

UC Berkeley

UC Berkeley Electronic Theses and Dissertations

Title

Learning Parametric Optimizers: Approximation, Proxies, and Geometry

Permalink

<https://escholarship.org/uc/item/69v9r2bd>

ISBN

9798189271793

Author

Dong, Qiran

Publication Date

2026-05-01

Peer reviewed|Thesis/dissertation

Learning Parametric Optimizers: Approximation, Proxies, and Geometry

By

Qiran Dong

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Industrial Engineering and Operations Research

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Associate Professor Paul Grigas, Chair

Assistant Professor Ying Cui

Assistant Professor Alexander Strang

Spring 2026

Learning Parametric Optimizers: Approximation, Proxies, and Geometry

Copyright 2026

by

Qiran Dong

Abstract

Learning Parametric Optimizers: Approximation, Proxies, and Geometry

by

Qiran Dong

Doctor of Philosophy in Industrial Engineering and Operations Research

University of California, Berkeley

Associate Professor Paul Grigas, Chair

Many decision-making systems require repeatedly solving a family of optimization problems as an instance parameter $\lambda \in \Lambda$ varies. Classical approaches compute this solution map by discretizing the parameter domain, solving many optimization problems, and interpolating between the resulting solutions. This thesis develops an alternative view: learn the optimizer itself as a map from problem instances to decisions. By parameterizing the solution map as $\Phi(\lambda)\beta$, the computation of an entire family of optimizers can be lifted to a single stochastic optimization problem over the coefficient vector β .

The first part of the thesis studies this idea in the language of learning solution paths. Under Euclidean smoothness and strong convexity assumptions, we relate suboptimality in the lifted stochastic problem to uniform solution-path error. We show that the lifted problem satisfies a relaxed weak growth condition, which yields linear convergence of constant step-size SGD to a neighborhood determined by the approximation error of the chosen basis. When the true solution map lies in the span of the basis, this limiting error vanishes. Combining this optimization analysis with approximation theory gives explicit complexity guarantees for differentiable and analytic solution paths, showing when learning the path can require substantially fewer gradient evaluations than discretization.

The second part reframes and generalizes the method as learning an optimization proxy: a fast decision rule that approximates the optimizer of a parametric optimization problem without requiring a new solve at inference time. We introduce complexity-penalized proxy error and show how regularization stabilizes the theory when strong convexity or spectral lower-bound conditions may fail. We prove proxy-performance decompositions that connect in-distribution stochastic training progress to uniform out-of-distribution optimality guarantees. The analysis identifies a fundamental approximation–optimization tradeoff: more expressive proxy classes can reduce the limiting error, but may also worsen conditioning and slow first-order training. We further extend the theory from Euclidean geometry to Bregman geometry through stochastic reference function descent, allowing the training method to respect non-Euclidean domains such as positive orthants and simplices.

The final part isolates the algorithmic geometry underlying these proxy-learning methods. We study stochastic steepest descent under general norm geometries, with particular attention to ℓ_p geometry. This analysis clarifies how the choice of geometry changes descent

directions, stochastic noise behavior, and convergence. In Euclidean geometry, stochastic steepest descent reduces to ordinary SGD, recovering the dense update structure used in the first part of the thesis. In ℓ_1 geometry, the method resembles greedy coordinate selection, making it a first step toward sparse, stagewise, and boosting-style proxy construction. The analysis shows that general norm geometries can introduce batch-size stability thresholds that are absent from the Euclidean RWGC theory, and the numerical experiments suggest that this issue is visible for ℓ_1 updates. Together, the three parts of the thesis provide a theoretical framework for learning parametric optimizers, connecting approximation of solution maps, robustness of optimization proxies, and the geometry of stochastic first-order methods.

To the great mentors I met along the way,
who saw more in me than I saw in myself.

To all my loved ones,
my source of courage and strength.

Contents

List of Figures	iv
List of Tables	vi
1 Introduction	1
1.1 A Coherent Thread	2
1.2 Comparison of the Chapters	2
1.3 Approximation, Optimization, and Geometry	4
1.4 Outlook	4
2 Beyond Discretization: Learning the Solution Path	6
2.1 Introduction	7
2.1.1 Our Contributions	8
2.1.2 Other Related Work	10
2.2 Model Setup and Preliminaries	11
2.3 SGD to Learn the Solution Path	12
2.3.1 Solving Equation (2.2) with SGD	13
2.4 Specialized Results for $\Lambda = [-1, 1]$	15
2.5 Implementation Guidelines	16
2.6 Numerical Experiments	18
2.6.1 Weighted Binary Classification	18
2.6.2 Portfolio Allocation	19
2.6.3 Portfolio Allocation with Moderate Dimensional Λ	21
2.7 Conclusion	23
3 Learning Optimization Proxies under General Convexity	24
3.1 Introduction	24
3.1.1 Our Contributions	26
3.2 Preliminaries	27
3.2.1 Complexity-Penalized Proxy Error	28
3.2.2 Proxy-Performance Decomposition	29
3.3 SGD to Learn an Optimization Proxy	34

3.3.1	Relaxed Weak Growth Condition (RWGC)	34
3.3.2	Solve Problem (3.2) with SGD	34
3.4	Specialized Results	39
3.4.1	Fourier Basis for Periodic Solution Map on $\Lambda = [0, 2\pi]$	39
3.5	From ℓ_2 to General Reference Functions	47
3.5.1	Relative Strong Convexity and Smoothness	47
3.5.2	RWGC and Stochastic Mirror Descent Convergence	48
3.5.3	Stochastic Reference Function Descent to Learn an Optimization Proxy	55
3.5.4	Examples	62
3.6	Numerical Results	67
3.6.1	Moderate-Dimensional Portfolio Allocation in Euclidean Geometry	67
3.6.2	Multitask D-Optimal Design in Bregman Geometry	68
4	Stochastic Steepest Descent	72
4.1	Deterministic Steepest Descent	73
4.1.1	General Steepest Descent under Arbitrary Norms	73
4.1.2	Relationship to ℓ_p Geometry	74
4.1.3	Examples	75
4.1.4	Relationship to Boosting and Stagewise Proxy Construction	76
4.1.5	Relationship to Reference Function Descent	76
4.1.6	Relationship to Stochastic Gradient Descent	78
4.1.7	Convergence under Smoothness and Strong Convexity	78
4.2	Stochastic Steepest Descent	80
4.2.1	Convergence under Smoothness, Strong Convexity, and Mini-Batch RWGC	81
4.3	Numerical Results	86
	Bibliography	92
A	Appendix	97
A.1	Omitted Proofs	97
A.1.1	Proofs from Section 2.2	97
A.1.2	Proofs from Section 2.3	100
A.1.3	Proofs from Section 2.4	102
A.2	Implementation Details	107
A.2.1	Calibrating Uniform Discretization	107
A.2.2	Basis for Moderate Dimensional Portfolio Allocation	109
A.3	Additional Figures	110

List of Figures

2.1	Learning Solution Path of Weighted Binary Classification. See Section 2.6 for setup. We compare our method with $q = 5, 7, 9$ <i>Legendre</i> polynomials as our basis with uniform discretization. Orange line is our adaptive method (c.f. Algorithm 1). Vertical lines indicate when new basis functions are added.	9
2.2	LSP and ALSP for Weighted Binary Classification. (a) Compares methods using Jacobi polynomials. (b) Error in solution path as a function of λ using Jacobi polynomials. (c) and (d) compare $\theta_3(\lambda)$ across λ in the Legendre and Jacobi bases respectively with 7 polynomials.	20
2.3	LSP and ALSP for Portfolio Allocation. Compares methods using the first $q = 4, 9, 16$ bivariate-Legendre polynomials.	21
2.4	Portfolio Allocation, $\dim(\Lambda) = 10$. Comparing the ERM objective for varying q . Panel a) training, Panel b) validation.	22
3.1	Portfolio allocation with random ReLU features. The curves show the unregularized stochastic objective for different values of the Euclidean regularization parameter γ . Larger γ gives faster initial convergence but plateaus at a higher objective; smaller γ converges more gradually and reaches a lower final value.	69
3.2	Portfolio allocation with random Fourier features. The same γ -dependent tradeoff appears with random Fourier features, indicating that the phenomenon is not tied to the particular random basis family.	69
3.3	Multitask D-optimal design. Unregularized stochastic objective for several fixed values of γ in the log-barrier Bregman geometry. Larger γ accelerates the early iterations but reaches a higher plateau, while smaller γ descends more slowly and eventually attains a lower objective.	71

4.1	Objective value versus iteration for stochastic steepest descent under different mini-batch sizes. Each panel fixes the mini-batch size and compares the ℓ_1 stochastic method, the ℓ_2 stochastic method, and deterministic coordinate descent. The deterministic coordinate descent curve provides a full-gradient benchmark for the ℓ_1 coordinate update.	88
4.2	ℓ_1 stochastic steepest descent for different mini-batch sizes, together with deterministic coordinate descent. The plot highlights the batch-size regimes predicted by the theory: very small batches can be unstable, intermediate batches may converge with substantial stochastic effects, and sufficiently large batches show more regular decay.	89
4.3	ℓ_2 stochastic steepest descent for different mini-batch sizes, together with deterministic coordinate descent as a benchmark. The ℓ_2 geometry exhibits the expected stabilization as B increases, with larger batches producing smoother and more nearly linear convergence on the logarithmic scale.	90
A.1	Compare: Ideal Stopping Criterion, Grid Pass Error, and Solution Path Error. (a) weighted binary classification; (b) portfolio allocation. Solution path error and grid pass error are plotted against the corresponding total number of gradient calls incurred in practice. Target solution path error δ is plotted against $c_1 \log(c_2/\delta)/\sqrt{\delta}$, which also equals the total number of gradient calls incurred in practice. Each discrete point in the line plots above corresponds to a discretization grid point $\delta \in G(\delta)$; thus smaller δ is associated with more gradient calls.	109
A.2	$\theta^*(\lambda)$ for Portfolio Allocation from Section 2.6.2. The true solution paths are very smooth in each dimension, suggesting that this problem is highly interpolable by a polynomial basis.	110
A.3	LSP and ALSP for Portfolio Allocation. Compares methods using the first $q = 4, 9, 16$ bivariate-Legendre polynomials. Differs from Figure 2.3 as the y -axis records the number of function evaluations during line-search of L-BFGS instead of gradient calls.	111
A.4	Probability Density Function(PDF) of Beta Distribution over $[0, 1]$ with Various a, b.	112

List of Tables

1.1	Comparison of the three chapters. The dissertation moves from learning a Euclidean solution path, to learning general optimization proxies, to studying the stochastic algorithmic geometry that could support future sparse or boosting-style proxy methods.	3
A.1	Parameters Used in Experiments. The table reports the constants c_1 and c_2 and the discretization grid Δ used for the weighted binary classification and portfolio allocation experiments.	108

Acknowledgments

This dissertation was made possible by the guidance of generous mentors, the help of many kind people, and the love of those closest to me.

I am deeply grateful to my advisor, Paul Grigas, for his guidance, patience, and support throughout my doctoral studies. His insight shaped not only the technical direction of this dissertation, but also the way I think about research. I especially appreciate his patience and encouragement when challenges arose. Our discussions gave me both intellectual excitement and intellectual freedom, and they helped me look more deeply into the mathematical structure underlying algorithmic behavior.

I would also like to thank my collaborator, Vishal Gupta, for his generous discussions, careful feedback, and shared work on the ideas developed in this dissertation. Our conversations helped me see how many of our technical findings could be organized into clearer narratives, which in turn shaped the way they were formalized and presented in this dissertation. I am especially grateful for the many insights he offered on numerical experiments, which helped me grow quickly in a direction that was new to me.

I am also grateful to my dissertation committee members, Professor Ying Cui and Professor Alexander Strang, for their time, thoughtful feedback, and support throughout this process. Their questions and suggestions strengthened the dissertation and encouraged me to view the work from broader perspectives.

I also wish to thank the administrative staff in the Industrial Engineering and Operations Research Department: Ginnie Sadil, Heather Iwata, Jenny Huang, Erica Diffenderfer, and Goldie Negelev, for their help with funding, scheduling, paperwork, milestones, and many other details. Their work made many parts of graduate school smoother, often in ways that were easy to take for granted but are deeply appreciated.

I wish to thank Ziyang Liu and Hao Wang for our weekend adventures around the Bay Area, for exploring good food and culture together, and for helping me recharge during the long journey of graduate school. Those moments offered comfort, joy, and a sense of balance when I needed them most.

I wish to thank Runhan Xie, who went the extra mile to help me with the formatting of this dissertation. More than that, I am grateful for the warmth, care, and companionship in the time we spent together. Every minute helped me feel embraced and recharged.

I would also like to thank Haiyue Zhang, Shaoqing Yi, Lingfei Zhong, Zhihao Liu, Tianyao Zheng, and Zhijian Li for their companionship and friendship when I was new and alone in Berkeley. Their presence helped make an unfamiliar environment feel less distant and more like home.

I wish to thank Renyuan Ma, who encouraged and inspired me to follow this path. I am grateful for our shared excitement about mathematics and many other things, for the confidence he helped me build, and for helping me feel, for the first time, that I was deeply connected to another human being.

I wish to thank Jiaxun Li and Jonathan Huang for making daily routines and house chores feel not like burdens, but like enjoyable fragments of life. Their companionship brought

lightness, humor, and warmth to ordinary days.

I also want to thank Qinghong Wang and Hongyu Lu for their support and friendship across years, land, and ocean. I am grateful for the constancy of their care despite distance and time.

I want to thank Xinyi Wang. Our time together always nourishes my soul, as we synchronize how we see different facets of things and people. She has always reminded me that many of my feelings are not mine alone, but shared. She has also taught me that life is not only about the goals we hope to reach, but also about the small, ordinary moments through which each day quietly unfolds.

I also want to thank Xinyu Li for believing in me and giving me abundant support during some of the most difficult times. Her overflowing courage and vibrant energy helped me move quickly out of turmoil. From her, I learned to act more, think less, and accept and learn from the outcome without being too harsh on myself.

I want to thank Leung Man-tao, @8fenban, and Imaginist for repairing my soul from the daily wear of a world that, in many ways, can erode humanity and alienate us from our true selves. Their words, thoughts, and creations gave me moments of restoration and reflection.

I also want to thank my extended family in New Jersey: my aunt, my cousin and cousin-in-law, and my niece Iris and nephew Ethan. Their warmth and care gave me a sense of family and belonging far from home.

Finally, I would like to extend my deepest appreciation to my parents for their encouragement, love, and support throughout my life. Their sacrifices, trust, and unconditional care have carried me through every stage of this journey. This dissertation would not have been possible without them.

Chapter 1

Introduction

Parametric optimization problems arise when the objective, constraints, data distribution, or design priorities depend on an external parameter. For each value $\lambda \in \Lambda$, one may define an optimizer

$$\boldsymbol{\theta}^*(\lambda) \in \arg \min_{\boldsymbol{\theta}} h(\boldsymbol{\theta}, \lambda).$$

The computational challenge is not merely to solve one such problem, but to understand and approximate the entire input-output relationship

$$\lambda \mapsto \boldsymbol{\theta}^*(\lambda).$$

This relationship is useful whenever decisions must be produced repeatedly across changing scenarios: regularization paths, portfolio tradeoffs, inverse problems, personalized decision rules, resource allocation, and data-driven optimization pipelines all fit this pattern. A direct approach is to discretize Λ , solve the optimization problem at many grid points, and interpolate between solutions. This approach is simple and often effective, but it treats a coherent family of optimization problems as many unrelated solves.

The central idea of this dissertation is to replace that collection of solves by a learned optimization object. Instead of computing $\boldsymbol{\theta}^*(\lambda)$ separately at each parameter value, we approximate the optimizer map by a parametric decision rule, such as

$$\lambda \mapsto \boldsymbol{\Phi}(\lambda)\boldsymbol{\beta},$$

and train the coefficients $\boldsymbol{\beta}$ by solving a single stochastic optimization problem over sampled values of λ . The learned object can then be evaluated quickly at new parameter values, turning a family of optimization problems into a reusable proxy for optimization.

This point of view raises three intertwined questions. First, when does optimizing the lifted stochastic problem produce a uniformly accurate approximation to the solution map? Second, how should this idea be generalized beyond smooth Euclidean solution paths to broader optimization proxies, weaker convexity assumptions, and constrained non-Euclidean domains? Third, what role does the geometry of the training algorithm play, especially when

one wants to move beyond dense gradient updates toward sparse, greedy, or boosting-style methods? The three chapters of the dissertation answer these questions in sequence.

1.1 A Coherent Thread

The dissertation begins with solution paths because they are the most direct expression of the idea. A solution path is a map from a parameter to an optimizer. Chapter 1 asks whether this map can be learned as a single object rather than computed by discretization. The answer developed there is affirmative under Euclidean smoothness and strong convexity assumptions: if the solution map is well approximated by a chosen basis and if the stochastic training problem is solved accurately enough, then the learned map approximates the entire solution path. The theory combines approximation error, optimization error, and the conditioning of the lifted stochastic problem.

Chapter 2 broadens the language from solution paths to optimization proxies. This shift matters because in many applications the goal is not to trace a geometric path in parameter space, but to build a fast decision rule for a distribution or family of optimization instances. The chapter introduces complexity-penalized proxy error, studies in-distribution and uniform proxy performance, and uses regularization to stabilize settings where the assumptions of Chapter 1 may fail. It also extends the training geometry from Euclidean SGD to stochastic reference-function descent, allowing the method to respect domains such as simplices and positive orthants.

Chapter 3 steps back from the proxy model itself and studies the geometry of stochastic first-order training. The first two chapters show that convergence depends not only on approximation power, but also on the geometry of the algorithm used to train the proxy. Chapter 3 therefore studies stochastic steepest descent under general norm geometries. This provides a first step toward understanding sparse and stagewise updates, including the boosting-inspired idea of building a proxy by greedily adding basis elements. In Euclidean geometry, stochastic steepest descent reduces to ordinary SGD; in ℓ_1 geometry, it resembles greedy coordinate or weak-learner selection. The chapter shows that these non-Euclidean stochastic directions introduce stability issues that do not appear in the same way in the Euclidean theory.

1.2 Comparison of the Chapters

The three chapters can be read as a progression in what is being learned and in what part of the learning process is placed under theoretical scrutiny. Chapter 1 starts with the most concrete object: the optimizer map itself. Chapter 2 keeps the same stochastic-learning construction but expands the object into a general optimization proxy, where average and uniform decision quality both matter. Chapter 3 then shifts attention from the proxy class to the training direction, asking how the geometry of stochastic first-order updates changes the behavior of the learning algorithm.

The comparison in Table 1.1 summarizes this progression along three axes. The first axis is the primary object of study: solution paths, optimization proxies, and stochastic descent directions. The second is the main theoretical question asked in each chapter. The third is the algorithmic geometry used for training, which evolves from Euclidean SGD to Bregman reference-function methods and then to general norm steepest descent.

Chapter	Primary object	Main question	Algorithmic Geometry for Training
Chapter 1: Learning the Solution Path	The optimizer map $\lambda \mapsto \theta^*(\lambda)$, approximated by $\Phi(\lambda)\beta$.	Can a whole solution path be learned by solving one stochastic optimization problem instead of many discretized problems?	Euclidean SGD. Establishes proxy/path error decomposition, verifies RWGC, and connects basis approximation error to gradient complexity.
Chapter 2: Learning Optimization Proxies	A fast decision rule for a family or distribution of optimization instances.	How can the solution-path idea be generalized under weaker convexity, regularization, in-distribution performance, and constrained domains?	Euclidean and Bregman geometry. Introduces complexity-penalized proxy error, sharper in-distribution guarantees, uniform proxy bounds, and stochastic reference-function descent.
Chapter 3: Stochastic Steepest Descent	The stochastic first-order direction used to train a proxy or build it stagewise.	How does the norm geometry of stochastic steepest descent affect stability, batching, and the possibility of boosting-style proxy construction?	General norm geometry, with emphasis on ℓ_p . Shows that ℓ_2 recovers SGD, ℓ_1 resembles greedy coordinate selection, and general norms can introduce batch-size thresholds.

Table 1.1: Comparison of the three chapters. The dissertation moves from learning a Euclidean solution path, to learning general optimization proxies, to studying the stochastic algorithmic geometry that could support future sparse or boosting-style proxy methods.

This structure is meant to make the chapters cumulative rather than separate. Chapter 1 supplies the basic approximation–optimization template: represent the optimizer map, train the coefficients by stochastic optimization, and translate training error into path error. Chapter 2 shows that the same template survives in a broader proxy-learning setting

once regularization, distributional performance, and non-Euclidean constraints are included. Chapter 3 identifies a remaining algorithmic question: if one wants future proxy methods to be adaptive or stagewise, then the stochastic training direction itself becomes part of the theory. In this sense, the dissertation moves from learning the object, to measuring the quality of the learned object, to understanding the geometry of the algorithm that learns it.

1.3 Approximation, Optimization, and Geometry

Across the dissertation, the same decomposition appears in different forms. A learned proxy is only as good as the class used to represent it, so approximation error determines the limiting performance. At the same time, the proxy must be trained by an algorithm, so optimization error determines how quickly the learned coefficients approach the best available proxy in the chosen class. Finally, the geometry of the training method determines both the conditioning of the optimization problem and the stability of stochastic updates.

Chapter 1 emphasizes approximation and optimization in Euclidean geometry. The chosen basis controls how well the solution path can be represented, while the spectral properties of the basis affect the convergence rate of SGD. The result is a concrete approximation–optimization tradeoff: adding basis functions can reduce limiting error, but can also worsen conditioning.

Chapter 2 shows that this tradeoff persists in a broader proxy-learning framework. Regularization improves conditioning and makes the theory robust to weak convexity or degenerate spectral structure, but it also introduces bias through the complexity penalty. The chapter also distinguishes between in-distribution and uniform proxy guarantees. The former aligns with average decision quality on future instances from the training distribution; the latter gives stronger worst-case control over the parameter space.

Chapter 3 focuses on geometry itself. The Euclidean setting has a special structure: stochastic steepest descent is ordinary SGD, and the relaxed weak growth condition leads to clean constant-step guarantees. Under a general norm, however, the stochastic gradient is transformed into a geometry-dependent steepest direction. This transformation can amplify noise and create a batch-size threshold in the convergence analysis. The numerical experiments suggest that this issue is visible in ℓ_1 geometry, where the stochastic gradient determines which coordinate is selected.

1.4 Outlook

The dissertation therefore develops a progression from representation, to proxy performance, to training geometry. Chapter 1 introduces the central representation idea: learn the optimizer map directly. Chapter 2 turns that idea into a more general theory of optimization proxies, including regularization and non-Euclidean domains. Chapter 3 studies the stochastic geometry needed to move toward adaptive, sparse, and stagewise proxy training methods.

One possible continuation is a true boosting-style algorithm for optimization proxies. Such a method would start with a large dictionary of candidate basis functions and build the proxy sequentially, adding components that most improve the stochastic proxy objective. The analysis in Chapter 3 does not complete that program, but it identifies a central obstacle: stochastic greedy directions can behave differently across geometries. Understanding when this instability is avoidable, and when it is intrinsic, remains an important question for future work.

Chapter 2

Beyond Discretization: Learning the Solution Path

This dissertation begins with a basic question about parametric convex optimization: when an optimization problem depends on a parameter $\lambda \in \Lambda$, how should one compute the optimizer $\theta^*(\lambda)$ across the entire parameter domain? A traditional answer is discretization: select a grid of parameters, solve the optimization problem at each grid point, and interpolate between the resulting solutions. This approach is natural and often effective, but it treats the family of problems as a collection of separate tasks.

The first step in the research program of this dissertation is to replace this collection of solves with a single learning problem. We represent the solution map by a basis expansion $\Phi(\lambda)\beta$ and choose the coefficients by minimizing one stochastic objective over randomly sampled parameter values. This lifts the computation of the entire solution path into a finite-dimensional stochastic optimization problem. The main theoretical question is then whether solving this lifted problem also gives a uniformly accurate approximation to the original solution path.

This chapter develops that first theory under Euclidean smoothness and strong convexity assumptions. It relates optimization suboptimality in the lifted stochastic problem to solution-path error, verifies a relaxed weak growth condition, and uses this structure to prove convergence guarantees for constant step-size SGD. It then combines these optimization results with approximation theory to show how the smoothness of $\lambda \mapsto \theta^*(\lambda)$ determines the number of basis functions and gradient calls needed to reach a target error.

This chapter is the point where the central idea of this research journey first appears: instead of following or discretizing a path, learn it as a single object. The next chapter broadens this idea from a solution path language to the more general language of optimization proxies, where the learned object is a decision rule for a family of optimization instances.

2.1 Introduction

Many decision-making applications entail solving a family of *parametrized* optimization problems:

$$\boldsymbol{\theta}^*(\lambda) \in \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^d} h(\boldsymbol{\theta}, \lambda), \quad \lambda \in \Lambda, \quad (2.1)$$

where Λ is an arbitrary set of parameters indexing the problems. (We assume Problem (2.1) admits an optimal solution for each $\lambda \in \Lambda$.) In such applications, we often seek to compute the entire solution path $\{\boldsymbol{\theta}^*(\lambda) : \lambda \in \Lambda\}$ in order to present experts with a portfolio of possible solutions to compare and assess tradeoffs.

As an example, consider the p -norm fair facility location problem [24], which minimizes facility opening costs while incorporating l_p -regularization to promote fairness across socio-economic groups. Since there is no obvious choice for p , we might prefer computing solutions for all p and allowing experts to (qualitatively) assess the resulting solutions. Many other applications entail navigating similar tradeoffs, including upweighting the minority class in binary classification to tradeoff Type I and II errors or aggregating features to increase interpretability at the expense of accuracy. In each case, we seek the entire solution path because selecting the “best” solution requires weighing a variety of criteria, some of which may be qualitative. These settings differ from classical hyperparameter tuning where there is a clear auxiliary criterion (like out-of-sample performance), and it would be enough to identify the single $\lambda^* \in \Lambda$ such that $\boldsymbol{\theta}^*(\lambda^*)$ optimizes this criteria.

The most common approach to learning the entire solution path is discretization: discretize Λ , solve Problem (2.1) at each grid point, and interpolate the resulting solutions. With enough grid points, interpolated solutions are approximately optimal along the entire path. Several authors have sought to determine the minimal the number of discretization points needed to achieve a target level of accuracy, usually under minimal assumptions on $h(\boldsymbol{\theta}, \lambda)$.

[22] considers convex optimization problems over the unit simplex when $\Lambda \subseteq \mathbb{R}$ and show that learning the solution path to accuracy ϵ requires at least $O(1/\epsilon)$ grid points. [21] consider the case where $h(\boldsymbol{\theta}, \lambda)$ is concave in λ and $\Lambda \subseteq \mathbb{R}$ and show only $O(1/\sqrt{\epsilon})$ points are needed. More recently, [36] relate the required number of points to the regularity of $h(\boldsymbol{\theta}, \lambda)$, arguing that if h is uniformly convex of order d , one requires $O(1/\sqrt[d]{\epsilon})$ points.

A potential criticism of this line of research is that the computational complexity of these methods depends not only on the number of grid points but also depends on the amount of work per grid point, e.g., as measured by the number of gradient calls used by a first-order method. Generally speaking, these methods do not share much information across grid points, at most warm-starting subsequent optimization problems. However, gradient evaluations at nearby grid points contain useful information for optimizing the current grid point, and leveraging this information presents an opportunity to reduce the total work.

2.1.1 Our Contributions

We propose a novel, simple algorithmic procedure to learn the solution path that applies to an arbitrary set Λ .

The key idea is to replace the family of problems in (2.1) with a *single* stochastic optimization problem. This stochastic optimization problem depends on two user-specified components: a distribution $\mathbb{P}_\lambda$ over values of $\lambda \in \Lambda$ and a collection of basis functions $\Phi_j(\cdot) : \Lambda \rightarrow \mathbb{R}^d$, $j = 1, \dots, p$. We then seek to approximate $\theta^*(\lambda)$ as a linear combination of basis functions, $\Phi_{1:p}(\lambda)\hat{\beta}$, where $\Phi_{1:p} := [\Phi_1 \ \Phi_2 \ \dots \ \Phi_p]$ and $\hat{\beta}$ is an approximate solution to

$$\min_{\beta \in \mathbb{R}^p} \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} [h(\Phi_{1:p}(\tilde{\lambda})\beta, \tilde{\lambda})]. \quad (2.2)$$

In contrast to discretization schemes which only leverage local information, (stochastic) gradient evaluations of Problem (2.2) inform *global* structure. Moreover, through a suitable choice of basis functions, we can naturally accomodate complex sets Λ in contrast to earlier work that only treats $\Lambda \subseteq \mathbb{R}$. Finally, any stochastic optimization routine can be used on Problem (2.2) beyond just SGD (see, e.g., [29, 9], among others).

Despite its simplicity, our approach can approximate the solution path to higher accuracy than discretization with fewer gradient evaluations. See Figure 2.1 for a sample of our numerical results on weighted binary classification using SGD to solve Problem (2.2). We prove this behavior is typical. Loosely,

- i) When using constant step-size SGD to solve Problem (2.2), we prove that the uniform error of our learned path $\Phi_{1:p}(\lambda)\hat{\beta}$ to the true path $\theta^*(\lambda)$ exhibits linear convergence to an irreducible constant that is proportional to the expressiveness of the basis (Theorem 2.3.4). This behavior is already visible in Figure 2.1.

The proof of this result utilizes ideas from the convergence of SGD under various “growth conditions” of the gradient [9, 40, 53, 32, 8]. See [27] for a summary and comparison of these various conditions. Our contribution to this literature is to relate the expressiveness of the basis $\Phi_{1:p}(\lambda)$ to a relaxed, weak-growth condition of [23, Lemma 2.4]. Indeed, we show that under some assumptions, Problem (2.2) always satisfies this relaxed weak-growth condition, and if the solution path lies in the span of the basis, it satisfies the weak-growth condition of [53]. This allows us to leverage results from those works to establish Theorem 2.3.4.

In special cases, we can leverage a priori knowledge of the structure of $\theta^*(\lambda)$ to prove stronger results. For example, suppose $\Lambda = [-1, 1]$, and we use Legendre polynomials as our basis. We prove that

- ii) If the solution path $\theta^*(\lambda)$ is ν -times differentiable, then using $p = O\left(\epsilon^{\frac{1}{2(1-\nu)}}\right)$ polynomials ensures that after $T = O\left(\epsilon^{\frac{1}{1-\nu}} \log(1/\epsilon)\right)$ gradient calls, we obtain an ϵ -approximation to the solution path (Theorem 2.4.3).

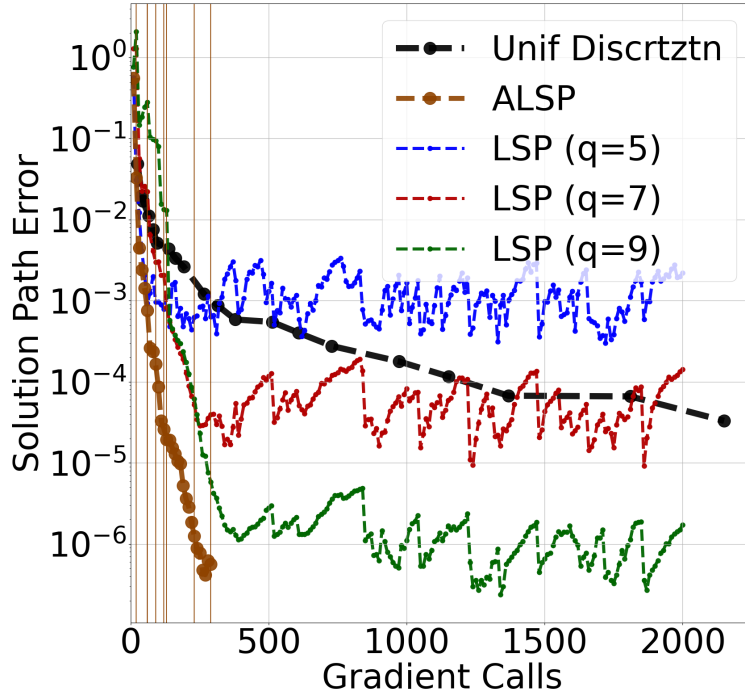


Figure 2.1: **Learning Solution Path of Weighted Binary Classification.** See Section 2.6 for setup. We compare our method with $q = 5, 7, 9$ *Legendre* polynomials as our basis with uniform discretization. Orange line is our adaptive method (c.f. Algorithm 1). Vertical lines indicate when new basis functions are added.

For comparison, [36] implies that when $h(\boldsymbol{\theta}, \lambda)$ is strongly-convex, discretization requires at least $O(\epsilon^{-1/2})$ points. Hence, even if the optimization problem at each grid point can be solved with 1 gradient evaluation, our approach requires asymptotically fewer evaluations whenever $\nu > 3$. If ν is large, the savings is substantive.

- iii) If the solution path $\boldsymbol{\theta}^*(\lambda)$ is analytic ($\nu = \infty$), then using $p = O(\log(1/\epsilon))$ basis polynomials ensures that after $T = O(\log^2(1/\epsilon) \log \log(1/\epsilon))$ iterations, we obtain an ϵ -approximation to the solution path (Theorem 2.4.5).

This is almost *exponentially* less work than discretization for unidimensional hyperparameter space.

These specialized results strongly suggest that in the general setting our approach should be competitive as long as we use enough basis functions. This observation motivates a natural heuristic in which we adaptively add basis functions whenever the stochastic optimization routine (e.g., SGD) stalls (see Algorithm 1 and Section 2.5 for details). We illustrate this idea in Figure 2.1 (orange line) where we can see that by progressively adding functions we drive the uniform error to zero rapidly.

Algorithm 1 Adaptively Learn the Solution Path (ALSP)

- 1: **Initialize:** $\Phi_{1:0}(\cdot) \leftarrow [\]$ and $\hat{\beta}_0 \leftarrow [\]$
- 2: **Return:** A sequence of coefficients: $\hat{\beta}_1, \hat{\beta}_2, \dots$
- 3: **At iteration** $p = 1, 2, \dots$ **do**
- 4: Append a new basis function:

$$\Phi_{1:p}(\cdot) \leftarrow [\Phi_{1:p-1}(\cdot) \quad \Phi_p(\cdot)]$$

- 5: Run a first-order method, starting from
- 6: $[\hat{\beta}_{p-1}, 0] \in \mathbb{R}^p$, to obtain

$$\hat{\beta}_p \approx \arg \min_{\beta \in \mathbb{R}^p} \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} [h(\Phi_{1:p}(\tilde{\lambda})\beta, \tilde{\lambda})].$$

2.1.2 Other Related Work

There is a rich literature on specialized methods for computing the solution path under *regularization*, i.e., where $\lambda \in \mathbb{R}_+$ is the weight on a convex regularizer. These methods generally employ a path-following algorithm. See, for example, [47, 19, 43] for cases where $\theta^*(\lambda)$ is smooth, and [48] for when it is piecewise linear. The LASSO, or ℓ_1 regularized regression, has received special emphasis [41, 15, 50], as has support vector machines [26] and certain structured regularizers [4]. [33] adopts a slightly different perspective than this existing literature, using ordinary differential equations to motivate a second-order method to compute the path. Overall, like our work, these works generally consider the total computational time to compute the path, not just the number of discretization points.

However, our work differs from this literature in two important ways. First, our proposed method applies to general $\lambda \in \Lambda$ which may be multidimensional, whereas the previous literature largely focuses on the case of $\lambda \in \mathbb{R}_+$. This limitation of the previous literature is perhaps fundamental as *path*-following algorithms do not generalize easily to multidimensional settings. Second, we treat a general function $h(\cdot, \cdot)$, not just the case of regularization. Despite our generality, our paper performs a relatively precise computational analysis by bounding total computation time in terms of the number of gradient calls, i.e., we treat the gradient as a block-box oracle instead of a more complicated black-box consisting of minimizing the objective $h(\cdot, \cdot)$ at fixed parameter values. Furthermore, our numerical experiments go beyond simple regularization examples that are common in the literature by considering the upweighing of a minority class in classification and weighting different components of a multi-objective optimization problem. Thus, our method applies more broadly than the aforementioned works.

2.2 Model Setup and Preliminaries

We denote the ℓ_2 -norm by $\|\cdot\|$ throughout. We focus on the case of smooth, strongly convex functions:

Assumption 2.2.1 (Uniform Smoothness and Strong Convexity). There are constants $0 < \mu \leq L$ such that, for all $\lambda \in \Lambda$, $h(\cdot, \lambda)$ is μ -strongly convex and L -smooth, i.e., for all $\boldsymbol{\theta}, \bar{\boldsymbol{\theta}} \in \mathbb{R}^d$,

$$\frac{\mu}{2} \|\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}\|^2 \leq h(\boldsymbol{\theta}, \lambda) - h(\bar{\boldsymbol{\theta}}, \lambda) - \nabla h(\bar{\boldsymbol{\theta}}, \lambda)^\top (\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}) \leq \frac{L}{2} \|\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}\|^2.$$

For any candidate solution path $\boldsymbol{\theta}(\cdot) : \Lambda \rightarrow \mathbb{R}^d$, we define the *solution path error* of $\boldsymbol{\theta}(\cdot)$ by

$$\epsilon_{\text{sp}}(\boldsymbol{\theta}(\cdot)) := \sup_{\lambda \in \Lambda} \{h(\boldsymbol{\theta}(\lambda), \lambda) - h(\boldsymbol{\theta}^*(\lambda), \lambda)\}.$$

An ϵ -*solution path* is a solution path $\boldsymbol{\theta}(\cdot)$ such that $\epsilon_{\text{sp}}(\boldsymbol{\theta}(\cdot)) < \epsilon$. Finally, given any vector of coefficients $\boldsymbol{\beta}$ and basis functions $\boldsymbol{\Phi}_{1:p}(\cdot)$, we define $\epsilon_{\text{sp}}(\boldsymbol{\beta}) := \epsilon_{\text{sp}}(\boldsymbol{\Phi}_{1:p}(\cdot)\boldsymbol{\beta})$ to be the solution path error of $\boldsymbol{\Phi}_{1:p}(\cdot)\boldsymbol{\beta}$.

As mentioned, our method depends on two user-chosen parameters: a distribution $\mathbb{P}_\lambda$ over $\lambda \in \Lambda$ and a series of basis functions $\boldsymbol{\Phi}_j : \Lambda \mapsto \mathbb{R}^d$, for $j = 1, \dots, p$. We require some minor assumptions on these choices:

Assumption 2.2.2. (Distribution, Basis Functions, and Linear Independence).

- i) It is easy to generate i.i.d. samples from $\mathbb{P}_\lambda$.
- ii) It is easy to compute $\boldsymbol{\Phi}_j(\lambda) \in \mathbb{R}^d$ for any $\lambda \in \Lambda$, and $1 \leq j \leq p$.
- iii) There does not exist $\boldsymbol{\beta} \in \mathbb{R}^p$ with $\boldsymbol{\beta} \neq \mathbf{0}$ such that $\boldsymbol{\Phi}_{1:p}(\tilde{\lambda})\boldsymbol{\beta} = \mathbf{0}$ holds $\mathbb{P}_\lambda$ -almost everywhere.

This last condition is a linear independence assumption. If violated, one can remove a function without affecting the span of the basis on the support of $\mathbb{P}_\lambda$.

Given $\boldsymbol{\Phi}_{1:p}$, we define the minimal solution path error for this basis by

$$\epsilon_{\text{sp}}^* := \inf_{\boldsymbol{\beta} \in \mathbb{R}^p} \epsilon_{\text{sp}}(\boldsymbol{\beta}).$$

We also define the following auxiliary constants:

$$\begin{aligned} C &:= \sup_{\lambda \in \Lambda} \sigma_{\max}(\boldsymbol{\Phi}(\lambda)^\top \boldsymbol{\Phi}(\lambda)), \\ c &:= \sigma_{\min} \left\{ \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} \left[\boldsymbol{\Phi}(\tilde{\lambda})^\top \boldsymbol{\Phi}(\tilde{\lambda}) \right] \right\}. \end{aligned} \tag{2.3}$$

In words, C is a uniform bound on the largest eigenvalue of the positive semidefinite matrix $\Phi(\lambda)^\top \Phi(\lambda)$, and c is the smallest eigenvalue of the corresponding expected matrix. By construction, $0 \leq c \leq C$. Under Assumption 2.2.2, both constants are strictly positive.

Lemma 2.2.3 (Positive Spectral Values). *If Assumption 2.2.2 holds, then $0 < c \leq C$.*

We can now state our first key result which relates the suboptimality of a feasible solution β in Equation (2.2) to its solution path error. Define

$$\beta_{\text{avg}}^* \in \arg \min_{\beta \in \mathbb{R}^p} \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} [h(\Phi(\tilde{\lambda})\beta, \tilde{\lambda})].$$

Theorem 2.2.4 (Relating Suboptimality to Solution Path Error). *Under Assumptions 2.2.1 and 2.2.2, for any $\beta \in \mathbb{R}^p$, we have*

$$\epsilon_{\text{sp}}(\beta) \leq 2CL\|\beta - \beta_{\text{avg}}^*\|^2 + \left(\frac{8CL}{c\mu}\right)\epsilon_{\text{sp}}^*.$$

This bound decomposes into two terms, one proportional to $\|\beta - \beta_{\text{avg}}^*\|^2$, which represents the suboptimality of β and the other proportional to ϵ_{sp}^* , which measures the maximal expressiveness of the basis. By solving Equation (2.2) to greater accuracy, we can drive down the first term, but we will not affect the second term. To reduce the second term, we must add basis functions to obtain a better quality approximation.

Theorem 2.2.4 shows any algorithm for solving Equation (2.2) can be used to approximate the solution path. In the next section, we argue that when ϵ_{sp}^* is small, constant step-size SGD for Equation (2.2) exhibits linear convergence to a constant proportional to ϵ_{sp}^* , making it an ideal algorithm to study.

2.3 SGD to Learn the Solution Path

In this section, we apply constant step-size SGD to solve Equation (2.2). The key idea is to show that Equation (2.2) satisfies a certain growth condition and apply [23, Theorem 3.1]. One minor detail is that [23] proves their result under a stronger “expected smoothness” condition for the setting where the objective Equation (2.2) is a finite sum. However, the result holds more generally under a weaker condition (Eq. (9) of their work). Hence, for clarity, we first restate this condition and the more general result.

The following definition is motivated by [23, Lemma 2.4]:

Definition 2.3.1 (Relaxed Weak Growth Condition (RWGC)). *Consider a family of functions $g(\cdot, z) : \mathbb{R}^d \rightarrow \mathbb{R}$, $z \sim \mathbb{P}_z$, and $G(\cdot) := \mathbb{E}_{z \sim \mathbb{P}_z} [g(\cdot, z)]$. Let $G^* := \min_{\mathbf{w} \in \mathbb{R}^d} G(\mathbf{w})$. Then, g and G are said to satisfy the relaxed weak growth condition with constants $\rho \geq 0$ and $\sigma \geq 0$, if for all $\mathbf{w} \in \mathbb{R}^d$,*

$$\mathbb{E}_{z \sim \mathbb{P}_z} [\|\nabla g(\mathbf{w}, z)\|^2] \leq 2\rho(G(\mathbf{w}) - G^*) + \sigma^2.$$

When $\sigma = 0$, Definition 2.3.1 recovers the *weak growth condition* of [53]. In this case, the variance of the stochastic gradients goes to zero as we approach an optimal solution. Hence, at optimality, not only is the expectation of the gradient zero, but it is zero almost surely in z . For regression problems, this condition corresponds to interpolation.

Definition 2.3.1 is implied by a standard second moment condition on the gradients (take $\rho = 0$). However, the most interesting cases are when σ^2 is small relative to ρ and $\rho > 0$. (This will be the case for Equation (2.2). See Section 2.3.1.)

We then have the following theorem.

Theorem 2.3.2 ([23]). *Suppose that the RWGC holds for the family of functions $g(\cdot, z) : \mathbb{R}^d \rightarrow \mathbb{R}$, and that $G(\cdot) := \mathbb{E}_{z \sim \mathbb{P}_z} [g(\cdot, z)]$ is μ_g -strongly convex for some $\mu_g > 0$. Consider the SGD algorithm, initialized at $\mathbf{w}_0 \in \mathbb{R}^d$, with iterations*

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \frac{\bar{\eta}}{\rho} \nabla g(\mathbf{w}_t, z_t),$$

where $z_t \sim \mathbb{P}_z$ is a random sample and $\bar{\eta} \in \left(0, \min \left\{1, \frac{\rho}{\mu_g}\right\}\right]$ parametrizes the step-size. Let $\mathbf{w}^* := \arg \min_{\mathbf{w} \in \mathbb{R}^d} G(\mathbf{w})$. Then for all $t \geq 0$, we have

$$\mathbb{E}[\|\mathbf{w}_t - \mathbf{w}^*\|^2] \leq \left(1 - \frac{\bar{\eta}\mu_g}{\rho}\right)^t \|\mathbf{w}_0 - \mathbf{w}^*\|^2 + \frac{\bar{\eta}\sigma^2}{\rho\mu_g}.$$

In other words, constant step-size SGD under the RWGC exhibits a “fast” linear convergence in expectation up to a constant that is directly proportional to σ^2 , after which it stalls. When $\sigma^2 = 0$, we exactly recover the result of [53]. To keep the exposition self-contained, we provide a proof in the appendix.

2.3.1 Solving Equation (2.2) with SGD

We next prove that Equation (2.2) satisfies RWGC, and in particular, that the “ σ^2 ” term depends on ϵ_{sp}^* , the minimal solution path error of the basis. To our knowledge, we are the first to make this observation.

Define

$$\begin{aligned} f(\beta, \lambda) &:= h(\Phi(\lambda)\beta, \lambda), \\ F(\beta) &:= \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} [f(\beta, \tilde{\lambda})] = \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} [h(\Phi(\tilde{\lambda})\beta, \tilde{\lambda})], \end{aligned} \tag{2.4a}$$

Lemma 2.3.3 (Equation (2.2) satisfies RWGC). *Suppose that Assumptions 2.2.1 and 2.2.2 hold and recall the constants defined in Equation (2.3). Then, the family of functions $f(\cdot, \lambda)$ and the function $F(\cdot)$, defined in Equation (2.4), satisfy the relaxed weak growth condition*

(RWGC) with constants $\rho = CL$ and $\sigma^2 = 2CL\epsilon_{\text{sp}}^*$. Namely, for all $\beta \in \mathbb{R}^d$,

$$\mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} [\|\nabla f(\beta, \tilde{\lambda})\|^2] \leq 2CL(F(\beta) - F^*) + 2CL\epsilon_{\text{sp}}^*.$$

In addition, $F(\cdot)$ is $c\mu$ -strongly convex.

We now combine Theorems 2.2.4 and 2.3.2 and Lemma 2.3.3 to yield our main result: a bound on the expected solution path error for constant step-size SGD. Recall constant step-size SGD in this setting yields the iteration

$$\beta_{t+1} \leftarrow \beta_t - \eta \nabla_{\beta} h(\Phi(\tilde{\lambda}_t)\beta_t, \tilde{\lambda}_t),$$

where

$$\nabla_{\beta} h(\Phi(\tilde{\lambda}_t)\beta_t, \tilde{\lambda}_t) = \Phi(\tilde{\lambda}_t)^\top \nabla_{\theta} h(\Phi(\tilde{\lambda}_t)\beta_t, \tilde{\lambda}_t),$$

and $\tilde{\lambda}_t \sim \mathbb{P}_\lambda$. We assume that the gradient $\nabla_{\theta} h(\theta, \lambda)$ is easily computable for any θ and λ .

We then have:

Theorem 2.3.4 (Expected Solution Path Error Convergence for SGD). *Under Assumptions 2.2.1 and 2.2.2, consider applying SGD to Equation (2.2) with a constant step-size $\eta = \frac{\bar{\eta}}{CL}$ parameterized by $\bar{\eta} \in (0, 1]$. Then, after T iterations, the expected solution path error is at most*

$$\begin{aligned} \mathbb{E} [\epsilon_{\text{sp}}(\beta_T)] &\leq 2CL\|\beta_0 - \beta_{\text{avg}}^*\|^2 \left(1 - \frac{\bar{\eta}c\mu}{CL}\right)^T \\ &\quad + \left(\frac{4CL(\bar{\eta} + 2)}{c\mu}\right) \epsilon_{\text{sp}}^*. \end{aligned}$$

In particular, when $\epsilon_{\text{sp}}^* > 0$, then for $T \geq \left\lceil \frac{CL}{\bar{\eta}c\mu} \log \left(\frac{c\mu\|\beta_0 - \beta_{\text{avg}}^*\|^2}{2\bar{\eta}\epsilon_{\text{sp}}^*} \right) \right\rceil$,

$$\mathbb{E} [\epsilon_{\text{sp}}(\beta_T)] \leq \left(\frac{8CL(\bar{\eta} + 1)}{c\mu} \right) \epsilon_{\text{sp}}^*.$$

When $\epsilon_{\text{sp}}^* = 0$, then for any $\epsilon > 0$ and $T \geq \left\lceil \frac{CL}{\bar{\eta}c\mu} \log \left(\frac{2CL\|\beta_0 - \beta_{\text{avg}}^*\|^2}{\epsilon} \right) \right\rceil$,

$$\mathbb{E} [\epsilon_{\text{sp}}(\beta_T)] \leq \epsilon.$$

Theorem 2.3.4 highlights the role of the basis functions in our results. First we see that as $T \rightarrow \infty$, the expected solution path error plateaus at a constant proportional to $\frac{CL}{c\mu} \epsilon_{\text{sp}}^*$. As mentioned, ϵ_{sp}^* measures the expressiveness of the basis, and we expect ϵ_{sp}^* to be small as we add more basis functions. We interpret $\frac{CL}{c\mu}$ as a condition number for Equation (2.2), which also depends on the choice of basis through Equation (2.3). This constant increases as we grow the basis. Finally, the iteration complexity scales linearly with this condition

number. Thus, an “ideal” basis must navigate this tradeoff between ϵ_{sp}^* and C/c .

Fortunately, there exists a rich theory on function approximation that studies the relationship between basis functions, uniform error, and eigenspectra. In the next section we leverage this theory to provide a comparison of our method with existing discretization techniques in a specialized setting.

2.4 Specialized Results for $\Lambda = [-1, 1]$

We next leverage results from function approximation theory to bound the number of basis functions needed to achieve a target solution path error. We focus on the case $\Lambda = [-1, 1]$ as it facilitates simple comparisons to existing results and elucidates key intuition.

We use a simple basis: we approximate each component $i = 1, \dots, d$ of $\theta_i^*(\cdot)$ by the first q Legendre polynomials. Hence the total number of basis functions is $p = qd$. Recall, the Legendre polynomials form an orthogonal basis on $[-1, 1]$ with respect to the uniform distribution, i.e., $\mathbb{E}_{\tilde{\lambda} \sim \text{Unif}[-1, 1]} [P_n(\tilde{\lambda})P_m(\tilde{\lambda})] = \frac{2}{2n+1} \mathbb{1}\{n = m\}$, where P_n and P_m are the n^{th} and m^{th} Legendre polynomial, respectively. Since we approximate each component separately, the matrix $\Phi(\lambda) \in \mathbb{R}^{d \times qd}$ is block-diagonal, with d blocks of size $1 \times q$.

For this basis, depending on the value of $p = qd$ implied by the choice of q , let C_p, c_p refer to the constants (2.3). We can bound the constant C_p/c_p :

Lemma 2.4.1 (C_p/c_p for Legendre Polynomials). *Take $\Lambda = [-1, 1]$ and $\mathbb{P}_\lambda$ to be the uniform distribution on $[-1, 1]$. Then, for the above basis, $C_p/c_p \leq q^2$.*

Notice, that this constant is independent of d and grows mildly with q . This is not true of all polynomial bases. One can check empirically that for the monomial basis, C_p/c_p grows exponentially fast in q .

Bounding ϵ_{sp}^* depends on the properties $\theta^*(\lambda)$. As a first example,

Lemma 2.4.2 (ϵ_{sp}^* for ν -Differentiable Solution Paths). *Let $\Lambda = [-1, 1]$. Suppose Assumption 2.2.1 holds. Further assume that there exists an integer $\nu \geq 0$ and constant $V > 0$ such that for all $i = 1, \dots, d$, $\theta_i^*(\cdot)$ has ν derivatives, where $\theta_i^*(\lambda), \dots, \theta_i^{(\nu-1)*}(\lambda)$ are absolutely continuous and $\theta_i^{(\nu)*}(\lambda)$ has total variation bounded by V . Then, for any $q \geq \nu + 1$, for the basis described above, $\epsilon_{\text{sp}}^* \leq \frac{dL}{2} \left(\frac{2V}{\pi\nu(q-\nu)^\nu} \right)^2$.*

The proof of Lemma 2.4.2 is constructive; we exhibit a polynomial with the given solution path error. The bound confirms the intuition that if $\theta^*(\cdot)$ is smooth (has many derivatives), then adding basis function drives down the approximation error rapidly. Problems arising in many application domains, including machine learning problems like ridge regression and relatives, often exhibit highly or even infinitely differentiable solution paths.

Combining these lemmas with Theorem 2.3.4 allows us to calculate the requisite basis size and number of iterations needed to achieve a target solution-path error:

Theorem 2.4.3 (SGD for ν -differentiable Solution Paths). *Let $\Lambda = [-1, 1]$, $\mathbb{P}_\lambda$ be the uniform distribution on $[-1, 1]$. Then, under the conditions of Lemma 2.4.2 and assume*

$\nu \geq 2$, if we use $q = O(\epsilon^{\frac{1}{2(1-\nu)}})$ polynomials in the previous basis, and run constant-step size SGD for $O(\epsilon^{\frac{1}{1-\nu}} \log(1/\epsilon))$ iterations, the resulting iterate β_T satisfies $\mathbb{E} [\epsilon_{\text{sp}}(\beta_T)] \leq \epsilon$.

For clarity, while Lemma 2.4.2 holds for any $\nu \geq 0$, Theorem 2.4.3 requires $\nu \geq 2$. Moreover, both big “Oh” terms should be interpreted as $\epsilon \rightarrow 0$, and both suppress constants not depending on ϵ (but possibly depending on h). The theorem establishes that the “smoother” $\theta^*(\cdot)$ is (i.e. larger ν), the fewer iterations required by our method to achieve a target tolerance.

Recall, [36] established that for strongly convex functions, discretization requires at least $O(\epsilon^{-1/2})$ points. One might expect the number of gradient evaluations per point to scale like $O(\log(1/\epsilon))$. Hence, if $\nu \geq 3$, our approach requires asymptotically less work. The larger ν , the larger the savings.

This gap becomes more striking as $\nu \rightarrow \infty$. Hence, we next consider the case where $\theta_i^*(\lambda)$ is analytic on $[-1, 1]$, i.e, its Taylor Series is absolutely convergent on this interval.

Lemma 2.4.4 (ϵ_{sp}^* for Analytic Solution Paths). *Suppose Assumption 2.2.1 holds and that for each $i = 1, \dots, d$, $\theta_i^*(\cdot)$ is analytic on the interval $[-1, 1]$. Then, there exist constants $\omega > 1$ and $M > 0$ such that for any $q > 0$, with the basis described above, $\epsilon_{\text{sp}}^* \leq \frac{dL}{2} \left(\frac{2M\omega^{-q}}{\omega-1} \right)^2$.*

The proof is again constructive. The constants ω and M pertain to the analytic continuation of $\theta_i^*(\cdot)$ to the complex plane. Importantly, the solution path error now dies geometrically fast (like ω^{-2q}). Using this faster decay rate yields,

Theorem 2.4.5 (SGD for Analytic Solution Paths). *Let $\Lambda = [-1, 1]$, and $\mathbb{P}_\lambda$ be the uniform distribution on $[-1, 1]$. Then, under the conditions of Lemma 2.4.4, if we use $q = \log(1/\epsilon)/\log(\omega)$ polynomials in the previous basis, and run constant-step size SGD for $O(\log^2(1/\epsilon) \log \log(1/\epsilon))$ iterations, the resulting iterate β_T satisfies $\mathbb{E} [\epsilon_{\text{sp}}(\beta_T)] \leq \epsilon$.*

Again, the big “Oh” hides constants that do not depend on ϵ . Compared to [36], the amount of work required is almost exponentially smaller for $\dim(\Lambda) = 1$.

In higher dimensions, the curse of dimensionality causes the required discretization points to grow exponentially with $\dim(\Lambda)$. In contrast, our method’s performance depends not on $\dim(\Lambda)$ but on the minimal solution path error ϵ_{sp}^* . The key advantage arises when we either select a basis ensuring a small ϵ_{sp}^* using prior knowledge or adopt a highly flexible, adaptive approach, as we next discuss in Section 2.5.

2.5 Implementation Guidelines

Although Section 2.4 provides insight on how to select a basis (in certain cases), choosing the “right” basis apriori from approximation theory remains a difficult challenge. To circumvent this issue, we re-introduce Algorithm 1. Algorithm 1 is more practical, as we only need to specify: i) a sequence of basis functions, ii) a distribution $\mathbb{P}_\lambda$ and, iii) a criterion for deciding when to add a new function to the basis. We next provide intuition and practical guidance on these choices.

Intuition for an “Ideal” Basis. Consider

$$h(\boldsymbol{\theta}, \boldsymbol{\lambda}) = \boldsymbol{\theta}^\top \mathbf{Q}(\boldsymbol{\lambda}) \boldsymbol{\theta} + \mathbf{b}(\boldsymbol{\lambda})^\top \boldsymbol{\theta}.$$

In this special case, Equation (2.2) becomes

$$\min_{\boldsymbol{\beta}} \boldsymbol{\beta}^\top \mathbb{E} [\boldsymbol{\Phi}(\tilde{\boldsymbol{\lambda}})^\top \mathbf{Q}(\tilde{\boldsymbol{\lambda}}) \boldsymbol{\Phi}(\tilde{\boldsymbol{\lambda}})] \boldsymbol{\beta} + \mathbb{E} [\mathbf{b}(\tilde{\boldsymbol{\lambda}})^\top \boldsymbol{\Phi}(\tilde{\boldsymbol{\lambda}})] \boldsymbol{\beta}.$$

The complexity of this problem depends strongly on its condition number (in our theory, this is bounded by $\frac{CL}{c\mu}$), which equals the condition number of the matrix $\mathbb{E} [\boldsymbol{\Phi}(\tilde{\boldsymbol{\lambda}})^\top \mathbf{Q}(\tilde{\boldsymbol{\lambda}}) \boldsymbol{\Phi}(\tilde{\boldsymbol{\lambda}})]$ and is directly computable in this case. Hence, an ideal basis would ensure this condition number is as close to 1 (its lower bound) as possible. In particular, if the basis functions $\boldsymbol{\Phi}_j(\cdot)$ are orthonormal with respect to the inner product

$$\langle \boldsymbol{\Phi}_j(\cdot), \boldsymbol{\Phi}_k(\cdot) \rangle \equiv \mathbb{E}_{\tilde{\boldsymbol{\lambda}} \sim \mathbb{P}_\lambda} [\boldsymbol{\Phi}_j(\tilde{\boldsymbol{\lambda}})^\top \mathbf{Q}(\tilde{\boldsymbol{\lambda}}) \boldsymbol{\Phi}_k(\tilde{\boldsymbol{\lambda}})],$$

then, the resulting condition number is 1. For general h , insofar as $F(\boldsymbol{\beta}) = \mathbb{E}_{\tilde{\boldsymbol{\lambda}} \sim \mathbb{P}_\lambda} [h(\boldsymbol{\Phi}(\tilde{\boldsymbol{\lambda}})\boldsymbol{\beta}, \tilde{\boldsymbol{\lambda}})]$ might be approximated by a second-order Taylor series expansion around the optimum $\boldsymbol{\beta}_{\text{avg}}^*$, a good basis might be orthonormal w.r.t. to the Hessian of $F(\cdot)$ at optimality. We verify this in Section 2.6.3. Absent knowledge about the Hessian at optimality, the best approach is to use a family of flexible and expressive basis functions. In Sections 2.6.1 and 2.6.2, we use orthogonal polynomials for the following reasons: i) Adding new basis functions does not require altering the existing basis, and ii) Intuitively, orthogonality suggests that if $\hat{\boldsymbol{\beta}}_{p-1}$ is near optimal in the $(p-1)^{\text{th}}$ iteration, then $[\hat{\boldsymbol{\beta}}_{p-1}, 0]$ is likely to be a near optimal (and good warm-start) solution in the p^{th} iteration. The first benefit is not shared, for example, by other approaches like cubic splines or neural networks.

Interplay Between Basis and Distribution. Although *any* sequence of basis functions and *any* distribution can be used in Algorithm 1, we suggest making these choices in concert. In our experiments, we focus on sequences of polynomials that are orthogonal with respect to $\mathbb{P}_\lambda$. The Legendre polynomials and the uniform distribution on $[-1, 1]$ is one such pair, but there are many canonical examples including Hermite polynomials with the normal distribution and Laguerre polynomials with the exponential distribution. There are performance-optimized implementations of these families in standard software (see, e.g., `scipy.special`). Although polynomials map to $\mathbb{R}$, by approximating each dimension separately as in Section 2.4, we can extend polynomials to a basis for $d > 1$. Finally, polynomials are highly expressive as they are uniform approximators for Lipschitz functions.

Deciding the Number of Basis Functions “On the Fly”. As reflected by our theoretical convergence guarantees, choosing a large, flexible basis induces a tradeoff. A larger basis increases the condition number of Equation (2.2) (as measured by the ratio $\frac{CL}{c\mu}$), and hence convergence will be slower. This observation motivates our adaptive approach that only adds basis functions as necessary until we reach a desired accuracy. One approach

is to empirically approximate the objective of Equation (2.2) using a hold-out validation set, and add a basis function when performance stalls. In our experiment with low dimensional hyperparameters (Section 2.6.1 and 6.2), we use Gauss-Legendre quadrature to evaluate the stochastic objective exactly over the entire hyperparameter space. In the high dimensional hyperparameter experiment (Section 2.6.3), we instead compute the performance of our learned β through an ERM version of the stochastic objective on a validation set of size 1000.

2.6 Numerical Experiments

We next empirically compare i) our approach using a fixed, large basis (denoted *LSP* for “Learning the Solution Path”) ii) our adaptive Algorithm 1 (denoted *ALSP* for “Adaptive LSP”) and iii) uniform discretization, a natural benchmark. We aim to show that both our approaches not only outperform the benchmark but that the qualitative insights from our theoretical results hold for more general optimization procedures than constant step-size SGD. We also provide preliminary results on how the choice of basis affects performance.

We choose uniform discretization to be our benchmark in lieu of other schemes (like geometric spacing) because i) it matches the theoretical lower bound from [36] and ii) we see it as most intuitive for learning the solution path with small *uniform* error. Specifically, for various ϵ , we consider a uniform spacing of size $\sqrt{\epsilon}$ in each dimension of λ , and run (warm-started) gradient descent for $O(\log(1/\epsilon))$ iterations. The constant hidden by big “Oh” here is calibrated in an oracle fashion to achieve a solution-path error of $O(\epsilon)$ (see Section A.2.1 for details) giving the benchmark a small advantage.

2.6.1 Weighted Binary Classification

We consider a binary classification problem using a randomly selected subset of 1000 cases from the highly imbalanced Law School Admission Bar Passage dataset [54]. Of the 1000 cases, there are 956 positive instances and 44 negatives. Standard logistic regression predicts 992 positives with a false positive rate of 0.86. When identifying students likely to fail is key, the default classifier may not be useful. Reweighting cases is a standard approach to improve false positive rate at the cost of overall accuracy.

We take

$$h(\theta, \lambda) = (1 - \lambda)l_{\text{pos}}(\theta) + \lambda l_{\text{neg}}(\theta) + 0.125\|\theta\|^2,$$

where $l_{\text{pos}}(\theta)$ and $l_{\text{neg}}(\theta)$ denote the negative log-likelihood on the positive and negative classes respectively. Specifically, letting $(\mathbf{x}_i, y_i) \in \mathbb{R}^{45} \times \{0, 1\}$ for $i = 1, \dots, n$ denote the data,

$$l_{\text{pos}}(\theta) = \frac{1}{|\{i : y_i = 1\}|} \sum_{i: y_i=1} \log(1 + e^{(-2y_i+1)\mathbf{x}_i^\top \theta}),$$

and similarly for $l_{\text{neg}}(\theta)$.

We consider two different choices of distribution for $\lambda \in [0, 1]$ together with two different orthogonal polynomial bases:

- i) A $\text{Unif}[0, 1]$ distribution. Here we use (scaled and shifted) Legendre polynomials.
- ii) A $\text{Beta}(b + 1, a + 1)$ distribution. Here we use (shifted) Jacobi polynomials with parameters $(a, b) = (-.3, -.7)$.

The ground truth $\theta^*(\lambda)$ is computed via 5000 iterations of (warm-started) gradient descent over a uniform grid of 2^{10} points. Solution path error is approximated by the uniform error over this grid.

For LSP, we run SGD using `torch.optim.SGD`. In lieu of a constant learning rate, we dynamically reduce the learning rate according to the Distance Diagnostic of [46, Section 4]. This dynamic updating is more reflective of practice. We use the suggested parameters from [46] with the exception of q^1 , which we tune by examining the performance after 200 iterations. (We take $q = 1.3$ for both the Legendre and Jacobi bases.) For ALSP, we initialize the algorithm with 5 polynomials and stop it after reaching 12 polynomials ($q = 5$ to 12).

Figure 2.1 in the introduction and Figure 2.2 panel (a) show the performance for the Legendre and Jacobi bases respectively. As we can see, LSP methods converge rapidly to an irreducible error, and this error decreases as we add basis functions. Our adaptive method converges rapidly. This performance is resonated by Figure 2.2 panels (c) and (d), where we plot the coefficient profile for $\theta_3^*(\lambda)$ when using 7 Legendre and Jacobi polynomials. The approximation improves as we increase the number of iterations, and eventually become closely aligned with the true solution path.

Figure 2.2 panel (b) shows that the largest errors occur almost periodically at a few places across $\Lambda = [0, 1]$, confirming the equioscillation theorem.

Practically, this experiment suggests that when the region of Λ is compact, taking a polynomial basis paired with a distribution supported everywhere over Λ is favorable. The polynomial basis can then be taken to be orthogonal to $\mathbb{P}_\lambda$ as discussed in Section 2.5.

2.6.2 Portfolio Allocation

We next consider a portfolio allocation problem calibrated to real data where $\theta \in \mathbb{R}^d$ represents the weights on $d = 10$ different asset classes. Namely, let $\mu \in \mathbb{R}^d$ and $\Sigma \in \mathbb{R}^{d \times d}$ be the mean and covariance matrix of the returns of the different asset classes. We fit these parameters to the monthly return data from Aug. 2014 to July 2024 using the Fama-French 10 Industry index dataset.²

We then solve

$$\min_{\theta} \quad -\lambda_1 \mu^\top \theta + \lambda_2 \theta^\top \Sigma \theta + \bar{\ell}_1(\theta).$$

¹Note that q , defined as a diagnostic frequency control parameter in [46, Section 4], is different from q , the number of polynomials in our paper.

²https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html

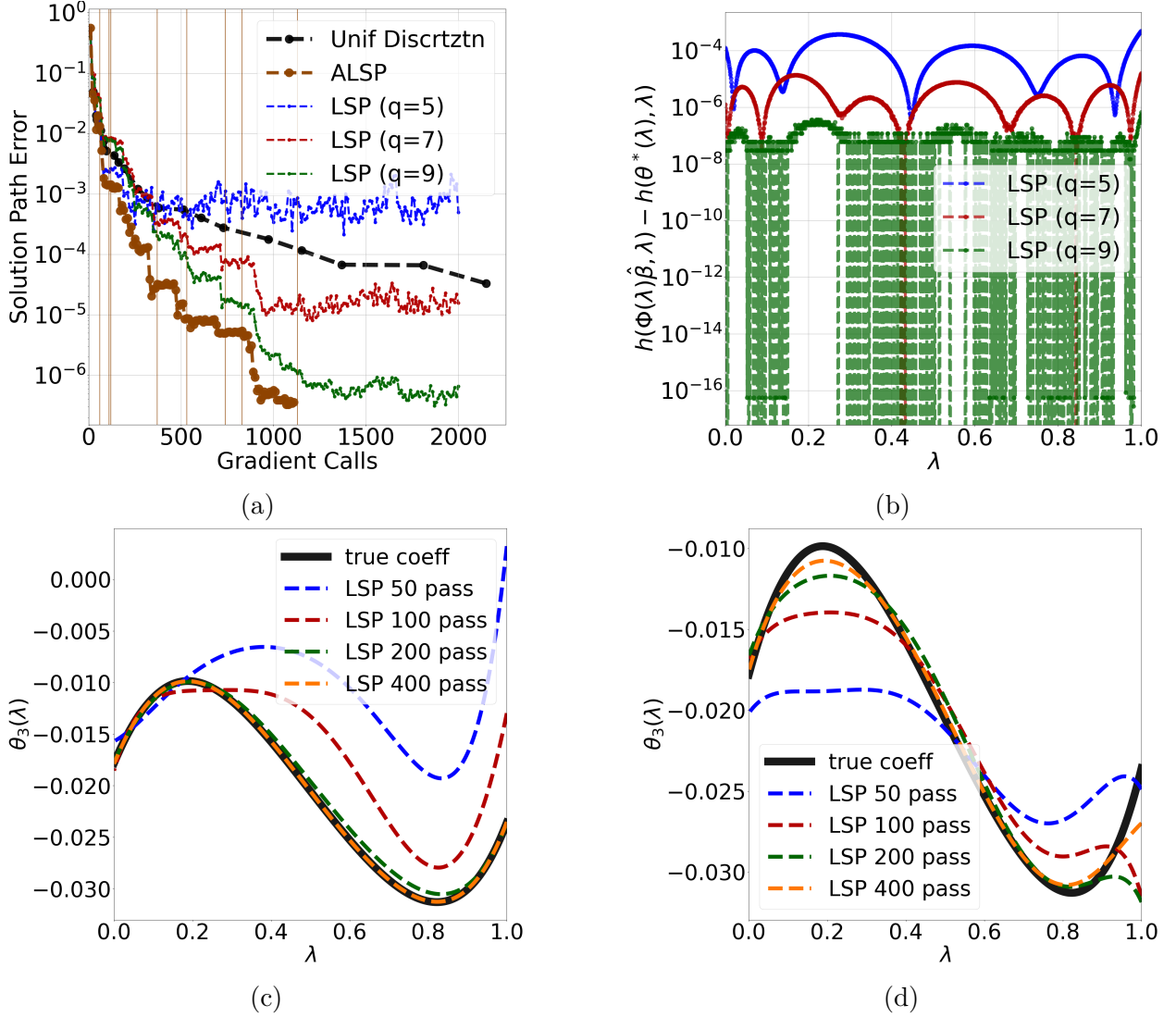


Figure 2.2: **LSP and ALSP for Weighted Binary Classification.** (a) Compares methods using Jacobi polynomials. (b) Error in solution path as a function of λ using Jacobi polynomials. (c) and (d) compare $\theta_3(\lambda)$ across λ in the Legendre and Jacobi bases respectively with 7 polynomials.

Here the first term represents the (negative) expected return, the second represents the risk (variance) and the third is a smoothed version ℓ_1 regularization to induce sparsity. Namely, $\bar{\ell}_1(\theta) := \sum_{i=1}^d \sqrt{\theta_i^2 + .01^2} - .01$. The parameters $\lambda_1 \in [0, 1]$ and $\lambda_2 \in [0.2, 1]$ control the tradeoffs in these multiple objectives.

Ground truth $\theta^*(\lambda)$ is computed over a 100×100 grid.

We focus on bivariate-Legendre polynomials for our basis, scaled and shifted to be orthogonal to uniform distribution on $[0, 1] \times [.2, 1]$. We initialize with ALSP with 2 polyno-

mials in each dimension, and iterate by adding one polynomial in each dimension, so that $q = 4, 9, 16, 25$, after which we stop.

Unlike our previous experiment, to showcase that the qualitative insights of our theory hold for other algorithms other than SGD, we use `torch.optim.LBFGS` for LSP, ALSP and uniform discretization (for a fair comparison). Unlike SGD, L-BFGS uses both function and gradient evaluations. We restrict it to use only 10 function calls per gradient step so that the total work is still proportional to the number of gradient calls.

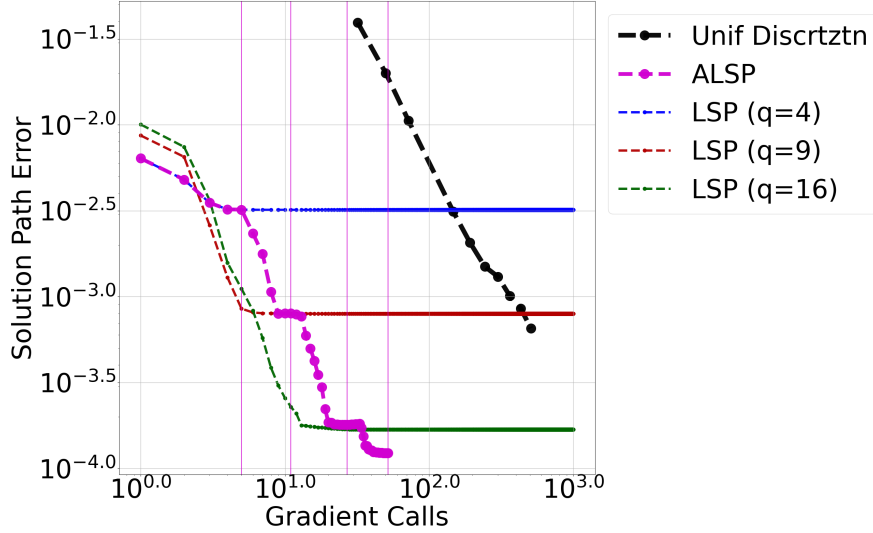


Figure 2.3: **LSP and ALSP for Portfolio Allocation.** Compares methods using the first $q = 4, 9, 16$ bivariate-Legendre polynomials.

As in our previous experiment, we see that LSP rapidly converges to an irreducible error and then plateaus. By contrast, ALSP seems to make continued progress as we add polynomials. Both methods substantively outperform uniform discretization.

The improved performance over uniform discretization is partially attributable to the increased dimension of Λ (because discretization suffers from the curse of dimensionality), but is also because (traditionally) discretization interpolates solutions in a piecewise constant fashion. In this example, $\theta^*(\lambda)$ is very smooth. See Figure A.2 in Appendix. Hence, polynomial can learn to interpolate values very fast, while uniform discretization needs a great deal more resolution.

2.6.3 Portfolio Allocation with Moderate Dimensional Λ

We next modify the portfolio allocation problem in Section 2.6.2 to include transaction costs relative to the current portfolio as follows:

$$h(\theta, \lambda) = -\lambda_1 \cdot \mu^\top \theta + \lambda_2 \cdot \theta^\top \Sigma \theta + \|\theta - \lambda_{3:12}\|_2^2.$$

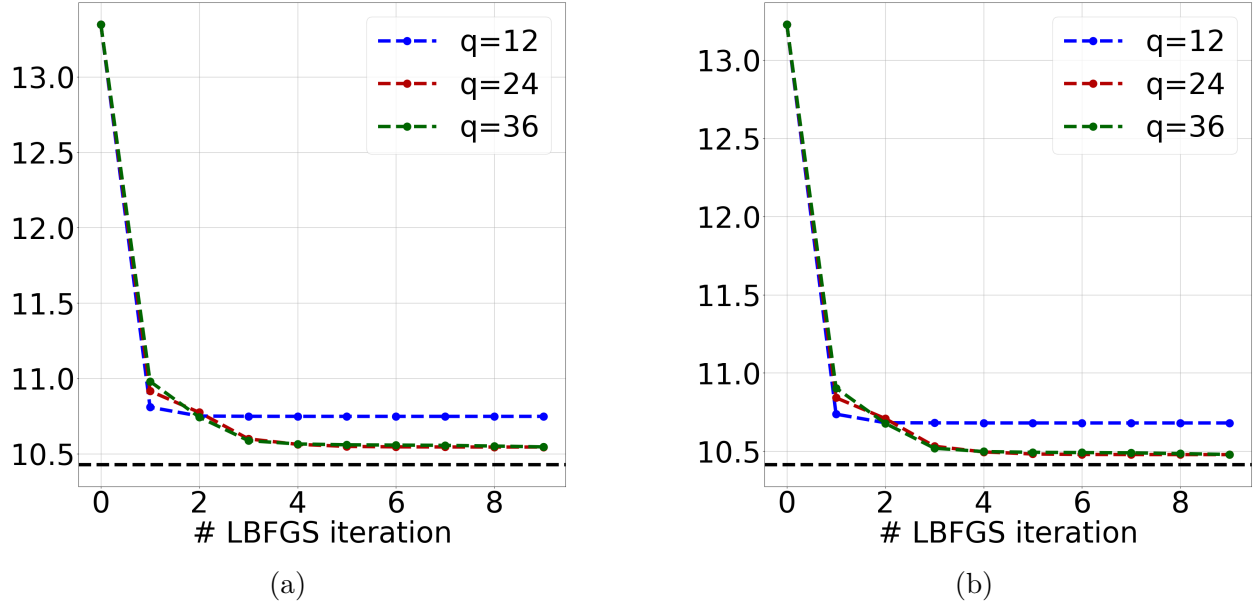


Figure 2.4: **Portfolio Allocation**, $\dim(\Lambda) = 10$. Comparing the ERM objective for varying q . Panel a) training, Panel b) validation.

Here $\lambda_{3:12} \in \mathbb{R}^{10}$ represents the current portfolio holding, and $\lambda \in \mathbb{R}^{12}$. We use the following basis

$$\lambda_{(j \bmod 12)} \cdot \lambda_2^{\lfloor j/12 \rfloor}, \quad j = 1, \dots, q,$$

for each component of $\hat{\theta}_i(\cdot)$, for $i = 1, \dots, 10$, and let $q = 12, 24, 36$ in our experiments. As $q \rightarrow \infty$, this basis contains the optimal solution path in its span for any choice of μ, Σ (c.f. Section A.2.2).

For simplicity, we ran LSP using `torch.optim.LBFGS` to solve an ERM version of the problem in Step 6 of Algorithm 1 on a training set of size 1000, and present results for an independent validation set of size 1000.

Discretization on a 12-dimensional grid is computationally challenging. Hence, to assess our method we instead compute an ERM approximation to $\mathbb{E} [h(\theta^*(\tilde{\lambda}), \tilde{\lambda})]$ over the training/validation sets, i.e., compute $\frac{1}{1000} \sum_{i=1}^{1000} h(\theta^*(\lambda_i), \lambda_i)$ by solving 1000 separate optimization problems. This value is the dotted grey line in Figure 2.4. The smaller the gap, the closer $\theta^*(\lambda_i)$ is to the span of the basis.

Similar to previous experiments, as the basis size increases, performance improves, and even with a fairly small basis, the error is quite small.

2.7 Conclusion

We propose a new method for learning the optimal solution path of a family of problems by reframing it as a single stochastic optimization problem over a linear combination of pre-specified basis functions. Compared to discretization schemes, our method offers flexibility and scalability by taking a global perspective on the solution path. We prove that our problem satisfies a certain relaxed weak-growth condition that allows us to solve the single optimization problem very efficiently when using sufficiently rich bases. Theoretical results in special cases and numerical experiments in more general settings support these findings. Future research might more carefully examine the interplay between the parameterization of the family of problems and the choice of basis. One might also consider other universal function approximators (e.g. deep neural networks, forests of trees) within this context.

Chapter 3

Learning Optimization Proxies under General Convexity

The previous chapter developed the dissertation’s core construction in the language of solution paths: a parametric optimizer is approximated by a learned map $\lambda \mapsto \Phi(\lambda)\beta$, trained through a single stochastic optimization problem. This chapter takes the next step and places the same construction in the broader setting of optimization proxies. The shift in language is important because the learned object need not be understood as tracing a path; it is a fast decision rule for a family of optimization instances.

The purpose of this chapter is to relax and generalize the first theory. We introduce complexity-penalized proxy error, allow the ridge penalty to stabilize cases where the original strong convexity or spectral lower-bound assumptions may fail, and connect in-distribution training progress to both average-case and uniform out-of-distribution proxy performance. The average-case guarantee is especially aligned with how optimization proxies are often evaluated: train on representative problem instances and measure expected decision quality on future instances from the same task distribution. We also move beyond Euclidean geometry by analyzing stochastic reference function descent, which makes the proxy-learning framework compatible with domains such as simplices and positive orthants.

In the development of the dissertation, this chapter turns the initial solution-path method into a general theory of learning optimization proxies. It identifies approximation error as the quantity governing the limiting performance of the trained proxy, while showing how regularization and geometry shape the speed of convergence.

3.1 Introduction

Many modern decision systems solve a related optimization problem repeatedly as a problem instance changes. In power systems, the instance may be a vector of loads or contingencies; in resource allocation, it may be a demand forecast or a vector of task weights; in statistical learning, it may be a regularization parameter or a user-specified tradeoff. If the instance is

denoted by $\lambda \in \Lambda$, the optimizer defines a solution map

$$\lambda \mapsto \theta^*(\lambda).$$

Classical parametric optimization methods approximate this map by repeatedly solving the optimization problem at selected instances and interpolating or following the solution as λ varies. Chapter 1 studied this perspective under the name of learning the optimal solution path. In this chapter, we adopt the language of optimization proxies: rather than solving a new optimization problem every time a new instance arrives, we train a model $\lambda \mapsto \Phi(\lambda)\beta$ that can be evaluated directly as a fast approximate optimizer.

This proxy viewpoint has appeared in several forms across machine learning and optimization. In power systems, learned approximations to optimal power flow replace expensive repeated solves with neural mappings from loads to dispatch decisions, often with reconstruction, repair, or feasibility-enforcement steps [42, 17, 44]. More generally, recent work on constrained optimization learning studies neural approximators that output feasible or near-feasible decisions for hard constrained problems [14, 31]. Van Hentenryck frames this broader program as optimization learning, where the learned object is an optimization proxy for a family of parametric optimization problems [52]; see also the survey of end-to-end constrained optimization learning in [28].

Optimization proxies are adjacent to, but distinct from, predict-then-optimize and differentiable optimization layer approaches. In predict-then-optimize, a learning model predicts uncertain problem data and a downstream solver still computes the decision; decision-focused losses train the prediction model according to the quality of the resulting decision [16, 13]. Differentiable optimization layers similarly keep an optimization problem inside the computational graph, allowing gradients to pass through a solver by implicit differentiation or related techniques [2, 1]. These approaches are powerful when a solver remains part of the prediction pipeline. By contrast, the proxy considered here is the decision rule itself: after training, the map $\lambda \mapsto \Phi(\lambda)\beta$ is evaluated in place of a fresh optimization solve.

Despite the substantial empirical literature, the training dynamics of optimization proxies remain less developed theoretically. A central difficulty is that a proxy is trained by minimizing an expected stochastic objective over sampled instances, while the desired guarantee often concerns its performance uniformly over Λ , including instances not emphasized by the training distribution. Another difficulty is the interaction between expressiveness and optimization: a richer basis can reduce approximation error, but it can also worsen conditioning and slow down first-order training. This chapter develops a convergence theory that makes this tradeoff explicit and, to our knowledge, gives the first general theoretical framing of optimization-proxy training through this approximation–optimization lens. We show that the stochastic objective used to train the proxy satisfies a relaxed weak growth condition, so constant step-size stochastic gradient methods converge rapidly to a neighborhood whose size is governed by the best-in-class proxy error. We then show how the same approximation error controls uniform proxy performance, thereby connecting in-distribution training progress to out-of-distribution instance-wise guarantees.

The chapter also broadens the geometry of Chapter 1. The Euclidean theory is appropriate for unconstrained or simply constrained decisions, but many optimization proxies must respect domains such as positive orthants or simplices. For these settings, Euclidean smoothness can be too restrictive or badly conditioned. We therefore extend the analysis to stochastic reference function descent, or stochastic mirror descent, under relative smoothness and relative strong convexity. This Bregman geometry lets the training algorithm align with the feasible domain and provides a common framework for the Poisson inverse and multitask optimal-design examples studied later in the chapter.

3.1.1 Our Contributions

We summarize our contributions as follows:

- We formulate the training of an optimization proxy as a single stochastic optimization problem over the coefficient vector β . Under the Euclidean assumptions of Assumptions 3.2.1 and 3.2.2, we show that this stochastic objective satisfies a relaxed weak growth condition. Consequently, constant step-size SGD converges linearly to a neighborhood controlled by the best-in-class proxy error $\epsilon_{\text{sp},\gamma}^*$. The result formalizes a basic approximation–optimization tradeoff: increasing the expressive power of the proxy can reduce the limiting error, while the associated conditioning constants can slow the transient convergence.
- We prove a proxy-performance decomposition that connects the in-distribution stochastic training objective to uniform performance over Λ . This gives a robustness guarantee for optimization proxies: the worst-case optimality gap of the learned decision rule is controlled by optimization suboptimality together with the approximation error of the model class. In particular, SGD convergence for the stochastic objective implies convergence of the expected uniform proxy error.
- We specialize the theory to periodic one-dimensional parameter domains using real Fourier features. Under differentiability or analyticity assumptions on the solution map, classical approximation theory bounds ϵ_{sp}^* and yields explicit basis-size and iteration-complexity guarantees. These results illustrate how smoothness of the underlying solution map translates into learnability of the proxy.
- We extend the analysis from Euclidean geometry to general Bregman geometry. For stochastic reference function descent, we establish convergence under a generalized relaxed weak growth condition and combine this convergence with a local proxy-performance decomposition. This allows the proxy-training theory to cover non-Euclidean domains and reference functions, including the log-barrier geometries used in the Poisson inverse and multitask D-optimal design examples.
- We provide numerical experiments for both Euclidean and Bregman settings. The experiments show the predicted effect of the regularization parameter γ : larger values

improve conditioning and accelerate early progress, while smaller values allow the trained proxy to reach lower final objective values once the optimization transient has passed.

3.2 Preliminaries

Let Λ denote the index set of problem instances. For each $\lambda \in \Lambda$, consider an objective function $\boldsymbol{\theta} \mapsto h(\boldsymbol{\theta}, \lambda)$. We aim to find the family of *parametrized* optimizers

$$\boldsymbol{\theta}^*(\lambda) := \arg \min_{\boldsymbol{\theta}} h(\boldsymbol{\theta}, \lambda), \quad \lambda \in \Lambda. \quad (3.1)$$

To this end, we propose replacing the family of parametrized objectives in (3.1) with a *single* stochastic optimization problem.

Specifically, we consider parametric solution-map approximations of the form

$$\boldsymbol{\theta}^*(\lambda) \approx \boldsymbol{\Phi}(\lambda)\boldsymbol{\beta}, \quad \lambda \in \Lambda,$$

where $\boldsymbol{\Phi}(\lambda) \in \mathbb{R}^{d \times p}$ is a feature map, and $\boldsymbol{\beta} \in \mathbb{R}^p$ is a parameter vector, obtained by approximately solving

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^p} \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} [h(\boldsymbol{\Phi}(\tilde{\lambda})\boldsymbol{\beta}, \tilde{\lambda})] + \frac{\gamma}{2} \|\boldsymbol{\beta}\|^2, \quad (3.2)$$

where $\mathbb{P}_\lambda$ is a probability distribution over Λ .

Both $\boldsymbol{\Phi}$ and $\mathbb{P}_\lambda$ are user-specified components.

In contrast to the original formulation in [12], the above formulation in (3.2) introduces a ridge-regularized variant of the proxy error definition. This modification removes the need for strict positivity of the strong convexity constant, which is an essential obstruction in infinite-dimensional settings.

We use $\|\cdot\|$ to denote the ℓ_2 -norm throughout.

We also make the following assumptions on $h(\cdot, \lambda)$ and the spectral bound of the prescribed basis $\boldsymbol{\Phi}(\lambda)$:

Assumption 3.2.1 ([12], Assumption 2.1). There are constants $0 \leq \mu \leq L$ such that, for all $\lambda \in \Lambda$, $h(\cdot, \lambda)$ is μ -strongly convex and L -smooth, i.e., for all $\boldsymbol{\theta}, \bar{\boldsymbol{\theta}} \in \mathbb{R}^d$,

$$\frac{\mu}{2} \|\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}\|^2 \leq h(\boldsymbol{\theta}, \lambda) - h(\bar{\boldsymbol{\theta}}, \lambda) - \nabla h(\bar{\boldsymbol{\theta}}, \lambda)^\top (\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}) \leq \frac{L}{2} \|\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}\|^2.$$

Here, we make a relaxation on the strict positivity condition of μ .

Assumption 3.2.2 (Uniform Spectral Bound). There exists constant $C \geq 0$ defined as

$$C := \sup_{\lambda \in \Lambda, \|\mathbf{v}\|=1} \|\Phi(\lambda)\mathbf{v}\|^2,$$

C is the uniform bound on the largest eigenvalue of the positive semidefinite matrix $\Phi(\lambda)^\top \Phi(\lambda)$. We also defined a lower bound on an expected operator norm

$$c := \inf_{\|\mathbf{v}\|=1} \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} \left[\|\Phi(\tilde{\lambda})\mathbf{v}\|^2 \right], \quad (3.3)$$

c is the smallest eigenvalue of the expected matrix $\mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} [\Phi(\lambda)^\top \Phi(\lambda)]$. By construction, $0 \leq c \leq C$.

Remark 3.2.3 (Relaxation of Strong Convexity Constant and Spectral Lower Bound). *Because of guaranteed strong convexity induced by the positive complexity-penalty parameter γ , we do not require μ or c to be strictly positive for proxy-performance decomposition or SGD convergence in Sections 2.1 and 2.2. However, the positivity of c is essential for increasing basis expressiveness when we expand it. We will see this in the specialized results analysis in Section 3.*

3.2.1 Complexity-Penalized Proxy Error

Fix $\gamma > 0$, feature map Φ , and probability distribution $\mathbb{P}_\lambda$. Define the complexity-penalized proxy error over the parameter (β) space as

$$\epsilon_{\text{sp},\gamma}(\beta) := \sup_{\lambda \in \Lambda} \{h(\Phi(\lambda)\beta, \lambda) - h(\theta^*(\lambda), \lambda)\} + \frac{\gamma}{2}\|\beta\|^2.$$

The best-in-class proxy error and the corresponding parameter vector are defined as

$$\begin{aligned} \epsilon_{\text{sp},\gamma}^* &:= \inf_{\beta \in \mathbb{R}^p} \epsilon_{\text{sp},\gamma}(\beta), \\ \beta_{\text{sp},\gamma}^* &\in \arg \min_{\beta \in \mathbb{R}^p} \epsilon_{\text{sp},\gamma}(\beta). \end{aligned}$$

$\epsilon_{\text{sp},\gamma}^*$ characterizes the best achievable uniform approximation to the solution map when complexity is controlled via a penalty rather than an explicit constraint.

We also define the following auxiliary ridge-regularized objectives

$$\begin{aligned} f_\gamma(\beta, \lambda) &:= h(\Phi(\lambda)\beta, \lambda) + \frac{\gamma}{2}\|\beta\|^2, \\ F_\gamma(\beta) &:= \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} [f_\gamma(\beta, \tilde{\lambda})] \\ &= \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} [h(\Phi(\tilde{\lambda})\beta, \tilde{\lambda})] + \frac{\gamma}{2}\|\beta\|^2. \end{aligned} \quad (3.4)$$

$F_\gamma(\cdot)$ is the stochastic objective appeared in Problem (3.2), which can be solved with a user-specified stochastic optimization algorithm. Further define $f(\cdot, \lambda)$, $F(\cdot)$ as the unregularized counterparts of (3.4) when $\gamma = 0$.

Define the solution of Problem (3.2) as

$$\beta_{\text{avg}, \gamma}^* \in \arg \min_{\beta \in \mathbb{R}^p} F_\gamma(\beta).$$

Remark 3.2.4 (Reduction to the unregularized setting). *When $\gamma = 0$, the complexity-penalized proxy error reduces to the original (unregularized) solution-path error studied in our prior work, and $\beta_{\text{avg}, \gamma}^*$, $\beta_{\text{sp}, \gamma}^*$ reduce to their unregularized counterparts. Thus, as γ is driven down to 0, $\Phi(\lambda)\beta_{\text{avg}, \gamma}^*$ provides an increasingly better approximation of $\theta^*(\lambda)$.*

Proposition 3.2.5 (Uniform Strong Convexity and Smoothness of Ridge-Regularized Objective). *Under Assumptions 3.2.1 and 3.2.2, $f_\gamma(\cdot, \lambda)$ is uniformly $(CL + \gamma)$ -smooth for all $\lambda \in \Lambda$. Furthermore, $F_\gamma(\cdot)$ is $(CL + \gamma)$ -smooth and $(c\mu + \gamma)$ -strongly convex.*

The proof is an simple extension of [12, Proposition A.1] and thus omitted here.

3.2.2 Proxy-Performance Decomposition

We develop the proxy-performance decomposition theorem, generalizing [12, Theorem 2.4]. The bound applies to both unpenalized and penalized supremum proxy error, and the penalty enters only through the constants.

Theorem 3.2.6 (Uniform Proxy Performance Bound). *Suppose Assumptions 3.2.1 and 3.2.2 hold. For all $\beta \in \mathbb{R}^p$,*

1. *when $c\mu > 0$,*

$$\epsilon_{\text{sp}}(\beta) \leq \frac{3CL}{2} \|\beta - \beta_{\text{avg}, \gamma}^*\|^2 + \left(\frac{3L}{\mu} + \frac{3CL}{c\mu + \gamma} \right) \cdot \epsilon_{\text{sp}, \gamma}^*.$$

2. *when $c\mu = 0$,*

$$\epsilon_{\text{sp}}(\beta) \leq \epsilon_{\text{sp}, \gamma}(\beta) \leq \frac{3(CL + \gamma)}{2} \|\beta - \beta_{\text{avg}, \gamma}^*\|^2 + \left(\frac{6(CL + \gamma)}{\gamma} + 1 \right) \cdot \epsilon_{\text{sp}, \gamma}^*.$$

Proof. Proof of Part (1).

Fix $\lambda \in \Lambda$. By L -smoothness of $h(\cdot, \lambda)$,

$$h(\Phi(\lambda)\beta, \lambda) - h(\theta^*(\lambda), \lambda) \leq \frac{L}{2} \|\Phi(\lambda)\beta - \theta^*(\lambda)\|^2. \quad (3.5)$$

Add and subtract $\Phi(\lambda)\beta_{\text{avg},\gamma}^*$ and $\Phi(\lambda)\beta_{\text{sp},\gamma}^*$, and use $(a+b+c)^2 \leq 3(a^2+b^2+c^2)$ to obtain

$$\begin{aligned} \|\Phi(\lambda)\beta - \theta^*(\lambda)\|^2 &\leq 3\|\Phi(\lambda)(\beta - \beta_{\text{avg},\gamma}^*)\|^2 \\ &\quad + 3\|\Phi(\lambda)(\beta_{\text{avg},\gamma}^* - \beta_{\text{sp},\gamma}^*)\|^2 + 3\|\Phi(\lambda)\beta_{\text{sp},\gamma}^* - \theta^*(\lambda)\|^2. \end{aligned} \quad (3.6)$$

By Assumption 3.2.2, the first and second term transform to

$$\|\Phi(\lambda)(\beta - \beta_{\text{avg},\gamma}^*)\|^2 \leq C\|\beta - \beta_{\text{avg},\gamma}^*\|^2,$$

$$\|\Phi(\lambda)(\beta_{\text{avg},\gamma}^* - \beta_{\text{sp},\gamma}^*)\|^2 \leq C\|\beta_{\text{avg},\gamma}^* - \beta_{\text{sp},\gamma}^*\|^2.$$

Since F_γ is $(c\mu + \gamma)$ -strongly convex and minimized at $\beta_{\text{avg},\gamma}^*$,

$$\|\beta_{\text{avg},\gamma}^* - \beta_{\text{sp},\gamma}^*\|^2 \leq \frac{2}{c\mu + \gamma} (F_\gamma(\beta_{\text{sp},\gamma}^*) - F_\gamma(\beta_{\text{avg},\gamma}^*)).$$

Moreover,

$$\begin{aligned} &F_\gamma(\beta_{\text{sp},\gamma}^*) - F_\gamma(\beta_{\text{avg},\gamma}^*) \\ &\leq F_\gamma(\beta_{\text{sp},\gamma}^*) - \mathbb{E} \left[h(\theta^*(\tilde{\lambda}), \tilde{\lambda}) + \frac{\gamma}{2} \|\beta_{\text{avg},\gamma}^*\|^2 \right] \quad (\text{Optimality of } \theta^*(\cdot)) \\ &\leq F_\gamma(\beta_{\text{sp},\gamma}^*) - \mathbb{E} \left[h(\theta^*(\tilde{\lambda}), \tilde{\lambda}) \right] \quad (\text{Non-negativity of } \frac{\gamma}{2} \|\beta_{\text{avg},\gamma}^*\|^2) \\ &\leq \epsilon_{\text{sp},\gamma}^*. \quad (\text{Definition of } \epsilon_{\text{sp},\gamma}^*(\cdot)) \end{aligned}$$

Combining the above two inequalities yields

$$\|\beta_{\text{avg},\gamma}^* - \beta_{\text{sp},\gamma}^*\|^2 \leq \frac{2}{c\mu + \gamma} \cdot \epsilon_{\text{sp},\gamma}^*.$$

Finally, for the last term, μ -strong convexity of $h(\cdot, \lambda)$ gives

$$\|\Phi(\lambda)\beta_{\text{sp},\gamma}^* - \theta^*(\lambda)\|^2 \leq \frac{2}{\mu} (h(\Phi(\lambda)\beta_{\text{sp},\gamma}^*, \lambda) - h(\theta^*(\lambda), \lambda)) \leq \frac{2}{\mu} \epsilon_{\text{sp},\gamma}^*.$$

Combined. Substituting these bounds into Equations (3.5) and (3.6),

$$h(\beta, \lambda) - h(\theta^*(\lambda), \lambda) \leq \frac{3CL}{2} \|\beta - \beta_{\text{avg},\gamma}^*\|^2 + \left(\frac{3L}{\mu} + \frac{3CL}{c\mu + \gamma} \right) \cdot \epsilon_{\text{sp},\gamma}^*.$$

Taking the supremum over $\lambda \in \Lambda$ yields the final result stated in the theorem. $\square$

Proof. Proof of Part (2) When $h(\cdot, \lambda)$ is not strongly convex, we instead rely on the strong convexity of $f_\gamma(\cdot, \lambda)$ provided by the regularization term. The proof works in the β -space directly and follows a similar strategy as Part (1). The only difference is that we first translate the pointwise optimality condition of $h(\cdot, \lambda)$ to a pointwise near-optimality condition of $f_\gamma(\cdot, \lambda)$, and then replace the strong convexity condition from h to f_γ where applied.

Define an auxiliary term

$$\beta_\gamma^*(\lambda) := \arg \min_{\beta \in \mathbb{R}^p} f_\gamma(\beta, \lambda).$$

Then, on every $\lambda \in \Lambda$, $\beta_\gamma^*(\lambda)$ beats $\beta_{\text{sp}, \gamma}^*$, yielding

$$\begin{aligned} f_\gamma(\beta_\gamma^*(\lambda), \lambda) - h(\theta^*(\lambda), \lambda) \\ &\leq f_\gamma(\beta_{\text{sp}, \gamma}^*, \lambda) - h(\theta^*(\lambda), \lambda) \\ &\leq \epsilon_{\text{sp}, \gamma}^*. \end{aligned}$$

Fix $\lambda \in \Lambda$. Then,

$$\begin{aligned} f_\gamma(\beta, \lambda) - h(\theta^*(\lambda), \lambda) \\ &= f_\gamma(\beta, \lambda) - f_\gamma(\beta_\gamma^*(\lambda), \lambda) + f_\gamma(\beta_\gamma^*(\lambda), \lambda) - h(\theta^*(\lambda), \lambda) \\ &\leq f_\gamma(\beta, \lambda) - f_\gamma(\beta_\gamma^*(\lambda), \lambda) + \epsilon_{\text{sp}, \gamma}^*. \end{aligned} \tag{3.7}$$

We are left to deal with the suboptimality in $f_\gamma(\cdot, \lambda)$. The following proof steps use Proposition 3.2.5.

Fix $\lambda \in \Lambda$. By $(CL + \gamma)$ -smoothness of $f_\gamma(\cdot, \lambda)$ and optimality of $\beta_\gamma^*(\lambda)$,

$$f_\gamma(\beta, \lambda) - f_\gamma(\beta_\gamma^*(\lambda), \lambda) \leq \frac{CL + \gamma}{2} \|\beta - \beta_\gamma^*(\lambda)\|^2. \tag{3.8}$$

Add and subtract $\beta_{\text{avg}, \gamma}^*$ and $\beta_{\text{sp}, \gamma}^*$, and use $(a + b + c)^2 \leq 3(a^2 + b^2 + c^2)$ to obtain

$$\|\beta - \beta_\gamma^*(\lambda)\|^2 \leq 3\|\beta - \beta_{\text{avg}, \gamma}^*\|^2 + 3\|\beta_{\text{avg}, \gamma}^* - \beta_{\text{sp}, \gamma}^*\|^2 + 3\|\beta_{\text{sp}, \gamma}^* - \beta_\gamma^*(\lambda)\|^2. \tag{3.9}$$

We deal with the second and last term separately.

Second Term. Since F_γ is γ -strongly convex ($c\mu = 0$) and minimized at $\beta_{\text{avg}, \gamma}^*$,

$$\|\beta_{\text{avg}, \gamma}^* - \beta_{\text{sp}, \gamma}^*\|^2 \leq \frac{2}{\gamma} (F_\gamma(\beta_{\text{sp}, \gamma}^*) - F_\gamma(\beta_{\text{avg}, \gamma}^*)).$$

Moreover,

$$\begin{aligned}
F_\gamma(\beta_{\text{sp},\gamma}^*) - F_\gamma(\beta_{\text{avg},\gamma}^*) & \leq F_\gamma(\beta_{\text{sp},\gamma}^*) - \mathbb{E} \left[h(\theta^*(\tilde{\lambda}), \tilde{\lambda}) + \frac{\gamma}{2} \|\beta_{\text{avg},\gamma}^*\|^2 \right] && \text{(Optimality of } \theta^*(\cdot) \text{)} \\
& \leq F_\gamma(\beta_{\text{sp},\gamma}^*) - \mathbb{E} \left[h(\theta^*(\tilde{\lambda}), \tilde{\lambda}) \right] && \text{(Non-negativity of } \frac{\gamma}{2} \|\beta_{\text{avg},\gamma}^*\|^2 \text{)} \\
& \leq \epsilon_{\text{sp},\gamma}^* && \text{(Definition of } \epsilon_{\text{sp},\gamma}(\cdot) \text{)}
\end{aligned}$$

Combine the above two inequalities yields

$$\|\beta_{\text{avg},\gamma}^* - \beta_{\text{sp},\gamma}^*\|^2 \leq \frac{2}{\gamma} \cdot \epsilon_{\text{sp},\gamma}^*.$$

Last Term. By γ -strong convexity of $f_\gamma(\cdot)$,

$$\begin{aligned}
\|\beta_{\text{sp},\gamma}^* - \beta_\gamma^*(\lambda)\|^2 & \leq \frac{2}{\gamma} (f_\gamma(\beta_{\text{sp},\gamma}^*) - f_\gamma(\beta_\gamma^*(\lambda))) \\
& \leq \frac{2}{\gamma} (f_\gamma(\beta_{\text{sp},\gamma}^*) - h(\theta^*(\lambda), \lambda) - \frac{\gamma}{2} \|\beta_\gamma^*(\lambda)\|^2) && \text{(Optimality of } \theta^*(\cdot) \text{)} \\
& \leq \frac{2}{\gamma} (f_\gamma(\beta_{\text{sp},\gamma}^*) - h(\theta^*(\lambda), \lambda)) && \text{(Non-negativity of } \frac{\gamma}{2} \|\beta_{\text{avg},\gamma}^*\|^2 \text{)} \\
& \leq \frac{2}{\gamma} \cdot \epsilon_{\text{sp},\gamma}^* && \text{(Definition of } \epsilon_{\text{sp},\gamma}(\cdot) \text{)}
\end{aligned}$$

Combined. Combine Equations (3.8) and (3.9), and the analysis above,

$$f_\gamma(\beta, \lambda) - f_\gamma(\beta_\gamma^*(\lambda), \lambda) \leq \frac{3(CL + \gamma)}{2} \|\beta - \beta_{\text{avg},\gamma}^*\|^2 + \frac{6(CL + \gamma)}{\gamma} \cdot \epsilon_{\text{sp},\gamma}^*.$$

Insert the above into Equation (3.7), yielding

$$f_\gamma(\beta, \lambda) - h(\theta^*(\lambda), \lambda) \leq \frac{3(CL + \gamma)}{2} \|\beta - \beta_{\text{avg},\gamma}^*\|^2 + \left(\frac{6(CL + \gamma)}{\gamma} + 1 \right) \cdot \epsilon_{\text{sp},\gamma}^*.$$

Taking the supremum over $\lambda \in \Lambda$, observing the non-negativity of the regularization term, and organizing the constants yield the final result stated in the theorem. $\square$

The preceding result controls the worst-case proxy error uniformly over $\lambda \in \Lambda$. Many optimization proxy applications, however, are evaluated in distribution: a proxy is trained on representative instances from $\mathbb{P}_\lambda$ and judged by its average decision quality on future in-

stances drawn from the same population. The next result gives this in-distribution guarantee directly. It bounds the expected objective-value gap between the proxy decision and the true optimizer by separating optimization error from the approximation error of the chosen proxy class.

Theorem 3.2.7 (In-Distribution Proxy Error Decomposition). *Suppose Assumptions 3.2.1 and 3.2.2 hold. For all $\beta \in \mathbb{R}^p$,*

$$F_\gamma(\beta) - \mathbb{E} [h(\theta^*(\tilde{\lambda}), \tilde{\lambda})] \leq \frac{CL + \gamma}{2} \|\beta - \beta_{\text{avg}, \gamma}^*\|^2 + \epsilon_{\text{sp}, \gamma}^*.$$

The first term in Theorem 3.2.7 is algorithmic: it measures how far the learned coefficients are from the population optimizer $\beta_{\text{avg}, \gamma}^*$. The second term is intrinsic to the proxy class: even exact optimization cannot improve on the best complexity-penalized proxy error $\epsilon_{\text{sp}, \gamma}^*$. Compared with the uniform bound, the in-distribution decomposition avoids the price of converting average training progress into worst-case control, which yields a sharper and more transparent performance statement.

Proof.

The proof is straightforward, using the smoothness of F_γ and the optimality of $\beta_{\text{avg}, \gamma}^*$. For any $\lambda \in \Lambda$,

$$\begin{aligned} f_\gamma(\beta, \lambda) - h(\theta^*(\lambda), \lambda) &= (f_\gamma(\beta, \lambda) - f_\gamma(\beta_{\text{avg}, \gamma}^*, \lambda)) + (f_\gamma(\beta_{\text{avg}, \gamma}^*, \lambda) - f_\gamma(\beta_{\text{sp}, \gamma}^*, \lambda)) \\ &\quad + \underbrace{(f_\gamma(\beta_{\text{sp}, \gamma}^*, \lambda) - h(\theta^*(\lambda), \lambda))}_{\leq \epsilon_{\text{sp}, \gamma}^*}. \end{aligned}$$

Taking expectation on both sides yields

$$\begin{aligned} F_\gamma(\beta) - \mathbb{E} [h(\theta^*(\tilde{\lambda}), \tilde{\lambda})] &\leq (F_\gamma(\beta) - F_\gamma(\beta_{\text{avg}, \gamma}^*)) + (F_\gamma(\beta_{\text{avg}, \gamma}^*) - F_\gamma(\beta_{\text{sp}, \gamma}^*)) + \epsilon_{\text{sp}, \gamma}^* \\ &\leq (F_\gamma(\beta) - F_\gamma(\beta_{\text{avg}, \gamma}^*)) + \epsilon_{\text{sp}, \gamma}^* \\ &\leq \frac{CL + \gamma}{2} \|\beta - \beta_{\text{avg}, \gamma}^*\|^2 + \epsilon_{\text{sp}, \gamma}^*. \end{aligned}$$

The first inequality above uses the definition of $\epsilon_{\text{sp}, \gamma}^*$ as a supremum proxy error, the second uses optimality of $\beta_{\text{avg}, \gamma}^*$, and the third uses $(CL + \gamma)$ -smoothness of $F_\gamma(\cdot)$. $\square$

3.3 SGD to Learn an Optimization Proxy

3.3.1 Relaxed Weak Growth Condition (RWGC)

Recall the following definition motivated by [23, Lemma 2.4]:

Definition 3.3.1 (Relaxed Weak Growth Condition (RWGC)). *Consider a family of functions $g(\cdot, z) : \mathbb{R}^d \rightarrow \mathbb{R}$, $z \sim \mathbb{P}_z$, and $G(\cdot) := \mathbb{E}_{z \sim \mathbb{P}_z} [g(\cdot, z)]$. Let $G^* := \min_{\mathbf{w} \in \mathbb{R}^d} G(\mathbf{w})$. Then, g and G are said to satisfy the relaxed weak growth condition with constants $\rho \geq 0$ and $\sigma \geq 0$, if for all $\mathbf{w} \in \mathbb{R}^d$,*

$$\mathbb{E}_{z \sim \mathbb{P}_z} [\|\nabla g(\mathbf{w}, z)\|^2] \leq 2\rho(G(\mathbf{w}) - G^*) + \sigma^2.$$

Furthermore, recall the following theorem for SGD convergence under RWGC.

Lemma 3.3.2 ([23]). *Suppose that the RWGC holds for the family of functions $g(\cdot, z) : \mathbb{R}^d \rightarrow \mathbb{R}$, and that $G(\cdot) := \mathbb{E}_{z \sim \mathbb{P}_z} [g(\cdot, z)]$ is μ_g -strongly convex for some $\mu_g > 0$. Consider the SGD algorithm, initialized at $\mathbf{w}_0 \in \mathbb{R}^d$, with iterations*

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \frac{\bar{\eta}}{\rho} \nabla g(\mathbf{w}_t, z_t),$$

where $z_t \sim \mathbb{P}_z$ is a random sample and $\bar{\eta} \in \left(0, \min \left\{1, \frac{\rho}{\mu_g}\right\}\right]$ parametrizes the step-size. Let $\mathbf{w}^* := \arg \min_{\mathbf{w} \in \mathbb{R}^d} G(\mathbf{w})$. Then for all $t \geq 0$, we have

$$\mathbb{E}[\|\mathbf{w}_t - \mathbf{w}^*\|^2] \leq \left(1 - \frac{\bar{\eta}\mu_g}{\rho}\right)^t \|\mathbf{w}_0 - \mathbf{w}^*\|^2 + \frac{\bar{\eta}\sigma^2}{\rho\mu_g}.$$

3.3.2 Solve Problem (3.2) with SGD

The Gower-Vaswani result applies unchanged. What remains is to verify that the Ridge-regularized stochastic objective satisfies the RWGC.

Lemma 3.3.3 (Problem (3.2) satisfies RWGC). *Suppose that Assumption 3.2.1 holds. Then, the family of functions $f_\gamma(\cdot, \lambda)$ and the function $F_\gamma(\cdot)$, defined in (3.4), satisfy the relaxed weak growth condition (RWGC) in Definition 3.3.1 $\rho = CL + \gamma$ and $\sigma^2 = 2(CL + \gamma)\epsilon_{\text{sp}, \gamma}^*$. Namely, for all $\beta \in \mathbb{R}^d$,*

$$\mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} [\|\nabla f_\gamma(\beta, \tilde{\lambda})\|^2] \leq 2(CL + \gamma)(F_\gamma(\beta) - F_\gamma^*) + 2(CL + \gamma)\epsilon_{\text{sp}, \gamma}^*.$$

Proof. Recall that Proposition 3.2.5 already establishes the $(CL + \gamma)$ -smoothness of $f_\gamma(\cdot, \lambda)$.

Define an auxiliary term

$$\beta_\gamma^*(\lambda) \in \arg \min_{\beta \in \mathbb{R}^p} f_\gamma(\beta, \lambda).$$

By $(CL + \gamma)$ -smoothness, we have

$$\begin{aligned} f_\gamma(\beta, \lambda) &\geq f_\gamma(\beta_\gamma^*(\lambda), \lambda) + \nabla f_\gamma(\beta_\gamma^*(\lambda), \lambda)^\top (\beta - \beta_\gamma^*(\lambda)) \\ &\quad + \frac{1}{2(CL + \gamma)} \|\nabla f_\gamma(\beta, \lambda) - \nabla f_\gamma(\beta_\gamma^*(\lambda), \lambda)\|^2. \end{aligned}$$

Using $\nabla f_\gamma(\beta_\gamma^*(\lambda), \lambda) = 0$ for all λ , rearranging and recalling the definitions of $\beta_{\text{avg}, \gamma}^*$ and $\beta_{\text{sp}, \gamma}^*$ yields

$$\begin{aligned} &\frac{1}{2(CL + \gamma)} \|\nabla f_\gamma(\beta, \lambda)\|^2 \\ &\leq f_\gamma(\beta, \lambda) - f_\gamma(\beta_\gamma^*(\lambda), \lambda) \\ &\leq f_\gamma(\beta, \lambda) - h(\theta^*(\lambda), \lambda) \\ &= (f_\gamma(\beta, \lambda) - f_\gamma(\beta_{\text{avg}, \gamma}^*, \lambda)) \\ &\quad + (f_\gamma(\beta_{\text{avg}, \gamma}^*, \lambda) - f_\gamma(\beta_{\text{sp}, \gamma}^*, \lambda)) + (f_\gamma(\beta_{\text{sp}, \gamma}^*, \lambda) - h(\theta^*(\lambda), \lambda)) \\ &\leq (f_\gamma(\beta, \lambda) - f_\gamma(\beta_{\text{avg}, \gamma}^*, \lambda)) \\ &\quad + (f_\gamma(\beta_{\text{avg}, \gamma}^*, \lambda) - f_\gamma(\beta_{\text{sp}, \gamma}^*, \lambda)) + \epsilon_{\text{sp}, \gamma}^*, \end{aligned}$$

where the second inequality uses optimality of $\theta^*(\lambda)$ and the third uses that $\epsilon_{\text{sp}, \gamma}^*$ is attained at $\beta_{\text{sp}, \gamma}^*$.

Taking expectations on both sides, we get

$$\begin{aligned} &\frac{1}{2(CL + \gamma)} \mathbb{E} [\|\nabla f_\gamma(\beta, \tilde{\lambda})\|^2] \\ &\leq F_\gamma(\beta) - F_\gamma(\beta_{\text{avg}, \gamma}^*) + F_\gamma(\beta_{\text{avg}, \gamma}^*) - F_\gamma(\beta_{\text{sp}, \gamma}^*) + \epsilon_{\text{sp}, \gamma}^* \\ &\leq F_\gamma(\beta) - F_\gamma(\beta_{\text{avg}, \gamma}^*) + \epsilon_{\text{sp}, \gamma}^*, \end{aligned}$$

where we used that the definition of $\beta_{\text{avg}, \gamma}^*$ implies that $F_\gamma(\beta_{\text{avg}, \gamma}^*) \leq F_\gamma(\beta_{\text{sp}, \gamma}^*)$. Multiplying through by $2(CL + \gamma)$ completes the proof. $\square$

Next, we combine Theorems 2.2.4 and 2.3.2 and Lemma 2.3.3 to yield our main result: a bound on the expected proxy error for constant step-size SGD. Recall constant step-size SGD in this setting yields the iteration

$$\beta_{t+1} \leftarrow \beta_t - \eta \nabla_\beta f_\gamma(\beta_t, \tilde{\lambda}_t),$$

where

$$\nabla_{\beta} f_{\gamma}(\beta_t, \tilde{\lambda}_t) = \Phi(\tilde{\lambda}_t)^{\top} \nabla_{\theta} h(\Phi(\tilde{\lambda}_t) \beta_t, \tilde{\lambda}_t) + \gamma \cdot \beta_t,$$

and $\tilde{\lambda}_t \sim \mathbb{P}_{\lambda}$. We assume that the gradient $\nabla_{\theta} h(\theta, \lambda)$ is easily computable for any θ and λ .

Similar to Theorem 3.2.6, we present the result in two parts: when strict strong convexity is satisfied, and when it is not.

Theorem 3.3.4 (Expected Proxy Error Convergence for SGD). *Under Assumption 3.2.1, consider applying SGD to the expected Problem (3.4) with a constant step-size $\eta = \frac{\bar{\eta}}{CL+\gamma}$ parameterized by $\bar{\eta} \in (0, 1]$. Then, after T iterations, the expected proxy error satisfies the following relationships:*

1. When $c\mu > 0$,

$$\mathbb{E} [\epsilon_{\text{sp}}(\beta_T)] \leq \frac{3CL}{2} \left(1 - \frac{\bar{\eta}(c\mu + \gamma)}{CL + \gamma}\right)^T \|\beta_0 - \beta_{\text{avg}, \gamma}^*\|^2 + \left(\frac{3L}{\mu} + \frac{3CL(\bar{\eta} + 1)}{c\mu + \gamma}\right) \cdot \epsilon_{\text{sp}, \gamma}^*.$$

$$\text{In particular, for } T \geq \left\lceil \frac{CL}{\bar{\eta}(c\mu + \gamma)} \log \left(\frac{(c\mu + \gamma) \|\beta_0 - \beta_{\text{avg}, \gamma}^*\|^2}{2\bar{\eta}\epsilon_{\text{sp}, \gamma}^*} \right) \right\rceil,$$

$$\mathbb{E} [\epsilon_{\text{sp}}(\beta_T)] \leq \left(\frac{3L}{\mu} + \frac{3CL(2\bar{\eta} + 1)}{c\mu + \gamma} \right) \epsilon_{\text{sp}, \gamma}^*.$$

$$\text{When } \epsilon_{\text{sp}, \gamma}^* = 0, \text{ then for any } \epsilon > 0 \text{ and } T \geq \frac{CL}{\bar{\eta}(c\mu + \gamma)} \log \left(\frac{3CL \|\beta_0 - \beta_{\text{avg}, \gamma}^*\|^2}{2\epsilon} \right),$$

$$\mathbb{E} [\epsilon_{\text{sp}, \gamma}(\beta_T)] \leq \epsilon.$$

2. When $c\mu = 0$,

$$\begin{aligned} \mathbb{E} [\epsilon_{\text{sp}}(\beta_T)] &\leq \mathbb{E} [\epsilon_{\text{sp}, \gamma}(\beta_T)] \\ &\leq \frac{3(CL + \gamma)}{2} \left(1 - \frac{\bar{\eta}\gamma}{CL + \gamma}\right)^T \|\beta_0 - \beta_{\text{avg}, \gamma}^*\|^2 \\ &\quad + \left(\frac{3(\bar{\eta} + 2)(CL + \gamma)}{\gamma} + 1 \right) \epsilon_{\text{sp}, \gamma}^*. \end{aligned}$$

$$\text{In particular, for } T \geq \left\lceil \frac{CL + \gamma}{\bar{\eta}\gamma} \log \left(\frac{\gamma \|\beta_0 - \beta_{\text{avg}, \gamma}^*\|^2}{2\bar{\eta}\epsilon_{\text{sp}, \gamma}^*} \right) \right\rceil,$$

$$\mathbb{E} [\epsilon_{\text{sp}}(\beta_T)] \leq \left(\frac{6(CL + \gamma)(\bar{\eta} + 1)}{\gamma} + 1 \right) \epsilon_{\text{sp}, \gamma}^*.$$

Remark 3.3.5. *The interpolable special case under $c\mu > 0$ implies $\epsilon_{\text{sp}, \gamma}^* = \epsilon_{\text{sp}}^* = 0$ due*

to non-negativity of both the suboptimality and the regularization. Thus, either $\gamma = 0$ or $\beta_{\text{sp},\gamma}^* = \mathbf{0}$.

When $\gamma = 0$, the first part of Theorem 3.3.4 recovers [12, Theorem 3.4] up to some constant round-off.

When γ is large, the proxy error is primarily driven by the penalty term. In this limit, $\beta_{\text{sp},\gamma}^*, \beta_{\text{avg},\gamma}^* \rightarrow \mathbf{0}$, and $\epsilon_{\text{sp},\gamma}^* \rightarrow \epsilon_{\text{sp}}(\mathbf{0})$. Moreover, $\frac{CL+\gamma}{\gamma} \rightarrow 1$. As a result, the required number of iterations T reduces to a constant order. This suggests that SGD converges very quickly for large γ while involving a fundamental tradeoff: a large optimal penalized error $\epsilon_{\text{sp},\gamma}^*$, and correspondingly, so is the expected proxy error at time T .

Proof. Our strategy will be to invoke Lemma 3.3.3 and apply Lemma 3.3.2, in conjunction with Theorem 3.2.6. To that end, notice that the upper bound on the step-size parameter $\bar{\eta}$ given in Lemma 3.3.2 becomes $\frac{CL+\gamma}{c\mu+\gamma} \geq 1$. Hence, we need only constrain $0 < \bar{\eta} \leq 1$.

Applying Lemma 3.3.2 yields

$$\mathbb{E} [\|\beta_T - \beta_{\text{avg},\gamma}^*\|^2] \leq \left(1 - \frac{\bar{\eta}(c\mu + \gamma)}{CL + \gamma}\right)^T \|\beta_0 - \beta_{\text{avg},\gamma}^*\|^2 + \frac{2\bar{\eta}\epsilon_{\text{sp},\gamma}^*}{c\mu + \gamma}.$$

Taking the expectation of Theorem 3.2.6 on both sides and substituting the above, we have

1. When $c\mu > 0$,

$$\begin{aligned} \mathbb{E} [\epsilon_{\text{sp}}(\beta_T)] &\leq \mathbb{E} [\epsilon_{\text{sp},\gamma}(\beta_T)] \\ &\leq \frac{3CL}{2} \mathbb{E} [\|\beta_T - \beta_{\text{avg},\gamma}^*\|^2] + \left(\frac{3L}{\mu} + \frac{3CL}{c\mu + \gamma}\right) \cdot \epsilon_{\text{sp},\gamma}^* \\ &\leq \frac{3CL}{2} \left(1 - \frac{\bar{\eta}(c\mu + \gamma)}{CL + \gamma}\right)^T \|\beta_0 - \beta_{\text{avg},\gamma}^*\|^2 \\ &\quad + \frac{3CL\bar{\eta}}{c\mu + \gamma} \cdot \epsilon_{\text{sp},\gamma}^* + \left(\frac{3L}{\mu} + \frac{3CL}{c\mu + \gamma}\right) \cdot \epsilon_{\text{sp},\gamma}^*. \end{aligned} \quad (3.10)$$

2. When $c\mu = 0$,

$$\begin{aligned} \mathbb{E} [\epsilon_{\text{sp}}(\beta_T)] &\leq \mathbb{E} [\epsilon_{\text{sp},\gamma}(\beta_T)] \\ &\leq \frac{3(CL + \gamma)}{2} \mathbb{E} [\|\beta_T - \beta_{\text{avg},\gamma}^*\|^2] + \left(\frac{6(CL + \gamma)}{\gamma} + 1\right) \cdot \epsilon_{\text{sp},\gamma}^* \\ &\leq \frac{3(CL + \gamma)}{2} \left(1 - \frac{\bar{\eta}\gamma}{CL + \gamma}\right)^T \|\beta_0 - \beta_{\text{avg},\gamma}^*\|^2 \\ &\quad + \frac{3(CL + \gamma)\bar{\eta}}{\gamma} \cdot \epsilon_{\text{sp},\gamma}^* + \left(\frac{6(CL + \gamma)}{\gamma} + 1\right) \cdot \epsilon_{\text{sp},\gamma}^*. \end{aligned} \quad (3.11)$$

Rearranging yields the expected proxy error upper bounds.

Next, we prove the bound of the iteration complexity. The strategy is identical for the two cases, so we only demonstrate the first case ($c\mu > 0$) here.

First, consider the case $\epsilon_{\text{sp},\gamma}^* > 0$.

Our goal will be to choose T large enough to drive the first term on the right side of (3.10) to below $\left(\frac{3CL\bar{\eta}}{c\mu+\gamma} \cdot \epsilon_{\text{sp},\gamma}^*\right)$. To simplify exposition, let $\kappa_1 = \frac{CL}{c\mu+\gamma}$. Then, we need to ensure that

$$T \log(1 - \bar{\eta}/\kappa_1) \leq \log \left(\frac{2\kappa_1 \bar{\eta} \epsilon_{\text{sp},\gamma}^*}{CL \|\beta_0 - \beta_{\text{avg},\gamma}^*\|^2} \right).$$

We can upper bound the left side using the identity $\log(1+x) \leq x$. Hence, it would be sufficient if

$$T \geq \frac{\kappa_1}{\bar{\eta}} \log \left(\frac{CL \|\beta_0 - \beta_{\text{avg},\gamma}^*\|^2}{2\kappa_1 \bar{\eta} \epsilon_{\text{sp},\gamma}^*} \right).$$

Replacing κ_1 with its definition yields the iteration complexity bound.

For the second case where $\epsilon_{\text{sp}}^* = 0$, we follow a similar argument but seek to drive the error to below ϵ . We omit the details.

The result for $c\mu = 0$ is obtained by replacing κ_1 with $\kappa_2 = \frac{CL+\gamma}{\gamma}$, and changing the optimization target to driving the first term on the right side of (3.11) to below $\left(\frac{3(CL+\gamma)\bar{\eta}}{\gamma} \cdot \epsilon_{\text{sp},\gamma}^*\right)$. $\square$

The preceding theorem converts SGD progress into uniform proxy control. We now record the sharper in-distribution counterpart, which is often the primary performance metric for optimization proxies. Instead of asking for worst-case control over all $\lambda \in \Lambda$, this result asks how well the learned proxy performs on a fresh instance from the same distribution used for training.

Theorem 3.3.6 (In-Distribution Proxy Error Convergence under SGD). *Under Assumption 3.2.1, consider applying SGD to the expected Problem (3.4) with a constant step-size $\eta = \frac{\bar{\eta}}{CL+\gamma}$ parameterized by $\bar{\eta} \in (0, 1]$. Then, after T iterations, the expected in-distribution proxy error satisfies*

$$\begin{aligned} F_\gamma(\beta_T) - \mathbb{E} [h(\theta^*(\tilde{\lambda}), \tilde{\lambda})] \\ \leq \frac{CL+\gamma}{2} \left(1 - \frac{\bar{\eta}(c\mu+\gamma)}{CL+\gamma}\right)^T \|\beta_0 - \beta_{\text{avg},\gamma}^*\|^2 + \left(\frac{(CL+\gamma)\bar{\eta}}{c\mu+\gamma} + 1\right) \cdot \epsilon_{\text{sp},\gamma}^*, \end{aligned}$$

where the expectation is taken over the stochasticity of SGD iterations. In particular, for

$$T \geq \left\lceil \frac{CL+\gamma}{\bar{\eta}(c\mu+\gamma)} \log \left(\frac{(c\mu+\gamma) \|\beta_0 - \beta_{\text{avg},\gamma}^*\|^2}{2\bar{\eta} \epsilon_{\text{sp},\gamma}^*} \right) \right\rceil,$$

we have

$$F_\gamma(\beta_T) - \mathbb{E} [h(\theta^*(\tilde{\lambda}), \tilde{\lambda})] \leq \left(\frac{2(CL + \gamma)\bar{\eta}}{c\mu + \gamma} + 1 \right) \epsilon_{\text{sp}, \gamma}^*.$$

When $\epsilon_{\text{sp}, \gamma}^* = 0$, then for any $\epsilon > 0$ and

$$T \geq \frac{CL + \gamma}{\bar{\eta}(c\mu + \gamma)} \log \left(\frac{(CL + \gamma)\|\beta_0 - \beta_{\text{avg}, \gamma}^*\|^2}{2\epsilon} \right),$$

we have

$$F_\gamma(\beta_T) - \mathbb{E} [h(\theta^*(\tilde{\lambda}), \tilde{\lambda})] \leq \epsilon.$$

The bound has the same two-term structure as Theorem 3.2.7, now with the coefficient error replaced by the explicit SGD convergence rate. The exponentially decaying term is the cost of finite training, while the residual term is proportional to $\epsilon_{\text{sp}, \gamma}^*$, the best achievable proxy error in the chosen feature class. In the realizable case $\epsilon_{\text{sp}, \gamma}^* = 0$, constant step-size SGD drives expected decision suboptimality below any prescribed tolerance at a linear rate.

Proof. Taking expectation on both sides of Theorem 3.2.7 and applying Lemma 3.3.2 yields the upper bound on expected in-distribution proxy error in the theorem statement.

The proof of iteration complexity bound uses exactly the same strategy in Theorem 3.3.4 by setting

$$\kappa = \frac{CL + \gamma}{c\mu + \gamma},$$

and driving the exponentially decaying term

$$\frac{CL + \gamma}{2} \left(1 - \frac{\bar{\eta}(c\mu + \gamma)}{CL + \gamma} \right)^T \|\beta_0 - \beta_{\text{avg}, \gamma}^*\|^2$$

below the target level

$$\frac{(CL + \gamma)\bar{\eta}}{c\mu + \gamma} \epsilon_{\text{sp}, \gamma}^* \quad (\text{or below } \epsilon \text{ when } \epsilon_{\text{sp}, \gamma}^* = 0).$$

We omit the details here. □

3.4 Specialized Results

3.4.1 Fourier Basis for Periodic Solution Map on $\Lambda = [0, 2\pi]$

We next provide another specialization for periodic parameter domains by taking $\Lambda = [0, 2\pi]$. Our goal is to leverage classical approximation theory and the structure of an orthogonal basis

to control the conditioning constants that appear in Section 2, Assumption 3.2.2 and Equation (3.3).

Fourier basis and feature map. We approximate each component $i = 1, \dots, d$ of the real-valued path $\theta_i^*(\cdot)$ using the first q *real* Fourier basis functions on $[0, 2\pi]$. Throughout this section we take $\mathbb{P}_\lambda$ to be the uniform distribution on $[0, 2\pi]$. For an integer truncation level $K \geq 0$, define the real Fourier feature vector $\varphi_K(\lambda) \in \mathbb{R}^{2K+1}$ as

$$\varphi_K(\lambda) = (\phi_0(\lambda), \phi_1(\lambda), \dots, \phi_{2K}(\lambda))^\top, \quad (3.12)$$

where

$$\phi_0(\lambda) = 1, \quad \phi_{2n-1}(\lambda) = \cos(n\lambda), \quad \phi_{2n}(\lambda) = \sin(n\lambda), \quad n \geq 1.$$

Let $q := 2K + 1$ denote the number of basis functions per coordinate, so the total number of basis functions is $p = qd = (2K + 1)d$. As in the Legendre specialization, the feature matrix $\Phi(\lambda) \in \mathbb{R}^{d \times qd}$ is block-diagonal with d blocks of size $1 \times q$, each block equal to $\varphi_K(\lambda)^\top$.

Orthogonality under the uniform distribution. The real Fourier basis (trigonometric polynomials) is orthogonal in $L^2(\text{Unif}[0, 2\pi])$:

$$\begin{aligned} \mathbb{E} [\cos(n\tilde{\lambda}) \cos(m\tilde{\lambda})] &= \mathbb{E} [\sin(n\tilde{\lambda}) \sin(m\tilde{\lambda})] = \frac{1}{2} \mathbb{I}\{n = m \geq 1\}, \\ \mathbb{E} [\sin(n\tilde{\lambda}) \cos(m\tilde{\lambda})] &= \mathbb{E} [\cos(n\tilde{\lambda})] = \mathbb{E} [\sin(n\tilde{\lambda})] = 0 \quad (n \geq 1), \end{aligned}$$

where $\tilde{\lambda} \sim \text{Unif}[0, 2\pi]$.

For this basis, depending on the value of $p = qd$ implied by the choice of K , We can bound the ratio C_p/c_p in a dimension-free manner, analogous to [12, Lemma 4.1].

Lemma 3.4.1 (C_p/c_p for Real Fourier Basis). *Take $\Lambda = [0, 2\pi]$ and $\mathbb{P}_\lambda$ to be the uniform distribution on $[0, 2\pi]$. Consider the block-diagonal basis described above with $q = 2K + 1$ real Fourier features per coordinate given by (3.12). Then*

$$\frac{C_p}{c_p} = 2q.$$

Proof. Similar to the proof of [12, Lemma 4.1], $\Phi(\lambda) \in \mathbb{R}^{d \times qd}$ is block-diagonal with d identical blocks of size $1 \times q$. Denote one block by $\psi(\lambda) \in \mathbb{R}^{1 \times q}$, where $\psi(\lambda)^\top = \varphi_K(\lambda)$. The nonzero eigenvalues of $\Phi(\lambda)^\top \Phi(\lambda)$ coincide with those of $\psi(\lambda)^\top \psi(\lambda)$, which has a single nonzero eigenvalue equal to $\|\psi(\lambda)\|^2$.

We first compute C_p . By definition,

$$C_p = \sup_{\lambda \in [0, 2\pi]} \|\psi(\lambda)\|^2 = \sup_{\lambda \in [0, 2\pi]} \|\varphi_K(\lambda)\|^2.$$

Using $\cos^2(t) + \sin^2(t) = 1$ for each harmonic,

$$\|\varphi_K(\lambda)\|^2 = 1 + \sum_{n=1}^K (\cos^2(n\lambda) + \sin^2(n\lambda)) = 1 + 2K = q,$$

for every $\lambda \in [0, 2\pi]$. Hence $C_p = q$.

Next we compute c_p . By definition,

$$c_p = \sigma_{\min} \left\{ \mathbb{E} \left[\psi(\tilde{\lambda})^\top \psi(\tilde{\lambda}) \right] \right\} = \sigma_{\min} \left\{ \mathbb{E} \left[\varphi_K(\tilde{\lambda}) \varphi_K(\tilde{\lambda})^\top \right] \right\}.$$

By the orthogonality identities listed above, the matrix $\mathbb{E} \left[\varphi_K(\tilde{\lambda}) \varphi_K(\tilde{\lambda})^\top \right] \in \mathbb{R}^{q \times q}$ is diagonal with entries

$$\mathbb{E} [\phi_0(\tilde{\lambda})^2] = 1, \quad \mathbb{E} [\cos^2(n\tilde{\lambda})] = \frac{1}{2}, \quad \mathbb{E} [\sin^2(n\tilde{\lambda})] = \frac{1}{2} \quad (n = 1, \dots, K),$$

and all cross-terms equal to 0. Therefore,

$$c_p = \min \left\{ 1, \underbrace{\frac{1}{2}, \dots, \frac{1}{2}}_{2K \text{ times}} \right\} = \frac{1}{2} \quad \text{for } K \geq 1.$$

Combining the two bounds yields

$$\frac{C_p}{c_p} = 2q.$$

□

Even better than the Legendre case, for the real Fourier basis the ratio C_p/c_p scales *linearly* with q up to universal constants, reflecting the exact energy identity $\cos^2(n\lambda) + \sin^2(n\lambda) = 1$ and the strong orthogonality of the basis under the uniform distribution.

Lemma 3.4.2 (ϵ_{sp}^* for ν -Differentiable Periodic Solution Maps (Real Fourier Basis)).

Suppose Assumption 3.2.1 holds. Let $\Lambda = [0, 2\pi]$ and assume $\boldsymbol{\theta}^*(\cdot) \in C_{2\pi}^\nu$ for integers $\nu \geq 1$, i.e. $\boldsymbol{\theta}^*(\cdot)$ is 2π -periodic and ν differentiable over Λ . Further assume there exists constant $V > 0$ such that for all $i = 1, \dots, d$, $\boldsymbol{\theta}_i^{(\nu)*}(\lambda)$ has total variation bounded by V . Then, for any $K \geq 0$ and the real Fourier basis described above,

$$\epsilon_{\text{sp}}^* \leq \frac{dL}{2} \left(\frac{\pi}{q+1} \right)^{2\nu} V^2,$$

where $q = 2K + 1$.

Proof. The proof follows the same template as [12, Lemma 4.2]. This time, for each coordinate $i \in [d]$, we approximate $\theta_i^*(\lambda)$ by real Fourier basis (trigonometric polynomials) of degree K in the space

$$\mathcal{T}_K := \left\{ a_0 + \sum_{n=1}^K (a_n \cos(n\lambda) + b_n \sin(n\lambda)) : a_n, b_n \in \mathbb{R} \right\}.$$

Again, let $\bar{\beta}$ be the coefficients corresponding to the best approximating polynomial from $\mathcal{T}_K$. We will show that $\epsilon_{\text{sp}}(\bar{\beta})$ satisfies the claimed bound using the well-known Jackson's theorem.

Since $\theta_i^*(\cdot)$ is 2π -periodic and $\theta_i^{(\nu)*}$ has bounded variation V , the Jackson's theorem statement for trigonometric approximation in [3, Theorem 1.8] ensures existence of a truncated Fourier series $s_{i,K} \in \mathcal{T}_K$ such that

$$\sup_{\lambda \in [0, 2\pi]} |s_{i,K}(\lambda) - \theta_i^*(\lambda)| \leq \left(\frac{\pi}{2(K+1)} \right)^\nu \cdot V, \quad \forall i \in [d]. \quad (3.13)$$

Let $\bar{\beta}$ collect these coefficients across i (so that $\Phi(\lambda)\bar{\beta}$ produces the vector $(s_{i,K}(\lambda))_{i \in [d]}$). Then, by definition,

$$\epsilon_{\text{sp}}(\bar{\beta}) = \sup_{\lambda \in \Lambda} \left\{ h((s_{i,K}(\lambda))_{i \in [d]}, \lambda) - h(\theta^*(\lambda), \lambda) \right\}.$$

By smoothness of $h(\cdot, \lambda)$ (as in [12, Lemma 4.2]), for any $\lambda \in \Lambda$,

$$h((s_{i,K}(\lambda))_{i \in [d]}, \lambda) - h(\theta^*(\lambda), \lambda) \leq \frac{L}{2} \left\| (s_{i,K}(\lambda))_{i \in [d]} - \theta^*(\lambda) \right\|^2.$$

Combining this with the definition of the ℓ_2 norm gives

$$\begin{aligned} \epsilon_{\text{sp}}(\bar{\beta}) &\leq \sup_{\lambda \in [0, 2\pi]} \left\{ \frac{L}{2} \left\| (s_{i,K}(\lambda))_{i \in [d]} - \theta^*(\lambda) \right\|^2 \right\} \\ &\leq \sup_{\lambda \in [0, 2\pi]} \left\{ \frac{dL}{2} \cdot \sup_{i \in [d]} |s_{i,K}(\lambda) - \theta_i^*(\lambda)|^2 \right\} \\ &\leq \frac{dL}{2} \left(\sup_{i \in [d]} \sup_{\lambda \in [0, 2\pi]} |s_{i,K}(\lambda) - \theta_i^*(\lambda)| \right)^2. \end{aligned}$$

Applying (3.13) yields

$$\epsilon_{\text{sp}}(\bar{\beta}) \leq \frac{dL}{2} \left(\frac{\pi}{2(K+1)} \right)^{2\nu} \cdot V^2.$$

Since $\epsilon_{\text{sp}}^* \leq \epsilon_{\text{sp}}(\bar{\beta})$ by optimality of ϵ_{sp}^* , the claim follows. Finally, substituting $q = 2K + 1$ gives the stated expression in terms of q . $\square$

Combining Lemma 3.4.1 and Lemma 3.4.2 with Section 3.2, Theorem 3.3.4 allows us to compute the requisite basis size and number of iterations needed to achieve a target solution-map error for periodic domains:

Theorem 3.4.3 (SGD for ν -Differentiable Periodic Solution Maps (Real Fourier Basis)).

Let $\Lambda = [0, 2\pi]$ and $\mathbb{P}_\lambda$ be the uniform distribution on $[0, 2\pi]$. Assume the conditions of Lemma 3.4.2 hold with an integer $\nu \geq 1$. If we use $q = O(\epsilon^{\frac{1}{1-2\nu}})$ real Fourier basis functions per coordinate (i.e. $q = 2K + 1$ for some K), and run constant-step size SGD for $O(\epsilon^{\frac{1}{1-2\nu}} \log(1/\epsilon))$ iterations, then the resulting iterate β_T satisfies $\mathbb{E}[\epsilon_{\text{sp}}(\beta_T)] \leq \epsilon$.

Similar to [12, Theorem 4.3], both big “Oh” terms should be interpreted as $\epsilon \rightarrow 0$ and suppress constants possibly depending on h and Λ but not on ϵ . Compared to the polynomial (Legendre) specialization, the real Fourier specialization improves the conditioning factor from quadratic to linear in q , which in turn improves the SGD iteration complexity from $O(q^2 \log q)$ to $O(q \log q)$.

Proof. As in the proof of [12, Theorem 4.3], we focus on asymptotics as $\epsilon \rightarrow 0$ and suppress constants that do not depend on ϵ . In particular, we write $a \lesssim b$ if there exists a constant C (not depending on ϵ but possibly depending on h and Λ) such that $a \leq Cb$.

Recall that we take $q = 2K + 1$ basis functions per coordinate and thus $p = qd$. To achieve a small solution-map error, we will need q large. By Section 3.2, Theorem 3.3.4, our first goal is to choose q large enough that

$$\epsilon \geq \frac{8L(\bar{\eta} + 1)}{\mu} \frac{C_q}{c_q} \cdot \epsilon_{\text{sp}}^*(q). \quad (3.14)$$

By Lemma 3.4.2, the optimal solution-map error satisfies

$$\epsilon_{\text{sp}}^*(q) \leq \frac{dL}{2} \left(\frac{\pi}{q+1} \right)^{2\nu} V^2 \lesssim q^{-2\nu}. \quad (3.15)$$

By Lemma 3.4.1, we have $C_q/c_q = 2q$. Hence the right-hand side of (3.14) is bounded by

$$\frac{8L(\bar{\eta} + 1)}{\mu} \frac{C_q}{c_q} \cdot \epsilon_{\text{sp}}^*(q) \lesssim q \cdot q^{-2\nu} = q^{1-2\nu}.$$

Since $\nu \geq 1$, the exponent $1 - 2\nu < 0$, so it suffices to take

$$q \gtrsim \epsilon^{\frac{1}{1-2\nu}} \quad (\text{equivalently, } q = O(\epsilon^{\frac{1}{1-2\nu}}) \text{ as } \epsilon \rightarrow 0)$$

basis functions per coordinate to ensure (3.14) holds.

By Section 3.2, Theorem 3.3.4, the iteration count should exceed

$$\frac{L}{\bar{\eta}\mu} \frac{C_q}{c_q} \log \left(\frac{c_q \mu \|\beta_{\text{avg}}^*\|^2}{2\bar{\eta} \epsilon_{\text{sp}}^*(q)} \right) \lesssim q \cdot (\log \|\beta_{\text{avg}}^*\|^2 + \log q),$$

where we used $C_q/c_q = 2q$ and $c_q = \frac{1}{2}$ (for $K \geq 1$) from the proof of Lemma 3.4.1.

To simplify the logarithmic term, we show $\|\beta_{\text{avg}}^*\| \lesssim 1$ as $q \rightarrow \infty$. Let $\bar{\beta}$ be constructed by approximating each coordinate $\theta_i^*(\lambda)$ by a degree- K real Fourier basis functions in $\mathcal{T}_K$ (as in the proof of Lemma 3.4.2). Write the coefficients for coordinate i as

$$s_{i,K}(\lambda) = a_{i,0} + \sum_{n=1}^K (a_{i,n} \cos(n\lambda) + b_{i,n} \sin(n\lambda)),$$

and let $\bar{\beta}$ stack $\{a_{i,n}, b_{i,n}\}$ across i .

Under the assumption that $\theta_i^{(\nu)*}$ has bounded variation (uniformly in i), standard Fourier series approximation results implies Fourier coefficient decay¹ of order $|a_{i,n}| + |b_{i,n}| \lesssim n^{-(\nu+1)}$ for $n \geq 1$, with constants independent of K . Therefore

$$\|\bar{\beta}\|^2 = \sum_{i=1}^d \left(|a_{i,0}|^2 + \sum_{n=1}^K (|a_{i,n}|^2 + |b_{i,n}|^2) \right) \lesssim \sum_{n=1}^K \frac{1}{n^{2(\nu+1)}} \lesssim \sum_{n=1}^{\infty} \frac{1}{n^{2(\nu+1)}} \lesssim 1.$$

Moreover, from (3.15) and the construction of $\bar{\beta}$, we have $\epsilon_{\text{sp}}(\bar{\beta}) \lesssim q^{-2\nu}$, hence (as in the Legendre proof) $F(\bar{\beta}) - F(\beta_{\text{avg}}^*) \lesssim q^{-2\nu}$, and by μ -strong convexity, $\|\bar{\beta} - \beta_{\text{avg}}^*\| \lesssim q^{-\nu}$. Thus

$$\|\beta_{\text{avg}}^*\| \leq \|\beta_{\text{avg}}^* - \bar{\beta}\| + \|\bar{\beta}\| \lesssim q^{-\nu} + 1 \lesssim 1.$$

Substituting $\|\beta_{\text{avg}}^*\| \lesssim 1$ and $\epsilon_{\text{sp}}^*(q) \lesssim q^{-2\nu}$ into the iteration bound yields

$$T \lesssim q \cdot \log(q^{2\nu}) = O(q \log q).$$

Finally, with $q \asymp \epsilon^{\frac{1}{1-2\nu}}$, we obtain

$$T = O\left(\epsilon^{\frac{1}{1-2\nu}} \log(1/\epsilon)\right),$$

which completes the proof. □

Similar to the Legendre approximation in [12, Section 4], the solution-map error decays exponentially for analytic $\theta_i^*(\cdot)$.

Lemma 3.4.4 (ϵ_{sp}^* for Analytic Periodic Solution Maps (Real Fourier Basis)). *Sup-*

¹see [45, Section 3.2.2] for a very nice and detailed discussion of the decay rates in modern language under various smoothness conditions.

pose Assumption 3.2.1 holds. Let $\Lambda = [0, 2\pi]$ and assume that for each $i = 1, \dots, d$, the function $\boldsymbol{\theta}_i^*(\cdot)$ is 2π -periodic and real analytic on Λ . Then there exist constants $C > 0$ and $\delta > 0$ (depending on $\{\boldsymbol{\theta}_i^*\}_{i=1}^d$ but not on K) such that for any $K \geq 0$ and the real Fourier basis described above,

$$\epsilon_{\text{sp}}^* \leq \frac{dL}{2} C^2 e^{-2\delta K}.$$

Equivalently, since $q = 2K + 1$,

$$\epsilon_{\text{sp}}^* \leq \frac{dL}{2} C^2 e^{-\delta(q-1)}.$$

Proof. Fix any coordinate $i \in [d]$. By [3, Theorem 1.11], there exist constants $C > 0$ and $\delta > 0$ such that for each integer $K \geq 0$,

$$\inf_{s \in \mathcal{T}_K} \sup_{\lambda \in [0, 2\pi]} |s(\lambda) - \boldsymbol{\theta}_i^*(\lambda)| \leq C e^{-\delta K}.$$

Let $s_{i,K} \in \mathcal{T}_K$ attain (or ε -attain) this infimum. Stack the coefficients of $\{s_{i,K}\}_{i=1}^d$ into $\bar{\boldsymbol{\beta}}$ so that $\boldsymbol{\Phi}(\lambda)\bar{\boldsymbol{\beta}} = (s_{i,K}(\lambda))_{i \in [d]}$.

Proceeding exactly as in the proof of Lemma 3.4.2, by L -smoothness of $h(\cdot, \lambda)$, for any $\lambda \in [0, 2\pi]$ we have

$$h((s_{i,K}(\lambda))_{i \in [d]}, \lambda) - h(\boldsymbol{\theta}^*(\lambda), \lambda) \leq \frac{L}{2} \left\| (s_{i,K}(\lambda))_{i \in [d]} - \boldsymbol{\theta}^*(\lambda) \right\|^2.$$

Taking the supremum over λ and using $\|x\|^2 \leq d\|x\|_\infty^2$ coordinatewise yields

$$\epsilon_{\text{sp}}(\bar{\boldsymbol{\beta}}) \leq \frac{dL}{2} \left(\sup_{i \in [d]} \sup_{\lambda \in [0, 2\pi]} |s_{i,K}(\lambda) - \boldsymbol{\theta}_i^*(\lambda)| \right)^2 \leq \frac{dL}{2} C^2 e^{-2\delta K}.$$

Since $\epsilon_{\text{sp}}^* \leq \epsilon_{\text{sp}}(\bar{\boldsymbol{\beta}})$ by optimality, the claim follows. Finally, substituting $q = 2K + 1$ gives the stated q -form. □

Theorem 3.4.5 (SGD for Analytic Periodic Solution Maps (Real Fourier Basis)).

Let $\Lambda = [0, 2\pi]$ and $\mathbb{P}_\lambda$ be the uniform distribution on $[0, 2\pi]$. Assume the conditions of Lemma 3.4.4 hold. Then it suffices to use

$$q = O(\log(1/\epsilon))$$

real Fourier basis functions per coordinate (i.e. $q = 2K + 1$ for some K) and run constant-step size SGD for

$$T = O(\log \log(1/\epsilon))$$

iterations to obtain an iterate β_T satisfying $\mathbb{E} [\epsilon_{\text{sp}}(\beta_T)] \leq \epsilon$.

Proof. As in the proof of Theorem 3.4.3, we focus on asymptotics as $\epsilon \rightarrow 0$ and suppress constants independent of ϵ .

Recall $q = 2K + 1$ and $p = qd$. By Lemma 3.4.1, $\frac{C_q}{c_q} = 2q$ (and $c_q = \frac{1}{2}$ for $K \geq 1$). By Lemma 3.4.4, there exist constants $C, \delta > 0$ such that

$$\epsilon_{\text{sp}}^*(q) \leq \frac{dL}{2} C^2 e^{-2\delta K}. \quad (3.16)$$

By Section 3.2, Theorem 3.3.4, it suffices that

$$\epsilon \geq \frac{8L(\bar{\eta} + 1)}{\mu} \frac{C_q}{c_q} \epsilon_{\text{sp}}^*(q) \lesssim q \cdot e^{-2\delta K}.$$

Since $q = 2K + 1 \asymp K$, the right-hand side is $Ke^{-2\delta K}$ up to constants, and thus it suffices to take

$$K \gtrsim \log(1/\epsilon) \iff q = O(\log(1/\epsilon)).$$

Again by Section 3.2, Theorem 3.3.4, the iteration count should exceed

$$T \gtrsim \frac{C_q}{c_q} \log \left(\frac{c_q \mu \|\beta_{\text{avg}}^*\|^2}{\epsilon_{\text{sp}}^*(q)} \right) \lesssim q \cdot \log \left(\frac{1}{\epsilon_{\text{sp}}^*(q)} \right),$$

where we used $\frac{C_q}{c_q} = 2q$ and absorbed constants. Using (3.16),

$$\log \left(\frac{1}{\epsilon_{\text{sp}}^*(q)} \right) = \log(\Theta(e^{2\delta K})) = \Theta(K) = \Theta(q),$$

hence $T \lesssim q \cdot q = O(q^2)$. With $q = O(\log(1/\epsilon))$ from Step 1, we obtain

$$T = O(\log \log(1/\epsilon)).$$

This completes the proof. □

Remark 3.4.6. *If we transform an aperiodic solution map on $[-1, 1]$ to a periodic one on $[0, 2\pi]$ by a change of variables $\lambda := \cos x$, then the results in this section corresponds to the results of approximating with Chebyshev polynomials in [12, Section 4] (See [3, Section 2.3]).*

3.5 From ℓ_2 to General Reference Functions

In this section, we motivate the use of Bregman geometry as a natural extension of the classical ℓ_2 geometry underlying standard first-order methods. In the Euclidean setting, smoothness and strong convexity are defined with respect to the squared ℓ_2 norm, which induces a quadratic notion of distance and leads to gradient-based updates with uniform geometry across the domain. However, many optimization problems arising in modern applications, including those with simplex constraints, multiplicative structure, or highly anisotropic curvature, are poorly aligned with this geometry, resulting in suboptimal conditioning and conservative step sizes. Bregman geometry generalizes this framework by replacing the squared ℓ_2 norm with a problem-dependent reference function, thereby inducing a tailored notion of distance via the Bregman divergence. This allows the geometry to adapt to the intrinsic structure of the domain and objective, often yielding tighter curvature characterizations. In particular, the notions of relative smoothness and relative strong convexity extend their Euclidean counterparts by measuring curvature with respect to the reference function, rather than a fixed norm. This perspective enables the design and analysis of first-order methods that remain effective even when standard Lipschitz gradient or strong convexity assumptions fail under ℓ_2 geometry, while recovering classical results as a special case when the reference function is chosen to be quadratic.

3.5.1 Relative Strong Convexity and Smoothness

Let Q be a closed convex set in a finite-dimensional vector space.

Definition 3.5.1 (Bregman Distance and Reference Function). *Let $\psi(\cdot) : Q \rightarrow \mathbb{R}$ be a given convex and differentiable function, which we refer to as the reference function. The Bregman distance $D_\psi(\cdot, \cdot)$ w.r.t. to $\psi(\cdot)$ is defined by:*

$$D_\psi(\mathbf{w}, \bar{\mathbf{w}}) := \psi(\mathbf{w}) - \psi(\bar{\mathbf{w}}) - \nabla\psi(\bar{\mathbf{w}})^\top(\mathbf{w} - \bar{\mathbf{w}}) \quad \text{for all } \mathbf{w}, \bar{\mathbf{w}} \in Q.$$

Definition 3.5.2 (Relative Smoothness). *A function $g(\cdot) : Q \rightarrow \mathbb{R}$ is said to be smooth relative to reference function $\psi(\cdot)$ if there exists constant $L > 0$ such that for all $\mathbf{w}, \bar{\mathbf{w}} \in Q$,*

$$g(\bar{\mathbf{w}}) \leq g(\mathbf{w}) + \nabla g(\mathbf{w})^\top(\bar{\mathbf{w}} - \mathbf{w}) + LD_\psi(\bar{\mathbf{w}}, \mathbf{w}).$$

Definition 3.5.3 (Relative Strong Convexity). *A function $g(\cdot) : Q \rightarrow \mathbb{R}$ is said to be strongly convex relative to reference function $\psi(\cdot)$ if there exists constant $\mu > 0$ such that for all $\mathbf{w}, \bar{\mathbf{w}} \in Q$,*

$$g(\bar{\mathbf{w}}) \geq g(\mathbf{w}) + \nabla g(\mathbf{w})^\top(\bar{\mathbf{w}} - \mathbf{w}) + \mu D_\psi(\bar{\mathbf{w}}, \mathbf{w}).$$

3.5.2 RWGC and Stochastic Mirror Descent Convergence

In this section, we develop a critical tool for solving a general stochastic optimization problem in the Bregman geometry.

Consider a family of functions $g(\cdot, z) : Q \rightarrow \mathbb{R}$, $z \sim \mathbb{P}_z$, and $G(\cdot) := \mathbb{E}_{z \sim \mathbb{P}_z} [g(\cdot, z)]$. The *stochastic reference function descent*, sometimes referred to as stochastic mirror descent, is carried out in the following manner.

Algorithm 2 Stochastic Reference Function Descent

- 1: **Initialize:** $\mathbf{w}_0 \in Q$. Let $\psi(\cdot)$ be a reference function of strong convexity, η denote stepsize.
- 2: **Return:** A sequence of iterates: $\mathbf{w}_1, \mathbf{w}_2, \dots$
- 3: **For** $t = 0, 1, \dots$ **do**
- 4: Sample the unbiased gradient estimator $\nabla g(\mathbf{w}_t, z_t)$, which satisfies $\mathbb{E} [\nabla g(\mathbf{w}_t, z_t)] = \nabla G(\mathbf{w}_t)$.
- 5: Update

$$\mathbf{w}_{t+1} \leftarrow \arg \min_{\mathbf{w} \in Q} \{ \eta \nabla g(\mathbf{w}_t, z_t)^\top (\mathbf{w} - \mathbf{w}_t) + D_\psi(\mathbf{w}, \mathbf{w}_t) \}. \quad (3.17)$$

3.5.2.1 Algorithm 2 Convergence under Generalized RWGC

The algorithm convergence happens under two key conditions: the stochastic objective being strong convexity relative to the reference function, and a generalized relaxed weak growth condition (RWGC). We give the two assumptions here and will discuss the connection between generalized RWGC and interpolation in the next subsection.

Assumption 3.5.4. Assume G is μ_G -strongly convex relative to reference function ψ , i.e. there exists $\mu_G > 0$ such that

$$G(\bar{\mathbf{w}}) \geq G(\mathbf{w}) + \nabla G(\mathbf{w})^\top (\bar{\mathbf{w}} - \mathbf{w}) + \mu_G D_\psi(\bar{\mathbf{w}}, \mathbf{w}), \quad \forall \mathbf{w}, \bar{\mathbf{w}} \in Q.$$

Denote $G^* := \min_{\mathbf{w} \in Q} G(\mathbf{w})$ as the unique optimality point of G (assuming that it exists). The following assumption defines the generalized version of the relaxed weak growth condition (RWGC) and is inspired by [25], Assumption 5.1.

Assumption 3.5.5 (Generalized Relaxed Weak Growth Condition (RWGC)).

Given $\eta > 0$, there exists constants $\rho \geq 0$ and $\sigma \geq 0$, such that for any $\bar{\mathbf{w}} \in Q$,

$$\mathbb{E}_{z \sim \mathbb{P}_z} \left[\frac{1}{\eta} \nabla g(\bar{\mathbf{w}}, z)^\top (\bar{\mathbf{w}} - \mathbf{w}_z^+) - \frac{1}{\eta^2} D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}}) \right] \leq \rho (G(\bar{\mathbf{w}}) - G^*) + \sigma^2,$$

where

$$\mathbf{w}_z^+ \leftarrow \arg \min_{\mathbf{w} \in Q} \{ \eta \nabla g(\bar{\mathbf{w}}, z)^\top (\mathbf{w} - \bar{\mathbf{w}}) + D_\psi(\mathbf{w}, \bar{\mathbf{w}}) \}.$$

Remark 3.5.6. *Multiplying the term inside the expectation by η^2 gives the mirror descent step. If we additionally assume ψ is μ_ψ -strongly convex (with the standard definition), then by substituting the pair $\bar{\mathbf{w}}, \mathbf{w}_z^+$ into the definition of strong convexity twice in different order, and then applying Cauchy-Schwarz, we obtain*

$$\|\bar{\mathbf{w}} - \mathbf{w}_z^+\| \leq \frac{1}{\mu_\psi} \|\nabla\psi(\bar{\mathbf{w}}) - \nabla\psi(\mathbf{w}_z^+)\| = \frac{1}{\mu_\psi} \|\eta\nabla g(\bar{\mathbf{w}}, z)\|.$$

The equality comes from the point of view of mirror descent originally established by [56], which says that $\nabla\psi$ maps $\bar{\mathbf{w}}$ into the mirror space, where we take a descent step at time t by going against the scaled unbiased gradient estimator $\eta\nabla g(\bar{\mathbf{w}}, z)$. Then,

$$\begin{aligned} & \mathbb{E}_{z \sim \mathbb{P}_z} \left[\frac{1}{\eta} \nabla g(\bar{\mathbf{w}}, z)^\top (\bar{\mathbf{w}} - \mathbf{w}_z^+) - \frac{1}{\eta^2} D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}}) \right] \\ & \leq \mathbb{E}_{z \sim \mathbb{P}_z} \left[\frac{1}{\eta} \nabla g(\bar{\mathbf{w}}, z)^\top (\bar{\mathbf{w}} - \mathbf{w}_z^+) \right] \\ & \leq \mathbb{E}_{z \sim \mathbb{P}_z} \left[\frac{1}{\eta} \|\nabla g(\bar{\mathbf{w}}, z)\| \|\bar{\mathbf{w}} - \mathbf{w}_z^+\| \right] \\ & \leq \mathbb{E}_{z \sim \mathbb{P}_z} \left[\frac{1}{\mu_\psi} \|\nabla g(\bar{\mathbf{w}}, z)\|^2 \right]. \end{aligned}$$

In other words, when ψ is μ_ψ -strongly convex, our RWGC assumption is implied by the standard RWGC condition with constants $\mu_\psi\rho$ and $\mu_\psi\sigma^2$.

When $\psi(\cdot) := \frac{1}{2} \|\cdot\|^2$, Assumption 3.5.5 is equivalent to the standard RWGC condition.

For this reference function, $\mu_\psi = 1$, so the standard RWGC condition implies our RWGC assumption with ρ and σ^2 .

To see the other direction, the mirror space coincides with the original space, so the increment $\bar{\mathbf{w}} - \mathbf{w}_z^+$ is no other than the scaled gradient estimator, and Assumption 3.5.5 ensures

$$\begin{aligned} & \mathbb{E}_{z \sim \mathbb{P}_z} \left[\frac{1}{\eta} \nabla g(\bar{\mathbf{w}}, z)^\top (\bar{\mathbf{w}} - \mathbf{w}_z^+) - \frac{1}{\eta^2} D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}}) \right] \\ & = \mathbb{E}_{z \sim \mathbb{P}_z} \left[\|\nabla g(\bar{\mathbf{w}}, z)\|^2 - \frac{1}{2\eta^2} \|\mathbf{w}_z^+ - \bar{\mathbf{w}}\|^2 \right] \\ & = \mathbb{E}_{z \sim \mathbb{P}_z} \left[\|\nabla g(\bar{\mathbf{w}}, z)\|^2 - \frac{1}{2} \|\nabla g(\bar{\mathbf{w}}, z)\|^2 \right] \\ & = \mathbb{E}_{z \sim \mathbb{P}_z} \left[\frac{1}{2} \|\nabla g(\bar{\mathbf{w}}, z)\|^2 \right] \\ & \leq \rho(G(\bar{\mathbf{w}}) - G^*) + \sigma^2, \end{aligned}$$

which exactly recovers the standard definition of RWGC except for a scaling factor of 2 on

the noise term σ^2 .

To develop the convergence analysis for Algorithm 2, we invoke a famous generalization of the Pythagorean identity from Hilbert space to Banach space, namely, the *three point property*. The proof of this lemma appears frequently in the literature (e.g. [25]) and is omitted here.

Lemma 3.5.7 (Three Point Property of Tseng). *Let $\psi(\cdot)$, $\varphi(\cdot) : Q \rightarrow \mathbb{R}$ be convex and differentiable. Let $D_\psi(\cdot, \cdot)$ be a Bregman distance. For some given $\bar{\mathbf{w}} \in Q$ denote*

$$\mathbf{w}^+ := \arg \min_{\mathbf{w} \in Q} \varphi(\mathbf{w}) + D_\psi(\mathbf{w}, \bar{\mathbf{w}}).$$

Then, for any $\mathbf{w} \in Q$,

$$\varphi(\mathbf{w}) + D_\psi(\mathbf{w}, \bar{\mathbf{w}}) \geq \varphi(\mathbf{w}^+) + D_\psi(\mathbf{w}^+, \bar{\mathbf{w}}) + D_\psi(\mathbf{w}, \mathbf{w}^+).$$

Now, we are ready to state the convergence result of Algorithm 2.

Theorem 3.5.8 (Convergence of Stochastic Reference Function Descent).

Suppose function $G(\cdot)$ satisfies Assumption 3.5.4, $G(\cdot)$ and $g(\cdot, z)$ satisfy Assumption 3.5.5. Performing t iterations of Algorithm 2 with $\eta = \frac{\bar{\eta}}{\rho}$, $\bar{\eta} \in (0, \min \{1, \frac{\rho}{\mu_G}\}]$, yields

$$\mathbb{E} [D_\psi(\mathbf{w}^*, \mathbf{w}_t)] \leq \left(1 - \frac{\bar{\eta}\mu_G}{\rho}\right)^t D_\psi(\mathbf{w}^*, \mathbf{w}_0) + \frac{\bar{\eta}\sigma^2}{\rho\mu_G}$$

Proof.

Set $\bar{\mathbf{w}} := \mathbf{w}_t$, $\mathbf{w}^+ := \mathbf{w}_{t+1}$, $\varphi(\mathbf{w}) := \eta \nabla g(\mathbf{w}_t, z_t)^\top (\mathbf{w} - \mathbf{w}_t)$, the Three Point Property ensures that for any $\mathbf{w} \in Q$,

$$\begin{aligned} & \eta \nabla g(\mathbf{w}_t, z_t)^\top (\mathbf{w} - \mathbf{w}_t) + D_\psi(\mathbf{w}, \mathbf{w}_t) \\ & \geq \eta \nabla g(\mathbf{w}_t, z_t)^\top (\mathbf{w}_{t+1} - \mathbf{w}_t) + D_\psi(\mathbf{w}_{t+1}, \mathbf{w}_t) + D_\psi(\mathbf{w}, \mathbf{w}_{t+1}). \end{aligned}$$

Suppose we choose $\mathbf{w} := \mathbf{w}^* = \arg \min_{\mathbf{w} \in Q} G(\mathbf{w})$, the optimality input of G . Then,

$$\begin{aligned} D_\psi(\mathbf{w}^*, \mathbf{w}_{t+1}) & \leq D_\psi(\mathbf{w}^*, \mathbf{w}_t) + \eta \nabla g(\mathbf{w}_t, z_t)^\top (\mathbf{w}^* - \mathbf{w}_t) \\ & \quad - \eta \nabla g(\mathbf{w}_t, z_t)^\top (\mathbf{w}_{t+1} - \mathbf{w}_t) - D_\psi(\mathbf{w}_{t+1}, \mathbf{w}_t) \\ & = D_\psi(\mathbf{w}^*, \mathbf{w}_t) + \eta \nabla g(\mathbf{w}_t, z_t)^\top (\mathbf{w}^* - \mathbf{w}_t) \\ & \quad + \eta \nabla g(\mathbf{w}_t, z_t)^\top (\mathbf{w}_t - \mathbf{w}_{t+1}) - D_\psi(\mathbf{w}_{t+1}, \mathbf{w}_t) \end{aligned}$$

Note that if we take $\psi(\cdot) := \frac{1}{2} \|\cdot\|^2$, then $D_\psi(\mathbf{w}, \bar{\mathbf{w}}) = \|\mathbf{w} - \bar{\mathbf{w}}\|^2$, and the above inequality becomes an equality using the decomposition of inner product.

Taking conditional expectations yields

$$\begin{aligned}
\mathbb{E} [D_\psi(\mathbf{w}^*, \mathbf{w}_{t+1}) \mid \mathbf{w}_t] &= D_\psi(\mathbf{w}^*, \mathbf{w}_t) + \eta \nabla G(\mathbf{w}_t)^\top (\mathbf{w}^* - \mathbf{w}_t) + \\
&\quad + \mathbb{E} [\eta \nabla g(\mathbf{w}_t, z_t)^\top (\mathbf{w}_t - \mathbf{w}_{t+1}) - D_\psi(\mathbf{w}_{t+1}, \mathbf{w}_t) \mid \mathbf{w}_t] \\
&\leq D_\psi(\mathbf{w}^*, \mathbf{w}_t) + \eta \nabla G(\mathbf{w}_t)^\top (\mathbf{w}^* - \mathbf{w}_t) + \\
&\quad + \eta^2 (\rho(G(\mathbf{w}_t) - G^*) + \sigma^2) \\
&\leq D_\psi(\mathbf{w}^*, \mathbf{w}_t) + \eta (G^* - G(\mathbf{w}_t) - \mu_G D_\psi(\mathbf{w}^*, \mathbf{w}_t)) + \\
&\quad + \eta^2 (\rho(G(\mathbf{w}_t) - G^*) + \sigma^2) \\
&= (1 - \eta \mu_G) D_\psi(\mathbf{w}^*, \mathbf{w}_t) + (\eta^2 \rho - \eta)(G(\mathbf{w}_t) - G^*) + \eta^2 \sigma^2.
\end{aligned} \tag{3.18}$$

The first inequality above uses generalized RWGC, setting $\bar{\mathbf{w}} := \mathbf{w}_t$ and so $\mathbf{w}_z^+ = \mathbf{w}_{t+1}$. The second inequality uses relative strong convexity.

The final inequality holds since $\eta^2 \rho - \eta = \left(\frac{\bar{\eta}}{\rho}\right)^2 \rho - \frac{\bar{\eta}}{\rho} = \left(\frac{\bar{\eta}}{\rho}\right) (\bar{\eta} - 1)$ and $\bar{\eta} \leq 1$. Furthermore, since $\bar{\eta} \leq \frac{\rho}{\mu_G}$, we have $\eta \mu_G = \left(\frac{\bar{\eta}}{\rho}\right) \mu_G \leq 1$; therefore, by iterating Equation (3.18) and taking overall expectations, we get

$$\begin{aligned}
\mathbb{E} [D_\psi(\mathbf{w}^*, \mathbf{w}_t)] &\leq (1 - \eta \mu_G)^t D_\psi(\mathbf{w}^*, \mathbf{w}_0) + \eta^2 \sigma^2 \sum_{k=0}^{t-1} (1 - \eta \mu_G)^k \\
&\leq (1 - \eta \mu_G)^t D_\psi(\mathbf{w}^*, \mathbf{w}_0) + \frac{\eta^2 \sigma^2}{1 - (1 - \eta \mu_G)} \\
&= (1 - \eta \mu_G)^t D_\psi(\mathbf{w}^*, \mathbf{w}_0) + \frac{\eta \sigma^2}{\mu_G},
\end{aligned}$$

where the second inequality uses the geometric series bound. Recalling that $\eta = \frac{\bar{\eta}}{\rho}$ completes the proof. $\square$

3.5.2.2 Interpolation Conditions and Generalized RWGC

In this section, we explore the implication relationship between interpolation conditions and the generalized RWGC condition defined in Assumption 3.5.5.

Recall that WGC corresponds to the case when $\sigma = 0$ in Assumption 3.5.5. We demonstrate that mixed interpolation is equivalent to WGC, matching the discussion of standard WGC in [53] when we take $\psi(\cdot) := \frac{1}{2} \|\cdot\|^2$.

[53] considered a finite sum objective setup, i.e. $G(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^n g_i(\mathbf{w})$. In this setting, standard WGC implies mixed interpolation, and the converse is true if we assume both uniform smoothness and convexity ([53], Section 5). We consider a general expectation

objective setup instead, i.e.

$$G(\mathbf{w}) = \mathbb{E}_{z \sim \mathbb{P}_z} [g(\mathbf{w}, z)].$$

First, we verify the forward implication between generalized WGC and mixed interpolation.

Proposition 3.5.9 (Generalized WGC Implies Interpolation).

Assume for chosen stepsize $\eta > 0$ and distribution $\mathbb{P}_z$, there exists constants $\rho \geq 0$, such that for all $\bar{\mathbf{w}} \in Q$,

$$\mathbb{E}_{z \sim \mathbb{P}_z} \left[\frac{1}{\eta} \nabla g(\bar{\mathbf{w}}, z)^\top (\bar{\mathbf{w}} - \mathbf{w}_z^+) - \frac{1}{\eta^2} D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}}) \right] \leq \rho(G(\bar{\mathbf{w}}) - G^*),$$

where

$$\mathbf{w}_z^+ \leftarrow \arg \min_{\mathbf{w} \in Q} \{ \eta \nabla g(\bar{\mathbf{w}}, z)^\top (\mathbf{w} - \bar{\mathbf{w}}) + D_\psi(\mathbf{w}, \bar{\mathbf{w}}) \}.$$

Then, $\nabla g(\mathbf{w}^*, z) = 0$ almost surely, i.e. mixed interpolation is satisfied.

Proof.

Take $\bar{\mathbf{w}} := \mathbf{w}^*$. By definition of $\mathbf{w}_z^+$, for any z ,

$$\begin{aligned} \frac{1}{\eta} \nabla g(\mathbf{w}^*, z)^\top (\mathbf{w}^* - \mathbf{w}_z^+) - \frac{1}{\eta^2} D_\psi(\mathbf{w}_z^+, \mathbf{w}^*) &= -\frac{1}{\eta^2} (\eta \nabla g(\mathbf{w}^*, z)^\top (\mathbf{w}_z^+ - \mathbf{w}^*) + D_\psi(\mathbf{w}_z^+, \mathbf{w}^*)) \\ &\leq -\frac{1}{\eta^2} (\eta \nabla g(\mathbf{w}^*, z)^\top (\mathbf{w}^* - \mathbf{w}^*) + D_\psi(\mathbf{w}^*, \mathbf{w}^*)) \\ &= 0, \end{aligned}$$

Moreover, when $\bar{\mathbf{w}} = \mathbf{w}^*$, generalized WGC implies that

$$\mathbb{E}_{z \sim \mathbb{P}_z} \left[\frac{1}{\eta} \nabla g(\mathbf{w}^*, z)^\top (\mathbf{w}^* - \mathbf{w}_z^+) - \frac{1}{\eta^2} D_\psi(\mathbf{w}_z^+, \mathbf{w}^*) \right] \leq \rho(G(\mathbf{w}^*) - G(\mathbf{w}^*)) = 0.$$

When the expectation of a non-negative random variable is non-positive, the random variable must be 0 almost surely. Thus, we get that almost surely

$$\frac{1}{\eta} \nabla g(\mathbf{w}^*, z)^\top (\mathbf{w}^* - \mathbf{w}_z^+) - \frac{1}{\eta^2} D_\psi(\mathbf{w}_z^+, \mathbf{w}^*) = 0. \quad (3.19)$$

Using the three point property with $\mathbf{w} = \bar{\mathbf{w}} := \mathbf{w}^*$, $\mathbf{w}^+ := \mathbf{w}_z^+$, one gets

$$0 \geq \eta \nabla g(\mathbf{w}^*, z)^\top (\mathbf{w}_z^+ - \mathbf{w}^*) + D_\psi(\mathbf{w}_z^+, \mathbf{w}^*) + D_\psi(\mathbf{w}^*, \mathbf{w}_z^+).$$

Multiply Equation (3.19) by η^2 and substitute into the RHS of the above inequality, we get

$$0 \geq D_\psi(\mathbf{w}^*, \mathbf{w}_z^+).$$

By the non-negativity of Bregman distance, we then conclude that $D_\psi(\mathbf{w}^*, \mathbf{w}_z^+) = 0$, which by the definition of Bregman distance, is equivalent to $\psi(\mathbf{w}^*) = \psi(\mathbf{w}_z^+) - \nabla\psi(\mathbf{w}_z^+)^\top(\mathbf{w}^* - \mathbf{w}_z^+)$. For convex and differentiable $\psi(\cdot)$, the gradient $\nabla\psi(\cdot)$ is monotone. Thus, one can easily show that the gradient has to stay the same on the line segment from $\mathbf{w}^*$ to $\mathbf{w}_z^+$, and we conclude $\nabla\psi(\mathbf{w}_z^+) = \nabla\psi(\mathbf{w}^*)$.

Furthermore, the first-order optimality condition for the unconstrained mirror step at $\bar{\mathbf{w}} = \mathbf{w}^*$ is

$$0 = \eta \nabla g(\mathbf{w}^*, z) + \nabla\psi(\mathbf{w}_z^+) - \nabla\psi(\mathbf{w}^*).$$

Using the fact that $\nabla\psi(\mathbf{w}_z^+) = \nabla\psi(\mathbf{w}^*)$ and $\eta > 0$, we conclude $\nabla g(\mathbf{w}^*, z) = 0$. □

Conversely, WGC implies mixed interpolation.

For the following propositions, we assume the existence of

$$\mathbf{w}^* \in \arg \min_{\mathbf{w} \in Q} G(\mathbf{w})$$

and

$$\mathbf{w}_z^* \in \arg \min_{\mathbf{w} \in Q} g(\mathbf{w}, z).$$

Further, denote $G^* := G(\mathbf{w}^*)$.

Proposition 3.5.10 (Interpolation Implies Generalized WGC).

Assume for $z \sim \mathbb{P}_z$, the following are satisfied almost surely:

1. $g(\cdot, z)$ is convex over Q .
2. $\nabla g(\mathbf{w}^*, z) = 0$ (mixed interpolation).
3. $g(\cdot, z)$ is uniformly L_g -smooth relative to reference function ψ , i.e.

$$g(\mathbf{w}, z) \leq g(\bar{\mathbf{w}}, z) + \nabla g(\bar{\mathbf{w}}, z)^\top(\mathbf{w} - \bar{\mathbf{w}}) + L_g D_\psi(\mathbf{w}, \bar{\mathbf{w}}), \quad \forall \mathbf{w}, \bar{\mathbf{w}} \in Q.$$

Then, for any $\eta \leq \frac{1}{L_g}$,

$$\mathbb{E}_{z \sim \mathbb{P}_z} \left[\frac{1}{\eta} \nabla g(\bar{\mathbf{w}}, z)^\top(\bar{\mathbf{w}} - \mathbf{w}_z^+) - \frac{1}{\eta^2} D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}}) \right] \leq \frac{1}{\eta} (G(\bar{\mathbf{w}}) - G^*).$$

Proof.

Directly use relative L_g -smoothness, we get almost surely

$$\begin{aligned}
\eta \nabla g(\bar{\mathbf{w}}, z)^\top (\bar{\mathbf{w}} - \mathbf{w}_z^+) - D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}}) & \\
&\leq \eta(g(\bar{\mathbf{w}}, z) - g(\mathbf{w}_z^+, z) + L_g D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}})) - D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}}) \\
&= \eta(g(\bar{\mathbf{w}}, z) - g(\mathbf{w}_z^+, z)) + (\eta L_g - 1) D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}}) \\
&= \eta(g(\bar{\mathbf{w}}, z) - g(\mathbf{w}^*, z) + g(\mathbf{w}^*, z) - g(\mathbf{w}_z^+, z)) + (\eta L_g - 1) D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}}) \\
&\leq \eta(g(\bar{\mathbf{w}}, z) - g(\mathbf{w}^*, z)) + (\eta L_g - 1) D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}})
\end{aligned}$$

The second inequality is due to the interpolation and convexity assumption of $g(\cdot, z)$. Thus, for any $\eta \leq \frac{1}{L_g}$,

$$\begin{aligned}
&\mathbb{E}_{z \sim \mathbb{P}_z} \left[\frac{1}{\eta} \nabla g(\bar{\mathbf{w}}, z)^\top (\bar{\mathbf{w}} - \mathbf{w}_z^+) - \frac{1}{\eta^2} D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}}) \right] \\
&\leq \mathbb{E}_{z \sim \mathbb{P}_z} \left[\frac{1}{\eta} (g(\bar{\mathbf{w}}, z) - g(\mathbf{w}^*, z)) + \frac{\eta L_g - 1}{\eta^2} D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}}) \right] \\
&\leq \frac{1}{\eta} (G(\bar{\mathbf{w}}) - G(\mathbf{w}^*)).
\end{aligned}$$

□

Mixed interpolation can be too strong for our proxy error problem in general. We derive a sufficient condition for the relaxed ($\sigma > 0$), non-interpolation version in the following proposition, requiring only relative smoothness.

Proposition 3.5.11 (Relative Smoothness Implies Generalized RWGC).

Assume for $z \sim \mathbb{P}_z$, the following are satisfied almost surely:

1. $g(\cdot, z)$ is convex over Q .
2. $g(\cdot, z)$ is uniformly L_g -smooth relative to reference function ψ , i.e.

$$g(\mathbf{w}, z) \leq g(\bar{\mathbf{w}}, z) + \nabla g(\bar{\mathbf{w}}, z)^\top (\mathbf{w} - \bar{\mathbf{w}}) + L_g D_\psi(\mathbf{w}, \bar{\mathbf{w}}), \quad \forall \mathbf{w}, \bar{\mathbf{w}} \in Q.$$

Then, for any $\eta \leq \frac{1}{L_g}$,

$$\begin{aligned}
&\mathbb{E}_{z \sim \mathbb{P}_z} \left[\frac{1}{\eta} \nabla g(\bar{\mathbf{w}}, z)^\top (\bar{\mathbf{w}} - \mathbf{w}_z^+) - \frac{1}{\eta^2} D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}}) \right] \\
&\leq \frac{1}{\eta} \left(G(\bar{\mathbf{w}}) - G^* + \mathbb{E}_{z \sim \mathbb{P}_z} [g(\mathbf{w}^*, z) - g(\mathbf{w}_z^*, z)] \right).
\end{aligned}$$

Proof.

Directly use relative L_g -smoothness, we get almost surely

$$\begin{aligned}
& \eta \nabla g(\bar{\mathbf{w}}, z)^\top (\bar{\mathbf{w}} - \mathbf{w}_z^+) - D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}}) \\
& \leq \eta (g(\bar{\mathbf{w}}, z) - g(\mathbf{w}_z^+, z) + L_g D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}})) - D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}}) \\
& = \eta (g(\bar{\mathbf{w}}, z) - g(\mathbf{w}_z^+, z)) + (\eta L_g - 1) D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}}) \\
& = \eta (g(\bar{\mathbf{w}}, z) - g(\mathbf{w}^*, z) + g(\mathbf{w}^*, z) - g(\mathbf{w}_z^*, z) \\
& \quad + g(\mathbf{w}_z^*, z) - g(\mathbf{w}_z^+, z)) + (\eta L_g - 1) D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}}) \\
& \leq \eta (g(\bar{\mathbf{w}}, z) - g(\mathbf{w}^*, z) + g(\mathbf{w}^*, z) - g(\mathbf{w}_z^*, z)) + (\eta L_g - 1) D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}})
\end{aligned}$$

The second inequality obtained directly by optimality of $\mathbf{w}_z^*$.

Thus, for any $\eta \leq \frac{1}{L_g}$,

$$\begin{aligned}
& \mathbb{E}_{z \sim \mathbb{P}_z} \left[\frac{1}{\eta} \nabla g(\bar{\mathbf{w}}, z)^\top (\bar{\mathbf{w}} - \mathbf{w}_z^+) - \frac{1}{\eta^2} D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}}) \right] \\
& \leq \mathbb{E}_{z \sim \mathbb{P}_z} \left[\frac{1}{\eta} (g(\bar{\mathbf{w}}, z) - g(\mathbf{w}^*, z) + g(\mathbf{w}^*, z) - g(\mathbf{w}_z^*, z)) + \frac{\eta L_g - 1}{\eta^2} D_\psi(\mathbf{w}_z^+, \bar{\mathbf{w}}) \right] \\
& \leq \mathbb{E}_{z \sim \mathbb{P}_z} \left[\frac{1}{\eta} (g(\bar{\mathbf{w}}, z) - g(\mathbf{w}^*, z) + g(\mathbf{w}^*, z) - g(\mathbf{w}_z^*, z)) \right] \\
& = \frac{1}{\eta} (G(\bar{\mathbf{w}}) - G(\mathbf{w}^*) + \mathbb{E}_{z \sim \mathbb{P}_z} [g(\mathbf{w}^*, z) - g(\mathbf{w}_z^*, z)]).
\end{aligned}$$

□

3.5.3 Stochastic Reference Function Descent to Learn an Optimization Proxy

In this section, we consider a generalization of the complexity-penalized proxy error formulation. With a slight abuse of notation, we reuse symbols from Section 3.2 by replacing the ridge-regularization term with a general reference function $\psi(\cdot)$.

3.5.3.1 Proxy-Performance Decomposition under Local Lipschitz Continuity

Fix $\gamma > 0$. Define the complexity-penalized proxy error as

$$\epsilon_{\text{sp}, \gamma}(\boldsymbol{\beta}) := \sup_{\lambda \in \Lambda} \{h(\boldsymbol{\Phi}(\lambda)\boldsymbol{\beta}, \lambda) - h(\boldsymbol{\theta}^*(\lambda), \lambda)\} + \gamma \cdot \psi(\boldsymbol{\beta}). \quad (3.20)$$

The best-in-class proxy error and the corresponding parameter vector definitions remain unchanged:

$$\epsilon_{\text{sp}, \gamma}^* := \inf_{\boldsymbol{\beta} \in \mathbb{R}^p} \epsilon_{\text{sp}, \gamma}(\boldsymbol{\beta}),$$

$$\beta_{\text{sp},\gamma}^* \in \arg \min_{\beta \in \mathbb{R}^p} \epsilon_{\text{sp},\gamma}(\beta).$$

Then, we aim to solve a generalized version of Problem (3.2):

$$\min_{\beta \in \mathbb{R}^p} \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} [h(\Phi(\tilde{\lambda})\beta, \tilde{\lambda})] + \gamma \cdot \psi(\beta), \quad (3.21)$$

Similarly, we generalize the Ridge penalty term in Equation (3.4) to $\psi(\cdot)$, producing

$$\begin{aligned} f_\gamma(\beta, \lambda) &:= h(\Phi(\lambda)\beta, \lambda) + \gamma \cdot \psi(\beta), \\ F_\gamma(\beta) &:= \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} [f_\gamma(\beta, \tilde{\lambda})] \\ &= \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} [h(\Phi(\tilde{\lambda})\beta, \tilde{\lambda})] + \gamma \cdot \psi(\beta). \end{aligned} \quad (3.22)$$

The solution of Problem (3.21) is then defined as

$$\beta_{\text{avg},\gamma}^* \in \arg \min_{\beta \in \mathbb{R}^p} F_\gamma(\beta).$$

Again, we define $f(\cdot, \lambda)$, $F(\cdot)$ as the unregularized counterparts of (3.22) when $\gamma = 0$.

In the Ridge (ℓ_2) regularized case, we assumed uniform strong convexity and smoothness of $f(\cdot, \lambda)$. In contrast, for the general case, we only assume uniform relative strong convexity of $F(\cdot)$ and restrict the growth of $f(\cdot, \lambda)$ around the optimality points of Problem (3.20) and (3.21).

Assumption 3.5.12 (Strong Convexity Relative to $\psi(\cdot)$). There exists constant $\mu_F \geq 0$ such that, for all $\lambda \in \Lambda$, $F(\cdot)$ is μ_F -strongly convex relative to $\psi(\cdot)$, i.e., for all $\beta, \bar{\beta} \in \mathbb{R}^d$,

$$\mu_F D_\psi(\beta, \bar{\beta}) \leq F(\beta) - F(\bar{\beta}) - \nabla F(\bar{\beta})^\top (\beta - \bar{\beta}).$$

Corollary 3.5.13 (Relative Strong Convexity of F_γ). *Under Assumption 3.5.12, $F_\gamma(\cdot)$ is $(\mu_F + \gamma)$ -strongly convex with respect to $\psi(\cdot)$.*

We impose a local Lipschitz-like condition on the pointwise suboptimality w.r.t. $f(\cdot, \lambda)$ within some neighborhood of $\beta_{\text{avg},\gamma}^*$ and $\beta_{\text{sp},\gamma}^*$ for every γ . Note that Bregman distance is asymmetric. We only constrain the divergence of the first-order Taylor's approximation from one side of the neighborhood center.

Definition 3.5.14 (δ -Ball in Bregman Distance). *For fixed $\beta \in \mathbb{R}^p$, $\lambda \in \Lambda$, $\delta > 0$, a δ -ball centered at $\bar{\beta}$ with respect to a general reference function $\psi(\cdot)$, is defined as*

$$B_\psi(\bar{\beta}, \delta) := \{\beta \in \mathbb{R}^p : \sqrt{D_\psi(\bar{\beta}, \beta)} < \delta\}.$$

Definition 3.5.15 (Local Lipschitz Continuity Constant). *Given a δ -ball $B_\psi(\bar{\beta}, \delta)$, the local Lipschitz continuity constant on this ball with respect to function $f(\cdot, \lambda)$, if exists, is*

defined as

$$\text{Lip}_{f(\cdot, \lambda)}(\delta) := \sup_{\beta \in B_\psi(\bar{\beta}, \delta)} \frac{|f(\beta, \lambda) - f(\bar{\beta}, \lambda)|}{\sqrt{D_\psi(\bar{\beta}, \beta)}} > 0.$$

When the reference function $\psi(\cdot)$ also has limited growth with the neighborhoods of $\beta_{\text{avg}, \gamma}^*$ and $\beta_{\text{sp}, \gamma}^*$, the local Lipschitz-like condition is transferrable to the regularized pointwise objective $f_\gamma(\cdot, \lambda)$. We shall see this in the following assumption and proposition.

Assumption 3.5.16 (Local Lipschitz Continuity Centered at $\beta_{\text{avg}, \gamma}^*$ and $\beta_{\text{sp}, \gamma}^*$).

For any $\delta > 0$, $\gamma \geq 0$, there exists the local Lipschitz constant $\text{Lip}_{f(\cdot, \lambda)}(\delta)$, $\text{Lip}_{\psi(\cdot)}(\delta) > 0$ defined in Definition 3.5.15. Therefore, for all $\lambda \in \Lambda$, and $\beta \in B_\psi(\beta_{\text{avg}, \gamma}^*, \delta)$, $\bar{\beta} \in B_\psi(\beta_{\text{sp}, \gamma}^*, \delta) \subset \mathbb{R}^p$,

$$|f(\beta, \lambda) - f(\beta_{\text{avg}, \gamma}^*, \lambda)| \leq \text{Lip}_{f(\cdot, \lambda)}(\delta) \cdot \sqrt{D_\psi(\beta_{\text{avg}, \gamma}^*, \beta)},$$

$$|\psi(\beta) - \psi(\beta_{\text{avg}, \gamma}^*)| \leq \text{Lip}_{\psi(\cdot)}(\delta) \cdot \sqrt{D_\psi(\beta_{\text{avg}, \gamma}^*, \beta)},$$

and

$$|f(\bar{\beta}, \lambda) - f(\beta_{\text{sp}, \gamma}^*, \lambda)| \leq \text{Lip}_{f(\cdot, \lambda)}(\delta) \cdot \sqrt{D_\psi(\beta_{\text{sp}, \gamma}^*, \bar{\beta})},$$

$$|\psi(\bar{\beta}) - \psi(\beta_{\text{sp}, \gamma}^*)| \leq \text{Lip}_{\psi(\cdot)}(\delta) \cdot \sqrt{D_\psi(\beta_{\text{sp}, \gamma}^*, \bar{\beta})}.$$

Proposition 3.5.17 (Local Lipschitz Continuity of f_γ). *Given Assumption 3.5.16, there exists $\text{Lip}_\gamma(\delta) := \text{Lip}_{f_\gamma(\cdot, \lambda)}(\delta) \leq \text{Lip}_{f(\cdot, \lambda)}(\delta) + \gamma \cdot \text{Lip}_{\psi(\cdot)}$ such that,*

$$|f_\gamma(\beta, \lambda) - f_\gamma(\beta_{\text{avg}, \gamma}^*, \lambda)| \leq \text{Lip}_\gamma(\delta) \cdot \sqrt{D_\psi(\beta_{\text{avg}, \gamma}^*, \beta)},$$

and

$$|f_\gamma(\beta, \lambda) - f_\gamma(\beta_{\text{sp}, \gamma}^*, \lambda)| \leq \text{Lip}_\gamma(\delta) \cdot \sqrt{D_\psi(\beta_{\text{sp}, \gamma}^*, \beta)}.$$

We can now state the proxy-performance decomposition theorem that relates the suboptimality of a feasible solution β in Problem (3.21) to its proxy error.

Theorem 3.5.18 (Suboptimality of Problem (3.21) and Proxy Error). *Let Assumptions 3.5.12 and 3.5.16 be satisfied. Then, for any $\beta \in \mathbb{R}^p$, we have*

$$\begin{aligned} \epsilon_{\text{sp}, \gamma}(\beta) &\leq \text{Lip}_\gamma(D_\psi(\beta_{\text{avg}, \gamma}^*, \beta)) \cdot \sqrt{D_\psi(\beta_{\text{avg}, \gamma}^*, \beta)} \\ &\quad + \text{Lip}_\gamma(D_\psi(\beta_{\text{sp}, \gamma}^*, \beta_{\text{avg}, \gamma}^*)) \cdot \sqrt{D_\psi(\beta_{\text{sp}, \gamma}^*, \beta_{\text{avg}, \gamma}^*)} + \epsilon_{\text{sp}, \gamma}^*. \end{aligned}$$

Proof.

For any $\lambda \in \Lambda$ and $\beta \in \mathbb{R}^d$,

$$\begin{aligned}
& h(\Phi(\lambda)\beta, \lambda) + \gamma \cdot \psi(\beta) - h(\theta^*(\lambda), \lambda) \\
&= (h(\Phi(\lambda)\beta, \lambda) + \gamma \cdot \psi(\beta) - h(\Phi(\lambda)\beta_{\text{avg},\gamma}^*, \lambda) - \gamma \cdot \psi(\beta_{\text{avg},\gamma}^*)) \\
&\quad + (h(\Phi(\lambda)\beta_{\text{avg},\gamma}^*, \lambda) + \gamma \cdot \psi(\beta_{\text{avg},\gamma}^*) - h(\Phi(\lambda)\beta_{\text{sp},\gamma}^*, \lambda) - \gamma \cdot \psi(\beta_{\text{sp},\gamma}^*)) \\
&\quad + (h(\Phi(\lambda)\beta_{\text{sp},\gamma}^*, \lambda) + \gamma \cdot \psi(\beta_{\text{sp},\gamma}^*) - h(\theta^*(\lambda), \lambda)) \\
&= (f_\gamma(\beta, \lambda) - f_\gamma(\beta_{\text{avg},\gamma}^*, \lambda)) + (f_\gamma(\beta_{\text{avg},\gamma}^*, \lambda) - f_\gamma(\beta_{\text{sp},\gamma}^*, \lambda)) \\
&\quad + (h(\Phi(\lambda)\beta_{\text{sp},\gamma}^*, \lambda) + \gamma \cdot \psi(\beta_{\text{sp},\gamma}^*) - h(\theta^*(\lambda), \lambda)).
\end{aligned}$$

First Term. Automatically by definition, $\beta \in B_\psi(\beta_{\text{avg},\gamma}^*, D_\psi(\beta_{\text{avg},\gamma}^*, \beta))$. Thus, the local Lipschitz continuity Assumption 3.5.16 and Proposition 3.5.17 provide an immediate upper bound $\text{Lip}_\gamma(D_\psi(\beta_{\text{avg},\gamma}^*, \beta)) \cdot \sqrt{D_\psi(\beta_{\text{avg},\gamma}^*, \beta)}$ on the first term.

Second Term. Again, we upper bound the second term with Assumption 3.5.16 and Proposition 3.5.17 yielding $\text{Lip}_\gamma(D_\psi(\beta_{\text{sp},\gamma}^*, \beta_{\text{avg},\gamma}^*)) \cdot \sqrt{D_\psi(\beta_{\text{sp},\gamma}^*, \beta_{\text{avg},\gamma}^*)}$.

Third Term. By definition of $\beta_{\text{sp},\gamma}^*$, the third term can be upper-bounded by $\epsilon_{\text{sp},\gamma}^*$.

Combined. Combining the individual upper bounds for each term, we obtain the inequality stated in the theorem. $\square$

Further, observe that the best-in-class approximation should be near the optimality of the surrogate expected problem. Define an auxiliary term

$$\delta_\gamma := \frac{\epsilon_{\text{sp},\gamma}^* - \gamma \cdot \psi(\beta_{\text{avg},\gamma}^*)}{\mu_F + \gamma}. \quad (3.23)$$

This serves as the upper-bound on the distance between $\beta_{\text{sp},\gamma}^*$ and $\beta_{\text{avg},\gamma}^*$ in the parameter space, as shown in the next corollary. When $\gamma \rightarrow 0$, δ_γ converges to $\epsilon_{\text{sp}}/\mu_F$.

Corollary 3.5.19 (Best-in-class Approximation Near-optimal). *The result in Theorem 3.5.18 can be further specified as*

$$\epsilon_{\text{sp},\gamma}(\beta_t) \leq \text{Lip}_\gamma(D_\psi(\beta_{\text{avg},\gamma}^*, \beta_t)) \cdot \sqrt{D_\psi(\beta_{\text{avg},\gamma}^*, \beta_t)} + \text{Lip}_\gamma(\delta_\gamma) \cdot \sqrt{\delta_\gamma} + \epsilon_{\text{sp},\gamma}^*.$$

Proof.

We simply prove the fact that

$$D_\psi(\beta_{\text{sp},\gamma}^*, \beta_{\text{avg},\gamma}^*) \leq \delta_\gamma.$$

To see why, recall that by the relative strong-convexity of $F_\gamma(\cdot)$,

$$\begin{aligned} D_\psi(\beta_{\text{sp},\gamma}^*, \beta_{\text{avg},\gamma}^*) &\leq \frac{1}{\mu_F + \gamma} (F_\gamma(\beta_{\text{sp},\gamma}^*) - F_\gamma(\beta_{\text{avg},\gamma}^*)) \\ &= \frac{1}{\mu_F + \gamma} (F_\gamma(\beta_{\text{sp},\gamma}^*) - \mathbb{E} [h(\Phi(\tilde{\lambda})\beta_{\text{avg},\gamma}^*, \tilde{\lambda})] - \gamma \cdot \psi(\beta_{\text{avg},\gamma}^*)) \\ &\leq \frac{1}{\mu_F + \gamma} (F_\gamma(\beta_{\text{sp},\gamma}^*) - \mathbb{E} [h(\theta^*(\tilde{\lambda}), \tilde{\lambda})] - \gamma \cdot \psi(\beta_{\text{avg},\gamma}^*)) \\ &\leq \frac{1}{\mu_F + \gamma} (\epsilon_{\text{sp},\gamma}^* - \gamma \cdot \psi(\beta_{\text{avg},\gamma}^*)) \\ &= \delta_\gamma. \end{aligned}$$

The second inequality leverages the optimality of $\theta^*(\cdot)$. □

3.5.3.2 Solving Problem (3.21) with Algorithm 2

Similar to the ℓ_2 case, we first verify that Problem (3.21) satisfies the generalized RWGC defined in Assumption 3.5.5, then leverage the convergence of Algorithm 2 and the proxy-performance decomposition result in combination to drive down the expected proxy error.

To this end, we need uniform smoothness of the sample objective $f(\cdot, \lambda)$ relative to $\psi(\cdot)$.

Assumption 3.5.20 (Uniform Smoothness Relative to $\psi(\cdot)$). There exists constant $L_f \geq 0$ such that, for all $\lambda \in \Lambda$, $f(\cdot, \lambda)$ is uniformly L_f -smooth relative to $\psi(\cdot)$ for all $\lambda \in \Lambda$, i.e., for all $\beta, \bar{\beta} \in \mathbb{R}^d$, and $\lambda \in \Lambda$,

$$L_f D_\psi(\beta, \bar{\beta}) \geq f(\beta, \lambda) - f(\bar{\beta}, \lambda) - \nabla f(\bar{\beta}, \lambda)^\top (\beta - \bar{\beta}).$$

Corollary 3.5.21 (Uniform Relative Smoothness of Regularized Objectives). *Under Assumption 3.5.20, $f_\gamma(\cdot, \lambda)$ is uniformly $(L_f + \gamma)$ -smooth with respect to $\psi(\cdot)$ for all $\lambda \in \Lambda$. Furthermore, $F_\gamma(\cdot)$ is $(L_f + \gamma)$ -smooth with respect to $\psi(\cdot)$.*

We now have all the required pieces for RWGC.

Lemma 3.5.22 (Problem (3.21) satisfies RWGC). *Suppose Assumption 3.5.20 holds. Then, the family of functions $f_\gamma(\cdot, \lambda)$ and the function $F_\gamma(\cdot)$, defined in Equation (3.22), satisfy the generalized relaxed weak growth condition (RWGC) with constants $\rho_\gamma = L_f + \gamma$ and*

$\sigma_\gamma^2 = (L_f + \gamma) \left(\epsilon_{\text{sp},\gamma}^* - \gamma \cdot \psi(\beta_{\tilde{\lambda},\gamma}^*) \right)$. Namely, for all $\bar{\beta} \in \mathbb{R}^p$,

$$\begin{aligned} \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_{\tilde{\lambda}}} \left[\frac{1}{\eta} \nabla f_\gamma(\bar{\beta}, \tilde{\lambda})^\top (\bar{\beta} - \beta_{\tilde{\lambda},\gamma}^+) - \frac{1}{\eta^2} D_\psi(\beta_{\tilde{\lambda},\gamma}^+, \bar{\beta}) \right] \\ \leq (L_f + \gamma) \left(F_\gamma(\bar{\beta}) - F_\gamma^* + \left(\epsilon_{\text{sp},\gamma}^* - \gamma \cdot \psi(\beta_{\tilde{\lambda},\gamma}^*) \right) \right), \end{aligned}$$

where

$$\beta_{\lambda,\gamma}^+ \leftarrow \arg \min_{\beta \in \mathbb{R}^p} \{ \eta \nabla f_\gamma(\bar{\beta}, \lambda)^\top (\beta - \bar{\beta}) + D_\psi(\beta, \bar{\beta}) \},$$

$$\beta_{\lambda,\gamma}^* \in \arg \min_{\beta \in \mathbb{R}^p} f_\gamma(\beta, \lambda).$$

Proof. Simply observe that by the optimality of $\theta^*(\lambda)$ and $\beta_{\text{sp},\gamma}^*$,

$$\begin{aligned} \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_{\tilde{\lambda}}} [f_\gamma(\beta_{\text{avg},\gamma}^*, \tilde{\lambda}) - f_\gamma(\beta_{\tilde{\lambda}}^*, \tilde{\lambda})] \\ \leq \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_{\tilde{\lambda}}} [f_\gamma(\beta_{\text{avg},\gamma}^*, \tilde{\lambda}) - h(\theta^*(\tilde{\lambda}), \tilde{\lambda}) - \gamma \cdot \psi(\beta_{\tilde{\lambda},\gamma}^*)] \\ = \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_{\tilde{\lambda}}} [f_\gamma(\beta_{\text{avg},\gamma}^*, \tilde{\lambda}) - f_\gamma(\beta_{\text{sp},\gamma}^*, \tilde{\lambda}) + f_\gamma(\beta_{\text{sp},\gamma}^*, \tilde{\lambda}) - h(\theta^*(\tilde{\lambda}), \tilde{\lambda}) - \gamma \cdot \psi(\beta_{\tilde{\lambda},\gamma}^*)] \\ = F_\gamma(\beta_{\text{avg},\gamma}^*) - F_\gamma(\beta_{\text{sp},\gamma}^*) + \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_{\tilde{\lambda}}} [f(\beta_{\text{sp},\gamma}^*, \tilde{\lambda}) - h(\theta^*(\tilde{\lambda}), \tilde{\lambda}) - \gamma \cdot \psi(\beta_{\tilde{\lambda},\gamma}^*)] \\ \leq \epsilon_{\text{sp},\gamma}^* - \gamma \cdot \psi(\beta_{\tilde{\lambda},\gamma}^*). \end{aligned}$$

Then, by Proposition 3.5.11, for any $\eta \leq \frac{1}{L_f + \gamma}$,

$$\begin{aligned} \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_{\tilde{\lambda}}} \left[\frac{1}{\eta} \nabla f_\gamma(\bar{\beta}, \tilde{\lambda})^\top (\bar{\beta} - \beta_{\tilde{\lambda},\gamma}^+) - \frac{1}{\eta^2} D_\psi(\beta_{\tilde{\lambda},\gamma}^+, \bar{\beta}) \right] \\ \leq \frac{1}{\eta} \left(F_\gamma(\bar{\beta}) - F_\gamma^* + \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_{\tilde{\lambda}}} [f_\gamma(\beta_{\text{avg},\gamma}^*, \tilde{\lambda}) - f_\gamma(\beta_{\tilde{\lambda}}^*, \tilde{\lambda})] \right) \\ \leq \frac{1}{\eta} \left(F_\gamma(\bar{\beta}) - F_\gamma^* + (\epsilon_{\text{sp},\gamma}^* - \gamma \cdot \psi(\beta_{\tilde{\lambda},\gamma}^*)) \right). \end{aligned}$$

□

Theorem 3.5.23 (Expected Proxy Error Convergence for Algorithm 2). *Suppose Assumptions 3.5.12, 3.5.16 and 3.5.20 hold. Consider running Algorithm 2 on F_γ with step size $\eta = \frac{\bar{\eta}}{L_f + \gamma}$, where $\bar{\eta} \in (0, 1]$. Then for all iteration steps $t \geq 0$,*

$$\begin{aligned} \mathbb{E} [\epsilon_{\text{sp},\gamma}(\beta_t)] &\leq \sqrt{\mathbb{E} [\text{Lip}_\gamma^2(D_\psi(\beta_{\text{avg},\gamma}^*, \beta_t))]} \cdot \sqrt{\left(1 - \frac{\bar{\eta}(\mu_F + \gamma)}{L_f + \gamma}\right)^t D_\psi(\beta_{\text{avg},\gamma}^*, \beta_0) + \bar{\eta} \cdot \delta_\gamma} \\ &\quad + \text{Lip}_\gamma(\delta_\gamma) \cdot \sqrt{\delta_\gamma} + \epsilon_{\text{sp},\gamma}^*, \end{aligned}$$

where

$$\delta_\gamma := \frac{\epsilon_{\text{sp},\gamma}^* - \gamma \cdot \psi(\beta_{\text{avg},\gamma}^*)}{\mu_F + \gamma}$$

as defined in Equation (3.23).

Proof.

By Corollary 3.5.19, take expectation on both sides with respect to randomness in the iterative process,

$$\mathbb{E} [\epsilon_{\text{sp},\gamma}(\beta_t)] \leq \mathbb{E} \left[\text{Lip}_\gamma(D_\psi(\beta_{\text{avg},\gamma}^*, \beta_t)) \cdot \sqrt{D_\psi(\beta_{\text{avg},\gamma}^*, \beta_t)} \right] + \text{Lip}_\gamma(\delta_\gamma) \cdot \sqrt{\delta_\gamma} + \epsilon_{\text{sp},\gamma}^*.$$

Apply Holder's inequality on the first term,

$$\begin{aligned} &\mathbb{E} \left[\text{Lip}_\gamma(D_\psi(\beta_{\text{avg},\gamma}^*, \beta_t)) \cdot D_\psi(\beta_{\text{avg},\gamma}^*, \beta_t) \right] \\ &\leq \sqrt{\mathbb{E} \left[|\text{Lip}_\gamma(D_\psi(\beta_{\text{avg},\gamma}^*, \beta_t))|^2 \right]} \cdot \sqrt{\mathbb{E} \left[\left(\sqrt{D_\psi(\beta_{\text{avg},\gamma}^*, \beta_t)} \right)^2 \right]} \\ &= \sqrt{\mathbb{E} [\text{Lip}_\gamma^2(D_\psi(\beta_{\text{avg},\gamma}^*, \beta_t))]} \cdot \sqrt{\mathbb{E} [D_\psi(\beta_{\text{avg},\gamma}^*, \beta_t)]}. \end{aligned}$$

Furthermore, Lemma 3.5.22 and Theorem 3.5.8 guarantee that running t iterations of Algorithm 2 on $F_\gamma(\cdot)$ yields

$$\begin{aligned} \mathbb{E} [D_\psi(\beta_{\text{avg},\gamma}^*, \beta_t)] &\leq \left(1 - \frac{\bar{\eta}(\mu_F + \gamma)}{\rho_\gamma}\right)^t D_\psi(\beta_{\text{avg},\gamma}^*, \beta_0) + \frac{\bar{\eta}\sigma_\gamma^2}{\rho_\gamma(\mu_F + \gamma)} \\ &= \left(1 - \frac{\bar{\eta}(\mu_F + \gamma)}{L_f + \gamma}\right)^t D_\psi(\beta_{\text{avg},\gamma}^*, \beta_0) + \frac{\bar{\eta}(\epsilon_{\text{sp},\gamma}^* - \gamma \cdot \psi(\beta_{\text{avg},\gamma}^*))}{\mu_F + \gamma} \\ &= \left(1 - \frac{\bar{\eta}(\mu_F + \gamma)}{L_f + \gamma}\right)^t D_\psi(\beta_{\text{avg},\gamma}^*, \beta_0) + \bar{\eta} \cdot \delta_\gamma. \end{aligned}$$

Combine the above yields

$$\begin{aligned} \mathbb{E} [\epsilon_{\text{sp},\gamma}(\beta_t)] &\leq \\ &\sqrt{\mathbb{E} [\text{Lip}_\gamma^2 (D_\psi(\beta_{\text{avg},\gamma}^*, \beta_t))]} \cdot \sqrt{\left(1 - \frac{\bar{\eta}(\mu_F + \gamma)}{L_f + \gamma}\right)^t D_\psi(\beta_{\text{avg},\gamma}^*, \beta_0) + \bar{\eta} \cdot \delta_\gamma} \\ &\quad + \text{Lip}_\gamma(\delta_\gamma) \cdot \sqrt{\delta_\gamma} + \epsilon_{\text{sp},\gamma}^*. \end{aligned}$$

□

3.5.4 Examples

3.5.4.1 State-Dependent Poisson Regression

We consider a Poisson regression problem where the underlying data-generating process depends on an observable state parameter $\lambda \in \Lambda$. As a motivating example, λ may represent external conditions such as time-of-day or weather in a traffic network, which affect observed flows.

Let $\mathbf{X} \in \mathbb{R}^{m \times d}$ be a fixed design matrix. For each state λ , we observe counts $\mathbf{y}^{(\lambda)} \in \mathbb{N}^m$ generated as

$$\mathbf{y}^{(\lambda)} \sim \text{Poisson}(\mathbf{X}\boldsymbol{\theta}^*(\lambda)),$$

where $\boldsymbol{\theta}^*(\lambda) \in \mathbb{R}_+^d$ is a state-dependent parameter.

Given observations collected at a finite set of states

$$\Lambda_{\text{train}} = \{\lambda_1, \dots, \lambda_K\},$$

we define the empirical negative log-likelihood at state λ_k as

$$h(\boldsymbol{\theta}, \lambda_k) = \sum_{i=1}^m \left(\mathbf{X}_i \boldsymbol{\theta} - \bar{y}_i^{(k)} \log(\mathbf{X}_i \boldsymbol{\theta}) \right),$$

where $\bar{\mathbf{y}}^{(k)}$ denotes the empirical mean at λ_k . Rather than learning independent parameters at each state, we aim to learn a mapping $\lambda \mapsto \boldsymbol{\theta}(\lambda)$ that generalizes beyond the observed states. At test time, we evaluate $\boldsymbol{\theta}(\lambda)$ for states $\lambda \notin \Lambda_{\text{train}}$.

Regularization. We impose a log-barrier regularization, defined as

$$\psi(\boldsymbol{\beta}) := - \sum_j \log \beta_j.$$

The choice is inspired by [5, 25], where it has been shown to serve as a legit reference function for the deterministic non-parametric Poisson inverse problem. We verify that it also works

for our elevated expected objective.

Lemma 3.5.24 (Poisson Linear Inverse Problem Satisfies Convergence Condition with Log-barrier).

Suppose we use a non-negative basis $\Phi(\lambda) : \Lambda \mapsto \mathbb{R}_+^{d \times p}$ and a non-negative parameter space $\beta \in \mathbb{R}_+^p$, then the auxiliary functions defined in (3.22) satisfy the following necessary convergence conditions for Algorithm 2:

1. $f_\gamma(\cdot, \lambda_k)$ is convex and uniformly $(L_f + \gamma)$ -smooth relative to $\psi(\cdot)$ for all $\lambda_k \in \Lambda$, for any L_f satisfying

$$L_f \geq \max_k \sum_{i=1}^m \bar{y}_i^{(k)}.$$

2. $F_\gamma(\cdot)$ is γ -strongly convex relative to ψ .

Proof. By adding the regularization, $F_\gamma(\cdot)$ is automatically γ -strongly convex relative to $\psi(\cdot)$.

It remains to confirm that $f(\cdot, \lambda_k)$ is convex and uniformly L_f -smooth relative to $\psi(\cdot)$ for all $\lambda_k \in \Lambda$. In other words, we need to verify the convexity of $L_f \cdot \psi - f$ for some L_f . We may directly leverage [5, Lemma 7] by making a simple observation that the point-wise objective in β -space is also a Poisson inverse problem.

Fix $\lambda_k \in \Lambda$, then

$$\begin{aligned} f(\beta, \lambda_k) &= h(\Phi(\lambda_k)\beta, \lambda_k) \\ &= \sum_{i=1}^m \left(\mathbf{X}_i \Phi(\lambda_k) \beta - \bar{y}_i^{(k)} \log \mathbf{X}_i \Phi(\lambda_k) \beta \right) \\ &= \sum_{i=1}^m \left(\mathbf{A}_i^{(k)} \beta - \bar{y}_i^{(k)} \log \mathbf{A}_i^{(k)} \beta \right), \end{aligned}$$

where

$$\mathbf{A}^{(k)} := \mathbf{X} \Phi(\lambda_k) \in \mathbb{R}_+^{m \times p}.$$

From [5, Lemma 7], we easily derive that $f(\cdot, \lambda_k)$ is relatively $L(\lambda_k)$ -smooth with respect to $\psi(\cdot)$ for any

$$L(\lambda_k) \geq \sum_{i=1}^m \bar{y}_i^{(k)}.$$

Therefore, $f_\gamma(\cdot, \lambda_k)$ is $(L_f + \gamma)$ -smooth relative to $\psi(\cdot)$ for any

$$L_f \geq \max_k L(\lambda_k).$$

□

3.5.4.2 Multitask D-Optimal Design

Consider a single experiment in which we distribute surveys across d different communities. Let $\boldsymbol{\theta} \in \mathbb{R}_+^d$ denote the survey allocation, where

$$\boldsymbol{\theta}_i \geq 0, \quad \sum_{i=1}^d \boldsymbol{\theta}_i = 1,$$

and $\boldsymbol{\theta}_i$ is the fraction of the total survey budget sent to community i .

Linear Regression Model for Each Task. The downstream task is to fit several different linear regressions using different subsets of features based on the survey responses.

Suppose there are S different prediction tasks, indexed by $s = 1, \dots, S$. For task s , we posit the model

$$\mathbf{y}^{(s)} = \mathbf{X}^{(s)} \mathbf{b}^{(s)} + \varepsilon^{(s)},$$

where $\mathbf{X}^{(s)}$ is the design matrix using only a subset of all features, $\mathbf{b}^{(s)}$ is the parameter vector, and the error term $\varepsilon^{(s)}$ is a normally distributed random variable centered at 0.

We want to decide the optimal proportion of surveys to send to each community. The Fisher information matrix for task s is proportional to

$$\mathbf{M}^{(s)}(\boldsymbol{\theta}) = \mathbf{X}^{(s)\top} \text{Diag}(\boldsymbol{\theta}) \mathbf{X}^{(s)}.$$

Thus, minimizing uncertainty is equivalent to minimizing $-\log \det \mathbf{M}^{(s)}(\boldsymbol{\theta})$.

Weighted D-optimal Design Objective. Assign weights to the tasks

$$\lambda^{(s)} \geq 0, \quad \sum_{s=1}^S \lambda^{(s)} = 1.$$

The sample objective with a fixed λ is

$$h(\boldsymbol{\theta}, \lambda) := \sum_{s=1}^S \lambda^{(s)} (-\log \det \mathbf{M}^{(s)}(\boldsymbol{\theta})), \quad \boldsymbol{\theta} \in \Delta_d.$$

As λ varies, the solution map selects the optimizer $\boldsymbol{\theta}^*(\lambda)$. For example, heavier weight on income prediction may push $\boldsymbol{\theta}^*(\lambda)$ toward communities with strong income-related variation.

Regularization. We impose a log-barrier regularizer, defined as

$$\psi(\boldsymbol{\beta}) := -\sum_j \log \beta_j.$$

Lemma 3.5.25 (Multitask Design Problem Satisfies Convergence Condition with Log-barrier).

Suppose the basis and parameter space satisfies that $\Phi(\lambda) : \Lambda \mapsto \mathbb{R}_+^{d \times p}$,

$$\sum_{i=1}^d \Phi(\lambda)_{ij} = 1 \quad \forall j,$$

and $\beta \in \Delta_{p,++}$. Then, the auxiliary functions defined in (3.22) satisfy the following necessary convergence conditions for Algorithm 2:

1. $f_\gamma(\cdot, \lambda)$ is convex and uniformly $(1 + \gamma)$ -smooth relative to $\psi(\cdot)$ for all $\lambda \in \Lambda$.
2. $F_\gamma(\cdot)$ is γ -strongly convex relative to ψ .

Proof. By adding the regularization, $F_\gamma(\cdot)$ is automatically γ -strongly convex relative to $\psi(\cdot)$.

We are left to confirm that $f(\cdot, \lambda)$ is convex and uniformly 1-smooth relative to $\psi(\cdot)$ for all $\lambda \in \Lambda$.

Adapting the proof of [34, Proposition 2.2], observe that the Hessian of $f(\cdot, \lambda)$ is

$$\nabla_\beta^2 f(\beta, \lambda) = \Phi(\lambda)^\top (\mathbf{A}^{(s)}(\beta, \lambda) \circ \mathbf{A}^{(s)}(\beta, \lambda)) \Phi(\lambda),$$

where $\circ$ denotes the Hadamard product and

$$\mathbf{A}^{(s)}(\beta, \lambda) := \mathbf{X}^{(s)} (\mathbf{M}^{(s)}(\Phi(\lambda)\beta))^{-1} \mathbf{X}^{(s)\top}.$$

Convexity is straightforward from the positive semi-definite Hessian.

Next, we verify relative smoothness. Since $\mathbf{A}^{(s)}(\beta, \lambda) \circ \mathbf{A}^{(s)}(\beta, \lambda) \preceq \text{Diag}^{-2}(\theta)$,

$$\Phi(\lambda)^\top (\mathbf{A}^{(s)}(\beta, \lambda) \circ \mathbf{A}^{(s)}(\beta, \lambda)) \Phi(\lambda) \preceq \Phi(\lambda)^\top \text{Diag}^{-2}(\theta(\lambda)) \Phi(\lambda) = \text{Diag}^{-1}(\beta) \mathbf{B}^\top \mathbf{B} \text{Diag}^{-1}(\beta),$$

where

$$\mathbf{B} := \text{Diag}^{-1}(\theta(\lambda)) \Phi(\lambda) \text{Diag}(\beta).$$

Observe that $\mathbf{B}$ is a block diagonal matrix with row blocks, and for all i, j ,

$$B_{ij} = \frac{\Phi_{ij}(\lambda) \beta_j}{\theta_i} \leq 1.$$

Thus, in the block-diagonal matrix $\mathbf{B}^\top \mathbf{B}$, all entries are $\in [0, 1]$, and the i th row sum is

$$\sum_j B_i^\top B_{ij} = B_i^\top \sum_j \frac{\Phi_{ij}(\lambda) \beta_j}{\theta_i} = B_i^\top,$$

where $\mathbf{B}_i^\top$ denotes the only non-zero element in row i of $\mathbf{B}^\top$.

$$\|\mathbf{B}^\top \mathbf{B}\| \leq \max_i \mathbf{B}_i^\top \leq 1 \quad \Rightarrow \quad \mathbf{B}^\top \mathbf{B} \preceq \mathbf{I}.$$

In other words,

$$\nabla_{\boldsymbol{\beta}}^2 f(\boldsymbol{\beta}, \lambda) = \boldsymbol{\Phi}(\lambda)^\top (\mathbf{A}^{(s)}(\boldsymbol{\beta}, \lambda) \circ \mathbf{A}^{(s)}(\boldsymbol{\beta}, \lambda)) \boldsymbol{\Phi}(\lambda) \preceq \text{Diag}^{-2}(\boldsymbol{\beta}) = \nabla^2 \psi(\boldsymbol{\beta}).$$

□

Solve the subproblem (3.17) in Algorithm 2 (inspired by [34]). The stochastic objective according to (3.21) is

$$\min_{\boldsymbol{\beta} \in \Delta_p} \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} \left[\sum_{s=1}^S \tilde{\lambda}^{(s)} \left(-\log \det \mathbf{M}^{(s)}(\boldsymbol{\Phi}(\tilde{\lambda})\boldsymbol{\beta}) \right) \right] + \gamma \cdot \psi(\boldsymbol{\beta}).$$

The stochastic sample gradient obtained at time t is

$$\nabla_{\boldsymbol{\beta}} f_\gamma(\boldsymbol{\beta}_t, \tilde{\lambda}_t) = \nabla_{\boldsymbol{\beta}} f(\boldsymbol{\beta}_t, \tilde{\lambda}_t) + \gamma \cdot \nabla \psi(\boldsymbol{\beta}_t),$$

where

$$\begin{aligned} f(\boldsymbol{\beta}_t, \tilde{\lambda}_t) &= \sum_{s=1}^S \tilde{\lambda}_t^{(s)} \left(-\log \det \mathbf{M}^{(s)}(\boldsymbol{\Phi}(\tilde{\lambda})\boldsymbol{\beta}) \right), \\ \nabla_{\boldsymbol{\beta}} f(\boldsymbol{\beta}_t, \tilde{\lambda}_t) &= \sum_{s=1}^S \tilde{\lambda}_t^{(s)} \boldsymbol{\Phi}(\tilde{\lambda})^\top \text{Diag}(-\mathbf{A}^{(s)}(\boldsymbol{\beta}, \lambda)). \end{aligned}$$

Define

$$\mathbf{v} := \eta \nabla_{\boldsymbol{\beta}} f_\gamma(\boldsymbol{\beta}_t, \tilde{\lambda}_t) - \nabla \psi(\boldsymbol{\beta}_t) = \eta \nabla_{\boldsymbol{\beta}} f(\boldsymbol{\beta}_t, \tilde{\lambda}_t) - (1 - \eta\gamma) \nabla \psi(\boldsymbol{\beta}_t).$$

The subproblem (3.17) can be written as

$$\min_{\boldsymbol{\beta} \in \Delta_p} \mathbf{v}^\top \boldsymbol{\beta} - \sum_j \log \beta_j,$$

and the first-order optimality conditions derived from KKT are

$$\boldsymbol{\beta} > 0, \quad \boldsymbol{\beta}^\top \mathbf{1} = 1, \quad \text{and } \mathbf{v} - \text{Diag}^{-1}(\boldsymbol{\beta}) \mathbf{1} = -\xi \mathbf{1}$$

for some scalar multiplier ξ . Given ξ , it then follows that $\beta_j = 1/(\mathbf{v}_j + \xi)$, and it remains to determine the value of the scalar ξ . [34] established that there exists ξ that satisfies the

necessary condition:

$$\sum_j \frac{1}{\mathbf{v}_j + \xi} = 1$$

for some ξ in the interval $(-\min_j \mathbf{v}_j, \infty)$.

3.6 Numerical Results

3.6.1 Moderate-Dimensional Portfolio Allocation in Euclidean Geometry

Our first experiment tests the regularized Euclidean theory on a portfolio-allocation problem with a moderately high-dimensional parameter space. The decision variable $\boldsymbol{\theta} \in \mathbb{R}^{10}$ represents the portfolio weights across the Fama-French 10 industry portfolios. We estimate the mean return vector $\boldsymbol{\mu} \in \mathbb{R}^{10}$ and covariance matrix $\Sigma \in \mathbb{R}^{10 \times 10}$ from monthly returns between August 2014 and July 2024.² For a parameter vector $\lambda \in \Lambda \subset \mathbb{R}^{12}$, we consider the sample objective

$$h(\boldsymbol{\theta}, \lambda) = -\lambda_1 \boldsymbol{\mu}^\top \boldsymbol{\theta} + \lambda_2 \boldsymbol{\theta}^\top \Sigma \boldsymbol{\theta} + \|\boldsymbol{\theta} - \lambda_{3:12}\|_2^2.$$

The first two coordinates of λ control the mean-variance tradeoff, while $\lambda_{3:12}$ encodes a reference portfolio. The last term therefore penalizes transaction costs relative to the current holding. In the implementation, $\lambda_1 \in [0.2, 1]$, $\lambda_2 \in [0, 1]$, and $\lambda_{3:12} \in [-2, 2]^{10}$.

We learn an optimization proxy $\hat{\boldsymbol{\theta}}(\lambda)$ by minimizing the regularized stochastic objective

$$\min_{\beta} \mathbb{E}_{\tilde{\lambda}} [h(\hat{\boldsymbol{\theta}}_{\beta}(\tilde{\lambda}), \tilde{\lambda})] + \frac{\gamma}{2} \|\beta\|_2^2,$$

where $\hat{\boldsymbol{\theta}}_{\beta}(\lambda) = B\phi(\lambda)$ is linear in the learned coefficient matrix B . We report the unregularized objective $\mathbb{E}[h(\hat{\boldsymbol{\theta}}_{\beta}(\tilde{\lambda}), \tilde{\lambda})]$, because this is the quantity that measures the quality of the learned decision rule after removing the optimization device introduced by γ . The expectations in the recorded training and validation curves are approximated by independent Monte Carlo samples of size 1000. Following the stepsize scaling suggested by Theorems 3.3.4 and 3.3.6, the SGD stepsize is chosen proportional to $(CL + \gamma)^{-1}$, where L is the smoothness scale of the unregularized objective and C accounts for the basis-dependent operator bound.

The experiment uses two nonlinear random feature maps, each with $q = 97$ features. The random ReLU basis is

$$\phi_{\text{ReLU}}(\lambda) = \left[1, \sqrt{\frac{2}{q-1}} \max\{0, w_1^\top \lambda + b_1\}, \dots, \sqrt{\frac{2}{q-1}} \max\{0, w_{q-1}^\top \lambda + b_{q-1}\} \right]^\top,$$

²https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html

with fixed random weights and biases. The random Fourier basis first rescales λ coordinate-wise to $[-1, 1]$ ¹² and then uses

$$\phi_{\text{RFF}}(\lambda) = \left[1, \sqrt{\frac{2}{q-1}} \cos(\omega_1^\top \tilde{\lambda} + \tau_1), \dots, \sqrt{\frac{2}{q-1}} \cos(\omega_{q-1}^\top \tilde{\lambda} + \tau_{q-1}) \right]^\top.$$

All random features are sampled once and held fixed during training.

Figures 3.1 and 3.2 show the effect of varying γ . In both bases, large values of γ produce the fastest initial reduction in the unregularized objective. This agrees with the contraction behavior predicted by the regularized theory in Theorem 3.3.6: increasing γ improves the conditioning of the stochastic objective and therefore accelerates the early transient. At the same time, larger γ imposes a stronger preference for small coefficient norm. The iterates consequently settle at a higher objective value, while smaller γ values descend more slowly but ultimately reach lower plateaus.

This bias–optimization tradeoff is the same qualitative phenomenon observed in our earlier basis-enrichment experiments [12]. There, adding basis functions reduced the approximation bias after a fixed finite-dimensional model had saturated. Here the basis is fixed, but decreasing γ plays an analogous role: once the strongly regularized model reaches its plateau, a smaller regularization parameter allows the learned path to use more of the available span of ϕ and continue decreasing the target objective. The validation panels closely track the training panels, suggesting that the plateaus are primarily driven by regularization and approximation rather than by overfitting to the Monte Carlo sample.

3.6.2 Multitask D-Optimal Design in Bregman Geometry

The second experiment tests the Bregman-geometry theory on a multitask D-optimal design problem. The decision variable $\theta \in \Delta_d$ allocates a survey budget across d communities, and the parameter $\lambda \in \Delta_S$ specifies the relative importance of S downstream regression tasks. We use the Ames Housing dataset,³ clean the selected numerical variables, and cluster the 2930 remaining homes into $d = 20$ communities by k-means. The design points are the resulting community-level feature averages.

For each task $s = 1, \dots, S$, we consider a linear model

$$y^{(s)} = X^{(s)}b^{(s)} + \varepsilon^{(s)},$$

where $X^{(s)}$ is the community-level design matrix for the features used by task s . If θ is the allocation over communities, the Fisher information matrix for task s is

$$M^{(s)}(\theta) = X^{(s)\top} \text{Diag}(\theta) X^{(s)}.$$

³<https://www.openml.org/d/42165>

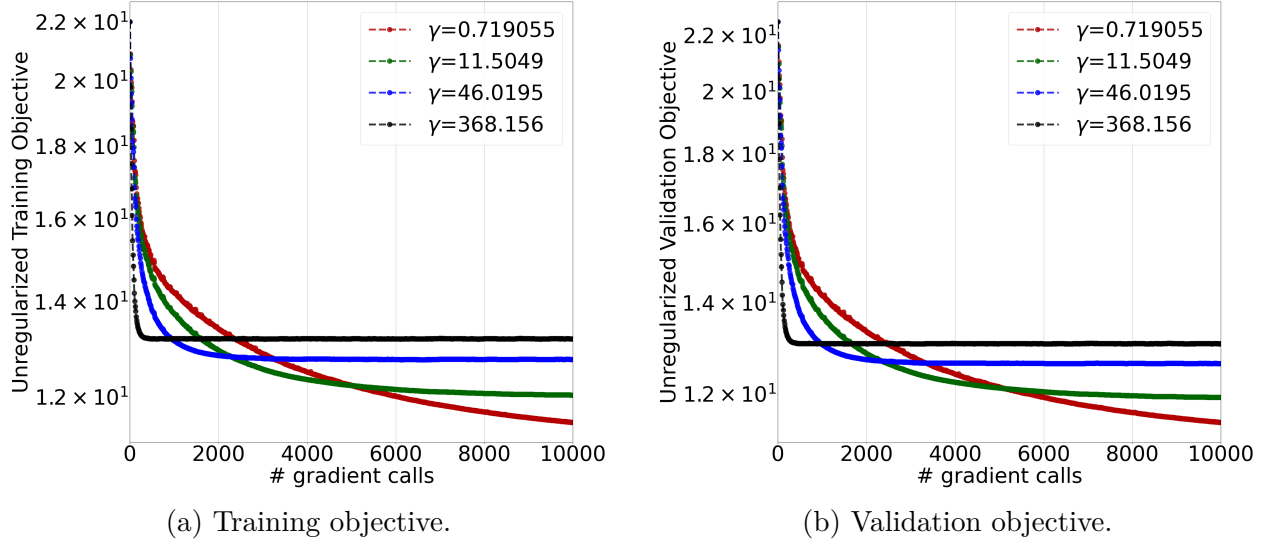


Figure 3.1: **Portfolio allocation with random ReLU features.** The curves show the unregularized stochastic objective for different values of the Euclidean regularization parameter γ . Larger γ gives faster initial convergence but plateaus at a higher objective; smaller γ converges more gradually and reaches a lower final value.

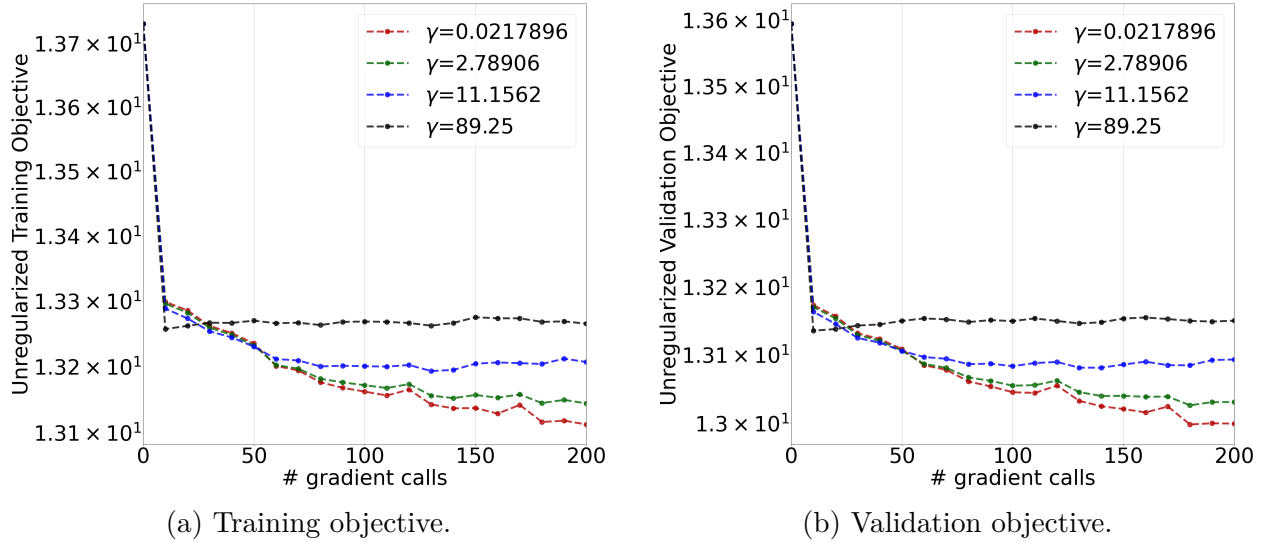


Figure 3.2: **Portfolio allocation with random Fourier features.** The same γ -dependent tradeoff appears with random Fourier features, indicating that the phenomenon is not tied to the particular random basis family.

The weighted D-optimal design objective is

$$h(\boldsymbol{\theta}, \lambda) = \sum_{s=1}^S \lambda_s \left(-\log \det M^{(s)}(\boldsymbol{\theta}) \right), \quad \boldsymbol{\theta} \in \Delta_d.$$

We use four tasks based on housing structure and valuation: predicting sale price, above-ground living area, garage area, and basement area from overlapping subsets of numerical housing attributes. These tasks are deliberately related but not identical, so changing λ shifts the allocation toward communities that are informative for different parts of the housing market.

To learn the solution map $\boldsymbol{\theta}^*(\lambda)$ while preserving the simplex constraint, we use a block-diagonal basis construction: for each component of $\boldsymbol{\theta}$, a shared collection of $q = 20$ random directions in $\mathbb{R}^S$ is evaluated at λ and normalized with a softmax. The learned coefficient vector $\boldsymbol{\beta}$ lies in the simplex Δ_{qd} , so the fitted allocation $\boldsymbol{\Phi}(\lambda)\boldsymbol{\beta}$ is also in Δ_d . We sample task weights from the uniform Dirichlet distribution,

$$\tilde{\lambda} \sim \text{Dirichlet}(1, \dots, 1),$$

and minimize the regularized stochastic objective

$$\min_{\boldsymbol{\beta} \in \Delta_{qd}} \mathbb{E}_{\tilde{\lambda}} [h(\boldsymbol{\Phi}(\tilde{\lambda})\boldsymbol{\beta}, \tilde{\lambda})] + \gamma\psi(\boldsymbol{\beta}), \quad \psi(\boldsymbol{\beta}) = -\sum_j \log \beta_j.$$

The update is the stochastic reference-function descent step derived above, with the log-barrier reference function. The reported quantity in Figure 3.3 is the regularized objective, since in the Bregman experiment the regularized geometry is the object directly controlled by the theory. In each phase, the stepsize follows the relative-smoothness prescription from Theorem 3.5.23: it is set proportional to $(1 + \gamma)^{-1}$.

The same pattern from the Euclidean portfolio experiment reappears. Larger γ values produce steeper initial descent, consistent with the relative strong convexity and relative smoothness constants. They also stabilize at higher objective values because the log-barrier term keeps $\boldsymbol{\beta}$ closer to the center of the coefficient simplex. Smaller γ values permit more expressive, more concentrated combinations of the basis elements; they therefore converge more slowly but can attain lower objective levels. The agreement between the Euclidean and Bregman experiments is the main empirical point of this section: the role of γ is not an artifact of a Hilbert-space proof, but persists under the non-Euclidean geometry required by simplex-constrained design.

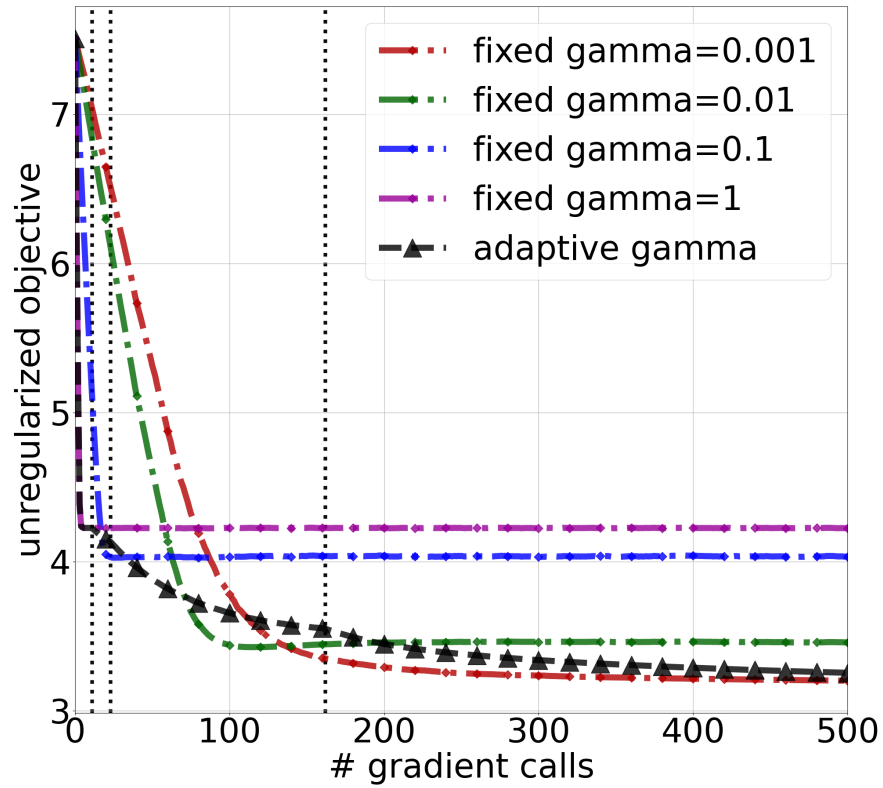


Figure 3.3: **Multitask D-optimal design.** Unregularized stochastic objective for several fixed values of γ in the log-barrier Bregman geometry. Larger γ accelerates the early iterations but reaches a higher plateau, while smaller γ descends more slowly and eventually attains a lower objective.

Chapter 4

Stochastic Steepest Descent

The first two chapters study learning optimization proxies by solving stochastic optimization problems over the coefficients of a learned decision rule. The convergence results there depend not only on the approximation power of the proxy class, but also on the geometry used by the training algorithm. Chapter 1 works in Euclidean geometry, while Chapter 2 shows that Bregman geometry can better match constrained domains and non-Euclidean curvature.

This final chapter isolates that algorithmic theme. Rather than focusing on a particular parametric optimization model, we study stochastic steepest descent under general norm geometries, with special attention to ℓ_p geometry. The goal is to understand how the choice of geometry changes descent directions, gradient noise amplification, and convergence behavior. In this sense, the chapter abstracts one of the lessons from proxy learning: the right first-order method is not only a matter of stepsize, but also of the geometry in which progress is measured.

This direction was also motivated by a possible connection between optimization proxies and boosting-style algorithms. Classical boosting builds a predictor stage by stage, repeatedly selecting a weak learner or additive component that improves the current objective [18, 49]. Later work recast boosting as additive modeling, functional gradient descent, and optimization in Bregman geometry [20, 35, 11]. A proxy-learning analogue would replace a fixed feature map by a large dictionary of candidate basis functions and train the proxy by greedily adding the component that most improves the stochastic proxy objective. From this viewpoint, boosting is not only a prediction method; it is also a template for constructing the proxy class while optimizing over it.

Chapter 3 should be read as an initial step toward that program. It does not yet propose a full boosting algorithm for optimization proxies. Instead, it studies the first-order geometry that such a method would rely on. In particular, ℓ_1 steepest descent selects a single coordinate, making it a norm-geometric analogue of stagewise weak-learner selection, while ℓ_2 steepest descent gives the dense gradient update used in the first proxy-learning theory. Understanding the stochastic behavior of these geometries is therefore a prerequisite for turning the boosting analogy into an algorithmic framework for proxy learning.

This comparison also clarifies a special feature of the Euclidean theory developed in

Chapter 1. There, stochastic gradient descent is exactly stochastic steepest descent in ℓ_2 geometry, and Theorem 2.3.2 gives a constant-step convergence guarantee under RWGC without imposing an explicit mini-batch threshold. The analysis can exploit the squared Euclidean norm and its inner-product structure to control stochastic gradients directly; in particular, the triangle-inequality step used to compare proxy coefficients, solution-map error, and stochastic-gradient error remains aligned with the same ℓ_2 geometry throughout. For general norm geometries, the stochastic steepest direction is nonlinear in the sampled gradient, and the analogous proof route introduces a batch-size threshold. A central question of this chapter is therefore whether this threshold is merely an artifact of the general-norm analysis, or whether non-Euclidean geometries genuinely require enough batching to stabilize the stochastic descent direction.

Thus the dissertation develops in three stages. Chapter 1 introduces the idea of learning an entire solution map through one stochastic problem. Chapter 2 reframes and generalizes that idea as optimization-proxy learning under broader convexity and geometry. Chapter 3 then studies the stochastic first-order geometry that underlies and extends these training methods.

4.1 Deterministic Steepest Descent

We begin by introducing deterministic ℓ_p descent, which may be viewed as a generalized steepest descent method under an arbitrary norm geometry. This framework unifies several classical first-order methods, including standard gradient descent, coordinate descent, mirror descent, and more general reference-function-based descent schemes.

4.1.1 General Steepest Descent under Arbitrary Norms

Let $\|\cdot\|$ denote an arbitrary norm on $\mathbb{R}^d$, and let $\|\cdot\|_*$ denote its dual norm:

$$\|\mathbf{g}\|_* := \sup_{\|\mathbf{x}\| \leq 1} \mathbf{g}^\top \mathbf{x}.$$

Consider the unconstrained optimization problem

$$\min_{\mathbf{w} \in \mathbb{R}^d} G(\mathbf{w}),$$

where $G : \mathbb{R}^d \rightarrow \mathbb{R}$ is differentiable.

The deterministic ℓ_p descent method is obtained by choosing the steepest descent direction induced by the norm geometry. At each iteration, the method selects a unit vector that maximizes the linearized decrease of the objective and scales the step by the dual norm of the gradient.

Algorithm 3 Deterministic Steepest Descent

-
- 1: **Initialize:** $\mathbf{w}_0 \in \mathbb{R}^d$. Let $\|\cdot\|$ be a norm on $\mathbb{R}^d$, $\|\cdot\|_*$ denote its dual norm, and $\eta > 0$ denote the stepsize.
 - 2: **Return:** A sequence of iterates: $\mathbf{w}_1, \mathbf{w}_2, \dots$
 - 3: **For** $t = 0, 1, \dots$ **do**
 - 4: Compute the gradient $\nabla G(\mathbf{w}_t)$.
 - 5: Choose a steepest descent direction

$$\mathbf{y}_t \in \arg \max_{\|\mathbf{y}\|=1} \nabla G(\mathbf{w}_t)^\top \mathbf{y}. \quad (4.1)$$

- 6: Update

$$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t - \eta \|\nabla G(\mathbf{w}_t)\|_* \mathbf{y}_t. \quad (4.2)$$

By the definition of the dual norm, the selected direction satisfies

$$\nabla G(\mathbf{w}_t)^\top \mathbf{y}_t = \|\nabla G(\mathbf{w}_t)\|_*.$$

Therefore, the update moves in the direction of steepest first-order decrease under the geometry induced by $\|\cdot\|$. This formulation corresponds to classical steepest descent under general norm geometries; see, for example, [10, 39, 6].

4.1.2 Relationship to ℓ_p Geometry

A particularly important special case is obtained by choosing

$$\|\mathbf{w}\| = \|\mathbf{w}\|_p,$$

for some $p \in [1, \infty]$. The dual norm is then

$$\|\mathbf{g}\|_* = \|\mathbf{g}\|_q, \quad \frac{1}{p} + \frac{1}{q} = 1.$$

For $1 < p < \infty$, the maximizer in (4.1) admits the closed form

$$(\mathbf{y}_t)_i = \frac{\text{sgn}((\nabla G(\mathbf{w}_t))_i) |(\nabla G(\mathbf{w}_t))_i|^{q-1}}{\|\nabla G(\mathbf{w}_t)\|_q^{q-1}}.$$

Hence the deterministic ℓ_p descent update becomes

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \|\nabla G(\mathbf{w}_t)\|_q \mathbf{y}_t.$$

This perspective reveals that the geometry of the optimization algorithm is determined by

the underlying norm.

4.1.3 Examples

Example 1: Euclidean Gradient Descent. When

$$\|\cdot\| = \|\cdot\|_2,$$

the dual norm is also ℓ_2 , and the steepest descent direction becomes

$$\mathbf{y}_t = \frac{\nabla G(\mathbf{w}_t)}{\|\nabla G(\mathbf{w}_t)\|_2}.$$

Therefore,

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \nabla G(\mathbf{w}_t),$$

which is precisely classical gradient descent.

Example 2: Greedy Coordinate Descent[55, 38]. When

$$\|\cdot\| = \|\cdot\|_1,$$

the dual norm is ℓ_∞ . The steepest descent direction selects the coordinate with largest gradient magnitude:

$$i^* \in \arg \max_i |(\nabla G(\mathbf{w}_t))_i|.$$

The update becomes

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \|\nabla G(\mathbf{w}_t)\|_\infty \operatorname{sgn}((\nabla G(\mathbf{w}_t))_{i^*}) \mathbf{e}_{i^*}.$$

Thus, greedy coordinate descent may be interpreted as steepest descent under ℓ_1 geometry. This case is the closest norm-geometric analogue of boosting-style stagewise fitting: if the coordinates index candidate basis functions or weak learners, then each step chooses the component with the largest current first-order improvement. This connection mirrors the optimization view of boosting as greedy additive modeling or gradient descent in function space [20, 35].

Example 3: ℓ_∞ Descent. When

$$\|\cdot\| = \|\cdot\|_\infty,$$

the dual norm is ℓ_1 . A steepest descent direction is given by

$$(\mathbf{y}_t)_i = \operatorname{sgn}((\nabla G(\mathbf{w}_t))_i).$$

The resulting method updates all coordinates simultaneously according to their gradient signs:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \|\nabla G(\mathbf{w}_t)\|_1 \text{sgn}(\nabla G(\mathbf{w}_t)).$$

This geometry encourages dense updates and is closely related to sign-based descent methods.

4.1.4 Relationship to Boosting and Stagewise Proxy Construction

The ℓ_1 example also connects stochastic steepest descent to the boosting motivation behind this chapter. Boosting algorithms construct predictors additively, selecting weak learners stage by stage and combining them into a final decision rule [18, 49]. A major line of work later showed that this procedure can be understood as optimization: boosting can be viewed as additive logistic modeling, as gradient descent in function space, and as an optimization procedure related to Bregman distances [20, 35, 11]. These interpretations are useful here because optimization proxies are also learned additive objects once a dictionary of basis functions has been chosen.

In the proxy-learning setting, one can imagine replacing a fixed finite feature map Φ by a large dictionary of candidate basis functions. A boosting-style proxy method would then build the map $\lambda \mapsto \Phi(\lambda)\beta$ sequentially, selecting at each iteration the basis element that most decreases the stochastic proxy objective. When the dictionary coefficients are viewed as coordinates, this is precisely the kind of sparse stagewise update induced by ℓ_1 steepest descent. Thus, the norm geometry studied in this chapter provides a first-order abstraction of the algorithmic question: how stable is stochastic greedy basis selection when the objective being optimized is a proxy-training objective rather than a classical prediction loss?

This chapter does not resolve that full boosting-for-proxies problem. Instead, it isolates a necessary ingredient. Before designing a proxy algorithm that adaptively expands its basis, we need to understand how stochastic steepest directions behave under sparse and non-Euclidean geometries. The ℓ_1 theory and experiments below should therefore be interpreted as an initial exploration of the geometry underlying a future stagewise proxy-learning method.

4.1.5 Relationship to Reference Function Descent

The deterministic steepest descent framework is closely related to reference-function descent and mirror descent. Although these two terms often lead to mathematically equivalent update rules, they usually appear in slightly different strands of the optimization literature and emphasize different aspects of the same underlying geometry.

Mirror descent originates from the classical non-Euclidean first-order optimization literature, especially the work of Nemirovski and Yudin on optimization in general normed spaces [37, 7, 39]. Its main motivation is to replace Euclidean projection with a geometry adapted to the feasible region or the structure of the problem. Given a strongly convex

mirror map ψ , mirror descent updates

$$\mathbf{w}_{t+1} = \arg \min_{\mathbf{w} \in Q} \left\{ \eta \nabla G(\mathbf{w}_t)^\top (\mathbf{w} - \mathbf{w}_t) + D_\psi(\mathbf{w}, \mathbf{w}_t) \right\},$$

where

$$D_\psi(\mathbf{w}, \bar{\mathbf{w}}) = \psi(\mathbf{w}) - \psi(\bar{\mathbf{w}}) - \nabla \psi(\bar{\mathbf{w}})^\top (\mathbf{w} - \bar{\mathbf{w}})$$

is the Bregman divergence generated by ψ . In this literature, the emphasis is typically algorithmic and geometric: the reference function ψ determines the geometry in which projection and descent are performed. For example, the Euclidean reference function $\psi(\mathbf{w}) = \frac{1}{2} \|\mathbf{w}\|_2^2$ recovers projected gradient descent, while the entropy reference function leads to multiplicative-weights-type updates on the simplex.

Reference-function descent is often used in the relative smoothness and Bregman optimization literature [5, 34]. In this setting, the reference function ψ is not only used to define a non-Euclidean proximal step, but also to characterize the curvature of the objective itself. Instead of assuming that G has a globally Lipschitz gradient in the Euclidean sense, one assumes that G is smooth relative to ψ , meaning that for some $L > 0$,

$$G(\mathbf{w}) \leq G(\bar{\mathbf{w}}) + \nabla G(\bar{\mathbf{w}})^\top (\mathbf{w} - \bar{\mathbf{w}}) + LD_\psi(\mathbf{w}, \bar{\mathbf{w}}), \quad \forall \mathbf{w}, \bar{\mathbf{w}} \in Q.$$

Under this condition, the update

$$\mathbf{w}_{t+1} = \arg \min_{\mathbf{w} \in Q} \left\{ \eta \nabla G(\mathbf{w}_t)^\top (\mathbf{w} - \mathbf{w}_t) + D_\psi(\mathbf{w}, \mathbf{w}_t) \right\}$$

can be interpreted as a natural analogue of gradient descent for objectives whose curvature is better controlled by D_ψ than by a squared Euclidean norm.

Thus, mirror descent and reference-function descent share the same basic update structure, but their emphases are different. Mirror descent is usually presented as a geometry-aware first-order method: the reference function defines the geometry of the algorithm. Reference-function descent is usually presented as a curvature-aware descent method: the reference function defines the geometry relative to which the objective is smooth or strongly convex. In this sense, mirror descent is primarily an algorithmic framework, while reference-function descent is often a modeling and analysis framework for non-Euclidean smoothness.

The deterministic steepest descent method considered here can be viewed as a norm-induced counterpart of these methods. Rather than choosing a general Bregman divergence D_ψ , we choose a norm $\|\cdot\|$ and construct the steepest descent direction under that norm. When the norm geometry is generated by an appropriate reference function, the resulting method is closely connected to mirror descent. When the objective admits smoothness or strong convexity bounds relative to that reference function, the same update can also be interpreted through the lens of reference-function descent.

When

$$\psi(\mathbf{w}) = \frac{1}{2} \|\mathbf{w}\|_2^2,$$

reference-function descent reduces to Euclidean gradient descent. More generally, different choices of ψ induce different local geometries and different descent dynamics. The ℓ_p descent framework may therefore be viewed as a norm-induced analogue of reference-function descent: the geometry is specified directly by a norm, while reference-function descent specifies the geometry through a Bregman divergence.

4.1.6 Relationship to Stochastic Gradient Descent

The stochastic ℓ_p descent method considered later in this chapter replaces the full gradient $\nabla G(\mathbf{w}_t)$ with a stochastic estimator:

$$\nabla g(\mathbf{w}_t, z_t), \quad \mathbb{E} [\nabla g(\mathbf{w}_t, z_t) \mid \mathbf{w}_t] = \nabla G(\mathbf{w}_t).$$

The resulting update becomes

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \|\nabla g(\mathbf{w}_t, z_t)\|_* \mathbf{y}_t,$$

where

$$\mathbf{y}_t \in \arg \max_{\|\mathbf{y}\|=1} \nabla g(\mathbf{w}_t, z_t)^\top \mathbf{y}.$$

When the norm is Euclidean, this reduces to classical stochastic gradient descent. Hence SGD may be interpreted as stochastic steepest descent under ℓ_2 geometry. More generally, stochastic ℓ_p descent provides a unified framework for studying stochastic first-order methods under arbitrary norm geometries.

4.1.7 Convergence under Smoothness and Strong Convexity

We now state the standard convergence guarantee for deterministic steepest descent under arbitrary norm geometries.

Assumption 4.1.1. Assume G is μ_g -strongly convex with respect to $\|\cdot\|$, i.e.

$$G(\bar{\mathbf{w}}) \geq G(\mathbf{w}) + \nabla G(\mathbf{w})^\top (\bar{\mathbf{w}} - \mathbf{w}) + \frac{\mu_g}{2} \|\bar{\mathbf{w}} - \mathbf{w}\|^2,$$

for all $\mathbf{w}, \bar{\mathbf{w}} \in \mathbb{R}^d$.

Assumption 4.1.2. Assume G is L_g -smooth with respect to $\|\cdot\|$, i.e.

$$G(\bar{\mathbf{w}}) \leq G(\mathbf{w}) + \nabla G(\mathbf{w})^\top (\bar{\mathbf{w}} - \mathbf{w}) + \frac{L_g}{2} \|\bar{\mathbf{w}} - \mathbf{w}\|^2,$$

for all $\mathbf{w}, \bar{\mathbf{w}} \in \mathbb{R}^d$.

Theorem 4.1.3 (Linear Convergence of Deterministic ℓ_p Descent). *Suppose Assumptions 4.1.1 and 4.1.2 hold. Consider Algorithm 3 with constant stepsize*

$$\eta = \frac{1}{L_g}.$$

Then,

$$G(\mathbf{w}_t) - G^* \leq \left(1 - \frac{\mu_g}{L_g}\right)^t (G(\mathbf{w}_0) - G^*),$$

where

$$G^* := \min_{\mathbf{w} \in \mathbb{R}^d} G(\mathbf{w}).$$

Proof. Using L_g -smoothness and the update rule,

$$\begin{aligned} G(\mathbf{w}_{t+1}) &\leq G(\mathbf{w}_t) + \nabla G(\mathbf{w}_t)^\top (\mathbf{w}_{t+1} - \mathbf{w}_t) + \frac{L_g}{2} \|\mathbf{w}_{t+1} - \mathbf{w}_t\|^2 \\ &= G(\mathbf{w}_t) - \eta \|\nabla G(\mathbf{w}_t)\|_* \nabla G(\mathbf{w}_t)^\top \mathbf{y}_t + \frac{L_g \eta^2}{2} \|\nabla G(\mathbf{w}_t)\|_*^2 \|\mathbf{y}_t\|^2 \\ &= G(\mathbf{w}_t) - \eta \|\nabla G(\mathbf{w}_t)\|_*^2 + \frac{L_g \eta^2}{2} \|\nabla G(\mathbf{w}_t)\|_*^2 \\ &= G(\mathbf{w}_t) - \eta \left(1 - \frac{L_g \eta}{2}\right) \|\nabla G(\mathbf{w}_t)\|_*^2. \end{aligned}$$

Setting $\eta = 1/L_g$ gives

$$G(\mathbf{w}_{t+1}) \leq G(\mathbf{w}_t) - \frac{1}{2L_g} \|\nabla G(\mathbf{w}_t)\|_*^2.$$

Strong convexity implies

$$\|\nabla G(\mathbf{w}_t)\|_*^2 \geq 2\mu_g (G(\mathbf{w}_t) - G^*).$$

Combining the two inequalities yields

$$G(\mathbf{w}_{t+1}) - G^* \leq \left(1 - \frac{\mu_g}{L_g}\right) (G(\mathbf{w}_t) - G^*),$$

and the result follows by recursion. □

4.2 Stochastic Steepest Descent

We now extend deterministic steepest descent to the stochastic setting. The deterministic method in Algorithm 3 uses the exact gradient $\nabla G(\mathbf{w}_t)$ to construct the steepest descent direction induced by the norm geometry. In many statistical learning and stochastic optimization problems, however, the objective takes the form

$$G(\mathbf{w}) := \mathbb{E}_{z \sim \mathbb{P}_z} [g(\mathbf{w}, z)],$$

and the exact gradient $\nabla G(\mathbf{w}_t)$ may be unavailable or computationally expensive to evaluate. Instead, we observe a stochastic gradient estimator $\nabla g(\mathbf{w}_t, z_t)$ satisfying

$$\mathbb{E} [\nabla g(\mathbf{w}_t, z_t) \mid \mathbf{w}_t] = \nabla G(\mathbf{w}_t).$$

The stochastic steepest descent method replaces the deterministic gradient in the steepest descent update with this stochastic gradient estimator.

Algorithm 4 Stochastic Steepest Descent

- 1: **Initialize:** $\mathbf{w}_0 \in \mathbb{R}^d$. Let $\|\cdot\|$ be a norm on $\mathbb{R}^d$, $\|\cdot\|_*$ denote its dual norm, and $\eta > 0$ denote the stepsize.
- 2: **Return:** A sequence of iterates: $\mathbf{w}_1, \mathbf{w}_2, \dots$
- 3: **For** $t = 0, 1, \dots$ **do**
- 4: Sample the unbiased gradient estimator $\nabla g(\mathbf{w}_t, z_t)$, which satisfies

$$\mathbb{E} [\nabla g(\mathbf{w}_t, z_t) \mid \mathbf{w}_t] = \nabla G(\mathbf{w}_t).$$

- 5: Choose a stochastic steepest descent direction

$$\mathbf{y}_t \in \arg \max_{\|\mathbf{y}\|=1} \nabla g(\mathbf{w}_t, z_t)^\top \mathbf{y}. \quad (4.3)$$

- 6: Update

$$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t - \eta \|\nabla g(\mathbf{w}_t, z_t)\|_* \mathbf{y}_t. \quad (4.4)$$

When $\|\cdot\| = \|\cdot\|_2$, the dual norm is also ℓ_2 , and the stochastic steepest descent direction is

$$\mathbf{y}_t = \frac{\nabla g(\mathbf{w}_t, z_t)}{\|\nabla g(\mathbf{w}_t, z_t)\|_2}.$$

Therefore, Algorithm 4 reduces to the classical SGD update

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \nabla g(\mathbf{w}_t, z_t).$$

Thus, stochastic steepest descent may be interpreted as stochastic gradient descent under a

general norm geometry.

This Euclidean reduction is more than a notational special case. In the ℓ_2 setting, the stochastic steepest direction and its dual-norm scaling collapse to the stochastic gradient itself, so the RWGC analysis in Theorem 2.3.2 can work directly with $\mathbb{E} [\|\nabla g(\mathbf{w}_t, z_t)\|_2^2]$ and obtain a single-sample constant-step guarantee. The same Euclidean geometry supports the triangle-inequality comparisons used in Chapter 1 to pass from coefficient error to solution-path error. In a general norm, the method first transforms the sampled gradient into a steepest direction through the dual norm. This nonlinear transformation makes the error in the stochastic direction harder to control, and our analysis below pays for that difficulty through a mini-batch condition.

4.2.1 Convergence under Smoothness, Strong Convexity, and Mini-Batch RWGC

Consider a family of functions $g(\cdot, z) : \mathbb{R}^d \rightarrow \mathbb{R}$, $z \sim \mathbb{P}_z$, and $G(\cdot) := \mathbb{E}_{z \sim \mathbb{P}_z} [g(\cdot, z)]$. Also consider a pair of norm $\|\cdot\|$ and its dual norm $\|\cdot\|_*$.

Assumption 4.2.1. Assume G is μ_g strongly-convex w.r.t. an arbitrary norm $\|\cdot\|$, i.e.

$$G(\bar{\mathbf{w}}) \geq G(\mathbf{w}) + \nabla G(\mathbf{w})^\top (\bar{\mathbf{w}} - \mathbf{w}) + \frac{\mu_g}{2} \|\bar{\mathbf{w}} - \mathbf{w}\|^2, \quad \forall \mathbf{w}, \bar{\mathbf{w}} \in \mathbb{R}^p.$$

Then, we may denote $G^* := \min_{\mathbf{w} \in \mathbb{R}^d} G(\mathbf{w})$ as the unique optimality point of G .

Assumption 4.2.2. Assume G is L_g -smooth w.r.t. the dual norm $\|\cdot\|_*$, i.e.

$$\|\nabla G(\mathbf{w}) - \nabla G(\bar{\mathbf{w}})\|_* \leq L_g \|\mathbf{w} - \bar{\mathbf{w}}\|_*, \quad \forall \mathbf{w}, \bar{\mathbf{w}} \in \mathbb{R}^p.$$

Assumption 4.2.3 (Mini-batch RWGC). If we sample a mini-batch of independent random variables $z_1, \dots, z_B \sim \mathbb{P}_z$, then there exist constants $\rho \geq 0$ and $\sigma \geq 0$ such that for all $\mathbf{w} \in \mathbb{R}^d$ and all $B \in \mathbb{N}$,

$$\begin{aligned} \mathbb{E}_{z_k \sim \mathbb{P}_z} \left[\left\| \frac{1}{B} \sum_{k=1}^B \nabla g(\mathbf{w}, z_k) \right\|_*^2 \right] &\leq \frac{\kappa}{B} (2\rho(G(\mathbf{w}) - G^*) + \sigma^2), \\ \mathbb{E}_{z_k \sim \mathbb{P}_z} \left[\left\| \frac{1}{B} \sum_{k=1}^B \nabla g(\mathbf{w}, z_k) - \nabla G(\mathbf{w}) \right\|_*^2 \right] &\leq \frac{\kappa}{B} (2\rho(G(\mathbf{w}) - G^*) + \sigma^2). \end{aligned}$$

The statement in this assumption is true by norm equivalence. The question becomes what is the smallest value of κ . If κ is order d , then the batch size would need to grow in some order of d . In this sense, Assumption 4.2.3 is the general-norm analogue of the RWGC condition used for Euclidean SGD in Chapter 1, but it is not yet as sharp. The

Euclidean result avoids an explicit threshold because the stochastic update remains linear in the sampled gradient. For general norms, it remains open whether one can bypass the batch-size requirement with a more refined argument, or whether the threshold reflects an intrinsic instability of stochastic steepest directions outside ℓ_2 geometry.

Theorem 4.2.4 (Convergence of General Stochastic Steepest Descent). *Consider the general stochastic steepest descent algorithm, initialized at $\mathbf{w}_0 \in \mathbb{R}^d$, with iterations*

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \|\nabla g(\mathbf{w}_t, z_t)\|_* \mathbf{y}_t$$

where with a slight abuse of notation,

$$\nabla g(\mathbf{w}_t, z_t) := \frac{1}{B} \sum_{k=1}^B \nabla g(\mathbf{w}, z_{t,k}),$$

$$\mathbf{y}_t \in \arg \max_{\|\mathbf{y}\|=1} \nabla g(\mathbf{w}_t, z_t)^\top \mathbf{y},$$

$z_t \sim \mathbb{P}_z$ is a random sample, and $\eta := \eta_B = \frac{1}{L} \left(1 - \frac{\rho}{B\mu}\right)$ is the step-size. Then for all $t \geq 0$, we have

$$\mathbb{E}[G(\mathbf{w}_t)] - G^* \leq (1 - \gamma_B)^T (\mathbb{E}[G(\mathbf{w}_0)] - G^*) + \frac{\eta_B \sigma^2}{B \cdot \gamma_B}.$$

where $G^* := \min_{\mathbf{w} \in \mathbb{R}^d} G(\mathbf{w})$ denotes the unique optimality output of G , and $\gamma_B = \frac{\mu}{L} \left(1 - \frac{\rho}{B\mu}\right)^2$.

Remark 4.2.5 (Batch size and convergence regimes). *The theorem identifies the mini-batch size as a stability parameter for stochastic steepest descent. In particular, the contraction factor*

$$\gamma_B = \frac{\mu}{L} \left(1 - \frac{\rho}{B\mu}\right)^2$$

is positive only when

$$B > \frac{\rho}{\mu}.$$

Thus, when the batch size is below this threshold, the bound does not certify contraction, and the stochastic direction may be too noisy to yield stable descent. Consequently, one should expect three qualitative regimes: for very small B , the method can be unstable or divergent; for intermediate B , the iterates may make progress but exhibit noise-dominated or apparently sublinear behavior; for sufficiently large B , the method enters the linear convergence regime predicted by the contraction term.

This regime description should not be interpreted as saying that the finite-time error is monotone in B . For example, in the ℓ_1 geometry, the update is coordinate-wise and the mini-batch gradient determines which coordinate is selected. A moderate batch size can introduce enough randomness to explore useful coordinates early in the run, while the deterministic

full-gradient coordinate rule may be more conservative. Therefore a smaller stochastic batch may decrease the objective faster over an initial time window even when the larger batch exhibits the cleaner asymptotic linear regime.

The contrast with Theorem 2.3.2 is important. In ℓ_2 geometry, stochastic steepest descent is ordinary SGD, and RWGC gives a clean single-sample guarantee up to the usual approximation-dependent error floor. The present theorem shows that the same conclusion does not follow automatically once the geometry changes: our general-norm proof certifies contraction only after the mini-batch is large enough to control the stochastic steepest direction. Whether this gap can be closed for arbitrary norms is an open problem. The numerical evidence below suggests that, at least for ℓ_1 geometry, the batch-size effect is not only a proof artifact.

For the longer-term proxy-learning goal, the ℓ_1 case is especially important. A boosting-style proxy algorithm would likely use a large dictionary and update only a few active basis functions at a time. Such updates are naturally sparse and stagewise, but their stochastic stability is governed by the same coordinate-selection mechanism analyzed here.

Proof.

Using L_g -smoothness of G ,

$$\begin{aligned}
G(\mathbf{w}_{t+1}) &\leq G(\mathbf{w}_t) + \nabla G(\mathbf{w}_t)^\top (\mathbf{w}_{t+1} - \mathbf{w}_t) + \frac{L_g}{2} \|\mathbf{w}_{t+1} - \mathbf{w}_t\|^2 \\
&= G(\mathbf{w}_t) - \eta \nabla G(\mathbf{w}_t)^\top (\|\nabla g(\mathbf{w}_t, z_t)\|_* \mathbf{y}_t) + \frac{L_g \eta^2}{2} \|\nabla g(\mathbf{w}_t, z_t)\|_*^2 \|\mathbf{y}_t\|^2 \\
&= G(\mathbf{w}_t) - \eta \|\nabla g(\mathbf{w}_t, z_t)\|_* \nabla G(\mathbf{w}_t)^\top \mathbf{y}_t + \frac{L_g \eta^2}{2} \|\nabla g(\mathbf{w}_t, z_t)\|_*^2 \\
&= G(\mathbf{w}_t) - \eta \|\nabla g(\mathbf{w}_t, z_t)\|_* \nabla g(\mathbf{w}_t, z_t)^\top \mathbf{y}_t + \frac{L_g \eta^2}{2} \|\nabla g(\mathbf{w}_t, z_t)\|_*^2 \\
&\quad + \eta \|\nabla g(\mathbf{w}_t, z_t)\|_* (\nabla g(\mathbf{w}_t, z_t) - \nabla G(\mathbf{w}_t))^\top \mathbf{y}_t \\
&= G(\mathbf{w}_t) + \frac{\eta(L_g \eta - 2)}{2} \|\nabla g(\mathbf{w}_t, z_t)\|_*^2 \\
&\quad + \eta \|\nabla g(\mathbf{w}_t, z_t)\|_* (\nabla g(\mathbf{w}_t, z_t) - \nabla G(\mathbf{w}_t))^\top \mathbf{y}_t \\
&= G(\mathbf{w}_t) - \frac{\eta(2 - L_g \eta)}{2} \|\nabla g(\mathbf{w}_t, z_t)\|_*^2 \\
&\quad + \eta \|\nabla g(\mathbf{w}_t, z_t)\|_* \|\nabla g(\mathbf{w}_t, z_t) - \nabla G(\mathbf{w}_t)\|_*
\end{aligned}$$

Assume $L_g \eta < 2$. Take a conditional expectation w.r.t. $\mathcal{F}_t$ and apply Jensen's inequality,

yielding

$$\begin{aligned}
\mathbb{E}[G(\mathbf{w}_{t+1}) \mid \mathcal{F}_t] &\leq G(\mathbf{w}_t) - \frac{\eta(2 - L_g\eta)}{2} \mathbb{E}[\|\nabla g(\mathbf{w}_t, z_t)\|_*^2 \mid \mathcal{F}_t] \\
&\quad + \eta \mathbb{E}[\|\nabla g(\mathbf{w}_t, z_t)\|_* \|\nabla g(\mathbf{w}_t, z_t) - \nabla G(\mathbf{w}_t)\|_* \mid \mathcal{F}_t] \\
&\leq G(\mathbf{w}_t) - \frac{\eta(2 - L_g\eta)}{2} \|\nabla G(\mathbf{w}_t)\|_*^2 + \eta s_t,
\end{aligned} \tag{4.5}$$

where

$$\begin{aligned}
s_t &:= \mathbb{E}[\|\nabla g(\mathbf{w}_t, z_t)\|_* \|\nabla g(\mathbf{w}_t, z_t) - \nabla G(\mathbf{w}_t)\|_* \mid \mathcal{F}_t] \\
&\leq \sqrt{\mathbb{E}[\|\nabla g(\mathbf{w}_t, z_t)\|_*^2 \mid \mathcal{F}_t]} \cdot \sqrt{\mathbb{E}[\|\nabla g(\mathbf{w}_t, z_t) - \nabla G(\mathbf{w}_t)\|_*^2 \mid \mathcal{F}_t]} \\
&\leq \frac{\kappa}{B} (2\rho(G(\mathbf{w}_t) - G^*) + \sigma^2).
\end{aligned} \tag{4.6}$$

The first inequality above is a result of Hölder's inequality and the second uses Assumption 4.2.3.

Furthermore, strong-convexity of G gives us the following

$$\begin{aligned}
G^* &\geq G(\mathbf{w}) - \sup_{\bar{\mathbf{w}} \in \mathbb{R}^d} \left\{ (-\nabla G(\mathbf{w}))^\top (\bar{\mathbf{w}} - \mathbf{w}) - \frac{\mu_g}{2} \|\bar{\mathbf{w}} - \mathbf{w}\|^2 \right\} \\
&= G(\mathbf{w}) - \left(\frac{\mu_g}{2} \|\cdot\|^2 \right)^* (-\nabla G(\mathbf{w})) \\
&= G(\mathbf{w}) - \frac{1}{2\mu_g} \|\nabla G(\mathbf{w})\|_*^2 \\
\Rightarrow \quad -\|\nabla G(\mathbf{w}_t)\|_*^2 &\leq -2\mu_g (G(\mathbf{w}_t) - G^*).
\end{aligned} \tag{4.7}$$

Substitute Equation (4.6) and Equation (4.7) into Equation (4.5), then subtract G^* from both sides, we obtain

$$\mathbb{E}[G(\mathbf{w}_{t+1}) \mid \mathcal{F}_t] - G^* \leq \left(1 - \mu_g\eta(2 - L_g\eta) + 2\frac{\kappa\eta\rho}{B} \right) (G(\mathbf{w}_t) - G^*) + \frac{\kappa\eta\sigma^2}{B}$$

Now take an overall expectation on both sides:

$$\mathbb{E}[G(\mathbf{w}_{t+1})] - G^* \leq \left(1 - \eta \left(2\mu_g - L_g\mu_g\eta - 2\frac{\kappa\rho}{B} \right) \right) (\mathbb{E}[G(\mathbf{w}_t)] - G^*) + \frac{\kappa\eta\sigma^2}{B}. \tag{4.8}$$

Define the following variables:

$$\Delta_t := \mathbb{E}[G(\mathbf{w}_t)] - G^*,$$

Suppose Assumption 4.2.3 holds and assume we are using an increasing batch size B_t . Define:

$$S_t := \frac{\eta_t \sigma^2}{B_t},$$

$$\gamma_t := \eta_t \left(2\mu - \eta_t \mu L - \frac{2\rho}{B_t} \right),$$

We want to make the γ_t a shrinking sequence. We can do so by define a shrinking sequence of η_t . This way, the noise term S_t will also shrink. Also define:

$$\Gamma_t := \begin{cases} 1, & t = 0, \\ (1 - \gamma_{t-1})\Gamma_{t-1}, & t > 0. \end{cases}$$

We may rewrite Equation (4.8) in the following form:

$$\Delta_{t+1} \leq (1 - \gamma_t)\Delta_t + S_t.$$

We can prove the following using the same line of deduction in Lemma 1 of [30]:

$$\Delta_T \leq \Gamma_T \Delta_0 + \Gamma_T \sum_{t=1}^T \frac{S_{t-1}}{\Gamma_t}, \quad (4.9)$$

where

$$\Gamma_T \sum_{t=1}^T \frac{S_{t-1}}{\Gamma_t} = \sum_{t=1}^T \left(\prod_{k=t}^{T-1} (1 - \eta_k (2\mu - \eta_k \mu L - \frac{2\rho}{B_k})) \right) \cdot \frac{\eta_{t-1} \sigma^2}{B_{t-1}}.$$

Consider the case $\eta_t := \eta$, $B_t = B$ for all t , i.e. we use fixed step-size and batch-size. Then,

$$\gamma_t = \gamma = \eta \left(2\mu - \eta \mu L - \frac{2\rho}{B} \right),$$

$$S_t = \frac{\eta \sigma^2}{B},$$

$$\Gamma_t = (1 - \gamma)^t.$$

The key to making Equation (4.8) a contraction is to ensure $\gamma \in (0, 1]$.

For any fixed $B > 0$, γ is maximized at $\eta_B := \frac{1}{L} \left(1 - \frac{\rho}{B\mu}\right)$, which yields

$$\begin{aligned}\gamma_B &= \frac{1}{L} \left(1 - \frac{\rho}{B\mu}\right) \left(2\mu - \frac{1}{L} \left(1 - \frac{\rho}{B\mu}\right) \cdot \mu L - \frac{2\rho}{B}\right) \\ &= \frac{\mu}{L} \left(1 - \frac{\rho}{B\mu}\right)^2.\end{aligned}$$

We want to choose η_B as our stepsize and ensure

$$0 < \gamma_B = \frac{\mu}{L} \left(1 - \frac{\rho}{B\mu}\right)^2 \leq 1.$$

Since $0 < \frac{\mu}{L} \leq 1$, the following is sufficient:

$$0 < 1 - \frac{\rho}{B\mu} \leq 1 \quad \Leftrightarrow \quad B > \frac{\rho}{\mu}.$$

Then, Equation (4.9) becomes

$$\begin{aligned}\mathbb{E}[G(\mathbf{w}_T)] - G^* &\leq (1 - \gamma_B)^T (\mathbb{E}[G(\mathbf{w}_0)] - G^*) + \sum_{t=1}^T (1 - \gamma_B)^{T-t} \left(\frac{\eta_B \sigma^2}{B}\right) \\ &\leq (1 - \gamma_B)^T (\mathbb{E}[G(\mathbf{w}_0)] - G^*) + \sum_{t=1}^{\infty} (1 - \gamma_B)^t \left(\frac{\eta_B \sigma^2}{B}\right) \\ &= (1 - \gamma_B)^T (\mathbb{E}[G(\mathbf{w}_0)] - G^*) + \frac{\eta_B \sigma^2}{B \cdot \gamma_B}.\end{aligned}$$

□

4.3 Numerical Results

We now illustrate the preceding convergence behavior in a controlled least-squares experiment. The goal is to compare stochastic steepest descent under the ℓ_1 and ℓ_2 geometries as the mini-batch size varies, and to relate the observed behavior to the batch-size regimes described in Remark 4.2.5. The comparison is motivated by two questions from the proxy-learning program. First, the Euclidean guarantee in Theorem 2.3.2 shows that ℓ_2 stochastic steepest descent can inherit the usual RWGC analysis for SGD, while it is unclear whether a comparable single-sample guarantee should hold for non-Euclidean steepest descent. Second, a boosting-style proxy method would resemble a stochastic ℓ_1 or coordinate-wise procedure, because each iteration would select one basis function or weak learner to add to the current proxy.

We generate an interpolable linear regression instance with $n = 100$ samples and $d = 20$ features. The design matrix has independent standard normal entries, $X \in \mathbb{R}^{n \times d}$, and the response is generated as

$$y = X\mathbf{w}^*,$$

where $\mathbf{w}^* \in \mathbb{R}^d$ is fixed. Thus the minimum empirical loss is zero. The objective is

$$G(\mathbf{w}) = \frac{1}{2n} \|X\mathbf{w} - y\|_2^2,$$

The Hessian is therefore

$$H = \frac{1}{n} X^\top X.$$

For the ℓ_2 steepest descent geometry, we use the smoothness parameter

$$L_2 = \lambda_{\max}(H).$$

For the ℓ_1 geometry, the steepest descent step updates the coordinate with largest absolute gradient, and we use the corresponding coordinate-wise quadratic smoothness parameter

$$L_1 = \max_{1 \leq j \leq d} H_{jj}.$$

The stepsizes are chosen according to the scaling suggested by the theorem:

$$\eta_1 = \frac{\alpha}{2L_1}, \quad \eta_2 = \frac{\alpha}{2L_2},$$

where $\alpha > 0$ is a fixed damping factor. For each mini-batch size $B \in \{1, 5, 20, 100\}$, the mini-batch is sampled with replacement at every iteration. We also include a deterministic coordinate descent benchmark, which uses the full empirical gradient and the same ℓ_1 coordinate selection rule.

Figure 4.1 shows that the effect of the mini-batch size is strongly geometry-dependent. For the ℓ_2 method, increasing the batch size rapidly stabilizes the trajectory and produces a clear linear decay on the logarithmic scale. For the ℓ_1 method, the behavior is less monotone: the smallest batch is unstable in this experiment, while intermediate and large batches both converge. In particular, the run with $B = 20$ can initially decrease the loss faster than the run with $B = 100$. This does not contradict Remark 4.2.5; rather, it reflects the distinction between finite-time transient behavior and the eventual linear regime. Because the ℓ_1 step selects a single coordinate, moderate stochasticity in the mini-batch gradient can change the coordinate selection and improve early exploration, whereas the full-gradient coordinate rule is deterministic.

Figures 4.2 and 4.3 separate the two geometries. The ℓ_2 plot is consistent with the standard intuition that variance reduction from larger mini-batches improves stability and reveals the linear contraction predicted by the deterministic part of the theorem. The ℓ_1

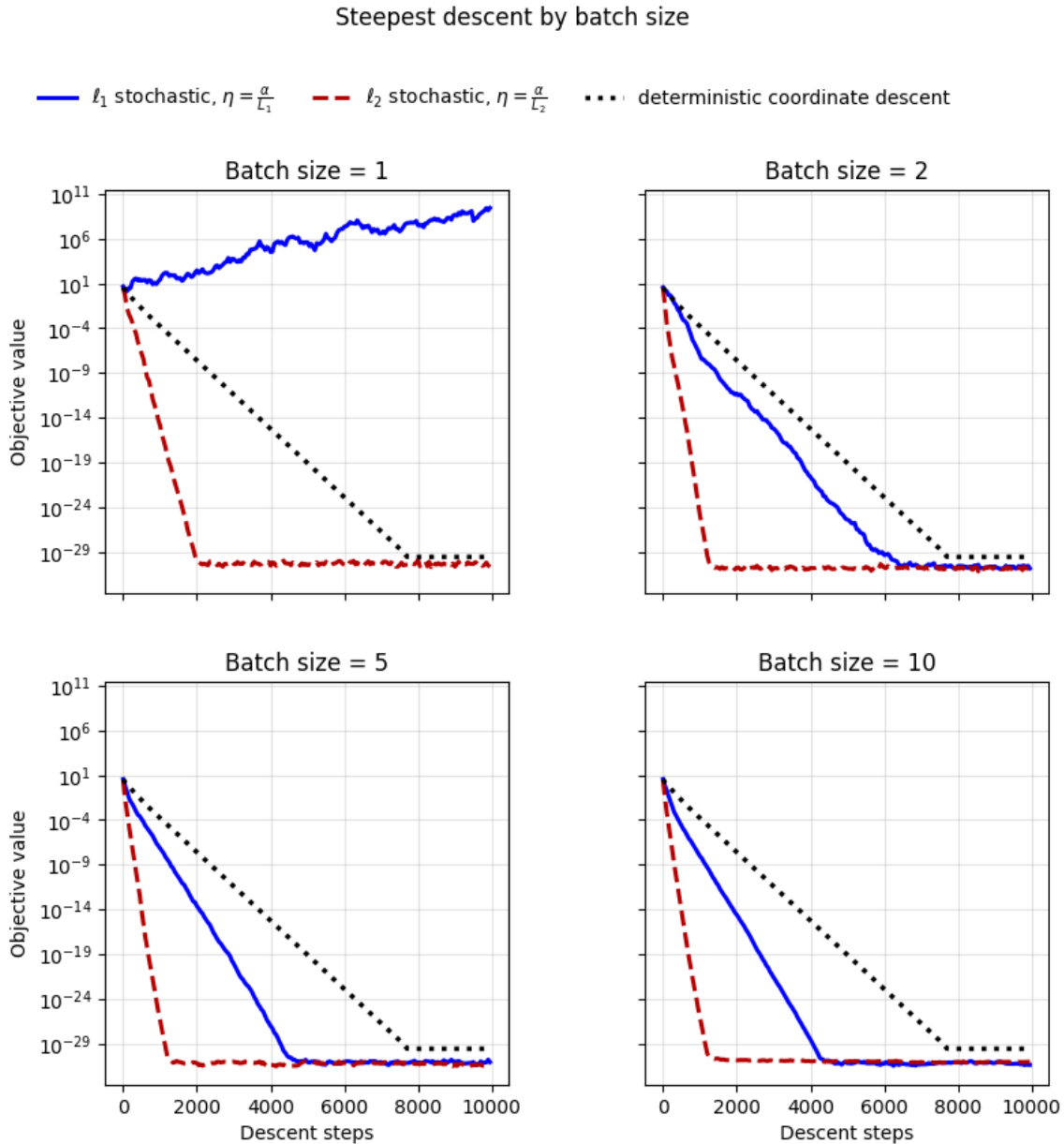


Figure 4.1: **Objective value versus iteration for stochastic steepest descent under different mini-batch sizes.** Each panel fixes the mini-batch size and compares the ℓ_1 stochastic method, the ℓ_2 stochastic method, and deterministic coordinate descent. The deterministic coordinate descent curve provides a full-gradient benchmark for the ℓ_1 coordinate update.

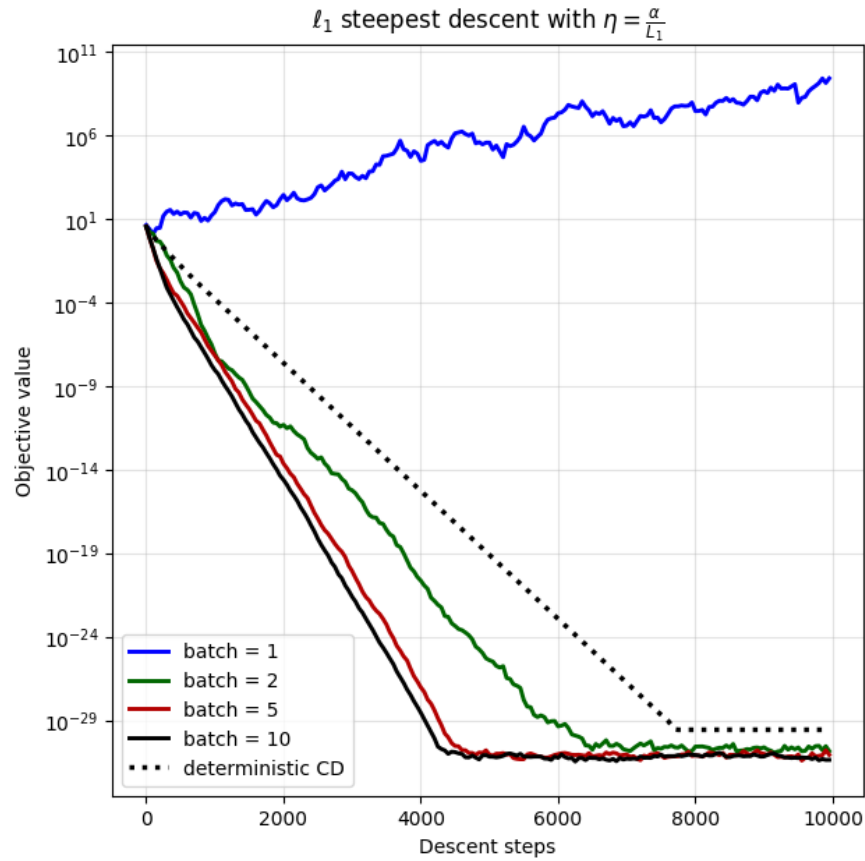


Figure 4.2: ℓ_1 stochastic steepest descent for different mini-batch sizes, together with deterministic coordinate descent. The plot highlights the batch-size regimes predicted by the theory: very small batches can be unstable, intermediate batches may converge with substantial stochastic effects, and sufficiently large batches show more regular decay.

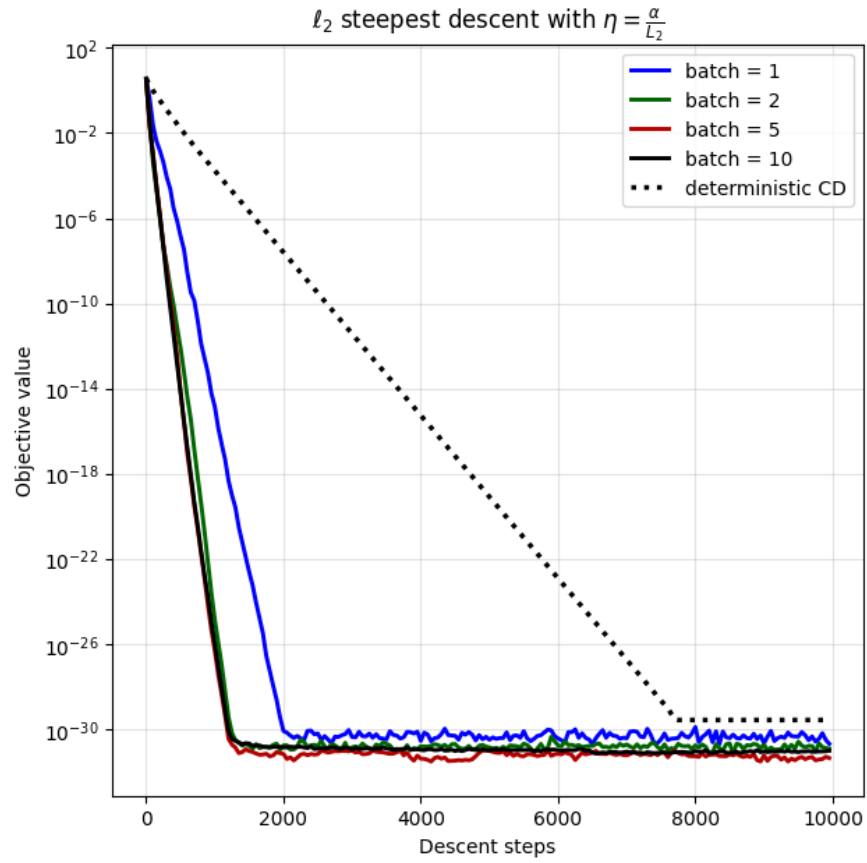


Figure 4.3: ℓ_2 stochastic steepest descent for different mini-batch sizes, together with deterministic coordinate descent as a benchmark. The ℓ_2 geometry exhibits the expected stabilization as B increases, with larger batches producing smoother and more nearly linear convergence on the logarithmic scale.

plot, by contrast, shows that the relationship between batch size and convergence speed can be nonmonotone over a finite horizon. This is precisely the phenomenon anticipated in Remark 4.2.5: the threshold conditions describe when contraction and a small enough stochastic error floor can be guaranteed, but they do not imply that every larger batch must dominate every smaller batch at every iteration. Taken together, the experiments support the interpretation that the Euclidean case is structurally special. The ℓ_2 method behaves like standard SGD, while the ℓ_1 method exhibits a visible dependence on batch size because the sampled gradient determines a coordinate choice. This does not prove that every general norm requires a batch-size threshold, but it suggests that bypassing the threshold outside ℓ_2 would require new structure beyond the RWGC argument used in Chapter 1.

Bibliography

- [1] Akshay Agrawal et al. “Differentiable Convex Optimization Layers”. In: *Advances in Neural Information Processing Systems*. Vol. 32. 2019.
- [2] Brandon Amos and J. Zico Kolter. “OptNet: Differentiable Optimization as a Layer in Neural Networks”. In: *Proceedings of the 34th International Conference on Machine Learning*. Vol. 70. Proceedings of Machine Learning Research. PMLR, 2017, pp. 136–145.
- [3] Douglas N Arnold. “A concise introduction to numerical analysis”. In: *Institute for Mathematics and its Applications, Minneapolis* (2001).
- [4] Runxue Bao, Bin Gu, and Heng Huang. “Efficient approximate solution path algorithm for order weight l_1 -norm with accuracy guarantee”. In: *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE. 2019, pp. 958–963.
- [5] Heinz H. Bauschke, Jérôme Bolte, and Marc Teboulle. “A Descent Lemma Beyond Lipschitz Gradient Continuity: First-Order Methods Revisited and Applications”. In: *Mathematics of Operations Research* 42.2 (2017), pp. 330–348. DOI: 10.1287/moor.2016.0817.
- [6] Amir Beck. *First-Order Methods in Optimization*. Vol. 25. MOS-SIAM Series on Optimization. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2017. ISBN: 9781611974980. DOI: 10.1137/1.9781611974997.
- [7] Amir Beck and Marc Teboulle. “Mirror Descent and Nonlinear Projected Subgradient Methods for Convex Optimization”. In: *Operations Research Letters* 31.3 (2003), pp. 167–175. DOI: 10.1016/S0167-6377(02)00231-6.
- [8] DP Bertsekas. “Neuro-dynamic programming”. In: *Athena Scientific* (1996).
- [9] Léon Bottou, Frank E Curtis, and Jorge Nocedal. “Optimization methods for large-scale machine learning”. In: *SIAM review* 60.2 (2018), pp. 223–311.
- [10] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge: Cambridge University Press, 2004. ISBN: 9780521833783. URL: <https://web.stanford.edu/~boyd/cvxbook/>.
- [11] Michael Collins, Robert E. Schapire, and Yoram Singer. “Logistic Regression, AdaBoost and Bregman Distances”. In: *Machine Learning* 48.1–3 (2002), pp. 253–285. DOI: 10.1023/A:1013912006537.

- [12] Qiran Dong, Paul Grigas, and Vishal Gupta. “Beyond Discretization: Learning the Optimal Solution Path”. In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2025, pp. 4528–4536.
- [13] Priya Donti, Brandon Amos, and J. Zico Kolter. “Task-Based End-to-End Model Learning in Stochastic Optimization”. In: *Advances in Neural Information Processing Systems*. Vol. 30. 2017.
- [14] Priya L. Donti, David Rolnick, and J. Zico Kolter. “DC3: A Learning Method for Optimization with Hard Constraints”. In: *International Conference on Learning Representations*. 2021.
- [15] Bradley Efron et al. “Least Angle Regression”. In: *Annals of Statistics* (2004), pp. 407–451.
- [16] Adam N. Elmachtoub and Paul Grigas. “Smart “Predict, then Optimize””. In: *Management Science* 68.1 (2022), pp. 9–26. DOI: 10.1287/mnsc.2020.3922.
- [17] Ferdinando Fioretto, Terrence W. K. Mak, and Pascal Van Hentenryck. “Predicting AC Optimal Power Flows: Combining Deep Learning and Lagrangian Dual Methods”. In: *Proceedings of the AAAI Conference on Artificial Intelligence* 34.1 (2020), pp. 630–637.
- [18] Yoav Freund and Robert E. Schapire. “A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting”. In: *Journal of Computer and System Sciences* 55.1 (1997), pp. 119–139. DOI: 10.1006/jcss.1997.1504.
- [19] Jerome Friedman, Trevor Hastie, and Rob Tibshirani. “Regularization paths for generalized linear models via coordinate descent”. In: *Journal of Statistical Software* 33.1 (2010), p. 1.
- [20] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. “Additive Logistic Regression: A Statistical View of Boosting”. In: *The Annals of Statistics* 28.2 (2000), pp. 337–407. DOI: 10.1214/aos/1016218223.
- [21] J. Giesen et al. “Approximating concavely parameterized optimization problems”. In: *Advances in Neural Information Processing Systems* 3 (Jan. 2012), pp. 2105–2113.
- [22] Joachim Giesen, Martin Jaggi, and Sören Laue. “Approximating parameterized convex optimization problems”. In: *ACM Transactions on Algorithms (TALG)* 9.1 (2012), pp. 1–17.
- [23] Robert Mansel Gower et al. “SGD: General analysis and improved rates”. In: *International conference on machine learning*. PMLR. 2019, pp. 5200–5209.
- [24] Swati Gupta, Jai Moondra, and Mohit Singh. “Which Lp norm is the fairest? Approximations for fair facility location across all” p”. In: *Proceedings of the 24th ACM Conference on Economics and Computation*. 2023, pp. 817–817.
- [25] Filip Hanzely and Peter Richtárik. “Fastest rates for stochastic mirror descent methods”. In: *Computational Optimization and Applications* 79.3 (2021), pp. 717–766.

- [26] Trevor Hastie et al. “The entire regularization path for the support vector machine”. In: *Journal of Machine Learning Research* 5.Oct (2004), pp. 1391–1415.
- [27] Ahmed Khaled and Peter Richtárik. “Better theory for SGD in the nonconvex world”. In: *arXiv preprint arXiv:2002.03329* (2020).
- [28] James Kotary et al. “End-to-End Constrained Optimization Learning: A Survey”. In: *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*. 2021, pp. 4475–4482. DOI: 10.24963/ijcai.2021/610.
- [29] Guanghai Lan. *First-order and Stochastic Optimization Methods for Machine Learning*. Vol. 1. Springer, 2020.
- [30] Guanghai Lan. “The complexity of large-scale convex programming under a linear optimization oracle”. In: *arXiv preprint arXiv:1309.5550* (2013).
- [31] Meiyi Li, Soheil Kolouri, and Javad Mohammadi. “Learning to Solve Optimization Problems With Hard Linear Constraints”. In: *IEEE Access* 11 (2023), pp. 59995–60004. DOI: 10.1109/ACCESS.2023.3285199.
- [32] Chaoyue Liu et al. “Aiming towards the minimizers: fast convergence of SGD for overparametrized problems”. In: *Advances in neural information processing systems* 36 (2024).
- [33] Heyuan Liu and Paul Grigas. “New Methods for Parametric Optimization via Differential Equations”. In: *arXiv preprint arXiv:2306.08812* (2023).
- [34] Haihao Lu, Robert M. Freund, and Yurii Nesterov. “Relatively Smooth Convex Optimization by First-Order Methods, and Applications”. In: *SIAM Journal on Optimization* 28.1 (2018), pp. 333–354. DOI: 10.1137/16M1099546.
- [35] Llew Mason et al. “Boosting Algorithms as Gradient Descent”. In: *Advances in Neural Information Processing Systems*. Vol. 12. 2000, pp. 512–518.
- [36] Eugene Ndiaye et al. “Safe grid search with optimal complexity”. In: *International Conference on Machine Learning*. PMLR. 2019, pp. 4771–4780.
- [37] Arkadi S. Nemirovski and David B. Yudin. *Problem Complexity and Method Efficiency in Optimization*. Chichester: John Wiley & Sons, 1983. ISBN: 9780471103455.
- [38] Yurii Nesterov. “Efficiency of Coordinate Descent Methods on Huge-Scale Optimization Problems”. In: *SIAM Journal on Optimization* 22.2 (2012), pp. 341–362. DOI: 10.1137/100802001.
- [39] Yurii Nesterov. *Lectures on Convex Optimization*. 2nd ed. Vol. 137. Springer Optimization and Its Applications. Springer, 2018. ISBN: 9783319915777. DOI: 10.1007/978-3-319-91578-4.
- [40] Lam Nguyen et al. “SGD and Hogwild! convergence without the bounded gradients assumption”. In: *International Conference on Machine Learning*. PMLR. 2018, pp. 3750–3758.

- [41] Michael R Osborne, Brett Presnell, and Berwin A Turlach. “A new approach to variable selection in least squares problems”. In: *IMA Journal of Numerical Analysis* 20.3 (2000), pp. 389–403.
- [42] Xiang Pan et al. “DeepOPF: A Deep Neural Network Approach for Security-Constrained DC Optimal Power Flow”. In: *IEEE Transactions on Power Systems* 36.3 (2021), pp. 1725–1735. DOI: 10.1109/TPWRS.2020.3026379.
- [43] Mee Young Park and Trevor Hastie. “L1-Regularization Path Algorithm for Generalized Linear Models”. In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 69.4 (Aug. 2007), pp. 659–677.
- [44] Seonho Park and Pascal Van Hentenryck. “Self-supervised primal-dual learning for constrained optimization”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 37. 4. 2023, pp. 4052–4060.
- [45] María Cristina Pereyra and Lesley A Ward. *Harmonic analysis: from Fourier to wavelets*. Vol. 63. American Mathematical Soc., 2012.
- [46] Scott Pesme, Aymeric Dieuleveut, and Nicolas Flammarion. “On convergence-diagnostic based step sizes for stochastic gradient descent”. In: *International Conference on Machine Learning*. PMLR. 2020, pp. 7641–7651.
- [47] Saharon Rosset. “Following curved regularized optimization solution paths”. In: *Advances in Neural Information Processing Systems* 17 (2004).
- [48] Saharon Rosset and Ji Zhu. “Piecewise linear regularized solution paths”. In: *The Annals of Statistics* (2007), pp. 1012–1030.
- [49] Robert E. Schapire and Yoram Singer. “Improved Boosting Algorithms Using Confidence-Rated Predictions”. In: *Machine Learning* 37.3 (1999), pp. 297–336. DOI: 10.1023/A:1007614523901.
- [50] Ryan J. Tibshirani and Jonathan Taylor. “The solution path of the generalized lasso”. In: *The Annals of Statistics* 39.3 (2011), pp. 1335–1371.
- [51] Lloyd N Trefethen. *Approximation Theory and Approximation Practice*. SIAM, 2013.
- [52] Pascal Van Hentenryck. “Optimization Learning”. In: *arXiv preprint arXiv:2501.03443* (2025).
- [53] Sharan Vaswani, Francis Bach, and Mark Schmidt. “Fast and faster convergence of sgd for over-parameterized models and an accelerated perceptron”. In: *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR. 2019, pp. 1195–1204.
- [54] Linda F Wightman. “LSAC National Longitudinal Bar Passage Study. LSAC Research Report Series.” In: (1998).
- [55] Stephen J. Wright. “Coordinate Descent Algorithms”. In: *Mathematical Programming* 151.1 (2015), pp. 3–34. DOI: 10.1007/s10107-015-0892-3.

- [56] David Borisovich Yudin. *Problem complexity and method efficiency in optimization*. Wiley, 1983.

Appendix A

Appendix

A.1 Omitted Proofs

A.1.1 Proofs from Section 2.2

Proof for Lemma 2.2.3. Suppose by contradiction that $c = 0$. Then there exists a corresponding eigenvector $\mathbf{v} \in \mathbb{R}^p$ with $\mathbf{v} \neq \mathbf{0}$ such that

$$\mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} \left[\Phi(\tilde{\lambda})^\top \Phi(\tilde{\lambda}) \right] \mathbf{v} = \mathbf{0} \implies \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} \left[\|\Phi(\tilde{\lambda})\mathbf{v}\|_2^2 \right] = 0.$$

This further implies that $\Phi(\tilde{\lambda})\mathbf{v} = 0$ almost everywhere. By the linear independence assumption in Assumption 2.2.2, we then must have $\mathbf{v} = \mathbf{0}$, but this is a contradiction since $\mathbf{v}$ is an eigenvector.

For the second statement, notice

$$c \leq \sigma_{\max} \left\{ \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} \left[\Phi(\tilde{\lambda})^\top \Phi(\tilde{\lambda}) \right] \right\} \leq \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} \left[\sigma_{\max} \left\{ \Phi(\tilde{\lambda})^\top \Phi(\tilde{\lambda}) \right\} \right] \leq C,$$

where the penultimate inequality follows from Jensen's inequality and the convexity of the maximal eigenvalue function. $\square$

Next, recall the family of functions $f(\cdot, \lambda)$ and the function $F(\cdot)$ defined in (2.4):

$$f(\beta, \lambda) := h(\Phi(\lambda)\beta, \lambda), \quad F(\beta) := \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} \left[f(\beta, \tilde{\lambda}) \right] = \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_\lambda} \left[h(\Phi(\tilde{\lambda})\beta, \tilde{\lambda}) \right].$$

Before proceeding, we establish the smoothness and strong convexity of these functions.

Proposition A.1.1 (Uniform Smoothness and Strong Convexity). *The family of functions $f(\cdot, \lambda)$, over all $\lambda \in \Lambda$, is CL -smooth. Moreover, $F(\cdot)$ is CL -smooth and $c\mu$ -strongly convex.*

Proof for Proposition A.1.1. Fix an arbitrary $\lambda \in \Lambda$. Then, for any $\beta, \bar{\beta} \in \mathbb{R}^p$, smoothness

of $f(\cdot, \lambda)$ is certified by

$$\begin{aligned}
\|\nabla f(\beta, \lambda) - \nabla f(\bar{\beta}, \lambda)\| &= \|\nabla_{\beta} h(\Phi(\lambda)\beta, \lambda) - \nabla_{\beta} h(\Phi(\lambda)\bar{\beta}, \lambda)\| \\
&\leq \|\Phi(\lambda)^{\top} (\nabla_{\theta} h(\Phi(\lambda)\beta, \lambda) - \nabla_{\theta} h(\Phi(\lambda)\bar{\beta}, \lambda))\| \quad (\text{chain rule}) \\
&\leq \|\Phi(\lambda)^{\top}\| \cdot L \|\Phi(\lambda)(\beta - \bar{\beta})\| \quad (L\text{-smoothness}) \\
&\leq CL \|\beta - \bar{\beta}\|
\end{aligned}$$

for any $\beta, \bar{\beta} \in \mathbb{R}^d$. To obtain the same conclusions for $F(\cdot)$, we may take expectation over $\lambda \sim \mathbb{P}_{\lambda}$ on the above inequalities, and invoke the linearity of expectations as well as the property that

$$\|\mathbb{E}[g(\cdot)]\| \leq \mathbb{E}[\|g(\cdot)\|].$$

Next, we verify the strong convexity of $F(\cdot)$. Fixing $\beta, \bar{\beta} \in \mathbb{R}^d$, for any $\lambda \in \Lambda$, define $\theta(\lambda) := \Phi(\lambda)\beta$, $\bar{\theta}(\lambda) := \Phi(\lambda)\bar{\beta}$. Then, by strong convexity of h , we have

$$h(\bar{\theta}(\lambda), \lambda) \geq h(\theta(\lambda), \lambda) + \nabla_{\theta} h(\theta(\lambda), \lambda)^{\top} (\bar{\theta}(\lambda) - \theta(\lambda)) + \frac{\mu}{2} \|\bar{\theta}(\lambda) - \theta(\lambda)\|^2.$$

Take expectation w.r.t. $\tilde{\lambda} \sim \mathbb{P}_{\lambda}$ on both sides, we obtain

$$\begin{aligned}
F(\bar{\beta}) &= \mathbb{E}[h(\Phi(\tilde{\lambda})\bar{\beta}, \tilde{\lambda})] \\
&\geq \mathbb{E}[h(\Phi(\tilde{\lambda})\beta, \tilde{\lambda})] + \mathbb{E}[\nabla_{\theta} h(\theta(\tilde{\lambda}), \tilde{\lambda})^{\top} (\Phi(\tilde{\lambda})(\bar{\beta} - \beta))] + \frac{\mu}{2} \mathbb{E}[\|\Phi(\tilde{\lambda})(\bar{\beta} - \beta)\|^2] \\
&= F(\beta) + \mathbb{E}[\Phi(\tilde{\lambda})^{\top} \nabla_{\theta} h(\theta(\tilde{\lambda}), \tilde{\lambda})]^{\top} (\bar{\beta} - \beta) + \frac{\mu}{2} (\bar{\beta} - \beta)^{\top} \mathbb{E}[\Phi(\tilde{\lambda})^{\top} \Phi(\tilde{\lambda})] (\bar{\beta} - \beta) \\
&\geq F(\beta) + \nabla F(\beta)^{\top} (\bar{\beta} - \beta) + \frac{c\mu}{2} (\bar{\beta} - \beta)^{\top} (\bar{\beta} - \beta)
\end{aligned}$$

The second term in the last inequality comes from the Leibniz integral rule and the third term invokes a well-known property of the smallest eigenvalue of a positive definite matrix. $\square$

Proof for Theorem 2.2.4.

First, we prove that ϵ_{sp}^* is attained at some $\beta_{\text{sp}}^* \in \mathbb{R}^p$. Observe that $\epsilon_{\text{sp}}(\cdot)$ is continuous. Thus, its level set $M(1) := \{\beta \in \mathbb{R}^p : \epsilon_{\text{sp}}(\beta) \leq 1\}$ is closed. Furthermore, $M(1)$ is bounded. To see this, for any $\beta \in M(1)$, by strong convexity of F ,

$$\begin{aligned}
\frac{c\mu}{2} \|\beta - \beta_{\text{avg}}^*\|^2 &\leq F(\beta) - F(\beta_{\text{avg}}^*) \\
&= \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_{\lambda}} [h(\Phi(\tilde{\lambda})\beta, \tilde{\lambda}) - h(\Phi(\tilde{\lambda})\beta_{\text{avg}}^*, \tilde{\lambda})] \\
&\leq \mathbb{E}_{\tilde{\lambda} \sim \mathbb{P}_{\lambda}} [h(\Phi(\tilde{\lambda})\beta, \tilde{\lambda}) - h(\theta^*(\tilde{\lambda}), \tilde{\lambda})] \\
&\leq \epsilon_{\text{sp}}(\beta) \\
&\leq 1.
\end{aligned} \tag{A.0}$$

Since β is arbitrary, for any $\beta, \bar{\beta} \in M(1)$,

$$\|\beta - \bar{\beta}\| \leq \|\beta - \beta_{\text{avg}}^*\| + \|\bar{\beta} - \beta_{\text{avg}}^*\| \leq \frac{2\sqrt{2}}{\sqrt{c\mu}}.$$

Applying the Weierstrass Theorem, there exists $\beta_{\text{sp}}^* \in M(1) \subset \mathbb{R}^p$ s.t.

$$\epsilon_{\text{sp}}(\beta_{\text{sp}}^*) = \epsilon_{\text{sp}}^*.$$

Next, we prove the main result of the theorem. Again, fix an arbitrary λ . By L -smoothness of $h(\cdot, \lambda)$,

$$h(\Phi(\lambda)\beta, \lambda) - h(\theta^*(\lambda), \lambda) \leq \frac{L}{2} \|\Phi(\lambda)\beta - \theta^*(\lambda)\|^2 \quad (\text{A.1})$$

Using the identity $(a + b + c)^2 \leq 4a^2 + 4b^2 + 4c^2$ and triangle inequality on the right-hand side, this inequality becomes

$$h(\Phi(\lambda)\beta, \lambda) - h(\theta^*(\lambda), \lambda) \leq 2L(\|\Phi(\lambda)(\beta - \beta_{\text{avg}}^*)\|^2 + \|\Phi(\lambda)(\beta_{\text{avg}}^* - \beta_{\text{sp}}^*)\|^2 + \|\Phi(\lambda)\beta_{\text{sp}}^* - \theta^*(\lambda)\|^2).$$

By strong convexity of $h(\cdot, \lambda)$ and definition of β_{sp}^* ,

$$\|\Phi(\lambda)\beta_{\text{sp}}^* - \theta^*(\lambda)\|^2 \leq \frac{2}{\mu} \sup_{\lambda \in \Lambda} \{h(\Phi(\lambda)\beta_{\text{sp}}^*, \lambda) - h(\theta^*(\lambda), \lambda)\} = \frac{2}{\mu} \epsilon_{\text{sp}}(\beta_{\text{sp}}^*) = \frac{2}{\mu} \epsilon_{\text{sp}}^*,$$

So

$$\epsilon_{\text{sp}}(\beta) \leq 2CL(\|\beta - \beta_{\text{avg}}^*\|^2 + \|\beta_{\text{avg}}^* - \beta_{\text{sp}}^*\|^2) + \frac{4L}{\mu} \epsilon_{\text{sp}}^*. \quad (\text{A.2})$$

Next, we bound $\|\beta_{\text{avg}}^* - \beta_{\text{sp}}^*\|^2$. Observe that due to the optimality of $\theta^*(\lambda)$ for each λ ,

$$\mathbb{E} [h(\Phi(\tilde{\lambda})\beta_{\text{sp}}^*, \tilde{\lambda})] - \mathbb{E} [h(\Phi(\tilde{\lambda})\beta_{\text{avg}}^*, \tilde{\lambda})] \leq \mathbb{E} [h(\Phi(\tilde{\lambda})\beta_{\text{sp}}^*, \tilde{\lambda}) - h(\theta^*(\tilde{\lambda}), \tilde{\lambda})] \leq \epsilon_{\text{sp}}^*.$$

On the other hand, from the optimality of β_{avg}^* and strong-convexity of $F(\cdot)$,

$$F(\beta_{\text{sp}}^*) - F(\beta_{\text{avg}}^*) = \mathbb{E} [h(\Phi(\tilde{\lambda})\beta_{\text{sp}}^*, \tilde{\lambda})] - \mathbb{E} [h(\Phi(\tilde{\lambda})\beta_{\text{avg}}^*, \tilde{\lambda})] \geq \frac{c\mu}{2} \|\beta_{\text{avg}}^* - \beta_{\text{sp}}^*\|^2.$$

Combining this inequality with the previous one shows $\frac{c\mu}{2} \|\beta_{\text{avg}}^* - \beta_{\text{sp}}^*\|^2 \leq \epsilon_{\text{sp}}^*$, which implies that

$$\|\beta_{\text{avg}}^* - \beta_{\text{sp}}^*\|^2 \leq \frac{2\epsilon_{\text{sp}}^*}{c\mu}.$$

Substitute the above into Equation (A.2). Take expectations over sample paths on both

sides, we conclude that

$$\epsilon_{\text{sp}}(\beta) \leq 2CL \left(\|\beta - \beta_{\text{avg}}^*\|^2 + \frac{2\epsilon_{\text{sp}}^*}{c\mu} \right) + \frac{4L}{\mu} \epsilon_{\text{sp}}^*.$$

Rearranging and using the fact that $C/c \geq 1$ verifies the statement. $\square$

A.1.2 Proofs from Section 2.3

Proof for Theorem 2.3.2. The proof follows the proof of Theorem 3.1 of [23]; we include it for completeness under the assumption of the RWGC Definition 2.3.1. Let $\eta := \frac{\bar{\eta}}{\rho}$ denote the step-size, $\mathbf{w}^* := \arg \min_{\mathbf{w} \in \mathbb{R}^d} G(\mathbf{w})$ denote the optimality input of G . For $t \geq 0$, we have

$$\begin{aligned} \|\mathbf{w}_{t+1} - \mathbf{w}^*\|^2 &= \|\mathbf{w}_t - \mathbf{w}^* - \eta \nabla g(\mathbf{w}_t, z_t)\|^2 \\ &= \|\mathbf{w}_t - \mathbf{w}^*\|^2 - 2\eta \nabla g(\mathbf{w}_t, z_t)^\top (\mathbf{w}_t - \mathbf{w}^*) + \eta^2 \|\nabla g(\mathbf{w}_t, z_t)\|^2. \end{aligned}$$

Taking conditional expectations yields

$$\begin{aligned} \mathbb{E} [\|\mathbf{w}_{t+1} - \mathbf{w}^*\|^2 \mid \mathbf{w}_t] &= \|\mathbf{w}_t - \mathbf{w}^*\|^2 - 2\eta \nabla G(\mathbf{w}_t)^\top (\mathbf{w}_t - \mathbf{w}^*) + \eta^2 \mathbb{E} [\|\nabla g(\mathbf{w}_t, z_t)\|^2 \mid \mathbf{w}_t] \\ &\leq \|\mathbf{w}_t - \mathbf{w}^*\|^2 - 2\eta \nabla G(\mathbf{w}_t)^\top (\mathbf{w}_t - \mathbf{w}^*) + \eta^2 [2\rho(G(\mathbf{w}_t) - G^*) + \sigma^2] \\ &\leq \|\mathbf{w}_t - \mathbf{w}^*\|^2 + 2\eta (G^* - G(\mathbf{w}_t) - \frac{\mu_g}{2} \|\mathbf{w}_t - \mathbf{w}^*\|^2) + \eta^2 [2\rho(G(\mathbf{w}_t) - G^*) + \sigma^2] \\ &= (1 - \eta\mu_g) \|\mathbf{w}_t - \mathbf{w}^*\|^2 + 2(\eta^2\rho - \eta)(G(\mathbf{w}_t) - G^*) + \eta^2\sigma^2 \\ &\leq (1 - \eta\mu_g) \|\mathbf{w}_t - \mathbf{w}^*\|^2 + \eta^2\sigma^2 \end{aligned} \tag{A.3}$$

The first inequality above uses the RWGC and the second uses strong convexity. The final inequality holds since $\eta^2\rho - \eta = \left(\frac{\bar{\eta}}{\rho}\right)^2 \rho - \frac{\bar{\eta}}{\rho} = \left(\frac{\bar{\eta}}{\rho}\right)(\bar{\eta} - 1)$ and $\bar{\eta} \leq 1$. Furthermore, since $\bar{\eta} \leq \frac{\rho}{\mu_g}$, we have $\eta\mu_g = \left(\frac{\bar{\eta}}{\rho}\right)\mu_g \leq 1$; therefore, by iterating Equation (A.3) and taking overall expectations, we get

$$\begin{aligned} \mathbb{E} [\|\mathbf{w}_t - \mathbf{w}^*\|^2] &\leq (1 - \eta\mu_g)^t \|\mathbf{w}_0 - \mathbf{w}^*\|^2 + \eta^2\sigma^2 \sum_{k=0}^{t-1} (1 - \eta\mu_g)^k \\ &\leq (1 - \eta\mu_g)^t \|\mathbf{w}_0 - \mathbf{w}^*\|^2 + \frac{\eta^2\sigma^2}{1 - (1 - \eta\mu_g)} \\ &= (1 - \eta\mu_g)^t \|\mathbf{w}_0 - \mathbf{w}^*\|^2 + \frac{\eta\sigma^2}{\mu_g}, \end{aligned}$$

where the second inequality uses the geometric series bound. Recalling that $\eta = \frac{\bar{\eta}}{\rho}$ completes the proof. $\square$

Proof for Lemma 2.3.3. Recall that Proposition A.1.1 already establishes the smoothness of f and smoothness and strong convexity of F .

Define

$$\bar{\beta}(\lambda) \in \arg \min_{\beta \in \mathbb{R}^p} f(\beta, \lambda) = \arg \min_{\beta \in \mathbb{R}^p} h(\Phi(\lambda)\beta, \lambda).$$

By CL -smoothness, we have

$$f(\beta, \lambda) \geq f(\bar{\beta}(\lambda), \lambda) + \nabla f(\bar{\beta}(\lambda), \lambda)^\top (\beta - \bar{\beta}(\lambda)) + \frac{1}{2CL} \|\nabla f(\beta, \lambda) - \nabla f(\bar{\beta}(\lambda), \lambda)\|^2.$$

Using $\nabla f(\bar{\beta}(\lambda), \lambda) = 0$ for all λ , rearranging and recalling the definitions of β_{avg}^* (Section 2.2) and β_{sp}^* (proof of Theorem 2.2.4) yields

$$\begin{aligned} \frac{1}{2CL} \|\nabla f(\beta, \lambda)\|^2 &\leq f(\beta, \lambda) - f(\bar{\beta}(\lambda), \lambda) \\ &\leq f(\beta, \lambda) - h(\theta^*(\lambda), \lambda) \\ &= (f(\beta, \lambda) - f(\beta_{\text{avg}}^*, \lambda)) + (f(\beta_{\text{avg}}^*, \lambda) - f(\beta_{\text{sp}}^*, \lambda)) + (f(\beta_{\text{sp}}^*, \lambda) - h(\theta^*(\lambda), \lambda)) \\ &\leq (f(\beta, \lambda) - f(\beta_{\text{avg}}^*, \lambda)) + (f(\beta_{\text{avg}}^*, \lambda) - f(\beta_{\text{sp}}^*, \lambda)) + \epsilon_{\text{sp}}^* \end{aligned}$$

where the second inequality uses optimality of $\theta^*(\lambda)$ and the third uses that ϵ_{sp}^* is attained at β_{sp}^* . Taking expectations on both sides, we get

$$\frac{1}{2CL} \mathbb{E} [\|\nabla f(\beta, \tilde{\lambda})\|^2] \leq F(\beta) - F(\beta_{\text{avg}}^*) + F(\beta_{\text{avg}}^*) - F(\beta_{\text{sp}}^*) + \epsilon_{\text{sp}}^* \leq F(\beta) - F(\beta_{\text{avg}}^*) + \epsilon_{\text{sp}}^*,$$

where we used that the definition of β_{avg}^* implies that $F(\beta_{\text{avg}}^*) \leq F(\beta_{\text{sp}}^*)$. Multiplying through by $2CL$ completes the proof. $\square$

Proof for Theorem 2.3.4. Our strategy will be to invoke Lemma 2.3.3 and apply Theorem 2.3.2, in conjunction with Theorem 2.2.4. To that end, notice that the upper bound on the step-size parameter $\bar{\eta}$ given in Theorem 2.3.2 becomes $\frac{CL}{c\mu} \geq 1$. Hence, we need only constrain $0 < \bar{\eta} \leq 1$.

Applying Theorem 2.3.2 yields

$$\mathbb{E} [\|\beta_T - \beta_{\text{avg}}^*\|^2] \leq \left(1 - \bar{\eta} \frac{c\mu}{CL}\right)^T \|\beta_0 - \beta_{\text{avg}}^*\|^2 + \frac{2\bar{\eta}\epsilon_{\text{sp}}^*}{c\mu}.$$

Taking the expectation of Theorem 2.2.4 on both sides and substituting the above, we have

$$\begin{aligned} \mathbb{E} [\epsilon_{\text{sp}}(\beta_T)] &\leq 2CL \mathbb{E} [\|\beta_T - \beta_{\text{avg}}^*\|^2] + \frac{8CL}{c\mu} \epsilon_{\text{sp}}^* \\ &\leq 2CL \left(1 - \bar{\eta} \frac{c\mu}{CL}\right)^T \|\beta_0 - \beta_{\text{avg}}^*\|^2 + 4\frac{\bar{\eta}CL}{c\mu} \epsilon_{\text{sp}}^* + \frac{8CL}{c\mu} \epsilon_{\text{sp}}^*. \end{aligned} \quad (\text{A.4})$$

Rearranging yields the first result. Next, we prove the two bounds of the iteration complexity.

First, consider the case $\epsilon_{\text{sp}}^* > 0$. Our goal will be to choose T large enough to drive the first term on the right side of (A.4) to below $4\frac{CL}{c\mu}\bar{\eta}\epsilon_{\text{sp}}^*$. To simplify exposition, let $\kappa = \frac{CL}{c\mu}$. Then, we need to ensure that

$$T \log(1 - \bar{\eta}/\kappa) \leq \log \left(\frac{2\kappa\bar{\eta}\epsilon_{\text{sp}}^*}{CL\|\beta_0 - \beta_{\text{avg}}^*\|^2} \right).$$

We can upper bound the left side using the identity $\log(1 + x) \leq x$. Hence, it would be sufficient if

$$T \geq \frac{\kappa}{\bar{\eta}} \log \left(\frac{CL\|\beta_0 - \beta_{\text{avg}}^*\|^2}{2\kappa\bar{\eta}\epsilon_{\text{sp}}^*} \right).$$

Replacing κ with its definition yields the first case.

For the second case where $\epsilon_{\text{sp}}^* = 0$, (A.4) reduces to

$$\mathbb{E} [\epsilon_{\text{sp}}(\beta_T)] \leq 2CL \left(1 - \bar{\eta} \frac{c\mu}{CL} \right)^T \|\beta_0 - \beta_{\text{avg}}^*\|^2.$$

Given $\epsilon > 0$, we need to ensure that

$$T \log(1 - \bar{\eta}/\kappa) \leq \log \left(\frac{\epsilon}{2CL\|\beta_0 - \beta_{\text{avg}}^*\|^2} \right).$$

Again using $\log(1 + x) \leq x$, it would be sufficient if

$$T \geq \frac{\kappa}{\bar{\eta}} \log \left(\frac{2CL\|\beta_0 - \beta_{\text{avg}}^*\|^2}{\epsilon} \right).$$

Replacing κ with its definition yields the second case.

□

A.1.3 Proofs from Section 2.4

Proof of Lemma 2.4.1. As described in the main text, $\Phi(\lambda) \in \mathbb{R}^{d \times qd}$ is block-diagonal with d blocks of size $1 \times q$. Denote this block by $\psi(\lambda) \in \mathbb{R}^{1 \times q}$ and note the elements of $\psi(\lambda)$ are precisely the first q Legendre polynomials evaluated at λ .

Now the matrix $\Phi(\lambda)^\top \Phi(\lambda) \in \mathbb{R}^{qd \times qd}$ is also block diagonal with d copies of the matrix $\psi(\lambda)^\top \psi(\lambda)$. As a consequence, the eigenvectors of $\Phi(\lambda)^\top \Phi(\lambda)$ are the stacked copies of the eigenvectors of $\psi(\lambda)^\top \psi(\lambda)$ with the same eigenvalues.

Since $\psi(\lambda)^\top \psi(\lambda)$ is a rank-one matrix, it has at most one non-zero eigenvalue, and by

inspection, this eigenvalue is $\|\psi(\lambda)\|^2$ with eigenvector $\psi(\lambda)^\top$. Then,

$$C_p = \sup_{\lambda \in [-1, 1]} \|\psi(\lambda)\|^2 = q,$$

because the Legendre polynomials achieve their maxima at 1 with a value of 1.

By a nearly identical argument, we can see that

$$\begin{aligned} c_p &= \sigma_{\min} \left\{ \mathbb{E} \left[\psi(\tilde{\lambda})^\top \psi(\tilde{\lambda}) \right] \right\} \\ &= \min_{n=0, \dots, q-1} \frac{2}{2n+1} \\ &= \frac{2}{2q-1}, \end{aligned}$$

because by the orthogonality of the Legendre polynomials, the above matrix is diagonal. Hence

$$C_p/c_p = \frac{2q^2 - q}{2} \leq q^2.$$

□

Proof for Lemma 2.4.2. Our proof is constructive and we will show a slightly stronger result. We will approximate $\theta_i^*(\lambda)$ by its Chebyshev truncation up to degree q for each i . Letting $\bar{\beta}$ be the coefficients corresponding to the resulting polynomials, we will show that $\epsilon_{\text{sp}}(\bar{\beta})$ satisfies the bound in Lemma 2.4.2, which implies that ϵ_{sp}^* also satisfies the bound.

Let $T_n(\lambda)$ denote the n^{th} Chebyshev polynomial of the first kind. For the i^{th} dimension of the solution path, let $a_{i,n}$ denote the coefficient of T_n in the Chebyshev truncation of θ_i^* up to degree q .

We assumed that $\theta_i^*(\lambda), \dots, \theta_i^{(\nu-1)*}(\lambda)$ are absolutely continuous and $\theta_i^{(\nu)*}(\lambda)$ has bounded variation V . When $\nu = 0$, we are assuming simply that $\theta_i^*(\lambda)$ is of bounded variation V . Thus, for any $q \geq \nu + 1$ and $i \in [d]$, [51, Theorem 7.2] guarantees that

$$\sup_{\lambda \in [-1, 1]} \left| \sum_{n=0}^q a_{i,n} T_n(\lambda) - \theta_i^*(\lambda) \right| \leq \frac{2V}{\pi \nu (q - \nu)^\nu}.$$

Then,

$$\epsilon_{\text{sp}}(\bar{\beta}) = \sup_{\lambda \in \Lambda} \left\{ h \left(\left(\sum_{n=0}^q a_{i,n} T_n(\lambda) \right)_{i=[d]}, \lambda \right) - h(\theta^*(\lambda), \lambda) \right\}.$$

Moreover, smoothness of $h(\cdot, \lambda)$ ensures that for any $\lambda \in \Lambda$,

$$h\left(\left(\sum_{n=0}^q a_{i,n} T_n(\lambda)\right)_{i=[d]}, \lambda\right) - h(\boldsymbol{\theta}^*(\lambda), \lambda) \leq \frac{L}{2} \left\| \left(\sum_{n=0}^q a_{i,n} T_n(\lambda)\right)_{i=[d]} - \boldsymbol{\theta}^*(\lambda) \right\|^2.$$

Combine the above and using the definition of l_2 -norm, we deduce that

$$\begin{aligned} \epsilon_{\text{sp}}(\bar{\beta}) &\leq \sup_{\lambda \in \Lambda} \left\{ \frac{L}{2} \left\| \left(\sum_{n=0}^q a_{i,n} T_n(\lambda)\right)_{i=[d]} - \boldsymbol{\theta}^*(\lambda) \right\|^2 \right\} \\ &\leq \sup_{\lambda \in [-1, 1]} \left\{ \frac{dL}{2} \cdot \sup_{i \in [d]} \left| \sum_{n=0}^q a_{i,n} T_n(\lambda) - \boldsymbol{\theta}_i^*(\lambda) \right|^2 \right\} \\ &\leq \frac{dL}{2} \left(\frac{2V}{\pi\nu(q-\nu)^\nu} \right)^2. \end{aligned}$$

We note that this result is presented and holds for any $\nu \geq 0$. (When $\nu = 0$, the bound is infinity and hence trivially valid.) As will be seen in the proof of Theorem 2.4.3 below, however, we will only apply the result when $\nu \geq 2$. □

Proof of Theorem 2.4.3. As we are only interested in asymptotic behavior as $\epsilon \rightarrow 0$, we will often suppress any constants that do not depend on ϵ below. In particular, we will write $a \lesssim b$ whenever there exists a constant C (not depending on ϵ but perhaps depending on h and Λ) such that $a \leq Cb$.

Note that in order to attain a small solution path error, we will need to use a large number of polynomials q . Based on Theorem 2.3.4, our first goal is to choose q large enough that

$$\epsilon \geq \frac{8L(\bar{\eta} + 1)}{\mu} \frac{C_q}{c_q} \cdot \epsilon_{\text{sp}}^*(q). \quad (\text{A.5})$$

First observe that Lemma 2.4.2 establishes that the solution path error is at most

$$\epsilon_{\text{sp}}^*(q) \leq \frac{4dLV^2}{2\pi^2\nu^2(q-\nu)^{2\nu}} \lesssim q^{-2\nu}. \quad (\text{A.6})$$

Using Lemma 2.4.1, we can upper bound the right side of Equation (A.5) by

$$\frac{8L(\bar{\eta} + 1)}{\mu} \frac{C_q}{c_q} \cdot \epsilon_{\text{sp}}^*(q) \lesssim q^{2(1-\nu)}.$$

Hence it suffices to take

$$\epsilon^{-\frac{1}{2(\nu-1)}} \lesssim q$$

polynomials to achieve our target error.

We now seek to bound the iteration count. By Theorem 2.3.4, the iteration count should exceed

$$\frac{L}{\bar{\eta}\mu} \frac{C_q}{c_q} \log \left(\frac{c_q \mu \|\beta_{\text{avg}}^*\|^2}{2\bar{\eta}\epsilon_{\text{sp}}^*} \right) \lesssim q^2 (\log \|\beta_{\text{avg}}^*\|^2 + \log(q))$$

where we have assumed SGD was initialized at $\beta = 0$ and used Equation (A.6) and $c_q = \frac{2}{2q-1}$ (cf. the proof of Lemma 2.4.1) to simplify.

To complete the proof we need to bound $\|\beta_{\text{avg}}^*\|^2$. To this end, we first bound $\|\bar{\beta}\|$, where $\bar{\beta}$ is constructed by approximating each component $\theta_i^*(\lambda)$ by its Chebyshev truncation to degree q for each i .

Then,

$$\begin{aligned} \|\bar{\beta}\|^2 &= \sum_{i=1}^d \sum_{k=1}^q |a_{ik}|^2 \\ &= \sum_{i=1}^d \sum_{k=1}^{\nu} |a_{ik}|^2 + \sum_{i=1}^d \sum_{k=\nu+1}^q |a_{ik}|^2. \\ &\lesssim \sum_{k=\nu+1}^q \frac{1}{k^{2(\nu+1)}}, \\ &\lesssim \sum_{k=\nu+1}^{\infty} \frac{1}{k^{2(\nu+1)}}, \end{aligned}$$

where the penultimate inequality collects constants and invokes [51, Theorem 7.1] to bound the second summation. Note, for $\nu \geq 0$, this last sum is summable, so that $\|\bar{\beta}\| \lesssim 1$.

Furthermore, from the proof of Lemma 2.4.2, $\epsilon_{\text{sp}}(\bar{\beta}) \lesssim q^{-2\nu}$. Hence, since the solution-path error upper bounds the stochastic error, we also have $F(\bar{\beta}) - F(\beta_{\text{avg}}^*) \lesssim q^{-2\nu}$. From strong convexity, this implies that $\|\bar{\beta} - \beta_{\text{avg}}^*\| \lesssim q^{-\nu}$.

Putting it together, we have that

$$\|\beta_{\text{avg}}^*\| \leq \|\beta_{\text{avg}}^* - \bar{\beta}\| + \|\bar{\beta}\| \lesssim q^{-\nu} + 1 \lesssim 1.$$

Substituting above shows that it suffices to take $O(q^2 \log q)$ iterations as $\epsilon \rightarrow 0$. Given our previous calculation of q , this amounts to $O(\bar{\epsilon}^{\frac{1}{1-\nu}} \log(1/\bar{\epsilon}))$ iterations. $\square$

Proof for Lemma 2.4.4. The proof is very similar to the proof of Lemma 2.4.2, with the only difference being that we use Theorem 8.2 from [51] instead of Theorem 7.2.

Recall that a Bernstein ellipse $\mathcal{E}_\omega$ with radius ω in the complex plane is the ellipse

$$\mathcal{E}_\omega = \left\{ z = \frac{1}{2} \left(\omega e^{\theta\sqrt{-1}} + \omega^{-1} e^{-\theta\sqrt{-1}} \right) : \theta \in [0, 2\pi) \right\}.$$

Since $\theta_i^*(\lambda)$ is analytic, there exists an analytic continuation of $\theta_i^*(\lambda)$ to a Bernstein ellipse of radius $\omega_i > 1$. The value of ω_i generally depends on if $\theta_i^*(\lambda)$ has any singularities in the complex plane, but is guaranteed to be larger than 1. Let M_i be $\max_{z \in \mathcal{E}_{\omega_i}} |\theta_i^*(z)|$. Finally, let $\omega \equiv \min_{i=1, \dots, d} \omega_i > 1$ and $M = \max_{i=1, \dots, d} M_i < \infty$.

Now, define $T_n(\lambda)$ and $a_{i,n}$ as in the proof for Lemma 2.4.2.

For any $q > 0$ and $i \in [d]$, Theorem 8.2 in [51] guarantees that

$$\sup_{\lambda \in [-1, 1]} \left| \sum_{n=0}^q a_{i,n} T_n(\lambda) - \theta_i^*(\lambda) \right| \leq \frac{2M\omega^{-q}}{\omega - 1}.$$

Plug this result into the last inequality of the proof for Lemma 2.4.2, we obtain

$$\epsilon_{\text{sp}}(\bar{\beta}) \leq \sup_{\lambda \in [-1, 1]} \left\{ \frac{dL}{2} \cdot \sup_{i \in [d]} \left| \sum_{n=0}^q a_{i,n} T_n(\lambda) - \theta_i^*(\lambda) \right|^2 \right\} \leq \frac{dL}{2} \left(\frac{2M\omega^{-q}}{\omega - 1} \right)^2.$$

□

Proof of Theorem 2.4.5. The proof is quite similar to the proof of Theorem 2.4.3. Again, we only consider asymptotic behavior as $\epsilon \rightarrow 0$ and suppress all other constants. Hence, $a \lesssim b$ means there exists a constant C (not depending on ϵ) such that $a \leq Cb$.

Again, our first goal is to identify a q sufficiently large to achieve our target error. By Theorem 2.3.4 we seek q large enough that

$$\epsilon \geq \frac{8L(\bar{\eta} + 1)}{\mu} \frac{C_q}{c_q} \epsilon_{\text{sp}}^*(q).$$

Lemma 2.4.4 shows that there exists an $\omega > 1$ such that

$$\epsilon_{\text{sp}}^*(q) \lesssim \omega^{-2q},$$

hence it suffice to take q large enough that

$$q^2 \omega^{-2q} \lesssim \bar{\epsilon}.$$

Solving this equation exactly requires the Lambert-W function. Instead, we take $q = \frac{\log(1/\bar{\epsilon})}{\log \omega}$. Then,

$$q^2 \omega^{-2q} = \frac{\epsilon^2 \log^2 \left(\frac{1}{\epsilon} \right)}{\log^2(\omega)} \lesssim \bar{\epsilon}$$

for $\bar{\epsilon}$ sufficiently small.

Again, to bound the number of iterations, we will need to bound $\|\beta_{\text{avg}}^* - \beta_0\|$. We again will assume that $\beta_0 = \mathbf{0}$, and first consider bounding $\bar{\beta}$. Recall, $\bar{\beta}$ is obtained by approximating each component i by the Chebyshev truncation of $\theta^*(\lambda)$ to degree q . Then,

$$\begin{aligned}\|\bar{\beta}\|^2 &= \sum_{i=1}^d \sum_{k=1}^q |a_{ik}|^2 \\ &\lesssim \sum_{k=1}^q \omega^{-k} \\ &\leq \sum_{k=1}^{\infty} \omega^{-k}.\end{aligned}$$

Here, the second to last inequality uses [51, Theorem 8.1]. Since the last summation is summable, we again conclude that $\|\bar{\beta}\|_2 \lesssim 1$.

The proof of Lemma 2.4.4 establishes that $\epsilon_{\text{sp}}(\bar{\beta}) \lesssim \omega^{-2q}$. Since solution path error upper-bounds the stochastic error, we conclude that $F(\bar{\beta}) - F(\beta_{\text{avg}}^*) \lesssim \omega^{-2q}$, and by strong convexity, $\|\bar{\beta} - \beta_{\text{avg}}^*\| \lesssim \omega^{-q}$.

Putting it together, we have that

$$\|\beta_{\text{avg}}^*\| \leq \|\beta_{\text{avg}}^* - \bar{\beta}\| + \|\bar{\beta}\| \lesssim \omega^{-q} + 1 \lesssim 1.$$

Substituting into the iteration complexity shows that it suffices to take

$$q^2 (\log \|\beta_{\text{avg}}^*\| + \log q) \lesssim \log^2(1/\bar{\epsilon}) \log \log(1/\bar{\epsilon})$$

iterations, by using our condition on q . □

A.2 Implementation Details

A.2.1 Calibrating Uniform Discretization

To apply uniform discretization in practice one must decide i) the number of grid points to use, and ii) the number of gradient calls to make at each grid point. Loosely speaking, our approach to these two decisions is to first fix a set of desired “target” solution path errors denoted Δ . Then, for each $\delta \in \Delta$, we determine the number of grid points and gradient calls to approximately achieve a solution path error of δ .

More specifically, [36] suggests that to achieve a solution path error of δ , we require $O(1/\sqrt{\delta})$ grid points. Thus, for each $\delta \in \Delta$, we construct a uniform discretization with $1/\sqrt{\delta}$ grid points across every dimension of the hyperparameter space. We denote this grid by $G(\delta)$. Recall that deterministic gradient descent requires $O(\log(1/\delta))$ steps to achieve

an error of $O(\delta)$. Motivated by this result, we use $c_1 \log(c_2/\delta)$ gradient calls per grid point. The total number of gradient calls is thus $c_1 \log(c_2/\delta)/\sqrt{\delta}$.

The specific values of c_1 , c_2 , and Δ used in our experiments are recorded in Table A.1.

Experiment	c_1	c_2	Δ grid
Weighted Binary Classification	1	0.5	$\{2^{-6}, 2^{-6.5}, 2^{-7}, \dots, 2^{-18}\}$
Portfolio Allocation	0.65	1	$\{4^{-2}, 5^{-2}, \dots, 19^{-2}\}$

Table A.1: **Parameters Used in Experiments.** The table reports the constants c_1 and c_2 and the discretization grid Δ used for the weighted binary classification and portfolio allocation experiments.

We next argue that this procedure with these constants is roughly efficient. Define the *grid pass error* as

$$\sup_{\lambda \in G(\delta)} h(\hat{\boldsymbol{\theta}}(\lambda), \lambda) - h(\boldsymbol{\theta}^*(\lambda), \lambda).$$

For comparison, the solution-path error is

$$\epsilon_{\text{sp}}(\hat{\boldsymbol{\theta}}(\cdot)) = \sup_{\lambda \in \Lambda} h(\hat{\boldsymbol{\theta}}(\lambda), \lambda) - h(\boldsymbol{\theta}^*(\lambda), \lambda).$$

Since $G(\delta) \subseteq \Lambda$, the grid pass error is always less than the solution path error.

By comparing the grid pass error, the solution path error, and the target solution path error, we can calibrate the amount of work done at each grid point and the total number of grid points. Specifically, if the grid pass error is much smaller than the solution pass error, it suggests we have allocated too much work to gradient calls at grid points and have insufficient grid points. On the other hand, if the grid pass error is very far from the target solution path error, it suggests we have not allocated enough work to gradient calls at the grid points and have too many grid points. Based on this trade-off, we tune the constants c_1 and c_2 .

Figure A.1 compares the solution path error and grid pass error to the target solution path error for our experiments. Figure A.1a illustrates that in the weighted binary classification experiment, the discretization scheme performs well, satisfying both objectives as the solution path error and grid pass error closely align with the target solution path error. Figure A.1b demonstrates that for the portfolio allocation problem, both solution path error and grid pass error are still close to each other, but fall below the target solution path error. This is primarily due to the high precision of L-BFGS, though only very few gradient calls are performed at each grid.

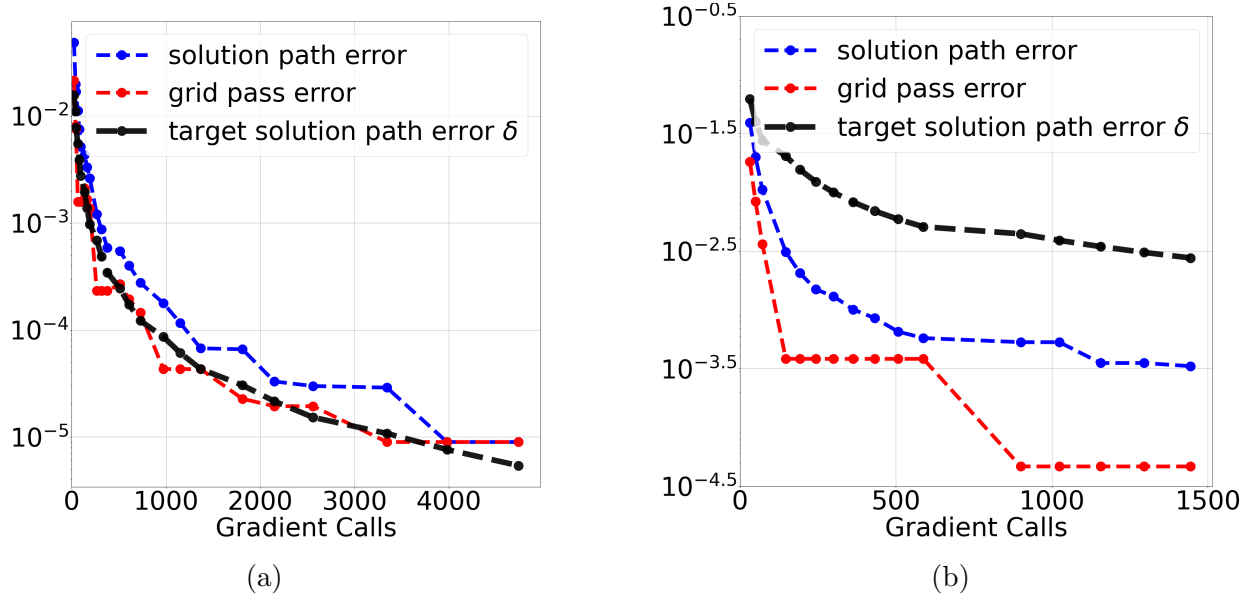


Figure A.1: **Compare: Ideal Stopping Criterion, Grid Pass Error, and Solution Path Error.** (a) weighted binary classification; (b) portfolio allocation. Solution path error and grid pass error are plotted against the corresponding total number of gradient calls incurred in practice. Target solution path error δ is plotted against $c_1 \log(c_2/\delta)/\sqrt{\delta}$, which also equals the total number of gradient calls incurred in practice. Each discrete point in the line plots above corresponds to a discretization grid point $\delta \in G(\delta)$; thus smaller δ is associated with more gradient calls.

A.2.2 Basis for Moderate Dimensional Portfolio Allocation

Recall that this experiment uses the following objective function:

$$h(\theta, \lambda) = -\lambda_1 \cdot \mu^\top \theta + \lambda_2 \cdot \theta^\top \Sigma \theta + \|\theta - \lambda_{3:12}\|_2^2,$$

where $\lambda_{3:12}$ represents the current holdings.

Since this is a quadratic objective, let us first directly compute the optimal solution path

$$\theta^*(\lambda) = \frac{1}{2} (+ \lambda_2 \Sigma)^{-1} (\lambda_1 \mu + 2\lambda_{3:12}).$$

Consider the eigendecomposition of Σ , which takes the form $\Sigma = P^\top D P$ for some orthonormal matrix P and diagonal matrix $D = \text{diag}(d_1, \dots, d_{10})$. Plugging into the above closed form optimal solution path yields

$$\theta^*(\lambda) = \frac{1}{2} P^\top \text{diag} \left(\frac{1}{1 + \lambda_2 d_1}, \dots, \frac{1}{1 + \lambda_2 d_{10}} \right) P (\lambda_1 \mu + 2\lambda_{3:12}).$$

Using the Taylor series expansion for $\frac{1}{1+a\lambda_2}$ around 0

$$\frac{1}{1+a\lambda_2} = \sum_{i=0}^{\infty} (-1)^i (a\lambda_2)^i, \quad a \in \mathbb{R},$$

we conclude that, as $q \rightarrow \infty$, each component of the optimal solution path $\theta_i^*(\lambda)$ is in the span of our chosen basis:

$$\lambda_{(j \bmod 12)} \cdot \lambda_2^{\lfloor j/12 \rfloor}, \quad j = 1, \dots, q.$$

A.3 Additional Figures

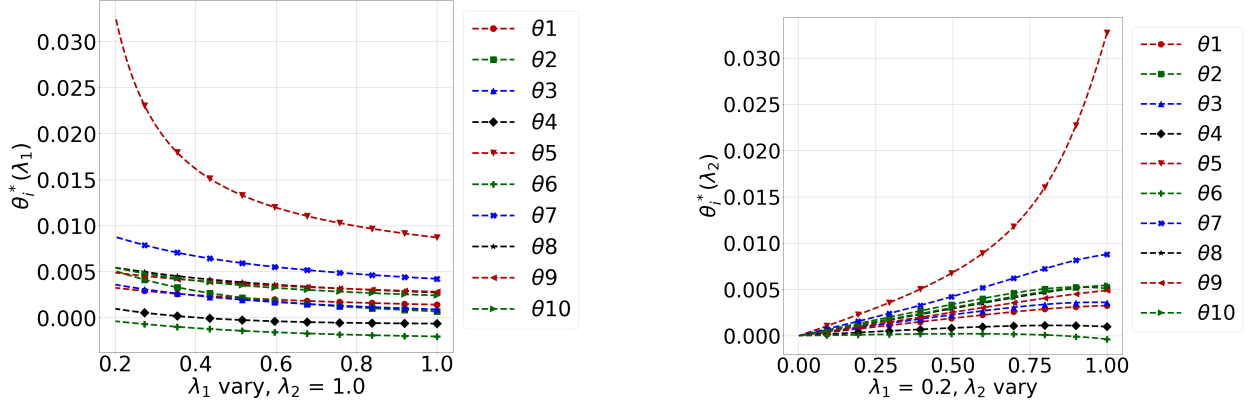


Figure A.2: $\theta^*(\lambda)$ for Portfolio Allocation from Section 2.6.2. The true solution paths are very smooth in each dimension, suggesting that this problem is highly interpolable by a polynomial basis.

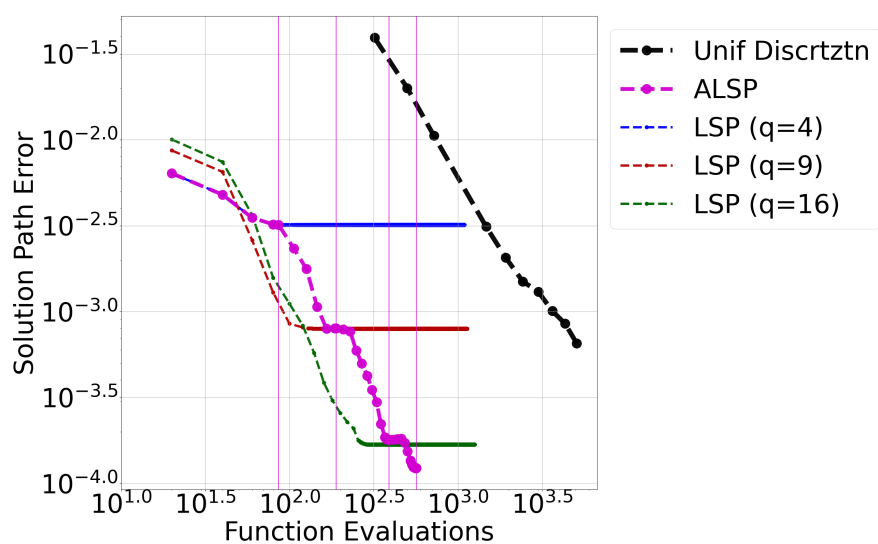


Figure A.3: **LSP and ALSP for Portfolio Allocation.** Compares methods using the first $q = 4, 9, 16$ bivariate-Legendre polynomials. Differs from Figure 2.3 as the y -axis records the number of function evaluations during line-search of L-BFGS instead of gradient calls.

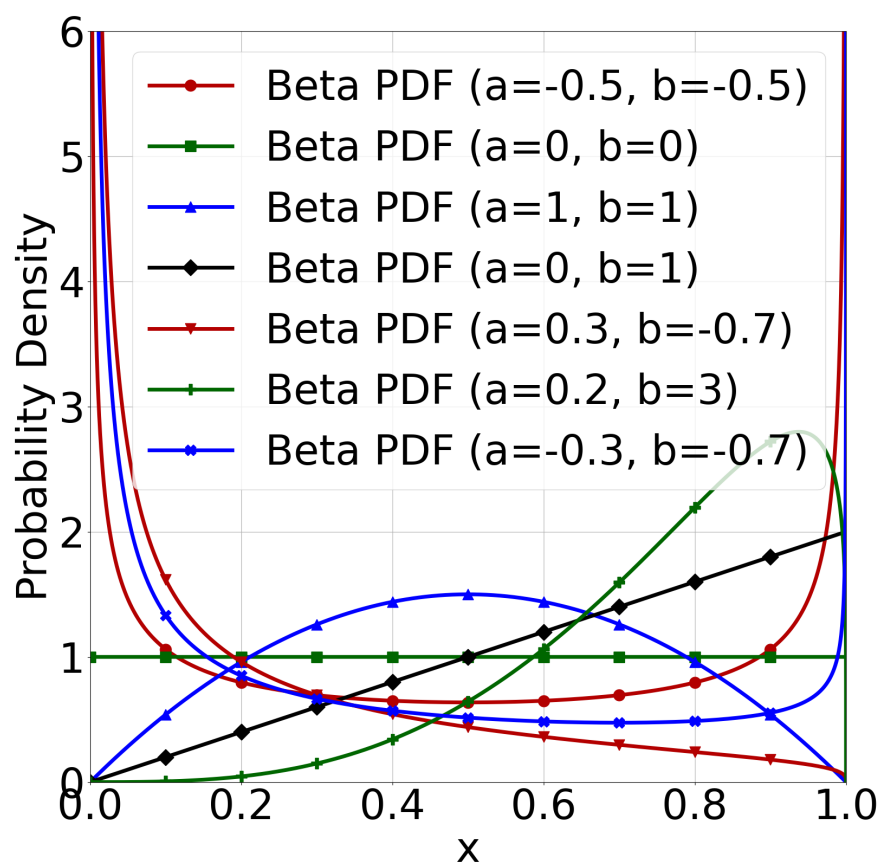


Figure A.4: Probability Density Function(PDF) of Beta Distribution over $[0, 1]$ with Various a, b .