Randomized Iterative Algorithms in Numerical Linear Algebra

Jackie Lok

A Dissertation
Presented to the Faculty
of Princeton University
in Candidacy for the Degree
of Doctor of Philosophy

Recommended for Acceptance
by the Department of
Operations Research and Financial Engineering

Adviser: Elizaveta Rebrova

May 2026

Abstract

Randomized iterative algorithms are fundamental computational tools for efficiently processing large-scale or high-dimensional data. The idea of introducing randomization allows for the development of algorithms that are faster, more flexible, or more robust, which has had a significant impact on the scale of problems that can be reliably solved. In this thesis, we develop the theoretical foundations and algorithmic principles of randomized iterative algorithms through the lens of numerical linear algebra.

In the first part, we study randomized iterative solvers for systems of linear equations. We develop a subspace-constrained framework for the randomized Kaczmarz and randomized coordinate descent methods, which are lightweight representatives of the sketch-and-project family of solvers. This idea allows us to propose efficient solvers for linear systems with approximately low-rank structure, based on a connection between the convergence rate and the problem of low-rank matrix approximation, as well as a robust solver that can leverage external knowledge to help solve corrupted linear systems.

In the second part, we analyze the learning dynamics of linear models trained by gradient descent, which provides a simple and tractable setting for deriving insights into the behavior of machine learning models. We investigate the dynamics of mini-batch gradient descent with random reshuffling, which is analytically challenging due to the dependencies arising from the sampling process, as well as the regularization effects of early stopping.

Finally, in the third part, we study an iterative approach for eigenvalue problems based on repeated random sparsification that is motivated by applications where the solution vector itself may be too large to be stored. We provide an analysis of a randomly sparsified power method applied to stochastic matrices, establishing that it can exploit the approximate sparsity of the leading eigenvector to achieve beyond-Monte Carlo convergence rates and dimension-independent computational costs.

Acknowledgements

First of all, I would like to express my gratitude to my adviser Liza Rebrova, who has supported me throughout my journey at Princeton. She has always been very generous with her time and effort, keeping an open door for whatever questions or help that I need. She has also been very encouraging, providing me with the freedom to explore various research topics and promoting numerous opportunities to collaborate, present, and travel.

I would like to thank Jianqing Fan and Boris Hanin for serving on my generals and FPO committees. I would also like to thank Jason Klusowski for serving as a dissertation reader. I am grateful to all of my teachers at Princeton, from whom I have learned a lot.

I would like to thank all of the ORFE staff for keeping everything running smoothly, especially Kim Lupinacci for handling all the administrative aspects related to being a graduate student at ORFE. I am grateful to Ransford Pinto and the McGraw Center for the opportunity to serve as a graduate teaching fellow. I am honoured to have received support from the Quad Fellowship during my PhD.

I would like to thank all of my collaborators that I had the fortune of learning from and working with, including Liza Rebrova, Roxanne He, Rishi Sonthalia, Jamie Haddock, Robert Webber, and Jonathan Weare. I would like to especially thank Robert Webber for his support during the last year of my PhD. I would like to thank Yuji Nakatsukasa for his hospitality during my visit to Oxford. I am grateful to Nathan Ross for encouraging a collaboration with his student.

I would like to thank all of my fellow graduate students that have made Sherrerd Hall and Princeton a brighter place, including Ben Budway, Rajita Chandak, Stefan Clarke, Giulia Crippa, Carla Crucianelli, Thomas Foster, Nicolas Garcia, Jiawei Ge, Yucheng Guo, Felix Höfer, Louis Hoffenberg, Erica Lai, Aaradhya Pandey, Alex Raistrick, Vinit Ranjan, Dan Rigobon, Till Saenger, Rajiv Sambharya, Boris Shigida, Sofiia Shvaiko, David Snyder, Shambhavi Suryanarayanan, Sylvia Tao, Hansen Tjo, Will Underwood, Kevin Wong, Rae

Yu, Henry Zhao, Kevin Zhang, and Xiaonan Zhu. A special shoutout to everyone who has joined in on some entertaining Avalon quests.

Finally, I would like to thank my family back home in Sydney—my parents, brother, and grandpa—for their support throughout my studies. I would also like to thank my family in New Jersey—my brother Kelvin and sister-in-law Selena, as well as Uncle Wayne and Aunt Lisa—for all their care and hospitality. Most of all, I am thankful to Jess for her unconditional support, love, and simply being here.

To the memory of my grandma, 婆婆

Contents

Abstract 3

Acknowledgements 4

1 Introduction **13**

1.1 Outline of thesis 14

1.1.1 Randomized iterative solvers for linear systems 15

1.1.2 Dynamics of gradient descent for linear models 21

1.1.3 Randomized iterative algorithms for eigenvalue problems 24

1.2 Related publications 27

2 A subspace-constrained randomized Kaczmarz method for structure or external knowledge exploitation **28**

2.1 Introduction 28

2.1.1 Setup and notation 30

2.1.2 Methods and main results 30

2.1.3 Organization 37

2.2 Related works 37

2.3 Analysis of subspace-constrained randomized Kaczmarz (SC-RK) 40

2.3.1 Simplified SC-RK update formula and proof of Theorem 2.1.1 40

2.3.2 SC-RK convergence on inconsistent linear systems 43

2.3.3 Exploiting structure with the SC-RK method 50

2.3.4 SC-RK on random data and dimension reduction 55
2.4 Analysis of the QuantileSC-RK algorithm 61
2.4.1 A deterministic condition for convergence 64
2.4.2 Proof of Theorem 2.4.1 67
2.5 Numerical experiments 71
2.5.1 SC-RK method for systems with correlated rows 71
2.5.2 SC-RK method for systems with low-rank structure 72
2.5.3 SC-RK method for noisy systems 74
2.5.4 QuantileSC-RK algorithm 74
2.5.5 Systems of differential equations with inconsistent initial conditions . 76
2.5.6 CT image reconstruction 78
2.6 Concluding remarks 79

3 Subspace-constrained randomized coordinate descent for linear systems with good low-rank matrix approximations 80

3.1 Introduction 80
3.1.1 Notation 84
3.1.2 Main results 84
3.1.3 Related works 94
3.1.4 Organization 97
3.2 A general framework for subspace-constrained sketch-and-project 97
3.2.1 The subspace-constrained sketch-and-project framework 99
3.2.2 Update rule and comparison with unconstrained sketch-and-project . 100
3.2.3 Convergence rate and block size 105
3.2.4 Example: randomized block Kaczmarz 108
3.3 Analysis of subspace-constrained randomized coordinate descent (SC-RCD) . 113
3.3.1 Convergence rate of SC-RCD 113
3.3.2 Implementation and complexity of SC-RCD 118

3.4 Numerical experiments . . . 122
3.4.1 Synthetic psd system . . . 122
3.4.2 KRR problem on real-life dataset with fast spectral decay . . . 124
3.4.3 KRR problem with slower spectral decay . . . 126
3.5 Concluding remarks . . . 127
3.6 Subspace-constrained sketch-and-project technical proofs . . . 128
3.7 Extension of SC-RCD for least-squares problems . . . 130
3.8 Additional numerical experiments . . . 133
3.9 Accelerated subspace-constrained sketch-and-project . . . 137
3.9.1 Proof of Theorem 3.9.3 . . . 141
3.9.2 Proof of Lemma 3.9.4 . . . 147
3.10 Accelerated subspace-constrained randomized coordinate descent . . . 149
3.10.1 Acceleration parameters with block size $\ell = 1$ . . . 150
3.10.2 Convergence rate of accelerated SC-RCD with $\ell = 1$ . . . 151

4 Dynamics of mini-batch gradient descent with random reshuffling 154
4.1 Introduction . . . 154
4.1.1 Related works . . . 157
4.2 Problem setup . . . 160
4.3 Analysis of mini-batch gradient descent with random reshuffling . . . 162
4.3.1 Training error dynamics . . . 164
4.3.2 Generalization error dynamics . . . 169
4.3.3 Asymptotic analysis . . . 171
4.4 Concluding remarks . . . 176
4.5 Additional background on full-batch gradient descent . . . 177
4.6 Technical proofs for mini-batch gradient descent . . . 180
4.6.1 Mini-batch gradient descent . . . 180
4.6.2 Two-batch gradient descent . . . 188

4.6.3 Asymptotic analysis . . . 192
4.7 Additional details on free probability computations . . . 195
4.7.1 Additional background . . . 195
4.7.2 Algorithm . . . 196
4.8 Additional numerical experiments . . . 199
4.8.1 Full-batch diverges, mini-batch converges . . . 200
4.8.2 Overparameterized regime . . . 200
4.8.3 Underparameterized regime . . . 201

5 Regularization via early stopping for linear least squares regression 203
5.1 Introduction . . . 203
5.1.1 Related work . . . 206
5.1.2 Problem setup and preliminaries . . . 208
5.2 Exact trajectories . . . 210
5.2.1 Formulas for the function φ . . . 216
5.3 Early stopping and generalized ridge regularization . . . 218
5.4 Should we stop early? . . . 223
5.4.1 Early stopped risk . . . 224
5.4.2 When is early stopping beneficial? . . . 228
5.5 Optimal stopping time estimate . . . 238
5.5.1 Experimental validation . . . 239
5.6 Concluding remarks . . . 241
5.7 Additional properties of the q-Pochhammer Symbol . . . 242

6 Analysis of a randomly sparsified power method 246
6.1 Introduction . . . 246
6.1.1 Deterministically sparsified power method . . . 248
6.1.2 Randomly sparsified power method . . . 250

6.1.3 Beyond-Monte Carlo rates . . . 253
6.1.4 Numerical demonstration with the Ising model . . . 253
6.1.5 Outline . . . 256
6.1.6 Notation . . . 257
6.2 Related works . . . 257
6.2.1 Deterministically sparsified power method . . . 257
6.2.2 Randomly sparsified power method . . . 258
6.3 Application to PageRank . . . 259
6.3.1 Computing the PageRank vector . . . 260
6.3.2 Numerical demonstration . . . 261
6.3.3 Comparison with randomly sparsified Richardson iteration . . . 263
6.4 Preliminaries . . . 263
6.4.1 Pivotal sparsification . . . 263
6.4.2 Stochastic matrices . . . 265
6.5 Proofs for deterministically sparsified power method . . . 269
6.5.1 Properties of deterministic sparsification . . . 269
6.5.2 Error bound with strict contractivity . . . 274
6.5.3 Failure mode of deterministic sparsification . . . 277
6.6 Proofs for randomly sparsified power method . . . 281
6.6.1 Full version of the main result . . . 282
6.6.2 Proof outline . . . 284
6.6.3 Analysis of bias . . . 284
6.6.4 Analysis of variance: general framework . . . 285
6.6.5 Variance bounds with Monte Carlo rates . . . 287
6.6.6 Error bounds at fixed times . . . 288
6.6.7 Variance bounds with improved rates . . . 297
6.7 Concluding remarks . . . 302

6.8 Additional discussion of the Ising model 303
6.8.1 Background 303
6.8.2 Additional numerics 304
6.8.3 Approximate sparsity of the Ising model 306
6.9 Proofs for properties of ℓ^1 contraction coefficients 309

Bibliography **311**

Chapter 1

Introduction

The development of efficient algorithms that are principled and supported by rigorous theoretical guarantees is increasingly important in a world where the volume and variety of data available continue to grow. The challenges posed by the massive datasets arising in machine learning and data science have led to a tremendous growth in the development and use of randomized iterative algorithms. These methods have the significant advantage of being lightweight and having a low memory footprint, enabling the handling of data that cannot be fully stored in memory.

Introducing randomization into numerical algorithms can lead to other key benefits. Randomized algorithms can be faster, both in theory and in practice. Furthermore, randomization allows for the design of flexible methods that are simple to implement and adaptable to different computational architectures, e.g., to facilitate parallelism or to reduce communication costs and memory movement. Finally, randomized algorithms can be more robust, since injecting randomness can protect against adversarial inputs and result in performance that typically mirrors average-case behavior.

The idea of using randomness as a computational resource is not new, and can be traced back to the introduction of the Monte Carlo method by S. Ulam and J. von Neumann for approximating intractable quantities by random sampling [MU49; FL50; Met+53]. It is an

extremely active area of research within the theoretical computer science community, where it is well-recognized that randomized algorithms can be faster or simpler than any known deterministic alternative [MR95]. Furthermore, stochastic optimization techniques such as stochastic gradient descent (SGD) and its variants are the workhorse of modern machine learning [BCN18].

The field of *randomized numerical linear algebra*, which studies the use of randomized algorithms for large-scale linear algebra computations—such as solving systems of linear equations, least squares regression, low-rank matrix approximation, and eigenvalue problems [GV13]—has developed rapidly over the past two decades. The contemporary subject originated in the theoretical computer science literature [FKV04; Pap+00; AM07; DKM06a] and has more recently attracted substantial contributions from the numerical analysis, statistics, and machine learning communities. By now, the field has reached a meaningful stage of maturity, as reflected by various surveys documenting its theoretical and algorithmic foundations [Mah11; Woo14; DM16; KV17; MT20; Mur+23; KT23; DM24; Epp25; PM25].

Despite this establishment, the foundational role of numerical linear algebra in machine learning, scientific computing, and data science continues to generate new applications and motivating problems. Consequently, significant opportunities remain for future research, including fundamental theoretical questions and the development of effective randomized algorithms that can be reliably applied in practice.

1.1 Outline of thesis

In this thesis, we develop the theoretical foundations and algorithmic principles of randomized iterative algorithms in numerical linear algebra and related fields. In the first part, we study randomized iterative algorithms for solving systems of linear equations (Chapters 2 and 3). The second part investigates the dynamics of (stochastic) gradient descent for linear

models (Chapters 4 and 5). Finally, the third part analyzes a randomly sparsified power method for eigenvalue problems (Chapter 6).

In the rest of this section, we will provide a brief outline of the problem studied in each part and the main contributions that are made in this thesis.

1.1.1 Randomized iterative solvers for linear systems

The problem of solving a system of linear equations $\mathbf{A}\mathbf{x} = \mathbf{b}$, where $\mathbf{A} \in \mathbb{R}^{m\times n}$ is a matrix and $\mathbf{b} \in \mathbb{R}^m$ is a vector, is a fundamental computational primitive. In some applications, such as image reconstruction, signal processing, statistical inference, or the numerical solution of differential equations, it is the primary computational task. In others, it arises as a subroutine embedded within a larger algorithm, such as for optimization, that must be solved repeatedly.

The particular properties of the linear system to be solved—such as its size, sparsity pattern, noise level, or any underlying algebraic, spectral or other exploitable problem-specific structure—are key factors underlying the most suitable methods that are capable of efficiently and reliably computing a solution to the desired accuracy. For large-scale problems, iterative algorithms are an essential tool since direct methods, such as those based on matrix factorizations [GV13], are often inefficient or simply infeasible if the matrix cannot even be stored in memory.

For solving large-scale, highly overdetermined systems (i.e., $m \gg n$), the *Kaczmarz method* [Kac37] is an effective and extremely lightweight row-action method that solves one equation at a time using a given row ordering (Figure 1.1). In each iteration of the Kaczmarz method, a row $\mathbf{a}_j \in \mathbb{R}^n$ of the matrix $\mathbf{A}$ is selected, and the current iterate $\mathbf{x}^k \in \mathbb{R}^n$ is projected onto the hyperplane $\{\mathbf{x} \in \mathbb{R}^n : \mathbf{a}_j^\mathsf{T}\mathbf{x} = b_j\}$ by

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \frac{b_j - \mathbf{a}_j^\mathsf{T}\mathbf{x}^k}{\|\mathbf{a}_j\|_2^2} \cdot \mathbf{a}_j,$$

where $\|\cdot\|_2$ denotes the ℓ^2 norm of a vector. The Kaczmarz method is a classical algorithm in image reconstruction [Nat01; KS01; Her09], where it was rediscovered under the name of algebraic reconstruction technique (ART) and implemented in the first computed tomography scanner [Hou73; GBH70; SL74; AK84].

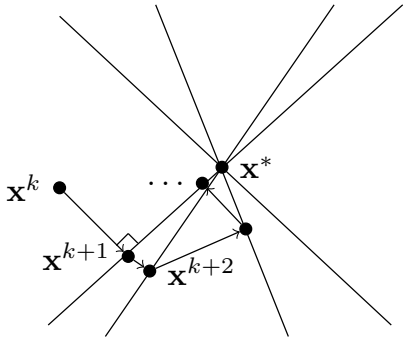


Figure 1.1: The solution $\mathbf{x}^*$ of a consistent linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$ lies at the intersection of all of the hyperplanes corresponding to the solution space of each equation $\mathbf{a}_j^\mathsf{T}\mathbf{x} = b_j$ for $j = 1, \ldots, m$. In the k^{th} iteration of the Kaczmarz method, a row $\mathbf{a}_j$ is selected, and the current iterate $\mathbf{x}^k$ is projected onto the associated hyperplane to obtain $\mathbf{x}^{k+1}$.

In their seminal paper, Strohmer and Vershynin [SV09] proposed the *randomized Kaczmarz* (RK) algorithm, which samples each row independently with probability proportional to its squared norm $\|\mathbf{a}_j\|_2^2$ in each iteration. They proved that if the linear system is consistent and has full rank, then the RK method converges linearly in mean squared error to the unique solution $\mathbf{x}^*$ with a rate that depends on the geometric properties of $\mathbf{A}$:

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|_2^2 \leq \left(1 - \frac{\sigma_n(\mathbf{A})^2}{\|\mathbf{A}\|_F^2}\right)^k \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_2^2,$$

where $\sigma_1(\mathbf{A}) \geq \ldots \geq \sigma_n(\mathbf{A}) \geq 0$ are the singular values of $\mathbf{A}$ and $\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^n \sigma_i(\mathbf{A})^2}$ is the Frobenius norm of $\mathbf{A}$. More precisely, the rate depends on the *scaled condition number* $\|\mathbf{A}\|_F/\sigma_{\min}(\mathbf{A})$ of $\mathbf{A}$. If $\mathbf{A}$ is well-conditioned, then this guarantees that the RK method converges very efficiently. For example, if the traditional condition number $\sigma_1(\mathbf{A})/\sigma_n(\mathbf{A})$ is upper bounded by some absolute constant C, then $O(n\log(1/\varepsilon))$ iterations and $O(n^2\log(1/\varepsilon))$

arithmetic operations suffice to obtain an ε-relative error approximate solution $\widehat{\mathbf{x}}$ satisfying $\mathbb{E}\|\widehat{\mathbf{x}} - \mathbf{x}^*\|_2^2 \leq \varepsilon \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_2^2$.

However, the convergence rate of randomized Kaczmarz rapidly deteriorates with the presence of any spectral outliers—that is, a few leading singular values that are much larger than the rest. This can easily arise if the rows of $\mathbf{A}$ are correlated with each other (i.e., $\mathbf{a}_i^\mathsf{T}\mathbf{a}_j > 0$ for many pairs i, j), or more generally if the matrix $\mathbf{A}$ exhibits approximate low-rank (spectral) structure, which is ubiquitous in practice.

In Chapter 2, we propose and analyze a *subspace-constrained randomized Kaczmarz* (SC-RK) method, where the dynamics of the Kaczmarz algorithm are confined within the solution space of the linear subsystem $\mathbf{A}_{\mathcal{I}_0,:}\mathbf{x} = \mathbf{b}_{\mathcal{I}_0}$ corresponding to a selected set of rows $\mathcal{I}_0 \subseteq \{1, \ldots, m\}$ (see Figure 1.2). Given any initial iterate $\mathbf{x}^0$ in the chosen affine subspace, we derive the following update formula:

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \frac{b_j - \mathbf{a}_j^\mathsf{T}\mathbf{x}^k}{\|\mathbf{P}\mathbf{a}_j\|_2^2} \cdot \mathbf{P}\mathbf{a}_j,$$

where $\mathbf{P}$ denotes the orthogonal projector onto $\mathrm{null}(\mathbf{A}_{\mathcal{I}_0,:})$. This resembles the Kaczmarz algorithm, and retains the advantage of having a low computational and storage cost as long as the size of the subspace constraint $|\mathcal{I}_0|$ is not too large.

We prove that the SC-RK method leads to an accelerated convergence rate, especially for linear systems where the matrix $\mathbf{A}$ has *approximate low-rank structure*, in the sense that $\sum_{i=r+1}^n \sigma_i(\mathbf{A})^2 \ll \sum_{i=1}^n \sigma_i(\mathbf{A})^2 = \|\mathbf{A}\|_F^2$ for some $r \ll n$. By connecting the improvement in the convergence rate to the *row subset selection problem* in numerical linear algebra, we demonstrate that a good subspace $\mathcal{I}_0$ can be found by randomized sampling. Our analysis also covers the case where the linear system is inconsistent, showing that the iterates converge to the least squares solution up to an error horizon.

Furthermore, we use the subspace constraint framework as a building block for a *robust* randomized iterative solver for *corrupted linear systems*. In this setting, the goal is to re-

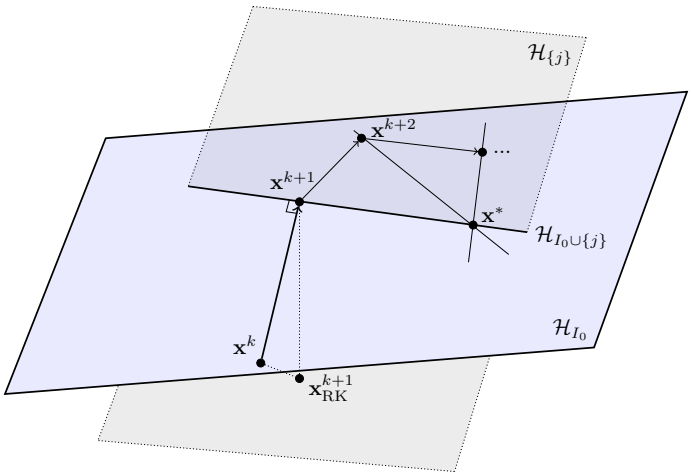


Figure 1.2: In each iteration of the subspace-constrained randomized Kaczmarz algorithm, a row $\mathbf{a}_j$ is selected, and the current iterate is projected onto the solution space $\mathbf{a}_j^\mathsf{T}\mathbf{x} = b_j$, constrained to remain within a fixed affine subspace $\{\mathbf{x} \in \mathbb{R}^n : \mathbf{A}_{\mathcal{I}_0,:}\mathbf{x} = \mathbf{b}_{\mathcal{I}_0}\}$. The rows corresponding to $\mathcal{I}_0$ can be learned algorithmically or specified using external knowledge.

construct the solution $\mathbf{x}^*$ of an underlying linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$ given a set of corrupted measurements $\widetilde{\mathbf{b}} = \mathbf{b} + \mathbf{b}_\mathcal{C}$, where $\mathbf{b}_\mathcal{C}$ is a sparse vector supported on some corrupted indices $\mathcal{C} \subseteq \{1, \dots, m\}$. This models applications where some measurements are corrupted by arbitrarily large errors, which may occur during the data collection, transmission, or storage process due to faulty components or adversarial attacks. Moreover, we suppose that we possess *external knowledge* in the form of a set $\mathcal{I}_0 \subseteq \{1, \dots, m\}$ of corruption-free measurements that we would like to exploit to improve the stability of the recovery algorithm.

By taking inspiration from a quantile-based modification of the randomized Kaczmarz algorithm by Haddock et al. [Had+22], we develop the *QuantileSC-RK algorithm*, which constrains the dynamics within the solution space of the trustworthy subsystem $\mathbf{A}_{\mathcal{I}_0,:}\mathbf{x} = \mathbf{b}_{\mathcal{I}_0}$. We demonstrate that QuantileSC-RK is able to efficiently utilize external knowledge about corruption-free equations to achieve convergence in difficult settings, such as when there are many corruptions (e.g., scaling linearly with the total number of measurements) or when there is not much redundancy in the measurements (i.e., the measurement matrix $\mathbf{A}$ is near-square).

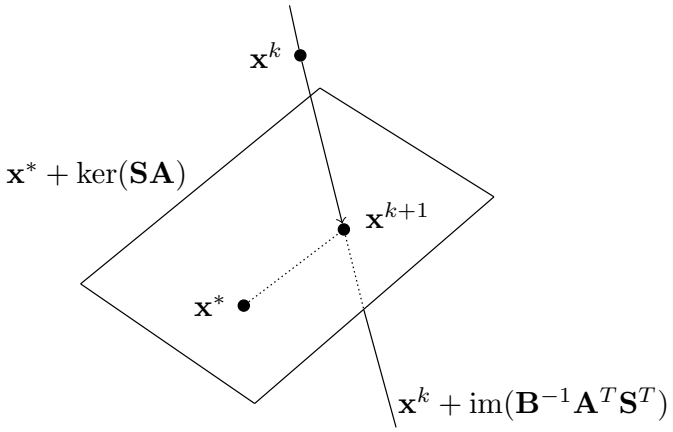


Figure 1.3: In each iteration of the sketch-and-project algorithm, a sketching matrix $\mathbf{S} \equiv \mathbf{S}^k \in \mathbb{R}^{\ell \times m}$ is drawn from a given distribution $\mathcal{D}$, and the current iterate is projected onto the solution space of the sketched linear system $\mathbf{SAx} = \mathbf{Sb}$ with respect to the $\|\cdot\|_{\mathbf{B}}$ norm.

In Chapter 3, we generalize the subspace-constrained framework to a wider family of randomized iterative solvers. The randomized Kaczmarz algorithm is a special case of the *sketch-and-project method* for solving $\mathbf{Ax} = \mathbf{b}$ introduced by Gower and Richtárik [GR15a]. Given a positive definite matrix parameter $\mathbf{B} \succ \mathbf{0}$ and a distribution $\mathcal{D}$ over matrices in $\mathbb{R}^{\ell \times m}$ with $\ell \leq \min\{m, n\}$, each iteration of sketch-and-project draws a sketching matrix $\mathbf{S}^k \sim \mathcal{D}$, forms a wide, underdetermined "sketch" $\mathbf{S}^k\mathbf{Ax} = \mathbf{S}^k\mathbf{b}$ of the linear system, and projects the current iterate $\mathbf{x}^k$ onto the corresponding solution space with respect to the norm $\|\mathbf{z}\|_{\mathbf{B}} = \sqrt{\mathbf{z}^\mathsf{T}\mathbf{Bz}}$ (see Figure 1.3):

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x} \in \mathbb{R}^n} \|\mathbf{x} - \mathbf{x}^k\|_{\mathbf{B}} \quad \text{such that} \quad \mathbf{S}^k\mathbf{Ax} = \mathbf{S}^k\mathbf{b}.$$

For example, the randomized Kaczmarz algorithm follows from choosing the identity matrix for the geometry parameter, $\mathbf{B} = \mathbf{I}$, and sketching matrices of the form $\mathbf{S}^k = \mathbf{e}_j^\mathsf{T} \in \mathbb{R}^{1 \times m}$, where $\mathbf{e}_j$ is the standard basis vector in $\mathbb{R}^m$ corresponding to the coordinate that is sampled.

Specifically, we develop the *subspace-constrained sketch-and-project method*, which constrains the dynamics of sketch-and-project within the affine subspace $\{\mathbf{x} \in \mathbb{R}^n : \mathbf{QAx} = \mathbf{Qb}\}$ corresponding to an additional matrix parameter $\mathbf{Q} \in \mathbb{R}^{d \times m}$ with $d < m$. We prove that the

updates are essentially the same as for the usual sketch-and-project with $\mathbf{AP_B}$ in place of $\mathbf{A}$, where $\mathbf{P_B}$ is the oblique projector onto null($\mathbf{QA}$) with respect to the $\|\cdot\|_{\mathbf{B}}$ norm.

As a concrete application, we analyze a subspace-constrained (block) generalization of the *randomized coordinate descent* (RCD) algorithm studied by Leventhal and Lewis [LL10] for solving linear systems $\mathbf{Ax} = \mathbf{b}$ where $\mathbf{A} \in \mathbb{R}^{n\times n}$ is a positive semidefinite (psd) matrix. In our analysis, the rank-d Nyström matrix approximation $\mathbf{A}\langle\mathcal{S}\rangle \in \mathbb{R}^{n\times n}$ corresponding to a subset of d indices $\mathcal{S} \subseteq \{1, 2, \ldots, n\}$ selected for the subspace constraint emerges as a key algorithmic component. By crystallizing the connection between the convergence rate and the quality of the low-rank matrix approximation, we show that an *efficient* linear solver can be developed if we can efficiently compute a good low-rank matrix approximation. For the task of approximating the psd matrix $\mathbf{A}$, we suggest using the *RPCholesky algorithm*, recently proposed and analyzed by Chen et al. [Che+25], which is an effective algorithm based on adaptive diagonal sampling that outputs a near-optimal Nyström approximation.

We prove that the resulting *subspace-constrained randomized coordinate descent* (SC-RCD) method can be used to compute an approximation $\widehat{\mathbf{x}}$ of the solution $\mathbf{x}^*$ satisfying $\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{A}}^2 \leq \varepsilon \cdot \mathbb{E}\|\mathbf{x}^0 - \mathbf{x}^*\|_{\mathbf{A}}^2$ using $O\big((n^2 + nd^2) \cdot \bar{\kappa}_r(\mathbf{A}) \log(1/\varepsilon)\big)$ arithmetic operations. Here, $\bar{\kappa}_r(\mathbf{A}) = (n-r)^{-1} \sum_{i>r} \lambda_i(\mathbf{A})/\lambda_{\min}^+(\mathbf{A})$ is the *normalized tail condition number* of $\mathbf{A}$ for some $r \approx d$, where we denote the eigenvalues of $\mathbf{A}$ by $\lambda_1(\mathbf{A}) \geq \ldots \geq \lambda_n(\mathbf{A})$ and the smallest non-zero eigenvalue by $\lambda_{\min}^+(\mathbf{A})$. In particular, under a mild assumption that $\mathbf{A}$ is not extremely ill-conditioned, this result implies that if $\bar{\kappa}_r(\mathbf{A}) = O(1)$ for some $r = O(\sqrt{n}/\log n)$, then the SC-RCD method can compute an ε-approximate solution using $O(n^2 \log(1/\varepsilon))$ arithmetic operations, which is optimal in terms of n for dense $n \times n$ systems.

Thus, the SC-RCD method is an efficient solver for psd linear systems with approximate low-rank structure, which is ubiquitous in many applications in machine learning and scientific computing. For example, high-dimensional real-life datasets are often intrinsically low-rank, and kernel matrices commonly exhibit rapid spectral decay. We provide numerical

experiments that demonstrate the efficiency of the SC-RCD method for large-scale kernel ridge regression problems.

1.1.2 Dynamics of gradient descent for linear models

Modern machine learning models are mainly trained using gradient-based methods on very large datasets. Since it is typically impractical to compute the entire gradient using all of the available data, *stochastic gradient descent* (SGD) and its variants are routinely used in practice. Studying the dynamics of (stochastic) gradient descent is an important problem for understanding the training dynamics and generalization capabilities of the model that is learned, guiding important hyperparameter choices such as the learning rates used, the amount of training to perform, or the mini-batch size.

We can develop theoretical insights into this problem by studying the dynamics of gradient descent for linear models, which provides a simple and tractable setting where many fascinating empirical phenomena observed in more complex models can be reproduced. Suppose that we are given n data points $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$, each drawn independently from an underlying distribution $\mathcal{D}$, where $\mathbf{x}_i \in \mathbb{R}^p$ is the feature vector and the $y_i \in \mathbb{R}$ is the associated response variable. We assume that there is an underlying vector of parameters $\boldsymbol{\beta}_* \in \mathbb{R}^p$ such that $y_i = \mathbf{x}_i^\mathsf{T} \boldsymbol{\beta}_* + \varepsilon_i$ for some random noise ε_i. The learning problem is to compute an estimate $\widehat{\boldsymbol{\beta}} \in \mathbb{R}^p$ using the training dataset that *generalizes well:* i.e., given an unseen feature vector $\mathbf{x}$, $\mathbf{x}^\mathsf{T} \widehat{\boldsymbol{\beta}}$ is a good predictor of its associated response y.

We can estimate the parameters using least squares regression. Let $\mathbf{X} \in \mathbb{R}^{n \times p}$ denote the matrix obtained from stacking the n feature vectors in the training dataset as rows, and $\mathbf{y} \in \mathbb{R}^n$ denote the vector of the corresponding response variables. Note that in a linear model, we can write $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}_* + \boldsymbol{\varepsilon}$, where $\boldsymbol{\varepsilon} \in \mathbb{R}^n$ denotes the vector of the noise terms. Then, the goal is to solve

$$\operatorname*{arg\,min}_{\boldsymbol{\beta} \in \mathbb{R}^p} L(\boldsymbol{\beta}) \quad \text{where} \quad L(\boldsymbol{\beta}) = \frac{1}{2n} \|\mathbf{X}\boldsymbol{\beta} - \mathbf{y}\|_2^2 = \frac{1}{2n} \sum_{i=1}^n (\mathbf{x}_i^\mathsf{T} \boldsymbol{\beta} - y_i)^2.$$

We can use gradient descent to solve this optimization problem by iterating

$$\boldsymbol{\beta}_k = \boldsymbol{\beta}_{k-1} - \eta_k \nabla L(\boldsymbol{\beta}_{k-1}) = \boldsymbol{\beta}_{k-1} - \frac{\eta_k}{n} \mathbf{X}^\mathsf{T}(\mathbf{X}\boldsymbol{\beta}_{k-1} - \mathbf{y}),$$

where $\{\eta_k\}_{k \geq 1}$ is a sequence of *step sizes* or *learning rates*. In stochastic gradient descent, the gradient $\nabla L(\boldsymbol{\beta}_{k-1})$ is replaced with an estimate $\widehat{\nabla} L(\boldsymbol{\beta}_{k-1})$. For example, since the loss function is a sum over the individual loss for each training data point, the simplest estimate can be obtained by sampling one of the n data points $(\mathbf{x}_j, y_j)$ and using the gradient of its associated loss: $\widehat{\nabla} L(\boldsymbol{\beta}_{k-1}) = (\mathbf{x}_j^\mathsf{T} \boldsymbol{\beta}_{k-1} - y_j) \cdot \mathbf{x}_j$. In this case, the iterations of SGD read

$$\boldsymbol{\beta}_k = \boldsymbol{\beta}_{k-1} - \eta_k (\mathbf{x}_j^\mathsf{T} \boldsymbol{\beta}_{k-1} - y_j) \cdot \mathbf{x}_j.$$

Note that the randomized Kaczmarz algorithm discussed in the previous section (with the correspondence $(\mathbf{A}, \mathbf{b}, \mathbf{x}) \leftrightarrow (\mathbf{X}, \mathbf{y}, \boldsymbol{\beta})$ in notation) can be cast as an instance of SGD where the data point $(\mathbf{x}_j, y_j)$ is sampled with probability proportional to $\|\mathbf{x}_j\|_2^2$, and the specific choice of step size $\eta_k = 1/\|\mathbf{x}_j\|_2^2$ is used. Indeed, this correspondence, which was observed by [NSW16], suggests that ideas and techniques can be transferred between separate bodies of literature.

It is an interesting problem to study the dynamics of stochastic gradient descent even though the underlying least squares optimization problem is fairly simple. An important reason is that machine models are often *highly overparameterized* (i.e., $p \gg n$), meaning that the solution is not unique. In this case, the choice of algorithm—including all of its hyperparameters—provides an important *implicit bias* towards the solution that is computed: e.g., gradient descent, initialized at zero, converges to the min-norm solution of the least squares problem. Moreover, since iterative algorithms are not run to convergence in practice, discerning the properties of the model $\widehat{\boldsymbol{\beta}} = \boldsymbol{\beta}_T$ that is returned at some stopping time T requires a more delicate understanding of the dynamics.

In Chapter 4, we study a variant of SGD known as *mini-batch gradient descent*, where a subset of the training data is used in each iteration. We assume that *random reshuffling* is used, which means the dataset is partitioned into B mini-batches $\{(\mathbf{X}_b, \mathbf{y}_b)\}_{b=1}^{B}$, randomly permuted, and iterated through. This form of SGD is commonly used in practice. However, the introduction of dependencies between mini-batches due to sampling without replacement significantly complicates the theoretical analysis of the dynamics, compared to schemes where mini-batches are sampled independently in each iteration. Our goal is to contribute towards a better understanding of mini-batch gradient descent with random reshuffling by studying the dynamics of the mean iterate for least squares regression.

We show that the training and generalization errors depend on a sample cross-covariance matrix $\mathbf{Z} = n^{-1}\widetilde{\mathbf{X}}^{\mathsf{T}}\mathbf{X}$ between the original features $\mathbf{X}$ and a set of new features $\widetilde{\mathbf{X}}$ in which each feature is modified by the mini-batches that appear before it during the learning process in an averaged way. Using this representation, we establish that the dynamics of mini-batch and full-batch gradient descent agree up to leading order with respect to the step size using the linear scaling rule. However, mini-batch gradient descent with random reshuffling exhibits a subtle dependence on the step size that a gradient flow analysis cannot detect—for example, it may converge to a limit that depends on the step size. By comparing $\mathbf{Z}$, a non-commutative polynomial of random matrices, with the sample covariance matrix of $\mathbf{X}$ asymptotically, we demonstrate that batching affects the dynamics by resulting in a form of shrinkage on the eigenvalue spectrum of the features matrix.

In Chapter 5, we study the effects of early stopping for (full-batch) gradient descent. Since machine learning models are usually trained on noisy data, there is generally a trade-off between fitting the signal (i.e., bias) and the noise (i.e., variance) as the number of iterations increases. This raises fundamental questions related to the properties of early stopped models and how to decide when to stop training. Heuristically, it is known that early stopping induces a form of ℓ^2 regularization, also known as *ridge* or *Tikhonov regularization*. However, formalizing this intuition under minimal assumptions has proven to be challenging.

By exactly characterizing the trajectory of the parameters and the expected generalization error with arbitrary learning rates and data, we show that the early stopped solution $\boldsymbol{\beta}_T$ is equivalent to the minimum norm solution of a related generalized ridge regression problem. We also establish that early stopping is beneficial for many common learning rate schedules, but that it may not be advantageous for some others. We provide an estimate for the optimal stopping time to minimize the generalization error and empirically demonstrate the accuracy of our estimate.

1.1.3 Randomized iterative algorithms for eigenvalue problems

Solving eigenvalue problems $\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$ with $\mathbf{A} \in \mathbb{C}^{n\times n}$ and $\mathbf{v} \in \mathbb{C}^n$ is another fundamental computation in numerical linear algebra. For large-scale problems, iterative methods that can exploit sparsity in the matrix $\mathbf{A}$ are crucial. The power method is perhaps the simplest iterative algorithm for computing the leading eigenvector $\mathbf{v}_1$ associated with the largest-magnitude eigenvalue $\lambda_1(\mathbf{A})$ of a matrix $\mathbf{A}$, which has iterations

$$\mathbf{x}_t = \frac{\mathbf{A}\mathbf{x}_{t-1}}{\|\mathbf{A}\mathbf{x}_{t-1}\|},$$

where $\|\cdot\|$ is some norm chosen for normalization in each iteration. It is well-known that if $\mathbf{A}$ is normal and has a large spectral gap, i.e., $|\lambda_2(\mathbf{A})|/|\lambda_1(\mathbf{A})| \ll 1$, where $\lambda_2(\mathbf{A})$ is the second largest-magnitude eigenvalue of $\mathbf{A}$, then the power iterations converge quickly.

However, the extremely large scale of the matrices increasingly encountered in modern applications means that classical iterative approaches can be too expensive—for instance, the solution vector $\mathbf{x}$ itself may be too large to be stored. Motivated by the need to mitigate the computational and storage costs for such problems, randomized iterative algorithms for "extreme-scale linear algebra" are receiving renewed interest to extend the boundary of problems that can be solved.

Motivated by the success of diffusion Monte Carlo algorithms that have been applied to eigenvalue problems as large as $10^{108} \times 10^{108}$ [She+12], a general approach called *fast randomized iteration*, which is based on randomly imposing sparsity in between the updates of an iterative method, was proposed by Lim and Weare [LW17]. The *randomly sparsified power method* is a particular instantiation of this framework, which has iterations

$$\mathbf{y}_t = \frac{\mathbf{A}\mathbf{x}_{t-1}}{\|\mathbf{A}\mathbf{x}_{t-1}\|}, \quad \mathbf{x}_t = \boldsymbol{\varphi}_t(\mathbf{y}_t),$$

where $\boldsymbol{\varphi}_t$ is an independent realization of an unbiased *random sparsification operator* $\boldsymbol{\varphi} : \mathbb{C}^n \to \mathbb{C}^n$ that maintains a user-chosen sparsity level; i.e., $\mathbb{E}\boldsymbol{\varphi}(\mathbf{x}) = \mathbf{x}$ and $\|\boldsymbol{\varphi}(\mathbf{x})\|_0 \leq m$ for all $\mathbf{x} \in \mathbb{C}^n$ (see Figure 1.4). Here, $\|\mathbf{x}\|_0$ denotes the number of non-zero entries of a vector $\mathbf{x}$. More intricate versions of the randomly sparsified power method have been developed and applied to large-scale eigenvalue problems in quantum chemistry [Gre+19; Gre+20; Gre+22a; Gre+22b].

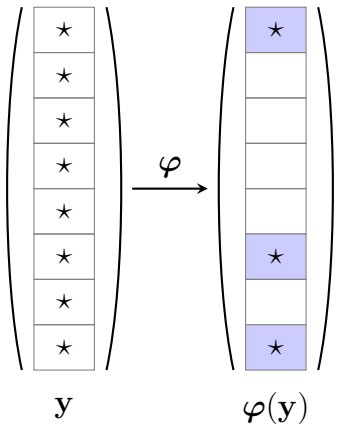


Figure 1.4: Representation of the sparsification operator $\boldsymbol{\varphi}$ with sparsification parameter m, which is used to control computational and storage costs in an iterative scheme.

Note that if each column of $\mathbf{A}$ has at most q non-zero entries, then each power iteration typically costs $O(nq)$ operations since the iterate $\mathbf{x}_t$ will rapidly become dense due to fill-in. However, each randomly sparsified power iteration costs $O(mq)$ operations, which can be significantly lower if $m \ll n$. The hope is that a relatively small sparsification parameter m

can be chosen—ideally independent of or growing sublinearly with the input dimension n—to obtain a stable randomized iteration. Heuristically, this might be possible if the leading eigenvector $\mathbf{v}_1$ is approximately sparse, in the sense that the magnitudes of its entries decay rapidly. However, a complete mathematical analysis of the algorithm, which would rigorously justify its applicability for certain classes of problems, remains elusive.

In Chapter 6, we provide an analysis of the randomly sparsified power method applied to computing the leading eigenvector of a column-stochastic matrix $\mathbf{A} \in [0,1]^{n\times n}$. This is equivalent to computing the stationary distribution $\mathbf{v} \in [0,1]^n$ of a Markov chain with probability transition matrix $\mathbf{A}$ that solves $\mathbf{A}\mathbf{v} = \mathbf{v}$. In this setting, each iteration of the randomly sparsified power method reads

$$\mathbf{y}_t = \mathbf{A}\mathbf{x}_{t-1}, \quad \mathbf{x}_t = \boldsymbol{\varphi}_t(\mathbf{y}_t),$$

where ℓ^1 normalization is chosen so that the power iteration step is linear, and a particular sparsification scheme based on pivotal sampling is applied in each iteration to sample m entries without replacement. In the context of column-stochastic matrices, this procedure is a generalization of the Markov chain Monte Carlo (MCMC) sampler.

We establish error bounds in the general case where $\mathbf{A}$ has a spectral gap, showing that the randomly sparsified power method can achieve beyond-Monte Carlo convergence rates—i.e., $m^{-1/2}$ as a function of the sparsification parameter m—by exploiting the approximate sparsity of the leading eigenvector $\mathbf{v}$. Specifically, we prove that as long as m is sufficiently large, based on the mixing time of the Markov chain associated with $\mathbf{A}$, then the error of the randomly sparsified power method scales like $m^{-1/2}$ multiplied by the tail sum $\sum_{i=m}^{n} \mathbf{v}^{\downarrow}(i)$, where $\mathbf{v}^{\downarrow}$ denotes any weakly decreasing rearrangement of $\mathbf{v}$ with $\mathbf{v}^{\downarrow}(1) \geq \ldots \geq \mathbf{v}^{\downarrow}(n)$. Hence, the randomly sparsified power method can be used to produce a high-accuracy solution of sparse eigenvalue problems with dimension-independent computational costs if the entries of $\mathbf{v}$ rapidly decay. We also showed that a sparsified power iteration based on deter-

ministic sparsification can only provide guaranteed accuracy control when $\mathbf{A}$ is a strict ℓ^1 contraction and can fail for a class of hard problems, which identifies a theoretical separation between randomized algorithms and their deterministic counterparts in this setting.

1.2 Related publications

- Chapter 2 is based on the following joint work with Elizaveta Rebrova:

  J. Lok and E. Rebrova. "A subspace constrained randomized Kaczmarz method for structure or external knowledge exploitation". *Linear Algebra and its Applications* **698**, 2024, pp. 220–260. arXiv: 2309.04889 [math.NA]. DOI: 10.1016/j.laa.2024.06.010

- Chapter 3 is based on the following joint work with Elizaveta Rebrova:

  J. Lok and E. Rebrova. "Subspace-constrained randomized coordinate descent for linear systems with good low-rank matrix approximations". *SIAM Journal on Matrix Analysis and Applications*, 2026. To appear. arXiv: 2506.09394 [math.NA]

- Chapter 4 is based on the following joint work with Rishi Sonthalia and Elizaveta Rebrova:

  J. Lok, R. Sonthalia, and E. Rebrova. "Error dynamics of mini-batch gradient descent with random reshuffling for least squares regression". *Proceedings of the 36th International Conference on Algorithmic Learning Theory*. 2025. arXiv: 2406.03696 [stat.ML]

- Chapter 5 is based on the following joint work with Rishi Sonthalia and Elizaveta Rebrova:

  R. Sonthalia, J. Lok, and E. Rebrova. "On Regularization via Early Stopping for Least Squares Regression", 2024. Preprint, arXiv:2406.04425. arXiv: 2406.04425 [cs.LG]

- Chapter 6 is based on a joint work with Robert J. Webber and Jonathan Weare, currently in preparation.

Chapter 2

A subspace-constrained randomized Kaczmarz method for structure or external knowledge exploitation

This chapter is based on the following joint work with Elizaveta Rebrova:

J. Lok and E. Rebrova. "A subspace constrained randomized Kaczmarz method for structure or external knowledge exploitation". *Linear Algebra and its Applications* **698**, 2024, pp. 220–260. arXiv: 2309.04889 [math.NA]. DOI: 10.1016/j.laa.2024.06.010

2.1 Introduction

A ubiquitous problem across the sciences is solving large-scale systems of linear equations $\mathbf{Ax} = \mathbf{b}$, for which scalable and efficient iterative methods are useful when it is too slow or infeasible to solve the system directly. Instead of solving such problems obliviously, it is natural to have insights into the structural properties of the linear system of interest, such as being approximately low-rank. Moreover, external knowledge about trustworthy observations in the presence of corrupted measurements could be available.

Now, additional information about the structure of a linear system influences the choice of the most suitable solver. In this work, we take an adaptive, problem-aware approach to account for various types of auxiliary information by augmenting the iterations of a generic iterative linear solver based on a distinguished subsystem of equations.

The generic solver that we consider is the *Kaczmarz algorithm* [Kac37], which is an iterative, row-action method for solving large-scale, typically overdetermined systems of linear equations. It is a special case of the alternating projection method that has low computational cost and storage per iteration, and can be used in the streaming setting where a single row (or block of rows) of the system can be accessed at a time. Besides its traditional applications in areas such as image reconstruction [Nat01; Her09] and signal processing [Cen+92], the Kaczmarz algorithm has recently been used as a building block for more sophisticated methods to design linear solvers [DY24; Du+21], and to address problems such as phase retrieval [TV19] and tensor recovery [CQ21].

In each iteration of the Kaczmarz algorithm, a row $\mathbf{a}_j$ of the matrix $\mathbf{A}$ is selected, and the current iterate $\mathbf{x}^k$ is projected onto the hyperplane $\mathbf{a}_j^\mathsf{T}\mathbf{x} = b_j$ by

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \frac{b_j - \mathbf{a}_j^\mathsf{T}\mathbf{x}^k}{\|\mathbf{a}_j\|} \cdot \frac{\mathbf{a}_j}{\|\mathbf{a}_j\|}. \tag{2.1}$$

In their seminal paper, Strohmer and Vershynin [SV09] show that if the system of linear equations is consistent and has unique solution $\mathbf{x}^*$, then the *randomized Kaczmarz (RK) algorithm*, which samples each row independently with probability $\|\mathbf{a}_j\|^2/\|\mathbf{A}\|_F^2$ at each iteration, converges to $\mathbf{x}^*$ in expectation with an exponential rate (i.e., linearly) that depends on the geometric properties of $\mathbf{A}$ (or more precisely, its scaled condition number $\|\mathbf{A}\|_F/\sigma_{\min}(\mathbf{A})$):

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|^2 \leq \left(1 - \frac{\sigma_{\min}(\mathbf{A})^2}{\|\mathbf{A}\|_F^2}\right)^k \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|. \tag{2.2}$$

Subsequently, many variants of the randomized Kaczmarz method have been analyzed; we defer a detailed discussion of related works to Section 2.2 after presenting our results.

In this chapter, we propose two Kaczmarz-based algorithms that can exploit (a) *approximately low-rank structure and geometric properties of the matrix in a system of linear equations to accelerate convergence*; and (b) *external knowledge about corruption-free equations for linear systems with arbitrary sparse corruptions to enable convergence even in the highly corrupted regime.*

2.1.1 Setup and notation

We consider a consistent, overdetermined (i.e., tall) system of linear equations $\mathbf{A}\mathbf{x} = \mathbf{b}$, or *linear system* for short, where the rows of $\mathbf{A} \in \mathbb{R}^{m\times n}$ are denoted by $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_m \in \mathbb{R}^n$, $\mathbf{b} \in \mathbb{R}^m$, and $m \geq n$. We assume throughout that $\mathbf{A}$ has full rank, and denote the unique solution of the linear system by $\mathbf{x}^* \in \mathbb{R}^n$. We work in the real setting for simplicity, but everything can be generalized to the complex setting.

Vectors, oriented as columns by default, and matrices are written in boldface. The vector ℓ_2-norm is denoted by $\|\cdot\|$, and the matrix spectral and Frobenius norms are denoted by $\|\cdot\|$ and $\|\cdot\|_F$. The singular values of a matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ are denoted by $\sigma_{\max}(\mathbf{A}) = \sigma_1(\mathbf{A}) \geq \sigma_2(\mathbf{A}) \geq \cdots \geq \sigma_{\min\{m,n\}}(\mathbf{A}) = \sigma_{\min}(\mathbf{A})$, and the smallest non-zero singular value is denoted by $\sigma^+_{\min}(\mathbf{A})$. The Moore-Penrose pseudoinverse of $\mathbf{A}$ is denoted by $\mathbf{A}^\dagger$. We refer to the row submatrix of $\mathbf{A}$ (resp. subvector of $\mathbf{b}$) indexed by $I \subseteq [m] \coloneqq \{1, 2, \ldots, m\}$ by $\mathbf{A}_I$ (resp. $\mathbf{b}_I$). The solution space $\mathbf{A}_I\mathbf{x} = \mathbf{b}_I$ of a linear system refers to the affine subspace $\{\mathbf{x} \in \mathbb{R}^n : \mathbf{A}_I\mathbf{x} = \mathbf{b}_I\}$.

2.1.2 Methods and main results

The SC-RK method

Fix a subset $I_0 \subset [m]$ of indices of rows of $\mathbf{A}$ with $m_0 \coloneqq |I_0| < n$, and denote the remaining indices by $I_1 \coloneqq [m] \setminus I_0$. We define a variant of the RK algorithm that confines the iterates within the solution space $\mathbf{A}_{I_0}\mathbf{x} = \mathbf{b}_{I_0}$, which we will refer to as the *subspace-constrained*

randomized Kaczmarz (SC-RK) method. Each update of the SC-RK algorithm consists of a projection of the current iterate $\mathbf{x}^k$ onto the solution space $\mathbf{A}_{I_0\cup\{j\}}\mathbf{x} = \mathbf{b}_{I_0\cup\{j\}}$, where the row corresponding to $j \in I_1$ is sampled according to an input probability distribution, and can be algebraically expressed by

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{A}^\dagger_{I_0\cup\{j\}}(\mathbf{b}_{I_0\cup\{j\}} - \mathbf{A}_{I_0\cup\{j\}}\mathbf{x}^k). \tag{2.3}$$

This is essentially a block Kaczmarz update [Elf80; NT14], but with I_0 fixed throughout the iterations so that the iterates are confined within the selected solution space $\mathbf{A}_{I_0}\mathbf{x} = \mathbf{b}_{I_0}$. Reusing the same block allows for properties of the distinguished subsystem $\mathbf{A}_{I_0}\mathbf{x} = \mathbf{b}_{I_0}$ to be exploited, and also leads to a more efficient update formula: in Lemma 2.3.1, we prove that as long as $\mathbf{x}^k$ satisfies $\mathbf{A}_{I_0}\mathbf{x}^k = \mathbf{b}_{I_0}$ and $\mathbf{a}_j \notin \mathrm{range}(\mathbf{A}^\mathsf{T}_{I_0})$, (2.3) simplifies to

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \frac{b_j - \mathbf{a}_j^\mathsf{T}\mathbf{x}^k}{\|\mathbf{P}\mathbf{a}_j\|} \cdot \frac{\mathbf{P}\mathbf{a}_j}{\|\mathbf{P}\mathbf{a}_j\|}, \tag{2.4}$$

where $\mathbf{P} \coloneqq \mathbf{I} - \mathbf{A}^\dagger_{I_0}\mathbf{A}_{I_0}$ is the orthogonal projector onto $\mathrm{null}(\mathbf{A}_{I_0}) = \mathrm{range}(\mathbf{A}^\mathsf{T}_{I_0})^\perp$. Unlike the block update (2.3), this does not require a new pseudoinverse to be computed at every iteration and thus can be performed faster. The SC-RK method, which leverages (2.4), is summarized in Algorithm 2.1. For concreteness, we fix a particular sampling distribution for the rows of $\mathbf{A}_{I_1}$ that leads to an especially simple and interpretable analysis. By varying the distribution, better convergence rates may be possible [GR15a; AWL14].

On a conceptual level, the SC-RK update (2.4) is reminiscent of the usual Kaczmarz update (2.1), with the new direction $\mathbf{P}\mathbf{a}_j$ representing the "extra information" offered by $\mathbf{a}_j$ beyond that which is already known from being in the solution space $\mathbf{A}_{I_0}\mathbf{x} = \mathbf{b}_{I_0}$.

The following result, proved in Section 2.3.1, shows that the SC-RK method converges linearly in expectation to the solution $\mathbf{x}^*$ of $\mathbf{A}\mathbf{x} = \mathbf{b}$ under minimal assumptions.

Theorem 2.1.1. *Suppose that the rows of* $\mathbf{A}$ *are partitioned into two blocks* $\mathbf{A}_{I_0}$ *and* $\mathbf{A}_{I_1}$ *of sizes* m_0 *and* $m - m_0$*, respectively. Let* $\mathbf{P} = \mathbf{I} - \mathbf{A}^\dagger_{I_0}\mathbf{A}_{I_0}$ *be the orthogonal projector onto*

Algorithm 2.1 Subspace-constrained randomized Kaczmarz (SC-RK)

procedure SC-RK($\mathbf{A}, \mathbf{b}, I_0, K$)
  $\mathbf{P} = \mathbf{I} - \mathbf{A}_{I_0}^{\dagger}\mathbf{A}_{I_0}$ ▷ Orthogonal projector onto null($\mathbf{A}_{I_0}$)
  initialize $\mathbf{x}^0 = \mathbf{A}_{I_0}^{\dagger}\mathbf{b}_{I_0}$ ▷ Initial iterate $\mathbf{x}^0$ solves $\mathbf{A}_{I_0}\mathbf{x}^0 = \mathbf{b}_{I_0}$
  for $k = 1, \dots, K$ **do**
    sample $j \in [m] \setminus I_0$ with prob. $\|\mathbf{P}\mathbf{a}_j\|^2 / \|\mathbf{A}_{I_1}\mathbf{P}\|_F^2$ ▷ Sample row in $\mathbf{A}_{I_1}$
    $\mathbf{x}^k = \mathbf{x}^{k-1} + \frac{b_j - \mathbf{a}_j^{\mathsf{T}}\mathbf{x}^{k-1}}{\|\mathbf{P}\mathbf{a}_j\|} \cdot \frac{\mathbf{P}\mathbf{a}_j}{\|\mathbf{P}\mathbf{a}_j\|}$ ▷ Project onto $\mathbf{A}_{I_0 \cup \{j\}}\mathbf{x} = \mathbf{b}_{I_0 \cup \{j\}}$
  end for
  return $\mathbf{x}^K$
end procedure

null($\mathbf{A}_{I_0}$), *and* $\sigma_{\min}^{+}(\mathbf{A}_{I_1}\mathbf{P})$ *be the smallest non-zero singular value of* $\mathbf{A}_{I_1}\mathbf{P}$. *Then the SC-RK iterates* $\mathbf{x}^k$ *from Algorithm 2.1 converge to the solution* $\mathbf{x}^*$ *in expectation with*

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|^2 \leq \left(1 - \frac{\sigma_{\min}^{+}(\mathbf{A}_{I_1}\mathbf{P})^2}{\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2}\right)^k \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|^2. \tag{2.5}$$

In the special case that the row spaces of $\mathbf{A}_{I_0}$ and $\mathbf{A}_{I_1}$ are orthogonal (i.e., $\mathbf{A}_{I_0}\mathbf{A}_{I_1}^{\mathsf{T}} = \mathbf{0}$), the SC-RK updates (2.4) reduce to the usual Kaczmarz updates (2.1) since $\mathbf{P}\mathbf{a}_j = \mathbf{a}_j$ for all $j \in I_1$, and hence we immediately deduce the following:

Corollary 2.1.2. *Consider the same setup as Theorem 2.1.1. If* $\mathbf{A}_{I_0}\mathbf{A}_{I_1}^{\mathsf{T}} = \mathbf{0}$, *then*

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|^2 \leq \left(1 - \frac{\sigma_{\min}^{+}(\mathbf{A}_{I_1})^2}{\|\mathbf{A}_{I_1}\|_F^2}\right)^k \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|^2. \tag{2.6}$$

Noisy linear systems. In Section 2.3.2, we prove that for inconsistent systems of linear equations where a noisy measurement vector $\widehat{\mathbf{b}} \neq \mathbf{b}$ is observed, the SC-RK method converges at the same rate up to an error horizon around the solution $\mathbf{x}^*$ with a radius that depends on the noise in I_0 and I_1, as well as the geometries of $\mathbf{A}_{I_0}$ and $\mathbf{A}_{I_1}\mathbf{P}$ (Theorem 2.3.4). This expands on a phenomenon that is known from previous analyses of Kaczmarz methods [Nee10; NT14].

The analysis of inconsistent linear systems requires developing technical results involving a two-step decomposition of the block update (2.3), which provides a partial generalization of the two-subspace Kaczmarz method in [NW13] (see Remark 2.3.9).

Remark 2.1.3 (Per-iteration complexity)**.** Each SC-RK iteration can be computed in $O(m_0 n)$ flops, with calculating $\mathbf{P}\mathbf{a}_j = \mathbf{a}_j - \mathbf{A}_{I_0}^{\dagger}\mathbf{A}_{I_0}\mathbf{a}_j$ being the most expensive step. This requires directly computing $\mathbf{A}_{I_0}^{\dagger}$ or an orthonormal basis for range$(\mathbf{A}_{I_0}^{\mathsf{T}})$[1] only once, using a method based on QR decomposition or SVD. This per-iteration cost is comparable to the $O(n)$ flops per iteration of RK if m_0 is not too large. The overall complexity is then determined by multiplying the per-iteration cost by the number of iterations required to reach a desired error, which we will elaborate upon below.

Using SC-RK with a larger m_0 may still be feasible if $\mathbf{A}_{I_0}$ possesses special structure (e.g., it admits a fast multiply or is sparse). For example, an inner iterative least-squares solver (e.g., CGLS) can be used to apply $\mathbf{A}_{I_0}^{\dagger}$ using matrix-vector multiplies with $\mathbf{A}_{I_0}$ and $\mathbf{A}_{I_0}^{\mathsf{T}}$, which avoids forming the pseudoinverse explicitly. We refer to [NT14] for more related discussion on the implementation of the block Kaczmarz method.

Exploiting low-rank structure with the SC-RK method. The convergence rate of the SC-RK algorithm depends on the geometric properties of $\mathbf{A}$ and $\mathbf{P}$: Theorem 2.1.1 shows that $k_\varepsilon \coloneqq \kappa(\mathbf{A}_{I_1}\mathbf{P})^2 \log(1/\varepsilon)$ iterations suffice to achieve the relative error guarantee $\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|^2 \leq \varepsilon \|\mathbf{x}^0 - \mathbf{x}^*\|^2$, where $\kappa(\mathbf{A}_{I_1}\mathbf{P}) \coloneqq \|\mathbf{A}_{I_1}\mathbf{P}\|_F / \sigma_{\min}^{+}(\mathbf{A}_{I_1}\mathbf{P})$ is a scaled condition number of $\mathbf{A}_{I_1}\mathbf{P}$. For the same guarantee using RK, from (2.2), $\kappa(\mathbf{A})^2 \log(1/\varepsilon)$ iterations are required, where $\kappa(\mathbf{A}) \coloneqq \|\mathbf{A}\|_F / \sigma_{\min}(\mathbf{A})$.

Since $\kappa(\mathbf{A}_{I_1}\mathbf{P}) \leq \kappa(\mathbf{A})$, we see that the projector $\mathbf{P}$ acts as a right preconditioner for $\mathbf{A}$, improving the convergence rate of SC-RK compared to RK. In particular, we can expect a significant per-iteration improvement (and hence overall advantage) if $\|\mathbf{A}_{I_1}\mathbf{P}\|_F \ll \|\mathbf{A}\|_F$ or $\sigma_{\min}^{+}(\mathbf{A}_{I_1}\mathbf{P}) \gg \sigma_{\min}(\mathbf{A})$. We examine the connection between the geometry of $\mathbf{A}$ and

[1]The projector $\mathbf{A}_{I_0}^{\dagger}\mathbf{A}_{I_0}$ can be written as $\mathbf{Q}\mathbf{Q}^{\mathsf{T}}$ where $\mathbf{Q} \in \mathbb{R}^{n \times m_0}$ is a matrix whose columns form an orthonormal basis of range$(\mathbf{A}_{I_0}^{\mathsf{T}})$, which can be computed in $O(m_0^2 n)$ flops.

convergence rates in more detail in Section 2.3.3 to show that the SC-RK method is able to exploit approximately low-rank structure and geometric properties of $\mathbf{A}$ to accelerate convergence.

We also describe how a good subset I_0 of rows, if not explicitly known, can actually be *efficiently found* via a connection to low-rank matrix approximation in Section 2.3.3.

SC-RK on random data and dimension reduction. In a somewhat complementary setting, we show that for "unstructured" matrices, the subspace constraint imposed by the projector $\mathbf{P}$ *acts as a form of dimension reduction that effectively increases the aspect ratio of the system to reflect the dimensionality of the solution that remains unsolved* in Section 2.3.4. More precisely, we prove that when $\mathbf{A} \in \mathbb{R}^{m \times n}$ is drawn from a generic class of "Gaussian-like" random matrices, the SC-RK method typically converges with a rate that is approximately $1 - 1/(n - m_0)$ as long as the "effective aspect ratio" $(m - m_0)/(n - m_0)$ of the system is sufficiently large (Theorem 2.3.17). Note that $1 - 1/(n - m_0)$ is the best possible rate that can be achieved by the RK algorithm (with any sampling distribution) on a consistent $(m - m_0) \times (n - m_0)$ linear system (see [GR15a]).

The QuantileSC-RK method

We also propose a modification of the SC-RK method for solving corrupted systems of linear equations. This setting models applications where some measurements are corrupted by arbitrarily large errors, which may occur during the data collection, transmission, and storage process due to malfunctioning sensors or faulty components (for more examples, see [Stu+12; HN19]). Unlike the noisy setting above, the error horizon is not very meaningful since significant outliers can be introduced. Hence, the aim is to converge to the solution $\mathbf{x}^*$ *exactly* by identifying and avoiding corruptions, which may be possible if the number of corruptions is relatively small and the system is highly overdetermined.

Our model for *corrupted linear systems* is defined as follows. Let $\mathcal{C} \subseteq [m]$ and $\mathbf{b}_\mathcal{C} \in \mathbb{R}^n$ be a *sparse* vector of *arbitrary* (possibly adversarial) corruptions supported on $\mathcal{C}$. Moreover,

suppose that we possess external knowledge in the form of a corruption-free subset $I_0 \subset [m]$ of size m_0 such that $(\mathbf{b}_{\mathcal{C}})_{I_0} = \mathbf{0}$; for example, this could reflect a set of trustworthy measurements by a reliable source, or infallible equations arising from physical laws. The goal is to reconstruct the solution $\mathbf{x}^*$ of $\mathbf{A}\mathbf{x} = \mathbf{b}$ given $\mathbf{A}$, I_0, and the corrupted measurements $\widetilde{\mathbf{b}} \coloneqq \mathbf{b} + \mathbf{b}_{\mathcal{C}}$.

To achieve convergence, we take inspiration from the QuantileRK method proposed in [Had+22], which modifies the RK algorithm so that each projection is sampled from a set of admissible rows whose residuals $|b_j - \mathbf{a}_j^\mathsf{T}\mathbf{x}^k|$ are smaller than the q^{th} quantile of residual sizes at each iteration for some parameter $q \in (0, 1]$. This modification is based on the heuristic that large residuals should be indicative of corrupted measurements, and small residuals lead to small steps that cannot divert the iterate too far away from the solution. We propose to exploit the auxiliary information by confining the iterates of QuantileRK within the "trusted" solution space $\mathbf{A}_{I_0}\mathbf{x} = \mathbf{b}_{I_0}$. We will refer to this procedure, summarized in Algorithm 2.2, as the *QuantileSC-RK method*.

Algorithm 2.2 Quantile subspace-constrained randomized Kaczmarz (QuantileSC-RK)

procedure QUANTILESC-RK($\mathbf{A}, \widetilde{\mathbf{b}}, I_0, q, K$)
  $\mathbf{P} = \mathbf{I} - \mathbf{A}_{I_0}^\dagger \mathbf{A}_{I_0}$ ▷ Orthogonal projector onto null($\mathbf{A}_{I_0}$)
  initialize $\mathbf{x}^0 = \mathbf{A}_{I_0}^\dagger \widetilde{\mathbf{b}}_{I_0}$ ▷ Initial iterate $\mathbf{x}^0$ solves $\mathbf{A}_{I_0}\mathbf{x}^0 = \widetilde{\mathbf{b}}_{I_0}$
  for $k = 1, \dots, K$ **do**
    $\gamma_q = q$-quantile $\left\{ |\widetilde{b}_j - \mathbf{a}_j^\mathsf{T}\mathbf{x}^k|,\ j \in [m] \setminus I_0 \right\}$ ▷ Threshold based on residuals
    $J = \left\{ j \in [m] \setminus I_0 : |\widetilde{b}_j - \mathbf{a}_j^\mathsf{T}\mathbf{x}^k| \le \gamma_q \right\}$ ▷ Set of admissible rows
    sample $j \in J$ with prob. proportional to $\|\mathbf{P}\mathbf{a}_j\|^2$ ▷ Sample admissible row
    $\mathbf{x}^k = \mathbf{x}^{k-1} + \frac{\widetilde{b}_j - \mathbf{a}_j^\mathsf{T}\mathbf{x}^{k-1}}{\|\mathbf{P}\mathbf{a}_j\|} \cdot \frac{\mathbf{P}\mathbf{a}_j}{\|\mathbf{P}\mathbf{a}_j\|}$ ▷ Project onto $\mathbf{A}_{I_0 \cup \{j\}}\mathbf{x} = \widetilde{\mathbf{b}}_{I_0 \cup \{j\}}$
  end for
  return $\mathbf{x}^K$
end procedure

We prove the following result for QuantileSC-RK, a simplified version of Theorem 2.4.1 that we defer the precise statement of to Section 2.4. It shows that for unstructured matrices modelled by continuous "Gaussian-like" random matrices, the QuantileSC-RK method ro-

bustly and efficiently converges, provided that there is enough external knowledge (in terms of m_0) to make the *effective aspect ratio* $(m-m_0)/(n-m_0)$ large enough, and the proportion of corrupted measurements, $\beta := |\mathcal{C}|/(m-m_0)$, is not too large:

Theorem 2.1.4 (Simplified version of Theorem 2.4.1)**.** *Assuming that* $\mathbf{A}$ *is a continuous "Gaussian-like" random matrix, there exist positive constants* $R \geq 1$, $\beta_0 < 1$, c_1 *and* c_2*, which are independent of* m *and* n*, such that if* $(m-m_0)/(n-m_0) \geq R$ *and* $\beta \leq \beta_0$*, then with probability at least* $1 - e^{-c_1(m-m_0)}$ *over the randomness in* $\mathbf{A}$*, the QuantileSC-RK iterates* $\mathbf{x}^k$ *from Algorithm 2.2 converge to the solution* $\mathbf{x}^*$ *with*

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|^2 \leq \left(1 - \frac{c_2}{n-m_0}\right)^k \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|^2. \tag{2.7}$$

Since we are interested in large-scale systems with $m, n \gg 1$, the values of the constants c_1 and c_2 are dominated by m and n (e.g., the probability guarantee is exponentially close to one for large m). Note that this result applies to almost-square matrices with $m = (1+o(1))n$ rows provided m_0 is big enough, which lies outside the scope of existing QuantileRK theory. Experimentally, we found that the QuantileSC-RK method works well for more general data models than described by the theory, such as when $\mathbf{A}$ is a structured sparse matrix in an image reconstruction problem (see Section 2.5.6).

Remark 2.1.5. (i) (Rejection sampling). To avoid recomputing the normalizing constant $Z_J = \sum_{j\in J}\|\mathbf{P}\mathbf{a}_j\|^2$ in every iteration of Algorithm 2.2 for sampling a row from the admissible set J, which depends on $\mathbf{x}^k$, rejection sampling (as originally proposed in [Had+22]) can be used: i.e., in each iteration, a row $j \in I_1$ is sampled with probability $\|\mathbf{P}\mathbf{a}_j\|/\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2$, and the projection is made if and only if $|b_j - \mathbf{a}_j^\top \mathbf{x}^k| \leq \gamma_q$.

(ii) (Uniform sampling). It is computationally more efficient to sample rows $\mathbf{a}_j$ uniformly at random from I_1, together with rejection sampling, in Algorithm 2.2. By using the threshold $\widetilde{\gamma}_q = q\text{-quantile}\left\{|b_j - \mathbf{a}_j^\top \mathbf{x}^k|/\|\mathbf{P}\mathbf{a}_j\|,\ j \in I_1\right\}$, which has been modified to capture the heterogeneity of the projected row norms, instead of γ_q, it can be shown

that analogues of our results (e.g. Theorem 2.4.1 and Lemma 2.4.2) still hold, except that the relevant spectral quantities come from the matrix $\mathbf{D}\mathbf{A}_{I_1}\mathbf{P}$, where $\mathbf{D}$ is the diagonal matrix with entries $\|\mathbf{P}\mathbf{a}_j\|^{-1}$, $j \in I_1$, instead of $\mathbf{A}_{I_1}\mathbf{P}$.

2.1.3 Organization

Section 2.2 discusses related works. Section 2.3 analyzes the SC-RK algorithm: we prove the convergence result (Theorem 2.1.1) in Section 2.3.1 and generalize it to the noisy setting in Section 2.3.2 (Theorem 2.3.4). We provide several results on using the SC-RK method to exploit low-rank structure and geometric properties of $\mathbf{A}$ in Section 2.3.3. Furthermore, we show that the subspace constraint acts as a form of dimension reduction when $\mathbf{A}$ is a Gaussian-like random matrix in Section 2.3.4. Section 2.4 analyzes the QuantileSC-RK algorithm for solving corrupted linear systems. We provide various numerical experiments in Section 2.5 to complement our theoretical results, and conclude in Section 2.6.

2.2 Related works

Kaczmarz-type methods Kaczmarz-type algorithms are related to a variety of modern algorithms in (randomized) numerical linear algebra and (stochastic) optimization. Randomized Kaczmarz (RK) can be viewed as an instance of the stochastic gradient descent (SGD) algorithm with a particular step size, which, based on this connection, has led to new insights into both methods, such as highlighting the role of weighted sampling for SGD [NSW16]. Furthermore, the RK method is one of the basic representatives of the sketch-and-project method [GR15a], which provides a unified framework for iteratively solving linear systems—including the randomized coordinate descent method, related block variants, and the randomized Newton method—and can also be directly extended to non-linear optimization problems [Gow+19a].

Recently, methods based on the Kaczmarz algorithm have been used in the design of more sophisticated linear solvers. A GMRES-type solver preconditioned by randomized and greedy Kaczmarz inner-iterations is studied in [Du+21]. Within the sketch-and-project framework, a randomized block Kaczmarz algorithm that uses the preconditioned conjugate gradient method to perform an inexact projection in each iteration is analyzed in [DY24], and the method is proven to be especially computationally efficient when the matrix $\mathbf{A}$ has a flat-tailed spectrum. An iterative solver that further combines these ideas with momentum and sparse sketching matrices is analyzed in [Der+25a]. In this chapter, we focus on the randomized Kaczmarz algorithm for solving systems of linear equations.

Randomized Kaczmarz The analysis of the RK algorithm in [SV09] spurred many developments and variants, including randomized block Kaczmarz methods [NT14; Nec19] and Kaczmarz-Motzkin methods that combine sampling and greedy row selection [DHN17; HM21]. It is shown that the Kaczmarz method can be extended to solve least squares problems in [ZF13; NZZ15; MNR15], and systems of linear inequalities in [LL10; BN15]. The duality between RK and randomized coordinate descent (also called randomized Gauss-Seidel) has motivated a unified description of the two methods and their extended versions [MNR15]. By using ideas from optimization, RK methods with varying step sizes [NSW16], acceleration [LW16], and that promote sparsity [Lor+14; SL19; Sch+22] have also been studied.

The SC-RK method resembles the block Kaczmarz method studied in [NT14]: the difference is that the blocks differ by one row between iterations and thus the update simplifies so that computing new pseudoinverses is not required. Therefore, the SC-RK method can offer a similar advantage from using blocks in an efficient manner if a "good" block $\mathbf{A}_{I_0}$ can be found (see Section 2.3.3 for a discussion of what a good block is, and how one might be found). In [NW13; Wu22], a two-subspace Kaczmarz method that iteratively projects onto the solution space associated with two rows is shown to significantly outperform the RK

method when the system has correlated rows. The SC-RK method can be considered as an extension of this idea to higher dimensional subspaces (see Remark 2.3.9).

The idea of constraining the iterates of the RK algorithm has also implicitly appeared in the design of a fast solver for Laplacian systems in the theoretical computer science literature in [Kel+13], where the row spaces of the blocks $\mathbf{A}_{I_0}$ and $\mathbf{A}_{I_1}$ are orthogonal by construction and hence Corollary 2.1.2 applies. The SC-RK method offers a general framework for analyzing convergence when $\mathbf{A}_{I_0}$ and $\mathbf{A}_{I_1}$ are not orthogonal. Randomized sketch descent methods for solving optimization problems subject to linear constraints are also studied in [NT21], where in each iteration the coordinate space (corresponding to $\mathbf{x}$) is randomly sketched for dimensionality reduction and (random) projection matrices, analogous to $\mathbf{P}$, enforce the linear constraints.

Corrupted linear systems The literature on solving linear systems with arbitrary sparse corruptions is abundant: see, e.g., [AK95; ABH05; DT19; Can+06; Stu+12]. Such problems are often tackled within the compressed sensing and robust statistics literature using methods based on linear or SDP relaxations. The closest line of work to our approach is on iterative, row-action, corruption-avoiding algorithms. The first Kaczmarz-type method was proposed in [HN19], which introduced the idea that large residuals should be indicative of corrupted equations, but makes strong restrictions on the number of corrupted measurements (scaling sublinearly with m). The QuantileRK method, introduced in [Had+22], utilizes quantile-based steps based on this residual heuristic. An important bottleneck of this method is that the linear system generally needs to be sufficiently overdetermined (i.e., $m \geq Cn$ for a large constant C) to guarantee convergence. We show that with enough external knowledge (i.e., m_0 large enough), the QuantileSC-RK method works even for almost-square systems. Other works studying the QuantileRK method include [Ste23; JN21; Che+23]; in particular, we adapt a deterministic sufficient condition for convergence from [Ste23] for QuantileSC-RK (see Lemma 2.4.2). Another Kaczmarz-type method based on obtaining sparse least

squares solutions is analyzed in [Sch+22] and demonstrated to be able to solve linear systems corrupted by impulsive noise.

2.3 Analysis of subspace-constrained randomized Kaczmarz (SC-RK)

In this section, we provide theoretical analysis of the SC-RK method (Algorithm 2.1). Recall that $\mathbf{P} = \mathbf{I} - \mathbf{A}_{I_0}^{\dagger}\mathbf{A}_{I_0}$ is the orthogonal projector onto $\operatorname{null}(\mathbf{A}_{I_0})$, which is equal to $\operatorname{range}(\mathbf{A}_{I_0}^{\mathsf{T}})^{\perp}$, the orthogonal complement of the row space of $\mathbf{A}_{I_0}$.

2.3.1 Simplified SC-RK update formula and proof of Theorem 2.1.1

First, we provide a proof of how the block update (2.3) simplifies to the more interpretable and computationally efficient formula (2.4).

Lemma 2.3.1. *Let* $\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{A}_{I_0\cup\{j\}}^{\dagger}(\mathbf{b}_{I_0\cup\{j\}} - \mathbf{A}_{I_0\cup\{j\}}\mathbf{x}^k)$ *and* $\mathbf{P} = \mathbf{I} - \mathbf{A}_{I_0}^{\dagger}\mathbf{A}_{I_0}$. *If* $\mathbf{x}^k$ *solves* $\mathbf{A}_{I_0}\mathbf{x}^k = \mathbf{b}_{I_0}$ *and* $\mathbf{P}\mathbf{a}_j \neq \mathbf{0}$, *then*

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \frac{b_j - \mathbf{a}_j^{\mathsf{T}}\mathbf{x}^k}{\|\mathbf{P}\mathbf{a}_j\|} \cdot \frac{\mathbf{P}\mathbf{a}_j}{\|\mathbf{P}\mathbf{a}_j\|}.$$

Proof. We may assume $b_j - \mathbf{a}_j^{\mathsf{T}}\mathbf{x}^k \neq 0$, otherwise $\mathbf{x}^{k+1} = \mathbf{x}^k$. Since $\mathbf{x}^{k+1}$ is the orthogonal projection of $\mathbf{x}^k$ onto the solution space $\mathbf{A}_{I_0\cup\{j\}}\mathbf{x} = \mathbf{b}_{I_0\cup\{j\}}$, the increment $\mathbf{z} \coloneqq \mathbf{x}^{k+1} - \mathbf{x}^k$ is the solution of the following optimization problem:

$$\min_{\mathbf{z}\in\mathbb{R}^n} \frac{1}{2}\|\mathbf{z}\|^2 \quad \text{subject to} \quad \mathbf{A}_{I_0}(\mathbf{x}^k + \mathbf{z}) = \mathbf{b}_{I_0},\ \mathbf{a}_j^{\mathsf{T}}(\mathbf{x}^k + \mathbf{z}) = b_j. \tag{2.8}$$

This can be solved by introducing the Lagrange multipliers $\boldsymbol{\lambda} \in \mathbb{R}^n$ and $\tau \in \mathbb{R}$ for the two constraints. Since $\mathbf{A}_{I_0}\mathbf{x}^k = \mathbf{b}_{I_0}$, the first constraint is equivalent to $\mathbf{A}_{I_0}\mathbf{z} = \mathbf{0}$, and thus $\mathbf{z}$

solves

$$\mathbf{z} + \mathbf{A}_{I_0}^\mathsf{T}\boldsymbol{\lambda} + \tau\mathbf{a}_j = \mathbf{0} \tag{2.9}$$

whilst satisfying $\mathbf{A}_{I_0}\mathbf{z} = \mathbf{0}$ and $\mathbf{a}_j^\mathsf{T}\mathbf{z} = b_j - \mathbf{a}_j^\mathsf{T}\mathbf{x}^k$. Since $\mathbf{P}$ is the orthogonal projector onto $\text{null}(\mathbf{A}_{I_0})$, $\mathbf{P}\mathbf{A}_{I_0}^\mathsf{T} = \mathbf{0}$ and the first constraint is equivalent to $\mathbf{P}\mathbf{z} = \mathbf{z}$. Hence, pre-multiplying (2.9) by $\mathbf{P}$ implies that $\mathbf{z} = -\tau\mathbf{P}\mathbf{a}_j$. Furthermore, pre-multiplying (2.9) by $\mathbf{z}^\mathsf{T}$ and using the constraints implies that $\tau = -\|\mathbf{z}\|^2/(b_j - \mathbf{a}_j^\mathsf{T}\mathbf{x}^k) = -\tau^2\|\mathbf{P}\mathbf{a}_j\|^2/(b_j - \mathbf{a}_j^\mathsf{T}\mathbf{x}^k)$. Solving for τ yields $\tau = -(b_j - \mathbf{a}_j^\mathsf{T}\mathbf{x}^k)/\|\mathbf{P}\mathbf{a}_j\|^2$, which completes the proof. □

Remark 2.3.2. From the optimization formulation (2.8), it can also be shown that the unit direction $\mathbf{P}\mathbf{a}_j/\|\mathbf{P}\mathbf{a}_j\|$ taken from $\mathbf{x}^k$ to reach $\mathbf{x}^{k+1}$ maximizes $|\mathbf{a}_j^\mathsf{T}\tilde{\mathbf{z}}|^2$ over all unit vectors $\tilde{\mathbf{z}} \in \text{null}(\mathbf{A}_{I_0})$. This provides a nice geometric interpretation of the SC-RK update: *the direction* $\mathbf{P}\mathbf{a}_j$ *taken to reach the solution space* $\mathbf{A}_{I_0\cup\{j\}}\mathbf{x} = \mathbf{b}_{I_0\cup\{j\}}$ *minimizes the angle from the optimal direction* $\mathbf{a}_j$ *for reaching the solution space* $\mathbf{a}_j^\mathsf{T}\mathbf{x} = b_j$ *within the subspace* $\text{null}(\mathbf{A}_{I_0})$; see Figure 2.1 for an illustration. For an alternative algebraic proof of a more general version of Lemma 2.3.1, see Remark 2.3.9 later.

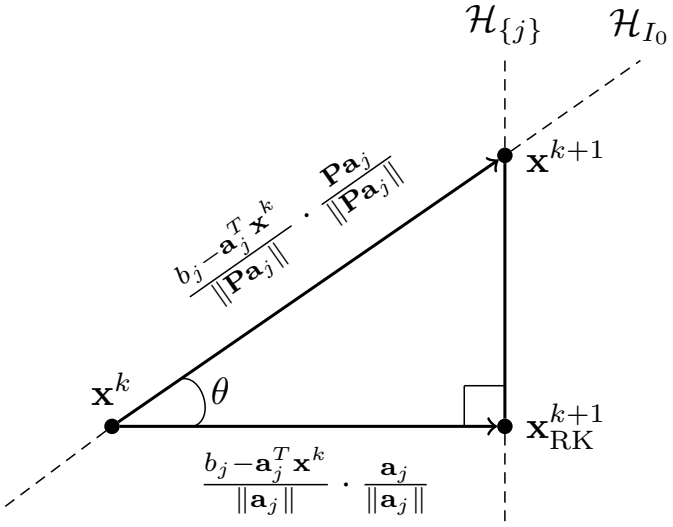


Figure 2.1: SC-RK update from the current iterate $\mathbf{x}^k$ for reaching the vector $\mathbf{x}^{k+1}$ in the solution space $\mathcal{H}_{\{j\}} = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{a}_j^\mathsf{T}\mathbf{x} = b_j\}$ whilst remaining within $\mathcal{H}_{I_0} = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{A}_{I_0}\mathbf{x} = \mathbf{b}_{I_0}\}$, compared to the RK update for reaching $\mathbf{x}_{\text{RK}}^{k+1}$ alone.

We will now use the simplified update formula in Lemma 2.3.1 to prove Theorem 2.1.1.

Proof of Theorem 2.1.1. Consider the k^{th} iterate $\mathbf{x}^k$. Suppose that $\mathbf{a}_j$ is sampled in the next iteration (with $\mathbf{P}\mathbf{a}_j \neq \mathbf{0}$). By subtracting $\mathbf{x}^*$ from both sides of (2.4) and noting that $\mathbf{a}_j^\mathsf{T}(\mathbf{x}^k - \mathbf{x}^*) = \mathbf{a}_j^\mathsf{T}\mathbf{P}(\mathbf{x}^k - \mathbf{x}^*)$ for any $j \in I_1$ since $\mathbf{x}^k - \mathbf{x}^* \in \text{null}(\mathbf{A}_{I_0})$, we have

$$\mathbf{x}^{k+1} - \mathbf{x}^* = \left(\mathbf{I} - \mathbf{v}\mathbf{v}^\mathsf{T}\right)(\mathbf{x}^k - \mathbf{x}^*) \quad \text{for } \mathbf{v} \coloneqq \frac{\mathbf{P}\mathbf{a}_j}{\|\mathbf{P}\mathbf{a}_j\|}.$$

Since $\mathbf{v}\mathbf{v}^\mathsf{T}$ is an orthogonal projector, $(\mathbf{x}^{k+1} - \mathbf{x}^*) \perp \mathbf{v}\mathbf{v}^\mathsf{T}(\mathbf{x}^k - \mathbf{x}^*)$. Thus, by Pythagoras' theorem,

$$\|\mathbf{x}^{k+1} - \mathbf{x}^*\|^2 = \|\mathbf{x}^k - \mathbf{x}^*\|^2 - \|\mathbf{v}\mathbf{v}^\mathsf{T}(\mathbf{x}^k - \mathbf{x}^*)\|^2 = \|\mathbf{x}^k - \mathbf{x}^*\|^2 - \left|\mathbf{v}^\mathsf{T}(\mathbf{x}^k - \mathbf{x}^*)\right|^2.$$

By taking expectation (where each row $\mathbf{a}_j$ is sampled with probability $\|\mathbf{P}\mathbf{a}_j\|^2/\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2$), conditional on all the choices up to the k^{th} iteration, we obtain

$$\begin{aligned}
\mathbb{E}_k\|\mathbf{x}^{k+1} - \mathbf{x}^*\|^2 &= \|\mathbf{x}^k - \mathbf{x}^*\|^2 - \sum_{j \in I_1 : \mathbf{P}\mathbf{a}_j \neq \mathbf{0}} \frac{\|\mathbf{P}\mathbf{a}_j\|^2}{\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2} \cdot \left| \left(\frac{\mathbf{P}\mathbf{a}_j}{\|\mathbf{P}\mathbf{a}_j\|}\right)^\mathsf{T} (\mathbf{x}^k - \mathbf{x}^*) \right|^2 \\
&= \|\mathbf{x}^k - \mathbf{x}^*\|^2 - \frac{1}{\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2} \|\mathbf{A}_{I_1}\mathbf{P}(\mathbf{x}^k - \mathbf{x}^*)\|^2 \\
&= \left(1 - \frac{\theta^2}{\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2}\right) \cdot \|\mathbf{x}^k - \mathbf{x}^*\|^2, \quad \text{where } \theta \coloneqq \left\| \mathbf{A}_{I_1}\mathbf{P} \left(\frac{\mathbf{x}^k - \mathbf{x}^*}{\|\mathbf{x}^k - \mathbf{x}^*\|}\right) \right\|.
\end{aligned}$$

The next step is to estimate θ from below, which requires more care than a similar estimate used to prove convergence of the RK method [SV09] since $\mathbf{A}_{I_1}\mathbf{P}$ has a nontrivial nullspace. A similar case where the system matrix has a nontrivial nullspace was also treated in [LRS18]. First, observe that $\text{null}(\mathbf{A}_{I_1}\mathbf{P}) = \text{null}(\mathbf{P})$. Indeed, the nontrivial inclusion $\text{null}(\mathbf{A}_{I_1}\mathbf{P}) \subseteq \text{null}(\mathbf{P})$ follows from the observation that $\mathbf{A}_{I_1}\mathbf{P}\mathbf{y} = \mathbf{0}$ implies that $\mathbf{P}\mathbf{y} \in \text{null}(\mathbf{A}_{I_0}) \cap \text{null}(\mathbf{A}_{I_1}) = \text{null}(\mathbf{A}) = \{\mathbf{0}\}$, since $\mathbf{A}$ has full rank. Therefore, since $\mathbf{x}^k - \mathbf{x}^* \in \text{null}(\mathbf{A}_{I_0})$ is orthogonal to $\text{null}(\mathbf{P}) = \text{null}(\mathbf{A}_{I_1}\mathbf{P})$,

$$\theta^2 = \left\| \mathbf{A}_{I_1}\mathbf{P} \left(\frac{\mathbf{x}^k - \mathbf{x}^*}{\|\mathbf{x}^k - \mathbf{x}^*\|}\right) \right\|^2 \geq \min_{\substack{\mathbf{z} \in \text{null}(\mathbf{A}_{I_0}) \\ \|\mathbf{z}\| = 1}} \|\mathbf{A}_{I_1}\mathbf{z}\|^2 = \sigma_{\min}^+(\mathbf{A}_{I_1}\mathbf{P})^2. \tag{2.10}$$

This implies the following bound for one step of the SC-RK method:

$$\mathbb{E}_k\|\mathbf{x}^{k+1}-\mathbf{x}^*\|^2 \leq \left(1-\frac{\sigma_{\min}^+(\mathbf{A}_{I_1}\mathbf{P})^2}{\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2}\right)\cdot\|\mathbf{x}^k-\mathbf{x}^*\|^2.$$

By iterating and taking the full expectation, this concludes the proof of Theorem 2.1.1. □

Remark 2.3.3. If $\mathbf{v}_\ell$ is a right singular vector of $\mathbf{A}_{I_1}\mathbf{P}$ corresponding to the ℓ^{th} largest singular value $\sigma_\ell(\mathbf{A}_{I_1}\mathbf{P})$, it can be shown that

$$\mathbb{E}\left\langle \mathbf{x}^k-\mathbf{x}^*, \mathbf{v}_\ell\right\rangle = \left(1-\frac{\sigma_\ell(\mathbf{A}_{I_1}\mathbf{P})^2}{\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2}\right)^k\cdot\left\langle \mathbf{x}^0-\mathbf{x}^*, \mathbf{v}_\ell\right\rangle.$$

This shows that the residual vector $\mathbf{x}^k-\mathbf{x}^*$ decays fastest in the directions corresponding to the largest singular values of $\mathbf{A}_{I_1}$ restricted to $\text{null}(\mathbf{A}_{I_0})$. This phenomenon was proved by Steinerberger [Ste21] for the RK method.

2.3.2 SC-RK convergence on inconsistent linear systems

In the general case, the measurement vector $\mathbf{b}$ might not be known exactly, but only accessible through a set of noisy observations $\widehat{\mathbf{b}} \coloneqq \mathbf{b}+\mathbf{R}$, where $\mathbf{R}$ is an arbitrary error vector (which is considered to be small). Similar to previous analyses of Kaczmarz methods [Nee10; NT14; RN21], we prove that if the SC-RK method is used with the noisy measurements $\widehat{\mathbf{b}}$, then the iterates converge to the solution $\mathbf{x}^*$ up to an error horizon:

Theorem 2.3.4. *Suppose that the rows of* $\mathbf{A}$ *are partitioned into two blocks* $\mathbf{A}_{I_0}$ *and* $\mathbf{A}_{I_1}$ *of sizes* m_0 *and* $m-m_0$*, respectively, and assume that* $\mathbf{A}_{I_0}$ *has full row rank. If* $\widehat{\mathbf{x}}^k$ *denotes the sequence of SC-RK iterates from Algorithm 2.1 where the noisy measurement vector* $\widehat{\mathbf{b}}=\mathbf{b}+\mathbf{R}$ *is used in place of* $\mathbf{b}$*, and the initial iterate* $\widehat{\mathbf{x}}^0$ *solves* $\mathbf{A}_{I_0}\widehat{\mathbf{x}}^0=\widehat{\mathbf{b}}_{I_0}$*, then*

$$\mathbb{E}\|\widehat{\mathbf{x}}^k-\mathbf{x}^*\|^2 \leq \left(1-\frac{\sigma_{\min}^+(\mathbf{A}_{I_1}\mathbf{P})^2}{\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2}\right)^k\cdot\|\widehat{\mathbf{x}}^0-\mathbf{x}^*\|^2+\gamma_0+\gamma_1,$$

where $\gamma_0, \gamma_1 \geq 0$ *are given by*

$$\gamma_0 = \frac{2\|\mathbf{R}_{I_0}\|^2}{\sigma_{\min}(\mathbf{A}_{I_0})^2} - \|\mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0}\|^2 \quad \textit{and} \quad \gamma_1 = \frac{\|\mathbf{R}_{I_1} - \mathbf{A}_{I_1}\mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0}\|^2}{\sigma_{\min}^+(\mathbf{A}_{I_1}\mathbf{P})^2}.$$

Remark 2.3.5 (Error horizon)**.** Note that γ_0 only depends on the noise in the measurements corresponding to the fixed block I_0. In particular, if $\mathbf{R}_{I_0} = 0$, then $\gamma_0 = 0$ and γ_1 only depends on $\|\mathbf{R}_{I_1}\|^2$. On the other hand, if $\mathbf{R}_{I_1} = 0$, then γ_1 only depends on $\|\mathbf{A}_{I_1}\mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0}\|^2 = \sum_{j \in I_1} |\mathbf{a}_j^\mathsf{T} \mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0}|^2$. Note that $|\mathbf{a}_j^\mathsf{T} \mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0}|$ corresponds to the angle between the row $\mathbf{a}_j$ and $\mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0}$; the vector $\mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0}$ accounts for how noise in the measurements corresponding to the fixed block $\mathbf{A}_{I_0}$ shifts the solution space (see Lemma 2.3.8a). Finally, if $\mathbf{R} = 0$, then $\gamma_0 = \gamma_1 = 0$ and we recover Theorem 2.1.1.

Remark 2.3.6 (Least squares)**.** Given a set of noisy measurements $\widehat{\mathbf{b}}$, our setup can easily be translated to the problem of solving the *inconsistent* system of linear equations $\mathbf{A}\mathbf{x} \approx \widehat{\mathbf{b}}$ in a least squares sense. In this setting, by defining $\mathbf{x}^*$ to be the least squares solution (i.e., $\mathbf{x}^* \coloneqq \arg\min_{\mathbf{x} \in \mathbb{R}^n} \|\mathbf{A}\mathbf{x} - \widehat{\mathbf{b}}\|^2 = \mathbf{A}^\dagger \widehat{\mathbf{b}}$), and setting $\mathbf{b} \coloneqq \mathbf{A}\mathbf{x}^*$ and $\mathbf{R} \coloneqq \widehat{\mathbf{b}} - \mathbf{A}\mathbf{x}^*$, Theorem 2.3.4 can be applied to deduce that the SC-RK iterates converge to the least squares solution up to the same error horizon.

We develop some technical results before proving Theorem 2.3.4. Note that with noisy measurements $\widehat{\mathbf{b}}$, the relationship $\widehat{\mathbf{x}}^k - \mathbf{x}^* \in \mathrm{null}(\mathbf{A}_{I_0})$ does not necessarily hold anymore. Thus, it will be more convenient to work directly with the block update (2.3) instead. First, we present a decomposition of the pseudoinverse $\mathbf{A}_{I \cup J}^\dagger$ in terms of $\mathbf{A}_I$ and $\mathbf{A}_J$.

Lemma 2.3.7. *Let* $\mathbf{A} \in \mathbb{R}^{m \times n}$ *and* $I, J \subseteq [m]$ *be two disjoint subsets of row indices. If* $\mathbf{A}_{I \cup J}$ *has full row rank, then the pseudoinverse* $\mathbf{A}_{I \cup J}^\dagger$ *admits the block representation*

$$\mathbf{A}_{I \cup J}^\dagger = \left(\mathbf{A}_I^\dagger - (\mathbf{A}_J \mathbf{P})^\dagger \mathbf{A}_J \mathbf{A}_I^\dagger \;\middle|\; (\mathbf{A}_J \mathbf{P})^\dagger \right), \tag{2.11}$$

where $\mathbf{P} = \mathbf{I} - \mathbf{A}_I^\dagger \mathbf{A}_I$ *is the orthogonal projection operator onto* $\mathrm{null}(\mathbf{A}_I)$.

Proof of Lemma 2.3.7. First, we record the key algebraic property that will be used repeatedly:

$$\mathbf{X}^\dagger = \mathbf{X}^\mathsf{T}(\mathbf{X}\mathbf{X}^\mathsf{T})^{-1} \quad \text{for } \mathbf{X} = \mathbf{A}_{I\cup J},\ \mathbf{A}_I, \text{ or } \mathbf{A}_J\mathbf{P}. \tag{2.12}$$

This follows since all three matrices have full row rank: $\mathbf{A}_{I\cup J}$ by assumption, $\mathbf{A}_I$ as its row subset, and $\mathbf{A}_J\mathbf{P}$ from the observation that if its rows were linearly dependent then there would exist some nonzero $\boldsymbol{\alpha} \in \mathbb{R}^{|J|}$ such that $\sum_{j\in J} \alpha_j \mathbf{a}_j^\mathsf{T}\mathbf{P} = \mathbf{0}$, which would imply that $\sum_{j\in J} \alpha_j \mathbf{a}_j \in \mathrm{null}(\mathbf{P}) = \mathrm{range}(\mathbf{A}_I^\mathsf{T})$ and thus contradict the assumption that $\mathbf{A}_{I\cup J}$ has full row rank. We have

$$\mathbf{A}_{I\cup J}\mathbf{A}_{I\cup J}^\mathsf{T} = \left(\begin{array}{c|c} \mathbf{A}_I\mathbf{A}_I^\mathsf{T} & \mathbf{A}_I\mathbf{A}_J^\mathsf{T} \\ \hline (\mathbf{A}_I\mathbf{A}_J^\mathsf{T})^\mathsf{T} & \mathbf{A}_J\mathbf{A}_J^\mathsf{T} \end{array}\right) =: \left(\begin{array}{c|c} \mathbf{A}_{II} & \mathbf{A}_{IJ} \\ \hline \mathbf{A}_{IJ}^\mathsf{T} & \mathbf{A}_{JJ} \end{array}\right).$$

Since $\mathbf{A}_I$ has full row rank, $\mathbf{A}_{II}$ is invertible and

$$\begin{aligned} \mathbf{A}_{I\cup J}^\dagger &\overset{(2.12)}{=} \mathbf{A}_{I\cup J}^\mathsf{T}\left(\mathbf{A}_{I\cup J}\mathbf{A}_{I\cup J}^\mathsf{T}\right)^{-1} \\ &= \left(\mathbf{A}_I^\mathsf{T} \,\middle|\, \mathbf{A}_J^\mathsf{T}\right) \left(\begin{array}{c|c} \mathbf{A}_{II}^{-1} + \mathbf{A}_{II}^{-1}\mathbf{A}_{IJ}\mathbf{R}^{-1}\mathbf{A}_{IJ}^\mathsf{T}\mathbf{A}_{II}^{-1} & -\mathbf{A}_{II}^{-1}\mathbf{A}_{IJ}\mathbf{R}^{-1} \\ \hline -\mathbf{R}^{-1}\mathbf{A}_{IJ}^\mathsf{T}\mathbf{A}_{II}^{-1} & \mathbf{R}^{-1} \end{array}\right), \end{aligned} \tag{2.13}$$

where $\mathbf{R} \coloneqq \mathbf{A}_{JJ} - \mathbf{A}_{IJ}^\mathsf{T}\mathbf{A}_{II}^{-1}\mathbf{A}_{IJ}$ is the Schur complement [HJ12] of the block $\mathbf{A}_{II}$. Recall that $\mathbf{P} = \mathbf{I} - \mathbf{A}_I^\dagger\mathbf{A}_I$, and so

$$\mathbf{R} = \mathbf{A}_J\mathbf{A}_J^\mathsf{T} - \mathbf{A}_J\mathbf{A}_I^\mathsf{T}(\mathbf{A}_I\mathbf{A}_I^\mathsf{T})^{-1}\mathbf{A}_I\mathbf{A}_J^\mathsf{T} \overset{(2.12)}{=} \mathbf{A}_J[\mathbf{I} - \mathbf{A}_I^\dagger\mathbf{A}_I]\mathbf{A}_J^\mathsf{T} = \mathbf{A}_J\mathbf{P}\mathbf{A}_J^\mathsf{T},$$

which implies that

$$\mathbf{P}\mathbf{A}_J^\mathsf{T}\mathbf{R}^{-1} = \mathbf{P}\mathbf{A}_J^\mathsf{T}(\mathbf{A}_J\mathbf{P}\mathbf{A}_J^\mathsf{T})^{-1} = (\mathbf{A}_J\mathbf{P})^\mathsf{T}((\mathbf{A}_J\mathbf{P})(\mathbf{A}_J\mathbf{P})^\mathsf{T})^{-1} \overset{(2.12)}{=} (\mathbf{A}_J\mathbf{P})^\dagger. \tag{2.14}$$

Next we compute expressions for the two blocks of $\mathbf{A}_{I\cup J}^\dagger$ in (2.13). The first block, $\mathbf{A}_I^\mathsf{T}[\mathbf{A}_{II}^{-1} + \mathbf{A}_{II}^{-1}\mathbf{A}_{IJ}\mathbf{R}^{-1}\mathbf{A}_{IJ}^\mathsf{T}\mathbf{A}_{II}^{-1}] + \mathbf{A}_J^\mathsf{T}[-\mathbf{R}^{-1}\mathbf{A}_{IJ}^\mathsf{T}\mathbf{A}_{II}^{-1}]$, simplifies to

$$\mathbf{A}_I^\dagger - [-\mathbf{A}_I^\dagger\mathbf{A}_I + \mathbf{I}]\mathbf{A}_J^\mathsf{T}\mathbf{R}^{-1}\mathbf{A}_J\mathbf{A}_I^\dagger = \mathbf{A}_I^\dagger - \mathbf{P}\mathbf{A}_J^\mathsf{T}\mathbf{R}^{-1}\mathbf{A}_J\mathbf{A}_I^\dagger \overset{(2.14)}{=} \mathbf{A}_I^\dagger - (\mathbf{A}_J\mathbf{P})^\dagger\mathbf{A}_J\mathbf{A}_I^\dagger.$$

The second block, $\mathbf{A}_I^\mathsf{T}[-\mathbf{A}_{II}^{-1}\mathbf{A}_{IJ}\mathbf{R}^{-1}] + \mathbf{A}_J^\mathsf{T}[\mathbf{R}^{-1}]$, simplifies to

$$[-\mathbf{A}_I^\dagger\mathbf{A}_I + \mathbf{I}]\mathbf{A}_J^\mathsf{T}\mathbf{R}^{-1} = \mathbf{P}\mathbf{A}_J^\mathsf{T}\mathbf{R}^{-1} \overset{(2.14)}{=} (\mathbf{A}_J\mathbf{P})^\dagger.$$

Combining the two preceding displayed equations completes the proof. □

Next, we describe how the noise affects the geometry of the solution spaces.

Lemma 2.3.8. *Denote the true and noisy solution spaces associated with $I \subset [m]$ by*

$$\mathcal{H}_I = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{A}_I\mathbf{x} = \mathbf{b}_I\} \quad \text{and} \quad \widehat{\mathcal{H}}_I = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{A}_I\mathbf{x} = \mathbf{b}_I + \mathbf{R}_I\}, \tag{2.15}$$

respectively. If $\mathbf{A}_I$ has full row rank, then $\mathcal{H}_I$ and $\widehat{\mathcal{H}}_I$ satisfy the following:

(a) $\widehat{\mathcal{H}}_I = \mathcal{H}_I + \mathbf{A}_I^\dagger\mathbf{R}_I$.

(b) $\widehat{\mathcal{H}}_I - \mathbf{x}^* = \text{null}(\mathbf{A}_I) + \mathbf{A}_I^\dagger\mathbf{R}_I$.

(c) *The vector $\mathbf{A}_I^\dagger\mathbf{R}_I$ is orthogonal to* $\text{null}(\mathbf{A}_I)$.

Proof. (a): Since $\mathbf{A}_I$ has full row rank, $\mathbf{A}_I\mathbf{A}_I^\dagger = \mathbf{I}$. Therefore, for any $\mathbf{x} \in \mathcal{H}_I$, we have $\mathbf{A}_I(\mathbf{x} + \mathbf{A}_I^\dagger\mathbf{R}_I) = \mathbf{b}_I + \mathbf{R}_I$ and so $\mathbf{x} + \mathbf{A}_I^\dagger\mathbf{R}_I \in \widehat{\mathcal{H}}_I$. Conversely, for any $\widehat{\mathbf{x}} \in \widehat{\mathcal{H}}_I$, we have $\mathbf{A}_I(\widehat{\mathbf{x}} - \mathbf{A}_I^\dagger\mathbf{R}_I) = \mathbf{b}_I$. Thus, $(\widehat{\mathbf{x}} - \mathbf{A}_I^\dagger\mathbf{R}_I) \in \mathcal{H}_I$, and so

$$\widehat{\mathbf{x}} = (\widehat{\mathbf{x}} - \mathbf{A}_I^\dagger\mathbf{R}_I) + \mathbf{A}_I^\dagger\mathbf{R}_I \in \mathcal{H}_I + \mathbf{A}_I^\dagger\mathbf{R}_I.$$

(b): Since $\mathbf{x}^* \in \mathcal{H}_I$, we have $\mathcal{H}_I - \mathbf{x}^* = \text{null}(\mathbf{A}_I)$. Together with part (a), this implies (b). Finally, (c) follows from the fact $\text{range}(\mathbf{A}_I^\dagger) = \text{range}(\mathbf{A}_I^\mathsf{T}) = \text{null}(\mathbf{A}_I)^\perp$. □

We will now prove Theorem 2.3.4 by using the expression for $\mathbf{A}^{\dagger}_{I_0\cup\{j\}}$ given in Lemma 2.3.7, as well as the geometry of the shifted solution spaces described by Lemma 2.3.8.

Proof of Theorem 2.3.4. Consider the k^{th} iterate $\widehat{\mathbf{x}}^k$. Suppose that $\mathbf{a}_j$ is sampled in the next iteration (with $\mathbf{Pa}_j \neq \mathbf{0}$ and hence $\mathbf{A}_{I_0\cup\{j\}}$ has full row rank). Then one step of the SC-RK algorithm with noisy measurements corresponds to the projection of $\widehat{\mathbf{x}}^k$ onto the noisy solution space $\widehat{\mathcal{H}}_{I_0\cup\{j\}}$, namely,

$$\widehat{\mathbf{x}}^{k+1} = \widehat{\mathbf{x}}^k + \mathbf{A}^{\dagger}_{I_0\cup\{j\}}(\mathbf{b}_{I_0\cup\{j\}} + \mathbf{R}_{I_0\cup\{j\}} - \mathbf{A}_{I_0\cup\{j\}}\widehat{\mathbf{x}}^k).$$

We will compare $\widehat{\mathbf{x}}^{k+1}$ with the projection of $\widehat{\mathbf{x}}^k$ onto the true solution space $\mathcal{H}_{I_0\cup\{j\}}$, denoted by

$$\mathbf{x}^{k+1} := \widehat{\mathbf{x}}^k + \mathbf{A}^{\dagger}_{I_0\cup\{j\}}(\mathbf{b}_{I_0\cup\{j\}} - \mathbf{A}_{I_0\cup\{j\}}\widehat{\mathbf{x}}^k).$$

Step 1. Exact computations. Note that

$$\widehat{\mathbf{x}}^{k+1} - \mathbf{x}^* = (\mathbf{x}^{k+1} - \mathbf{x}^*) + \mathbf{A}^{\dagger}_{I_0\cup\{j\}}\mathbf{R}_{I_0\cup\{j\}},$$

and $\mathbf{A}^{\dagger}_{I_0\cup\{j\}}\mathbf{R}_{I_0\cup\{j\}} \perp (\mathbf{x}^{k+1} - \mathbf{x}^*) \in \text{null}(\mathbf{A}_{I_0\cup\{j\}})$ by Lemma 2.3.8c. By using Pythagoras' theorem twice (and orthogonality of the true Kaczmarz projections), we have

$$\begin{aligned}\|\widehat{\mathbf{x}}^{k+1} - \mathbf{x}^*\|^2 &= \|\mathbf{x}^{k+1} - \mathbf{x}^*\|^2 + \|\mathbf{A}^{\dagger}_{I_0\cup\{j\}}\mathbf{R}_{I_0\cup\{j\}}\|^2 \\ &= \|\widehat{\mathbf{x}}^k - \mathbf{x}^*\|^2 - \|\mathbf{A}^{\dagger}_{I_0\cup\{j\}}\mathbf{A}_{I_0\cup\{j\}}(\widehat{\mathbf{x}}^k - \mathbf{x}^*)\|^2 + \|\mathbf{A}^{\dagger}_{I_0\cup\{j\}}\mathbf{R}_{I_0\cup\{j\}}\|^2. \qquad (2.16)\end{aligned}$$

By using Lemma 2.3.7 with $I = I_0$ and $J = \{j\}$, we can simplify the last two terms: firstly,

$$\begin{aligned}\mathbf{A}^{\dagger}_{I_0\cup\{j\}}\mathbf{R}_{I_0\cup\{j\}} &= \left(\mathbf{A}^{\dagger}_{I_0} - \tfrac{\mathbf{Pa}_j\mathbf{a}_j^{\mathsf{T}}\mathbf{A}^{\dagger}_{I_0}}{\|\mathbf{Pa}_j\|^2} \;\middle|\; \tfrac{\mathbf{Pa}_j}{\|\mathbf{Pa}_j\|^2}\right)\begin{pmatrix}\mathbf{R}_{I_0}\\ r_j\end{pmatrix} \\ &= \mathbf{A}^{\dagger}_{I_0}\mathbf{R}_{I_0} + (r_j - \mathbf{a}_j^{\mathsf{T}}\mathbf{A}^{\dagger}_{I_0}\mathbf{R}_{I_0})\cdot\frac{\mathbf{Pa}_j}{\|\mathbf{Pa}_j\|^2}. \qquad (2.17)\end{aligned}$$

Next, since $\widehat{\mathbf{x}}^k - \mathbf{x}^* - \mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0} \in \mathrm{null}(\mathbf{A}_{I_0})$ (from Lemma 2.3.8b) is a fixed point for $\mathbf{P}$,

$$\begin{aligned}\mathbf{A}_{I_0\cup\{j\}}^\dagger \mathbf{A}_{I_0\cup\{j\}}(\widehat{\mathbf{x}}^k - \mathbf{x}^*) &= \left(\mathbf{A}_{I_0}^\dagger - \tfrac{\mathbf{P}\mathbf{a}_j \mathbf{a}_j^\mathsf{T} \mathbf{A}_{I_0}^\dagger}{\|\mathbf{P}\mathbf{a}_j\|^2} \;\middle|\; \tfrac{\mathbf{P}\mathbf{a}_j}{\|\mathbf{P}\mathbf{a}_j\|^2}\right) \begin{pmatrix} \mathbf{R}_{I_0} \\ \hline \mathbf{a}_j^\mathsf{T}(\widehat{\mathbf{x}}^k - \mathbf{x}^*) \end{pmatrix} \\ &= \mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0} + \frac{(\mathbf{P}\mathbf{a}_j)^\mathsf{T}(\widehat{\mathbf{x}}^k - \mathbf{x}^* - \mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0})}{\|\mathbf{P}\mathbf{a}_j\|} \cdot \frac{\mathbf{P}\mathbf{a}_j}{\|\mathbf{P}\mathbf{a}_j\|}, \end{aligned} \tag{2.18}$$

Furthermore, by Lemma 2.3.8c, $\mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0} \perp \mathbf{P}\mathbf{a}_j \in \mathrm{null}(\mathbf{A}_{I_0})$, which implies that the two summands in both (2.17) and (2.18) are orthogonal. Hence, we can further expand (2.16) to show that $\|\widehat{\mathbf{x}}^{k+1} - \mathbf{x}^*\|^2$ is equal to

$$\|\widehat{\mathbf{x}}^k - \mathbf{x}^*\|^2 - \|\mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0}\|^2 - \left|\frac{(\mathbf{P}\mathbf{a}_j)^\mathsf{T}(\widehat{\mathbf{x}}^k - \mathbf{x}^* - \mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0})}{\|\mathbf{P}\mathbf{a}_j\|}\right|^2 + \|\mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0}\|^2 + \frac{|r_j - \mathbf{a}_j^\mathsf{T} \mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0}|^2}{\|\mathbf{P}\mathbf{a}_j\|^2}.$$

By cancelling identical terms and taking the expectation, conditional on all the choices of the algorithm up to the k^{th} iteration (similar to the proof of Theorem 2.1.1), we obtain

$$\mathbb{E}_k\|\widehat{\mathbf{x}}^{k+1} - \mathbf{x}^*\|^2 = \|\widehat{\mathbf{x}}^k - \mathbf{x}^*\|^2 - \frac{\|\mathbf{A}_{I_1}\mathbf{P}(\widehat{\mathbf{x}}^k - \mathbf{x}^* - \mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0})\|^2}{\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2} + \frac{\|\mathbf{R}_{I_1} - \mathbf{A}_{I_1}\mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0}\|^2}{\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2}. \tag{2.19}$$

Step 2. Spectral bounds. Recall that $\mathrm{null}(\mathbf{A}_{I_1}\mathbf{P}) = \mathrm{null}(\mathbf{P})$ from the proof of Theorem 2.1.1. Since $\widehat{\mathbf{x}}^k - \mathbf{x}^* - \mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0} \in \mathrm{null}(\mathbf{A}_{I_0})$, which is orthogonal to $\mathrm{null}(\mathbf{P}) = \mathrm{null}(\mathbf{A}_{I_1}\mathbf{P})$, arguing as in (2.10) shows that

$$\|\mathbf{A}_{I_1}\mathbf{P}(\widehat{\mathbf{x}}^k - \mathbf{x}^* - \mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0})\|^2 \geq \sigma_{\min}^+(\mathbf{A}_{I_1}\mathbf{P})^2 \cdot \|\widehat{\mathbf{x}}^k - \mathbf{x}^* - \mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0}\|^2$$

By expanding the square, $\|\widehat{\mathbf{x}}^k - \mathbf{x}^* - \mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0}\|^2$ is equal to $\|\widehat{\mathbf{x}}^k - \mathbf{x}^*\|^2 + \|\mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0}\|^2 - 2(\widehat{\mathbf{x}}^k - \mathbf{x}^*)^\mathsf{T}\mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0}$. Since $\mathbf{A}_{I_0}^\dagger = \mathbf{A}_{I_0}^\mathsf{T}(\mathbf{A}_{I_0}\mathbf{A}_{I_0}^\mathsf{T})^{-1}$ and $\mathbf{A}_{I_0}(\widehat{\mathbf{x}}^k - \mathbf{x}^*) = \mathbf{R}_{I_0}$,

$$(\widehat{\mathbf{x}}^k - \mathbf{x}^*)^\mathsf{T}\mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0} = \mathbf{R}_{I_0}^\mathsf{T}(\mathbf{A}_{I_0}\mathbf{A}_{I_0}^\mathsf{T})^{-1}\mathbf{R}_{I_0} \leq \frac{\|\mathbf{R}_{I_0}\|^2}{\sigma_{\min}(\mathbf{A}_{I_0})^2}.$$

Hence, we can bound the second term of (2.19) from below by

$$\frac{\|\mathbf{A}_{I_1}\mathbf{P}(\widehat{\mathbf{x}}^k - \mathbf{x}^* - \mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0})\|^2}{\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2} \geq \frac{\sigma_{\min}^+(\mathbf{A}_{I_1}\mathbf{P})^2}{\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2}\left(\|\widehat{\mathbf{x}}^k - \mathbf{x}^*\|^2 + \|\mathbf{A}_{I_0}^\dagger \mathbf{R}_{I_0}\|^2 - \frac{2\|\mathbf{R}_{I_0}\|^2}{\sigma_{\min}(\mathbf{A}_{I_0})^2}\right).$$

By instating the definitions of γ_0 and γ_1, we have shown that

$$\mathbb{E}_k\|\widehat{\mathbf{x}}^{k+1} - \mathbf{x}^*\|^2 \leq \left(1 - \frac{\sigma_{\min}^+(\mathbf{A}_{I_1}\mathbf{P})^2}{\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2}\right) \cdot \|\widehat{\mathbf{x}}^k - \mathbf{x}^*\|^2 + \frac{\sigma_{\min}^+(\mathbf{A}_{I_1}\mathbf{P})^2}{\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2}(\gamma_0 + \gamma_1). \tag{2.20}$$

By iterating (2.20), we deduce that $\mathbb{E}\|\widehat{\mathbf{x}}^k - \mathbf{x}^*\|^2$ is upper bounded by

$$\left(1 - \frac{\sigma_{\min}^+(\mathbf{A}_{I_1}\mathbf{P})^2}{\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2}\right)^k \cdot \|\widehat{\mathbf{x}}^0 - \mathbf{x}^*\|^2 + \sum_{i=0}^{k-1}\left(1 - \frac{\sigma_{\min}^+(\mathbf{A}_{I_1}\mathbf{P})^2}{\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2}\right)^i \cdot \frac{\sigma_{\min}^+(\mathbf{A}_{I_1}\mathbf{P})^2}{\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2}(\gamma_1 + \gamma_2).$$

We conclude by bounding the geometric series by $\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2/\sigma_{\min}^+(\mathbf{A}_{I_1}\mathbf{P})^2$. □

Remark 2.3.9. A natural generalization of the SC-RK update (2.3) is to project onto the solution space $\mathbf{A}_{I_0\cup J}\mathbf{x} = \mathbf{b}_{I_0\cup J}$, where $J \subseteq [m] \setminus I_0$ is a block of row indices disjoint from I_0 with $|J| \geq 1$:

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{A}_{I_0\cup J}^\dagger(\mathbf{b}_{I_0\cup J} - \mathbf{A}_{I_0\cup J}\mathbf{x}^k). \tag{2.21}$$

Assuming that $\mathbf{A}_{I_0\cup J}$ has full row rank, Lemma 2.3.7 implies that (2.21) can be computed by the following two-step procedure (which does not require $\mathbf{x}^k$ to satisfy $\mathbf{A}_{I_0}\mathbf{x}^k = \mathbf{b}_{I_0}$):

(1) Project $\mathbf{x}^k$ onto the solution space $\mathbf{A}_{I_0}\mathbf{x} = \mathbf{b}_{I_0}$ to obtain $\mathbf{y}^k$:

$$\mathbf{y}^k = \mathbf{x}^k + \mathbf{A}_{I_0}^\dagger(\mathbf{b}_{I_0} - \mathbf{A}_{I_0}\mathbf{x}^k).$$

(2) Compute the new measurements $\boldsymbol{\beta}_J \coloneqq \mathbf{b}_J - \mathbf{A}_J\mathbf{A}_{I_0}^\dagger\mathbf{b}_{I_0} \in \mathbb{R}^{|J|}$, then project $\mathbf{y}^k$ onto the solution space $\mathbf{A}_J\mathbf{x} = \mathbf{b}_J$ whilst remaining in the solution space of $\mathbf{A}_{I_0}\mathbf{x} = \mathbf{b}_{I_0}$ for $\mathbf{x}^{k+1}$:

$$\mathbf{x}^{k+1} = \mathbf{y}^k + (\mathbf{A}_J\mathbf{P})^\dagger(\boldsymbol{\beta}_J - (\mathbf{A}_J\mathbf{P})\mathbf{y}^k).$$

In particular, by restricting to a single row $J = \{j\}$ and imposing the condition $\mathbf{A}_{I_0}\mathbf{x}^k = \mathbf{b}_{I_0}$, we recover the simplified update formula (2.4), which provides an alternative algebraic proof of Lemma 2.3.1.

By further restricting to the special case $I = \{i\}$, the two-step procedure above reduces to an update of the two-subspace Kaczmarz method of [NW13]. Thus, the SC-RK method can be seen as a partial generalization of the two-subspace Kaczmarz method, except that the subset I_0 is fixed throughout the iterations to exploit specific features of the block $\mathbf{A}_{I_0}$, and similar results concerning coherence with respect to more general subsets of equations can be obtained (see Remark 2.3.13).

Finally, while all the convergence results in this work are stated for the case $|J| = 1$, we believe that similar techniques can be extended to the case of $|J| > 1$.

2.3.3 Exploiting structure with the SC-RK method

In this section, we discuss how the SC-RK method can exploit approximately low-rank structure and geometric properties of the data matrix $\mathbf{A}$ to accelerate convergence. For simplicity, we will restrict our attention to the noiseless case. Our goal is to study the per-iteration convergence rate (i.e., with $k = 1$). First, note that the SC-RK rate (2.5) is as good as the RK rate (2.2). Indeed,

$$\sigma_{\min}^{+}(\mathbf{A}_{I_1}\mathbf{P}) = \min_{\substack{\mathbf{z}\in\mathrm{null}(\mathbf{A}_{I_0}) \\ \|\mathbf{z}\|=1}} \|\mathbf{A}_{I_1}\mathbf{z}\| = \min_{\substack{\mathbf{z}\in\mathrm{null}(\mathbf{A}_{I_0}) \\ \|\mathbf{z}\|=1}} \|\mathbf{A}\mathbf{z}\| \geq \min_{\substack{\mathbf{z}\in\mathbb{R}^n \\ \|\mathbf{z}\|=1}} \|\mathbf{A}\mathbf{z}\| = \sigma_{\min}(\mathbf{A}), \tag{2.22}$$

and $\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2 \leq \|\mathbf{P}\|^2\|\mathbf{A}_{I_1}\|_F^2 \leq \|\mathbf{A}\|_F^2$. Therefore,

$$1 - \frac{\sigma_{\min}^{+}(\mathbf{A}_{I_1}\mathbf{P})^2}{\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2} \leq 1 - \frac{\sigma_{\min}(\mathbf{A})^2}{\|\mathbf{A}\|_F^2}. \tag{2.23}$$

However, since each SC-RK iteration requires more computation (as discussed in Remark 2.1.3), we would like to understand when the SC-RK method is advantageous to RK

overall. In the following, we first examine what features of the matrices $\mathbf{A}$ and $\mathbf{A}_{I_0}$ lead to such an advantage in Section 2.3.3. Furthermore, we discuss how a good subset I_0 of rows, if not explicitly given, can actually be efficiently found when $\mathbf{A}$ has approximately low-rank structure in Section 2.3.3.

Geometry of the matrix and convergence rates

As highlighted by (2.23), either $\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2 \ll \|\mathbf{A}\|_F^2$ or $\sigma_{\min}^+(\mathbf{A}_{I_1}\mathbf{P}) \gg \sigma_{\min}(\mathbf{A})$ leads to significant per-iteration advantage of SC-RK over RK. We describe two specific motivating examples of systems with such structure before generalizing our observations in Corollary 2.3.12 below. For these examples, consider an arbitrary $(m-m_0)$-dimensional subspace $\mathcal{U}$ of $\mathbb{R}^n$. Let $\{\mathbf{u}_1,\dots,\mathbf{u}_{m_0}\}$ and $\{\mathbf{c}_{m_0+1},\mathbf{c}_{m_0+2},\dots,\mathbf{c}_n\}$ be orthonormal bases for $\mathcal{U}$ and $\mathcal{U}^\perp$, respectively, and $\varepsilon \approx 0$ be a small positive constant.

Example 2.3.10 ($\sigma_{\min}^+(\mathbf{A}_{I_1}\mathbf{P}) \gg \sigma_{\min}(\mathbf{A})$)**.** This can happen if the equations in the selected block $\mathbf{A}_{I_0}$ are almost collinear, but the system $\mathbf{A}_{I_1}\mathbf{P}$ with projected rows is well-conditioned. Let $\overline{\mathbf{u}} \coloneqq \frac{1}{m_0}\sum_{i=1}^{m_0}\mathbf{u}_i$, and $\mathbf{A} \in \mathbb{R}^{n\times n}$ be the matrix where the first m_0 rows are given by $\mathbf{a}_j \coloneqq (1-\varepsilon)\overline{\mathbf{u}} + \varepsilon\mathbf{u}_i$, $j = 1,\dots,m_0$, and the remaining rows are $\mathbf{c}_{m_0+1},\dots,\mathbf{c}_n$. Choose $I_0 = [m_0]$ so that $\mathbf{P}$ is the orthogonal projection onto $\mathcal{U}^\perp$. Then $\sigma_{\min}^+(\mathbf{A}_{I_1}\mathbf{P}) = 1 \gg \varepsilon \geq \sigma_{\min}(\mathbf{A})$. Indeed, if $\mathbf{e}_i$ is the i^{th} standard basis vector, then

$$\sigma_{\min}(\mathbf{A}) = \min_{\mathbf{x}\in\mathbb{R}^n:\|\mathbf{x}\|=1}\|\mathbf{A}^\mathsf{T}\mathbf{x}\| \leq \left\|\mathbf{A}^\mathsf{T}\frac{(\mathbf{e}_1-\mathbf{e}_2)}{\sqrt{2}}\right\| = \frac{\varepsilon}{\sqrt{2}}\|\mathbf{u}_1-\mathbf{u}_2\| = \varepsilon.$$

Furthermore, since the rows of $\mathbf{A}_{I_1}\mathbf{P}$ form an orthonormal basis for $\mathcal{U}^\perp$, for any unit vector $\mathbf{x} = \sum_{i=m_0+1}^n \alpha_i\mathbf{c}_i$ in $\mathcal{U}^\perp$, we have $\|\mathbf{A}_{I_1}\mathbf{x}\|^2 = \sum_{i=m_0+1}^n \alpha_i^2 = 1$, and hence $\sigma_{\min}^+(\mathbf{A}_{I_1}\mathbf{P}) = \min_{\mathbf{x}\in\mathcal{U}^\perp:\|\mathbf{x}\|=1}\|\mathbf{A}_{I_1}\mathbf{x}\| = 1$.

Example 2.3.11 ($\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2 \ll \|\mathbf{A}\|_F^2$)**.** This can happen if the block $\mathbf{A}_{I_0}$ is highly correlated with the remaining rows $(\mathbf{a}_j)_{j\in I_1}$. Let $\mathbf{A} \in \mathbb{R}^{n\times n}$ be the matrix where the first m_0 rows are $\mathbf{u}_1,\dots,\mathbf{u}_{m_0}$, and the remaining rows are $\mathbf{a}_j \coloneqq (1-\varepsilon)\mathbf{v}_j + \varepsilon\mathbf{c}_j$, where $\mathbf{v}_j$ is any unit vector in

$\mathcal{U}$, for $j = m_0 + 1, \ldots, n$. Choose $I_0 = [m_0]$, so that $\text{range}(\mathbf{A}_{I_0}^\mathsf{T}) = \mathcal{U}$ and $\mathbf{P}$ is the orthogonal projection onto $\mathcal{U}^\perp$. Then $\mathbf{P}\mathbf{a}_j = \varepsilon \mathbf{c}_j$ and $\|\mathbf{P}\mathbf{a}_j\| = \varepsilon \ll 1 = \|\mathbf{a}_j\|$ for all $j \in I_1$, and hence $\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2 = \sum_{j \in I_1} \|\mathbf{P}\mathbf{a}_j\|^2 \leq \varepsilon^2 \sum_{j=1}^m \|\mathbf{a}_j\|^2 = \varepsilon^2 \|\mathbf{A}\|_F^2$.

The calculations in the preceding example, together with Theorem 2.1.1 and (2.22), generalize to the following result:

Corollary 2.3.12. *Consider the same setup as Theorem 2.1.1, and assume that for some* $\delta \in [0, 1)$,

$$\frac{\|\mathbf{P}\mathbf{a}_j\|^2}{\|\mathbf{a}_j\|^2} \leq 1 - \delta^2 \quad \textit{for all } j \in I_1. \tag{2.24}$$

Then the SC-RK iterates $\mathbf{x}^k$ *converge to* $\mathbf{x}^*$ *in expectation with*

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|^2 \leq \left(1 - \frac{1}{1-\delta^2} \cdot \frac{\sigma_{\min}(\mathbf{A})^2}{\|\mathbf{A}\|_F^2}\right)^k \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|^2. \tag{2.25}$$

Remark 2.3.13. Since $\mathbf{P}$ is the orthogonal projection onto $\text{range}(\mathbf{A}_{I_0}^\mathsf{T})^\perp$, $\|\mathbf{P}\mathbf{a}_j\|^2 / \|\mathbf{a}_j\|^2 = \sin^2 \theta_j$ where θ_j is the principal angle between the subspaces $\text{range}(\mathbf{a}_j)$ and $\text{range}(\mathbf{A}_{I_0}^\mathsf{T})$. Therefore, the quantity δ in (2.24) measures the coherence between the row space of the fixed block $\mathbf{A}_{I_0}$ and each of the remaining rows $(\mathbf{a}_j)_{j \in I_1}$. A value of δ close to one means that the principal angles are uniformly small; i.e., all of the $\mathbf{a}_j$ are close to the row space of $\mathbf{A}_{I_0}$ and offer little new information by themselves. By projecting each row with $\mathbf{P}$, the shared information is effectively modded out, and thus each SC-RK iteration is able to make more meaningful progress towards the solution.

In particular, if we take $\mathbf{A}_{I_0} = \mathbf{a}_i^\mathsf{T}$ to be a single row and assume that $\|\mathbf{a}_i\| = 1 = \|\mathbf{a}_j\|$, then $\|\mathbf{P}\mathbf{a}_j\|^2 = 1 - |\mathbf{a}_i^\mathsf{T}\mathbf{a}_j|^2$, where $|\mathbf{a}_i^\mathsf{T}\mathbf{a}_j|$ is the correlation between $\mathbf{a}_i$ and $\mathbf{a}_j$. It is shown in [NW13] that the two-subspace Kaczmarz method, which iteratively projects onto the solution space associated with two random rows, significantly improves upon RK if $\mathbf{A}$ has highly correlated rows. Thus, Corollary 2.3.12 quantifies a similar phenomenon for the SC-RK method for higher dimensional subspaces.

Sampling rows to find a good subspace

Previously, we showed that bounds of the form $\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2 \ll \|\mathbf{A}\|_F^2$ using a specific choice of rows I_0 imply significant improvements in the convergence rate of the SC-RK method over randomized Kaczmarz. However, what if we are not explicitly given a good set I_0, even though there is latent low-rank structure in $\mathbf{A}$—in the sense that the matrix has $r \ll n$ dominant singular values—that can be exploited? We begin by considering a motivating hypothetical example where the row span of $\mathbf{A}_{I_0}$ is able to align perfectly with the leading right singular subspace.

Example 2.3.14 ($\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2 \ll \|\mathbf{A}\|_F^2$)**.** Let $\mathbf{A}_{(r)} \coloneqq \mathbf{U}_{(r)}\mathbf{\Sigma}_{(r)}\mathbf{V}_{(r)}^\mathsf{T}$ be the best rank-r approximation of $\mathbf{A}$ (with respect to $\|\cdot\|_F$), where $\mathbf{\Sigma}_{(r)} = \operatorname{diag}(\sigma_1(\mathbf{A}), \ldots, \sigma_r(\mathbf{A}))$ is the diagonal matrix of the top r singular values of $\mathbf{A}$, and the columns of $\mathbf{V}_{(r)} \in \mathbb{R}^{n\times r}$ and $\mathbf{U}_{(r)} \in \mathbb{R}^{m\times r}$ contain the corresponding right and left singular vectors. Suppose that the row span of $\mathbf{A}_{I_0}$ equals range($\mathbf{V}_{(r)}$). Then

$$\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2 = \|\mathbf{A}\mathbf{P}\|_F^2 = \|\mathbf{P}\mathbf{A}^\mathsf{T}\|_F^2 = \|\mathbf{A}^\mathsf{T} - \mathbf{P}^\perp\mathbf{A}^\mathsf{T}\|_F^2 = \|\mathbf{A}^\mathsf{T} - \mathbf{A}_{(r)}^\mathsf{T}\|_F^2 = \sum_{i=r+1}^{n} \sigma_i(\mathbf{A})^2.$$

If the top r singular values of $\mathbf{A}$ are much larger than the rest, then $\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2$ is much smaller than $\|\mathbf{A}\|_F^2 = \sum_{i=1}^{n} \sigma_i(\mathbf{A})^2$.

Note that in general, such a subset of rows does not exist in $\mathbf{A}$. This raises the following question: can we efficiently find a small subset I_0 of rows of $\mathbf{A}$ so that the row span of $\mathbf{A}_{I_0}$ is a good approximation of the top r-dimensional right singular subspace of $\mathbf{A}$? This is known as the problem of finding an approximate CX decomposition in the randomized numerical linear algebra literature. Algorithms have been proposed that sample rows of $\mathbf{A}$ according to their Euclidean norms [DKM06b] or their leverage scores $(\ell_j)_{j\in[m]}$ [DMM08], where ℓ_j is the squared ℓ_2 norm of the j^{th} row of $\mathbf{U}_{(r)}$, using the same notation as the example above (for more details, see [Mah11]). The following summarizes guarantees for these two sampling schemes proved in [DMM08; BMD09] and [DKM06b]:

Theorem 2.3.15 (Theorem 1 [DMM08] and [BMD09]; Theorem 4 [DKM06b]). *Suppose that c rows of $\mathbf{A}$ are independently sampled, where row j is selected with probability p_j in each trial. Let I_0 be the set of indices of the sampled rows, and $\mathbf{P}$ be the orthogonal projection onto* $\mathrm{null}(\mathbf{A}_{I_0})$.

(i) *If $p_j = \ell_j / r$ and $c = O(r \log r / \varepsilon^2)$, then with probability at least 0.9,*

$$\|\mathbf{AP}\|_F^2 \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{A}_{(r)}\|_F^2 = (1+\varepsilon) \sum_{i=r+1}^{n} \sigma_i(\mathbf{A})^2. \tag{2.26}$$

(ii) *If $p_j = \|\mathbf{a}_j\|^2 / \|\mathbf{A}\|_F^2$ and $c = O(r/\varepsilon^2)$, then with probability at least 0.9,*

$$\|\mathbf{AP}\|_F^2 \leq \|\mathbf{A} - \mathbf{A}_{(r)}\|_F^2 + \varepsilon\|\mathbf{A}\|_F^2 = (1+\varepsilon) \sum_{i=r+1}^{n} \sigma_i(\mathbf{A})^2 + \varepsilon \sum_{i=1}^{r} \sigma_i(\mathbf{A})^2. \tag{2.27}$$

Theorem 2.3.15 implies that sampling $c \approx r \log r$ rows of $\mathbf{A}$ produces a subspace that tames the leading r singular values of $\mathbf{A}$ with high probability. In practice, it has been observed that a modest oversampling factor (i.e., c is a small constant times r) usually suffices [DMM08]. The relative-error bound (2.26) is better than the additive-error bound (2.27); however it is more costly because it requires the estimation of the leverage scores (see, e.g., [Dri+12; HMT11]). By combining Theorem 2.3.15 and (2.22) with the SC-RK convergence result (Theorem 2.1.1), we deduce the following:

Corollary 2.3.16. *Suppose that $I_0 \subset [m]$ contains $m_0 = O(r \log r / \varepsilon^2)$ rows of $\mathbf{A}$, randomly sampled according to the leverage scores of $\mathbf{A}$ relative to its best rank-r approximation as described in Theorem 2.3.15, and partition $\mathbf{A}$ into blocks $\mathbf{A}_{I_0}$ and $\mathbf{A}_{I_1}$ with $I_1 = [m] \setminus I_0$. Then with probability at least 0.9 over the sampling of I_0, the SC-RK iterates $\mathbf{x}^k$ satisfy*

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|^2 \leq \left(1 - \frac{\sigma_n(\mathbf{A})^2}{(1+\varepsilon)\sum_{i=r+1}^{n} \sigma_i(\mathbf{A})^2}\right)^k \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|^2.$$

Thus, if the rank of $\mathbf{A}$ is effectively less than r (in the sense $\sum_{i=r+1}^{n} \sigma_i(\mathbf{A})^2 \ll \|\mathbf{A}\|_F^2$), then the SC-RK method with iterates constrained to the solution space corresponding to $m_0 = O(r \log r)$ randomly sampled rows significantly improves upon RK. Note that similar results are known for the sketch-and-project method [DR24]. Moreover, the effective rank of a large-scale matrix $\mathbf{A}$ can be estimated in a data-driven manner by sketching [MN24].

2.3.4 SC-RK on random data and dimension reduction

Previously, we discussed how the SC-RK method accelerates the iterative solver when the matrix $\mathbf{A}$ has approximately low-rank structure. In this section, we consider a somewhat complementary setting to study the effect of the subspace constraint when $\mathbf{A}$ *is unstructured and homogeneous*: namely, when $\mathbf{A}$ is drawn from a class of generic random matrices (precisely defined below) whose rows behave like independent standard Gaussian vectors. Such a matrix is typically well-conditioned as long as its aspect ratio m/n is large enough, and hence the corresponding linear system is easily solved using randomized Kaczmarz. However, for almost-square systems with an aspect ratio close to one, the convergence rate is far from optimal.

First, we review some definitions from probability theory (we refer to [Ver18] for more details). If $\mathbf{a} \in \mathbb{R}^n$ is a random vector, we say that $\mathbf{a}$ is *mean-zero* if $\mathbb{E}[\mathbf{a}] = \mathbf{0}$, and $\mathbf{a}$ is *isotropic* if $\mathbb{E}\left[\mathbf{a}\mathbf{a}^\mathsf{T}\right] = \mathbf{I}$. We say that a scalar random variable X is K*-subgaussian* if its subgaussian norm $\|X\|_{\psi_2} \coloneqq \inf_{t>0}\{\mathbb{E}[\exp(X^2/t^2) \leq 2]\}$ is bounded by $K > 0$; informally, this means that X concentrates around its mean with a light, exponentially decaying tail. Furthermore, a random vector $\mathbf{a}$ is K-subgaussian if all of its one-dimensional marginals are K-subgaussian: $\|\mathbf{a}\|_{\psi_2} \coloneqq \sup_{\mathbf{z}\in\mathbb{R}^n:\|\mathbf{z}\|=1} \|\langle \mathbf{z}, \mathbf{a}\rangle\|_{\psi_2} \leq K$.

As before, we will continue to assume that $\mathbf{A}$ has full rank (almost surely). For our model, we will allow $\mathbf{A}_{I_0} \in \mathbb{R}^{m_0 \times n}$ to be arbitrary, and we assume that $\mathbf{A}_{I_1} \in \mathbb{R}^{(m-m_0)\times n}$ is a random matrix that satisfies the following:

A1. The rows of $\mathbf{A}_{I_1}$ are independent, mean-zero, isotropic, and K-subgaussian random vectors.

The canonical example for our model is a standard Gaussian matrix $\mathbf{A}_{I_1}$ whose entries are independent standard normal random variables. In this special case, exact computations are often possible (e.g., if $\boldsymbol{A}_{I_1}$ is a standard Gaussian matrix, then we can choose $K = \sqrt{8/3} \approx 1.633$). More generically, Assumption A1 models unstructured matrices containing homogeneous data (that is centered and isotropic) with light tails.

Our main result in this section shows that for such matrices, the subspace constraint imposed by $\mathbf{P}$ acts as a form of dimension reduction, typically resulting in a near-optimal convergence rate of approximately $1 - 1/(n - m_0)$ as long as the "effective aspect ratio" $(m - m_0)/(n - m_0)$, which may be much larger than m/n, is large enough.

Theorem 2.3.17. *Suppose that the rows of $\mathbf{A}$ are partitioned into two blocks $\mathbf{A}_{I_0}$ and $\mathbf{A}_{I_1}$ of sizes m_0 and $m - m_0$, respectively, where $\mathbf{A}_{I_0}$ is arbitrary and $\mathbf{A}_{I_1}$ is a random matrix that satisfies Assumption A1. There exist constants $c, R > 0$ (only depending on K) such that if*

$$r := \frac{m - m_0}{n - m_0} \geq R,$$

then for any $\varepsilon \in (0, 1)$, with probability at least $1 - 3\exp\left\{-c\varepsilon^2\left(1 - \sqrt{\frac{R}{r}}\right)(m - m_0)\right\}$ over the randomness in $\mathbf{A}_{I_1}$, the SC-RK iterates $\mathbf{x}^k$ satisfy

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|^2 \leq \left(1 - \frac{(1-\varepsilon)^2}{1+\varepsilon}\left(1 - \sqrt{\frac{R}{r}}\right)^2 \cdot \frac{1}{n - m_0}\right)^k \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|^2. \tag{2.28}$$

In the special case where $\mathbf{A}_{I_1}$ is standard Gaussian, this result holds with $R = 1$.

The values of R and c depend on the precise distributional properties of the random matrix, and are, importantly, independent of m and n. Note that for tall, large-scale systems with $m, n \gg 1$ and $r \gg 1$, the requirement $r \geq R$ is not difficult to meet, and taking $\varepsilon \approx 0$

shows that the convergence rate is approximately $1-1/(n-m_0)$ with a probability guarantee that is exponentially close to one.

Proof of Theorem 2.3.17

To study the typical convergence rate with a random matrix, we will obtain tail bounds for $\sigma_{\min}^{+}(\mathbf{A}_{I_1}\mathbf{P})$ and $\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2$. The first lemma is deterministic, and shows that instead of studying the non-zero singular values of the $(m-m_0)\times n$ matrix $\mathbf{A}_{I_1}\mathbf{P}$, we can study the singular values of a thinner $(m-m_0)\times(n-m_0)$ matrix after rotating.

Lemma 2.3.18. *Let $\mathbf{X}\in\mathbb{R}^{m\times n}$ be a matrix, and $\mathbf{P}\in\mathbb{R}^{n\times n}$ be an orthogonal projection onto a d-dimensional subspace of $\mathbb{R}^n$. Suppose that the columns of $\bar{\mathbf{Q}}\in\mathbb{R}^{n\times d}$ form an orthonormal basis for* range$(\mathbf{P})$. *Then the non-zero singular values of $\mathbf{XP}\in\mathbb{R}^{m\times n}$ and $\mathbf{X}\bar{\mathbf{Q}}\in\mathbb{R}^{m\times d}$ are the same.*

Proof. Note that $\mathbf{P}=\bar{\mathbf{Q}}\bar{\mathbf{Q}}^{\mathsf{T}}$. Let $\mathbf{X}\bar{\mathbf{Q}}=\mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathsf{T}}$ be a compact singular value decomposition of $\mathbf{X}\bar{\mathbf{Q}}$, which means that $\mathbf{\Sigma}$ is a square diagonal matrix containing the rank$(\mathbf{X})$ non-zero singular values of $\mathbf{X}\bar{\mathbf{Q}}$, and $\mathbf{U},\mathbf{V}$ are rectangular matrices with orthonormal columns. Observe that $\mathbf{XP}=\mathbf{U}\mathbf{\Sigma}(\mathbf{V}')^{\mathsf{T}}$ where $\mathbf{V}'\coloneqq\bar{\mathbf{Q}}\mathbf{V}$ also has orthonormal columns. This allows us to conclude the desired result since the non-zero singular values of $\mathbf{XP}$ are presented in the same matrix $\mathbf{\Sigma}$. □

Remark 2.3.19 (Gaussian distribution)**.** Suppose that the rows $\mathbf{a}_j$ of $\mathbf{A}_{I_1}$ are independent standard Gaussian vectors in $\mathbb{R}^n$. Note that the matrix $\bar{\mathbf{Q}}$ can be represented as $\bar{\mathbf{Q}}=\mathbf{Q}\mathbf{R}^{\mathsf{T}}$ where $\mathbf{Q}\in\mathbb{R}^{n\times n}$ is an orthogonal matrix and $\mathbf{R}:\mathbb{R}^n\to\mathbb{R}^{n-m_0}$ is the restriction operator $\mathbf{R}=\begin{pmatrix}\mathbf{I}_{n-m_0} & \mathbf{0}\end{pmatrix}$. The distribution of each row $\mathbf{a}_j$ is rotation invariant and so $\mathbf{A}_{I_1}\overset{\mathrm{d}}{=}\mathbf{A}_{I_1}\mathbf{Q}$. Hence, by Lemma 2.3.18,

$$\big(\sigma_1(\mathbf{A}_{I_1}\mathbf{P}),\,\sigma_2(\mathbf{A}_{I_1}\mathbf{P}),\,\ldots,\,\sigma_{n-m_0}(\mathbf{A}_{I_1}\mathbf{P})\big)\overset{\mathrm{d}}{=}\big(\sigma_1(\mathbf{A}_{I_1}\mathbf{R}^{\mathsf{T}}),\,\sigma_2(\mathbf{A}_{I_1}\mathbf{R}^{\mathsf{T}}),\,\ldots,\,\sigma_{n-m_0}(\mathbf{A}_{I_1}\mathbf{R}^{\mathsf{T}})\big).$$

Note that the rows of $\mathbf{A}_{I_1}\mathbf{R}^{\mathsf{T}}\in\mathbb{R}^{(m-m_0)\times(n-m_0)}$ are independent Gaussian vectors in $\mathbb{R}^{n-m_0}$.

The upcoming probabilistic results show that the same intuition extends to more generic matrices with Gaussian-like tails. First, the following result shows that the smallest non-zero singular value of $\mathbf{A}_{I_1}\mathbf{P}$ can be lower bounded with very high probability.

Lemma 2.3.20. *Let $\mathbf{P}$ be an orthogonal projection onto a fixed $(n-m_0)$-dimensional subspace. Suppose that the random matrix $\mathbf{A}_{I_1} \in \mathbb{R}^{(m-m_0)\times n}$ satisfies Assumption A1. Then there exists an absolute constant $C > 0$ such that for all $s > 0$, with probability at least $1 - 2e^{-s^2(m-m_0)}$, the smallest and largest non-zero singular values of $\mathbf{A}_{I_1}\mathbf{P}$ satisfy*

$$\sigma^+_{\min}(\mathbf{A}_{I_1}\mathbf{P}) \geq \sqrt{m-m_0} - CK^2(\sqrt{n-m_0} + s\sqrt{m-m_0}) \quad \text{and}$$
$$\sigma_{\max}(\mathbf{A}_{I_1}\mathbf{P}) \leq \sqrt{m-m_0} + CK^2(\sqrt{n-m_0} + s\sqrt{m-m_0}).$$

In the case where $\mathbf{A}_{I_1}$ is Gaussian, the inequalities hold with CK^2 replaced by one.

Proof. Let $\bar{\mathbf{Q}} \in \mathbb{R}^{(m-m_0)\times(n-m_0)}$ be a matrix whose columns form an orthonormal basis for range$(\mathbf{P})$. By Lemma 2.3.18, the smallest and largest non-zero singular values of $\mathbf{A}_{I_1}\mathbf{P} \in \mathbb{R}^{(m-m_0)\times n}$ and $\mathbf{B} := \mathbf{A}_{I_1}\bar{\mathbf{Q}} \in \mathbb{R}^{(m-m_0)\times(n-m_0)}$ are equal. It can be directly checked that the rows of $\mathbf{B}$ are also independent, mean-zero, isotropic, K-subgaussian random vectors in $\mathbb{R}^{n-m_0}$. Hence, using a standard tail bound for the extremal singular values $\sigma_{\min}(\mathbf{B})$ and $\sigma_{\max}(\mathbf{B})$ of the random matrix $\mathbf{B}$ (see [Ver18, Theorem 4.6.1]) implies the claimed inequalities. In the Gaussian case, the precise constants can be computed using Gaussian concentration tools (see [Ver18, Corollary 7.3.3, Exercise 7.3.4]). □

Remark 2.3.21. The minimum restricted singular value of random matrices has also been studied in the context of universality laws for randomized dimension reduction in [OT18]. If the entries of $\mathbf{A}_{I_1}$ are independent random variables satisfying some mild regularity conditions, then [OT18, Theorem II] establishes that $\sigma^+_{\min}(\mathbf{A}_{I_1}\mathbf{P}) \approx \sqrt{m-m_0} - C\sqrt{n-m_0}$ with high probability since range$(\mathbf{P})$ is a $(n-m_0)$-dimensional subspace in $\mathbb{R}^n$. Thus, $\sigma^+_{\min}(\mathbf{A}_{I_1}\mathbf{P})$ is of comparable order for a wide class of distributions. However, the maximum restricted singular value is not necessarily universal.

Next, our goal is to obtain tail bounds for $\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2 = \sum_{j\in I_1}\|\mathbf{P}\mathbf{a}_j\|^2$. In the setting where the rows $\mathbf{a}_j$ of $\mathbf{A}_{I_1}$ are mean-zero, K-subgaussian random vectors, it is proved in [Jin+19] that the Euclidean norms $\|\mathbf{a}_j\|$ are $O(K\sqrt{n})$-subgaussian. The next lemma states that the norms of the projected vectors $\|\mathbf{P}\mathbf{a}_j\|$ are $O(K\sqrt{n-m_0})$-subgaussian.

Lemma 2.3.22. *Let* $\mathbf{a}$ *be a mean-zero,* K*-subgaussian random vector in* $\mathbb{R}^n$*, and* $\mathbf{P}$ *be an orthogonal projection onto a fixed* d*-dimensional subspace. Then the subgaussian norm of* $\|\mathbf{P}\mathbf{a}\|$ *is bounded by* $CK\sqrt{d}$ *for some absolute constant* $C > 0$*.*

The proof uses the following geometric observation about unit spheres of subspaces, which we record for later reference. We say that $\mathcal{N}$ is an *ε-net* of a set $S \subseteq \mathbb{R}^n$ if $\mathcal{N} \subseteq S$ and every point in S is within distance ε of some point in $\mathcal{N}$. It is known that there exists an ε-net of the d-dimensional unit sphere $\mathbb{S}^{d-1} := \{\mathbf{x} \in \mathbb{R}^d : \|\mathbf{x}\| = 1\}$ with cardinality bounded by $(1 + 2/\varepsilon)^d$ for any d (see, e.g., [Ver18, Corollary 4.2.13]). Thus, if $\mathcal{U}$ is a d-dimensional subspace of $\mathbb{R}^n$, then by identifying $\mathcal{U} \cong \mathbb{R}^d$ (using the fact that rotations are isometries) and obtaining a net of $\mathbb{S}^{d-1}$, we deduce the following:

Lemma 2.3.23. *Let* $\mathcal{U}$ *be a* d*-dimensional subspace of* $\mathbb{R}^n$*. Then for any* $\varepsilon > 0$*, there exists an* ε*-net* $\mathcal{N}$ *of* $\mathcal{U} \cap \mathbb{S}^{n-1}$ *with cardinality* $|\mathcal{N}| \le (1 + 2/\varepsilon)^d$*.*

Proof of Lemma 2.3.22. The proof is similar to the proof of [Jin+19, Lemma 1]; we provide it for completeness. First, it can be checked that $\mathbf{P}\mathbf{a}$ is also a mean-zero K-subgaussian random vector. Next, by Lemma 2.3.23, we can fix a 1/2-net $\mathcal{N}$ of $\mathrm{range}(\mathbf{P}) \cap \mathbb{S}^{n-1}$ with cardinality $|\mathcal{N}| \le 5^{n-m_0}$. By using these observations, we will show that

$$\mathbb{P}\left(\|\mathbf{P}\mathbf{a}\| \ge t\right) \le 2e^{-t^2/(10K^2(n-m_0))} \quad \text{for all } t \ge 0. \tag{2.29}$$

Indeed, for any realization of $\mathbf{P}\mathbf{a}$, there exists $\mathbf{v} \in \mathcal{N}$ such that $\left\|\frac{\mathbf{P}\mathbf{a}}{\|\mathbf{P}\mathbf{a}\|} - \mathbf{v}\right\| \le 1/2$, and we can write

$$\|\mathbf{P}\mathbf{a}\| = \langle \mathbf{v}, \mathbf{P}\mathbf{a}\rangle + \left\langle \frac{\mathbf{P}\mathbf{a}}{\|\mathbf{P}\mathbf{a}\|} - \mathbf{v}, \mathbf{P}\mathbf{a}\right\rangle \le \langle \mathbf{v}, \mathbf{P}\mathbf{a}\rangle + \left\|\frac{\mathbf{P}\mathbf{a}}{\|\mathbf{P}\mathbf{a}\|} - \mathbf{v}\right\| \cdot \|\mathbf{P}\mathbf{a}\| \le \langle \mathbf{v}, \mathbf{P}\mathbf{a}\rangle + \frac{\|\mathbf{P}\mathbf{a}\|}{2}$$

to deduce that $\|\mathbf{Pa}\| \le 2\langle \mathbf{v}, \mathbf{Pa}\rangle$. Therefore, because $\langle \mathbf{v}, \mathbf{Pa}\rangle$ is a mean-zero, K-subgaussian random variable, a union bound implies that for all $t \ge 0$,

$$\mathbb{P}\left(\|\mathbf{Pa}\| \ge t\right) \le \mathbb{P}\left(\exists \mathbf{v} \in \mathcal{N} : \langle \mathbf{v}, \mathbf{Pa}\rangle \ge \frac{t}{2}\right) \le 5^{n-m_0} \cdot e^{-t^2/(4K^2)}.$$

We claim that this implies (2.29). If $t^2 \le 4\log(5)K^2(n-m_0)$, then (2.29) trivially holds. Otherwise, if $t^2 = 4\log(5)K^2(n-m_0) + s$ for $s > 0$, then

$$5^{n-m_0} \cdot e^{-t^2/(4K^2)} = e^{-s/(4K^2)} \le e^{-s/(6\log(5)K^2(n-m_0))} \le 2e^{-t^2/(10K^2(n-m_0))}.$$

Thus, the tail bound (2.29) holds for all $t \ge 0$, which implies that (by, e.g., [Ver18, Proposition 2.5.2]), $\|\mathbf{Pa}\|$ has subgaussian norm bounded by $CK\sqrt{n-m_0}$ for some absolute constant $C > 0$. □

Lemma 2.3.24. *Consider the same setup as Lemma 2.3.20. Then there exists an absolute constant $c > 0$ such that for all $\varepsilon > 0$, with probability at least $1 - e^{-c\min\{\varepsilon,\varepsilon^2\}(m-m_0)/K^4}$,*

$$\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2 \le (1+\varepsilon)(m-m_0)(n-m_0). \tag{2.30}$$

Proof. Since $\mathbf{P}$ is an orthogonal projection onto an $(n-m_0)$-dimensional subspace and $\mathbf{a}_j$ is isotropic, using the cyclic property of trace implies that for all $j \in I_1$,

$$\mathbb{E}\|\mathbf{Pa}_j\|^2 = \mathbb{E}\left[\mathrm{tr}(\mathbf{a}_j^\mathsf{T}\mathbf{Pa}_j)\right] = \mathrm{tr}(\mathbb{E}\left[\mathbf{a}_j\mathbf{a}_j^\mathsf{T}\right]\mathbf{P}) = \mathrm{tr}(\mathbf{P}) = n - m_0.$$

Therefore,

$$\mathbb{E}\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2 = \mathbb{E}\left[\sum_{j\in I_1}\|\mathbf{Pa}_j\|^2\right] = |I_1| \cdot \mathbb{E}\|\mathbf{Pa}_1\|^2 = (m-m_0)(n-m_0).$$

Now, the random variables $\|\mathbf{Pa}_j\|$ are independent and, by Lemma 2.3.22, $O(K\sqrt{n-m_0})$-subgaussian. Hence, by centering and Bernstein's inequality [Ver18, Theorem 2.8.1], there

exists an absolute constant $c > 0$ such that

$$\mathbb{P}\left(\|\mathbf{A}_{I_1}\mathbf{P}\|_F^2 - (m-m_0)(n-m_0) \geq t\right) \leq \exp\left(-\frac{c}{K^4}\min\left\{\frac{t^2}{(m-m_0)(n-m_0)^2}, \frac{t}{n-m_0}\right\}\right)$$

for all $t \geq 0$. By choosing $t = \varepsilon(m-m_0)(n-m_0)$, we obtain (2.30). □

The main result of this section now easily follows from the tail bounds for $\mathbf{A}_{I_1}\mathbf{P}$.

Proof of Theorem 2.3.17. Suppose that the random matrix $\mathbf{A}_{I_1}\mathbf{P}$ satisfies the events in Lemma 2.3.20, using $s = \varepsilon\left(1 - \frac{CK^2}{\sqrt{r}}\right)\frac{1}{CK^2}$ and relabelling C^2K^4 by R, and Lemma 2.3.24. If this occurs, which holds with the claimed probability after simplifying, the convergence result (2.28) then directly follows from the SC-RK convergence result, Theorem 2.1.1. □

2.4 Analysis of the QuantileSC-RK algorithm

In this section, we consider the QuantileSC-RK method for solving corrupted linear systems (Algorithm 2.2). Recall that in our model, we are given a corrupted measurement vector $\widetilde{\mathbf{b}} \coloneqq \mathbf{b} + \mathbf{b}_{\mathcal{C}}$, where $\mathbf{b}_{\mathcal{C}}$ is a sparse vector of arbitrary corruptions supported on $\mathcal{C} \subseteq [m]$, as well as a corruption-free subset $I_0 \subseteq [m]$ of size m_0 such that $(\mathbf{b}_{\mathcal{C}})_{I_0} = \mathbf{0}$. Our goal is to reconstruct the solution $\mathbf{x}^*$ of the linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$.

Our main result in this section is Theorem 2.4.1, which shows that the QuantileSC-RK method is able to converge robustly and efficiently when $\mathbf{A}$ is an unstructured random matrix as long as the effective aspect ratio $(m-m_0)/(n-m_0)$ is tall enough and the proportion of corrupted measurements $|\mathcal{C}|/(m-m_0)$ is not too large. Specifically, we consider the class of "Gaussian-like" random matrices previously considered in Section 2.3.4, and assume that $\mathbf{A}_{I_1}$ is a random matrix that satisfies Assumption A1 in addition to the following continuity assumption:

A2. Each row of $\mathbf{A}_{I_1}$ either has a log-concave distribution[2] or has independent entries with bounded probability densities.[3]

The class of log-concave distributions is a generalization of the standard Gaussian distribution that allows for some dependence between the entries of a random vector; for example, the uniform distribution over any convex body in $\mathbb{R}^n$ is log-concave. For more details and examples, we refer to [SW14].

Assumption A2 is essentially needed for technical reasons for our proof of Theorem 2.4.1. Empirically, convergence is observed even if $\mathbf{A}$ has random discrete entries [Had+22], or if $\mathbf{A}$ is a structured, sparse matrix in an imaging problem (see Section 2.5.6). The assumption of having independent coordinates with bounded densities in each row was previously considered in [Had+22], and we extend the model by allowing for log-concave distributions.

We can now state our main result:

Theorem 2.4.1. *Suppose that the rows of $\mathbf{A}$ are partitioned into blocks $\mathbf{A}_{I_0}$ and $\mathbf{A}_{I_1}$ of sizes m_0 and $m - m_0$, respectively, where $\mathbf{A}_{I_0}$ is arbitrary and $\mathbf{A}_{I_1}$ is a random matrix that satisfies Assumptions A1 and A2. Suppose that the corrupted measurement vector $\widetilde{\mathbf{b}} = \mathbf{b} + \mathbf{b}_{\mathcal{C}}$ is observed, $(\mathbf{b}_{\mathcal{C}})_{I_0} = \mathbf{0}$, and a quantile parameter $q \in (0, 1)$ is fixed. There exist constants $\beta_0 \in (0, 1)$ and $R \geq 1$ (only depending on q and K) such that if*

$$\frac{m - m_0}{n - m_0} \geq R \quad \text{and} \quad \beta \coloneqq \frac{|\mathcal{C}|}{m - m_0} \leq \beta_0, \tag{2.31}$$

then for some constants $c_1, c_2 > 0$ (only depending on q, β, and K), with probability at least $1 - 6e^{-c_1(m-m_0)}$ over the randomness in $\mathbf{A}_{I_1}$, the QuantileSC-RK iterates $\mathbf{x}^k$ from Algorithm 2.2 converge to the solution $\mathbf{x}^*$ in expectation with

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|^2 \leq \left(1 - \frac{c_2}{n - m_0}\right)^k \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|^2. \tag{2.32}$$

[2] A log-concave distribution in $\mathbb{R}^n$ has a probability density f that satisfies $f(\lambda \mathbf{x} + (1-\lambda)\mathbf{y}) \geq f(\mathbf{x})^\lambda f(\mathbf{y})^{1-\lambda}$ for all $\lambda \in [0, 1]$ and $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$.

[3] By scaling, we may assume without loss of generality that the densities are bounded by one.

As mentioned previously, the values of the constants c_1 and c_2 are dominated in large-scale systems with $m, n \gg 1$, and the requirement $(m - m_0)/(n - m_0) \geq R$ is not difficult to meet if the system is tall and there is enough external knowledge (i.e., $m \gg n$, $m_0 \gg 1$). In addition, we believe that it should be possible to obtain sharper theoretical estimates for β_0 and R.

The strategy to prove Theorem 2.4.1 is to combine a deterministic sufficient condition for the convergence of QuantileSC-RK, adapting a result for QuantileRK proved by [Ste23], with probabilistic results for the spectra of the projected random matrix $\mathbf{A}_{I_1}\mathbf{P}$. First, we define some spectral quantities that will be needed. For $\alpha \in (0, 1]$, define

$$\sigma^+_{\alpha,\min}(\mathbf{A}_{I_1}\mathbf{P}) \coloneqq \inf_{\substack{T \subseteq I_1 \\ |T| = \alpha(m-m_0)}} \sigma^+_{\min}((\mathbf{A}_{I_1}\mathbf{P})_T). \tag{2.33}$$

For simplicity, we will assume throughout that $\alpha(m - m_0)$ is an integer. This quantity, which represents the uniform minimum singular value over all row submatrices of $\mathbf{A}_{I_1}\mathbf{P}$ with $\alpha(m - m_0)$ rows, has appeared in previous analyses of the QuantileRK algorithm [Had+22; Ste23], and quantifies whether there are any poorly-conditioned row submatrices that are particularly susceptible to corruptions. Similarly, define

$$Z_\alpha \coloneqq \sup_{\substack{T \subseteq I_1 \\ |T| = \alpha(m-m_0)}} \|(\mathbf{A}_{I_1}\mathbf{P})_T\|_F^2. \tag{2.34}$$

Together, (2.33) and (2.34) provide a uniform upper bound for the scaled condition numbers of all row submatrices of $\mathbf{A}_{I_1}\mathbf{P}$ possibly containing the uncorrupted, admissible rows (i.e., whose residuals are smaller than the quantile threshold). This is the key step which guarantees that the expected improvement from moving in an uncorrupted direction offsets the expected deterioration caused by a corruption.

2.4.1 A deterministic condition for convergence

The following lemma provides a deterministic condition that guarantees the convergence of the QuantileSC-RK algorithm for any arbitrary sparse corruption vector $\mathbf{b}_{\mathcal{C}}$, which may be of independent interest. This is adapted from a similar condition for convergence of the QuantileRK algorithm that was proved by Steinerberger [Ste23]. It is a very strong, albeit restrictive, deterministic result that holds for all arbitrary corruptions that affect at most a β proportion of the rows in I_1, and depends on the spectral properties of the projected matrix $\mathbf{A}_{I_1}\mathbf{P}$.

Lemma 2.4.2. *Recall that $\mathcal{C} \subseteq I_1$ are the indices of the corrupted measurements, and $\beta = |\mathcal{C}|/(m - m_0)$. Suppose that $\beta < q < 1 - \beta$. Define*

$$C_{q,\beta} \coloneqq \frac{1}{Z_{q-\beta}} \left\{ \sigma^{+}_{q-\beta,\min}(\mathbf{A}_{I_1}\mathbf{P})^2 - \sigma_{\max}(\mathbf{A}_{I_1}\mathbf{P})^2 \left(\frac{\beta}{1-q} + 2\sqrt{\frac{\beta}{1-q}} \right) \right\}, \tag{2.35}$$

where $\sigma^{+}_{q-\beta,\min}(\mathbf{A}_{I_1}\mathbf{P})$ is defined in (2.33), and $Z_{q-\beta}$ in (2.34). If $C_{q,\beta} > 0$, or equivalently,

$$\frac{\sigma^{+}_{q-\beta,\min}(\mathbf{A}_{I_1}\mathbf{P})^2}{\sigma_{\max}(\mathbf{A}_{I_1}\mathbf{P})^2} > \frac{\beta}{1-q} + 2\sqrt{\frac{\beta}{1-q}}, \tag{2.36}$$

then the QuantileSC-RK iterates $\mathbf{x}^k$ from Algorithm 2.2 converge to the solution $\mathbf{x}^*$ in expectation with

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|^2 \leq (1 - C_{q,\beta})^k \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|^2. \tag{2.37}$$

The proof of Lemma 2.4.2 follows the same strategy as [Ste23] with a minor improvement in the condition (2.36) for convergence. For completeness, we provide the full details.

Proof. Consider the iterate $\mathbf{x}^k$. Recall that $J = J(q,k)$ is the set of indices of the admissible rows that satisfy $|b_j - \mathbf{a}_j^{\mathsf{T}}\mathbf{x}^k| \leq \gamma_q = q\text{-quantile}\left\{|b_j - \mathbf{a}_j^{\mathsf{T}}\mathbf{x}^k| : j \in I_1\right\}$, with $|J| = q(m - m_0)$. Let $S \coloneqq \mathcal{C} \cap J$ be the indices of the corrupted yet admissible rows, which satisfies $0 \leq |S| \leq \beta(m - m_0)$. Recall that the row j is sampled from J with probability equal to $\|\mathbf{P}\mathbf{a}_j\|^2/Z_J$,

where $Z_J \coloneqq \sum_{j\in J}\|\mathbf{Pa}_j\|^2$ is the normalizing constant. Conditional on all the choices up to the k^{th} iteration, we have

$$\mathbb{E}_k\|\mathbf{x}^{k+1}-\mathbf{x}^*\|^2 = \left[1-\frac{\sum_{j\in S}\|\mathbf{Pa}_j\|^2}{Z_J}\right]\mathbb{E}_{\{j\in J\setminus S\}}\|\mathbf{x}^{k+1}-\mathbf{x}^*\|^2 \tag{2.38}$$

$$+\frac{\sum_{j\in S}\|\mathbf{Pa}_j\|^2}{Z_J}\mathbb{E}_{\{j\in S\}}\|\mathbf{x}^{k+1}-\mathbf{x}^*\|^2, \tag{2.39}$$

where $\mathbb{E}_{\{j\in S\}}$ denotes the expectation further conditional on $j\in S$, and similarly for $\mathbb{E}_{\{j\in J\setminus S\}}$. We proceed to estimate the two summands (2.38) and (2.39) individually.

Step 1: Lower bounding the improvement from selecting an uncorrupted equation. Conditional on sampling an admissible, uncorrupted equation $j\in J\setminus S$, the improvement is given by one step of the SC-RK method applied to the row submatrix $(\mathbf{A}_{I_1}\mathbf{P})_{J\setminus S}$. Thus, by Theorem 2.1.1,

$$\mathbb{E}_{\{j\in J\setminus S\}}\|\mathbf{x}^{k+1}-\mathbf{x}^*\|^2 \le \left(1-\frac{\sigma^+_{\min}((\mathbf{A}_{I_1}\mathbf{P})_{J\setminus S})^2}{\sum_{j\in J\setminus S}\|\mathbf{Pa}_j\|^2}\right)\cdot\|\mathbf{x}^k-\mathbf{x}^*\|^2.$$

Since $|J\setminus S|\ge(q-\beta)(m-m_0)$, by using the definition of $\sigma^+_{q-\beta,\min}(\mathbf{A}_{I_1}\mathbf{P})$ in (2.33) together with the fact that adding rows to a matrix can only increase its minimum singular value, we obtain the following upper bound for the first term (2.38):

$$\left(1-\frac{\sum_{j\in S}\|\mathbf{Pa}_j\|^2}{Z_J}\right)\cdot\|\mathbf{x}^k-\mathbf{x}^*\|^2-\frac{\sigma^+_{q-\beta,\min}(\mathbf{A}_{I_1}\mathbf{P})^2}{Z_J}\cdot\|\mathbf{x}^k-\mathbf{x}^*\|^2. \tag{2.40}$$

Step 2: Upper bounding the deterioration from selecting a corrupted equation. The second term (2.39) represents the possible deterioration from selecting a corrupted yet admissible row that may take $\mathbf{x}^k$ further away from the solution $\mathbf{x}^*$. By expanding the

square, it is equal to

$$\begin{aligned}
&\frac{\sum_{j\in S}\|\mathbf{P}\mathbf{a}_j\|^2}{Z_J}\sum_{j\in S}\frac{\|\mathbf{P}\mathbf{a}_j\|^2}{\sum_{i\in S}\|\mathbf{P}\mathbf{a}_i\|^2}\left\|\mathbf{x}^k-\mathbf{x}^*+\frac{b_j-\mathbf{a}_j^\mathsf{T}\mathbf{x}^k}{\|\mathbf{P}\mathbf{a}_j\|^2}\mathbf{P}\mathbf{a}_j\right\|^2\\
&=\frac{\sum_{j\in S}\|\mathbf{P}\mathbf{a}_j\|^2}{Z_J}\|\mathbf{x}^k-\mathbf{x}^*\|^2+\frac{1}{Z_J}\sum_{j\in S}|b_j-\mathbf{a}_j^\mathsf{T}\mathbf{x}^k|^2+\frac{2}{Z_J}\sum_{j\in S}(b_j-\mathbf{a}_j^\mathsf{T}\mathbf{x}^k)(\mathbf{P}\mathbf{a}_j)^\mathsf{T}(\mathbf{x}^k-\mathbf{x}^*)\\
&\le\frac{\sum_{j\in S}\|\mathbf{P}\mathbf{a}_j\|^2}{Z_J}\|\mathbf{x}^k-\mathbf{x}^*\|^2+\frac{1}{Z_J}|S|\gamma_q^2+\frac{2}{Z_J}\gamma_q\sqrt{|S|}\,\|(\mathbf{A}_{I_1}\mathbf{P})_C(\mathbf{x}^k-\mathbf{x}^*)\|, \qquad (2.41)
\end{aligned}$$

where the definition of the quantile γ_q and Cauchy-Schwarz is used for the inequality.

Step 3: Bounding the q-quantile of a sample. Since any uncorrupted row $\mathbf{a}_j$ with $j\in I_1\setminus\mathcal{C}$ satisfies $\mathbf{a}_j^\mathsf{T}\mathbf{x}^*=b_j$, we have

$$b_j-\mathbf{a}_j^\mathsf{T}\mathbf{x}^k=\mathbf{a}_j^\mathsf{T}(\mathbf{x}^*-\mathbf{x}^k)=(\mathbf{P}\mathbf{a}_j)^\mathsf{T}(\mathbf{x}^*-\mathbf{x}^k),$$

recalling that $\mathbf{x}^*-\mathbf{x}^k\in\mathrm{null}(\mathbf{A}_{I_0})$. Since there are at least $(1-q)(m-m_0)-(\beta(m-m_0)-|S|)=(1-q-\beta)(m-m_0)+|S|$ uncorrupted equations in I_1 whose residual is larger than γ_q, we have

$$\begin{aligned}
((1-q-\beta)(m-m_0)+|S|)\gamma_q^2&\le\sum_{j\in I_1\setminus\mathcal{C}}|b_j-\mathbf{a}_j^\mathsf{T}\mathbf{x}^k|^2\le\sum_{j\in I_1}\left|(\mathbf{P}\mathbf{a}_j)^\mathsf{T}(\mathbf{x}^*-\mathbf{x}^k)\right|^2\\
&=\|\mathbf{A}_{I_1}\mathbf{P}(\mathbf{x}^k-\mathbf{x}^*)\|^2\le\sigma_{\max}(\mathbf{A}_{I_1}\mathbf{P})^2\cdot\|\mathbf{x}^k-\mathbf{x}^*\|^2.
\end{aligned}$$

Therefore, the q-quantile of the sizes of the residuals can be bounded by

$$\gamma_q\le\frac{\sigma_{\max}(\mathbf{A}_{I_1}\mathbf{P})}{\sqrt{(m-m_0)(1-q-\beta)+|S|}}\cdot\|\mathbf{x}^k-\mathbf{x}^*\|. \qquad (2.42)$$

Step 4: Conclude. Combining (2.40) and (2.41) with the bound on γ_q shows that the expected relative improvement $\mathbb{E}_k\|\mathbf{x}^{k+1}-\mathbf{x}^*\|^2/\|\mathbf{x}^k-\mathbf{x}^*\|^2$ is upper bounded by

$$1-\frac{1}{Z_J}\left(\sigma^+_{q-\beta,\min}(\mathbf{A}_{I_1}\mathbf{P})^2-\frac{|S|\gamma_q^2}{\|\mathbf{x}^k-\mathbf{x}^*\|^2}-\frac{2\gamma_q\sqrt{|S|}\,\|(\mathbf{A}_{I_1}\mathbf{P})_C(\mathbf{x}^k-\mathbf{x}^*)\|}{\|\mathbf{x}^k-\mathbf{x}^*\|^2}\right)$$
$$\leq 1-\frac{1}{Z_J}\left(\sigma^+_{q-\beta,\min}(\mathbf{A}_{I_1}\mathbf{P})^2-\left[\frac{|S|\cdot\sigma_{\max}(\mathbf{A}_{I_1}\mathbf{P})^2}{\theta+|S|}+\frac{2\sqrt{|S|}\cdot\sigma_{\max}(\mathbf{A}_{I_1}\mathbf{P})^2}{\sqrt{\theta+|S|}}\right]\right), \tag{2.43}$$

where $\theta \coloneqq (m-m_0)(1-q-\beta)$. Now, we can upper bound Z_J by $Z_{q-\beta}$ from (2.34). Next, consider the function $f(x)=\frac{x}{\theta+x}+\frac{2\sqrt{x}}{\sqrt{\theta+x}}$, and observe that $f(|S|)$ appears in the upper bound (2.43). Since $f'(x)>0$ for all $x>0$, the upper bound is increasing in $|S|$. Because $|S|\leq\beta(m-m_0)$, we conclude that the most pessimistic bound, independent of $|S|$ and J (and hence k), is obtained by setting $|S|=\beta(m-m_0)$, which implies that

$$\mathbb{E}_k\|\mathbf{x}^{k+1}-\mathbf{x}^*\|^2\leq(1-C_{q,\beta})\cdot\|\mathbf{x}^k-\mathbf{x}^*\|^2, \qquad \text{where } C_{q,\beta} \text{ is defined in (2.35).}$$

To ensure that the mean squared error contracts after each step, it suffices for $C_{q,\beta}$ to be positive: this is exactly secured by the condition (2.36). By iterating, we obtain (2.37). □

Note that Lemma 2.4.2 only provides a sufficient condition for convergence in the worst case (see [Ste23] for further discussion). Empirically, convergence is observed for larger values of β because the corruptions are quickly detected and trapped beyond the threshold. The dependence on $|S|$ in (2.43) shows that if the number of admissible, corrupted equations is small, then far less is demanded of the spectral quantities of $\mathbf{A}_{I_1}\mathbf{P}$ for the mean squared error to contract. For similar reasons, the QuantileRK method also empirically outperforms currently available theoretical convergence guarantees [Had+22; Che+23].

2.4.2 Proof of Theorem 2.4.1

To prove Theorem 2.4.1, our strategy will be to show that the ratio of $\sigma_{\max}(\mathbf{A}_{I_1}\mathbf{P})$ and $\sigma^+_{q-\beta,\min}(\mathbf{A}_{I_1}\mathbf{P})$ is of the same order with high probability. Together with the condition (2.36)

for convergence in Lemma 2.4.2, this implies that the QuantileSC-RK method will efficiently converge if the proportion of corruptions is small enough.

First, we show that $\sigma^+_{q-\beta,\min}(\mathbf{A}_{I_1}\mathbf{P})$ can be lower bounded with high probability as long as the effective aspect ratio $(m-m_0)/(n-m_0)$ is tall enough. This is proved using a similar technique as [Had+22, Proposition 3.4].

Lemma 2.4.3. *Let $\mathbf{P}$ be an orthogonal projection onto a fixed $(n-m_0)$-dimensional subspace, $\alpha \in (0,1]$, and $\mathbf{A}_{I_1} \in \mathbb{R}^{(m-m_0)\times n}$ be a random matrix that satisfies Assumptions A1 and A2. Then there exist absolute constants $C, \theta > 0$ such that if*

$$\frac{m-m_0}{n-m_0} \geq \frac{24}{\alpha}\log\left(\frac{36\theta(1+CK^2)}{\alpha^{3/2}}\right), \tag{2.44}$$

then with probability at least $1-3e^{-\alpha(m-m_0)/24}$,

$$\inf_{\substack{T\subseteq I_1 \\ |T|=\alpha(m-m_0)}} \sigma^+_{\min}((\mathbf{A}_{I_1}\mathbf{P})_T) \geq \frac{\alpha^{3/2}}{32}\sqrt{m-m_0}. \tag{2.45}$$

In the case where $\mathbf{A}_{I_1}$ is Gaussian, this result holds with CK^2 replaced by one and $\theta = 2\sqrt{2}$.

Proof. Recall that $\mathbb{S}^{n-1} = \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\| = 1\}$ denotes the unit sphere. By Lemma 2.3.23, we can fix an ε-net $\mathcal{N}$ of $\text{range}(\mathbf{P}) \cap \mathbb{S}^{n-1}$ with cardinality $|\mathcal{N}| \leq (3/\varepsilon)^{n-m_0}$ for some $\varepsilon \in (0,1]$ to be chosen later. Fix any $T \subseteq I_1$ with $|T| = \alpha(m-m_0)$. Since for any $\mathbf{z} \in \text{range}(\mathbf{P}) \cap \mathbb{S}^{n-1}$, there exists $\mathbf{x} \in \mathcal{N}$ such that $\|\mathbf{z}-\mathbf{x}\| \leq \varepsilon$, using the reverse triangle inequality and $\|(\mathbf{A}_{I_1}\mathbf{P})_T\| \leq \|\mathbf{A}_{I_1}\mathbf{P}\|$ implies that

$$\sigma^+_{\min}((\mathbf{A}_{I_1}\mathbf{P})_T) \geq \inf_{\mathbf{z}\in\text{range}(\mathbf{P})\cap\mathbb{S}^{n-1}} \|(\mathbf{A}_{I_1}\mathbf{P})_T)\mathbf{z}\| \geq \inf_{\mathbf{x}\in\mathcal{N}} \|(\mathbf{A}_{I_1}\mathbf{P})_T\mathbf{z}\| - \varepsilon\|\mathbf{A}_{I_1}\mathbf{P}\|. \tag{2.46}$$

Firstly, by Lemma 2.3.20 (with $s=1$), we have that with probability at least $1-2e^{-(m-m_0)}$,

$$\|\mathbf{A}_{I_1}\mathbf{P}\| \leq (1+CK^2)\sqrt{m-m_0}. \tag{2.47}$$

Next, our goal is to define an event $\mathcal{E}$ on which a good bound for $\inf_{\mathbf{x}\in\mathcal{N}}\|(\mathbf{A}_{I_1}\mathbf{P})_T)\mathbf{z}\|$ that is independent of T holds. More precisely, for every $j \in I_1$ and $\mathbf{x} \in \mathcal{N}$, define the "bad" event

$$\mathcal{E}_j^{\mathbf{x}} := \{|\langle \mathbf{a}_j, \mathbf{x}\rangle| \le \alpha/(4\theta)\},$$

where θ is some constant to be specified later. Let $\mathcal{E}^{\mathbf{x}}$ be the "good" event where less than $\alpha(m-m_0)/2$ of the events $(\mathcal{E}_j^{\mathbf{x}})_{j\in I_1}$ occur, and $\mathcal{E} := \bigcap_{\mathbf{x}\in\mathcal{N}} \mathcal{E}^{\mathbf{x}}$. Observe that $\langle \mathbf{P}\mathbf{a}_j, \mathbf{x}\rangle = \langle \mathbf{a}_j, \mathbf{P}\mathbf{x}\rangle = \langle \mathbf{a}_j, \mathbf{x}\rangle$ since $\mathbf{x} \in \text{range}(\mathbf{P})$. Therefore, on $\mathcal{E}$, at least half of the rows of $(\mathbf{A}_{I_1}\mathbf{P})_T$ have nontrivial correlation with any $\mathbf{x} \in \mathcal{N}$, which implies that

$$\inf_{\mathbf{x}\in\mathcal{N}}\|(\mathbf{A}_{I_1}\mathbf{P})_T\mathbf{x}\| = \inf_{\mathbf{x}\in\mathcal{N}}\sqrt{\sum_{j\in T}|\langle \mathbf{P}\mathbf{a}_j, \mathbf{x}\rangle|^2} \ge \sqrt{\frac{\alpha(m-m_0)}{2}\cdot\frac{\alpha^2}{16\theta^2}} \ge \frac{\alpha^{3/2}}{6\theta}\sqrt{m-m_0}. \quad (2.48)$$

To balance (2.47) and (2.48), we choose $\varepsilon = \alpha^{3/2}/(12\theta(1+CK^2))$. Therefore, if both events $\mathcal{E}$ and (2.47) hold, then (2.46) implies that the desired bound (2.45) holds.

It remains to bound the probability of the event $\mathcal{E}$, for which we will combine an anti-concentration result with a Chernoff bound. By using either [RV15, Theorem 1.2] if the row $\mathbf{a}_j$ has independent entries with bounded densities (with $\theta \ge 2\sqrt{2}$), or [CW01, Theorem 8] if $\mathbf{a}_j$ has a log-concave distribution (increasing the value of θ based on the absolute constant in this result), we deduce that $\mathbb{P}\left(\mathcal{E}_j^{\mathbf{x}}\right) \le \alpha/4$ for all $j \in I_1$. Hence, a standard Chernoff bound implies that $\mathbb{P}\left(\mathcal{E}^{\mathbf{x}}\right) \ge 1 - e^{-\alpha(m-m_0)/12}$ for all $\mathbf{x} \in \mathcal{N}$, and a union bound shows that $\mathcal{E}$ fails to hold with probability less than

$$|\mathcal{N}|\cdot\exp\left(\frac{-\alpha(m-m_0)}{12}\right) \le \exp\left((n-m_0)\log\left(\frac{3}{\varepsilon}\right) - \frac{\alpha(m-m_0)}{12}\right) \le e^{-\alpha(m-m_0)/24},$$

where the condition (2.44) is used for the final inequality. Combining this with the probability bound for (2.47) to hold completes the proof. □

Next, the following lemma bounds $Z_{q-\beta}$ from above with high probability. Note that for any fixed $T \subseteq I_1$ with $|T| = \alpha(m - m_0)$, the expected value of $\|(\mathbf{A}_{I_1}\mathbf{P})_T\|_F^2 = \sum_{j\in T}\|\mathbf{P}\mathbf{a}_j\|^2$ is $\alpha(m - m_0)(n - m_0)$.

Lemma 2.4.4. *Let $\mathbf{P}$ be an orthogonal projection onto a fixed $(n - m_0)$-dimensional subspace, $\alpha \in (0, 1]$, and $\mathbf{A}_{I_1} \in \mathbb{R}^{(m-m_0)\times n}$ be a random matrix that satisfies Assumptions A1 and A2. Then there exists an absolute constant $c > 0$ such that with probability at least $1 - e^{-c\alpha(m-m_0)/K^4}$,*

$$\sup_{\substack{T\subseteq I_1 \\ |T|=\alpha(m-m_0)}} \|(\mathbf{A}_{I_1}\mathbf{P})_T\|_F^2 \leq \left(2 + \frac{K^4}{c}\log\left(\frac{e}{\alpha}\right)\right)\alpha(m - m_0)(n - m_0). \tag{2.49}$$

Proof. For all fixed $T \subseteq I_1$ with $|T| = \alpha(m - m_0)$, Lemma 2.3.24 applied to the submatrix $(\mathbf{A}_{I_1}\mathbf{P})_T$ implies that there exists an absolute constant $c > 0$ such that for all $\varepsilon > 0$,

$$\mathbb{P}\left(\|(\mathbf{A}_{I_1}\mathbf{P})_T\|_F^2 \geq (1 + \varepsilon)\alpha(m - m_0)(n - m_0)\right) \leq e^{-c\min\{\varepsilon^2,\varepsilon\}\alpha(m-m_0)/K^4}.$$

Hence, by a union bound over all $\binom{m-m_0}{\alpha(m-m_0)} < e^{\alpha(m-m_0)\log(e/\alpha)}$ such subsets T, we deduce that the probability that the event (2.49) does not hold is not greater than $\exp\left\{-\alpha(m - m_0)\left(\frac{c\varepsilon}{K^4} - \log\left(\frac{e}{\alpha}\right)\right)\right\}$ for $\varepsilon \geq 1$. In particular, choosing $\varepsilon = 1 + \frac{K^4}{c}\log\left(\frac{e}{\alpha}\right)$ leads to the claimed probability guarantee. □

By combining our tail bounds for $\sigma_{\max}(\mathbf{A}_{I_1}\mathbf{P})$ and $\sigma^+_{q-\beta,\min}(\mathbf{A}_{I_1}\mathbf{P})$ as well as $Z_{q-\beta}$, we can now prove Theorem 2.4.1.

Proof of Theorem 2.4.1. In this proof, the various constants of the form $c_1, C_1, \ldots$ that appear only depend on K. By Lemma 2.3.20, $\sigma_{\max}(\mathbf{A}_{I_1}\mathbf{P}) \leq C_1\sqrt{m - m_0}$ with probability at least $1 - 2e^{-c_1(m-m_0)}$. By Lemma 2.4.3, $\sigma^+_{q-\beta,\min}(\mathbf{A}_{I_1}\mathbf{P}) \geq C_2(q - \beta)^{3/2}\sqrt{m - m_0}$ with probability at least $1 - 3e^{-c_2(q-\beta)(m-m_0)}$, given that the condition (2.31) is satisfied. Therefore, if

both of these events hold, then

$$\frac{\sigma^+_{q-\beta,\min}(\mathbf{A}_{I_1}\mathbf{P})^2}{\sigma_{\max}(\mathbf{A}_{I_1}\mathbf{P})^2} \geq \left(\frac{C_2}{C_1}\right)^2 (q-\beta)^3.$$

Hence, by Lemma 2.4.2 we deduce that the QuantileSC-RK algorithm converges if

$$\left(\frac{C_2}{C_1}\right)^2 (q-\beta)^3 > \frac{\beta}{1-q} + 2\sqrt{\frac{\beta}{1-q}}.$$

Since q is fixed and the right-hand side can be made arbitrarily small by decreasing β, it follows that this condition is satisfied as long as β is sufficiently small. Finally, Lemma 2.4.4 implies that $Z_{q-\beta} \leq C_{q,\beta}(m-m_0)(n-m_0)$ with probability at least $1 - e^{-c_3(q-\beta)(m-m_0)}$ for some constant $C_{q,\beta} > 0$ that only depends on q, β, and K. If all of these events hold, then Lemma 2.4.2 implies that QuantileSC-RK converges with the claimed rate (2.32). The proof is completed after simplifying the probability bound. □

2.5 Numerical experiments

In this section, we present numerical experiments that demonstrate various features of the SC-RK method (Algorithm 2.1) and the QuantileSC-RK method (Algorithm 2.2). For the plots with random, simulated data that follow, the lines represent the median over 200 trials, and the shaded regions indicate the 0.1- and 0.9-quantiles around the corresponding medians. The log relative error refers to the quantity $\log(\|\mathbf{x}^k - \mathbf{x}^*\|/\|\mathbf{x}^0 - \mathbf{x}^*\|)$. The experiments were performed on a MacBook Air M1 with 8GB RAM using Python 3.11.

2.5.1 SC-RK method for systems with correlated rows

In **Figure 2.2**, we compare the performance of the SC-RK method on a system with highly correlated rows for various sizes m_0 of I_0. It is known that RK performs poorly in this setting [NT14; NW13]. The entries of $\mathbf{A} \in \mathbb{R}^{2,000\times 1,000}$ are independently and uniformly

distributed on $[0.9, 1.1]$, and the solution $\mathbf{x}^* \in \mathbb{R}^{1,000}$ is a standard Gaussian vector. The same initial iterate starting in the solution space corresponding to the biggest block (i.e., $m_0 = 200$) is used for each variation.

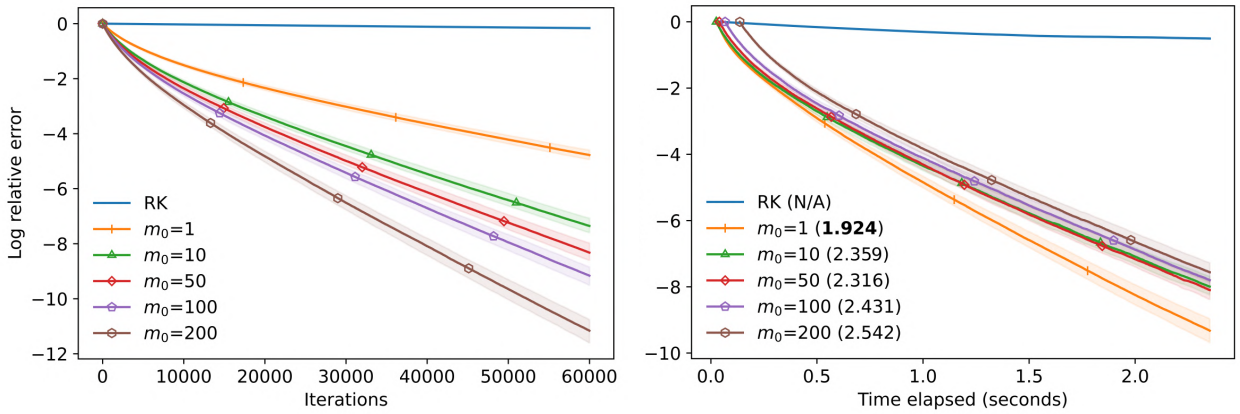


Figure 2.2: Performance of SC-RK on a system with highly correlated rows for various sizes m_0 of $\mathbf{A}_{I_0}$. **(Left)** Log relative error at each iteration. **(Right)** Log relative error against time elapsed, including the initial cost of precomputing $\mathbf{A}_{I_0}^\dagger$ for each m_0. The time taken to reach a log relative error of less than -8 is reported in brackets (N/A indicates that this was not reached in 30 seconds).

As predicted by Corollary 2.3.12, the SC-RK method with I_0 as the first m_0 rows of $\mathbf{A}$ outperforms RK for any $m_0 \geq 1$ since the pairwise row correlations of $\mathbf{A}$ are bounded from below. Moreover, increasing m_0 increases the rate of convergence (see Theorem 2.3.17). However, since increasing m_0 leads to heavier iterations and a higher initial cost from computing $\mathbf{A}_{I_0}^\dagger$ (see Remark 2.1.3), the optimal block size for a given target error and time budget is not necessarily the largest as highlighted by Figure 2.2 (right).

2.5.2 SC-RK method for systems with low-rank structure

In **Figure 2.3**, we consider the performance of the SC-RK method on a structured matrix $\mathbf{A} \in \mathbb{R}^{2,000 \times 1,000}$, constructed as in Example 2.3.11. The first $r = 20$ rows of $\mathbf{A}$ are normalized standard Gaussian vectors. The remaining $m-r$ rows $(\mathbf{a}_j)_{j>r}$ are equal to $\mathbf{a}_j \coloneqq (1-\varepsilon)\mathbf{a}'_j + \varepsilon \mathbf{c}_j$, where $\varepsilon = 0.1$, $\mathbf{a}'_j$ is sampled from $\{\mathbf{a}_1, \ldots, \mathbf{a}_r\}$, and $\mathbf{c}_j$ is sampled from $\text{span}(\{\mathbf{a}_1, \ldots, \mathbf{a}_r\})^\perp$

and normalized; i.e., $\mathbf{a}_j$ mainly consists of a row from the special top block plus some noise in the orthogonal direction. The solution $\mathbf{x}^* \in \mathbb{R}^{1,000}$ is a standard Gaussian vector.

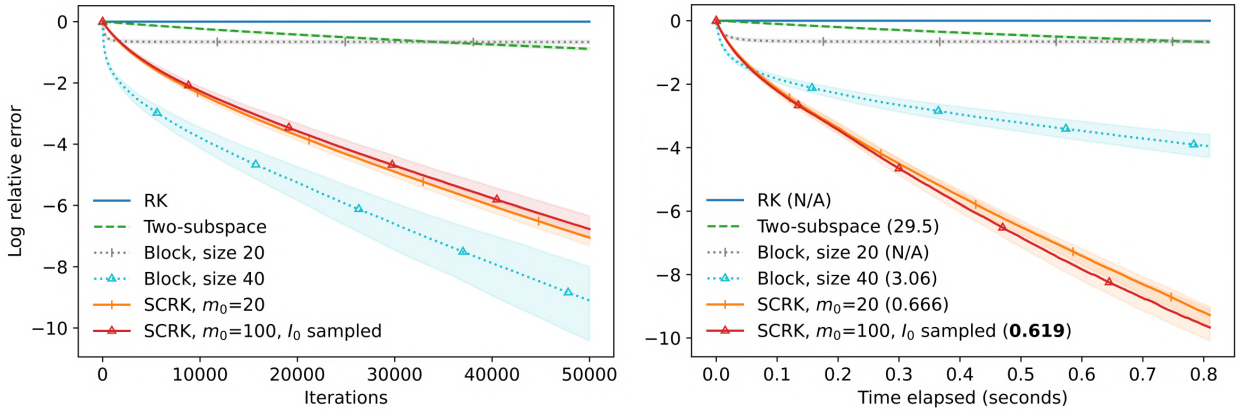


Figure 2.3: Performance of SC-RK on a coherent system with low-rank structure using a "perfect" block (with $m_0 = 20$) and a randomly sampled block (with $m_0 = 100$) as described in the main text. The two-subspace Kaczmarz method [NW13] and randomized block Kaczmarz method [NT14] (with two block sizes) are also included. **(Left)** Log relative error at each iteration. **(Right)** Log relative error against time elapsed, not including the initial costs of precomputing pseudoinverses for SC-RK and block Kaczmarz. The time taken to reach a log relative error of less than -8 is reported in the brackets (N/A indicates that this was not reached in 30 seconds).

The SC-RK algorithm is run with two choices of I_0: the first uses the "perfect" block of size $m_0 = 20$ with the rows $\{\mathbf{a}_1, \ldots, \mathbf{a}_r\}$ that generate the coherence structure. The second variant uses a block of $m_0 = 5r = 100$ rows of $\mathbf{A}$ sampled (without replacement) uniformly at random. This represents the case where the source of coherence is unknown, but sampling the effective rank of $\mathbf{A}$ (with an appropriate oversampling factor) should find a good block $\mathbf{A}_{I_0}$ as predicted by Theorem 2.3.15. Indeed, Figure 2.3 shows that both choices of I_0 converge effectively: the dramatic improvement in the per-iteration convergence rate of SC-RK over RK shown by the left plot is explained by the (inverse) scaled condition number $\sigma^+_{\min}(\mathbf{A}_{I_1}\mathbf{P})/\|\mathbf{A}_{I_1}\mathbf{P}\|_F = 9.33 \times 10^{-3}$ of $\mathbf{A}_{I_1}\mathbf{P}$ with $m_0 = 20$ (and similarly 9.56×10^{-3} with $m_0 = 100$) being significantly larger than the corresponding quantity $\sigma_{\min}(\mathbf{A})/\|\mathbf{A}\|_F = 3.29 \times 10^{-5}$ for $\mathbf{A}$ (see Section 2.3.3).

The two-subspace Kaczmarz method [NW13] and randomized block Kaczmarz method [NT14] (using equally-sized blocks of size 20 and 40 chosen uniformly at random, and pre-computed pseudoinverses) are also included. The same initial iterate as SC-RK with $m_0 = 20$ is used. It is known that these algorithms perform well in systems with highly correlated rows, such as the one previously considered in Figure 2.2. However, Figure 2.3 shows that the effectiveness of two-subspace Kaczmarz and block Kaczmarz with blocks of size 20 that are "too small" is impeded by the coherence structure of $\mathbf{A}$.

On the other hand, block Kaczmarz with blocks of size 40 that are "large enough" (relative to $r = 20$ for this problem) converges effectively. While Figure 2.3 (left) shows that it converges with a greater per-iteration rate than SC-RK (since it effectively uses 40 new rows in each iteration instead of just one), Figure 2.3 (right) shows that the lighter iterations of the SC-RK method actually makes it more efficient on a time basis.

2.5.3 SC-RK method for noisy systems

In **Figure 2.4**, we consider the performance of the SC-RK algorithm on a noisy system to demonstrate the validity of the error horizon predicted by Theorem 2.3.4. The rows of $\mathbf{A} \in \mathbb{R}^{300\times 100}$ are independent normalized standard Gaussian vectors, the solution $\mathbf{x}^* \in \mathbb{R}^{100}$ is a standard Gaussian vector, and the entries of the noise vector $\mathbf{R}$ are independently and uniformly distributed on $[-0.01, 0.01]$.

2.5.4 QuantileSC-RK algorithm

In **Figure 2.5**, we compare the performance of the QuantileSC-RK and QuantileRK [Had+22] methods on Gaussian systems $\mathbf{A}$ with different aspect ratios, where the measurements are corrupted by a sparse vector with c non-zero entries independently and uniformly distributed on $[-1, 1]$. The rows of $\mathbf{A}$ are independent normalized standard Gaussian vectors, and the solution $\mathbf{x}^*$ is a standard Gaussian vector.

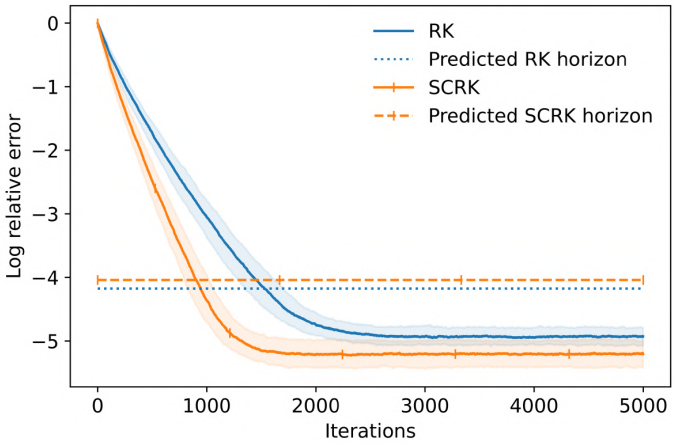
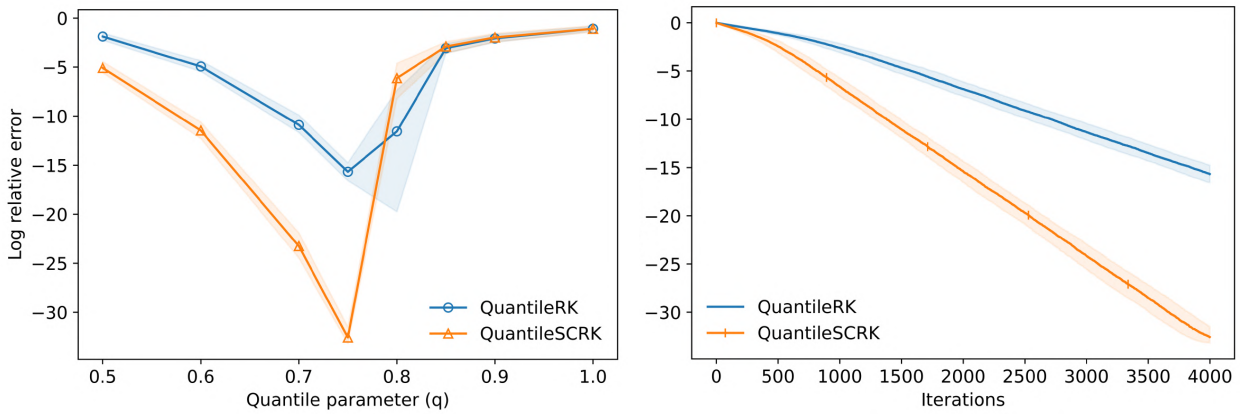


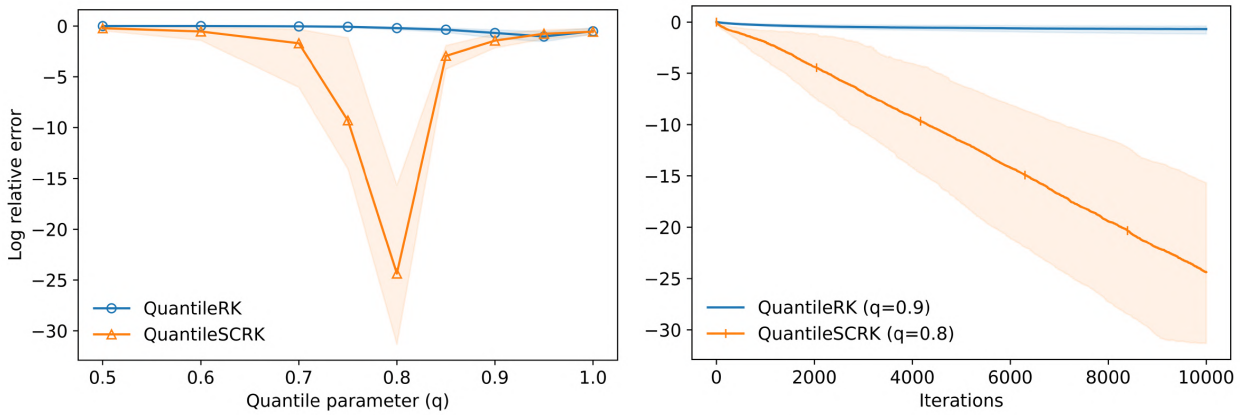
Figure 2.4: Convergence paths for the SC-RK (with I_0 equal to the first $m_0 = 25$ rows) and RK methods on a noisy system. The dashed/dotted lines indicate the predicted error horizons $\gamma_0 + \gamma_1$ from Theorem 2.3.4 and $\gamma = \|\mathbf{R}\|^2/\sigma_{\min}(\mathbf{A})^2$ from [Nee10], respectively.

Tall systems are considered in **Figure 2.5a**, where $100/500 = 20\%$ (resp. $100/480 \approx 20.8\%$) of the rows of $\mathbf{A}$ (resp. $\mathbf{A}_{I_1}$) correspond to corrupted measurements. These plots replicate the finding that the QuantileRK method converges effectively for tall, Gaussian-like matrices even in the presence of numerous corruptions [Had+22], and also show that exploiting information about corruption-free measurements using the QuantileSC-RK method accelerates convergence (see Theorem 2.4.1).

Almost-square systems are considered in **Figure 2.5b**, where $10/130 \approx 7.7\%$ (resp. $10/55 \approx 18.2\%$) of the rows of $\mathbf{A}$ (resp. $\mathbf{A}_{I_1}$) correspond to corrupted measurements. It is clear that the QuantileRK method is unable to make any progress in this setting. On the other hand, the QuantileSC-RK method converges for q around 0.8, which demonstrates that exploiting external knowledge in the form of a large block $\mathbf{A}_{I_0}$ corresponding to corruption-free measurements can enable convergence in such challenging settings.



(a) Tall system $\mathbf{A}^{500\times 50}$ with $c = 100$, $m_0 = 20$, $k = 4,000$, and $q_{\mathrm{RK}} = q_{\mathrm{SCRK}} = 0.75$.



(b) Almost-square systems $\mathbf{A}^{130\times 100}$ with $c = 10$, $m_0 = 75$, $k = 10,000$, $q_{\mathrm{RK}} = 0.9$, $q_{\mathrm{SCRK}} = 0.8$.

Figure 2.5: Performance of the QuantileSC-RK method, given a corruption-free block of size m_0, compared to the QuantileRK method [Had+22] on Gaussian systems with different aspect ratios and c corrupted measurements. **(Left)** Log relative error after k iterations for various values of the quantile parameter q. **(Right)** Convergence paths using the best quantile parameters q_{RK} and q_{SCRK}.

2.5.5 Systems of differential equations with inconsistent initial conditions

We consider the problem of numerically solving a system of differential equations given competing data for the initial conditions as another application of the QuantileSC-RK method. After discretization via a finite difference scheme, two types of equations arise: the first describe the underlying law and can be considered to be known exactly, and the second type encode the initial conditions, which can be obtained from real data with potentially faulty

measurements. Thus, the problem can be viewed as one about detecting and disregarding the "corrupted" equations coming from inconsistent initial conditions, given that the majority of the equations of the first type can be "trusted".

In **Figure 2.6**, we consider the linear system obtained from discretizing the differential equation $y'' = 0$ for a line illustrate this idea. The top 98×100 block is a Toeplitz matrix with entries $1, -2, 1$ along the diagonal before normalization, which we take to be $\mathbf{A}_{I_0}$. We consider two sets of initial conditions corresponding to two lines: Line 1 being $y = x$ with 10 initial conditions, and Line 2 being $y = 25 - x/2$ with 5 initial conditions.

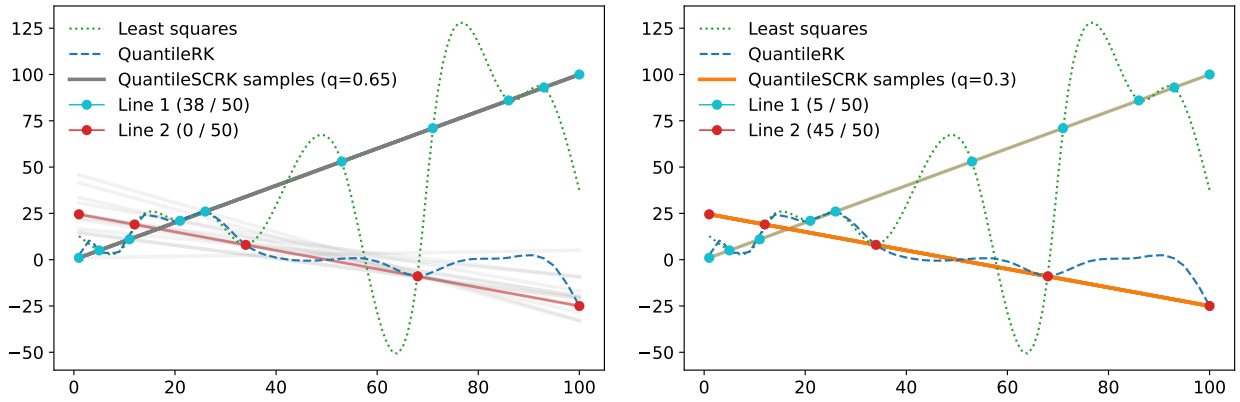


Figure 2.6: Solving a discretized differential equation for a line in the plane given two sets of inconsistent initial conditions as described in the main text. **(Left)** 50 outputs after 10,000 iterations of QuantileSC-RK with $q = 0.65$ (translucent gray lines); 38 out of the 50 converged to Line 1 (in the sense $\|\mathbf{A}_{\text{Line 1}}\mathbf{x}^k - \mathbf{b}_{\text{Line 1}}\|_2 < 10^{-3}$). **(Right)** 50 outputs after 10,000 iterations of the QuantileSC-RK algorithm with $q = 0.3$ (translucent orange lines); 45 out of the 50 converged to Line 2.

The plots show that solving this system using least squares or QuantileRK produces poor solutions. However, using QuantileSC-RK with a careful choice of the quantile parameter enables convergence to one line or the other as the algorithm is able to find a set of consistent initial conditions: when $q = 0.65$, QuantileSC-RK converges to Line 1 a majority of the time (left), and when $q = 0.3$, QuantileSC-RK converges to Line 2 instead (right). We also observed that the initial iterate $\mathbf{x}^0$ has a significant biasing effect on which solution is preferred for convergence. Our initialization—a random Gaussian vector projected onto the

solution space of $\mathbf{A}_{I_0}$—intuitively corresponds to a random line centered on the x-axis on average, which is closer to Line 2.

2.5.6 CT image reconstruction

Finally, we investigate the performance of QuantileSC-RK on a realistic dataset. We consider the Shepp–Logan phantom [SL74], generated using the Air Tools II package [HJ18] with parameters $N = 50$ (the image is $N \times N$), $\theta = \{0, 2, 4, \ldots, 178\}$ (angles used), and $\rho = 50$ (number of parallel rays). The image is encoded by the measurement matrix $\mathbf{A} \in \mathbb{R}^{4,500 \times 2,500}$ and measurements $\mathbf{b} \in \mathbb{R}^{4,500}$. A subset I_0 of $m_0 = 500$ rows of $\mathbf{A}$ was randomly chosen to be corruption-free (e.g., corresponding to trustworthy measurements), and a random set of $c = 1,125$ of the remaining measurements were corrupted by quantities uniformly distributed in $[2, 6]$ to produce the corrupted measurements $\widetilde{\mathbf{b}}$.

In **Figure 2.7**, we show various reconstructions given $\mathbf{A}$ and the corrupted measurements $\widetilde{\mathbf{b}}$. It is clear that the least squares solution is very poor. The QuantileRK method, initialized from zero, achieves a noisy reconstruction that does not recover the fine details. Using QuantileSC-RK with the corruption-free block $\mathbf{A}_{I_0}$ achieves the best reconstruction, even though a significant proportion (25%) of the measurements have been corrupted.

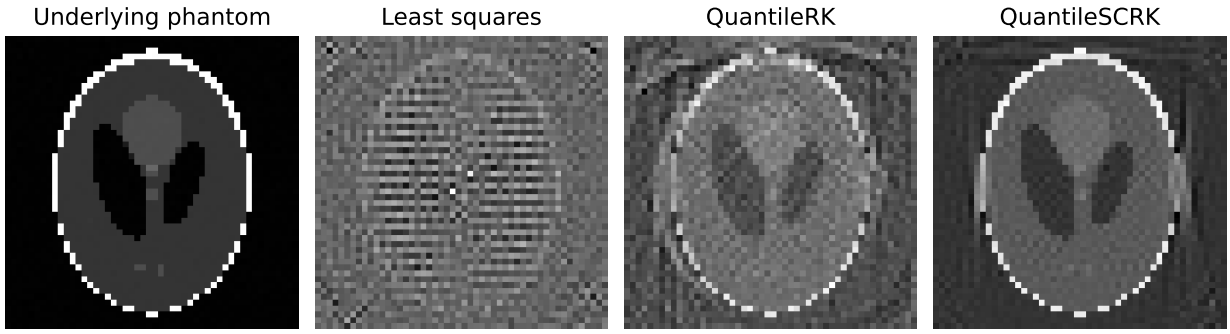


Figure 2.7: Reconstructions of the Shepp–Logan phantom from $\mathbf{A} \in \mathbb{R}^{4,500 \times 2,500}$ and $\widetilde{\mathbf{b}} \in \mathbb{R}^{4,500}$ with $c = 1,125$ corruptions as described in the main text. The QuantileSC-RK method, given a corruption-free block $\mathbf{A}_{I_0}$ of size $m_0 = 500$, and the QuantileRK method were both run using $q = 0.7$ for $60m = 270,000$ iterations, obtaining a final ℓ_2 error $\|\mathbf{x}^k - \mathbf{x}\|_2$ of 3.47 and 6.85, respectively.

2.6 Concluding remarks

In this chapter, we introduced the subspace-constrained randomized Kaczmarz (SC-RK) method for solving consistent, overdetermined systems of linear equations $\mathbf{A}\mathbf{x} = \mathbf{b}$, which provides a framework for studying the dynamics of the randomized Kaczmarz algorithm when the iterates are confined to a selected solution space $\mathbf{A}_{I_0}\mathbf{x} = \mathbf{b}_{I_0}$. We described the convergence rate of the SC-RK method in terms of the spectral properties of the matrix $\mathbf{A}_{I_1}\mathbf{P}$, where $\mathbf{P}$ is the orthogonal projector onto $\text{null}(\mathbf{A}_{I_0})$. We also demonstrated, both theoretically and empirically, how the SC-RK method can exploit approximately low-rank structure to accelerate convergence.

We also proposed the QuantileSC-RK method for solving corrupted linear systems, which is able to exploit external knowledge about corruption-free subsystems. In addition to theoretical convergence analysis, we demonstrated numerically that it is able to converge for almost-square corrupted linear systems, where existing iterative methods are ineffective, and that it can be useful for solving differential equations with inconsistent initial conditions and image reconstruction from significantly corrupted measurements.

The framework of subspace-constrained iterations raises many possible future directions. For example, since our analysis showed that the SC-RK updates simplify to a version of the usual Kaczmarz updates with skewed step directions and the projector $\mathbf{P}$ acts as a right preconditioner for $\mathbf{A}$ to improve the convergence rate, it seems plausible that similar ideas could be applied to related solvers such as the sketch-and-project algorithm [GR15a] or iterative projection methods for solving systems of linear inequalities [LL10; BN15]. It would also be interesting to develop and analyze a QuantileSC-RK method in which the trusted solution space is built up adaptively in a data-driven way, based on the information accumulated during the iteration process, which could lead to an effective way for solving corrupted linear systems even in the absence of external knowledge.

Chapter 3

Subspace-constrained randomized coordinate descent for linear systems with good low-rank matrix approximations

This chapter is based on the following joint work with Elizaveta Rebrova:
J. Lok and E. Rebrova. "Subspace-constrained randomized coordinate descent for linear systems with good low-rank matrix approximations". *SIAM Journal on Matrix Analysis and Applications*, 2026. To appear. arXiv: 2506.09394 [math.NA]

3.1 Introduction

The problem of solving large-scale systems of linear equations $\mathbf{Ax} = \mathbf{b}$ is ubiquitous in machine learning and scientific computing. The growing size of datasets presents new challenges and demands on the algorithms used. For instance, the entire matrix $\mathbf{A}$ may not fit into

memory, which means that memory-efficient iterative methods are practically advantageous or even necessary. Furthermore, if evaluating entries of the matrix $\mathbf{A}$ is associated with a nontrivial cost, then it may also be desirable to seek algorithms that are based on accessing small parts of the matrix or are more easily parallelizable, compared to algorithms based on other primitives such as matrix–vector products.

A key motivation for this work is the development of an efficient solver for dense, positive semidefinite (psd) *linear systems that requires a limited number of entry accesses at a time.* As a primary example, consider the problem of kernel ridge regression (KRR) [SS02]. Given n data points, this is a method for nonparametric regression that is equivalent to solving a linear system $(\mathbf{K} + \lambda\mathbf{I})\mathbf{x} = \mathbf{y}$, where $\mathbf{K} \in \mathbb{R}^{n\times n}$ is a positive definite kernel matrix, which has entries measuring the similarity between each pair of input feature vectors, and $\lambda \geq 0$ is a chosen regularization parameter. The matrix $\mathbf{K}$ is typically dense, and the regularized system $\mathbf{K} + \lambda\mathbf{I}$ can be very poorly conditioned for small values of λ. Loading the entire kernel matrix in memory requires storing $O(n^2)$ entries, and solving the linear system using a standard direct method requires $O(n^3)$ arithmetic operations; for large n, both can be infeasible.

Good low-rank matrix approximations. Fortunately, matrices arising from many real-life datasets often possess fast spectral decay [UT19], and so can be well-approximated by much smaller low-rank matrices. A particularly effective method for computing a good low-rank approximation of a psd $n \times n$ matrix $\mathbf{A}$ is the *randomly pivoted Cholesky* (RPCholesky) algorithm, recently proposed and analyzed by Chen et al. [Che+25]. Given an approximation rank parameter $d \geq 0$, RPCholesky efficiently finds a near-optimal low-rank approximation of $\mathbf{A}$ using only $O(d^2n)$ arithmetic operations, $O(dn)$ entry evaluations, and $O(dn)$ storage. It can also be practically accelerated using block computations and rejection sampling [ETW24].

Briefly, the RPCholesky algorithm selects a random set of d pivots $\mathcal{S} \subseteq [n] := \{1, 2, \ldots, n\}$ using adaptive diagonal sampling, and outputs a (column) *Nyström approximation*

$$\mathbf{A}\langle\mathcal{S}\rangle = \mathbf{F}\mathbf{F}^{\mathsf{T}}$$

of $\mathbf{A}$ in factorized form, where $\mathbf{F} \in \mathbb{R}^{n\times d}$ is a lower-triangular matrix after permuting the rows to bring the pivots $\mathcal{S}$ to the top, and the residual matrix $\mathbf{A} - \mathbf{A}\langle\mathcal{S}\rangle$ has trace-norm error comparable with the best rank-r approximation of $\mathbf{A}$ (from truncated SVD) for some $r \approx d$. More precisely, the approximation quality is described by

Theorem 3.1.1 ([Che+25, Theorem 5.1])**.** *Let $\mathbf{A} \in \mathbb{R}^{n\times n}$ be a psd matrix with eigenvalues $\lambda_1(\mathbf{A}) \geq \ldots \geq \lambda_n(\mathbf{A}) \geq 0$, and denote $\log_+(t) = \max\{\log t, 0\}$ for $t > 0$. Suppose that for some fixed integer $r \geq 0$ and real $\delta > 0$, the approximation rank parameter d satisfies*

$$d \geq \frac{r}{\delta} + \min\left\{ r\log\left(\frac{1}{\delta\eta_r}\right), r + r\log_+\left(\frac{2^r}{\delta}\right)\right\}, \quad \textit{where} \quad \eta_r := \frac{\sum_{i>r}\lambda_i(\mathbf{A})}{\sum_{i=1}^{n}\lambda_i(\mathbf{A})}.$$

Then, the rank-d Nyström approximation $\mathbf{A}\langle\mathcal{S}\rangle$ output by RPCholesky satisfies

$$\mathbb{E}[\mathrm{tr}(\mathbf{A} - \mathbf{A}\langle\mathcal{S}\rangle)] \leq (1+\delta)\cdot\sum_{i>r}\lambda_i(\mathbf{A}).$$

Among other applications, matrix approximation techniques based on RPCholesky have recently been used to build efficient preconditioners for conjugate gradient-based linear solvers for KRR [Día+24].

Randomized block coordinate descent. A convenient and widely-used optimization approach that requires a limited number of local entry accesses is the coordinate descent method [LL10; Nes12]. For solving the psd system $\mathbf{A}\mathbf{x} = \mathbf{b}$, the algorithm has basic iterations of the form

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \frac{(\mathbf{A}_{:,j})^{\mathsf{T}}\mathbf{x}^k - \mathbf{b}_j}{\mathbf{A}_{j,j}}\mathbf{e}_j, \tag{3.1}$$

where $\mathbf{e}_j \in \mathbb{R}^n$ denotes the j^{th} standard basis vector in $\mathbb{R}^n$. Observe that the updates (3.1) are very lightweight, requiring only access to a single column of $\mathbf{A}$ and $O(n)$ arithmetic operations. Leventhal and Lewis [LL10] showed that if the coordinate $j \in [n]$ is sampled with probability $\mathbf{A}_{j,j}/\operatorname{tr}(\mathbf{A})$, then the corresponding *randomized coordinate descent* (RCD) method converges linearly with a rate that depends on the spectrum of $\mathbf{A}$. More precisely, if $\mathbf{x}^*$ is any solution of $\mathbf{A}\mathbf{x} = \mathbf{b}$ and $\lambda^+_{\min}(\mathbf{A})$ is the smallest non-zero eigenvalue of $\mathbf{A}$, then

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|^2_{\mathbf{A}} \leq \left(1 - \frac{\lambda^+_{\min}(\mathbf{A})}{\operatorname{tr}(\mathbf{A})}\right)^k \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|^2_{\mathbf{A}}, \tag{3.2}$$

where $\|\mathbf{z}\|_{\mathbf{A}} = \sqrt{\mathbf{z}^\mathsf{T}\mathbf{A}\mathbf{z}}$ is the norm induced by $\mathbf{A}$. The iteration (3.1) readily generalizes to the *block coordinate descent* update

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \mathbf{e}_{\mathcal{J}}(\mathbf{A}_{\mathcal{J},\mathcal{J}})^\dagger\big((\mathbf{A}_{:,\mathcal{J}})^\mathsf{T}\mathbf{x}^k - \mathbf{b}_{\mathcal{J}}\big), \tag{3.3}$$

where $\mathbf{e}_{\mathcal{J}} \in \mathbb{R}^{n\times|\mathcal{J}|}$ denotes the matrix whose columns are the standard basis vectors in $\mathbb{R}^n$ corresponding to a subset $\mathcal{J} \subseteq [n]$ (i.e., only the coordinates in $\mathcal{J}$ are updated). However, with block size $\ell = |\mathcal{J}| > 1$, bounds on the convergence rate have only been obtained in special cases: e.g., if given a well-conditioned partitioning of the columns [NT14; WN18], or if the blocks $\mathcal{J}$ are sampled with a special distribution (which is not easy to compute) [RK20; MDK20].

The convergence rate bound (3.2) implies that if the matrix $\mathbf{A}$ is very well-conditioned, in the sense that $\lambda^+_{\min}(\mathbf{A})$ is of the same order as the maximum eigenvalue and so $\lambda^+_{\min}(\mathbf{A})/\operatorname{tr}(\mathbf{A}) = O(n)$, then $O(n\log(1/\varepsilon))$ iterations and $O(n^2\log(1/\varepsilon))$ arithmetic operations suffice to compute an approximate solution with relative error in $\|\cdot\|_{\mathbf{A}}$ bounded by ε. However, if $\mathbf{A}$ is not so well-conditioned, then the bound rapidly deteriorates and RCD can easily be demonstrated to converge very slowly. Recent works such as [Der+25a; Der+25b] show that properly designed block generalizations of RCD can produce competitive linear solvers with implicit preconditioning properties for systems with large spectral outliers.

Key question. Motivated by these observations, we focus on the following question: *Can we use an efficiently-computable low-rank approximation to improve the rate of convergence for solving a psd system* $\mathbf{Ax} = \mathbf{b}$ *with a lightweight, memory-efficient iterative algorithm such as randomized block coordinate descent, especially when* $\mathbf{A}$ *exhibits rapid spectral decay?*

3.1.1 Notation

We write vectors and matrices in boldface. We denote the identity matrix by $\mathbf{I}$, and the Moore–Penrose pseudoinverse of a matrix $\mathbf{A}$ by $\mathbf{A}^\dagger$. Given a subset $\mathcal{J} \subseteq [n] := \{1, 2, \ldots, n\}$, we denote the column submatrix of $\mathbf{A} \in \mathbb{R}^{n \times n}$ indexed by $\mathcal{J}$ by $\mathbf{A}_{:,\mathcal{J}}$. Similarly, we indicate the row and principal submatrix indexed by $\mathcal{J}$ by $\mathbf{A}_{\mathcal{J},:}$ and $\mathbf{A}_{\mathcal{J},\mathcal{J}}$, respectively. Furthermore, we write $\mathbf{e}_\mathcal{J} \in \mathbb{R}^{n \times |\mathcal{J}|}$ to denote the matrix whose columns are the standard basis vectors in $\mathbb{R}^n$ corresponding to the coordinates in $\mathcal{J}$. We denote the eigenvalues of a symmetric matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ in non-increasing order by $\lambda_1(\mathbf{A}) \geq \ldots \geq \lambda_n(\mathbf{A}) = \lambda_{\min}(\mathbf{A})$, and the smallest non-zero eigenvalue by $\lambda^+_{\min}(\mathbf{A})$. We use the Loewner order $\mathbf{A} \preceq \mathbf{B}$ for symmetric matrices $\mathbf{A}$, $\mathbf{B}$ to mean that $\mathbf{B} - \mathbf{A}$ is positive semidefinite (psd). Given a psd matrix $\mathbf{A} \succeq \mathbf{0}$, the $\mathbf{A}$-(semi)norm is defined by $\|\mathbf{z}\|_\mathbf{A} = \sqrt{\mathbf{z}^\mathsf{T}\mathbf{A}\mathbf{z}}$.

3.1.2 Main results

In this section, we will describe an efficient algorithm based on randomized block coordinate descent that provides a solution to our key question. We will also describe a more general subspace-constrained framework that we developed to obtain our theoretical results.

I. Subspace-constrained randomized coordinate descent (SC-RCD). We show that a column Nyström approximation $\mathbf{A}\langle\mathcal{S}\rangle := \mathbf{A}_{:,\mathcal{S}}(\mathbf{A}_{\mathcal{S},\mathcal{S}})^\dagger\mathbf{A}_{\mathcal{S},:}$ of the psd matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ can be combined with randomized block coordinate descent by constraining the iterates $\mathbf{x}^k$ of

RCD within the following affine subspace, parameterized by the pivot set $\mathcal{S} \subseteq [n]$:

$$\mathbf{x}^k \in \{\mathbf{x} \in \mathbb{R}^n : \mathbf{A}_{\mathcal{S},:}\mathbf{x} = \mathbf{b}_{\mathcal{S}}\} \quad \text{for all } k \geq 0.$$

We show that when the blocks $\mathcal{J} \subseteq [n]$ in each iteration consist of ℓ coordinates independently sampled with probability proportional to the diagonal of the *residual matrix*

$$\mathbf{A}^\circ := \mathbf{A} - \mathbf{A}\langle\mathcal{S}\rangle, \tag{3.4}$$

then the convergence rate depends on the spectrum of $\mathbf{A}^\circ$ instead of $\mathbf{A}$. That is, *the restriction of the dynamics to the subspace corresponding to a given low-rank approximation effectively preconditions the system*, and a significant improvement in the convergence rate can be achieved when a good low-rank matrix approximation is efficiently computable, using an algorithm such as RPCholesky to select $\mathcal{S}$.

The SC-RCD method. Given an approximation rank parameter $d \geq 0$ and block size parameter $\ell \geq 1$, the resulting algorithm, which we refer to as *subspace-constrained randomized coordinate descent* (SC-RCD)[1], can be described as follows:

- A rank-d Nyström approximation $\mathbf{A}\langle\mathcal{S}\rangle = \mathbf{F}\mathbf{F}^\mathsf{T}$ with corresponding pivots $\mathcal{S}$ is efficiently computed using an algorithm such as RPCholesky, and an initial iterate $\mathbf{x}^0$ is obtained by solving $\mathbf{A}_{\mathcal{S},:}\mathbf{x}^0 = \mathbf{b}_{\mathcal{S}}$. Let $\mathbf{C} := (\mathbf{A}_{\mathcal{S},\mathcal{S}})^\dagger \mathbf{A}_{\mathcal{S},:} \in \mathbb{R}^{d\times n}$.

- In the $(k+1)^{\text{st}}$ iteration, given the iterate $\mathbf{x}^k$ and corresponding residual vector $\mathbf{r}^k = \mathbf{A}\mathbf{x}^k - \mathbf{b}$, we form a random subset $\mathcal{J} = \{j_1, \ldots, j_\ell\} \subseteq [n] \setminus \mathcal{S}$ of ℓ coordinates, each sampled independently with probability proportional to the corresponding diagonal entry

[1] If the block size $\ell > 1$, then the updates of the algorithm are more accurately described as randomized *block* coordinate descent, but we will use the same acronym SC-RCD for brevity.

of the residual matrix $\mathbf{A}^\circ = \mathbf{A} - \mathbf{A}\langle\mathcal{S}\rangle$. Then, we compute

$$\boldsymbol{\alpha}^k = (\mathbf{A}^\circ_{\mathcal{J},\mathcal{J}})^\dagger \mathbf{r}^k_{\mathcal{J}} \in \mathbb{R}^\ell \quad \text{and} \quad \boldsymbol{\beta}^k = \mathbf{C}_{:,\mathcal{J}}\boldsymbol{\alpha}^k \in \mathbb{R}^d,$$

and use these vectors to update the coordinates of $\mathbf{x}^k$ in $\mathcal{J}$ and $\mathcal{S}$:

$$\mathbf{x}^{k+1}_{\mathcal{J}} = \mathbf{x}^k_{\mathcal{J}} - \boldsymbol{\alpha}^k, \quad \mathbf{x}^{k+1}_{\mathcal{S}} = \mathbf{x}^k_{\mathcal{S}} + \boldsymbol{\beta}^k. \tag{3.5}$$

The other coordinates of $\mathbf{x}^k$ remain unchanged. Furthermore, the residual vector is updated by

$$\mathbf{r}^{k+1} = \mathbf{r}^k - \mathbf{A}^\circ_{:,\mathcal{J}}\boldsymbol{\alpha}^k. \tag{3.6}$$

Remark 3.1.2. If we denote a solution of the system $\mathbf{Ax} = \mathbf{b}$ by $\mathbf{x}^*$, then it can be shown that given $\mathbf{x}^k$, the next iterate $\mathbf{x}^{k+1}$ produced by the update (3.5) satisfies

$$\mathbf{x}^{k+1} = \underset{\mathbf{x}\in\mathbb{R}^n}{\arg\min} \|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{A}} \quad \text{such that} \quad \mathbf{x} = \mathbf{x}^k + \mathbf{e}_{\mathcal{J}}\boldsymbol{\alpha} + \mathbf{e}_{\mathcal{S}}\boldsymbol{\beta}, \tag{3.7}$$

where only $\boldsymbol{\alpha} \in \mathbb{R}^\ell$ and $\boldsymbol{\beta} \in \mathbb{R}^d$ are free to vary, i.e., we minimize the $\mathbf{A}$-norm error with respect to the coordinates in $\mathcal{J}$ and $\mathcal{S}$. Without the subspace constraint, we would only optimize over the coordinates in the sampled subset $\mathcal{J}$, which corresponds to the usual block coordinate descent update (3.3). With the additional subspace constraint, the coordinates in $\mathcal{S}$ must also be modified to ensure that the iterates continue to satisfy $\mathbf{A}_{\mathcal{S},:}\mathbf{x} = \mathbf{b}_{\mathcal{S}}$. What distinguishes the iteration-dependent subset $\mathcal{J}$ from the fixed subset $\mathcal{S}$ is that we know that $\mathbf{x}^k$ satisfies $\mathbf{A}_{\mathcal{S},:}\mathbf{x}^k = \mathbf{b}_{\mathcal{S}}$ a priori, which allows for the update for $\mathcal{S}$ to be computed more efficiently using the update for $\mathcal{J}$.

See Algorithm 3.1 for pseudocode for an implementation of SC-RCD. The comments provide pointers to parts of Section 3.3.2, where more details on efficient implementation

Algorithm 3.1 Subspace-constrained randomized coordinate descent (SC-RCD)

Require: Psd matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$, vector $\mathbf{b} \in \mathbb{R}^n$, approximation rank d, block size ℓ
Ensure: Approximate solution $\mathbf{x} \in \mathbb{R}^n$ of $\mathbf{A}\mathbf{x} = \mathbf{b}$, residual vector $\mathbf{r} = \mathbf{A}\mathbf{x} - \mathbf{b} \in \mathbb{R}^n$

Compute pivot set $\mathcal{S} \subseteq [n]$ and partial pivoted Cholesky factor $\mathbf{F} \in \mathbb{R}^{n \times d}$ defining the Nyström approximation $\mathbf{A}\langle\mathcal{S}\rangle = \mathbf{F}\mathbf{F}^\mathsf{T}$ ▷ E.g., RPCholesky [Che+25; ETW24]
Compute $\boldsymbol{\beta} \leftarrow (\mathbf{F}_{\mathcal{S},:})^{-\mathsf{T}}(\mathbf{F}_{\mathcal{S},:})^{-1}\mathbf{b}_{\mathcal{S}} \in \mathbb{R}^d$ ▷ Triangular solves with $\mathbf{F}$
Set $\mathbf{x} \leftarrow \mathbf{0}_{n\times 1}$, $\mathbf{x}_{\mathcal{S}} \leftarrow \boldsymbol{\beta}$, and $\mathbf{r} \leftarrow \mathbf{A}_{:,\mathcal{S}}\boldsymbol{\beta} - \mathbf{b} \in \mathbb{R}^n$ ▷ Initial iterate (i)
Compute $\mathbf{C} \leftarrow (\mathbf{F}_{\mathcal{S},:})^{-\mathsf{T}}\mathbf{F}^\mathsf{T} \in \mathbb{R}^{d\times n}$ ▷ Triangular solves with $\mathbf{F}$ (ii)
Set $\mathbf{p} \leftarrow \mathbf{0}_{n\times 1}$, compute $\mathbf{p}_j \leftarrow \mathbf{A}_{j,j} - \|\mathbf{F}_{j,:}\|_2^2$ for $j \in [n] \setminus \mathcal{S}$, and normalize: $\mathbf{p} \leftarrow \mathbf{p}/\sum_j \mathbf{p}_j$
for $k = 1, 2, \ldots$ **do**
    Sample $\mathcal{J} = \{j_1, \ldots, j_\ell\}$ with $j_1, \ldots, j_\ell \sim \mathbf{p}$ i.i.d. ▷ Possibly without replacement (iii)
    Solve $\big(\mathbf{A}_{\mathcal{J},\mathcal{J}} - \mathbf{F}_{\mathcal{J},:}(\mathbf{F}_{\mathcal{J},:})^\mathsf{T}\big)\boldsymbol{\alpha} = \mathbf{r}_{\mathcal{J}}$ for $\boldsymbol{\alpha} \in \mathbb{R}^\ell$ ▷ Possibly inexactly (iv)
    $\boldsymbol{\beta} \leftarrow \mathbf{C}_{:,\mathcal{J}}\boldsymbol{\alpha} \in \mathbb{R}^d$
    $\mathbf{x}_{\mathcal{J}} \leftarrow \mathbf{x}_{\mathcal{J}} - \boldsymbol{\alpha}$, $\mathbf{x}_{\mathcal{S}} \leftarrow \mathbf{x}_{\mathcal{S}} + \boldsymbol{\beta}$ ▷ Update coordinates of iterate in $\mathcal{J}$ and $\mathcal{S}$
    $\mathbf{r} \leftarrow \mathbf{r} - \big(\mathbf{A}_{:,\mathcal{J}}\boldsymbol{\alpha} - \mathbf{F}((\mathbf{F}_{\mathcal{J},:})^\mathsf{T}\boldsymbol{\alpha})\big)$ ▷ Update residual vector
end for

(such as taking advantage of the lower triangular structure of $\mathbf{F}_{\mathcal{S},:}$) and computational costs are discussed.

While the updates in (3.5) have been described in terms of the residual matrix $\mathbf{A}^\circ$, we emphasize that neither $\mathbf{A}$ nor $\mathbf{A}^\circ$ actually need to be stored in memory. We only assume that the auxiliary matrix $\mathbf{C} \in \mathbb{R}^{d\times n}$ is precomputed and stored, which is feasible if $d \ll n$. Each iteration of SC-RCD only needs to access the entries $\mathbf{A}_{:,\mathcal{J}} \in \mathbb{R}^{n\times\ell}$ in the ℓ columns indexed by the sampled subset $\mathcal{J}$. Furthermore, each iteration can be performed in $O((\ell + d)n)$ arithmetic operations, provided that the block size ℓ is at most $O(\sqrt{n})$.

SC-RCD convergence analysis. Conditional on the quality of the low-rank approximation output by the RPCholesky algorithm, we show that the SC-RCD method converges linearly in expectation with a rate that depends on the eigenvalues of $\mathbf{A}$ without the largest spectral outliers:

Theorem 3.1.3. *Let $\mathbf{A}\mathbf{x} = \mathbf{b}$ be a consistent linear system with psd $\mathbf{A} \in \mathbb{R}^{n\times n}$ and solution $\mathbf{x}^*$. Let the eigenvalues of $\mathbf{A}$ be $\lambda_1(\mathbf{A}) \geq \cdots \geq \lambda_n(\mathbf{A}) \geq 0$ with smallest non-zero eigenvalue $\lambda_{\min}^+(\mathbf{A})$. Let $\mathbf{A}\langle\mathcal{S}\rangle$ be a rank-d Nyström approximation computed using the RPCholesky*

algorithm, and $\{\mathbf{x}^k\}_{k\geq 0}$ *denote the corresponding sequence of SC-RCD iterates with block size* $\ell \geq 1$. *Fix an integer* $r \geq 0$ *and real* $\delta > 0$, $\rho \in (0,1)$. *If the approximation rank* d *satisfies*

$$d \geq \frac{r}{\delta} + r \log\left(\frac{1}{\delta \eta_r}\right), \quad \text{where} \quad \eta_r := \frac{\sum_{i>r} \lambda_i(\mathbf{A})}{\sum_{i=1}^n \lambda_i(\mathbf{A})}, \tag{3.8}$$

then, with probability at least $1-\rho$, *the residual matrix* $\mathbf{A}^\circ = \mathbf{A} - \mathbf{A}\langle \mathcal{S} \rangle$ *satisfies*

$$\operatorname{tr}(\mathbf{A}^\circ) \leq \rho^{-1}(1+\delta) \sum_{i>r} \lambda_i(\mathbf{A}). \tag{3.9}$$

Furthermore, conditional on the event that (3.9) *holds, denoted by* $\mathcal{E}$, *the expected relative error in the* $\mathbf{A}$*-norm satisfies*

$$\mathbb{E}\left[\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{A}}^2 \mid \mathcal{E}\right] \leq \left(1 - \frac{\lambda_{\min}^+(\mathbf{A})}{\rho^{-1}(1+\delta)\sum_{i>r}\lambda_i(\mathbf{A})}\right)^{k\ell} \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_{\mathbf{A}}^2. \tag{3.10}$$

The proof of Theorem 3.1.3 is given in Section 3.3. More generally, we show that if the subspace constraint is defined by an arbitrary Nyström approximation $\mathbf{A}\langle \mathcal{S} \rangle$, not necessarily computed using RPCholesky, then the SC-RCD algorithm converges linearly with a rate that depends on the spectrum of the residual matrix $\mathbf{A}^\circ = \mathbf{A} - \mathbf{A}\langle \mathcal{S} \rangle$ (see Theorem 3.3.2).

Remark 3.1.4 (Randomized low-rank approximation)**.** The probability of the event $\mathcal{E}$ in (3.9), which ensures the quality of the randomized low-rank approximation for the iterative phase, can be boosted by the standard trick of running RPCholesky $T \geq 1$ times and choosing the best approximation with the smallest trace-norm. Then, by applying (3.9) to each run with probability parameter $\rho = 1/2$, say, we conclude that the best approximation fails to satisfy the same bound in $\mathcal{E}$ with probability at most 2^{-T}. Note that this procedure is very easy to run in parallel.

Remark 3.1.5 (Approximation rank)**.** A technical caveat is the dependence of the approximation rank d on η_r, the relative trace-norm error of the best rank-r approximation, in (3.8). Theoretically, it is possible to compute a randomized rank-d Nyström approximation $\widehat{\mathbf{A}}$ such

that

$$\mathbb{E}[\operatorname{tr}(\mathbf{A} - \widehat{\mathbf{A}})] \leq (1+\delta) \sum_{i>r} \lambda_i(\mathbf{A}) \quad \text{as long as } d \geq \frac{r}{\delta} + r - 1,$$

instead of (3.8), by using a more computationally expensive method for sampling the pivots $\mathcal{S}$ known as *determinantal point process* (DPP) *sampling* (see [Che+25, Table 2] and the references therein). Using Markov's inequality to define the event $\mathcal{E}$ in (3.9), the same convergence rate (3.10) holds with such an approximation.

Remark 3.1.6 (Block size and sampling)**.** Choosing a larger block size ℓ for the SC-RCD method means that a heavier $\ell \times \ell$ inner linear system has to be solved (directly or iteratively) in each iteration. This increased cost may be offset by two benefits. The first is that modern computational architectures can realize computational benefits from using block computations due to parallelism, caching, etc. (e.g., see [ETW24, App. A]). The second is that larger blocks results in more progress to be made with each projection; this is reflected by the bound in Theorem 3.1.3, which shows that the iteration complexity improves at least linearly in the block size ℓ. Later, we also show that the entries in each block can be sampled without replacement to obtain a guarantee that is at least as good (see Remark 3.3.5).

However, we expect (and empirically observe) the actual improvement from projections onto bigger blocks to be even better due to the implicit "low-rank approximation effect": e.g., rigorous results showing more precise rates depending on the entire spectrum are known for blocks obtained using DPP sampling [MDK20; RK20; XXZ25] or subgaussian sketches [DR24]. Proving tighter bounds with computationally cheap schemes is more challenging. In a recent line of work, it has been shown that DPP sampling can be approximated by uniform sampling if the input matrix satisfies some incoherence properties [DY24; Der+25a; Der+25b]. When the entries in each block are uniformly sampled in SC-RCD, an analogous result as Theorem 3.1.3 also holds (see Proposition 3.3.3).

SC-RCD complexity. Theorem 3.1.3 shows that the convergence of SC-RCD depends on the approximation rank d, the block size ℓ, and on the *normalized tail condition number* of

$\mathbf{A}$, defined as

$$\bar{\kappa}_r(\mathbf{A}) := \frac{1}{n-r}\frac{\sum_{i>r}\lambda_i(\mathbf{A})}{\lambda^+_{\min}(\mathbf{A})}, \qquad \text{for } r < \text{rank}(\mathbf{A}). \tag{3.11}$$

Note that $\bar{\kappa}_r(\mathbf{A})$ lower bounds the classical condition number of $\mathbf{A} - \mathbf{A}_r$, where $\mathbf{A}_r$ is the best rank-r approximation of $\mathbf{A}$ in the Frobenius norm; i.e., $\lambda_{r+1}(\mathbf{A})/\lambda^+_{\min}(\mathbf{A})$. When implemented as Algorithm 3.1, the SC-RCD method satisfies the following complexity result in terms of the size of the system n and the parameters d, ℓ, and $\bar{\kappa}_r(\mathbf{A})$, which is proved in Section 3.3.2.

Theorem 3.1.7. *Let $\mathbf{A}\mathbf{x} = \mathbf{b}$ be a consistent linear system with psd $\mathbf{A} \in \mathbb{R}^{n\times n}$ and solution $\mathbf{x}^*$. The SC-RCD method (Algorithm 3.1) with approximation rank $d \geq 0$ and block size $\ell \geq 1$, combined with the boosting procedure described in Remark 3.1.4, computes an approximate solution $\mathbf{x}^k$ that satisfies $\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|^2_{\mathbf{A}} \leq \varepsilon \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|^2_{\mathbf{A}}$ after $k = \lceil 4(n/\ell)\bar{\kappa}_r(\mathbf{A})\log(2/\varepsilon)\rceil$ iterations. In total,*

$$O\big(nd^2\log(1/\varepsilon)\big) + O\left(\left(\frac{n^2(d+\ell)}{\ell} + n\ell^2 + n\ell d\right)\cdot \bar{\kappa}_r(\mathbf{A})\log(1/\varepsilon)\right) \quad \textit{operations} \tag{3.12}$$

are required, where r is the largest integer satisfying $d \geq r + r\log(1/\eta_r)$ with η_r defined as in (3.8). Moreover, $O(nd\log(1/\varepsilon)) + O(n^2\cdot\bar{\kappa}_r(\mathbf{A})\log(1/\varepsilon))$ entry evaluations of $\mathbf{A}$, and $O(n(d+\ell))$ storage are required. Here, big O notation is used to hide absolute constants.

In particular, if we choose $\ell = d$, then (3.12) simplifies to

$$O\big((n^2 + nd^2)\cdot\bar{\kappa}_r(\mathbf{A})\log(1/\varepsilon)\big) \quad \text{operations.} \tag{3.13}$$

We see that the SC-RCD method takes advantage of the flat-tailed structure of the spectrum of $\mathbf{A}$ (i.e., when $\bar{\kappa}_r(\mathbf{A})$ is well-bounded), which can appear in practice due to explicit regularization or the effects of noise in the data.

Remark 3.1.8. To illustrate the claim of Theorem 3.1.7, let us suppose that $\mathbf{A}$ is not extremely ill-conditioned. Specifically, suppose that there exists an absolute constant $\tau > 0$

such that

$$\frac{\sum_{i=1}^{n} \lambda_i(\mathbf{A})}{\lambda_{\min}^{+}(\mathbf{A})} \leq O(n^{\tau}). \tag{3.14}$$

Then, r can be taken as large as $r = O(d/\log n)$ in the complexity bounds in Theorem 3.1.7. An important setting in which (3.14) is often satisfied is kernel ridge regression, where $\mathbf{A} = \mathbf{K} + \lambda \mathbf{I}$, the kernel matrix $\mathbf{K}$ has unit diagonals ($\mathbf{K}_{ii} = 1$ for all i), and the regularization parameter λ is not too small (i.e., $\lambda \geq Cn^{-\tau}$). In addition, if $\mathbf{K}$ also exhibits rapid spectral decay, then $\mathbf{A}$ would be expected to have a well-bounded normalized tail condition number (governed by λ). In particular, if we assume that $\bar{\kappa}_r(\mathbf{A}) = O(1)$ for some $r = O(\sqrt{n}/\log n)$, then, Theorem 3.1.7 implies that SC-RCD with $d = O(\sqrt{n})$ and $\ell = O(\sqrt{n})$ can solve the KRR problem to ε-relative error using $O(n^2 \log(1/\varepsilon))$ arithmetic operations. Note that this is optimal in terms of n for solving a dense $n \times n$ linear system.

II. A subspace-constrained sketch-and-project framework. To analyze the SC-RCD method, we consider the algorithm as an instance of a more general *subspace-constrained sketch-and-project* framework, which we develop in Section 3.2 and may be of independent interest. The *sketch-and-project method* [GR15a] encompasses a class of iterative algorithms for solving linear systems $\mathbf{Ax} = \mathbf{b}$ with $\mathbf{A} \in \mathbb{R}^{m \times n}$ (not necessarily psd), including the randomized Kaczmarz [SV09], randomized coordinate descent [LL10], and randomized Newton [Gow+19a] methods.

In each iteration of the standard sketch-and-project algorithm, a sketching matrix $\mathbf{S} \in \mathbb{R}^{\ell \times m}$ is drawn independently from an input distribution $\mathcal{D}$, and the current iterate $\mathbf{x}^k \in \mathbb{R}^n$ is projected onto the sketched linear system $\mathbf{SAx} = \mathbf{Sb}$ with respect to the norm $\|\mathbf{x}\|_{\mathbf{B}} = \sqrt{\mathbf{x}^{\mathsf{T}}\mathbf{Bx}}$ induced by a positive definite matrix parameter $\mathbf{B} \in \mathbb{R}^{n \times n}$:

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x} \in \mathbb{R}^n} \|\mathbf{x} - \mathbf{x}^k\|_{\mathbf{B}} \quad \text{such that} \quad \mathbf{SAx} = \mathbf{Sb}. \tag{3.15}$$

We propose to constrain the dynamics of the iterates from (3.15) within a particular (affine) subspace parameterized by a matrix parameter $\mathbf{Q} \in \mathbb{R}^{d \times m}$:

$$\mathbf{x}^k \in \{\mathbf{x} \in \mathbb{R}^n : \mathbf{QAx} = \mathbf{Qb}\} \quad \text{for all } k \geq 0.$$

In Section 3.2, we show that the theory for the subspace-constrained sketch-and-project method parallels the standard sketch-and-project method. The main difference is the emergence of a projector $\mathbf{P_B}$ onto the subspace null($\mathbf{QA}$) that acts to constrain the iterates. We prove that the convergence rate depends on a projected version of the original matrix, $\mathbf{AP_B}$, instead of $\mathbf{A}$, which shows that the subspace constraint effectively acts as a preconditioner and can speed up the convergence in various general cases (see Theorem 3.2.4 and Remark 3.2.6).

If the matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ is positive definite, then the SC-RCD method can be derived as a special case of this framework by choosing $\mathbf{B} = \mathbf{A}$; extremely sparse sketching matrices of the form $\mathbf{S} = \mathbf{e}_{\mathcal{J}}^{\mathsf{T}}$, defined as in (3.3); and a subspace constraint defined by the matrix $\mathbf{Q} = \mathbf{e}_{\mathcal{S}}^{\mathsf{T}}$, which can be identified with the set of pivots $\mathcal{S} \subseteq [n]$ corresponding to a Nyström approximation $\mathbf{A}\langle\mathcal{S}\rangle$ of $\mathbf{A}$.

III. Randomized block Kaczmarz convergence rate. As a part of our analysis of the sketch-and-project framework—which is not specific to the subspace-constrained version—we show that the iteration complexity improves at least linearly in the block size ℓ when the sketching matrices $\mathbf{S} \in \mathbb{R}^{\ell \times m}$ consist of ℓ independent and identically distributed rows (in Proposition 3.2.9).

Previously, bounds for block sketch-and-project methods have only been obtained for more special sketching matrices that are dense [Der+20; DR24] or based on more computationally expensive DPP sampling schemes [MDK20; Der+25a]. With more randomness, the rates obtained are generally sharper. However, Proposition 3.2.9 applies more generically, such as when standard coordinate basis vectors are sampled using a simple scheme.

Concretely, this result applies to the randomized block Kaczmarz method [Elf80; NT14] when the blocks consist of i.i.d. rows of $\mathbf{A}$, sampled with probability proportional to the squared row norms, which is a direct block generalization of the classical randomized Kaczmarz method [SV09].

Proposition 3.1.9. *Consider solving the consistent linear system* $\mathbf{A}\mathbf{x} = \mathbf{b}$ *with* $\mathbf{A} \in \mathbb{R}^{m \times n}$ *and solution* $\mathbf{x}^*$. *Let* $\{\mathbf{x}^k\}_{k \geq 0}$ *be the iterates of the randomized block Kaczmarz method, defined by*

$$\mathbf{x}^{k+1} = \mathbf{x}^k - (\mathbf{A}_{\mathcal{J},:})^\dagger(\mathbf{A}_{\mathcal{J},:}\mathbf{x}^k - \mathbf{b}_{\mathcal{J}}),$$

where the block $\mathcal{J} = \{j_1, \ldots, j_\ell\}$ *in each iteration consists of* ℓ *rows of* $\mathbf{A}$, *independently sampled from the distribution* $\{\|\mathbf{A}_{j,:}\|_2^2/\|\mathbf{A}\|_F^2\}_{j=1}^m$. *If* $\sigma_{\min}^+(\mathbf{A})$ *denotes the smallest non-zero singular value of* $\mathbf{A}$, *then*

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|_2^2 \leq \left(1 - \frac{\sigma_{\min}^+(\mathbf{A})^2}{\|\mathbf{A}\|_F^2}\right)^{\ell k} \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_2^2.$$

Proposition 3.1.9 is a special case of Corollary 3.2.11 for the subspace-constrained randomized block Kaczmarz method, which appears later in Section 3.2. To the best of our knowledge, this bound, which describes the explicit dependence of the rate on the block size, is new even for the standard randomized block Kaczmarz method.

In prior work [NT14], a randomized block Kaczmarz method, which additionally requires $\mathbf{A}$ to be row-normalized, was analyzed based on sampling non-intersecting blocks that partition the rows of the matrix. The sampling strategy in Proposition 3.1.9 is more general and slightly differs: in particular, it can result in repeated rows in the blocks. However, the convergence rate can only improve if the blocks are sampled without replacement instead, and we refer to Remark 3.3.5, which appears later in Section 3.3.2, for more details.

3.1.3 Related works

Coordinate descent and sketch-and-project. Important instances of the sketch-and-project framework [GR15a; RT20; Gow+18; Gow+21] for solving linear systems include the randomized coordinate descent (RCD) [LL10; Nes12]—also known as randomized Gauss–Seidel—and the closely-related randomized Kaczmarz (RK) [SV09] methods. The RK and RCD methods and their variants have been extensively studied, including [MNR15; HNR17; NSW16; NT14; SL19; Had+22; EGW26; Rat+25a]. Notably, the subspace-constrained sketch-and-project framework developed in this work generalizes a subspace-constrained RK method analyzed in [LR24]. Note that coordinate descent algorithms for more general objective functions have also been extensively studied in the optimization literature: e.g., see [RT14; RT16; QR16; TY10].

An accelerated sketch-and-project method with Nesterov momentum was analyzed in [Gow+18], building on earlier works on accelerated RK [LW16] and RCD [Nes12; LS13; NS17; Tu+17]. The accelerated method requires additional tuning parameters: theoretically, with a specific choice of values (which are not easily computable), the accelerated method leads to an improved convergence rate bound. In particular, [Tu+17] presents experiments showing that accelerated RCD can outperform RCD and the conjugate gradient method for large-scale KRR problems in machine learning.

A closely related line of works [DY24; Der+25a; Der+25b] analyzes randomized solvers based on sketch-and-project that are also especially effective for solving approximately low-rank systems, building on the insight that sketch-and-project can exploit rapid spectral decay [DR24]. Most recently, Dereziński et al. [Der+25b] showed that an algorithm based on accelerated RCD called CD++ can exploit large spectral outliers. Specifically, they prove that for any $\tilde{O}(1) \leq r \leq n$ (where $\tilde{O}$ hides polylog factors in n), a solution of the psd system $\mathbf{Ax} = \mathbf{b}$ with ε-relative error in $\|\cdot\|_{\mathbf{A}}$ can be computed using

$$\tilde{O}(nr^2) + \tilde{O}(n^2\sqrt{\bar{\kappa}_r(\mathbf{A})}\log(1/\varepsilon)) \quad \text{operations ([Der+25b, §5]).} \tag{3.16}$$

The CD++ algorithm achieves (3.16) through the use and analysis of techniques such as adaptive acceleration, approximate regularized projections, randomized Hadamard preconditioning, and block memoization. One of the main takeaways is that the rate of CD++ cannot be outperformed by any solver based on matrix-vector products, such as Krylov subspace-based solvers ([Der+25a, Theorem 3]).

The SC-RCD method with $d = O(r \log n)$, $\ell = d$, and $1 \leq r \leq O(\sqrt{n}/\log n)$ has a similar complexity bound as (3.16) for flat-tailed systems with $\bar{\kappa}_r(\mathbf{A}) = O(1)$ satisfying condition (3.14). Otherwise, CD++ has a better dependence on the normalized tail condition number due to the incorporation of Nesterov's momentum in the algorithm, which could be integrated into SC-RCD as a part of future work. At the same time, SC-RCD does not require storing the entire input matrix, while the CD++ guarantee (3.16) assumes that $O(n^2)$ memory is available for storing a preprocessed version of the matrix. Algorithmically, the main difference is that CD++ *implicitly* captures the leading part of the spectrum of $\mathbf{A}$, whereas it is *explicitly* learned in SC-RCD by constraining the dynamics of RCD. The SC-RCD method is based on the combination of two fundamental ideas: low-rank matrix approximation and a simple iterative solver. We believe that this allows for flexibility: e.g., the aforementioned innovations in CD++ can be combined with the subspace constraint idea.

Conjugate gradient method. Another popular class of iterative solvers is based on Krylov subspaces [NT24], which includes the *conjugate gradient method* (CG) [Saa03] for solving psd systems. In practice, preconditioning is often crucial for these methods to be effective. For kernel ridge regression (KRR) specifically, a *preconditioned conjugate gradient* (PCG) method for solving $(\mathbf{K}+\lambda\mathbf{I})\mathbf{x} = \mathbf{y}$ using a low-rank Nyström approximation $\widehat{\mathbf{K}} = \mathbf{F}\mathbf{F}^\mathsf{T}$ of $\mathbf{K}$ computed with RPCholesky was analyzed by Díaz et al. [Día+24]. We note that there many other methods, based on dimension reduction, have been proposed to solve large-scale

KRR problems: e.g., we refer to [Rat+25b; Día+24; ACW17; Mea+20; RCR00; RR07; SB00; WS00] and the references therein.

To give a brief overview of the PCG method of [Día+24], a preconditioner $\mathbf{M} = \widehat{\mathbf{K}} + \lambda\mathbf{I}$ is constructed from an SVD of $\mathbf{F}$, and the spectrum of $\mathbf{M}^{-1/2}(\mathbf{K} + \lambda\mathbf{I})\mathbf{M}^{-1/2}$ determines the resulting rate of convergence. It is shown that if the approximation rank d is larger than the λ-tail rank of $\mathbf{K}$, which is defined as the smallest integer r such that $\sum_{i>r} \lambda_i(\mathbf{K}) \leq \lambda$, then the preconditioned matrix has constant condition number, meaning that $O(n^2 \log(1/\varepsilon))$ operations suffices to compute a solution with ε-relative error in $\|\cdot\|_{\mathbf{A}}$ ([Día+24, Theorem 2.2]). This is a similar result as Theorem 3.1.7 for SC-RCD: however, we note that SC-RCD can be applied to general psd systems.

Low-rank approximation. Low-rank matrix approximation is a fundamental idea in numerical linear algebra [MT20]. The most relevant forms for SC-RCD are based on (column) Nyström approximation or interpolative decomposition [Che+25; Don+25; ETW24]. These are aligned with the standard coordinate basis, which means that the corresponding subspace constraint can be enforced by updating a fixed subset of coordinates. These algorithms use random, adaptive pivoting, and are accompanied by provable guarantees of quality that are comparable to the best low-rank approximation. Other simple strategies such as uniform sampling and greedy pivoting may work well in practice, but do not have good error bounds in general. DPP sampling [DM21], which is based on sampling blocks weighted by their squared volumes, offers the best known near-optimal bounds, but are not easily computable. Recently, a greedy deterministic method based on maximization of a trace-norm is also analyzed in [FL24].

More generally, the subspace constraint for sketch-and-project can be defined using low-rank approximations from *random embeddings* of $\mathbf{A}$, which mix across all the coordinates to perform dimension reduction (e.g., multiplying by a Gaussian matrix). These methods are often more robust and lead to a better approximation than those based on sampling. We refer

to the survey [TW23, §5.4] for a comprehensive discussion of Nyström-based adaptations of matrix approximation algorithms such as randomized SVD [HMT11], as well as extensions based on block Krylov iteration, which are more accurate for matrices with slow spectral decay.

3.1.4 Organization

The rest of the chapter is structured as follows. Section 3.2 describes a more abstract subspace-constrained sketch-and-project framework for solving general linear systems. In Section 3.3, the analysis is specialized to the SC-RCD method for solving psd linear systems. Section 3.4 presents numerical experiments demonstrating the performance of SC-RCD for solving psd systems, generated synthetically and coming from kernel ridge regression problems using real-life datasets. Finally, some concluding remarks are given in Section 3.5.

The remaining sections contain additional technical details. Section 3.6 describes some omitted proofs, Section 3.7 outlines an extension of the SC-RCD method for solving least squares problems, and Section 3.8 presents additional numerical experiments for SC-RCD. Section 3.9 describes an accelerated subspace-constrained sketch-and-project method, and Section 3.10 discusses an accelerated SC-RCD variant.

3.2 A general framework for subspace-constrained sketch-and-project

Consider the goal of solving a consistent system of linear equations $\mathbf{A}\mathbf{x} = \mathbf{b}$, where $\mathbf{A} \in \mathbb{R}^{m\times n}$. Since the matrix $\mathbf{A}$ may not necessarily be square or have full rank in this generality, we aim to find the min-norm solution

$$\mathbf{x}^* := \arg\min_{\mathbf{x}\in\mathbb{R}^n} \|\mathbf{x} - \mathbf{x}^0\|_{\mathbf{B}} \quad \text{such that} \quad \mathbf{A}\mathbf{x} = \mathbf{b}, \tag{3.17}$$

where $\mathbf{x}^0 \in \mathbb{R}^n$ represents any initial approximate solution and the norm is induced by a psd matrix parameter $\mathbf{B} \in \mathbb{R}^{n\times n}$. Observe that $\mathbf{x}^*$ is the projection of $\mathbf{x}^0$ onto $\mathbf{Ax} = \mathbf{b}$ with respect to the $\mathbf{B}$-norm, so equivalently,

$$\mathbf{x}^* = \mathbf{x}^0 - \mathbf{B}^{-1}\mathbf{A}^\mathsf{T}(\mathbf{A}\mathbf{B}^{-1}\mathbf{A}^\mathsf{T})^\dagger(\mathbf{A}\mathbf{x}^0 - \mathbf{b}). \tag{3.18}$$

Overview of the standard sketch-and-project method. In each iteration of the vanilla sketch-and-project framework [GR15a], a sketching matrix $\mathbf{S} \equiv \mathbf{S}^k$ is drawn independently from an input distribution $\mathcal{D}$, and the current iterate $\mathbf{x}^k \in \mathbb{R}^n$ is projected onto the sketched linear system $\mathbf{SAx} = \mathbf{Sb}$ with respect to the $\mathbf{B}$-norm as in (3.15). As shown in [GR15a], this update can be written in closed form as

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \mathbf{B}^{-1}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T}(\mathbf{S}\mathbf{A}\mathbf{B}^{-1}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T})^\dagger\mathbf{S}(\mathbf{A}\mathbf{x}^k - \mathbf{b}), \tag{3.19}$$

or expressed as a fixed point iteration in terms of the solution $\mathbf{x}^*$ from (3.17) as

$$\mathbf{x}^{k+1} - \mathbf{x}^* = (\mathbf{I} - \widetilde{\mathbf{Z}})(\mathbf{x}^k - \mathbf{x}^*), \tag{3.20}$$

where $\widetilde{\mathbf{Z}}$ is the orthogonal projector onto range$(\mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T})$: that is,

$$\widetilde{\mathbf{Z}} := \mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T}(\mathbf{S}\mathbf{A}\mathbf{B}^{-1}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T})^\dagger\mathbf{S}\mathbf{A}\mathbf{B}^{-1/2}. \tag{3.21}$$

The convergence rate of the sketch-and-project algorithm depends on the eigenvalue spectrum of the expected projection matrix $\mathbb{E}[\widetilde{\mathbf{Z}}]$, where the expectation is taken over the random sketching matrix $\mathbf{S} \sim \mathcal{D}$ ([GR15a, Theorem 4.6]). For a wide class of sketching matrices (including $\mathbf{S}$ with independent subgaussian entries as well as sparse counterparts), it has been shown that the convergence rate of sketch-and-project precisely depends on the entire singular value spectrum of $\mathbf{A}$ [Der+20; DR24].

3.2.1 The subspace-constrained sketch-and-project framework

Let $\mathbf{Q} \in \mathbb{R}^{d\times m}$ with $d < m$ be an arbitrary matrix parameter defining the (affine) subspace $\{\mathbf{x} \in \mathbb{R}^n : \mathbf{QAx} = \mathbf{Qb}\}$. We formulate the *subspace-constrained sketch-and-project method* as the process starting from an initial iterate $\mathbf{x}^0 \in \mathbb{R}^n$ solving $\mathbf{QAx}^0 = \mathbf{Qb}$, and with subsequent iterates given by

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}\in\mathbb{R}^n} \|\mathbf{x} - \mathbf{x}^k\|_{\mathbf{B}} \quad \text{such that} \quad \mathbf{SAx} = \mathbf{Sb},\ \mathbf{QAx} = \mathbf{Qb}, \tag{3.22}$$

where the sketching matrices $\mathbf{S} \equiv \mathbf{S}^k$ in each iteration are drawn independently from an input distribution $\mathcal{D}$. This is equivalent to the sketch-and-project method with the iterates $\mathbf{x}^k$ confined within the subspace $\{\mathbf{x} \in \mathbb{R}^n : \mathbf{QAx} = \mathbf{Qb}\}$.

The matrix $\mathbf{Q}$ could be chosen deterministically and consist of a selection of d rows of the system of linear equations. It could also be a random embedding of the linear system. Informally, constraining the iterates to a subspace parametrized by $\mathbf{Q}$ is computationally beneficial when

1. we can efficiently solve the smaller system $\mathbf{QAx} = \mathbf{Qb}$ to obtain an initial iterate $\mathbf{x}^0$; and

2. we can efficiently project onto the nullspace of $\mathbf{QAB}^{-1/2}$ in each iteration.

In the rest of this section, our goal is to develop general theory for the subspace-constrained sketch-and-project method (3.22). Specifically, we will derive closed-form expressions for the iterations and estimates for the convergence rate, and demonstrate the theoretical advantage of having a subspace constraint.

3.2.2 Update rule and comparison with unconstrained sketch-and-project

Our first goal is to show that the iterations of subspace-constrained sketch-and-project admit the following closed-form expressions, analogous to (3.19) and (3.20) for the unconstrained method.

Lemma 3.2.1 (Closed-form updates)**.** *Let* $\mathbf{P}$ *and* $\mathbf{Z}$ *be the orthogonal projection matrices onto* $\text{null}(\mathbf{QAB}^{-1/2})$ *and* $\text{range}(\mathbf{PB}^{-1/2}\mathbf{A}^\top\mathbf{S}^\top)$*, respectively; that is,*

$$\begin{aligned}\mathbf{P} &:= \mathbf{I} - (\mathbf{QAB}^{-1/2})^\dagger \mathbf{QAB}^{-1/2},\\ \mathbf{Z} &:= \mathbf{PB}^{-1/2}\mathbf{A}^\top\mathbf{S}^\top(\mathbf{SAB}^{-1/2}\mathbf{PB}^{-1/2}\mathbf{A}^\top\mathbf{S}^\top)^\dagger \mathbf{SAB}^{-1/2}\mathbf{P}.\end{aligned} \tag{3.23}$$

If $\mathbf{x}^0$ *is any vector satisfying* $\mathbf{QAx}^0 = \mathbf{Qb}$*, then the subspace-constrained sketch-and-project iterates* $\{\mathbf{x}^k\}_{k\geq 0}$ *from* (3.22) *satisfy the following:*

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \mathbf{B}^{-1/2}\mathbf{PB}^{-1/2}\mathbf{A}^\top\mathbf{S}^\top(\mathbf{SAB}^{-1/2}\mathbf{PB}^{-1/2}\mathbf{A}^\top\mathbf{S}^\top)^\dagger \mathbf{S}(\mathbf{Ax}^k - \mathbf{b}). \tag{3.24}$$

Furthermore, if $\mathbf{x}^*$ *is the min-norm solution as in* (3.17)*, then*

$$\mathbf{B}^{1/2}(\mathbf{x}^{k+1} - \mathbf{x}^*) = (\mathbf{I} - \mathbf{Z})\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*). \tag{3.25}$$

The proof of Lemma 3.2.1, which is technical, can be found in Section 3.6. As a quick sanity check, observe that with $\mathbf{Q} = \mathbf{0}$ and $\mathbf{P} = \mathbf{I}$ in Lemma 3.2.1, which corresponds to having no subspace constraint, we recover the sketch-and-project updates (3.19) and (3.20).

A consequence of the closed form update formulas is the following auxiliary lemma, whose proof can also be found in Section 3.6.

Lemma 3.2.2 (Invariant subspace property)**.** *Let* $\mathbf{x}^*$ *be the min-norm solution to* $\mathbf{Ax} = \mathbf{b}$ *defined in* (3.17) *and* $\mathbf{x}^k$ *be the* k^{th} *iterate from the subspace-constrained sketch-and-project method* (3.22)*. Then for all* $k \geq 0$, $\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*) \in \text{range}(\mathbf{PB}^{-1/2}\mathbf{A}^\mathsf{T})$.

The next result gives a formula for the decrease in error for each iteration of subspace-constrained sketch-and-project, and justifies that it is always at least as large as the error decrease from the corresponding unconstrained update.

Lemma 3.2.3 (Error decrease)**.** *With the same notation as in Lemma 3.2.1, the decrease in error in each iteration of subspace-constrained sketch-and-project is given by*

$$\|\mathbf{B}^{1/2}(\mathbf{x}^{k+1} - \mathbf{x}^*)\|_2^2 = \|\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)\|_2^2 - \|\mathbf{ZB}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)\|_2^2. \tag{3.26}$$

Furthermore, if $\widetilde{\mathbf{Z}}$ *is the corresponding projection matrix for the unconstrained sketch-and-project method from* (3.21) *with the same sketching matrix* $\mathbf{S}$*, then the error decrease with the subspace constraint is not smaller than the error decrease in the unconstrained case:*

$$\|\mathbf{ZB}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)\|_2 \geq \|\widetilde{\mathbf{Z}}\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)\|_2. \tag{3.27}$$

Proof. Since $\mathbf{Z}$ is an orthogonal projector, the per-iteration error decrease (3.26) in terms of the Euclidean norm follows from the fixed point equation (3.25) in Lemma 3.2.1 and Pythagoras' theorem.

Next, observe that showing (3.27) is equivalent to showing

$$\boldsymbol{\xi}^\mathsf{T}(\mathbf{SAB}^{-1/2}\mathbf{PB}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T})^\dagger\boldsymbol{\xi} \geq \boldsymbol{\xi}^\mathsf{T}(\mathbf{SAB}^{-1}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T})^\dagger\boldsymbol{\xi} \tag{3.28}$$

for all $\boldsymbol{\xi} = \mathbf{SA}(\mathbf{x}^k - \mathbf{x}^*) = \mathbf{SAB}^{-1/2}\mathbf{PB}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)$, recalling that $\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*) \in \text{range}(\mathbf{P})$ by Lemma 3.2.2. Since $\mathbf{P} \preceq \mathbf{I}$ in the psd order, we immediately have that

$$\mathbf{SAB}^{-1/2}\mathbf{PB}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T} \preceq \mathbf{SAB}^{-1}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T}. \tag{3.29}$$

Now, let $\mathcal{L} := \text{null}(\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\top\mathbf{S}^\top)$. Observe that $\boldsymbol{\xi} \in \text{range}(\mathbf{S}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P}) = \mathcal{L}^\perp$ and

$$\text{null}(\mathbf{S}\mathbf{A}\mathbf{B}^{-1}\mathbf{A}^\top\mathbf{S}^\top) = \text{null}(\mathbf{B}^{-1/2}\mathbf{A}^\top\mathbf{S}^\top) \subseteq \mathcal{L} = \text{null}(\mathbf{S}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\top\mathbf{S}^\top),$$

and so $\left(\mathbf{S}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\top\mathbf{S}^\top\big|_{\mathcal{L}^\perp}\right)^\dagger \succeq \left(\mathbf{S}\mathbf{A}\mathbf{B}^{-1}\mathbf{A}^\top\mathbf{S}^\top\big|_{\mathcal{L}^\perp}\right)^\dagger$, which implies (3.28). □

We will now prove that the subspace-constrained sketch-and-project method converges (in mean squared error and the $\mathbf{B}$-norm) with a rate that depends on the eigenvalue spectrum of the expected projection matrix $\mathbb{E}[\mathbf{Z}]$, where $\mathbf{Z}$ is defined in (3.23) and the expectation is taken over the distribution of the random sketching matrix $\mathbf{S}$.

Theorem 3.2.4. *Suppose that the same notation as Lemma 3.2.1 is used. Assume that the following exactness condition holds:*

$$\text{null}(\mathbb{E}[\mathbf{Z}]) = \text{null}(\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P}). \tag{3.30}$$

Then the subspace-constrained sketch-and-project iterates $\{\mathbf{x}^k\}_{k\geq 0}$ satisfy

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{B}}^2 \leq \left(1 - \lambda_{\min}^+(\mathbb{E}[\mathbf{Z}])\right)^k \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_{\mathbf{B}}^2,$$

where

$$\lambda_{\min}^+(\mathbb{E}[\mathbf{Z}]) = \min_{\substack{\mathbf{x}\in\text{range}(\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\top) \\ \|\mathbf{x}\|_2=1}} \mathbf{x}^\top(\mathbb{E}[\mathbf{Z}])\mathbf{x}.$$

Proof of Theorem 3.2.4. By considering the decrease in error in the $(k+1)^{\text{st}}$ iteration from Lemma 3.2.3, taking the expectation conditional on all the randomness up to the k^{th} iteration, which we denote by $\mathbb{E}_k$, and using the linearity of expectation, we obtain

$$\mathbb{E}_k\|\mathbf{x}^{k+1} - \mathbf{x}^*\|_{\mathbf{B}}^2 = \|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{B}}^2 - (\mathbf{x}^k - \mathbf{x}^*)^\top\mathbf{B}^{1/2}(\mathbb{E}[\mathbf{Z}])\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*).$$

By Lemma 3.2.2, the vectors $\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*) \in \text{range}(\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\top) = \text{null}(\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P})^\perp$ for all $k \geq 0$. Since $\text{null}(\mathbb{E}[\mathbf{Z}]) = \text{null}(\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P})$ under the assumption (3.30), we may expand $\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)$ in the orthonormal basis of eigenvectors of $\mathbb{E}[\mathbf{Z}]$ corresponding to positive eigenvalues only, and so

$$(\mathbf{x}^k - \mathbf{x}^*)^\top \mathbf{B}^{1/2}(\mathbb{E}[\mathbf{Z}])\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*) \geq \lambda_{\min}^+(\mathbb{E}[\mathbf{Z}]) \cdot \|\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)\|_2^2.$$

Hence, the decrease in error in the $(k+1)^{\text{st}}$ iteration can be bounded by

$$\mathbb{E}_k\|\mathbf{x}^{k+1} - \mathbf{x}^*\|_{\mathbf{B}}^2 \leq \left(1 - \lambda_{\min}^+(\mathbb{E}[\mathbf{Z}])\right) \cdot \|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{B}}^2.$$

We conclude by iterating and using the tower rule for conditional expectations. □

Remark 3.2.5 (Exactness condition)**.** The assumption (3.30) is a technical condition to ensure that the convergence rate is (strictly) positive. It is similar to an exactness condition from the sketch-and-project literature [RT20; Gow+21; GR15b], which relaxes stronger requirements such as $\mathbf{A}$ having full column rank, and holds for most practical sketching techniques. Intuitively, the exactness condition fails to hold if the distribution of sketching matrices $\mathbf{S} \sim \mathcal{D}$ does not cover the entire space. For example, if the updates are sampled from the same low-rank subspace, then $\mathbf{x}^0 - \mathbf{x}^*$ may have components that are unable to be resolved.

Remark 3.2.6 ($\mathbf{B}$-inner product geometry and oblique projections)**.** The natural geometry of the sketch-and-project method is defined by the $\mathbf{B}$-inner product $\langle \mathbf{x}, \mathbf{y} \rangle_{\mathbf{B}} := \mathbf{x}^\top \mathbf{B}\mathbf{y}$ corresponding to the positive definite $\mathbf{B} \in \mathbb{R}^{n \times n}$. By expressing the projectors $\mathbf{P}$ and $\mathbf{Z}$ from Lemma 3.2.1 in terms of this geometry via a similarity transformation, the update formulas for the subspace-constrained sketch-and-project method admit the following equivalent, more natural expressions. Specifically, let

$$\mathbf{P}_{\mathbf{B}} := \mathbf{B}^{-1/2}\mathbf{P}\mathbf{B}^{1/2} \quad \text{and} \quad \mathbf{Z}_{\mathbf{B}} := \mathbf{B}^{-1/2}\mathbf{Z}\mathbf{B}^{1/2} \tag{3.31}$$

be the *oblique* projection matrices onto null($\mathbf{QA}$) and range($\mathbf{B}^{-1}\mathbf{P}_B^\mathsf{T}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T}$) with respect to the $\mathbf{B}$-norm, respectively.[2] Note that $\mathbf{P_B}\mathbf{B}^{-1}\mathbf{P_B^\mathsf{T}} = \mathbf{P_B}\mathbf{B}^{-1} = \mathbf{B}^{-1}\mathbf{P_B^\mathsf{T}}$ and $\mathbf{P_B}(\mathbf{x}^k - \mathbf{x}^*) = \mathbf{x}^k - \mathbf{x}^*$. Then (3.24) and (3.25) in Lemma 3.2.1 can be written as

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \mathbf{B}^{-1}\mathbf{P_B^\mathsf{T}}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T}(\mathbf{S}\mathbf{A}\mathbf{P_B}\mathbf{B}^{-1}\mathbf{P_B^\mathsf{T}}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T})^\dagger\mathbf{S}(\mathbf{A}\mathbf{x}^k - \mathbf{b}), \tag{3.32}$$

$$\mathbf{x}^{k+1} - \mathbf{x}^* = (\mathbf{I} - \mathbf{Z_B})(\mathbf{x}^k - \mathbf{x}^*). \tag{3.33}$$

These expressions are almost the same as (3.19) and (3.20) for the unconstrained sketch-and-project method using a "new matrix" $\mathbf{AP_B}$ in place of $\mathbf{A}$, and the same sketched residuals $\mathbf{S}(\mathbf{A}\mathbf{x}^k - \mathbf{b})$. Similarly, the error decrease in Lemma 3.2.3 can be written in terms of the $\mathbf{B}$-norm as

$$\|\mathbf{x}^{k+1} - \mathbf{x}^*\|_\mathbf{B}^2 = \|\mathbf{x}^k - \mathbf{x}^*\|_\mathbf{B}^2 - \|\mathbf{Z_B}(\mathbf{x}^k - \mathbf{x}^*)\|_\mathbf{B}^2.$$

Remark 3.2.7 (Connections with the nullspace method)**.** Observe that to solve $\mathbf{A}\mathbf{x} = \mathbf{b}$, we can first solve $\mathbf{Q}\mathbf{A}\mathbf{x}^0 = \mathbf{Q}\mathbf{b}$ for $\mathbf{x}^0$. Then, given $\mathbf{x}^0$, we can (approximately) solve $\mathbf{A}\mathbf{P_B}\mathbf{w} \approx \mathbf{b} - \mathbf{A}\mathbf{x}^0$ for $\mathbf{w} \in \text{range}(\mathbf{P_B}) = \text{null}(\mathbf{QA})$. The overall solution can then be constructed as $\mathbf{x}^1 = \mathbf{x}^0 + \mathbf{w}$, which continues to satisfy the constraints $\mathbf{Q}\mathbf{A}\mathbf{x}^1 = \mathbf{Q}\mathbf{b}$. In our setting, we form $\mathbf{x}^1$ by solving the sketched system $\mathbf{S}\mathbf{A}\mathbf{P_B}\mathbf{w} = \mathbf{S}(\mathbf{b} - \mathbf{A}\mathbf{x}^0)$, and iterate this procedure to obtain the iterates $\{\mathbf{x}^k\}_{k\geq 0}$.

This setup resembles the nullspace approach for solving least squares problems with linear equality constraints (e.g., see [ST22]), and illuminates the observation in Remark 3.2.6 that the subspace-constrained updates parallel the unconstrained updates using a projected version of the matrix, $\mathbf{AP_B}$, in place of $\mathbf{A}$. In our context, the main difference is that the subspace constraint $\mathbf{Q}\mathbf{A}\mathbf{x} = \mathbf{Q}\mathbf{b}$ is not specified by the problem a priori, but generated algorithmically from the data. For our purposes, *the subspace constraint effectively acts as a preconditioner*: ideally, the properties of $\mathbf{AP_B}$ should improve the downstream iterative

[2]Note that if $\mathbf{\Pi}$ is the orthogonal projector onto some subspace $\mathcal{M}$, then $\mathbf{\Pi_B} = \mathbf{B}^{-1/2}\mathbf{\Pi}\mathbf{B}^{1/2}$ is the orthogonal projector onto $\mathcal{M}$ with respect to the $\mathbf{B}$-norm (i.e., $\mathbf{\Pi_B}\mathbf{x} = \arg\min_{\mathbf{y}\in\mathcal{M}}\|\mathbf{x} - \mathbf{y}\|_\mathbf{B}$).

process in a suitable way. For instance, for the SC-RCD method, we will seek $\mathbf{Q}$ such that $\mathbf{AP_B}$ is a good low-rank approximation of $\mathbf{A}$ in trace-norm.

Remark 3.2.8 (Deflated Krylov subspace methods)**.** The subspace constraint framework also resembles the *deflation* technique used to inject spectral information into Krylov subspace solvers [SS07]. The explicit deflation procedure for the conjugate gradient method studied in [Nic87; Saa+00] is most closely related to our approach, especially for solving positive semidefinite linear systems. These works establish that after obtaining an initial iterate in an approximate eigenspace, the conjugate gradient method can also be preconditioned by restricting onto the remaining subspace, and the resulting algorithm enjoys similar recurrences. The main difference of these methods, compared with the subspace-constrained sketch-and-project algorithm that we analyze, is in the underlying projection mechanism (i.e., projection onto a Krylov subspace vs. projection onto a sketched linear system $\mathbf{SAx} = \mathbf{Sb}$ with respect to $\|\cdot\|_{\mathbf{B}}$). Later, for the SC-RCD method, we also focus on the computational aspects of learning a good subspace from the data via a connection to low-rank matrix approximation.

3.2.3 Convergence rate and block size

In many cases, $\mathbb{E}[\mathbf{Z}]$ is difficult to compute or even estimate. However, the expected projection matrix $\mathbb{E}[\mathbf{Z}_1]$ corresponding to the sketching matrix $\mathbf{S}_1$ with a single row is often exactly computable.

In this section, we consider the special class of sketching matrices where $\mathbf{S} \in \mathbb{R}^{\ell \times m}$ consists of ℓ independent and identically distributed rows $\mathbf{s}_1, \ldots, \mathbf{s}_\ell \in \mathbb{R}^m$. This encompasses a range of practical sketching schemes, including block versions of the randomized Kaczmarz [SV09; NT14] and coordinate descent [LL10] methods. The following result shows that in this setting, the iteration complexity improves at least linearly in the block size ℓ compared to the single-row case; i.e., if using $\mathbf{S}_1 = \mathbf{s}_1^\mathsf{T}$ requires τ iterations to reach a desired tolerance in mean squared error, then using $\mathbf{S}$ requires at most τ/ℓ iterations.

Proposition 3.2.9. *Suppose that* $\mathbf{S}_\ell \in \mathbb{R}^{\ell\times m}$ *is a matrix with i.i.d. random rows* $\mathbf{s}_1,\dots,\mathbf{s}_\ell \in \mathbb{R}^m$*, and let* $\mathbf{Z}_1 = (\mathbf{s}_1^\mathsf{T}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P})^\dagger \mathbf{s}_1^\mathsf{T}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P}$ *be the orthogonal projector onto* $\operatorname{range}(\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}_1^\mathsf{T})$*. Assume that* $\operatorname{null}(\mathbb{E}[\mathbf{Z}_1]) = \operatorname{null}(\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P})$*. If* $\{\mathbf{x}^k\}_{k\geq 1}$ *are the subspace-constrained sketch-and-project iterates with sketching matrix* $\mathbf{S}_\ell$*, then*

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{B}}^2 \leq \left(1 - \lambda_{\min}^+(\mathbb{E}[\mathbf{Z}_1])\right)^{\ell k} \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_{\mathbf{B}}^2. \tag{3.34}$$

The key technical ingredient behind Proposition 3.2.9 is the following rank-one update formula for orthogonal projection matrices from [Der+20], which is derived using a rank-one update formula for the Moore–Penrose pseudoinverse ([Mey73, Theorem 1]).

Lemma 3.2.10 ([Der+20, Lemma 1])**.** *Let* $\mathbf{X} \in \mathbb{R}^{t\times n}$ *and* $\mathbf{X}_{-t} \in \mathbb{R}^{(t-1)\times n}$ *be the matrix* $\mathbf{X}$ *without its last row* $\mathbf{x}_t \in \mathbb{R}^n$*. Suppose that* $\mathbf{\Pi} = \mathbf{X}^\dagger\mathbf{X}$ *and* $\mathbf{\Pi}_{-t} = \mathbf{X}_{-t}^\dagger\mathbf{X}_{-t}$ *are the orthogonal projectors onto the ranges of* $\mathbf{X}^\mathsf{T}$ *and* $\mathbf{X}_{-t}^\mathsf{T}$*, respectively. If* $\mathbf{x}_t^\mathsf{T}(\mathbf{I} - \mathbf{\Pi}_{-t})\mathbf{x}_t \neq \mathbf{0}$*, then*

$$\mathbf{\Pi} - \mathbf{\Pi}_{-t} = \frac{(\mathbf{I} - \mathbf{\Pi}_{-t})\mathbf{x}_t\mathbf{x}_t^\mathsf{T}(\mathbf{I} - \mathbf{\Pi}_{-t})}{\mathbf{x}_t^\mathsf{T}(\mathbf{I} - \mathbf{\Pi}_{-t})\mathbf{x}_t}.$$

Otherwise, if $\mathbf{x}_t^\mathsf{T}(\mathbf{I} - \mathbf{\Pi}_{-t})\mathbf{x}_t = 0$*, then* $\mathbf{x}_t \in \operatorname{range}(\mathbf{X}_{-t}^\mathsf{T})$ *and* $\mathbf{\Pi} = \mathbf{\Pi}_{-t}$*, so the decomposition above also holds provided that the right hand side is interpreted as zero.*

We can now prove Proposition 3.2.9 using Lemma 3.2.10 to decompose the orthogonal projector $\mathbf{Z}$ into a sum of rank-one projections onto a growing sequence of subspaces.

Proof of Proposition 3.2.9. It will be useful to define the sketching matrices for all intermediate block sizes. For $1 \leq t \leq \ell$, let $\mathbf{S}_t \in \mathbb{R}^{t\times m}$ be the matrix with rows $\mathbf{s}_1,\dots,\mathbf{s}_t$, and $\mathbf{X}_t := \mathbf{S}_t\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P} \in \mathbb{R}^{t\times n}$ be the matrix with rows $\mathbf{x}_1,\dots,\mathbf{x}_t$, where $\mathbf{x}_i := \mathbf{s}_i^\mathsf{T}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P}$. Furthermore, let $\mathbf{Z}_t = \mathbf{X}_t^\dagger\mathbf{X}_t$ be the orthogonal projector onto the range of $\mathbf{X}_t^\mathsf{T}$ (with $\mathbf{Z}_0 := \mathbf{0}$). In particular, we get the following representation

$$\mathbf{X}_1^\dagger\mathbf{X}_1 = \frac{\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}\mathbf{s}_1\mathbf{s}_1^\mathsf{T}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P}}{\mathbf{s}_1^\mathsf{T}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}\mathbf{s}_1}. \tag{3.35}$$

Next, observe that the exactness condition $\mathrm{null}(\mathbb{E}[\mathbf{Z}_t]) = \mathrm{null}(\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P})$ holds for all $\mathbf{Z}_t$. Indeed, $\mathbf{Z}_t \succeq \mathbf{Z}_1$ and by the exactness assumption on $\mathbf{Z}_1$,

$$\mathrm{null}(\mathbb{E}[\mathbf{Z}_t]) \subseteq \mathrm{null}(\mathbb{E}[\mathbf{Z}_1]) = \mathrm{null}(\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P}) \subseteq \mathrm{null}(\mathbb{E}[\mathbf{Z}_t]).$$

Hence, all the inclusions are equalities.

We claim that for all $1 \le t \le \ell$ and unit vectors $\mathbf{y} \in \mathrm{range}(\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T})$,

$$\mathbf{y}^\mathsf{T}\mathbb{E}[\mathbf{Z}_t]\mathbf{y} \ge 1 - \left(1 - \lambda_{\min}^+(\mathbb{E}[\mathbf{Z}_1])\right)^t. \tag{3.36}$$

We will prove (3.36) by induction on t. When $t = 1$, the base case $\mathbf{y}^\mathsf{T}\mathbb{E}[\mathbf{Z}_1]\mathbf{y} \ge \lambda_{\min}^+(\mathbb{E}[\mathbf{Z}_1])$ follows because $\mathbf{y} \in \mathrm{range}(\mathbb{E}[\mathbf{Z}_1])$. Assuming that (3.36) holds for $t-1 < \ell$, we want to prove that (3.36) holds for t. Using Lemma 3.2.10, we have the following decomposition of $\mathbf{Z}_t$:

$$\mathbf{Z}_t = \mathbf{Z}_{t-1} + (\mathbf{Z}_t - \mathbf{Z}_{t-1}) = \mathbf{Z}_{t-1} + \frac{(\mathbf{I} - \mathbf{Z}_{t-1})\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}\mathbf{s}_t\mathbf{s}_t^\mathsf{T}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P}(\mathbf{I} - \mathbf{Z}_{t-1})}{\mathbf{s}_t^\mathsf{T}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P}(\mathbf{I} - \mathbf{Z}_{t-1})\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}\mathbf{s}_t}. \tag{3.37}$$

Observe that $\mathbf{s}_t^\mathsf{T}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P}(\mathbf{I} - \mathbf{Z}_{t-1})\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}\mathbf{s}_t \le \mathbf{s}_t^\mathsf{T}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}\mathbf{s}_t$. Hence, by taking expectation over the randomness in $\mathbf{s}_t$ only, conditional on $\mathbf{s}_1, \ldots, \mathbf{s}_{t-1}$, and using linearity of expectation, we obtain

$$\begin{aligned}
&\mathbf{y}^\mathsf{T}\mathbb{E}_{\mathbf{s}_t|\mathbf{s}_1,\ldots,\mathbf{s}_{t-1}}[\mathbf{Z}_t]\mathbf{y} \\
&\quad \ge \mathbf{y}^\mathsf{T}\mathbf{Z}_{t-1}\mathbf{y} + \mathbf{y}^\mathsf{T}(\mathbf{I} - \mathbf{Z}_{t-1})\mathbb{E}_{\mathbf{s}_t|\mathbf{s}_1,\ldots,\mathbf{s}_{t-1}}\left[\frac{\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}\mathbf{s}_t\mathbf{s}_t^\mathsf{T}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P}}{\mathbf{s}_t^\mathsf{T}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}\mathbf{s}_t}\right](\mathbf{I} - \mathbf{Z}_{t-1})\mathbf{y} \\
&\quad = \mathbf{y}^\mathsf{T}\mathbf{Z}_{t-1}\mathbf{y} + \mathbf{y}^\mathsf{T}(\mathbf{I} - \mathbf{Z}_{t-1})\mathbb{E}[\mathbf{Z}_1](\mathbf{I} - \mathbf{Z}_{t-1})\mathbf{y}.
\end{aligned}$$

The last equality follows from comparing the random matrix inside the inner expectation to (3.35) and using the fact that $\mathbf{s}_t$ has the same distribution as $\mathbf{s}_1$. Therefore, since $(\mathbf{I} - \mathbf{Z}_{t-1})\mathbf{y} \in \mathrm{range}(\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T})$ by definition of $\mathbf{Z}_t$, and $\mathbf{Z}_{t-1}$ is an orthogonal projector, this

implies that

$$\mathbf{y}^\mathsf{T}\mathbb{E}_{\mathbf{s}_t|\mathbf{s}_1,\dots,\mathbf{s}_{t-1}}[\mathbf{Z}_t]\mathbf{y} \geq \mathbf{y}^\mathsf{T}\mathbf{Z}_{t-1}\mathbf{y} + \lambda_{\min}^+(\mathbb{E}[\mathbf{Z}_1]) \cdot \mathbf{y}^\mathsf{T}(\mathbf{I} - \mathbf{Z}_{t-1})\mathbf{y}.$$

By taking the full expectation and using the induction hypothesis, we deduce that

$$\begin{aligned}\mathbf{y}^\mathsf{T}\mathbb{E}[\mathbf{Z}_t]\mathbf{y} &\geq \lambda_{\min}^+(\mathbb{E}[\mathbf{Z}_1]) + \big(1 - \lambda_{\min}^+(\mathbb{E}[\mathbf{Z}_1])\big) \cdot \mathbf{y}^\mathsf{T}\mathbb{E}[\mathbf{Z}_{t-1}]\mathbf{y} \\ &\geq \lambda_{\min}^+(\mathbb{E}[\mathbf{Z}_1]) + \big(1 - \lambda_{\min}^+(\mathbb{E}[\mathbf{Z}_1])\big) \cdot \big(1 - \big(1 - \lambda_{\min}^+(\mathbb{E}[\mathbf{Z}_1])\big)^{t-1}\big) \\ &= 1 - \big(1 - \lambda_{\min}^+(\mathbb{E}[\mathbf{Z}_1])\big)^t,\end{aligned}$$

and so, $1 - \lambda_{\min}^+(\mathbb{E}[\mathbf{Z}_\ell]) \leq \big(1 - \lambda_{\min}^+(\mathbb{E}[\mathbf{Z}_1])\big)^\ell$. Employing Theorem 3.2.4 completes the proof. □

3.2.4 Example: randomized block Kaczmarz

The subspace-constrained sketch-and-project method generalizes the *subspace-constrained randomized Kaczmarz* (SC-RK) method analyzed in [LR24], which can be described as follows. Let $\mathbf{B} = \mathbf{I}$ and $\mathbf{Q} \in \mathbb{R}^{d\times m}$ be a matrix parameter so that the iterates are confined within the solution space of the sketched system $\mathbf{QAx} = \mathbf{Qb}$, and $\mathbf{P} = \mathbf{I} - (\mathbf{QA})^\dagger\mathbf{QA}$ be the orthogonal projector onto null($\mathbf{QA}$). From Lemma 3.2.1, the updates of the SC-RK method are given by

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \frac{\mathbf{a}_j^\mathsf{T}\mathbf{x}^k - \mathbf{b}_j}{\|\mathbf{Pa}_j\|_2^2}\mathbf{Pa}_j.$$

Suppose that each row $\mathbf{a}_j^\mathsf{T} = \mathbf{A}_{j,:}$ of the matrix $\mathbf{A}$ is independently sampled with probability $\|\mathbf{Pa}_j\|_2^2/\|\mathbf{AP}\|_F^2$ in each iteration. Since $\mathbf{Z} = \mathbf{Pa}_j\mathbf{a}_j^\mathsf{T}\mathbf{P}/\|\mathbf{Pa}_j\|_2^2$ if row j is sampled, it follows that $\mathbb{E}[\mathbf{Z}] = \mathbf{PA}^\mathsf{T}\mathbf{AP}/\|\mathbf{AP}\|_F^2$. Note that the exactness condition (3.30) holds trivially because null($\mathbf{PA}^\mathsf{T}\mathbf{AP}$) = null($\mathbf{AP}$).

The SC-RK method studied in [LR24] uses a subspace constraint parameter $\mathbf{Q} = \mathbf{e}_\mathcal{S}^\mathsf{T}$ aligned with the standard coordinate basis for some set of coordinates $\mathcal{S} \subseteq [m]$. Thus, the

initial iterate is required to satisfy the subsystem $\mathbf{A}_{\mathcal{S},:}\mathbf{x} = \mathbf{b}_{\mathcal{S}}$, and $\mathbf{P} = \mathbf{I} - (\mathbf{A}_{\mathcal{S},:})^{\dagger}\mathbf{A}_{\mathcal{S},:}$ is the orthogonal projector onto $\mathrm{null}(\mathbf{A}_{\mathcal{S},:})$.

More generally, if a block $\mathcal{J} \subseteq [m]$ of $\ell \geq 1$ rows are sampled independently as above, then the block update

$$\mathbf{x}^{k+1} = \mathbf{x}^k - (\mathbf{A}_{\mathcal{J},:}\mathbf{P})^{\dagger}(\mathbf{A}_{\mathcal{J},:}\mathbf{x}^k - \mathbf{b}_{\mathcal{J}}). \tag{3.38}$$

describes a subspace-constrained *block* randomized Kaczmarz method. From Proposition 3.2.9, the following convergence rate bound holds:

Corollary 3.2.11. *Let $\{\mathbf{x}^k\}_{k\geq 0}$ be the iterates defined by (3.38), where $\mathbf{x}^0$ solves $\mathbf{QAx} = \mathbf{Qb}$ and the block $\mathcal{J} = \{j_1, \ldots, j_\ell\}$ in each iteration consists of ℓ rows independently sampled from the distribution $\{\|\mathbf{Pa}_j\|_2^2/\|\mathbf{AP}\|_F^2\}_{j=1}^m$. Then*

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|_2^2 \leq \left(1 - \frac{\sigma_{\min}^+(\mathbf{AP})^2}{\|\mathbf{AP}\|_F^2}\right)^{\ell k} \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_2^2,$$

where $\sigma_{\min}^+(\mathbf{AP})^2 = \lambda_{\min}^+(\mathbf{PA}^{\mathsf{T}}\mathbf{AP})$ is the smallest non-zero squared singular value of $\mathbf{AP}$.

By taking $\mathbf{Q} = \mathbf{0}$ and thus $\mathbf{P} = \mathbf{I}$ (i.e., no subspace constraint), Corollary 3.2.11 implies Proposition 3.1.9 for the convergence rate of the randomized block Kaczmarz method.

Randomized rangefinder. Besides allowing for a block generalization of the SC-RK method from [LR24], the formulation of the subspace constraint $\mathbf{QAx} = \mathbf{Qb}$ allows for a matrix parameter $\mathbf{Q}$ that is not aligned with the standard coordinate basis. This arrangement allows for the subspace to be learned using more general low-rank approximation techniques beyond row selection, which was explored in [LR24] using leverage score sampling. The desirability of this additional flexibility is motivated by the observation that using a random embedding that mixes different rows often produces a higher quality approximation than one based purely on coordinate sampling [MT20, §9.6].

A particularly noteworthy example that we will discuss is the *randomized rangefinder* procedure, which underlies the celebrated randomized SVD algorithm introduced by Halko,

Martinsson and Tropp [HMT11]. A prototype algorithm of the randomized rangefinder for computing an approximate basis of the row space of $\mathbf{A} \in \mathbb{R}^{m \times n}$ can be described as follows:

1. Draw a random Gaussian matrix $\mathbf{\Omega} \in \mathbb{R}^{d \times m}$ with i.i.d. $N(0,1)$ entries.

2. Compute the matrix $\mathbf{\Omega A} \in \mathbb{R}^{d \times n}$

3. Construct a matrix $\mathbf{U} \in \mathbb{R}^{n \times d}$ whose columns form an orthonormal basis for the row space of $\mathbf{\Omega A}$, e.g., by computing a QR decomposition of its transpose $(\mathbf{\Omega A})^{\mathsf{T}} = \mathbf{UR}$.

The prototype randomized rangefinder algorithm can be implemented in $O(dmn + d^2 n)$ arithmetic operations, which is typically dominated by the dense matrix multiplication step. When $\mathbf{\Omega} \in \mathbb{R}^{d \times m}$ is Gaussian, techniques from random matrix theory can be used to prove that the procedure produces a near-optimal rank-r approximation for some $r \approx d$. More precisely, [HMT11] proved the following result (after some simplifications for interpretability) on the approximation quality in terms of the Frobenius norm, both in expectation and with very high probability:

Theorem 3.2.12 ([HMT11, Theorems 10.5 and 10.7])**.** *Let $\sigma_1(\mathbf{A}) \geq \sigma_2(\mathbf{A}) \geq \ldots$ be the singular values of $\mathbf{A} \in \mathbb{R}^{m \times n}$. Given a target rank $r \in \mathbb{N}$ and a sketch size $d \in \mathbb{N}$ with $d \geq r + 4$, let $\mathbf{\Omega} \in \mathbb{R}^{d \times m}$ be a random Gaussian matrix and $\mathbf{U} \in \mathbb{R}^{n \times d}$ be a matrix whose columns contain an orthonormal basis of $(\mathbf{\Omega A})^{\mathsf{T}}$. Then*

$$\mathbb{E}\|\mathbf{A} - \mathbf{AUU}^{\mathsf{T}}\|_F^2 \leq \left(1 + \frac{r}{d - r - 1}\right) \cdot \sum_{i > r} \sigma_i(\mathbf{A})^2.$$

Furthermore, the following event holds with probability at least $1 - 7e^{-(d-r)}$:

$$\|\mathbf{A} - \mathbf{AUU}^{\mathsf{T}}\|_F \leq \left(1 + e\sqrt{12}\sqrt{\frac{r}{d-r}}\right) \cdot \left(\sum_{i>r} \sigma_i(\mathbf{A})^2\right)^{1/2} + e^2\sqrt{2}\sqrt{\frac{d}{d-r+1}} \cdot \sigma_{r+1}(\mathbf{A}).$$

For example, if we oversample by a multiplicative factor of two, i.e., choose a sketch size $d = 2r$ for a target rank $r \geq 2$, then Theorem 3.2.12 implies that $\mathbb{E}\|\mathbf{A} - \mathbf{AUU}^{\mathsf{T}}\|_F^2 \leq$

$3\sum_{i>r}\sigma_i(\mathbf{A})^2$. Moreover, since $\sigma_{r+1} \leq \left(\sum_{i>r}\sigma_i(\mathbf{A})^2\right)^{1/2}$, it implies that with probability at least $1-7e^{-r}$, $\|\mathbf{A}-\mathbf{A}\mathbf{U}\mathbf{U}^\mathsf{T}\|_F^2 \leq 635\sum_{i>r}\sigma_i(\mathbf{A})^2$. (Note that we did not attempt to optimize the constants as the goal is only to highlight the comparison with the best rank-r error.) In practice, oversampling by a small constant amount (e.g., $d = r + 10$) is often enough to obtain an excellent approximation. Furthermore, for matrices with slow spectral decay, the randomized rangefinder can be combined with power iteration to significantly improve the approximation quality [HMT11; TW23]

Currently, using a dense Gaussian sketch $\mathbf{\Omega}$ is viewed to be the gold standard in terms of the approximation quality, and allows for rigorous a priori error control (Theorem 3.2.12). In practice, more practical sketching matrices $\mathbf{\Omega}$ can be used with very similar performance, including fast transforms (e.g., the subsampled randomized Hadamard transform) [AC06; Woo+08] or sparse embeddings [AM07; Cam+25]. For a more thorough discussion of related computational aspects (e.g., a posteriori error estimation, and variations of the algorithm for different environments), we refer to the excellent survey [MT20].

Once we have generated the random embedding $\mathbf{\Omega}$, computed $\mathbf{\Omega}\mathbf{A}$ (which we may assume to be full rank for simplicity), and constructed a basis $\mathbf{U}$ for its row space, the idea is that we can run the SC-RK algorithm confined within the affine subspace $\mathbf{\Omega}\mathbf{A}\mathbf{x} = \mathbf{\Omega}\mathbf{b}$. This directly fits into the subspace-constrained sketch-and-project framework with $\mathbf{Q} = \mathbf{\Omega}$. Recall that $\mathbf{P} = \mathbf{I} - (\mathbf{\Omega}\mathbf{A})^\dagger\mathbf{\Omega}\mathbf{A} = \mathbf{I} - \mathbf{U}\mathbf{U}^\mathsf{T}\mathbf{A}$ is the orthogonal projector onto $\mathrm{null}(\mathbf{\Omega}\mathbf{A}) = \mathrm{range}(\mathbf{A}^\mathsf{T}\mathbf{\Omega}^\mathsf{T})$. Thus, each (block) SC-RK update (3.38) can be implemented as

$$\mathbf{x}^{k+1} = \mathbf{x}^k - (\mathbf{A}_{\mathcal{J},:} - \mathbf{A}_{\mathcal{J},:}\mathbf{U}\mathbf{U}^\mathsf{T})^\dagger(\mathbf{A}_{\mathcal{J},:}\mathbf{x}^k - \mathbf{b}_\mathcal{J}). \tag{3.39}$$

Compared to randomized bock Kaczmarz, each iteration (3.39) costs an additional $O(\ell nd)$ arithmetic operations to compute the projection of the block $\mathbf{A}_{\mathcal{J},:}$.

By combining Corollary 3.2.11 with the high probability control of $\|\mathbf{A}\mathbf{P}\|_F^2$ from Theorem 3.2.12, we can formulate the following bound on the convergence rate of the (block) SC-

RK method with a rank-d subspace constraint computed using the randomized rangefinder. Compared with Proposition 3.1.9, this result shows that the denominator in the convergence rate improves from $\|\mathbf{A}\|_F^2 = \sum_{i \geq 1} \sigma_i(\mathbf{A})^2$ to a constant multiple of $\sum_{i > \lfloor d/2 \rfloor} \sigma_i(\mathbf{A})^2$. If $\mathbf{A}$ is approximately low-rank, then the latter can be much smaller for some $d \ll n$, which implies a significant improvement in the convergence rate with a relatively small increase in the per-iteration computational cost.

Proposition 3.2.13. *Let $\boldsymbol{\Omega} \in \mathbb{R}^{d \times m}$ be a random Gaussian matrix with $d \geq 10$, and $\mathbf{U} \in \mathbb{R}^{n \times d}$ be a matrix whose columns contain an orthonormal basis of $(\boldsymbol{\Omega}\mathbf{A})^\mathsf{T}$. Let $\{\mathbf{x}^k\}_{k \geq 0}$ be the iterates defined by (3.39), where $\mathbf{x}^0$ solves $\boldsymbol{\Omega}\mathbf{A}\mathbf{x} = \boldsymbol{\Omega}\mathbf{b}$ and the block $\mathcal{J} = \{j_1, \ldots, j_\ell\}$ in each iteration consists of ℓ rows independently sampled from the distribution $\{\|\mathbf{a}_j - \mathbf{U}\mathbf{U}^\mathsf{T}\mathbf{a}_j\|_2^2 / \|\mathbf{A} - \mathbf{A}\mathbf{U}\mathbf{U}^\mathsf{T}\|_F^2\}_{j=1}^m$. Then, conditional on an event $\mathcal{E}$ that occurs with probability at least $1 - 7e^{-\lfloor d/2 \rfloor}$,*

$$\mathbb{E}[\|\mathbf{x}^k - \mathbf{x}^*\|_2^2 \mid \mathcal{E}] \leq \left(1 - \frac{\sigma_{\min}^+(\mathbf{A})^2}{635 \sum_{i > \lfloor d/2 \rfloor} \sigma_i(\mathbf{A})^2}\right)^{\ell k} \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_2^2,$$

where $\sigma_{\min}^+(\mathbf{A})$ is the smallest non-zero singular value of $\mathbf{A}$.

Proof. Let $\mathcal{E}$ be the event that $\|\mathbf{A}\mathbf{P}\|_F^2 = \|\mathbf{A} - \mathbf{A}\mathbf{U}\mathbf{U}^\mathsf{T}\|_F^2 \leq 635 \sum_{i > d/2} \sigma_i(\mathbf{A})^2$, which occurs with probability at least $1 - 7e^{-\lfloor d/2 \rfloor}$ by Theorem 3.2.12 with $r = \lfloor d/2 \rfloor$. Next, note that the smallest non-zero singular value of $\mathbf{A}\mathbf{P}$ is always as large as the smallest non-zero singular value of $\mathbf{A}$. This can be shown by the variational principle, using the fact that $\mathrm{null}(\mathbf{A}\mathbf{P})^\perp = \mathrm{range}(\mathbf{P}\mathbf{A}^\mathsf{T}) \subseteq \mathrm{range}(\mathbf{P})$ and $\mathrm{range}(\mathbf{P}\mathbf{A}^\mathsf{T}) \subseteq \mathrm{range}(\mathbf{A}^\mathsf{T})$:

$$\sigma_{\min}^+(\mathbf{A}\mathbf{P}) = \min_{\mathbf{x} \in \mathrm{range}(\mathbf{P}\mathbf{A}^\mathsf{T})} \frac{\|\mathbf{A}\mathbf{P}\mathbf{x}\|_2}{\|\mathbf{x}\|_2} = \min_{\mathbf{x} \in \mathrm{range}(\mathbf{P}\mathbf{A}^\mathsf{T})} \frac{\|\mathbf{A}\mathbf{x}\|_2}{\|\mathbf{x}\|_2} \geq \min_{\mathbf{x} \in \mathrm{range}(\mathbf{A}^\mathsf{T})} \frac{\|\mathbf{A}\mathbf{x}\|_2}{\|\mathbf{x}\|_2} = \sigma_{\min}^+(\mathbf{A}).$$

Therefore, conditional on the event $\mathcal{E}$, we can apply the upper and lower bounds on $\|\mathbf{A}\mathbf{P}\|_F^2$ and $\sigma_{\min}^+(\mathbf{A}\mathbf{P})^2$ from above to the bound on the SC-RK error from Corollary 3.2.11, which shows that the claimed inequality holds. □

3.3 Analysis of subspace-constrained randomized coordinate descent (SC-RCD)

In this section, we analyze the subspace-constrained randomized coordinate descent (SC-RCD) method for solving the linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$, where $\mathbf{A} \in \mathbb{R}^{n\times n}$ is a positive semidefinite matrix.

3.3.1 Convergence rate of SC-RCD

Update formulas. In the case when $\mathbf{A}$ is also positive definite, the SC-RCD method is an instance of the subspace-constrained sketch-and-project method with geometry parameter $\mathbf{B} = \mathbf{A}$ and sparse sketching matrices aligned with the standard coordinate basis of the form $\mathbf{S} = \mathbf{e}_{\mathcal{J}}^{\mathsf{T}}$, defined as in (3.3), for randomly chosen subsets $\mathcal{J} \subseteq [n]$. Furthermore, the subspace constraint is defined by the matrix $\mathbf{Q} = \mathbf{e}_{\mathcal{S}}^{\mathsf{T}} \in \mathbb{R}^{d\times n}$, parameterized by a (small) subset $\mathcal{S} \subseteq [n]$ of $d \ll n$ coordinates representing d salient data points or landmarks, which we propose to efficiently choose using an algorithm such as RPCholesky.

Recall that $\mathbf{x}^0 \in \mathbb{R}^n$ is any initial iterate satisfying $\mathbf{A}_{\mathcal{S},:}\mathbf{x}^0 = \mathbf{b}_{\mathcal{S}}$. By Lemma 3.2.1, the update formula of the SC-RCD method is given by

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \mathbf{A}^{-1/2}\mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_J(\mathbf{e}_J^{\mathsf{T}}\mathbf{A}^{1/2}\mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_J)^{\dagger}(\mathbf{A}_{J,:}\mathbf{x}^k - \mathbf{b}_J), \tag{3.40}$$

where $\mathbf{P} = \mathbf{I} - \mathbf{A}^{1/2}\mathbf{e}_{\mathcal{S}}(\mathbf{e}_{\mathcal{S}}^{\mathsf{T}}\mathbf{A}\mathbf{e}_{\mathcal{S}})^{\dagger}\mathbf{e}_{\mathcal{S}}^{\mathsf{T}}\mathbf{A}^{1/2}$ is the orthogonal projector onto $\mathrm{null}(\mathbf{e}_{\mathcal{S}}^{\mathsf{T}}\mathbf{A}^{1/2})$.

Note that the rank-d Nyström approximation of $\mathbf{A}$ with respect to the coordinates indexed by $\mathcal{S}$ (e.g., see [MT20, §19.2], [Che+25, §2.1]) can be written as

$$\mathbf{A}\langle\mathcal{S}\rangle = \mathbf{A}_{:,\mathcal{S}}(\mathbf{A}_{\mathcal{S},\mathcal{S}})^{\dagger}\mathbf{A}_{\mathcal{S},:}. \tag{3.41}$$

Using this expression, we can make the key observation that the residual matrix satisfies

$$\mathbf{A}^{\circ} = \mathbf{A} - \mathbf{A}\langle \mathcal{S} \rangle = \mathbf{A}^{1/2}\mathbf{P}\mathbf{A}^{1/2}. \tag{3.42}$$

Thus, the update formula (3.40) can be written as

$$\begin{aligned} \mathbf{x}^{k+1} &= \mathbf{x}^k - (\mathbf{e}_{\mathcal{J}} - \mathbf{e}_{\mathcal{S}}(\mathbf{A}_{\mathcal{S},\mathcal{S}})^{\dagger}\mathbf{A}_{\mathcal{S},:}\mathbf{e}_{\mathcal{J}})(\mathbf{A}^{\circ}_{\mathcal{J},\mathcal{J}})^{\dagger}(\mathbf{A}_{\mathcal{J},:}\mathbf{x}^k - \mathbf{b}_{\mathcal{J}}) \\ &= \mathbf{x}^k - \mathbf{e}_{\mathcal{J}}\boldsymbol{\alpha}^k + \mathbf{e}_{\mathcal{S}}\boldsymbol{\beta}^k, \end{aligned} \tag{3.43}$$

where, with the corresponding residual vector denoted by $\mathbf{r}^k = \mathbf{A}\mathbf{x}^k - \mathbf{b}$,

$$\boldsymbol{\alpha}^k := (\mathbf{A}^{\circ}_{\mathcal{J},\mathcal{J}})^{\dagger}\mathbf{r}^k_{\mathcal{J}}, \quad \boldsymbol{\beta}^k := \mathbf{C}_{:,\mathcal{J}}\boldsymbol{\alpha}^k, \quad \text{and} \quad \mathbf{C} := (\mathbf{A}_{\mathcal{S},\mathcal{S}})^{\dagger}\mathbf{A}_{\mathcal{S},:}.$$

Note that the formula (3.43) remains well-defined even if $\mathbf{A}$ is rank deficient. Hence, the SC-RCD method can be defined directly through the update formula (3.43), and remains well-defined in the general positive semidefinite case.

With the update (3.43) for the iterate $\mathbf{x}^k$, the following update for the residual vector $\mathbf{r}^{k+1} = \mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}$ can be derived:

$$\begin{aligned} \mathbf{r}^{k+1} &= (\mathbf{A}\mathbf{x}^k - \mathbf{b}) - \mathbf{A}^{\circ}_{:,\mathcal{J}}(\mathbf{A}^{\circ}_{\mathcal{J},\mathcal{J}})^{\dagger}(\mathbf{A}_{\mathcal{J},:}\mathbf{x}^k - \mathbf{b}_{\mathcal{J}}) \\ &= \mathbf{r}^k - \mathbf{A}^{\circ}_{:,\mathcal{J}}(\mathbf{A}^{\circ}_{\mathcal{J},\mathcal{J}})^{\dagger}\mathbf{r}^k_{\mathcal{J}} \\ &= \mathbf{r}^k - \mathbf{A}^{\circ}_{:,\mathcal{J}}\boldsymbol{\alpha}^k. \end{aligned} \tag{3.44}$$

Finally, using (3.42), the orthogonal projector $\mathbf{Z}$ onto range$(\mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_{\mathcal{J}})$ can be written

$$\begin{aligned} \mathbf{Z} &= \mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_{\mathcal{J}}(\mathbf{e}_{\mathcal{J}}^{\mathsf{T}}\mathbf{A}^{1/2}\mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_{\mathcal{J}})^{\dagger}\mathbf{e}_{\mathcal{J}}^{\mathsf{T}}\mathbf{A}^{1/2}\mathbf{P} \\ &= \mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_{\mathcal{J}}(\mathbf{A}^{\circ}_{\mathcal{J},\mathcal{J}})^{\dagger}\mathbf{e}_{\mathcal{J}}^{\mathsf{T}}\mathbf{A}^{1/2}\mathbf{P}. \end{aligned} \tag{3.45}$$

Remark 3.3.1 (Properties of the Nyström approximation)**.** The residual matrix $\mathbf{A}^\circ = \mathbf{A} - \mathbf{A}\langle\mathcal{S}\rangle$ is the (generalized) *Schur complement* of $\mathbf{A}$ with respect to the coordinates $\mathcal{S} \subseteq [n]$ (e.g., see [HZ05]). From this observation, the following well-known properties of the Nyström approximation $\mathbf{A}\langle\mathcal{S}\rangle$ can be derived. (i) $\mathbf{A}^\circ \succeq \mathbf{0}$, so $\mathbf{A} \succeq \mathbf{A}\langle\mathcal{S}\rangle \succeq \mathbf{0}$. (ii) The columns indexed by $\mathcal{S}$ in $\mathbf{A}\langle\mathcal{S}\rangle$ and $\mathbf{A}$ are equal: $(\mathbf{A}\langle\mathcal{S}\rangle)_{:,\mathcal{S}} = \mathbf{A}_{:,\mathcal{S}}$. (iii) The range of $\mathbf{A}\langle\mathcal{S}\rangle$ coincides with the span of the columns $\mathcal{S}$ in $\mathbf{A}$: $\operatorname{range}(\mathbf{A}\langle\mathcal{S}\rangle) = \operatorname{range}(\mathbf{A}_{:,\mathcal{S}})$. (iv) If the $d \times d$ block $(\mathbf{A}\langle\mathcal{S}\rangle)_{\mathcal{S},\mathcal{S}}$ is invertible, then $\operatorname{rank}(\mathbf{A}^\circ) = \operatorname{rank}(\mathbf{A}) - d$ ([HZ05, Theorem 1.6]), and the eigenvalues of $\mathbf{A}^\circ$ interlace those of $\mathbf{A}$: $\lambda_i(\mathbf{A}) \geq \lambda_i(\mathbf{A}^\circ) \geq \lambda_{i+d}(\mathbf{A})$ for $1 \leq i \leq n-d$ ([Liu05, Theorem 2.1]).[3] (v) Finally, if $\mathbf{A}\langle\mathcal{S}\rangle$ is computed using RPCholesky, then $(\mathbf{A}\langle\mathcal{S}\rangle)_{\mathcal{S},\mathcal{S}}$ is invertible by virtue of the adaptive diagonal sampling process for the pivots.

Convergence rate. In the positive definite case, if the block $\mathcal{J} \subseteq [n]$ is formed by the simple scheme of sampling ℓ coordinates independently from the same distribution, the convergence rate of the SC-RCD method can be deduced from the general theory for the subspace-constrained sketch-and-project method (Theorem 3.2.4 and Proposition 3.2.9). The following result obtains the resulting rate when the coordinates are sampled proportionally to the diagonal of the residual matrix $\mathbf{A}^\circ$:

Theorem 3.3.2 (Diagonal sampling)**.** *Let $\mathbf{A} \in \mathbb{R}^{n\times n}$ be a positive semidefinite matrix and $\mathbf{x}^*$ be any solution of $\mathbf{A}\mathbf{x} = \mathbf{b}$. Suppose that $\{\mathbf{x}^k\}_{k\geq 0}$ are the iterates defined by (3.43) with a fixed subset $\mathcal{S} \subseteq [n]$, and the block $\mathcal{J} = \{j_1, \ldots, j_\ell\}$ in each iteration consists of ℓ coordinates independently sampled according to the distribution $\operatorname{diag}(\mathbf{A}^\circ)/\operatorname{tr}(\mathbf{A}^\circ)$. Then*

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{A}}^2 \leq \left(1 - \frac{\lambda_{\min}^+(\mathbf{A}^\circ)}{\operatorname{tr}(\mathbf{A}^\circ)}\right)^{k\ell} \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_{\mathbf{A}}^2.$$

Proof. First, suppose that $\mathbf{A}$ is positive definite. Since the blocks consist of i.i.d. samples, the SC-RCD method fits into the framework of Proposition 3.2.9. Hence, it suffices to analyze

[3] If $(\mathbf{A}\langle\mathcal{S}\rangle)_{\mathcal{S},\mathcal{S}}$ is not invertible, these properties still hold and can be proved using the same arguments using the generalized Aitken block-diagonalization formula ([PS05, Eq. (6.0.20)]) instead.

$\mathbb{E}[\mathbf{Z}_1]$, where $\mathbf{Z}_1$ is the orthogonal projector onto range$(\mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_j)$ from (3.45) with $j \in [n]$ and block size $\ell = 1$.

The key idea is that the expectation $\mathbb{E}[\mathbf{Z}_1]$ admits a nice formula in terms of the residual matrix when the coordinate $j \in [n]$ is sampled with probability $\mathbf{A}^\circ_{j,j}/\operatorname{tr}(\mathbf{A}^\circ)$ in each iteration:

$$\begin{aligned}\mathbb{E}[\mathbf{Z}_1] &= \sum_{j:\mathbf{A}^\circ_{j,j}>0} \frac{\mathbf{A}^\circ_{j,j}}{\operatorname{tr}(A^\circ)} \cdot \frac{1}{\mathbf{A}^\circ_{j,j}} \mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_j\mathbf{e}_j^\mathsf{T}\mathbf{A}^{1/2}\mathbf{P} \\ &= \frac{1}{\operatorname{tr}(\mathbf{A}^\circ)} \mathbf{P}\mathbf{A}^{1/2} \left(\mathbf{I} - \sum_{j:\mathbf{A}^\circ_{j,j}=0} \mathbf{e}_j\mathbf{e}_j^\mathsf{T}\right) \mathbf{A}^{1/2}\mathbf{P} = \frac{1}{\operatorname{tr}(\mathbf{A}^\circ)} \mathbf{P}\mathbf{A}\mathbf{P}. \end{aligned} \tag{3.46}$$

Here we use the fact that $\sum_{j=1}^n \mathbf{e}_j\mathbf{e}_j^\mathsf{T} = \mathbf{I}$, and $\mathbf{A}^\circ_{j,j} = 0$ implies $\mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_j\mathbf{e}_j^\mathsf{T}\mathbf{A}^{1/2}\mathbf{P} = \mathbf{0}$, since this is a rank one, psd matrix with zero trace. Note that the exactness condition is trivially satisfied because

$$\operatorname{null}(\mathbb{E}[\mathbf{Z}_1]) = \operatorname{null}((\mathbf{A}^{1/2}\mathbf{P})^\mathsf{T}\mathbf{A}^{1/2}\mathbf{P}) = \operatorname{null}(\mathbf{A}^{1/2}\mathbf{P}).$$

Thus, by Proposition 3.2.9, SC-RCD with block size ℓ results in the expected error

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|_\mathbf{A}^2 \leq \left(1 - \frac{\lambda_{\min}^+(\mathbf{P}\mathbf{A}\mathbf{P})}{\operatorname{tr}(\mathbf{A}^\circ)}\right)^{k\ell} \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_\mathbf{A}^2.$$

To conclude, we observe that the eigenvalues of $\mathbf{P}\mathbf{A}\mathbf{P}$ and $\mathbf{A}^{1/2}\mathbf{P}\mathbf{A}^{1/2} = \mathbf{A}^\circ$ coincide.

Finally, we claim that the same result holds if $\mathbf{A}$ is positive semidefinite but not invertible. In this case, the solution $\mathbf{x}^*$ is not unique since it can be shifted by any vector in null$(\mathbf{A})$. However, we only need to track $\|\mathbf{x}^k - \mathbf{x}^*\|_\mathbf{A}^2$, which is constant for any choice of solution $\mathbf{x}^*$.[4] Although we cannot directly use the results from the subspace-constrained sketch-and-project framework, it can be shown by direct calculation that if we *define* the orthogonal projector $\mathbf{Z}$ by (3.45), then the fixed point equation $\mathbf{A}^{1/2}(\mathbf{x}^{k+1} - \mathbf{x}^*) = (\mathbf{I} - \mathbf{Z})\mathbf{A}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)$

[4]For example, we can write $\|\mathbf{x}^k - \mathbf{x}^*\|_\mathbf{A}^2 = f(\mathbf{x}^k) - f(\mathbf{x}^*)$, where $f : \mathbf{x} \mapsto \mathbf{x}^\mathsf{T}\mathbf{A}\mathbf{x} - 2\mathbf{b}^\mathsf{T}\mathbf{x}$ ([LL10, Eq. (8)]). Since $\nabla f(\mathbf{x}) = 2(\mathbf{A}\mathbf{x} - \mathbf{b})$, any solution $\mathbf{x}^*$ of $\mathbf{A}\mathbf{x} = \mathbf{b}$ with psd $\mathbf{A}$ is a minimizer of the convex quadratic f.

still holds. Hence, the same arguments in Theorem 3.2.4 and Proposition 3.2.9 can be used to reach the same conclusions. □

If the blocks are sampled uniformly at random instead, then we can obtain a similar result, except that the rate depends on the spectrum of the diagonal-normalized residual matrix. The proof is similar and will be omitted.

Proposition 3.3.3 (Uniform sampling)**.** *Consider the same setup as in Theorem 3.3.2. If instead, the block $\mathcal{J}$ consists of ℓ coordinates independently sampled from $\{j \in [n] : \mathbf{A}^\circ_{j,j} > 0\}$ uniformly at random in each iteration, then with $\mathbf{D} \in \mathbb{R}^{n\times n}$ the diagonal matrix with entries $\mathbf{D}_{j,j} = \mathbf{A}^\circ_{j,j}$,*

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{A}}^2 \leq \left(1 - \frac{\lambda_{\min}^+(\mathbf{D}^{\dagger/2}\mathbf{A}^\circ\mathbf{D}^{\dagger/2})}{n-d}\right)^{k\ell} \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_{\mathbf{A}}^2.$$

Note that when $\ell = 1$, the convergence rate in Theorem 3.3.2 is analogous to the bound (3.2) proved for randomized coordinate descent, which depends on $\lambda_{\min}^+(\mathbf{A})/\operatorname{tr}(\mathbf{A})$ and is tight in general. By using the fact that the eigenvalues of the residual matrix $\mathbf{A}^\circ$ interlace those of $\mathbf{A}$ (Remark 3.3.1), or directly applying the variational principle, the numerator of the rate is guaranteed to be no smaller: $\lambda_{\min}^+(\mathbf{A}^\circ) \geq \lambda_{\min}^+(\mathbf{A})$. However, a significant improvement can be realized if the denominator is much smaller; i.e., $\operatorname{tr}(\mathbf{A}^\circ) \ll \operatorname{tr}(\mathbf{A})$. This depends on the quality of the low-rank approximation in trace-norm and relates to how the pivots $\mathcal{S}$ are selected. When $\mathcal{S}$ is selected using RPCholesky, we are able to use its approximation guarantees to prove Theorem 3.1.3.

Proof of Theorem 3.1.3. Let $\mathbf{A}\langle\mathcal{S}\rangle$ be the Nyström approximation of $\mathbf{A}$ output by RPCholesky with randomly sampled pivot set $\mathcal{S}$. For a given $\mathcal{S}$, Theorem 3.3.2 implies that the error of the SC-RCD method satisfies

$$\mathbb{E}\left[\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{A}}^2 \mid \mathcal{S}\right] \leq \left(1 - \frac{\lambda_{\min}^+(\mathbf{A} - \mathbf{A}\langle\mathcal{S}\rangle)}{\operatorname{tr}(\mathbf{A} - \mathbf{A}\langle\mathcal{S}\rangle)}\right)^{k\ell} \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_{\mathbf{A}}^2.$$

Note that $\lambda_{\min}^{+}(\mathbf{A} - \mathbf{A}\langle S\rangle) \geq \lambda_{\min}^{+}(\mathbf{A})$. By using Theorem 3.1.1 and Markov's inequality, the event $\mathcal{E}$ where $\mathrm{tr}(\mathbf{A} - \mathbf{A}\langle\mathcal{S}\rangle) \leq \rho^{-1}(1+\delta)\sum_{i>r}\lambda_i(A)$ occurs with probability at least $1-\rho$. Conditional on this event, substituting this bound into the displayed equation above completes the proof. □

3.3.2 Implementation and complexity of SC-RCD

A practical implementation of the updates (3.43) and (3.44) of the SC-RCD method is summarized in the pseudocode in Algorithm 3.1, presented in Section 3.1.2. Its efficiency, as well as its complexity estimates used in Theorem 3.1.7, relies on several computational considerations discussed below.

(i) The partial pivoted Cholesky factor $\mathbf{F} \in \mathbb{R}^{n\times d}$ output by RPCholesky can be used to efficiently compute an initial iterate $\mathbf{x}^0$ solving $\mathbf{A}_{\mathcal{S},:}\mathbf{x}^0 = \mathbf{b}_{\mathcal{S}}$ (line 3). From Remark 3.3.1, observe that $\mathbf{A}_{\mathcal{S},\mathcal{S}} = (\mathbf{A}\langle\mathcal{S}\rangle)_{\mathcal{S},\mathcal{S}} = \mathbf{F}_{\mathcal{S},:}(\mathbf{F}_{\mathcal{S},:})^{\mathsf{T}}$, where $\mathbf{F}_{\mathcal{S},:} \in \mathbb{R}^{d\times d}$ is an invertible, lower triangular matrix, and so $(\mathbf{A}_{\mathcal{S},\mathcal{S}})^{-1} = (\mathbf{F}_{\mathcal{S},:})^{-\mathsf{T}}(\mathbf{F}_{\mathcal{S},:})^{-1}$. Hence, given any vector $\mathbf{x} \in \mathbb{R}^n$, we can compute

$$\mathbf{x}^0 = \mathbf{x} - \mathbf{e}_{\mathcal{S}}(\mathbf{A}_{\mathcal{S},\mathcal{S}})^{-1}(\mathbf{A}_{\mathcal{S},:}\mathbf{x} - \mathbf{b}_{\mathcal{S}})$$

by only modifying the coordinates of $\mathbf{x}$ in $\mathcal{S}$: first, we solve $\mathbf{F}_{\mathcal{S},:}\mathbf{w} = \mathbf{A}_{\mathcal{S},:}\mathbf{x} - \mathbf{b}_{\mathcal{S}}$ for $\mathbf{w} \in \mathbb{R}^d$ using forward substitution, and then solve $(\mathbf{F}_{\mathcal{S},:})^{\mathsf{T}}\boldsymbol{\beta} = \mathbf{w}$ for $\boldsymbol{\beta} \in \mathbb{R}^d$ using back substitution. Finally, we set $\mathbf{x}^0 \leftarrow \mathbf{x}$ and $\mathbf{x}^0_{\mathcal{S}} \leftarrow \mathbf{x}^0_{\mathcal{S}} - \boldsymbol{\beta}$. In total, this requires $O(d^2)$ flops, which represents a substantial improvement over the $O(d^2 n)$ flops from solving $\mathbf{A}_{\mathcal{S},:}\mathbf{x} = \mathbf{b}_{\mathcal{S}}$ naively.

Given any vector $\mathbf{x}$ and $\mathbf{r} = \mathbf{A}\mathbf{x} - \mathbf{b}$ (e.g., $\mathbf{x} = \mathbf{0}$ and $\mathbf{r} = -\mathbf{b}$), the residual vector $\mathbf{r}^0$ associated with $\mathbf{x}^0$ can also be computed by $\mathbf{r}^0 \leftarrow \mathbf{r} - \mathbf{A}_{:,\mathcal{S}}\boldsymbol{\beta}$, which costs $O(dn)$ flops, instead of $O(n^2)$ flops from computing $\mathbf{r}^0 = \mathbf{A}\mathbf{x}^0 - \mathbf{b}$ directly.

(ii) Similarly, the columns of the auxiliary matrix $\mathbf{C} = (\mathbf{A}_{\mathcal{S},\mathcal{S}})^\dagger \mathbf{A}_{\mathcal{S},:} = (\mathbf{F}_{\mathcal{S},:})^{-\mathsf{T}} \mathbf{F}^\mathsf{T} \in \mathbb{R}^{d \times n}$ (line 4) can be computed by solving a sequence of upper triangular linear systems with back substitution (each requiring $O(d^2)$ flops). Specifically, for each $j \in [n] \setminus \mathcal{S}$, the j^{th} column $\mathbf{C}_{:,j}$ is the solution of $(\mathbf{F}_{\mathcal{S},:})^\mathsf{T} \mathbf{C}_{:,j} = (\mathbf{F}_{j,:})^\mathsf{T}$. (The submatrix $\mathbf{C}_{:,\mathcal{S}}$ is the identity, but it is not used.) In total, computing $\mathbf{C}$ with this procedure requires $O(d^2(n-d))$ flops.

(iii) The block $\mathcal{J} \subseteq [n]$ of ℓ coordinates can also be sampled *without replacement* (line 7) with probabilities proportional to $\operatorname{diag}(\mathbf{A}^\circ)$, which will always result in a larger error decrease in each iteration (see the upcoming Remark 3.3.5). Another even simpler alternative is to sample the block $\mathcal{J}$ of ℓ coordinates *uniformly at random* (with or without replacement), which will be especially effective if $\mathbf{A}^\circ$ has incoherence properties (e.g., see [Der+25a, Lemma 10]).

(iv) Line 8 computes $\boldsymbol{\alpha}^{k-1} = (\mathbf{A}^\circ_{\mathcal{J},\mathcal{J}})^\dagger \mathbf{r}^{k-1}_{\mathcal{J}}$ by finding the min-norm solution of the $\ell \times \ell$ linear system $\mathbf{A}^\circ_{\mathcal{J},\mathcal{J}} \boldsymbol{\alpha} = \mathbf{r}^{k-1}_{\mathcal{J}}$.[5] For small block sizes ℓ, $\boldsymbol{\alpha}$ can be solved directly using a method based on QR or SVD using $O(\ell^3)$ flops. For larger block sizes, $\boldsymbol{\alpha}$ can be computed inexactly using an iterative method such as CG, noting that $\mathbf{A}^\circ_{\mathcal{J},\mathcal{J}}$ is psd (see the upcoming Remark 3.3.4).

Complexity estimates. Based on the considerations above, the computational costs of SC-RCD (Algorithm 3.1) can be analyzed in two stages:

- **Initialization:** learning the rank-d Nyström approximation $\mathbf{A}\langle\mathcal{S}\rangle = \mathbf{F}\mathbf{F}^\mathsf{T}$ using RPCholesky requires $O(d^2 n)$ arithmetic operations, $O(dn)$ entry evaluations of $\mathbf{A}$, and $O(dn)$ storage [Che+25; ETW24]. Note that the normalized diagonal of the residual matrix $\mathbf{A}^\circ$ can be read off the output of RPCholesky to obtain the sampling probabilities $\mathbf{p}$

[5] The linear system has a solution since $\mathbf{r}^k_{\mathcal{J}} \in \operatorname{range}(\mathbf{A}^\circ_{\mathcal{J},\mathcal{J}})$. This follows from using the fact that $\mathbf{P}\mathbf{A}^{1/2}(\mathbf{x}^k - \mathbf{x}^*) = \mathbf{A}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)$ to write $\mathbf{r}^k_{\mathcal{J}} = \mathbf{e}_{\mathcal{J}}^\mathsf{T}\mathbf{A}^{1/2}\mathbf{A}^{1/2}(\mathbf{x}^k - \mathbf{x}^*) = \mathbf{e}_{\mathcal{J}}^\mathsf{T}\mathbf{A}^{1/2}\mathbf{P}\mathbf{A}^{1/2}(\mathbf{x}^k - \mathbf{x}^*) = \mathbf{A}^\circ_{\mathcal{J},:}(\mathbf{x}^k - \mathbf{x}^*)$. Finally, we conclude by noting that $\operatorname{range}(\mathbf{A}^\circ_{\mathcal{J},:}) \subseteq \operatorname{range}(\mathbf{A}^\circ_{\mathcal{J},\mathcal{J}})$ since $\mathbf{A}^\circ$ is psd ([HZ05, Theorem 1.20]).

without any additional cost. Next, the initial iterate $\mathbf{x}$ and auxiliary matrix $\mathbf{C}$ can be computed with $O(d^2)$ and $O(d^2 n)$ operations, respectively, and dn entries of $\mathbf{C}$ have to be stored in memory.

- **Iterations:** in each iteration, the residual submatrix $\mathbf{A}^{\circ}_{\mathcal{J},\mathcal{J}} = \mathbf{A}_{\mathcal{J},\mathcal{J}} - \mathbf{F}_{\mathcal{J},:}(\mathbf{F}_{\mathcal{J},:})^{\mathsf{T}}$ is needed for the projection step. If $\mathbf{A}^{\circ}$ cannot be stored in memory, this requires accessing and storing ℓn entries of $\mathbf{A}$ in $\mathbf{A}_{:,\mathcal{J}}$ (since the entire columns are also needed to update the residual vector $\mathbf{r}$) and $O(\ell^2 d)$ arithmetic operations. Then, solving for $\boldsymbol{\alpha}$ requires $O(\ell^3)$ operations (possibly fewer if solved inexactly), computing $\boldsymbol{\beta}$ requires $O(\ell d)$ operations, and updating the iterate $\mathbf{x}$ and residual vector $\mathbf{r}$ requires $\ell + d$ and $O(\ell n) + O(\ell d + dn)$ operations, respectively. In summary, each iteration requires $O(dn + \ell n + \ell^3 + \ell^2 d)$ operations. If $\ell = O(\sqrt{n})$, then this simplifies to $O((\ell + d)n)$ operations per iteration.

Combining Theorem 3.1.3 with the analysis of the computational costs of the SC-RCD method allows us to prove Theorem 3.1.7 on the overall complexity of SC-RCD.

Proof of Theorem 3.1.7. If RPCholesky is independently run $T = \lceil \log_2(2/\varepsilon) \rceil$ times and the output $\mathcal{S}$ with the smallest residual trace-norm is chosen as in Remark 3.1.4, then the event $\mathcal{E}$ where $\operatorname{tr}(\mathbf{A} - \mathbf{A}\langle\mathcal{S}\rangle) \leq 2(1+1)\sum_{i>r}\lambda_i(\mathbf{A})$ occurs with probability at least $1 - 2^{-T} \geq 1 - \varepsilon/2$. Applying Theorem 3.1.3 with $\delta = 1$ implies that conditional on the event $\mathcal{E}$, the expected relative error after $k = \lceil 4(n/\ell)\bar{\kappa}_r(\mathbf{A}) \log(2/\varepsilon) \rceil$ iterations satisfies

$$\begin{aligned}\mathbb{E}\left[\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{A}}^2 \mid \mathcal{E}\right] &\leq \exp\left(-4n\bar{\kappa}_r(\mathbf{A})\log(2/\varepsilon) \cdot \frac{\lambda_{\min}^{+}(\mathbf{A})}{2(1+1)\sum_{i>r}\lambda_i(\mathbf{A})}\right) \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_{\mathbf{A}}^2 \\ &\leq (\varepsilon/2) \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_{\mathbf{A}}^2,\end{aligned}$$

where we used the elementary inequality $1 - t \leq e^{-t}$ for the first inequality. By using the monotonicity property $\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{A}}^2 \leq \|\mathbf{x}^0 - \mathbf{x}^*\|_{\mathbf{A}}^2$ in the event that $\mathcal{E}$ does not hold, denoted

by $\mathcal{E}^c$, the overall expectation can be bounded by

$$\begin{aligned}\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{A}}^2 &= \mathbb{E}\left[\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{A}}^2 \mid \mathcal{E}\right] \cdot \mathbb{P}\left(\mathcal{E}\right) + \mathbb{E}\left[\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{A}}^2 \mid \mathcal{E}^c\right] \cdot \mathbb{P}\left(\mathcal{E}^c\right) \\ &\leq (\varepsilon/2) \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_{\mathbf{A}}^2 + (\varepsilon/2) \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_{\mathbf{A}}^2 = \varepsilon \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_{\mathbf{A}}^2,\end{aligned}$$

as desired. It remains to compute the computational costs. Running RPCholesky $T = O(\log(1/\varepsilon))$ times and initializing requires $O(d^2 n \log(1/\varepsilon))$ arithmetic operations, $O(dn \log(1/\varepsilon))$ entry evaluations, and $O(dn)$ storage. Subsequently, each SC-RCD iteration requires accessing ℓn entries of $\mathbf{A}$ and $O(dn + \ell n + \ell^3 + \ell^2 d)$ arithmetic operations, so in total $O(n^2 \cdot \bar{\kappa}_r(\mathbf{A}) \log(1/\varepsilon))$ entry evaluations and $O((n^2(d+\ell)/\ell + \ell^2 n + \ell dn) \cdot \bar{\kappa}_r(\mathbf{A}) \log(1/\varepsilon))$ arithmetic operations are required. □

Remark 3.3.4 (Inexact projections)**.** Note that each update (3.24) requires finding the min-norm solution $\mathbf{z}$ of the linear system $\mathbf{S}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P}\mathbf{z} = \mathbf{S}(\mathbf{A}\mathbf{x}^k - \mathbf{b})$ to compute $(\mathbf{S}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T})^\dagger \mathbf{S}(\mathbf{A}\mathbf{x}^k - \mathbf{b})$. For practical efficiency, it is possible for an approximate solution to be computed; e.g., using an inner iterative method such as preconditioned CG. See [DY24, §4.3], [Der+25a, §6], or [TRG16] for results along these lines, where similar theoretical bounds as in Theorem 3.2.4 can be derived with the loss of a small multiplicative factor in the rate.

Remark 3.3.5 (Sampling without replacement)**.** If the coordinates in the blocks are sampled without replacement in Theorem 3.3.2 or Proposition 3.3.3, then the same bounds hold because the convergence rate can only improve. To see this, suppose that $\mathcal{J}$ and $\mathcal{J}'$ consists of ℓ coordinates sampled with and without replacement, respectively, and let $\mathbf{Z} \equiv \mathbf{Z}(\mathcal{J})$ and $\mathbf{Z}' \equiv \mathbf{Z}'(\mathcal{J}')$ denote the corresponding orthogonal projectors onto range$(\mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_{\mathcal{J}})$ and range$(\mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_{\mathcal{J}'})$ from (3.45). The key observation is that $\mathcal{J}$ and $\mathcal{J}'$ can be coupled such that $\mathcal{J} \subseteq \mathcal{J}'$ by rejection sampling (e.g., $\mathcal{J}'$ can be formed by proposing the same indices sampled for $\mathcal{J}$ and resampling any duplicates), and hence $\mathbf{Z}' \succeq \mathbf{Z}$. Combined with Lemma 3.2.3, this implies that the error decrease in each iteration with $\mathbf{Z}'$ is always at least as large as with $\mathbf{Z}$.

3.4 Numerical experiments

In this section, we present some numerical experiments demonstrating various features of the SC-RCD method. The experiments were performed using Python 3.12.7 on a 2.6 GHz Intel Skylake CPU with 32GB RAM. The code is available at https://github.com/jackielok/subspace-constrained-rcd.

3.4.1 Synthetic psd system

In the first experiment, we demonstrate the effectiveness of the SC-RCD method (Algorithm 3.1) for solving approximately low-rank systems. We simulate a $n \times n$ psd linear system with $n = 8,192 = 2^{13}$, where the first $r = 400$ eigenvalues are equal to one (i.e., are "large" up to normalization) and subsequently decay as $\lambda_i = i^{-3/2}$ for $i > 400$, by defining a diagonal matrix $\mathbf{\Sigma}$ with $\mathbf{\Sigma}_{i,i} = \lambda_i$ and rotating with a uniformly random orthogonal matrix $\mathbf{U}$ to form $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{U}^\mathsf{T}$.

Figure 3.1 (right) shows that the residual matrix $\mathbf{A}^\circ = \mathbf{A} - \mathbf{A}\langle\mathcal{S}\rangle$ with approximation rank $d = 500 \approx 5.5\sqrt{n}$ is much better conditioned than $\mathbf{A}$. Accordingly, Figure 3.1 (left) shows that SC-RCD, using the corresponding rank-d Nyström approximation and block size $\ell = 500$, converges effectively. For comparison, we also show the convergence rate, measured on an epoch-basis, for related methods including the conjugate gradient method (CG); randomized coordinate descent (RCD) with blocks of size $\ell = 500$, sampled in the same way as SC-RCD; and the recently proposed CD++ method from Dereziński et al. [Der+25b], which combines RCD with techniques such as adaptive acceleration and Hadamard preconditioning.

Figure 3.2 shows the relative residual norm after 200 epochs for SC-RCD, RCD, and CD++ using various block sizes ℓ and approximation ranks d. Figure 3.2 (left) shows that the SC-RCD error decreases as ℓ increases. The theory that we develop (Theorem 3.3.2) implies that this curve should be non-decreasing; however, the actual performance (significantly) exceeds this bound, which reflects how larger blocks are able to implicitly capture larger

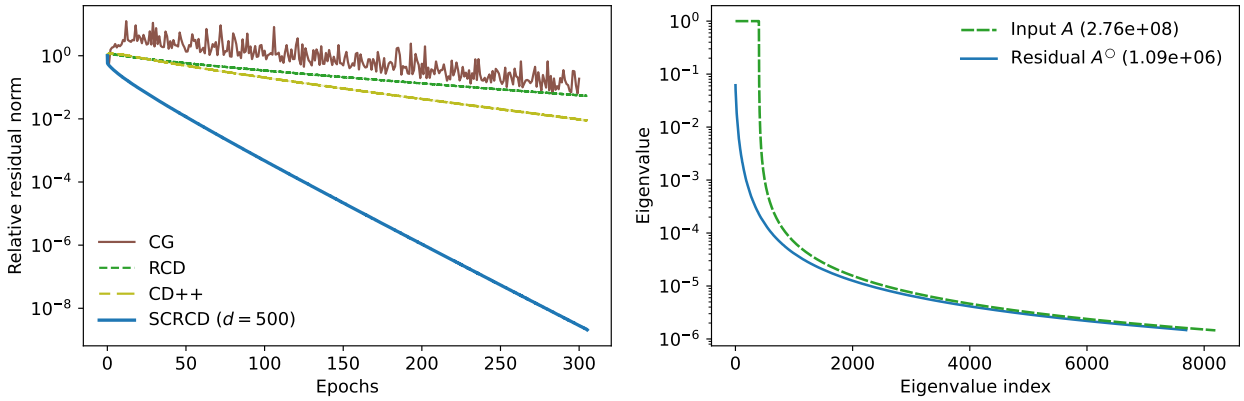


Figure 3.1: Solving a synthetic $8,192 \times 8,192$ psd system $\mathbf{Ax} = \mathbf{b}$ with approximate rank $r = 400$. **(Left)** Relative residual norm $\|\mathbf{Ax}^k - \mathbf{b}\|_2 / \|\mathbf{Ax}^0 - \mathbf{b}\|_2$ over 300 epochs for SC-RCD (with $d = 500$ and $\ell = 500$), as well as CG, RCD and CD++ (also with $\ell = 500$), using the same initial iterate as SC-RCD. Each epoch corresponds to a single pass over the entire dataset (i.e., one iteration of SC-RCD/RCD/CD++ corresponds to ℓ/n epochs). The lines depict the median over 100 independent runs with the same Nyström approximation. **(Right)** Eigenvalue spectra of $\mathbf{A}$ and the residual matrix $\mathbf{A}^\circ$ corresponding to the rank-d approximation, with their condition numbers $\sum_i \lambda_i / \lambda_{\min}^+$ reported in brackets.

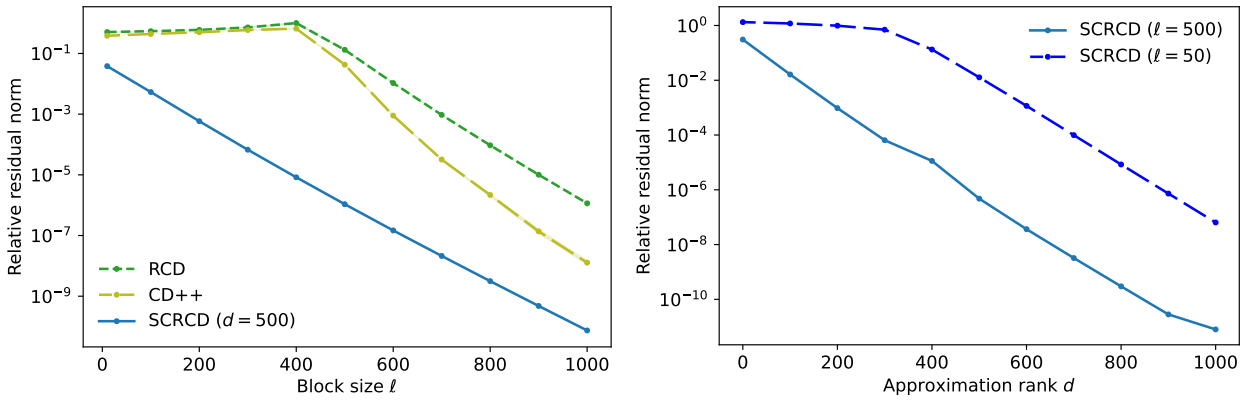


Figure 3.2: Continuing the same setup as in Figure 3.1, the plots show the relative residual norm after 200 epochs for **(Left)** SC-RCD (with $d = 500$), RCD, and CD++ with various block sizes; and **(Right)** SC-RCD (with $\ell \in \{50, 500\}$) with various approximation ranks.

parts of the spectrum of $\mathbf{A}$ (see [DR24; Der+25b] for related theory). It also shows that RCD and CD++ do not converge effectively until ℓ is large enough to implicitly capture the leading $r = 400$ eigenvalues, after which they significantly improve. Figure 3.2 (right) shows that the SC-RCD error also decreases as d increases and RPCholesky computes a higher quality matrix approximation (Theorem 3.1.1). We observe a significant improvement once

d is large enough, in combination with ℓ (due to the implicit effects of block size), to capture the r large spectral outliers of $\mathbf{A}$ in each iteration.

We note that while each iteration of SC-RCD incurs an additional computational cost of $O(nd)$ to enforce the subspace constraint compared to RCD, it is not the dominating term in the complexity of each iteration when $d, \ell \approx O(\sqrt{n})$ (see Section 3.3.2). Correspondingly, the time to run SC-RCD and RCD in Figures 3.1 and 3.2 with the same block size ℓ was found to be very close. The iterations of CD++ with the same block size are somewhat faster due to its use of techniques such as approximate regularized projections and block memoization, which could be integrated with SC-RCD for practical efficiency as a part of future work. However, we note that the theoretical guarantee (3.16) for CD++ requires the matrix $\mathbf{A}$ to be stored in memory and preprocessed by a randomized Hadamard transform, or otherwise requires $\mathbf{A}$ to possess some natural incoherence properties. This makes it more difficult to apply for large-scale problems, such as in the upcoming experiment.

3.4.2 KRR problem on real-life dataset with fast spectral decay

In the next experiment, we investigate the performance of the SC-RCD method for solving large-scale kernel ridge regression (KRR) problems where the matrix cannot be stored in memory, and the dominant computational cost comes from matrix evaluations.

To give a brief overview of KRR, suppose that we are given n data points $\{(\mathbf{z}_i, y_i)\}_{i=1}^n$ with features $\mathbf{z}_i \in \mathbb{R}^p$ and response variable $y_i \in \mathbb{R}$. A kernel matrix $\mathbf{K} \in \mathbb{R}^{n \times n}$ is formed with $\mathbf{K}_{i,j} = \kappa(\mathbf{z}_i, \mathbf{z}_j)$ for some positive definite kernel function $\kappa : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$. For our experiments, we will use the Gaussian kernel with bandwidth parameter $\sigma > 0$, defined by $\kappa(\mathbf{z}_i, \mathbf{z}_j) = \exp\left(-\|\mathbf{z}_i - \mathbf{z}_j\|_2^2/(2\sigma^2)\right)$. Then, given a regularization parameter $\lambda > 0$, the goal of KRR is to find a vector $\mathbf{x} \in \mathbb{R}^n$ to minimize $\|\mathbf{K}\mathbf{x} - \mathbf{y}\|_2^2 + \lambda \mathbf{x}^\mathsf{T}\mathbf{K}\mathbf{x}$, which is equivalent to solving the positive definite system $(\mathbf{K} + \lambda \mathbf{I})\mathbf{x} = \mathbf{y}$.

In Figure 3.3, we take $n = 100,000$ samples from the `hls4ml_lhc_jets` dataset [Pie+20], which consists of features $\mathbf{z}_i \in \mathbb{R}^{16}$ for predicting jet classes from LHC proton-proton colli-

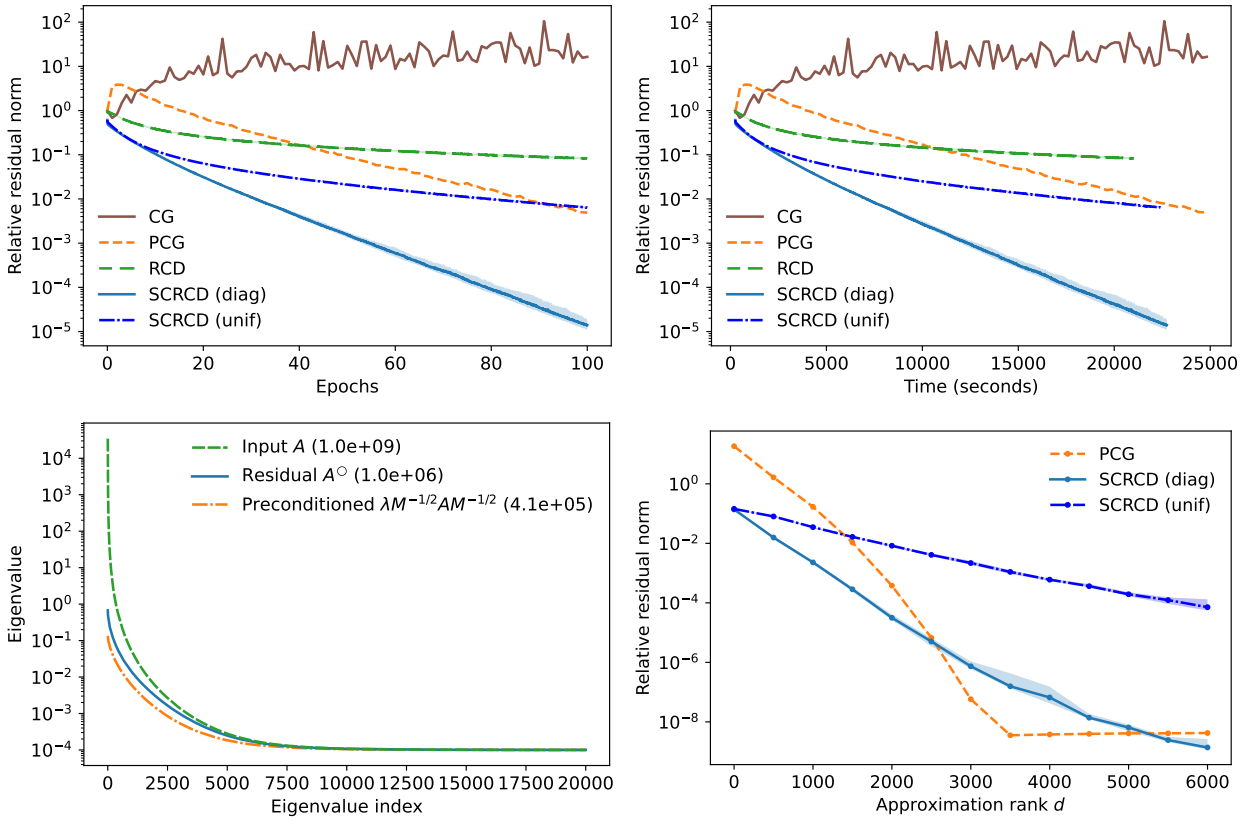


Figure 3.3: Solving the KRR problem $(\mathbf{K} + \lambda \mathbf{I})\mathbf{x} = \mathbf{y}$ on the `hls4ml_lhc_jets` dataset with $n = 100,000$ samples and a very small amount of regularization $\lambda = 10^{-9}n$. **(Top left)** Relative residual norm $\|(\mathbf{K}+\lambda\mathbf{I})\mathbf{x}^k - \mathbf{y}\|_2 / \|\mathbf{y}\|_2$ over 100 epochs for SC-RCD (with $d = 1,000$ and $\ell = 1,000$), as well as RCD (also with $\ell = 1,000$), CG, and PCG (also with $d = 1,000$). The lines (resp. shaded interval) depict the median (resp. 0.2- and 0.8-quantiles) over 100 independent runs. **(Top right)** Error in terms of time elapsed. The kernel matrix $\mathbf{K}$ is not stored in memory, and entry evaluations represent the dominant computational cost. **(Bottom left)** The leading $20,000$ eigenvalues of $\mathbf{A} = \mathbf{K} + \lambda\mathbf{I}$, the residual $\mathbf{A}^\circ$, and the preconditioned $\lambda\mathbf{M}^{-1/2}\mathbf{A}\mathbf{M}^{-1/2}$ corresponding to the rank-d Nyström approximation, with their condition numbers $\sum_i \lambda_i / \lambda_{\min}^+$ reported in brackets. **(Bottom right)** The error after 50 epochs for SC-RCD and PCG with various approximation ranks d.

sions. We consider solving $(\mathbf{K} + \lambda\mathbf{I})\mathbf{x} = \mathbf{y}$ using the Gaussian kernel with bandwidth $\sigma = 3$ and a small regularization parameter $\lambda = 10^{-9}n$, which results in a more ill-conditioned and challenging system to solve. Figure 3.3 (bottom) shows that the eigenvalues of $\mathbf{K}$ decay exponentially, so the regularized system has a flat-tailed spectrum that quickly decays to λ.

We consider the SC-RCD method where the blocks consist of indices sampled with weights proportional to the diagonal of the residual matrix or uniformly, as well as RCD (where

both forms of sampling are equivalent since $\mathbf{K}$ has unit diagonals). In each iteration, we perform inexact projections, where $\boldsymbol{\alpha}$ is solved up to a relative error of 0.05 using CG with a simple Jacobi preconditioner (i.e., diagonal normalization). For comparison, we also solve the system using the preconditioned CG method (PCG) proposed by Díaz et al. [Día+24], which uses a preconditioner $\mathbf{M} = \mathbf{F}\mathbf{F}^\mathsf{T} + \lambda\mathbf{I}$, constructed from an approximation $\widehat{\mathbf{K}} = \mathbf{F}\mathbf{F}^\mathsf{T}$ of $\mathbf{K}$ using RPCholesky.

Figure 3.3 (top) show the convergence rate of these iterative solvers, measured in terms of the number of epochs completed (left) and the total time elapsed (right). We observe that the SC-RCD method with a relatively small Nyström approximation significantly improves upon RCD, analogous to the improvement of PCG over CG as shown by [Día+24]. Furthermore, SC-RCD converges faster than PCG with the same approximation rank $d = 1,000$ used. However, Figure 3.3 (bottom right) shows that the improvement in the rate of PCG with larger d is faster than for SC-RCD. In practice, the choice of d may be limited by the availability of memory.

This experiment provides limited evidence of how coordinate descent-based methods can be competitive with methods such as preconditioned CG for large-scale problems where entry evaluations are costly. There may be further computational advantages of CD-based methods, such as the possibility for acceleration [Tu+17], parallelization (e.g., averaging over mini-batches [RT20, Algorithm 2]), and asynchronization [ADG15], but we do not investigate these possibilities.

3.4.3 KRR problem with slower spectral decay

In the final experiment, we investigate the performance of SC-RCD for solving another KRR problem on a dataset with slower spectral decay. We also demonstrate that how the blocks are sampled can play a critical role in the convergence rate of SC-RCD.

Specifically, we consider solving $(\mathbf{K} + \lambda\mathbf{I})\mathbf{x} = \mathbf{y}$ using $n = 20,000$ samples from the `sensorless` dataset [CL11], which consists of features $\mathbf{z}_i \in \mathbb{R}^{48}$, the Gaussian kernel with

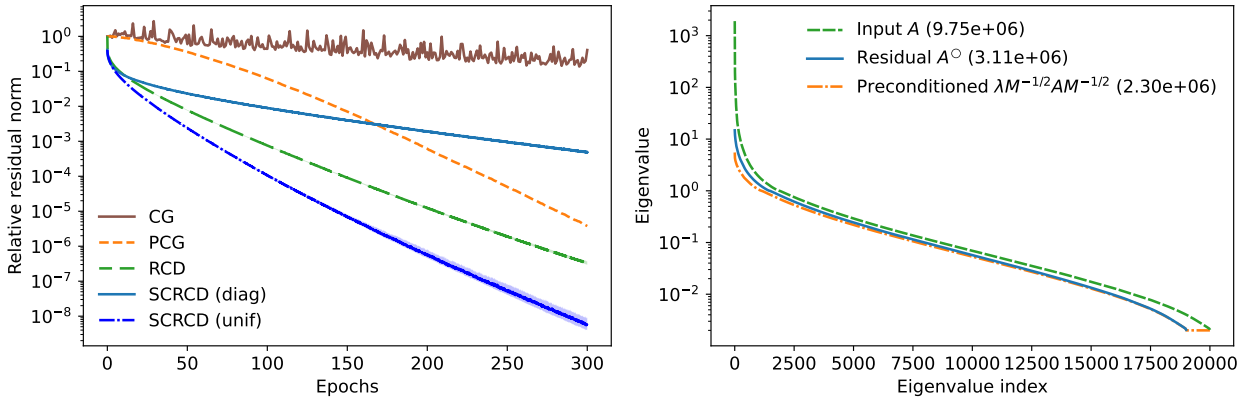


Figure 3.4: Solving the KRR problem $(\mathbf{K} + \lambda\mathbf{I})\mathbf{x} = \mathbf{y}$ on the `sensorless` dataset with $n = 20,000$ samples and a small regularization parameter $\lambda = 10^{-7}n$. **(Left)** Relative residual norm over 300 epochs for SC-RCD (with $d = 1,000$ and $\ell = 1,000$), RCD (also with $\ell = 1,000$), CG, and PCG (also with $d = 1,000$). **(Right)** Eigenvalue spectra of $\mathbf{A} = \mathbf{K} + \lambda\mathbf{I}$, the residual $\mathbf{A}^{\circ}$, and the preconditioned $\lambda\mathbf{M}^{-1/2}\mathbf{A}\mathbf{M}^{-1/2}$ corresponding to the rank-d Nyström approximation, with their condition numbers $\sum_i \lambda_i/\lambda_{\min}^{+}$ reported in brackets.

bandwidth $\sigma = 3$, and a small regularization parameter $\lambda = 10^{-7}n$. This dataset was identified as one of the more difficult KRR problems studied in [Día+24]. Figure 3.4 (right) confirms that the kernel matrix exhibits much slower spectral decay, making it far more difficult to find a good low-rank approximation. Figure 3.4 (left) shows that SC-RCD with uniformly sampled blocks exhibits the fastest convergence rate, and diagonal sampling—which has been the most effective for systems with rapid spectral decay so far—actually performs poorly. We observe that SC-RCD with uniform sampling improves upon RCD, as expected from Proposition 3.3.3.

3.5 Concluding remarks

We proposed and analyzed the SC-RCD method for solving psd linear systems $\mathbf{A}\mathbf{x} = \mathbf{b}$, which combines the classical randomized block coordinate descent algorithm with a rank-d matrix approximation, efficiently computable using an algorithm such as RPCholesky. We proved that it is a lightweight algorithm that can obtain an ε-relative error solution using $O(nd)$ memory and $O\big((n^2 + nd^2) \cdot \bar{\kappa}_r(\mathbf{A}) \log(1/\varepsilon)\big)$ arithmetic operations, where $\bar{\kappa}_r(\mathbf{A}) =$

$\sum_{i>r} \lambda_i(\mathbf{A})/\lambda_{\min}^+(\mathbf{A})$ is the normalized tail condition number of $\mathbf{A}$ and r is typically close to the approximation rank d. This makes SC-RCD effective for solving large-scale, dense systems with rapid spectral decay, such as those arising in kernel ridge regression. We presented numerical experiments in support of these results.

Some directions for future work include combining the subspace-constrained framework with other computational techniques, such as those employed in [Der+25b], for further practical efficiency. For example, momentum-based acceleration would help attain an improved complexity in terms of the condition number dependence. A more general question suggested by this work is how one can efficiently learn subspaces that control different parts of the spectrum, such as small spectral outliers. Another future direction is to investigate how constraining the dynamics within a selected subspace can be used to accelerate other iterative algorithms based on the sketch-and-project approach, particularly those addressing more general nonlinear problems.

3.6 Subspace-constrained sketch-and-project technical proofs

In this section, we give the technical proofs of Lemmas 3.2.1 and 3.2.2 for subspace-constrained sketch-and-project from Section 3.2.

Proof of Lemma 3.2.1. Given the iterate $\mathbf{x}^k$ after the k^{th} iteration, define $\bar{\mathbf{z}} := \mathbf{x}^k - \mathbf{x}^{k+1}$. Then from (3.22), together with the change of variables $\mathbf{w} = \mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x})$, we have

$$\mathbf{B}^{1/2}\bar{\mathbf{z}} = \begin{bmatrix} \arg\min\limits_{\mathbf{w}\in\mathbb{R}^n} & \mathbf{w}^\mathsf{T}\mathbf{w} \\ \text{such that} & \mathbf{S}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{w} = \mathbf{S}(\mathbf{A}\mathbf{x}^k - \mathbf{b}), \\ & \mathbf{Q}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{w} = \mathbf{Q}(\mathbf{A}\mathbf{x}^k - \mathbf{b}). \end{bmatrix} \tag{3.47}$$

Note that since $\mathbf{QAx}^k = \mathbf{Qb}$, the second constraint is equivalent to $\mathbf{w} \in \text{null}(\mathbf{QAB}^{-1/2})$, or $\mathbf{w} = \mathbf{Pw}$. By introducing Lagrange multipliers $\boldsymbol{\lambda} \in \mathbb{R}^\ell$, $\boldsymbol{\tau} \in \mathbb{R}^d$, we deduce that the optimal $\mathbf{w}_* = \mathbf{B}^{1/2}\bar{\mathbf{z}}$ solves the first-order conditions

$$\begin{aligned} \mathbf{w}_* + \mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T}\boldsymbol{\lambda} + \mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{Q}^\mathsf{T}\boldsymbol{\tau} &= \mathbf{0}, \\ \mathbf{SAB}^{-1/2}\mathbf{w}_* &= \mathbf{S}(\mathbf{Ax}^k - \mathbf{b}), \\ \mathbf{Pw}_* &= \mathbf{w}_*. \end{aligned}$$

First, by definition of $\mathbf{P}$, we have $\mathbf{PB}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{Q}^\mathsf{T} = \mathbf{0}$. Hence, by multiplying the first equation by $\mathbf{P}$, we obtain $\mathbf{Pw}_* + \mathbf{PB}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T}\boldsymbol{\lambda} = \mathbf{0}$. By combining this with $\mathbf{Pw}_* = \mathbf{w}_*$, we deduce that

$$\mathbf{w}_* = -\mathbf{PB}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T}\boldsymbol{\lambda}.$$

Thus, $\mathbf{w}_* \in \text{range}(\mathbf{PB}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T})$, which implies that $\mathbf{Zw}_* = \mathbf{w}_*$. This shows that

$$\begin{aligned} \mathbf{w}_* = \mathbf{Zw}_* &= \mathbf{PB}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T}(\mathbf{SAB}^{-1/2}\mathbf{PB}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T})^\dagger\mathbf{SAB}^{-1/2}\mathbf{Pw}_* \\ &= \mathbf{PB}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T}(\mathbf{SAB}^{-1/2}\mathbf{PB}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T})^\dagger\mathbf{S}(\mathbf{Ax}^k - \mathbf{b}), \end{aligned} \tag{3.48}$$

where in the last line we used that $\mathbf{Pw}_* = \mathbf{w}_*$, and so $\mathbf{SAB}^{-1/2}\mathbf{Pw}_* = \mathbf{S}(\mathbf{Ax}^k - \mathbf{b})$. Recalling that $\mathbf{x}^{k+1} = \mathbf{x}^k - \mathbf{B}^{-1/2}\mathbf{w}_*$, the update rule (3.24) follows from (3.48).

Next, from the update rule (3.24), we obtain

$$\begin{aligned} \mathbf{B}^{1/2}(\mathbf{x}^{k+1} - \mathbf{x}^*) &= \mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*) - \mathbf{PB}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T}(\mathbf{SAB}^{-1/2}\mathbf{PB}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T})^\dagger\mathbf{SA}(\mathbf{x}^k - \mathbf{x}^*) \\ &= \mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*) - \mathbf{ZB}^{1/2}(\mathbf{x}^k - \mathbf{x}^*), \end{aligned}$$

where in the last line we used $\mathbf{x}^k - \mathbf{x}^* = \mathbf{B}^{-1/2}\mathbf{PB}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)$. Indeed, since $\mathbf{QA}(\mathbf{x}^k - \mathbf{x}^*) = \mathbf{Q}(\mathbf{Ax}^k - \mathbf{b}) = \mathbf{0}$, we have $\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*) \in \text{null}(\mathbf{QAB}^{-1/2})$ and so $\mathbf{PB}^{1/2}(\mathbf{x}^k - \mathbf{x}^*) = \mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)$. This concludes the proof of the fixed point iteration (3.25). □

Proof of Lemma 3.2.2. From the formula (3.18) for $\mathbf{x}^*$, we see that $\mathbf{B}^{1/2}(\mathbf{x}^0 - \mathbf{x}^*) \in \mathrm{range}(\mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T})$. Furthermore, since the initial iterate solves $\mathbf{QAx}^0 = \mathbf{Qb}$, we have $\mathbf{QA}(\mathbf{x}^0 - \mathbf{x}^*) = \mathbf{Q}(\mathbf{Ax}^0 - \mathbf{b}) = \mathbf{0}$, and thus $\mathbf{B}^{1/2}(\mathbf{x}^0 - \mathbf{x}^*) \in \mathrm{null}(\mathbf{QAB}^{-1/2}) = \mathrm{range}(\mathbf{P})$. Hence,

$$\mathbf{B}^{1/2}(\mathbf{x}^0 - \mathbf{x}^*) = \mathbf{PB}^{1/2}(\mathbf{x}^0 - \mathbf{x}^*) \in \mathrm{range}(\mathbf{PB}^{-1/2}\mathbf{A}^\mathsf{T}).$$

Next, observe that $\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*) \in \mathrm{range}(\mathbf{P})$ for all $k \geq 0$ since the subsequent iterates $\mathbf{x}^k$ continue to solve $\mathbf{QAx}^k = \mathbf{Qb}$. From the fixed point iteration (3.25) in Lemma 3.2.1, we have $\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*) = \mathbf{B}^{1/2}(\mathbf{x}^{k-1} - \mathbf{x}^*) - \mathbf{ZB}^{1/2}(\mathbf{x}^{k-1} - \mathbf{x}^*)$. Since $\mathbf{Z}$ is the orthogonal projector onto $\mathrm{range}(\mathbf{PB}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T})$, it follows from induction that $\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*) \in \mathrm{range}(\mathbf{PB}^{-1/2}\mathbf{A}^\mathsf{T})$ for all $k \geq 0$. □

3.7 Extension of SC-RCD for least-squares problems

Algorithm 3.2 SC-RCD: least squares

Require: Matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, vector $\mathbf{b} \in \mathbb{R}^m$, approximation rank d, block size ℓ
Ensure: Approximate solution $\mathbf{x} \in \mathbb{R}^n$ of $\arg\min_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|_2$, residual vector $\mathbf{r} = \mathbf{Ax} - \mathbf{b} \in \mathbb{R}^m$

Compute pivot set $\mathcal{S} \subseteq [n]$ and $\mathbf{Q} \in \mathbb{R}^{m \times d}$, $\mathbf{R} \in \mathbb{R}^{d \times n}$ defining column-pivoted partial QR decomp. $\widehat{\mathbf{A}} = \mathbf{QR}$, and set $\mathbf{A}^\circ \leftarrow \mathbf{A} - \widehat{\mathbf{A}}$ ▷ E.g., [Che+25, Alg. 7]
Compute $\mathbf{D} \leftarrow (\mathbf{R}_{:,\mathcal{S}})^{-1}\mathbf{Q}^\mathsf{T} \in \mathbb{R}^{d \times m}$ and $\mathbf{C} \leftarrow \mathbf{DA} \in \mathbb{R}^{d \times n}$
Set $\mathbf{x} \leftarrow \mathbf{Db} \in \mathbb{R}^n$ and $\mathbf{r} \leftarrow \mathbf{Ax} - \mathbf{b} \in \mathbb{R}^m$
Set $\mathbf{p} \leftarrow \mathbf{0}_{n \times 1}$, and compute $\mathbf{p}_j \leftarrow \|\mathbf{A}^\circ_{:,j}\|_2^2 / \|\mathbf{A}^\circ\|_F^2$ for $j \in [n] \setminus \mathcal{S}$
for $k = 1, 2, \ldots$ **do**
    Sample subset $\mathcal{J} = \{j_1, \ldots, j_\ell\}$ of ℓ columns with $j_1, \ldots, j_\ell \sim \mathbf{p}$ i.i.d.
    Solve $(\mathbf{A}^\circ_{:,\mathcal{J}})^\mathsf{T}\mathbf{A}^\circ_{:,\mathcal{J}}\boldsymbol{\alpha} = (\mathbf{A}^\circ_{:,\mathcal{J}})^\mathsf{T}\mathbf{r}$ for $\boldsymbol{\alpha} \in \mathbb{R}^\ell$
    $\boldsymbol{\beta} \leftarrow \mathbf{C}_{:,\mathcal{J}}\boldsymbol{\alpha} \in \mathbb{R}^d$
    $\mathbf{x}_\mathcal{J} \leftarrow \mathbf{x}_\mathcal{J} - \boldsymbol{\alpha}$, $\mathbf{x}_\mathcal{S} \leftarrow \mathbf{x}_\mathcal{S} + \boldsymbol{\beta}$
    $\mathbf{r} \leftarrow \mathbf{r} - \mathbf{A}^\circ_{:,\mathcal{J}}\boldsymbol{\alpha}$
end for

In this section, we will briefly explain how the SC-RCD method can be adapted to solving the least-squares problem

$$\arg\min_{\mathbf{x}\in\mathbb{R}^n} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2, \quad \text{where } \mathbf{A}\in\mathbb{R}^{m\times n}. \tag{3.49}$$

Since (3.49) reduces to the solution of the normal equations $\mathbf{A}^\mathsf{T}\mathbf{A}\mathbf{x} = \mathbf{A}^\mathsf{T}\mathbf{b}$, this can be solved by applying the psd SC-RCD method on the normal equations. In the following, we will show that the algorithm can be implemented without explicitly forming the psd matrix $\mathbf{A}^\mathsf{T}\mathbf{A}$ as a column-action method; see Algorithm 3.2 for pseudocode.

In this setting, the natural analogue of the Nyström approximation is the *column projection approximation* $\mathbf{\Pi}_{\mathbf{A},\mathcal{S}}\mathbf{A}$ of $\mathbf{A}$, where $\mathbf{\Pi}_{\mathbf{A},\mathcal{S}}$ is the orthogonal projector onto the span of the columns of $\mathbf{A}$ indexed by $\mathcal{S}$. The key observation is that the entries of the Gram matrix $(\mathbf{A}_{:,\mathcal{S}})^\mathsf{T}\mathbf{A}_{:,\mathcal{S}}$ give the inner products between the columns of $\mathbf{A}$ indexed by $\mathcal{S}$; in particular, the squared column norms can be read off the diagonal (see, e.g., [Che+25, §3] for more details on this classical connection).

Indeed, the RPCholesky algorithm can be naturally adapted to a *randomly pivoted QR* algorithm that outputs a column projection approximation in the form of a partial QR decomposition $\widehat{\mathbf{A}} = \mathbf{Q}\mathbf{R}$ of $\mathbf{A}$, where $\mathbf{Q}\in\mathbb{R}^{m\times d}$ has orthonormal columns and $\mathbf{R}\in\mathbb{R}^{d\times n}$ is upper triangular (after pivoting to bring the columns in $\mathcal{S}$ to the front), such that $\widehat{\mathbf{A}}_{:,\mathcal{S}} = \mathbf{A}_{:,\mathcal{S}}$ and $\operatorname{range}(\widehat{\mathbf{A}}) = \operatorname{range}(\mathbf{A}_{:,\mathcal{S}})$; see [Che+25, Algorithm 7]. We note that there are many other approaches for solving the corresponding *column subset selection problem*, including an elegant strategy based on adaptive randomized pivoting recently analyzed in [CK26].

Derivation of Algorithm 3.2

Suppose that we are given a column projection approximation $\mathbf{\Pi}_{\mathbf{A},\mathcal{S}}\mathbf{A}$ of $\mathbf{A}$ with d pivots $\mathcal{S}\subseteq[n]$ and an initial iterate $\mathbf{x}^0$ satisfying $(\mathbf{A}^\mathsf{T}\mathbf{A})_{\mathcal{S},:}\mathbf{x}^0 = (\mathbf{A}^\mathsf{T}\mathbf{b})_{\mathcal{S}}$, or equivalently $(\mathbf{A}_{:,\mathcal{S}})^\mathsf{T}(\mathbf{A}\mathbf{x}^0 -$

$\mathbf{b}) = \mathbf{0}$. Let

$$\widehat{\mathbf{A}} := \mathbf{\Pi}_{\mathbf{A},\mathcal{S}}\mathbf{A} \quad \text{and} \quad \mathbf{A}^{\circ} := \mathbf{A} - \widehat{\mathbf{A}} \tag{3.50}$$

be the column projection approximation of $\mathbf{A}$ with respect to the columns indexed by $\mathcal{S}$ and the corresponding residual matrix, respectively. Observe that if

$$\mathbf{P} = \mathbf{I} - (\mathbf{A}^{\mathsf{T}}\mathbf{A})^{1/2}\mathbf{e}_{\mathcal{S}}((\mathbf{A}^{\mathsf{T}}\mathbf{A})_{\mathcal{S},\mathcal{S}})^{\dagger}\mathbf{e}_{\mathcal{S}}^{\mathsf{T}}(\mathbf{A}^{\mathsf{T}}\mathbf{A})^{1/2}$$

is the orthogonal projector onto $\text{null}(\mathbf{e}_{\mathcal{S}}^{\mathsf{T}}(\mathbf{A}^{\mathsf{T}}\mathbf{A})^{1/2})$, then

$$(\mathbf{A}^{\mathsf{T}}\mathbf{A})^{1/2}\mathbf{P}(\mathbf{A}^{\mathsf{T}}\mathbf{A})^{1/2} = \mathbf{A}^{\mathsf{T}}(\mathbf{I} - \mathbf{\Pi}_{\mathbf{A},\mathcal{S}})\mathbf{A} = (\mathbf{A}^{\circ})^{\mathsf{T}}\mathbf{A}^{\circ}. \tag{3.51}$$

Hence, after some algebraic manipulations, the update (3.43) for the iterate $\mathbf{x}^k$ for solving the psd system $\mathbf{A}^{\mathsf{T}}\mathbf{A}\mathbf{x} = \mathbf{A}^{\mathsf{T}}\mathbf{b}$ with SC-RCD is equivalent to the following:

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \mathbf{e}_{\mathcal{J}}\boldsymbol{\alpha}^k + \mathbf{e}_{\mathcal{S}}\boldsymbol{\beta}^k, \tag{3.52}$$

where

$$\boldsymbol{\alpha}^k := (((\mathbf{A}^{\circ})^{\mathsf{T}}\mathbf{A}^{\circ})_{\mathcal{J},\mathcal{J}})^{\dagger}(\mathbf{A}_{:,\mathcal{J}})^{\mathsf{T}}\mathbf{r}^k \quad \text{and} \quad \boldsymbol{\beta}^k := \mathbf{C}_{:,\mathcal{J}}\boldsymbol{\alpha}^k,$$

with $\mathbf{C} := ((\mathbf{A}^{\mathsf{T}}\mathbf{A})_{\mathcal{S},\mathcal{S}})^{\dagger}(\mathbf{A}^{\mathsf{T}}\mathbf{A})_{\mathcal{S},:}$ and $\mathbf{r}^k = \mathbf{A}\mathbf{x}^k - \mathbf{b}$. Note that the subspace constraint maintains the invariant $(\mathbf{A}_{:,\mathcal{S}})^{\mathsf{T}}(\mathbf{A}\mathbf{x}^k - \mathbf{b}) = \mathbf{0}$. Therefore, $(\widehat{\mathbf{A}}_{:,\mathcal{J}})^{\mathsf{T}}\mathbf{r}^k = (\mathbf{A}_{:,\mathcal{J}})^{\mathsf{T}}\mathbf{\Pi}_{\mathbf{A},\mathcal{S}}\mathbf{r}^k = \mathbf{0}$, and we can replace $(\mathbf{A}_{:,\mathcal{J}})^{\mathsf{T}}\mathbf{r}^k$ with $(\mathbf{A}_{:,\mathcal{J}}^{\circ})^{\mathsf{T}}\mathbf{r}^k$ (i.e., $\boldsymbol{\alpha}^k$ is the solution of a highly overdetermined least squares problem $\arg\min_{\boldsymbol{\alpha}}\|\mathbf{A}_{:,\mathcal{J}}^{\circ}\boldsymbol{\alpha} - \mathbf{r}^k\|_2$). Furthermore, the update (3.44) for the residual vector $\mathbf{r}^k$ is equivalent to

$$\mathbf{r}^{k+1} = \mathbf{r}^k - (\mathbf{I} - \mathbf{\Pi}_{\mathbf{A},\mathcal{S}})\mathbf{A}_{:,\mathcal{J}}\boldsymbol{\alpha}^k = \mathbf{r}^k - \mathbf{A}_{:,\mathcal{J}}^{\circ}\boldsymbol{\alpha}^k. \tag{3.53}$$

The updates (3.52) and (3.53) are summarized in the pseudocode in Algorithm 3.2. Each iteration requires accessing the columns of $\mathbf{A}$ and $\widehat{\mathbf{A}}$ indexed by the sampled block $\mathcal{J}$. Similar

to the psd case (Section 3.3.2), the partial QR structure of the approximation $\widehat{\mathbf{A}}$ can be used to compute the auxiliary matrix $\mathbf{C} = (\mathbf{A}_{:,\mathcal{S}})^\dagger \mathbf{A}$ and a valid initialization $\mathbf{x}^0 = (\mathbf{A}_{:,\mathcal{S}})^\dagger \mathbf{b}$ more efficiently by observing that $(\mathbf{A}_{:,\mathcal{S}})^\dagger = (\mathbf{Q}\mathbf{R}_{:,\mathcal{S}})^\dagger = (\mathbf{R}_{:,\mathcal{S}})^{-1}\mathbf{Q}^\mathsf{T}$ if $\mathbf{A}_{:,\mathcal{S}}$ has full rank, recalling that $\mathbf{R}_{:,\mathcal{S}}$ is upper triangular.

The following result states the convergence rate of the least squares SC-RCD method (Algorithm 3.2), which immediately follows from Theorem 3.3.2 for a fixed pivot set $\mathcal{S}$. As in the psd SC-RCD case, this can be further combined with bounds for the quality of the low-rank approximation $\widehat{\mathbf{A}}$ such as [Che+25, Corollary 5.2], which we do not elaborate on.

Theorem 3.7.1. *Let $\mathbf{A} \in \mathbb{R}^{m\times n}$ and $\mathbf{x}^*$ be any solution of the least squares problem (3.49). Suppose that $\{\mathbf{x}^k\}_{k\geq 0}$ are the iterates defined by (3.52) with a fixed subset $\mathcal{S} \subseteq [n]$, and the block $\mathcal{J} = \{j_1, \dots, j_\ell\}$ in each iteration consists of ℓ columns independently sampled according to the distribution $\{\|\mathbf{A}^\circ_{:,j}\|_2^2/\|\mathbf{A}^\circ\|_F^2\}_{j=1}^n$. Then*

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|^2_{\mathbf{A}^\mathsf{T}\mathbf{A}} \leq \left(1 - \frac{\sigma^+_{\min}(\mathbf{A}^\circ)^2}{\|\mathbf{A}^\circ\|_F^2}\right)^{k\ell} \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|^2_{\mathbf{A}^\mathsf{T}\mathbf{A}},$$

where $\mathbf{A}^\circ = \mathbf{A} - \mathbf{\Pi}_{\mathbf{A},\mathcal{S}}\mathbf{A}$, and $\sigma^+_{\min}(\mathbf{A}^\circ)$ is the smallest non-zero singular value of $\mathbf{A}^\circ$. Note that $\|\mathbf{x}^k - \mathbf{x}^*\|^2_{\mathbf{A}^\mathsf{T}\mathbf{A}} = \|\mathbf{A}\mathbf{x}^k - \mathbf{b}\|_2^2 - \|\mathbf{A}\mathbf{x}^* - \mathbf{b}\|_2^2$ measures the suboptimality in the least squares objective.

3.8 Additional numerical experiments

In this section, we present additional experiments to corroborate the findings reported in Section 3.4. We adopt a similar KRR setup as in Figures 3.3 and 3.4: we take $n = 20,000$ samples $(\mathbf{z}_i, y_i) \in \mathbb{R}^p$ from a selection of datasets considered in [Día+24, Table 1], which are sourced from OpenML [Van+13] and LibSVM [CL11], and solve $(\mathbf{K} + \lambda\mathbf{I})\mathbf{x} = \mathbf{y}$ using the Gaussian kernel $\mathbf{K}$ with bandwidth $\sigma = 3$ and a small regularization parameter $\lambda = 10^{-8}n$.

We use SC-RCD (with $d = 1,000 \approx 7\sqrt{n}$ and $\ell = 1,000$), RCD (with $\ell = 1,000$), CG, and PCG (with $d = 1,000$).

Figures 3.5 and 3.6 report the convergence trajectories, showing the relative residual norm $\|(\mathbf{K} + \lambda\mathbf{I})\mathbf{x}^k - \mathbf{y}\|_2/\|\mathbf{y}\|_2$ over the first 300 epochs (the median and 0.2/0.8-quantiles over 10 independent runs are reported), and the corresponding eigenvalue spectra for eight datasets, loosely grouped in terms of whether the kernel matrix exhibits rapid or slower spectral decay.

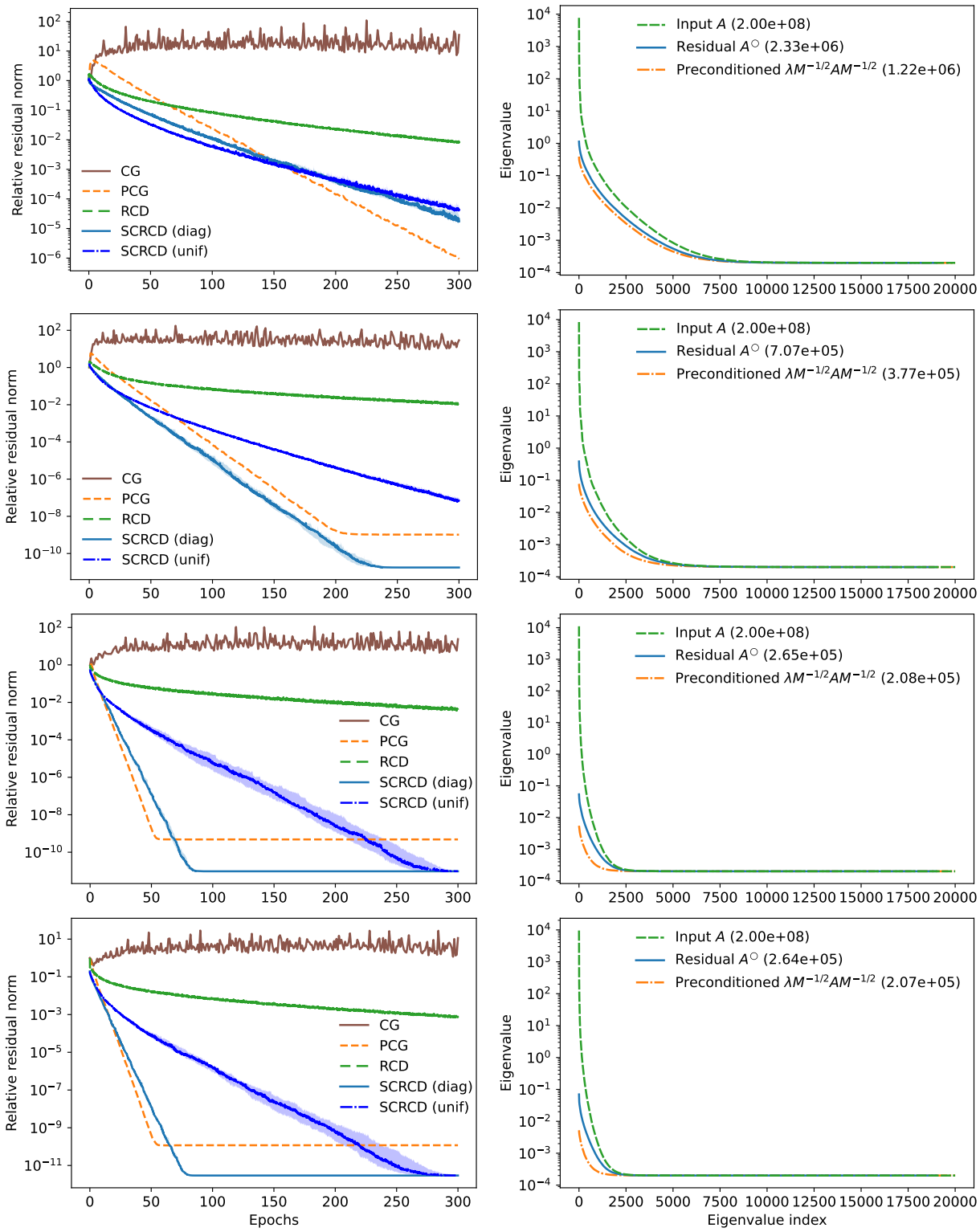


Figure 3.5: For kernel matrices exhibiting rapid spectral decay, the SC-RCD method is particularly effective: **(Left)** Convergence trajectories and **(right)** eigenvalues for the (i) `ACSIncome` ($p = 11$), (ii) `Airlines_DepDelay_1M` ($p = 9$), (iii) `cod-rna` ($p = 8$), and (iv) `diamonds` ($p = 9$) datasets.

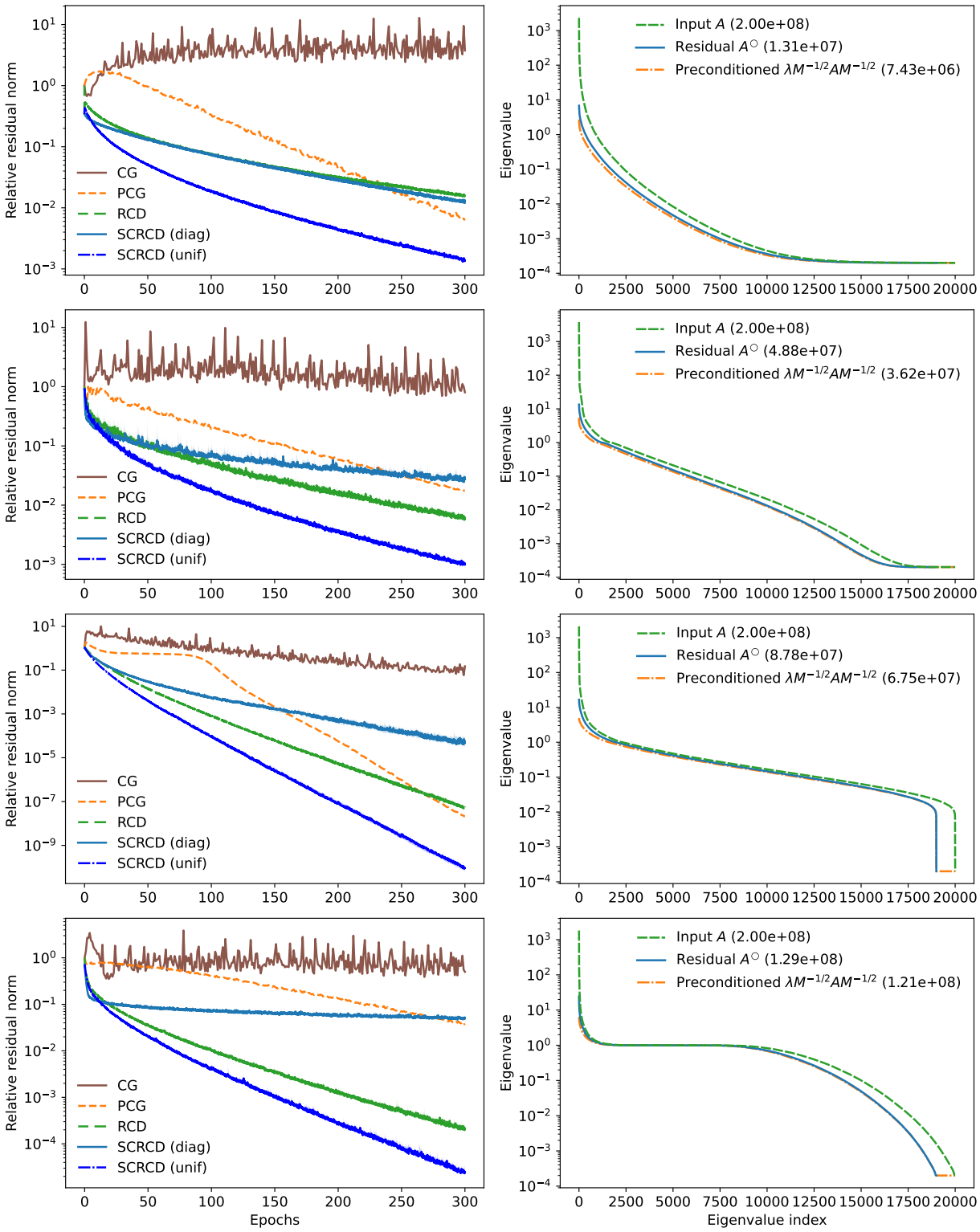


Figure 3.6: For matrices with slower spectral decay, SC-RCD with uniformly sampled blocks typically works quite well: **(Left)** Convergence trajectories and **(right)** eigenvalues for the (i) `covtype.binary` ($p = 54$), (ii) `creditcard` ($p = 29$), (iii) `HIGGS` ($p = 28$), and (iv) `SensIT Vehicle` ($p = 100$) datasets.

3.9 Accelerated subspace-constrained sketch-and-project

The sketch-and-project method can be interpreted as a form of stochastic gradient descent (SGD) [RT20; Gow+21]. Analogously, the subspace-constrained version can be interpreted as a form of *projected SGD* where the iterates are confined within (affine) subspace corresponding to the solutions of $\mathbf{QAx} = \mathbf{Qb}$ throughout. Indeed, from Lemma 3.2.1, the updates of subspace-constrained sketch-and-project are of the form $\mathbf{x}^{k+1} = \mathbf{x}^k - \mathbf{g}^k$, where $\mathbf{g}^k$ is a *projected gradient* associated with a random sketch of the linear system.

In this section, we will describe an *accelerated subspace-constrained sketch-and-project algorithm* for solving $\mathbf{Ax} = \mathbf{b}$ with $\mathbf{A} \in \mathbb{R}^{m\times n}$ that incorporates a Nesterov momentum term. We will adapt a formulation of the accelerated randomized block Kaczmarz algorithm recently proposed by [Der+25b], building on previous accelerated algorithms analyzed by [Tu+17; Gow+18]. Recall that $\mathbf{P}$ and $\mathbf{Z}$ are the orthogonal projectors onto $\text{null}(\mathbf{QAB}^{-1/2})$ and $\text{range}(\mathbf{PB}^{-1/2}\mathbf{A}^\top\mathbf{S}^\top)$, respectively.

Given parameters $\widehat{\rho}, \widehat{\eta} \in [0, 1]$ and any initial iterate $\mathbf{x}^0$ satisfying $\mathbf{QAx}^0 = \mathbf{Qb}$, consider the following procedure. Set $\mathbf{m}^0 = \mathbf{0}$, and for $k = 0, 1, 2, \ldots$, draw a sketching matrix $\mathbf{S} \equiv \mathbf{S}^{k+1}$ independently from an input distribution $\mathcal{D}$, and compute

$$\mathbf{g}^k = \mathbf{B}^{-1/2}\mathbf{PB}^{-1/2}\mathbf{A}^\top\mathbf{S}^\top(\mathbf{SAB}^{-1/2}\mathbf{PB}^{-1/2}\mathbf{A}^\top\mathbf{S}^\top)^\dagger\mathbf{S}(\mathbf{Ax}^k - \mathbf{b}), \tag{3.54}$$

$$\mathbf{m}^{k+1} = \frac{1-\widehat{\rho}}{1+\widehat{\rho}}(\mathbf{m}^k - \mathbf{g}^k), \tag{3.55}$$

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \mathbf{g}^k + \widehat{\eta}\cdot\mathbf{m}^{k+1}. \tag{3.56}$$

The following result bounds the convergence rate of accelerated subspace-constrained sketch-and-project if the parameters $\widehat{\rho}$ and $\widehat{\eta}$ are chosen based on quantities related to the spectrum of $\mathbb{E}[\mathbf{Z}]$ (analogous to the first and second moments).

Theorem 3.9.1. *Let* $\mathbf{P}$ *and* $\mathbf{Z}$ *be the orthogonal projection matrices onto* $\text{null}(\mathbf{QAB}^{-1/2})$ *and* $\text{range}(\mathbf{PB}^{-1/2}\mathbf{A}^\top\mathbf{S}^\top)$*, respectively, as in Lemma 3.2.1. Assume that the exactness con-*

dition (3.30) *holds:* $\text{null}(\mathbb{E}[\mathbf{Z}]) = \text{null}(\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P})$. *Define*

$$\mu := \lambda_{\min}^{+}(\mathbb{E}[\mathbf{Z}]) \quad \textit{and} \quad \nu := \lambda_{\max}\left(\mathbb{E}\left[(\mathbb{E}[\mathbf{Z}]^{\dagger/2}\mathbf{Z}\mathbb{E}[\mathbf{Z}]^{\dagger/2})^2\right]\right). \tag{3.57}$$

If $\{\mathbf{x}^k\}_{k\geq 0}$ *is computed according to the procedure* (3.54)–(3.56) *using parameters* $\widehat{\eta}, \widehat{\rho} \in [0,1]$ *that satisfy*

$$\widehat{\eta} \leq \frac{\nu^{-1} - \widehat{\rho}}{1 - \widehat{\rho}} \quad \textit{and} \quad \frac{\widehat{\rho}(\widehat{\rho} - \mu)}{1 - \widehat{\rho}} \leq \mu\widehat{\eta}, \tag{3.58}$$

then

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{B}}^2 \leq 8\left(1 - \widehat{\rho}\right)^k \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_{\mathbf{B}}^2. \tag{3.59}$$

Remark 3.9.2.

1. A choice of parameters that saturate the bounds in (3.58) are given by

$$\rho := \sqrt{\frac{\mu}{\nu}} \quad \text{and} \quad \eta := \frac{\nu^{-1} - \rho}{1 - \rho}. \tag{3.60}$$

   By choosing $\widehat{\eta} = \eta$ and $\widehat{\rho} = \rho$, (3.59) implies that the iterates $\{\mathbf{x}^k\}_{k\geq 0}$ converge with rate $1 - \sqrt{\mu/\nu}$. It can be shown that $1 \leq \nu \leq 1/\mu$ (see Lemma 3.9.5). Hence, *the accelerated method with the optimal parameters converges faster than the unaccelerated method*, which has rate $1-\mu$ from Theorem 3.2.4. On the other end, note that choosing $\widehat{\eta} = 0$ and (necessarily) $\widehat{\rho} = \mu$ also recovers the unaccelerated method. Thus, the choice of $\widehat{\eta}$ and $\widehat{\rho}$ satisfying the conditions (3.58) offers a smooth interpolation between the optimal accelerated and unaccelerated methods, and suggests a certain robustness in the estimation of the parameters.

2. Given estimates $\widehat{\mu}$ and $\widehat{\nu}$ that satisfy $\widehat{\mu} \leq \mu$, $\widehat{\nu} \geq \nu$, and $\widehat{\nu} \leq 1/\widehat{\mu}$, a valid set of parameters $\widehat{\rho}, \widehat{\eta}$ satisfying (3.58) can be obtained by setting

$$\widehat{\rho} = \sqrt{\frac{\widehat{\mu}}{\widehat{\nu}}} \quad \text{and} \quad \widehat{\eta} = \frac{\widehat{\nu}^{-1} - \widehat{\rho}}{1 - \widehat{\rho}}. \tag{3.61}$$

3. The conditions on $\widehat{\rho}$ in (3.58) imply that $\widehat{\rho} \leq \rho$. Indeed, substituting the first condition into the second yields $\widehat{\rho}^2 - \mu\widehat{\rho} = \widehat{\rho}(\widehat{\rho} - \mu) \leq \mu\widehat{\eta}(1 - \widehat{\rho}) \leq \mu(\nu^{-1} - \widehat{\rho}) = \rho^2 - \mu\widehat{\rho}$. The first condition also implies that $\widehat{\eta} \leq 1/\nu$.

To prove Theorem 3.9.1, we will consider another formulation of the accelerated sketch-and-project method [Tu+17; Gow+18] that embeds the sequence $\{\mathbf{x}^k\}_{k\geq 0}$ from (3.54), and allows for the convergence rate to be bounded using a Lyapunov-style analysis [WRJ21]. Specifically, let $\widehat{\mu}$ and $\widehat{\nu}$ be positive parameters satisfying $\widehat{\mu} \leq \widehat{\nu}$, and define

$$\beta := 1 - \sqrt{\frac{\widehat{\mu}}{\widehat{\nu}}}, \quad \gamma := \frac{1}{\sqrt{\widehat{\mu}\widehat{\nu}}}, \quad \alpha := \frac{1}{1 + \gamma\widehat{\nu}}. \tag{3.62}$$

Given any initial iterate $\mathbf{x}^0$ satisfying $\mathbf{QAx}^0 = \mathbf{Qb}$, set $\mathbf{v}^0 = \mathbf{y}^0 = \mathbf{x}^0$. For $k = 0, 1, 2, \ldots$, draw a sketching matrix $\mathbf{S} \equiv \mathbf{S}^{k+1}$ independently from an input distribution $\mathcal{D}$, and compute

$$\mathbf{g}^k = \mathbf{B}^{-1/2}\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T}(\mathbf{S}\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T}\mathbf{S}^\mathsf{T})^\dagger\mathbf{S}(\mathbf{A}\mathbf{x}^k - \mathbf{b}), \tag{3.63}$$

$$\mathbf{y}^{k+1} = \mathbf{x}^k - \mathbf{g}^k, \tag{3.64}$$

$$\mathbf{v}^{k+1} = \beta\mathbf{v}^k + (1 - \beta)\mathbf{x}^k - \gamma\mathbf{g}^k, \tag{3.65}$$

$$\mathbf{x}^{k+1} = \alpha\mathbf{v}^{k+1} + (1 - \alpha)\mathbf{y}^{k+1}. \tag{3.66}$$

Note that from (3.64), $\mathbf{y}^{k+1}$ is obtained from a subspace-constrained sketch-and-project update from $\mathbf{x}^k$ (Lemma 3.2.1). Furthermore, observe that the differences $\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*), \mathbf{B}^{1/2}(\mathbf{y}^k - \mathbf{x}^*), \mathbf{B}^{1/2}(\mathbf{v}^k - \mathbf{x}^*) \in \mathrm{range}(\mathbf{P}) = \mathrm{null}(\mathbf{QAB}^{-1/2})$ throughout, which shows that the iterates $\mathbf{x}^k, \mathbf{y}^k, \mathbf{v}^k$ are constrained within the solution space $\mathbf{QAx} = \mathbf{Qb}$.

The following theorem describes the convergence rate of the accelerated subspace-constrained sketch-and-project method formulated in (3.63)–(3.66). In the statement, we use the notation $\|\mathbf{z}\|^2_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}} = \mathbf{z}^\mathsf{T}\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}\mathbf{z}$ to denote the semi-norm induced by $\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}$.

Theorem 3.9.3. *Suppose that the same notation as Theorem 3.9.1 is used. Assume that the exactness condition (3.30) holds:* $\text{null}(\mathbb{E}[\mathbf{Z}]) = \text{null}(\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P})$. *Suppose that* $\{(\mathbf{x}^k, \mathbf{y}^k, \mathbf{v}^k)\}_{k\geq 0}$ *are the iterates from the procedure (3.63)–(3.66) using positive parameters* $\widehat{\mu}, \widehat{\nu} > 0$ *that satisfy* $\widehat{\mu} \leq \mu$ *and* $\widehat{\nu} \geq \nu$. *If we define*

$$\Delta^k := \|\mathbf{v}^k - \mathbf{x}^*\|^2_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}} + \widehat{\mu}^{-1}\|\mathbf{y}^k - \mathbf{x}^*\|^2_{\mathbf{B}},$$

then

$$\mathbb{E}\Delta^k \leq \left(1 - \sqrt{\frac{\widehat{\mu}}{\widehat{\nu}}}\right)^k \cdot \Delta^0. \tag{3.67}$$

In particular, this implies the bounds

$$\mathbb{E}\|\mathbf{y}^k - \mathbf{x}^*\|^2_{\mathbf{B}} \leq 2\left(1 - \sqrt{\frac{\widehat{\mu}}{\widehat{\nu}}}\right)^k \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|^2_{\mathbf{B}} \tag{3.68}$$

and

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|^2_{\mathbf{B}} \leq 8\left(1 - \sqrt{\frac{\widehat{\mu}}{\widehat{\nu}}}\right)^k \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|^2_{\mathbf{B}}. \tag{3.69}$$

The proof of Theorem 3.9.3 uses essentially the same argument used in the proof of [Gow+18, Theorem 3] for the accelerated sketch-and-project method, and involves some fairly technical and lengthy calculations. The main difference is that the subspace-constrained version incorporates the projector $\mathbf{P}$ to enforce the subspace constraint, which requires more careful reasoning about subspaces. Moreover, our formulation allows for the algorithm to use an *underestimate* $\widehat{\mu}$ of μ and an *overestimate* $\widehat{\nu}$ of ν. We will prove Theorem 3.9.3 in Section 3.9.1.

To show that Theorem 3.9.1 follows from Theorem 3.9.3 , we simply have to identify how the two formulations of accelerated subspace-constrained sketch-and-project are equivalent under appropriate reparameterizations, which is given by the following lemma.

Lemma 3.9.4. *The iterates $\{x^k\}_{k\geq 0}$ from the procedure (3.63)–(3.66) using parameters $\widehat{\mu}$ and $\widehat{\nu}$ are the same as the iterates $\{x^k\}_{k\geq 0}$ from the procedure (3.54)–(3.56) using parameters*

$$\widehat{\rho} = \sqrt{\frac{\widehat{\mu}}{\widehat{\nu}}} \quad \text{and} \quad \widehat{\eta} = \frac{\widehat{\nu}^{-1} - \widehat{\rho}}{1 - \widehat{\rho}}. \tag{3.70}$$

Conversely, the iterates $\{x^k\}_{k\geq 0}$ from the procedure (3.54)–(3.56) using parameters $\widehat{\rho}$ and $\widehat{\eta}$ are the same as the iterates $\{x^k\}_{k\geq 0}$ from the procedure (3.63)–(3.66) using parameters

$$\widehat{\mu} = \widehat{\rho}^2\widehat{\nu} \quad \text{and} \quad \widehat{\nu} = \frac{1}{\widehat{\rho} + \widehat{\eta}(1 - \widehat{\rho})}. \tag{3.71}$$

The proof of equivalence follows from some straightforward algebra and the definitions, and will be deferred to Section 3.9.2. To conclude, we finish by verifying that Theorem 3.9.1 follows from Theorem 3.9.3 and Lemma 3.9.4.

Proof of Theorem 3.9.1. Since Lemma 3.9.4 shows that the sequence $\{x^k\}_{k\geq 0}$ from (3.56) embeds into the sequence from (3.66) with parameters $\widehat{\mu} = \widehat{\rho}^2\widehat{\nu}$ and $\widehat{\nu} = 1/(\widehat{\rho} + \widehat{\eta}(1 - \widehat{\rho}))$, it suffices to check that $\widehat{\mu} \leq \mu$ and $\widehat{\nu} \geq \nu$ so that (3.69) from Theorem 3.9.3 can be applied. Indeed, the conditions in (3.58) are rearrangements of the inequalities $\widehat{\nu} \geq \nu$ and $\widehat{\mu} \leq \mu$, respectively, in terms of $\widehat{\rho}$ and $\widehat{\eta}$. □

3.9.1 Proof of Theorem 3.9.3

In this section, we will prove Theorem 3.9.3, which bounds the convergence rate of the subspace-constrained sketch-and-project procedure (3.63)–(3.66). First, we prove some properties of the moments μ and ν related to the expected projector $\mathbb{E}[\mathbf{Z}]$ as in (3.57). The approach is similar to the proofs in [Gow+18, Appendix A.1].

Lemma 3.9.5 (Properties of μ and ν)**.** *Assume that the exactness condition (3.30) holds:* $\operatorname{null}(\mathbb{E}[\mathbf{Z}]) = \operatorname{null}(\mathbf{A}\mathbf{B}^{-1/2}\mathbf{P})$. *Let $\mu = \lambda_{\min}^{+}(\mathbb{E}[\mathbf{Z}])$ and $\nu = \lambda_{\max}\left(\mathbb{E}\left[(\mathbb{E}[\mathbf{Z}]^{\dagger/2}\mathbf{Z}\mathbb{E}[\mathbf{Z}]^{\dagger/2})^2\right]\right)$.*

Then, we have the following equivalent variational characterizations of μ and ν:

$$\mu = \inf_{\mathbf{x}\in\text{range}(\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T})} \frac{\langle \mathbb{E}[\mathbf{Z}]\mathbf{x}, \mathbf{x}\rangle}{\langle \mathbf{x}, \mathbf{x}\rangle} \quad \textit{and} \quad \nu = \sup_{\mathbf{x}\in\text{range}(\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T})} \frac{\left\langle \mathbb{E}[\mathbf{Z}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{Z}]\mathbf{x}, \mathbf{x}\right\rangle}{\langle \mathbb{E}[\mathbf{Z}]\mathbf{x}, \mathbf{x}\rangle}. \tag{3.72}$$

Furthermore, $\nu \geq 1$ and $\mu \leq 1/\nu$.

Proof. The variational characterization for $\mu = \lambda_{\min}^+(\mathbb{E}[\mathbf{Z}])$ follows from applying the standard min-max theorem for the smallest eigenvalue of a symmetric matrix and using the exactness condition to identify the nullspace. We do the same for $\nu = \lambda_{\max}\left(\mathbb{E}[\mathbf{Z}]^{\dagger/2}\mathbb{E}\left[\mathbf{Z}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{Z}\right]\mathbb{E}[\mathbf{Z}]^{\dagger/2}\right)$, together with the substitution $\mathbf{y} = \mathbb{E}[\mathbf{Z}]^{1/2}\mathbf{x}$ and using the fact that $\mathbb{E}[\mathbf{Z}]^{\dagger/2}\mathbb{E}[\mathbf{Z}]^{1/2}$ is the identity on range$(\mathbb{E}[\mathbf{Z}])$.

Next, by Jensen's inequality and convexity of the map $\mathbf{Z} \mapsto \left\langle \mathbf{Z}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{Z}\mathbf{x}, \mathbf{x}\right\rangle = \|\mathbb{E}[\mathbf{Z}]^{\dagger/2}\mathbf{Z}\mathbf{x}\|^2$ for fixed $\mathbf{x}$, we have

$$\mathbb{E}\left[\left\langle \mathbf{Z}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{Z}\mathbf{x}, \mathbf{x}\right\rangle\right] \geq \left\langle \mathbb{E}[\mathbf{Z}]\mathbb{E}[\mathbf{Z}]^\dagger\mathbb{E}[\mathbf{Z}]\mathbf{x}, \mathbf{x}\right\rangle = \left\langle \mathbb{E}[\mathbf{Z}]\mathbf{x}, \mathbf{x}\right\rangle.$$

Therefore, from the variational characterization of ν we deduce that

$$\nu \geq \sup_{\mathbf{x}\in\text{range}(\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T})} \frac{\langle \mathbb{E}[\mathbf{Z}]\mathbf{x}, \mathbf{x}\rangle}{\langle \mathbb{E}[\mathbf{Z}]\mathbf{x}, \mathbf{x}\rangle} = 1.$$

By using the variational characterization of ν again and making the substitution $\mathbf{x} = \mathbb{E}[\mathbf{Z}]^{\dagger/2}\mathbf{y}$, we have

$$\nu = \sup_{\mathbf{x}\in\text{range}(\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T})} \frac{\mathbb{E}\left[\left\langle \mathbb{E}[\mathbf{Z}]^\dagger\mathbf{Z}\mathbf{x}, \mathbf{Z}\mathbf{x}\right\rangle\right]}{\langle \mathbb{E}[\mathbf{Z}]\mathbf{x}, \mathbf{x}\rangle} \leq \sup_{\mathbf{x}\in\text{range}(\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T})} \frac{\|\mathbb{E}[\mathbf{Z}]^\dagger\|\mathbb{E}\|\mathbf{Z}\mathbf{x}\|^2}{\langle \mathbb{E}[\mathbf{Z}]\mathbf{x}, \mathbf{x}\rangle} = \|\mathbb{E}[\mathbf{Z}]^\dagger\| \leq \frac{1}{\mu}.$$

This completes the proof. □

Furthermore, from the definition of the iterates, it is clear that the differences satisfy the following invariance property, analogous to Lemma 3.2.2.

Lemma 3.9.6 (Invariance property)**.** *For all* $k \geq 0$, $\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*), \mathbf{B}^{1/2}(\mathbf{y}^k - \mathbf{x}^*), \mathbf{B}^{1/2}(\mathbf{v}^k - \mathbf{x}^*) \in \text{range}(\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T})$.

We can now proceed to prove Theorem 3.9.3.

Proof of Theorem 3.9.3. Let $r_k := \|\mathbf{v}^k - \mathbf{x}^*\|_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}}$. Our goal is to bound the expectation, conditional on the randomness up to time k, of r_{k+1}^2 and $\widehat{\mu}^{-1}\|\mathbf{y}^{k+1} - \mathbf{x}^*\|_{\mathbf{B}}^2$ in terms of an expression involving r_k^2 and $\widehat{\mu}^{-1}\|\mathbf{y}^k - \mathbf{x}^*\|_{\mathbf{B}}^2$ to set up a recursion.

Since $\mathbf{v}^{k+1} = \beta\mathbf{v}^k + (1-\beta)\mathbf{x}^k - \gamma\mathbf{g}^k$ and $\mathbf{g}^k = \mathbf{B}^{-1/2}\mathbf{Z}\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)$, we have

$$\begin{aligned} r_{k+1}^2 &= \|\mathbf{v}^{k+1} - \mathbf{x}^*\|_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}}^2 \\ &= \|\beta\mathbf{v}^k + (1-\beta)\mathbf{x}^k - \mathbf{x}^* - \gamma\mathbf{B}^{-1/2}\mathbf{Z}\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)\|_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}}^2. \end{aligned}$$

By expanding the square, we obtain

$$\begin{aligned} r_{k+1}^2 = \|\beta\mathbf{v}^k + (1-\beta)\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger}^2 + \gamma^2\|\mathbf{B}^{-1/2}\mathbf{Z}\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)\|_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}}^2 \\ - 2\gamma\left\langle\beta\mathbf{v}^k + (1-\beta)\mathbf{x}^k - \mathbf{x}^*, \mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}\mathbf{B}^{-1/2}\mathbf{Z}\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)\right\rangle. \end{aligned} \tag{3.73}$$

We will bound the three terms of (3.73) individually. For the first term, we expand the square and use the parallelogram identity ($2\langle\mathbf{u}, \mathbf{w}\rangle = \|\mathbf{u}\|^2 + \|\mathbf{w}\|^2 - \|\mathbf{u} - \mathbf{w}\|^2$) to compute

$$\begin{aligned} \|\beta\mathbf{v}^k + (1-\beta)\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}}^2 &= \|\beta(\mathbf{v}^k - \mathbf{x}^*) + (1-\beta)(\mathbf{x}^k - \mathbf{x}^*)\|_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}}^2 \\ &= \beta^2\|\mathbf{v}^k - \mathbf{x}^*\|_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger}^2 + (1-\beta)^2\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}}^2 \\ &\quad + 2\beta(1-\beta)\left\langle\mathbf{v}^k - \mathbf{x}^*, \mathbf{x}^k - \mathbf{x}^*\right\rangle_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}} \\ &= \beta\|\mathbf{v}^k - \mathbf{x}^*\|_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}}^2 + (1-\beta)\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}}^2 \\ &\quad - \beta(1-\beta)\|\mathbf{v}^k - \mathbf{x}^k\|_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}}^2. \end{aligned}$$

Therefore, because $\beta \leq 1$ and $\|\mathbb{E}[\mathbf{Z}]^\dagger\| = 1/\mu$, the first term satisfies the bound

$$\|\beta \mathbf{v}^k + (1-\beta)\mathbf{x}^k - \mathbf{x}^*\|^2_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger \mathbf{B}^{1/2}} \leq \beta r_k^2 + \frac{1-\beta}{\mu}\|\mathbf{x}^k - \mathbf{x}^*\|^2_{\mathbf{B}}. \tag{3.74}$$

Next, we bound the conditional expectation of the second term of (3.73) over the randomness in the $(k+1)^{\text{th}}$ iteration only. By using linearity of expectation and the fact that $\mathbf{Z}$ is identically distributed in each iteration, we obtain

$$\begin{aligned}
&\mathbb{E}_k\|\mathbf{B}^{-1/2}\mathbf{Z}\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)\|^2_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}} \\
&\quad = \mathbb{E}_k\left[\left\langle \mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}\mathbf{B}^{-1/2}\mathbf{Z}\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*), \mathbf{B}^{-1/2}\mathbf{Z}\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)\right\rangle\right] \\
&\quad = \left\langle \mathbb{E}\left[\mathbf{Z}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{Z}\right]\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*), \mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)\right\rangle.
\end{aligned}$$

Note that $\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*) \in \text{range}(\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T})$ from Lemma 3.9.6. Hence, by using the variational characterization of ν from Lemma 3.9.5 and $\nu \leq \widehat{\nu}$, we obtain the following bound for the second term:

$$\begin{aligned}
\mathbb{E}_k\|\mathbf{B}^{-1/2}\mathbf{Z}\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)\|^2_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}} &\leq \nu\left\langle \mathbb{E}[\mathbf{Z}]\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*), \mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)\right\rangle \\
&\leq \widehat{\nu}\|\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)\|^2_{\mathbb{E}[\mathbf{Z}]}.
\end{aligned} \tag{3.75}$$

Finally, we compute the conditional expectation of the third term of (3.73):

$$\begin{aligned}
&\mathbb{E}_k\left[\left\langle \beta\mathbf{v}^k + (1-\beta)\mathbf{x}^k - \mathbf{x}^*, \mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}\mathbf{B}^{-1/2}\mathbf{Z}\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)\right\rangle\right] \\
&\quad = \left\langle \beta\mathbf{v}^k + (1-\beta)\mathbf{x}^k - \mathbf{x}^*, \mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbb{E}[\mathbf{Z}]B^{1/2}(\mathbf{x}^k - \mathbf{x}^*)\right\rangle \\
&\quad = \left\langle \beta\mathbf{v}^k + (1-\beta)\mathbf{x}^k - \mathbf{x}^*, \mathbf{B}^{1/2}\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*)\right\rangle.
\end{aligned}$$

For the last equality, we used the fact that $\mathbb{E}[\mathbf{Z}]^\dagger\mathbb{E}[\mathbf{Z}]$ is the orthogonal projector onto range$(\mathbb{E}[\mathbf{Z}])$, and $\mathbf{B}^{1/2}(\mathbf{x}^k - \mathbf{x}^*) \in \text{range}(\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T}) = \text{range}(\mathbb{E}[\mathbf{Z}])$ from Lemma 3.9.6 and

the exactness assumption. Continuing, since $\mathbf{v}^k = \frac{1}{\alpha}(\mathbf{x}^k - (1-\alpha)\mathbf{y}^k)$,

$$\begin{aligned}
&\mathbb{E}_k\left[\left\langle \beta\mathbf{v}^k + (1-\beta)\mathbf{x}^k - \mathbf{x}^*, B^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}\mathbf{B}^{-1/2}\mathbf{Z}\mathbf{B}^{1/2}(\mathbf{x}^k-\mathbf{x}^*)\right\rangle\right] \\
&= \left\langle \beta\frac{1}{\alpha}\mathbf{x}^k - \beta\frac{1-\alpha}{\alpha}\mathbf{y}^k + (1-\beta)\mathbf{x}^k - \mathbf{x}^*, \mathbf{B}(\mathbf{x}^k-\mathbf{x}^*)\right\rangle \\
&= \left\langle \mathbf{x}^k - \mathbf{x}^* + \beta\frac{1-\alpha}{\alpha}(\mathbf{x}^k-\mathbf{y}^k), \mathbf{B}(\mathbf{x}^k-\mathbf{x}^*)\right\rangle \\
&= \|\mathbf{x}^k-\mathbf{x}^*\|_{\mathbf{B}}^2 + \beta\frac{1-\alpha}{\alpha}\left\langle \mathbf{x}^k-\mathbf{y}^k, \mathbf{x}^k-\mathbf{x}^*\right\rangle_{\mathbf{B}} \\
&= \|\mathbf{x}^k-\mathbf{x}^*\|_{\mathbf{B}}^2 - \beta\frac{1-\alpha}{2\alpha}(\|\mathbf{y}^k-\mathbf{x}^*\|_{\mathbf{B}}^2 - \|\mathbf{x}^k-\mathbf{y}^k\|_{\mathbf{B}}^2 - \|\mathbf{x}^k-\mathbf{x}^*\|_{\mathbf{B}}^2), \qquad (3.76)
\end{aligned}$$

where we used the parallelogram identity again for the last equality.

By collecting the bounds in (3.74), (3.75) and (3.76) for (3.73), we have shown that

$$\begin{aligned}
\mathbb{E}_k[r_{k+1}^2] \le \beta r_k^2 + \frac{1-\beta}{\mu}\|\mathbf{x}^k-\mathbf{x}^*\|_{\mathbf{B}}^2 + \gamma^2\widehat{\nu}\|\mathbf{B}^{1/2}(\mathbf{x}^k-\mathbf{x}^*)\|_{\mathbb{E}[\mathbf{Z}]}^2 \\
- 2\gamma\left(\|\mathbf{x}^k-\mathbf{x}^*\|_{\mathbf{B}}^2 - \beta\frac{1-\alpha}{2\alpha}(\|\mathbf{y}^k-\mathbf{x}^*\|_{\mathbf{B}}^2 - \|\mathbf{x}^k-\mathbf{y}^k\|_{\mathbf{B}}^2 - \|\mathbf{x}^k-\mathbf{x}^*\|_{\mathbf{B}}^2)\right). \qquad (3.77)
\end{aligned}$$

Next, our goal is to rewrite $\|\mathbf{B}^{1/2}(\mathbf{x}^k-\mathbf{x}^*)\|_{\mathbb{E}[\mathbf{Z}]}^2$ in terms of the $\mathbf{B}$-norm. Note that from Lemma 3.2.1, we have $\mathbf{B}^{1/2}(\mathbf{y}^{k+1}-\mathbf{x}^*) = (\mathbf{I}-\mathbf{Z})\mathbf{B}^{1/2}(\mathbf{x}^k-\mathbf{x}^*)$, so

$$\begin{aligned}
\mathbb{E}_k\|\mathbf{y}^{k+1}-\mathbf{x}^*\|_{\mathbf{B}}^2 &= \left\langle (\mathbf{I}-\mathbb{E}[\mathbf{Z}])\mathbf{B}^{1/2}(\mathbf{x}^k-\mathbf{x}^*), \mathbf{B}^{1/2}(\mathbf{x}^k-\mathbf{x}^*)\right\rangle \\
&= \|\mathbf{B}^{1/2}(\mathbf{x}^k-\mathbf{x}^*)\|^2 - \|\mathbf{B}^{1/2}(\mathbf{x}^k-\mathbf{x}^*)\|_{\mathbb{E}[\mathbf{Z}]}^2.
\end{aligned}$$

Rearranging, $\|\mathbf{B}^{1/2}(\mathbf{x}^k-\mathbf{x}^*)\|_{\mathbb{E}[\mathbf{Z}]}^2 = \|\mathbf{x}^k-\mathbf{x}^*\|_{\mathbf{B}}^2 - \mathbb{E}_k\|\mathbf{y}^{k+1}-\mathbf{x}^*\|_{\mathbf{B}}^2$. By substituting this into (3.77), we obtain

$$\begin{aligned}
\mathbb{E}_k[r_{k+1}^2] \le \beta r_k^2 + \frac{1-\beta}{\mu}\|\mathbf{x}^k-\mathbf{x}^*\|_{\mathbf{B}}^2 + \gamma^2\widehat{\nu}(\|\mathbf{x}^k-\mathbf{x}^*\|_{\mathbf{B}}^2 - \mathbb{E}_k\|\mathbf{y}^{k+1}-\mathbf{x}^*\|_{\mathbf{B}}^2) \\
- 2\gamma\left(\|\mathbf{x}^k-\mathbf{x}^*\|_{\mathbf{B}}^2 - \beta\frac{1-\alpha}{2\alpha}(\|\mathbf{y}^k-\mathbf{x}^*\|_{\mathbf{B}}^2 - \|\mathbf{x}^k-\mathbf{y}^k\|_{\mathbf{B}}^2 - \|\mathbf{x}^k-\mathbf{x}^*\|_{\mathbf{B}}^2)\right).
\end{aligned}$$

With some further rearranging, this implies that

$$\begin{aligned}\mathbb{E}_k\left[r_{k+1}^2+\gamma^2\widehat{\nu}\|\mathbf{y}^{k+1}-\mathbf{x}^*\|_{\mathbf{B}}^2\right] \leq \beta\left(r_k^2+\gamma\frac{1-\alpha}{\alpha}\|\mathbf{y}^k-\mathbf{x}^*\|_{\mathbf{B}}^2\right)\\ +\left(\frac{1-\beta}{\mu}+\gamma^2\widehat{\nu}-2\gamma-\beta\gamma\frac{1-\alpha}{\alpha}\right)\|\mathbf{x}^k-\mathbf{x}^*\|_{\mathbf{B}}^2.\end{aligned} \tag{3.78}$$

The inspired choice of the parameters β, γ, and α is such that

$$\gamma\frac{1-\alpha}{\alpha}=\gamma^2\widehat{\nu} \quad \text{and} \quad \frac{1-\beta}{\mu}+\gamma^2\widehat{\nu}-2\gamma-\beta\gamma\frac{1-\alpha}{\alpha}\leq 0. \tag{3.79}$$

(Note that if $\widehat{\mu}=\mu$ and $\widehat{\nu}=\nu$, then we have equality for the latter equation.) This can be directly verified from the following algebraic identities, together with $\widehat{\mu}\leq\mu$:

$$\frac{1-\beta}{\mu}=\frac{1}{\sqrt{\widehat{\mu}\widehat{\nu}}}\cdot\frac{\widehat{\mu}}{\mu}\leq\gamma, \quad \frac{1-\alpha}{\alpha}=\gamma\widehat{\nu}, \quad \text{and} \quad \beta\gamma^2\widehat{\nu}=\gamma^2\widehat{\nu}-\gamma.$$

Furthermore, observe that $\gamma^2\widehat{\nu}=1/\widehat{\mu}$. To conclude, from combining the bound (3.78) with the conditions (3.79) resulting from the choice of parameters β, γ, and α, we have shown that

$$\mathbb{E}_k\left[r_{k+1}^2+\frac{1}{\widehat{\mu}}\|\mathbf{y}^{k+1}-\mathbf{x}^*\|_{\mathbf{B}}^2\right]\leq\beta\left(r_k^2+\frac{1}{\widehat{\mu}}\|\mathbf{y}^k-\mathbf{x}^*\|_{\mathbf{B}}^2\right).$$

That is, $\mathbb{E}_k\Delta^{k+1}\leq\left(1-\sqrt{\frac{\widehat{\mu}}{\widehat{\nu}}}\right)\cdot\Delta^k$. By iterating, this implies (3.67).

Next, to show that (3.67) implies (3.68), it suffices to observe that

$$\|\mathbf{v}^0-\mathbf{x}^*\|_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}}^2\leq\frac{1}{\widehat{\mu}}\|\mathbf{v}^0-\mathbf{x}^*\|_{\mathbf{B}}^2=\frac{1}{\widehat{\mu}}\|\mathbf{x}^0-\mathbf{x}^*\|_{\mathbf{B}}^2,$$

where we used the fact that $\|\mathbb{E}[\mathbf{Z}]^\dagger\|=1/\mu\leq 1/\widehat{\mu}$ again. This implies that $\Delta^0\leq 2\widehat{\mu}^{-1}\|\mathbf{x}^0-\mathbf{x}^*\|_{\mathbf{B}}^2$. Since $\|\mathbf{y}^k-\mathbf{x}^*\|_{\mathbf{B}}^2\leq\widehat{\mu}\Delta^k$, combining these bounds in (3.67) leads to (3.68).

Finally, we will show that (3.67) implies (3.69). Since $\mathbf{x}^k - \mathbf{x}^* = \alpha(\mathbf{v}^k - \mathbf{x}^*) + (1-\alpha)(\mathbf{y}^k - \mathbf{x}^*)$, we have

$$\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{B}}^2 \leq 2\alpha^2 \|\mathbf{v}^k - \mathbf{x}^*\|_{\mathbf{B}}^2 + 2(1-\alpha)^2 \|\mathbf{y}^k - \mathbf{x}^*\|_{\mathbf{B}}^2.$$

Note that since (as operators) $\mathbb{E}[\mathbf{Z}]^\dagger\big|_{\text{range}(\mathbb{E}[\mathbf{Z}])} \succeq \mathbf{I}\big|_{\text{range}(\mathbb{E}[\mathbf{Z}])}$ and $\mathbf{B}^{1/2}(\mathbf{v}^k - \mathbf{x}^*) \in \text{range}(\mathbf{P}\mathbf{B}^{-1/2}\mathbf{A}^\mathsf{T}) = \text{range}(\mathbb{E}[\mathbf{Z}])$ from Lemma 3.9.6 and the exactness assumption, we have

$$\|\mathbf{v}^k - \mathbf{x}^*\|_{\mathbf{B}}^2 \leq \|\mathbf{v}^k - \mathbf{x}^*\|_{\mathbf{B}^{1/2}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{B}^{1/2}}^2 \leq \Delta^k.$$

Hence, using (3.67) and $\|\mathbf{y}^k - \mathbf{x}^*\|_{\mathbf{B}}^2 \leq \widehat{\mu}\Delta^k$ and $\Delta^0 \leq 2\widehat{\mu}^{-1}\|\mathbf{x}^0 - \mathbf{x}^*\|_{\mathbf{B}}^2$ from above, we have

$$\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{B}}^2 \leq 2(\alpha^2 + \widehat{\mu}(1-\alpha)^2)\Delta^k \leq 4\left(\frac{\alpha^2}{\widehat{\mu}} + (1-\alpha)^2\right) \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_{\mathbf{B}}^2.$$

From some brief calculation, it follows that

$$\frac{\alpha^2}{\mu} + (1-\alpha)^2 = \frac{1 + \frac{1}{\widehat{\nu}}}{\left(1 + \sqrt{\frac{\widehat{\mu}}{\widehat{\nu}}}\right)^2} \leq 2,$$

since $\widehat{\nu} \geq \nu \geq 1$ and $\widehat{\mu}, \widehat{\nu} \geq 0$. Combining this with the previous displayed bound leads to (3.69), which completes the proof. □

3.9.2 Proof of Lemma 3.9.4

Next, we will prove Lemma 3.9.4, which shows the equivalence between the two formulations (3.54)–(3.56) and (3.63)–(3.66) of accelerated subspace-constrained sketch-and-project under appropriate reparameterizations.

Proof of Lemma 3.9.4. Let $\{(\mathbf{x}^k, \mathbf{y}^k, \mathbf{v}^k)\}_{k\geq 0}$ be the iterates from the procedure (3.63)–(3.66) using parameters $\widehat{\mu}$ and $\widehat{\nu}$. Observe that from (3.66), we have

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \mathbf{g}^k + \alpha(\mathbf{v}^{k+1} - \mathbf{y}^{k+1})$$

With some algebra, we can write

$$\frac{\beta(1-\alpha)}{\gamma - 1}(\mathbf{v}^{k+1} - \mathbf{y}^{k+1}) = \beta(1-\alpha)\left[\frac{\beta(1-\alpha)}{\gamma-1}(\mathbf{v}^k - \mathbf{y}^k) - \mathbf{g}^k\right].$$

Hence, if we define

$$\mathbf{m}^k = \frac{\beta(1-\alpha)}{\gamma-1}(\mathbf{v}^k - \mathbf{y}^k),$$

with $\mathbf{m}^0 = \mathbf{0}$, then we see that the sequence $\{\mathbf{m}^k\}_{k\geq 0}$ satisfies $\mathbf{m}^{k+1} = \beta(1-\alpha)(\mathbf{m}^k - \mathbf{g}^k)$. Furthermore, we can rewrite (3.66) as

$$\begin{aligned}\mathbf{x}^{k+1} &= \mathbf{x}^k - \mathbf{g}^k + \alpha(\gamma-1)(\mathbf{m}^k - \mathbf{g}^k) \\ &= \mathbf{x}^k - \mathbf{g}^k + \frac{\alpha(\gamma-1)}{\beta(1-\alpha)}\mathbf{m}^{k+1}.\end{aligned}$$

If we define $\widehat{\rho} = \widehat{\mu}/\widehat{\nu}$, then $\alpha = \widehat{\rho}/(1+\widehat{\rho})$, $\beta = 1 - \widehat{\rho}$, and $\gamma = 1/(\widehat{\nu}\widehat{\rho})$. Therefore,

$$\beta(1-\alpha) = \frac{1-\widehat{\rho}}{1+\widehat{\rho}} \quad \text{and} \quad \frac{\alpha(\gamma-1)}{\beta(1-\alpha)} = \frac{\widehat{\nu}^{-1} - \widehat{\rho}}{1-\widehat{\rho}} =: \widehat{\eta}.$$

This shows that the sequence $\{\mathbf{x}^k\}_{k\geq 0}$ can be written as

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \mathbf{g}^k + \widehat{\eta}\cdot\mathbf{m}^{k+1}, \quad \text{and} \quad \mathbf{m}^{k+1} = \frac{1-\widehat{\rho}}{1+\widehat{\rho}}(\mathbf{m}^k - \mathbf{g}^k),$$

which is exactly (3.56) with the parameters $\widehat{\eta}$ and $\widehat{\rho}$.

For the reverse direction, solving the equations for $\widehat{\rho}$ and $\widehat{\eta}$ from above in terms of $\widehat{\mu}$ and $\widehat{\nu}$ yields $\widehat{\nu} = 1/(\widehat{\rho} + \widehat{\eta}(1-\widehat{\rho}))$ and $\widehat{\mu} = \widehat{\rho}^2\widehat{\nu}$. Hence, unrolling the argument backwards

shows that the sequence $\{\mathbf{x}^k\}_{k\geq 0}$ from the procedure (3.54)–(3.56) is the same as the one from (3.63)–(3.66) with the parameters $\widehat{\mu}$ and $\widehat{\nu}$. □

3.10 Accelerated subspace-constrained randomized coordinate descent

We can derive an *accelerated subspace-constrained randomized coordinate descent method* for solving psd linear systems $\mathbf{A}\mathbf{x} = \mathbf{b}$ with $\mathbf{A} \in \mathbb{R}^{n\times n}$ by casting it as an instance of the subspace-constrained sketch-and-project algorithm from Section 3.9 and applying the procedure (3.54)–(3.56). In particular, note that we can reuse the majority of our analysis of the SC-RCD method from Section 3.3 and the pseudocode in Algorithm 3.1).

Specifically, by copying down the form of the SC-RCD gradient from (3.43), the accelerated subspace-constrained randomized coordinate descent method has the following update steps, given acceleration parameters $\widehat{\rho}, \widehat{\eta} \in [0, 1]$. First, we obtain a pivot set $\mathcal{S} \subseteq [n]$ using, e.g., RPCholesky and compute any initial $\mathbf{x}^0 \in \mathbb{R}^n$ satisfying $\mathbf{A}_{\mathcal{S},:}\mathbf{x}^0 = \mathbf{b}_{\mathcal{S}}$. Then, for each $k = 0, 1, 2, \ldots$, we sample a block $\mathcal{J} \subseteq [n]$ of ℓ i.i.d. coordinates with probability proportional to the diagonal of $\mathbf{A}^\circ$, and compute

$$\begin{aligned} \mathbf{g}^k &= (\mathbf{e}_{\mathcal{J}} - \mathbf{e}_{\mathcal{S}}(\mathbf{A}_{\mathcal{S},\mathcal{S}})^\dagger \mathbf{A}_{\mathcal{S},:}\mathbf{e}_{\mathcal{J}})(\mathbf{A}^\circ_{\mathcal{J},\mathcal{J}})^\dagger(\mathbf{A}_{\mathcal{J},:}\mathbf{x}^k - \mathbf{b}_{\mathcal{J}}), \\ \mathbf{m}^{k+1} &= \frac{1-\widehat{\rho}}{1+\widehat{\rho}}(\mathbf{m}^k - \mathbf{g}^k), \\ \mathbf{x}^{k+1} &= \mathbf{x}^k - \mathbf{g}^k + \widehat{\eta}\cdot\mathbf{m}^{k+1}. \end{aligned} \tag{3.80}$$

It remains to clarify how we should choose the parameters $\widehat{\rho}, \widehat{\eta}$. In the following section, we will show that for block size $\ell = 1$, there is a nice choice in terms of spectral quantities related to $\mathbf{A}^\circ$ that leads to a clean convergence rate.

3.10.1 Acceleration parameters with block size $\ell = 1$

From the proof of Theorem 3.46, we know that with block size one, the parameter $\mu = \lambda_{\min}^{+}(\mathbb{E}[\mathbf{Z}])$ from (3.57) is equal to $\mu = \lambda_{\min}^{+}(\mathbf{A}^{\circ})/\operatorname{tr}(\mathbf{A}^{\circ})$. The following result shows that $\nu = \lambda_{\max}\left(\mathbb{E}\left[(\mathbb{E}[\mathbf{Z}]^{\dagger/2}\mathbf{Z}\mathbb{E}[\mathbf{Z}]^{\dagger/2})^2\right]\right)$ also admits a simple, exact expression with block size one. This extends a result originally obtained for accelerated randomized coordinate descent by Tu et al. in [Tu+17] to the subspace-constrained framework.

Lemma 3.10.1 (Acceleration parameters with $\ell = 1$)**.** *Let* $\mathbf{Z} = \mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_{\mathcal{J}}(\mathbf{A}^{\circ}_{\mathcal{J},\mathcal{J}})^{\dagger}\mathbf{e}_{\mathcal{J}}^{\mathsf{T}}\mathbf{A}^{1/2}\mathbf{P}$ *be the orthogonal projector onto* $\operatorname{range}(\mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_{\mathcal{J}})$ *associated with the SC-RCD method. If* $\mathcal{J} = \{j\}$ *is sampled with probability* $\mathbf{A}^{\circ}_{j,j}/\operatorname{tr}(\mathbf{A}^{\circ})$*, then*

$$\mu = \frac{\lambda_{\min}^{+}(\mathbf{A}^{\circ})}{\operatorname{tr}(\mathbf{A}^{\circ})} \quad \text{and} \quad \nu \leq \frac{\operatorname{tr}(\mathbf{A}^{\circ})}{\min_{j:\mathbf{A}^{\circ}_{j,j}>0}\mathbf{A}^{\circ}_{j,j}}.$$

Proof. Recall that $\mathbb{E}[\mathbf{Z}] = \frac{1}{\operatorname{tr}(\mathbf{A}^{\circ})}\mathbf{P}\mathbf{A}\mathbf{P}$ from (3.46), which implies $\mu = \lambda_{\min}^{+}(\mathbf{A}^{1/2}\mathbf{P}\mathbf{A}^{1/2})/\operatorname{tr}(\mathbf{A}^{\circ})$. By linearity of expectation, $\nu = \lambda_{\max}\left(\mathbb{E}[\mathbf{Z}]^{\dagger/2}\mathbb{E}\left[\mathbf{Z}\mathbb{E}[\mathbf{Z}]^{\dagger}\mathbf{Z}\right]\mathbb{E}[\mathbf{Z}]^{\dagger/2}\right)$, so it suffices to compute $\mathbb{E}\left[\mathbf{Z}\mathbb{E}[\mathbf{Z}]^{\dagger}\mathbf{Z}\right]$ first. Since $\mathbb{E}[\mathbf{Z}]^{\dagger} = \operatorname{tr}(\mathbf{A}^{\circ})\cdot(\mathbf{P}\mathbf{A}\mathbf{P})^{\dagger}$, and $\mathbf{A}^{1/2}\mathbf{P}(\mathbf{P}\mathbf{A}\mathbf{P})^{\dagger} = (\mathbf{P}\mathbf{A}^{1/2})^{\dagger}$,

$$\begin{aligned}\mathbf{Z}\mathbb{E}[\mathbf{Z}]^{\dagger}\mathbf{Z} &= \operatorname{tr}(\mathbf{A}^{\circ})\cdot\frac{1}{(\mathbf{A}^{\circ}_{j,j})^2}\mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_j\mathbf{e}_j^{\mathsf{T}}\mathbf{A}^{1/2}\mathbf{P}(\mathbf{P}\mathbf{A}\mathbf{P})^{\dagger}\mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_j\mathbf{e}_j^{\mathsf{T}}\mathbf{A}^{1/2}\mathbf{P}\\ &= \operatorname{tr}(\mathbf{A}^{\circ})\cdot\frac{1}{(\mathbf{A}^{\circ}_{j,j})^2}\mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_j\mathbf{e}_j^{\mathsf{T}}(\mathbf{P}\mathbf{A}^{1/2})^{\dagger}\mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_j\mathbf{e}_j^{\mathsf{T}}\mathbf{A}^{1/2}\mathbf{P}.\end{aligned}$$

Observe that $(\mathbf{P}\mathbf{A}^{1/2})^{\dagger}\mathbf{P}\mathbf{A}^{1/2}$ is the orthogonal projector onto $\operatorname{range}(\mathbf{A}^{1/2}\mathbf{P})$. Thus, the diagonal elements of this matrix satisfy $\mathbf{e}_j^{\mathsf{T}}(\mathbf{P}\mathbf{A}^{1/2})^{\dagger}\mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_j \in [0,1]$ for all $j \in [n]$.[6] It follows that

$$\mathbf{Z}\mathbb{E}[\mathbf{Z}]^{\dagger}\mathbf{Z} \preceq \operatorname{tr}(\mathbf{A}^{\circ})\cdot\frac{1}{(\mathbf{A}^{\circ}_{j,j})^2}\mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_j\mathbf{e}_j^{\mathsf{T}}\mathbf{A}^{1/2}\mathbf{P}.$$

[6] Since $\operatorname{range}(\mathbf{A}^{1/2}\mathbf{P}) = \operatorname{range}(\mathbf{A}^{1/2}\mathbf{P}\mathbf{A}^{1/2}) = \operatorname{range}(\mathbf{A}^{\circ})$, $\mathbf{\Pi}_{\mathbf{A}^{\circ}} := (\mathbf{P}\mathbf{A}^{1/2})^{\dagger}\mathbf{P}\mathbf{A}^{1/2}$ is the orthogonal projector onto $\operatorname{range}(\mathbf{A}^{\circ})$, and $\{\mathbf{e}_j^{\mathsf{T}}\mathbf{\Pi}_{\mathbf{A}^{\circ}}\mathbf{e}_j\}_{j\in[n]}$ are the *leverage scores* of each coordinate with respect to $\mathbf{A}^{\circ}$. If we assume that $\mathbf{A}$ is positive definite, then all of the inequalities in this argument are actually *equalities*, and the expression for ν is exact. To see this, note that since $\mathbf{A}^{\circ}$ must also be positive definite, $\operatorname{tr}(\mathbf{\Pi}_{\mathbf{A}^{\circ}}) = \operatorname{rank}(\mathbf{A}^{\circ}) = n - d$. Since $\mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_j = \mathbf{0}$ if and only if $\mathbf{A}^{\circ}_{j,j} = 0$, this implies that we must have $\mathbf{e}_j^{\mathsf{T}}\mathbf{\Pi}_{\mathbf{A}^{\circ}}\mathbf{e}_j = 1$ if $\mathbf{A}^{\circ}_{j,j} > 0$.

By taking expectation, where each $j \in [n]$ is sampled with probability $\mathbf{A}^\circ_{j,j}/\operatorname{tr}(\mathbf{A}^\circ)$, we obtain

$$\mathbb{E}\left[\mathbf{Z}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{Z}\right] \preceq \sum_{j:\mathbf{A}^\circ_{j,j}>0} \frac{1}{\mathbf{A}^\circ_{j,j}}\mathbf{P}\mathbf{A}^{1/2}\mathbf{e}_j\mathbf{e}_j^\top\mathbf{A}^{1/2}\mathbf{P} = \mathbf{P}\mathbf{A}^{1/2}\left(\sum_{j:\mathbf{A}^\circ_{j,j}>0} \frac{1}{\mathbf{A}^\circ_{j,j}}\mathbf{e}_j\mathbf{e}_j^\top\right)\mathbf{A}^{1/2}\mathbf{P}.$$

Note that $\mathbf{D} := \left(\sum_{j:\mathbf{A}^\circ_{j,j}>0} \frac{1}{\mathbf{A}^\circ_{j,j}}\mathbf{e}_j\mathbf{e}_j^\top\right)$ is a diagonal matrix with $\mathbf{D}_{j,j} = (\mathbf{A}^\circ)^\dagger_{j,j}$. Hence,

$$\mathbb{E}[\mathbf{Z}]^{\dagger/2}\mathbb{E}\left[\mathbf{Z}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{Z}\right]\mathbb{E}[\mathbf{Z}]^{\dagger/2} \preceq \mathbb{E}[\mathbf{Z}]^{\dagger/2}\mathbf{P}\mathbf{A}^{1/2}\mathbf{D}\mathbf{A}^{1/2}\mathbf{P}\mathbb{E}[\mathbf{Z}]^{\dagger/2}.$$

Using the fact that $\mathbf{XY}$ and $\mathbf{YX}$ have the same eigenvalues for any (square) matrices $\mathbf{X}, \mathbf{Y}$, the eigenvalues of the matrix in the upper bound displayed above are the same as those of

$$\begin{aligned}\mathbf{D}^{1/2}\mathbf{A}^{1/2}\mathbf{P}\mathbb{E}[\mathbf{Z}]^\dagger\mathbf{P}\mathbf{A}^{1/2}\mathbf{D}^{1/2} &= \operatorname{tr}(\mathbf{A}^\circ)\cdot\mathbf{D}^{1/2}\mathbf{A}^{1/2}\mathbf{P}(\mathbf{P}\mathbf{A}\mathbf{P})^\dagger\mathbf{P}\mathbf{A}^{1/2}\mathbf{D}^{1/2}\\ &= \operatorname{tr}(\mathbf{A}^\circ)\cdot\mathbf{D}^{1/2}(\mathbf{P}\mathbf{A}^{1/2})^\dagger\mathbf{P}\mathbf{A}^{1/2}\mathbf{D}^{1/2} \preceq \operatorname{tr}(\mathbf{A}^\circ)\cdot\mathbf{D},\end{aligned}$$

where we use the fact that $(\mathbf{P}\mathbf{A}^{1/2})^\dagger\mathbf{P}\mathbf{A}^{1/2} \preceq \mathbf{I}$ is an orthogonal projector again. Hence,

$$\nu \le \lambda_{\max}\left(\mathbb{E}[\mathbf{Z}]^{\dagger/2}\mathbf{P}\mathbf{A}^{1/2}\mathbf{D}\mathbf{A}^{1/2}\mathbf{P}\mathbb{E}[\mathbf{Z}]^{\dagger/2}\right) \le \operatorname{tr}(\mathbf{A}^\circ)\cdot\lambda_{\max}(\mathbf{D}) = \frac{\operatorname{tr}(\mathbf{A}^\circ)}{\min_{j:\mathbf{A}^\circ_{j,j}>0}\mathbf{A}^\circ_{j,j}}. \qquad \square$$

3.10.2 Convergence rate of accelerated SC-RCD with $\ell = 1$

By applying Theorem 3.9.1, we can immediately deduce the following bound on the convergence of accelerated subspace-constrained randomized coordinate descent (3.80) with block size $\ell = 1$. It shows that if acceleration parameters $\widehat{\rho}, \widehat{\eta} \in [0, 1]$ based on good estimates of the parameters μ and ν from Lemma 3.10.1 are used, then the accelerated method leads to an improved convergence rate over SC-RCD (Theorem 3.3.2). Specifically, a factor of $\sqrt{\lambda^+_{\min}(\mathbf{A}^\circ)}$ in the convergence rate is replaced with $\sqrt{\min_{j:\mathbf{A}^\circ_{j,j}>0}\mathbf{A}^\circ_{j,j}}$, which is no smaller.

Theorem 3.10.2. *Let $\mathbf{A} \in \mathbb{R}^{n\times n}$ be a positive semidefinite matrix and $\mathbf{x}^*$ be any solution of $\mathbf{A}\mathbf{x} = \mathbf{b}$. Given a fixed subset $\mathcal{S} \subseteq [n]$, let $\mathbf{A}^\circ = \mathbf{A} - \mathbf{A}\langle\mathcal{S}\rangle$. Suppose that we are given*

estimates

$$\widehat{\mu} \leq \frac{\lambda_{\min}^{+}(\mathbf{A}^{\circ})}{\operatorname{tr}(\mathbf{A}^{\circ})} \quad \text{and} \quad \widehat{\nu} \geq \frac{\operatorname{tr}(\mathbf{A}^{\circ})}{\min_{j:\mathbf{A}^{\circ}_{j,j}>0} \mathbf{A}^{\circ}_{j,j}}.$$

that satisfy $\widehat{\nu} \leq 1/\widehat{\mu}$. *Let* $\{\mathbf{x}^k\}_{k\geq 0}$ *be the iterates defined by* (3.80) *where the block* $\mathcal{J} = \{j\}$ *is sampled with probability* $\mathbf{A}^{\circ}_{j,j}/\operatorname{tr}(\mathbf{A}^{\circ})$ *in each iteration, and the acceleration parameters* $\widehat{\rho} = \sqrt{\widehat{\mu}/\widehat{\nu}}$ *and* $\widehat{\eta} = (\widehat{\nu}^{-1} - \widehat{\rho})/(1 - \widehat{\rho})$ *are used. Then,*

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{A}}^2 \leq 8\,(1-\widehat{\rho})^k \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_{\mathbf{A}}^2.$$

In particular, if the exact parameters $\rho = \sqrt{\mu/\nu}$ *and* $\eta = (\nu^{-1} - \rho)/(1-\rho)$ *are used, then*

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{A}}^2 \leq 8\left(1 - \frac{\sqrt{\lambda_{\min}^{+}(\mathbf{A}^{\circ})\min_{j:\mathbf{A}^{\circ}_{j,j}>0}\mathbf{A}^{\circ}_{j,j}}}{\operatorname{tr}(\mathbf{A}^{\circ})}\right)^k \cdot \|\mathbf{x}^0 - \mathbf{x}^*\|_{\mathbf{A}}^2.$$

Proof. If $\mathbf{A}$ is positive definite, then this immediately follows from applying the convergence rate bound for the accelerated subspace-constrained sketch-and-project method in Theorem 3.9.1 for randomized coordinate descent specifically, combined with the discussion of the relationship between $\widehat{\rho}, \widehat{\eta}$ and $\widehat{\mu}, \widehat{\nu}$ in Remark 3.9.2). If $\mathbf{A}$ is only positive semidefinite but not invertible, the same argument can essentially be used to draw the same conclusions, as discussed in the proof of Theorem 3.3.2, and we will not elaborate further. □

Theorem 3.10.2 suggests that the acceleration effect depends on having a good estimate of $\lambda_{\min}^{+}(\mathbf{A}^{\circ})$, which is not feasible to compute. In practice, a pragmatic possibility is to use an adaptive scheme to estimate the acceleration parameters $\widehat{\rho}, \widehat{\eta}$ (e.g., as proposed by [Der+25b]), but we do not investigate this further.

Another interesting future direction is to extend Theorem 3.10.2 by proving a bound on the convergence rate of accelerated subspace-constrained randomized coordinate descent (3.80) with block size $\ell > 1$. Intuitively, we would expect that using a larger block size should improve the convergence rate approximately linearly, relative to using block size one.

We were able to formalize this (in terms of the iteration complexity) for the parameter μ in Proposition 3.2.9. However, it remains an open problem to compute or obtain reasonably tight estimates for the parameter ν with $\ell > 1$ in terms of quantities related to $\mathbf{A}^{\circ}$.

Chapter 4

Dynamics of mini-batch gradient descent with random reshuffling

This chapter is based on the following joint work with Rishi Sonthalia and Elizaveta Rebrova:

J. Lok, R. Sonthalia, and E. Rebrova. "Error dynamics of mini-batch gradient descent with random reshuffling for least squares regression". *Proceedings of the 36th International Conference on Algorithmic Learning Theory*. 2025. arXiv: 2406.03696 [stat.ML]

4.1 Introduction

Modern machine learning models are primarily trained via gradient-based methods on large datasets. Since it is typically not feasible to compute the entire gradient, stochastic gradient descent (SGD) and its variants are often the algorithm of choice [Bot12]. In a variant of SGD known as *mini-batch gradient descent*, a subset of the training data, or *mini-batch*, is used in each iteration. Studying the dynamics of gradient descent is an important problem for understanding the training dynamics and generalization capabilities of the learned parame-

ters, especially for overparameterized models [Gun+18b; Gun+18a]. However, the effects of mini-batching on the error dynamics are less well-understood.

There are also different ways to sample the mini-batch in each iteration. The most commonly studied method is sampling with replacement, where a random subset of data is used to select the mini-batch in each iteration. Thus, in each epoch, the same data point may be used more than once. However, in practice, *random reshuffling* is typically used: at the beginning of each epoch, the dataset is partitioned into mini-batches, randomly permuted, and iterated through. It has been observed that sampling without replacement in this way often leads to faster convergence [Bot09; Bot12]. However, the introduction of dependencies between batches makes theoretical analysis of the dynamics more difficult [GOP21; HS19].

In this work, we aim to contribute towards a better understanding of sampling without replacement. By analyzing the discrete dynamics of mini-batch gradient descent with random reshuffling for the fundamental problem of least squares regression, we find that there are higher-order effects introduced by sampling without replacement that are not present when sampling with replacement, which result in subtly different trajectories.

Contributions. Our main contributions are the following:

- **Exact characterization of error dynamics.** We show that the training dynamics (Theorem 4.3.3) and generalization error (Theorem 4.3.9) of the mean iterate of mini-batch gradient descent with random reshuffling, averaged over the permutations of mini-batches in each epoch, are governed by a sample cross-covariance matrix $\mathbf{Z} := \frac{1}{n}\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}$ that captures the interaction between the original features $\mathbf{X}$ and a set of modified features $\widetilde{\mathbf{X}}$ (defined in Section 4.3). The matrix $\mathbf{Z}$ encapsulates the influence of preceding mini-batches on each feature in an averaged manner, providing a framework for analyzing the learning process. Our results are stated under minimal assumptions on the data, learning rate, and mini-batches.

- **Comparison with full-batch gradient descent/sampling with replacement.** Our analysis demonstrates that the error dynamics of mini-batch gradient descent with random reshuffling are controlled by the sample cross-covariance matrix $\mathbf{Z}$ in a way that is analogous to how full-batch gradient descent (or SGD when sampling with replacement) depends on the sample covariance matrix $\mathbf{W} := \frac{1}{n}\mathbf{X}^\mathsf{T}\mathbf{X}$. We find that $\mathbf{Z}$, which is a non-commutative polynomial in the sample covariance matrices of each mini-batch, matches $\mathbf{W}$ up to leading order with respect to the step size α. Based on this connection, we establish that the linear scaling rule, which calls for the step size to be scaled proportionally by the number of batches, matches the error dynamics of full-batch and mini-batch gradient descent for infinitesimal step sizes (Remark 4.3.6). However, for finite step sizes, mini-batch gradient descent with random reshuffling exhibits a subtle dependence on the step size that a continuous-time gradient flow analysis cannot detect; for example, it may converge to a step size-dependent limit that differs from the usual shifted minimum-norm solution obtained with full-batch gradient descent or SGD when sampling with replacement (Corollary 4.3.5).

- **Effects of batching.** We analyze the effects of batching on the error dynamics compared to full-batch gradient descent by comparing the asymptotic properties of the matrices $\mathbf{Z}$ and $\mathbf{W}$. As the number of data samples n tends to infinity and the dimension of the parameters p is fixed, we establish that asymptotically, while $\mathbf{Z}$ and $\mathbf{W}$ share the same eigenvectors, the eigenvalues of $\mathbf{Z}$ are systematically shrunk compared to those of $\mathbf{W}$ (Proposition 4.3.13), which directly affects the training and generalization errors (Proposition 4.3.14). Furthermore, we demonstrate that batching results in a similar effect in the more complicated proportional regime where $p/n \to \gamma \in (0, \infty)$ by numerically computing the limiting spectrum of $\mathbf{Z}$ under a more specific Gaussian random matrix model using tools from free probability theory (Section 4.3.3).

4.1.1 Related works

Gradient flow. The error dynamics of gradient descent has typically been analyzed from the perspective of continuous-time gradient flow, which is a good approximation assuming that infinitesimal learning rates are used. This perspective is adopted in [SGB94; ASS20; AKT19; ADT20] to study the effects of early stopping and implicit regularization via connections with ridge regression. One of the key themes in these works is that the trajectory and generalization error of (full-batch, continuous-time) gradient descent for least squares regression are determined by the spectrum of the sample covariance matrix $\mathbf{W} = \frac{1}{n}\mathbf{X}^\mathsf{T}\mathbf{X}$ of the data. In our work, we consider the discrete dynamics of gradient descent with finite learning rates, and show that the error dynamics of mini-batch gradient descent with random reshuffling depends analogously on a cross-covariance matrix $\mathbf{Z} = \frac{1}{n}\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}$ that involves a set of modified features $\widetilde{\mathbf{X}}$.

SGD: Sampling with replacement. Stochastic gradient descent has most commonly been studied assuming that the mini-batches are independently sampled with replacement in each iteration, which makes the process more amenable to theoretical analysis. From an optimization perspective, the convergence rates of SGD have been well-studied under various assumptions on the objective function and with different sampling schemes [BM11; BM13; NSW16; MBB18; Gow+19b]. Explanations of the good generalization properties of SGD, based on properties such as the width of the final minima obtained or the optimal batch size and learning rate, have also been offered based on analogies with stochastic differential equations [MHB17; SL18; Jas+18; LTE17; LTE19; LMA21; Mal+22]. These works assume that the mini-batches are sampled independently with replacement in each epoch, and also typically assume that vanishing learning rates are used.

SGD: Sampling without replacement. A line of work that analyzes the implicit bias of SGD with finite learning rates uses the technique of backward error analysis, beginning with

the analysis of gradient descent in [BD21; Miy22], and extended to analyze SGD with random reshuffling—which is the same model that we consider—in [Smi+21]. Specifically, it is shown that the mean iterate, averaged over the permutations of mini-batches in each epoch, is close to the path of gradient flow on a modified loss with an additional regularization term that penalizes the norms of the mini-batch gradients. The mean evolution of SGD using sampling without replacement is also studied in [Ben23] under weaker assumptions. Backward error analysis has also been used to show that adaptive algorithms such as Adam and RMSProp have similar implicit regularization in [CKS24]. Compared to these works, we consider linear models specifically instead of general loss functions; however, our results are presented with minimal assumptions and essentially apply to any input data, choice of mini-batches, batch size, and step size.

Another notable line of work from the stochastic optimization literature studies the convergence rates of SGD when sampling without replacement. In one of the earliest theoretical results, [GOP21] shows that for quadratic objective functions (or more generally, strongly convex smooth objectives), SGD with random reshuffling, using a prescribed sequence of step sizes, converges asymptotically at a rate of $O(1/k^2)$ where k is the number of epochs, which is superior to the $O(1/k)$ rate of SGD when sampling with replacement. In subsequent works [Sha16; HS19; NJN19; RGP20; Ngu+21; MKR20], the complexity advantage of random reshuffling over sampling with replacement (with a primary focus on mini-batches of size one) is analyzed for more general optimization problems where assumptions such as strong convexity, smoothness, and bounded gradients are relaxed. In our work, we study random reshuffling from a different perspective for the special case of least squares regression by providing an exact description of the error dynamics for various batch sizes in terms of the spectrum of the data covariance matrix instead of complexity bounds. As such, our results are not directly comparable with these prior results.

Linear scaling rule. An important aspect of SGD is the choice of batch size and the learning rate. It has been observed that training with larger mini-batches can be more efficient [Smi+18; Gei+22] and lead to better generalization properties [LWM19; Lew+20]. A connection between the batch size and the learning rate for SGD known as the *linear scaling rule* states that by adjusting the mini-batch size and learning rate proportionally by the same factor, the training dynamics do not change. This was empirically discovered to be a practically useful heuristic for training deep neural networks using SGD [Kri14; Goy+18; Smi+18; HLT19], and theoretical explanations have been proposed based on the effect of noise on the estimation of the gradient in each mini-batch for SGD.[1] For more general convex losses, it is also shown in [MBB18] that SGD with mini-batch sizes below a certain threshold is consistent with the linear scaling rule. In our work, we use a different approach to find that the linear scaling rule emerges naturally from analyzing the dynamics of the mean SGD iterate for linear models under no assumptions on the batch size or the noise from mini-batch gradient estimation. Our approach also shows that the linear scaling rule can fail to hold (dramatically) for large step sizes; see Remark 4.3.7.

Linear models. Linear models in the high-dimensional regime have recently been extensively studied in the high-dimensional statistics literature, offering explanations for many interesting empirical phenomena in deep learning, such as double descent and the benefits of overparameterization. It has been shown that neural networks in a "lazy" training regime in which the weights do not change much around initialization are essentially equivalent to linear models [Chi19; Du+19b; Du+19a; MM23]. The generalization errors of ridge(less) regression with random data are precisely described in [DW18; Has+22; MM22; KSS24]. From a dynamical perspective, [Paq+21; Lee+22; Paq+22] show that the trajectories of SGD for ridge regression with finite step sizes and high-dimensional random data concentrate on a deterministic function determined by a Volterra equation, assuming the batch

[1] Interestingly, different optimizers may have different scaling rules: a square root scaling rule has been derived for adaptive gradient algorithms such as Adam and RMSProp using random matrix theory [GZR22] and SDE approximation [Mal+22].

sizes are vanishingly small as a fraction of the sample size. Analogous concentration results for the trajectories of SGD for a wider class of models such as two-layer neural networks have also been derived concurrently [SS95; Gol+19; BGJ22; Arn+23]. Our work provides exact formulas for the training and generalization errors for linear models trained by mini-batch gradient descent with random reshuffling, establishing an analogy with the more well-studied dynamics of full-batch gradient descent or SGD with replacement.

4.2 Problem setup

Suppose that we are given n i.i.d. data samples $(\mathbf{x}_i, y_i)$, where $\mathbf{x}_i \in \mathbb{R}^p$ is the feature vector and $y_i \in \mathbb{R}$ is the response given by $y_i = \mathbf{x}_i^\mathsf{T}\boldsymbol{\beta}_* + \eta_i$, with $\boldsymbol{\beta}_* \in \mathbb{R}^p$ an underlying parameter vector and η_i a noise term. We will assume that the (uncentered) covariance matrix of the features $\mathbf{x}_i$ is given by $\mathbb{E}\left[\mathbf{x}_i\mathbf{x}_i^\mathsf{T}\right] = \Sigma$, and the noise terms η_i have mean $\mathbb{E}\left[\eta_i \mid \mathbf{x}_i\right] = 0$ and variance $\mathbb{E}\left[\eta_i^2 \mid \mathbf{x}_i\right] = \sigma^2$, conditional on the features. By arranging each observation as a row, we can write the linear model in matrix form as $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}_* + \boldsymbol{\eta}$, where $\mathbf{y} \in \mathbb{R}^n$ and $\mathbf{X} \in \mathbb{R}^{n\times p}$.

We consider the following model of *mini-batch gradient descent with random reshuffling* using $B \geq 1$ mini-batches, initialized at $\boldsymbol{\beta}_0 \in \mathbb{R}^p$. For simplicity, we will assume that B divides n. Suppose that the data $\mathbf{X}$ is partitioned into B equally-sized mini-batches $\mathbf{X}_1, \ldots, \mathbf{X}_B \in \mathbb{R}^{(n/B)\times p}$, and let $\mathbf{y}_1, \ldots, \mathbf{y}_B$ and $\boldsymbol{\eta}_1, \ldots, \boldsymbol{\eta}_B$ denote the corresponding entries of $\mathbf{y}$ and $\boldsymbol{\eta}$. In each epoch, a permutation $\tau = (\tau(1), \tau(2), \ldots, \tau(B))$ of the B mini-batches is chosen uniformly at random, and B iterations of gradient descent with step size α are performed with respect to the loss functions

$$L_b(\boldsymbol{\beta}) := \frac{B}{2n}\|\mathbf{y}_b - \mathbf{X}_b\boldsymbol{\beta}\|_2^2 \tag{4.1}$$

for $b = \tau(1), \ldots, \tau(B)$ using this ordering. That is, if $\boldsymbol{\beta}_k^{(b)}$ denotes the parameters after the first b iterations using the mini-batches $\mathbf{X}_{\tau(1)}, \ldots, \mathbf{X}_{\tau(b)}$ in the k^{th} epoch, then

$$\boldsymbol{\beta}_k^{(b)} = \boldsymbol{\beta}_k^{(b-1)} - \frac{B\alpha}{n}\mathbf{X}_{\tau(b)}^{\mathsf{T}}(\mathbf{X}_{\tau(b)}\boldsymbol{\beta}_k^{(b-1)} - \mathbf{y}_{\tau(b)}), \quad b = 1, 2, \ldots, B, \tag{4.2}$$

with $\boldsymbol{\beta}_k^{(0)} := \boldsymbol{\beta}_{k-1}^{(B)}$ and $\boldsymbol{\beta}_0^{(B)} := \boldsymbol{\beta}_0$. Denote the set of all permutations of B elements by S_B. Let

$$\bar{\boldsymbol{\beta}}_k := \mathbb{E}_{\tau \sim \text{Unif}(S_B)}\left[\boldsymbol{\beta}_k^{(B)}\right] \tag{4.3}$$

be the *mean iterate* after k epochs, averaged over the random permutations of the mini-batches in each epoch. Note that *full-batch gradient descent* corresponds to $B = 1$ with this setup.

Our goal is to study the dynamics of the error vector $\bar{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_*$ (i.e., the *training dynamics*), as well as the corresponding *generalization error* $R_{\mathbf{X}}(\bar{\boldsymbol{\beta}}_k)$, representing the prediction error on an out-of-sample observation, defined by

$$R_{\mathbf{X}}(\boldsymbol{\beta}) := \mathbb{E}_{\mathbf{x},\boldsymbol{\eta}}\left[(\mathbf{x}^{\mathsf{T}}\boldsymbol{\beta} - \mathbf{x}^{\mathsf{T}}\boldsymbol{\beta}_*)^2 \mid \mathbf{X}\right] = \mathbb{E}_{\boldsymbol{\eta}}\left[\|\boldsymbol{\beta} - \boldsymbol{\beta}_*\|_{\Sigma}^2 \mid \mathbf{X}\right]. \tag{4.4}$$

Here, the expectation, conditional on the data $\mathbf{X}$, is taken over a newly sampled feature vector $\mathbf{x}$ and the randomness in $\boldsymbol{\eta}$, and $\|\mathbf{z}\|_{\Sigma}^2 = \mathbf{z}^{\mathsf{T}}\Sigma\mathbf{z}$ denotes the norm induced by Σ.

Outline. The rest of the chapter is structured as follows. Section 4.3 describes our main results on analyzing mini-batch gradient descent. After defining the modified features $\widetilde{\mathbf{X}}$ and the matrix $\mathbf{Z}$, we provide exact formulae for the training dynamics and generalization error in Sections 4.3.1 and 4.3.2, respectively. In Section 4.3.3, we consider the asymptotic properties of $\mathbf{Z}$ to evaluate these expressions and provide more insights into the effects of batching. We will defer most of the proofs and technical details to the end of the chapter.

4.3 Analysis of mini-batch gradient descent with random reshuffling

In this section, we will show that the error dynamics of mini-batch gradient descent with random reshuffling are governed by a set of features that are modified by the other mini-batches. Specifically, for $b = 1, \dots, B$, let $\mathbf{W}_b := \frac{B}{n}\mathbf{X}_b^\mathsf{T}\mathbf{X}_b$ denote the sample covariance matrix of each mini-batch, and define the *modified mini-batches* $\widetilde{\mathbf{X}}_b := \mathbf{X}_b\Pi_b$, where[2]

$$\Pi_b := \mathbb{E}_{\tau\sim\mathrm{Unif}(S_B)}\left[\prod_{j:j<\tau^{-1}(b)}(\mathbf{I}-\alpha\mathbf{W}_{\tau(j)})\right] = \frac{1}{B!}\sum_{\tau\in S_B}\prod_{j:j<\tau^{-1}(b)}(\mathbf{I}-\alpha\mathbf{W}_{\tau(j)}). \tag{4.5}$$

That is, each feature $\mathbf{x}_i$ in $\mathbf{X}_b$ corresponds to the feature $\Pi_b\mathbf{x}_i$ in $\widetilde{\mathbf{X}}_b$, which has been modified by all the other mini-batches that appear before it in the learning process in an averaged way. Let $\widetilde{\mathbf{X}} \in \mathbb{R}^{n\times p}$ be the concatenation of the modified mini-batches $\widetilde{\mathbf{X}}_b$ in the same order as the original partition, and define

$$\mathbf{Z} := \frac{1}{n}\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X} = \frac{1}{n}\sum_{b=1}^{B}\Pi_b\mathbf{X}_b^\mathsf{T}\mathbf{X}_b \tag{4.6}$$

to be the $p \times p$ *sample cross-covariance matrix* of the modified features with the original features. The following technical lemma describes some key properties of $\mathbf{Z}$; its proof, which uses the properties of the symmetric group S_B in the definition of $\widetilde{\mathbf{X}}_b$, can be found in Section 4.6.1.

Lemma 4.3.1. *Let* $\widetilde{\mathbf{X}}$ *and* $\mathbf{Z}$ *be defined as in* (4.5) *and* (4.6). *Then* $\mathbf{Z}$ *is a symmetric matrix, and hence all of its eigenvalues are real. Furthermore,* $\mathrm{Range}(\mathbf{Z}) \subseteq \mathrm{Range}(\widetilde{\mathbf{X}}^\mathsf{T}) \subseteq \mathrm{Range}(\mathbf{X}^\mathsf{T})$, *where* range$\cdot$ *denotes the column space of a matrix.*

[2]By convention, we identify each permutation τ in S_B, the set of all permutations of B elements, with a list $(\tau(1), \tau(2), \dots, \tau(B))$ of matrices that are multiplied from right to left in the product. Thus, $\tau^{-1}(b)$ denotes the position of mini-batch b in the epoch. Furthermore, we take the product over an empty set to be the identity matrix.

Finally, observe that $\widetilde{\mathbf{X}} \equiv \widetilde{\mathbf{X}}(\alpha)$ and $\mathbf{Z} \equiv \mathbf{Z}(\alpha)$ are functions of the step size α. In particular, it follows from the definition of the modified features $\widetilde{\mathbf{X}}_b = \mathbf{X}_b\Pi_b$ in (4.3.1) that we can write

$$\mathbf{Z}(\alpha) = \frac{1}{n}\sum_{b=1}^{B} \mathbf{X}_b^\mathsf{T}\mathbf{X}_b + O(\alpha) = \mathbf{W} + O(\alpha), \tag{4.7}$$

where $\mathbf{W} := \frac{1}{n}\mathbf{X}^\mathsf{T}\mathbf{X} = \frac{1}{n}\sum_{b=1}^{B}\mathbf{X}_b^\mathsf{T}\mathbf{X}_b$, and $O(\alpha)$ denotes terms of order α or smaller as $\alpha \to 0$. This shows that $\mathbf{Z}$ matches $\mathbf{W}$, the sample covariance matrix of the features, up to leading order in the step size α. In general, $\mathbf{Z}$ *is a complicated non-commutative polynomial of the mini-batch sample covariance matrices* $\mathbf{W}_1, \ldots, \mathbf{W}_B$.

Example 4.3.2 (Two-batch gradient descent)**.** For a concrete example where we can write down a tractable, explicit expression for $\mathbf{Z}$, consider the case of *two-batch gradient descent* with $B = 2$ and mini-batches $\mathbf{X}_1, \mathbf{X}_2 \in \mathbb{R}^{(n/2)\times p}$. Here, the sample covariance matrices of the mini-batches are $\mathbf{W}_1 = \frac{2}{n}\mathbf{X}_1^\mathsf{T}\mathbf{X}_1$ and $\mathbf{W}_2 = \frac{2}{n}\mathbf{X}_2^\mathsf{T}\mathbf{X}_2$, and the modified mini-batches are given by

$$\widetilde{\mathbf{X}}_1 \equiv \widetilde{\mathbf{X}}_1(\alpha) = \mathbf{X}_1\left(\mathbf{I} - \frac{1}{2}\alpha\mathbf{W}_2\right) \quad \text{and} \quad \widetilde{\mathbf{X}}_2 \equiv \widetilde{\mathbf{X}}_2(\alpha) = \mathbf{X}_2\left(\mathbf{I} - \frac{1}{2}\alpha\mathbf{W}_1\right). \tag{4.8}$$

Thus, the features in $\widetilde{\mathbf{X}}_1$, corresponding to the first mini-batch, are given by $\left(\mathbf{I} - \frac{1}{2}\alpha\mathbf{W}_2\right)\mathbf{x}_i$. The sample cross-covariance matrix of the modified features $\widetilde{\mathbf{X}}$ with the original features is given by

$$\begin{aligned}\mathbf{Z} \equiv \mathbf{Z}(\alpha) &= \frac{1}{n}(\widetilde{\mathbf{X}}_1(\alpha)^\mathsf{T}\mathbf{X}_1 + \widetilde{\mathbf{X}}_2(\alpha)^\mathsf{T}\mathbf{X}_2) = \frac{1}{2}\left(\mathbf{I} - \frac{1}{2}\alpha\mathbf{W}_2\right)\mathbf{W}_1 + \frac{1}{2}\left(\mathbf{I} - \frac{1}{2}\alpha\mathbf{W}_1\right)\mathbf{W}_2 \\ &= \frac{1}{2}\left(\mathbf{W}_1 + \mathbf{W}_2\right) - \frac{1}{4}\alpha\left(\mathbf{W}_2\mathbf{W}_1 + \mathbf{W}_1\mathbf{W}_2\right).\end{aligned} \tag{4.9}$$

Since $\frac{1}{2}(\mathbf{W}_1 + \mathbf{W}_2) = \frac{1}{n}(\mathbf{X}_1^\mathsf{T}\mathbf{X}_1 + \mathbf{X}_2^\mathsf{T}\mathbf{X}_2) = \mathbf{W}$, it is easily seen that $\mathbf{Z} = \mathbf{W} + O(\alpha)$. Even in this simple setting, $\mathbf{Z}$ is already non-trivial to analyze since it involves interactions between the two mini-batches in the term $\mathbf{W}_2\mathbf{W}_1 + \mathbf{W}_1\mathbf{W}_2$, known as the *anticommutator* of $\mathbf{W}_1$ and $\mathbf{W}_2$.

4.3.1 Training error dynamics

First, we derive an expression for the dynamics of the mean error $\bar{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_*$ under mini-batch gradient descent with random reshuffling. The expression depends on *the spectrum of the sample cross-covariance matrix* $\mathbf{Z}$ and *the alignment of the initial error* $\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*$ *with the eigenspaces of* $\mathbf{Z}$.

Theorem 4.3.3. *Let* $\bar{\boldsymbol{\beta}}_k \in \mathbb{R}^p$ *be the mean iterate after* k *epochs of gradient descent with* B *mini-batches, step size* $\alpha \geq 0$*, and initialization* $\boldsymbol{\beta}_0 \in \mathbb{R}^p$*. Let* $\widetilde{\mathbf{X}} \in \mathbb{R}^{n \times p}$ *be defined as in* (4.5) *and* $\mathbf{Z} = \frac{1}{n}\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}$*, and assume that* $\mathrm{range}\widetilde{\mathbf{X}}^\mathsf{T} \subseteq \mathrm{range}\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}$*. Then for all* $k \geq 0$*,*

$$\bar{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_* = (\mathbf{I} - B\alpha\mathbf{Z})^k(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) + \frac{1}{n}\left[\mathbf{I} - (\mathbf{I} - B\alpha\mathbf{Z})^k\right]\mathbf{Z}^\dagger\widetilde{\mathbf{X}}^\mathsf{T}\boldsymbol{\eta}. \tag{4.10}$$

Furthermore, if $\mathbf{P}_{\mathbf{Z},0} := \mathbf{I} - \mathbf{Z}^\dagger\mathbf{Z}$ *and* $\mathbf{P}_\mathbf{Z} := \mathbf{I} - \mathbf{P}_{\mathbf{Z},0}$ *denote the orthogonal projectors onto the nullspace and row (or column) space of* $\mathbf{Z}$*, respectively (where* $(\cdot)^\dagger$ *is the Moore–Penrose pseudoinverse of a matrix), then we may decompose the first term as*

$$(\mathbf{I} - B\alpha\mathbf{Z})^k(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) = \mathbf{P}_{\mathbf{Z},0}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) + (\mathbf{I} - B\alpha\mathbf{Z})^k\mathbf{P}_\mathbf{Z}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*). \tag{4.11}$$

The proof of Theorem 4.3.3 is given in Section 4.6.1; the main technical part involves developing some algebraic identities relating $\mathbf{Z}$ and products of the form $\mathbf{I} - \alpha\mathbf{W}_b$ for each mini-batch. The requirement $\mathrm{range}\widetilde{\mathbf{X}}^\mathsf{T} \subseteq \mathrm{range}\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}$ is purely a technical assumption to ensure that $\mathbf{P}_\mathbf{Z}\widetilde{\mathbf{X}}^\mathsf{T} = \widetilde{\mathbf{X}}^\mathsf{T}$ in order to control the learned noise, otherwise the iterate will always diverge.[3] The requirement appears to be generic; for example, in the overparameterized regime where $p \geq n$, it simply follows from the natural assumption that $\mathbf{X}$ has full rank.

The first term $\mathbf{P}_{\mathbf{Z},0}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)$ of (4.11) corresponds to the components of $\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*$ that cannot be learned by mini-batch gradient descent with random reshuffling—referred to as a *"frozen subspace"* of weights in [ASS20] in the context of (full-batch) gradient descent—and

[3] For full-batch gradient descent, the corresponding requirement is $\mathrm{range}\mathbf{X}^\mathsf{T} \subseteq \mathrm{range}\mathbf{X}^\mathsf{T}\mathbf{X}$, which always holds.

the second term $\mathbf{P}_{\mathbf{Z}}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)$ corresponds to the *learnable* components. In particular, note that the projector $\mathbf{P}_{\mathbf{Z},0}$ is always non-trivial in the overparameterized regime where $p > n$.

Comparison with full-batch and mini-batching with replacement

First, we recall the known result that the full-batch gradient descent iterate $\widehat{\boldsymbol{\beta}}_k$ satisfies the following (for a proof and additional background, we refer to Section 4.5):

$$\widehat{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_* = (\mathbf{I} - \alpha\mathbf{W})^k(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) + \frac{1}{n}\left[\mathbf{I} - (\mathbf{I} - \alpha\mathbf{W})^k\right]\mathbf{W}^\dagger\mathbf{X}^\mathsf{T}\boldsymbol{\eta}. \tag{4.12}$$

Remark 4.3.4 (Sampling with replacement)**.** Suppose that in each iteration, we sample a mini-batch *with replacement* uniformly at random from the fixed set of B mini-batches $\mathbf{X}_1, \ldots, \mathbf{X}_B$ instead. Then it can be shown that after k epochs (or Bk iterations), the error corresponding to the mean iterate of this sampling process also satisfies (4.12) up to a time change by a factor of B; i.e., the same equation holds with k replaced by Bk. For the details, see Section 4.6.1.

Therefore, comparing (4.12) with Theorem 4.3.3, we see that the sample cross-covariance matrix $\mathbf{Z}$ plays an analogous role as the sample covariance matrix $\mathbf{W}$ in the training dynamics of full-batch gradient descent or mini-batch gradient descent when sampling with replacement.

Comparing the limiting vectors. Furthermore, recall that the iterates of full-batch gradient descent with step size $\alpha < 2/\|n^{-1}\mathbf{X}^\mathsf{T}\mathbf{X}\|$ tend to the shifted min-norm solution

$$\widehat{\boldsymbol{\beta}}_\infty := \mathbf{P}_{\mathbf{X},0}\boldsymbol{\beta}_0 + (\mathbf{X}^\mathsf{T}\mathbf{X})^\dagger\mathbf{X}^\mathsf{T}\mathbf{y}, \tag{4.13}$$

where $\mathbf{P}_{\mathbf{X},0} := \mathbf{I} - \mathbf{X}^\dagger\mathbf{X}$ is the orthogonal projector onto null$\mathbf{X}$, and $\|\cdot\|$ denotes the spectral norm of a matrix. From Remark 4.3.4, this is the same limit for mini-batching with replacement. In particular, note that this limit is always independent of the step size α.

On the other hand, as a corollary of Theorem 4.3.3, we see that mini-batch gradient descent with random reshuffling, using a step size small enough so that $\|(\mathbf{I} - B\alpha\mathbf{Z})\mathbf{P}_{\mathbf{Z}}\| < 1$ (i.e., based on the non-zero eigenvalues of $\mathbf{Z}$), converges to a solution $\bar{\boldsymbol{\beta}}_\infty$ that can exhibit *more complex interactions between the mini-batches* and *a dependence on the step size*.

Corollary 4.3.5 (Limit with random reshuffling)**.** *Consider the same setup as Theorem 4.3.3. If* $\mathbf{Z}$ *is positive semidefinite and* $B\alpha < 2/\left\|n^{-1}\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}\right\|$*, then* $\bar{\boldsymbol{\beta}}_k \to \bar{\boldsymbol{\beta}}_\infty$ *as* $k \to \infty$*, where*

$$\bar{\boldsymbol{\beta}}_\infty \equiv \bar{\boldsymbol{\beta}}_\infty(\alpha) := \mathbf{P}_{\mathbf{Z},0}\boldsymbol{\beta}_0 + (\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X})^\dagger\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{y}.$$

We can examine the two limits $\bar{\boldsymbol{\beta}}_\infty$ and $\widehat{\boldsymbol{\beta}}_\infty$ from Corollary 4.3.5 and (4.13) in the over and underparameterized regimes more carefully. For simplicity, we will assume that $\mathbf{X}$ is full rank here to avoid the complexities in the rank deficient case. Recall that $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}_* + \boldsymbol{\eta}$, and since $\mathbb{R}^n = \text{range}\mathbf{X} \oplus \text{null}\mathbf{X}^\mathsf{T}$, we can write $\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\theta} + \boldsymbol{\xi}$ for some $\boldsymbol{\theta} \in \mathbb{R}^p$ and $\boldsymbol{\xi} \in \text{null}\mathbf{X}^\mathsf{T}$.

- In the overparameterized regime ($p \geq n$), we have $\widehat{\boldsymbol{\beta}}_\infty = \mathbf{P}_{\mathbf{X},0}\boldsymbol{\beta}_0 + \mathbf{P}_{\mathbf{X}^\mathsf{T}}\boldsymbol{\beta}_* + \mathbf{P}_{\mathbf{X}^\mathsf{T}}\boldsymbol{\theta}$ and $\bar{\boldsymbol{\beta}}_\infty = \mathbf{P}_{\mathbf{Z},0}\boldsymbol{\beta}_0 + \mathbf{P}_{\mathbf{Z}}\boldsymbol{\beta}_* + \mathbf{P}_{\mathbf{Z}}\boldsymbol{\theta}$, since $\boldsymbol{\xi} = \mathbf{0}$ (here, $\mathbf{P}_{\mathbf{X}^\mathsf{T}}$ is the orthogonal projector onto $\text{range}\mathbf{X}^\mathsf{T}$). Thus, if the ranges of $\mathbf{X}^\mathsf{T}$ and $\mathbf{Z}$ are close, then the two limits are also similar, regardless of the noise vector $\boldsymbol{\eta}$. Specifically, the two subspaces can be shown to coincide if $\text{range}\mathbf{X}^\mathsf{T} \subseteq \text{range}\mathbf{X}^\mathsf{T}\widetilde{\mathbf{X}}$. Therefore, if $\widetilde{\mathbf{X}}$ is also full rank (which is typical), then the two limits are actually the same (in particular, the dependence of $\bar{\boldsymbol{\beta}}_\infty$ on the step size vanishes). However, we emphasize that in this case, the two *trajectories* still differ in a step size-dependent way.

- In the underparameterized regime ($p < n$), we have $\widehat{\boldsymbol{\beta}}_\infty = \boldsymbol{\beta}_* + \boldsymbol{\theta}$, but, assuming that $\widetilde{\mathbf{X}}$ is also full rank (so $\mathbf{Z} = \widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}$ is invertible), $\bar{\boldsymbol{\beta}}_\infty = \boldsymbol{\beta}_* + \boldsymbol{\theta} + (\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X})^{-1}\widetilde{\mathbf{X}}^\mathsf{T}\boldsymbol{\xi}$. Since the nullspaces of $\mathbf{X}^\mathsf{T}$ and $\widetilde{\mathbf{X}}^\mathsf{T}$ are not necessarily close (so $\widetilde{\mathbf{X}}^\mathsf{T}\boldsymbol{\xi} \neq \mathbf{0}$), we find that the two limits easily exhibit a step size-dependent difference in this case with non-zero noise $\boldsymbol{\eta}$.

Comparing the trajectories. Note that from Theorem 4.3.3, the error of mini-batch gradient descent with random reshuffling depends on $B\alpha\mathbf{Z}$. This can be compared with a dependence on $\alpha\mathbf{W}$ in the full-batch case. Since $\mathbf{Z}$ matches $\mathbf{W}$ up to leading order (4.7) in α, this suggests that if a step size of α/B is used for mini-batch gradient descent with B mini-batches, then its dynamics should be very similar to those of full-batch gradient descent with step size α. The following remark establishes this intuition rigorously for infinitesimal step sizes.

Remark 4.3.6 (Linear scaling and gradient flow)**.** From Theorem 4.3.3, initialized at $\bar{\boldsymbol{\beta}}_{k-1}$, and using the fact that $\mathbf{Z}^{\dagger}\mathbf{Z}\widetilde{\mathbf{X}}^{\mathsf{T}} = \widetilde{\mathbf{X}}^{\mathsf{T}}$, the error vector of mini-batch gradient descent with random reshuffling with B mini-batches and step size α/B satisfies

$$\bar{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_* = \left(\mathbf{I} - \frac{\alpha}{n}\widetilde{\mathbf{X}}^{\mathsf{T}}\mathbf{X}\right)(\bar{\boldsymbol{\beta}}_{k-1} - \boldsymbol{\beta}_*) + \frac{\alpha}{n}\widetilde{\mathbf{X}}^{\mathsf{T}}\boldsymbol{\eta}.$$

By rearranging this expression, recalling that $\mathbf{Z} = \mathbf{W} + O(\alpha)$, we obtain

$$\frac{\bar{\boldsymbol{\beta}}_k - \bar{\boldsymbol{\beta}}_{k-1}}{\alpha} = \frac{1}{n}\mathbf{X}^{\mathsf{T}}(\mathbf{y} - \mathbf{X}\bar{\boldsymbol{\beta}}_{k-1}) + O(\alpha).$$

Hence, by taking the limit as $\alpha \to 0$, we deduce that the continuous dynamics correspond to the ordinary differential equation

$$\frac{\mathrm{d}}{\mathrm{d}t}\bar{\boldsymbol{\beta}}(t) = \frac{1}{n}\mathbf{X}^{T}(\mathbf{y} - \mathbf{X}\bar{\boldsymbol{\beta}}(t)). \tag{4.14}$$

This is the same differential equation for the gradient flow corresponding to full-batch gradient descent (e.g., [ASS20; AKT19]). Naturally, this also coincides with the continuous-time dynamics of the model of SGD when sampling with replacement discussed in Remark 4.3.4. *As a consequence, we deduce that a gradient flow analysis cannot distinguish the effects of batching when sampling without replacement.*

Remark 4.3.7 (Large step sizes)**.** While Remark 4.3.6 shows that the dynamics of mini-batch gradient descent with random reshuffling are similar to those of full-batch gradient descent small step sizes using linear scaling, the two dynamics can be dramatically different for large step sizes. For example, if the step size satisfies $\alpha > 2/\|n^{-1}\mathbf{X}^\mathsf{T}\mathbf{X}\|$ but $B\alpha < 2/\left\|n^{-1}\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}\right\|$, then full-batch gradient descent diverges while mini-batch gradient descent still converges. For a simple numerical demonstration of this phenomenon, see Section 4.8.

Next, a natural question is whether an explicit condition, based only on the data $\mathbf{X}$, can be formulated for how small the step size α needs to be for mini-batch gradient descent with random reshuffling to converge as guaranteed by Corollary 4.3.5. We can show the following sufficient condition for two-batch gradient descent: recall from Example 4.3.2 that in this setting, $\mathbf{Z} = \frac{1}{2}(\mathbf{W}_1 + \mathbf{W}_2) - \frac{1}{4}\alpha(\mathbf{W}_2\mathbf{W}_1 + \mathbf{W}_1\mathbf{W}_2)$ is a non-commutative polynomial of the mini-batch covariances $\mathbf{W}_1$, $\mathbf{W}_2$.

Proposition 4.3.8. *If full-batch gradient descent with step size 2α converges (i.e., $\alpha < 1/(n^{-1}\|\mathbf{X}^\mathsf{T}\mathbf{X}\|)$), then two-batch gradient descent with step size α also converges (i.e., $\|(\mathbf{I} - 2\alpha\mathbf{Z})\mathbf{P}_\mathbf{Z}\| < 1$).*

The proof of Proposition 4.3.8, which uses some matrix analysis, is given in Section 4.6.2. To explain why this seemingly-simple statement is non-trivial, observe that it is not even immediately obvious when $\mathbf{Z}$ is positive semidefinite since it may have negative eigenvalues if α is large enough (unlike the covariance matrix $\mathbf{W}$). Note that the converse of Proposition 4.3.8 is not true as discussed previously in Remark 4.3.7. Furthermore, based on the correspondence with full-batch gradient descent using the linear scaling rule, Proposition 4.3.8 suggests that mini-batch gradient descent with random reshuffling has some sort of shrinkage effect on the operator norm of $\mathbf{Z}$ compared to $\mathbf{W}$.

4.3.2 Generalization error dynamics

Next, we provide an exact formula for the generalization error of the mean iterate of mini-batch gradient descent with random reshuffling, which corresponds to the usual bias-variance decomposition, The following result shows that the bias component of the generalization error (i.e., the first two terms) only depends on the sample cross-covariance matrix $\mathbf{Z}$, and the variance component (i.e., the last term) depends on $\mathbf{Z}$ and the modified features through $\widetilde{\mathbf{X}}^\top\widetilde{\mathbf{X}}$.

Theorem 4.3.9. *Consider the same setup as Theorem 4.3.3. Then for all $k \geq 0$, the generalization error (4.4) of the mean mini-batch gradient descent iterate $\bar{\boldsymbol{\beta}}_k$ is given by*

$$\begin{aligned} R_{\mathbf{X}}(\bar{\boldsymbol{\beta}}_k) &= (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)^\top \mathbf{P}_{\mathbf{Z},0} \Sigma \mathbf{P}_{\mathbf{Z},0} (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) \\ &\quad + (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)^\top \mathbf{P}_{\mathbf{Z}} (\mathbf{I} - B\alpha\mathbf{Z})^k \Sigma (\mathbf{I} - B\alpha\mathbf{Z})^k \mathbf{P}_{\mathbf{Z}} (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) \\ &\quad + \frac{\sigma^2}{n} \operatorname{Tr}\left(\left[\mathbf{I} - (\mathbf{I} - B\alpha\mathbf{Z})^k\right] \Sigma \left[\mathbf{I} - (\mathbf{I} - B\alpha\mathbf{Z})^k\right] \mathbf{Z}^\dagger \left(\frac{1}{n} \widetilde{\mathbf{X}}^\top \widetilde{\mathbf{X}} \right) \mathbf{Z}^\dagger \right). \end{aligned}$$

The proof of Theorem 4.3.9, which uses the error dynamics from Theorem 4.3.3, appears in Section 4.6.1. Theorem 4.3.9 shows that the generalization errors of mini-batch and full-batch gradient descent also correspond under the linear scaling rule, which is consistent with Remark 4.3.6 (for numerical experiments demonstrating this, see Section 4.8).

As a straightforward corollary, we can also write down the limiting risk of mini-batch gradient descent with a small enough step size, complementing Corollary 4.3.5, which shows that the limiting risk consists of the constant term corresponding to $\mathbf{P}_{\mathbf{Z},0}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)$, the components of the initial error in the frozen subspace, and a term corresponding to overfitting the noise that is magnified by small eigenvalues of $\mathbf{Z}$.

Corollary 4.3.10. *Consider the same setup as Theorem 4.3.9. If $\|(\mathbf{I} - B\alpha\mathbf{Z})\mathbf{P}_{\mathbf{Z}}\| < 1$, then $\bar{\boldsymbol{\beta}}_k \to \bar{\boldsymbol{\beta}}_\infty = \mathbf{P}_{\mathbf{Z},0}\boldsymbol{\beta}_0 + (\widetilde{\mathbf{X}}^\top\mathbf{X})^\dagger \widetilde{\mathbf{X}}^\top \mathbf{y}$ as $k \to \infty$, and the limiting generalization error is given*

by

$$R_{\mathbf{X}}(\bar{\boldsymbol{\beta}}_\infty) = (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)^\mathsf{T}\mathbf{P}_{\mathbf{Z},0}\Sigma\mathbf{P}_{\mathbf{Z},0}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) + \frac{\sigma^2}{n}\operatorname{Tr}\left(\Sigma\mathbf{Z}^\dagger\Big(\frac{1}{n}\widetilde{\mathbf{X}}^\mathsf{T}\widetilde{\mathbf{X}}\Big)\mathbf{Z}^\dagger\right).$$

Note that the generalization error depends on both $\mathbf{Z}$ and the covariance of the modified features $\widetilde{\mathbf{X}}^\mathsf{T}\widetilde{\mathbf{X}}$, rather than only on $\mathbf{Z}$. If we specialize to the case of two-batch gradient descent again, then we are able to show the following result, which bounds the generalization error within a narrow interval that only depends on $\mathbf{Z}$, under a natural assumption on the step size α that was shown in Proposition 4.3.8 to imply convergence.

Proposition 4.3.11. *Consider the same setup as Theorem 4.3.9 with $B = 2$. If $\alpha \le 1/(n^{-1}\|\mathbf{X}^\mathsf{T}\mathbf{X}\|)$, then for all $k \ge 0$, $R_{\mathbf{X}}(\bar{\boldsymbol{\beta}}_k) \in [R_-(k), R_+(k)]$, where*

$$\begin{aligned} R_\pm(k) &:= (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)^\mathsf{T}\mathbf{P}_{\mathbf{Z},0}\Sigma\mathbf{P}_{\mathbf{Z},0}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) \\ &\quad + (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)^\mathsf{T}\mathbf{P}_{\mathbf{Z}}(\mathbf{I} - 2\alpha\mathbf{Z})^k\Sigma(\mathbf{I} - 2\alpha\mathbf{Z})^k\mathbf{P}_{\mathbf{Z}}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) \\ &\quad + \left(1 \pm \alpha n^{-1}\|\mathbf{X}^\mathsf{T}\mathbf{X}\|\right)\frac{\sigma^2}{n}\operatorname{Tr}\left(\left[\mathbf{I} - (\mathbf{I} - 2\alpha\mathbf{Z})^k\right]\Sigma\left[\mathbf{I} - (\mathbf{I} - 2\alpha\mathbf{Z})^k\right]\mathbf{Z}^\dagger\right). \end{aligned}$$

The upper bound is tight if $\mathbf{W}_1 = \mathbf{W}_2 = c\mathbf{I}$ for some $c > 0$ and $\alpha = 2/c$. Furthermore, $\bar{\boldsymbol{\beta}}_k \to \bar{\boldsymbol{\beta}}_\infty$ as $k \to \infty$, and the limiting generalization error lies in the interval

$$R_{\mathbf{X}}(\bar{\boldsymbol{\beta}}_\infty) \in (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)^\mathsf{T}\mathbf{P}_{\mathbf{Z},0}\Sigma\mathbf{P}_{\mathbf{Z},0}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) + \left(1 \pm \alpha n^{-1}\|\mathbf{X}^\mathsf{T}\mathbf{X}\|\right)\frac{\sigma^2}{n}\operatorname{Tr}\left(\Sigma\mathbf{Z}^\dagger\right).$$

The proof of Proposition 4.3.11, which relies on some matrix analysis, is given in Section 4.6.2. Note that the width of the interval is linear in the step size α, and thus shrinks to zero as $\alpha \to 0$.

Remark 4.3.12. Instead of studying the generalization error of the mean iterate $\bar{\boldsymbol{\beta}}_k = \mathbb{E}_\tau[\boldsymbol{\beta}_k^{(B)}]$, it would also be of interest to understand the expected generalization error of the random iterate $\boldsymbol{\beta}_k^{(B)}$ itself; that is, $\mathbb{E}_{\tau,\mathbf{x},\boldsymbol{\eta}}[(\mathbf{x}^\mathsf{T}\boldsymbol{\beta}_k^{(B)} - \mathbf{x}^\mathsf{T}\boldsymbol{\beta}_*)^2 \mid \mathbf{X}] = \mathbb{E}_{\tau,\boldsymbol{\eta}}[\|\boldsymbol{\beta}_k^{(B)} - \boldsymbol{\beta}_*\|_\Sigma^2 \mid \mathbf{X}]$, where the expectation over the random reshuffling process is taken at the end. By a bias–variance decomposition for the norm of a random vector (e.g., [GR15a, Lemma 4.1]), it can

be shown that $\mathbb{E}_{\tau,\boldsymbol{\eta}}[\|\boldsymbol{\beta}_k^{(B)} - \boldsymbol{\beta}_*\|_\Sigma^2 \mid \mathbf{X}] = R_{\mathbf{X}}(\bar{\boldsymbol{\beta}}_k) + \mathbb{E}_{\tau,\boldsymbol{\eta}}[\|\boldsymbol{\beta}_k^{(B)} - \bar{\boldsymbol{\beta}}_k\|_\Sigma^2 \mid \mathbf{X}]$. Thus, our exact description of $R_{\mathbf{X}}(\bar{\boldsymbol{\beta}}_k)$ provides a lower bound for this notion of expected generalization error. The difference between these two quantities, $\mathbb{E}_{\tau,\boldsymbol{\eta}}[\|\boldsymbol{\beta}_k^{(B)} - \bar{\boldsymbol{\beta}}_k\|_\Sigma^2 \mid \mathbf{X}]$ corresponds to the variance of $\boldsymbol{\beta}_k^{(B)}$ over the random reshuffling process (averaged over the noise).

4.3.3 Asymptotic analysis

In this section, we aim to provide more insights into the effects of batching without replacement by interpreting our main results on the training error (Theorem 4.3.3) and generalization error (Theorem 4.3.9) asymptotically. This will allow us to obtain a finer characterization of the sample cross-covariance matrix $\mathbf{Z}(\alpha/B) = \frac{1}{n}\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}$, which is quite non-trivial to analyze in general as it is a non-commutative polynomial of the mini-batch covariance matrices, and compare it with the sample covariance matrix $\mathbf{W} = \frac{1}{n}\mathbf{X}^\mathsf{T}\mathbf{X}$, its full-batch analogue (under linear scaling).

Asymptotic analysis in the large n, fixed p regime

We begin by considering the more classical statistical regime where p is fixed and $n \to \infty$. Since we assume that the features $\mathbf{x}_i$ are i.i.d. with $\mathbb{E}\left[\mathbf{x}_i\mathbf{x}_i^\mathsf{T}\right] = \Sigma$, by the law of large numbers, the sample covariances $\mathbf{W} = \frac{1}{n}\mathbf{X}^\mathsf{T}\mathbf{X}$ and $\mathbf{W}_b = \frac{B}{n}\mathbf{X}_b^\mathsf{T}\mathbf{X}_b$, $b = 1, \dots B$, tend to Σ as $n \to \infty$, almost surely. Therefore, by independence, $\mathbf{Z}(\alpha/B)$ tends to $\Sigma(\mathbf{I} - p_{B,\alpha}(\Sigma))$, where $p_{B,\alpha}$ is a certain polynomial that depends on the number of mini-batches B and step size α. If we denote the eigenvalues of Σ by λ_i, then the limiting eigenvalues of $\mathbf{Z}(\alpha/B)$ are given by $\lambda_i(1 - p_{B,\alpha}(\lambda_i))$.

For example, if $B = 2$, then $\mathbf{Z}(\alpha/2) = \frac{1}{2}(\mathbf{W}_1 + \mathbf{W}_2) - \frac{1}{8}\alpha(\mathbf{W}_2\mathbf{W}_1 + \mathbf{W}_1\mathbf{W}_2)$ converges to $\Sigma - \frac{1}{4}\alpha\Sigma^2$, so $p_{2,\alpha}(\Sigma) = \frac{1}{4}\alpha\Sigma$. In particular, note that the limiting spectrum of $\mathbf{W}$ is shrunk compared to $\mathbf{Z}$.[4] In general, for any B, we have the following expression for $p_{B,\alpha}$:

[4]For another example, if $B = 3$, then $\mathbf{Z}(\alpha/3) = \frac{1}{3}(\mathbf{W}_1 + \mathbf{W}_2 + \mathbf{W}_3) - \frac{1}{18}\alpha(\mathbf{W}_1\mathbf{W}_2 + \mathbf{W}_1\mathbf{W}_3 + \mathbf{W}_2\mathbf{W}_1 + \mathbf{W}_2\mathbf{W}_3 + \mathbf{W}_3\mathbf{W}_1 + \mathbf{W}_3\mathbf{W}_2) + \frac{1}{162}\alpha^2(\mathbf{W}_1\mathbf{W}_2\mathbf{W}_3 + \mathbf{W}_1\mathbf{W}_3\mathbf{W}_2 + \mathbf{W}_2\mathbf{W}_1\mathbf{W}_3 + \mathbf{W}_2\mathbf{W}_3\mathbf{W}_1 + \mathbf{W}_3\mathbf{W}_1\mathbf{W}_2 + \mathbf{W}_3\mathbf{W}_2\mathbf{W}_1)$, which converges to $\Sigma - \frac{1}{3}\alpha\Sigma^2 + \frac{1}{27}\alpha^2\Sigma^3$, so $p_{3,\alpha}(\Sigma) = \frac{1}{3}\alpha\Sigma - \frac{1}{27}\alpha^2\Sigma^2$.

Proposition 4.3.13. *Suppose that p is fixed. Then as $n \to \infty$, $\mathbf{Z}(\alpha/B) \to \Sigma(\mathbf{I} - p_{B,\alpha}(\Sigma))$ almost surely, where*

$$p_{B,\alpha}(\Sigma) = \sum_{i=1}^{B-1} (-1)^{i+1} \frac{(B-1)!(B-1-i)!}{(i+1)!} \left(\frac{\alpha}{B}\right)^i \Sigma^i.$$

The proof of this result uses the algebraic representation of Π_b as a function of all the other mini-batches to write down the limit of each $\mathbf{W}_b\Pi_b$, which allows for the limit of $\mathbf{Z} = \frac{1}{B}\sum_{b=1}^{B} \mathbf{W}_b\Pi_b$ to be obtained by symmetry. For the details, see Section 4.6.3.

Observe that Proposition 4.3.13 implies that the sample cross-covariance $\mathbf{Z}$ is not a consistent estimator of the true (uncentered) covariance matrix Σ of the features, unlike $\mathbf{W}$. Moreover, we see that although $\mathbf{Z}$ matches $\mathbf{W}$ up to leading order in α, asymptotically, batching results in a step size-dependent shrinkage of the spectrum of $\mathbf{W}$ (for small enough α satisfying, say, $\alpha\|\mathbf{W}\| \leq 1$).

The key idea behind Proposition 4.3.13 is that by exploiting the independence of each mini-batch and the algebraic properties of Π_b, the matrix Π_b that modifies each mini-batch turns out to be asymptotically independent of b as $n \to \infty$; in fact, the limit of each Π_b is exactly the matrix $\mathbf{I} - p_{B,\alpha}(\Sigma)$ from Proposition 4.3.13. We can take this idea and make it an explicit assumption (justified by the fact that it holds asymptotically) to elaborate on the implications of batching by providing an *explicit description* of how the *trajectories* of mini-batch gradient descent with random reshuffling differ from the full-batch case under linear scaling.

Proposition 4.3.14. *Suppose that $\Pi_b = \Pi := \mathbf{I} - p(\mathbf{W})$ for each $b = 1, \dots, B$, where $p \equiv p_\alpha$ is some polynomial, and that $\mathbf{X}$ and Π are invertible. Let $\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^\mathsf{T}$ be a singular value decomposition of $\mathbf{X}$, so that $\mathbf{W} = \mathbf{V}(\frac{1}{n}\mathbf{S}^\mathsf{T}\mathbf{S})\mathbf{V}^\mathsf{T}$ where $\frac{1}{n}\mathbf{S}^\mathsf{T}\mathbf{S}$ is a diagonal matrix with (non-zero) eigenvalues denoted by $\hat{\lambda}_1, \dots, \hat{\lambda}_p$. Then for $i = 1, \dots, p$, the i^{th} coordinate (in the eigenbasis $\mathbf{V}$) of the error vector $\bar{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_*$ after k epochs of mini-batch gradient descent is*

given by

$$[\mathbf{V}^\mathsf{T}(\bar{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_*)]_i = [1 - \alpha\hat{\lambda}_i(1 - p(\hat{\lambda}_i))]^k [\mathbf{V}^\mathsf{T}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)]_i + \frac{1}{\hat{\lambda}_i}\left(1 - [1 - \alpha\hat{\lambda}_i(1 - p(\hat{\lambda}_i))]^k\right)[\mathbf{U}^\mathsf{T}\boldsymbol{\eta}]_i. \tag{4.15}$$

If we make the further simplifying assumption that $\mathbf{V}^\mathsf{T}\Sigma\mathbf{V} = \Lambda$ *is diagonal with eigenvalues* $\lambda_1, \ldots, \lambda_p$,[5] *then the corresponding generalization error is given by*

$$R_{\mathbf{X}}(\bar{\boldsymbol{\beta}}_k) = \sum_{i=1}^{p} \lambda_i [1 - \alpha\hat{\lambda}_i(1 - p(\hat{\lambda}_i))]^{2k} [\mathbf{V}^\mathsf{T}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)]_i + \frac{\sigma^2}{n}\sum_{i=1}^{p} \frac{\lambda_i}{\hat{\lambda}_i}\left(1 - [1 - \alpha\hat{\lambda}_i(1 - p(\hat{\lambda}_i))]^k\right)^2. \tag{4.16}$$

The proof of these claims is obtained from the dynamics described in Theorems 4.3.3 and 4.3.9 under the specific assumptions imposed, and can be found in Section 4.6.3.

For comparison, the corresponding quantities for full-batch gradient descent are the same as those in Proposition 4.3.14 with $\hat{\lambda}_i(1 - p_{B,\alpha}(\hat{\lambda}_i))$ replaced by $\hat{\lambda}_i$. Therefore, if we take $p = p_{B,\alpha}$ from Proposition 4.3.13 as the specific polynomial that motivated the setup, then we deduce that the convergence rate $[1 - \alpha\hat{\lambda}_i(1 - p(\hat{\lambda}_i))]$ for mini-batch gradient descent (4.15) is comparatively smaller due to the shrinkage effect on the spectrum of $\mathbf{W}$. The overall impact on the generalization error (4.16) is less clear, since the change in the convergence rate implies a different trade-off between fitting the signal (i.e., bias) and the noise (i.e., variance). However, if early stopping is used to minimize the generalization error (e.g., [SLR24]), then a particular consequence is mini-batch gradient descent with random reshuffling may have a *different optimal stopping time and a different early stopped risk (possibly lower)*, compared with full-batch gradient descent.

[5]For example, this holds if we assume that the features $\mathbf{x}_i$ are isotropic so that Σ is a scalar multiple of the identity.

Asymptotic analysis in the proportional regime: large n and p

Next, we consider the proportional regime in which both $n, p \to \infty$ such that $p/n \to \gamma \in (0, \infty)$. This setting has been extensively studied in the context of modern large-scale machine learning in prior theoretical works (e.g., [Has+22; CL22; MM22; Ba+22; WSH24]). In this regime, the sample covariance $\mathbf{W}$ does not have a deterministic limit in general. However, its limiting spectral distribution can be studied using tools from random matrix theory if assume that the features $\mathbf{x}_i$ satisfy some concentration properties.

We will also consider the more tractable setting of two-batch gradient descent, recalling that $\alpha\mathbf{Z}(\alpha/2) = \frac{1}{2}\alpha(\mathbf{W}_1 + \mathbf{W}_2) - \frac{1}{8}\alpha^2(\mathbf{W}_2\mathbf{W}_1 + \mathbf{W}_1\mathbf{W}_2)$. This case is also already difficult to analyze in the proportional regime since it requires finding a non-trivial limiting distribution of a non-commutative polynomial of random matrices.

For the remainder of this section, we will assume that the entries of $\mathbf{x}_i$ are i.i.d. standard Gaussian.[6] In this case, it is well-known [MP67; BS10] that almost surely, the empirical spectral distribution[7] $F_{\alpha\mathbf{W}}(x)$ of $\alpha\mathbf{W}$ (known as a *Wishart matrix*) converges in distribution to the *Marchenko-Pastur distribution* with ratio parameter γ and variance α, which has probability measure $\nu_{\gamma,\alpha}$ given by

$$\mathrm{d}\nu_{\gamma,\alpha}(x) := \frac{1}{2\pi\alpha\gamma x}\sqrt{(x_+ - x)(x - x_-)} + \left(1 - \frac{1}{\gamma}\right)_+ \mathbb{1}_{\{x=0\}}, \quad \text{where } x_\pm := \alpha(1 \pm \sqrt{\gamma})^2.$$

That is, $\nu_{\gamma,\alpha}$ has a density supported on $[x_-, x_+]$, and a point mass of $(1 - \gamma^{-1})$ at zero if and only if $\gamma > 1$ (i.e., in the overparameterized regime).

[6] While the limiting spectrum of $\mathbf{W}$ can be described under more general models, such as assuming that $\mathbf{x}_i = \Sigma^{1/2}\mathbf{z}_i$ for some $\mathbf{z}_i$ with i.i.d. coordinates [DW18; Has+22], or that $\mathbf{x}_i$ is a random vector that is subgaussian or satisfies convex concentration [CL22], we will require this strong assumption to study the limiting spectrum of $\mathbf{Z}$ using tools from free probability theory.

[7] The empirical spectral distribution of a symmetric matrix $\mathbf{A} \in \mathbb{R}^{p\times p}$ with eigenvalues $\lambda_i(\mathbf{A})$ is defined by $F_{\mathbf{A}}(x) := \frac{1}{p}\sum_{i=1}^p \mathbb{1}_{\{\lambda_i(\mathbf{A}) \le x\}}$.

To understand the limiting spectrum of $\alpha\mathbf{Z}(\alpha/2)$, our starting point is the observation that

$$\alpha\mathbf{Z}(\alpha/2) = \frac{\alpha}{2}(\mathbf{W}_1 + \mathbf{W}_2) - \frac{\alpha^2}{8}(\mathbf{W}_2\mathbf{W}_1 + \mathbf{W}_2\mathbf{W}_1) = p\left(\frac{\alpha}{2}\mathbf{W}_1, \frac{\alpha}{2}\mathbf{W}_2\right) \tag{4.17}$$

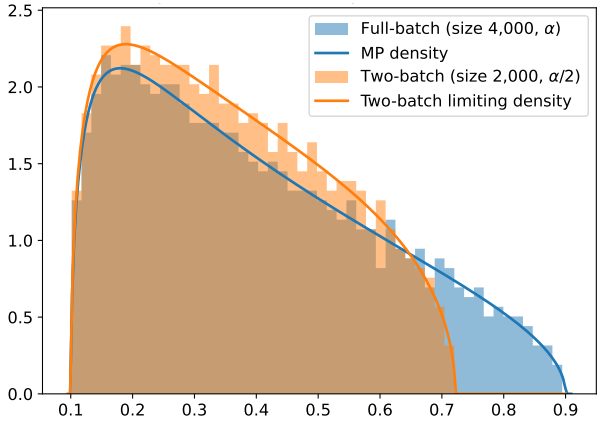
is a *non-commutative polynomial* $p(x, y) = x+y-\frac{1}{2}(xy+yx)$ *in the independent Wishart matrices* $\frac{\alpha}{2}\mathbf{W}_1$ *and* $\frac{\alpha}{2}\mathbf{W}_2$. To understand its spectrum, we need tools from free probability theory, which, roughly speaking, deals with a notion of *free independence* for non-commutative random variables: for the precise mathematical setup, we refer to a standard reference, e.g., [MS17].

The key result needed is that under the Gaussian assumption on $\mathbf{x}_i$, $\frac{\alpha}{2}\mathbf{W}_1$ and $\frac{\alpha}{2}\mathbf{W}_2$ are asymptotically free [MS17, Section 4.5.1], which implies that the limiting spectral distribution of $\alpha\mathbf{Z}(\alpha/2)$ is the spectral distribution of the polynomial $p(w_1, w_2)$ of two freely independent Marchenko-Pastur distributions w_1, w_2 with ratio parameter 2γ and variance $\alpha/2$.

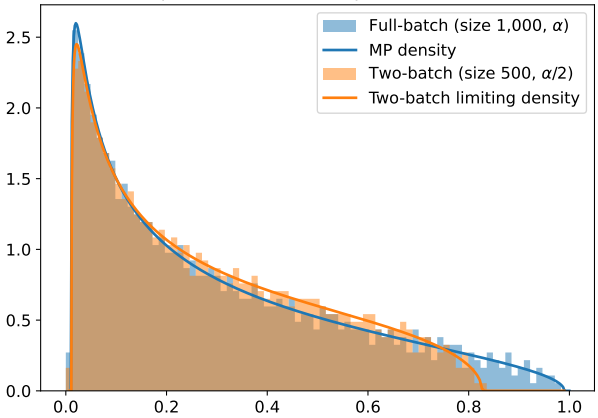
However, there is no closed-form or convenient analytical expression for characterizing the limiting spectral distribution of $\alpha\mathbf{Z}(\alpha/2)$. Instead, we were able to use a general algorithm for computing the spectral distribution of a polynomial of free random variables from [BMS17] for this task by lifting to the space of *operator-valued random variables*. Specifically, after developing a linearization of the non-commutative polynomial in (4.17), we implemented the algorithm in [BMS17] to compute the operator-valued Stieltjes transform of the linearization, from which we could numerically extract the desired spectral distribution of $p(w_1, w_2)$. For a detailed description of our procedure, see Section 4.7.

Figure 4.1 demonstrates our results from computing the limiting spectral distributions of $\alpha\mathbf{Z}(\alpha/2)$ and $\alpha\mathbf{W}$ in the underparameterized ($\gamma < 1$) and overparameterized ($\gamma > 1$) regimes. We observe that batching results in a non-linear shrinkage effect of the spectrum of $\alpha\mathbf{W}$. This is consistent with the conclusions from Section 4.3.3 in a different asymptotic

regime (which does not allow for the overparameterized case). The close adherence between the theoretical predictions and simulations with moderately-sized matrices also highlights the predictive capacity of the asymptotic theory.



(a) Underparameterized case: $\gamma = 1/4$ ($n = 4,000$, $p = 1,000$) and $\alpha = 0.4$.



(b) Overparameterized case: $\gamma = 3/2$ ($n = 1,000$, $p = 1,500$) and $\alpha = 0.2$. Each distribution has a point mass of $(1-\gamma)^{-1}$ at zero not shown.

Figure 4.1: Limiting spectral distributions (lines) of $\alpha\mathbf{W}$ (full-batch) and $\alpha\mathbf{Z}(\alpha/2)$ (two-batch) compared with empirical distribution of a single $n \times p$ standard Gaussian matrix (histogram).

4.4 Concluding remarks

In this work, we showed that the training and generalization error dynamics of mini-batch gradient descent with random reshuffling for least squares regression depend on a sample cross-covariance matrix $\mathbf{Z}$ between the original features and a set of new features that have been modified by the other mini-batches. Using this connection, we established that while the linear scaling rule for the step size matches the dynamics of mini-batch and full-batch gradient descent up to leading order, sampling without replacement results in subtle differences that a continuous-time gradient flow analysis cannot detect. We demonstrated that asymptotically, batching leads to non-linear shrinkage effects on the spectrum of the sample covariance matrix $\mathbf{W}$, which directly affects the mini-batch error dynamics.

Some future directions include studying the dynamics of mini-batch gradient descent with random reshuffling under more specific assumptions to gain insights into the optimal choice of batch size and learning rate for generalization, as well as generalizing the results to more realistic models such as one-layer networks with non-linearities. Finally, there are some random matrix questions on better understanding $\mathbf{Z}$ asymptotically in the proportional regime, such as obtaining precise analytical expressions in the Gaussian setting.

4.5 Additional background on full-batch gradient descent

In this section, we state formulas for the error dynamics and generalization error of full-batch gradient descent (i.e., with $B = 1$). These results are not novel, having appeared in the literature in many varying forms (e.g., [AKT19; RDR22]). However, they are helpful for the purposes of comparison with analogous results for mini-batch gradient descent with random reshuffling.

The first lemma gives an exact expression for the error vector that is driven by the sample covariance matrix $\mathbf{W} := \frac{1}{n}\mathbf{X}^\mathsf{T}\mathbf{X}$ of the features (i.e., Hessian of the least squares problem).

Lemma 4.5.1. *Let $(\boldsymbol{\beta}_k)_{k\geq 0}$ be the sequence of full-batch gradient descent iterates for the least squares problem with step size $\alpha \geq 0$ and initialization $\boldsymbol{\beta}_0 \in \mathbb{R}^p$. Then for all $k \geq 0$,*

$$\boldsymbol{\beta}_k - \boldsymbol{\beta}_* = (\mathbf{I} - \alpha\mathbf{W})^k (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) + \frac{1}{n}\left[\mathbf{I} - (\mathbf{I} - \alpha\mathbf{W})^k\right] \mathbf{W}^\dagger \mathbf{X}^\mathsf{T}\boldsymbol{\eta}. \tag{4.18}$$

Furthermore, if $\mathbf{P}_{\mathbf{X},0} := \mathbf{I} - (\mathbf{X}^\mathsf{T}\mathbf{X})^\dagger(\mathbf{X}^\mathsf{T}\mathbf{X})$ and $\mathbf{P}_{\mathbf{X}^\mathsf{T}} := \mathbf{I} - \mathbf{P}_0$ denote the orthogonal projectors onto the nullspace and row space of $\mathbf{X}$, respectively, then we may decompose the first term as

$$(\mathbf{I} - \alpha\mathbf{W})^k (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) = \mathbf{P}_{\mathbf{X},0}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) + (\mathbf{I} - \alpha\mathbf{W})^k \mathbf{P}_{\mathbf{X}^\mathsf{T}}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*). \tag{4.19}$$

Proof. Since $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}_* + \boldsymbol{\eta}$, the error vector satisfies the recursive relationship

$$\boldsymbol{\beta}_k - \boldsymbol{\beta}_* = \left(\mathbf{I} - \frac{\alpha}{n}\mathbf{X}^\mathsf{T}\mathbf{X}\right)(\boldsymbol{\beta}_{k-1} - \boldsymbol{\beta}_*) + \frac{\alpha}{n}\mathbf{X}^\mathsf{T}\boldsymbol{\eta}.$$

By recursively applying this relationship, and instating the definition of $\mathbf{W} = \frac{1}{n}\mathbf{X}^\mathsf{T}\mathbf{X}$, we obtain

$$\boldsymbol{\beta}_k - \boldsymbol{\beta}_* = (\mathbf{I} - \alpha\mathbf{W})^k(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) + \frac{\alpha}{n}\sum_{j=1}^{k}(\mathbf{I} - \alpha\mathbf{W})^{k-j}\mathbf{X}^\mathsf{T}\boldsymbol{\eta}.$$

The proof of (4.18) is completed by using the following identity to simplify the expression for the sum above, which follows from considering the eigendecomposition of the symmetric matrix $\mathbf{X}$:

$$\sum_{j=1}^{k}(\mathbf{I} - \alpha\mathbf{W})^{k-j}\mathbf{X}^\mathsf{T} = \left[\mathbf{I} - (\mathbf{I} - \alpha\mathbf{W})^k\right](\alpha\mathbf{W})^\dagger\mathbf{X}^\mathsf{T}.$$

Finally, by incorporating the decomposition of the initial error

$$\boldsymbol{\beta}_0 - \boldsymbol{\beta}_* = \mathbf{P}_{\mathbf{X},0}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) + \mathbf{P}_{\mathbf{X}^\mathsf{T}}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*),$$

noting that $(\mathbf{I} - \alpha\mathbf{W})^k\mathbf{P}_{\mathbf{X},0} = \mathbf{P}_{\mathbf{X},0}$, we obtain (4.19). □

The following lemma gives a formula for the generalization error of full-batch gradient descent, corresponding to the usual bias-variance decomposition. It reveals that the generalization error is characterized by the *eigenvalue spectrum of the sample covariance matrix* $\mathbf{W}$, the *alignment of the initial error* $\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*$ *with the eigenspaces of* $\mathbf{W}$, as well as the *covariance of the features* Σ.

Lemma 4.5.2. *Consider the same setup as Lemma 4.5.1. Then for all $k \geq 0$, the generalization error ((4.4)) of the full-batch gradient descent iterates $\boldsymbol{\beta}_k$ is given by*

$$\begin{aligned} R_{\mathbf{X}}(\boldsymbol{\beta}_k) &= (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)^{\mathsf{T}} \mathbf{P}_{\mathbf{X},0} \Sigma \mathbf{P}_{\mathbf{X},0} (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) \\ &\quad + (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)^{\mathsf{T}} \mathbf{P}_{\mathbf{X}^{\mathsf{T}}} \left(\mathbf{I} - \alpha \mathbf{W}\right)^k \Sigma \left(\mathbf{I} - \alpha \mathbf{W}\right)^k \mathbf{P}_{\mathbf{X}^{\mathsf{T}}} (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) \\ &\quad + \frac{\sigma^2}{n} \operatorname{Tr}\left(\left[\mathbf{I} - \left(\mathbf{I} - \alpha \mathbf{W}\right)^k \right] \Sigma \left[\mathbf{I} - \left(\mathbf{I} - \alpha \mathbf{W}\right)^k \right] \mathbf{W}^{\dagger} \right). \end{aligned}$$

Proof. Note that $\|\boldsymbol{\beta}_k - \boldsymbol{\beta}_*\|_{\Sigma}^2 = \|\Sigma^{1/2}(\boldsymbol{\beta}_k - \boldsymbol{\beta}_*)\|_2^2$, where $\|\cdot\|_2$ is the usual ℓ_2 norm. Hence, we may expand the square in (4.18) of Lemma 4.5.1, and use the fact that the cross-terms with a linear dependence on the mean-zero noise term $\boldsymbol{\eta}$ vanish upon taking expectation. The first term of this expansion, combined with the decomposition of the initial error in (4.19), yields the first two terms of the claimed generalization error, corresponding to the bias. The remaining variance term follows from writing the second term of the expansion as a trace (i.e., writing $\|\mathbf{z}^{\mathsf{T}}\mathbf{z}\|_2^2 = \operatorname{Tr}\left(\mathbf{z}\mathbf{z}^{\mathsf{T}}\right)$), using the fact that $\mathbb{E}\left[\boldsymbol{\eta}\boldsymbol{\eta}^{\mathsf{T}}\right] = \sigma^2 \mathbf{I}$, the cyclic property of trace, and the property $\mathbf{W}^{\dagger}\mathbf{W}\mathbf{W}^{\dagger} = \mathbf{W}^{\dagger}$ of the pseudoinverse. □

Observe that by taking the limit as $k \to \infty$ with a small enough step size, Lemma 4.5.1 shows that gradient descent converges to the min-norm solution $(\mathbf{X}^{\mathsf{T}}\mathbf{X})^{\dagger}\mathbf{X}^{\mathsf{T}}\mathbf{y}$ of the least squares problem, shifted by the projection of $\boldsymbol{\beta}_0$ onto the nullspace of $\mathbf{X}$. Additionally, Lemma 4.5.2 shows that the resulting generalization error is increased by small eigenvalues of $\mathbf{W}$, which corresponds to overfitting the noise.

Corollary 4.5.3. *Consider the same setup as Lemma 4.5.1. Let $\boldsymbol{\beta}_{\infty} := \mathbf{P}_{\mathbf{X},0}\boldsymbol{\beta}_0 + (\mathbf{X}^{\mathsf{T}}\mathbf{X})^{\dagger}\mathbf{X}^{\mathsf{T}}\mathbf{y}$. If $\alpha < 2/(n^{-1}\|\mathbf{X}^{\mathsf{T}}\mathbf{X}\|)$, then $\boldsymbol{\beta}_k \to \boldsymbol{\beta}_{\infty}$ as $k \to \infty$, and the limiting generalization error is given by*

$$R_{\mathbf{X}}(\boldsymbol{\beta}_{\infty}) = (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)^{\mathsf{T}} \mathbf{P}_{\mathbf{X},0} \Sigma \mathbf{P}_{\mathbf{X},0} (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) + \frac{\sigma^2}{n} \operatorname{Tr}\left(\Sigma \mathbf{W}^{\dagger}\right).$$

4.6 Technical proofs for mini-batch gradient descent

In this section, we provide the technical proofs for our results on mini-batch gradient descent with random reshuffling. Specifically, in Section 4.6.1, we prove the general results for general mini-batch gradient descent (Lemma 4.3.1, Theorem 4.3.3, and Theorem 4.3.9). In Section 4.6.2, we prove some more precise results in the specific setting of two-batch gradient descent with $B = 2$ (Propositions 4.3.8 and 4.3.11). Finally, in Section 4.6.3, we prove the results obtained in the asymptotic regime as $n \to \infty$ with fixed p (Propositions 4.3.13 and 4.3.14).

4.6.1 Mini-batch gradient descent

As a reminder of our setup, recall that we have partitioned the data matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ into B equally-sized batches $\mathbf{X}_1, \dots, \mathbf{X}_B \in \mathbb{R}^{(m/B) \times n}$, and we denote the corresponding sample covariance matrix of $\mathbf{X}_b$ by $\mathbf{W}_b = \frac{B}{n} \mathbf{X}_b^\mathsf{T} \mathbf{X}_b$. The modified mini-batches are defined by $\widetilde{\mathbf{X}}_b := \mathbf{X}_b \Pi_b$, where Π_b was defined in (4.5) as

$$\Pi_b = \frac{1}{B!} \sum_{\tau \in S_B} \prod_{j : j < \tau^{-1}(b)} (\mathbf{I} - \alpha \mathbf{W}_{\tau(j)}). \tag{4.20}$$

Another equivalent expression for Π_b is the following:

$$\Pi_b = \frac{1}{B!} \sum_{\tau \in S_B} \prod_{j : j > \tau^{-1}(b)} (\mathbf{I} - \alpha \mathbf{W}_{\tau(j)}). \tag{4.21}$$

This is because each summand corresponding to a permutation τ in (4.20) can be matched one-to-one to a summand corresponding to a permutation τ' in (4.21) by swapping the sub-permutations to the left and right of the batch b (with position $\tau^{-1}(b)$). For example, without loss of generality, consider $\tau = (1, \dots, b-1, b, b+1, \dots, B)$, and let $\tau' = (B, \dots, b+1, b, 1, \dots, b-1)$. Then the summand in (4.20) for τ is $(\mathbf{I} - \alpha \mathbf{W}_{b-1}) \dots (\mathbf{I} - \alpha \mathbf{W}_1)$, which exactly corresponds to the summand for τ' in (4.21).

Furthermore, by expanding, it can be verified that another expression for Π_b is the following:[8]

$$\Pi_b = \mathbf{I} - \sum_{i=1}^{B-1} \left\{ \frac{(-1)^{i+1}}{(i+1)!} \cdot \alpha^i \sum_{\substack{\{b_1,\ldots,b_i\} \subseteq [B] \setminus \{b\} \\ b_1,\ldots,b_i \text{ distinct, ordered}}} \mathbf{W}_{b_1} \ldots \mathbf{W}_{b_i} \right\}. \tag{4.22}$$

Finally, $\widetilde{\mathbf{X}}$ denotes the concatenation of the B modified mini-batches in the same order, and $\mathbf{Z} = \frac{1}{n}\widetilde{\mathbf{X}}^\top \mathbf{X} = \frac{1}{n}\sum_{b=1}^{B} \Pi_b \mathbf{X}_b^\top \mathbf{X}_b$ was defined in (4.6) to be the sample cross-covariance matrix of the modified features with the original features.

Proof of Lemma 4.3.1

Since $\mathbf{Z} = \frac{1}{n}\widetilde{\mathbf{X}}^\top \mathbf{X} = \frac{1}{n}\sum_{b=1}^{B} \widetilde{\mathbf{X}}_b^\top \mathbf{X}_b = \frac{1}{n}\sum_{b=1}^{B} \Pi_b \mathbf{X}_b^\top \mathbf{X}_b$, it suffices to show that

$$\sum_{b=1}^{B} \Pi_b \mathbf{W}_b = \sum_{b=1}^{B} \mathbf{W}_b \Pi_b$$

to prove that $\mathbf{Z}$ is symmetric, where Π_b is defined as in (4.21). Fix $b \in \{1, \ldots, B\}$. Note that $\Pi_b \mathbf{W}_b$ and $\mathbf{W}_b \Pi_b$ are polynomials in the non-commuting variables $\mathbf{W}_1, \ldots, \mathbf{W}_B$, and that Π_b does not contain the term $\mathbf{W}_b$. Hence, it suffices to argue that the word ending in $\mathbf{W}_b$ on the left hand side (i.e., $\Pi_b \mathbf{W}_b$) matches the word ending in $\mathbf{W}_b$ on the right hand side (i.e., the sum of the words ending in $\mathbf{W}_b$ in $\sum_{j \neq b} \mathbf{W}_j \Pi_j$).

Observe that $\Pi_b \mathbf{W}_b$ is a sum of words of the form $a_{i_1,\ldots,i_\ell} \mathbf{W}_{i_1} \mathbf{W}_{i_2} \cdots \mathbf{W}_{i_\ell} \mathbf{W}_b$, where each of the indices are distinct and $a_{i_1,\ldots,i_\ell} \in \mathbb{R}$ is a constant. From the form of Π_b, this term arises as a sum over permutations τ from a set, say $\mathcal{T} \equiv \mathcal{T}_{i_1,\ldots,i_\ell}$, such that $\tau^{-1}(b) < \tau^{-1}(i_\ell) < \cdots < \tau^{-1}(i_2) < \tau^{-1}(i_1)$:

$$a_{i_1,\ldots,i_\ell} \mathbf{W}_{i_1} \mathbf{W}_{i_2} \cdots \mathbf{W}_{i_\ell} \mathbf{W}_b = \left(\frac{1}{B!} \sum_{\tau \in \mathcal{T}} (-\alpha)^\ell \mathbf{W}_{i_1} \mathbf{W}_{i_2} \cdots \mathbf{W}_{i_\ell} \right) \mathbf{W}_b.$$

[8] For example: with $B = 2$, $\Pi_1 = \mathbf{I} - \frac{1}{2}\alpha \mathbf{W}_2$; with $B = 3$, $\Pi_1 = \mathbf{I} - \frac{1}{2}\alpha(\mathbf{W}_2 + \mathbf{W}_3) + \frac{1}{6}\alpha^2(\mathbf{W}_2\mathbf{W}_3 + \mathbf{W}_3\mathbf{W}_2)$; with $B = 4$, $\Pi_1 = \mathbf{I} - \frac{1}{2}\alpha(\mathbf{W}_2 + \mathbf{W}_3 + \mathbf{W}_4) + \frac{1}{6}\alpha^2(\mathbf{W}_2\mathbf{W}_3 + \mathbf{W}_2\mathbf{W}_4 + \mathbf{W}_3\mathbf{W}_2 + \mathbf{W}_3\mathbf{W}_4 + \mathbf{W}_4\mathbf{W}_2 + \mathbf{W}_4\mathbf{W}_3) - \frac{1}{24}\alpha^3(\mathbf{W}_2\mathbf{W}_3\mathbf{W}_4 + \mathbf{W}_2\mathbf{W}_4\mathbf{W}_3 + \mathbf{W}_3\mathbf{W}_2\mathbf{W}_4 + \mathbf{W}_3\mathbf{W}_4\mathbf{W}_2 + \mathbf{W}_4\mathbf{W}_2\mathbf{W}_3 + \mathbf{W}_4\mathbf{W}_3\mathbf{W}_2)$; and so on.

The same word arises in the expression $\sum_{j\neq b}\mathbf{W}_j\Pi_j$ from the single term $\mathbf{W}_{i_1}\Pi_{i_1}$ with $\mathbf{W}_{i_1}$ as the leftmost matrix in the product. For each $\tau\in\mathcal{T}$, consider shifting the sub-permutation $(b,i_\ell,\ldots,i_2,i_1)$ in τ cyclically to the right (keeping the other entries fixed) to obtain the permutation τ' with sub-permutation $(i_1,b,i_\ell,\ldots,i_2)$. If $\mathcal{T}'$ denotes the set of permutations obtained from $\mathcal{T}$ in this way, then by summing over all $\tau'\in\mathcal{T}$ in Π_{i_1} — choosing the term $-\alpha\mathbf{W}_{\tau'(j)}$ for each $j\in\{\tau^{-1}(b),\tau^{-1}(i_\ell),\ldots,\tau^{-1}(i_2)\}$, and $\mathbf{I}$ for the rest of the indices in the product over τ' — this shows that the word $a'_{i_2,\ldots,i_\ell,b}\mathbf{W}_{i_1}\mathbf{W}_{i_2}\cdots\mathbf{W}_{i_\ell}\mathbf{W}_b$ appearing in $\mathbf{W}_{i_1}\Pi_{i_1}$ is equal to

$$\mathbf{W}_{i_1}\left(\frac{1}{B!}\sum_{\tau'\in\mathcal{T}'}(-\alpha)^\ell\mathbf{W}_{i_2}\cdots\mathbf{W}_{i_\ell}\mathbf{W}_b\right)=a_{i_1,\ldots,i_\ell}\mathbf{W}_{i_1}\mathbf{W}_{i_2}\cdots\mathbf{W}_{i_\ell}\mathbf{W}_b.$$

Thus, we conclude that $\sum_{b=1}^B\Pi_b\mathbf{W}_b=\sum_{b=1}^B\mathbf{W}_b\Pi_b$, and hence $\mathbf{Z}$ is symmetric.

Next, we will prove that range$\mathbf{Z}\subseteq$ range$\widetilde{\mathbf{X}}^\mathsf{T}\subseteq$ range$\mathbf{X}^\mathsf{T}$. Since $\mathbf{Z}\mathbf{w}=\frac{1}{n}\sum_{b=1}^B\widetilde{\mathbf{X}}_b^\mathsf{T}\mathbf{X}_b\mathbf{w}$ for any $\mathbf{w}\in\mathbb{R}^p$, it is clear that range$\mathbf{Z}\subseteq$ range$\widetilde{\mathbf{X}}^\mathsf{T}$. Next, let $\mathbf{y}\in\mathbb{R}^n$ be a generic vector partitioned into $\mathbf{y}_1,\ldots,\mathbf{y}_B$ in the same way as the batches $\mathbf{X}_1,\ldots,\mathbf{X}_b$. From expanding the product in Π_b, we can write $\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{y}=\sum_{b=1}^B\Pi_b\mathbf{X}_b^\mathsf{T}\mathbf{y}_b=\sum_{b=1}^B\mathbf{X}_b^\mathsf{T}\mathbf{y}_b+\sum_{b=1}^B\alpha_b\mathbf{X}_b^\mathsf{T}\mathbf{X}_b\mathbf{v}_b$ for some coefficients $\alpha_b\in\mathbb{R}$ and vectors $\mathbf{v}_b$. Hence, range$\widetilde{\mathbf{X}}^\mathsf{T}\subseteq$ range$\mathbf{X}^\mathsf{T}$.

Proof of Theorem 4.3.3

Lemma 4.6.1. *Let $\mathbf{A}\in\mathbb{R}^{p\times p}$ be a symmetric matrix, and define $\mathbf{P}=(\mathbf{I}-\mathbf{A})(\mathbf{I}-\mathbf{A})^\dagger$ and $\mathbf{P}_0=\mathbf{I}-\mathbf{P}$ to be the orthogonal projectors onto the range and nullspace of $\mathbf{I}-\mathbf{A}$, respectively. Then we have*

$$(\mathbf{I}+\mathbf{A}+\cdots+\mathbf{A}^{k-1})=(\mathbf{I}-\mathbf{A}^k)(\mathbf{I}-\mathbf{A})^\dagger+k\mathbf{P}_0.$$

Proof. Since $\mathbf{I}=\mathbf{P}+\mathbf{P}_0$, we can write $(\mathbf{I}+\mathbf{A}+\cdots+\mathbf{A}^{k-1})=(\mathbf{I}+\mathbf{A}+\cdots+\mathbf{A}^{k-1})(\mathbf{P}+\mathbf{P}_0)$. By multiplying both sides of the algebraic identity $(\mathbf{I}+\mathbf{A}+\cdots+\mathbf{A}^{k-1})(\mathbf{I}-\mathbf{A})=(\mathbf{I}-\mathbf{A}^k)$ by $(\mathbf{I}-\mathbf{A})^\dagger$, we have $(\mathbf{I}+\mathbf{A}+\cdots+\mathbf{A}^{k-1})\mathbf{P}=(\mathbf{I}-\mathbf{A}^k)(\mathbf{I}-\mathbf{A})^\dagger$, which yields the first term.

For the second term, note that $\mathbf{A}^\ell \mathbf{P}_0 = \mathbf{P}_0$ for any $\ell \geq 1$, since $\mathbf{A}\mathbf{x} = \mathbf{x}$ for any $\mathbf{x}$ in the nullspace of $\mathbf{I} - \mathbf{A}$. Thus, $(\mathbf{I} + \mathbf{A} + \cdots + \mathbf{A}^{k-1})\mathbf{P}_0 = k\mathbf{P}_0$, which yields the second term. $\square$

Proof of Theorem 4.3.3. Recall that from (4.2), the iterates $\boldsymbol{\beta}_k^{(b)}$ from mini-batch gradient descent after b iterations over the mini-batches in the k^{th} epoch satisfy

$$\boldsymbol{\beta}_k^{(b)} = \boldsymbol{\beta}_k^{(b)} - \frac{B\alpha}{n} \mathbf{X}_{\tau(b)}^\mathsf{T} (\mathbf{X}_{\tau(b)} \boldsymbol{\beta}_k^{(b-1)} - \mathbf{y}_{\tau(b)}), \quad b = 1, 2, \ldots, B,$$

given a permutation $\tau = (\tau(1), \tau(2), \ldots, \tau(B))$ of the mini-batches in the k^{th} epoch, where $\boldsymbol{\beta}_k^{(0)} := \boldsymbol{\beta}_{k-1}^{(B)}$ and $\boldsymbol{\beta}_0^{(B)} := \boldsymbol{\beta}_0$. By using the fact that $\mathbf{y}_b = \mathbf{X}_b \boldsymbol{\beta}_* + \boldsymbol{\eta}_b$ for each mini-batch, the displayed equation above rearranges to

$$\boldsymbol{\beta}_k^{(b)} - \boldsymbol{\beta}_* = \left(\mathbf{I} - \frac{B\alpha}{n} \mathbf{X}_{\tau(b)}^\mathsf{T} \mathbf{X}_{\tau(b)} \right) (\boldsymbol{\beta}_k^{(b-1)} - \boldsymbol{\beta}_*) + \frac{B\alpha}{n} \mathbf{X}_{\tau(b)}^\mathsf{T} \boldsymbol{\eta}_{\tau(b)}, \quad b = 1, 2, \ldots, B.$$

By iterating this relationship, we deduce that the estimate at the end of the k^{th} epoch satisfies

$$\boldsymbol{\beta}_k^{(B)} - \boldsymbol{\beta}_* = \prod_{b=1}^{B} (\mathbf{I} - \alpha \mathbf{W}_{\tau(b)}) (\boldsymbol{\beta}_{k-1}^{(B)} - \boldsymbol{\beta}_*) + \frac{B\alpha}{n} \sum_{b=1}^{B} \prod_{j : j > \tau^{-1}(b)} (\mathbf{I} - \alpha \mathbf{W}_{\tau(j)}) \mathbf{X}_b^\mathsf{T} \boldsymbol{\eta}_b. \tag{4.23}$$

Recall that $\bar{\boldsymbol{\beta}}_k = \mathbb{E}_{\tau \sim \text{Unif}(S_B)} \left[\boldsymbol{\beta}_k^{(B)} \right]$. Hence, by taking the expectation over the random permutations of the batches in each epoch, drawn uniformly from the $B!$ permutations in the symmetric group S_B of B elements, the error vector $\bar{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_*$ satisfies the recursive relationship

$$\begin{aligned} \bar{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_* = \frac{1}{B!} \sum_{\tau \in S_B} \prod_{b=1}^{B} (\mathbf{I} - \alpha \mathbf{W}_{\tau(b)}) (\bar{\boldsymbol{\beta}}_{k-1} - \boldsymbol{\beta}_*) \\ + \frac{B\alpha}{n} \left\{ \frac{1}{B!} \sum_{\tau \in S_B} \sum_{b=1}^{B} \prod_{j : j > \tau^{-1}(b)} (\mathbf{I} - \alpha \mathbf{W}_{\tau(j)}) \mathbf{X}_b^\mathsf{T} \right\} \boldsymbol{\eta}_b. \end{aligned} \tag{4.24}$$

By moving the sum over b outside, recognizing the definition of Π_b from (4.21), and recalling that $\widetilde{\mathbf{X}}_b^\mathsf{T} = \Pi_b \mathbf{X}_b$, the second term is equal to

$$\frac{B\alpha}{n} \sum_{b=1}^{B} \widetilde{\mathbf{X}}_b^\mathsf{T} \boldsymbol{\eta}_b = \frac{B\alpha}{n} \widetilde{\mathbf{X}}^\mathsf{T} \boldsymbol{\eta}.$$

Next, by writing $\mathbf{Z} = \frac{1}{n} \sum_{b=1}^{B} \widetilde{\mathbf{X}}_b^\mathsf{T} \mathbf{X}_b$, we have

$$\mathbf{Z} = \frac{1}{B\alpha} \left(\frac{1}{B!} \sum_{\tau \in S_B} \sum_{b=1}^{B} \prod_{j : j > \tau^{-1}(b)} (\mathbf{I} - \alpha \mathbf{W}_{\tau(j)}) \alpha \mathbf{W}_b \right). \tag{4.25}$$

We claim that the identity

$$\frac{1}{B!} \sum_{\tau \in S_B} \sum_{b=1}^{B} \prod_{j : j > \tau^{-1}(b)} (\mathbf{I} - \alpha \mathbf{W}_{\tau(j)}) \alpha \mathbf{W}_b = \mathbf{I} - \frac{1}{B!} \sum_{\tau \in S_B} \prod_{b=1}^{B} (\mathbf{I} - \alpha \mathbf{W}_{\tau(b)}) \tag{4.26}$$

holds. Assuming that this is true for now, combining (4.25) and (4.26) shows that (4.24) can be written as

$$\bar{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_* = (\mathbf{I} - B\alpha \mathbf{Z})(\bar{\boldsymbol{\beta}}_{k-1} - \boldsymbol{\beta}_*) + \frac{B\alpha}{n} \widetilde{\mathbf{X}}^\mathsf{T} \boldsymbol{\eta}. \tag{4.27}$$

Hence, by recursively applying this relationship, we obtain

$$\bar{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_* = (\mathbf{I} - B\alpha \mathbf{Z})^k (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) + \frac{B\alpha}{n} \sum_{j=1}^{k} (\mathbf{I} - B\alpha \mathbf{Z})^{k-j} \widetilde{\mathbf{X}}^\mathsf{T} \boldsymbol{\eta}.$$

The proof of (4.10) is completed by using the following identity from Lemma 4.6.1 to simplify the expression for the sum above:

$$\sum_{j=1}^{k} (\mathbf{I} - B\alpha \mathbf{Z})^{k-j} \, \widetilde{\mathbf{X}}^\mathsf{T} = \left[\mathbf{I} - (\mathbf{I} - B\alpha \mathbf{Z})^k \right] (B\alpha \mathbf{Z})^\dagger \, \widetilde{\mathbf{X}}^\mathsf{T}.$$

Here, we use the assumption that range$\widetilde{\mathbf{X}}^\mathsf{T} \subseteq$ range$\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X} =$ range$\mathbf{Z}$ to deduce that $\mathbf{P}_{\mathbf{Z},0}\widetilde{\mathbf{X}}^\mathsf{T} = \mathbf{0}$. Furthermore, by incorporating the decomposition of the initial error

$$\boldsymbol{\beta}_0 - \boldsymbol{\beta}_* = \mathbf{P}_{\mathbf{Z},0}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) + \mathbf{P}_{\mathbf{Z}}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*),$$

noting that $(\mathbf{I} - B\alpha\mathbf{Z})^k\mathbf{P}_{\mathbf{Z},0} = \mathbf{P}_{\mathbf{Z},0}$, we obtain (4.11).

Finally, it remains to prove that the identity (4.26) holds. We prove the equivalent identity, noting that $|S_B| = B!$ so that the identity matrix $\mathbf{I}$ can be brought inside the sum:

$$\frac{1}{B!}\sum_{\tau\in S_B}\sum_{b=1}^{B}\prod_{j:j>\tau^{-1}(b)}(\mathbf{I}-\alpha\mathbf{W}_{\tau(j)})\alpha\mathbf{W}_b = \frac{1}{B!}\sum_{\tau\in S_B}\left(\mathbf{I}-\prod_{b=1}^{B}(\mathbf{I}-\alpha\mathbf{W}_{\tau(b)})\right). \tag{4.28}$$

We will prove this by matching each summand on the left hand side to a summand on the right hand side. Fix a permutation $\tau \in S_B$; without loss of generality, we may assume that $\tau = (1, 2, \ldots, B-1, B)$. On the left hand side, the summand corresponding to τ is

$$\alpha\mathbf{W}_B + (\mathbf{I}-\alpha\mathbf{W}_B)\alpha\mathbf{W}_{B-1} + \cdots + (\mathbf{I}-\alpha\mathbf{W}_B)\cdots(\mathbf{I}-\alpha\mathbf{W}_3)(\mathbf{I}-\alpha\mathbf{W}_2)\alpha\mathbf{W}_1. \tag{4.29}$$

On the right hand side, the summand corresponding to τ is

$$\mathbf{I} - (\mathbf{I}-\alpha\mathbf{W}_B)(\mathbf{I}-\alpha\mathbf{W}_{B-1})\cdots(\mathbf{I}-\alpha\mathbf{W}_2)(\mathbf{I}-\alpha\mathbf{W}_1). \tag{4.30}$$

Consider expanding the product by choosing a term from each bracket going from right to left. For the last bracket, choosing $\alpha\mathbf{W}_1$ yields the term $(\mathbf{I}-\alpha\mathbf{W}_B)\cdots(\mathbf{I}-\alpha\mathbf{W}_3)(\mathbf{I}-\alpha\mathbf{W}_2)\alpha\mathbf{W}_1$ ending in $\alpha\mathbf{W}_1$, matching the left hand side. Otherwise, choosing $\mathbf{I}$ results in a smaller product to which the same argument can be applied recursively. In the end, we are left with the single term $(\alpha\mathbf{W}_B - \mathbf{I}) - \mathbf{I}$, so that the identity vanishes and we are left with $\alpha\mathbf{W}_B$. Thus, we see that (4.29) and (4.30) correspond to the exact same expression, and summing over all $\tau \in S_B$ completes the proof of the claim (4.28). □

Proof of Corollary 4.3.5. Recall that $\mathbf{P}_{\mathbf{Z},0} = \mathbf{I} - \mathbf{Z}^\dagger\mathbf{Z}$ and $\mathbf{Z} = \frac{1}{n}\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}$. From (4.10) and (4.11) of Theorem 4.3.3, it is clear that if $\|(\mathbf{I} - B\alpha\mathbf{W})\mathbf{P}_\mathbf{Z}\| < 1$, then $\bar{\boldsymbol{\beta}}_k$ converges as $k \to \infty$ to the vector

$$\mathbf{P}_{\mathbf{Z},0}\boldsymbol{\beta}_0 + \mathbf{Z}^\dagger\mathbf{Z}\boldsymbol{\beta}_* + \frac{1}{n}\mathbf{Z}^\dagger\widetilde{\mathbf{X}}^\mathsf{T}\boldsymbol{\eta} = \mathbf{P}_{\mathbf{Z},0}\boldsymbol{\beta}_0 + (\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X})^\dagger\widetilde{\mathbf{X}}^\mathsf{T}(\mathbf{X}\boldsymbol{\beta}_* + \boldsymbol{\eta}).$$

Since $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}_* + \boldsymbol{\eta}$, we obtain the claimed expression for the limiting vector $\bar{\boldsymbol{\beta}}_\infty$. □

On the assumptions in Theorem 4.3.3

In this section, we expand upon the discussion on the assumption in Theorem 4.3.3 that range$\widetilde{\mathbf{X}}^\mathsf{T} \subseteq$ range$\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}$.

- In the overparameterized case ($p \geq n$), this follows if $\mathbf{X} \in \mathbb{R}^{n\times p}$ has rank n. Thus, for any $\boldsymbol{\eta} \in \mathbb{R}^n$, we can write $\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\theta}$ for some $\boldsymbol{\theta} \in \mathbb{R}^p$. Hence, $\widetilde{\mathbf{X}}^\mathsf{T}\boldsymbol{\eta} = \widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}\boldsymbol{\theta} \in$ range$\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}$.

- In the underparameterized case ($p < n$), this also follows if we assume that $\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}$ (or equivalently $\mathbf{Z}$) has rank p since range$\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X} = \mathbb{R}^p$.

  Next, using less trivial assumptions, the condition also follows if we assume that range$\widetilde{\mathbf{X}} \subseteq$ range$\mathbf{X}$. For $\boldsymbol{\eta} \in \mathbb{R}^n$, let $\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}\boldsymbol{\theta}$ be the projection of $\widetilde{\mathbf{X}}^\mathsf{T}\boldsymbol{\eta}$ onto range$\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}$, where $\boldsymbol{\theta} \in \mathbb{R}^p$. Thus, $\widetilde{\mathbf{X}}^\mathsf{T}\boldsymbol{\eta} - \widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}\boldsymbol{\theta}$ is orthogonal to range$\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}$, or in other words,

  $$\mathbf{X}^\mathsf{T}\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^\mathsf{T}(\boldsymbol{\eta} - \mathbf{X}\boldsymbol{\theta}) = \mathbf{0}.$$

  We claim that $\widetilde{\mathbf{X}}^\mathsf{T}\boldsymbol{\eta} = \widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}\boldsymbol{\theta}$. Since range$\widetilde{\mathbf{X}} \subseteq$ range$\mathbf{X}$, we have $\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^\mathsf{T}(\boldsymbol{\eta} - \mathbf{X}\boldsymbol{\theta}) \in$ range$\mathbf{X}$, and thus $\mathbf{X}^\mathsf{T}\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^\mathsf{T}(\boldsymbol{\eta} - \mathbf{X}\boldsymbol{\theta}) = \mathbf{0}$ if and only if $\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^\mathsf{T}(\boldsymbol{\eta} - \mathbf{X}\boldsymbol{\theta}) = \mathbf{0}$. Furthermore, $\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^\mathsf{T}(\boldsymbol{\eta} - \mathbf{X}\boldsymbol{\theta}) = \mathbf{0}$ if and only if $\widetilde{\mathbf{X}}^\mathsf{T}(\boldsymbol{\eta} - \mathbf{X}\boldsymbol{\theta}) = \mathbf{0}$, which completes the proof.

The assumption in the overparameterized case (which is arguably the more interesting case for machine learning applications) is natural, and does not depend on the structure of the

mini-batches or the step size. The underparameterized case seems to be more delicate, and it remains unclear what the necessary assumptions on the structure of the mini-batches or on the step size are in this regime for the required condition to hold. However, in our numerical experiments, $\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}$ was always observed to have the same rank as $\mathbf{X}$, so the assumption is likely to be typically satisfied generically.

In fact, we observed that range$\mathbf{X}^\mathsf{T}$ and range$\widetilde{\mathbf{X}}^\mathsf{T}\mathbf{X}$ always appeared to be very similar, if not identical, which suggests that it may be possible to prove that the two subspaces coincide under a set of generic assumptions.

Mini-batching with replacement

Here, we will provide more details on our claims in Remark 4.3.4 on the error dynamics when the mini-batches are sampled *with replacement*. Specifically, suppose that in each iteration, we sample a mini-batch with replacement uniformly at random from the same set of B mini-batches $\mathbf{X}_1, \dots, \mathbf{X}_b$ instead. If $\widehat{\boldsymbol{\beta}}_k$ denotes the mean iterate after k epochs (which corresponds to Bk iterations), averaged over the i.i.d. sampling of the mini-batches in each iteration, then we will show that

$$\widehat{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_* = (\mathbf{I} - \alpha\mathbf{W})^{Bk}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) + \frac{1}{n}\left[\mathbf{I} - (\mathbf{I} - \alpha\mathbf{W})^{Bk}\right]\mathbf{W}^\dagger\mathbf{X}^\mathsf{T}\boldsymbol{\eta}. \tag{4.31}$$

Thus, we see that the error dynamics when sampling with replacement are essentially identical to those of full-batch gradient descent up to a time change by a factor of B.

Proof of (4.31). Let $\widehat{\boldsymbol{\beta}}_{k,t}$ denote the mean parameters in the k^{th} epoch after t iterations. (Thus, $\widehat{\boldsymbol{\beta}}_k = \widehat{\boldsymbol{\beta}}_{k,0} = \widehat{\boldsymbol{\beta}}_{k-1,B}$.) Note that $\frac{1}{n}\sum_{b=1}^B \mathbf{X}_b^\mathsf{T}\mathbf{X}_b = \frac{1}{n}\mathbf{X}^\mathsf{T}\mathbf{X} = \mathbf{W}$, and $\sum_{b=1}^B \mathbf{X}_b^\mathsf{T}\boldsymbol{\eta}_b = \mathbf{X}^\mathsf{T}\boldsymbol{\eta}$. Conditional on the iterate $\widehat{\boldsymbol{\beta}}_{k,t}$, the next mini-batch is sampled uniformly at random from the B mini-batches $\mathbf{X}_1, \dots \mathbf{X}_B$. Hence, we can develop the following recursive expression

for the expected error vector after one iteration:

$$\begin{aligned}\widehat{\boldsymbol{\beta}}_{k,t+1} - \boldsymbol{\beta}_* &= \frac{1}{B}\sum_{b=1}^{B}\left\{\left(\mathbf{I} - \frac{B\alpha}{n}\mathbf{X}_b^\mathsf{T}\mathbf{X}_b\right)(\widehat{\boldsymbol{\beta}}_{k,t} - \boldsymbol{\beta}_*) - \frac{B\alpha}{n}\mathbf{X}_b^\mathsf{T}\boldsymbol{\eta}_b\right\} \\ &= (\mathbf{I} - \alpha\mathbf{W})(\widehat{\boldsymbol{\beta}}_{k,t} - \boldsymbol{\beta}_*) - \frac{\alpha}{n}\mathbf{X}^\mathsf{T}\boldsymbol{\eta}.\end{aligned}$$

By iterating over Bk iterations until the end of the k^{th} epoch, and simplifying the matrix geometric series using Lemma 4.6.1, we obtain (4.31). □

Proof of Theorem 4.3.9

By expanding the square in (4.10) of Theorem 4.3.3 and using the fact that the cross-terms vanish upon taking expectation with respect to the mean-zero noise $\boldsymbol{\eta}$, denoted by $\mathbb{E}_{\boldsymbol{\eta}}$, the generalization error $R_{\mathbf{X}}(\bar{\boldsymbol{\beta}}_k)$ is equal to

$$\mathbb{E}_{\boldsymbol{\eta}}\|\bar{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_*\|_\Sigma^2 = \|\Sigma^{1/2}(\mathbf{I} - B\alpha\mathbf{Z})^k(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)\|_2^2 + \mathbb{E}_{\boldsymbol{\eta}}\left\|\frac{1}{n}\Sigma^{1/2}\left[\mathbf{I} - (\mathbf{I} - B\alpha\mathbf{Z})^k\right]\mathbf{Z}^\dagger\widetilde{\mathbf{X}}^\mathsf{T}\boldsymbol{\eta}\right\|_2^2.$$

Since $\mathbf{Z}$ is symmetric, the first term is equal to $(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)^\mathsf{T}(\mathbf{I} - B\alpha\mathbf{Z})^k\Sigma(\mathbf{I} - B\alpha\mathbf{Z})^k(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)$. When combined with the decomposition of the initial error in (4.11), this yields the first two terms of the claimed generalization error, corresponding to the bias. The second term of the expansion above, written as a trace using the cyclic property, is equal to

$$\frac{1}{n^2}\operatorname{Tr}\left(\Sigma\left[\mathbf{I} - (\mathbf{I} - B\alpha\mathbf{Z})^k\right]\mathbf{Z}^\dagger\widetilde{\mathbf{X}}^\mathsf{T}\mathbb{E}_{\boldsymbol{\eta}}[\boldsymbol{\eta}\boldsymbol{\eta}^\mathsf{T}]\widetilde{\mathbf{X}}\mathbf{Z}^\dagger\left[\mathbf{I} - (\mathbf{I} - B\alpha\mathbf{Z})^k\right]\right).$$

Since $\mathbb{E}_{\boldsymbol{\eta}}[\boldsymbol{\eta}\boldsymbol{\eta}^\mathsf{T}] = \sigma^2\mathbf{I}$, this completes the proof.

4.6.2 Two-batch gradient descent

For the following proofs, we use the Loewner order defined by the cone of positive semidefinite matrices: that is, for symmetric matrices $\mathbf{A}, \mathbf{B}$, we have $\mathbf{A} \preceq \mathbf{B}$ if and only if $\mathbf{B} - \mathbf{A}$ is

positive semidefinite, or equivalently $\mathbf{x}^\mathsf{T}\mathbf{A}\mathbf{x} \le \mathbf{x}^\mathsf{T}\mathbf{B}\mathbf{x}$ for all unit vectors $\mathbf{x}$. We recall some basic properties of the Loewner order: if $\mathbf{A} \preceq \mathbf{B}$ and $\mathbf{C} \preceq \mathbf{D}$, then

- (Preserved by conjugation) $\mathbf{C}^\mathsf{T}\mathbf{A}\mathbf{C} \preceq \mathbf{C}^\mathsf{T}\mathbf{B}\mathbf{C}$ for any $\mathbf{C}$ with compatible dimensions
- $\mathbf{A} + \mathbf{B} \preceq \mathbf{C} + \mathbf{D}$ and $\alpha\mathbf{A} \preceq \alpha\mathbf{B}$ for any $\alpha \ge 0$.
- (Preserved by trace) $\operatorname{Tr}\mathbf{A} \le \operatorname{Tr}\mathbf{B}$.

Furthermore, recall that $\mathbf{W}_1 + \mathbf{W}_2 = 2n^{-1}\mathbf{X}^\mathsf{T}\mathbf{X}$. Therefore, the assumption $\alpha \le 1/(n^{-1}\|\mathbf{X}^\mathsf{T}\mathbf{X}\|)$ is simply the same as $\alpha \le 2/\|\mathbf{W}_1 + \mathbf{W}_2\|$ in different notation.

Proof of Proposition 4.3.8

The claim follows if we can show that $\mathbf{Z} \succ \mathbf{0}$ and $2\alpha\mathbf{Z} \prec 2\mathbf{I}$, assuming $\alpha\|\mathbf{W}_1 + \mathbf{W}_2\| < 2$.

- $\mathbf{Z} \succ \mathbf{0}$:[9] the key observation is that we can write

$$\begin{aligned}\mathbf{Z} &= \frac{1}{2}(\mathbf{W}_1 + \mathbf{W}_2) - \frac{1}{4}\alpha(\mathbf{W}_1 + \mathbf{W}_2)^2 + \frac{1}{4}\alpha(\mathbf{W}_1^2 + \mathbf{W}_2^2) \\ &= \frac{1}{2}(\mathbf{W}_1 + \mathbf{W}_2)\left[\mathbf{I} - \frac{1}{2}\alpha(\mathbf{W}_1 + \mathbf{W}_2)\right] + \frac{1}{4}\alpha(\mathbf{W}_1^2 + \mathbf{W}_2^2).\end{aligned}$$

  Since $\mathbf{W}_1, \mathbf{W}_2 \succeq \mathbf{0}$, we have $\mathbf{W}_1^2 + \mathbf{W}_2^2 \succeq \mathbf{0}$, and using the assumption $\frac{1}{2}\alpha(\mathbf{W}_1 + \mathbf{W}_2) \prec \mathbf{I}$, we deduce that the first term is also positive semidefinite. Hence, $\mathbf{Z} \succeq \mathbf{0}$.

- $2\alpha\mathbf{Z} \prec 2\mathbf{I}$: we can write

$$\begin{aligned}2\alpha\mathbf{Z} &= \alpha\mathbf{W}_1\left(\mathbf{I} - \frac{1}{2}\alpha\mathbf{W}_2\right) + \alpha\mathbf{W}_2\left(\mathbf{I} - \frac{1}{2}\alpha\mathbf{W}_1\right) \\ &= \alpha\left(\mathbf{W}_1 + \mathbf{W}_2\right)\left(2\mathbf{I} - \frac{1}{2}\alpha(\mathbf{W}_1 + \mathbf{W}_2)\right) - \alpha\mathbf{W}_1\left(\mathbf{I} - \frac{1}{2}\alpha\mathbf{W}_1\right) - \alpha\mathbf{W}_2\left(\mathbf{I} - \frac{1}{2}\alpha\mathbf{W}_2\right).\end{aligned}$$

[9] Even though $\mathbf{W}_1 + \mathbf{W}_2 \succeq \mathbf{0}$, this is not immediately obvious since the anticommutator $\mathbf{W}_1\mathbf{W}_2 + \mathbf{W}_2\mathbf{W}_1$ is not positive semidefinite in general.

Since $\alpha\mathbf{W}_1 \prec 2\mathbf{I}$ and $\alpha\mathbf{W}_2 \prec 2\mathbf{I}$ by assumption, we have $(\mathbf{I}-\frac{1}{2}\alpha\mathbf{W}_1) \succ \mathbf{0}$ and $(\mathbf{I}-\frac{1}{2}\alpha\mathbf{W}_2) \succ \mathbf{0}$. Thus,

$$2\alpha\mathbf{Z} \prec 2\alpha\left(\mathbf{W}_1+\mathbf{W}_2\right) - \frac{1}{2}\alpha^2\left(\mathbf{W}_1+\mathbf{W}_2\right)^2.$$

By considering the eigenvalues of $\alpha(\mathbf{W}_1+\mathbf{W}_2)$, which satisfy $\|\alpha(\mathbf{W}_1+\mathbf{W}_2)\| < 2$ by assumption, we deduce that the operator norm of the upper bound is at most 2. Hence, we conclude that $2\alpha\mathbf{Z} \prec 2\mathbf{I}$.

Proof of Proposition 4.3.11

Our goal is to bound the generalization error given in Theorem 4.3.9 (with $B = 2$) by bounding the trace term (corresponding to the variance component). The key observation is that in the two-batch case, we have the explicit relationship between $\frac{1}{n}\widetilde{\mathbf{X}}^\mathsf{T}\widetilde{\mathbf{X}}$ and $\mathbf{Z}$:

$$\frac{1}{n}\widetilde{\mathbf{X}}^\mathsf{T}\widetilde{\mathbf{X}} = \mathbf{Z} + \frac{\alpha}{4}\left[\left(\frac{1}{2}\alpha\mathbf{W}_1-\mathbf{I}\right)\mathbf{W}_2\mathbf{W}_1 + \left(\frac{1}{2}\alpha\mathbf{W}_2-\mathbf{I}\right)\mathbf{W}_1\mathbf{W}_2\right]. \tag{4.32}$$

By using the property $\mathbf{Z}^\dagger\mathbf{Z}\mathbf{Z}^\dagger = \mathbf{Z}^\dagger$ of the pseudoinverse, and the fact that the trace preserves the Loewner order, the claimed upper bound follows if we can show that

$$\frac{1}{4}\left[\left(\frac{1}{2}\alpha\mathbf{W}_1-\mathbf{I}\right)\mathbf{W}_2\mathbf{W}_1 + \left(\frac{1}{2}\alpha\mathbf{W}_2-\mathbf{I}\right)\mathbf{W}_1\mathbf{W}_2\right] \preceq \frac{1}{2}\|\mathbf{W}_1+\mathbf{W}_2\|\mathbf{Z}, \tag{4.33}$$

assuming that $\alpha\|\mathbf{W}_1+\mathbf{W}_2\| \le 2$. Since $\mathbf{Z} = \frac{1}{2}(\mathbf{W}_1+\mathbf{W}_2) - \frac{1}{4}\alpha(\mathbf{W}_2\mathbf{W}_1+\mathbf{W}_1\mathbf{W}_2)$, the claim (4.33) is equivalent to showing that

$$\begin{aligned}&\frac{\alpha}{8}\left(\mathbf{W}_2\mathbf{W}_1\mathbf{W}_2+\mathbf{W}_1\mathbf{W}_2\mathbf{W}_1\right) - \frac{1}{4}(\mathbf{W}_2\mathbf{W}_1+\mathbf{W}_1\mathbf{W}_2)\\ &\qquad\preceq \frac{1}{4}\|\mathbf{W}_1+\mathbf{W}_2\|(\mathbf{W}_1+\mathbf{W}_2) - \frac{\alpha}{8}\|\mathbf{W}_1+\mathbf{W}_2\|(\mathbf{W}_2\mathbf{W}_1+\mathbf{W}_1\mathbf{W}_2),\end{aligned}$$

or, by rearranging,

$$\begin{aligned}&\frac{\alpha}{8}\left\{(\mathbf{W}_2\mathbf{W}_1\mathbf{W}_2+\mathbf{W}_1\mathbf{W}_2\mathbf{W}_1)+\|\mathbf{W}_1+\mathbf{W}_2\|(\mathbf{W}_2\mathbf{W}_1+\mathbf{W}_1\mathbf{W}_2)\right\}\\&\quad\preceq\frac{1}{4}\left\{\|\mathbf{W}_1+\mathbf{W}_2\|(\mathbf{W}_1+\mathbf{W}_2)+(\mathbf{W}_2\mathbf{W}_1+\mathbf{W}_1\mathbf{W}_2)\right\}\end{aligned}\tag{4.34}$$

Since $\mathbf{W}_1 \preceq \|\mathbf{W}_1\|\mathbf{I} \preceq \|\mathbf{W}_1+\mathbf{W}_2\|\mathbf{I}$, and similarly $\mathbf{W}_2 \preceq \|\mathbf{W}_1+\mathbf{W}_2\|\mathbf{I}$, the left hand side of (4.34) is bounded from above in the Loewner order by

$$\frac{\alpha}{8}\|\mathbf{W}_1+\mathbf{W}_2\|\left\{(\mathbf{W}_1^2+\mathbf{W}_2^2)+(\mathbf{W}_2\mathbf{W}_1+\mathbf{W}_1\mathbf{W}_2)\right\}\preceq\frac{1}{4}(\mathbf{W}_1+\mathbf{W}_2)^2,$$

where we use the assumption $\alpha\|\mathbf{W}_1+\mathbf{W}_2\| \leq 2$ for the second inequality. Next, since $\|\mathbf{W}_1+\mathbf{W}_2\|(\mathbf{W}_1+\mathbf{W}_2) \succeq \mathbf{W}_1^2+\mathbf{W}_2^2$, the right hand side of (4.34) is bounded from below by

$$\frac{1}{4}\left\{(\mathbf{W}_1^2+\mathbf{W}_2^2)+(\mathbf{W}_2\mathbf{W}_1+\mathbf{W}_1\mathbf{W}_2)\right\}=\frac{1}{4}(\mathbf{W}_1+\mathbf{W}_2)^2.$$

Combining the preceding two displayed equations shows that (4.34) holds. If $\mathbf{W}_1=\mathbf{W}_2=c\mathbf{I}$ and $\alpha=2/c$ for some $c>0$, then it is also clear that (4.34) holds with equality.

Similarly as above, the lower bound follows if we can show that

$$\frac{1}{4}\left[\left(\frac{1}{2}\alpha\mathbf{W}_1-\mathbf{I}\right)\mathbf{W}_2\mathbf{W}_1+\left(\frac{1}{2}\alpha\mathbf{W}_2-\mathbf{I}\right)\mathbf{W}_1\mathbf{W}_2\right]\succeq-\frac{1}{2}\|\mathbf{W}_1+\mathbf{W}_2\|\mathbf{Z},\tag{4.35}$$

assuming that $\alpha\|\mathbf{W}_1+\mathbf{W}_2\| \leq 2$. This is equivalent to showing that

$$\begin{aligned}&\frac{\alpha}{8}(\mathbf{W}_2\mathbf{W}_1\mathbf{W}_2+\mathbf{W}_1\mathbf{W}_2\mathbf{W}_1)-\frac{1}{4}(\mathbf{W}_2\mathbf{W}_1+\mathbf{W}_1\mathbf{W}_2)\\&\quad\succeq-\frac{1}{4}\|\mathbf{W}_1+\mathbf{W}_2\|(\mathbf{W}_1+\mathbf{W}_2)+\frac{\alpha}{8}\|\mathbf{W}_1+\mathbf{W}_2\|(\mathbf{W}_2\mathbf{W}_1+\mathbf{W}_1\mathbf{W}_2).\end{aligned}$$

By rearranging and using the fact that $\mathbf{W}_2\mathbf{W}_1\mathbf{W}_2+\mathbf{W}_1\mathbf{W}_2\mathbf{W}_1 \succeq \mathbf{0}$, this is implied by

$$\frac{1}{4}\|\mathbf{W}_1+\mathbf{W}_2\|(\mathbf{W}_1+\mathbf{W}_2)\succeq\frac{1}{4}\left(\frac{\alpha}{2}\|\mathbf{W}_1+\mathbf{W}_2\|+1\right)(\mathbf{W}_2\mathbf{W}_1+\mathbf{W}_1\mathbf{W}_2).$$

By using the assumption $\alpha\|\mathbf{W}_1 + \mathbf{W}_2\| \leq 2$, and the fact that $\|\mathbf{W}_1 + \mathbf{W}_2\|(\mathbf{W}_1 + \mathbf{W}_2) \succeq (\mathbf{W}_1 + \mathbf{W}_2)^2$, this is further implied by

$$\frac{1}{4}(\mathbf{W}_1 + \mathbf{W}_2)^2 \succeq \frac{1}{2}(\mathbf{W}_2\mathbf{W}_1 + \mathbf{W}_1\mathbf{W}_2).$$

Since $(\mathbf{W}_1 + \mathbf{W}_2)^2 = \mathbf{W}_1^2 + \mathbf{W}_2^2 + \mathbf{W}_2\mathbf{W}_1 + \mathbf{W}_1\mathbf{W}_2$, this is equivalent to

$$\frac{1}{4}(\mathbf{W}_1 - \mathbf{W}_2)^2 = \frac{1}{4}(\mathbf{W}_1^2 + \mathbf{W}_2^2 - \mathbf{W}_2\mathbf{W}_1 - \mathbf{W}_1\mathbf{W}_2) \succeq \mathbf{0},$$

which is indeed true, and hence we conclude that the claim (4.35) holds.

Finally, the convergence of $\bar{\boldsymbol{\beta}}_k$ to $\bar{\boldsymbol{\beta}}_\infty$ follows from Corollary 4.3.5 (with $B = 2$), using the sufficient condition on the step size α given in Proposition 4.3.8. The resulting bound for the limiting generalization error, expressed in terms of $\mathbf{Z}$, is obtained from the fact that $(\mathbf{I} - 2\alpha\mathbf{Z})^k \to 0$.

4.6.3 Asymptotic analysis

In this section, we consider the asymptotics of $\mathbf{Z}(\alpha/B)$ as $n \to \infty$ with p fixed, and evaluate the impact on the error trajectory and generalization error of mini-batch gradient descent with random reshuffling as compared to full-batch gradient descent.

Proof of Proposition 4.3.13

As $n \to \infty$, each $\mathbf{W}_b$ tends to Σ almost surely. Therefore, by independence, we deduce that on a set of probability one, $\mathbf{W}_b \to \Sigma$ for all $b = 1, \ldots, B$. Since $\mathbf{Z}(\alpha) = \frac{1}{B}\sum_{b=1}^{B} \mathbf{W}_b\Pi_b$, it suffices to compute the limiting expression for a fixed $\mathbf{W}_b\Pi_b$ by symmetry. The starting

point is the expression for Π_b from (4.22):

$$\Pi_b = \mathbf{I} - \sum_{i=1}^{B-1} \left\{ \frac{(-1)^{i+1}}{(i+1)!} \cdot \alpha^i \sum_{\substack{\{b_1,\ldots,b_i\} \subseteq [B] \setminus b \\ b_1,\ldots,b_i \text{ distinct}}} \mathbf{W}_{b_1} \ldots \mathbf{W}_{b_i} \right\}.$$

Indeed, we simply have to count the number of terms for the internal summand for a fixed $i \in \{1, 2, \ldots, B-1\}$, which indicates the number of distinct sample covariances that appear. There are $i!\binom{B-1}{i}$ (ordered) ways to choose i distinct indices $\{b_1, \ldots, b_i\}$ from $[B] \setminus \{b\}$. For each such choice, the limit of $\mathbf{W}_{b_1} \ldots \mathbf{W}_{b_i}$ is Σ^i. Therefore, introducing an extra factor of Σ for $\mathbf{W}_b$ (which is not in any of the terms in Π_b), we have, as $n \to \infty$,

$$\mathbf{W}_b \Pi_b \to \Sigma - \sum_{i=1}^{B-1} (-1)^{i+1} \frac{\binom{B-1}{i}}{i+1} \alpha^i \Sigma^{i+1} = \Sigma \left\{ \mathbf{I} - \sum_{i=1}^{B-1} (-1)^{i+1} \frac{(B-1)!(B-1-i)!}{(i+1)!} \alpha^i \Sigma^i \right\}.$$

The proof is completed by replacing α with α/B to obtain the claimed expression for $\mathbf{Z}(\alpha/B)$.

Proof of Proposition 4.3.14

Recall from Theorem 4.3.3 that the error trajectory of mini-batch gradient descent with random reshuffling and step size α/B is given by the following (with $\mathbf{Z} \equiv \mathbf{Z}(\alpha/B)$:

$$\bar{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_* = (\mathbf{I} - \alpha \mathbf{Z})^k (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) + \frac{1}{n} \left[\mathbf{I} - (\mathbf{I} - \alpha \mathbf{Z})^k \right] \mathbf{Z}^\dagger \widetilde{\mathbf{X}}^\mathsf{T} \boldsymbol{\eta}.$$

By using the assumption that $\Pi_b \equiv \Pi = \mathbf{I} - p(\mathbf{W})$, we have

$$\mathbf{Z} = \frac{1}{n} \sum_{b=1}^{B} \Pi_b \mathbf{X}_b^\mathsf{T} \mathbf{X}_b = \Pi \left(\frac{1}{n} \sum_{b=1}^{B} \mathbf{X}_b^\mathsf{T} \mathbf{X}_b \right) = \Pi \mathbf{W}.$$

Since $\mathbf{Z}$ is symmetric, $\mathbf{Z} = \mathbf{W}\Pi = \mathbf{W}(\mathbf{I} - p(\mathbf{W}))$. Furthermore, $\widetilde{\mathbf{X}}^\mathsf{T}\boldsymbol{\eta} = \sum_{b=1}^B \Pi_b \mathbf{X}_b^\mathsf{T}\boldsymbol{\eta}_b = \Pi(\sum_{b=1}^B \mathbf{X}_b^\mathsf{T}\boldsymbol{\eta}_b) = \Pi\mathbf{X}^\mathsf{T}\boldsymbol{\eta}$. Therefore,

$$\mathbf{Z}^\dagger\widetilde{\mathbf{X}}^\mathsf{T}\boldsymbol{\eta} = \mathbf{W}^{-1}\Pi^{-1}\Pi\mathbf{X}^\mathsf{T}\boldsymbol{\eta} = \mathbf{W}^{-1}\mathbf{X}^\mathsf{T}\boldsymbol{\eta} = n\mathbf{X}^\dagger\boldsymbol{\eta}$$

Since $\mathbf{W} = \mathbf{V}\widehat{\mathbf{S}}\mathbf{V}^\mathsf{T}$, where $\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^\mathsf{T}$ is the SVD of $\mathbf{X}$ with $\mathbf{U} \in \mathbb{R}^{n\times n}$ and $\mathbf{V} \in \mathbb{R}^{p\times p}$ orthogonal and $\mathbf{S} \in \mathbb{R}^{n\times p}$ diagonal, and $\widehat{\mathbf{S}} := \frac{1}{n}\mathbf{S}^\mathsf{T}\mathbf{S} \in \mathbb{R}^{p\times p}$, we have

$$\mathbf{V}^\mathsf{T}(\bar{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_*) = [\mathbf{I} - \alpha\widehat{\mathbf{S}}(\mathbf{I} - p(\widehat{\mathbf{S}}))]^k\mathbf{V}^\mathsf{T}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) + \left[\mathbf{I} - [\mathbf{I} - \alpha\widehat{\mathbf{S}}(\mathbf{I} - p(\widehat{\mathbf{S}}))]^k\right]\mathbf{S}^\dagger\mathbf{U}^\mathsf{T}\boldsymbol{\eta}.$$

The point of expressing the error in the eigenbasis $\mathbf{V}$ is that the dynamics decouple (since $\widehat{\mathbf{S}}$ is diagonal). Therefore, for $i = 1, 2, \ldots, p$, the i^{th} coordinate of $\mathbf{V}^\mathsf{T}(\bar{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_*)$ satisfies

$$[\mathbf{V}^\mathsf{T}(\bar{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_*)]_i = [1 - \alpha\hat{\lambda}_i(1 - p(\hat{\lambda}_i))]^k[\mathbf{V}^\mathsf{T}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)]_i + \frac{1}{\hat{\lambda}_i}\left(1 - [1 - \alpha\hat{\lambda}_i(1 - p(\hat{\lambda}_i))]^k\right)[\mathbf{U}^\mathsf{T}\boldsymbol{\eta}]_i.$$

Next, we can apply Theorem 4.3.9 for the corresponding generalization error:

$$\begin{aligned} R_{\mathbf{X}}(\bar{\boldsymbol{\beta}}_k) &= (\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)^\mathsf{T}(\mathbf{I} - \alpha\mathbf{Z})^k\Sigma(\mathbf{I} - \alpha\mathbf{Z})^k(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*) \\ &\quad + \frac{\sigma^2}{n}\operatorname{Tr}\left(\left[\mathbf{I} - (\mathbf{I} - \alpha\mathbf{Z})^k\right]\Sigma\left[\mathbf{I} - (\mathbf{I} - \alpha\mathbf{Z})^k\right]\mathbf{Z}^\dagger\left(\frac{1}{n}\widetilde{\mathbf{X}}^\mathsf{T}\widetilde{\mathbf{X}}\right)\mathbf{Z}^\dagger\right). \end{aligned}$$

From the calculations above, we have $\mathbf{Z}^\dagger(\frac{1}{n}\widetilde{\mathbf{X}}^\mathsf{T}\widetilde{\mathbf{X}})\mathbf{Z}^\dagger = \mathbf{W}^{-1}(\frac{1}{n}\mathbf{X}^\mathsf{T}\mathbf{X})\mathbf{W}^{-1} = \mathbf{W}^{-1}$. Therefore, changing to the eigenbasis $\mathbf{V}$ again, we have

$$\begin{aligned} R_{\mathbf{X}}(\bar{\boldsymbol{\beta}}_k) &= (\mathbf{V}^\mathsf{T}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*))^\mathsf{T}[\mathbf{I} - \alpha\widehat{\mathbf{S}}(\mathbf{I} - p(\widehat{\mathbf{S}}))]^k\mathbf{V}^\mathsf{T}\Sigma\mathbf{V}[\mathbf{I} - \alpha\widehat{\mathbf{S}}(\mathbf{I} - p(\widehat{\mathbf{S}}))]^k(\mathbf{V}^\mathsf{T}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)) \\ &\quad + \frac{\sigma^2}{n}\operatorname{Tr}\left(\left[\mathbf{I} - [\mathbf{I} - \alpha\widehat{\mathbf{S}}(\mathbf{I} - p(\widehat{\mathbf{S}}))]^k\right]\mathbf{V}^\mathsf{T}\Sigma\mathbf{V}\left[\mathbf{I} - [\mathbf{I} - \alpha\widehat{\mathbf{S}}(\mathbf{I} - p(\widehat{\mathbf{S}}))]^k\right]\widehat{\mathbf{S}}^{-1}\right). \end{aligned}$$

By using the assumption that $\mathbf{V}^\mathsf{T}\Sigma\mathbf{V} = \Lambda$ (i.e., that Σ and $\mathbf{W}$ are simultaneously diagonalizable), then we have again obtained an expression that decouples since all the matrices

involved are diagonal, and we can write the result as

$$R_{\mathbf{X}}(\bar{\boldsymbol{\beta}}_k) = \sum_{i=1}^{p} \lambda_i [1 - \alpha\hat{\lambda}_i(1 - p(\hat{\lambda}_i))]^{2k} [\mathbf{V}^\mathsf{T}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}_*)]_i \\ + \frac{\sigma^2}{n} \sum_{i=1}^{p} \frac{\lambda_i}{\hat{\lambda}_i} \left(1 - [1 - \alpha\hat{\lambda}_i(1 - p(\hat{\lambda}_i))]^k\right)^2.$$

This completes the proof of the claimed expressions for mini-batch gradient descent.

Finally, it is straightforward to show that the corresponding quantities for full-batch gradient descent (under the same assumptions) are the same as the given expressions with $\hat{\lambda}_i(1 - p(\hat{\lambda}_i))$ replaced by $\hat{\lambda}_i$ using the same strategy and the usual expressions for the error dynamics of full-batch gradient descent (Lemmas 4.5.1 and 4.5.2).

4.7 Additional details on free probability computations

In this section, we provide a high-level overview of and details on the algorithm from [BMS17] for calculating the spectral distribution of a self-adjoint polynomial of free random variables.

4.7.1 Additional background

Techniques for computing the distribution of a sum or product of free random variables have been developed (e.g., see [MS17; RE08]). The reason that we could not apply these techniques is that while we could compute the distribution of $w_1 w_2$ or $w_2 w_1$ individually (where w_1, w_2 are free random variables), we cannot compute the distribution of $w_2 w_1 + w_1 w_2$ since the two summands are not freely independent.

More generally, the problem of describing the distribution of a *general polynomial of free random variables* in terms of its individual marginals—such as its density or smoothness properties—remains a difficult open problem, even in pure mathematics. Recent theoretical progress in [Ari+24] provides a general description of the atoms: in particular, [Ari+24, Theorem 1.3] implies that asymptotically, $\alpha\mathbf{Z}(\alpha/2)$ and $\mathbf{W}$ have the *same point mass* of

$(1-\gamma^{-1})$ at zero if and only if $\gamma > 1$ (i.e., in the overparameterized regime). To interpret this result, note that the point mass at zero effectively corresponds to the "dimension" of the frozen subspaces of weights for gradient descent; i.e., the rank of the projectors $\mathbf{P}_{\mathbf{Z},0}$ and $\mathbf{P}_0$ for mini-batch gradient descent with random reshuffling and full-batch gradient descent, respectively.

4.7.2 Algorithm

In this section, we describe our implementation of the general algorithm from [BMS17] for calculating the spectral distribution of the non-commutative polynomial

$$p(w_1, w_2) = w_1 + w_2 - \frac{1}{2}(w_2 w_1 + w_1 w_2)$$

of two freely independent Marchenko-Pastur distributions w_1, w_2 with ratio parameter γ and variance α. When $\gamma = 2\lim_{n,p\to\infty} p/n$, this corresponds to the limiting spectral distributions of the scaled sample covariances $\alpha\mathbf{W}_1$ and $\alpha\mathbf{W}_2$ of the two mini-batches in two-batch gradient descent with step size α. For the statement of the algorithm for computing the spectral distributions of general polynomials of free random variables as well as the technical details and proofs, we refer to [BMS17] (in particular, Theorems 4.1 and 2.2 of their paper).

First, we state some preliminaries on the Marchenko-Pastur distribution $\nu_{\gamma,\alpha}$ with ratio parameter γ and variance α. The *Stieltjes transform* of $\nu_{\gamma,\alpha}$ is given by

$$m_{\gamma,\alpha}(z) := \mathbb{E}_{Y\sim\nu_{\gamma,\alpha}}[(Y-z)^{-1}] = \frac{\alpha(1-\gamma) - z + \sqrt{(z-\alpha(\gamma+1))^2 - 4\gamma\alpha^2}}{2\alpha\gamma z} \tag{4.36}$$

for $z \in \mathbb{C}_+$, where $\mathbb{C}_+ = \{z \in \mathbb{C} : \mathrm{Im}(z) > 0\}$ is the complex upper half-plane, and the branch of the complex square root is chosen with positive imaginary part. The *Cauchy transform* is given by $G(z) = -m(z)$. The Stieltjes transform of a real-valued random variable (or equivalently its Cauchy transform) uniquely determines its distribution through the Stieltjes inversion theorem (e.g., [MS17, Theorem 6]).

The algorithm of [BMS17] computes the Cauchy transform G_p of $p(w_1, w_2)$, which uniquely determines its distribution, given the individual Cauchy transforms of w_1, w_2 by the following steps:

(1) Compute a *linearization* $\mathbf{L}_p(w_1, w_2)$ of the non-commutative polynomial $p(w_1, w_2) = w_1 + w_2 - \frac{1}{2}(w_2 w_1 + w_1 w_2)$ in the sense of [BMS17, Definition 3.1]: that is, we want to find

$$\mathbf{L}_p(w_1, w_2) = \begin{pmatrix} 0 & \mathbf{u}^\mathsf{T} \\ \mathbf{v} & \mathbf{Q} \end{pmatrix}$$

such that $p(w_1, w_2) = -\mathbf{u}^\mathsf{T}\mathbf{Q}^{-1}\mathbf{v}$, where $\mathbf{u}, \mathbf{v}$ are vectors with entries in $\mathbb{C}\langle w_1, w_2\rangle$, the algebra generated by w_1, w_2 over the field of complex numbers, and $\mathbf{Q}$ is a matrix with entries in $\mathbb{C}\langle w_1, w_2\rangle$. Specifically, we use

$$\mathbf{L}_p(w_1, w_2) = \begin{pmatrix} 0 & 1 & w_1 & w_2 \\ 1 & -1 & -1 & -1 \\ w_1 & -1 & -1 & 1 \\ w_2 & -1 & 1 & -1 \end{pmatrix}. \tag{4.37}$$

It may be easily checked that

$$\mathbf{Q}^{-1} = \begin{pmatrix} -1 & -1 & -1 \\ -1 & -1 & 1 \\ -1 & 1 & -1 \end{pmatrix}^{-1} = \frac{1}{2}\begin{pmatrix} 0 & -1 & -1 \\ -1 & 0 & 1 \\ -1 & 1 & 0 \end{pmatrix},$$

so that $w_1 + w_2 - \frac{1}{2}(w_2 w_1 + w_1 w_2) = -\mathbf{u}^\mathsf{T}\mathbf{Q}^{-1}\mathbf{v}$. We also define the matrices

$$\mathbf{b}_0 := \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & -1 & -1 & -1 \\ 0 & -1 & -1 & 1 \\ 0 & -1 & 1 & -1 \end{pmatrix}, \quad \mathbf{b}_1 := \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{b}_2 := \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix},$$

so that we can write $\mathbf{L}_p(w_1, w_2) = \mathbf{b}_0 \otimes 1 + \mathbf{b}_1 \otimes w_1 + \mathbf{b}_2 \otimes w_2$. Finally, $\mathbf{b}_1 \otimes w_1$ and $\mathbf{b}_2 \otimes w_2$ (i.e., matrices whose entries consist of w_1 and w_2, respectively) are freely independent operator-valued random variables.

(2) The *operator-valued Cauchy transform* $\mathbf{G}_{\mathbf{b}_1 \otimes w_1}(\mathbf{b})$ of $\mathbf{b}_1 \otimes w_1$ is defined by $\mathbf{G}_{\mathbf{b}_1 \otimes w_1}(\mathbf{b}) := \mathbb{E}\left[(\mathbf{b} - \mathbf{b}_1 \otimes w_1)^{-1}\right] = \int_{\mathbb{R}} (\mathbf{b} - t\mathbf{b}_j)^{-1} \mathrm{d}\nu_{\gamma,\alpha}(t)$ for complex-valued matrices $\mathbf{b}$ in the operator upper half-plane (i.e., whose imaginary part has only positive eigenvalues). By the Stieltjes inversion theorem, it can be calculated by the limiting formula

$$\mathbf{G}_{\mathbf{b}_1 \otimes w_1}(\mathbf{b}) = \lim_{\varepsilon \downarrow 0} \frac{-1}{\pi} \int_{\mathbb{R}} (\mathbf{b} - t\mathbf{b}_1)^{-1} \mathrm{Im}(G_{w_1}(t + i\varepsilon)) \, \mathrm{d}t,$$

where the integral is taken elementwise, and the (scalar-valued) Cauchy transform G_{w_1} for the distribution $\nu_{\gamma,\alpha}$ is (the negative) of (4.36) above. (In our implementation, we found that computing this integral with parameters $\varepsilon \sim 10^{-6}$ and $t \sim 100$ worked well; in particular, t does not need to be large since the matrices involved have bounded operator norm and the Marchenko-Pastur distribution has compact support.) Similarly, the operator-valued Cauchy transform $\mathbf{G}_{\mathbf{b}_2 \otimes w_2}$ of $\mathbf{b}_2 \otimes w_2$ can be computed in the same way with $\mathbf{b}_1, w_1$ replaced by $\mathbf{b}_2, w_2$.

(3) Let $f_{\mathbf{b}}$ be the map defined by

$$f_{\mathbf{b}}(\mathbf{a}) = \mathbf{h}_{\mathbf{b}_2 \otimes w_2}(\mathbf{h}_{\mathbf{b}_1 \otimes w_1}(\mathbf{a}) + \mathbf{b}) + \mathbf{b},$$

where $\mathbf{h}_{\mathbf{b}_1 \otimes w_1}(\mathbf{a}) = (\mathbf{G}_{\mathbf{b}_1 \otimes w_1}(\mathbf{a}))^{-1} - \mathbf{a}$ and $\mathbf{h}_{\mathbf{b}_2 \otimes w_2}(\mathbf{a}) = (\mathbf{G}_{\mathbf{b}_2 \otimes w_2}(\mathbf{a}))^{-1} - \mathbf{a}$ are the so-called "h-transforms" of $\mathbf{b}_1 \otimes w_1$ and $\mathbf{b}_2 \otimes w_2$, respectively.

The *operator-valued Cauchy transform of the sum* $\mathbf{b}_1 \otimes w_1 + \mathbf{b}_2 \otimes w_2$ satisfies $G_{\mathbf{b}_1 \otimes w_1 + \mathbf{b}_2 \otimes w_2}(\mathbf{b}) = G_{\mathbf{b}_1 \otimes w_1}(\omega(\mathbf{b}))$, where $\omega(\mathbf{b})$ is the *unique fixed point of the map* $f_{\mathbf{b}}$ [BMS17, Theorem 2.2]. (In our implementation, we compute $\omega(\mathbf{b})$ by iterating

$\omega_i = \mathbf{f}_{\mathbf{b}}(\omega_{i-1})$ until the maximum elementwise difference between the iterates ω_i does not exceed a specified tolerance parameter $\sim 10^{-6}$.)

Thus, the operator-valued Cauchy transform of $\mathbf{L}_p(w_1, w_2) = \mathbf{b}_0 \otimes 1 + \mathbf{b}_1 \otimes w_1 + \mathbf{b}_2 \otimes w_2$ can be computed by $\mathbf{G}_{\mathbf{L}_p}(\mathbf{b}) = \mathbf{G}_{\mathbf{b}_1 \otimes w_1 + \mathbf{b}_2 \otimes w_2}(\mathbf{b} - \mathbf{b}_0) = G_{\mathbf{b}_1 \otimes w_1}(\omega(\mathbf{b} - \mathbf{b}_0))$.

(4) Finally, the *scalar-valued Cauchy transform $G_p(z)$ of $p(w_1, w_2)$ can be extracted from the first entry of the operator-valued Cauchy transform $\mathbf{G}_{L_p}$ of $\mathbf{L}_p(w_1, w_2)$*, evaluated at a diagonal matrix $\Lambda_\varepsilon(z)$, as $\varepsilon \downarrow 0$ [BMS17, Corollary 3.6]:

$$G_p(z) = \lim_{\varepsilon \downarrow 0} [\mathbf{G}_{L_p}(\Lambda_\varepsilon(z))]_{1,1}, \quad \text{where} \quad \Lambda_\varepsilon(z) := \begin{pmatrix} z & & & \\ & i\varepsilon & & \\ & & \ddots & \\ & & & i\varepsilon \end{pmatrix}.$$

(In our implementation, we found that evaluating $\mathbf{G}_{L_p}(\Lambda_\varepsilon(z))$ with $\varepsilon \sim 10^{-6}$ worked well.)

Thus, the algorithm above allows us to compute the Cauchy transform G_p, which completely determines the distribution of $p(w_1, w_2)$. For example, using G_p, we can compute the density f_p of $p(w_1, w_2)$ at $x \in \mathbb{R}$ by

$$f_p(x) = \lim_{\varepsilon \downarrow 0} \frac{-1}{\pi} \mathrm{Im}(G_p(x + i\varepsilon)).$$

For example, see [CL22, Theorem 2.1]. Furthermore, we can compute the point mass $g_p(x)$ at $x \in \mathbb{R}$ (if there is one) by

$$g_p(x) = \lim_{\varepsilon \downarrow 0} i\varepsilon G_p(x + i\varepsilon).$$

4.8 Additional numerical experiments

In this section, we present some additional numerical experiments.

4.8.1 Full-batch diverges, mini-batch converges

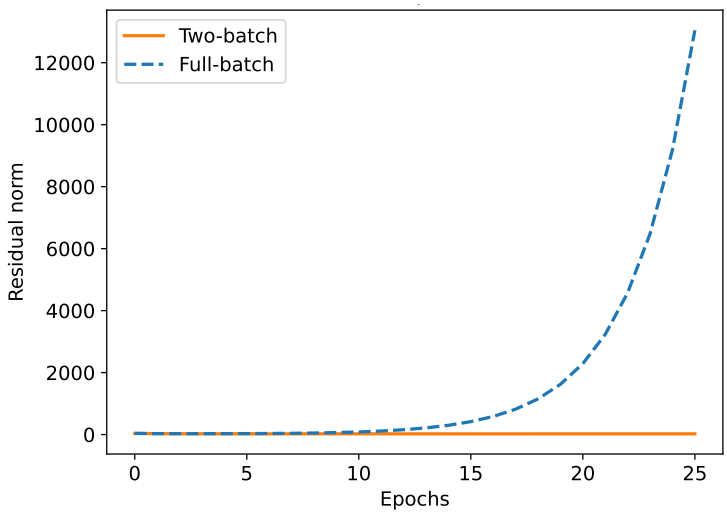


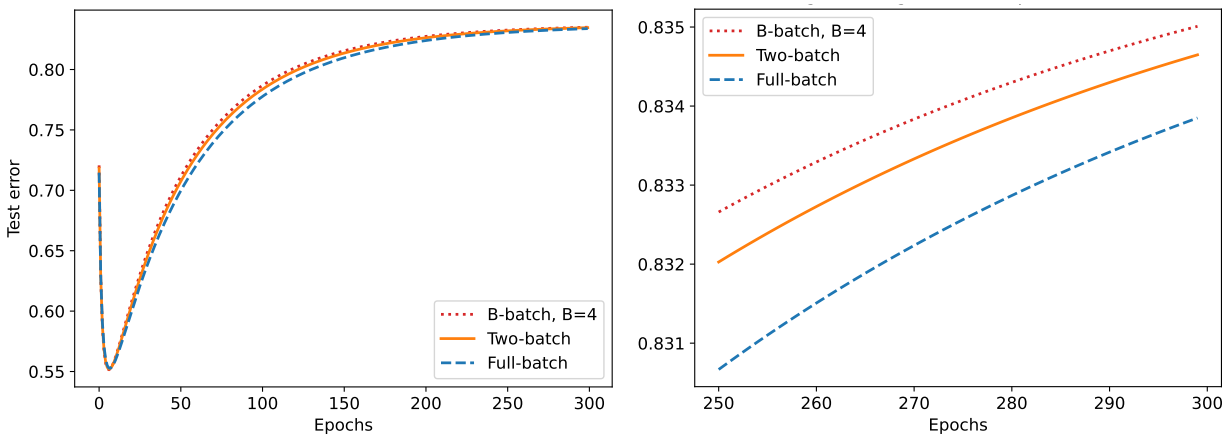
Figure 4.2: Training error dynamics where the entries of the data $\mathbf{X} \in \mathbb{R}^{n \times p}$ with $n = 1,000$ and $p = 1,500$, noise $\boldsymbol{\eta}$, and $\boldsymbol{\beta}_*$ are i.i.d. standard Gaussians, and a step size of $\alpha = 0.5$ is used. (This choice of α is slightly larger than $2/(1 + \sqrt{p/n})^2$, which is the almost sure limit of $\|\mathbf{X}\|$ by the Bai-Yin law.)

Consider the dynamics of gradient descent with step size α and two-batch gradient descent ($B = 2$) with step size $\alpha/2$. Then it is possible for the full-batch iterate to diverge (i.e., $\|\mathbf{I} - \alpha \frac{1}{n} \mathbf{X}^\mathsf{T} \mathbf{X}\| > 1$), while the two-batch iterate still converges (i.e., $\|\mathbf{I} - \alpha \mathbf{Z}\| > 1$).

4.8.2 Overparameterized regime

Figure 4.3 shows the generalization error dynamics of full-batch gradient descent with step size α and B-batch gradient descent with step size α/B for $B = 2, 4$ in the overparameterized regime. Overall, the difference is slight (according to the scaling of the figures), highlighting how the full-batch and mini-batch dynamics are matched using the linear scaling rule. Nonetheless, the difference is visually apparent during the middle of training.

For an extreme illustration of the differences that can be caused by large step sizes, recall that we discussed in Remark 4.3.7 (and demonstrated in Section 4.8.1) that with a larger choice of α, full-batch gradient descent can diverge, but two-batch gradient descent still converges.



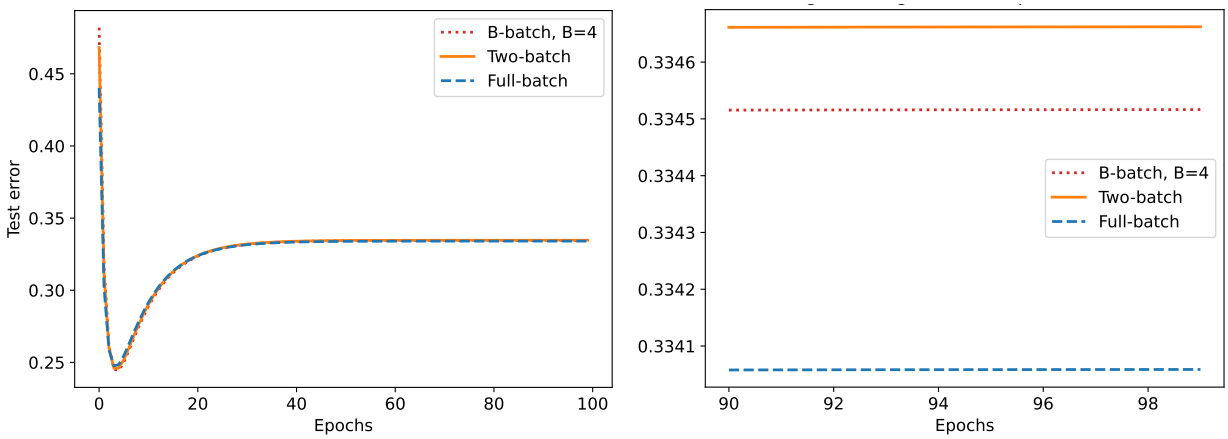
(a) Entire trajectory over 300 epochs.

(b) Limiting trajectory in the last 50 epochs.

Figure 4.3: Generalization error dynamics with standard Gaussian data $\mathbf{X} \in \mathbb{R}^{1,000 \times 1,500}$ ($\gamma = 3/2$), $\sigma = 0.5$, and $\boldsymbol{\beta}_*$ sampled uniformly at random from the unit sphere. Gradient descent with step size $\alpha = 0.2$ compared to B-batch gradient descent with step size α/B for $B = 2, 4$. The test error is averaged over $1,000$ simulations with $1,000$ test samples each.

4.8.3 Underparameterized regime

In Figure 4.4, we compare full-batch gradient descent and mini-batch gradient descent with $B = 2, 4$ mini-batches using the linear scaling rule for the step size in the underparameterized regime. Similar to the observations for Figure 4.3, the generalization error trajectories of mini-batch and full-batch gradient descent are closely matched. However there are very slight differences; for instance, the limiting risk of two-batch gradient descent is greater by about ~ 0.05.



(a) Entire trajectory over 100 epochs.

(b) Limiting trajectory in the last 10 epochs.

Figure 4.4: Generalization error dynamics with standard Gaussian data $\mathbf{X} \in \mathbb{R}^{4,000 \times 1,000}$ ($\gamma = 1/4$), $\sigma = 1$, and $\boldsymbol{\beta}_*$ sampled uniformly at random from the unit sphere. Gradient descent with step size $\alpha = 0.4$ compared to B-batch gradient descent with step size α/B for $B = 2, 4$. The test error is averaged over $1,000$ simulations with $1,000$ test samples each.

Chapter 5

Regularization via early stopping for linear least squares regression

This chapter is based on the following joint work with Rishi Sonthalia and Elizaveta Rebrova:

R. Sonthalia, J. Lok, and E. Rebrova. "On Regularization via Early Stopping for Least Squares Regression", 2024. Preprint, arXiv:2406.04425. arXiv: 2406.04425 [cs.LG]

5.1 Introduction

Early stopping is a common strategy used during training to regularize machine learning models, yet our understanding of the properties of early stopped models is far from complete. Recent work has shown that linear models trained with gradient descent can exhibit grokking, a phenomenon where the model initially overfits and generalizes poorly, only to later achieve better generalization after prolonged training [LBB24]. As a result, the strategy of early stopping naturally raises a range of important questions, including: (a) *What are the properties of early stopped models?* (b) *Under what circumstances is early stopping beneficial?* (c) *How do we decide when to stop training?*

Whether early stopping is beneficial or not, in terms of the out-of-sample generalization error, can depend on various parameters of the model. For example, for models trained by gradient descent, a crucial parameter is the choice of learning rates (or step sizes). Most prior research on early stopping for gradient descent considers constant step size schedules, frequently in the continuous-time gradient flow regime where the step sizes are assumed to be negligibly small [YRC07; RWY14]. However, the gradient flow approach does not explain how to quantify the optimal stopping time under more general and practical learning rate schedules.

Another known perspective views early stopping as inducing a form of ℓ^2 regularization. Intuitively, limiting the number of training iterations ensures that the obtained parameters remain relatively close to their initialization. Hence, it is commonly believed that with zero initialization, early stopping induces ℓ^2-type regularization, suggesting that the optimal stopping time should scale inversely with the minimum eigenvalue of the sample covariance matrix of the features [GBC16; HL22; RWY14]. However, formalizing this intuition has proven to be difficult.

Again, most prior works approach this by considering the continuous-time gradient flow dynamics for linear regression [SGB94], and impose Gaussian assumptions on the feature matrix. For example, by studying the exact generalization dynamics under gradient flow, Advani et al. [ASS20] provides an estimate for stopping after $O\big(\lambda^{-1}\log(1+\lambda)\big)$ iterations, where λ is a modal eigenvalue of the sample covariance of the feature matrix. Furthermore, [AKT19] shows that gradient flow and ridge regression are tightly connected—namely, under the correspondence $\mu = 1/t$ between the ridge regularization parameter μ and time parameter t, the relative risk is always between 1 and 1.6862 ([AKT19, Theorems 1 and 2]). The authors also derive a formula for the risk of gradient flow under the Marchenko–Pastur limit with arbitrary covariance [AKT19, Theorem 5]. Finally, [SGM22] provides a description of the optimal stopping time (up to constants) under the Gaussian model that is valid with high probability. However, without strong assumptions such as negligible learning rates and

Gaussianity, questions such as when to stop to optimize generalization error remain open [ASS20; AKT19; SGM22; Xu+23].

In this work, we focus on the *non-asymptotic discrete dynamics of gradient descent for least squares regression*. This point of view allows us to derive an explicit formula for the parameters of our model after k steps of gradient descent, denoted by $\boldsymbol{\beta}_k$, which can be expressed in a closed form for many different learning rate schedules used in practice. Using this, we obtain several novel, more precise answers to the questions above related to early stopping. Specifically,

1. **Exact trajectories.** We provide exact formulas for the discrete dynamics of $\boldsymbol{\beta}_k$ obtained when solving the ridge regression problem using gradient descent that make no assumptions on the data, learning rate schedule, or noise distribution (Proposition 5.2.2). We state simple expressions for many common learning rate schedules.

2. **Equivalence to (generalized) ridge regression.** For the ridgeless case, we show that for generic data, learning rate schedules, and stopping time T, the solution obtained after T iterations is equivalent to the minimum norm solution for a generalized ridge regression problem (Theorem 5.3.1). Additionally, we show that any solution to the ridge regression problem can also be obtained via early stopping if we can pick a distinct learning rate in each eigenspace of the sample covariance matrix of the features (Theorem 5.3.3).

3. **Sufficient conditions for early stopping to be beneficial**. We provide sufficient conditions (Theorems 5.4.4 and 5.4.8) for when early stopping improves the generalization performance under some general assumptions. Conversely, we also provide sufficient conditions for when early stopping is not beneficial (Theorems 5.4.6 and 5.4.8). As a corollary, we show that early stopping is beneficial for many common learning rate schedules, independent of the input data distribution, and that it is not beneficial for some other learning rates (Remarks 5.4.7 and 5.4.9).

4. **Optimal stopping time estimate for ill-conditioned covariance and non-constant step sizes.** We propose an estimate for the optimal stopping time for generic data and a large class of learning rate schedules: see (5.37) in Section 5.5. Our estimate generalizes a prior estimate from [ASS20] that only considered constant step size schedules and requires the covariance matrix of the features to be well-conditioned. We numerically verify the accuracy of our estimate on synthetic and real datasets.

5.1.1 Related work

The dynamics of gradient descent and its related variants have been the subject of extensive research in recent years. Without attempting to cover all the relevant literature, we will highlight below some key works and approaches most closely related to early stopping and our methodology.

Gradient flow dynamics. A common approach to understanding gradient-based methods is to study gradient flow, which assumes that the learning rates are negligibly small. Prior works such as [SGB94; ASS20; AKT19; LBB24; SGM22; BF25] have studied the gradient flow approximation of full-batch gradient descent. Other works have also studied the dynamics of stochastic gradient descent or flow [ADT20; Paq+25; Paq+22]. Here, we study the discrete dynamics of full-batch gradient descent instead of gradient flow, which, in particular, allows us to derive nontrivial results when non-constant step sizes are used.

Generalization error and regularization. In this work, we study the regularization effects of early stopping in terms of the out-of-sample generalization error. There has been a lot of recent work that uses tools from random matrix theory to understand the generalization performance of linear regression with other forms of regularization. Some works studying generalization for the ridge regression problem include [DW18; Has+22; KLS20; YH22; Nak+21; WSH24; Jac+20b; Jac+20a]. One of the surprising results from this line of research is that the optimal ridge parameter can be negative [YH22]. Other works such as [SN23;

KSS24; DL21; CZ23] study the effects of noise regularization. Recent work [SLG23] also studies the problem with both noise and ridge regularization. However, our focus is on the regularization effects of early stopping. Recent work by Stark and Steinerberger [SS25] shows that if the ridge regularization parameter is large enough, then early stopping for gradient descent with constant step sizes is not beneficial, which we have established independently for general step size schedules (see Theorem 5.4.8 and Remark 5.4.9). In addition, [SS25] elaborates on and proves optimality results for a fully data-driven methodology for estimating the optimal ridge regularization parameter.

Early stopping for statistical inverse problems. Early stopping of iterative procedures is also studied in the inverse problems literature [BHR18a; BHR18b; HR25; MR25], where regularization is crucial to recover a signal from ill-posed problems. In particular, early stopping for gradient descent, which is known as the Landweber iteration in this literature, is studied in [BHR18b]. The adaptive early stopping rules studied in this literature are based on the discrepancy principle [EHN96], which halts when the residual is of the same order as the noise level (which is assumed to be known or estimable a priori). Moreover, the quality of the solution is typically measured in terms of the distance to the underlying signal or the residual error (i.e., in-sample training error), in contrast to the generalization error on an unseen test sample that we focus on. In these works, the model is typically situated in the underparameterized setting, whereas our work allows for overparameterized models.

Other related works. Understanding the dynamics of gradient descent for linear models can provide insights into the dynamics of gradient descent for neural networks. Specifically, [Chi19] showed that in a *lazy training regime*, which holds under some circumstances such as with a large variance initialization, the trajectory of the parameters of a neural network is close to the trajectory of the parameters of a corresponding linearized model. Note that we can think of the linearized network as a kernel ridge regression problem with the neural tangent kernel [JGH18]. Recent work such as [Gei+20] seeks to understand the relationship

between feature learning and lazy training, and it is shown in [Kum+24] that grokking for neural networks occurs due to the transition from lazy training to rich training.

While our work focuses on studying gradient descent, studying the dynamics and regularization effects of early stopping for stochastic gradient descent (SGD) is an important related problem. Different learning rate schedules have been analyzed for SGD, and we will only mention a few works that do so. Learning rate schedules with linear decay and switching from constant to linear decay were studied in [Gow+19b]. In the streaming setting, SGD with square root decaying step sizes and oblivious noise was investigated in [PF20], and SGD with exponentially decaying step sizes and semi-adversarial noise was analyzed in [JNR25].

5.1.2 Problem setup and preliminaries

Suppose that we are given n training data points $(\mathbf{z}_i, y_i)$ drawn i.i.d. from a distribution $\mathcal{D}$, where $\mathbf{z}_i \in \mathbb{R}^d$ are the input vectors and $y_i \in \mathbb{R}$ are the responses. Let $\Psi : \mathbb{R}^d \to \mathbb{R}^p$ be a feature map and $\mathbf{x}_i = \Psi(\mathbf{z}_i)$. Let $\mathbf{X} \in \mathbb{R}^{n \times p}$ be the matrix containing the feature vectors $\mathbf{x}_i$ of the training data as rows, and $\mathbf{y} \in \mathbb{R}^n$ denote the response vector. Furthermore, we assume that there exists an underlying vector of parameters $\boldsymbol{\beta}_* \in \mathbb{R}^p$ such that $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}_* + \boldsymbol{\varepsilon}$, where $\boldsymbol{\varepsilon} \in \mathbb{R}^n$ is a residual or noise term with i.i.d. coordinates $\varepsilon_i = y_i - \mathbf{x}_i^\mathsf{T}\boldsymbol{\beta}_*$.

Given a regularization parameter $\mu \geq 0$, the *ridge regression problem* aims to minimize the following loss:

$$L_\mu(\boldsymbol{\beta}) = \frac{1}{2n}\|\mathbf{X}\boldsymbol{\beta} - \mathbf{y}\|_2^2 + \frac{\mu}{2}\|\boldsymbol{\beta}\|_2^2. \tag{5.1}$$

We solve the ridge regression problem (5.1) using gradient descent. Let $\boldsymbol{\beta}_0 \in \mathbb{R}^p$ be the chosen initialization and $\{\eta_k\}_{k \geq 1}$ denote the sequence of step sizes used in each iteration. If we denote the iterate after k iterations of gradient descent by $\boldsymbol{\beta}_k \in \mathbb{R}^p$, then the gradient descent update for the k^{th} iteration is given by

$$\boldsymbol{\beta}_k = \boldsymbol{\beta}_{k-1} - \frac{\eta_k}{n}\mathbf{X}^\mathsf{T}(\mathbf{X}\boldsymbol{\beta}_{k-1} - \mathbf{y}) - \eta_k \mu \boldsymbol{\beta}_{k-1}. \tag{5.2}$$

For any estimator $\boldsymbol{\beta} \in \mathbb{R}^p$, the *excess risk* with respect to $\boldsymbol{\beta}_*$ is given by

$$\mathcal{R}_{\mathbf{X}}(\boldsymbol{\beta}) := \mathbb{E}_{\mathbf{z}\sim\mathcal{D}_{\text{test}}}\left[\left\|\Psi(z)^\mathsf{T}\boldsymbol{\beta} - \Psi(z)^\mathsf{T}\boldsymbol{\beta}_*\right\|_2^2 \mid \mathbf{X}\right] = \|\boldsymbol{\beta} - \boldsymbol{\beta}_*\|_{\boldsymbol{\Sigma}}^2, \tag{5.3}$$

where $\boldsymbol{\Sigma} := \mathbb{E}_{\mathbf{z}\sim\mathcal{D}_{\text{test}}}[\Psi(\mathbf{z})\Psi(\mathbf{z})^\mathsf{T}]$ is the (uncentered) covariance matrix of the feature vectors $\mathbf{x} = \Psi(\mathbf{z})$, drawn according to some test distribution $\mathbf{z} \sim \mathcal{D}_{\text{test}}$, $\|\cdot\|_2$ denotes the Euclidean norm, $\|\mathbf{v}\|_{\boldsymbol{\Sigma}}^2 = \mathbf{v}^\mathsf{T}\boldsymbol{\Sigma}\mathbf{v}$, and the expectation is taken over a newly drawn test sample, conditional on the training features $\mathbf{X}$.

Finally, we introduce the following notation. Let $\widehat{\boldsymbol{\Sigma}} := n^{-1}\mathbf{X}^\mathsf{T}\mathbf{X} \in \mathbb{R}^{p\times p}$ be the sample (uncentered) covariance matrix of the features $\mathbf{X}$. Let $\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}_{\mathbf{X}}\mathbf{V}^\mathsf{T}$ and $\mathbf{X}^\mathsf{T}\mathbf{X} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^\mathsf{T}$ be the singular value decomposition and eigendecomposition of $\mathbf{X}$ and $\mathbf{X}^\mathsf{T}\mathbf{X}$, respectively, where $\mathbf{U} \in \mathbb{R}^{n\times n}$ and $\mathbf{V} \in \mathbb{R}^{p\times p}$ are orthogonal matrices, and $\boldsymbol{\Sigma}_{\mathbf{X}} \in \mathbb{R}^{n\times p}$ and $\boldsymbol{\Lambda} \in \mathbb{R}^{p\times p}$ are diagonal (which we may assume to be in non-increasing order). Note that $\boldsymbol{\Lambda} = \boldsymbol{\Sigma}_{\mathbf{X}}^\mathsf{T}\boldsymbol{\Sigma}_{\mathbf{X}} = n\mathbf{V}^\mathsf{T}\widehat{\boldsymbol{\Sigma}}\mathbf{V}$. We will also be interested in the representations of the parameters $\boldsymbol{\beta}_k$ and residual $\boldsymbol{\varepsilon}$ in the eigenbasis of $\mathbf{X}^\mathsf{T}\mathbf{X}$ (i.e., $\mathbf{V}$) and $\mathbf{X}\mathbf{X}^\mathsf{T}$ (i.e., $\mathbf{U}$), respectively, which we will denote by

$$\widetilde{\boldsymbol{\beta}}_k := \mathbf{V}^\mathsf{T}\boldsymbol{\beta}_k, \quad \widetilde{\boldsymbol{\beta}}_* := \mathbf{V}^\mathsf{T}\boldsymbol{\beta}_*, \quad \text{and} \quad \check{\boldsymbol{\varepsilon}} := \mathbf{U}^\mathsf{T}\boldsymbol{\varepsilon}. \tag{5.4}$$

We shall denote the identity matrix by $\mathbf{I}$. Given a matrix $\mathbf{A}$, we shall denote its Frobenius norm by $\|\mathbf{A}\|_F = \sqrt{\operatorname{Tr}(\mathbf{A}^\mathsf{T}\mathbf{A})}$ and its Moore-Penrose pseudoinverse by $\mathbf{A}^\dagger$.

Remark 5.1.1. We do not restrict the class of feature maps; for example, Ψ could represent a random feature model [RR07], the features learned by the final layer of a neural network, or the feature map for the neural tangent kernel [JGH18]. Moreover, we do not require that the distribution $\mathcal{D}_{\text{test}}$ for $\mathbf{x} = \Psi(\mathbf{z})$ at test time from (5.3) to be the same as the distribution $\mathcal{D}$ used to obtain the training data. This allows for *covariate shift*, which is an important problem that has been widely studied; see, e.g., [TAP21; KSS24].

5.2 Exact trajectories

To understand the regularization effects of early stopping and to determine when to stop, we begin by quantifying the trajectory of the parameters $\boldsymbol{\beta}_k$ and its associated excess risk. The aim of this section is to prove Proposition 5.2.2, which gives exact expressions for the dynamics of $\boldsymbol{\beta}_k$ and its representation $\widetilde{\boldsymbol{\beta}}_k$ in the eigenbasis V.

To build intuition, we first consider the unregularized problem ($\mu = 0$) with constant step size $\eta_k \equiv \eta$ and zero initialization $\boldsymbol{\beta}_0 = 0$. Simple calculations (e.g., [GBC16, Chapter 7.8]) show that we can recursively unravel the gradient descent dynamics from (5.2) to obtain

$$\boldsymbol{\beta}_k = \left(\mathbf{I} - \frac{\eta}{n}\mathbf{X}^\mathsf{T}\mathbf{X}\right)^k \boldsymbol{\beta}_0 + \sum_{i=1}^{k} \frac{\eta}{n}\left(\mathbf{I} - \frac{\eta}{n}\mathbf{X}^\mathsf{T}\mathbf{X}\right)^{k-i} \mathbf{X}^\mathsf{T}\mathbf{y}. \tag{5.5}$$

From (5.5), we can deduce the classical convergence result: if η is small enough, then the first term tends to zero as $k \to \infty$, and the second term converges to the minimum norm solution $(\mathbf{X}^\mathsf{T}\mathbf{X})^\dagger \mathbf{X}^\mathsf{T}\mathbf{y}$.

Similarly, if a sequence of non-negative step sizes $\{\eta_k\}_{k\geq 1}$ is used, then the above changes to

$$\boldsymbol{\beta}_k = \prod_{i=1}^{k}\left(\mathbf{I} - \frac{\eta_i}{n}\mathbf{X}^\mathsf{T}\mathbf{X}\right)\boldsymbol{\beta}_0 + \sum_{i=1}^{k}\frac{\eta_i}{n}\prod_{j=i+1}^{k}\left(\mathbf{I} - \frac{\eta_j}{n}\mathbf{X}^\mathsf{T}\mathbf{X}\right)\mathbf{X}^\mathsf{T}\mathbf{y}. \tag{5.6}$$

The general expression (5.6) involves an unfactored matrix polynomial with coefficients that depend on the step sizes. With constant step sizes, the dynamics are tractable because the polynomial expression can be simplified using the formula for the geometric series. One of the technical contributions of our work is a technique for factoring the polynomial above that works for any given step size schedule. With this goal in mind, we shall define the following function φ:

Definition 5.2.1. Given a learning rate schedule $\{\eta_i\}_{i\geq 1}$ and a real number ζ, define the function $\varphi(k;\zeta,\{\eta_i\})$ for positive integers k by

$$\varphi(k;\zeta,\{\eta_i\}) := \varphi(0;\zeta)\cdot\prod_{i=1}^{k}(1-\eta_i\zeta),$$

where $\varphi(0;\zeta)$ is a positive number that only depends on ζ and not on the learning rate schedule. Whenever it is clear from context, we shall suppress the dependence of φ on ζ and $\{\eta_i\}_{i\geq 1}$. Furthermore, given $\mu \geq 0$ and a diagonal matrix $\mathbf{\Lambda} = \mathrm{diag}(\{\Lambda_j\}_{j=1}^{p})$, we define $\mathbf{\Phi}(k;\mu,\mathbf{\Lambda}) \in \mathbb{R}^{p\times p}$ to be the diagonal matrix whose j^{th} diagonal entry is given by

$$\Phi(k;\mu,\mathbf{\Lambda})_j := \frac{\varphi(k;\mu+n^{-1}\Lambda_j)}{\varphi(0;\mu+n^{-1}\Lambda_j)}. \tag{5.7}$$

In matrix form, we can write

$$\mathbf{\Phi}(k;\mu,\mathbf{\Lambda}) = \prod_{i=1}^{k}\big(\mathbf{I}-\eta_i(\mu\mathbf{I}+n^{-1}\mathbf{\Lambda})\big). \tag{5.8}$$

Whenever it is clear from context, we shall also suppress the dependence of $\mathbf{\Phi}$ on μ and $\mathbf{\Lambda}$.

In particular, we will choose $\mathbf{\Lambda}$ to be the diagonal matrix with the eigenvalues of $\mathbf{X}^\mathsf{T}\mathbf{X}$. Therefore, $n^{-1}\Lambda_j$ is the j^{th} largest eigenvalue of the sample covariance matrix of the features, $\widehat{\mathbf{\Sigma}} = n^{-1}\mathbf{X}^\mathsf{T}\mathbf{X}$, and $\Phi(k;0,\mathbf{\Lambda})_j = 1$ if $j > \mathrm{rank}(\mathbf{X})$.

In Section 5.2.1, we will derive explicit expressions for the function φ for a variety of standard learning rates. The following result exactly characterizes the trajectory of $\boldsymbol{\beta}_k$, and is completely generic in that it makes no assumptions on the data, the learning rate schedule, or the initialization.

Proposition 5.2.2 (Trajectory)**.** *Let $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}_\mathbf{X}\mathbf{V}^\mathsf{T}$ be any feature matrix and $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}_* + \boldsymbol{\varepsilon}$. If $\boldsymbol{\beta}_k$ are the parameters after k steps of gradient descent for the ridge regression problem (5.1) with regularization parameter $\mu \geq 0$, initialized at $\boldsymbol{\beta}_0$ and with arbitrary learning rate schedule*

$\{\eta_k\}_{k\geq 1}$, *then we have*

$$\boldsymbol{\beta}_k = \mathbf{V}\boldsymbol{\Phi}(k;\mu)\mathbf{V}^\mathsf{T}\boldsymbol{\beta}_0 + (\mathbf{I} - \mathbf{V}\boldsymbol{\Phi}(k;\mu)\mathbf{V}^\mathsf{T})(\mu n\mathbf{I} + \mathbf{X}^\mathsf{T}\mathbf{X})^\dagger\mathbf{X}^\mathsf{T}\mathbf{y}. \tag{5.9}$$

Moreover, recalling that $\widetilde{\boldsymbol{\beta}}_k = \mathbf{V}^\mathsf{T}\boldsymbol{\beta}_k$, $\widetilde{\boldsymbol{\beta}}_* = \mathbf{V}^\mathsf{T}\boldsymbol{\beta}_*$, *and* $\check{\boldsymbol{\varepsilon}} = \mathbf{U}^\mathsf{T}\boldsymbol{\varepsilon}$, *we have*

$$\widetilde{\boldsymbol{\beta}}_k - \widetilde{\boldsymbol{\beta}}_* = \boldsymbol{\Phi}(k;\mu)(\widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_*) + (\mu n\mathbf{I} + \boldsymbol{\Lambda})^\dagger(\mathbf{I} - \boldsymbol{\Phi}(k;\mu))(\boldsymbol{\Sigma}_\mathbf{X}^\mathsf{T}\check{\boldsymbol{\varepsilon}} - \mu n\widetilde{\boldsymbol{\beta}}_*). \tag{5.10}$$

Remark 5.2.3 (Why is early stopping beneficial?)**.** Proposition 5.2.2 suggests that early stopping can be beneficial for a wide variety of learning rate schedules $\{\eta_k\}_{k\geq 1}$. Specifically, if the step size is sufficiently small and $\boldsymbol{\Phi}(k)$ is monotonically decreasing, then the error from learning the signal $\boldsymbol{\Phi}(k)(\widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_*)$ is monotonically decreasing, while the second term $(\mu n\mathbf{I}+\boldsymbol{\Lambda})^\dagger(\mathbf{I}-\boldsymbol{\Phi}(k;\mu))(\boldsymbol{\Sigma}_\mathbf{X}^\mathsf{T}\check{\boldsymbol{\varepsilon}}-\mu n\widetilde{\boldsymbol{\beta}}_*)$, which corresponds to fitting the noise, is monotonically increasing. Thus, stopping early may allow for the minimum to be achieved by balancing the two quantities. We make this intuition precise in Section 5.4 by providing conditions on the learning rates under which early stopping is beneficial (or not).

Remark 5.2.4 (Decoupling of the learning dynamics)**.** Equation (5.9) shows that $\boldsymbol{\beta}_k$ can be thought of as a convex combination of the initialization $\boldsymbol{\beta}_0$ and the minimum norm solution $(\mu n\mathbf{I} + \mathbf{X}^\mathsf{T}\mathbf{X})^\dagger\mathbf{X}^\mathsf{T}\mathbf{y}$ with weights given by $\mathbf{V}\boldsymbol{\Phi}(k)\mathbf{V}^\mathsf{T}$. Equation (5.10) also shows that the dynamics of $\boldsymbol{\beta}_k - \boldsymbol{\beta}_*$ decouple when the parameters are expressed in the eigenbasis $\mathbf{V}$: recalling that $\boldsymbol{\Phi}(k;\mu)$ is a diagonal matrix, it can be rearranged to

$$\widetilde{\boldsymbol{\beta}}_k - \widetilde{\boldsymbol{\beta}}_* = \boldsymbol{\Phi}(k;\mu)\left[(\widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_*) - (\mu n\mathbf{I} + \boldsymbol{\Lambda})^\dagger(\boldsymbol{\Sigma}_\mathbf{X}^\mathsf{T}\check{\boldsymbol{\varepsilon}} - \mu n\widetilde{\boldsymbol{\beta}}_*)\right] + (\mu n\mathbf{I} + \boldsymbol{\Lambda})^\dagger(\boldsymbol{\Sigma}_\mathbf{X}^\mathsf{T}\check{\boldsymbol{\varepsilon}} - \mu n\widetilde{\boldsymbol{\beta}}_*). \tag{5.11}$$

Thus, each coordinate of $\widetilde{\boldsymbol{\beta}}_k - \widetilde{\boldsymbol{\beta}}_*$ evolves independently of the others.

The proof of Proposition 5.2.2 follows from matrix computations. We present it below to illuminate the notation and for completeness.

Proof of Proposition 5.2.2. From (5.2), the gradient update is given by

$$\boldsymbol{\beta}_{k+1} = \boldsymbol{\beta}_k - \frac{\eta_{k+1}}{n}\mathbf{X}^\mathsf{T}(\mathbf{X}\boldsymbol{\beta}_k - \mathbf{X}\boldsymbol{\beta}_* - \boldsymbol{\varepsilon}) - \eta_{k+1}\mu\boldsymbol{\beta}_k.$$

Subtracting $\boldsymbol{\beta}_*$ from both sides, we obtain

$$\begin{aligned}
\boldsymbol{\beta}_{k+1} - \boldsymbol{\beta}_* &= (1-\eta_{k+1}\mu)(\boldsymbol{\beta}_k - \boldsymbol{\beta}_*) - \frac{\eta_{k+1}}{n}\mathbf{X}^\mathsf{T}\mathbf{X}(\boldsymbol{\beta}_k - \boldsymbol{\beta}_*) + \frac{\eta_{k+1}}{n}(\mathbf{X}^\mathsf{T}\boldsymbol{\varepsilon} - \mu n\boldsymbol{\beta}_*) \\
&= \left[(1-\eta_{k+1}\mu)\mathbf{I} - \frac{\eta_{k+1}}{n}\mathbf{X}^\mathsf{T}\mathbf{X}\right](\boldsymbol{\beta}_k - \boldsymbol{\beta}_*) + \frac{\eta_{k+1}}{n}\left(\mathbf{X}^\mathsf{T}\boldsymbol{\varepsilon} - \mu n\boldsymbol{\beta}_*\right) \\
&= \mathbf{V}\left[(1-\eta_{k+1}\mu)\mathbf{I} - \frac{\eta_{k+1}}{n}\boldsymbol{\Lambda}\right]\mathbf{V}^\mathsf{T}(\boldsymbol{\beta}_k - \boldsymbol{\beta}_*) + \frac{\eta_{k+1}}{n}\mathbf{V}\left(\boldsymbol{\Sigma}_\mathbf{X}^\mathsf{T}\mathbf{U}^\mathsf{T}\boldsymbol{\varepsilon} - \mu n\mathbf{V}^\mathsf{T}\boldsymbol{\beta}_*\right).
\end{aligned}$$

By multiplying both sides by $\mathbf{V}^\mathsf{T}$ to perform a change of basis and using the notations in (5.4), we have

$$\widetilde{\boldsymbol{\beta}}_{k+1} - \widetilde{\boldsymbol{\beta}}_* = \left[(1-\eta_{k+1}\mu)\mathbf{I} - \frac{\eta_{k+1}}{n}\boldsymbol{\Lambda}\right](\widetilde{\boldsymbol{\beta}}_k - \widetilde{\boldsymbol{\beta}}_*) + \frac{\eta_{k+1}}{n}\left(\boldsymbol{\Sigma}_\mathbf{X}^\mathsf{T}\check{\boldsymbol{\varepsilon}} - \mu n\widetilde{\boldsymbol{\beta}}_*\right).$$

We can rewrite this as the matrix equation

$$\begin{bmatrix} \widetilde{\boldsymbol{\beta}}_{k+1} - \widetilde{\boldsymbol{\beta}}_* \\ \boldsymbol{\Sigma}_\mathbf{X}^\mathsf{T}\check{\boldsymbol{\varepsilon}} - \mu n\widetilde{\boldsymbol{\beta}}_* \end{bmatrix} = \begin{bmatrix} [(1-\eta_{k+1}\mu)\mathbf{I} - \eta_{k+1}n^{-1}\boldsymbol{\Lambda}] & \eta_{k+1}n^{-1}\mathbf{I} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \widetilde{\boldsymbol{\beta}}_k - \widetilde{\boldsymbol{\beta}}_* \\ \boldsymbol{\Sigma}_\mathbf{X}^\mathsf{T}\check{\boldsymbol{\varepsilon}} - \mu n\widetilde{\boldsymbol{\beta}}_* \end{bmatrix}.$$

Note that the product of block upper triangular matrices is also block upper triangular. Thus, we have

$$\begin{aligned}
&\begin{bmatrix} \widetilde{\boldsymbol{\beta}}_{k+1} - \widetilde{\boldsymbol{\beta}}_* \\ \boldsymbol{\Sigma}_\mathbf{X}^\mathsf{T}\check{\boldsymbol{\varepsilon}} - \mu n\widetilde{\boldsymbol{\beta}}_* \end{bmatrix} = \prod_{i=1}^{k+1}\begin{bmatrix} [(1-\eta_i\mu)\mathbf{I} - \eta_i n^{-1}\boldsymbol{\Lambda}] & \eta_i n^{-1}\mathbf{I} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_* \\ \boldsymbol{\Sigma}_\mathbf{X}^\mathsf{T}\check{\boldsymbol{\varepsilon}} - \mu n\widetilde{\boldsymbol{\beta}}_* \end{bmatrix} \\
&= \begin{bmatrix} \prod_{i=1}^{k+1}[(1-\eta_i\mu)\mathbf{I} - \eta_i n^{-1}\boldsymbol{\Lambda}] & \sum_{i=1}^{k+1}\left(\prod_{j=i+1}^{k+1}[(1-\eta_j\mu)\mathbf{I} - \eta_j n^{-1}\boldsymbol{\Lambda}]\right)\eta_i n^{-1}\mathbf{I} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_* \\ \boldsymbol{\Sigma}_\mathbf{X}^\mathsf{T}\check{\boldsymbol{\varepsilon}} - \mu n\widetilde{\boldsymbol{\beta}}_* \end{bmatrix}.
\end{aligned}$$

By inserting the notation (5.8) from Definition 5.2.1, we have shown that

$$\begin{bmatrix} \widetilde{\boldsymbol{\beta}}_{k+1} - \widetilde{\boldsymbol{\beta}}_* \\ \boldsymbol{\Sigma}_{\mathbf{X}}^{\mathsf{T}} \check{\boldsymbol{\varepsilon}} - \mu n \widetilde{\boldsymbol{\beta}}_* \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Phi}(k+1; \mu, \boldsymbol{\Lambda}) & \sum_{i=1}^{k+1} \left(\prod_{j=i+1}^{k+1} [(1-\eta_j \mu)\mathbf{I} - \eta_j n^{-1} \boldsymbol{\Lambda}] \right) \eta_i n^{-1} \mathbf{I} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_* \\ \boldsymbol{\Sigma}_{\mathbf{X}}^{\mathsf{T}} \check{\boldsymbol{\varepsilon}} - \mu n \widetilde{\boldsymbol{\beta}}_* \end{bmatrix}.$$

We can rewrite the sum in the top right block as follows. For the ℓ^{th} diagonal entry, writing $a_\ell := \mu + n^{-1}\Lambda_\ell$ for brevity, we have

$$\begin{aligned} \frac{1}{n} \sum_{i=1}^{k+1} \eta_i \left(\prod_{j=i+1}^{k+1} [1 - \eta_j (\mu + n^{-1}\Lambda_\ell)] \right) &= \frac{1}{n} \sum_{i=1}^{k+1} \eta_i \left(\frac{\prod_{j=1}^{k+1} [1 - \eta_j (\mu + n^{-1}\Lambda_\ell)]}{\prod_{j=1}^{i} [1 - \eta_j (\mu + n^{-1}\Lambda_\ell)]} \right) \\ &= \frac{1}{n} \sum_{i=1}^{k+1} \eta_i \left(\frac{\varphi(k+1; a_\ell)}{\varphi(i; a_\ell)} \right). \end{aligned} \tag{5.12}$$

If $a_\ell \neq 0$, then we have the following identity, which is formally stated and proved in Lemma 5.2.5 below:

$$\frac{1}{n} \sum_{i=1}^{k+1} \eta_i \left(\frac{\varphi(k+1; a_\ell)}{\varphi(i; a_\ell)} \right) = \frac{1}{n} \frac{1}{a_\ell} \left(1 - \frac{\varphi(k+1; a_\ell)}{\varphi(0; a_\ell)} \right).$$

If $a_\ell = 0$, then $\mu = \Lambda_\ell = (\boldsymbol{\Sigma}_{\mathbf{X}}^{\mathsf{T}} \boldsymbol{\Sigma}_{\mathbf{X}})_{\ell,\ell} = 0$, and so the corresponding entry of the vector $(\boldsymbol{\Sigma}_{\mathbf{X}}^{\mathsf{T}} \check{\boldsymbol{\varepsilon}} - \mu n \widetilde{\boldsymbol{\beta}}_*)_\ell$ is also zero. Thus, we obtain

$$\begin{bmatrix} \widetilde{\boldsymbol{\beta}}_{k+1} - \widetilde{\boldsymbol{\beta}}_* \\ \boldsymbol{\Sigma}_{\mathbf{X}}^{\mathsf{T}} \check{\boldsymbol{\varepsilon}} - \mu n \widetilde{\boldsymbol{\beta}}_* \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Phi}(k+1; \mu, \boldsymbol{\Lambda}) & (\mu n \mathbf{I} + \boldsymbol{\Lambda})^{\dagger} (\mathbf{I} - \boldsymbol{\Phi}(k+1; \mu, \boldsymbol{\Lambda})) \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_* \\ \boldsymbol{\Sigma}_{\mathbf{X}}^{\mathsf{T}} \check{\boldsymbol{\varepsilon}} - \mu n \widetilde{\boldsymbol{\beta}}_* \end{bmatrix}.$$

This implies that (5.10) holds. Now, we will show how this implies (5.9). Since $\mathbf{\Lambda}$ and $\mathbf{\Phi}(k;\mu)$ are diagonal, we have

$$\begin{aligned}\widetilde{\boldsymbol{\beta}}_k &= \mathbf{\Phi}(k;\mu)(\widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_*) + (\mathbf{I} - \mathbf{\Phi}(k;\mu))(\mu n\mathbf{I} + \mathbf{\Lambda})^\dagger(\mathbf{\Sigma}_{\mathbf{X}}^\mathsf{T}\check{\boldsymbol{\varepsilon}} - \mu n\widetilde{\boldsymbol{\beta}}_*) + \widetilde{\boldsymbol{\beta}}_* \\ &= \mathbf{\Phi}(k;\mu)(\widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_*) + (\mathbf{I} - \mathbf{\Phi}(k;\mu))(\mu n\mathbf{I} + \mathbf{\Lambda})^\dagger(\mathbf{\Sigma}_{\mathbf{X}}^\mathsf{T}\check{\boldsymbol{\varepsilon}} + \mathbf{\Lambda}\widetilde{\boldsymbol{\beta}}_*) \\ &\qquad - (\mathbf{I} - \mathbf{\Phi}(k;\mu))(\mu n\mathbf{I} + \mathbf{\Lambda})^\dagger(\mu n\mathbf{I} + \mathbf{\Lambda})\widetilde{\boldsymbol{\beta}}_* + \widetilde{\boldsymbol{\beta}}_* \\ &= \mathbf{\Phi}(k;\mu)\widetilde{\boldsymbol{\beta}}_0 + (\mathbf{I} - \mathbf{\Phi}(k;\mu))(\mu n\mathbf{I} + \mathbf{\Lambda})^\dagger(\mathbf{\Sigma}_{\mathbf{X}}^\mathsf{T}\check{\boldsymbol{\varepsilon}} + \mathbf{\Lambda}\widetilde{\boldsymbol{\beta}}_*).\end{aligned}$$

The last step holds since $\mathbf{\Lambda}$ is a diagonal matrix whose leading $r \times r$ principal submatrix is non-zero, recalling that $\operatorname{rank}(\mathbf{X}) = r$, and $\mathbf{\Phi}(k;\mu)$ is diagonal with entries $\Phi(k;\mu)_j = 1$ for $j > r$, so that

$$(\mathbf{I} - \mathbf{\Phi}(k;\mu))(\mu n\mathbf{I} + \mathbf{\Lambda})^\dagger(\mu n\mathbf{I} + \mathbf{\Lambda}) = \mathbf{I} - \mathbf{\Phi}(k;\mu).$$

By multiplying by $\mathbf{V}$ to change back to the original basis, recalling that $\mathbf{V}\mathbf{\Lambda}\mathbf{V}^\mathsf{T} = \mathbf{X}^\mathsf{T}\mathbf{X}$, we have

$$\boldsymbol{\beta}_k = \mathbf{V}\mathbf{\Phi}(k;\mu)\mathbf{V}^\mathsf{T}\boldsymbol{\beta}_0 + (\mathbf{I} - \mathbf{V}\mathbf{\Phi}(k;\mu)\mathbf{V}^\mathsf{T})(\mu n\mathbf{I} + \mathbf{X}^\mathsf{T}\mathbf{X})^\dagger\mathbf{X}^\mathsf{T}(\mathbf{X}\boldsymbol{\beta}_* + \boldsymbol{\varepsilon}).$$

Since $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}_* + \varepsilon$, this implies (5.9), which completes the proof. □

Lemma 5.2.5 (Technical sum identity)**.** *For any non-zero real number ζ, we have*

$$\sum_{i=1}^{k} \eta_i \frac{1}{\varphi(i;\zeta)} = \frac{1}{\zeta}\left(\frac{1}{\varphi(k;\zeta)} - \frac{1}{\varphi(0;\zeta)}\right).$$

Proof of Lemma 5.2.5. For notational simplicity, we will write $\varphi(i) = \varphi(i;\zeta)$. We will prove the identity using induction. For the base case with $k = 1$, the left hand side is $\eta_1/\varphi(1)$, and the right hand side is equal to

$$\frac{1}{\zeta}\left(\frac{1}{\varphi(1)} - \frac{1}{\varphi(0)}\right) = \frac{1}{\zeta}\left(\frac{1 - (1 - \eta_1\zeta)}{\varphi(1)}\right) = \frac{\eta_1}{\varphi(1)}.$$

Hence, the base case holds. For the inductive step, assuming that the identity holds for some k, we can write

$$\sum_{i=1}^{k+1} \eta_i \frac{1}{\varphi(i)} = \frac{\eta_{k+1}}{\varphi(k+1)} + \sum_{i=1}^{k} \eta_i \frac{1}{\varphi(i)} = \frac{\eta_{k+1}}{\varphi(k+1)} + \frac{1}{\zeta}\left(\frac{1}{\varphi(k)} - \frac{1}{\varphi(0)}\right)$$
$$= \frac{\eta_{k+1}\zeta}{\varphi(k+1)\zeta} + \frac{(1-\eta_{k+1}\zeta)}{\varphi(k+1)\zeta} - \frac{1}{\zeta\varphi(0)} = \frac{1}{\zeta}\left(\frac{1}{\varphi(k+1)} - \frac{1}{\varphi(0)}\right).$$

This shows that the identity also holds for $k+1$. This completes the proof. □

5.2.1 Formulas for the function φ

For generic learning rate schedules $\{\eta_i\}_{i\geq 1}$, we may not have closed-form expressions for $\varphi(k;\zeta,\{\eta_i\})$. However, for many common learning rate schedules, φ can indeed be evaluated in terms of analytic functions. For *constant learning rate schedules* (i.e., $\eta_k \equiv \eta$), we can see that Definition 5.2.1 is satisfied by

$$\varphi(k;\zeta,\{\eta\}) = (1-\eta\zeta)^k \quad \text{and} \quad \varphi(0;\zeta) = 1.$$

The following two results give expressions for φ in terms of the Gamma function for *learning rates with polynomial decay* or *constant additive decay*.

Proposition 5.2.6 (Polynomial decay)**.** *If $\eta_k = \eta/k^m$ for some integer $m \geq 1$, then we have*

$$\varphi(k;\zeta) = \frac{1}{\Gamma(k+1)^m}\prod_{j=1}^{m}\Gamma\big(k+1-\omega_j(\eta\zeta)^{1/m}\big) \quad \textit{and} \quad \varphi(0;\zeta) = \prod_{j=1}^{m}\Gamma\big(1-\omega_j(\eta\zeta)^{1/m}\big), \tag{5.13}$$

where $\omega_1,\ldots,\omega_m$ are the m^{th} roots of unity, and $\Gamma(z) = \int_0^\infty t^{z-1}e^{-t}\,\mathrm{d}t$ is the Gamma function.

Proof. First, note that using the roots of unity, we have the following polynomial factorization:

$$1 - \frac{\eta}{k^m}\zeta = \prod_{j=1}^{m}\left(1 - \frac{\omega_j}{k}(\eta\zeta)^{1/m}\right).$$

Thus, given a learning rate schedule with polynomial decay with $\eta_k = \eta/k^m$, by using this identity and switching the order of the products, $\varphi(k;\zeta)$ and $\varphi(0;\zeta)$ must satisfy

$$\begin{aligned}\varphi(k;\zeta) &= \varphi(0;\zeta)\cdot\prod_{i=1}^{k}\left(1-\frac{\eta}{k^m}\zeta\right)\\ &= \varphi(0;\zeta)\cdot\prod_{i=1}^{k}\prod_{j=1}^{m}\left(1-\frac{\omega_j}{i}(\eta\zeta)^{1/m}\right) = \frac{\varphi(0;\zeta)}{\Gamma(k+1)^m}\cdot\prod_{j=1}^{m}\prod_{i=1}^{k}\left(i-\omega_j(\eta\zeta)^{1/m}\right).\end{aligned}$$

From the fundamental property $\Gamma(z+1) = z\Gamma(z)$ of the Gamma function, we have

$$\prod_{i=1}^{k}\left(i-\omega_j(\eta\zeta)^{1/m}\right) = \frac{\Gamma(k+1-\omega_j(\eta\zeta)^{1/m})}{\Gamma(1-\omega_j(\eta\zeta)^{1/m})}.$$

Thus, defining $\varphi(k;\zeta)$ and $\varphi(0;\zeta)$ as in (5.13) leads to a valid expression according to Definition 5.2.1. □

Proposition 5.2.7 (Constant additive decay)**.** *If $\eta_k = \eta_0 - k\eta$, then we have that*

$$\varphi(k;\zeta) = (\eta\zeta)^k\cdot\Gamma\left(k+1+\frac{1-\eta_0\zeta}{\eta\zeta}\right) \quad \textit{and} \quad \varphi(0;\zeta) = \Gamma\left(1+\frac{1-\eta_0\zeta}{\eta\zeta}\right), \tag{5.14}$$

where $\Gamma(z) = \int_0^\infty t^{z-1}e^{-t}\,\mathrm{d}t$ is the Gamma function.

Proof. The function $\varphi(k;\zeta)$ and $\varphi(0;\zeta)$ must satisfy

$$\varphi(k;\zeta) = \varphi(0;\zeta)\cdot\prod_{i=1}^{k}(1-\eta_0\zeta+i\eta\zeta) = \varphi(0;\zeta)\cdot(\eta\zeta)^k\prod_{i=1}^{k}\left(i+\frac{1-\eta_0\zeta}{\eta\zeta}\right).$$

Using the property $\Gamma(z+1) = z\Gamma(z)$ again verifies the validity of the claimed expressions. □

Finally, for *learning rates with exponential decay*, φ can be evaluated in terms of a function known as the *q-Pochhammer symbol*, which is defined by

$$(a;q)_n = \prod_{i=0}^{n-1}(1-aq^i). \tag{5.15}$$

From the definition of the q-Pochhammer symbol (5.15), we immediately obtain the following:

Proposition 5.2.8 (Exponential decay)**.** *If $\eta_k = \eta^k$, then we have that*

$$\varphi(k;\zeta) = (\zeta;\eta)_{k+1} \quad \textit{and} \quad \varphi(0;\zeta) = 1-\zeta. \tag{5.16}$$

We can also obtain a formula for composite learning schedules by multiplying the corresponding φ functions. For example, given a cyclic learning rate schedule with period T such that $\eta_{k+T} = \eta_k$, for any iteration $k = qT + r$, we have that

$$\varphi(k;\zeta) = \left(\frac{\varphi(T;\zeta)}{\varphi(0;\zeta)}\right)^q \cdot \varphi(r;\zeta).$$

In summary, we can obtain closed-form expressions for φ for many (but not all) learning rate schedules used in practice. One prominent learning rate schedule that we do not have a formula for is cosine annealing [LH17].

5.3 Early stopping and generalized ridge regularization

In this section, we show an equivalence between early stopping for the least squares problem (with $\mu = 0$) and generalized ridge regularization. Specifically, given a matrix $\mathbf{D} \in \mathbb{R}^{p\times p}$, the *generalized ridge regression problem* is the following:

$$\underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\arg\min}\, \frac{1}{2n}\|\mathbf{X}\boldsymbol{\beta} - \mathbf{y}\|_2^2 + \frac{1}{2}\|\mathbf{D}\boldsymbol{\beta}\|_2^2. \tag{5.17}$$

The generalized ridge regression parameter allows for a different regularization strength for each coordinate of $\widetilde{\boldsymbol{\beta}}_k$ in the eigenspaces of $\mathbf{X}^\mathsf{T}\mathbf{X}$. Recall that $\boldsymbol{\Phi}(T) \equiv \boldsymbol{\Phi}(T;0,\boldsymbol{\Lambda})$ is the diagonal matrix defined in (5.8). Using Proposition 5.2.2, we can show that the early stopped solution after T iterations solves a generalized ridge regression problem corresponding to some matrix $\mathbf{D}$ that depends on $\boldsymbol{\Phi}(T)$:

Theorem 5.3.1 (Early stopping $\implies$ generalized ridge regression)**.** *Let* $\mathbf{X}$ *be any feature matrix and* $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}_* + \boldsymbol{\varepsilon}$*. Suppose that* $\boldsymbol{\beta}_T$ *are the parameters after* T *steps of gradient descent for the least squares problem* (5.1) *with* $\mu = 0$*, initialized at* $\boldsymbol{\beta}_0 = \mathbf{0}$*, and with arbitrary learning rate schedule* $\{\eta_k\}_{k \geq 1}$*. Then, the early stopped solution* $\boldsymbol{\beta}_T$ *is the minimum norm solution to the generalized ridge regression problem* (5.17) *with*

$$\mathbf{D} = \left(\frac{1}{n} \boldsymbol{\Lambda} \boldsymbol{\Phi}(T) (\mathbf{I} - \boldsymbol{\Phi}(T))^{\dagger} \right)^{1/2} \mathbf{V}^{\mathsf{T}}.$$

Remark 5.3.2. Crucially, note that Theorem 5.3.1 makes no assumptions on the data or the learning rate schedule. In fact, if we consider large learning rates, then $\boldsymbol{\Phi}(T)$ could have negative entries! In this case, the entries of $\mathbf{D}$ would be complex. Prior work [YH22] has shown that the optimal ridge regularization parameter can be negative. Here, we see that this is analogous to training with $\mu = 0$ and large learning rates. A version of Theorem 5.3.1 was proved in [AKT19, Lemma 3] for constant step size schedules.

The main idea behind the proof is that we have a closed form expression for the minimum norm solution of the generalized ridge regression problem, which can be shown to coincide with the expression for $\boldsymbol{\beta}_T$ from Proposition 5.2.2.

Proof of Theorem 5.3.1. First, observe that the generalized ridge regression problem can be reformulated as an augmented least squares problem:

$$\frac{1}{2n} \|\mathbf{X}\boldsymbol{\beta} - \mathbf{y}\|_2^2 + \frac{1}{2} \|\mathbf{D}\boldsymbol{\beta}\|_2^2 = \frac{1}{2n} \left\| \begin{bmatrix} \mathbf{y} \\ \mathbf{0} \end{bmatrix} - \begin{bmatrix} \mathbf{X} \\ \sqrt{n}\mathbf{D} \end{bmatrix} \boldsymbol{\beta} \right\|_2^2.$$

The min-norm solution $\boldsymbol{\beta}^{(\mathbf{D})}$ to the generalized ridge regression problem (5.17) is given by

$$\begin{aligned}
\boldsymbol{\beta}^{(\mathbf{D})} &= \left(\begin{bmatrix} \mathbf{X}^\mathsf{T} & \sqrt{n}\mathbf{D}^\mathsf{T} \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ \sqrt{n}\mathbf{D} \end{bmatrix} \right)^\dagger \begin{bmatrix} \mathbf{X}^\mathsf{T} & \sqrt{n}\mathbf{D}^\mathsf{T} \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{0} \end{bmatrix} \\
&= \left(\mathbf{X}^\mathsf{T}\mathbf{X} + n\mathbf{D}^\mathsf{T}\mathbf{D}\right)^\dagger \mathbf{X}^\mathsf{T}\mathbf{y} \\
&= \mathbf{V}\left(\boldsymbol{\Lambda} + n\mathbf{V}^\mathsf{T}\mathbf{D}^\mathsf{T}\mathbf{D}\mathbf{V}\right)^\dagger \mathbf{V}^\mathsf{T}\mathbf{X}^\mathsf{T}(\mathbf{X}\boldsymbol{\beta}_* + \boldsymbol{\varepsilon}) \\
&= \mathbf{V}\left(\boldsymbol{\Lambda} + n\mathbf{V}^\mathsf{T}\mathbf{D}^\mathsf{T}\mathbf{D}\mathbf{V}\right)^\dagger \boldsymbol{\Lambda}\widetilde{\boldsymbol{\beta}_*} + \mathbf{V}\left(\boldsymbol{\Lambda} + n\mathbf{V}^\mathsf{T}\mathbf{D}^\mathsf{T}\mathbf{D}\mathbf{V}\right)^\dagger \boldsymbol{\Sigma}_\mathbf{X}^\mathsf{T}\check{\boldsymbol{\varepsilon}}.
\end{aligned}$$

Let $\widetilde{\boldsymbol{\beta}}^{(\mathbf{D})} := \mathbf{V}^\mathsf{T}\boldsymbol{\beta}^{(\mathbf{D})}$. By multiplying by $\mathbf{V}^\mathsf{T}$ and subtracting $\widetilde{\boldsymbol{\beta}}_*$ on both sides, we obtain

$$\widetilde{\boldsymbol{\beta}}^{(\mathbf{D})} - \widetilde{\boldsymbol{\beta}}_* = \left[\left(\boldsymbol{\Lambda} + n\mathbf{V}^\mathsf{T}\mathbf{D}^\mathsf{T}\mathbf{D}\mathbf{V}\right)^\dagger \boldsymbol{\Lambda} - \mathbf{I}\right] \widetilde{\boldsymbol{\beta}}_* + \left(\boldsymbol{\Lambda} + n\mathbf{V}^\mathsf{T}\mathbf{D}^\mathsf{T}\mathbf{D}\mathbf{V}\right)^\dagger \boldsymbol{\Sigma}_\mathbf{X}^\mathsf{T}\check{\boldsymbol{\varepsilon}}. \tag{5.18}$$

Next, substituting in the chosen value of $\mathbf{D}$, we have that

$$\boldsymbol{\Lambda} + n\mathbf{V}^\mathsf{T}\mathbf{D}^\mathsf{T}\mathbf{D}\mathbf{V} = \boldsymbol{\Lambda} + \boldsymbol{\Lambda}\boldsymbol{\Phi}(T)(\mathbf{I} - \boldsymbol{\Phi}(T))^\dagger.$$

To simplify this expression, note that all of the matrices are diagonal matrices. Recalling that $r = \operatorname{rank}(\mathbf{X})$, and writing $\mathbf{A}_{1:r,1:r}$ to denote the leading $r \times r$ principal submatrix of a matrix $\mathbf{A}$, we have

$$\boldsymbol{\Lambda} = \begin{bmatrix} \boldsymbol{\Lambda}_{1:r,1:r} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \quad \text{and} \quad \boldsymbol{\Phi}(T) = \begin{bmatrix} \boldsymbol{\Phi}(T)_{1:r,1:r} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}.$$

Thus, we see that

$$\begin{aligned}\boldsymbol{\Lambda} + n\mathbf{V}^\mathsf{T}\mathbf{D}^\mathsf{T}\mathbf{D}\mathbf{V} &= \boldsymbol{\Lambda} + \boldsymbol{\Lambda}\boldsymbol{\Phi}(T)(\mathbf{I} - \boldsymbol{\Phi}(T))^\dagger \\ &= \begin{bmatrix} \boldsymbol{\Lambda}_{1:r,1:r} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \left(\mathbf{I} + \begin{bmatrix} \boldsymbol{\Phi}(T)_{1:r,1:r} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} - \boldsymbol{\Phi}(T)_{1:r,1:r} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}^\dagger \right) \\ &= \begin{bmatrix} \boldsymbol{\Lambda}_{1:r,1:r} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{I} + \boldsymbol{\Phi}(T)_{1:r,1:r}(\mathbf{I} - \boldsymbol{\Phi}(T)_{1:r,1:r})^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}.\end{aligned}$$

Observe that for the leading principal $r \times r$ submatrix, we have

$$\begin{aligned}\mathbf{I} + \boldsymbol{\Phi}(T)_{1:r,1:r}(\mathbf{I} - \boldsymbol{\Phi}(T)_{1:r,1:r})^{-1} &= \left[(\mathbf{I} - \boldsymbol{\Phi}(T)_{1:r,1:r}) + \boldsymbol{\Phi}(T)_{1:r,1:r}\right](\mathbf{I} - \boldsymbol{\Phi}(T)_{1:r,1:r})^{-1} \\ &= (\mathbf{I} - \boldsymbol{\Phi}(T)_{1:r,1:r})^{-1}.\end{aligned}$$

Hence, we have shown that

$$\left(\boldsymbol{\Lambda} + n\mathbf{V}^\mathsf{T}\mathbf{D}^\mathsf{T}\mathbf{D}\mathbf{V}\right)^\dagger = \begin{bmatrix} (\mathbf{I} - \boldsymbol{\Phi}(T)_{1:r,1:r})\boldsymbol{\Lambda}_{1:r,1:r}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

Substituting this back into (5.18) shows that

$$\widetilde{\boldsymbol{\beta}}^{(\mathbf{D})} - \widetilde{\boldsymbol{\beta}}_* = \begin{bmatrix} \boldsymbol{\Phi}(T)_{1:r,1:r} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} (-\widetilde{\boldsymbol{\beta}}_*) + \boldsymbol{\Lambda}^\dagger(\mathbf{I} - \boldsymbol{\Phi}(T)_{1:r,1:r})\boldsymbol{\Sigma}_\mathbf{X}^\mathsf{T}\check{\boldsymbol{\varepsilon}}.$$

From Proposition 5.2.2, we see that with $\mu = 0$ and $\boldsymbol{\beta}_0 = \mathbf{0}$, the parameter estimate after T iterations of gradient descent also satisfies

$$\widetilde{\boldsymbol{\beta}}_T - \widetilde{\boldsymbol{\beta}}_* = \begin{bmatrix} \boldsymbol{\Phi}(T)_{1:r,1:r} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} (-\widetilde{\boldsymbol{\beta}}_*) + \boldsymbol{\Lambda}^\dagger(\mathbf{I} - \boldsymbol{\Phi}(T)_{1:r,1:r})\boldsymbol{\Sigma}_\mathbf{X}^\mathsf{T}\check{\boldsymbol{\varepsilon}},$$

which completes the proof. □

Next, we shall present a partial converse to Theorem 5.3.1. We show that for any regularization parameter μ, the minimum norm solution of the ridge regression problem (5.1) can be obtained via early stopping. However, similar to Theorem 5.3.1 where we had to regularize each component of $\widetilde{\boldsymbol{\beta}}_k$ in the eigenspaces of $\mathbf{X}^\mathsf{T}\mathbf{X}$ independently, we require a different (constant) learning rate for each direction.

Theorem 5.3.3 (Ridge regularization $\implies$ early stopping)**.** *Let* $\mathbf{X}$ *be any feature matrix and* $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}_* + \boldsymbol{\varepsilon}$*. For any regularization parameter* μ*, let* $\boldsymbol{\beta}^{(\mu)} = \left(\mu n I + \mathbf{X}^\mathsf{T}\mathbf{X}\right)^\dagger \mathbf{X}^\mathsf{T} y$ *be the minimum norm solution to the ridge regression problem* (5.1)*. Suppose that for each* j*, we choose*

$$\eta^{(j)} = \frac{n}{\mu n + \Lambda_j}$$

to be the learning rate for the j^{th} *coordinate of* $\widetilde{\boldsymbol{\beta}}$ *(if* $\mu n + \Lambda_j = 0$*, then we choose* $\eta^{(j)} = 0$*). Then, after one step of gradient descent for the unregularized least squares problem* (5.1)*, initialized at* $\boldsymbol{\beta}_0 = \mathbf{0}$*, we obtain* $\boldsymbol{\beta}^{(\mu)}$*.*

Proof. Recall that the dynamics of $\boldsymbol{\beta}_k - \boldsymbol{\beta}_*$ decouple when expressed in the eigenbasis $\mathbf{V}$ from (5.11). With $\boldsymbol{\beta}_0 = \mathbf{0}$, one step of gradient descent for the unregularized least squares

problem with step size $\eta^{(j)}$ for each coordinate of $\widetilde{\boldsymbol{\beta}}$ can be written as

$$\begin{aligned}
\widetilde{\boldsymbol{\beta}}_1 &= \frac{1}{n}\begin{bmatrix} \eta^{(1)} & 0 & \dots & & 0 \\ 0 & \eta^{(2)} & \dots & & 0 \\ \vdots & \vdots & \ddots & & \vdots \\ & & & \eta^{(p-1)} & 0 \\ 0 & 0 & \dots & 0 & \eta^{(p)} \end{bmatrix} \left[\boldsymbol{\Lambda}\widetilde{\boldsymbol{\beta}}_* + \boldsymbol{\Sigma}_{\mathbf{X}}^{\mathsf{T}}\check{\boldsymbol{\varepsilon}}\right] \\
&= (\mu n\mathbf{I} + \boldsymbol{\Lambda})^{\dagger}\left[\boldsymbol{\Lambda}\widetilde{\boldsymbol{\beta}}_* + \boldsymbol{\Sigma}_{\mathbf{X}}^{\mathsf{T}}\check{\boldsymbol{\varepsilon}}\right] \\
&= \mathbf{V}^{\mathsf{T}}\left(\mu n\mathbf{I} + \mathbf{V}\Lambda\mathbf{V}^{\mathsf{T}}\right)^{\dagger}\left[\mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^{\mathsf{T}}\boldsymbol{\beta}_* + \mathbf{X}^{\mathsf{T}}\boldsymbol{\varepsilon}\right] \\
&= \mathbf{V}^{\mathsf{T}}\left(\mu n\mathbf{I} + \mathbf{X}^{\mathsf{T}}\mathbf{X}\right)^{\dagger}\mathbf{X}^{\mathsf{T}}\left[\mathbf{X}\boldsymbol{\beta}_* + \boldsymbol{\varepsilon}\right] \\
&= \mathbf{V}^{\mathsf{T}}\left(\mu n\mathbf{I} + \mathbf{X}^{\mathsf{T}}\mathbf{X}\right)^{\dagger}\mathbf{X}^{\mathsf{T}}\mathbf{y}.
\end{aligned}$$

Multiplying both sides by $\mathbf{V}$ shows that $\boldsymbol{\beta}_1 = \left(\mu n\mathbf{I} + \mathbf{X}^{\mathsf{T}}\mathbf{X}\right)^{\dagger}\mathbf{X}^{\mathsf{T}}\mathbf{y} = \boldsymbol{\beta}^{(\mu)}$, as desired. □

Remark 5.3.4. A version of Theorem 5.3.3 can be presented given any learning rate schedule $\{\eta_i\}_{i\geq 1}$ such that $\varphi(k;\zeta,\{\eta_i\}) \to 0$ monotonically as $k \to \infty$ for any ζ. In this case, for each coordinate j of $\widetilde{\boldsymbol{\beta}}$, we would choose a separate stopping time T_j such that $\varphi(T_j; n^{-1}\Lambda_j, \{\eta_i\})$ is closest to $n/(\mu n + \Lambda_j)$. However, since we cannot guarantee equality for generic learning rate schedules $\{\eta_i\}_{i\geq 1}$, we can only obtain an approximation in this way.

5.4 Should we stop early?

In the previous section, we showed that early stopping acts like a form of ℓ^2 regularization. However, that does not tell us whether (a) regularization is beneficial for improving generalization; and (b) if it is beneficial, what the optimal stopping time is. In this section, we provide conditions for when early stopping is beneficial (or not).

5.4.1 Early stopped risk

To determine when to stop, we need to understand the dynamics of the *expected excess risk*

$$R_{\mathbf{X}}(\boldsymbol{\beta}_k) := \mathbb{E}_{\boldsymbol{\varepsilon}}[\mathcal{R}_{\mathbf{X}}(\boldsymbol{\beta}_k) \mid \mathbf{X}] = \mathbb{E}_{\boldsymbol{\varepsilon}}[\|\boldsymbol{\beta} - \boldsymbol{\beta}_*\|_{\boldsymbol{\Sigma}}^2 \mid \mathbf{X}] \tag{5.19}$$

during training, where the expectation is taken over the residual $\boldsymbol{\varepsilon} = \mathbf{y} - \mathbf{X}\boldsymbol{\beta}_*$ (which has i.i.d. coordinates $\varepsilon_i = y_i - \mathbf{x}_i^\mathsf{T}\boldsymbol{\beta}_*$), conditional on the feature matrix $\mathbf{X}$. To formulate our result, we shall impose the following assumption on the first and second moments of the residual $\boldsymbol{\varepsilon}$:

Assumption 1. The coordinates of $\boldsymbol{\varepsilon}$ satisfy $\mathbb{E}[\varepsilon_i \mid \mathbf{x}_i] = 0$ and $\mathbb{E}[\varepsilon_i^2 \mid \mathbf{x}_i] = \tau^2 < \infty$ for all $1 \le i \le n$.

Note that Assumption 1 is more general than the common assumption that the conditional distribution of the residual $\boldsymbol{\varepsilon}$ given $\mathbf{X}$ is subgaussian [Bar+20; Xu+23; Zou+23; RWY14].

Recall that $\boldsymbol{\Phi}(k;\mu)$ is the diagonal matrix defined in (5.8), and $\widetilde{\boldsymbol{\beta}}_k = \mathbf{V}^\mathsf{T}\boldsymbol{\beta}_k$ and $\widetilde{\boldsymbol{\beta}}_* = \mathbf{V}^\mathsf{T}\boldsymbol{\beta}_*$ from (5.4). We can leverage the formula for $\widetilde{\boldsymbol{\beta}}_k - \widetilde{\boldsymbol{\beta}}_*$ given in Proposition 5.2.2 to obtain the following exact formula for the expected excess risk $R_{\mathbf{X}}(\boldsymbol{\beta}_k)$:

Proposition 5.4.1 (Risk with ridge regularization)**.** *Let $\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}_{\mathbf{X}}\mathbf{V}^\mathsf{T}$ be any feature matrix and $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}_* + \boldsymbol{\varepsilon}$. Assume that Assumption 1 holds. If $\boldsymbol{\beta}_k$ are the parameters after k steps of gradient descent for the ridge regression problem (5.1) with ridge regularization parameter $\mu \ge 0$, initialized at any $\boldsymbol{\beta}_0$, and with arbitrary learning rate schedule $\{\eta_k\}_{k\ge 1}$, then*

$$\begin{aligned} R_{\mathbf{X}}(\boldsymbol{\beta}_k) = &\left\| \boldsymbol{\Sigma}^{1/2}\mathbf{V}\left[\boldsymbol{\Phi}(k;\mu)(\widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_*) - \mu n(\mu n\mathbf{I} + \boldsymbol{\Lambda})^\dagger(\mathbf{I} - \boldsymbol{\Phi}(k;\mu))\widetilde{\boldsymbol{\beta}}_*\right]\right\|_2^2 \\ &+ \tau^2 \left\| \boldsymbol{\Sigma}^{1/2}\mathbf{V}(\mu n\mathbf{I} + \boldsymbol{\Lambda})^\dagger(\mathbf{I} - \boldsymbol{\Phi}(k;\mu))\boldsymbol{\Lambda}^{1/2}\right\|_F^2. \end{aligned}$$

Proof. From Equation (5.10) of Proposition 5.2.2, we have that

$$\begin{aligned}\widetilde{\boldsymbol{\beta}}_k - \widetilde{\boldsymbol{\beta}}_* &= \boldsymbol{\Phi}(k;\mu)(\widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_*) + (\mu n\mathbf{I} + \boldsymbol{\Lambda})^\dagger(\mathbf{I} - \boldsymbol{\Phi}(k;\mu))(\boldsymbol{\Sigma}_{\mathbf{X}}^\mathsf{T}\check{\boldsymbol{\varepsilon}} - \mu n\widetilde{\boldsymbol{\beta}}_*) \\ &= \underbrace{\boldsymbol{\Phi}(k;\mu)(\widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_*) - \mu n(\mu n\mathbf{I} + \boldsymbol{\Lambda})^\dagger(\mathbf{I} - \boldsymbol{\Phi}(k;\mu))\widetilde{\boldsymbol{\beta}}_*}_{=:\mathbf{a}} + \underbrace{(\mu n\mathbf{I} + \boldsymbol{\Lambda})^\dagger(\mathbf{I} - \boldsymbol{\Phi}(k;\mu))\boldsymbol{\Sigma}_{\mathbf{X}}^\mathsf{T}\mathbf{U}^\mathsf{T}\boldsymbol{\varepsilon}}_{=:\mathbf{b}}.\end{aligned} \tag{5.20}$$

Our goal is to compute the expected excess risk:

$$R_{\mathbf{X}}(\boldsymbol{\beta}_k) = \mathbb{E}_{\boldsymbol{\varepsilon}}[\mathcal{R}_{\mathbf{X}}(\boldsymbol{\beta}_k) \mid \mathbf{X}] = \mathbb{E}_{\boldsymbol{\varepsilon}}\left[\left\|\boldsymbol{\Sigma}^{1/2}(\boldsymbol{\beta}_k - \boldsymbol{\beta}_*)\right\|_2^2 \mid \mathbf{X}\right] = \mathbb{E}_{\boldsymbol{\varepsilon}}\left[\left\|\boldsymbol{\Sigma}^{1/2}\mathbf{V}\left(\widetilde{\boldsymbol{\beta}}_k - \widetilde{\boldsymbol{\beta}}_*\right)\right\|_2^2 \mid \mathbf{X}\right].$$

By substituting in (5.20) and expanding the square, we have

$$\begin{aligned}\mathbb{E}_{\boldsymbol{\varepsilon}}\left[\left\|\boldsymbol{\Sigma}^{1/2}\mathbf{V}(\widetilde{\boldsymbol{\beta}}_k - \widetilde{\boldsymbol{\beta}}_*)\right\|_2^2 \mid \mathbf{X}\right] &= \mathbb{E}_{\boldsymbol{\varepsilon}}\left[\left\|\boldsymbol{\Sigma}^{1/2}\mathbf{V}\mathbf{a}\right\|_2^2 \mid \mathbf{X}\right] + \mathbb{E}_{\boldsymbol{\varepsilon}}\left[\left\|\boldsymbol{\Sigma}^{1/2}\mathbf{V}\mathbf{b}\right\|_2^2 \mid \mathbf{X}\right] \\ &\quad + 2\mathbb{E}_{\boldsymbol{\varepsilon}}\left[\mathbf{a}^\mathsf{T}\mathbf{V}^\mathsf{T}\boldsymbol{\Sigma}\mathbf{V}\mathbf{b} \mid \mathbf{X}\right].\end{aligned} \tag{5.21}$$

Recall that $\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}_{\mathbf{X}}\mathbf{V}^\mathsf{T}$, and that $\boldsymbol{\Lambda} = \boldsymbol{\Sigma}_{\mathbf{X}}^\mathsf{T}\boldsymbol{\Sigma}_{\mathbf{X}}$ is diagonal. Thus, $\mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^\mathsf{T} = \mathbf{X}^\mathsf{T}\mathbf{X}$, and from (5.8),

$$\mathbf{V}\boldsymbol{\Phi}(k;\mu)\mathbf{V}^\mathsf{T} = \prod_{i=1}^{k}(\mathbf{I} - \eta_i(\mu\mathbf{I} + n^{-1}\mathbf{X}^\mathsf{T}\mathbf{X})).$$

The significance of this observation is that $\mathbf{V}\boldsymbol{\Phi}(k;\mu)\mathbf{V}^\mathsf{T}$ is a function of the feature matrix $\mathbf{X}$. Hence, using the fact that $\mathbf{V}$ is orthogonal (i.e., $\mathbf{V}^\mathsf{T}\mathbf{V} = \mathbf{V}\mathbf{V}^\mathsf{T} = \mathbf{I}$), we can write

$$\begin{aligned}\mathbf{V}\mathbf{a} &= [\mathbf{V}\boldsymbol{\Phi}(k;\mu)\mathbf{V}^\mathsf{T}](\boldsymbol{\beta}_* - \boldsymbol{\beta}_0) - \mu n[\mathbf{V}(\mu n\mathbf{I} + \boldsymbol{\Lambda})^\dagger\mathbf{V}^\mathsf{T}][\mathbf{V}(\mathbf{I} - \boldsymbol{\Phi}(k;\mu))\mathbf{V}^\mathsf{T}]\boldsymbol{\beta}_* \\ &= [\mathbf{V}\boldsymbol{\Phi}(k;\mu)\mathbf{V}^\mathsf{T}](\boldsymbol{\beta}_* - \boldsymbol{\beta}_0) - \mu n(\mu n\mathbf{I} + \mathbf{X}^\mathsf{T}\mathbf{X})^\dagger(\mathbf{I} - \mathbf{V}^\mathsf{T}\boldsymbol{\Phi}(k;\mu)\mathbf{V})\boldsymbol{\beta}_*\end{aligned} \tag{5.22}$$

Hence, for the first term of (5.21), we have $\mathbb{E}_{\boldsymbol{\varepsilon}}\left[\|\boldsymbol{\Sigma}^{1/2}\mathbf{V}\mathbf{a}\|_2^2 \mid \mathbf{X}\right] = \|\boldsymbol{\Sigma}^{1/2}\mathbf{V}\mathbf{a}\|_2^2$ since $\boldsymbol{\Sigma}^{1/2}\mathbf{V}\mathbf{a}$ is a function of $\mathbf{X}$. Similarly, to compute the second term of (5.21), we can write

$$\begin{aligned}\mathbf{V}\mathbf{b} &= [\mathbf{V}(\mu n\mathbf{I} + \boldsymbol{\Lambda})^{\dagger}\mathbf{V}^{\mathsf{T}}][\mathbf{V}(\mathbf{I} - \boldsymbol{\Phi}(k;\mu))\mathbf{V}^{\mathsf{T}}][\mathbf{V}\boldsymbol{\Sigma}_{\mathbf{X}}^{\mathsf{T}}\mathbf{U}^{\mathsf{T}}]\boldsymbol{\varepsilon} \\ &= (\mu n\mathbf{I} + \mathbf{X}^{\mathsf{T}}\mathbf{X})^{\dagger}(\mathbf{I} - \mathbf{V}\boldsymbol{\Phi}(k;\mu)\mathbf{V}^{\mathsf{T}})\mathbf{X}^{\mathsf{T}}\boldsymbol{\varepsilon}. \end{aligned} \tag{5.23}$$

Also,

$$\mathbf{V}\mathbf{b}\mathbf{b}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}} = (\mu n\mathbf{I}+\mathbf{X}^{\mathsf{T}}\mathbf{X})^{\dagger}(\mathbf{I}-\mathbf{V}\boldsymbol{\Phi}(k;\mu)\mathbf{V}^{\mathsf{T}})\mathbf{X}^{\mathsf{T}}\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^{\mathsf{T}}\mathbf{X}(\mathbf{I}-\mathbf{V}\boldsymbol{\Phi}(k;\mu)\mathbf{V}^{\mathsf{T}})(\mu n\mathbf{I}+\mathbf{X}^{\mathsf{T}}\mathbf{X})^{\dagger}. \tag{5.24}$$

Since the residual has covariance matrix $\mathbb{E}_{\boldsymbol{\varepsilon}}[\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^{\mathsf{T}} \mid \mathbf{X}] = \tau^2\mathbf{I}$ by Assumption 1, it follows from using the linearity of expectation and (5.24) that

$$\begin{aligned}&\mathbb{E}[\mathrm{Tr}(\boldsymbol{\Sigma}\mathbf{V}\mathbf{b}\mathbf{b}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}) \mid \mathbf{X}] \\ &\quad= \mathrm{Tr}(\boldsymbol{\Sigma}(\mu n\mathbf{I} + \mathbf{X}^{\mathsf{T}}\mathbf{X})^{\dagger}(\mathbf{I} - \mathbf{V}\boldsymbol{\Phi}(k;\mu)\mathbf{V}^{\mathsf{T}})\mathbf{X}^{\mathsf{T}}\mathbb{E}\left[\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^{\mathsf{T}} \mid \mathbf{X}\right]\mathbf{X}(\mathbf{I} - \mathbf{V}\boldsymbol{\Phi}(k;\mu)\mathbf{V}^{\mathsf{T}})(\mu n\mathbf{I} + \mathbf{X}^{\mathsf{T}}\mathbf{X})^{\dagger}) \\ &\quad= \tau^2\,\mathrm{Tr}(\boldsymbol{\Sigma}(\mu n\mathbf{I} + \mathbf{X}^{\mathsf{T}}\mathbf{X})^{\dagger}(\mathbf{I} - \mathbf{V}\boldsymbol{\Phi}(k;\mu)\mathbf{V}^{\mathsf{T}})\mathbf{X}^{\mathsf{T}}\mathbf{X}(\mathbf{I} - \mathbf{V}\boldsymbol{\Phi}(k;\mu)\mathbf{V}^{\mathsf{T}})(\mu n\mathbf{I} + \mathbf{X}^{\mathsf{T}}\mathbf{X})^{\dagger}) \\ &\quad= \tau^2\,\mathrm{Tr}(\boldsymbol{\Sigma}\mathbf{V}(\mu n\mathbf{I} + \boldsymbol{\Lambda})^{\dagger}(\mathbf{I} - \boldsymbol{\Phi}(k;\mu))\boldsymbol{\Lambda}(\mathbf{I} - \boldsymbol{\Phi}(k;\mu))(\mu n\mathbf{I} + \boldsymbol{\Lambda})^{\dagger}\mathbf{V}^{\mathsf{T}}).\end{aligned}$$

Hence, by using the cyclic property of trace, the second term of (5.21) is equal to

$$\begin{aligned}&\mathbb{E}_{\boldsymbol{\varepsilon}}[\|\boldsymbol{\Sigma}^{1/2}\mathbf{V}\mathbf{b}\|_2^2 \mid \mathbf{X}] \\ &\quad= \mathbb{E}_{\boldsymbol{\varepsilon}}[\mathrm{Tr}(\mathbf{b}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\boldsymbol{\Sigma}\mathbf{V}\mathbf{b}) \mid \mathbf{X}] = \mathbb{E}_{\boldsymbol{\varepsilon}}[\mathrm{Tr}(\boldsymbol{\Sigma}\mathbf{V}\mathbf{b}\mathbf{b}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}) \mid \mathbf{X}] \\ &\quad= \tau^2\,\mathrm{Tr}(\boldsymbol{\Sigma}^{1/2}\mathbf{V}(\mu n\mathbf{I} + \boldsymbol{\Lambda})^{\dagger}(\mathbf{I} - \boldsymbol{\Phi}(k;\mu))\boldsymbol{\Sigma}_{\mathbf{X}}^{\mathsf{T}}\boldsymbol{\Sigma}_{\mathbf{X}}(\mathbf{I} - \boldsymbol{\Phi}(k;\mu))(\mu n\mathbf{I} + \boldsymbol{\Lambda})^{\dagger}\mathbf{V}^{\mathsf{T}}\boldsymbol{\Sigma}^{1/2}) \\ &\quad= \tau^2\left\|\boldsymbol{\Sigma}^{1/2}\mathbf{V}(\mu n\mathbf{I} + \boldsymbol{\Lambda})^{\dagger}(\mathbf{I} - \boldsymbol{\Phi}(k;\mu))\boldsymbol{\Sigma}_{\mathbf{X}}^{\mathsf{T}}\right\|_F^2.\end{aligned}$$

Finally, for the third term of (5.21), using the observation (5.22) that $\mathbf{Va}$ is a function of $\mathbf{X}$, and the expression (5.23) for $\mathbf{Vb}$, we have

$$\begin{aligned}\mathbb{E}_{\boldsymbol{\varepsilon}}\left[\mathbf{a}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\boldsymbol{\Sigma}\mathbf{V}\mathbf{b} \mid \mathbf{X}\right] &= \mathbb{E}_{\boldsymbol{\varepsilon}}\left[\mathbf{a}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\boldsymbol{\Sigma}(\mu n\mathbf{I}+\mathbf{X}^{\mathsf{T}}\mathbf{X})^{\dagger}(\mathbf{I}-\mathbf{V}\boldsymbol{\Phi}(k;\mu)\mathbf{V}^{\mathsf{T}})\mathbf{X}^{\mathsf{T}}\boldsymbol{\varepsilon} \mid \mathbf{X}\right] \\ &= \mathbf{a}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\boldsymbol{\Sigma}(\mu n\mathbf{I}+\mathbf{X}^{\mathsf{T}}\mathbf{X})^{\dagger}(\mathbf{I}-\mathbf{V}\boldsymbol{\Phi}(k;\mu)\mathbf{V}^{\mathsf{T}})\mathbf{X}^{\mathsf{T}}\mathbb{E}_{\boldsymbol{\varepsilon}}\left[\boldsymbol{\varepsilon} \mid \mathbf{X}\right] = \mathbf{0},\end{aligned}$$

since $\mathbb{E}_{\boldsymbol{\varepsilon}}[\boldsymbol{\varepsilon} \mid \mathbf{X}] = \mathbf{0}$ by Assumption 1. Thus, the cross-term vanishes, and we conclude that $R_{\mathbf{X}}(\boldsymbol{\beta}_k) = \|\boldsymbol{\Sigma}^{1/2}\mathbf{Va}\|_2^2 + \mathbb{E}_{\boldsymbol{\varepsilon}}[\|\boldsymbol{\Sigma}^{1/2}\mathbf{Vb}\|_2^2 \mid \mathbf{X}]$, which leads to the claimed result. □

As an immediate corollary of Proposition 5.4.1, we can read off the expected excess risk for the ridgeless case with $\mu = 0$.

Corollary 5.4.2 (Ridgeless risk)**.** *Let* $\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}_{\mathbf{X}}\mathbf{V}^{\mathsf{T}}$ *be any feature matrix and* $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}_* + \boldsymbol{\varepsilon}$*. Assume that Assumption 1 holds. If* $\boldsymbol{\beta}_k$ *are the parameters after* k *steps of gradient descent for the least squares problem* (5.1) *with* $\mu = 0$*, initialized at any* $\boldsymbol{\beta}_0$*, and with arbitrary learning rate schedule* $\{\eta_k\}_{k\geq 1}$*, then, recalling that* $\Phi(k) \equiv \Phi(k;0)$*,*

$$R_{\mathbf{X}}(\boldsymbol{\beta}_k) = \left\|\boldsymbol{\Sigma}^{1/2}\mathbf{V}\boldsymbol{\Phi}(k)(\widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_*)\right\|_2^2 + \tau^2\left\|\boldsymbol{\Sigma}^{1/2}\mathbf{V}(\mathbf{I}-\boldsymbol{\Phi}(k))\boldsymbol{\Sigma}_{\mathbf{X}}^{\dagger}\right\|_F^2.$$

Remark 5.4.3 (Early stopping as regularization)**.** Corollary 5.4.2 offers another viewpoint on the regularization effects of early stopping to complement Theorem 5.3.1, which describes the regularizing effects of early stopping on the solution obtained from training the model with a zero initialization. On the other hand, Corollary 5.4.2 allows us to understand early stopping from the perspective of the generalization error with arbitrary initialization. Indeed, prior work [DW18; CM24] has shown that for generic data, the expected excess risk for the solution $\boldsymbol{\beta}^{(\mu)}$ to the ridge regression problem (5.1) with regularization parameter μ is given by

$$R_{\mathbf{X}}(\boldsymbol{\beta}^{(\mu)}) = \mu^2\boldsymbol{\beta}_*^{\mathsf{T}}(\widehat{\boldsymbol{\Sigma}}+\mu\mathbf{I})^{-1}\boldsymbol{\Sigma}(\widehat{\boldsymbol{\Sigma}}+\mu\mathbf{I})^{-1}\boldsymbol{\beta}_* + \frac{\tau^2}{n}\operatorname{Tr}\left(\boldsymbol{\Sigma}\widehat{\boldsymbol{\Sigma}}(\widehat{\boldsymbol{\Sigma}}+\mu\mathbf{I})^{-2}\right).$$

From Corollary 5.4.2, we see that the early stopped risk after k iterations of gradient descent for the unregularized least squares problem (with $\mu = 0$) is given by

$$R_{\mathbf{X}}(\boldsymbol{\beta}_k) = (\boldsymbol{\beta}_* - \boldsymbol{\beta}_0)^\mathsf{T}(\mathbf{V}\boldsymbol{\Phi}(k)\mathbf{V}^\mathsf{T})\boldsymbol{\Sigma}(\mathbf{V}\boldsymbol{\Phi}(k)\mathbf{V}^\mathsf{T})(\boldsymbol{\beta}_* - \boldsymbol{\beta}_0) + \frac{\tau^2}{n}\operatorname{Tr}\left(\boldsymbol{\Sigma}\widehat{\boldsymbol{\Sigma}}^\dagger(\mathbf{I} - \mathbf{V}\boldsymbol{\Phi}(k)\mathbf{V}^\mathsf{T})^2\right),$$

recalling that $\widehat{\boldsymbol{\Sigma}} = n^{-1}\mathbf{X}^\mathsf{T}\mathbf{X}$. We see that this is similar to the risk for $\boldsymbol{\beta}^{(\mu)}$ under the correspondence

$$\mathbf{V}\boldsymbol{\Phi}(k)\mathbf{V}^\mathsf{T} \leftrightarrow \mu(\widehat{\boldsymbol{\Sigma}} + \mu\mathbf{I})^{-1}.$$

Indeed, note that if $\mathbf{I} - \mathbf{V}\boldsymbol{\Phi}(k)\mathbf{V}^\mathsf{T} = \mathbf{I} - \mu(\widehat{\boldsymbol{\Sigma}} + \mu\mathbf{I})^{-1} = \widehat{\boldsymbol{\Sigma}}(\widehat{\boldsymbol{\Sigma}} + \mu\mathbf{I})^{-1}$, then $\widehat{\boldsymbol{\Sigma}}^\dagger(\mathbf{I} - \mathbf{V}\boldsymbol{\Phi}(k)\mathbf{V}^\mathsf{T})^2 = \widehat{\boldsymbol{\Sigma}}(\widehat{\boldsymbol{\Sigma}} + \mu\mathbf{I})^{-2}$. Recall that the j^{th} diagonal entry of $\boldsymbol{\Phi}(k) \equiv \boldsymbol{\Phi}(k; 0)$ is equal to $\prod_{i=1}^{k}\big(1 - \eta_i n^{-1}\Lambda_j\big)$.

5.4.2 When is early stopping beneficial?

Given Proposition 5.4.1, all we need to do to find the optimal iteration k to minimize the expected excess risk is, in principle, to differentiate $R_{\mathbf{X}}(\boldsymbol{\beta}_k)$ with respect to k and compute the critical points. However, the expression is only defined for discrete values of k. To circumvent this technicality, we impose the following assumption on the function φ, which implies that $\boldsymbol{\Phi}(k; \mu)$ can be extended to a differentiable function of k:

Assumption 2. For all fixed ζ and learning rate schedules $\{\eta_i\}_{i\geq 1}$ such that for all i, $\eta_i \leq \zeta^{-1}$, the function $k \mapsto \varphi(k; \zeta, \{\eta_i\})$ can be extended to a monotonic differentiable function on $[1, \infty)$.

The differentiability of the extension is satisfied by many learning rate schedules. For example, recall that with constant step sizes $\eta_k \equiv \eta$, $\varphi(k; \zeta) = (1 - \eta\zeta)^k$, which can clearly be extended to the differentiable function $x \mapsto (1 - \eta\zeta)^x$ on $\mathbb{R}$. Propositions 5.2.6 and 5.2.7 show that learning rate schedules with polynomial decay ($\eta_k = \eta/k^m$) and constant additive decay ($\eta_k = \eta_0 - k\eta$) also satisfy Assumption 2 since the Gamma function is known to

be differentiable on $[1, \infty)$. Finally, learning rates with exponential decay ($\eta_k = \eta^k$) also satisfy this assumption; this is implied by Proposition 5.2.8 and the differentiability of the q-Pochhammer symbol, the technical details of which are established in Section 5.7.

Additionally, we shall also make the following statistical assumption on $\boldsymbol{\beta}_* - \boldsymbol{\beta}_0$, which is similar to a common statistical *spherical prior assumption* that $\boldsymbol{\beta}_* \sim N(\mathbf{0}, \sigma^2 \mathbf{I})$ with zero initialization [DW18; ASS20; AKT19]:

Assumption 3. The entries of $\boldsymbol{\beta}_* - \boldsymbol{\beta}_0$ are i.i.d. and have mean 0 and variance σ^2.

This means that the results address the performance of generic signals on average. Under Assumption 3, we shall also take an additional expectation over $\boldsymbol{\beta}_* - \boldsymbol{\beta}_0$ and analyze the *Bayes excess risk*:

$$\bar{R}_{\mathbf{X}}(\boldsymbol{\beta}_k) := \mathbb{E}_{\boldsymbol{\beta}_* - \boldsymbol{\beta}_0}[R_{\mathbf{X}}(\boldsymbol{\beta}_k)] \tag{5.25}$$

To build intuition, we begin by presenting our results on when early stopping for the least squares problem (5.1) with regularization parameter $\mu = 0$ is beneficial.

Theorem 5.4.4 (Early stopping)**.** *Let $\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}_{\mathbf{X}}\mathbf{V}^\mathsf{T}$ be any feature matrix with $\operatorname{rank}(\mathbf{X}) = r$, and $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}_* + \boldsymbol{\varepsilon}$. Recall that Λ_1 is the largest eigenvalue of $\mathbf{X}^\mathsf{T}\mathbf{X}$. Suppose that Assumptions 1, 2, and 3 hold. Let $\boldsymbol{\beta}_k$ be the parameters after k steps of gradient descent for the least squares problem (5.1) with $\mu = 0$, initialized at any $\boldsymbol{\beta}_0$. If the learning rate schedule $\{\eta_k\}_{k \geq 1}$ is such that $\eta_k \leq 1/(n^{-1}\Lambda_1)$ for all k, and for all $j \leq r$, we have that*

$$\lim_{k \to \infty} \frac{\varphi(k; n^{-1}\Lambda_j)}{\varphi(0; n^{-1}\Lambda_j)} < \frac{\tau^2}{\Lambda_j \sigma^2 + \tau^2}, \tag{5.26}$$

then there is a finite T such that for all $k \geq T$, $\bar{R}_{\mathbf{X}}(\boldsymbol{\beta}_k) \geq \bar{R}_{\mathbf{X}}(\boldsymbol{\beta}_T)$. That is, early stopping is beneficial. Furthermore, if $\mathbf{P} = \mathbf{V}^\mathsf{T}\boldsymbol{\Sigma}\mathbf{V}$, then for all k,

$$\bar{R}_{\mathbf{X}}(\boldsymbol{\beta}_k) \geq \sigma^2 \left[\sum_{j=1}^{r} P_{j,j} \frac{\tau^2}{\Lambda_j \sigma^2 + \tau^2} + \sum_{j=r+1}^{p} P_{j,j} \right]. \tag{5.27}$$

Proof. From Corollary 5.4.2, noting that $\mathbf{\Lambda}^\dagger = (\mathbf{\Sigma}_\mathbf{X}^\mathsf{T}\mathbf{\Sigma}_\mathbf{X})^\dagger$ is a rank r diagonal matrix, the expected excess risk after k iterations is given by

$$\begin{aligned} R_\mathbf{X}(\boldsymbol{\beta}_k) &= (\widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_*)^\mathsf{T}\mathbf{\Phi}(k)\mathbf{P}\mathbf{\Phi}(k)(\widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_*) + \tau^2 \operatorname{Tr}\left(\mathbf{\Lambda}^\dagger(\mathbf{I} - \mathbf{\Phi}(k))\mathbf{P}(\mathbf{I} - \mathbf{\Phi}(k))\right) \\ &= \sum_{i,j=1}^{p} P_{i,j}\left[\frac{\varphi(k; n^{-1}\Lambda_i)\varphi(k; n^{-1}\Lambda_j)}{\varphi(0; n^{-1}\Lambda_i)\varphi(0; n^{-1}\Lambda_j)}(\widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_*)_i(\widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_*)_j\right] \\ &\quad + \sum_{j=1}^{r} P_{j,j} \cdot \tau^2 \frac{1}{\Lambda_j}\left(1 - \frac{\varphi(k; n^{-1}\Lambda_j)}{\varphi(0; n^{-1}\Lambda_j)}\right)^2. \end{aligned}$$

Taking the expectation over $\boldsymbol{\beta}_* - \boldsymbol{\beta}_0$ under Assumption 3, we see that the cross terms vanish, and the Bayes excess risk is given by

$$\bar{R}_\mathbf{X}(\boldsymbol{\beta}_k) = \sum_{j=1}^{p} P_{j,j} \cdot \frac{\varphi(k; n^{-1}\Lambda_j)^2}{\varphi(0; n^{-1}\Lambda_j)^2} \cdot \sigma^2 + \sum_{j=1}^{r} P_{j,j} \cdot \tau^2 \frac{1}{\Lambda_j}\left(1 - \frac{\varphi(k; n^{-1}\Lambda_j)}{\varphi(0; n^{-1}\Lambda_j)}\right)^2. \tag{5.28}$$

Taking the derivative in k and noting that $\varphi(k; n^{-1}\Lambda_j)$ is constant for $j > r$, we get

$$\begin{aligned} \partial_k \bar{R}_\mathbf{X}(\boldsymbol{\beta}_k) &= 2\sum_{j=1}^{r} P_{j,j} \cdot \frac{\partial_k \varphi(k; n^{-1}\Lambda_j)}{\varphi(0; n^{-1}\Lambda_j)}\left[\frac{\varphi(k; n^{-1}\Lambda_j)}{\varphi(0; n^{-1}\Lambda_j)} \cdot \sigma^2 - \tau^2 \frac{1}{\Lambda_j}\left(1 - \frac{\varphi(k; n^{-1}\Lambda_j)}{\varphi(0; n^{-1}\Lambda_j)}\right)\right] \\ &= 2\sum_{j=1}^{r} \frac{P_{j,j}}{\Lambda_j} \cdot \frac{\partial_k \varphi(k; n^{-1}\Lambda_j)}{\varphi(0; n^{-1}\Lambda_j)}\left[\frac{\varphi(k; n^{-1}\Lambda_j)}{\varphi(0; n^{-1}\Lambda_j)}\left(\sigma^2\Lambda_j + \tau^2\right) - \tau^2\right]. \end{aligned}$$

Note that since $\eta_k \le 1/(n^{-1}\Lambda_1)$ for all k, $\varphi(k; n^{-1}\Lambda_j)/\varphi(0; n^{-1}\Lambda_j)$ is a non-increasing function of k by Assumption 2, and hence $\partial_k \varphi(k; n^{-1}\Lambda_j)/\varphi(0; n^{-1}\Lambda_j) \le 0$. Since $\Lambda_j \ge 0$ and $P_{j,j} \ge 0$ (because $\mathbf{P}$ is a positive semidefinite matrix), the sign of the derivative $\partial_k \bar{R}_\mathbf{X}(\boldsymbol{\beta}_k)$ is determined by

$$\frac{\varphi(k; n^{-1}\Lambda_j)}{\varphi(0; n^{-1}\Lambda_j)}\left(\sigma^2\Lambda_j + \tau^2\right) - \tau^2.$$

Therefore, by using the assumption (5.26) that $\lim_{k\to\infty} \varphi(k; n^{-1}\Lambda_j)/\varphi(0; n^{-1}\Lambda_j) < \tau^2/(\sigma^2\Lambda_j + \tau^2)$, we deduce that for all sufficiently large k, $\partial_k \bar{R}_\mathbf{X}(\boldsymbol{\beta}_k) \ge 0$. This shows that the derivative of the Bayes excess risk is eventually positive, from which we conclude that we should have stopped earlier to minimize the risk.

To obtain the lower bound on the Bayes excess risk, observe that we can minimize the expected excess risk independently in each eigendirection. Since φ is monotonic, we see that solving

$$\frac{\varphi(k_j; n^{-1}\Lambda_j)}{\varphi(0; n^{-1}\Lambda_j)} = \frac{\tau^2}{\sigma^2\Lambda_j + \tau^2}$$

for k_j achieves a global minimum in the j^{th} eigendirection for $j \le r$. By plugging this into the expression (5.28) for the Bayes excess risk for each $j \le r$, we obtain the lower bound as follows. For $j \le r$, the corresponding term in the summation is $P_{j,j}$ times

$$\begin{aligned}\sigma^2\left(\frac{\tau^2}{\sigma^2\Lambda_j+\tau^2}\right)^2 + \tau^2\frac{1}{\Lambda_j}\left(1-\frac{\tau^2}{\sigma^2\Lambda_j+\tau^2}\right)^2 &= \sigma^2\left(\frac{\tau^2}{\sigma^2\Lambda_j+\tau^2}\right)^2 + \tau^2\frac{1}{\Lambda_j}\left(\frac{\sigma^2\Lambda_j}{\sigma^2\Lambda_j+\tau^2}\right)^2 \\ &= \frac{\sigma^2\tau^2}{\sigma^2\Lambda_j+\tau^2}.\end{aligned}$$

For $j > r$, note that $\varphi(k; n^{-1}\Lambda_j)/\varphi(0; n^{-1}\Lambda_j) = 1$. This completes the proof. □

Remark 5.4.5. The only reason that Assumption 3 is needed is to address the cross terms in (5.28). However, another condition that would lead the cross terms to vanish is if the matrix $\mathbf{P} = \mathbf{V}^\mathsf{T}\boldsymbol{\Sigma}\mathbf{V}$ is diagonal. This is satisfied if the features are isotropic, i.e., $\boldsymbol{\Sigma} = \mathbf{I}$, or more generically, if $\boldsymbol{\Sigma}$ and $\widehat{\boldsymbol{\Sigma}}$ are simultaneously diagonalizable by the same basis of eigenvectors $\mathbf{V}$. The latter is similar to a common requirement in works studying the problem of covariate shift, where assumptions on the alignment between the eigenspaces of the covariance matrices of the training and test distributions are required [TAP21]. If $\mathbf{P}$ is diagonal, then a version of Theorem 5.4.4 showing that early stopping is beneficial for minimizing the expected excess risk $R_{\mathbf{X}}(\boldsymbol{\beta}_k)$ holds under the assumption that for all $j \le \operatorname{rank}(\mathbf{X})$,

$$\lim_{k\to\infty}\frac{\varphi(k; n^{-1}\Lambda_j)}{\varphi(0; n^{-1}\Lambda_j)} < \frac{\tau^2}{\Lambda_j(\widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_*)_j^2 + \tau^2},$$

and a lower bound on the expected excess risk is given by

$$R_{\mathbf{X}}(\boldsymbol{\beta}_k) \ge \sum_{j=1}^{r} P_{j,j}\cdot\frac{\tau^2(\widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_*)_j^2}{\Lambda_j(\widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_*)_j^2 + \tau^2} + \sum_{j=r+1}^{p} P_{j,j}\cdot(\widetilde{\boldsymbol{\beta}}_0 - \widetilde{\boldsymbol{\beta}}_*)_j^2.$$

Next, as a counterpart to Theorem 5.4.4, we shall provide sufficient conditions for when early stopping for the least squares problem (5.1) with regularization parameter $\mu = 0$ is *not beneficial*.

Theorem 5.4.6 (Early stopping converse)**.** *Consider the same setup as Theorem 5.4.4. If the learning rate schedule $\{\eta_k\}_{k\geq 1}$ is such that $\eta_k \leq 1/(n^{-1}\Lambda_1)$ for all k, and for all $j \leq \operatorname{rank}(X)$, we have that*

$$\lim_{k\to\infty} \frac{\varphi(k; n^{-1}\Lambda_j)}{\varphi(0; n^{-1}\Lambda_j)} \geq \frac{\tau^2}{\Lambda_j \sigma^2 + \tau^2}, \tag{5.29}$$

then early stopping is not beneficial.

Proof. The proof is the same as for Theorem 5.4.4, except that the inequality for $\partial_k \bar{R}_{\mathbf{X}}(\boldsymbol{\beta}_k)$ is reversed, and we deduce that the derivative of the Bayes excess risk is always negative. Hence, early stopping is not beneficial since the risk can always be decreased by further iterations. □

Remark 5.4.7 (Step size schedules and late generalization)**.** In summary, Theorems 5.4.4 and 5.4.6 show that the learning rate schedule affects whether early stopping is beneficial or not. Theorem 5.4.4 provides a sufficient condition for when early stopping is beneficial. In particular, if $\varphi(k; \zeta, \{\eta_i\}) \to 0$ as $k \to \infty$ for all ζ, then we see that early stopping is *always beneficial, independent of the spectrum of the covariance matrices of the training and test data*. Examples of learning rate schedules $\{\eta_k\}_{k\geq 1}$ that satisfy this assumption include constant learning rates ($\eta_k \equiv \eta < n^{-1}\Lambda_1$) and learning rates with linear decay ($\eta_k = \eta/k$). Furthermore, if the learning rates satisfy Assumption 2 along with the *Robbins–Monro conditions* from stochastic optimization:

$$\sum_{k=1}^{\infty} \eta_k = \infty \quad \text{and} \quad \sum_{k=1}^{\infty} \eta_k^2 < \infty, \tag{5.30}$$

then $\varphi(k; \zeta) \to 0$ as $k \to \infty$ for all ζ, and hence early stopping is *always beneficial*.

We can interpret *late generalization* or *grokking* as the phenomena where we want to keep training, even after overfitting the noise. On the other hand, when early stopping is beneficial, we have shown that we do not want to overfit the noise. Hence, we show that for many different learning rate schedules, linear models trained by gradient descent do not exhibit late generalization.

Theorem 5.4.6 allows us to construct examples of learning rate schedules for which it is possible that early stopping is not beneficial, including fast decaying step sizes such as learning rates with polynomial decay ($\eta_k = \eta/k^m$ with $m > 1$) or exponential decay ($\eta_k = \eta^k$ with $\eta < 1$).

Having built intuition in the ridgeless setting, we now characterize when early stopping is beneficial for solving the ridge regression problem (5.1) with regularization parameter $\mu \geq 0$.

Theorem 5.4.8 (Early stopping with ridge regularization)**.** *Let $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}_\mathbf{X}\mathbf{V}^\mathsf{T}$ be any feature matrix and $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}_* + \boldsymbol{\varepsilon}$. Recall that Λ_1 is the largest eigenvalue of $\mathbf{X}^\mathsf{T}\mathbf{X}$. Suppose that Assumptions 1, 2, and 3 hold. Let $\boldsymbol{\beta}_k$ be the parameters after k steps of gradient descent for the ridge regression problem (5.1) with regularization parameter $\mu > 0$, initialized at $\boldsymbol{\beta}_0 = \mathbf{0}$. If the learning rate schedule $\{\eta_k\}_{k\geq 1}$ is such that $\eta_k \leq (\mu + n^{-1}\Lambda_1)^{-1}$ for all k, and for all $j = 1, \ldots, p$, we have that*

$$\lim_{k\to\infty} \frac{\varphi(k; \mu + n^{-1}\Lambda_j)}{\varphi(0; \mu + n^{-1}\Lambda_j)} < \frac{\tau^2 - \sigma^2\mu n}{\tau^2 + \sigma^2\Lambda_j}, \tag{5.31}$$

then there exists a finite T such that for all $k \geq T$, $\bar{R}_\mathbf{X}(\boldsymbol{\beta}_k) \geq \bar{R}_\mathbf{X}(\boldsymbol{\beta}_T)$. That is, early stopping is beneficial.

On the other hand, if for all $j = 1, \ldots, p$, we have that

$$\lim_{k\to\infty} \frac{\varphi(k; \mu + n^{-1}\Lambda_j)}{\varphi(0; \mu + n^{-1}\Lambda_j)} \geq \frac{\tau^2 - \sigma^2\mu n}{\tau^2 + \sigma^2\Lambda_j}, \tag{5.32}$$

then early stopping is not beneficial.

Proof of Theorem 5.4.8. Let $\mathbf{P} = \mathbf{V}^\mathsf{T}\mathbf{\Sigma}\mathbf{V}$ as in the proof of Theorem 5.4.4. From Proposition 5.4.1, the expected excess risk after k iterations is given by

$$R_{\mathbf{X}}(\boldsymbol{\beta}_k) = \boldsymbol{\gamma}^\mathsf{T}\mathbf{P}\boldsymbol{\gamma} + \tau^2 \operatorname{Tr}\left(\mathbf{P}\left[(\mu n\mathbf{I} + \mathbf{\Lambda})^\dagger\right]^2 (\mathbf{I} - \Phi(k;\mu))^2\mathbf{\Lambda}\right), \tag{5.33}$$

where

$$\gamma := \left[\mathbf{\Phi}(k;\mu)(-\widetilde{\boldsymbol{\beta}}_*) - \mu n(\mu n\mathbf{I} + \mathbf{\Lambda})^\dagger(\mathbf{I} - \mathbf{\Phi}(k;\mu))\widetilde{\boldsymbol{\beta}}_*\right].$$

By expanding the inner product, we can write the first term of (5.33) as the sum of the following three terms:

$$\begin{aligned}\boldsymbol{\gamma}^\mathsf{T}\mathbf{P}\boldsymbol{\gamma} = &\left[\mathbf{\Phi}(k;\mu)(-\widetilde{\boldsymbol{\beta}}_*)\right]^\mathsf{T}\mathbf{P}\mathbf{\Phi}(k;\mu)(-\widetilde{\boldsymbol{\beta}}_*)\\ &+ \left[\mu n(\mu n\mathbf{I} + \mathbf{\Lambda})^\dagger(\mathbf{I} - \mathbf{\Phi}(k;\mu))\widetilde{\boldsymbol{\beta}}_*\right]^\mathsf{T}\mathbf{P}\mu n(\mu n\mathbf{I} + \mathbf{\Lambda})^\dagger(\mathbf{I} - \mathbf{\Phi}(k;\mu))\widetilde{\boldsymbol{\beta}}_*\\ &- 2\left[\mu n(\mu n\mathbf{I} + \mathbf{\Lambda})^\dagger(\mathbf{I} - \mathbf{\Phi}(k;\mu))\widetilde{\boldsymbol{\beta}}_*\right]^\mathsf{T}\mathbf{P}\mathbf{\Phi}(k;\mu)(-\widetilde{\boldsymbol{\beta}}_*).\end{aligned}$$

After taking expectations with respect to $\boldsymbol{\beta}_*$ under Assumption 3, we see that the first summand becomes

$$\sum_{j=1}^{p} P_{j,j} \cdot \Phi(k)_j^2 \cdot \sigma^2,$$

where $\Phi(k)_j = \varphi(k;\mu+n^{-1}\Lambda_j)/\varphi(0;\mu+n^{-1}\Lambda_j)$ is the j^{th} diagonal entry of $\mathbf{\Phi}(k;\mu)$. Similarly, the second summand becomes

$$\sum_{j=1}^{p} P_{j,j} \cdot \frac{\mu^2 n^2}{(\mu n + \Lambda_j)^2}(1 - \Phi(k)_j)^2 \cdot \sigma^2,$$

and the third summand becomes

$$2\sum_{j=1}^{p} P_{j,j} \cdot \frac{\mu n}{\mu n + \Lambda_j}\Phi(k)_j(1 - \Phi(k)_j) \cdot \sigma^2.$$

Next, we can write the second term of (5.33) as the following:

$$\tau^2 \operatorname{Tr}\left(\mathbf{P}\left[(\mu n\mathbf{I}+\mathbf{\Lambda})^\dagger\right]^2(\mathbf{I}-\mathbf{\Phi}(k;\mu))^2\mathbf{\Lambda}\right) = \sum_{j=1}^{p} P_{j,j}\cdot\frac{\tau^2\Lambda_j}{(\mu n+\Lambda_j)^2}(1-\Phi(k)_j)^2.$$

Thus, summing the displayed expressions above shows that the Bayes excess risk is given by

$$\bar{R}_{\mathbf{X}}(\boldsymbol{\beta}_k) = \sum_{j=1}^{p} P_{j,j}\left[\Phi(k)_j^2\sigma^2 + \left(\frac{\mu^2n^2\sigma^2+\tau^2\Lambda_j}{(\mu n+\Lambda_j)^2}\right)(1-\Phi(k)_j)^2 + 2\sigma^2\frac{\mu n}{\mu n+\Lambda_j}\Phi(k)_j(1-\Phi(k)_j)\right].$$

To analyze the benefit of early stopping, we compute the derivative with respect to k:

$$\partial_k\bar{R}_{\mathbf{X}}(\boldsymbol{\beta}_k) = 2\sum_{j=1}^{p} P_{j,j}\cdot\partial_k\Phi(k)_j\left[\Phi(k)_j\sigma^2 - \frac{\mu^2n^2\sigma^2+\tau^2\Lambda_j}{(\mu n+\Lambda_j)^2}(1-\Phi(k)_j) + \sigma^2\frac{\mu n}{\mu n+\Lambda_j}(1-2\Phi(k)_j)\right].$$

Note that with a ridge regularization parameter $\mu > 0$, $\Phi(k)_j$ is not necessarily constant for $j > \operatorname{rank}(X)$. Since $\partial_k\Phi(k)_j \le 0$, the sign of $\partial_k\bar{R}_{\mathbf{X}}(\boldsymbol{\beta}_k)$ depends on the sign of the expression inside the brackets:

$$\Phi(k)_j\sigma^2 - \frac{\mu^2n^2\sigma^2+\tau^2\Lambda_j}{(\mu n+\Lambda_j)^2}(1-\Phi(k)_j) + \sigma^2\frac{\mu n}{\mu n+\Lambda_j}(1-2\Phi(k)_j).$$

Simplifying, the coefficient of the term with $\Phi(k)_j$ is given by

$$\begin{aligned}\sigma^2\left[1+\frac{\mu^2n^2}{(\mu n+\Lambda_j)^2}-2\frac{\mu n}{\mu n+\Lambda_j}\right]+\frac{\tau^2\Lambda_j}{(\mu n+\Lambda_j)^2} &= \sigma^2\frac{\Lambda_j^2}{(\mu n+\Lambda_j)^2}+\frac{\tau^2\Lambda_j}{(\mu n+\Lambda_j)^2}\\ &= \Lambda_j\cdot\frac{\sigma^2\Lambda_j+\tau^2}{(\mu n+\Lambda_j)^2},\end{aligned}$$

and the coefficient for the remaining terms without $\Phi(k)_j$ is given by

$$-\frac{\mu^2n^2\sigma^2+\tau^2\Lambda_j}{(\mu n+\Lambda_j)^2}+\sigma^2\frac{\mu n}{\mu n+\Lambda_j} = \Lambda_j\cdot\frac{\sigma^2\mu n-\tau^2}{(\mu n+\Lambda_j)^2}.$$

Thus, we have shown that

$$\partial_k \bar{R}_{\mathbf{X}}(\boldsymbol{\beta}_k) = 2 \sum_{j=1}^{r} P_{j,j} \cdot \partial_k \Phi(k)_j \cdot \frac{\Lambda_j}{(\mu n + \Lambda_j)^2} \left[\Phi(k)_j \cdot (\sigma^2 \Lambda_j + \tau^2) + (\sigma^2 \mu n - \tau^2) \right]. \quad (5.34)$$

Observe that for each j, solving for the value of $\Phi(k)_j$ for which the square bracket is zero yields

$$\Phi(k)_j = \frac{\tau^2 - \sigma^2 \mu n}{\tau^2 + \sigma^2 \Lambda_j}.$$

Therefore, if the condition (5.31) holds for all $j = 1, \ldots, p$, then the gradient is eventually positive, i.e., $\partial_k \bar{R}_{\mathbf{X}}(\boldsymbol{\beta}_k) \geq 0$ for all sufficiently large k, which implies that early stopping is beneficial. Conversely, if (5.32) holds for all $j = 1, \ldots, p$, then we deduce that $\partial_k \bar{R}_{\mathbf{X}}(\boldsymbol{\beta}_k) \leq 0$, which implies that early stopping is not beneficial. □

Remark 5.4.9 (Early stopping for ridge regression)**.** Theorem 5.4.8 extends Theorems 5.4.4 and 5.4.6 by providing conditions under which early stopping remains beneficial (or not) even in the presence of explicit ridge regularization. It reveals an interesting implication: if $\mu \geq \tau^2/(n\sigma^2)$, then early stopping is never beneficial. In other words, if the explicit ridge regularization is too strong, then the benefits of early stopping diminish. As discussed in the related works, this particular result was also established concurrently by Stark and Steinerberger [SS25, Theorem 2.5] for gradient descent with constant learning rates.

Suppose that we train our model until convergence for the ridge regularized problem. Then prior work [Nak+21] has shown that $\mu^* = \tau^2/(n\sigma^2)$ is an optimal ridge regularization parameter that minimizes the generalization error. Hence, if we choose $\mu = \mu^*$, then early stopping is not beneficial! However, this does not imply that the expected excess risk cannot be improved by using *both* ridge regularization and early stopping. In particular, the optimal early stopped solution with $\mu < \mu^*$ may outperform the converged solution with $\mu = \mu^*$.

We validate this numerically in Figure 5.1. In this experiment, we sample 40 data points $\mathbf{x} \sim \mathbf{z}\boldsymbol{\Sigma}^{1/2} \in \mathbb{R}^p$ with $p = 100$, where $\mathbf{z} \sim N(\mathbf{0}, p^{-1}\mathbf{I})$ and $\boldsymbol{\Sigma}$ is a diagonal matrix with entries $\Sigma_{j,j} = j^{-2}$. We sample $2,000$ parameters $\boldsymbol{\beta}_* \sim N(\mathbf{0}, p^{-1}\mathbf{I})$ and independent mean-

zero Gaussian noise with variance $\tau^2 = 1$; i.e., we work in the spherical prior setting with $\sigma^2 = 0.01$. We consider gradient descent with constant learning rate $\eta_k \equiv \eta = 0.01/(n^{-1}\Lambda_1)$ for four ridge regression problems with regularization parameters $\mu = 0, 0.5, 2.5, 4$, where Λ_1 is the largest eigenvalue of $\mathbf{X}^\mathsf{T}\mathbf{X}$. Here, the optimal ridge regularization parameter $\mu^* = 2.5$. We see that for $\mu < 2.5$, early stopping is beneficial, while for $\mu \geq 2.5$ early stopping is not beneficial. Additionally, we see that for $\mu > \mu^*$, the early stopped risk is very similar to the converged risk with $\mu = \mu^*$. Hence, we can obtain computational advantages by training for fewer iterations using smaller amounts of ridge regularization.

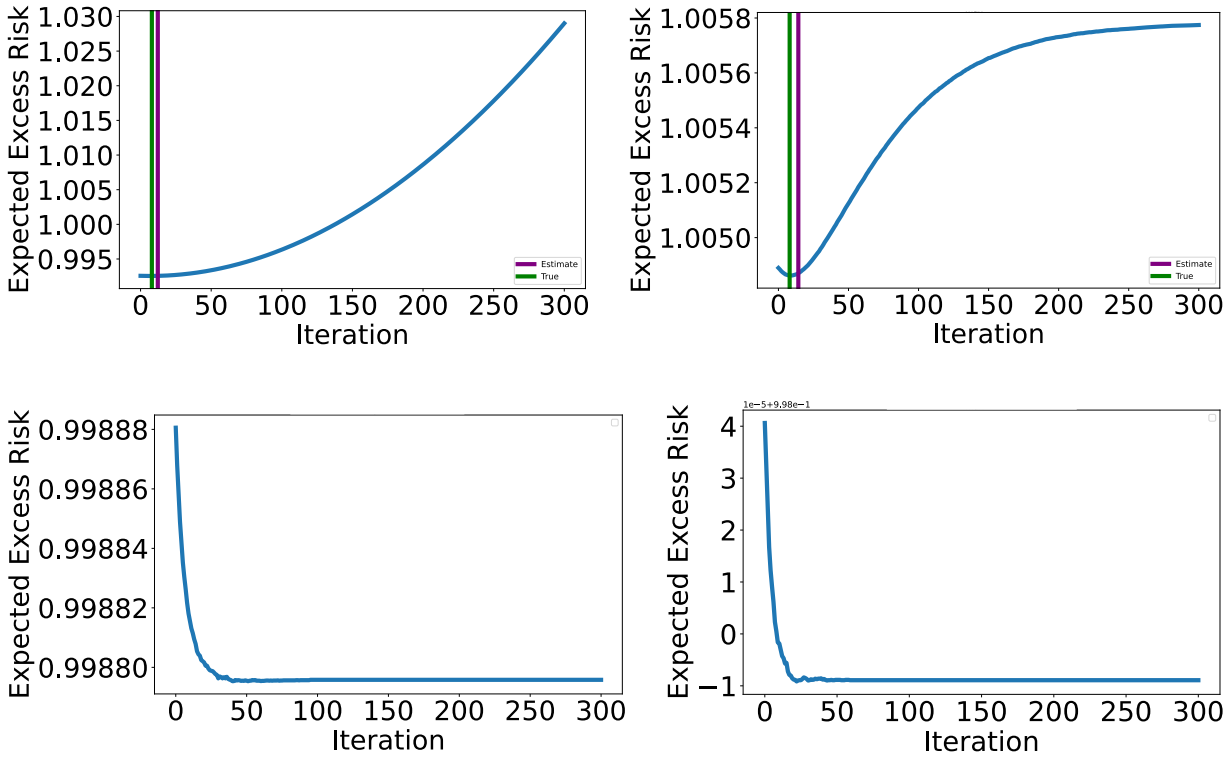


Figure 5.1: Bayes excess risk trajectories from using gradient descent with constant learning rate $\eta_k \equiv \eta = 0.01/(n^{-1}\Lambda_1)$ for four different ridge regression problems: from left to right, we have $\mu = 0, 0.5$ on the top, and $\mu = 2.5, 4$ on the bottom. The optimal ridge regularization parameter $\mu^* = 2.5$. When early stopping is beneficial (top), the purple line shows our estimate for the stopping time that we give later in (5.37), and the green line shows the empirical optimal stopping time.

5.5 Optimal stopping time estimate

Having shown that early stopping is beneficial for a wide variety of learning rate schedules, we will now provide an estimate for the optimal stopping time. Recall that from (5.34) in the proof of Theorem 5.4.8, the derivative of the Bayes excess risk is given by

$$\partial_k \bar{R}_{\mathbf{X}}(\boldsymbol{\beta}_k) = 2\sum_{j=1}^{p} P_{j,j} \cdot \partial_k \Phi(k)_j \cdot \frac{\Lambda_j}{(\mu n + \Lambda_j)^2} \left[\Phi(k)_j \cdot (\sigma^2\Lambda_j + \tau^2) + (\sigma^2 \mu n - \tau^2)\right].$$

Solving the equation above for a critical point is quite challenging. However, we can determine the optimal stopping time for each eigendirection by setting each summand to zero individually. That is, recalling that $\Phi(k)_j = \varphi(k; \mu + n^{-1}\Lambda_j)/\varphi(0; \mu + n^{-1}\Lambda_j)$, for each $j \leq r$, we need to find k_j such that

$$\prod_{i=1}^{k}(1 - \eta_i(\mu + n^{-1}\Lambda_j)) = \frac{\tau^2 - \sigma^2\mu n}{\tau^2 + \sigma^2\Lambda_j}. \tag{5.35}$$

We may assume that $\mu \leq \tau^2/(n\sigma^2)$, otherwise early stopping is never beneficial. By taking the logarithm of both sides and using the first order expansion of $\log(1+x)$, we obtain

$$-\log\left(\frac{\tau^2 + \sigma^2\Lambda_j}{\tau^2 - \sigma^2\mu n}\right) = \sum_{i=1}^{k_j} \log(1 - \eta_i(\mu + n^{-1}\Lambda_j)) \approx -(\mu + n^{-1}\Lambda_j) \cdot \sum_{i=1}^{k_j} \eta_i.$$

Hence, the choice of k_j should satisfy

$$\sum_{i=1}^{k_j} \eta_i \approx \frac{\log\left(\frac{\tau^2+\sigma^2\Lambda_j}{\tau^2-\sigma^2\mu n}\right)}{\mu + n^{-1}\Lambda_j}. \tag{5.36}$$

If $\mu = 0$ and constant step sizes $\eta_i \equiv \eta$ are used, then from (5.36), we obtain the estimate $\widehat{k}_j \approx \log\left(1 + \frac{\sigma^2}{\tau^2}\Lambda_j\right)/(\eta n^{-1}\Lambda_j)$, which corresponds to the discrete version of the estimate proposed in [ASS20]; see [ASS20, Eq. (16)]. However, the estimate (5.36) obtained from our method allows for general step size schedules $\{\eta_k\}_{k\geq 1}$ and the presence of explicit ridge

regularization. Furthermore, note that as μ approaches $\tau^2/(n\sigma^2)$, the optimal stopping time goes to infinity. Finally, the expression (5.36) provides a stopping time estimate for each eigendirection. A single estimate can be obtained by using the mean of the eigenvalues; i.e., replacing Λ_j in (5.36) by the mean $\bar{\Lambda} := r^{-1} \sum_{j=1}^{r} \Lambda_j$.

Putting everything together, Theorem 5.4.8 and the heuristics discussed above lead to the following estimate of the optimal stopping time:

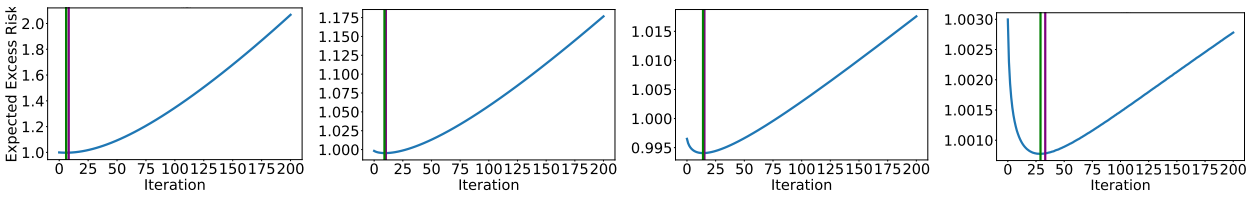
$$\widehat{k} = \arg\min_{k} \left\{ \sum_{i=1}^{k} \eta_i > \frac{\log\left(\frac{\tau^2+\sigma^2\bar{\Lambda}}{\tau^2-\sigma^2\mu n}\right)}{\mu + n^{-1}\bar{\Lambda}} \right\}. \tag{5.37}$$

Here, $\bar{\Lambda} = \mathrm{Tr}(\mathbf{X}^\mathsf{T}\mathbf{X})/\mathrm{rank}(\mathbf{X}^\mathsf{T}\mathbf{X})$, recalling that μ is the ridge regularization parameter, τ^2 is the variance of each entry of the residual ε_i under Assumption 1 (i.e., the strength of the noise), and σ^2 is the variance of $\boldsymbol{\beta}_* - \boldsymbol{\beta}_0$ under the prior in Assumption 3 (i.e., the strength of the signal). We test the performance of this estimate empirically in the next section.

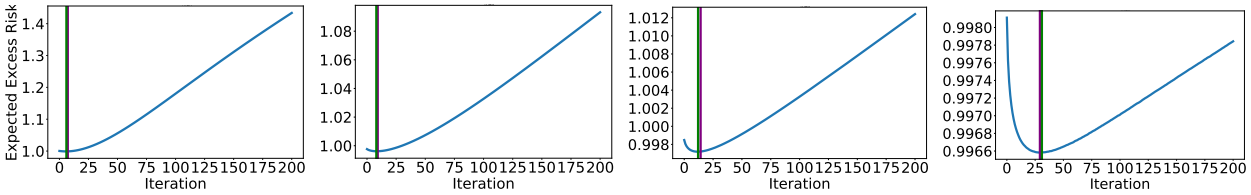
5.5.1 Experimental validation

Synthetic data. In this experiment, we sample n data points $\mathbf{x} \sim \mathbf{z}\boldsymbol{\Sigma}^{1/2} \in \mathbb{R}^p$, where $\mathbf{z} \sim N(\mathbf{0}, p^{-1}\mathbf{I})$ and $\boldsymbol{\Sigma}$ is a diagonal matrix with entries $\Sigma_{j,j} = j^{-\alpha}$. We consider gradient descent for the unregularized least squares problem ($\mu = 0$) with four different learning rate schedules given by $\eta_k = \eta/k^m$ for $m = 0, 1/4, 1/2, 3/4$ and $\eta = 0.9/(n^{-1}\Lambda_1)$, where Λ_1 is the largest eigenvalue of $\mathbf{X}^\mathsf{T}\mathbf{X}$. We shall work in the spherical prior setup (Assumption 3). For each setting, we sample 800 parameters $\boldsymbol{\beta}_* \sim N(\mathbf{0}, p^{-1}\mathbf{I})$ and independent, mean-zero Gaussian noise with variance τ^2.

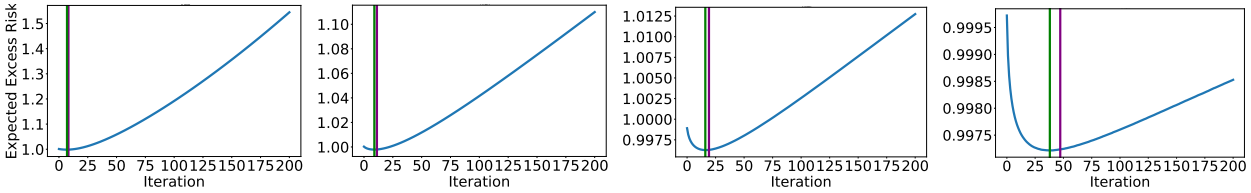
Figure 5.2 shows the Bayes excess risk curves, the empirical optimal stopping times, and the predicted stopping times from the estimate (5.37). As shown by the plots, our estimate is quite accurate. Furthermore, we see that the optimally early-stopped solutions with different learning schedules essentially all have the same risk (keeping all the other parameters p, n, τ, and α the same), which aligns with the lower bound from Theorem 5.4.4.



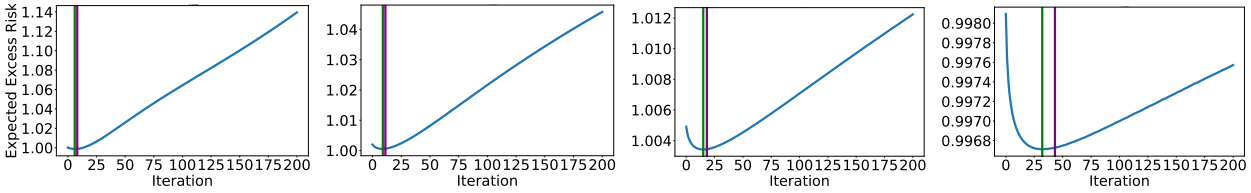
(a) $p = 40$, $n = 100$, $\tau = 1$, and $\alpha = 2$.



(b) $p = 40$, $n = 100$, $\tau = 1$, and $\alpha = 4$.



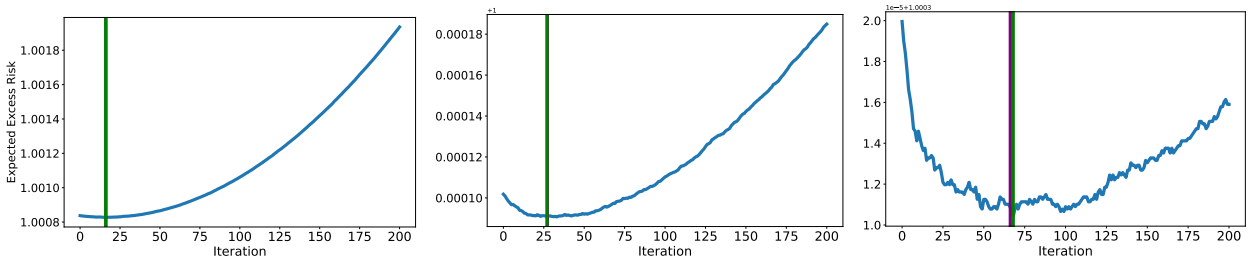
(c) $p = 100$, $n = 40$, $\tau = 0.15$, and $\alpha = 2$.



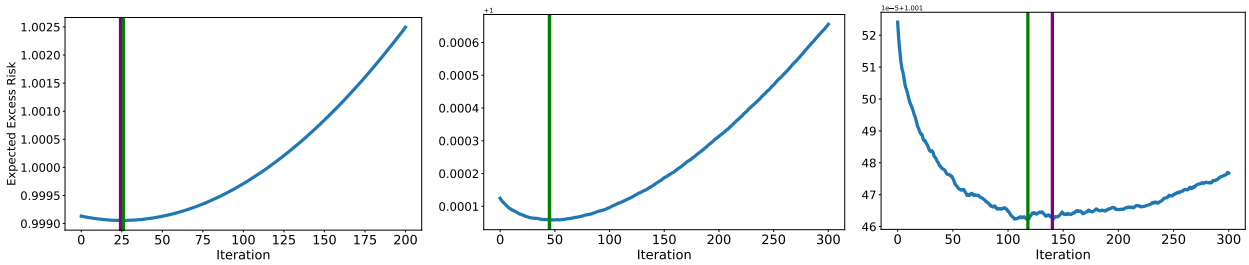
(d) $p = 100$, $n = 40$, $\tau = 0.15$, and $\alpha = 4$.

Figure 5.2: Bayes excess risk curves for different sets of parameters p, n, τ, and α, described in the main text. From left to right, the plots show the results from using the learning rate schedule $\eta_k = \eta/k^m$ for $m = 0, 1/4, 1/2, 3/4$ with $\eta = 0.9/(n^{-1}\Lambda_1)$. The purple line shows the estimated stopping time from (5.37), and the green line shows the empirical optimal stopping time.

Real data. We repeat the same experimental setup as above, where we sample n data points $\mathbf{x}$ from the CIFAR10 and MNIST datasets instead. We consider gradient descent the unregularized least squares problem ($\mu = 0$) with three different learning rate schedules given by $\eta_k = \eta/k^m$ for $m = 0, 1/4, 1/2$ and $\eta = 0.9/(n^{-1}\Lambda_1)$, where Λ_1 is the largest eigenvalue of $\mathbf{X}^\mathsf{T}\mathbf{X}$. The results, displayed in Figure 5.3, show that the same results also hold for real data.



(a) CIFAR10 dataset with $p = 3,072$, $n = 2,000$, $\tau = 150$.



(b) MNIST dataset with $p = 784$, $n = 1,000$, $\tau = 50$.

Figure 5.3: Bayes excess risk curves for the CIFAR10 (top) and MNIST (bottom) datasets with different sets of parameters p, n, and τ. From left to right, the learning rate schedule $\eta_k = \eta/k^m$ for $m = 0, 1/4, 1/2$ with $\eta = 0.9/(n^{-1}\Lambda_1)$ is used. The purple line shows the estimated stopping time from (5.37), and the green line shows the empirical optimal stopping time.

5.6 Concluding remarks

In this chapter, we analyzed the discrete dynamics of gradient descent for linear regression with generic data and learning rate schedules. By determining expressions for the exact trajectory of the parameters, we proved various results that formalize the intuition that early stopping is similar to ℓ^2 regularization. Furthermore, we established general conditions on the learning rate and spectrum of the sample covariance matrix of the features that show whether early stopping is beneficial or not. Finally, we provided an estimate for the optimal stopping time, which we verified empirically.

An important direction for future work is to extend the results to the non-linear case. While the current work is limited to linear models, we believe that our approach can serve as a foundation for more complex models, such as multi-layer neural networks. For example, a simple case that may be tractable is the analysis of two-layer neural networks with layer-

wise training. Existing works such as [Mon+24; WWF24] describe how the spectrum of the features learned by the first layer evolves during training. Once the spectrum is known, our framework can be used to analyze the risk for the overall model by introducing two time parameters, corresponding to the number of steps each layer is trained for, which can then be optimized to minimize the risk. Understanding the dynamics of jointly training both layers is a significant and challenging open problem.

Another natural direction for future work is to analyze early stopping for other training algorithms, such as stochastic mini-batch gradient descent. Recent work [LSR25] shows that the dynamics of mini-batch gradient descent evolves in a way that is analogous to full-batch gradient descent, and depends on the spectrum of a modified cross-covariance matrix that encodes dependencies between the mini-batches. An interesting question is to quantify early stopping behavior in terms of the eigenvalues of this modified matrix.

Finally, our analysis shows that the optimal stopping time is, in general, different for each eigendirection of the covariance of the features. This is impossible to implement in practice with the usual gradient descent algorithm. However, it would be interesting to develop and study principled methods that augment the learning dynamics in the later stages in order to enhance the movement along the directions with longer stopping times.

5.7 Additional properties of the q-Pochhammer Symbol

In this section, our goal is to establish some technical facts about the q-Pochhammer symbol $(a;q)_n = \prod_{i=0}^{n-1}(1-aq^i)$. In particular, we would like to justify being able to take derivatives in n when analyzing the risk of gradient descent with an exponentially decaying learning rate schedule (see Proposition 5.2.8 and Assumption 2). Since $(a;q)_n$ is only defined for integer n so far, we first need to find an extension to the set of real numbers. Throughout this section, we shall assume that $|q| < 1$.

First, observe that we can write

$$(a;q)_x = \frac{(a;q)_\infty}{(aq^x;q)_\infty}. \tag{5.38}$$

Here we see that the right hand side is defined for all $x \in \mathbb{R}$. The main result of this section is the following, which shows that the extension (5.38) of the q-Pochhammer symbol is differentiable.

Proposition 5.7.1. *Let $|q| < 1$. The function*

$$x \mapsto (a;q)_x := \frac{(a;q)_\infty}{(aq^x;q)_\infty}$$

is differentiable in x, and the derivative is given by

$$\partial_x (a;q)_x = \frac{(a;q)_\infty}{(aq^x;q)_\infty^2} aq^x \log(q) \sum_{j=0}^{\infty} \frac{q^j}{1-aq^{x+j}} (aq^x;q)_\infty.$$

We shall build up towards the proof of Proposition 5.7.1 by proving some technical lemmas.

Lemma 5.7.2. *For $|q| < 1$, the function $a \mapsto (a;q)_\infty$ is continuous.*

Proof. Let $\varepsilon > 0$. We shall show continuity at a fixed point a_0. Fix $\delta > 0$, which we will choose later, and let a be any point such that $|a - a_0| < \delta$. Define

$$M := \max\left\{ \left(\max_{i,j=1,\ldots,\infty} \prod_{k=i}^{j} |(1-aq^k)| \right) \cdot \left(\max_{i,j=1,\ldots,\infty} \prod_{k=i}^{j} |(1-a_0 q^k)| \right), 1 \right\}.$$

Note that M is finite because $|q| < 1$, and therefore only finitely many terms $(1-aq^k)$, $(1-a_0q^k)$ have magnitude greater than one. We claim that if $|a - a_0| < \delta$, then

$$|(a;q)_n - (a_0;q)_n| \leq \delta M \sum_{k=0}^{n-1} |q|^k \tag{5.39}$$

for all n. Assuming that this is true for now, then by taking the limit as $n \to \infty$, we obtain

$$|(a;q)_\infty - (a_0;q)_\infty| \leq \frac{\delta M}{1-|q|}.$$

Hence, choosing $\delta < (1-|q|)\varepsilon/M$ implies that $|(a;q)_\infty - (a_0;q)_\infty| < \varepsilon$, as desired. To complete the proof, we will prove the claim (5.39). Note that

$$\begin{aligned}
&|(a;q)_{n+1} - (a_0;q)_{n+1}| \\
&\quad = |(a;q)_n(1-aq^n) - (a_0;q)_n(1-a_0q^n) - (a_0;q)_n(1-aq^n) + (a_0;q)_n(1-aq^n)| \\
&\quad \leq |1-aq^n||(a;q)_n - (a_0;q)_n| + |(a_0;q)_n||a-a_0||q|^n \\
&\quad \leq |1-aq^n||(a;q)_n - (a_0;q)_n| + \delta|(a_0;q)_n||q|^n \\
&\quad \leq |1-aq^n|\Big[|1-aq^{n-1}||(a;q)_{n-1} - (a_0;q)_{n-1}| + \delta|(a_0;q)_{n-1}||q|^{n-1}\Big] + \delta|(a_0;q)_n||q|^n \\
&\quad \leq \ldots \leq \delta \sum_{k=0}^{n} c_k|q|^k,
\end{aligned}$$

where each c_k is of the form $c_k = \left(\prod_{\ell=i_k}^{j_k} |(1-aq^\ell)|\right) \cdot \left(\prod_{\ell=i'_k}^{j'_k} |(1-a_0q^\ell)|\right)$. Thus, $c_k \leq M$ for all k, which implies the claimed result. □

Now that we have continuity, we want to bootstrap this to get differentiability. To do this, we shall need to prove the following lemma:

Lemma 5.7.3. *Let $|q| < 1$. Then, for all a, $(a;q)_n \to (a;q)_\infty$ locally uniformly as $n \to \infty$.*

Proof. Let $\varepsilon > 0$ and a_0 be any fixed point. We will prove uniform convergence in the closed ball $B(a_0,\varepsilon) := \{a : |a-a_0| \leq \varepsilon\}$. Let $M = \max_{a \in B(a_0,\varepsilon)} |a|$. For any $a \in B(a_0,\varepsilon)$, we have

$$|(a;q)_n - (a;q)_\infty| = |(a;q)_n||1-(aq^n;q)_\infty|.$$

By Lemma 5.7.2, the function $a \mapsto (a;q)_\infty$ is continuous at zero. Thus, there exists a number $\delta > 0$ such that if $|\widetilde{a}| < \delta$, then

$$|1 - (\widetilde{a};q)_\infty| \leq \frac{\varepsilon}{\max_k |(a;q)_k|}.$$

Note that $\max_k (a;q)_k$ is again finite since $|q| < 1$ and $|a| \leq M$. Thus, by choosing N such that $aq^N < \delta$, we deduce that for all $n \geq N$,

$$|(a;q)_n||1 - (aq^n;q)_\infty| \leq |(a;q)_n| \frac{\varepsilon}{\max_k |(a;q)_k|} \leq \varepsilon.$$

This establishes the uniform convergence of $(a;q)_n$ to $(a;q)_\infty$ on the closed ball $B(a_0, \varepsilon)$, as desired. □

Lemma 5.7.4. *If $|q| < 1$, then*

$$\partial_x (aq^x;q)_\infty = -aq^x \log(q) \sum_{k=0}^{\infty} \frac{q^k}{1 - aq^{x+k}} (aq^x;q)_\infty.$$

Proof. Since we have locally uniform convergence from Lemma 5.7.3, we can use the formula for the derivative of an infinite product of analytic functions to compute

$$\begin{aligned}
\partial_x (aq^x;q)_\infty &= \sum_{k=0}^{\infty} \partial_x (1 - aq^{x+k}) \prod_{j \neq k} (1 - aq^{x+j}) \\
&= -\sum_{k=0}^{\infty} aq^x \log(q) q^k \prod_{j \neq k} (1 - aq^{x+j}) \\
&= -aq^x \log(q) \sum_{k=0}^{\infty} q^k \prod_{j \neq k} (1 - aq^{x+j}) \\
&= -aq^x \log(q) \sum_{k=0}^{\infty} \frac{q^k}{(1 - aq^{x+k})} (aq^x;q)_\infty.
\end{aligned}$$

□

Proof of Proposition 5.7.1. The differentiability of $(a;q)_x$ in x and the formula for the derivative follow from Lemma 5.7.4. □

Chapter 6

Analysis of a randomly sparsified power method

This chapter is based on a joint work with Robert J. Webber and Jonathan Weare, currently in preparation.

6.1 Introduction

The problem of computing the leading eigenvalue and eigenvector of a square matrix $\mathbf{A}$ is a fundamental problem in numerical linear algebra. The extremely large scale of the matrices increasingly encountered in modern applications means that classical iterative approaches can be too expensive, even if $\mathbf{A}$ is sparse—for instance, the solution vector $\mathbf{x}$ itself may be too large to be stored.

In this chapter, we consider the problem of computing the leading eigenvector $\mathbf{v} \in [0,1]^n$ of a column-stochastic matrix $\mathbf{A} \in [0,1]^{n\times n}$ associated with the eigenvalue one, also known as the *Perron–Frobenius eigenvector* [Sen81]. We will assume throughout that the second largest-magnitude eigenvalue $\lambda_2(\mathbf{A})$ of $\mathbf{A}$ satisfies $|\lambda_2(\mathbf{A})| < 1$, i.e., $\mathbf{A}$ has a spectral gap. This implies that the eigenvector $\mathbf{v}$ satisfying $\mathbf{A}\mathbf{v} = \mathbf{v}$ is unique. We will assume that $\mathbf{v}$ is normalized to be a stochastic vector with entries summing to one. Equivalently, the eigen-

vector $\mathbf{v}$ represents the unique stationary distribution of a Markov chain with probability transition matrix given by $\mathbf{A}$. We note that while the Markov chain perspective is good for building intuition, it will not be needed in this work, which emphasizes a linear algebraic perspective that focuses on the sparsity properties of $\mathbf{A}$ and $\mathbf{v}$.

The *power method* is a classical iterative approach for computing the leading eigenvector $\mathbf{v}$, which, starting from any initial stochastic vector $\mathbf{x}_0 \in [0,1]^n$, has updates

$$\mathbf{x}_t = \mathbf{A}\mathbf{x}_{t-1}. \tag{6.1}$$

Under our assumption that $\mathbf{A}$ has a spectral gap, the power iterations converge geometrically to the unique leading eigenvector ([Sen81, Theorem 4.7]). A natural way to quantify the convergence rate of the power method for stochastic matrices is in terms of its contractivity with respect to the ℓ^1 norm.

Definition 6.1.1 (Contraction coefficients and mixing time)**.** For any column-stochastic matrix $\mathbf{A} \in [0,1]^{n\times n}$ and index $r \in \mathbb{N}$, define the r-step ℓ^1 *contraction coefficients:*

$$\alpha_r(\mathbf{A}) := \max_{\mathbf{z}\in\mathbb{R}^n,\, \sum_{i=1}^n \mathbf{z}(i)=0} \frac{\|\mathbf{A}^r\mathbf{z}\|_1}{\|\mathbf{z}\|_1} \in [0,1]. \tag{6.2}$$

Furthermore, define the *mixing time* of $\mathbf{A}$ by

$$\tau_{\mathrm{mix}}(\mathbf{A}) := \arg\min_{\tau\in\mathbb{N}} \left\{ \max_{i\in\{1,\ldots,n\}} \|\mathbf{A}^\tau \mathbf{e}_i - \mathbf{v}\|_1 \le \frac{1}{2} \right\}. \tag{6.3}$$

This is the first time that the total variation distance (which is equal to half the ℓ^1 norm) between the distribution of the Markov chain with probability transition matrix $\mathbf{A}$ and its stationary distribution $\mathbf{v}$ is less than $1/4$.

Since $\mathbf{A}^t\mathbf{x}_0 - \mathbf{v} = \mathbf{A}^t(\mathbf{x}_0 - \mathbf{v})$ and $\mathbf{1}^\mathsf{T}(\mathbf{x}_0 - \mathbf{v}) = 0$, where $\mathbf{1}$ is the all-ones vector,

$$\|\mathbf{A}^t\mathbf{x}_0 - \mathbf{v}\|_1 \le \alpha_t(\mathbf{A}) \cdot \|\mathbf{x}_0 - \mathbf{v}\|_1. \tag{6.4}$$

Some properties of the ℓ^1 contraction coefficients will be discussed later in Section 6.4.2. In particular, it can be shown that $\alpha_t(\mathbf{A}) \sim |\lambda_2(\mathbf{A})|^t$ as $t \to \infty$, i.e., the power method converges with rate $|\lambda_2(\mathbf{A})|$ asymptotically. Moreover, if R is an integer such that $\alpha_R(\mathbf{A}) \leq 1/4$, then $\tau_{\text{mix}}(\mathbf{A}) \leq R$ (see Proposition 6.4.3).

The goal of this chapter is to develop the theoretical foundations of two algorithms—one deterministic and the other randomized—that systematically impose sparsity in between each power iteration in order to mitigate the computational and storage costs. Our results will show that the sparsified methods with a relatively small user-chosen sparsity level are effective if the underlying power method is performant, in the sense that its ℓ^1 contraction coefficients $\alpha_r(\mathbf{A})$ decay quickly, and the leading eigenvector $\mathbf{v}$ is *approximately sparse*, in the sense that its entries decay rapidly.

6.1.1 Deterministically sparsified power method

The first approach that we will study is a deterministic scheme. Given a sparsification parameter $m \leq n$, let $\boldsymbol{\psi} : [0,1]^n \to [0,1]^n$ denote the *deterministic sparsification operator* that keeps the m largest entries of an input stochastic vector, zeros out all the other entries, and redistributes their mass evenly among the entries that are kept. (Any ties in choosing the largest entries can be broken arbitrarily.) Then, the iterates of the *deterministically sparsified power method* are given by

$$\begin{aligned} \mathbf{y}_t &= \mathbf{A}\mathbf{x}_{t-1}, \\ \mathbf{x}_t &= \boldsymbol{\psi}(\mathbf{y}_t). \end{aligned} \tag{6.5}$$

This scheme is optimal in the sense that $\boldsymbol{\psi}$ deterministically outputs a m-sparse stochastic vector that optimally controls the ℓ^p sparsification error for any $p \in [1, \infty]$; see Lemma 6.5.1.

Remark 6.1.2 (Why sparsify?)**.** Suppose that each column of $\mathbf{A}$ has at most q non-zero entries. Then the sparsified power iteration (6.5) requires $O(mq)$ operations, whereas each

standard power iteration (6.1) requires $O(nq)$ operations since the iterate $\mathbf{x}_t$ will rapidly become dense due to fill-in. Moreover, the cost of storing the iterate reduces from $O(n)$ to $O(m)$. Thus, the sparsified power method with a relatively small sparsification parameter m, ideally independent of or growing sublinearly with the underlying dimension n, has significantly lower computational and storage costs.

If $\mathbf{A}$ is *strictly contractive* in the sense that the one-step ℓ^1 contraction coefficient satisfies $\alpha_1(\mathbf{A}) < 1$, then we can prove the following convergence guarantee for the deterministically sparsified power method. The bound is composed of a geometrically decaying error and an irreducible component that is proportional to the ℓ^1 tail mass of the leading eigenvector $\mathbf{v}$.

Theorem 6.1.3 (Error with deterministic sparsification)**.** *Assume that* $\alpha_1(\mathbf{A}) < 1$. *Let* $\mathbf{v}^{\downarrow} \in [0,1]^n$ *be a weakly decreasing rearrangement of the leading eigenvector* $\mathbf{v}$ *with* $|\mathbf{v}^{\downarrow}(1)| \geq \ldots \geq |\mathbf{v}^{\downarrow}(n)|$. *Suppose that the sparsification parameter* $m \in \mathbb{N}$ *satisfies*

$$m \geq 2m_\star(\mathbf{A}), \quad \text{where} \quad m_\star(\mathbf{A}) := \frac{4\alpha_1(\mathbf{A})}{1-\alpha_1(\mathbf{A})}.$$

Then, for any time $t \geq 1$, *the iterate* $\mathbf{x}_t$ *after* t *steps of the deterministically sparsified power method* (6.5) *satisfies*

$$\|\mathbf{x}_t - \mathbf{v}\|_1 \leq \frac{6\alpha_1(\mathbf{A})}{1-\alpha_1(\mathbf{A})}\left(\frac{1+\alpha_1(\mathbf{A})}{2}\right)^t \|\mathbf{x}_0 - \mathbf{v}\|_1 + \frac{2(1+\alpha_1(\mathbf{A}))}{(1-\alpha_1(\mathbf{A}))^2} \sum_{i=\lceil m/m_\star(\mathbf{A})\rceil}^{n} \mathbf{v}^{\downarrow}(i).$$

The proof of Theorem 6.1.3 will be given in Section 6.5. The requirement for strict contractivity is very strong. A stochastic matrix $\mathbf{A}$ with a spectral gap is guaranteed to have $\alpha_R(\mathbf{A}) < 1$ for some $R \geq 1$, but it will generally not satisfy $\alpha_1(\mathbf{A}) < 1$.

A natural question is whether Theorem 6.1.3 can be generalized to allow for a contraction over multiple steps, i.e., $\alpha_R(\mathbf{A}) < 1$ for some $R > 1$. The following result shows that for general matrices $\mathbf{A}$ and initializations $\mathbf{x}_0$, this cannot be done in a meaningful way. Namely, for any dimension n, we can present a stochastic matrix $\mathbf{A}$ with *constant mixing time* and

constant spectral gap (i.e., it is very well connected globally), such that the deterministically sparsified power method, given a poor initialization, can remain stuck *with maximal ℓ^1 error*, even if an unreasonably large sparsification parameter $m \sim O(n)$ is used.

Theorem 6.1.4 (Failure mode of deterministic sparsification)**.** *Fix an integer $n \in \mathbb{N}$. Then, there exists a column-stochastic matrix $\mathbf{A}$ with leading eigenvector $\mathbf{v}$ that has one-step ℓ^1 contraction coefficient $\alpha_1(\mathbf{A}) = 1$ and spectral gap $1 - |\lambda_2(\mathbf{A})| \geq 1/3$, as well as a stochastic vector $\mathbf{x}_0 \in [0,1]^n$, such that for any sparsification parameter $m \leq n/2$, the iterates $\mathbf{x}_1, \mathbf{x}_2, \ldots$ of the deterministically sparsified power method (6.5) initialized at $\mathbf{x}_0$ satisfy*

$$\|\mathbf{x}_t - \mathbf{v}\|_1 = 2 \quad \text{for all } t = 0, 1, 2, \ldots.$$

The proof of Theorem 6.1.4 will also be given in Section 6.5. The main issue is that the one-step nature of deterministic sparsification can be foiled by local bottlenecks in the connectivity graph associated with $\mathbf{A}$, which can trap mass in a suboptimal solution and inhibit movement towards the leading eigenvector.

6.1.2 Randomly sparsified power method

The second approach that we will study is a randomized scheme. Let $\boldsymbol{\varphi} : \mathbb{R}^n \to \mathbb{R}^n$ be a *random sparsification operator*, parameterized by a user-chosen sparsification parameter $m \in \mathbb{N}$, whose output is (i) unbiased, i.e., $\mathbb{E}[\boldsymbol{\varphi}(\mathbf{x})] = \mathbf{x}$, and (ii) m-sparse, i.e., the number of non-zero entries $\|\boldsymbol{\varphi}(\mathbf{x})\|_0$ is at most m for all $\mathbf{x}$. Then, the iterates of the *randomly sparsified power method* are given by

$$\begin{aligned} \mathbf{y}_t &= \mathbf{A}\mathbf{x}_{t-1}, \\ \mathbf{x}_t &= \boldsymbol{\varphi}_t(\mathbf{y}_t), \end{aligned} \tag{6.6}$$

where each $\boldsymbol{\varphi}_t$ is an independent realization of the random sparsification operator $\boldsymbol{\varphi}$. In particular, we will choose $\boldsymbol{\varphi}$ to be the *pivotal sparsification operator*, which exactly preserves some of the largest-magnitude entries of the input vector, and randomly samples a subset

of the remaining entries to produce an unbiased approximation with m entries in total. We will defer a detailed description of pivotal sparsification to Section 6.4.1.

When the sparsification parameter $m = 1$, the iterates $\mathbf{x}_t$ produced by (6.6) are equivalent to a Markov chain with probability transition matrix $\mathbf{A}$. Hence, in the context of column-stochastic matrices, the randomly sparsified power method is a generalization of the *Markov chain Monte Carlo* (MCMC) sampler.

Because of the random sparsification, the iterates $\mathbf{x}_1, \mathbf{x}_2, \ldots$ are noisy. However, we can reduce the noise by averaging after some *burn-in time* $t_{\mathrm{b}} \geq 0$ and reporting the *tail-averaged iterate*:

$$\overline{\mathbf{x}}_t = \frac{1}{t - t_{\mathrm{b}}} \sum_{r=t_{\mathrm{b}}+1}^{t} \mathbf{x}_r, \quad t > t_{\mathrm{b}}. \tag{6.7}$$

Observe that the sparsity of the tail-averaged iterate $\overline{\mathbf{x}}_t$ is also controlled: $\|\overline{\mathbf{x}}_t\|_0 \leq m(t - t_{\mathrm{b}})$. If it is not even possible to store a full dense vector due to fill-in for $t \gg 1$, it is still possible to estimate low-dimensional projections of $\mathbf{v}$ by simply keeping track of the numbers $\mathbf{u}^* \mathbf{x}_r \in \mathbb{C}$ for any fixed vector $\mathbf{u} \in \mathbb{C}^n$, and reporting

$$\mathbf{u}^* \overline{\mathbf{x}}_t = \frac{1}{t - t_b} \sum_{r=t_b+1}^{t} \mathbf{u}^* \mathbf{x}_r. \tag{6.8}$$

Note that computing (6.8) aligns with the goals of many MCMC schemes, where the object of interest is typically an average with respect to the stationary distribution $\mathbf{v}$, instead of completely characterizing the vector itself.

Our main result for the randomly sparsified power method shows that if the sparsification parameter m is large enough, relative to the mixing time $\tau_{\mathrm{mix}}(\mathbf{A})$ of $\mathbf{A}$, then we can bound the error of the tail-averaged iterate by a bias component that decays geometrically in multiples of the mixing time, and a variance component that is proportional to the ℓ^1 tail mass of the leading eigenvector $\mathbf{v}$.

Theorem 6.1.5 (Error with random sparsification)**.** *Let* $\tau_{\mathrm{mix}} \equiv \tau_{\mathrm{mix}}(\mathbf{A})$ *be the mixing time of* $\mathbf{A}$ *and* $\mathbf{v}^{\downarrow} \in [0, 1]^n$ *be a weakly decreasing rearrangement of the leading eigenvector* $\mathbf{v}$ *with*

$|\mathbf{v}^{\downarrow}(1)| \geq \ldots \geq |\mathbf{v}^{\downarrow}(n)|$. *Suppose that the sparsification parameter* $m \in \mathbb{N}$ *satisfies*

$$m \geq 256\tau_{\mathrm{mix}}(\mathbf{A}).$$

Then, for any time t *after a burn-in time of* $t_{\mathrm{b}} = \lfloor t/2 \rfloor$*, the randomly sparsified power method* (6.6) *produces a tail-averaged iterate* $\overline{\mathbf{x}}_t = (t - t_{\mathrm{b}})^{-1} \sum_{r=t_{\mathrm{b}}+1}^{t} \mathbf{x}_r$ *that satisfies*

$$\mathbb{E}\|\overline{\mathbf{x}}_t - \mathbf{v}\|_2^2 \leq \frac{86\tau_{\mathrm{mix}}^2}{t^2}\left(1 + \frac{12t}{m}\right)\left(\frac{7}{8}\right)^{t/\tau_{\mathrm{mix}}}\|\mathbf{x}_0 - \mathbf{v}\|_1^2 + \frac{1{,}024\tau_{\mathrm{mix}}^2}{mt}\left(\sum_{i=\lceil m/2 \rceil}^{n} \mathbf{v}^{\downarrow}(i)\right)^2.$$

This result is a simplified version of Theorem 6.6.2, which is stated and proved in Section 6.6. The error bound in Theorem 6.1.5 is stated in terms of the L^2 error for simplicity. The full version implies that the same bound holds in terms of a stronger "triple norm" $|||\overline{\mathbf{x}}_t - \mathbf{v}|||$ (see Definition 6.6.1), which implies the following guarantee for any estimate $\mathbf{u}^*\overline{\mathbf{x}}_t$ of $\mathbf{u}^*\mathbf{v}$ as in (6.8):

$$\sqrt{\mathbb{E}\left|\mathbf{u}^*\overline{\mathbf{x}}_t - \mathbf{u}^*\mathbf{v}\right|^2} \leq \|\mathbf{u}\|_\infty |||\overline{\mathbf{x}}_t - \mathbf{v}|||.$$

Note that if $\alpha_1(\mathbf{A}) < 1$, then $\tau_{\mathrm{mix}}(\mathbf{A}) \leq \lceil \log(4)/\log(1/\alpha_1(\mathbf{A})) \rceil$; see Proposition 6.4.3 and (6.20). Therefore, if $\mathbf{A}$ is strictly contractive, we can compare Theorem 6.1.5 with our error bound for the deterministically sparsified power method given in Theorem 6.1.3. Although the bound with deterministic sparsification is in terms of the stronger ℓ^1 norm, the bound with random sparsification has a better irreducible component that exhibits an improved dependence on the ℓ^1 tail mass of the leading eigenvector (i.e., it is deeper and smaller by a factor of $m^{-1/2}$), and reflects the possibility for further improvement with rate $t^{-1/2}$ by tail-averaging.

More importantly, Theorem 6.1.5 applies far more generally. For example, it implies that the randomly sparsified power method rapidly converges for the counterexample described in Theorem 6.1.4 for the deterministically sparsified power method, which is not strictly contractive (see Remark 6.5.5).

6.1.3 Beyond-Monte Carlo rates

Intuitively, we would expect the sparsified iterations to be effective if the leading eigenvector $\mathbf{v}$ is approximately sparse, in the sense that its entries decay rapidly. For example, if $\mathbf{v}$ were truly m-sparse, then it would indeed remain a fixed point of the (deterministically or randomly) sparsified iterations. To make this intuition more precise, we can make some idealized assumptions on the decay of the entries of $\mathbf{v}$ to derive the consequences on the convergence rate implied by our bounds. We will write $\mathbf{v}^{\downarrow} \in [0,1]^n$ to denote a weakly decreasing rearrangement of $\mathbf{v}$ such that $\mathbf{v}^{\downarrow}(1) \geq \mathbf{v}^{\downarrow}(2) \geq \ldots \geq \mathbf{v}^{\downarrow}(n)$.

1. *(Exponential decay).* If $\mathbf{v}^{\downarrow}(i) \leq Ce^{-ci}$ for some constants $C, c > 0$, then

$$\sum_{i=s}^{n} \mathbf{v}^{\downarrow}(i) \leq \sum_{i=s}^{n} Ce^{-ci} \leq \frac{Ce^{-cs}}{1-e^{-c}}.$$

2. *(Polynomial decay).* If $\mathbf{v}^{\downarrow}(i) \leq Ci^{-(1+c)}$ for some constants $C, c > 0$, then

$$\sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i) \leq \sum_{i=s+1}^{n} Ci^{-(1+c)} \leq C \int_{s}^{\infty} x^{-(1+c)} \, \mathrm{d}x = \frac{C}{c} s^{-c}.$$

By applying these calculations with $s = \lceil m/2 \rceil$ in Theorem 6.1.5, we see that the randomly sparsified power method can achieve *beyond-Monte Carlo rates* as a function of the sparsification parameter m based on the ℓ^1 tail decay of the leading eigenvector $\mathbf{v}$, i.e., faster than $O(m^{-1/2})$. More precisely, after a sufficiently long burn-in period $t_{\mathrm{b}} = O(\tau_{\mathrm{mix}}(\mathbf{A}))$ such that the bias component is negligible, the error $\left|\left|\left|\overline{\mathbf{x}}_t - \mathbf{v}\right|\right|\right|$ can scale as $O(e^{-cm})$ or $O(m^{-(1/2+c)})$ for some constant $c > 0$ if the entries of $\mathbf{v}^{\downarrow}$ decay exponentially or polynomially, respectively.

6.1.4 Numerical demonstration with the Ising model

Consider the ferromagnetic Ising model on a graph $G = (V, E)$ with inverse temperature $\beta > 0$ and external field $h \in \mathbb{R}$. This is a probability measure on each *configuration* $\sigma \in$

$\{\pm 1\}^V$, which is an assignment of spins from $\{+1, -1\}$ to every vertex, defined by

$$\mu_{\beta,h}(\sigma) = \frac{e^{-\beta H(\sigma)}}{Z(\beta, h)}, \quad \text{where} \quad H(\sigma) = -\sum_{(u,v)\in E} \sigma(u)\sigma(v) - h\sum_{v\in V} \sigma(v) \tag{6.9}$$

is the Hamiltonian, and $Z(\beta, h)$ is the normalization constant known as the partition function, which is computationally intractable.

The (heat-bath) *Glauber dynamics* is a classical dynamical system for simulating the Ising model that is studied in non-equilibrium statistical mechanics [Gla63] and related contexts [Lig85; CFL09]. From any given configuration $\sigma \in \{\pm 1\}^V$, the next configuration σ' is sampled by choosing a vertex $v \in V$ uniformly at random, and sampling a new spin $\sigma'(v) \in \{\pm 1\}$ from the Ising model (6.9), conditional on all other vertices being fixed. Each update is extremely local and easily computable. The Glauber dynamics is an irreducible, aperiodic, and reversible Markov chain that has the Ising model (6.9) as its unique stationary distribution (e.g., see [LPW17]). For additional background on the Ising model, see Section 6.8.

As a test of our theory, we will numerically investigate the performance of the deterministically and randomly sparsified power methods for solving the eigenvalue problem $\mathbf{A}\mathbf{v} = \mathbf{v}$, where $\mathbf{A}$ is the column-stochastic transition matrix corresponding to the Glauber dynamics on an $\ell \times \ell$ lattice with nearest-neighbor interactions and periodic boundary conditions (i.e., torus). The leading eigenvector $\mathbf{v}$ corresponds to the Ising model (6.9), and expectations of quantities with respect to the Ising measure (e.g., magnetization or correlations) can be estimated by estimates of the form $\mathbf{u}^*\overline{\mathbf{x}}_t$ as in (6.8). Note that $\mathbf{A}$ is very sparse: each column has $\ell^2 + 1$ non-zero entries, significantly smaller than the size of the state space $2^{\ell\times\ell}$, which grows exponentially in the number of vertices. For example, on a 2×2 torus, the 16×16

matrix $\mathbf{A}$ with each configuration indexed by its bitstring looks like

$$\mathbf{A} = \begin{pmatrix}
\star & p_-^{(-2)} & p_-^{(-2)} & & p_-^{(-2)} & & & & p_-^{(-2)} & & & & & & & \\
p_+^{(-2)} & \star & & p_-^{(0)} & & p_-^{(-2)} & & & & p_-^{(0)} & & & & & & \\
p_+^{(-2)} & & \star & p_-^{(0)} & & & p_-^{(0)} & & & & p_-^{(-2)} & & & & & \\
& p_+^{(0)} & p_+^{(0)} & \star & & & & p_-^{(0)} & & & & p_-^{(0)} & & & & \\
p_+^{(-2)} & & & & \star & p_-^{(-2)} & p_-^{(0)} & & & & & & p_-^{(0)} & & & \\
& p_+^{(-2)} & & & p_+^{(-2)} & \star & & p_-^{(2)} & & & & & & p_-^{(2)} & & \\
& & p_+^{(0)} & & p_+^{(0)} & & \star & p_-^{(0)} & & & & & & & p_-^{(0)} & \\
& & & p_+^{(0)} & & p_+^{(2)} & p_+^{(0)} & \star & & & & & & & & p_-^{(2)} \\
p_+^{(-2)} & & & & & & & & \star & p_-^{(0)} & p_-^{(-2)} & & p_-^{(0)} & & & \\
& p_+^{(0)} & & & & & & & p_+^{(0)} & \star & & p_-^{(0)} & & p_-^{(0)} & & \\
& & p_+^{(-2)} & & & & & & p_+^{(-2)} & & \star & p_-^{(2)} & & & p_-^{(2)} & \\
& & & p_+^{(0)} & & & & & & p_+^{(0)} & p_+^{(2)} & \star & & & & p_-^{(2)} \\
& & & & p_+^{(0)} & & & & p_+^{(0)} & & & & \star & p_-^{(0)} & p_-^{(0)} & \\
& & & & & p_+^{(2)} & & & & p_+^{(0)} & & & p_+^{(0)} & \star & & p_-^{(2)} \\
& & & & & & p_+^{(0)} & & & & p_+^{(2)} & & p_+^{(0)} & & \star & p_-^{(2)} \\
& & & & & & & p_+^{(2)} & & & & p_+^{(2)} & & p_+^{(2)} & p_+^{(2)} & \star
\end{pmatrix}.$$

Here, $p_+^{(\ell)} = (1+e^{-2\beta(\ell+h)})^{-1}/4$ and $p_-^{(\ell)} = (1+e^{2\beta(\ell+h)})^{-1}/4$ for $\ell \in \{-2, 0, 2\}$, corresponding to the sum of the neighboring spins of the vertex to be updated, and the diagonal entries are left implicitly defined such that $\mathbf{A}$ is column-stochastic.

Figure 6.1 presents the numerical results for the Ising model on a 4×4 torus with inverse temperature $\beta = 0.45$ and external field $h = 0.25$. The size of the grid is chosen for computational simplicity, since the size of the state space ($2^{4\times 4} = 65,536$) is non-trivial, yet small enough to handle explicitly. The parameters β and h are chosen to situate the Ising model in a low-temperature and strong external field regime, which speeds up the mixing of the underlying Glauber dynamics and promotes approximate sparsity of the Ising measure $\mathbf{v}$ (i.e., such that the entries $\mathbf{v}^{\downarrow}(i) \leq Ci^{-(1+c)}$ exhibit polynomial decay; see Proposition 6.8.1 in Section 6.8 for a precise statement).

The left plot shows that the ℓ^2 error of the randomly sparsified power method decreases at a beyond-Monte Carlo rate in m that follows the shape of the best m-sparse approximation of the leading eigenvector $\mathbf{v}$ and improves with tail-averaging, as predicted by Theorem 6.1.5. The deterministically sparsified power method, for which we do not have any guarantees in this case since $\mathbf{A}$ is not strictly contractive, is much less effective for this problem. The

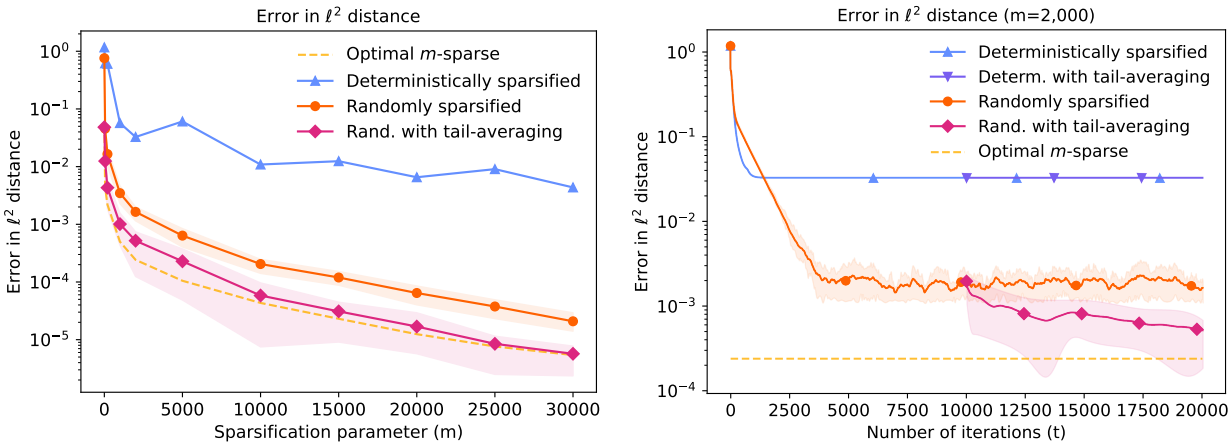


Figure 6.1: The ℓ^2 error $\|\mathbf{x}_t - \mathbf{v}\|_2$ from solving the eigenvalue problem $\mathbf{A}\mathbf{v} = \mathbf{v}$ corresponding to the Glauber dynamics for the Ising model (6.9) on a 4×4 torus in a low-temperature and strong external field regime with $\beta = 0.45$ and $h = 0.25$ using the deterministically and randomly sparsified power methods (with and without tail-averaging), initialized from a random configuration. The optimal m-sparse error represents the ℓ^2 error $\left(\sum_{i=m+1}^{n} \mathbf{v}^{\downarrow}(i)^2\right)^{1/2}$ from the best m-sparse approximation of the Ising model. **(Left)** The errors at time $t = 20,000$ after a sufficiently long burn-in time of $t_{\mathrm{b}} = 10,000$ as a function of the sparsification parameter m. **(Right)** The dynamics of the error with a fixed sparsification parameter $m = 2,000$. The mean errors over 30 independent runs of the randomized algorithms are reported, with the corresponding $0.2/0.8^{\mathrm{th}}$ quantiles indicated by the shaded intervals.

right plot shows the dynamics of the ℓ^2 error for $m = 2,000$ as a function of the number of iterations t, confirming that tail-averaging helps the randomly sparsified power method to achieve lower errors with further iterations. However, it does not help the deterministically sparsified power method since the solution vector is stuck in a suboptimal solution.

6.1.5 Outline

The rest of this chapter is structured as follows. Section 6.3 describes another application of the deterministically and randomly sparsified power methods for the PageRank problem. In Section 6.4, mathematical preliminaries related to the pivotal sparsification operator and ergodic coefficients for stochastic matrices are presented. The proofs of our main results are provided in Sections 6.5 and 6.6 for the deterministically and randomly sparsified power

methods, respectively. We present some concluding remarks in Section 6.7. Additional technical details are presented in Sections 6.8 and 6.9.

6.1.6 Notation

We will write vectors $\mathbf{v} \in \mathbb{R}^n$ and matrices $\mathbf{A} \in \mathbb{R}^{n \times n}$ in boldface, and denote their entries by $\mathbf{v}(i)$ and $\mathbf{A}(i,j)$. We write $\mathbf{I} \in \mathbb{R}^{n \times n}$ for the identity matrix, $\mathbf{e}_i \in \mathbb{R}^n$ for the i^{th} standard basis vector in $\mathbb{R}^n$, and $\mathbf{1}$ for the all-ones vector. We will use the vector ℓ^1, ℓ^2, and ℓ^∞ norms given by $\|\mathbf{v}\|_1 = \sum_{i=1}^n |\mathbf{v}(i)|$, $\|\mathbf{v}\|_2 = \sqrt{\sum_{i=1}^n |\mathbf{v}(i)|^2}$, and $\|\mathbf{v}\|_\infty = \max_{i=1,\ldots,n} |\mathbf{v}(i)|$, respectively. We will use the notation $\|\mathbf{v}\|_0$ to count the number of non-zero entries of a vector $\mathbf{v}$. We will also write $\mathbf{v}^\downarrow \in \mathbb{R}^n$ to denote any weakly decreasing rearrangement of a vector $\mathbf{v}$ such that $|\mathbf{v}^\downarrow(1)| \geq |\mathbf{v}^\downarrow(2)| \geq \ldots \geq |\mathbf{v}^\downarrow(n)|$. We will use the matrix ℓ^1 operator norm $\|\mathbf{A}\|_1 = \max_{\|\mathbf{z}\|_1=1} \|\mathbf{A}\mathbf{z}\|_1 = \max_{i=1,\ldots,n} \|\mathbf{A}\mathbf{e}_i\|_1$.

6.2 Related works

6.2.1 Deterministically sparsified power method

The deterministically sparsified power method is most closely related to the truncated power method analyzed by Yuan and Zhang in [YZ13], which also preserves the m largest-magnitude entries after each iteration. It is also similar to the power method with iterative hard thresholding studied in [Ma13], which uses a fixed threshold designed to filter out noise rather than an adaptive threshold for maintaining a fixed sparsity. The main difference is that these iterative algorithms, which are proposed for the problem of sparse principal component analysis (PCA) [ZHT06; ZX18], are studied from a statistically-oriented perspective, focusing on the statistical consistency and rates of the procedure for the recovery of an underlying sparse eigenvector. Our analysis focuses on the implications of maintaining a user-chosen sparsity level for computational reasons on the convergence rate.

6.2.2 Randomly sparsified power method

The randomly sparsified power method is an instance of a general approach based on imposing sparsity in between each update of an iterative method called *fast randomized iteration* (FRI), which was proposed by Lim and Weare [LW17] to mitigate the computational and storage costs for large-scale numerical linear algebra problems. These algorithms are motivated by the success of diffusion Monte Carlo algorithms, which have been applied to solve eigenproblems as large as $10^{108} \times 10^{108}$ [She+12].

Soon after, the research groups of Weare and Berkelbach developed more intricate versions of the randomly sparsified power method and applied them to solve large-scale eigenvalue problems in quantum chemistry [Gre+19; Gre+20; Gre+22a; Gre+22b]. However, these works leave open a range of mathematical questions, including:

- What is the smallest possible sparsification parameter m needed so that the randomized iteration is stable?
- What is the effect of increasing the sparsification parameter m on the convergence rate?
- What can be proved for a deterministically sparsified power method, where a deterministic sparsification operator is applied in each iteration, and how does it compare to the randomly sparsified case?

In this chapter, we provide mathematical analysis of the randomly and deterministically sparsified power methods for stochastic matrices that answer these questions.

More recently, Weare and Webber [WW25] analyzed a *randomly sparsified Richardson iteration* for solving linear systems $\mathbf{A}\mathbf{x} = \mathbf{b}$. Based on the observation that the solution of the linear system satisfies the fixed-point formula $\mathbf{x} = \mathbf{G}\mathbf{x} + \mathbf{b}$, where $\mathbf{G} := \mathbf{I} - \mathbf{A}$, the

randomly sparsified Richardson iteration begins with $\mathbf{x}_0 = \mathbf{0}$ and iterates:

$$\begin{aligned} \mathbf{y}_t &= (\mathbf{I} - \mathbf{A})\mathbf{x}_{t-1} + \mathbf{b}, \\ \mathbf{x}_t &= \boldsymbol{\varphi}_t(\mathbf{y}_t). \end{aligned} \tag{6.10}$$

The authors prove that if $\mathbf{G}$ is a strict 1-norm contraction ($\|\mathbf{G}\|_1 < 1$) and a sufficiently large sparsification parameter $m \geq 1/(1 - \|\mathbf{G}\|_1^2)$ is used, then the error of the randomly sparsified Richardson iteration can decay at a faster-than-Monte Carlo rate—e.g., polynomial m^{-c} or exponential e^{-cm} in m for some constant $c > 0$—for problems where the entries of the solution vector decrease quickly ([WW25, Theorem 2.1]). In our main result for the randomly sparsified power method (Theorem 6.1.5), we establish a similar result for eigenvalue problems.

6.3 Application to PageRank

Consider the PageRank problem [Gle15]: given a directed graph on n vertices $\{1, 2, \ldots, n\}$, a column-stochastic transition matrix $\mathbf{Q} \in [0, 1]^{n \times n}$, and a stochastic vector $\mathbf{r} \in [0, 1]^n$, a random walker flips a biased coin at each time step. If the coin lands heads with probability $\theta \in [0, 1]$, the walker travels from its current state i to an outgoing neighbor with probability $\mathbf{Q}(j, i)$. Otherwise, if the coin lands tails with probability $1 - \theta$, the walker travels to a node randomly selected from the renewal distribution $\mathbf{r}$.

We are interested in the long-run behavior of the random walker, which is given by the stationary distribution $\mathbf{v} \in [0, 1]^n$ that solves

$$\mathbf{A}\mathbf{v} = \mathbf{v}, \quad \text{where} \quad \mathbf{A} := \theta\mathbf{Q} + (1 - \theta)\mathbf{r}\mathbf{1}^\mathsf{T}. \tag{6.11}$$

In other words, $\mathbf{v}$ is the Perron–Frobenius eigenvector of the column-stochastic matrix $\mathbf{A}$. If $\theta \in [0, 1)$, then a unique stationary distribution $\mathbf{v}$ always exists [Gle15, §2]. Otherwise, if $\theta =$

1, then a unique stationary distribution exists if the graph is irreducible and aperiodic [Sen81, §1].

The PageRank problem was originally proposed by Google to find the most relevant websites in response to a search query [Pag+99]. The entries of the PageRank vector $\mathbf{v}$ can be used as a centrality measure for the importance of nodes in the graph: intuitively, nodes that are "important" have incoming edges from other "important" nodes. In particular, if $\theta < 1$ and $\mathbf{r}$ is sparse, the PageRank vector $\mathbf{v}$ conveys localized information about a region of the graph corresponding to the support of $\mathbf{r}$, and the problem is typically referred to as *personalized PageRank*. The PageRank problem has found numerous other applications in the analysis of social, physical, and information networks. For a more detailed discussion, we refer to the survey [Gle15].

6.3.1 Computing the PageRank vector

We may compute the stationary distribution $\mathbf{v}$ by applying the power method (6.1) to solve the eigenvalue problem (6.11). Starting from any initial stochastic vector $\mathbf{x}_0 \in [0,1]^n$ (e.g., $\mathbf{x}_0 = \mathbf{r}$), we iterate:

$$\mathbf{x}_t = \mathbf{A}\mathbf{x}_{t-1} = \theta\mathbf{Q}\mathbf{x}_{t-1} + (1-\theta)\mathbf{r}. \tag{6.12}$$

Note that $\mathbf{A}\mathbf{z} = \theta\mathbf{Q}\mathbf{z} + (1-\theta)\mathbf{r}\mathbf{1}^\mathsf{T}\mathbf{z} = \theta\mathbf{Q}\mathbf{z}$ for any $\mathbf{z}$ with $\mathbf{1}^\mathsf{T}\mathbf{z} = 0$. Therefore, we can bound the one-step ℓ^1 contraction coefficient by the parameter θ:

$$\alpha_1(\mathbf{A}) = \max_{\substack{\sum_{i=1}^n \mathbf{z}(i)=0 \\ \|\mathbf{z}\|_1=1}} \|\mathbf{A}\mathbf{z}\|_1 = \theta \cdot \max_{\substack{\sum_{i=1}^n \mathbf{z}(i)=0 \\ \|\mathbf{z}\|_1=1}} \|\mathbf{Q}\mathbf{z}\|_1 = \theta \cdot \alpha_1(\mathbf{Q}) \leq \theta. \tag{6.13}$$

From (6.4), combined with the submultiplicativity of the ℓ^1 contraction coefficients (Proposition 6.4.3), this immediately furnishes the following convergence guarantee for solving the PageRank problem using the power method (6.12):

$$\|\mathbf{x}_t - \mathbf{v}\|_1 \leq \theta^t \cdot \|\mathbf{x}_0 - \mathbf{v}\|_1. \tag{6.14}$$

We may also compute the PageRank eigenvector using the deterministically sparsified power method (6.5):

$$\begin{aligned} \mathbf{y}_t &= \mathbf{A}\mathbf{x}_{t-1} = \theta \mathbf{Q}\mathbf{x}_{t-1} + (1-\theta)\mathbf{r}, \\ \mathbf{x}_t &= \boldsymbol{\psi}(\mathbf{y}_t). \end{aligned} \tag{6.15}$$

If $\theta < 1$, then we can bound the error $\|\mathbf{x}_t - \mathbf{v}\|_1$ by applying Theorem 6.1.3 with $\alpha_1(\mathbf{A}) \leq \theta$.

Alternatively, we can compute the PageRank vector using the randomly sparsified power method (6.6):

$$\begin{aligned} \mathbf{y}_t &= \mathbf{A}\mathbf{x}_{t-1} = \theta \mathbf{Q}\mathbf{x}_{t-1} + (1-\theta)\mathbf{r}, \\ \mathbf{x}_t &= \boldsymbol{\varphi}_t(\mathbf{y}_t). \end{aligned} \tag{6.16}$$

If $\theta < 1$, then we can also obtain a bound on the error $\mathbb{E}\|\overline{\mathbf{x}}_t - \mathbf{v}\|_2^2$ by applying Theorem 6.1.5 with $\tau_{\text{mix}}(\mathbf{A}) \leq \lceil \log(4)/\log(1/\theta) \rceil$. However, if $\theta = 1$, we can also bound the error using Theorem 6.1.5 in terms of the mixing time of the random walk on the graph.

6.3.2 Numerical demonstration

We consider solving the personalized PageRank problem on the largest strongly connected component from the `twitter_combined` dataset [ML12], which consists of a social network where there is a directed edge from vertex A to vertex B if user A follows user B. The graph is irreducible, aperiodic, and has $n = 68,413$ vertices and $1,685,163$ edges. We construct a 10-sparse renewal distribution $\mathbf{r}$ by placing mass equally over 10 vertices chosen uniformly at random.

Figure 6.2 shows the ℓ^2 error from applying the deterministically and randomly sparsified power methods using various sparsification parameters m for the PageRank problem with $\theta \in \{0.85, 1\}$. For each θ, the burn-in time was chosen to be large enough for the random errors to stabilize, and the final tail-averaged iterate $\overline{\mathbf{x}}_t$ is averaged over the last $1,000$ iterations.

The plot on the left shows the results with $\theta = 0.85$, which adds a significant amount of regularization that makes the power iterations much more effective. In particular, Theo-

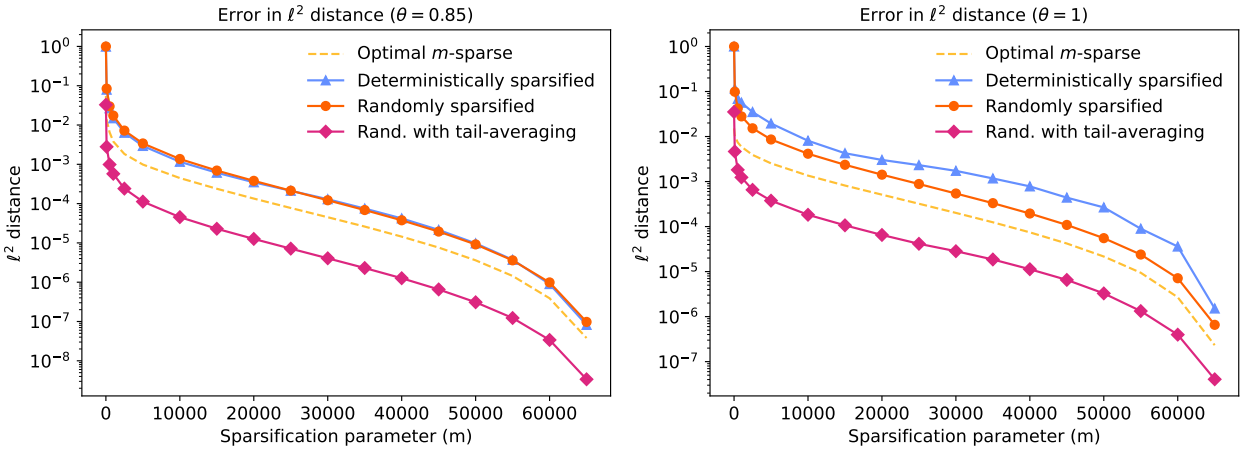


Figure 6.2: The ℓ^2 error $\|\mathbf{x}_t - \mathbf{v}\|_2$ from solving the personalized PageRank problem (6.11) on the `twitter_combined` dataset using the deterministically and randomly sparsified power methods (with and without tail-averaging), initialized from $\mathbf{x}_0 = \mathbf{r}$. The optimal m-sparse error represents the ℓ^2 error $\left(\sum_{i=m+1}^{n} \mathbf{v}^{\downarrow}(i)^2\right)^{1/2}$ from the best m-sparse approximation of the leading eigenvector. **(Left)** The errors with $\theta = 0.85$ at $t = 1,100$ after a burn-in time of $t_{\mathrm{b}} = 100$. **(Right)** The errors with $\theta = 1$ at $t = 3,000$ after a burn-in time of $t_{\mathrm{b}} = 2,000$. The mean errors over 10 independent runs of the randomized algorithms are reported.

rems 6.1.3 and 6.1.5 apply since $\mathbf{A}$ is strictly contractive, and we observe that deterministic and random sparsification are both effective and perform equally well in this setting, producing an error after a sufficiently long burn-in period close to the optimal m-sparse error. However, random sparsification allows for further improvement from tail-averaging, in contrast to deterministic sparsification which produces a fixed point that cannot be improved with tail-averaging.

The plot on the right shows the results with $\theta = 1$, where the power method is less effective and its convergence properties depend on the mixing time of the random walk on the underlying graph. Since $\mathbf{A}$ is not strictly contractive, we do not have any guarantees for the deterministically sparsified power method, but we can apply Theorem 6.1.5 to bound the error of the randomly sparsified power method. We observe a greater separation between the deterministically and randomly sparsified power methods, illustrating how deterministic sparsification can result in mass being trapped in a suboptimal solution.

6.3.3 Comparison with randomly sparsified Richardson iteration

Note that computing the leading eigenvector $\mathbf{v}$ of the PageRank problem (6.11) with $\theta < 1$ is equivalent to finding a solution of the following linear system (see [Gle15]):

$$(\mathbf{I} - \theta\mathbf{Q})\mathbf{v} = (1-\theta)\mathbf{r}. \tag{6.17}$$

The randomly sparsified Richardson iteration [WW25], applied to solving this linear system with zero initial vector $\mathbf{x}_0 = \mathbf{0}$ instead, results in the same iterations as the randomly sparsified power method (6.12). Therefore, Theorem 6.1.5 can be compared with the convergence bound obtained in [WW25, Proposition 3.1], which produces qualitatively similar predictions. In practice, initializing with the stochastic vector $\mathbf{x}_0 = \mathbf{r}$ is preferable ([Gle15, Remark 2.3]), in which case only our results on the power iterations are applicable. Moreover, if $\theta = 1$ and the matrix $\mathbf{A}$ is not strictly contractive, only Theorem 6.1.5 can be applied.

6.4 Preliminaries

In this section, we discuss some mathematical preliminaries for the pivotal sparsification operator used in the randomly sparsified power method, and ergodic coefficients for stochastic matrices.

6.4.1 Pivotal sparsification

For the randomly sparsified power method (6.6), we will use the *pivotal sparsification operator* $\boldsymbol{\varphi} : \mathbb{R}^n \to \mathbb{R}^n$ with sparsification parameter m. This procedure exactly preserves some of the largest-magnitude entries of the input vector based on an adaptive threshold, and randomly samples (without replacement) and rescales a subset of the remaining entries to produce an unbiased approximation with m entries in total.

More precisely, for any input vector $\mathbf{y} \in \mathbb{R}^n$, let $\mathbf{y}^{\downarrow} \in \mathbb{R}^n$ denote a rearrangement of the entries of $\mathbf{y}$ with weakly decreasing magnitudes, $|\mathbf{y}^{\downarrow}(1)| \geq |\mathbf{y}^{\downarrow}(2)| \geq \ldots \geq |\mathbf{y}^{\downarrow}(n)|$, and σ be the corresponding permutation such that $\mathbf{y}(\sigma(i)) = \mathbf{y}^{\downarrow}(i)$ for all $i \in \{1, 2, \ldots, n\}$. Then, pivotal sparsification $\mathbf{y} \mapsto \boldsymbol{\varphi}(\mathbf{y})$ is applied as follows:

(1) Determine a threshold

$$q_* = \min\left\{0 \leq q \leq m : |\mathbf{y}^{\downarrow}(q+1)| < \frac{1}{m-q} \sum_{i=q+1}^{n} |\mathbf{y}^{\downarrow}(i)|\right\}.$$

(2) Calculate a vector of inclusion probabilities $\mathbf{p} \in [0,1]^n$ by the formula

$$\begin{aligned} \mathbf{p}(i) &= 1 && \text{if } \sigma(i) \leq q_*, \\ \mathbf{p}(i) &= \frac{m - q_*}{\sum_{j > q_*} |\mathbf{y}^{\downarrow}(j)|} |\mathbf{y}(i)| < 1 && \text{if } \sigma(i) > q_*. \end{aligned}$$

(3) Sample a subset $\mathsf{K} \subseteq \{1, 2, \ldots, n\}$ with exactly m entries based on the inclusion probabilities $\mathbf{p}$: that is, $\mathbb{P}\{i \in \mathsf{K}\} = \mathbf{p}(i)$ for all $i \in \{1, 2, \ldots, n\}$.

(4) Set $\boldsymbol{\varphi}(\mathbf{y})(i) = \mathbf{y}(i)/\mathbf{p}(i)$ for each entry $i \in \mathsf{K}$ that is kept, and set $\boldsymbol{\varphi}(\mathbf{y})(i) = 0$ otherwise.

Remark 6.4.1 (Implementation)**.** The threshold q_* for Step (1) can be computed in $O(\|\mathbf{y}\|_0 \log m)$ time in a single pass by finding the top m entries of $\mathbf{y}$ using a min-heap. Alternatively, it can be done in $O(\|\mathbf{y}\|_0 + q_* \log m)$ average time using a selection method based on QuickSelect.

The subset K in Step (3) can be computed using *pivotal sampling* [DT98], which can be implemented using a single pass through the non-zero entries of $\mathbf{y}$ in $O(\|\mathbf{y}\|_0)$ operations; see [WW25, Algorithm 5.2] for a pseudocode. For additional discussion on the practical implementation of pivotal sparsification, including strategies for parallelization, see [Gre+22a, Appendix A].

As the most important feature, pivotal sparsification leads to a high-accuracy approximation whenever the entries $|\mathbf{y}^{\downarrow}(i)|$ decrease rapidly in magnitude. The following explicit bound was proved in [WW25, Proposition 5.2]:

$$\mathbb{E}\|\boldsymbol{\varphi}(\mathbf{y}) - \mathbf{y}\|_2^2 \leq \min_{0 \leq s \leq m} \frac{1}{m-s} \left(\sum_{i=s+1}^{n} |\mathbf{y}^{\downarrow}(i)| \right)^2. \tag{6.18}$$

In fact, it was also established in [WW25] that pivotal sparsification is the optimal unbiased, m-sparse sparsification scheme in terms of L^2 error.

In order to control the variance of the randomly sparsified power method, we will use the following bound on the triple norm error of pivotal sparsification for vectors with non-negative entries from [WW25]:

Lemma 6.4.2 ([WW25, Corollary 5.5])**.** *Let $\boldsymbol{\varphi}$ be the pivotal sparsification operator. Then for all non-negative valued vectors $\mathbf{y} \in \mathbb{R}_+^n$ and subsets $\mathsf{E} \subseteq \{1, \ldots, n\}$, it holds:*

$$\left|\!\left|\!\left| \mathbf{y} - \boldsymbol{\varphi}(\mathbf{y}) \right|\!\right|\!\right|^2 = \max_{\|\mathbf{u}\|_\infty \leq 1} \mathbb{E} \left| \mathbf{u}^* \big(\mathbf{y} - \boldsymbol{\varphi}(\mathbf{y}) \big) \right|^2 \leq \frac{1}{m - |\mathsf{E}|} \left(\sum_{i \notin \mathsf{E}} \mathbf{y}(i) \right)^2.$$

6.4.2 Stochastic matrices

Recall that the matrix $\mathbf{A} \in [0,1]^{n \times n}$ represents a column-stochastic matrix throughout the chapter, which has leading eigenvalue $\lambda_1(\mathbf{A}) = 1$ with corresponding left eigenvector $\mathbf{1}^\mathsf{T} = \mathbf{1}^\mathsf{T}\mathbf{A}$ and a right eigenvector $\mathbf{v} \in [0,1]^n$ by Perron–Frobenius theory [Sen81]. It is beneficial to identify $\mathbf{A}$ as the probability transition matrix of an associated Markov chain [LPW17], so that $\mathbf{v}$ represents a stationary distribution of $\mathbf{A}$. We also assume that $\mathbf{A}$ exhibits a spectral gap: i.e., the second-largest magnitude eigenvalue $\lambda_2(\mathbf{A})$ satisfies $|\lambda_2(\mathbf{A})| < 1$. A typical structural condition that guarantees this is that the matrix $\mathbf{A}$ is *irreducible* and *aperiodic*; see, e.g., [Sen81, §1] for the precise definitions of these terms related to the classification of non-negative matrices.

Let $\mathbf{P} := \mathbf{I} - \mathbf{v}\mathbf{1}^\mathsf{T}$ be the oblique projector onto $\{\mathbf{x} : \mathbf{1}^\mathsf{T}\mathbf{x} = 0\}$ along $\mathbf{v}$ that annihilates the leading left and right eigenvectors of $\mathbf{A}$. Observe that $\mathbf{P}$ commutes with $\mathbf{A}$, and $\mathbf{1}^\mathsf{T}\mathbf{w} = 0$ for any eigenvector $\mathbf{w}$ of $\mathbf{A}$ corresponding to eigenvalue $\lambda \neq 1$. Moreover, note that for any index $r \in \mathbb{N}$, the ℓ^1 operator norm of $\mathbf{A}^r\mathbf{P}$ is given by

$$\|\mathbf{A}^r\mathbf{P}\|_1 = \max_{i\in\{1,\dots,n\}} \|\mathbf{A}^r\mathbf{P}\mathbf{e}_i\|_1 = \max_{i\in\{1,\dots,n\}} \|\mathbf{A}^r\mathbf{e}_i - \mathbf{v}\|_1 \in [0, 2], \tag{6.19}$$

where $\mathbf{e}_i$ the i^{th} standard basis vector in $\mathbb{R}^n$. That is, $\|\mathbf{A}^r\mathbf{P}\|_1$ is twice the total variation distance between the distribution of the Markov chain with transition matrix $\mathbf{A}$ after r steps and its stationary distribution $\mathbf{v}$ in the worst-case initialization.

The following proposition collects various properties of the ℓ^1 contraction coefficients $\alpha_r(\mathbf{A})$ defined in Definition 6.1.1. These are known in various forms throughout the literature; e.g., see the survey [IS11] on ergodic coefficients. For completeness, we will include elementary proofs of the properties in Section 6.9.

Proposition 6.4.3 (Properties of ℓ^1 contraction coefficients)**.** *Let $\mathbf{A} \in [0, 1]^{n\times n}$ be a column-stochastic matrix with leading right eigenvector $\mathbf{v} \in [0, 1]^n$, and $\mathbf{P} = \mathbf{I} - \mathbf{v}\mathbf{1}^\mathsf{T}$. Then:*

1. (Scrambling identity). *The ℓ^1 contraction coefficients of $\mathbf{A}$ can be written:*

$$\alpha_r(\mathbf{A}) = \frac{1}{2}\max_{i,j}\|\mathbf{A}^r(\mathbf{e}_i - \mathbf{e}_j)\|_1 = 1 - \min_{i,j}\sum_{k=1}^{n}\min\{\mathbf{A}^r(k, i), \mathbf{A}^r(k, j)\}.$$

2. (Submultiplicativity). *The ℓ^1 contraction coefficients of $\mathbf{A}$ are submultiplicative:*

$$\alpha_r(\mathbf{A}) \leq \alpha_s(\mathbf{A})\alpha_{r-s}(\mathbf{A}) \quad \textit{for each } s \leq r.$$

3. (Oblique projection bounds). *The ℓ^1 contraction coefficients of $\mathbf{A}$ satisfy*

$$\frac{1}{2}\|\mathbf{A}^r\mathbf{P}\|_1 \leq \alpha_r(\mathbf{A}) \leq \|\mathbf{A}^r\mathbf{P}\|_1.$$

4. (Eigenvalue bounds). *Let* $\lambda_2(\mathbf{A})$ *be the second-largest magnitude eigenvalue of* $\mathbf{A}$. *If* $|\lambda_2(\mathbf{A})| < 1$, *then the* ℓ^1 *contraction coefficients of* $\mathbf{A}$ *satisfy*

$$|\lambda_2(\mathbf{A})|^r \leq \alpha_r(\mathbf{A}) \quad \textit{and} \quad \lim_{r\to\infty} \alpha_r(\mathbf{A})^{1/r} = |\lambda_2(\mathbf{A})|.$$

Example 6.4.4. As an illustrative example, consider a random walk on a circle with four vertices that stays put with probability 1/2 and moves clockwise with probability 1/2 as depicted in Figure 6.3, which is an irreducible and aperiodic Markov chain.

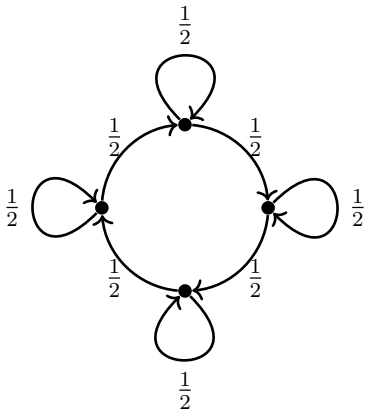


Figure 6.3: A random walk on a clockwise directed 4-cycle with self-loops.

The transition matrix $\mathbf{A}$, where $\mathbf{A}(j,i)$ contains the probability of moving from i to j, and its powers for $r \geq 1$ are given by

$$\mathbf{A}^r = \frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} + \left(\frac{1}{\sqrt{2}}\right)^{r-2} \begin{bmatrix} \cos\theta_r & -\sin\theta_r & -\cos\theta_r & \sin\theta_r \\ \sin\theta_r & \cos\theta_r & -\sin\theta_r & -\cos\theta_r \\ -\cos\theta_r & \sin\theta_r & \cos\theta_r & -\sin\theta_r \\ -\sin\theta_r & -\cos\theta_r & \sin\theta_r & \cos\theta_r \end{bmatrix},$$

where $\theta_r := r\pi/4$. Even though the matrix $\mathbf{A}$ is irreducible and aperiodic, the ℓ^1 contraction coefficients $\alpha_r(\mathbf{A})$, using Proposition 6.4.3, part 1, are given by $1, 1, \frac{1}{2}, \frac{1}{2}, \frac{1}{4}, \frac{1}{4}, \ldots$ for $r =$

$0, 1, 2, 3, 4, 5, \ldots$. In general,

$$\alpha_r(\mathbf{A}) = \begin{cases} (1/\sqrt{2})^r & \text{if } r \text{ is even,} \\ (1/\sqrt{2})^{r-1} & \text{if } r \text{ is odd.} \end{cases}$$

Furthermore, $|\lambda_2(\mathbf{A})| = 1/\sqrt{2}$. This confirms that $\alpha_r(\mathbf{A}) \geq |\lambda_2(\mathbf{A})|^r = (1/\sqrt{2})^r$ and $\lim_{r\to\infty} \alpha_r(\mathbf{A})^{1/r} = |\lambda_2(\mathbf{A})| = 1/\sqrt{2}$, which is consistent with Proposition 6.4.3, part 4.

Remark 6.4.5 (Scrambling matrices)**.** Assuming $\alpha_1(\mathbf{A}) < 1$ is much stronger than assuming the spectral gap condition $|\lambda_2(\mathbf{A})| < 1$. Indeed, from Proposition 6.4.3, part 1, the condition $\alpha_1(\mathbf{A}) < 1$ is equivalent to requiring that $\mathbf{A}$ is *"scrambling"*, i.e., every pair of columns $\mathbf{A}(:, i)$ and $\mathbf{A}(:, j)$ shares a common positive element. The proof of [LW17, Corollary 1] claims that $\alpha_1(\mathbf{A}) < 1$ whenever the column-stochastic matrix $\mathbf{A}$ is irreducible and aperiodic. However, this is apparently false.

Next, recall the definition of the mixing time $\tau_{\text{mix}}(\mathbf{A})$ from Definition 6.1.1. From (6.19), we see that equivalently,

$$\tau_{\text{mix}}(\mathbf{A}) = \arg\min_{\tau\in\mathbb{N}} \left\{ \|\mathbf{A}^\tau \mathbf{P}\|_1 \leq \frac{1}{2} \right\}. \tag{6.20}$$

Since the Markov chain corresponding to $\mathbf{A}$ converges to $\mathbf{v}$ under a spectral gap assumption, we know that the mixing time is finite: i.e., there exists an integer R such that $\|\mathbf{A}^R \mathbf{P}\|_1 \leq 1/2$. However, this does not say anything about the rate of convergence, and understanding the dependence of the mixing time on the size of the state space for different Markov chains is a fascinating and active area of research that is beyond the scope of our investigations. For a comprehensive treatise on the mixing times of Markov chains, we refer to [LPW17]. In the context of exponentially-sized state spaces with $n = 2^d$, a Markov chain that mixes in $O(\text{polylog}\, d)$ steps is commonly said to be *rapid mixing*, while one that mixes in $O(\exp(cd))$ steps is *slow mixing*.

The mixing time $\tau_{\text{mix}}(\mathbf{A})$ can be lower bounded by the reciprocal of the spectral gap of $\mathbf{A}$ ([LPW17, Theorem 12.5]):

$$\tau_{\text{mix}}(\mathbf{A}) \geq \left(\frac{1}{1-\lambda_2(\mathbf{A})} - 1\right)\log(2), \tag{6.21}$$

recalling that $\lambda_2(\mathbf{A})$ denotes the second largest-eigenvalue of $\mathbf{A}$. Thus, while the mixing time is a natural quantity for stochastic matrices, Theorem 6.1.5 suggests that a sparsification parameter that is at least of order $m \geq O\big(1/(1-\lambda_2(\mathbf{A}))\big)$ is necessary for the effectiveness of the randomly sparsified power method for more general (e.g., Hermitian) matrices.

6.5 Proofs for deterministically sparsified power method

In this section, we describe the proofs of our main results (Theorems 6.1.3 and 6.1.4) for the deterministically sparsified power method (6.5).

6.5.1 Properties of deterministic sparsification

We begin by establishing some properties of the deterministic sparsification operator $\boldsymbol{\psi}$. First, we prove that the deterministic sparsification operator $\boldsymbol{\psi}$ is optimal for stochastic vectors in an ℓ^p sense for any $p \in [1,\infty]$.

Lemma 6.5.1 (Optimality of deterministic sparsification)**.** *Let $\mathbf{y} \in [0,1]^n$ be any stochastic vector and $\boldsymbol{\psi}(\mathbf{y})$ be the output of deterministic sparsification with sparsification parameter m. Then, for any m-sparse stochastic vector $\mathbf{z} \in [0,1]^n$ and $p \in [1,\infty]$,*

$$\|\mathbf{y} - \boldsymbol{\psi}(\mathbf{y})\|_p \leq \|\mathbf{y} - \mathbf{z}\|_p.$$

Proof. Suppose that we are given any m-sparse stochastic vector $\mathbf{z}$ supported on $\mathsf{S} \subseteq \{1, 2, \ldots, n\}$. First, consider the ℓ^p error with $1 \leq p < \infty$,

$$\|\mathbf{y} - \mathbf{z}\|_p^p = \sum_{i \in \mathsf{S}} |\mathbf{y}(i) - \mathbf{z}(i)|^p + \sum_{i \notin \mathsf{S}} \mathbf{y}(i)^p.$$

Note that by using $|\mathsf{S}| = m$ and Hölder's inequality, we have

$$\left(\sum_{i \in \mathsf{S}} |\mathbf{z}(i) - \mathbf{y}(i)| \right)^p \leq m^{p-1} \sum_{i \in \mathsf{S}} |\mathbf{z}(i) - \mathbf{y}(i)|^p,$$

with equality if and only if $\mathbf{z}(i) - \mathbf{y}(i)$ is constant for $i \in \mathsf{S}$. Furthermore, by applying the triangle inequality, using the fact that $\sum_{i \in \mathsf{S}} \mathbf{z}(i) = 1$ and $\mathbf{y}$ is a stochastic vector, we have

$$\sum_{i \notin \mathsf{S}} \mathbf{y}(i) = 1 - \sum_{i \in \mathsf{S}} \mathbf{y}(i) = \sum_{i \in \mathsf{S}} \mathbf{z}(i) - \sum_{i \in \mathsf{S}} \mathbf{y}(i) \leq \sum_{i \in \mathsf{S}} |\mathbf{z}(i) - \mathbf{y}(i)|.$$

Hence, we deduce that

$$\begin{aligned} \|\mathbf{y} - \mathbf{z}\|_p^p &\geq m^{1-p} \left(\sum_{i \notin \mathsf{S}} \mathbf{y}(i) \right)^p + \sum_{i \notin \mathsf{S}} \mathbf{y}(i)^p \\ &\geq m^{1-p} \left(\sum_{i=m+1}^{n} \mathbf{y}^{\downarrow}(i) \right)^p + \sum_{i=m+1}^{n} \mathbf{y}^{\downarrow}(i)^p = \|\mathbf{y} - \boldsymbol{\psi}(\mathbf{y})\|_p^p. \end{aligned}$$

For the ℓ^∞ error, we can apply the following bespoke argument that uses the same ideas. Note that

$$\|\mathbf{y} - \mathbf{z}\|_\infty = \max \left\{ \max_{i \in \mathsf{S}} |\mathbf{y}(i) - \mathbf{z}(i)|, \max_{i \notin \mathsf{S}} \mathbf{y}(i) \right\}.$$

Observe that because $|\mathsf{S}| = m$, $\max_{i \notin \mathsf{S}} \mathbf{y}(i) \geq \mathbf{y}^{\downarrow}(m+1)$. Next, notice that the maximum over the set S is at least as large as the average over S. Therefore, using the triangle inequality

and the fact that $\sum_{i \in \mathsf{S}} \mathbf{z}(i) = 1$ and $\mathbf{y}$ is a stochastic vector again, we deduce that

$$\begin{aligned}\max_{i \in \mathsf{S}} |\mathbf{y}(i) - \mathbf{z}(i)| &\geq \frac{1}{m} \sum_{i \in \mathsf{S}} |\mathbf{y}(i) - \mathbf{z}(i)| \\ &\geq \sum_{i \in \mathsf{S}} \mathbf{z}(i) - \sum_{i \in \mathsf{S}} \mathbf{y}(i) = 1 - \sum_{i \in \mathsf{S}} \mathbf{y}(i) = \sum_{i \notin \mathsf{S}} \mathbf{y}(i) \geq \sum_{i=m+1}^{n} \mathbf{y}^{\downarrow}(i).\end{aligned}$$

Hence,

$$\|\mathbf{y} - \mathbf{z}\|_\infty \geq \max\left\{ \frac{1}{m} \sum_{i=m+1}^{n} \mathbf{y}^{\downarrow}(i), \mathbf{y}^{\downarrow}(m+1) \right\} = \mathbf{y}^{\downarrow}(m+1) = \|\mathbf{y} - \boldsymbol{\psi}(\mathbf{y})\|_\infty.$$

We conclude that for $p \in [1, \infty]$, the ℓ^p error of any m-sparse approximation $\mathbf{z}$ of $\mathbf{y}$ is minimized by choosing the support S to be the m largest entries of $\mathbf{y}$ and evenly redistributing the mass of $\mathbf{y}$ outside the support, i.e., $\mathbf{z}(i) = \mathbf{y}(i) + m^{-1} \sum_{i \notin \mathsf{S}} \mathbf{y}(i)$ for $i \in \mathsf{S}$ and $\mathbf{z}(i) = 0$ otherwise. □

Next, we prove the following bound on the error from applying the deterministic sparsification operator.

Proposition 6.5.2 (Bounded truncation error)**.** *Fix two stochastic vectors $\mathbf{y}, \mathbf{v} \in [0,1]^n$, and let $\boldsymbol{\psi}(\mathbf{y})$ denote the output of the deterministic sparsification operator applied to $\mathbf{y}$ with sparsification parameter m. Then, for any $s \leq m$, we have*

$$\|\boldsymbol{\psi}(\mathbf{y}) - \mathbf{v}\|_1 \leq \left[1 + \frac{2s}{m}\right] \|\mathbf{y} - \mathbf{v}\|_1 + 2\left(\sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i) \right). \tag{6.22}$$

Furthermore,

$$\|\boldsymbol{\psi}(\mathbf{y}) - \mathbf{y}\|_1 \leq \|\mathbf{y} - \mathbf{v}\|_1 + \left(\sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i) \right). \tag{6.23}$$

Remark 6.5.3 (Optimality of the bound)**.** For fixed integers $s \leq m$, consider the following example of a bad vector in $\mathbb{R}^{m+s}$ for deterministic sparsification:

$$\mathbf{v} = \big(\underbrace{1/s, \ldots, 1/s}_{s \text{ times}}, 0, \ldots, 0\big), \qquad \mathbf{y} = \big(\underbrace{1/(s+m), \ldots, 1/(s+m)}_{s+m \text{ times}}\big),$$
$$\boldsymbol{\psi}(\mathbf{y}) = \big(0, \ldots, 0, \underbrace{1/m, \ldots, 1/m}_{m \text{ times}}\big).$$

Essentially, the method is picking all the wrong entries to truncate. We can calculate

$$\|\boldsymbol{\psi}(\mathbf{y}) - \mathbf{v}\|_1 = 2, \qquad \sum_{i=s+1}^{m} \mathbf{v}^{\downarrow}(i) = 0,$$
$$\|\mathbf{y} - \mathbf{v}\|_1 = s\left(\frac{1}{s} - \frac{1}{s+m}\right) + m\left(\frac{1}{s+m}\right) = \frac{sm + sm}{s(s+m)} = \frac{2m}{s+m} = \frac{2}{(s/m)+1},$$

Therefore, in this case, the deterministic sparsification error is given by

$$\|\boldsymbol{\psi}(\mathbf{y}) - \mathbf{v}\|_1 = \left[1 + \frac{s}{m}\right] \|\mathbf{y} - \mathbf{v}\|_1.$$

This shows that the first term of the bound (6.22) in Proposition 6.5.2 is near-optimal, up to a constant factor of two. Furthermore, note that for general $\mathbf{y}$, $\|\mathbf{y} - \boldsymbol{\psi}(\mathbf{y})\|_1 = 2\sum_{i=m+1}^{n} \mathbf{y}^{\downarrow}(i)$, which shows that the second term in Proposition 6.5.2 is also optimal.

Proof of Proposition 6.5.2. Let the s largest entries in $\mathbf{v}$ be denoted T for "target", and the m largest entries in $\mathbf{y}$ be denoted K for "kept". By definition of the deterministic sparsification operator $\boldsymbol{\psi}$,

$$\begin{aligned}
\|\boldsymbol{\psi}(\mathbf{y}) - \mathbf{v}\|_1 &= \sum_{i \in \mathsf{K}} |\boldsymbol{\psi}(\mathbf{y})(i) - \mathbf{v}(i)| + \sum_{i \notin \mathsf{K}} |\boldsymbol{\psi}(\mathbf{y})(i) - \mathbf{v}(i)| \\
&\leq \sum_{i \in \mathsf{K}} |\boldsymbol{\psi}(\mathbf{y})(i) - \mathbf{y}(i)| + \sum_{i \in \mathsf{K}} |\mathbf{y}(i) - \mathbf{v}(i)| + \sum_{i \notin \mathsf{K}} |\boldsymbol{\psi}(\mathbf{y})(i) - \mathbf{v}(i)| \\
&= \sum_{i \notin \mathsf{K}} \mathbf{y}(i) + \sum_{i \in \mathsf{K}} |\mathbf{y}(i) - \mathbf{v}(i)| + \sum_{i \notin \mathsf{K}} \mathbf{v}(i).
\end{aligned}$$

Next, observe that we can bound

$$\begin{aligned}\sum_{i\notin \mathsf{K}} \mathbf{v}(i) &= \sum_{i\in \mathsf{K}^c\cap \mathsf{T}^c} \mathbf{v}(i) + \sum_{i\in \mathsf{T}\setminus \mathsf{K}} \mathbf{v}(i) \\ &\leq \sum_{i\in \mathsf{K}^c\cap \mathsf{T}^c} \mathbf{v}(i) + \sum_{i\in \mathsf{T}\setminus \mathsf{K}} |\mathbf{v}(i)-\mathbf{y}(i)| + \sum_{i\in \mathsf{T}\setminus \mathsf{K}} \mathbf{y}(i).\end{aligned}$$

Similarly,

$$\begin{aligned}\sum_{i\notin \mathsf{K}} \mathbf{y}(i) &= \sum_{i\in \mathsf{T}\setminus \mathsf{K}} \mathbf{y}(i) + \sum_{i\in \mathsf{K}^c\cap \mathsf{T}^c} \mathbf{y}(i) \\ &\leq \sum_{i\in \mathsf{T}\setminus \mathsf{K}} \mathbf{y}(i) + \sum_{i\in \mathsf{K}^c\cap \mathsf{T}^c} |\mathbf{y}(i)-\mathbf{v}(i)| + \sum_{i\in \mathsf{K}^c\cap \mathsf{T}^c} \mathbf{v}(i).\end{aligned}$$

Combining the preceding displayed bounds yields

$$\|\boldsymbol{\psi}(\mathbf{y})-\mathbf{v}\|_1 \leq 2\sum_{i\in \mathsf{T}\setminus \mathsf{K}} \mathbf{y}(i) + \|\mathbf{y}-\mathbf{v}\|_1 + 2\sum_{i\in \mathsf{K}^c\cap \mathsf{T}^c} \mathbf{v}(i).$$

Now, since K consists of the m largest entries of $\mathbf{y}$, we have the following inequality between the average entry of $\mathbf{y}$ in $\mathsf{T}\setminus\mathsf{K}$ and $\mathsf{K}\setminus\mathsf{T}$, assuming that $|\mathsf{T}\setminus\mathsf{K}|>0$ and $|\mathsf{K}\setminus\mathsf{T}|>0$:

$$\frac{1}{|\mathsf{T}\setminus\mathsf{K}|}\sum_{i\in \mathsf{T}\setminus \mathsf{K}} \mathbf{y}(i) \leq \frac{1}{|\mathsf{K}\setminus\mathsf{T}|}\sum_{i\in \mathsf{K}\setminus \mathsf{T}} \mathbf{y}(i).$$

Note that

$$\frac{|\mathsf{T}\setminus\mathsf{K}|}{|\mathsf{K}\setminus\mathsf{T}|} = \frac{s-|\mathsf{T}\cap\mathsf{K}|}{m-|\mathsf{T}\cap\mathsf{K}|} \leq \frac{s}{m}.$$

Therefore,

$$\begin{aligned}\|\boldsymbol{\psi}(\mathbf{y})-\mathbf{v}\|_1 &\leq 2\frac{|\mathsf{T}\setminus\mathsf{K}|}{|\mathsf{K}\setminus\mathsf{T}|}\sum_{i\in \mathsf{K}\setminus \mathsf{T}} \mathbf{y}(i) + \|\mathbf{y}-\mathbf{v}\|_1 + 2\sum_{i\in \mathsf{K}^c\cap \mathsf{T}^c} \mathbf{v}(i) \\ &\leq 2\frac{s}{m}\sum_{i\in \mathsf{K}\setminus \mathsf{T}} \mathbf{y}(i) + \|\mathbf{y}-\mathbf{v}\|_1 + 2\sum_{i\in \mathsf{K}^c\cap \mathsf{T}^c} \mathbf{v}(i).\end{aligned}$$

If $\mathsf{T} \subseteq \mathsf{K}$ and hence $|\mathsf{T} \setminus \mathsf{K}| = 0$, then this bound also holds trivially. By further routine application of the triangle inequality, we deduce that

$$\begin{aligned}\|\boldsymbol{\psi}(\mathbf{y}) - \mathbf{v}\|_1 &\leq 2\frac{s}{m}\sum_{i\in\mathsf{K}\setminus\mathsf{T}} \mathbf{v}(i) + 2\frac{s}{m}\sum_{i\in\mathsf{K}\setminus\mathsf{T}} |\mathbf{y}(i) - \mathbf{v}(i)| + \|\mathbf{y} - \mathbf{v}\|_1 + 2\sum_{i\in\mathsf{K}^c\cap\mathsf{T}^c} \mathbf{v}(i) \\ &\leq 2\max\left\{\frac{s}{m}, 1\right\}\sum_{i\notin\mathsf{T}} \mathbf{v}(i) + \left[1 + \frac{2s}{m}\right]\|\mathbf{y} - \mathbf{v}\|_1.\end{aligned}$$

After inserting the definition of T and using $s \leq m$, we obtain the first inequality.

For the second inequality, by a similar argument, we know that

$$\begin{aligned}\|\boldsymbol{\psi}(\mathbf{y}) - \mathbf{y}\|_1 &\leq \sum_{i\in\mathsf{T}\setminus\mathsf{K}} \mathbf{y}(i) + \sum_{i\in\mathsf{K}^c\cap\mathsf{T}^c} |\mathbf{y}(i) - \mathbf{v}(i)| + \sum_{i\in\mathsf{K}^c\cap\mathsf{T}^c} \mathbf{v}(i) \\ &\leq \frac{s}{m}\sum_{i\in\mathsf{K}\setminus\mathsf{T}} \mathbf{y}(i) + \sum_{i\in\mathsf{K}^c\cap\mathsf{T}^c} |\mathbf{y}(i) - \mathbf{v}(i)| + \sum_{i\in\mathsf{K}^c\cap\mathsf{T}^c} \mathbf{v}(i).\end{aligned}$$

By using $s/m \leq 1$ and combining terms, we conclude that

$$\begin{aligned}\|\boldsymbol{\psi}(\mathbf{y}) - \mathbf{y}\|_1 &\leq \sum_{i\in\mathsf{T}^c} |\mathbf{y}(i) - \mathbf{v}(i)| + \sum_{i\in\mathsf{T}^c} \mathbf{v}(i) \\ &\leq \|\mathbf{y} - \mathbf{v}\|_1 + \sum_{i\in\mathsf{T}^c} \mathbf{v}(i).\end{aligned}$$

□

6.5.2 Error bound with strict contractivity

We will now prove Theorem 6.1.3, which establishes a non-asymptotic error bound for the deterministically sparsified power method, assuming that $\mathbf{A}$ is strictly contractive with $\alpha_1(\mathbf{A}) < 1$.

Proof of Theorem 6.1.3. By the triangle inequality, we have

$$\begin{aligned}\|\mathbf{x}_t - \mathbf{v}\|_1 &\leq \|\mathbf{A}^t\mathbf{x}_0 - \mathbf{v}\|_1 + \|\mathbf{x}_t - \mathbf{A}^t\mathbf{x}_0\|_1 \\ &\leq \alpha_1(\mathbf{A})^t\|\mathbf{x}_0 - \mathbf{v}\|_1 + \|\mathbf{x}_t - \mathbf{A}^t\mathbf{x}_0\|_1.\end{aligned} \tag{6.24}$$

The first term corresponds to the error from the full power iterations without any sparsification (6.4), where we have used the fact that $\mathbf{A}^t\mathbf{x}_0 - \mathbf{v} = \mathbf{A}^t(\mathbf{x}_0 - \mathbf{v})$ and $\mathbf{1}^\mathsf{T}(\mathbf{x}_0 - \mathbf{v}) = 0$. The second term corresponds to the deviation of the deterministically sparsified power method from this trajectory, and our goal is to show that this can be controlled with a sufficiently large sparsification parameter m.

To do so, we will first write the deviation as a telescoping sum:

$$\begin{aligned}\|\mathbf{x}_t - \mathbf{A}^t\mathbf{x}_0\|_1 &= \left\|\sum_{r=0}^{t-1} \mathbf{A}^{t-r-1}\big[\mathbf{A}\mathbf{x}_r - \boldsymbol{\psi}(\mathbf{A}\mathbf{x}_r)\big]\right\|_1 \\ &\le \sum_{r=0}^{t-1} \big\|\mathbf{A}^{t-r-1}\big[\mathbf{A}\mathbf{x}_r - \boldsymbol{\psi}(\mathbf{A}\mathbf{x}_r)\big]\big\|_1 \\ &\le \sum_{r=0}^{t-1} \alpha_1(\mathbf{A})^{t-r-1}\big\|\mathbf{A}\mathbf{x}_r - \boldsymbol{\psi}(\mathbf{A}\mathbf{x}_r)\big\|_1.\end{aligned}$$

Fix integers $s \le m$. By using Proposition 6.5.2 to bound the one-step sparsification error, we obtain

$$\begin{aligned}\|\mathbf{x}_t - \mathbf{A}^t\mathbf{x}_0\|_1 &\le \sum_{r=0}^{t-1} \alpha_1(\mathbf{A})^{t-r-1}\left(\|\mathbf{A}\mathbf{x}_r - \mathbf{v}\|_1 + \sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i)\right) \\ &\le \sum_{r=0}^{t-1} \alpha_1(\mathbf{A})^{t-r}\|\mathbf{x}_r - \mathbf{v}\|_1 + \frac{1-\alpha_1(\mathbf{A})^t}{1-\alpha_1(\mathbf{A})}\sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i).\end{aligned} \tag{6.25}$$

Now, for $r \le t-1$, by using Proposition 6.5.2 again, we have

$$\begin{aligned}\|\mathbf{x}_r - \mathbf{v}\|_1 &= \|\boldsymbol{\psi}(\mathbf{A}\mathbf{x}_{r-1}) - \mathbf{v}\|_1 \\ &\le \left(1 + \frac{2s}{m}\right)\|\mathbf{A}\mathbf{x}_{r-1} - \mathbf{v}\|_1 + 2\sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i) \\ &\le \alpha_1(\mathbf{A})\left(1 + \frac{2s}{m}\right)\|\mathbf{x}_{r-1} - \mathbf{v}\|_1 + 2\sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i).\end{aligned}$$

Let $\beta := \alpha_1(\mathbf{A})(1 + 2s/m)$. Clearly $\alpha_1(\mathbf{A}) < \beta$. By iterating the previous inequality, we deduce that

$$\begin{aligned}\|\mathbf{x}_r - \mathbf{v}\|_1 &\leq \beta^r \|\mathbf{x}_0 - \mathbf{v}\|_1 + 2\Big(1 + \beta + \ldots + \beta^{r-1}\Big) \sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i) \\ &\leq \beta^r \|\mathbf{x}_0 - \mathbf{v}\|_1 + \frac{2(1-\beta^r)}{1-\beta} \sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i).\end{aligned} \tag{6.26}$$

For the contribution from the initial error to decay, this suggests that we require a large enough sparsification parameter m such that

$$\beta = \alpha_1(\mathbf{A})\Big(1 + \frac{2s}{m}\Big) < 1 \iff \frac{2s}{m} < \frac{1}{\alpha_1(\mathbf{A})} - 1. \tag{6.27}$$

Under this assumption, inserting (6.26) back into (6.25) yields

$$\begin{aligned}\|\mathbf{x}_t - \mathbf{A}^t \mathbf{x}_0\|_1 &\leq \sum_{r=0}^{t-1} \alpha_1(\mathbf{A})^{t-r} \Big(\beta^r \|\mathbf{x}_0 - \mathbf{v}\|_1 + \frac{2(1-\beta^r)}{1-\beta} \sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i)\Big) \\ &\quad + \frac{1 - \alpha_1(\mathbf{A})^t}{1 - \alpha_1(\mathbf{A})} \sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i) \\ &= \alpha_1(\mathbf{A})^t \sum_{r=0}^{t-1} \Big(\frac{\beta}{\alpha_1(\mathbf{A})}\Big)^r \|\mathbf{x}_0 - \mathbf{v}\|_1 \\ &\quad + \Bigg[\sum_{r=0}^{t-1} \frac{2\alpha_1(\mathbf{A})^{t-r}(1-\beta^r)}{1-\beta} + \frac{1 - \alpha_1(\mathbf{A})^t}{1 - \alpha_1(\mathbf{A})}\Bigg] \sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i).\end{aligned}$$

Note that

$$\alpha_1(\mathbf{A})^t \sum_{r=0}^{t-1} \Big(\frac{\beta}{\alpha_1(\mathbf{A})}\Big)^r = \alpha_1(\mathbf{A})^t \cdot \frac{(1 + 2s/m)^t - 1}{2s/m} \leq \frac{m}{2s} \beta^t.$$

Furthermore,

$$\begin{aligned}\sum_{r=0}^{t-1} \alpha_1(\mathbf{A})^{t-r}(1-\beta^r) &= \alpha_1(\mathbf{A})^t \cdot \frac{\alpha_1(\mathbf{A})^{-t} - 1}{\alpha_1(\mathbf{A})^{-1} - 1} - \alpha_1(\mathbf{A})^t \cdot \frac{(1 + 2s/m)^t - 1}{2s/m} \\ &\leq \alpha_1(\mathbf{A}) \cdot \frac{1 - \alpha_1(\mathbf{A})^t}{1 - \alpha_1(\mathbf{A})}.\end{aligned}$$

Hence, we deduce that

$$\begin{aligned}\|\mathbf{x}_t - \mathbf{A}^t\mathbf{x}_0\|_1 &\leq \frac{m}{2s}\beta^t\|\mathbf{x}_0 - \mathbf{v}\|_1 + \left[\frac{2\alpha_1(\mathbf{A})}{1-\beta} + 1\right]\frac{1-\alpha_1(\mathbf{A})^t}{1-\alpha_1(\mathbf{A})}\sum_{i=s+1}^{n}\mathbf{v}^{\downarrow}(i) \\ &\leq \frac{m}{2s}\beta^t\|\mathbf{x}_0 - \mathbf{v}\|_1 + \frac{1}{1-\beta}\frac{1+\alpha_1(\mathbf{A})}{1-\alpha_1(\mathbf{A})}\sum_{i=s+1}^{n}\mathbf{v}^{\downarrow}(i).\end{aligned}$$

Finally, inserting this back into (6.24) yields

$$\|\mathbf{x}_t - \mathbf{v}\|_1 \leq \left(\alpha_1(\mathbf{A})^t + \frac{m}{2s}\beta^t\right)\|\mathbf{x}_0 - \mathbf{v}\|_1 + \frac{1}{1-\beta}\frac{1+\alpha_1(\mathbf{A})}{1-\alpha_1(\mathbf{A})}\sum_{i=s+1}^{n}\mathbf{v}^{\downarrow}(i). \tag{6.28}$$

To conclude, we will provide a choice of s for any given m that satisfies (6.27) to derive an explicit form for β. Suppose that we choose

$$s = \left\lfloor \frac{1-\alpha_1(\mathbf{A})}{4\alpha_1(\mathbf{A})}m \right\rfloor = \left\lfloor \frac{m}{m_\star(\mathbf{A})} \right\rfloor. \tag{6.29}$$

Note that the assumption $m \geq 2m_*(\mathbf{A})$ implies s is non-trivial (i.e., $s \geq 1$). Then,

$$\frac{2s}{m} \leq \frac{1}{2}\left(\frac{1}{\alpha_1(\mathbf{A})} - 1\right) \quad \text{and} \quad \beta = \alpha_1(\mathbf{A})\left(1 + \frac{2s}{m}\right) \leq \frac{1+\alpha_1(\mathbf{A})}{2}.$$

Furthermore, using the assumption on m,

$$\frac{m}{2s} \leq \frac{m}{2\left(\frac{1-\alpha_1(\mathbf{A})}{4\alpha_1(\mathbf{A})}m - 1\right)} = \frac{2\alpha_1(\mathbf{A})}{1-\alpha_1(\mathbf{A})} \cdot \frac{1}{1 - \frac{4\alpha_1(\mathbf{A})}{1-\alpha_1(\mathbf{A})}\frac{1}{m}} \leq \frac{4\alpha_1(\mathbf{A})}{1-\alpha_1(\mathbf{A})}.$$

Inserting the choice (6.29) of s for the depth of the tail into (6.28) and applying the simplifications above completes the proof. □

6.5.3 Failure mode of deterministic sparsification

Finally, we will prove Theorem 6.1.4, which constructs a counterexample showing how the deterministically sparsified power method can fail.

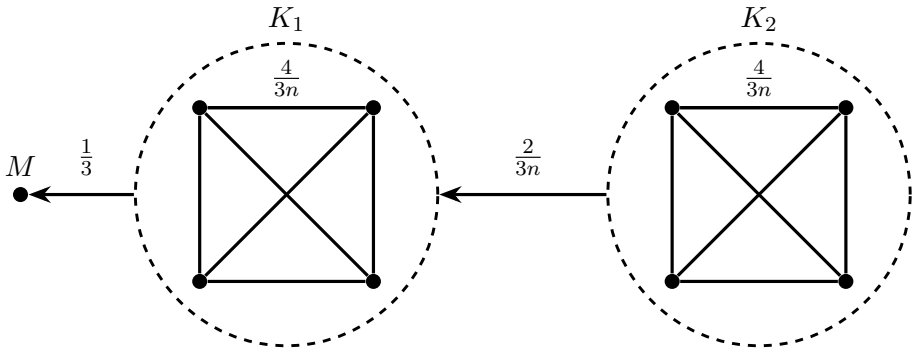


Figure 6.4: Markov chain on $n+1$ vertices with n even: K_1 and K_2 are two complete graphs on $n/2$ vertices with self-loops, directed edges in both directions, and edge weights $4/(3n)$. Each vertex in K_2 has a *directed* edge with weight $2/(3n)$ to each of the vertices in K_1, and each vertex in K_1 has a directed edge to a special vertex M with weight $1/3$. M has a self-loop with weight one.

Proof of Theorem 6.1.4. For simplicity, we will describe the construction for when n for when n is even. Consider the following column-stochastic matrix $\mathbf{A} \in [0,1]^{(n+1)\times(n+1)}$, written in block form:

$$\mathbf{A} = \begin{bmatrix} 1 & \frac{1}{3}\mathbf{1}_{n/2}^\mathsf{T} & \mathbf{0}^\mathsf{T} \\ \mathbf{0} & \frac{4}{3n}\mathbf{1}_{n/2\times n/2} & \frac{2}{3n}\mathbf{1}_{n/2\times n/2} \\ \mathbf{0} & \mathbf{0} & \frac{4}{3n}\mathbf{1}_{n/2\times n/2} \end{bmatrix}.$$

Here, $\mathbf{1}_{n/2} \in \mathbb{R}^{n/2}$ and $\mathbf{1}_{n/2\times n/2} \in \mathbb{R}^{n/2\times n/2}$ denotes the all-ones vector and all-ones matrix, respectively. That is, $\mathbf{A}$ is the probability transition matrix corresponding to the random walk on the graph in Figure 6.4, where $\mathbf{A}(j,i)$ gives the probability that a random walker at state i moves to state j in the next step. The first index corresponds to the special vertex M, the next $n/2$ indices correspond to the vertices in the clique K_1, and the last $n/2$ indices corresponding to the vertices in the clique K_2. (If n is odd, we can simply modify the construction by adding an extra vertex to the special cluster M, and the argument essentially remains the same.)

Note that the states in K_1 and K_2 are inessential, and therefore the stationary distribution of the Markov chain is supported on the special vertex M, which is an absorbing state. That

is, the leading eigenvector solving $\mathbf{A}\mathbf{v} = \mathbf{v}$ is given by

$$\mathbf{v} = (1, 0, \ldots, 0).$$

Next, note that from Proposition 6.4.3, part 1, we see that $\alpha_1(\mathbf{A}) = 1$; i.e., $\mathbf{A}$ is not scrambling. Using the block representation of $\mathbf{A}$, it can be verified that $\lambda_2(\mathbf{A}) = 2/3$ is an eigenvalue with corresponding eigenvector

$$\mathbf{v}_2 = (-n/2, \underbrace{1, \ldots, 1}_{n/2 \text{ times}}, \underbrace{0, \ldots, 0}_{n/2 \text{ times}}),$$

The remaining eigenvalues are equal to 0, with $n/2 - 1$ eigenvectors coming from placing a permutation of $(1, -1, 0, \ldots, 0) \in \mathbb{R}^{n/2}$ in the $n/2$ indices corresponding to K_1, and similarly $n/2 - 1$ eigenvectors coming from placing a permutation of the same vector in the last $n/2$ indices corresponding to K_2. Since the trace of $\mathbf{A}$ is equal to $1 + 2(2/3)$, it follows that the second largest-magnitude eigenvalue is indeed $\lambda_2(\mathbf{A}) = 2/3$, which has algebraic multiplicity two. Thus, the spectral gap of $\mathbf{A}$ is $1 - \lambda_2(\mathbf{A}) = 1/3$.

Now, we will show that with the initialization

$$\mathbf{x}_0 = \mathbf{e}_{n+1} = (0, \ldots, 0, 1)$$

in K_2, the deterministically sparsified power method with any sparsification parameter $m \leq n/2$ remains stuck in K_2. After the first iteration, we have

$$\mathbf{y}_1 = \mathbf{A}\mathbf{x}_0 = \frac{2}{3n}(0, \underbrace{1, \ldots 1}_{n/2 \text{ times}}, \underbrace{2, \ldots, 2}_{n/2 \text{ times}}).$$

After deterministic sparsification, the resulting vector consists of the unit mass equally distributed among m entries over the last $n/2$ indices in K_2. Without loss of generality, we

may assume that the indices $\{n/2+1+1, \ldots, n/2+1+m\}$ are chosen. Thus,

$$\mathbf{x}_1 = \boldsymbol{\psi}(\mathbf{y}_1) = \frac{1}{m}(0, \underbrace{0, \ldots 0}_{n/2 \text{ times}}, \underbrace{1, \ldots, 1}_{m \text{ times}}, 0, \ldots, 0).$$

By computing another deterministically sparsified power iteration, we have $\mathbf{y}_2 = \mathbf{y}_1$ and $\mathbf{x}_2 = \mathbf{x}_1$. Hence, by induction, we conclude that $\|\mathbf{x}_t - \mathbf{v}\|_1 = 2$ for all times $t = 0, 1, 2, \ldots$, which is the maximal ℓ^1 distance between two stochastic vectors. □

Remark 6.5.4 (Better initializations)**.** Note that the counterexample described in Theorem 6.1.4 can be "fixed" by a better initialization. For example, if $\mathbf{x}_0$ has enough mass in K_1 instead of K_2, then the deterministically sparsified power method will indeed converge geometrically to the leading eigenvector. Moreover, the matrix $\mathbf{A}$ in the counterexample can also easily be modified into an irreducible matrix by adding a single directed edge with tiny edge weight from the vertex M to any vertex in K_2, and the conclusions will not materially change.

Remark 6.5.5 (Convergence with random sparsification)**.** We can show that the randomly sparsified power method effectively converges for the counterexample described in Theorem 6.1.4. Using the same matrix $\mathbf{A}$ and notation from the proof (see Figure 6.4), we can compute the mixing time $\tau_{\text{mix}}(\mathbf{A})$ by calculating the maximal distance to stationarity: $\|\mathbf{A}^\tau \mathbf{P}\|_1 = \max_{i \in \{1,2,\ldots,n+1\}} \|\mathbf{A}^\tau \mathbf{e}_i - \mathbf{v}\|_1$. Clearly, the worst case corresponds to an initialization in K_2, say $\mathbf{e}_{n+1} = (0, \ldots, 0, 1)$. Similar to the calculations in the proof of Theorem 6.1.4, we can compute

$$\mathbf{A}\mathbf{e}_{n+1} = \Big(0, \frac{2}{3n}\mathbf{1}_{n/2}, \frac{4}{3n}\mathbf{1}_{n/2}\Big), \quad \mathbf{A}^2\mathbf{e}_{n+1} = \Big(\frac{1}{9}, \frac{8}{9n}\mathbf{1}_{n/2}, \frac{8}{9n}\mathbf{1}_{n/2}\Big),$$

and so forth, with corresponding ℓ^1 errors

$$\|\mathbf{A}\mathbf{e}_{n+1} - \mathbf{v}\|_1 = 2, \quad \|\mathbf{A}^2\mathbf{e}_{n+1} - \mathbf{v}\|_1 = \frac{16}{9}, \quad \ldots.$$

Observe that $\|\mathbf{A}^t\mathbf{x}_0 - \mathbf{v}\|_1$ is independent of n for all $5 \geq 0$. By continuing the elementary computations, we find that $\tau_{\text{mix}}(\mathbf{A}) = 8$; i.e., $\|\mathbf{A}^8\mathbf{e}_{n+1} - \mathbf{v}\|_1 \leq 1/2$. Hence, because $\mathbf{v}$ is a truly sparse vector, Theorem 6.1.5 implies that there exists an absolute constant C (independent of n) such that the randomly sparsified power method with a sparsification parameter $m \geq C$ converges geometrically to $\mathbf{v}$ in the triple norm.

6.6 Proofs for randomly sparsified power method

In this section, we will prove the full version of our main result (Theorem 6.1.5) for the randomly sparsified power method (6.6). Before stating the full result, we will define a stronger error metric than the L^2 error that was introduced by [LW17].

Definition 6.6.1 (Triple norm)**.** For any random vector $\mathbf{Z} \in \mathbb{C}^n$, define

$$|||\mathbf{Z}||| := \left(\max_{\mathbf{u} \in \mathbb{C}^n, \, \|\mathbf{u}\|_\infty \leq 1} \mathbb{E}\left|\mathbf{u}^* \mathbf{Z}\right|^2 \right)^{1/2}. \tag{6.30}$$

From [LW17, Eq. (27)], we have the relationship

$$\mathbb{E}\|\mathbf{Z}\|_2^2 \leq |||\mathbf{Z}|||^2 \leq \mathbb{E}\|\mathbf{Z}\|_1^2. \tag{6.31}$$

Both inequalities can be saturated, since $|||\mathbf{Z}||| = \|\mathbf{Z}\|_1$ if $\mathbf{Z}$ is deterministic, and $|||\mathbf{Z}|||^2 = \mathbb{E}\|\mathbf{Z}\|_2^2$ if $\mathbf{Z}$ has independent, mean-zero components.

The triple norm $|||\overline{\mathbf{x}}_t - \mathbf{v}|||$ bounds the error of any low-dimensional projection of the tail-averaged iterate $\overline{\mathbf{x}}_t$ from the corresponding projection of the leading eigenvector $\mathbf{v}$, and is stronger than the L^2 error by (6.31).

6.6.1 Full version of the main result

We will prove the following fine-grained bound on the triple norm error of the randomly sparsified power method that depends on the ℓ^1 tail decay of the leading eigenvector $\mathbf{v}$ as long as the sparsification parameter m is large enough:

Theorem 6.6.2 (Fine-grained error bound)**.** *Let $R \geq 1$ be an integer such that $\|\mathbf{A}^R\mathbf{P}\|_1 < 1$. Suppose that the sparsification parameter $m \in \mathbb{N}$ satisfies*

$$m \geq \frac{12\|\mathbf{AP}\|_1^2 R}{(1-\delta)\big(1 - \|\mathbf{A}^R\mathbf{P}\|_1^2\big)} + s \quad \text{for some } \delta \in [1/2, 1) \text{ and integer } s \in \mathbb{N}.$$

Then, for any time t after a burn-in time of t_{b}, the randomly sparsified power method (6.6) produces a tail-averaged iterate $\overline{\mathbf{x}}_t = (t - t_{\mathrm{b}})^{-1} \sum_{r=t_{\mathrm{b}}+1}^{t} \mathbf{x}_r$ that satisfies

$$\begin{aligned}\big|\big|\big|\overline{\mathbf{x}}_t - \mathbf{v}\big|\big|\big|^2 \leq \frac{\Big(\sum_{\ell=0}^{R-1} \|\mathbf{A}^\ell\mathbf{P}\|_1\Big)^2 \Big(\sum_{u=0}^{\lceil (t-t_{\mathrm{b}})/R \rceil} \|\mathbf{A}^R\mathbf{P}\|_1^u\Big)^2}{(t-t_{\mathrm{b}})^2} \\ \times \left\{ \left(1 + \frac{6t}{m-s}\right) \big(\delta \|\mathbf{A}^R\mathbf{P}\|_1^2 + (1-\delta)\big)^{\lfloor t_{\mathrm{b}}/R \rfloor} \|\mathbf{x}_0 - \mathbf{v}\|_1^2 + \frac{8t}{m-s} \left(\sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i) \right)^2 \right\}.\end{aligned} \tag{6.32}$$

Assuming that Theorem 6.6.2 holds for now, we will show how it implies Theorem 6.1.5 after some simplifications.

Proof of Theorem 6.1.5. Let $R = \tau_{\mathrm{mix}}(\mathbf{A})$ so that $\|\mathbf{A}^R\mathbf{P}\|_1 \leq 1/2$ by (6.20). Let $\delta = 1/2$ and $s = \lfloor m/2 \rfloor$. Then, we have $\delta\|\mathbf{A}^R\mathbf{P}\|_1^2 + (1-\delta) \leq 3/4$ and $m - s \geq m/2$. For $0 \leq \ell \leq R-1$, we can use the (very loose) bound $\|\mathbf{A}^\ell\mathbf{P}\|_1 \leq 2$, so that $\big(\sum_{\ell=0}^{R-1} \|\mathbf{A}^\ell\mathbf{P}\|_1\big)^2 \leq 4R^2$. Furthermore, we can bound the sum in multiples of the mixing time by a geometric series: $\big(\sum_{u=0}^{\lceil (t-t_{\mathrm{b}})/R \rceil} \|\mathbf{A}^R\mathbf{P}\|_1^u\big)^2 \leq \big(\sum_{u=0}^{\infty} (1/2)^u\big)^2 = 4$. By applying Theorem 6.6.2 with these

parameter choices and simplifications, we obtain the bound

$$\left|\left|\left|\overline{\mathbf{x}}_t - \mathbf{v}\right|\right|\right|^2 \leq \frac{16\tau_{\mathrm{mix}}^2(\mathbf{A})}{(t-t_{\mathrm{b}})^2}\left\{\left(1+\frac{12t}{m}\right)\left(\frac{3}{4}\right)^{\left\lfloor \frac{t_{\mathrm{b}}}{\tau_{\mathrm{mix}}(\mathbf{A})}\right\rfloor}\|\mathbf{x}_0-\mathbf{v}\|_1^2 + \frac{16t}{m}\left(\sum_{i=\lceil m/2\rceil}^{n}\mathbf{v}^{\downarrow}(i)\right)^2\right\}.$$

The stated bound in Theorem 6.1.5 follows from bounding $\mathbb{E}\|\overline{\mathbf{x}}_t-\mathbf{v}\|_2^2 \leq \left|\left|\left|\overline{\mathbf{x}}_t-\mathbf{v}\right|\right|\right|^2$ by (6.31), and inserting the numerical simplifications $t_{\mathrm{b}} = \lfloor t/2\rfloor$ and $\sqrt{3/4} \leq 7/8$.

Finally, it remains to verify the lower bound on m required. Since $12\|\mathbf{AP}\|_1^2 \leq 48$ and $(1-\delta)(1-\|\mathbf{A}^R\mathbf{P}\|_1^2) \geq 3/8$, the condition $m - s \geq m/2 \geq 128$ suffices. This completes the proof. □

Note that if $\alpha_1(\mathbf{A}) < 1$, then by using the submultiplicativity of the ℓ^1 contractivity coefficients (Proposition 6.4.3), we deduce that $\alpha_r(\mathbf{A}) \leq \alpha_1^r(\mathbf{A}) \leq 1/4$ whenever $r \geq \log(4)/\log(1/\alpha_1(\mathbf{A}))$, which implies that $\tau_{\mathrm{mix}}(\mathbf{A}) \leq \lceil \log(4)/\log(1/\alpha_1(\mathbf{A}))\rceil$.

Remark 6.6.3 (Asymptotic rate with large m)**.** Note that if $m = n$, then the randomly sparsified power iterations reduces to the usual power iteration (6.1), which we know has error decaying like $|\lambda_2(\mathbf{A})|^{2t_{\mathrm{b}}}$ asymptotically from (6.4). From Theorem 6.6.2, we can deduce a similar qualitative prediction for the randomly sparsified power method if m is very large. More precisely, by choosing $\delta = 1-\|\mathbf{A}^R\mathbf{P}\|_1^2$ and $s = \lfloor m/2\rfloor$ in Theorem 6.6.2 and simplifying the expression, we deduce that if

$$m \geq \frac{24\|\mathbf{AP}\|_1^2\tau_{\mathrm{mix}}(\mathbf{A})}{\|\mathbf{A}^R\mathbf{P}\|_1^2\left(1-\|\mathbf{A}^R\mathbf{P}\|_1^2\right)},$$

then the contribution of the initial error $\|\mathbf{x}_0-\mathbf{v}\|_1^2$ towards the error $\left|\left|\left|\overline{\mathbf{x}}_t-\mathbf{v}\right|\right|\right|^2$ decays as $\left(2\|\mathbf{A}^R\mathbf{P}\|_1^2\right)^{t_{\mathrm{b}}/R} \sim |\lambda_2(\mathbf{A})|^{2t_{\mathrm{b}}}$ as $R \to \infty$, where we use the fact that $\|\mathbf{A}^R\mathbf{P}\|_1^{1/R} \sim |\lambda_2(\mathbf{A})|$. Thus, the bound can capture the correct asymptotic dependence of the bias component on the second eigenvalue of $\mathbf{A}$.

6.6.2 Proof outline

The proof of Theorem 6.6.2 is based on a bias-variance decomposition for the triple norm. For any $\mathbf{u} \in \mathbb{C}^n$ with $\|\mathbf{u}\|_\infty \leq 1$, note that we have the following equality:

$$\mathbb{E}\left|\mathbf{u}^*\left(\overline{\mathbf{x}}_t - \mathbf{v}\right)\right|^2 = \left|\mathbf{u}^*\left(\mathbb{E}\left[\overline{\mathbf{x}}_t\right] - \mathbf{v}\right)\right|^2 + \mathbb{E}\left|\mathbf{u}^*\left(\overline{\mathbf{x}}_t - \mathbb{E}\left[\overline{\mathbf{x}}_t\right]\right)\right|^2.$$

By taking the supremum over $\mathbf{u}$, it follows that

$$\left|\left|\left|\overline{\mathbf{x}}_t - \mathbf{v}\right|\right|\right|^2 \leq \|\mathbb{E}[\overline{\mathbf{x}}_t] - \mathbf{v}\|_1^2 + \left|\left|\left|\overline{\mathbf{x}}_t - \mathbb{E}[\overline{\mathbf{x}}_t]\right|\right|\right|^2. \tag{6.33}$$

The bias is bounded in Section 6.6.3, and a general variance bound is derived in Section 6.6.4. These are sufficient to derive a simple bound (Proposition 6.6.7) showing that the randomly sparsified power method converges with Monte Carlo rates for any choice of sparsification parameter m, which is discussed in Section 6.6.5.

In order to prove the more fine-grained bound in Theorem 6.6.2, the most technically involved part of the proof is to control the variance more carefully. To this end, fixed-time error bounds on $\left|\left|\left|\mathbf{x}_T - \mathbf{v}\right|\right|\right|^2$ with improved rates are obtained in Section 6.6.6 by resolving a recursive inequality. These are then leveraged in Section 6.6.7 to bound the variance of the tail-averaged iterate $\overline{\mathbf{x}}_t$, which can then be combined with the bias-variance decomposition (6.33) to complete the proof.

6.6.3 Analysis of bias

The bias of the randomly sparsified power method is the same as the error of the deterministic power method with tail-averaging.

Lemma 6.6.4 (Bias bound). *The randomly sparsified power method (6.6) produces a tail-averaged iterate* $\overline{\mathbf{x}}_t = (t - t_{\mathrm{b}})^{-1} \sum_{r=t_{\mathrm{b}}+1}^{t} \mathbf{x}_r$ *that satisfies*

$$\|\mathbb{E}[\overline{\mathbf{x}}_t] - \mathbf{v}\|_1 \leq \frac{\alpha_{t_{\mathrm{b}}+1}(\mathbf{A})}{t - t_{\mathrm{b}}} \left[\sum_{r=0}^{t-t_{\mathrm{b}}-1} \alpha_r(\mathbf{A}) \right] \cdot \|\mathbf{x}_0 - \mathbf{v}\|_1.$$

Proof. Observe that $\mathbb{E}[\mathbf{x}_r] = \mathbf{A}^r \mathbf{x}_0$ for all $r \geq 0$. Therefore,

$$\mathbb{E}[\overline{\mathbf{x}}_t] - \mathbf{v} = \frac{1}{t - t_{\mathrm{b}}} (\mathbf{A}^{t_{\mathrm{b}}+1} + \ldots + \mathbf{A}^t)(\mathbf{x}_0 - \mathbf{v}).$$

Note that $\mathbf{1}^{\mathsf{T}} \mathbf{A}^r (\mathbf{x}_0 - \mathbf{v}) = \mathbf{1}^{\mathsf{T}} (\mathbf{x}_0 - \mathbf{v}) = 0$ for all $r \geq 0$. Thus, applying the triangle inequality and the definition of the ℓ^1 contraction coefficients $\alpha_r(\mathbf{A})$ yields

$$\begin{aligned} \|\mathbb{E}[\overline{\mathbf{x}}_t] - \mathbf{v}\|_1 &\leq \frac{1}{t - t_{\mathrm{b}}} \sum_{r=t_{\mathrm{b}}+1}^{t} \|\mathbf{A}^r (\mathbf{x}_0 - \mathbf{v})\|_1 \\ &\leq \frac{1}{t - t_{\mathrm{b}}} \left[\sum_{r=t_{\mathrm{b}}+1}^{t} \alpha_r(\mathbf{A}) \right] \|\mathbf{x}_0 - \mathbf{v}\|_1. \end{aligned}$$

Finally, we use the submultiplicativity of the ℓ^1 contraction coefficients $\alpha_r(\mathbf{A})$ (Proposition 6.4.3, part 2) to complete the proof. □

6.6.4 Analysis of variance: general framework

Next, we provide a general bound on the variance of the randomly sparsified power method in terms of the sparsification errors that are induced at each step.

Lemma 6.6.5 (General variance bound). *The randomly sparsified power method (6.6) produces a tail-averaged iterate* $\overline{\mathbf{x}}_t = (t - t_{\mathrm{b}})^{-1} \sum_{r=t_{\mathrm{b}}+1}^{t} \mathbf{x}_r$ *that satisfies*

$$\left|\!\left|\!\left| \overline{\mathbf{x}}_t - \mathbb{E}[\overline{\mathbf{x}}_t] \right|\!\right|\!\right|^2 \leq \frac{1}{(t - t_{\mathrm{b}})^2} \sum_{r=1}^{t} \left[\sum_{\ell=\max\{t_{\mathrm{b}}+1, r\}}^{t} \|\mathbf{A}^{\ell-r} \mathbf{P}\|_1 \right]^2 \left|\!\left|\!\left| \boldsymbol{\varphi}_r(\mathbf{y}_r) - \mathbf{y}_r \right|\!\right|\!\right|^2.$$

Proof. Introduce the Doob martingale $\mathbf{M}_r = \mathbb{E}[\overline{\mathbf{x}}_t \,|\, \mathbf{x}_1, \dots, \mathbf{x}_r]$, and write

$$\overline{\mathbf{x}}_t - \mathbb{E}[\overline{\mathbf{x}}_t] = \sum_{r=1}^{t} (\mathbf{M}_r - \mathbf{M}_{r-1}).$$

For any $\mathbf{u} \in \mathbb{C}^n$ with $\|\mathbf{u}\|_\infty \leq 1$, it follows that

$$\mathbb{E}\big|\mathbf{u}^*\big(\overline{\mathbf{x}}_t - \mathbb{E}[\overline{\mathbf{x}}_t]\big)\big|^2 = \sum_{r=1}^{t} \mathbb{E}\big|\mathbf{u}^*\big(\mathbf{M}_r - \mathbf{M}_{r-1}\big)\big|^2.$$

We will proceed to bound each of the summands $\big|\mathbf{u}^*\big(\mathbf{M}_r - \mathbf{M}_{r-1}\big)\big|^2$. For $r = 1, 2, \dots, t$, a calculation using the unbiasedness of the random sparsification operator yields

$$\begin{aligned}\mathbf{M}_r - \mathbf{M}_{r-1} &= \frac{1}{t - t_\mathrm{b}} \sum_{\ell=\max\{t_\mathrm{b}+1, r\}}^{t} \big(\mathbb{E}[\mathbf{x}_\ell \,|\, \mathbf{x}_r] - \mathbb{E}[\mathbf{x}_\ell \,|\, \mathbf{x}_{r-1}]\big) \\ &= \frac{1}{t - t_\mathrm{b}} \sum_{\ell=\max\{t_\mathrm{b}+1, r\}}^{t} \mathbf{A}^{\ell - r}(\mathbf{x}_r - \mathbf{y}_r).\end{aligned}$$

Therefore, since $\mathbf{x}_r = \boldsymbol{\varphi}_r(\mathbf{y}_r)$,

$$\big|\mathbf{u}^*\big(\mathbf{M}_r - \mathbf{M}_{r-1}\big)\big| \leq \frac{1}{t - t_\mathrm{b}} \sum_{\ell=\max\{t_b+1, r\}}^{t} \big|\mathbf{u}^* \mathbf{A}^{\ell - r}(\boldsymbol{\varphi}_r(\mathbf{y}_r) - \mathbf{y}_r)\big|.$$

Since $\mathbf{1}^\mathsf{T}\big(\boldsymbol{\varphi}_r(\mathbf{y}_r) - \mathbf{y}_r\big) = 0$, $\mathbf{A}^{\ell - r}(\boldsymbol{\varphi}_r(\mathbf{y}_r) - \mathbf{y}_r) = \mathbf{A}^{\ell - r}\mathbf{P}(\boldsymbol{\varphi}_r(\mathbf{y}_r) - \mathbf{y}_r)$. Because $\|\mathbf{u}\|_\infty \leq 1$, the vector $(\mathbf{A}^{\ell - r}\mathbf{P})^*\mathbf{u}$ satisfies $\|(\mathbf{A}^{\ell - r}\mathbf{P})^*\mathbf{u}\|_\infty \leq \|\mathbf{A}^{\ell - r}\mathbf{P}\|_1$. Therefore,

$$\begin{aligned}\big|\mathbf{u}^* \mathbf{A}^{\ell - r}(\boldsymbol{\varphi}_r(\mathbf{y}_r) - \mathbf{y}_r)\big| &= \big|\mathbf{u}^* \mathbf{A}^{\ell - r}\mathbf{P}(\boldsymbol{\varphi}_r(\mathbf{y}_r) - \mathbf{y}_r)\big| \\ &\leq \|\mathbf{A}^{\ell - r}\mathbf{P}\|_1 \cdot \big|\big|\big|\boldsymbol{\varphi}_r(\mathbf{y}_r) - \mathbf{y}_r\big|\big|\big|.\end{aligned}$$

Putting this all together, we have shown that for any $\mathbf{u} \in \mathbb{C}^n$ with $\|\mathbf{u}\|_\infty \le 1$,

$$\begin{aligned}\mathbb{E}\left|\mathbf{u}^*\left(\overline{\mathbf{x}}_t - \mathbb{E}[\overline{\mathbf{x}}_t]\right)\right|^2 &= \sum_{r=1}^{t} \mathbb{E}\left|\mathbf{u}^*\left(\mathbf{M}_r - \mathbf{M}_{r-1}\right)\right|^2 \\ &\le \frac{1}{(t-t_\mathrm{b})^2} \sum_{r=1}^{t} \left[\sum_{\ell=\max\{t_\mathrm{b}+1,r\}}^{t} \|\mathbf{A}^{\ell-r}\mathbf{P}\|_1\right]^2 \left|\!\left|\!\left|\boldsymbol{\varphi}_r(\mathbf{y}_r) - \mathbf{y}_r\right|\!\right|\!\right|^2.\end{aligned}$$

Since the right hand side is independent of $\mathbf{u}$, we can take another supremum over $\mathbf{u}$ to deduce that it is also an upper bound for $\left|\!\left|\!\left|\overline{\mathbf{x}}_t - \mathbb{E}[\overline{\mathbf{x}}_t]\right|\!\right|\!\right|^2$. □

6.6.5 Variance bounds with Monte Carlo rates

Since each $\mathbf{y}_r$ is a stochastic vector, combining Lemmas 6.6.5 and 6.4.2 immediately implies that the variance is of order $O(1/m)$ for any choice of sparsification parameter m.

Corollary 6.6.6 (Variance bound I)**.** *The randomly sparsified power method* (6.6) *produces a tail-averaged iterate* $\overline{\mathbf{x}}_t = (t-t_\mathrm{b})^{-1}\sum_{r=t_\mathrm{b}+1}^{t} \mathbf{x}_r$ *that satisfies*

$$\left|\!\left|\!\left|\overline{\mathbf{x}}_t - \mathbb{E}[\overline{\mathbf{x}}_t]\right|\!\right|\!\right|^2 \le \frac{1}{m}\frac{t}{(t-t_\mathrm{b})^2}\left[\sum_{r=0}^{t-t_\mathrm{b}-1} \|(\mathbf{AP})^r\|_1\right]^2.$$

Proof. Since $\mathbf{1}^\mathsf{T}\mathbf{y}_r = 1$ for any r, Lemma 6.4.2 with E chosen to be the empty set implies that

$$\left|\!\left|\!\left|\boldsymbol{\varphi}_r(\mathbf{y}_r) - \mathbf{y}_r\right|\!\right|\!\right|^2 \le \frac{1}{m}\left(\sum_{i=1}^{n} \mathbf{y}_r(i)\right)^2 \le \frac{1}{m}.$$

Therefore, from Lemma 6.6.5, we deduce that

$$\begin{aligned}\left|\!\left|\!\left|\overline{\mathbf{x}}_t - \mathbb{E}[\overline{\mathbf{x}}_t]\right|\!\right|\!\right|^2 &\le \frac{1}{m}\frac{1}{(t-t_\mathrm{b})^2}\sum_{r=1}^{t}\left[\sum_{\ell=\max\{t_\mathrm{b}+1,r\}}^{t} \|\mathbf{A}^{\ell-r}\mathbf{P}\|_1\right]^2 \\ &\le \frac{1}{m}\frac{1}{(t-t_\mathrm{b})^2}\sum_{r=1}^{t}\left[\sum_{\ell=0}^{t-t_\mathrm{b}-1} \|\mathbf{A}^{\ell}\mathbf{P}\|_1\right]^2.\end{aligned}$$

□

By combining Lemma 6.6.4 and Corollary 6.6.6, we can immediately that the randomly sparsified power method converges with a bias component that decays in terms of the ℓ^1 contraction coefficients $\alpha_r(\mathbf{A})$ from Definition 6.1.1—which can be compared with the bound (6.4) for the vanilla power method—and a variance component that scales with Monte Carlo rates with respect to the sparsification parameter m (i.e., $m^{-1/2}$).

Proposition 6.6.7 (Monte Carlo rates)**.** *For any $t > t_{\mathrm{b}}$, the randomly sparsified power method (6.6) with any sparsification parameter $m \in \mathbb{N}$ produces a tail-averaged iterate $\overline{\mathbf{x}}_t = (t - t_{\mathrm{b}})^{-1} \sum_{r=t_{\mathrm{b}}+1}^{t} \mathbf{x}_r$ that satisfies*

$$\left|\!\left|\!\left| \overline{\mathbf{x}}_t - \mathbf{v} \right|\!\right|\!\right|^2 \leq \left[\sum_{r=0}^{t-t_{\mathrm{b}}-1} \alpha_r(\mathbf{A}) \right]^2 \left\{ \frac{\alpha^2_{t_{\mathrm{b}}+1}(\mathbf{A})}{(t-t_{\mathrm{b}})^2} \cdot \|\mathbf{x}_0 - \mathbf{v}\|_1^2 + \frac{4t}{m(t-t_{\mathrm{b}})^2} \right\}.$$

Proof. By inserting the bounds in Lemma 6.6.4 and Corollary 6.6.6 into the bias-variance decomposition (6.33), we obtain

$$\left|\!\left|\!\left| \overline{\mathbf{x}}_t - \mathbf{v} \right|\!\right|\!\right|^2 \leq \frac{\alpha^2_{t_{\mathrm{b}}+1}(\mathbf{A})}{(t-t_{\mathrm{b}})^2} \left[\sum_{r=0}^{t-t_{\mathrm{b}}-1} \alpha_r(\mathbf{A}) \right]^2 \cdot \|\mathbf{x}_0 - \mathbf{v}\|_1^2 + \frac{1}{m} \frac{t}{(t-t_{\mathrm{b}})^2} \left[\sum_{r=0}^{t-t_{\mathrm{b}}-1} \|(\mathbf{AP})^r\|_1 \right]^2.$$

By using the inequality $\|\mathbf{A}^r\mathbf{P}\|_1 \leq 2\alpha_r(\mathbf{A})$ from Proposition 6.4.3, part 3, and simplifying the expression, the proof is completed. □

6.6.6 Error bounds at fixed times

To prove the fine-grained bound in Theorem 6.6.2, we will need to control the variance more carefully. We begin by obtaining bounds on the error $\left|\!\left|\!\left| \mathbf{x}_T - \mathbf{v} \right|\!\right|\!\right|^2$ at any fixed time T. The general variance bound from Lemma 6.6.5, combined with Lemma 6.4.2, implies that the variance of $\mathbf{x}_t$ is bounded by the mean squared error of a subset of the entries of all the iterates $\mathbf{y}_r = \mathbf{A}\mathbf{x}_{r-1}$ prior to $\mathbf{x}_t$. We would like to transfer this error to the target eigenvector $\mathbf{v}$, which is the limit of the usual deterministic power iterations: $\mathbb{E}\mathbf{y}_r = \mathbf{A}^r\mathbf{x}_0 \to \mathbf{v}$ as $r \to \infty$. The first technical observation provides a link between the error of $\mathbf{y}_r$ and $\mathbf{v}$.

Lemma 6.6.8. *For any iterate $\mathbf{y}_r$ of the randomly sparsified power method (6.6) and subset of indices $\mathsf{E} \subseteq \{1, 2, \dots, n\}$,*

$$\mathbb{E}\big|\mathbf{1}_{\mathsf{E}^c}^{\mathsf{T}}\mathbf{y}_r\big|^2 \le 2\big|\mathbf{1}_{\mathsf{E}^c}^{\mathsf{T}}\mathbf{v}\big|^2 + 2\mathbb{E}\big|\mathbf{1}_{\mathsf{E}^c}^{\mathsf{T}}(\mathbf{y}_r - \mathbf{v})\big|^2.$$

In particular, this implies that

$$\mathbb{E}\big|\mathbf{1}_{\mathsf{E}^c}^{\mathsf{T}}\mathbf{y}_r\big|^2 \le 2\big|\mathbf{1}_{\mathsf{E}^c}^{\mathsf{T}}\mathbf{v}\big|^2 + 2\big|\!\big|\!\big|\mathbf{y}_r - \mathbf{v}\big|\!\big|\!\big|^2.$$

Proof. The first displayed equation follows from applying the triangle inequality:

$$\mathbb{E}\big|\mathbf{1}_{\mathsf{E}^c}^{\mathsf{T}}\mathbf{y}_r\big|^2 = \mathbb{E}\big|\mathbf{1}_{\mathsf{E}^c}^{\mathsf{T}}\mathbf{v} + \mathbf{1}_{\mathsf{E}^c}^{\mathsf{T}}(\mathbf{y}_r - \mathbf{v})\big|^2 \le 2\big|\mathbf{1}_{\mathsf{E}^c}^{\mathsf{T}}\mathbf{v}\big|^2 + 2\mathbb{E}\big|\mathbf{1}_{\mathsf{E}^c}^{\mathsf{T}}(\mathbf{y}_r - \mathbf{v})\big|^2.$$

The second displayed equation follows from the observation $\|\mathbf{1}_{\mathsf{E}^c}^{\mathsf{T}}\|_\infty \le 1$ and the definition of the triple norm. □

The next technical result bounds the error $\big|\!\big|\!\big|\mathbf{x}_T - \mathbf{v}\big|\!\big|\!\big|^2$ at any fixed time T recursively in terms of the error incurred previously, up to any reference point $t < T$.

Lemma 6.6.9 (Recursive bound)**.** *For any $t < T$ and $s < m$, the iterate $\mathbf{x}_T$ of the randomly sparsified power method (6.6) satisfies*

$$\begin{aligned}\big|\!\big|\!\big|\mathbf{x}_T - \mathbf{v}\big|\!\big|\!\big|^2 \le \|\mathbf{A}^{T-t}\mathbf{P}\|_1^2 \cdot \big|\!\big|\!\big|\mathbf{x}_t - \mathbf{v}\big|\!\big|\!\big|^2 + \frac{2\|\mathbf{A}\mathbf{P}\|_1^2}{m-s}\sum_{r=t}^{T-1}\|\mathbf{A}^{T-r-1}\mathbf{P}\|_1^2 \cdot \big|\!\big|\!\big|\mathbf{x}_r - \mathbf{v}\big|\!\big|\!\big|^2 \\ + \frac{2}{m-s}\left(\sum_{r=0}^{T-t-1}\|\mathbf{A}^r\mathbf{P}\|_1^2\right)\left(\sum_{i=s+1}^{n}\mathbf{v}^{\downarrow}(i)\right)^2.\end{aligned}$$

Proof. Let $\mathbb{E}_t$ denote the expectation conditional on $\mathbf{x}_t$. Then, for any $\mathbf{u} \in \mathbb{C}^n$ with $\|\mathbf{u}\|_\infty \le 1$, we have the conditional bias-variance decomposition:

$$\mathbb{E}_t\big|\mathbf{u}^*\big(\mathbf{x}_T - \mathbf{v}\big)\big|^2 = \big|\mathbf{u}^*\big(\mathbb{E}_t[\mathbf{x}_T] - \mathbf{v}\big)\big|^2 + \mathbb{E}_t\big|\mathbf{u}^*\big(\mathbf{x}_T - \mathbb{E}_t\big[\mathbf{x}_t\big]\big)\big|^2.$$

First, for the bias term, note that $\mathbb{E}_t[\mathbf{x}_T] = \mathbf{A}^{T-t}\mathbf{x}_t$. Since $\mathbf{1}^\mathsf{T}(\mathbf{x}_t - \mathbf{v}) = 0$, $\mathbf{A}^{T-t}(\mathbf{x}_t - \mathbf{v}) = \mathbf{A}^{T-t}\mathbf{P}(\mathbf{x}_t - \mathbf{v})$. Furthermore, since $\|\mathbf{u}\|_\infty \le 1$, $\|(\mathbf{A}^{T-t}\mathbf{P})^*\mathbf{u}\|_\infty \le \|\mathbf{A}^{T-t}\mathbf{P}\|_1$. Thus, we have

$$\begin{aligned}\big|\mathbf{u}^*\big(\mathbb{E}_t[\mathbf{x}_T] - \mathbf{v}\big)\big|^2 &= \big|\mathbf{u}^*\mathbf{A}^{T-t}\mathbf{P}\big(\mathbf{x}_t - \mathbf{v}\big)\big|^2 \\ &\le \|\mathbf{A}^{T-t}\mathbf{P}\|_1^2 \cdot \big|\!\big|\!\big|\mathbf{x}_t - \mathbf{v}\big|\!\big|\!\big|^2.\end{aligned} \tag{6.34}$$

Next, for the variance component, we can apply Lemma 6.6.5, conditional on $\mathbf{x}_t$ (i.e., treating $\mathbf{x}_t$ as the fixed initial iterate) and without tail averaging (i.e., $t_\mathrm{b} = t-1$), to obtain

$$\mathbb{E}_t\big|\mathbf{u}^*\big(\mathbf{x}_T - \mathbb{E}_t\big[\mathbf{x}_T\big]\big)\big|^2 \le \sum_{r=t+1}^{T} \|\mathbf{A}^{T-r}\mathbf{P}\|_1^2 \cdot \big|\!\big|\!\big|\boldsymbol{\varphi}_r(\mathbf{y}_r) - \mathbf{y}_r\big|\!\big|\!\big|^2.$$

Then, for any subset $\mathsf{E} \subseteq \{1, 2, \ldots, n\}$ with $|\mathsf{E}| = s$, applying Lemma 6.4.2 yields

$$\mathbb{E}_t\big|\mathbf{u}^*\big(\mathbf{x}_T - \mathbb{E}_t\big[\mathbf{x}_T\big]\big)\big|^2 \le \frac{1}{m-s}\sum_{r=t+1}^{T} \|\mathbf{A}^{T-r}\mathbf{P}\|_1^2 \cdot \mathbb{E}_t|\mathbf{1}_{\mathsf{E}^c}^\mathsf{T}\mathbf{y}_r|^2.$$

In particular, by choosing E to be the set of indices corresponding to the s largest entries of $\mathbf{v}$ and applying Lemma 6.6.8, conditional on $\mathbf{x}_t$, we deduce that

$$\begin{aligned}\mathbb{E}_t\big|\mathbf{u}^*\big(\mathbf{x}_T - \mathbb{E}_t\big[\mathbf{x}_T\big]\big)\big|^2 \le{}& \frac{2}{m-s}\left(\sum_{r=0}^{T-t-1} \|\mathbf{A}^r\mathbf{P}\|_1^2\right)\left(\sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i)\right)^2 \\ &+ \frac{2}{m-s}\sum_{r=t+1}^{T} \|\mathbf{A}^{T-r}\mathbf{P}\|_1^2 \cdot \mathbb{E}_t\big|\mathbf{1}_{\mathsf{E}^c}^\mathsf{T}(\mathbf{y}_r - \mathbf{v})\big|^2.\end{aligned}$$

Note that $\|\mathbf{1}_{\mathsf{E}^c}\|_\infty \le 1$, so $\big|\mathbf{1}_{\mathsf{E}^c}^\mathsf{T}(\mathbf{y}_r - \mathbf{v})\big| \le \big|\!\big|\!\big|\mathbf{y}_r - \mathbf{v}\big|\!\big|\!\big|$. Since $\mathbf{y}_r = \mathbf{A}\mathbf{x}_{r-1}$, a similar argument as (6.34) implies that

$$\big|\!\big|\!\big|\mathbf{y}_r - \mathbf{v}\big|\!\big|\!\big| = \big|\!\big|\!\big|\mathbf{A}(\mathbf{x}_{r-1} - \mathbf{v})\big|\!\big|\!\big| = \big|\!\big|\!\big|\mathbf{A}\mathbf{P}(\mathbf{x}_{r-1} - \mathbf{v})\big|\!\big|\!\big| \le \|\mathbf{A}\mathbf{P}\|_1 \cdot \big|\!\big|\!\big|\mathbf{x}_{r-1} - \mathbf{v}\big|\!\big|\!\big| \tag{6.35}$$

Therefore, by taking the full expectation, we deduce that

$$\begin{aligned}\mathbb{E}\left|\mathbf{u}^*\left(\mathbf{x}_T - \mathbb{E}_t\left[\mathbf{x}_T\right]\right)\right|^2 \le \frac{2}{m-s}\left(\sum_{r=0}^{T-t-1}\|\mathbf{A}^r\mathbf{P}\|_1^2\right)\left(\sum_{i=s+1}^{n}\mathbf{v}^{\downarrow}(i)\right)^2 \\ + \frac{2\|\mathbf{AP}\|_1^2}{m-s}\sum_{r=t+1}^{T}\|\mathbf{A}^{T-r}\mathbf{P}\|_1^2 \cdot \left|\!\left|\!\left|\mathbf{x}_{r-1} - \mathbf{v}\right|\!\right|\!\right|^2.\end{aligned} \tag{6.36}$$

Combining (6.34) and (6.36) shows that

$$\begin{aligned}\mathbb{E}\left|\mathbf{u}^*\left(\mathbf{x}_T - \mathbf{v}\right)\right|^2 \le \|\mathbf{A}^{T-t}\mathbf{P}\|_1^2 \cdot \left|\!\left|\!\left|\mathbf{x}_t - \mathbf{v}\right|\!\right|\!\right|^2 + \frac{2\|\mathbf{AP}\|_1^2}{m-s}\sum_{r=t}^{T-1}\|\mathbf{A}^{T-r-1}\mathbf{P}\|_1^2 \cdot \left|\!\left|\!\left|\mathbf{x}_r - \mathbf{v}\right|\!\right|\!\right|^2 \\ + \frac{2}{m-s}\left(\sum_{r=0}^{T-t-1}\|\mathbf{A}^r\mathbf{P}\|_1^2\right)\left(\sum_{i=s+1}^{n}\mathbf{v}^{\downarrow}(i)\right)^2.\end{aligned}$$

Since the upper bound is independent of $\mathbf{u}$, we can take the supremum over $\|\mathbf{u}\|_\infty \le 1$ to deduce that it is also an upper bound for $\left|\!\left|\!\left|\mathbf{x}_T - \mathbf{v}\right|\!\right|\!\right|^2$, completing the proof. □

By resolving the recursive bound in Lemma 6.6.9, we can deduce the following fixed-time error bound on $\left|\!\left|\!\left|\mathbf{x}_T - \mathbf{v}\right|\!\right|\!\right|^2$, which is composed of a decaying bias component and a variance component that is proportional to the ℓ^1 tail mass of the limiting eigenvector $\mathbf{v}$. It requires the sparsification parameter m to be larger than a threshold of order $O(R)$, where R is a natural time scale related to the mixing time of $\mathbf{A}$ for which $\mathbf{A}^R$ is sufficiently contractive.

Lemma 6.6.10 (Fixed-time error bound)**.** *Suppose that* $\|\mathbf{A}^R\mathbf{P}\|_1 < 1$ *for some* $R \ge 1$*. If the sparsification parameter* m *satisfies*

$$m \ge \frac{12\|\mathbf{AP}\|_1^2 R}{(1-\delta)\left(1 - \|\mathbf{A}^R\mathbf{P}\|_1^2\right)} + s \quad \text{for some } \delta \in [0,1) \text{ and integer } s \in \mathbb{N}, \tag{6.37}$$

then

$$\rho := \|\mathbf{A}^R\mathbf{P}\|_1^2 + 2\left(\left(1 + \frac{4\|\mathbf{AP}\|_1^2}{m-s}\right)^R - 1\right) < \delta\|\mathbf{A}^R\mathbf{P}\|_1^2 + (1-\delta). \tag{6.38}$$

Furthermore, for any $t \geq 0$, the iterate $\mathbf{x}_{Rt}$ from the randomly sparsified power method (6.6) satisfies

$$\left|\left|\left|\mathbf{x}_{Rt} - \mathbf{v}\right|\right|\right|^2 \leq \rho^t \cdot \|\mathbf{x}_0 - \mathbf{v}\|_1^2 + \frac{3\sum_{r=0}^{R-1} \|\mathbf{A}^r\mathbf{P}\|_1^2}{1-\rho} \cdot \frac{1}{m-s}\left(\sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i)\right)^2. \tag{6.39}$$

Additionally, for any $0 < q < R$,

$$\left|\left|\left|\mathbf{x}_{Rt+q} - \mathbf{v}\right|\right|\right|^2 \leq 3\rho^t \cdot \|\mathbf{x}_0 - \mathbf{v}\|_1^2 + \frac{3\sum_{r=0}^{R-1} \|\mathbf{A}^r\mathbf{P}\|_1^2}{1-\rho} \cdot \frac{1}{m-s}\left(\sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i)\right)^2. \tag{6.40}$$

Proof. Our goal is to extract an explicit bound on $\left|\left|\left|\mathbf{x}_{Rt} - \mathbf{v}\right|\right|\right|^2$ from the discrete recursive inequality in Lemma 6.6.9. For notational simplicity, define the following scalar quantities:

- $x(t) := \left|\left|\left|\mathbf{x}_t - \mathbf{v}\right|\right|\right|^2$,
- $a(t) := \|\mathbf{A}^t\mathbf{P}\|_1^2$,
- $c := 2/(m-s)$,
- $v := \left(\sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i)\right)^2$.

Then, Lemma 6.6.9 (with $T \leftarrow t$ and $t \leftarrow 0$) is equivalent to the following recursive inequality:

$$x(t) \leq a(t)x(0) + cv\sum_{r=0}^{t-1} a(r) + ca(1)\sum_{r=0}^{t-1} a(t-r-1)x(r), \quad t \geq 0. \tag{6.41}$$

To upper bound $x(t)$, we will define two separate sequences $x_0(t)$ and $x_v(t)$. The first is a *decaying component* related to the initial error, recursively defined by

$$x_0(t) = a(t)x(0) + ca(1)\sum_{r=0}^{t-1} a(t-r-1)x_0(r), \quad t \geq 1, \tag{6.42}$$

with $x_0(0) = x(0)$. The second is a *residual component* related to the limiting vector, recursively defined by

$$x_v(t) = cv \sum_{r=0}^{t-1} a(r) + ca(1) \sum_{r=0}^{t-1} a(t-r-1)x_v(r), \quad t \geq 1, \tag{6.43}$$

with $x_v(0) = 0$. We can obtain upper bounds for these two sequences separately since (6.41), combined with an inductive argument, implies that they majorize $x(t)$ together:

$$x(t) \leq x_0(t) + x_v(t), \quad t \geq 0. \tag{6.44}$$

We will derive a bound for $x_0(t)$ and $x_v(t)$ over the first R steps. Since we only assume $a(R) < 1$, we will use the bound $a(r) \leq 2$ for $1 \leq r \leq R-1$ in the worst-case scenario. Specifically, considering $x_0(t)$ first, (6.42) implies that

$$x_0(R) \leq a(R)x(0) + 2ca(1) \sum_{r=0}^{R-1} x_0(r), \tag{6.45}$$

$$x_0(r) \leq 2x(0) + 2ca(1) \sum_{u=0}^{r-1} x_0(u), \quad 1 \leq r \leq R-1. \tag{6.46}$$

Let $\beta := 2ca(1)$. Define the function $h : \mathbb{N} \to \mathbb{R}$ by $h(0) := 1$, and for $t \geq 1$,

$$h(t) := \sum_{m=1}^{t} \beta^m \left(\sum_{\substack{t_1+\ldots+t_m=t \\ t_1,\ldots,t_m \geq 1}} 1 \right). \tag{6.47}$$

Note that $h(t)$ effectively counts the number of paths connecting 0 and t with segments of integral length at least one. By using the classical stars-and-bars technique to resolve the combinatorics, we obtain the formula

$$h(t) = \sum_{m=1}^{t} \beta^m \binom{t-1}{m-1} = \beta \sum_{m=0}^{t-1} \beta^m \binom{t-1}{m} = \beta(1+\beta)^{t-1}. \tag{6.48}$$

By using the function h, the bound for the decaying component $x_0(r)$ for $r \leq R-1$ from (6.46) can be written as follows:

$$x_0(r) \leq 2x(0) \sum_{\ell=0}^{r} h(\ell), \quad 1 \leq r \leq R-1. \tag{6.49}$$

The sum over ℓ counts the number of times the recursive inequality in the sum in (6.46) is accessed in the path from $x_0(r)$ to $x(0)$, each contributing a factor of $\beta = 2ca(1)$. Concretely, for a given ℓ, the recursion continues up to $x_0(r-\ell)$, and then the term $2x(0)$ is picked out to exit the recursion. For example, the edge case $\ell = 0$ captures the scenario where the term $2x(0)$ is immediately picked out, and the edge case $\ell = r$ captures the contribution through a path of length r entirely through the recursive sum.

Hence, by inserting (6.48) into (6.49), we have

$$x_0(r) \leq 2x(0)\left(1 + \sum_{\ell=1}^{r} \beta(1+\beta)^{\ell-1}\right) = 2x(0)(1+\beta)^r, \quad 1 \leq r \leq R-1. \tag{6.50}$$

Using this bound at the endpoint (6.45) yields

$$\begin{aligned} x_0(R) &\leq a(R)x(0) + 2\beta \sum_{r=0}^{R-1} (1+\beta)^r x(0) \\ &= \big(a(R) + 2\big((1+\beta)^R - 1\big)\big) \cdot x(0). \end{aligned} \tag{6.51}$$

Note that $\rho = a(R) + 2\big((1+\beta)^R - 1\big)$. We want $x_0(R)$ to decay, so we require $\rho < 1$.

Next, we will bound the residual component $x_v(t)$ over the first R steps. Similar to the argument above, using the bounds $a(r) \leq 2$ and $\sum_{t=0}^{r-1} a(t) \leq \sum_{t=0}^{R-1} a(t)$ for $1 \leq r \leq R-1$, (6.43) implies that

$$x_v(r) \leq cv\left(\sum_{t=0}^{R-1} a(t)\right) + 2ca(1) \sum_{u=0}^{r-1} x_v(u), \quad 1 \leq r \leq R. \tag{6.52}$$

Following a similar argument as for (6.49), we can also write the upper bound (6.52) for $x_v(r)$, $1 \le r \le R$, using the function h from (6.48) as follows:

$$\begin{aligned} x_v(r) &\le cv\left(\sum_{t=0}^{R-1} a(t)\right)\sum_{\ell=0}^{r} h(\ell) \\ &= cv\left(\sum_{t=0}^{R-1} a(t)\right)\left(1+\beta\sum_{\ell=1}^{r}(1+\beta)^{\ell-1}\right) = cv\left(\sum_{t=0}^{R-1} a(t)\right)(1+\beta)^r. \end{aligned} \tag{6.53}$$

Note that $\rho < 1$ implies that $(1+\beta)^R < 1 + (1-a(R))/2 \le 3/2$. Hence, we have shown that at the endpoint,

$$x_v(R) \le \frac{3c}{2}\left(\sum_{r=0}^{R-1} a(r)\right)v. \tag{6.54}$$

To summarize, by combining the bounds on $x_0(R)$ and $x_v(R)$ from (6.51) and (6.54), we have shown that, assuming $\rho = a(R) + 2\big((1+\beta)^R - 1\big) < 1$,

$$\begin{aligned} x(R) &\le x_0(R) + x_v(R) \\ &\le \rho \cdot x(0) + \frac{3c}{2}\left(\sum_{r=0}^{R-1} a(r)\right)v. \end{aligned} \tag{6.55}$$

Finally, to obtain a bound on $x(Rt+q)$ for some $t \ge 0$ and $0 \le q < R$, we will leverage the R-step bound (6.55). Note that Lemma 6.6.9 (with $T \leftarrow Rt+q$ and $t \leftarrow R(t-1)+q$) implies that

$$x(Rt+q) \le a(R)x(R(t-1)+q) + cv\sum_{r=0}^{R-1} a(r) + ca(1)\sum_{r=0}^{R-1} a(r)x(R(t-1)+q). \tag{6.56}$$

Observe that this takes the same form as (6.41), treating $x(R(t-1)+q)$ as the fixed initial reference point. Therefore, by applying (6.55), along with a shift in indices, we deduce that

$$x(Rt+q) \le \rho \cdot x(R(t-1)+q) + \frac{3c}{2}\left(\sum_{r=0}^{R-1} a(r)\right)v.$$

By iteratively applying this bound, we obtain

$$\begin{aligned} x(Rt+q) &\leq \rho^t \cdot x(q) + \frac{3c}{2}(1+\rho+\rho^2+\ldots+\rho^{t-1})\left(\sum_{r=0}^{R-1} a(r)\right)\cdot v \\ &\leq \rho^t \cdot x(q) + \frac{3c(1-\rho^t)\sum_{r=0}^{R-1} a(r)}{2(1-\rho)}\cdot v. \end{aligned} \tag{6.57}$$

If $q = 0$ (i.e., T is a multiple of R), then (6.57) is equivalent to the claimed bound (6.39) for $\left|\left|\left|\mathbf{x}_{Rt} - \mathbf{v}\right|\right|\right|^2$ after translating back to our problem-specific notation.

Otherwise, for $0 < q < R$, we can use the bound $x_0(q) \leq 2x(0)(1+\beta)^R$ from (6.50), recalling $(1+\beta)^R < 3/2$ under the assumption $\rho < 1$, and the bound $x_v(q) \leq cv\big(\sum_{r=0}^{R-1} a(r)\big)(1+\beta)^R$ from (6.53) to deduce that

$$x(q) \leq x_0(q) + x_v(q) \leq 3x(0) + \frac{3c}{2}\left(\sum_{r=0}^{R-1} a(r)\right)v.$$

Inserting this bound into (6.57) yields

$$x(Rt+q) \leq 3\rho^t \cdot x(0) + \frac{3c\sum_{r=1}^{R} a(r)}{2(1-\rho)}\cdot v. \tag{6.58}$$

This is equivalent to the claimed bound (6.40) for $\left|\left|\left|\mathbf{x}_{Rt+q} - \mathbf{v}\right|\right|\right|^2$ after translating back to our problem-specific notation.

To conclude, we will do some housekeeping to argue that the simplified condition (6.37) implies the bound $\rho < \delta a(R) + (1-\delta) < 1$ in (6.38). Note that for $0 \leq \Delta < 1 - a(R)$,

$$\begin{aligned} \rho < 1-\Delta &\iff \left(1+\frac{4a(1)}{m-s}\right)^R < \frac{1-a(R)-\Delta}{2} \\ &\iff m - s > \frac{4a(1)}{\left(1+\frac{1-a(R)-\Delta}{2}\right)^{1/R} - 1}. \end{aligned}$$

A standard calculus calculation furnishes the following inequality:

$$\frac{1}{R} \cdot \frac{1}{\left(1 + \frac{1-a(R)-\Delta}{2}\right)^{1/R} - 1} \leq \frac{1}{\log\left(1 + \frac{1-a(R)-\Delta}{2}\right)} \quad \text{for all } R \geq 1.$$

By using the elementary inequality $\log(1+x) > x/(1+x)$ for $x > 0$, we have

$$\frac{1}{\log\left(1 + \frac{1-a(R)-\Delta}{2}\right)} < \frac{2+1-a(R)-\Delta}{1-a(R)-\Delta} \leq \frac{3}{1-a(R)-\Delta}.$$

In particular, if we choose $\Delta = \delta(1-a(R))$, then we deduce that

$$m - s \geq \frac{12a(1)R}{(1-\delta)(1-a(R))} \quad \Longrightarrow \quad \rho < 1 - \delta(1-a(R)).$$

This completes the proof. □

6.6.7 Variance bounds with improved rates

Finally, we will use the fixed-time error bound obtained in Lemma 6.6.10 to derive a bound on the variance of the tail-averaged estimator $\overline{\mathbf{x}}_t$ from the randomly sparsified power method.

Lemma 6.6.11 (Variance bound II)**.** *Suppose that* $\|\mathbf{A}^R\mathbf{P}\|_1 < 1$. *Let* $\rho \in [0,1)$ *be defined as in Lemma 6.6.10. If the sparsification parameter* m *satisfies*

$$m \geq \frac{12\|\mathbf{AP}\|_1^2 R}{(1-\delta)\big(1 - \|\mathbf{A}^R\mathbf{P}\|_1^2\big)} + s \quad \text{for some } \delta \in [1/2, 1) \text{ and integer } s \in \mathbb{N},$$

then $\rho < \delta\|\mathbf{A}^R\mathbf{P}\|_1^2 + (1-\delta)$, *and for any time* t *after a burn-in time of* t_{b}, *we have*

$$\begin{aligned}
&\big|\!\big|\!\big|\overline{\mathbf{x}}_t - \mathbb{E}[\overline{\mathbf{x}}_t]\big|\!\big|\!\big|^2 \\
&\quad \leq \frac{t\Big(\sum_{\ell=0}^{R-1}\|\mathbf{A}^\ell\mathbf{P}\|_1\Big)^2\Big[\sum_{u=0}^{\lceil (t-t_{\mathrm{b}})/R\rceil}\|\mathbf{A}^R\mathbf{P}\|_1^u\Big]^2}{(t-t_{\mathrm{b}})^2(m-s)}\left\{6\rho^{\lfloor t_{\mathrm{b}}/R\rfloor}\|\mathbf{x}_0 - \mathbf{v}\|_1^2 + 8\left(\sum_{i=s+1}^{n}\mathbf{v}^{\downarrow}(i)\right)^2\right\}.
\end{aligned}$$

Proof. Let $\mathsf{E} \subseteq \{1, 2, \ldots, n\}$ be the set of indices corresponding to the s largest entries of $\mathbf{v}$. By using the primitive bound on the variance of the tail-averaged iterate from Lemma 6.6.5, combined with Lemmas 6.4.2 and 6.6.8, we have

$$\begin{aligned}\left|\!\left|\!\left| \overline{\mathbf{x}}_t - \mathbb{E}[\overline{\mathbf{x}}_t] \right|\!\right|\!\right|^2 \leq \frac{2}{(t-t_{\mathrm{b}})^2} \frac{1}{m-s} \sum_{r=0}^{t-1} \left[\sum_{\ell=\max\{t_{\mathrm{b}},r\}+1}^{t} \|\mathbf{A}^{\ell-r-1}\mathbf{P}\|_1 \right]^2 \left|\!\left|\!\left| \mathbf{y}_{r+1} - \mathbf{v} \right|\!\right|\!\right|^2 \\ + \frac{2t}{(t-t_{\mathrm{b}})^2} \left[\sum_{\ell=0}^{t-t_{\mathrm{b}}-1} \|\mathbf{A}^{\ell}\mathbf{P}\|_1 \right]^2 \cdot \frac{1}{m-s} \left(\sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i) \right)^2.\end{aligned} \tag{6.59}$$

Recall from (6.35) that $\mathbf{y}_{r+1} = \mathbf{A}\mathbf{x}_r$ satisfies

$$\left|\!\left|\!\left| \mathbf{y}_{r+1} - \mathbf{v} \right|\!\right|\!\right|^2 = \left|\!\left|\!\left| \mathbf{A}\mathbf{P}(\mathbf{x}_r - \mathbf{v}) \right|\!\right|\!\right|^2 \leq \|\mathbf{A}\mathbf{P}\|_1^2 \cdot \left|\!\left|\!\left| \mathbf{x}_r - \mathbf{v} \right|\!\right|\!\right|^2.$$

From Lemma 6.6.10, we have the following bound on the error at any fixed time r:

$$\left|\!\left|\!\left| \mathbf{x}_r - \mathbf{v} \right|\!\right|\!\right|^2 \leq 3\rho^{\lfloor r/R \rfloor} \cdot \|\mathbf{x}_0 - \mathbf{v}\|_1^2 + \frac{3\sum_{\ell=0}^{R-1} \|\mathbf{A}^{\ell}\mathbf{P}\|_1^2}{(1-\rho)(m-s)} \left(\sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i) \right)^2.$$

To simplify this bound, note that $1-\rho > \delta(1-\|\mathbf{A}^R\mathbf{P}\|_1^2)$. Since $\|\mathbf{A}^{\ell}\mathbf{P}\|_1 \leq 2$ for $0 \leq \ell \leq R-1$, by using our assumption on $m-s$ and $\delta \geq 1/2$, we obtain

$$\frac{3\sum_{\ell=0}^{R-1} \|\mathbf{A}^{\ell}\mathbf{P}\|_1^2}{(1-\rho)(m-s)} \leq \frac{12R}{(1-\rho)(m-s)} < \frac{(1-\delta)(1-\|\mathbf{A}^R\mathbf{P}\|_1^2)}{(1-\rho)\|\mathbf{A}\mathbf{P}\|_1^2} < \frac{1-\delta}{\delta\|\mathbf{A}\mathbf{P}\|_1^2} \leq \frac{1}{\|\mathbf{A}\mathbf{P}\|_1^2}.$$

Thus,

$$\left|\!\left|\!\left| \mathbf{y}_{r+1} - \mathbf{v} \right|\!\right|\!\right|^2 \leq 3\|\mathbf{A}\mathbf{P}\|_1^2 \rho^{\lfloor r/R \rfloor} \cdot \|\mathbf{x}_0 - \mathbf{v}\|_1^2 + 3\left(\sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i) \right)^2.$$

Inserting this bound into (6.59) yields

$$\left|\left|\left|\overline{\mathbf{x}}_t - \mathbb{E}[\overline{\mathbf{x}}_t]\right|\right|\right|^2 \le \frac{6}{(t-t_{\mathrm{b}})^2}\frac{\|\mathbf{AP}\|_1^2}{m-s}\sum_{r=0}^{t-1}\left[\sum_{\ell=\max\{t_{\mathrm{b}},r\}+1}^{t}\|\mathbf{A}^{\ell-r-1}\mathbf{P}\|_1\right]^2\rho^{\lfloor r/R\rfloor}\cdot\|\mathbf{x}_0-\mathbf{v}\|_1^2 + \frac{8t}{(t-t_{\mathrm{b}})^2}\left[\sum_{\ell=0}^{t-t_{\mathrm{b}}-1}\|\mathbf{A}^{\ell}\mathbf{P}\|_1\right]^2\cdot\frac{1}{m-s}\left(\sum_{i=s+1}^{n}\mathbf{v}^{\downarrow}(i)\right)^2. \tag{6.60}$$

It remains to control the contributions from the initial error, which is more delicate. To do this, we will split the first sum over r in (6.60) into blocks of size R (possibly including additional terms in the last block if t is not a multiple of R):

$$\sum_{r=0}^{t-1}\left[\sum_{\ell=\max\{t_{\mathrm{b}},r\}+1}^{t}\|\mathbf{A}^{\ell-r-1}\mathbf{P}\|_1\right]^2\rho^{\lfloor r/R\rfloor} \le \sum_{\tau=0}^{\lfloor t/R\rfloor-1}\sum_{r=R\tau}^{R(\tau+1)-1}\rho^{\tau}\left[\sum_{\ell=\max\{t_{\mathrm{b}},r\}+1}^{t}\|\mathbf{A}^{\ell-r-1}\mathbf{P}\|_1\right]^2.$$

We further split the sum over τ from $\tau=0$ to $\tau=\lfloor t_{\mathrm{b}}/R\rfloor-1$, and from $\tau=\lfloor t_{\mathrm{b}}/R\rfloor$ to $\tau=\lfloor t/R\rfloor-1$. For $\tau\le\lfloor t_{\mathrm{b}}/R\rfloor-1$, we will use the following bound for each block, which follows from using submultiplicativity and $\|\mathbf{A}^R\mathbf{P}\|_1<\rho^{1/2}$:

$$\begin{aligned}\sum_{r=R\tau}^{R(\tau+1)-1}\rho^{\tau}\left[\sum_{\ell=\max\{t_{\mathrm{b}},r\}+1}^{t}\|\mathbf{A}^{\ell-r-1}\mathbf{P}\|_1\right]^2 &= \sum_{r=R\tau}^{R(\tau+1)-1}\rho^{\tau}\left[\sum_{\ell=t_{\mathrm{b}}+1}^{t}\|\mathbf{A}^{\ell-r-1}\mathbf{P}\|_1\right]^2\\ &\le \sum_{r=R\tau}^{R(\tau+1)-1}\rho^{\tau}\left[\sum_{u=\lfloor t_{\mathrm{b}}/R\rfloor}^{\lceil t/R\rceil-1}\|\mathbf{A}^{R(u-\tau)}\mathbf{P}\|_1\left(\sum_{\ell=0}^{R-1}\|\mathbf{A}^{\ell}\mathbf{P}\|_1\right)\right]^2\\ &\le R\left(\sum_{\ell=0}^{R-1}\|\mathbf{A}^{\ell}\mathbf{P}\|_1\right)^2\cdot\rho^{\tau}\left[\sum_{u=\lfloor t_{\mathrm{b}}/R\rfloor}^{\lceil t/R\rceil-1}\|\mathbf{A}^{R}\mathbf{P}\|_1^{u-\tau}\right]^2\\ &\le R\left(\sum_{\ell=0}^{R-1}\|\mathbf{A}^{\ell}\mathbf{P}\|_1\right)^2\cdot\rho^{\lfloor t_{\mathrm{b}}/R\rfloor}\left[\sum_{u=0}^{\lceil t/R\rceil-\lfloor t_{\mathrm{b}}/R\rfloor-1}\|\mathbf{A}^{R}\mathbf{P}\|_1^{u}\right]^2.\end{aligned} \tag{6.61}$$

For $\tau \geq \lfloor t_{\mathrm{b}}/R \rfloor$, we will use the following bound for each block:

$$\begin{aligned}\sum_{r=R\tau}^{R(\tau+1)-1} \rho^{\tau} \left[\sum_{\ell=\max\{t_{\mathrm{b}},r\}+1}^{t} \|\mathbf{A}^{\ell-r-1}\mathbf{P}\|_1 \right]^2 &= \sum_{r=R\tau}^{R(\tau+1)-1} \rho^{\tau} \left[\sum_{\ell=r+1}^{t} \|\mathbf{A}^{\ell-r-1}\mathbf{P}\|_1 \right]^2 \\ &\leq \sum_{r=R\tau}^{R(\tau+1)-1} \rho^{\tau} \left[\sum_{u=0}^{\lceil t/R \rceil - \lfloor t_{\mathrm{b}}/R \rfloor - 1} \|\mathbf{A}^{Ru}\|_1 \left(\sum_{\ell=0}^{R-1} \|\mathbf{A}^{\ell}\mathbf{P}\|_1 \right) \right]^2 \\ &\leq R \left(\sum_{\ell=0}^{R-1} \|\mathbf{A}^{\ell}\mathbf{P}\|_1 \right)^2 \cdot \rho^{\tau} \left[\sum_{u=0}^{\lceil t/R \rceil - \lfloor t_{\mathrm{b}}/R \rfloor - 1} \|\mathbf{A}^{R}\mathbf{P}\|_1^u \right]^2 .\end{aligned} \tag{6.62}$$

For the first sum from $\tau = 0$ to $\tau = \lfloor t_{\mathrm{b}}/R \rfloor - 1$, we will apply (6.61) to each block to derive the following upper bound:

$$\begin{aligned}&\sum_{\tau=0}^{\lfloor t_{\mathrm{b}}/R \rfloor - 1} \sum_{r=R\tau}^{R(\tau+1)-1} \rho^{\tau} \left[\sum_{\ell=\max\{t_{\mathrm{b}},r\}+1}^{t} \|\mathbf{A}^{\ell-r-1}\mathbf{P}\|_1 \right]^2 \\ &\qquad \leq R\lfloor t_{\mathrm{b}}/R \rfloor \left(\sum_{\ell=0}^{R-1} \|\mathbf{A}^{\ell}\mathbf{P}\|_1 \right)^2 \cdot \left[\sum_{u=0}^{\lceil t/R \rceil - \lfloor t_{\mathrm{b}}/R \rfloor - 1} \|\mathbf{A}^{R}\mathbf{P}\|_1^u \right]^2 \cdot \rho^{\lfloor t_{\mathrm{b}}/R \rfloor}.\end{aligned} \tag{6.63}$$

For the second sum from $\tau = \lfloor t_{\mathrm{b}}/R \rfloor$ to $\tau = \lfloor t/R \rfloor - 1$, we will apply (6.62) to each block to derive the following upper bound:

$$\begin{aligned}&\sum_{\tau=\lfloor t_{\mathrm{b}}/R \rfloor}^{\lfloor t/R \rfloor - 1} \sum_{r=R\tau}^{R(\tau+1)-1} \rho^{\tau} \left[\sum_{\ell=\max\{t_{\mathrm{b}},r\}+1}^{t} \|\mathbf{A}^{\ell-r-1}\mathbf{P}\|_1 \right]^2 \\ &\qquad \leq R \left(\sum_{\ell=0}^{R-1} \|\mathbf{A}^{\ell}\mathbf{P}\|_1 \right)^2 \cdot \left[\sum_{u=0}^{\lceil t/R \rceil - \lfloor t_{\mathrm{b}}/R \rfloor - 1} \|\mathbf{A}^{R}\mathbf{P}\|_1^u \right]^2 \cdot \sum_{\tau=\lfloor t_{\mathrm{b}}/R \rfloor}^{\lfloor t/R \rfloor - 1} \rho^{\tau} \\ &\qquad \leq R(\lfloor t/R \rfloor - \lfloor t_{\mathrm{b}}/R \rfloor) \left(\sum_{\ell=0}^{R-1} \|\mathbf{A}^{\ell}\mathbf{P}\|_1 \right)^2 \cdot \left[\sum_{u=0}^{\lceil t/R \rceil - \lfloor t_{\mathrm{b}}/R \rfloor - 1} \|\mathbf{A}^{R}\mathbf{P}\|_1^u \right]^2 \cdot \rho^{\lfloor t_{\mathrm{b}}/R \rfloor}.\end{aligned} \tag{6.64}$$

We have used $\rho \leq 1$ in the last inequality. Altogether, the geometric decay from $\rho^{\lfloor t_{\mathrm{b}}/R \rfloor}$ dominates. By combining (6.63) and (6.64) in the majorization for (6.60), we have shown

that

$$\left|\left|\left|\overline{\mathbf{x}}_t - \mathbb{E}[\overline{\mathbf{x}}_t]\right|\right|\right|^2 \leq \frac{6t}{(t-t_{\mathrm{b}})^2}\frac{\|\mathbf{AP}\|_1^2}{m-s}\left(\sum_{\ell=0}^{R-1}\|\mathbf{A}^{\ell}\mathbf{P}\|_1\right)^2\left[\sum_{u=0}^{\lceil (t-t_{\mathrm{b}})/R\rceil}\|\mathbf{A}^{R}\mathbf{P}\|_1^u\right]^2\rho^{\lfloor t_{\mathrm{b}}/R\rfloor}\|\mathbf{x}_0-\mathbf{v}\|_1^2 + \frac{8t}{(t-t_{\mathrm{b}})^2}\left[\sum_{\ell=0}^{t-t_{\mathrm{b}}-1}\|\mathbf{A}^{\ell}\mathbf{P}\|_1\right]^2\frac{1}{m-s}\left(\sum_{i=s+1}^{n}\mathbf{v}^{\downarrow}(i)\right)^2.$$

We may further simplify the variance component by using submultiplicativity to peel off multiples of $\|\mathbf{A}^R\mathbf{P}\|_1$ in order to group terms. This completes the proof. □

By combining the bounds on the bias and variance of the tail-averaged estimator $\overline{\mathbf{x}}_T$ in Lemmas 6.6.4 and 6.6.11, respectively, we can now prove Theorem 6.6.2.

Proof of Theorem 6.6.2. Recall the bias-variance decomposition from (6.33):

$$\left|\left|\left|\overline{\mathbf{x}}_t - \mathbf{v}\right|\right|\right|^2 \leq \|\mathbb{E}[\overline{\mathbf{x}}_t] - \mathbf{v}\|_1^2 + \left|\left|\left|\overline{\mathbf{x}}_t - \mathbb{E}[\overline{\mathbf{x}}_t]\right|\right|\right|^2.$$

By combining the bias bound from Lemma 6.6.4 and the variance bound from Lemma 6.6.11, we deduce that

$$\left|\left|\left|\overline{\mathbf{x}}_t - \mathbf{v}\right|\right|\right|^2 \leq \frac{\alpha_{t_{\mathrm{b}}+1}^2(\mathbf{A})}{(t-t_{\mathrm{b}})^2}\left[\sum_{r=0}^{t-t_{\mathrm{b}}-1}\alpha_r(\mathbf{A})\right]^2\|\mathbf{x}_0-\mathbf{v}\|_1 + \frac{t\left(\sum_{\ell=0}^{R-1}\|\mathbf{A}^{\ell}\mathbf{P}\|_1\right)^2\left(\sum_{u=0}^{\lceil (t-t_{\mathrm{b}})/R\rceil}\|\mathbf{A}^{R}\mathbf{P}\|_1^u\right)^2}{(t-t_{\mathrm{b}})^2(m-s)}\left\{6\rho^{\lfloor t_{\mathrm{b}}/R\rfloor}\|\mathbf{x}_0-\mathbf{v}\|_1^2 + 8\left(\sum_{i=s+1}^{n}\mathbf{v}^{\downarrow}(i)\right)^2\right\}.$$

By using submultiplicativity and the fact that $\alpha_R^2(\mathbf{A}) \leq \|\mathbf{A}^R\mathbf{P}\|_1^2 < \rho$, we can simplify this expression to obtain

$$\left|\left|\left|\overline{\mathbf{x}}_t - \mathbf{v}\right|\right|\right|^2 \leq \frac{\left(\sum_{\ell=0}^{R-1}\|\mathbf{A}^{\ell}\mathbf{P}\|_1\right)^2\left(\sum_{u=0}^{\lceil (t-t_{\mathrm{b}})/R\rceil}\|\mathbf{A}^{R}\mathbf{P}\|_1^u\right)^2}{(t-t_{\mathrm{b}})^2} \times\left\{\left(1+\frac{6t}{m-s}\right)\cdot\rho^{\lfloor t_{\mathrm{b}}/R\rfloor}\cdot\|\mathbf{x}_0-\mathbf{v}\|_1^2 + \frac{8t}{m-s}\cdot\left(\sum_{i=s+1}^{n}\mathbf{v}^{\downarrow}(i)\right)^2\right\}.$$

After using the assumption on m to insert the bound $\rho < \delta\|\mathbf{A}^R\mathbf{P}\|_1^2 + (1-\delta)$, the proof is completed. □

6.7 Concluding remarks

We proved that the error of the randomly sparsified power method for computing the leading eigenvector $\mathbf{v}$ of a column-stochastic matrix $\mathbf{A}$ can be bounded independently of the matrix dimension, depending only on the mixing time of the stochastic matrix $\mathbf{A}$ and the rate of decay of the entries of the solution vector $\mathbf{v}$. We showed that the deterministically sparsified power method can only provide guaranteed accuracy control when $\mathbf{A}$ is a strict ℓ^1 contraction, and can fail for a class of hard problems.

A natural future direction is to develop mathematical guarantees for the randomly sparsified power method applied to eigenproblems where $\mathbf{A} \in \mathbb{C}^{n\times n}$ is a Hermitian matrix [LW17; Gre+19]. In this setting, the basic scheme has iterations

$$\mathbf{y}_t = \frac{\mathbf{A}\mathbf{x}_{t-1}}{\|\mathbf{A}\mathbf{x}_{t-1}\|_2},$$
$$\mathbf{x}_t = \boldsymbol{\varphi}_t(\mathbf{y}_t).$$

The main difficulty is the additional bias introduced by sparsification due to the nonlinear power iteration with normalization. It would also be interesting to prove theoretical guarantees for variants of this algorithm for computing multiple eigenvectors such as randomized subspace iteration [Gre+22a; Gre+22b]. Another direction is to develop a better understanding and characterization of the gaps between randomized methods and their deterministic counterparts in these settings. These efforts would contribute towards future developments of the mathematical and algorithmic foundations of the fast randomized iteration framework [LW17] and its applications.

6.8 Additional discussion of the Ising model

6.8.1 Background

The Ising model is a classical model of a magnet at equilibrium and phase transitions. Heuristically, at low temperatures and without an external field (i.e., large β and $h = 0$), the system is in an ordered phase, and all the spins are aligned. At high temperatures, the system is in a disordered phase, and the spins essentially behave independently.

The Ising model with zero external field ($h = 0$) has only been exactly solved in very specific cases. Notably, this includes the Ising model on the one-dimensional line by Ising [Isi25], and on the two-dimensional lattice by Onsager [Ons44], who demonstrated that the Ising model on the 2D lattice exhibits a second-order phase transition at the critical inverse temperature $\beta_c = \frac{1}{2}\log(1+\sqrt{2}) \approx 0.4407$. However, the exact solution of the Ising model on the 2D lattice with non-zero external field ($h \neq 0$), or on any higher-dimensional lattice remains open. Thus, the Ising model continues to be a source of deep mathematical questions (e.g., see [Dum22] for a recent survey).

Computing quantities related to the Ising model is a fundamental problem with applications in areas such as statistical physics [LB14], Bayesian statistics [RC04], image processing [GG84], and machine learning [FI14]. For example, quantities of interest may include the average magnetization $f(\sigma) = |V|^{-1}\sum_{v\in V}\sigma(v)$ and correlations $f(\sigma) = |V|^{-2}\sum_{u,v\in V}\sigma(u)\sigma(v)$. The main difficulty with computation is the exponentially large state space, which makes the normalization constant $Z(\beta, h)$ intractable. Therefore, Monte Carlo approaches are typically used in practice: MCMC algorithms are commonly used to generate (nearly) independent samples $\widehat{\sigma}^{(1)}, \ldots, \widehat{\sigma}^{(m)}$ of the Ising model, which can then be used to approximate expectations with respect to the Ising model by averaging. For MCMC, the goal is to design a Markov chain that quickly converges to stationarity.

We mainly consider the Glauber dynamics, which is an efficient algorithm for sampling in certain regimes where it is rapid mixing (in the precise sense that its mixing time scales

polylogarithmically in the number of vertices, as opposed to exponentially). Heuristically, the Glauber dynamics mix rapidly at high temperatures or with a large external field. There is a vast literature on proving these intuitions rigorously, which is beyond the scope of our investigations: e.g., there are sufficient conditions for rapid mixing known as the Dobrushin–Shlosman conditions [DGJ09], and a very precise picture on the lattice has been established [LS12; LS16]. We refer to [LPW17, Chapter 15] for a detailed discussion.

Additionally, we note that there are other non-local Markov chains used to sample the Ising model, such as the Swendsen–Wang and Wolff cluster algorithms [SW87; Wol89], which can be significantly more effective near the point of critical slowdown. However, the columns of the stochastic matrix associated with these dynamics are not sparse, which does not integrate very well with the random sparsification framework.

6.8.2 Additional numerics

In this section, we will present additional plots to supplement our numerical demonstration for the Ising model in Figure 6.1, which shows the ℓ^2 error. Specifically, we will consider the average magnetization error $|\mathbf{m}^\mathsf{T}\mathbf{x}_t - \mathbf{m}^\mathsf{T}\mathbf{v}|$, where the i^{th} entry of the vector $\mathbf{m} \in \mathbb{R}^{2^{\ell\times\ell}}$ denotes the average magnetization of the configuration $\sigma^{(i)}$ that it indexes, i.e., $\mathbf{m}(i) = \ell^{-2}\sum_{i=1}^{\ell\times\ell}\sigma^{(m)}(i)$. Note that this measures the error in estimates of a low-dimensional projection of $\mathbf{v}$ (i.e., an expectation with respect to the Ising measure) and $\|\mathbf{m}\|_\infty = 1$, so the error of the randomly sparsified power method is bounded by Figure 6.1.5. We will also consider the scaled ℓ^1 error $\frac{1}{2}\|\mathbf{x}_t - \mathbf{v}\|_1$, or equivalently the total variation distance between the two probability distributions $\mathbf{x}_t$ and $\mathbf{v}$, which our theory does not cover.

Figures 6.5 and 6.6 present the plots in terms of the average magnetization error and the scaled ℓ^1 error $\frac{1}{2}\|\mathbf{x}_t - \mathbf{v}\|_1$, respectively, for the Ising model on a 4×4 torus with inverse temperature $\beta = 0.45$ and external field $h = 0.25$. (With these parameters, the ground truth average magnetization equals $\mathbf{m}^\mathsf{T}\mathbf{v} = 0.841146$.) The plots verify that the same observations

made for the ℓ^2 errors of the deterministically and randomly sparsified power methods in Figure 6.1 continue to hold.

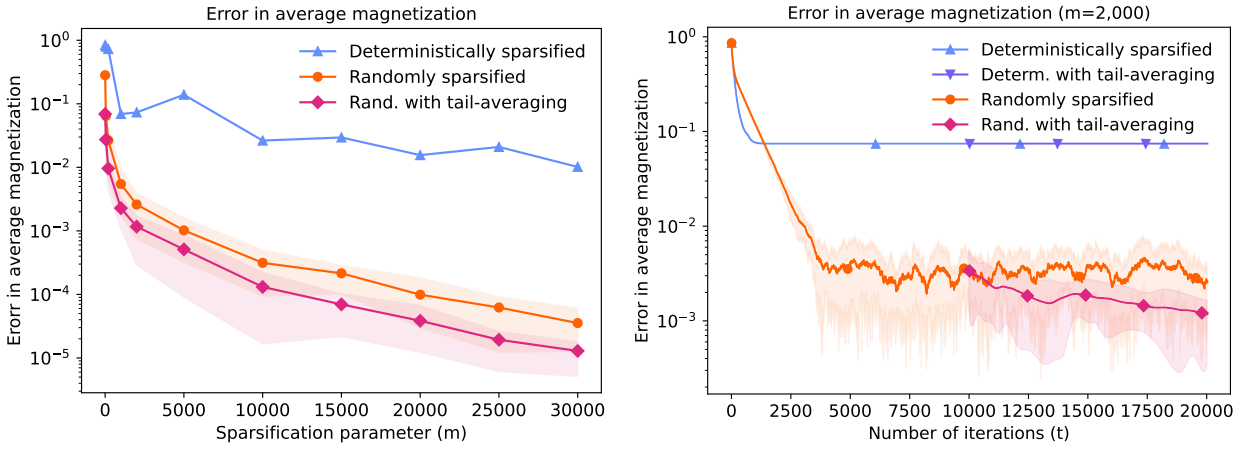


Figure 6.5: The average magnetization error $|\mathbf{m}^{\mathsf{T}}\mathbf{x}_t - \mathbf{m}^{\mathsf{T}}\mathbf{v}|$ from solving the eigenvalue problem $\mathbf{A}\mathbf{v} = \mathbf{v}$ corresponding to the Glauber dynamics for the Ising model (6.9) on a 4×4 torus in a low-temperature and strong external field regime. **(Left)** The errors at time $t = 20,000$ after a burn-in time of $t_{\mathrm{b}} = 10,000$ as a function of the sparsification parameter m. **(Right)** The dynamics of the error with $m = 2,000$. The mean errors over 30 independent runs of the randomized algorithms are reported, with the corresponding $0.2/0.8^{\mathrm{th}}$ quantiles indicated by the shaded intervals.

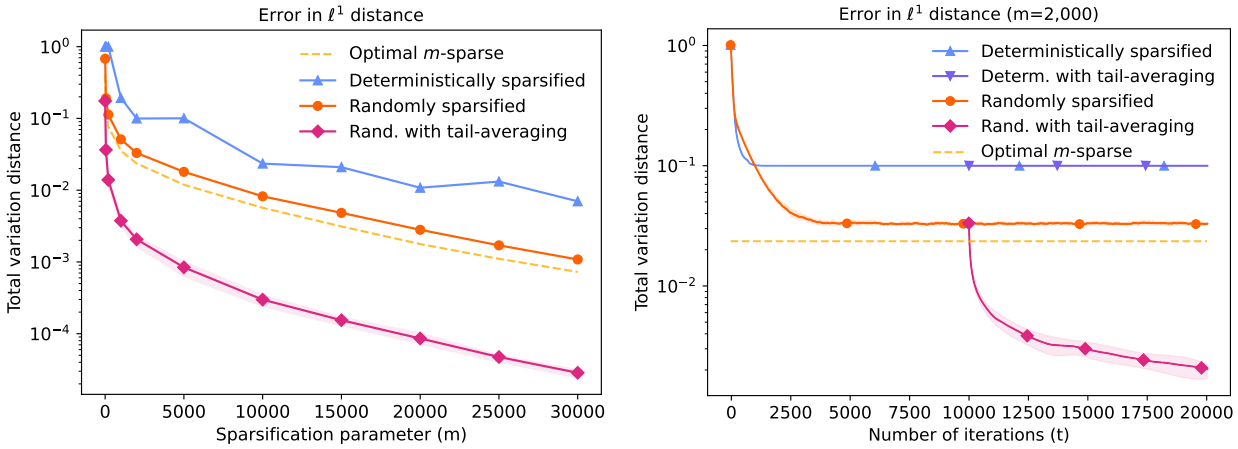


Figure 6.6: The scaled ℓ^1 error $\frac{1}{2}\|\mathbf{x}_t - \mathbf{v}\|_1$ (i.e., total variation distance) from solving the same eigenvalue problem $\mathbf{A}\mathbf{v} = \mathbf{v}$ as in Figure 6.5. The optimal m-sparse error represents the ℓ^1 error $\sum_{i=m+1}^{n} \mathbf{v}^{\downarrow}(i)$ from the best m-sparse approximation of the Ising model.

6.8.3 Approximate sparsity of the Ising model

Based on the error bound in Theorem 6.1.5, we would expect the randomly sparsified power method to be effective only if the Ising model admits a sparse approximation. This is not always the case: for example, all the configurations of the Ising model at infinitely high temperatures without an external field (i.e., $\beta = 0$ and $h = 0$) are equally likely. Therefore, the entries of the leading eigenvector $\mathbf{v}$ exhibits *no decay*, and exponentially many states $s = O(2^{\ell \times \ell})$ are needed to achieve a constant factor tail sum $\sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i) = O(1)$.

However, when the temperature is very low and the external field is very strong, we would expect the Ising model to be approximately sparse, with most of the probability mass concentrated near the configuration with all positive spins. This is established more precisely for our experimental setup on the $\ell \times \ell$ torus in the following result, which states the entries of $\mathbf{v}$ exhibit polynomial decay in such a regime. In fact, it shows that a constant factor tail sum can be achieved by a negligible proportion of the exponentially-sized state space.

Proposition 6.8.1 (Low temperature and strong external field regime)**.** *Let* $\mathbf{v} \in [0,1]^{\ell \times \ell}$ *contain the entries of the Ising model* (6.9) *with inverse temperature* $\beta > 0$ *and external field* $h > 0$ *on the* $\ell \times \ell$ *lattice. Suppose that* β *and* h *satisfy* $\beta(h+4) > \log(\ell)$*. Then, for all* $i = 1, \ldots, 2^{\ell \times \ell}$ *and* $s = 1, \ldots, 2^{\ell \times \ell}$*, we have*

$$\mathbf{v}^{\downarrow}(i) \leq C i^{-(1+c)} \quad \text{and} \quad \sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i) \leq \frac{C}{c} s^{-c}$$

with $c := \beta(h+4)/\log(\ell) - 1$ *and* $C = 2^c$*. Moreover,*

$$\mathbf{v}^{\downarrow}(1) \geq \exp\Big(-\exp\big(-2\beta(h+4) + 2\log(\ell)\big)\Big) \geq e^{-1}.$$

Proof. Let $d = \ell^2$ be the number of vertices, so that there are $n = 2^d$ states. Note in the presence of a positive external field $h > 0$, the configuration with all positive spins, denoted

by σ^+, is the unique global minimizer of the Hamiltonian with $H(\sigma^+) = -(2+h)d$, where we have used the fact that the square lattice with periodic boundary conditions is 4-regular.

For any configuration $\sigma \in \{\pm 1\}^d$, let $N^-(\sigma) := \{u : \sigma(u) = -1\}$ be the set of vertices with spin -1, and $\partial N^-(\sigma) := \{(u,v) : \sigma(u)\sigma(v) = -1\}$ be the set of edges between two opposite spins. Using the fact that the square lattice is 4-regular again, we have $|\partial N^-(\sigma)| \le 4|N^-(\sigma)|$. Hence, we have

$$\begin{aligned} H(\sigma) - H(\sigma^+) &= 2h|N^-(\sigma)| + 2|\partial N^-(\sigma)| \\ &\ge 2(h+4)|N^-(\sigma)|. \end{aligned} \tag{6.65}$$

Let $\mathcal{C}_k = \{\sigma \in \{\pm 1\}^d : |N^-(\sigma)| = k\}$ be the set of configurations with k spins that are -1. Observe that $|\mathcal{C}_k| = \binom{d}{k}$, and for all $\sigma \in \mathcal{C}_k$, $\mu_{\beta,h}(\sigma)$ takes on the same value, which we will denote by $\mu^{(k)}_{\beta,h}$. From (6.65), we have

$$\mu^{(k)}_{\beta,h} = \mu_{\beta,h}(\sigma) \le e^{-2\beta(h+4)k} \cdot \mu_{\beta,h}(\sigma^+) \le \left(e^{-2\beta(h+4)}\right)^k \quad \text{for any } \sigma \in \mathcal{C}_k. \tag{6.66}$$

From our observations so far, we deduce that the sorted entries of the Ising measure are given by

$$\mathbf{v}^{\downarrow} = (\mu_{\beta,h}(\sigma^+), \underbrace{\mu^{(1)}_{\beta,h}, \ldots, \mu^{(1)}_{\beta,h}}_{\binom{d}{1} \text{ times}}, \underbrace{\mu^{(2)}_{\beta,h}, \ldots, \mu^{(2)}_{\beta,h}}_{\binom{d}{2} \text{ times}}, \ldots, \underbrace{\mu^{(k)}_{\beta,h}, \ldots, \mu^{(k)}_{\beta,h}}_{\binom{d}{k} \text{ times}}, \ldots),$$

consisting of the $\binom{d}{k}$ configurations in the block $\mathcal{C}_k$ with k spins flipped to -1 as $k = 0, 1, 2, \ldots, d$. From this structure, we are able to obtain the exact expression for the entries of $\mathbf{v}^{\downarrow}$: for $i \ge 2$,

$$\mathbf{v}^{\downarrow}(i) = \mu^{(k(i))}_{\beta,h}, \quad \text{where} \quad k(i) = \arg\min_k \sum_{j=0}^{k} \binom{d}{j} \ge i \tag{6.67}$$

To obtain a simple bound on how $k(i)$ scales with i, note that we have, for $d \geq 2$,

$$\sum_{j=0}^{k} \binom{d}{j} \leq 1 + d + d^2 + \ldots + d^k = \frac{d^{k+1}}{d-1} \leq 2d^k.$$

Thus, by rearranging the inequality $2d^k \leq i$, we deduce that $k(i) \geq \log(i/2)/\log(d)$. From (6.67) and (6.66), we deduce that

$$\mathbf{v}^{\downarrow}(i) \leq \left(e^{-2\beta(h+4)}\right)^{k(i)} \leq \left(e^{-2\beta(h+4)}\right)^{\log(i/2)/\log(d)} = \left(\frac{i}{2}\right)^{-2\beta(h+4)/\log(d)}.$$

Thus, the entries of $\mathbf{v}^{\downarrow}$ exhibit polynomial decay as long as $2\beta(h+4) > \log(d)$, i.e., $\mathbf{v}^{\downarrow}(i) \leq Cs^{-(1+c)}$ with $c := 2\beta(h+4)/\log(d) - 1$ and $C := 2^{\alpha}$. In turn, this also implies that the tail sums exhibit polynomial decay, i.e., $\sum_{i=s+1}^{n} \mathbf{v}^{\downarrow}(i) \leq (C/c)s^{-c}$, following the same calculations as in Section 6.1.3.

In fact, we can show that in the low-temperature and/or strong external field regime that we are considering, a large proportion of the mass is actually taken up by the single state σ^{+}. More precisely, suppose that for some $\Delta \in (0,1)$,

$$2\beta(h+4) \geq \log(N) - \log(\log(1/\Delta)).$$

Then, using the elementary inequality $\log(1+x) \leq x$, we have

$$\log(1 + e^{-2\beta(h+4)}) \leq e^{-2\beta(h+4)} \leq \frac{1}{d}\log(1/\Delta) \iff (1 + e^{-2\beta(h+4)})^{-d} \geq \Delta.$$

Therefore, using (6.66), we can compute

$$\begin{aligned}\mathbf{v}^{\downarrow}(1) = \mu_{\beta,h}(\sigma^+) &= \frac{e^{-\beta H(\sigma^+)}}{e^{-\beta H(\sigma^+)} + \sum_{\sigma \neq \sigma^+} e^{-\beta H(\sigma)}} \\ &= \frac{1}{1 + \sum_{k=1}^{d} \sum_{\sigma \in \mathcal{C}_k} e^{-\beta(H(\sigma) - H(\sigma^+))}} \\ &\geq \frac{1}{1 + \sum_{k=1}^{d} \binom{d}{k} e^{-2\beta(h+4)k}} = \frac{1}{(1 + e^{-2\beta(h+4)})^d} \geq \Delta.\end{aligned}$$

That is, $\sum_{i=2}^{n} \mathbf{v}^{\downarrow}(i) \leq 1 - \Delta$. □

6.9 Proofs for properties of ℓ^1 contraction coefficients

In this section, we verify the properties of $\alpha_r(\mathbf{A})$ stated in Proposition 6.4.3.

Proof of Proposition 6.4.3. (Part 1). Note that the ℓ^1 contraction coefficient $\alpha_r(\mathbf{A})$ is the solution of the maximization problem $\max_{\mathbf{z} \in \mathbb{R}^n} \|\mathbf{A}^r \mathbf{z}\|_1$, subject to $\|\mathbf{z}\|_1 \leq 1$ and $\sum_{i=1}^{n} \mathbf{z}(i) = 0$. This is a convex maximization problem in a convex and compact domain, so the solution is attained at an extreme point, which takes the form $\mathbf{z} = \pm\frac{1}{2}(\mathbf{e}_i - \mathbf{e}_j)$ with $i \neq j$. The expression $\alpha_r(\mathbf{A}) = \frac{1}{2} \max_{i,j} \|\mathbf{A}^r(\mathbf{e}_i - \mathbf{e}_j)\|_1$ follows. By manipulating this explicit expression and using the fact that columns of $\mathbf{A}^r$ are stochastic vectors, we can verify that

$$\begin{aligned}\alpha_r(\mathbf{A}) &= \frac{1}{2} \max_{i,j} \sum_{k=1}^{n} |\mathbf{A}^r(k,i) - \mathbf{A}^r(k,j)| \\ &= \frac{1}{2} \max_{i,j} \sum_{k=1}^{n} \big[\mathbf{A}^r(k,i) + \mathbf{A}^r(k,j) - 2\min\{\mathbf{A}^r(k,i), \mathbf{A}^r(k,j)\}\big] \\ &= 1 - \min_{i,j} \sum_{k=1}^{n} \min\{\mathbf{A}^r(k,i), \mathbf{A}^r(k,j)\}.\end{aligned}$$

(Part 2). Fix any $\mathbf{z} \in \mathbb{R}^n$ with $\sum_{i=1}^{n} \mathbf{z}(i) = 0$. Observe that $\mathbf{1}^{\mathsf{T}} \mathbf{A}^s \mathbf{z} = \mathbf{1}^{\mathsf{T}} \mathbf{z} = 0$ for each $s \leq r$. Consequently, using the definition of the ℓ^1 contraction coefficients,

$$\|\mathbf{A}^r \mathbf{z}\|_1 = \|\mathbf{A}^{r-s} \mathbf{A}^s \mathbf{z}\|_1 \leq \alpha_{r-s}(\mathbf{A}) \|\mathbf{A}^s \mathbf{z}\|_1 \leq \alpha_{r-s}(\mathbf{A}) \alpha_s(\mathbf{A}) \|\mathbf{z}\|_1.$$

By optimizing over $\mathbf{z}$, we confirm the submultiplicativity property.

(Part 3). We showed that $\|\mathbf{A}^r\mathbf{z}\|_1 = \|\mathbf{A}^r\mathbf{P}\mathbf{z}\|_1 \leq \|\mathbf{A}^r\mathbf{P}\|_1\|\mathbf{z}\|_1$ for any $\mathbf{z} \in \mathbb{R}^n$ with $\sum_{i=1}^n \mathbf{z}(i) = 0$, which implies that $\alpha_r(\mathbf{A}) \leq \|\mathbf{A}^r\mathbf{P}\|_1$. To obtain a matching lower bound, we explicitly calculate, using $\mathbf{P} = \mathbf{I} - \mathbf{v}\mathbf{1}^\mathsf{T}$,

$$\|\mathbf{P}\|_1 = \max_i \|\mathbf{P}(:,i)\|_1 = \max_i \left[1 - \mathbf{v}(i) + \sum_{j \neq i} \mathbf{v}(j)\right] = 2 - 2\min_i \mathbf{v}(i) \leq 2.$$

Therefore, since $\mathbf{1}^\mathsf{T}\mathbf{P}\mathbf{z} = \mathbf{1}^\mathsf{T}\mathbf{z} - (\mathbf{1}^\mathsf{T}\mathbf{v})(\mathbf{1}^\mathsf{T}\mathbf{z}) = 0$ for any $\mathbf{z} \in \mathbb{R}^n$ with $\mathbf{1}^\mathsf{T}\mathbf{z} = 0$, we have

$$\|\mathbf{A}^r(\mathbf{P}\mathbf{z})\|_1 \leq \alpha_r(\mathbf{A})\|\mathbf{P}\mathbf{z}\|_1 \leq \alpha_r(\mathbf{A})\|\mathbf{P}\|_1\|\mathbf{z}\|_1 \leq 2\alpha_r(\mathbf{A})\|\mathbf{z}\|_1.$$

By optimizing over $\mathbf{z}$, we confirm that $\|\mathbf{A}^r\mathbf{P}\|_1 \leq 2\alpha_r(\mathbf{A})$, as desired.

(Part 4). Let $\mathbf{v}_2$ be the eigenvector corresponding to $\lambda_2(\mathbf{A})$, so that $\mathbf{A}\mathbf{v}_2 = \lambda_2(\mathbf{A})\mathbf{v}_2$. Since $\mathbf{A}$ is column-stochastic, multiplying both sides on the left by $\mathbf{1}^\mathsf{T}$ implies that $\mathbf{1}^\mathsf{T}\mathbf{v}_2 = \mathbf{1}^\mathsf{T}\mathbf{A}\mathbf{v}_2 = \lambda_2(\mathbf{A})\mathbf{1}^\mathsf{T}\mathbf{v}_2$. Since $\lambda_2(\mathbf{A}) \neq 1$ by assumption, we must have $\mathbf{1}^\mathsf{T}\mathbf{v}_2 = 0$. It follows that

$$|\lambda_2(\mathbf{A})|^r = \frac{\|\mathbf{A}^r\mathbf{v}_2\|_1}{\|\mathbf{v}_2\|_1} \leq \alpha_r(\mathbf{A}).$$

Recall that $\mathbf{P}$ and $\mathbf{A}$ commute. By invoking Gelfand's formula for matrix norms, we have

$$\lim_{r\to\infty} \|\mathbf{A}^r\mathbf{P}\|_1^{1/r} = \lim_{r\to\infty} \|(\mathbf{A}\mathbf{P})^r\|_1^{1/r} = \rho(\mathbf{A}\mathbf{P}),$$

where $\rho(\mathbf{A}\mathbf{P})$ is the spectral radius of $\mathbf{A}\mathbf{P}$, which equals $|\lambda_2(\mathbf{A})|$. In combination with the oblique projection bounds in part 3, this confirms that $\lim_{r\to\infty} \alpha_r(\mathbf{A})^{1/r} = |\lambda_2(\mathbf{A})|$. □

Bibliography

[ABH05] E. Amaldi, P. Belotti, and R. Hauser. "Randomized Relaxation Methods for the Maximum Feasible Subsystem Problem". In: *Integer Programming and Combinatorial Optimization*. Berlin, Heidelberg: Springer, 2005, pp. 249–264. DOI: 10.1007/11496915_19 (p. 39).

[AC06] N. Ailon and B. Chazelle. "Approximate nearest neighbors and the fast Johnson-Lindenstrauss transform". *ACM Symposium on Theory of Computing (STOC)*. 2006, pp. 557–563 (p. 111).

[ACW17] H. Avron, K. L. Clarkson, and D. P. Woodruff. "Faster Kernel Ridge Regression Using Sketching and Preconditioning". *SIAM Journal on Matrix Analysis and Applications* **38** (4), 2017, pp. 1116–1138. arXiv: 1611.03220 [math.NA]. DOI: 10.1137/16M1105396 (p. 96).

[ADG15] H. Avron, A. Druinsky, and A. Gupta. "Revisiting Asynchronous Linear Solvers: Provable Convergence Rate through Randomization". *Journal of the ACM* **62** (6), 2015, pp. 1–27. arXiv: 1304.6475 [cs.DC]. DOI: 10.1145/2814566 (p. 126).

[ADT20] A. Ali, E. Dobriban, and R. Tibshirani. "The Implicit Regularization of Stochastic Gradient Flow for Least Squares". *International Conference on Machine Learning*. 2020. arXiv: 2003.07802 [stat.ML] (pp. 157, 206).

[AK84] A. H. Andersen and A. C. Kak. "Simultaneous Algebraic Reconstruction Technique (SART): A superior implementation of the ART algorithm". *Ultrasonic Imaging* **6** (1), 1984, pp. 81–94. DOI: 10.1016/0161-7346(84)90008-7 (p. 16).

[AK95] E. Amaldi and V. Kann. "The complexity and approximability of finding maximum feasible subsystems of linear relations". *Theoretical Computer Science* **147** (1-2), 1995, pp. 181–210. DOI: 10.1016/0304-3975(94)00254-G (p. 39).

[AKT19] A. Ali, J. Z. Kolter, and R. J. Tibshirani. "A Continuous-Time View of Early Stopping for Least Squares Regression". *International Conference on Artificial Intelligence and Statistics*. 2019, pp. 1370–1378. arXiv: 1810.10082 [stat.ML] (pp. 157, 167, 177, 204–206, 219, 229).

[AM07] D. Achlioptas and F. McSherry. "Fast Computation of Low-Rank Matrix Approximations". *Journal of the ACM* **54** (2), 2007. DOI: 10.1145/1219092.1219097 (pp. 14, 111).

[Ari+24] O. Arizmendi, G. Cébron, R. Speicher, and S. Yin. "Universality of free random variables: Atoms for non-commutative rational functions". *Advances in Mathematics* **443**, 2024, p. 109595. arXiv: 2107.11507 [math.OA]. DOI: 10.1016/j.aim.2024.109595 (p. 195).

[Arn+23] L. Arnaboldi, L. Stephan, F. Krzakala, and B. Loureiro. "From high-dimensional & mean-field dynamics to dimensionless ODEs: A unifying approach to SGD in two-layers networks". *Conference on Learning Theory*. 2023. arXiv: 2302.05882 [stat.ML] (p. 160).

[ASS20] M. S. Advani, A. M. Saxe, and H. Sompolinsky. "High-dimensional dynamics of generalization error in neural networks". *Neural Networks* **132**, 2020, pp. 428–446. arXiv: 1710.03667 [stat.ML]. DOI: 10.1016/j.neunet.2020.08.022 (pp. 157, 164, 167, 204–206, 229, 238).

[AWL14] A. Agaskar, C. Wang, and Y. M. Lu. "Randomized Kaczmarz algorithms: Exact MSE analysis and optimal sampling probabilities". *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. 2014, pp. 389–393. DOI: 10.1109/GlobalSIP.2014.7032145 (p. 31).

[Ba+22] J. Ba, M. A. Erdogdu, T. Suzuki, Z. Wang, D. Wu, and G. Yang. "High-dimensional Asymptotics of Feature Learning: How One Gradient Step Improves the Representation". *Advances in Neural Information Processing Systems*. 2022. arXiv: 2205.01445 [stat.ML] (p. 174).

[Bar+20] P. Bartlett, P. M. Long, G. Lugosi, and A. Tsigler. "Benign Overfitting in Linear Regression". *Proceedings of the National Academy of Sciences* **117** (48), 2020, pp. 30063–30070. arXiv: 1906.11300 [stat.ML]. DOI: 10.1073/pnas.1907378117 (p. 224).

[BCN18] L. Bottou, F. E. Curtis, and J. Nocedal. "Optimization Methods for Large-Scale Machine Learning". *SIAM Review* **60** (2), 2018, pp. 223–311. arXiv: 1606.04838 [stat.ML]. DOI: 10.1137/16M1080173 (p. 14).

[BD21] D. G. T. Barrett and B. Dherin. "Implicit Gradient Regularization". *International Conference on Learning Representations*. 2021. arXiv: 2009.11162 [cs.LG] (p. 158).

[Ben23] P. Beneventano. *On the Trajectories of SGD Without Replacement*. Preprint, arXiv:2312.16143. 2023. arXiv: 2312.16143 [cs.LG] (p. 158).

[BF25] E. Boursier and N. Flammarion. "Early Alignment in Two-Layer Networks Training is a Two-Edged Sword". *Journal of Machine Learning Research* **26**, 2025, pp. 1–75. arXiv: 2401.10791 [cs.LG] (p. 206).

[BGJ22] G. Ben Arous, R. Gheissari, and A. Jagannath. "High-dimensional limit theorems for SGD: Effective dynamics and critical scaling". *Advances in Neural Information Processing Systems*. Vol. 35. 2022. arXiv: 2206.04030 [stat.ML] (p. 160).

[BHR18a] G. Blanchard, M. Hoffmann, and M. Reiß. "Early stopping for statistical inverse problems via truncated SVD estimation". *Electronic Journal of Statistics* **12** (2), 2018, pp. 3204–3231. arXiv: 1710.07278 [math.ST]. DOI: 10.1214/18-EJS1482 (p. 207).

[BHR18b] G. Blanchard, M. Hoffmann, and M. Reiß. "Optimal Adaptation for Early Stopping in Statistical Inverse Problems". *SIAM/ASA Journal on Uncertainty Quantification* **6** (3), 2018, pp. 1043–1075. arXiv: 1606.07702 [math.ST]. DOI: 10.1137/17M1154096 (p. 207).

[BM11] F. Bach and E. Moulines. "Non-Asymptotic Analysis of Stochastic Approximation Algorithms for Machine Learning". *Advances in Neural Information Processing Systems*. 2011 (p. 157).

[BM13] F. Bach and E. Moulines. "Non-strongly-convex smooth stochastic approximation with convergence rate $O(1/n)$". *Advances in Neural Information Processing Systems*. 2013. arXiv: 1306.2119 [cs.LG] (p. 157).

[BMD09] C. Boutsidis, M. W. Mahoney, and P. Drineas. "An Improved Approximation Algorithm for the Column Subset Selection Problem". *ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 2009, pp. 968–977. arXiv: 0812.4293 [cs.DS]. DOI: 10.1137/1.9781611973068.105 (pp. 53, 54).

[BMS17] S. T. Belinschi, T. Mai, and R. Speicher. "Analytic subordination theory of operator-valued free additive convolution and the solution of a general random matrix problem". *Journal für die reine und angewandte Mathematik* **2017** (732), 2017, pp. 21–53. arXiv: 1303.3196 [math.OA]. DOI: 10.1515/crelle-2014-0138 (pp. 175, 195–199).

[BN15] J. Briskman and D. Needell. "Block Kaczmarz Method with Inequalities". *Journal of Mathematical Imaging and Vision* **52**, 2015, pp. 385–396. arXiv: 1406.7339 [math.NA]. DOI: 10.1007/s10851-014-0539-7 (pp. 38, 79).

[Bot09] L. Bottou. "Curiously fast convergence of some stochastic gradient descent algorithms". Unpublished open problem offered to the attendance of the SLDS 2009 conference. 2009 (p. 155).

[Bot12] L. Bottou. "Stochastic Gradient Descent Tricks". In: *Neural Networks: Tricks of the Trade*. 2nd ed. Springer Berlin Heidelberg, 2012, pp. 421–436. DOI: 10.1007/978-3-642-35289-8_25 (pp. 154, 155).

[BS10] Z. Bai and J. W. Silverstein. *Spectral Analysis of Large Dimensional Random Matrices*. 2nd ed. Springer New York, NY, 2010. DOI: 10.1007/978-1-4419-0661-8 (p. 174).

[Cam+25] C. Camaño, E. N. Epperly, R. A. Meyer, and J. A. Tropp. "Faster Linear Algebra Algorithms with Structured Random Matrices", 2025. arXiv: 2508.21189 [cs.DS] (p. 111).

[Can+06] E. J. Candes, M. Rudelson, T. Tao, and R. Vershynin. "Error Correction via Linear Programming". *IEEE Symposium on Foundations of Computer Science (FOCS)*. 2006, pp. 668–681. DOI: 10.1109/SFCS.2005.5464411 (p. 39).

[Cen+92] C. Cenker, H. G. Feichtinger, M. Mayer, H. Steier, and T. Strohmer. "New variants of the POCS method using affine subspaces of finite codimension with applications to irregular sampling". *Visual Communications and Image Processing '92*. Vol. 1818. 1992, pp. 299–310 (p. 29).

[CFL09] C. Castellano, S. Fortunato, and V. Loreto. "Statistical physics of social dynamics". *Reviews of Modern Physics* **81** (2), 2009, pp. 591–646. arXiv: 0710.3256 [physics.soc-ph]. DOI: 10.1103/RevModPhys.81.591 (p. 254).

[Che+23] L. Cheng, B. Jarman, D. Needell, and E. Rebrova. "On block accelerations of quantile randomized Kaczmarz for corrupted systems of linear equations". *Inverse Problems* **39** (2), 2023, p. 024002. arXiv: 2206.12554 [math.NA]. DOI: 10.1088/1361-6420/aca78a (pp. 39, 67).

[Che+25] Y. Chen, E. N. Epperly, J. A. Tropp, and R. J. Webber. "Randomly pivoted Cholesky: Practical approximation of a kernel matrix with few entry evaluations". *Communications on Pure and Applied Mathematics* **78** (5), 2025, pp. 995–1041. arXiv: 2207.06503 [math.NA]. DOI: 10.1002/cpa.22234 (pp. 20, 81, 82, 87, 89, 96, 113, 119, 130, 131, 133).

[Chi19] Chizat, Lénaïc and Oyallon, Edouard and Bach, Francis. "On Lazy Training in Differentiable Programming". *Advances in Neural Information Processing Systems*. Vol. 32. 2019. arXiv: 1812.07956 [math.OC] (pp. 159, 207).

[CK26] A. Cortinovis and D. Kressner. "Adaptive Randomized Pivoting for Column Subset Selection, DEIM, and Low-Rank Approximation". *SIAM Journal on Matrix Analysis and Applications* **47** (1), 2026, pp. 25–47. arXiv: 2412.13992 [math.NA]. DOI: 10.1137/24M1719189 (p. 131).

[CKS24] M. D. Cattaneo, J. M. Klusowski, and B. Shigida. "On the Implicit Bias of Adam". *International Conference on Machine Learning*. 2024. arXiv: 2309.00079 [cs.LG] (p. 158).

[CL11] C.-C. Chang and C.-J. Lin. "LIBSVM: A library for support vector machines". *ACM Transactions on Intelligent Systems and Technology* **2** (3), 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm, pp. 1–27. DOI: 10.1145/1961189.1961199 (pp. 126, 133).

[CL22] R. Couillet and Z. Liao. *Random Matrix Methods for Machine Learning*. Cambridge University Press, 2022. DOI: 10.1017/9781009128490 (pp. 174, 199).

[CM24] C. Cheng and A. Montanari. "Dimension free ridge regression". *The Annals of Statistics* **52** (6), 2024, pp. 2879–2912. arXiv: 2210.08571 [math.ST]. DOI: 10.1214/24-AOS2449 (p. 227).

[CQ21] X. Chen and J. Qin. "Regularized Kaczmarz algorithms for tensor recovery". *SIAM Journal on Imaging Sciences* **14** (4), 2021, pp. 1439–1471. arXiv: 2102.06852 [math.OC]. DOI: 10.1137/21M1398562 (p. 29).

[CW01] A. Carbery and J. Wright. "Distributional and L^q norm inequalities for polynomials over convex bodies in $\mathbb{R}^n$". *Mathematical Research Letters* **8**, 2001, pp. 233–248. DOI: 10.4310/MRL.2001.v8.n3.a1 (p. 69).

[CZ23] H. Cui and L. Zdeborová. "High-dimensional Asymptotics of Denoising Autoencoders". *37th Conference on Neural Information Processing Systems*. 2023. arXiv: 2305.11041 [cs.LG] (p. 207).

[Der+20] M. Dereziński, F. Liang, Z. Liao, and M. W. Mahoney. "Precise expressions for random projections: Low-rank approximation and randomized Newton". *Advances in Neural Information Processing Systems*. Vol. 33. 2020, pp. 18272–18283. arXiv: 2006.10653 [cs.LG] (pp. 92, 98, 106).

[Der+25a] M. Dereziński, D. LeJeune, D. Needell, and E. Rebrova. "Fine-grained Analysis and Faster Algorithms for Iteratively Solving Linear Systems". *Journal of Machine Learning Research* **26** (144), 2025, pp. 1–49. URL: %7Bhttp://jmlr.org/papers/v26/24-1906.html%7D. arXiv: 2405.05818 [cs.DS] (pp. 38, 83, 89, 92, 94, 95, 119, 121).

[Der+25b] M. Dereziński, D. Needell, E. Rebrova, and J. Yang. "Randomized Kaczmarz Methods with Beyond-Krylov Convergence". *SIAM Journal on Matrix Analysis and Applications*, 2025. To appear. arXiv: 2501.11673 [math.NA] (pp. 83, 89, 94, 122, 123, 128, 137, 152).

[DGJ09] M. Dyer, L. A. Goldberg, and M. Jerrum. "Matrix norms and rapid mixing for spin systems". *The Annals of Applied Probability* **19** (1), 2009, pp. 71–107. arXiv: math/0702744. DOI: 10.1214/08-AAP532 (p. 304).

[DHN17] J. A. De Loera, J. Haddock, and D. Needell. "A Sampling Kaczmarz–Motzkin Algorithm for Linear Feasibility". *SIAM Journal on Scientific Computing* **39** (5), 2017, S66–S87. arXiv: 1605.01418 [math.OC]. DOI: 10.1137/16M1073807 (p. 38).

[Día+24] M. Díaz, E. N. Epperly, Z. Frangella, J. A. Tropp, and R. J. Webber. "Robust, randomized preconditioning for kernel ridge regression", 2024. arXiv: 2304.12465 [math.NA] (pp. 82, 95, 96, 126, 127, 133).

[DKM06a] P. Drineas, R. Kannan, and M. W. Mahoney. "Fast Monte Carlo Algorithms for Matrices I: Approximating Matrix Multiplication". *SIAM Journal on Computing* **36** (1), 2006, pp. 132–157. DOI: 10.1137/S0097539704442684 (p. 14).

[DKM06b] P. Drineas, R. Kannan, and M. W. Mahoney. "Fast Monte Carlo Algorithms for Matrices II: Computing a Low-Rank Approximation to a Matrix". *SIAM Journal on Computing* **36** (1), 2006, pp. 158–183. DOI: 10.1137/S0097539704442696 (pp. 53, 54).

[DL21] O. Dhifallah and Y. Lu. "On the Inherent Regularization Effects of Noise Injection During Training". *Proceedings of the 38th International Conference on Machine Learning*. Vol. 139. 2021. arXiv: 2102.07379 [cs.LG] (p. 207).

[DM16] P. Drineas and M. W. Mahoney. "RandNLA: Randomized Numerical Linear Algebra". *Communications of the ACM* **59** (6), 2016, pp. 80–90. DOI: 10.1145/2842602 (p. 14).

[DM21] M. Dereziński and M. W. Mahoney. "Determinantal Point Processes in Randomized Numerical Linear Algebra". *Notices of the American Mathematical Society* **68** (1), 2021, pp. 34–45. arXiv: 2005.03185 [cs.DS]. DOI: 10.1090/noti2202 (p. 96).

[DM24] M. Dereziński and M. W. Mahoney. "Recent and Upcoming Developments in Randomized Numerical Linear Algebra for Machine Learning". *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. Vol. 32. Full technical report available at arXiv:2406.11151. 2024, pp. 6470–6479. arXiv: 2406.11151 [cs.LG]. DOI: 10.1145/3637528.3671461 (p. 14).

[DMM08] P. Drineas, M. W. Mahoney, and S. Muthukrishnan. "Relative-Error *CUR* Matrix Decompositions". *SIAM Journal on Matrix Analysis and Applications* **30** (2), 2008, pp. 844–881. arXiv: 0708.3696 [cs.DS]. DOI: 10.1137/07070471X (pp. 53, 54).

[Don+25] Y. Dong, C. Chen, P.-G. Martinsson, and K. Pearce. "Robust Blockwise Random Pivoting: Fast and Accurate Adaptive Interpolative Decomposition". *SIAM Journal on Matrix Analysis and Applications* **46** (3), 2025, pp. 1791–1815. arXiv: 2309.16002 [math.NA]. DOI: 10.1137/24M1678027 (p. 96).

[DR24] M. Dereziński and E. Rebrova. "Sharp Analysis of Sketch-and-Project Methods via a Connection to Randomized Singular Value Decomposition". *SIAM Journal on Mathematics of Data Science* **6** (1), 2024, pp. 127–153. arXiv: 2208.09585 [math.OC]. DOI: 10.1137/23M1545537 (pp. 55, 89, 92, 94, 98, 123).

[Dri+12] P. Drineas, M. Magdon-Ismail, M. W. Mahoney, and D. P. Woodruff. "Fast Approximation of Matrix Coherence and Statistical Leverage". *The Journal of Machine Learning Research* **13** (1), 2012, pp. 3475–3506. arXiv: 1109.3843 [cs.DS]. DOI: 10.5555/2503308.2503352 (p. 54).

[DT19] A. Dalalyan and P. Thompson. "Outlier-robust estimation of a sparse linear model using ℓ_1-penalized Huber's M-estimator". *Advances in Neural Information Processing Systems*. Vol. 32. 2019. arXiv: 1904.06288 [math.ST] (p. 39).

[DT98] J.-C. Deville and Y. Tille. "Unequal Probability Sampling Without Replacement Through a Splitting Method". *Biometrika* **85** (1), 1998, pp. 89–101. DOI: 10.1093/biomet/85.1.89 (p. 264).

[Du+19a] S. Du, J. Lee, H. Li, L. Wang, and X. Zhai. "Gradient Descent Finds Global Minima of Deep Neural Networks". *International Conference on Machine Learning*. Vol. 97. 2019, pp. 1675–1685. arXiv: 1811.03804 [cs.LG] (p. 159).

[Du+19b] S. S. Du, X. Zhai, B. Poczos, and A. Singh. "Gradient Descent Provably Optimizes Over-parameterized Neural Networks". *International Conference on Learning Representations*. 2019. arXiv: 1810.02054 [cs.LG] (p. 159).

[Du+21] Y.-S. Du, K. Hayami, N. Zheng, K. Morikuni, and J.-F. Yin. "Kaczmarz-type inner-iteration preconditioned flexible GMRES methods for consistent linear systems". *SIAM Journal on Scientific Computing* **43** (5), 2021, S345–S366. arXiv: 2006.10818 [math.NA]. DOI: 10.1137/20M1344937 (pp. 29, 38).

[Dum22] H. Duminil-Copin. "100 years of the (critical) Ising model on the hypercubic lattice". *Proceedings of the International Congress of Mathematicians (ICM) 2022*. Vol. 1. EMS Press, 2022, pp. 164–210. arXiv: 2208.00864 [math.PR]. DOI: 10.4171/ICM2022/204 (p. 303).

[DW18] E. Dobriban and S. Wager. "High-dimensional asymptotics of prediction: Ridge regression and classification". *The Annals of Statistics* **46** (1), 2018, pp. 247–279. arXiv: 1507.03003 [math.ST]. DOI: 10.1214/17-AOS1549 (pp. 159, 174, 206, 227, 229).

[DY24] M. Dereziński and J. Yang. "Solving Dense Linear Systems Faster than via Preconditioning". *ACM Symposium on Theory of Computing (STOC)*. 2024. arXiv: 2312.08893 [cs.DS] (pp. 29, 38, 89, 94, 121).

[EGW26] E. N. Epperly, G. Goldshlager, and R. J. Webber. "Randomized Kaczmarz with tail averaging". *Applied and Computational Harmonic Analysis* **80**, 2026, p. 101812. arXiv: 2411.19877 [math.NA]. DOI: 10.1016/j.acha.2025.101812 (p. 94).

[EHN96] H. W. Engl, M. Hanke, and A. Neubauer. *Regularization of Inverse Problems*. Springer Dordrecht, 1996 (p. 207).

[Elf80] T. Elfving. "Block-Iterative Methods for Consistent and Inconsistent Linear Equations". *Numerische Mathematik* **35**, 1980, pp. 1–12. DOI: 10.1007/BF01396365 (pp. 31, 93).

[Epp25] E. N. Epperly. "Make the Most of What You Have: Resource-Efficient Randomized Algorithms for Matrix Computations". PhD thesis. California Institute of Technology, 2025 (p. 14).

[ETW24] E. N. Epperly, J. A. Tropp, and R. J. Webber. "Embrace rejection: Kernel matrix approximation by accelerated randomly pivoted Cholesky", 2024. arXiv: 2410.03969 [math.NA] (pp. 81, 87, 89, 96, 119).

[FI14] A. Fischer and C. Igel. "Training restricted Boltzmann machines: An introduction". *Pattern Recognition* **47** (1), 2014, pp. 25–39. DOI: 10.1016/j.patcog.2013.05.025 (p. 303).

[FKV04] A. Frieze, R. Kannan, and S. Vempala. "Fast Monte-Carlo Algorithms for Finding Low-Rank Approximations". *Journal of the ACM* **51** (6), 2004, pp. 1025–1041. DOI: 10.1145/1039488.1039494 (p. 14).

[FL24] M. Fornace and M. Lindsey. "Column and row subset selection using nuclear scores: algorithms and theory for Nyström approximation, CUR decomposition, and graph Laplacian reduction", 2024. arXiv: 2407.01698 [math.NA] (p. 96).

[FL50] G. E. Forsythe and R. A. Leibler. "Matrix Inversion by a Monte Carlo Method". *Mathematical Tables and Other Aids to Computation* **4** (31), 1950, pp. 127–129. DOI: 10.2307/2002508 (p. 13).

[GBC16] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016 (pp. 204, 210).

[GBH70] R. Gordon, R. Bender, and G. T. Herman. "Algebraic Reconstruction Techniques (ART) for three-dimensional electron microscopy and X-ray photography". *Journal of Theoretical Biology* **29** (3), 1970, pp. 471–476. DOI: 10.1016/0022-5193(70)90109-8 (p. 16).

[Gei+20] M. Geiger, S. Spigler, A. Jacot, and M. Wyart. "Disentangling feature and lazy training in deep neural networks". *Journal of Statistical Mechanics: Theory and Experiment* **11**, 2020, p. 113301. arXiv: 1906.08034 [cs.LG]. DOI: 10.1088/1742-5468/abc4de (p. 207).

[Gei+22] J. Geiping, M. Goldblum, P. Pope, M. Moeller, and T. Goldstein. "Stochastic Training is Not Necessary for Generalization". *International Conference on Learning Representations*. 2022. arXiv: 2109.14119 [cs.LG] (p. 159).

[GG84] S. Geman and D. Geman. "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images". *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-6** (6), 1984, pp. 721–741. DOI: 10.1109/TPAMI.1984.4767596 (p. 303).

[Gla63] R. J. Glauber. "Time-Dependent Statistics of the Ising Model". *Journal of Mathematical Physics* **4**, 1963, pp. 294–307. DOI: 10.1063/1.1703954 (p. 254).

[Gle15] D. F. Gleich. "PageRank Beyond the Web". *SIAM Review* **57** (3), 2015, pp. 321–363. arXiv: 1407.5107 [cs.SI]. DOI: 10.1137/140976649 (pp. 259, 260, 263).

[Gol+19] S. Goldt, M. Advani, A. M. Saxe, F. Krzakala, and L. Zdeborová. "Dynamics of stochastic gradient descent for two-layer neural networks in the teacher-student setup". *Advances in Neural Information Processing Systems*. 2019. arXiv: 1906.08632 [stat.ML] (p. 160).

[GOP21] M. Gürbüzbalaban, A. Ozdaglar, and P. A. Parrilo. "Why random reshuffling beats stochastic gradient descent". *Mathematical Programming* **186**, 2021, pp. 49–84. arXiv: 1510.08560 [math.OC]. DOI: 10.1007/s10107-019-01440-w (pp. 155, 158).

[Gow+18] R. M. Gower, F. Hanzely, P. Richtárik, and S. U. Stich. "Accelerated Stochastic Matrix Inversion: General Theory and Speeding up BFGS Rules for Faster Second-Order Optimization". *Advances in Neural Information Processing Systems*. 2018. arXiv: 1802.04079 [math.OC] (pp. 94, 137, 139–141).

[Gow+19a] R. M. Gower, D. Kovalev, F. Lieder, and P. Richtárik. "RSN: Randomized Subspace Newton". *Advances in Neural Information Processing Systems*. Vol. 32. 2019. arXiv: 1905.10874 [math.OC] (pp. 37, 91).

[Gow+19b] R. M. Gower, N. Loizou, X. Qian, A. Sailanbayev, E. Shulgin, and P. Richtárik. "SGD: General Analysis and Improved Rates". *International Conference on Machine Learning*. Vol. 97. 2019, pp. 5200–5209. arXiv: 1901.09401 [cs.LG] (pp. 157, 208).

[Gow+21] R. M. Gower, D. Molitor, J. Moorman, and D. Needell. "On Adaptive Sketch-and-Project for Solving Linear Systems". *SIAM Journal on Matrix Analysis and Applications* **42** (2), 2021, pp. 954–989. arXiv: 1909.03604 [math.NA]. DOI: 10.1137/19M1285846 (pp. 94, 103, 137).

[Goy+18] P. Goyal et al. *Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour*. Technical report, arXiv:1706.02677. 2018. arXiv: 1706.02677 [cs.CV] (p. 159).

[GR15a] R. M. Gower and P. Richtárik. "Randomized Iterative Methods for Linear Systems". *SIAM Journal on Matrix Analysis and Applications* **36** (4), 2015, pp. 1660–1690. arXiv: 1506.03296 [math.NA]. DOI: 10.1137/15M1025487 (pp. 19, 31, 34, 37, 79, 91, 94, 98, 170).

[GR15b] R. M. Gower and P. Richtárik. "Stochastic Dual Ascent for Solving Linear Systems", 2015. arXiv: 1512.06890 [math.NA] (p. 103).

[Gre+19] S. M. Greene, R. J. Webber, J. Weare, and T. C. Berkelbach. "Beyond Walkers in Stochastic Quantum Chemistry: Reducing Error Using Fast Randomized Iteration". *Journal of Chemical Theory and Computation* **15** (9), 2019, pp. 4834–4850. arXiv: 1905.00995 [physics.chem-ph]. DOI: 10.1021/acs.jctc.9b00422 (pp. 25, 258, 302).

[Gre+20] S. M. Greene, R. J. Webber, J. Weare, and T. C. Berkelbach. "Improved Fast Randomized Iteration Approach to Full Configuration Interaction". *Journal of Chemical Theory and Computation* **16** (9), 2020, pp. 5572–5585. arXiv: 2005.00654 [physics.comp-ph]. DOI: 10.1021/acs.jctc.0c00437 (pp. 25, 258).

[Gre+22a] S. M. Greene, R. J. Webber, T. C. Berkelbach, and J. Weare. "Approximating Matrix Eigenvalues by Subspace Iteration with Repeated Random Sparsification". *SIAM Journal on Scientific Computing* **44** (5), 2022, A3067–A3097. arXiv: 2103.12109 [math.NA]. DOI: 10.1137/21M1422513 (pp. 25, 258, 264, 302).

[Gre+22b] S. M. Greene, R. J. Webber, J. E. T. Smith, J. Weare, and T. C. Berkelbach. "Full Configuration Interaction Excited-State Energies in Large Active Spaces from Subspace Iteration with Repeated Random Sparsification". *Journal of Chemical Theory and Computation* **18** (12), 2022, pp. 7218–7232. arXiv: 2201.12164 [physics.comp-ph]. DOI: 10.1021/acs.jctc.2c00435 (pp. 25, 258, 302).

[Gun+18a] S. Gunasekar, J. D. Lee, D. Soudry, and N. Srebro. "Characterizing Implicit Bias in Terms of Optimization Geometry". *International Conference on Machine Learning*. Vol. 80. 2018, pp. 1832–1841. arXiv: 1802.08246 [stat.ML] (p. 155).

[Gun+18b] S. Gunasekar, J. D. Lee, D. Soudry, and N. Srebro. "Implicit Bias of Gradient Descent on Linear Convolutional Networks". *Advances in Neural Information Processing Systems*. Vol. 31. 2018. arXiv: 1806.00468 [cs.LG] (p. 155).

[GV13] G. H. Golub and C. F. Van Loan. *Matrix Computations*. 4th ed. The John Hopkins University Press, 2013 (pp. 14, 15).

[GZR22] D. Granziol, S. Zohren, and S. Roberts. "Learning Rates as a Function of Batch Size: A Random Matrix Theory Approach to Neural Network Training". *Journal of Machine Learning Research* **23** (173), 2022, pp. 1–65. arXiv: 2006.09092 [stat.ML] (p. 159).

[Had+22] J. Haddock, D. Needell, E. Rebrova, and W. Swartworth. "Quantile-Based Iterative Methods for Corrupted Systems of Linear Equations". *SIAM Journal on Matrix Analysis and Applications* **43** (2), 2022, pp. 605–637. arXiv: 2009.08089 [math.NA]. DOI: 10.1137/21M1429187 (pp. 18, 35, 36, 39, 62, 63, 67, 68, 74–76, 94).

[Has+22] T. Hastie, A. Montanari, S. Rosset, and R. J. Tibshirani. "Surprises in high-dimensional ridgeless least squares interpolation". *The Annals of Statistics* **50** (2), 2022, pp. 949–986. arXiv: 1903.08560 [math.ST]. DOI: 10.1214/21-AOS2133 (pp. 159, 174, 206).

[Her09] G. T. Herman. *Fundamentals of Computerized Tomography*. Advances in Computer Vision and Pattern Recognition. Springer-Verlag London, 2009. DOI: 10.1007/978-1-84628-723-7 (pp. 16, 29).

[HJ12] R. A. Horn and C. R. Johnson. *Matrix Analysis*. 2nd ed. Cambridge: Cambridge University Press, 2012. DOI: 10.1017/CBO9781139020411 (p. 45).

[HJ18] P. Hansen and J. Jørgensen. "AIR Tools II: algebraic iterative reconstruction methods, improved implementation". *Numerical Algorithms* **79**, 2018, pp. 107–137. DOI: 10.1007/s11075-017-0430-x (p. 78).

[HL22] T. Hu and Y. Lei. "Early Stopping for Iterative Regularization with General Loss Functions". *Journal of Machine Learning Research* **23** (339), 2022, pp. 1–36 (p. 204).

[HLT19] F. He, T. Liu, and D. Tao. "Control Batch Size and Learning Rate to Generalize Well: Theoretical and Empirical Evidence". *Advances in Neural Information Processing Systems*. Vol. 32. 2019 (p. 159).

[HM21] J. Haddock and A. Ma. "Greed Works: An Improved Analysis of Sampling Kaczmarz–Motzkin". *SIAM Journal on Mathematics of Data Science* **3** (1), 2021, pp. 342–368. arXiv: 1912.03544 [math.NA]. DOI: 10.1137/19M1307044 (p. 38).

[HMT11] N. Halko, P.-G. Martinsson, and J. A. Tropp. "Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions". *SIAM Review* **53** (2), 2011, pp. 217–288. arXiv: 0909.4061 [math.NA]. DOI: 10.1137/090771806 (pp. 54, 97, 110, 111).

[HN19] J. Haddock and D. Needell. "Randomized Projection Methods for Linear Systems with Arbitrarily Large Sparse Corruptions". *SIAM Journal on Scientific Computing* **41** (5), 2019, S19–S36. arXiv: 1803.08114 [math.NA]. DOI: 10.1137/18M1179213 (pp. 34, 39).

[HNR17] A. Hefny, D. Needell, and A. Ramdas. "Rows versus Columns: Randomized Kaczmarz or Gauss–Seidel for Ridge Regression". *SIAM Journal on Scientific Computing* **39** (5), 2017, S528–S542. arXiv: 1507.05844 [math.NA]. DOI: 10.1137/16M1077891 (p. 94).

[Hou73] G. N. Hounsfield. "Computerized transverse axial scanning (tomography): Part 1. Description of system". *British Journal of Radiology* **46** (552), 1973, pp. 1016–1022. DOI: 10.1259/0007-1285-46-552-1016 (p. 16).

[HR25] L. Hucker and M. Reiß. "Early stopping for conjugate gradients in statistical inverse problems". *Numerische Mathematik* **157**, 2025, pp. 1739–1791. arXiv: 2406.15001 [math.ST]. DOI: 10.1007/s00211-025-01469-4 (p. 207).

[HS19] J. HaoChen and S. Sra. "Random Shuffling Beats SGD after Finite Epochs". *International Conference on Machine Learning*. Vol. 97. 2019, pp. 2624–2633. arXiv: 1806.10077 [math.OC] (pp. 155, 158).

[HZ05] R. A. Horn and F. Zhang. "Basic Properties of the Schur Complement". In: *The Schur Complement and Its Applications*. Vol. 4. Numerical Methods and Algorithms. Springer, 2005, pp. 17–46. DOI: 10.1007/0-387-24273-2_2 (pp. 115, 119).

[IS11] I. C. F. Ipsen and T. M. Selee. "Ergodicity Coefficients Defined by Vector Norms". *SIAM Journal on Matrix Analysis and Applications* **32** (1), 2011, pp. 153–200. DOI: 10.1137/090752948 (p. 266).

[Isi25] E. Ising. "Beitrag zur Theorie des Ferromagnetismus". *Zeitschrift für Physik* **31**, 1925, pp. 253–258. DOI: 10.1007/BF02980577 (p. 303).

[Jac+20a] A. Jacot, B. Şimşek, F. Spadaro, C. Hongler, and F. Gabriel. "Implicit Regularization of Random Feature Models". *Proceedings of the 37th International Conference on Machine Learning*. 2020, pp. 7397–7406. arXiv: 2002.08404 [stat.ML] (p. 206).

[Jac+20b] A. Jacot, B. Şimşek, F. Spadaro, C. Hongler, and F. Gabriel. "Kernel Alignment Risk Estimator: Risk Prediction from Training Data". *Advances in Neural Information Processing Systems*. Vol. 33. 2020. arXiv: 2006.09796 [stat.ML] (p. 206).

[Jas+18] S. Jastrzębski et al. "Three Factors Influencing Minima in SGD". *International Conference on Artificial Neural Networks*. 2018. arXiv: 1711.04623 [cs.LG] (p. 157).

[JGH18] A. Jacot, F. Gabriel, and C. Hongler. "Neural Tangent Kernel: Convergence and Generalization in Neural Networks". *Advances in Neural Information Processing Systems*. Vol. 31. 2018. arXiv: 1806.07572 [cs.LG] (pp. 207, 209).

[Jin+19] C. Jin, P. Netrapalli, R. Ge, S. M. Kakade, and M. I. Jordan. *A Short Note on Concentration Inequalities for Random Vectors with SubGaussian Norm*. Preprint, arXiv:1902.03736. 2019. arXiv: 1902.03736 [math.PR] (p. 59).

[JN21] B. Jarman and D. Needell. "QuantileRK: Solving large-scale linear systems with corrupted, noisy data". *55th Asilomar Conference on Signals, Systems, and Computers*. 2021, pp. 1312–1316. arXiv: 2108.02304 [math.NA] (p. 39).

[JNR25] H. Jeong, D. Needell, and E. Rebrova. "Stochastic Gradient Descent for Streaming Linear and Rectified Linear Systems with Adversarial Corruptions". *SIAM Journal on Mathematics of Data Science* **7** (2), 2025, pp. 516–541. arXiv: 2403.01204 [cs.LG]. DOI: 10.1137/24M1652167 (p. 208).

[Kac37] S. Kaczmarz. "Angenäherte Auflösung von Systemen linearer Gleichungen". *Bulletin International de l'Academie Polonaise des Sciences et des Lettres* **35**, 1937, pp. 355–357 (pp. 15, 29).

[Kel+13] J. A. Kelner, L. Orecchia, A. Sidford, and Z. A. Zhu. "A Simple, Combinatorial Algorithm for Solving SDD Systems in Nearly-Linear Time". *ACM Symposium on Theory of Computing (STOC)*. 2013, pp. 911–920. arXiv: 1301.6628 [cs.DS]. DOI: 10.1145/2488608.2488724 (p. 39).

[KLS20] D. Kobak, J. Lomond, and B. Sanchez. "The Optimal Ridge Penalty for Real-world High-dimensional Data Can Be Zero or Negative due to the Implicit Ridge Regularization". *Journal of Machine Learning Research* **21** (169), 2020, pp. 1–16. arXiv: 1805.10939 [math.ST] (p. 206).

[Kri14] A. Krizhevsky. *One weird trick for parallelizing convolutional neural networks*. Preprint, arXiv:1404.5997. 2014. arXiv: 1404.5997 [cs.NE] (p. 159).

[KS01] A. C. Kak and M. Slaney. *Principles of Computerized Tomographic Imaging*. Classics in Applied Mathematics. Society For Industrial and Applied Mathematics, 2001. DOI: 10.1137/1.9780898719277 (p. 16).

[KSS24] C. Kausik, K. Srivastava, and R. Sonthalia. "Double Descent and Overfitting under Noisy Inputs and Distribution Shift for Linear Denoisers". *Transactions on Machine Learning Research*, 2024. arXiv: 2305.17297 [cs.LG] (pp. 159, 207, 209).

[KT23] A. Kireeva and J. A. Tropp. "Randomized matrix computations: themes and variations", 2023. CMS Lecture Notes 2023-02. July 2023. To appear in CIME Lecture Notes series. arXiv: 2402.17873 [math.NA]. DOI: 10.7907/7yade-5k351 (p. 14).

[Kum+24] T. Kumar, B. Bordelon, S. J. Gershman, and C. Pehlevan. "Grokking as the transition from lazy to rich training dynamics". *International Conference on Learning Representations*. 2024. arXiv: 2310.06110 [stat.ML] (p. 208).

[KV17] R. Kannan and S. Vempala. "Randomized algorithms in numerical linear algebra". *Acta Numerica* **26**, 2017, pp. 95–135. DOI: 10.1017/S0962492917000058 (p. 14).

[LB14] D. P. Landau and K. Binder. *A Guide to Monte Carlo Simulations in Statistical Physics*. 4th ed. Cambridge University Press, 2014. DOI: 10.1017/CBO9781139696463 (p. 303).

[LBB24] N. I. Levi, A. Beck, and Y. Bar-Sinai. "Grokking in Linear Estimators – A Solvable Model that Groks without Understanding". *International Conference on Learning Representations*. 2024. arXiv: 2310.16441 [stat.ML] (pp. 203, 206).

[Lee+22] K. Lee, A. N. Cheng, C. Paquette, and E. Paquette. "Trajectory of Mini-Batch Momentum: Batch Size Saturation and Convergence in High Dimensions". *Advances in Neural Information Processing Systems*. 2022. arXiv: 2206.01029 [math.OC] (p. 159).

[Lew+20] A. Lewkowycz, Y. Bahri, E. Dyer, J. Sohl-Dickstein, and G. Gur-Ari. *The large learning rate phase of deep learning: the catapult mechanism*. Preprint, arXiv:2003.02218. 2020. arXiv: 2003.02218 [stat.ML] (p. 159).

[LH17] I. Loshchilov and F. Hutter. "SGDR: Stochastic Gradient Descent with Warm Restarts". *International Conference on Learning Representations*. 2017. arXiv: 1608.03983 [cs.LG] (p. 218).

[Lig85] T. M. Liggett. *Interacting Particle Systems*. New York: Springer-Verlag, 1985. DOI: 10.1007/978-1-4613-8542-4 (p. 254).

[Liu05] J. Liu. "Eigenvalue and Singular Value Inequalities of Schur Complements". In: *The Schur Complement and Its Applications*. Vol. 4. Numerical Methods and Algorithms. Springer, 2005, pp. 47–82. DOI: 10.1007/0-387-24273-2_3 (p. 115).

[LL10] D. Leventhal and A. S. Lewis. "Randomized Methods for Linear Constraints: Convergence Rates and Conditioning". *Mathematics of Operations Research* **35** (3), 2010, pp. 641–654. arXiv: 0806.3015 [math.OC]. DOI: 10.1287/moor.1100.0456 (pp. 20, 38, 79, 82, 83, 91, 94, 105, 116).

[LMA21] Z. Li, S. Malladi, and S. Arora. "On the Validity of Modeling SGD with Stochastic Differential Equations (SDEs)". *Advances in Neural Information Processing Systems*. 2021. arXiv: 2102.12470 [cs.LG] (p. 157).

[Lor+14] D. A. Lorenz, S. Wenger, F. Schöpfer, and M. A. Magnor. "A sparse Kaczmarz solver and a linearized Bregman method for online compressed sensing". *IEEE International Conference on Image Processing (ICIP)*. 2014, pp. 1347–1351. arXiv: 1403.7543 [math.OC]. DOI: 10.1109/ICIP.2014.7025269 (p. 38).

[LPW17] D. Levin, Y. Peres, and E. Wilmer. *Markov Chains and Mixing Times*. 2nd ed. Providence, Rhode Island: American Mathematical Society, 2017 (pp. 254, 265, 268, 269, 304).

[LR24] J. Lok and E. Rebrova. "A subspace constrained randomized Kaczmarz method for structure or external knowledge exploitation". *Linear Algebra and its Applications* **698**, 2024, pp. 220–260. arXiv: 2309.04889 [math.NA]. DOI: 10.1016/j.laa.2024.06.010 (pp. 27, 28, 94, 108, 109).

[LR26] J. Lok and E. Rebrova. "Subspace-constrained randomized coordinate descent for linear systems with good low-rank matrix approximations". *SIAM Journal on Matrix Analysis and Applications*, 2026. To appear. arXiv: 2506.09394 [math.NA] (pp. 27, 80).

[LRS18] D. A. Lorenz, S. Rose, and F. Schöpfer. "The randomized Kaczmarz method with mismatched adjoint". *BIT Numerical Mathematics* **58**, 2018, pp. 1079–1098. arXiv: 1803.02848 [math.NA]. DOI: 10.1007/s10543-018-0717-x (p. 42).

[LS12] E. Lubetzky and A. Sly. "Critical Ising on the Square Lattice Mixes in Polynomial Time". *Communications in Mathematical Physics* **313**, 2012, pp. 815–836. arXiv: 1001.1613 [math.PR]. DOI: 10.1007/s00220-012-1460-9 (p. 304).

[LS13] Y. T. Lee and A. Sidford. "Efficient Accelerated Coordinate Descent Methods and Faster Algorithms for Solving Linear Systems". *Proceedings of the 2013 IEEE 54th Annual Symposium on Foundations of Computer Science*. 2013, pp. 147–156. arXiv: 1305.1922 [cs.DS]. DOI: 10.1109/FOCS.2013.24 (p. 94).

[LS16] E. Lubetzky and A. Sly. "Information percolation and cutoff for the stochastic Ising model". *Journal of the American Mathematical Society* **29** (3), 2016, pp. 729–774. arXiv: 1401.6065 [math.PR]. DOI: 10.1090/jams/841 (p. 304).

[LSR25] J. Lok, R. Sonthalia, and E. Rebrova. "Error dynamics of mini-batch gradient descent with random reshuffling for least squares regression". *Proceedings of the 36th International Conference on Algorithmic Learning Theory*. 2025. arXiv: 2406.03696 [stat.ML] (pp. 27, 154, 242).

[LTE17] Q. Li, C. Tai, and W. E. "Stochastic Modified Equations and Adaptive Stochastic Gradient Algorithms". *International Conference on Machine Learning*. 2017. arXiv: 1511.06251 [cs.LG] (p. 157).

[LTE19] Q. Li, C. Tai, and W. E. "Stochastic Modified Equations and Dynamics of Stochastic Gradient Algorithms I: Mathematical Foundations". *Journal of Machine Learning Research* **20**, 2019, pp. 1–47. arXiv: 1811.01558 [cs.LG] (p. 157).

[LW16] J. Liu and S. J. Wright. "An Accelerated Randomized Kaczmarz Algorithm". *Mathematics of Computation* **85** (297), 2016, pp. 153–178. arXiv: 1310.2887 [math.NA]. DOI: 10.1090/mcom/2971 (pp. 38, 94).

[LW17] L.-H. Lim and J. Weare. "Fast Randomized Iteration: Diffusion Monte Carlo through the Lens of Numerical Linear Algebra". *SIAM Review* **59** (3), 2017, pp. 547–587. arXiv: 1508.06104 [math.NA]. DOI: 10.1137/15M1040827 (pp. 25, 258, 268, 281, 302).

[LWM19] Y. Li, C. Wei, and T. Ma. "Towards Explaining the Regularization Effect of Initial Large Learning Rate in Training Neural Networks". *Advances in Neural Information Processing Systems*. 2019. arXiv: 1907.04595 [cs.LG] (p. 159).

[Ma13] Z. Ma. "Sparse principal component analysis and iterative thresholding". *The Annals of Statistics* **41** (2), 2013, pp. 772–801. arXiv: 1112.2432 [math.ST]. DOI: 10.1214/13-AOS1097 (p. 257).

[Mah11] M. W. Mahoney. "Randomized Algorithms for Matrices and Data". *Foundations and Trends in Machine Learning* **3** (2), 2011, pp. 123–224. arXiv: 1104.5557 [cs.DS]. DOI: 10.1561/2200000035 (pp. 14, 53).

[Mal+22] S. Malladi, K. Lyu, A. Panigrahi, and S. Arora. "On the SDEs and Scaling Rules for Adaptive Gradient Algorithms". *Advances in Neural Information Processing Systems*. 2022. arXiv: 2205.10287 [cs.LG] (pp. 157, 159).

[MBB18] S. Ma, R. Bassily, and M. Belkin. "The Power of Interpolation: Understanding the Effectiveness of SGD in Modern Over-parametrized Learning". *International Conference on Machine Learning*. Vol. 80. 2018, pp. 3325–3334. arXiv: 1712.06559 [cs.LG] (pp. 157, 159).

[MDK20] M. Mutný, M. Dereziński, and A. Krause. "Convergence Analysis of Block Coordinate Algorithms with Determinantal Sampling". *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*. 2020. arXiv: 1910.11561 [math.NA] (pp. 83, 89, 92).

[Mea+20] G. Meanti, L. Carratino, L. Rosasco, and A. Rudi. "Kernel methods through the roof: handling billions of points efficiently". *Proceedings of the 34th International Conference on Neural Information Processing Systems*. 2020, pp. 14410–14422. arXiv: 2006.10350 [cs.LG]. DOI: 10.5555/3495724.3496932 (p. 96).

[Met+53] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. "Equation of State Calculations by Fast Computing Machines". *The Journal of Chemical Physics* **21** (6), 1953, pp. 1087–1092. DOI: 10.1063/1.1699114 (p. 13).

[Mey73] C. D. Meyer Jr. "Generalized Inversion of Modified Matrices". *SIAM Journal on Applied Mathematics* **24** (3), 1973, pp. 315–323. DOI: 10.1137/0124033 (p. 106).

[MHB17] S. Mandt, M. D. Hoffman, and D. M. Blei. "Stochastic Gradient Descent as Approximate Bayesian Inference". *Journal of Machine Learning Research* **18**, 2017, pp. 1–35. arXiv: 1704.04289 [stat.ML] (p. 157).

[Miy22] T. Miyagawa. "Toward Equation of Motion for Deep Neural Networks: Continuous-time Gradient Descent and Discretization Error Analysis". *Advances in Neural Information Processing Systems*. 2022. arXiv: 2210.15898 [cs.LG] (p. 158).

[MKR20] K. Mishchenko, A. Khaled, and P. Richtárik. "Random Reshuffling: Simple Analysis with Vast Improvements". *Advances in Neural Information Processing Systems*. 2020. arXiv: 2006.05988 [math.OC] (p. 158).

[ML12] J. McAuley and J. Leskovec. "Learning to Discover Social Circles in Ego Networks". *Advances in Neural Information Processing Systems*. Dataset available from https://snap.stanford.edu/data/ego-Twitter.html. 2012 (p. 261).

[MM22] S. Mei and A. Montanari. "The Generalization Error of Random Features Regression: Precise Asymptotics and the Double Descent Curve". *Communications on Pure and Applied Mathematics* **75**, 2022, pp. 667–766. arXiv: 1908.05355 [math.ST]. DOI: 10.1002/cpa.22008 (pp. 159, 174).

[MM23] T. Misiakiewicz and A. Montanari. *Six Lectures on Linearized Neural Networks*. Preprint, arXiv:2308.13431. 2023. arXiv: 2308.13431 [stat.ML] (p. 159).

[MN24] M. Meier and Y. Nakatsukasa. "Fast Randomized Numerical Rank Estimation for Numerically Low-Rank Matrices". *Linear Algebra and its Applications* **686**, 2024, pp. 1–32. arXiv: 2105.07388 [math.NA]. DOI: 10.1016/j.laa.2024.01.001 (p. 55).

[MNR15] A. Ma, D. Needell, and A. Ramdas. "Convergence Properties of the Randomized Extended Gauss–Seidel and Kaczmarz Methods". *SIAM Journal on Matrix Analysis and Applications* **36** (4), 2015, pp. 1590–604. arXiv: 1503.08235 [math.NA]. DOI: 10.1137/15M1014425 (pp. 38, 94).

[Mon+24] B. Moniri, D. Lee, H. Hassani, and E. Dobriban. "A Theory of Non-Linear Feature Learning with One Gradient Step in Two-Layer Neural Networks". *Proceedings of the 41st International Conference on Machine Learning*. 2024. arXiv: 2310.07891 [stat.ML] (p. 242).

[MP67] V. A. Marčenko and L. A. Pastur. "Distribution of eigenvalues for some sets of random matrices". *Mathematics of the USSR-Sbornik* **1** (4), 1967, p. 457. DOI: 10.1070/SM1967v001n04ABEH001994 (p. 174).

[MR25] R. Miftachov and M. Reiß. *Early Stopping for Regression Trees*. Preprint, arXiv:2502.04709. 2025. arXiv: 2502.04709 [math.ST] (p. 207).

[MR95] R. Motwani and P. Raghavan. *Randomized Algorithms*. Cambridge University Press, 1995. DOI: 10.1017/CBO9780511814075 (p. 14).

[MS17] J. A. Mingo and R. Speicher. *Free Probability and Random Matrices*. 2nd ed. Springer New York, NY, 2017. DOI: 10.1007/978-1-4939-6942-5 (pp. 175, 195, 196).

[MT20] P.-G. Martinsson and J. A. Tropp. "Randomized numerical linear algebra: Foundations and algorithms". *Acta Numerica* **29**, 2020, pp. 403–572. arXiv: 2002.01387 [math.NA]. DOI: 10.1017/S0962492920000021 (pp. 14, 96, 109, 111, 113).

[MU49] N. Metropolis and S. Ulam. "The Monte Carlo Method". *Journal of the American Statistical Association* **44** (247), 1949, pp. 335–341. DOI: 10.1080/01621459.1949.10483310 (p. 13).

[Mur+23] R. Murray et al. "Randomized Numerical Linear Algebra: A Perspective on the Field With an Eye to Software". (LAPACK Working Note 299), 2023. arXiv: 2302.11474 [math.NA] (p. 14).

[Nak+21] P. Nakkiran, P. Venkat, S. M. Kakade, and T. Ma. "Optimal Regularization can Mitigate Double Descent". *International Conference on Learning Representations*. 2021. arXiv: 2003.01897 [cs.LG] (pp. 206, 236).

[Nat01] F. Natterer. *The Mathematics of Computerized Tomography*. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics, 2001. DOI: 10.1137/1.9780898719284 (pp. 16, 29).

[Nec19] I. Necoara. "Faster Randomized Block Kaczmarz Algorithms". *SIAM Journal on Matrix Analysis and Applications* **40** (4), 2019, pp. 1425–1452. arXiv: 1902.09946 [math.OC]. DOI: 10.1137/19M1251643 (p. 38).

[Nee10] D. Needell. "Randomized Kaczmarz solver for noisy linear systems". *BIT Numerical Mathematics* **50**, 2010, pp. 395–403. arXiv: 0902.0958 [math.NA]. DOI: 10.1007/s10543-010-0265-5 (pp. 32, 43, 75).

[Nes12] Y. Nesterov. "Efficiency of Coordinate Descent Methods on Huge-Scale Optimization Problems". *SIAM Journal on Optimization* **22** (2), 2012, pp. 341–362. DOI: 10.1137/100802001 (pp. 82, 94).

[Ngu+21] L. M. Nguyen, Q. Tran-Dinh, D. T. Phan, P. H. Nguyen, and M. van Dijk. "A Unified Convergence Analysis for Shuffling-Type Gradient Methods". *Journal of Machine Learning Research* **22** (207), 2021, pp. 1–44. arXiv: 2002.08246 [math.OC] (p. 158).

[Nic87] R. A. Nicolaides. "Deflation of Conjugate Gradients with Applications to Boundary Value Problems". *SIAM Journal on Numerical Analysis* **24** (2), 1987, pp. 355–365. DOI: 10.1137/0724027 (p. 105).

[NJN19] D. Nagaraj, P. Jain, and P. Netrapalli. "SGD without Replacement: Sharper Rates for General Smooth Convex Functions". *International Conference on Machine Learning*. Vol. 97. 2019, pp. 4703–4711. arXiv: 1903.01463 [math.OC] (p. 158).

[NS17] Y. Nesterov and S. U. Stich. "Efficiency of the Accelerated Coordinate Descent Method on Structured Optimization Problems". *SIAM Journal on Optimization* **27** (1), 2017, pp. 110–123. DOI: 10.1137/16M1060182 (p. 94).

[NSW16] D. Needell, N. Srebro, and R. Ward. "Stochastic gradient descent, weighted sampling, and the randomized Kaczmarz algorithm". *Mathematical Programming* **155**, 2016, pp. 549–573. arXiv: 1310.5715 [math.NA]. DOI: 10.1007/s10107-015-0864-7 (pp. 22, 37, 38, 94, 157).

[NT14] D. Needell and J. A. Tropp. "Paved with good intentions: Analysis of a randomized block Kaczmarz method". *Linear Algebra and its Applications* **441**, 2014, pp. 199–221. arXiv: 1208.3805 [math.NA]. DOI: 10.1016/j.laa.2012.12.022 (pp. 31–33, 38, 43, 71, 73, 74, 83, 93, 94, 105).

[NT21] I. Necoara and M. Takáč. "Randomized sketch descent methods for non-separable linearly constrained optimization". *IMA Journal of Numerical Analysis* **41** (2), 2021, pp. 1056–1092. arXiv: 1808.02530 [math.OC]. DOI: 10.1093/imanum/draa018 (p. 39).

[NT24] Y. Nakatsukasa and J. A. Tropp. "Fast and Accurate Randomized Algorithms for Linear Systems and Eigenvalue Problems". *SIAM Journal on Matrix Analysis and Applications* **45** (2), 2024, pp. 1183–1214. arXiv: 2111.00113 [math.NA]. DOI: 10.1137/23M1565413 (p. 95).

[NW13] D. Needell and R. Ward. "Two-Subspace Projection Method for Coherent Overdetermined Systems". *Journal of Fourier Analysis and Applications* **19**, 2013. Associated technical report arXiv:1204.0279, pp. 256–269. arXiv: 1204.0277 [math.NA]. DOI: 10.1007/s00041-012-9248-z (pp. 33, 38, 50, 52, 71, 73, 74).

[NZZ15] D. Needell, R. Zhao, and A. Zouzias. "Randomized Block Kaczmarz Method with Projection for Solving Least Squares". *Linear Algebra and its Applications* **484**, 2015, pp. 322–343. arXiv: 1403.4192 [math.NA]. DOI: 10.1016/j.laa.2015.06.027 (p. 38).

[Ons44] L. Onsager. "Crystal Statistics. I. A Two-Dimensional Model with an Order-Disorder Transition". *Physical Review* **65** (3-4), 1944, p. 117. DOI: 10.1103/PhysRev.65.117 (p. 303).

[OT18] S. Oymak and J. A. Tropp. "Universality laws for randomized dimension reduction, with applications". *Information and Inference: A Journal of the IMA* **7** (3), 2018, pp. 337–446. arXiv: 1511.09433 [math.PR]. DOI: 10.1093/imaiai/iax011 (p. 58).

[Pag+99] L. Page, S. Brin, R. Motwani, and T. Winograd. *The PageRank Citation Ranking: Bringing Order to the Web*. Tech. rep. 1999-66. Stanford InfoLab, 1999. URL: http://ilpubs.stanford.edu:8090/422/ (p. 260).

[Pap+00] C. H. Papadimitriou, P. Raghavan, H. Tamaki, and S. Vempala. "Latent Semantic Indexing: A Probabilistic Analysis". *Journal of Computer and System Sciences* **61** (2), 2000, pp. 217–235. DOI: 10.1006/jcss.2000.1711 (p. 14).

[Paq+21] C. Paquette, K. Lee, F. Pedregosa, and E. Paquette. "SGD in the Large: Average-case Analysis, Asymptotics, and Stepsize Criticality". *Conference on Learning Theory*. 2021. arXiv: 2102.04396 [math.OC] (p. 159).

[Paq+22] C. Paquette, E. Paquette, B. Adlam, and J. Pennington. "Implicit Regularization or Implicit Conditioning? Exact Risk Trajectories of SGD in High Dimensions". *Advances in Neural Information Processing Systems*. Vol. 35. 2022, pp. 35984–35999. arXiv: 2206.07252 [stat.ML] (pp. 159, 206).

[Paq+25] C. Paquette, E. Paquette, B. Adlam, and J. Pennington. "Homogenization of SGD in high-dimensions: exact dynamics and generalization properties". *Mathematical Programming* **214**, 2025, pp. 1–90. arXiv: 2205.07069 [math.ST]. DOI: 10.1007/s10107-024-02171-3 (p. 206).

[PF20] S. Pesme and N. Flammarion. "Online Robust Regression via SGD on the ℓ_1 loss". *Advances in Neural Information Processing Systems*. Vol. 33. 2020, pp. 2540–2552. arXiv: 2007.00399 [cs.LG] (p. 208).

[Pie+20] M. Pierini, J. M. Duarte, N. Tran, and M. Freytsis. *HLS4ML LHC Jet dataset (150 particles)*. Available from OpenML at https://www.openml.org/ (ID: 42468). 2020. arXiv: 1804.06913 [physics.ins-det]. DOI: 10.5281/zenodo.3602260 (p. 124).

[PM25] K. J. Pearce and P.-G. Martinsson. *Randomized Algorithms for Low-Rank Matrix and Tensor Decompositions*. Preprint, arXiv:2512.05286. 2025. arXiv: 2512.05286 [math.NA] (p. 14).

[PS05] S. Puntanen and G. P. H. Styan. "Schur Complements in Statistics and Probability". In: *The Schur Complement and Its Applications*. Vol. 4. Numerical Methods and Algorithms. Springer, 2005, pp. 163–226. DOI: 10.1007/0-387-24273-2_7 (p. 115).

[QR16] Z. Qu and P. Richtárik. "Coordinate descent with arbitrary sampling I: algorithms and complexity". *Optimization Methods and Software* **31** (5), 2016, pp. 829–857. arXiv: 1412.8060 [math.OC]. DOI: 10.1080/10556788.2016.1190360 (p. 94).

[Rat+25a] P. Rathore, Z. Frangella, S. Garg, S. Fazliani, M. Dereziński, and M. Udell. "Turbocharging Gaussian Process Inference with Approximate Sketch-and-Project". *Conference on Neural Information Processing Systems (NeurIPS)*. 2025. arXiv: 2505.13723 [cs.LG] (p. 94).

[Rat+25b] P. Rathore, Z. Frangella, J. Yang, M. Dereziński, and M. Udell. "Have ASkotch: A Neat Solution for Large-scale Kernel Ridge Regression", 2025. arXiv: 2407.10070 [cs.LG] (p. 96).

[RC04] C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. 2nd ed. Cambridge University Press, 2004. DOI: 10.1007/978-1-4757-4145-2 (p. 303).

[RCR00] A. Rudi, L. Carratino, and L. Rosasco. "FALKON: An Optimal Large Scale Kernel Method". *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 2000, pp. 3891–3901. arXiv: 1705.10958 [stat.ML]. DOI: 10.5555/3294996.3295145 (p. 96).

[RDR22] D. Richards, E. Dobriban, and P. Rebeschini. *Comparing Classes of Estimators: When does Gradient Descent Beat Ridge Regression in Linear Models?* Preprint, arXiv:2108.11872. 2022. arXiv: 2108.11872 [math.ST] (p. 177).

[RE08] N. Raj Rao and A. Edelman. "The Polynomial Method for Random Matrices". *Foundations of Computational Mathematics* **8**, 2008, pp. 649–702. arXiv: math/0601389. DOI: 10.1007/s10208-007-9013-x (p. 195).

[RGP20] S. Rajput, A. Gupta, and D. Papailiopoulos. "Closing the convergence gap of SGD without replacement". *International Conference on Machine Learning*. Vol. 119. 2020, pp. 7964–7973. arXiv: 2002.10400 [cs.LG] (p. 158).

[RK20] A. Rodomanov and D. Kropotov. "A Randomized Coordinate Descent Method with Volume Sampling". *SIAM Journal on Optimization* **30** (3), 2020, pp. 1878–904. arXiv: 1904.04587 [math.OC]. DOI: 10.1137/19M125532X (pp. 83, 89).

[RN21] E. Rebrova and D. Needell. "On block Gaussian sketching for the Kaczmarz method". *Numerical Algorithms* **86** (1), 2021, pp. 443–473. arXiv: 1905.08894 [math.PR]. DOI: 10.1007/s11075-020-00895-9 (p. 43).

[RR07] A. Rahimi and B. Recht. "Random Features for Large-Scale Kernel Machines". *Proceedings of the 21st International Conference on Neural Information Processing Systems*. 2007, pp. 1177–1184 (pp. 96, 209).

[RT14] P. Richtárik and M. Takáč. "Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function". *Mathematical Programming* **144**, 2014, pp. 1–38. arXiv: 1107.2848 [math.OC]. DOI: 10.1007/s10107-012-0614-z (p. 94).

[RT16] P. Richtárik and M. Takáč. "Parallel coordinate descent methods for big data optimization". *Mathematical Programming* **156**, 2016, pp. 433–484. arXiv: 1212.0873 [math.OC]. DOI: 10.1007/s10107-015-0901-6 (p. 94).

[RT20] P. Richtárik and M. Takáč. "Stochastic Reformulations of Linear Systems: Algorithms and Convergence Theory". *SIAM Journal on Matrix Analysis and Applications* **41** (2), 2020, pp. 487–524. arXiv: 1706.01108 [math.NA]. DOI: 10.1137/18M1179249 (pp. 94, 103, 126, 137).

[RV15] M. Rudelson and R. Vershynin. "Small Ball Probabilities for Linear Images of High-Dimensional Distributions". *International Mathematics Research Notices* **2015** (19), 2015, pp. 9594–9617. arXiv: 1402.4492 [math.PR]. DOI: 10.1093/imrn/rnu243 (p. 69).

[RWY14] G. Raskutti, M. J. Wainwright, and B. Yu. "Early Stopping and Non-parametric Regression: An Optimal Data-dependent Stopping Rule". *Journal of Machine Learning Research* **15** (11), 2014, pp. 335–366. arXiv: 1306.3574 [stat.ML] (pp. 204, 224).

[Saa+00] Y. Saad, M. Yeung, J. Erhel, and F. Guyomarc'h. "A Deflated Version of the Conjugate Gradient Algorithm". *SIAM Journal on Scientific Computing* **21** (5), 2000, pp. 1909–1926. DOI: 10.1137/S1064829598339761 (p. 105).

[Saa03] Y. Saad. *Iterative Methods for Sparse Linear Systems*. 2nd ed. Philadelphia: Society for Industrial and Applied Mathematics, 2003 (p. 95).

[SB00] A. J. Smola and P. Bartlett. "Sparse Greedy Gaussian Process Regression". *Proceedings of the 14th International Conference on Neural Information Processing Systems*. 2000, pp. 598–604 (p. 96).

[Sch+22] F. Schöpfer, D. A. Lorenz, L. Tondji, and M. Winkler. "Extended randomized Kaczmarz method for sparse least squares and impulsive noise problems". *Linear Algebra and its Applications* **652**, 2022, pp. 132–154. arXiv: 2201.08620 [math.NA]. DOI: 10.1016/j.laa.2022.07.003 (pp. 38, 40).

[Sen81] E. Seneta. *Non-negative Matrices and Markov Chains*. 2nd ed. Springer New York, NY, 1981. DOI: 10.1007/0-387-32792-4 (pp. 246, 247, 260, 265).

[SGB94] K. Skouras, C. Goutis, and M. Bramson. "Estimation in linear models using gradient descent with early stopping". *Statistics and Computing* **4**, 1994, pp. 271–278. DOI: 10.1007/BF00156750 (pp. 157, 204, 206).

[SGM22] R. Shen, L. Gao, and Y.-A. Ma. *On Optimal Early Stopping: Over-informative versus Under-informative Parametrization*. Preprint, arXiv:2202.09885. 2022. arXiv: 2202.09885 [cs.LG] (pp. 204–206).

[Sha16] O. Shamir. "Without-Replacement Sampling for Stochastic Gradient Methods". *Advances in Neural Information Processing Systems*. 2016. arXiv: 1603.00570 [cs.LG] (p. 158).

[She+12] J. J. Shepherd, G. Booth, A. Grüneis, and A. Alavi. "Full configuration interaction perspective on the homogeneous electron gas". *Physical Review B* **85** (8), 2012, p. 081103. DOI: 10.1103/PhysRevB.85.081103 (pp. 25, 258).

[SL18] S. L. Smith and Q. V. Le. "A Bayesian Perspective on Generalization and Stochastic Gradient Descent". *International Conference on Learning Representations*. 2018. arXiv: 1710.06451 [cs.LG] (p. 157).

[SL19] F. Schöpfer and D. Lorenz. "Linear convergence of the randomized sparse Kaczmarz method". *Mathematical Programming* **173**, 2019, pp. 509–536. arXiv: 1610.02889 [math.OC]. DOI: 10.1007/s10107-017-1229-1 (pp. 38, 94).

[SL74] L. A. Shepp and B. F. Logan. "The Fourier reconstruction of a head section". *IEEE Transactions on Nuclear Science* **21** (3), 1974, pp. 21–43. DOI: 10.1109/TNS.1974.6499235 (pp. 16, 78).

[SLG23] R. Sonthalia, X. Li, and B. Gu. "Under-Parameterized Double Descent for Ridge Regularized Least Squares Denoising of Data on a Line". *NeurIPS 2023 Workshop on Mathematics of Modern Machine Learning*. 2023. arXiv: 2305.14689 [stat.ML] (p. 207).

[SLR24] R. Sonthalia, J. Lok, and E. Rebrova. "On Regularization via Early Stopping for Least Squares Regression", 2024. Preprint, arXiv:2406.04425. arXiv: 2406.04425 [cs.LG] (pp. 27, 173, 203).

[Smi+18] S. L. Smith, P.-J. Kindermans, C. Ying, and Q. V. Le. "Don't Decay the Learning Rate, Increase the Batch Size". *International Conference on Learning Representations*. 2018. arXiv: 1711.00489 [cs.LG] (p. 159).

[Smi+21] S. L. Smith, B. Dherin, D. G. T. Barrett, and S. De. "On the Origin of Implicit Regularization in Stochastic Gradient Descent". *International Conference on Learning Representations*. 2021. arXiv: 2101.12176 [cs.LG] (p. 158).

[SN23] R. Sonthalia and R. R. Nadakuditi. "Training Data Size Induced Double Descent For Denoising Feedforward Neural Networks and the Role of Training Noise". *Transactions on Machine Learning Research*, 2023, pp. 1–75. arXiv: 2401.10791 [cs.LG] (p. 206).

[SS02] B. Schölkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. The MIT Press, 2002. DOI: 10.7551/mitpress/4175.001.0001 (p. 81).

[SS07] V. Simoncini and D. B. Szyld. "Recent computational developments in Krylov subspace methods for linear systems". *Numerical Linear Algebra with Applications* **14** (1), 2007, pp. 1–59. DOI: 10.1002/nla.499 (p. 105).

[SS25] T. Stark and L. Steinberger. *Implicit vs. explicit regularization for high-dimensional gradient descent*. Preprint, arXiv:2502.10578. 2025. arXiv: 2502.10578 [math.ST] (pp. 207, 236).

[SS95] D. Saad and S. Solla. "Dynamics of On-Line Gradient Descent Learning for Multilayer Neural Networks". *Advances in Neural Information Processing Systems*. 1995 (p. 160).

[ST22] J. Scott and M. Tůma. "Solving large linear least squares problems with linear equality constraints". *BIT Numerical Mathematics* **62**, 2022, pp. 1765–1787. arXiv: 2106.13142 [math.NA]. DOI: 10.1007/s10543-022-00930-2 (p. 104).

[Ste21] S. Steinerberger. "Randomized Kaczmarz converges along small singular vectors". *SIAM Journal on Matrix Analysis and Applications* **42** (2), 2021, pp. 608–615. arXiv: 2006.16978 [math.NA]. DOI: 10.1137/20M1350947 (p. 43).

[Ste23] S. Steinerberger. "Quantile-based Random Kaczmarz for corrupted linear systems of equations". *Information and Inference: A Journal of the IMA* **12** (1), 2023, pp. 448–465. arXiv: 2107.05554 [math.NA]. DOI: 10.1093/imaiai/iaab029 (pp. 39, 63, 64, 67).

[Stu+12] C. Studer, P. Kuppinger, G. Pope, and H. Bolcskei. "Recovery of Sparsely Corrupted Signals". *IEEE Transactions on Information Theory* **58** (5), 2012, pp. 3115–3130. arXiv: 1102.1621 [cs.IT]. DOI: 10.1109/TIT.2011.2179701 (pp. 34, 39).

[SV09] T. Strohmer and R. Vershynin. "A Randomized Kaczmarz Algorithm with Exponential Convergence". *J. Fourier Anal. Appl.* **15**, 2009, pp. 262–278. arXiv: math/0702226. DOI: 10.1007/s00041-008-9030-4 (pp. 16, 29, 38, 42, 91, 93, 94, 105).

[SW14] A. Saumard and J. A. Wellner. "Log-concavity and strong log-concavity: A review". *Statistics Surveys* **8**, 2014, pp. 45–114. arXiv: 1404.5886 [math.ST]. DOI: 10.1214/14-SS107 (p. 62).

[SW87] R. H. Swendsen and J.-S. Wang. "Nonuniversal critical dynamics in Monte Carlo simulations". *Physical Review Letters* **58** (2), 1987, pp. 86–88. DOI: 10.1103/PhysRevLett.58.86 (p. 304).

[TAP21] N. Tripuraneni, B. Adlam, and J. Pennington. *Covariate Shift in High-Dimensional Random Feature Regression*. Preprint, arXiv:2111.08234. 2021. arXiv: 2111.08234 [stat.ML] (pp. 209, 231).

[TRG16] R. Tappenden, P. Richtárik, and J. Gondzio. "Inexact Coordinate Descent: Complexity and Preconditioning". *Journal of Optimization Theory and Applications* **170**, 2016. math.OC, pp. 144–176. arXiv: 1304.5530. DOI: 10.1007/s10957-016-0867-4 (p. 121).

[Tu+17] S. Tu, S. Venkataraman, A. C. Wilson, A. Gittens, M. I. Jordan, and B. Recht. "Breaking Locality Accelerates Block Gauss-Seidel". *Proceedings of the 34th International Conference on Machine Learning*. 2017, pp. 3482–3491. arXiv: 1701.03863 [math.OC] (pp. 94, 126, 137, 139, 150).

[TV19] Y. S. T. Tan and R. Vershynin. "Phase retrieval via randomized Kaczmarz: theoretical guarantees". *Information and Inference: A Journal of the IMA* **8** (1), 2019, pp. 97–123. arXiv: 1706.09993 [math.NA]. DOI: 10.1093/imaiai/iay005 (p. 29).

[TW23] J. A. Tropp and R. J. Webber. "Randomized algorithms for low-rank matrix approximation: Design, analysis, and applications", 2023. arXiv: 2306.12418 [math.NA] (pp. 97, 111).

[TY10] P. Tseng and S. Yun. "A coordinate gradient descent method for linearly constrained smooth optimization and support vector machines training". *Computational Optimization and Applications* **47**, 2010, pp. 179–206. DOI: 10.1007/s10589-008-9215-4 (p. 94).

[UT19] M. Udell and A. Townsend. "Why Are Big Data Matrices Approximately Low Rank?" *SIAM Journal on Mathematics of Data Science* **1** (1), 2019, pp. 144–160. arXiv: 1705.07474 [cs.LG]. DOI: 10.1137/18M1183480 (p. 81).

[Van+13] J. Vanschoren, J. N. van Rijn, B. Bischl, and L. Torgo. "OpenML: networked science in machine learning". *SIGKDD Explorations* **15** (2), 2013, pp. 49–60. DOI: 10.1145/2641190.2641198 (p. 133).

[Ver18] R. Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge: Cambridge University Press, 2018. DOI: 10.1017/9781108231596 (pp. 55, 58–60).

[WN18] W. Wu and D. Needell. "Convergence of the Randomized Block Gauss-Seidel Method". *SIAM Undergraduate Research Online (SIURO)* **11**, 2018, pp. 369–382. DOI: 10.1137/17S015860 (p. 83).

[Wol89] U. Wolff. "Collective Monte Carlo Updating for Spin Systems". *Physical Review Letters* **62** (4), 1989, pp. 361–364. DOI: 10.1103/PhysRevLett.62.361 (p. 304).

[Woo+08] F. Woolfe, E. Liberty, V. Rokhlin, and M. Tygert. "A fast randomized algorithm for the approximation of matrices". *Applied and Computational Harmonic Analysis* **25** (3), 2008, pp. 335–366. DOI: 10.1016/j.acha.2007.12.002 (p. 111).

[Woo14] D. P. Woodruff. "Sketching as a Tool for Numerical Linear Algebra". *Foundations and Trends in Theoretical Computer Science* **10** (1-2), 2014, pp. 1–157. arXiv: 1411.4357 [cs.DS]. DOI: 10.1561/0400000060 (p. 14).

[WRJ21] A. C. Wilson, B. Recht, and M. I. Jordan. "A Lyapunov Analysis of Accelerated Methods in Optimization". *Journal of Machine Learning Research* **22**, 2021, pp. 1–34. URL: http://jmlr.org/papers/v22/20-195.html (p. 139).

[WS00] C. K. I. Williams and M. Seeger. "Using the Nyström Method to Speed Up Kernel Machines". *Proceedings of the 14th International Conference on Neural Information Processing Systems*. 2000, pp. 661–667. DOI: 10.5555/3008751.3008847 (p. 96).

[WSH24] Y. Wang, R. Sonthalia, and W. Hu. "Near-interpolators: Rapid norm growth and the trade-off between interpolation and generalization". *International Conference on Artificial Intelligence and Statistics*. 2024, pp. 4483–4491. arXiv: 2403.07264 [cs.LG] (pp. 174, 206).

[Wu22] W.-T. Wu. "On two-subspace randomized extended Kaczmarz method for solving large linear least-squares problems". *Numerical Algorithms* **89**, 2022, pp. 1–31. DOI: 10.1007/s11075-021-01104-x (p. 38).

[WW25] J. Weare and R. J. Webber. "Randomly sparsified Richardson iteration: A dimension-independent sparse linear solver". *Communications on Pure and Applied Mathematics*, 2025. To appear. arXiv: 2309.17270 [math.NA]. DOI: 10.1002/cpa.70012 (pp. 258, 259, 263–265).

[WWF24] Z. Wang, D. Wu, and Z. Fan. "Nonlinear spiked covariance matrices and signal propagation in deep neural networks". *Proceedings of the 37th Annual Conference on Learning Theory*. 2024, pp. 1–67. arXiv: 2402.10127 [stat.ML] (p. 242).

[Xu+23] J. Xu, J. Teng, Y. Yuan, and A. C. Yao. "Towards Data-Algorithm Dependent Generalization: a Case Study on Overparameterized Linear Regression". *Proceedings of the 37th Conference on Neural Information Processing Systems*. 2023 (pp. 205, 224).

[XXZ25] R. Xiang, J. Xie, and Q. Zhang. "Randomized block Kaczmarz with volume sampling: Momentum acceleration and efficient implementation", 2025. arXiv: 2503.13941 [math.NA] (p. 89).

[YH22] F. F. Yilmaz and R. Heckel. "Regularization-Wise Double Descent: Why it Occurs and How to Eliminate it". *IEEE International Symposium on Information Theory*. 2022. arXiv: 2206.01378 [cs.LG] (pp. 206, 219).

[YRC07] Y. Yao, L. Rosasco, and A. Caponnetto. "On Early Stopping in Gradient Descent Learning". *Constructive Approximation* **26**, 2007, pp. 289–315. DOI: 10.1007/s00365-006-0663-2 (p. 204).

[YZ13] X.-T. Yuan and T. Zhang. "Truncated Power Method for Sparse Eigenvalue Problems". *Journal of Machine Learning Research* **14** (28), 2013, pp. 899–925. arXiv: 1112.2679 [stat.ML] (p. 257).

[ZF13] A. Zouzias and N. M. Freris. "Randomized Extended Kaczmarz for Solving Least Squares". *SIAM Journal on Matrix Analysis and Applications* **34** (2), 2013, pp. 773–793. arXiv: 1205.5770 [math.NA]. DOI: 10.1137/120889897 (p. 38).

[ZHT06] H. Zou, T. Hastie, and R. Tibshirani. "Sparse Principal Component Analysis". *Journal of Computational and Graphical Statistics* **15** (2), 2006, pp. 265–286. DOI: 10.1198/106186006X113430 (p. 257).

[Zou+23] D. Zou, J. Wu, V. Braverman, Q. Gu, and S. Kakade. "Benign Overfitting of Constant-Stepsize SGD for Linear Regression". *Journal of Machine Learning Research* **24**, 2023, pp. 1–58. arXiv: 2103.12692 [cs.LG] (p. 224).

[ZX18] H. Zou and L. Xue. "A Selective Overview of Sparse Principal Component Analysis". *Proceedings of the IEEE* **106** (8), 2018, pp. 1311–1320. DOI: 10.1109/JPROC.2018.2846588 (p. 257).