Sparsity and robustness in modern statistical estimation

by

Martin Steven Copenhaver

B.S., Applied Mathematics, Georgia Institute of Technology (2013)

Submitted to the Sloan School of Management
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Operations Research

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2018

Author .....
Sloan School of Management
May 8, 2018

Certified by .....
Dimitris J. Bertsimas
Boeing Leaders for Global Operations Professor
Co-Director, Operations Research Center
Thesis Supervisor

Accepted by .....
Patrick Jaillet
Dugald C. Jackson Professor
Department of Electrical Engineering and Computer Science
Co-Director, Operations Research Center

Sparsity and robustness in modern statistical estimation

by

Martin Steven Copenhaver

Submitted to the Sloan School of Management
on May 8, 2018, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Operations Research

Abstract

Two principles at the forefront of modern machine learning and statistics are sparse modeling and robustness. Sparse modeling enables the construction of simpler statistical models, with examples including the Lasso and matrix completion. At the same time, statistical models need to be robust—they should perform well when data is noisy—in order to make reliable decisions.

While sparsity and robustness are often closely related, the exact relationship and subsequent trade-offs are not always transparent. For example, convex penalties like the Lasso are often motivated by sparsity considerations, yet the success of these methods is also driven by their robustness. In this thesis, we *develop new statistical methods* for sparse and robust modeling and *clarify the relationship* between these two principles.

The first portion of the thesis focuses on a new methodological approach to the old multivariate statistical problem of *Factor Analysis*: finding a low-dimensional description of covariance structure among a set of random variables. Here we propose and analyze a practically tractable family of estimators for this problem. Our approach allows us to exploit bilinearities and eigenvalue structure and thereby show that convex heuristics obtain optimal estimators in many instances.

In the latter portion of the thesis, we focus on developing a unified perspective on various penalty methods employed throughout statistical learning. In doing so, we provide a precise characterization of the relationship between robust optimization and a more traditional penalization approach. Further, we show how the threads of optimization under uncertainty and sparse modeling come together by focusing on the *trimmed Lasso*, a penalization approach to the best subset selection problem. We also contextualize the trimmed Lasso within the broader penalty methods literature by characterizing the relationship with usual separable penalty approaches; as a result, we show that this estimation scheme leads to a richer class of models.

Thesis Supervisor: Dimitris J. Bertsimas
Title: Boeing Leaders for Global Operations Professor
Co-Director, Operations Research Center

Acknowledgments

Completing a thesis is, at times, an exercise in contradictions. In my mind the most prominent example of this phenomenon is the following: while the work is often solitary in nature, the final resulting product is a reflection of input, both direct and indirect, from many different people. As such, I am incredibly grateful and deeply indebted to those who have influenced the work herein by contributing to my personal and intellectual development in the past several years.

The largest influence on my work has been Dimitris. Dimitris views research as a serious enterprise—a means of changing our world for the better. He has taught me that we as researchers (and as humans) are best served if we keep that principle at the heart of everything we do. I do not exaggerate when I say that he has profoundly influenced how I think about conducting research. I am tremendously grateful for his guidance and for our conversations in the last few years. Perhaps what I will take away most from my interactions with Dimitris is that I should not be afraid to challenge assumptions or popular convention.

Special thanks are also due to Rahul, to whom I have often referred as my "co-advisor in spirit." He has been conspicuously generous with his time and his ideas. Rahul's kind spirit is infinitely refreshing. One can only aspire to have his sense of curiosity and enthusiasm for research.

I would also like to extend special gratitude to my second family at MGH. I am eternally grateful to the entire team there, but would like to thank a few specific individuals:

> Cecilia—for being a genuine person, a mentor, and a friend; and for her constant support and advocacy for me. She is a person of truly exceptional character. Cecilia's guidance, insight, and mentorship have had an immeasurable impact on my life, and I extend to her my most heartfelt thanks.
>
> Wilton—for his mentorship, and for constantly inspiring and challenging me with his insatiable curiosity. I aspire to be the kind of leader and mentor that Wilton is, and I am tremendously grateful for all that he continues to teach me.

Bethany—for being an untiring advocate and supporter of mine.

In what follows, I would like to provide a non-exhaustive list of those without whom this thesis would not have been possible:

U.S. Department of Defense—for three years of generous financial support through the National Defense Science and Engineering Graduate Fellowship (NDSEG) Program.

Andrew, Laura, Nikki, and Patrick—for helping with administrative hurdles, and for their dedication to the ORC and its future.

Rob—for his advice and wisdom; for reminding me to advocate for myself; and, finally, for serving on my thesis committee.

Sivaram—for being a mentor and friend, and for reminding me never to forget to do something that positively affects our world.

Jerry and Iain (the tres bandidos)—for friendship and conversations on life, work, chicken, memes, and everything else under the sun.

Daniel, Nishanth, Rajan, Lennart, and Yee Sian—for being friends and collaborators, and for providing sufficient distractions in the office.

Brian and Mew—for sending me countless pictures of cats, among other things.

Cortney—for many years of friendship.

My past students—for inspiring me. Special thanks are due to Allison and Shamir for their untiring devotion to epicurean innovation.

And finally, my family—for everything.

Contents

List of Figures

List of Tables

For Frank E. and Norrell

Chapter 1

Introduction

The ever-increasing availability of complex data has been a substantial force driving modern machine learning and statistics. Two principles at the forefront are sparse modeling and robustness. Sparse modeling enables the construction of simpler statistical models; examples of such techniques include the Lasso and matrix completion. At the same time, statistical models need to be robust—they should perform well when data is noisy—in order to make reliable decisions.

While sparsity and robustness are often closely related, the exact relationship and subsequent trade-offs are not always transparent. For example, convex penalties like the Lasso are often motivated by sparsity considerations, yet the success of these methods is also driven by their robustness. In this thesis, we *develop new statistical methods* for sparse and robust modeling and *clarify the relationship* between these two principles.

The structure of the thesis mirrors these contributions. In particular, the first portion of the thesis focuses on a new methodological approach to an old multivariate statistical problem; then, in the latter portion of the thesis, we focus on developing a unified perspective on various penalty methods employed in statistical learning. Throughout we emphasize the underlying structure of the various optimization problems that arise.

In what follows, we broadly describe the context for the contributions made.

Factor analysis

One classical problem in sparse modeling is finding a low-dimensional representation of the covariance structure among a set of random variables in terms of a smaller number of *hidden factors*. One widely used approach is factor analysis ("FA"), which approximately decomposes the covariance matrix as the sum of two components: a low-rank matrix corresponding to the variances common to all of the random variables; and a diagonal matrix corresponding to the individual variances unique to each random variable.

Despite the ubiquity of factor models, most approaches to FA lack any optimality guarantees or rely on restrictive assumptions about the underlying data which are impossible to verify in practice. Moreover, these approaches can lead to nonsensical estimates such as negative variances.

A modern optimization-based proposal

In Chapter 2, we propose a new family of estimators for FA that uses nonlinear semidefinite optimization, handles problems with thousands of variables, and aids in statistical interpretability by ensuring that the covariance decomposition yields valid positive-semidefinite estimates of variance. A critical component of our approach as compared to others is that we do not rely on assumptions that cannot be verified in practice. Instead, the method produces *optimal* estimators by leveraging techniques like conditional gradient descent to quickly find high-quality solutions which are subsequently used in a branch-and-bound algorithm for global optimization. In particular, our approach exploits the underlying eigenvalue structure of the FA estimation problem, thereby allowing us to synthesize advances in matrix analysis and in global optimization to create a tailored approach.

This chapter appears in large part in the published paper [27].

New perspectives on penalty methods

Penalty methods form the principal focus of the latter portion of the thesis. These methods, such as the Lasso and matrix completion, have seen widespread success in

practice; however, these techniques often have multiple aims and the trade-off between these objectives is not always clear.

Robust optimization: an adversarial perspective

There has been a variety of work in robust optimization ("RO") to show that popular penalty methods correspond exactly to an RO approach, offering a new interpretation of the robustness of such methods. At its core, RO replaces probabilistic primitives with a deterministic *uncertainty set* that restricts allowed deviations from a nominal model; at the same time, the RO approach considers a *worst-case*, pessimistic objective over the uncertainty set.

In Chapter 3, we fully characterize the relationship between RO and the usual penalty-based approach as taken in problems like linear regression and matrix completion. By precisely connecting the RO and penalization problems we show that there is a fine line between the two—indeed, they are often different. This suggests that RO is not merely another view of penalty methods; instead, the RO problem can itself serve as a modeling primitive, giving new avenues for designing estimation schemes. This crucial distinction opens the way for a variety of possible research directions, from the data-driven construction of uncertainty sets to a more fine-grained analysis of the relative merits of the two approaches.

This chapter appears in the published paper [26].

Connecting robustness to sparsity

The RO perspective is also critical in creating a more precise understanding of the relationship between sparsity and robustness. In Chapter 4, we synthesize different views on robustness to show that, in a precise sense, sparse models arise under an *optimistic* model of robustness, in direct contrast with the pessimistic RO viewpoint.

Central to the analysis is a family of nonconvex penalty functions that we call the trimmed Lasso and which bridges the gap between a convex method like the Lasso and nonconvex approaches like best subset selection, which attempts to find the best linear model (in terms of least squares loss) using a specified number of the possible features. We show that the trimmed Lasso and variants thereof lead to a richer

class of estimators than many popularly used nonconvex penalty methods while still being amenable to state-of-the-art techniques in convex and discrete optimization. One immediate practical implication is that because the trimmed Lasso leads to a larger class of estimators, there is potential for such estimators (e.g. as selected via cross-validation) to display superior out-of-sample performance.

This chapter appears in large part in the submitted paper [28].

Chapter 2

Low Rank Factor Analysis

2.1 Introduction

Factor Analysis ("FA") [6, 14, 108], a widely employed methodology in classical and modern multivariate statistics, is used as a tool to obtain a parsimonious representation of the correlation structure among a set of variables in terms of a smaller number of common hidden factors. A basic FA model is of the form $\mathbf{x} = \mathbf{L}\mathbf{f} + \boldsymbol{\epsilon}$, where $\mathbf{x} \in \mathbb{R}^p$ is the observed random vector, $\mathbf{f} \in \mathbb{R}^{r_1}$ (with $r_1 \leq p$, though we do not necessarily restrict r_1 to be small) is a random vector of common factor variables or scores, $\mathbf{L} \in \mathbb{R}^{p \times r_1}$ is a matrix of factor loadings and $\boldsymbol{\epsilon} \in \mathbb{R}^p$ is a vector of uncorrelated random variables. We assume that the variables are mean-centered, $\mathbf{f}$ and $\boldsymbol{\epsilon}$ are uncorrelated, and without loss of generality, the covariance of $\mathbf{f}$ is the identity matrix. We will denote $\mathrm{Cov}(\boldsymbol{\epsilon}) = \boldsymbol{\Phi} = \mathrm{diag}(\Phi_1, \ldots, \Phi_p)$. It follows that

$$\boldsymbol{\Sigma} = \boldsymbol{\Sigma}_c + \boldsymbol{\Phi}, \tag{2.1}$$

where $\boldsymbol{\Sigma}$ is the covariance matrix of $\mathbf{x}$ and $\boldsymbol{\Sigma}_c := \mathbf{L}\mathbf{L}'$ is the covariance matrix corresponding to the common factors. Decomposition (2.1) suggests that $\boldsymbol{\Sigma}$ can be written as the sum of a positive semidefinite ("PSD") matrix $\boldsymbol{\Sigma}_c$ of rank r_1 and a nonnegative diagonal matrix ($\boldsymbol{\Phi}$) corresponding to the errors. In particular, the variance of the ith coordinate of $\mathbf{x} := (x_1, \ldots, x_p)$, i.e., $\mathrm{var}(x_i) = \sum_k L_{ik}^2 + \Phi_i, i = 1, \ldots, p$, splits into

two parts. The first part ($\sum_k L_{ik}^2$) is known as the *communality estimate* (since this is the variance of the factors common to all the x_i's) and the remaining part Φ_i is the variance specific to the ith variable (Φ_i's are also referred to as the *unique variances* or simply *uniquenesses*).

Formulation of the estimator: In decomposition (2.1), the assumption that the rank (r_1) of $\mathbf{\Sigma}_c$ is small compared to p is fairly stringent—see [74, 138, 149] for a historical overview of the concept. In a classical paper of [74], the author argued based on psychometric evidence that $\mathbf{\Sigma}_c$ is often found to have high algebraic rank. In psychometric case studies it is rather rare that the covariance structure can be *completely* explained by a *few* common factors corresponding to mental abilities—in fact, there is evidence of at least hundreds of common factors being present with the number growing without an upper bound. Formally, this means that instead of assuming that $\mathbf{\Sigma}_c$ has *exactly* low-rank it is practical to assume that it can be well-approximated by a low-rank matrix, namely, $\mathbf{L}_1\mathbf{L}_1'$ with $\mathbf{L}_1 \in \mathbb{R}^{p\times r}$. More precisely, $\mathbf{L}_1\mathbf{L}_1'$ is the best rank-r approximation to $\mathbf{\Sigma}_c$ in the matrix q-norm (also known as the Schatten norm), as defined in (2.5), and $(\mathbf{\Sigma}_c - \mathbf{L}_1\mathbf{L}_1')$ is the *residual* component. Following psychometric terminology, $\mathbf{L}_1$ corresponds to the r *most significant* factors representative of mental abilities and the residual $\mathbf{\Sigma}_c - \mathbf{L}_1\mathbf{L}_1'$ corresponds to the remaining psychometric factors unexplained by $\mathbf{L}_1\mathbf{L}_1'$. Thus we can rewrite decomposition (2.1) as

$$\mathbf{\Sigma} = \underbrace{\mathbf{L}_1\mathbf{L}_1'}_{=:\mathbf{\Theta}} + \underbrace{(\mathbf{\Sigma}_c - \mathbf{L}_1\mathbf{L}_1')}_{=:\mathcal{N}} + \mathbf{\Phi}, \tag{2.2}$$

where we use the notation $\mathbf{\Theta} = \mathbf{L}_1\mathbf{L}_1'$ and $\mathcal{N} = (\mathbf{\Sigma}_c - \mathbf{L}_1\mathbf{L}_1')$ with $\mathbf{\Theta} + \mathcal{N} = \mathbf{\Sigma}_c = \mathbf{\Sigma} - \mathbf{\Phi}$. Note that $\mathbf{\Theta}$ denotes the best rank-r approximation to $(\mathbf{\Sigma} - \mathbf{\Phi})$, with the residual component being $\mathcal{N} = \mathbf{\Sigma} - \mathbf{\Phi} - \mathbf{\Theta}$. Note that the entries in $\mathbf{\Phi}$ need to be nonnegative[1] and $\mathbf{\Sigma} - \mathbf{\Phi} \succcurlyeq \mathbf{0}$. In fact, in the words of [149, p. 326],

"*... However, when* $\mathbf{\Sigma} - \mathbf{\Phi}$ *the covariance matrix for the common parts of*

[1]Negative estimates of the diagonals of $\mathbf{\Phi}$ are unwanted since they correspond to variances, but some FA estimation procedures often lead to negative estimates of $\mathbf{\Phi}$—these are popularly known in the literature as Heywood cases and have invited a significant amount of discussion in the community.

the variables, would appear to be indefinite, that would be no less embarrassing than having a negative unique variance in $\mathbf{\Phi}$*...*"

We further refer the reader to [108] for a discussion of the importance of $\mathbf{\Sigma} - \mathbf{\Phi}$ being PSD.[2] We thus have the following natural structural constraints on the parameters:

$$\mathbf{\Theta} \succcurlyeq \mathbf{0}, \quad \mathbf{\Phi} = \operatorname{diag}(\Phi, \ldots, \Phi_p) \succcurlyeq \mathbf{0}, \quad \text{and} \quad \mathbf{\Sigma} - \mathbf{\Phi} \succcurlyeq \mathbf{0}. \tag{2.3}$$

Motivated by the above considerations, we present the following rank-constrained estimation problem for FA:

$$\begin{aligned} \min_{\mathbf{\Theta}, \mathbf{\Phi}} \ & \|\mathbf{\Sigma} - (\mathbf{\Theta} + \mathbf{\Phi})\|_q^q \\ \text{s. t.} \quad & \operatorname{rank}(\mathbf{\Theta}) \leq r \\ & \mathbf{\Theta} \succcurlyeq \mathbf{0} \\ & \mathbf{\Phi} = \operatorname{diag}(\Phi_1, \ldots, \Phi_p) \succcurlyeq \mathbf{0} \\ & \mathbf{\Sigma} - \mathbf{\Phi} \succcurlyeq \mathbf{0}, \end{aligned} \tag{2.4}$$

where $\mathbf{\Theta} \in \mathbb{R}^{p \times p}, \mathbf{\Phi} \in \mathbb{R}^{p \times p}$ are the optimization variables, and for a real symmetric matrix $\mathbf{A} \in \mathbb{R}^{p \times p}$, its matrix q-norm, also known as the Schatten norm (or Schatten-von-Neumann norm), is defined as

$$\|\mathbf{A}\|_q := \left(\sum_{i=1}^{p} |\lambda_i(\mathbf{A})|^q \right)^{1/q}, \tag{2.5}$$

where $\lambda_i(\mathbf{A}),\ i = 1, \ldots, p$, are the (real) eigenvalues of $\mathbf{A}$.

Interpreting the estimator: The estimation criterion (2.4) seeks to *jointly* obtain the (low-rank) common factors and uniquenesses that best explain $\mathbf{\Sigma}$ in terms of minimizing the matrix q-norm of the error $\mathbf{\Sigma} - (\mathbf{\Theta} + \mathbf{\Phi})$ under the PSD constraints (2.3). Note that criterion (2.4) does not necessarily assume that $\mathbf{\Sigma}$ *exactly*

[2] However, the estimation method described in [108] does not guarantee that $\mathbf{\Sigma} - \mathbf{\Phi} \succcurlyeq \mathbf{0}$.

decomposes into a low-rank PSD matrix and a nonnegative diagonal matrix. Problem (2.4) enjoys curious similarities with Principal Component Analysis ("PCA"). In PCA, given a PSD matrix $\mathbf{\Sigma}$ the leading r principal component directions of $\mathbf{\Sigma}$ are obtained by minimizing $\|\mathbf{\Sigma} - \mathbf{\Theta}\|_q$ subject to $\mathbf{\Theta} \succcurlyeq \mathbf{0}$ and $\text{rank}(\mathbf{\Theta}) \leq r$. If the optimal solution $\mathbf{\Phi}$ to Problem (2.4) is *given*, Problem (2.4) is analogous to a rank-r PCA on the residual matrix $\mathbf{\Sigma} - \mathbf{\Phi}$—thus it is naturally desirable to have $\mathbf{\Sigma} - \mathbf{\Phi} \succcurlyeq \mathbf{0}$. In PCA one is interested in understanding the proportion of variance explained by the top-r principal component directions: $\sum_{i=1}^{r} \lambda_i(\mathbf{\Sigma}) / \sum_{i=1}^{p} \lambda_i(\mathbf{\Sigma})$. The denominator $\sum_{i=1}^{p} \lambda_i(\mathbf{\Sigma}) = \text{Tr}(\mathbf{\Sigma})$ accounts for the total variance explained by the covariance matrix $\mathbf{\Sigma}$. Analogously, the proportion of variance explained by $\widehat{\mathbf{\Theta}}_r$ (which denotes the best rank r approximation to $\mathbf{\Sigma} - \mathbf{\Phi}$) is given by $\sum_{i=1}^{r} \lambda_i(\mathbf{\Sigma} - \mathbf{\Phi}) / \sum_{i=1}^{p} \lambda_i(\mathbf{\Sigma} - \mathbf{\Phi})$—for this quantity to be interpretable it is imperative that $\mathbf{\Sigma} - \mathbf{\Phi} \succcurlyeq \mathbf{0}$. In the above argument, of course, we assumed that $\mathbf{\Phi}$ is given. In general, $\mathbf{\Phi}$ needs to be estimated: Problem (2.4) achieves this goal by *jointly* learning $\mathbf{\Phi}$ and $\mathbf{\Theta}$. We note that certain popular approaches of FA (see Sections 2.1.1 and 2.1.2) do not impose the PSD constraint $\mathbf{\Sigma} - \mathbf{\Phi}$ as a part of the estimation scheme—leading to indefinite $\mathbf{\Sigma} - \mathbf{\Phi}$—thereby rendering statistical interpretations troublesome. Our numerical evidence suggests that the quality of estimates of $\mathbf{\Theta}$ and $\mathbf{\Phi}$ obtained from Problem (2.4) outperform those obtained by other competing procedures which do not incorporate the PSD constraints into their estimation criteria.

Choice of r: In exploratory FA, it is standard to consider several choices of r and study the manner in which the proportion of variance explained by the common factors *saturates* with increasing r. We refer the reader to popularly used methods described in [6, 14, 108] and more modern techniques [10, see also references therein] for the choice of r.

Estimate of Covariance Matrix: In the finite sample setting, we set $\mathbf{\Sigma}$ to be the sample covariance matrix. A *consequence* of solving Problem (2.4) is that we get an *estimate* for the covariance matrix given by $\hat{\mathbf{\Theta}} + \hat{\mathbf{\Phi}}$—in this sense, criterion (2.4) can

be viewed as a regularization scheme: the rank constraint on $\mathbf{\Theta}$ encourages parsimony and the PSD constraints encourage interpretability, as discussed above.

In this chapter, we propose a general computational framework to solve Problem (2.4) for any $q \geq 1$. The well-known Schatten q-norm appearing in the loss function is chosen for flexibility—it underlines the fact that our approach can be applied for *any* $q \geq 1$. Note that the estimation criterion (2.4) (even for the case $q = 1$) does not seem to appear in prior work on *approximate minimum rank Factor Analysis* (MRFA) [150, 139]. However, we show in Proposition 1 that Problem (2.4) for the special case $q = 1$ turns out to be equivalent to MRFA. For $q = 2$, the loss function is the familiar squared Frobenius norm also used in MINRES, though the latter formulation is not equivalent to Problem (2.4), as explained in Section 2.1.1. We place more emphasis on studying the computational properties for the more common norms $q \in \{1, 2\}$.

The presence of the rank constraint in Problem (2.4) makes the optimization problem nonconvex. Globally optimizing Problem (2.4), or for that matter obtaining a good stationary point, is quite challenging. We propose a new *equivalent* smooth formulation to Problem (2.4) which does not contain the combinatorial rank constraint. We employ simple and tractable sequential convex relaxation techniques with guaranteed convergence properties and excellent computational properties to obtain a stationary point for Problem (2.4). An important novelty of this chapter is to present certifiable lower bounds on Problem (2.4) without resorting to structural assumptions, thus making it possible to solve Problem (2.4) to *provable* optimality. Towards this end we propose new methods and ideas that incorporate state-of-the-art developments in nonlinear and global optimization.[3]

[3]The class of optimization problems studied here involve global minimization of nonconvex, continuous semidefinite optimization problems. Computational methods for this class of problems are in a nascent stage; further, such methods are significantly less developed when compared to those for mixed integer linear optimization problems, thus posing a major challenge in this work.

2.1.1 A selective overview of related FA estimators

FA has a long and influential history which dates back more than a hundred years. The notion of FA possibly first appeared in [143] for the one factor model, which was then generalized to the multiple factors model by various authors (see, for example, [155]). Significant contributions related to computational methods for FA have been nicely documented in [10, 76, 87, 88, 97, 99, 125, 138, 150], among others.

We will briefly describe some widely used approaches for FA that are closely related to the approach pursued herein and also point out their connections.

Constrained Minimum Trace Factor Analysis (MTFA): This approach [152, 128] seeks to decompose $\mathbf{\Sigma}$ exactly into the sum of a diagonal matrix and a low-rank component, which are estimated via the following convex optimization problem:

$$\begin{array}{ll} \underset{\mathbf{\Theta},\mathbf{\Phi}}{\min} & \mathrm{Tr}(\mathbf{\Theta}) \\ \text{s.t.} & \mathbf{\Theta} \succcurlyeq \mathbf{0} \\ & \mathbf{\Sigma} = \mathbf{\Theta} + \mathbf{\Phi} \\ & \mathbf{\Phi} = \mathrm{diag}(\Phi_1, \ldots, \Phi_p) \succcurlyeq \mathbf{0} \end{array} \tag{2.6}$$

Because $\mathbf{\Theta}$ is PSD, $\mathrm{Tr}(\mathbf{\Theta}) = \sum_{i=1}^{p} \lambda_i(\mathbf{\Theta})$ is a convex surrogate [61] for the rank of $\mathbf{\Theta}$. As such, Problem (2.6) may thus be viewed as a convexification of the rank minimization problem

$$\begin{array}{ll} \underset{\mathbf{\Theta},\mathbf{\Phi}}{\min} & \mathrm{rank}(\mathbf{\Theta}) \\ \text{s.t.} & \mathbf{\Theta} \succcurlyeq \mathbf{0} \\ & \mathbf{\Sigma} = \mathbf{\Theta} + \mathbf{\Phi} \\ & \mathbf{\Phi} = \mathrm{diag}(\Phi_1, \ldots, \Phi_p) \succcurlyeq \mathbf{0}. \end{array} \tag{2.7}$$

In general, Problems (2.6) and (2.7) are *not* equivalent. See [128, 135, 138] and references therein for further connections between the minimizers of (2.6) and (2.7).

A main difference between formulations (2.4) and (2.7) is that the former allows an error in the residual $(\mathbf{\Sigma} - \mathbf{\Theta} - \mathbf{\Phi})$ by constraining $\mathbf{\Theta}$ to have low-rank, unlike (2.7)

which imposes a hard constraint $\mathbf{\Sigma} = \mathbf{\Theta} + \mathbf{\Phi}$. As noted earlier, this can be quite restrictive in various applications. Even if one views Problem (2.6) as imposing a less stringent requirement than that of Problem (2.7), we see two distinct advantages of Problem (2.4) over Problem (2.6): it offers the modeling flexibility of controlling the complexity, viz. rank, of solutions via the choice of r; and it provides smaller of estimates of rank for a comparable amount of explained variance, as substantiated by experimental findings presented in Section 2.5.

Approximate Minimum Rank Factor Analysis (MRFA): This method [150, 139] considers the following optimization problem:

$$\begin{aligned} \min_{\mathbf{\Phi}} \quad & \sum_{i=r+1}^{p} \lambda_i(\mathbf{\Sigma} - \mathbf{\Phi}) \\ \text{s. t.} \quad & \mathbf{\Phi} = \text{diag}(\Phi_1, \ldots, \Phi_p) \succcurlyeq \mathbf{0} \\ & \mathbf{\Sigma} - \mathbf{\Phi} \succcurlyeq \mathbf{0}. \end{aligned} \tag{2.8}$$

Proposition 1 presented below establishes that Problem (2.8) is equivalent to the rank-constrained FA formulation (2.4) for the case $q = 1$. This connection does not appear to be formally established in [150]. We believe that criterion (2.4) for $q = 1$ is easier to interpret as an estimation criterion for FA models over (2.8). [152, 150] describe a method[4] for numerically optimizing (2.8)—as documented in the code for MRFA [151], their implementation can handle problems of size $p \leq 20$.

Principal Component (PC) Factor Analysis: Principal Component factor analysis ("PC") [50, 10] implicitly assumes that $\mathbf{\Phi}$ is a nonnegative scalar multiple of the identity and performs a low-rank PCA on $\mathbf{\Sigma}$. It is not clear how to estimate $\mathbf{\Phi}$ via this method such that $\Phi_i \geq 0$ and $\mathbf{\Sigma} - \mathbf{\Phi} \succcurlyeq \mathbf{0}$. Following [108], the $\mathbf{\Phi}$'s may be estimated after estimating $\widehat{\mathbf{\Theta}}$ via the update rule $\widehat{\mathbf{\Phi}} = \text{diag}(\mathbf{\Sigma} - \widehat{\mathbf{\Theta}})$—the estimates thus obtained, however, need not be nonnegative. Furthermore, it is not guaranteed that the condition $\mathbf{\Sigma} - \mathbf{\Phi} \succcurlyeq \mathbf{0}$ is met.

[4] The method is similar to Algorithm 1 presented herein for the case of $q = 1$; however, [152, 150] rely on a heuristic procedure, as described in [20], for the subproblem with respect to $\mathbf{\Phi}$.

Minimum Residual Factor Analysis (MINRES): This approach [76, 139] considers the problem

$$\min_{\mathbf{L}\in\mathbb{R}^{p\times r}} \quad \sum_{1\le i\neq j\le p} \left(\Sigma_{ij} - (\mathbf{L}\mathbf{L}')_{ij}\right)^2, \tag{2.9}$$

where the sum in the objective is taken over all the off-diagonal entries. Formulation (2.9) is equivalent to the nonconvex optimization problem

$$\begin{aligned} \min_{\boldsymbol{\Theta},\boldsymbol{\Phi}} \quad & \|\boldsymbol{\Sigma} - (\boldsymbol{\Theta} + \boldsymbol{\Phi})\|_2^2 \\ \text{s. t.} \quad & \operatorname{rank}(\boldsymbol{\Theta}) \le r \\ & \boldsymbol{\Theta} \succcurlyeq \mathbf{0} \\ & \boldsymbol{\Phi} = \operatorname{diag}(\Phi_1, \ldots, \Phi_p). \end{aligned} \tag{2.10}$$

Note that the variables Φ_i, $i = 1, \ldots, p$, are unconstrained. If $\widehat{\boldsymbol{\Theta}}$ is a minimizer of Problem (2.10), then *any* $\widehat{\mathbf{L}}$ satisfying $\widehat{\mathbf{L}}\widehat{\mathbf{L}}' = \widehat{\boldsymbol{\Theta}}$ minimizes (2.9) and vice versa.

Various heuristic approaches are used to for Problem (2.9). For example, the `R` package `psych` uses a black box gradient-based tool `optim` to minimize the nonconvex Problem (2.9) with respect to $\mathbf{L}$. Once $\widehat{\mathbf{L}}$ is estimated, the diagonal entries of $\boldsymbol{\Phi}$ are estimated as $\widehat{\Phi}_i = \Sigma_{ii} - (\widehat{\mathbf{L}}\widehat{\mathbf{L}}')_{ii}$ for $i \ge 1$. Note that $\widehat{\Phi}_i$ obtained in this fashion may be negative[5] and the condition $\boldsymbol{\Sigma} - \boldsymbol{\Phi} \succcurlyeq \mathbf{0}$ may be violated.

Generalized Least Squares, Principal Axis and variants: The Ordinary Least Squares (OLS) method for FA [14] considers formulation (2.10) with the additional constraint that $\Phi_i \ge 0\ \forall i$. The Weighted Least Squares (WLS) or the generalized least squares method (see for example, [14]) considers a weighted least squares objective:

$$\|\mathbf{W}\left(\boldsymbol{\Sigma} - (\boldsymbol{\Theta} + \boldsymbol{\Phi})\right)\|_2^2.$$

As in the ordinary least squares case, here too we assume that $\Phi_i \ge 0$. Various choices of $\mathbf{W}$ are possible depending upon the application, with $\mathbf{W} \in \{\boldsymbol{\Sigma}^{-1}, \boldsymbol{\Phi}^{-1}\}$ being a couple of popular choices.

[5] If $\widehat{\Phi}_i < 0$, some ad-hoc procedure is used to threshold it to a nonnegative quantity.

The Principal Axis (PA) FA method [14, 127] is popularly used to estimate factor model parameters based on criterion (2.10) along with the constraints $\Phi_i \geq 0 \; \forall i$. This method starts with a nonnegative estimate $\widehat{\mathbf{\Phi}}$ and performs a rank r eigendecomposition on $\mathbf{\Sigma} - \widehat{\mathbf{\Phi}}$ to obtain $\widehat{\mathbf{\Theta}}$. The matrix $\widehat{\mathbf{\Phi}}$ is then updated to match the diagonal entries of $\mathbf{\Sigma} - \widehat{\mathbf{\Theta}}$, and the above steps are repeated until the estimate $\widehat{\mathbf{\Phi}}$ stabilizes.[6] Note that in this procedure the estimate $\widehat{\mathbf{\Theta}}$ may fail to be PSD and the entries of $\widehat{\Phi}_i$ may be negative as well. Heuristic restarts and various initializations are often carried out to arrive at a reasonable solution (see for example discussions in [14]).

In summary, the least squares stylized methods described above may lead to estimates that violate one or more of the constraints: $\mathbf{\Sigma} - \mathbf{\Phi} \succcurlyeq \mathbf{0}$, $\mathbf{\Theta} \succcurlyeq \mathbf{0}$, and $\mathbf{\Phi} \succcurlyeq \mathbf{0}$.

Maximum Likelihood for Factor Analysis: This approach [9, 87, 108, 132, 133] is another widely used method in FA and typically assumes that the data follows a multivariate Gaussian distribution. This procedure maximizes a likelihood function and is quite different from the loss functions pursued herein and discussed above. The estimator need not exist for any $\mathbf{\Sigma}$—see, e.g., [129].

Most of the methods described in Section 2.1.1 are widely used and their implementations are available in statistical packages `psych` [127], `nFactors` [124], `GPArotation` [21], and others and are publicly available from `CRAN`.[7]

2.1.2 Broad categories of factor analysis estimators

A careful investigation of the methods described above suggests that they can be divided into two broad categories. Some of the above estimators explicitly incorporate a PSD structural assumption on the residual covariance matrix $\mathbf{\Sigma} - \mathbf{\Phi}$ in *addition* to requiring $\mathbf{\Theta} \succcurlyeq \mathbf{0}$ and $\mathbf{\Phi} \succcurlyeq \mathbf{0}$ while the others do not. As already pointed out, these constraints are important for statistical interpretability. We propose to distinguish between the following two broad categories of FA algorithms:

[6] However, we are not aware of a proof showing the convergence of this procedure.
[7] http://cran.us.r-project.org

(A) This category is comprised of FA estimation procedures cast as nonlinear Semidefinite Optimization (SDO) problems—estimation takes place in the presence of constraints of the form $\mathbf{\Sigma} - \mathbf{\Phi} \succcurlyeq \mathbf{0}$, along with $\mathbf{\Theta} \succcurlyeq \mathbf{0}$ and $\mathbf{\Phi} \succcurlyeq \mathbf{0}$. Members of this category are MRFA, MTFA and more generally Problem (2.4).

Existing approaches for these problems are typically not scalable: for example, we are not aware of any algorithm (prior to this work) for Problem (2.8) (MRFA) that scales to covariance matrices with p larger than thirty. Indeed, while theoretical guarantees of optimality exist in certain cases [135], the conditions required for such results to hold are generally difficult to verify in practice.

(B) This category includes classical FA methods which are not based on nonlinear SDO based formulations (as in Category (A)). MINRES, OLS, WLS, GLS, PC and PA based FA estimation procedures (as described in Section 2.1.1) belong to this category. These methods are generally scalable to problem sizes where p is of the order of a few thousand—significantly larger than most procedures belonging to Category (A)—and are implemented in open-source `R`-packages.

Contributions: Our contributions in this chapter may be summarized as follows:

1. We consider a flexible family of FA estimators which can be obtained as solutions to rank-constrained nonlinear SDO problems. In particular, our framework provides a unifying perspective on several existing FA estimation approaches.

2. We propose a novel *exact* reformulation of the rank-constrained FA problem (2.4) as a smooth optimization problem with convex compact constraints. We also develop a unified algorithmic framework utilizing modern optimization techniques to obtain high quality solutions to Problem (2.4). Our algorithms, at every iteration, simply require computing a low-rank eigendecomposition of a $p \times p$ matrix and a structured scalable convex SDO. Our proposal is capable of solving FA problems involving covariance matrices having dimensions up to a few thousand, thereby making it on par with the most scalable FA methods

used currently.[8]

3. Our SDO formulation enables us to estimate the underlying factors and unique variances under the restriction that the residual covariance matrix is PSD—a characteristic that is absent in several popularly used FA methods. This aids statistical interpretability, especially in drawing parallels with PCA and understanding the proportion of variance explained by a given number of factors. Methods proposed herein produce superior quality estimates, in terms of various performance metrics, when compared to existing FA approaches. To our knowledge, this is the first work demonstrating that certifiably optimal solutions to a rank-constrained problem can be found for problems of realistic sizes, without making any assumptions on the underlying data.

4. Using techniques from discrete and global optimization, we develop a branch-and-bound algorithm which proves that the low-rank solutions found are often optimal in seconds for problems on the order of $p = 10$ variables, in minutes for problems on the order of $p = 100$, and in days for some problems on the order of $p = 4000$. As the selected rank increases, so too does the computational burden of proving optimality. It is particularly crucial to note that the optimal solutions for all problems we consider are found very quickly, and that vast majority of computational time is then spent on proving optimality. Hence, for a practitioner who is not particularly concerned with certifying optimality, our techniques for finding feasible solutions provide high-quality estimates quickly.

5. We provide computational evidence demonstrating the favorable performance of our proposed method. Finally, to the best of our knowledge this is the first work that views various FA methods in a unified fashion via a modern optimization lens and attempts to compare a wide range of FA techniques in large scale.

Structure of the chapter: The chapter is organized as follows. In Section 2.1 we propose a flexible family of optimization Problems (2.4) for the task of statistical

[8] An implementation of our approach is available in Appendix D.

estimation in FA models. Section 2.2 presents an exact reformulation of Problem (2.4) as a nonlinear SDO without the rank constraint. Section 2.3 describes the use of nonlinear optimization techniques such as the Conditional Gradient (CG) method [23] adapted to provide feasible solutions (upper bounds) to our formulation. First order methods employed to compute the convex SDO subproblems are also described in the same section. In Section 2.4, we describe our method for certifying optimality of the solutions from Section 2.3 in the case when $q = 1$. In Section 2.5, we present computational results demonstrating the effectiveness of our proposed method in terms of (a) modeling flexibility in the choice of the number of factors r and the parameter q, (b) scalability, and (c) the quality of solutions obtained in a wide array of real and synthetic datasets—comparisons with several existing methods for FA are considered. Section 2.6 contains our conclusions.

2.2 Reformulations of Problem (2.4)

Let $\boldsymbol{\lambda}(\mathbf{A})$ denote the vector of eigenvalues of $\mathbf{A} \in \mathbb{R}^{p\times p}$, arranged in decreasing order, i.e.,

$$\lambda_1(\mathbf{A}) \geq \lambda_2(\mathbf{A}) \geq \ldots \geq \lambda_p(\mathbf{A}). \tag{2.11}$$

The following proposition presents the first reformulation of Problem (2.4) as a continuous eigenvalue optimization problem with convex compact constraints. Proofs of all results can be found in Appendix A.1.

Proposition 1. *(a) For any $q \geq 1$, Problem (2.4) is equivalent to*

$$\begin{array}{ll} \min\limits_{\boldsymbol{\Phi}} & f_q(\boldsymbol{\Phi};\boldsymbol{\Sigma}) := \sum\limits_{i=r+1}^{p} \lambda_i^q(\boldsymbol{\Sigma}-\boldsymbol{\Phi}) \\ \text{s. t.} & \boldsymbol{\Phi} = \operatorname{diag}(\Phi_1,\ldots,\Phi_p) \succcurlyeq \mathbf{0} \\ & \boldsymbol{\Sigma}-\boldsymbol{\Phi} \succcurlyeq \mathbf{0}. \end{array} \tag{CFA$_q$}$$

(b) Suppose $\mathbf{\Phi}^*$ *is a minimizer of Problem* (CFA$_q$)*, and let*

$$\mathbf{\Theta}^* = \mathbf{U}\ \mathrm{diag}\left(\lambda_1(\mathbf{\Sigma}-\mathbf{\Phi}^*),\ldots,\lambda_r(\mathbf{\Sigma}-\mathbf{\Phi}^*),\underbrace{0,\ldots,0}_{p-r\ \text{times}}\right)\mathbf{U}',$$

where $\mathbf{U}\in\mathbb{R}^{p\times p}$ *is the matrix of eigenvectors of* $\mathbf{\Sigma}-\mathbf{\Phi}^*$*. Then* $(\mathbf{\Theta}^*,\mathbf{\Phi}^*)$ *is a solution to Problem* (2.4)*.*

Problem (CFA$_q$) is a nonlinear SDO in $\mathbf{\Phi}$, unlike the original formulation (2.4) that estimates $\mathbf{\Theta}$ and $\mathbf{\Phi}$ jointly. Note that the rank constraint does not appear in Problem (CFA$_q$) and the constraint set of Problem (CFA$_q$) is convex and compact. However, Problem (CFA$_q$) is nonconvex due to the nonconvex objective function $\sum_{i=r+1}^{p}\lambda_i^q(\mathbf{\Sigma}-\mathbf{\Phi})$. For $q=1$ the function appearing in the objective of (CFA$_q$) is concave and for $q>1$, it is neither convex nor concave.

Proposition 2. *The estimation Problem* (2.4) *is equivalent to*

$$\begin{aligned}
\min_{\mathbf{\Theta},\mathbf{\Phi}}\quad & \|\mathbf{\Sigma}-(\mathbf{\Theta}+\mathbf{\Phi})\|_q^q \\
\text{s.t.}\quad & \mathrm{rank}(\mathbf{\Theta})\le r \\
& \mathbf{\Theta}\succcurlyeq\mathbf{0} \\
& \mathbf{\Phi}=\mathrm{diag}(\Phi_1,\ldots,\Phi_p)\succcurlyeq\mathbf{0} \\
& \mathbf{\Sigma}-\mathbf{\Phi}\succcurlyeq\mathbf{0} \\
& \mathbf{\Sigma}-\mathbf{\Theta}\succcurlyeq\mathbf{0}.
\end{aligned} \tag{2.12}$$

Note that Problem (2.12) has an additional PSD constraint $\mathbf{\Sigma}-\mathbf{\Theta}\succcurlyeq\mathbf{0}$ which does not explicitly appear in Problem (2.4). It is interesting to note that the two problems are equivalent. By virtue of Proposition 2, Problem (2.12) can as well be used as the estimation criterion for rank constrained FA. However, we will work with formulation (2.4) because it is easier to interpret from a statistical perspecitve.

Special instances of (CFA$_q$)**:** We show that some well-known FA estimation problems can be viewed as special cases of our general framework.

For $q = 1$, Problem (CFA_q) reduces to MRFA, as described in (2.8). For $q = 1$ and $r = 0$, Problem (CFA_q) reduces to MTFA (2.6). When $q = 2$, we get a variant of (2.10), i.e., a PSD *constrained* analogue of MINRES

$$\begin{aligned} \min_{\boldsymbol{\Phi}} \quad & \sum_{i=r+1}^{p} \lambda_i^2(\boldsymbol{\Sigma} - \boldsymbol{\Phi}) \\ \text{s.t.} \quad & \boldsymbol{\Phi} = \operatorname{diag}(\Phi_1, \ldots, \Phi_p) \succcurlyeq \mathbf{0} \\ & \boldsymbol{\Sigma} - \boldsymbol{\Phi} \succcurlyeq \mathbf{0}. \end{aligned} \tag{2.13}$$

Note that unlike Problem (2.10), Problem (2.13) explicitly imposes PSD constraints on $\boldsymbol{\Phi}$ and $\boldsymbol{\Sigma} - \boldsymbol{\Phi}$. The objective function in (CFA_q) is continuous but non-smooth. The function is differentiable at $\boldsymbol{\Phi}$ if and only if the r and $(r+1)$th eigenvalues of $\boldsymbol{\Sigma} - \boldsymbol{\Phi}$ are distinct [100, 139], i.e., $\lambda_{r+1}(\boldsymbol{\Sigma} - \boldsymbol{\Phi}) < \lambda_r(\boldsymbol{\Sigma} - \boldsymbol{\Phi})$. The non-smoothness of the objective function in Problem (CFA_q) makes the use of standard gradient based methods problematic [23]. Theorem 1 presents a reformulation of Problem (CFA_q) in which the objective function is continuously differentiable.

Theorem 1. *(a) The estimation criterion given by Problem (2.4) is equivalent to*[9]

$$\begin{aligned} \min_{\boldsymbol{\Phi}, \mathbf{W}} \quad & g_q(\mathbf{W}, \boldsymbol{\Phi}) := \operatorname{Tr}(\mathbf{W}(\boldsymbol{\Sigma} - \boldsymbol{\Phi})^q) \\ \text{s.t.} \quad & \boldsymbol{\Phi} = \operatorname{diag}(\Phi_1, \ldots, \Phi_p) \succcurlyeq \mathbf{0} \\ & \boldsymbol{\Sigma} - \boldsymbol{\Phi} \succcurlyeq \mathbf{0} \\ & \mathbf{I} \succcurlyeq \mathbf{W} \succcurlyeq \mathbf{0} \\ & \operatorname{Tr}(\mathbf{W}) = p - r. \end{aligned} \tag{2.14}$$

(b) The solution $\widehat{\boldsymbol{\Theta}}$ of Problem (2.4) can be recovered from the solution $\widehat{\mathbf{W}}, \widehat{\boldsymbol{\Phi}}$ of Problem (2.14) via

$$\widehat{\boldsymbol{\Theta}} := \widehat{\mathbf{U}} \operatorname{diag}(\widehat{\lambda}_1, \ldots, \widehat{\lambda}_r, \underbrace{0, \ldots, 0}_{p-r \text{ times}}) \widehat{\mathbf{U}}', \tag{2.15}$$

where $\widehat{\mathbf{U}}$ is the matrix formed by the p eigenvectors corresponding to the eigen-

[9] For any PSD matrix $\mathbf{A}$, with eigendecomposition $\mathbf{A} = \mathbf{U}_A \operatorname{diag}(\lambda_1, \ldots, \lambda_p)\mathbf{U}_A'$, we define $\mathbf{A}^q := \mathbf{U}_A \operatorname{diag}(\lambda_1^q, \ldots, \lambda_p^q)\mathbf{U}_A'$, for any $q \geq 1$.

values $\widehat{\lambda}_1, \ldots, \widehat{\lambda}_p$ *(arranged in decreasing order) of the matrix* $\mathbf{\Sigma} - \widehat{\mathbf{\Phi}}$. *Given* $\widehat{\mathbf{\Phi}}$, *any solution* $\widehat{\mathbf{\Theta}}$ *(given by* (2.15)*) is independent of* q.

In Problem (2.14), if we partially minimize the function $g_q(\mathbf{W}, \mathbf{\Phi})$ over $\mathbf{\Phi}$ (with fixed $\mathbf{W}$), the resulting function is concave in $\mathbf{W}$. This observation leads to the following proposition.

Proposition 3. *The function* $G_q(\mathbf{W})$ *obtained upon (partially) minimizing* $g_q(\mathbf{W}, \mathbf{\Phi})$ *over* $\mathbf{\Phi}$ *(with* $\mathbf{W}$ *fixed) in Problem* (2.14), *given by*

$$\begin{aligned} G_q(\mathbf{W}) := \min_{\mathbf{\Phi}} \quad & g_q(\mathbf{W}, \mathbf{\Phi}) \\ \text{s.t.} \quad & \mathbf{\Phi} = \operatorname{diag}(\Phi_1, \ldots, \Phi_p) \succcurlyeq \mathbf{0} \\ & \mathbf{\Sigma} - \mathbf{\Phi} \succcurlyeq \mathbf{0}, \end{aligned} \tag{2.16}$$

is concave in $\mathbf{W}$. *The subgradients of the function* $G_q(\mathbf{W})$ *exist and are given by*

$$\nabla G_q(\mathbf{W}) = (\mathbf{\Sigma} - \widehat{\mathbf{\Phi}}(\mathbf{W}))^q, \tag{2.17}$$

where $\widehat{\mathbf{\Phi}}(\mathbf{W})$ *is a minimizer of the convex optimization Problem* (2.16).

In light of Proposition 3, we present another reformulation of Problem (2.4) as the following concave minimization problem:

$$\begin{aligned} \min_{\mathbf{W}} \quad & G_q(\mathbf{W}) \\ \text{s.t.} \quad & \mathbf{I} \succcurlyeq \mathbf{W} \succcurlyeq \mathbf{0} \\ & \operatorname{Tr}(\mathbf{W}) = p - r, \end{aligned} \tag{2.18}$$

where the function $G_q(\mathbf{W})$ is differentiable if and only if $\widehat{\mathbf{\Phi}}(\mathbf{W})$ is unique.

Note that by virtue of Proposition 1, Problems (CFA_q) and (2.18) are equivalent. Therefore, it is natural to ask whether one formulation might be favored over the other from a computational perspective. Towards this end, note that both Problems (CFA_q) and (2.18) involve the minimization of a non-smooth objective function, over convex compact constraints. However, the objective function of Problem (CFA_q)

is nonconvex (for $q > 1$) whereas the one in Problem (2.18) is concave (for all $q \geq 1$). We will see in Section 2.3 that CG-based algorithms can be applied to a concave minimization problem (even if the objective function is not differentiable); however, CG applied to general non-smooth objective functions has limited convergence guarantees. Thus, formulation (2.18) is readily amenable to CG-based optimization algorithms, unlike formulation (CFA$_q$). This seems to make Problem (2.18) computationally more appealing than Problem (CFA$_q$).

2.3 Finding upper bounds

This section presents a unified computational framework for the class of problems (CFA$_q$). Problem (2.14) is a nonconvex smooth optimization problem and obtaining a stationary point is quite challenging. We propose iterative schemes based on the Conditional Gradient (CG) algorithm [23]—a generalization of the Frank-Wolfe algorithm [66]—to obtain a stationary point of the problem. The appealing aspect of our framework is that every iteration of the algorithm requires solving a convex SDO problem which is computationally tractable. While off-the-shelf interior point algorithms—for example `SDPT3` [158], `Yalmip` [107], and `MOSEK` [5]—can be used to solve the convex SDO problems, they typically do not scale well for large problems due to intensive memory requirements. In this vein, first order algorithms have received a lot of attention [116, 117, 118] in convex optimization of late, due to their low cost per iteration, low-memory requirements, and ability to deliver solutions of moderate accuracy for large problems within a modest time limit. We use first order methods to solve the convex SDO problems. We present one primary scheme based on the CG algorithm:

> **Algorithm 1:** This scheme, described in Section 2.3.1, applies CG on the optimization Problem (2.18), where the function $G_q(\mathbf{W})$ defined in (2.16) is concave (and possibly non-smooth).

In addition, in Appendix A.2 we present an alternative approach that applies CG to Problem (2.14), where the objective function $g_q(\mathbf{W}, \mathbf{\Phi})$ is smooth.

To make notation simpler, we will use the following shorthand:

$$\mathcal{W}_{p-r} = \{\mathbf{W} \in \mathbb{R}^{p\times p} : \mathbf{I} \succcurlyeq \mathbf{W} \succcurlyeq \mathbf{0},\ \ \mathrm{Tr}(\mathbf{W}) = p - r\}$$
$$\mathcal{F}_{\mathbf{\Sigma}} = \{\mathbf{\Phi} : \mathbf{\Sigma} - \mathbf{\Phi} \succcurlyeq \mathbf{0},\ \ \mathbf{\Phi} = \mathrm{diag}(\Phi_1, \ldots, \Phi_p) \succcurlyeq \mathbf{0}\}.$$

2.3.1 A CG-based algorithm for (2.18)

The CG method for Problem (2.18) requires solving a linearization of the concave objective function. At iteration k, if $\mathbf{W}^{(k)}$ is the current estimate of $\mathbf{W}$, the new estimate $\mathbf{W}^{(k+1)}$ is obtained by

$$\mathbf{W}^{(k+1)} \in \underset{\mathbf{W}\in\mathcal{W}_{p-r}}{\arg\min}\ \left\langle \nabla G_q(\mathbf{W}^{(k)}), \mathbf{W}\right\rangle = \underset{\mathbf{W}\in\mathcal{W}_{p-r}}{\arg\min}\ \mathrm{Tr}(\mathbf{W}(\mathbf{\Sigma} - \mathbf{\Phi}^{(k)})^q), \tag{2.19}$$

where by Proposition 3, $(\mathbf{\Sigma} - \mathbf{\Phi}^{(k)})^q$ is a subgradient of $G_q(\mathbf{W})$ at $\mathbf{W}^{(k)}$ with $\mathbf{\Phi}^{(k)}$ given by

$$\mathbf{\Phi}^{(k)} \in \underset{\mathbf{\Phi}\in\mathcal{F}_{\mathbf{\Sigma}}}{\arg\min}\ \ \mathrm{Tr}(\mathbf{W}^{(k)}(\mathbf{\Sigma} - \mathbf{\Phi})^q) \tag{2.20}$$

No explicit line search is necessary here because the minimum will always be at the new point, i.e., $\mathbf{\Phi}^{(k)}$, due to the concavity of the objective function. The sequence $\mathbf{W}^{(k)}$ is recursively computed via (2.19) until the convergence criterion

$$G_q(\mathbf{W}^{(k)}) - G_q(\mathbf{W}^{(k+1)}) \leq \texttt{TOL} \cdot G_q(\mathbf{W}^{(k)}), \tag{2.21}$$

is met for some user-defined tolerance $\texttt{TOL} > 0$. A short description of the procedure appears in Algorithm 1.

Algorithm 1 A CG based algorithm for formulation (2.18)

1. Initialize with $\mathbf{W}^{(1)}$ ($k = 1$), feasible for Problem (2.18) and repeat, for $k \geq 2$, Steps 2-3 until convergence criterion (2.21) is satisfied.
2. Update $\mathbf{\Phi}$ (with $\mathbf{W}$ fixed) by solving (2.20).
3. Update $\mathbf{W}$ (with $\mathbf{\Phi}$ fixed) by solving (2.19), to get $\mathbf{W}^{(k+1)}$.

Before we present the convergence rate of Algorithm 1, we will need to introduce some notation. For any point $\overline{\mathbf{W}}$ belonging to the feasible set of Problem (2.18) let us define $\Delta(\overline{\mathbf{W}})$ as follows:

$$\Delta(\overline{\mathbf{W}}) := \min_{\mathbf{W} \in \mathcal{W}_{p-r}} \langle \nabla G_q(\overline{\mathbf{W}}), \mathbf{W} - \overline{\mathbf{W}} \rangle. \tag{2.22}$$

Further, $\mathbf{W}^*$ satisfies the first order stationary condition for Problem (2.18) if $\mathbf{W}^*$ is feasible for the problem and $\Delta(\mathbf{W}^*) \geq 0$.

We now present Theorem 2 establishing the rate of convergence and associated convergence properties of Algorithm 1. The proof (along with all other omitted proofs) is contained in Appendix A.1.

Theorem 2. *If $\mathbf{W}^{(k)}$ is a sequence produced by Algorithm 1, then $G_q(\mathbf{W}^{(k)})$ is a monotone decreasing sequence and every limit point $\mathbf{W}^{(\infty)}$ of the sequence $\mathbf{W}^{(k)}$ is a stationary point of Problem (2.18). Furthermore, Algorithm 1 has a convergence rate of $O(1/K)$ (with K denoting the iteration index) to a first order stationary point of Problem (2.18), i.e.,*

$$\min_{i=1,\ldots,K} \left\{ -\Delta(\mathbf{W}^{(i)}) \right\} \leq \frac{G_q(\mathbf{W}^{(1)}) - G_q(\mathbf{W}^{(\infty)})}{K}. \tag{2.23}$$

2.3.2 Solving the convex SDO problems

Algorithm 1 requires sequentially solving convex SDO problems in $\mathbf{W}$ and $\mathbf{\Phi}$. We describe herein how these subproblems can be solved efficiently.

Solving the SDO problem with respect to W

A generic SDO problem associated with Problem (2.19) requires updating $\mathbf{W}$ as

$$\widehat{\mathbf{W}} \quad \in \quad \underset{\mathbf{W} \in \mathcal{W}_{p-r}}{\arg\min} \quad \langle \mathbf{W}, \widetilde{\mathbf{W}} \rangle, \tag{2.24}$$

for some fixed symmetric $\widetilde{\mathbf{W}} \in \mathbb{R}^{p \times p}$, depending upon the algorithm and the choice of q. For Algorithm 1 the update in $\mathbf{W}$ at iteration k for Problem (2.19), corresponds

to $\widetilde{\mathbf{W}} = (\boldsymbol{\Sigma} - \boldsymbol{\Phi}^{(k+1)})^q$.

A solution to Problem (2.24) is given by $\widehat{\mathbf{W}} = \sum_{i=r+1}^{p} \mathbf{u}_i \mathbf{u}_i'$, where $\mathbf{u}_1, \ldots, \mathbf{u}_p$ are the eigenvectors of the matrix $\widetilde{\mathbf{W}}$, corresponding to the eigenvalues $\lambda_1(\widetilde{\mathbf{W}}), \ldots, \lambda_p(\widetilde{\mathbf{W}})$.

Solving the SDO problem with respect to $\boldsymbol{\Phi}$

The SDO problem arising from the update of $\boldsymbol{\Phi}$ is not as straightforward as the update with respect to $\mathbf{W}$. Before presenting the general case, it helps to consider a few special cases of (CFA$_q$). For $q = 1$ the objective function of Problem (2.14)

$$g_1(\mathbf{W}, \boldsymbol{\Phi}) = \langle \mathbf{W}, \boldsymbol{\Sigma} \rangle - \sum_{i=1}^{p} w_{ii} \Phi_i \tag{2.25}$$

is linear in $\boldsymbol{\Phi}$ (for fixed $\mathbf{W}$). For $q = 2$, the objective function of Problem (2.14)

$$g_2(\mathbf{W}, \boldsymbol{\Phi}) = \mathrm{Tr}(\mathbf{W}\boldsymbol{\Sigma}^2) + \sum_{i=1}^{p} (w_{ii}\Phi_i^2 - 2\langle \mathbf{w}_i, \boldsymbol{\sigma}_i \rangle \Phi_i) \tag{2.26}$$

is a convex quadratic in $\boldsymbol{\Phi}$ (for fixed $\mathbf{W}$).

For Algorithm 1, the partial minimizations with respect to $\boldsymbol{\Phi}$, for $q = 1$ and $q = 2$, require minimizing Problems (2.25) and (2.26), respectively.

Various instances of optimization problems with respect to $\boldsymbol{\Phi}$ such as those appearing in Algorithm 1 can be viewed as special cases of the following family of SDO problems:

$$\min_{\boldsymbol{\Phi} \in \mathcal{F}_{\boldsymbol{\Sigma}}} \quad \sum_{i=1}^{p} (c_i \Phi_i^2 + d_i \Phi_i) \tag{2.27}$$

where $c_i \geq 0$ and d_i for $i = 1, \ldots, p$ are problem parameters that depend upon the choice of algorithm and q. We now present a first order convex optimization scheme for solving (2.27).

A first order scheme for (2.27)

With the intention of providing a simple and scalable algorithm for the convex SDO problem, we use the Alternating Direction Method of Multipliers [23, 37] (ADMM).

We introduce a splitting variable $\mathbf{\Lambda} = \mathbf{\Sigma} - \mathbf{\Phi}$ and rewrite Problem (2.27) in the following equivalent form:

$$\begin{aligned} \min_{\mathbf{\Phi},\mathbf{\Lambda}} \quad & \sum_{i=1}^{p}(c_i\Phi_i^2 + d_i\Phi_i) \\ \text{s.t.} \quad & \mathbf{\Phi} = \text{diag}(\Phi_1,\ldots,\Phi_p) \succcurlyeq \mathbf{0} \\ & \mathbf{\Lambda} \succcurlyeq \mathbf{0} \\ & \mathbf{\Lambda} = \mathbf{\Sigma} - \mathbf{\Phi}. \end{aligned} \tag{2.28}$$

The Augmented Lagrangian for the above problem is:

$$\mathcal{L}_\rho(\mathbf{\Phi},\mathbf{\Lambda},\boldsymbol{\nu}) := \sum_{i=1}^{p}(c_i\Phi_i^2 + d_i\Phi_i) + \langle \boldsymbol{\nu}, \mathbf{\Lambda} - (\mathbf{\Sigma} - \mathbf{\Phi})\rangle + \frac{\rho}{2}\|\mathbf{\Lambda} - (\mathbf{\Sigma} - \mathbf{\Phi})\|_2^2, \tag{2.29}$$

where $\rho > 0$ is a scalar and $\langle\cdot,\cdot\rangle$ denotes the standard trace inner product. ADMM involves the following three updates:

$$\mathbf{\Phi}^{(k+1)} \in \underset{\mathbf{\Phi}=\text{diag}(\Phi_1,\ldots,\Phi_p)\succcurlyeq\mathbf{0}}{\arg\min} \mathcal{L}_\rho(\mathbf{\Phi},\mathbf{\Lambda}^{(k)},\boldsymbol{\nu}^{(k)}), \tag{2.30}$$

$$\mathbf{\Lambda}^{(k+1)} \in \underset{\mathbf{\Lambda}\succcurlyeq\mathbf{0}}{\arg\min} \mathcal{L}_\rho(\mathbf{\Phi}^{(k+1)},\mathbf{\Lambda},\boldsymbol{\nu}^{(k)}), \tag{2.31}$$

$$\boldsymbol{\nu}^{(k+1)} = \boldsymbol{\nu}^{(k)} + \rho(\mathbf{\Lambda}^{(k+1)} - (\mathbf{\Sigma} - \mathbf{\Phi}^{(k+1)})), \tag{2.32}$$

and produces a sequence $\{(\mathbf{\Phi}^{(k)},\boldsymbol{\lambda}^{(k)},\boldsymbol{\nu}^{(k)})\}_{k\geq 1}$; the convergence properties of the algorithm are quite well-known [37].

Problem (2.30) can be solved in closed form as

$$\Phi_i^{(k+1)} = \frac{\rho}{\rho + 2c_i}\max\left\{(\sigma_{ii} - \lambda_{ii}^{(k)}) - \frac{(d_i + \nu_{ii}^{(k)})}{\rho}, 0\right\}, \quad i = 1,\ldots,p. \tag{2.33}$$

The update with respect to $\mathbf{\Lambda}$ in (2.31) requires an eigendecomposition:

$$\begin{aligned}\mathbf{\Lambda}^{(k+1)} =& \arg\min_{\mathbf{\Lambda}\succcurlyeq\mathbf{0}} \left\|\mathbf{\Lambda} - (\mathbf{\Sigma} - \mathbf{\Phi}^{(k+1)} - \frac{1}{\rho}\boldsymbol{\nu}^{(k)})\right\|_2^2 \\ =& \mathcal{P}_{S_p^+}\left(\mathbf{\Sigma} - \mathbf{\Phi}^{(k+1)} - \frac{1}{\rho}\boldsymbol{\nu}^{(k)}\right),\end{aligned} \tag{2.34}$$

where the operator $\mathcal{P}_{S_p^+}(\mathbf{A})$ denotes the projection of a symmetric matrix $\mathbf{A}$ onto the cone of PSD matrices of dimension $p \times p$:

$$\mathcal{P}_{S_p^+}(\mathbf{A}) = \mathbf{U}_A \operatorname{diag}\left(\max\left\{\lambda_1, 0\right\}, \ldots, \max\left\{\lambda_p, 0\right\}\right) \mathbf{U}_A',$$

where $\mathbf{A} = \mathbf{U}_A \operatorname{diag}(\lambda_1, \ldots, \lambda_p)\mathbf{U}_A'$ is the eigendecomposition of $\mathbf{A}$.

Stopping criterion: The ADMM iterations (2.30)—(2.32) are continued till the values of $\|\mathbf{\Lambda}^{(k+1)} - (\mathbf{\Sigma} - \mathbf{\Phi}^{(k+1)})\|_2$ and the relative change in the objective values of Problem (2.27) become smaller than a certain threshold, say, $\texttt{TOL} \times \alpha$, where $\alpha \in \{10^{-1}, \ldots, 10^{-3}\}$—this is typically taken to be smaller than the convergence threshold for the CG iterations ($\texttt{TOL}$).

Computational cost of Problem (2.27): The most intensive computational stage in the ADMM procedure is in performing the projection operation (2.34); this requires $O(p^3)$ operations due to the associated eigendecomposition. This needs to be done for as many ADMM steps, until convergence.

Since Problem (2.27) is embedded inside iterative procedures like Algorithm 1, the estimates of $(\mathbf{\Phi}, \mathbf{\Lambda}, \boldsymbol{\nu})$ obtained by solving Problem (2.27) for a iteration index (of the CG algorithm) provides a good warm-start for the Problem (2.27) in the subsequent CG iteration. This is often found to decrease the number of iterations required by the ADMM algorithm to converge to a prescribed level of accuracy.[10]

[10]The utility of warm starts is another compelling reason to apply a first-order-based approach instead of interior point methods. Indeed, warm starts are well-known to perform quite poorly when incorporated into interior point methods—often they can perform worse than cold starts [163, 86]. Given the need to repeatedly solve similarly structured SDOs for both upper bounds (as presented in this section) as well as lower bounds (as presented in Section 2.4), the ability to effectively incorporate warm start information is crucial.

2.3.3 Computational cost of Algorithm 1

For Algorithm 1 and other CG-based algorithms for factor analysis (see Appendix A.2), the computational bottleneck is in performing the eigendecomposition of a $p \times p$ matrix: the $\mathbf{W}$ update requires performing a low-rank eigendecomposition of a $p \times p$ matrix and the $\mathbf{\Phi}$ update requires solving a problem of the form (2.27), which also costs $O(p^3)$. Since eigendecompositions can easily be done for p of the order of a few thousands, the proposed algorithms can be applied to that scale.

Note that most existing popular algorithms for FA belonging to Category (B) (see Section 2.1.2) also perform an eigendecomposition with cost $O(p^3)$. Thus it appears that Category (B) and the algorithms proposed herein have the same computational complexity and hence these two estimation approaches are equally scalable.

2.4 Certificates of optimality via lower bounds

In this section, we outline our approach to computing lower bounds to (CFA_q) via techniques from global optimization and matrix analysis. In particular, we focus on the case when $q = 1$.[11] We begin with an overview of the method. We then discuss initialization parameters for the method as well as branching rules and other refinements employed in our approach.

2.4.1 Overview of method

Our primary problem of interest is to provide lower bounds to (CFA_1), i.e.,

$$\min_{\mathbf{\Phi} \in \mathcal{F}_{\mathbf{\Sigma}}} \sum_{i=r+1}^{p} \lambda_i(\mathbf{\Sigma} - \mathbf{\Phi}), \tag{2.35}$$

[11] The general case for $q > 1$ can be addressed using similar (although more complicated) techniques as applied to Problem (2.14), again applying principles developed in global optimization.

or equivalently via Theorem 1,

$$\min_{\substack{\mathbf{W}\in\mathcal{W}_{p-r}\\ \mathbf{\Phi}\in\mathcal{F}_{\mathbf{\Sigma}}}} \langle \mathbf{W}, \mathbf{\Sigma} - \mathbf{\Phi}\rangle. \tag{2.36}$$

One possible approach is to consider *convex* lower bounds to (2.36).

Definition 1. *For a function $f : \Gamma \subseteq \mathbb{R}^n \to \mathbb{R}$ we define its convex envelope on Γ, denoted* $\operatorname{convenv}_\Gamma(f)$*, to be the largest convex function g with $g \leq f$. In symbols,*

$$g = \sup\{h : \Gamma \to \mathbb{R} \mid h \text{ convex on } \Gamma \text{ and } h \leq f\}.$$

The convex envelope acts as the best possible convex lower bound for a given function. Further, its precise form is well-known for certain classes of functions; one such function of principle interest here is described in the following theorem. This is in some contexts referred to as a McCormick hull and is widely used throughout the nonconvex optimization literature [64, 148].

Theorem 3 ([2])**.** *If $f : \Gamma = [0,1] \times [\ell, u] \to \mathbb{R}$ is defined by $f(x,y) = -xy$, then the convex envelope of f on Γ is precisely*

$$\operatorname{convenv}_\Gamma(f)(x,y) = \max\{-ux, \ell - \ell x - y\}.$$

Further, $|f(x,y) - \operatorname{convenv}_\Gamma(f)(x,y)| \leq (u-\ell)/4$. In particular, if $|u - \ell| \to 0$, then $\operatorname{convenv}_\Gamma(f) \to f$.

Using this result we proceed to describe our approach for computing lower bounds to (CFA_1) as in (2.36). First observe that if $\Phi_i \in [\ell_i, u_i]$ and $W_{ii} \in [0,1]$, then

$$\begin{aligned}\operatorname{convenv}(-W_{ii}\Phi_i)(W_{ii}, \Phi_i) &= \max\{-u_i W_{ii}, \ell_i - \ell_i W_{ii} - \Phi_i\} \\ &= -\min\{u_i W_{ii}, \Phi_i + \ell_i W_{ii} - \ell_i\}.\end{aligned}$$

Hence, the best possible convex lower bound to the objective in Problem (2.36),

namely $\langle \mathbf{W}, \mathbf{\Sigma} - \mathbf{\Phi} \rangle = \langle \mathbf{W}, \mathbf{\Sigma} \rangle - \sum_i W_{ii}\Phi_i$, for $\boldsymbol{\ell} \leq \operatorname{diag}(\mathbf{\Phi}) \leq \mathbf{u}$ and $\mathbf{I} \succcurlyeq \mathbf{W} \succcurlyeq \mathbf{0}$ is

$$\langle \mathbf{W}, \mathbf{\Sigma} \rangle - \sum_{i=1}^{p} \min\{u_i W_{ii}, \Phi_i + \ell_i W_{ii} - \ell_i\}.$$

By introducing auxiliary variables $\mathbf{e} \in \mathbb{R}^p$ to represent these convex envelopes, we obtain the following linear SDO that is a lower bound to (2.36):

$$\begin{array}{lll} \min\limits_{\mathbf{W},\mathbf{\Phi},\mathbf{e}} & \langle \mathbf{W}, \mathbf{\Sigma} \rangle - \sum\limits_{i=1}^{p} e_i & \\ \text{s.t.} & \mathbf{\Phi} \in \mathcal{F}_{\mathbf{\Sigma}} & \\ & \mathbf{W} \in \mathcal{W}_{p-r} & \\ & \left.\begin{array}{r} \ell_i \leq \Phi_i \leq u_i \\ e_i \leq \Phi_i + \ell_i W_{ii} - \ell_i \\ e_i \leq u_i W_{ii} \end{array}\right\} & \forall\, i \end{array} \qquad (\mathrm{LS}_{\boldsymbol{\ell},\mathbf{u}})$$

The approach now lies in iteratively refining the lower and upper bounds on the diagonal entries of $\mathbf{\Phi}$, denoted $\boldsymbol{\ell}$ and $\mathbf{u}$, respectively, in order to improve the quality of the approximations obtained via convex envelopes (*cf.* Theorem 3). This classical scheme is known as *spatial branch and bound* and is shown in pseudocode in Algorithm 2 as it applies to solving (2.36) by way of using ($\mathrm{LS}_{\boldsymbol{\ell},\mathbf{u}}$).

In words, Algorithm 2 involves treating a given "node" $\mathtt{n} = [\boldsymbol{\ell}, \mathbf{u}]$, which represents bounds on $\mathbf{\Phi}$, namely, $\boldsymbol{\ell} \leq \operatorname{diag}(\mathbf{\Phi}) \leq \mathbf{u}$. Here we solve ($\mathrm{LS}_{\boldsymbol{\ell},\mathbf{u}}$) with the lower and upper bounds $\boldsymbol{\ell}$ and $\mathbf{u}$, respectively, and see whether the resulting new feasible solution is better (lower in objective value) than the best known incumbent solution encountered thus far. We then see if the the bound for this node as obtained via ($\mathrm{LS}_{\boldsymbol{\ell},\mathbf{u}}$) is better than the currently known best feasible solution; if it is not at least the current best feasible solution's objective value (up to some numerical tolerance), then we must further branch on this node, generating two new nodes $\mathtt{n}_1$ and $\mathtt{n}_2$ which partition the existing node $\mathtt{n}$. Throughout, we keep track of the worst lower bound encountered, thereby allowing the algorithm to be terminated early while still having a provable suboptimality guarantee on the best feasible solution $\mathbf{\Phi}_f \in \mathcal{F}_{\mathbf{\Sigma}}$ found thus

far.

In light of Theorem 3, we have as a corollary the following theorem.

Theorem 4. *Given numerical tolerance* `TOL` > 0*, Algorithm 2 (with an appropriate branching rule)*[12] *terminates in finitely many iterations and solves* (CFA_1) *to within an additive optimality gap of at most* `TOL`*. Further, if Algorithm 2 is terminated early (i.e., before* `Nodes` $= \emptyset$*), then the best feasible solution* $\mathbf{\Phi}_f$ *at termination is guaranteed to be within an additive optimality gap of* $z_f - z_{lb}$*.*

The algorithm we have considered here omits some important details. After discussing properties of ($\text{LS}_{\boldsymbol{\ell},\mathbf{u}}$) in Section 2.4.2, we will discuss various aspects of Algorithm 2. In Section 2.4.3, we detail how to choose input $\mathbf{u}^0$. We then turn our attention to branching (line 5) in Section 2.4.4. In Section 2.4.5, we use results from matrix analysis coupled with ideas from the modern practice of discrete optimization to make tailored refinements to Algorithm 2. Finally, in Section 2.4.6 we include a discussion of node selection strategies.

Global optimization—State of the art: We close this section by discussing similarities between the branch-and-bound approach we develop here and existing methods in nonconvex optimization. Our approach is very similar in spirit to approaches to global optimization [64], and in particular for (nonconvex) quadratic optimization problems, quadratically-constrained convex optimization problems, and bilinear optimization problems [75, 13, 147, 146, 51, 8, 113]. The primary similarity is that we work within a branch and bound framework using successively better convex lower bounds. However, while global optimization software for a variety of nonconvex problems with underlying vector variables is generally well-developed (as evidenced by solvers like `BARON`, see [134]), this is not the case for problems with underlying matrix variables and semidefinite constraints.

The presence of semidefinite structure presents several substantial computational challenges. First and foremost, algorithmic implementations for solving linear SDOs

[12] A branching rule that is sufficient for convergence is selecting $i \in \arg\max_i(u_i^c - \ell_i^c)$ and $\alpha = (u_i^c + \ell_i^c)/2$.

Algorithm 2 Spatial branch and bound scheme to solve (2.36). The inputs are as follows: (a) upper bounds $\mathbf{u}^0$ such that for any i and any $\mathbf{\Phi} \in \mathcal{F}$, we have $\Phi_i \leq u_i^0$ (see Section 2.4.3); (b) optimality tolerance `TOL`; and (c) initial feasible solution $\mathbf{\Phi}_f \in \mathcal{F}$.

1. Initialize $z_f \leftarrow \sum_{i>r} \lambda_i(\mathbf{\Sigma} - \mathbf{\Phi}_f)$; `Nodes` $\leftarrow \{([\mathbf{0}, \mathbf{u}^0], -\infty)\}$; and $z_{\text{lb}} \leftarrow -\infty$.
2. While `Nodes` $\neq \emptyset$, remove some node $([\boldsymbol{\ell}^c, \mathbf{u}^c], z^c) \in$ `Nodes`.
3. Solve ($\text{LS}_{\boldsymbol{\ell}^c, \mathbf{u}^c}$). Let $\mathbf{\Phi}$ be an optimal solution with z the optimal objective value; set $z_u \leftarrow \sum_{i>r} \lambda_i(\mathbf{\Sigma} - \mathbf{\Phi})$.
4. If $z_u < z_f$ (i.e., a better feasible solution is found), update the best feasible solution found thus far ($\mathbf{\Phi}_f$) to be $\mathbf{\Phi}$ and update the corresponding value (z_f) to z_u.
5. If $z < z_f -$ `TOL` (i.e., a `TOL`–optimal solution has not yet been found), then pick some $i \in \{1, \ldots, p\}$ and some $\alpha \in (\ell_i^c, u_i^c)$. Then add two new nodes to `Nodes`:
$$\left(\prod_{j<i}[\ell_j^c, u_j^c] \times [\ell_i^c, \alpha] \times \prod_{j>i}[\ell_j^c, u_j^c], z\right) \text{ and } \left(\prod_{j<i}[\ell_j^c, u_j^c] \times [\alpha, u_i^c] \times \prod_{j>i}[\ell_j^c, u_j^c], z\right).$$
6. Update the best lower bound $z_{\text{lb}} \leftarrow \min_{([\boldsymbol{\ell}, \mathbf{u}], z) \in \texttt{Nodes}} z$ and return to Step 2.

are not nearly as advanced as those which exist for linear optimization problems. Therefore, each subproblem, which is itself a linear SDO, carries a larger computational cost than the usual corresponding linear program which typically arises in other global optimization problems with vector variables. Secondly, a critical component of the success of global optimization software is the ability to quickly resolve multiple instances of subproblems which have similar structure. Corresponding methods for SDOs, as solved via interior point methods, are generally not well-developed. Finally, semidefinite structure complicates the traditional process of computing convex envelopes. Such computations are critical to the success of modern global optimization solvers like `BARON`.

There are a variety of other possible approaches to computing lower bounds to (CFA_q). One possibility is to utilize techniques for mixed semidefinite optimization, an approach that we detail in Appendix A.3. Another such approach is the method of moments [96]. However, for problems of the size we are considering, such an

approach is likely not computationally feasible, so we do not make a direct comparison here. There is also recent work in complementarity constraints literature [11] which connects rank-constrained optimization problems to copositive optimization [43]. In short, such an approach turns (2.35) into an equivalent convex problem; despite the transformation, the new problem is not particularly amenable to computation at this time. For this reason, we do not consider the copositive optimization approach.

2.4.2 Properties of ($\mathrm{LS}_{\boldsymbol{\ell},\mathbf{u}}$)

We now examine properties of ($\mathrm{LS}_{\boldsymbol{\ell},\mathbf{u}}$), the main subproblem of interest. Observe that it is a linear SDO problem, and therefore we can consider its dual, namely

$$\begin{array}{ll} \max\limits_{\substack{q,\mathbf{f}_u,\mathbf{f}_\ell,\boldsymbol{\kappa},\\ \boldsymbol{\sigma},\mathbf{m},\mathbf{n},\mathbf{p}}} & (p-r)q - \mathbf{u}'\mathbf{f}_u - \mathrm{Tr}(\mathbf{n}) - \langle \mathbf{p}, \boldsymbol{\Sigma} - \mathrm{diag}(\boldsymbol{\ell})\rangle \\ \text{s. t.} & \boldsymbol{\kappa} + \boldsymbol{\sigma} = \mathbf{1} \\ & \mathrm{diag}(\mathbf{p}) + \mathbf{f}_u - \mathbf{f}_\ell - \boldsymbol{\kappa} = \mathbf{0} \\ & \boldsymbol{\Sigma} - \mathrm{diag}(\mathbf{u}) + \mathbf{m} + \mathbf{n} + \mathrm{diag}(\mathrm{diag}(\mathbf{u}-\boldsymbol{\ell})\boldsymbol{\kappa}) - q\mathbf{I} = \mathbf{0} \\ & \mathbf{f}_u, \mathbf{f}_\ell, \boldsymbol{\kappa}, \boldsymbol{\sigma} \geq \mathbf{0} \\ & \mathbf{m}, \mathbf{n}, \mathbf{p} \succcurlyeq \mathbf{0}. \end{array} \qquad (\mathrm{DS}_{\boldsymbol{\ell},\mathbf{u}})$$

Observation 1. *We now include some remarks about structural properties of* ($\mathrm{LS}_{\boldsymbol{\ell},\mathbf{u}}$) *and its dual* ($\mathrm{DS}_{\boldsymbol{\ell},\mathbf{u}}$).

1. *If* $\mathrm{rank}(\boldsymbol{\Sigma}) = p$ *then the Slater condition [38] holds and hence there is strong duality, so we can work with* ($\mathrm{DS}_{\boldsymbol{\ell},\mathbf{u}}$) *instead of* ($\mathrm{LS}_{\boldsymbol{\ell},\mathbf{u}}$) *as an exact reformulation.*

2. *There exists an optimal solution to the dual with* $\mathbf{f}_u = \mathbf{0}$. *This is a variable reduction which is not immediately obvious. Note that* $\mathbf{f}_u$ *appears as the multiplier for the constraints in the primal of the form* $\mathrm{diag}(\boldsymbol{\Phi}) \leq \mathbf{u}$. *To claim that we can set* $\mathbf{f}_u = \mathbf{0}$ *it suffices to show that the corresponding constraints in the primal can be ignored. Namely, if* $(\mathbf{W}^*, \boldsymbol{\Phi}^*)$ *solves* ($\mathrm{LS}_{\boldsymbol{\ell},\mathbf{u}}$) *with the constraints* $\Phi_i \leq u_i \ \forall i$ *omitted, then the pair* $(\mathbf{W}^*, \widetilde{\boldsymbol{\Phi}})$ *is feasible and optimal to* ($\mathrm{LS}_{\boldsymbol{\ell},\mathbf{u}}$)

with all the constraints included, where $\widetilde{\mathbf{\Phi}}$ *is defined by*

$$\widetilde{\Phi}_i = \min\{\Phi^*_i, u_i\}.$$

Hereinafter we set $\mathbf{f}_u = \mathbf{0}$ *and omit the constraint* $\operatorname{diag}(\mathbf{\Phi}) \leq \mathbf{u}$ *(with the caveat that, upon solving a problem and identifying some* $\mathbf{\Phi}^*$*, we must instead work with* $\min\{\mathbf{\Phi}^*, \operatorname{diag}(\mathbf{u})\}$*, taken entrywise).*

Solving subproblems: We briefly detail how to solve ($\text{LS}_{\boldsymbol{\ell},\mathbf{u}}$). In light of the discussion in Section 2.3.2, we choose to apply a first-order method. Observe that we cannot solve the primal form ($\text{LS}_{\boldsymbol{\ell},\mathbf{u}}$) within the branch-and-bound framework unless we solve it to optimality. Therefore, we instead choose to work with its dual ($\text{DS}_{\boldsymbol{\ell},\mathbf{u}}$). We apply an off-the-shelf solver `SCS` [119] to solve ($\text{DS}_{\boldsymbol{\ell},\mathbf{u}}$) and find reasonably accurate, *feasible* solutions for this dual problem, which guarantees that we have a lower bound to ($\text{LS}_{\boldsymbol{\ell},\mathbf{u}}$).[13] In this way, we maintain the provable optimality properties of Algorithm 2 without needing to solve nodes in the branch-and-bound tree to full optimality.

2.4.3 Input parameters

In solving the root node of Algorithm 2, we must begin with some choice of $\mathbf{u}^0 = \mathbf{u}$. An obvious first choice for u_i is $u_i = \Sigma_{ii}$, but one can do better. Let us optimally set u_i, defining it as

$$u_i := \begin{array}{ll} \max\limits_{\eta \in \mathbb{R}} & \eta \\ \text{s.t.} & \mathbf{\Sigma} - \eta \mathbf{E}^{(i)} \succcurlyeq \mathbf{0}, \end{array} \tag{2.37}$$

where $\mathbf{E}^{(i)} \in \mathbb{R}^{p \times p}$ is a matrix with all zeros except $\mathbf{E}^{(i)}_{ii} = 1$. These bounds are useful because if $\mathbf{\Phi} \in \mathcal{F}_{\mathbf{\Sigma}}$, then $\Phi_i \leq u_i$. Note that problem (2.37) is a linear SDO for which

[13] One notable feature of the ADMM-based approach is that we can extract an approximately feasible primal solution $\mathbf{\Phi}$, which is useful for branching. Note that in Algorithm 2, we can replace the best incumbent solution if we find a new $\mathbf{\Phi}$ which has better objective value $\sum_{i>r} \sigma_i(\mathbf{\Sigma} - \mathbf{\Phi})$. Because $\mathbf{\Phi}$ may not be feasible (i.e., $\mathbf{\Phi} \notin \mathcal{F}$), we take care here. Namely, compute $t = \sum_{i>r} \lambda_i(\mathbf{\Sigma} - \mathbf{\Phi})$, where $\lambda_1(\mathbf{\Sigma} - \mathbf{\Phi}) \geq \cdots \geq \lambda_p(\mathbf{\Sigma} - \mathbf{\Phi})$ are the sorted eigenvalues of $\mathbf{\Sigma} - \mathbf{\Phi}$. If $t < z_f$, then we perform an iteration of CG scheme for finding feasible solutions (outlined in Section 2.3) to find a feasible $\bar{\mathbf{\Phi}} \in \mathcal{F}$. We then use this as a possible candidate for replacing the incumbent.

strong duality holds. Its dual is precisely

$$u_i = \begin{array}{ll} \min\limits_{\mathbf{M}\in\mathbb{R}^{p\times p}} & \langle \mathbf{M}, \mathbf{\Sigma}\rangle \\ \text{s. t.} & M_{ii} = 1 \\ & \mathbf{M} \succcurlyeq \mathbf{0}, \end{array} \tag{2.38}$$

a linear SDO in standard form with a single equality constraint. By a result of [15, 122], there exists a rank one solution to (2.38). This implies that (2.38) can actually be solved as a convex quadratic program:

$$u_i = \begin{array}{ll} \min\limits_{\mathbf{m}\in\mathbb{R}^p} & \mathbf{m}'\mathbf{\Sigma}\mathbf{m} \\ \text{s. t.} & m_i^2 = 1. \end{array} = \begin{array}{ll} \min\limits_{\mathbf{m}} & \mathbf{m}'\mathbf{\Sigma}\mathbf{m} \\ \text{s. t.} & m_i = 1. \end{array} \tag{2.39}$$

This formulation given in (2.39) is computationally inexpensive to solve (given a large number of specialized convex quadratic problem solvers), in contrast to both formulations (2.37) and (2.38).

Exact formula when $\mathbf{\Sigma} \succ \mathbf{0}$: In the case when $\mathbf{\Sigma} \succ \mathbf{0}$, it is not necessary to use quadratic optimization problems to compute $\mathbf{u}$. In this case one can apply a straightforward Schur complement argument [38] to show that $\mathbf{u}$ can be computed by solving for inverse of p different $(p-1)\times(p-1)$ matrices (or equivalently, by finding the diagonal of the precision matrix $\mathbf{\Sigma}^{-1}$). In particular,

$$u_i = 1/\left(\mathbf{\Sigma}^{-1}\right)_{ii} = 1/\langle \mathbf{\Sigma}^{-1}, \mathbf{E}^{(i)}\rangle,$$

where $\left(\mathbf{\Sigma}^{-1}\right)_{ii}$ denotes the ith diagonal entry of $\mathbf{\Sigma}^{-1}$.

Alternative approach using eigenvalue decomposition

Using techniques from matrix analysis, it is actually unnecessary to solve quadratic optimization problems of the form (2.39) to compute $\mathbf{u}^0$, even when $\mathbf{\Sigma}$ is rank-deficient (i.e., $\operatorname{rank}(\mathbf{\Sigma}) < p$). In particular, the following proposition entirely eliminates the need for solving p convex quadratic optimization problems, instead relying solely on

a standard eigenvalue decomposition.

Proposition 1. *Given eigenvalue decomposition* $\mathbf{\Sigma} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}'$ *for* $\mathbf{\Sigma} \succcurlyeq \mathbf{0}$ *(where* $\mathbf{U}\mathbf{U}' = \mathbf{U}'\mathbf{U} = \mathbf{I}$ *and* $\mathbf{\Lambda}$ *is diagonal with* $\Lambda_{11} \geq \Lambda_{22} \geq \cdots \geq \Lambda_{pp}$*) and* $\rho := \operatorname{rank}(\mathbf{\Sigma})$*, one has*

$$\begin{array}{ll} \max\limits_{\eta \in \mathbb{R}} & \eta \\ \text{s. t.} & \mathbf{\Sigma} - \eta \mathbf{E}^{(i)} \succcurlyeq \mathbf{0} \end{array} = \begin{cases} 0, & \sum_{j>\rho} U_{ij}^2 > 0 \\ 1/\left(\sum_{j=1}^{\rho} U_{ij}^2/\Lambda_{jj}\right), & \sum_{j>\rho} U_{ij}^2 = 0. \end{cases}$$

Proof. Fix i throughout. Our principal focus is on the minimum eigenvalue of $\mathbf{\Sigma} - \eta\mathbf{E}^{(i)}$ for $\eta \geq 0$. Let $N = p - \rho$ denote the nullity of $\mathbf{\Sigma}$, i.e., the dimension of the nullspace of $\mathbf{\Sigma}$. We assume throughout that $N > 0$. (The case when $N = 0$ follows in a similar manner, so long as we interpret the sum $\sum_{j \in \emptyset} U_{ij}^2 = 0$.) For simplicity, let $\lambda_j := \Lambda_{jj}\ \forall j$. By definition, $\lambda_\rho > 0$ and $\lambda_j = 0 \quad \forall\ j > \rho$.

Fix $\eta > 0$. Studying the eigenvalues of $\mathbf{\Sigma} - \eta\mathbf{E}^{(i)}$ is equivalent to studying the eigenvalues of $\mathbf{\Lambda} - \eta\mathbf{v}\mathbf{v}'$, where $\mathbf{v}$ is the ith row of $\mathbf{U}$ (eigenvalues are invariant under unitary conjugacy). Consider the characteristic polynomial π of $\mathbf{\Lambda} - \eta\mathbf{v}\mathbf{v}'$ in variable λ. As per [68, §5], this can be written as

$$\pi(\lambda) = \prod_j (\lambda_j - \lambda) - \eta \sum_j v_j^2 \prod_{k \neq j} (\lambda_k - \lambda).$$

Therefore, if for $\epsilon \geq 0$ we define $\underline{\lambda}_\epsilon = \prod_{j=1}^{\rho}(\lambda_j + \epsilon)$, then

$$\begin{aligned} \pi(-\epsilon) &= \prod_j (\lambda_j + \epsilon) - \eta \sum_j v_j^2 \prod_{k \neq j} (\lambda_k + \epsilon) \\ &= \underline{\lambda}_\epsilon \epsilon^{N-1} \left(\left(1 - \eta \sum_{j=1}^{\rho} v_j^2/(\lambda_j + \epsilon) \right) \epsilon - \eta \sum_{j>\rho} v_j^2 \right). \end{aligned}$$

Note that $\underline{\lambda}_\epsilon > 0$ whenever $\epsilon \geq 0$.

We will consider two scenarios. First suppose that $\sum_{j>\rho} v_j^2 > 0$. Then for $\epsilon > 0$ sufficiently small, $\pi(-\epsilon) < 0$. Combined with the fact that $\lim_{\lambda \to -\infty} \pi(\lambda) = \infty$ and the intermediate value theorem, this implies that π has a strictly negative root. As the choice of $\eta > 0$ was arbitrary, this implies that we must have that $\eta = 0$ to ensure

that $\mathbf{\Sigma} - \eta \mathbf{E}^{(i)} \succcurlyeq \mathbf{0}$.

Now let us consider the case when $\sum_{j>\rho} v_j^2 = 0$. We focus on two possibilities:

(a) Suppose that $1/\eta < \sum_{j=1}^{\rho} v_j^2/\lambda_j$. In this case, just like before we have $\pi(-\epsilon) < 0$ for $\epsilon > 0$ sufficiently small; another application of the intermediate value theorem shows that π has a strictly negative root. It follows that $\mathbf{\Sigma} - \eta \mathbf{E}^{(i)} \not\succcurlyeq \mathbf{0}$.

(b) Now suppose that $1/\eta > \sum_{j=1}^{\rho} v_j^2/\lambda_i$. (Note that it is impossible for the right-hand side to equal zero, based on the fact that $\sum_{j>\rho} v_j^2 = 0$ and that $\mathbf{v} \neq \mathbf{0}$.) This implies that $\pi(-\epsilon) > 0$ for $\epsilon > 0$ sufficiently small. We claim that π does not have a strictly negative root. For contradiction, suppose that π does have a strictly negative root, say λ^*. By the interlacing inequality [80, §4.3], we know that this root must have multiplicity one (i.e., it cannot have larger multiplicity). Yet, $\pi(-\epsilon) > 0$ for $\epsilon > 0$ small, $\pi(\lambda^*) = 0$, and $\lim_{\lambda \to -\infty} \pi(\lambda) = \infty$. Clearly this implies, via continuity of π', that $\pi'(\lambda^*) = 0$.[14] Yet this necessarily implies that λ^* has multiplicity strictly greater than one, yielding a contradiction. Therefore, we conclude that in this case π does not have a strictly negative root.

Combining the above scenarios with the fact that $v_j = U_{ij}$, we conclude that the largest $\eta \geq 0$ so that $\mathbf{\Sigma} - \eta \mathbf{E}^{(i)} \succcurlyeq \mathbf{0}$ is precisely as claimed:

$$\max_{\substack{\eta: \\ \mathbf{\Sigma} - \eta \mathbf{E}^{(i)} \succcurlyeq \mathbf{0}}} \eta \quad = \quad \begin{cases} 0, & \sum_{j>\rho} U_{ij}^2 > 0 \\ 1/\left(\sum_{j=1}^{\rho} U_{ij}^2/\Lambda_{jj} \right), & \sum_{j>\rho} U_{ij}^2 = 0. \end{cases}$$

□

2.4.4 Branching

Here we detail two methods for branching (line 5 in Algorithm 2). The problem of branching is as follows: having solved ($\mathrm{LS}_{\boldsymbol{\ell},\mathbf{u}}$) for some particular $\mathtt{n} = [\boldsymbol{\ell}, \mathbf{u}]$, we must choose some $i \in \{1, \ldots, p\}$ and split the interval $[\ell_i, u_i]$ to create two new subproblems. We begin with a simple branching rule. Given a solution $(\mathbf{W}^*, \mathbf{\Phi}^*, \mathbf{e}^*)$

[14] This observation implicitly uses the fact that, again by the interlacing inequality, π can have at most one strictly negative root.

to ($\text{LS}_{\boldsymbol{\ell},\mathbf{u}}$), compute $i \in \operatorname{argmax}_i |e_i^* - W_{ii}^* \Phi_i^*|$ and branch on variable Φ_i, generating two new subproblems with the intervals

$$\prod_{j<i}[\ell_j, u_j] \times [\ell_i, \Phi_i^*] \times \prod_{j>i}[\ell_j, u_j] \quad \text{and} \quad \prod_{j<i}[\ell_j, u_j] \times [\Phi_i^*, u_i] \times \prod_{j>i}[\ell_j, u_j].$$

Observe that, so long as $\max_i |e_i^* - W_{ii}^* \Phi_i^*| > 0$, the solution $(\mathbf{W}^*, \mathbf{\Phi}^*, \mathbf{e}^*)$ is not optimal for either of the subproblems created.

We now briefly describe an alternative rule which we employ instead. We again pick the branching index i as before, but now the two new nodes we generate are

$$\prod_{j<i}[\ell_j, u_j] \times [\ell_i, (1-\epsilon)\Phi_i^* + \epsilon\ell_i] \times \prod_{j>i}[\ell_j, u_j] \quad \text{and}$$
$$\prod_{j<i}[\ell_j, u_j] \times [(1-\epsilon)\Phi_i^* + \epsilon\ell_i, u_i] \times \prod_{j>i}[\ell_j, u_j],$$

where $\epsilon \in [0, 1)$ is some parameter. For the computational experiments, we set $\epsilon = 0.4$.

Such an approach, which lowers the location of the branch in the ith interval $[\ell_i, u_i]$ from Φ_i^*, serves to improve the objective value from the first node, while hurting the objective value from the second node (here by objective value, we mean the objective value of the optimal solution to the two new subproblems). In this way, it spreads out the distance between the two, and so it is more likely that the first node may have an objective value that is higher than $z_f - \texttt{TOL}$ than before, and hence, this would mean there are fewer nodes necessary to consider to solve for an additive gap of TOL. While this heuristic explanation is only partially satisfying, we have observed throughout a variety of numerical experiments that this rule, even though simple, performs better across a variety of example classes than the basic branching rule outlined. At the same time, recent work on the theory of branching rules supports such a heuristic rule [98]. In Section 2.5.4, we give evidence on the impact of the use of the modified branching rule.

2.4.5 Weyl's Method—Pruning and bound tightening

In this subsection, we develop another method for lower bounds for the factor analysis problem. While we use it to supplement our approach detailed throughout Section 2.4, it is of interest as a standalone method, particularly for its computational speed and simplicity. In Section 2.5.3, we discuss the performance of this approach in both contexts.

The central problem of interest in factor analysis involves the spectrum of a symmetric matrix ($\mathbf{\Sigma} - \mathbf{\Phi}$) which is the difference of two other symmetric matrices ($\mathbf{\Sigma}$ and $\mathbf{\Phi}$). From a linear algebraic perspective, the spectrum of the sum of real symmetric matrices is an extensively studied problem [80, 32].Therefore, it is natural to inquire how such results carry over to our setting. We discuss the implications of a well-known result from this literature, namely *Weyl's inequality* (see, e.g., [32]):[15]

Theorem 5 (Weyl's Inequality)**.** *For symmetric matrices* $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{p\times p}$ *with sorted eigenvalues*

$$\lambda_1(\mathbf{A}) \geq \lambda_2(\mathbf{A}) \geq \cdots \geq \lambda_p(\mathbf{A}) \quad \textit{and} \quad \lambda_1(\mathbf{B}) \geq \lambda_2(\mathbf{B}) \geq \cdots \geq \lambda_p(\mathbf{B})$$

one has for any $k \in \{1, \ldots, p\}$ *that*

$$\lambda_k(\mathbf{A} + \mathbf{B}) \geq \lambda_{k+j}(\mathbf{A}) + \lambda_{p-j}(\mathbf{B}) \; \forall j = 0, \ldots, p-k. \tag{2.40}$$

For any vector $\mathbf{x} \in \mathbb{R}^p$ we let $\{x_{(i)}\}_{i=1}^p$ denote sorted $\{x_i\}_{i=1}^p$ with $x_{(1)} \geq x_{(2)} \geq \cdots \geq x_{(p)}$. Using this notation, we arrive at the following theorem.

Theorem 6. *For any diagonal matrix* $\bar{\mathbf{\Phi}}$ *one has that*

$$\min_{\mathbf{\Phi}\in\mathcal{F}} \sum_{i>r} \lambda_i(\mathbf{\Sigma} - \mathbf{\Phi}) \geq \min_{\mathbf{\Phi}\in\mathcal{F}} \left(\sum_{i>r} \max_{j=0,\ldots,p-i} \left\{ \lambda_{i+j}(\mathbf{\Sigma} - \bar{\mathbf{\Phi}}) + \operatorname{diag}\left(\bar{\mathbf{\Phi}} - \mathbf{\Phi}\right)_{(p-j)}, 0 \right\} \right).$$

[15] There are numerous eigenvalue inequalities. To show the variety of techniques and their implications for factor analysis, we include in Appendix A.3 a full discussion of how to incorporate them with mixed integer semidefinite optimization techniques.

Proof. Apply Weyl's inequality with $\mathbf{A} = \mathbf{\Sigma} - \operatorname{diag}(\bar{\mathbf{\Phi}})$ and $\mathbf{B} = \bar{\mathbf{\Phi}} - \mathbf{\Phi}$, and use the fact that $\mathbf{\Phi} \in \mathcal{F}$ so $\mathbf{\Sigma} - \mathbf{\Phi} \succcurlyeq \mathbf{0}$. □

Weyl's Method: The new lower bound introduced in Theorem 6 is a nonconvex problem in general. We begin by discussing one situation in which Theorem 6 provides computationally tractable (and fast) lower bounds; we deem this *Weyl's method*, detailed as follows:

1. Compute bounds u_i as in (2.37), so that for all $\mathbf{\Phi} \in \mathcal{F}_{\mathbf{\Sigma}}$, one has $\Phi_i \leq u_i \; \forall i$.

2. For each $r \in \{1, \ldots, p\}$, one can compute a lower bound to (CFA_1) (for a given r) of
$$\sum_{i>r} \max\{\lambda_i(\mathbf{\Sigma} - \operatorname{diag}(\mathbf{u})), 0\}, \tag{2.41}$$
by taking Theorem 6 with $\bar{\mathbf{\Phi}} = \operatorname{diag}(\mathbf{u})$.

As per the above remarks, computing $\mathbf{u}$ in Step 1 of Weyl's method can be carried out efficiently. Step 2 only relies on computing the eigenvalues of $\mathbf{\Sigma} - \operatorname{diag}(\mathbf{u})$. Therefore, this lower bounding procedure is quite simple to carry out. What is perhaps surprising is that this simple lower bounding procedure is effective as a standalone method. We describe such results in Section 2.5.3. We now turn our attention to how Weyl's method can be utilized within the branch and bound tree as described in Algorithm 2.

Pruning: We begin by considering how Weyl's method can be used for *pruning*. The notion of pruning for branch and bound trees is grounded in the theory and practice of discrete optimization. In short, pruning in the elimination of nodes from the tree without actually solving them. We make this precise in our context.

Consider some point in the branch and bound process in Algorithm 2, where we have some collection of nodes, $([\boldsymbol{\ell}^c, \mathbf{u}^c], z^c) \in \texttt{Nodes}$. Recall that z^c is the optimal objective value of the parent node of $\texttt{n}$. Per Weyl's method, we know *a priori*, without

solving ($\text{LS}_{\boldsymbol{\ell}^c,\mathbf{u}^c}$), that

$$\min_{\substack{\boldsymbol{\Phi}\in\mathcal{F}_{\boldsymbol{\Sigma}}\\ \boldsymbol{\ell}^c\le\operatorname{diag}(\boldsymbol{\Phi})\le\mathbf{u}^c}} \sum_{i>r}\lambda_i(\boldsymbol{\Sigma}-\boldsymbol{\Phi}) \ge \sum_{i>r}\max\{\lambda_i(\boldsymbol{\Sigma}-\operatorname{diag}(\mathbf{u}^c)),0\}.$$

Hence, if $z_f - \texttt{TOL} < \sum_{i>r}\max\{\lambda_i(\boldsymbol{\Sigma}-\operatorname{diag}(\mathbf{u}^c)),0\}$, where z_f is as in Algorithm 2, then node $\mathtt{n}$ can be discarded, i.e., there is no need to actually compute ($\text{LS}_{\boldsymbol{\ell}^c,\mathbf{u}^c}$) or further consider this branch. This is because if we were to solve ($\text{LS}_{\boldsymbol{\ell}^c,\mathbf{u}^c}$), and then branch again, solving further down this branch to optimality, then the final lower bound obtained would necessarily be at least as large as the best feasible objective already found (within tolerance $\texttt{TOL}$).

In this way, because Weyl's method is relatively fast, this provides a simple method for pruning. In the computational results detailed in Section 2.5.3, we always use Weyl's method to discard nodes which are not fruitful to consider.

Bound tightening: We now turn our attention to another way in which Weyl's method can be used to improve the performance of Algorithm 2—*bound tightening*. In short, bound tightening is the use of implicit constraints to strengthen bounds obtained for a given node. We detail this with the same node notation as above. Namely, consider a given node $\mathtt{n} = [\boldsymbol{\ell}^c, \mathbf{u}^c]$. Fix some $j \in \{1,\dots,p\}$ and let $\alpha \in (\ell_j^c, u_j^c)$. If we have that

$$z_f - \texttt{TOL} < \sum_{i>r}\max\{\lambda_i(\boldsymbol{\Sigma}-\operatorname{diag}(\tilde{\mathbf{u}})),0\},$$

where $\tilde{\mathbf{u}}$ is $\mathbf{u}^c$ with the jth entry replaced by α, then we can replace the node $\mathtt{n}$ with the "tightened" node $\tilde{\mathtt{n}} = [\tilde{\boldsymbol{\ell}}, \mathbf{u}^c]$, where $\tilde{\boldsymbol{\ell}}$ is $\boldsymbol{\ell}^c$ with the jth entry replaced by α.

We consider why this is valid. Suppose that one were to solve ($\text{LS}_{\boldsymbol{\ell}^c,\mathbf{u}^c}$) and choose to branch on index j at α. Then one would create two new nodes: $[\boldsymbol{\ell}^c, \tilde{\mathbf{u}}]$ and $[\tilde{\boldsymbol{\ell}}, \mathbf{u}^c]$. We would necessarily then prune away the node $[\boldsymbol{\ell}^c, \tilde{\mathbf{u}}]$ as just described; hence, we can replace $[\boldsymbol{\ell}^c, \mathbf{u}^c]$ without loss of generality with $[\tilde{\boldsymbol{\ell}}, \mathbf{u}^c]$. Note that here $\alpha \in (\ell_j^c, u_j^c)$ and $j \in \{1,\dots,p\}$ were arbitrary. Hence, for each j, one can choose the largest such

$\alpha_j \in (\ell_j^c, u_j^c)$ (if one exists) so that $z_f - \texttt{TOL} < \sum_{i>k} \max\{\lambda_i(\boldsymbol{\Sigma} - \operatorname{diag}(\tilde{\mathbf{u}})), 0\}$, and then replace $\boldsymbol{\ell}^c$ by $\tilde{\boldsymbol{\ell}}$.[16]

Such a procedure is somewhat expensive (because of its use of repeated eigenvalue calculations), but can be thought of as "optimal" pruning via Weyl's method. In our experience the benefit of bound tightening does not warrant its computational cost when used at every node in the branch-and-bound tree except in a small number of problems. For this reason, in the computational results in Section 2.5.3 we only employ bound tightening at the root node $\mathtt{n} = [\mathbf{0}, \mathbf{u}^0]$.

2.4.6 Node selection

In this section, we briefly describe our method of node selection. The problem of node selection has been considered extensively in discrete optimization and is still an active area of research. Here we describe a simple node selection strategy.

To be precise, consider some point in Algorithm 2 were we have a certain collection of nodes, $([\boldsymbol{\ell}^c, \mathbf{u}^c], z^c) \in \texttt{Nodes}$. The most obvious node selection strategy is to pick the node $\mathtt{n}$ for which z^c is smallest among all nodes in $\texttt{Nodes}$. In this way, the algorithm is likely to improve the gap $z_f - z_{\text{lb}}$ at every iteration. Such greedy selection strategies tend to not perform particularly well in general global optimization problems (see, e.g., [148]).

For these reasons, we employ a slightly modified greedy selection strategy which utilizes Weyl's method. For a given node $\mathtt{n}$, we also consider its corresponding lower bound w^c obtained from Weyl's method, namely, $w^c := \sum_{i>r} \max\{\lambda_i(\boldsymbol{\Sigma} - \operatorname{diag}(\mathbf{u}^c)), 0\}$. For each node, we now consider $\max\{z^c, w^c\}$. There are two cases to consider:

1. With probability β, we select the node with smallest value of $\max\{z^c, w^c\}$.

2. In the remaining scenarios (occurring with probability $1 - \beta$), we choose randomly between selecting the node with smallest value of z^c and the node with

[16] An obvious choice to find such an α is a grid-search-based bisection method. For simplicity we use a linear search on a grid instead of resorting to the bisection method.

smallest value of w^c. To be precise, let Z be the minimum of z^c over all nodes and likewise let W be the minimum of w^c over all nodes. Then with (independent) probability β, we choose the node with worst z^c *or* w^c (i.e., with $\min\{z^c, w^c\} = \min\{Z, W\}$); with probability $1-\beta$, if $Z < W$ we choose a node with $w^c = W$, and if $Z > W$ we choose a node with $z^c = Z$.

In this way, we allow for the algorithm to switch between trying to make progress towards improving the convex envelope bounds and making progress towards improving the best of the two bounds (the convex envelope bounds along with the Weyl bounds). We set $\beta = 0.9$ for all computational experiments. It is possible that a more dynamic branching strategy could perform substantially better; however, the method here has a desirable level of simplicity.

We close by noting that while this node selection strategy appears naïve, it is not necessarily so simple. Improved node selection strategies from discrete optimization often take into account some sort of duality theory. Weyl's inequality is at its core a result from duality theory (principally Wielandt's minimax principle; see [32]), and therefore our strategy is not as unsophisticated as it might appear on first inspection.

2.5 Computational experiments

In this section, we perform various computational experiments to study the properties of our different algorithmic proposals for (CFA$_q$), for $q \in \{1, 2\}$. Using a variety of statistical measures, we compare our methods with existing popular approaches for FA, as implemented in standard `R` statistical packages `psych` [127], `nFactors` [124], and `GPArotation` [21]. We then turn our attention to certificates of optimality as described in Section 2.4 for (CFA$_1$).

2.5.1 Synthetic examples

For our synthetic experiments, we considered distinctly different groups of examples. Classes A_1 and A_2 have subspaces of the low-rank common factors which are random

and the values of Φ_i are taken to be equally spaced. The underlying matrix corresponding to the common factors in type A_1 is exactly low-rank, while this is not the case in type A_2.

Class $A_1(R/p)$. We generated a matrix $\mathbf{L} \in \mathbb{R}^{p \times R}$ (with $R < p$) with $L_{ij} \overset{\text{iid}}{\sim} N(0,1)$. The *unique variances* $\Phi_1, \ldots, \Phi_p$, are taken to be proportional to p equi-spaced values on the interval $[\lambda_R(\mathbf{L}'\mathbf{L}), \lambda_1(\mathbf{L}'\mathbf{L})]$ such that

$$\Phi_i = \bar{\phi} \cdot \left(\lambda_1(\mathbf{L}'\mathbf{L}) + (\lambda_R(\mathbf{L}'\mathbf{L}) - \lambda_1(\mathbf{L}'\mathbf{L})) \frac{i-1}{p} \right) \quad \text{for } 1 \leq i \leq p.$$

Here $\bar{\phi}$, which controls the ratio of the variances between the *uniquenesses* and the common latent factors, is chosen such that $\sum_{i=1}^{p} \Phi_i = \text{Tr}(\mathbf{L}\mathbf{L}')$, i.e., the contribution to the total variance from the common factors matches that from the uniqueness factors. The covariance matrix is thus given by: $\mathbf{\Sigma} = \mathbf{L}\mathbf{L}' + \mathbf{\Phi}$.

Class $A_2(p)$. Here $\mathbf{L} \in \mathbb{R}^{p \times p}$ is generated as $L_{ij} \overset{\text{iid}}{\sim} N(0,1)$. We did a full singular value decomposition on $\mathbf{L}$—let $\mathbf{U}_L$ denote the set of p (left) singular vectors. We created a positive definite matrix with exponentially decaying eigenvalues as follows: $\widetilde{\mathbf{L}}\widetilde{\mathbf{L}}' = \mathbf{U}_L \, \text{diag}(\lambda_1, \ldots, \lambda_p) \mathbf{U}_L'$, where the eigenvalues were chosen as $\lambda_i = 0.8^{i/2}, i = 1, \ldots, p$. We chose the diagonal entries of $\mathbf{\Phi}$ (like data type A_1), as a scalar multiple ($\bar{\phi}$) of a uniformly spaced grid in $[\lambda_p, \lambda_1]$ and $\bar{\phi}$ was chosen such that $\sum_i \Phi_i = \text{Tr}(\widetilde{\mathbf{L}}\widetilde{\mathbf{L}}')$.

In contrast, classes B_1, B_2, and B_3 are qualitatively different from the aforementioned ones—the subspaces corresponding to the common factors are more structured, and hence different from the coherence-like assumptions on the eigenvectors which are necessary for nuclear-norm-based methods [135] to work well.

Class $B_1(R/p)$. We set $\mathbf{\Theta} = \mathbf{L}\mathbf{L}'$, where $\mathbf{L} \in \mathbb{R}^{p \times R}$ is given by

$$L_{ij} = \begin{cases} 1, & i \leq j \\ 0, & i > j. \end{cases}$$

Class $B_2(r/R/p)$. Here we set $\boldsymbol{\Theta} = \mathbf{L}\mathbf{L}'$, where $\mathbf{L} \in \mathbb{R}^{p\times R}$ is such that

$$L_{ij} = \begin{cases} 1, & i,j = 1,\ldots,r \\ \overset{\text{iid}}{\sim} N(0,1), & i > r, j = 1,\ldots,R \\ 0, & i = 1,\ldots,r,\ j > r. \end{cases}$$

Class $B_3(r/R/p)$. Here we define $\boldsymbol{\Theta} = \mathbf{L}\mathbf{L}'$, where $\mathbf{L} \in \mathbb{R}^{p\times R}$ is such that

$$L_{ij} = \begin{cases} 1, & j = 1,\ldots,r,\ i \le j \\ \overset{\text{iid}}{\sim} N(0,1), & j > r,\ i = 1,\ldots,R \\ 0, & i > j,\ j = 1,\ldots,r. \end{cases}$$

In all the B classes, we generated $\Phi_i \overset{\text{iid}}{\sim} \text{abs}(N(0,1))$ and the covariance matrix $\boldsymbol{\Sigma}$ was taken to be $\boldsymbol{\Sigma} = \boldsymbol{\Theta} + \alpha\boldsymbol{\Phi}$, where α is so that $\text{Tr}(\boldsymbol{\Theta}) = \alpha\,\text{Tr}(\boldsymbol{\Phi})$.

Comparisons with other FA methods

We performed a suite of experiments using Algorithm 1 for the cases $q \in \{1,2\}$. We compared our proposed algorithm with the following popular FA estimation procedures as described in Section 2.1.1:

1. MINRES: minimum residual factor analysis
2. WLS: weighted least squares method with weights being the *uniquenesses*
3. PA: this is the principal axis factor analysis method
4. MTFA: constrained minimum trace factor analysis—formulation (2.6)
5. PC: The method of principal component factor analysis
6. MLE: this is the maximum likelihood estimator (MLE)
7. GLS: the generalized least squares method

For MINRES, WLS, GLS, and PA, we used the implementations available in the `R` package `psych` [127] available from `CRAN`. For MLE we investigated the methods `factanal` from `R` package `stats` and the `fa` function from `R` package `psych`. The

estimates obtained by the different MLE implementations were similar in our experiments; therefore, we report the results obtained from `factanal`.

For MTFA, we used our own implementation by adapting the ADMM algorithm (Section 2.3.2) to solve Problem (2.6). For the experiments in Section 2.5.1, we took the convergence thresholds for Algorithm 1 as $\texttt{TOL} = 10^{-5}$ and ADMM as $\texttt{TOL} \times \alpha = 10^{-9}$. For the PC method we followed the description in [10] (as described in Section 2.1.1)—the $\mathbf{\Phi}$ estimates were thresholded at zero if they became negative.

Note that all the methods considered in the experiments, apart from MTFA, allow the user to specify the desired number of factors in the problem. Since standard implementations of MINRES, WLS and PA require $\mathbf{\Sigma}$ to be a correlation matrix, we standardized all covariance matrices to correlation matrices at the outset.

Performance measures

We consider the following measures of "goodness of fit" (see [14] and references therein) to assess the performances of the different FA estimation procedures.

Estimation error in $\mathbf{\Phi}$: We use the following measure to assess the quality of an estimator for $\mathbf{\Phi}$:

$$\text{Error}(\mathbf{\Phi}) := \sum_{i=1}^{p} (\widehat{\Phi}_i - \Phi_i)^2. \tag{2.42}$$

The estimation of $\mathbf{\Phi}$ plays an important role in FA—given a good estimate $\widehat{\mathbf{\Phi}}$, the r-common factors can be obtained by a rank-r eigendecomposition on the residual covariance matrix $\mathbf{\Sigma} - \widehat{\mathbf{\Phi}}$.

Proportion of variance explained and semi-definiteness of $(\mathbf{\Sigma} - \mathbf{\Phi})$: A fundamental objective in FA lies in understanding how well the r-common factors explain the residual covariance, i.e., $(\mathbf{\Sigma} - \widehat{\mathbf{\Phi}})$—a direct analogue of what is done in PCA, as explained in Section 2.1. For a given r, the proportion of variance explained by the

r common factors is given by

$$\text{Explained Variance} = \sum_{i=1}^{r} \lambda_i(\widehat{\boldsymbol{\Theta}}) / \sum_{i=1}^{p} \lambda_i(\boldsymbol{\Sigma} - \widehat{\boldsymbol{\Phi}}). \tag{2.43}$$

As r increases, the explained variance increases to one. This trade-off between r and "Explained Variance" plays an important role in exploratory FA and in particular the choice of r. For the expression (2.43) to be meaningful, it is desirable to have $\boldsymbol{\Sigma} - \widehat{\boldsymbol{\Phi}} \succcurlyeq \mathbf{0}$. Note that our framework (CFA$_q$), and in particular MTFA, estimates $\boldsymbol{\Phi}$ under a PSD constraint on $\boldsymbol{\Sigma} - \boldsymbol{\Phi}$. However, as seen in our experiments $(\widehat{\boldsymbol{\Phi}}, \widehat{\boldsymbol{\Theta}})$ estimated by the remaining methods MINRES, PA, WLS, GLS, MLE and others often violate the PSD condition on $\boldsymbol{\Sigma} - \widehat{\boldsymbol{\Phi}}$ for some choices of r, thereby rendering the interpretation of "Explained Variance" troublesome.

For the MTFA method with estimator $\widehat{\boldsymbol{\Theta}}$, the measure (2.43) applies only for the value of $r = \text{rank}(\widehat{\boldsymbol{\Theta}})$ and the explained variance is one.

For the methods we have included in our comparisons, we report the values of "Explained Variance" as delivered by the `R`-package implementations.

Proximity between $\widehat{\boldsymbol{\Theta}}$ and $\boldsymbol{\Theta}$: A relevant measure of the proximity between $\boldsymbol{\Theta}$ and its estimate $(\widehat{\boldsymbol{\Theta}})$ is given by

$$\text{Error}(\boldsymbol{\Theta}) := \|\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}_r\|_2^2 / \|\boldsymbol{\Theta}_r\|_2^2, \tag{2.44}$$

where $\boldsymbol{\Theta}_r$ is the best rank-r approximation to $\boldsymbol{\Theta}$ and can be viewed as the natural "oracle" counterpart of $\widehat{\boldsymbol{\Theta}}$. Note that MTFA does not incorporate any constraint on $\text{rank}(\widehat{\boldsymbol{\Theta}})$ in its formulation. Since the estimates obtained by this procedure satisfy $\widehat{\boldsymbol{\Theta}} = \boldsymbol{\Sigma} - \widehat{\boldsymbol{\Phi}}$, $\text{rank}(\widehat{\boldsymbol{\Theta}})$ may be quite different from r.

Discussion of experimental results. We next discuss our findings from the numerical experiments for the synthetic datasets.

Table 2.1 shows the performances of the various methods for different choices of p and R for class A_1. For the problems (CFA$_q$), $q \in \{1, 2\}$, we present the

results of Algorithm 1. (Results obtained by the approach in Appendix A.2 were similar.) In all the examples, with the exception of MTFA, we set the number of factors to be $r = (R-1)$, one less than the rank of the covariance matrix for the common underlying factors; in other words, the "remaining" rank-one component can be considered as noise. For MTFA, the rank of $\widehat{\boldsymbol{\Theta}}$ was computed as the number of eigenvalues of $\widehat{\boldsymbol{\Theta}}$ larger than 10^{-5}. MTFA and (CFA$_q$), $q \in \{1, 2\}$, estimate $\boldsymbol{\Phi}$ with zero error—significantly better than competing methods. While MTFA and (CFA$_q$), $q \in \{1, 2\}$, result in estimates such that $\boldsymbol{\Sigma} - \widehat{\boldsymbol{\Phi}}$ is PSD, other methods, however, fail to do so; indeed, the discrepancy can often be quite large. MTFA performs poorly in terms of estimating $\boldsymbol{\Theta}$ since the estimated $\boldsymbol{\Theta}$ has rank different than r. In terms of the proportion of variance explained (CFA$_q$) performs significantly better than all other methods. The notion of "Explained Variance" by MTFA for $r = (R-1)$ is not applicable since the rank of the estimated $\boldsymbol{\Theta}$ is larger than r.

Performance measure	Method Used (CFA$_1$)	(CFA$_2$)	MTFA	MINRES	WLS	PA	PC	MLE	Problem size (R/p)
Error($\boldsymbol{\Phi}$)	0.0	0.0	0.0	39.85	39.18	39.28	2.47	40.04	10/200
Expl Var	0.937	0.937	-	0.445	0.445	0.445	0.469	0.445	
$\lambda_p(\boldsymbol{\Sigma} - \widehat{\boldsymbol{\Phi}})$	0.0	0.0	0.0	-0.204	-0.229	-0.233	-0.206	-0.204	
Error($\boldsymbol{\Theta}$)	0.0	0.0	35.94	0.053	0.040	0.042	2.47	0.056	
Error($\boldsymbol{\Phi}$)	0.0	0.0	0.0	301.94	301.08	301.04	160.29	302.1	10/500
Expl Var	0.929	0.929	-	0.444	0.444	0.444	0.454	0.444	
$\lambda_p(\boldsymbol{\Sigma} - \widehat{\boldsymbol{\Phi}})$	0.0	0.0	0.0	-0.329	-0.328	-0.328	-0.321	-0.330	
Error($\boldsymbol{\Theta}$)	0.0	0.0	291.44	0.051	0.042	0.041	2.395	0.052	
Error($\boldsymbol{\Phi}$)	0.0	0.0	0.0	1682	1681	1681	1311	1682	10/1000
Expl Var	0.915	0.915	-	0.436	0.436	0.436	0.441	0.436	
$\lambda_p(\boldsymbol{\Sigma} - \widehat{\boldsymbol{\Phi}})$	0.0	0.0	0.0	-0.264	-0.268	-0.268	-0.263	-0.264	
Error($\boldsymbol{\Theta}$)	0.0	0.0	1654.3	0.067	0.057	0.057	2.420	0.067	

Table 2.1: Comparative performances of the various FA methods for data of type A_1, for different choices of R and p. "Expl Var" denotes explained variance. In all the above methods (apart from MTFA), r was taken to be $(R-1)$. In all of the cases, rank($\widehat{\boldsymbol{\Theta}}$) obtained by MTFA is seen to be R. The "-" symbol implies that the notion of explained variance is not meaningful for MTFA for $r = R-1$. No method in Category (B) satisfies $\boldsymbol{\Sigma} - \boldsymbol{\Phi} \succcurlyeq \mathbf{0}$. Methods proposed herein seem to significantly outperform their competitors, as seen across the different performance measures.

Figure 2-1 displays results for type A_2. Here we present the results for (CFA$_q$), $q \in \{1, 2\}$, using Algorithm 1. For all the methods (with the exception of MTFA) we computed estimates of $\boldsymbol{\Theta}$ and $\boldsymbol{\Phi}$ for a range of values of r. MTFA and (CFA$_1$) do a

perfect job in estimating $\mathbf{\Phi}$ and both deliver PSD matrices $\mathbf{\Sigma} - \widehat{\mathbf{\Phi}}$. MTFA computes solutions ($\widehat{\mathbf{\Theta}}$) with a higher numerical rank and with large errors in estimating $\mathbf{\Theta}$ (for smaller values of r). Among the four performance measures corresponding to MTFA, Error($\mathbf{\Theta}$) is the only one that varies with different r values. Each of the other three measures deliver a single value corresponding to $r = \text{rank}(\widehat{\mathbf{\Theta}})$. Overall, it appears that (CFA$_q$) is significantly better than all other methods.

Figure 2-2 shows the results for classes B_1, B_2, and B_3. We present the results for (CFA$_q$) for $q \in \{1, 2\}$ using Algorithm 1 as before. Figure 2-2 shows the performance of the different methods in terms of four different metrics: error in $\mathbf{\Phi}$ estimation, proportion of variance explained, violation of the PSD constraint on $\mathbf{\Sigma} - \mathbf{\Phi}$, and error in $\mathbf{\Theta}$ estimation. For the case of B_1, we see that the proportion of explained variance for (CFA$_q$) reaches one at a rank smaller than that of MTFA—this shows that the nonconvex criterion (CFA$_q$) provides smaller estimates of the rank than its convex relaxation MTFA when one seeks a model that explains the full proportion of residual variance. This result is qualitatively different from the behavior seen for A_1 and A_2 where the benefit of (CFA$_q$) over MTFA was mainly due to its flexibility to control the rank of $\mathbf{\Theta}$. Algorithms in Category (A) do an excellent job in estimating $\mathbf{\Phi}$. All other competing methods perform poorly in estimating $\mathbf{\Phi}$ for small/moderate values of r. We observe that none of the methods apart from (CFA$_q$) and MTFA lead to PSD estimates of $\mathbf{\Sigma} - \widehat{\mathbf{\Phi}}$ (unless r becomes sufficiently large which corresponds to a model with a saturated fit). In terms of the proportion of variance explained, our proposal performs much better than the competing methods. We see that the error in $\mathbf{\Theta}$ estimation incurred by (CFA$_q$), increases marginally as soon as the rank r becomes larger than a certain value for B_1. Note that around the same values of r, the proportion of explained variance reaches one in both these cases, thereby suggesting that this is possibly not a region of statistical interest. In summary, Figure 2-2 suggests that (CFA$_q$) performs very well compared to all its competitors.

Summary: All methods of Category (B) (see Section 2.1.2) used in the experimental comparisons perform worse than Category (A) in terms of measures Error($\mathbf{\Phi}$),

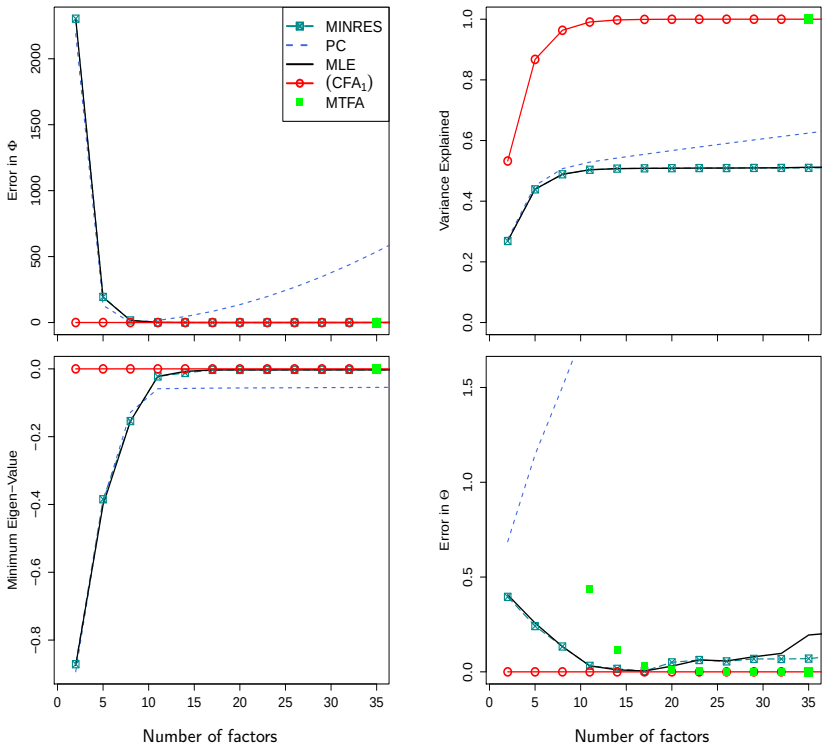


Figure 2-1: Performance of various FA methods for class $A_2(200)$ as a function of the number of factors. The vertical label "Minimum Eigenvalue" refers to $\lambda_p(\mathbf{\Sigma} - \widehat{\mathbf{\Phi}})$. We present the results of (CFA$_1$), as obtained via Algorithm 1—the results of (CFA$_2$) were similar, and hence omitted from the plot. Our methods seems to perform better than all the other competitors. For large values of r, as the fits saturate, the methods become similar. The methods (as available from the `R` package implementations) that experienced convergence difficulties do not appear in the plot.

Error($\mathbf{\Theta}$) and Explained Variance for small/moderate values of r. They also lead to indefinite estimates of $\mathbf{\Sigma} - \widehat{\mathbf{\Phi}}$. MTFA performs well in estimating $\mathbf{\Phi}$ but fails in estimating $\mathbf{\Theta}$ mainly due to the lack in flexibility of imposing a rank constraint; in some cases the trace heuristic falls short of doing a good job in approximating the rank function when compared to its nonconvex counterpart (CFA$_q$). The estimation methods proposed herein have a significant edge over existing methods in producing high quality solutions across various performance metrics.

2.5.2 Real data examples

This section describes the performance of different FA methods on some real-world benchmark datasets popularly used in the context of FA. These datasets can be found

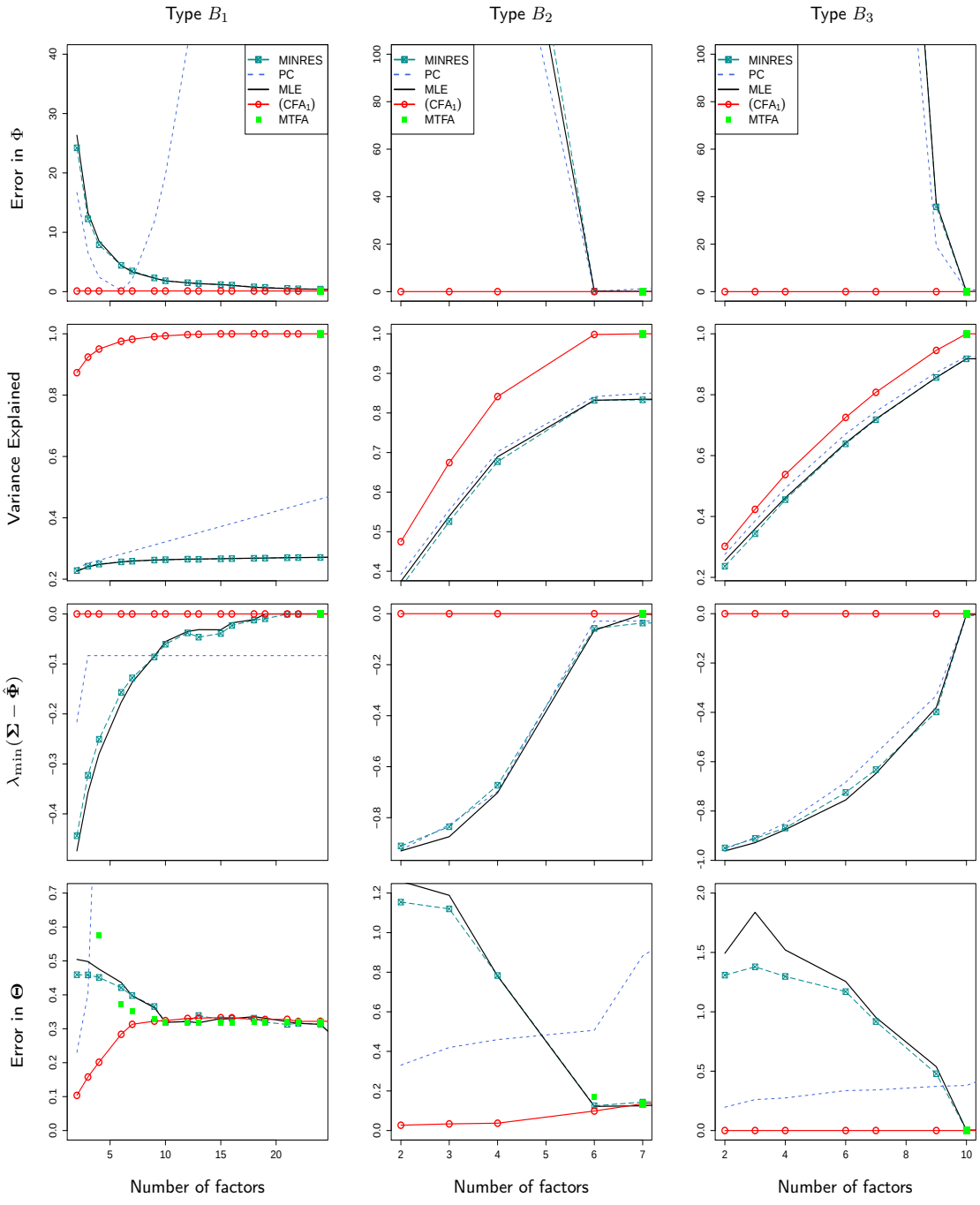


Figure 2-2: Performance of different methods for instances of $B_1(30/100)$, $B_2(5/10/100)$, and $B_3(5/10/100)$. We see that (CFA$_1$) exhibits very good performance across all instances, significantly outperforming the competing methods (the results of (CFA$_2$) were similar).

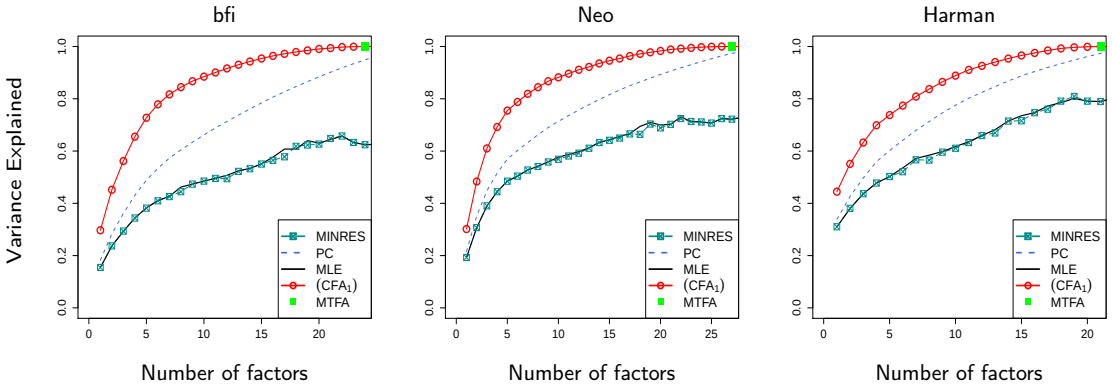


Figure 2-3: Proportion of variance explained by different methods for real-data examples. We see that in terms of the proportion of explained variance, (CFA_1) delivers the largest values for different values of r, which is indeed desirable. (CFA_1)also shows nice flexibility in delivering different models with varying r, in contrast to MTFA which delivers one model with proportion of variance explained equal to one. The results of (CFA_2) were found to be similar to (CFA_1).

in the `R` libraries `datasets` [123], `psych` [127], and `FactoMineR` [84] and are as follows:

- The bfi data set has 2800 observations on 28 variables (25 personality self-reported items and 3 demographic variables).
- The neo data set has 1000 measurements for $p = 30$ dimensions.
- The Harman data set is a correlation matrix of 24 psychological tests given to 145 seventh- and eighth-grade children.
- The geomorphology data set is a collection of geomorphological data collected across $p = 10$ variables and 75 observations. (The geomorphology data set originally has $p = 11$, but we remove the one categorical feature, leaving $p = 10$.)

We present the results in Figure 2-3. We also experimented with other methods—such as WLS and GLS—but the results were similar to MINRES and hence have not been shown in the figure. For the real examples, most of the performance measures described in Section 2.5.1 do not apply;[17] however, the notion of explained vari-

[17]Understanding performance of FA methods on real data is difficult because it is an unsupervised problem. However, we can understand the performances of different methods by drawing parallels with PCA in terms of the proportion of variance explained of the matrix $\mathbf{\Sigma} - \mathbf{\Phi}$—see our discussion in Section 2.1.

ance (2.43) does apply. We used this metric to compare the performance of different competing estimation procedures. We observe that solutions delivered by Category (B) explain the maximum amount of residual variance for a given rank r, which is indeed desirable, especially in the light of its analogy with PCA on the residual covariance matrix $\mathbf{\Sigma} - \mathbf{\Phi}$.

2.5.3 Certificates of optimality via Algorithm 2

We now turn our attention to certificates of optimality using Algorithm 2. Computational results of Algorithm 2 for a variety of problem sizes across all six classes can be found in Tables 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, and 2.10. In general, we provide results for p ranging between 10 and 4000. Parametric choices are outlined in depth in Table 2.2.[18] We always initialize Algorithm 2 with an initial feasible solution as found via Algorithm 1 so that we can understand if the estimators found via the CG-based approach are indeed optimal. In particular, if the best feasible objective value does not change throughout the branch-and-bound algorithm, then the initial estimator was indeed optimal (within the numerical tolerance).

Root node gap: Let us first consider the gap at the root node. In classes A_1, B_1, B_2, and B_3, we see that the Weyl bound at the root node often provides a better bound than the one given by using convex envelopes. Indeed, the bound provided by Weyl's method can in many instances certify optimality (up to numerical tolerance) at the *root node*. For example, this is the case in many instances of classes A_1 and half of the instances in B_3. Given that Weyl's method is computationally inexpensive (only requiring two eigenvalue decompositions), this suggests that Weyl's inequality as used within the context of factor analysis is particularly fruitful.

In contrast, in class A_2 and the real examples, we see that the convex envelope

[18] All computational experiments are performed in a shared cluster computing environment with highly variable demand, and therefore runtimes are not necessarily a reliable measure of problem complexity; hence, the number of nodes considered is always displayed. Further, Algorithm 2 is highly parallelizable, like many branch-and-bound algorithms; however, our implementation is serial. Therefore, with improvements in code design, it is very likely that runtimes can be substantially improved beyond those shown here.

	Root node			Terminal node			
Problem size (R/p)	Upper bound	CE LB	Weyl LB	Upper bound	Lower bound	Nodes	Time (s)
2/10	1.54	1.44	1.43	1.54	1.44	1	0.14
3/10	0.88	0.49	0.70	0.88	0.78	78	4.08
5/10	0.43	−0.90	0.10	0.43	0.33	28163	19432.70
2/20	3.99	3.91	3.94	3.99	3.94	1	0.19
3/20	2.33	2.06	2.24	2.33	2.33	1	0.21
5/20	0.61	−0.07	0.49	0.61	0.51	626	75.98

Table 2.2: Computational results for Algorithm 2 for class $A_1(R/p)$. All computations are performed in `julia` using SDO solvers `MOSEK` (for primal feasible solutions) and `SCS` (for dual feasible solutions within tolerance 10^{-3}) via the `JuMP` interface [56]. Computation time does not include preprocessing (such as computation of $\mathbf{u}$ as in (2.37) and finding an initial incumbent feasible solution $\mathbf{\Phi}_f$ as computed via the conditional gradient algorithm in Section 2.3). We always use default tolerance `TOL` $= 0.1$ for algorithm termination. Parameters for branching, pruning, node selection, etc., are detailed throughout Section 2.4. Upper bounds denote z_f, which is the best feasible solution found thus far (either at the root node or at algorithm termination). At the root node, we display two lower bounds: the lower bound arising from convex envelopes (denoted "CE LB") and the one arising from the Weyl bound (denoted "Weyl LB"). Note that for lower bound at the termination node, we mean the worst bound $\max\{z^c, w^c\}$ (see Section 2.4.6; z^c is from the convex envelope approach, while w^c comes from Weyl's method). "Nodes" indicates the number of nodes considered in the course of execution, while "Time (s)" denotes the runtime (in seconds). We set r^*, the rank used within Algorithm 2, to $r^* = R - 1$, where R is the generative rank displayed. All results displayed to two decimals. Computations run on high-demand, shared cluster computing environment with variable architectures. Runtime is capped at 400000s (approximately 5 days), and any instance which is still running at that time is marked with an asterisk next to its runtime.

	Root node			Terminal node			
Problem size (R/p)	Upper bound	CE LB	Weyl LB	Upper bound	Lower bound	Nodes	Time (s)
3/50	7.16	7.09	7.14	7.16	7.14	1	18.66
5/50	3.04	2.86	3.01	3.04	3.01	1	19.13
10/50	0.88	−0.15	0.81	0.88	0.81	1	6.22
3/100	13.72	13.69	13.72	13.72	13.72	1	125.17
5/100	8.54	8.46	8.53	8.54	8.53	1	117.71
10/100	2.62	2.10	2.58	2.62	2.58	1	36.91

Table 2.3: Computational results for larger examples from class $A_1(R/p)$. Same parameters as in Table 2.2. Again we set $r^* = R - 1$.

		Root node			Terminal node			
Problem size (p)	r^* used	Upper bound	CE LB	Weyl LB	Upper bound	Lower bound	Nodes	Time (s)
10	2	0.98	−0.29	0.26	0.96	0.86	1955	91.99
	3	0.53	−0.43	0.13	0.53	0.43	3822	409.68
20	2	5.13	4.14	2.13	5.13	5.03	29724	36857.46
	3	4.26	2.68	1.55	4.26	4.05	86687	400002.3*
100	3	38.65	37.50	33.08	38.65	38.18	20282	400015.2*
	5	30.98	29.05	25.69	30.98	29.91	16653	400005.6*

Table 2.4: Computational results for class $A_2(p)$. All parameters as per Table 2.2. Here we show the behavior across a variety of choices of the parameter r^*.

		Root node			Terminal node			
Problem size (r/p)	r^* chosen	Upper bound	CE LB	Weyl LB	Upper bound	Lower bound	Nodes	Time (s)
4/10	1	0.25	0.06	0.11	0.25	0.20	24	0.66
6/10	2	0.09	−0.27	0.00	0.09	0.04	77	0.62
4/20	1	0.26	0.10	0.13	0.26	0.21	21	0.17
6/20	2	0.10	−0.24	0.01	0.10	0.05	73	1.47
6/50	2	0.11	−0.15	0.04	0.11	0.06	50	0.66
10/50	3	0.17	−0.35	0.03	0.17	0.12	2283	333.62
6/100	2	0.12	−0.07	0.05	0.12	0.05	1	0.71
10/100	3	0.19	−0.22	0.06	0.19	0.14	1535	237.19

Table 2.5: Computational results for class $B_1(R/p)$. We choose r^* during computation as the largest r such that Algorithm 1 yields a feasible solution with strictly positive objective value (up to additive tolerance 0.05; we use a smaller value here because the objective values are smaller across this class). For this class, examples can be preprocessed because $\mathbf{\Sigma} \sim B_1(R/p)$ has a block of size $R \times R$ in the upper left, with all other entries set to zero except the diagonal. Hence, it suffices to perform factor analysis with the truncated matrix $\mathbf{\Sigma}_{1:R,1:R}$.

		Root node			Terminal node			
Problem size $(r/R/p)$	r^* chosen	Upper bound	CE LB	Weyl LB	Upper bound	Lower bound	Nodes	Time (s)
2/5/20	4	1.15	0.4	1.03	1.15	1.05	517	146.72
3/5/20	4	1.24	0.46	1.11	1.24	1.14	933	232.74
3/5/100	4	8.40	8.32	8.39	8.40	8.39	1	135.73
3/10/100	9	3.23	2.80	3.20	3.23	3.20	1	50.63

Table 2.6: Computational results for class $B_2(r/R/p)$. Parameters as set in Table 2.2. Here r^* is chosen during computation as the largest r' such that Algorithm 1 procedures a feasible solution with strictly positive objective (up to 0.1). Here $r^* = R - 1$ ends up being an appropriate choice.

		Root node			Terminal node			
Problem size $(r/R/p)$	r^* chosen	Upper bound	CE LB	Weyl LB	Upper bound	Lower bound	Nodes	Time (s)
2/5/20	3	0.30	−0.30	0.13	0.30	0.20	777	219.75
3/5/20	2	1.00	0.62	0.82	1.00	0.90	217	48.43
2/5/100	3	0.55	0.24	0.40	0.55	0.45	7099	47651.57
3/5/100	2	1.03	0.70	0.81	1.03	0.93	4906	34514.69
2/10/100	8	0.12	−0.58	0.05	0.12	0.05	1	77.38
3/10/100	6	0.33	−0.73	0.20	0.33	0.22	27770	400000.0*

Table 2.7: Computational results for class $B_3(r/R/p)$. Parameters as set in Table 2.2. As in Tables 2.5 and 2.6, here r^* is chosen during computation as largest r' such that Algorithm 1 produces a feasible solution with strictly positive objective (up to 0.1).

bound tends to perform better. Because of the structure of Weyl's inequality, Weyl's method is well-suited for matrices $\mathbf{\Sigma}$ with very quickly decaying eigenvalues. Examples in these two classes do not have such a spectrum, and indeed Weyl's method does not provide the best root node bound.[19] Because neither Weyl's method nor the convex envelope bound strictly dominate one another at the root node across all examples, our approach incorporating both can leverage the advantages of each.

Observe that the root node gap (either in terms of the absolute difference between the initial feasible solution found and the better of the convex envelope bound and the Weyl bound) tends to be smaller when r^* is much smaller than p. This suggests that the approach we take is well-suited to certify optimality of particularly low-rank

[19]However, it is worth remarking that Weyl's method still provides lower bounds on the rank of solutions to the noiseless factor analysis problem. Hence, even in settings where Weyl's method is not necessarily well-suited for proving optimality for the noisy factor analysis problem, it can still be applied successfully to lower bound rank for noiseless factor analysis.

	Root node			Terminal node			
r^* used	Upper bound	CE LB	Weyl LB	Upper bound	Lower bound	Nodes	Time (s)
1	4.06	3.78	2.53	4.06	3.96	44	0.64
2	2.64	2.04	1.42	2.64	2.54	1885	142.06
3	1.56	0.62	0.61	1.56	1.46	11056	2458.23
4	0.88	−0.33	0.28	0.88	0.78	66877	60612.61
5	0.36	−0.88	0.0	0.36	0.25	155759	400012.4*

Table 2.8: Computational results for the geomorphology example ($p = 10$). Parameters as set in Table 2.2. Here we display results for the choices of $r^* \in \{1, 2, 3, 4, 5\}$ (for $r^* > 5$, the upper bound is below the numerical tolerance, so we do not include those).

	Root node			Terminal node			
r^* used	Upper bound	CE LB	Weyl LB	Upper bound	Lower bound	Nodes	Time (s)
1	9.88	9.64	5.89	9.88	9.78	158	30.13
2	7.98	7.54	4.22	7.98	7.88	31710	49837.17
3	6.53	5.85	3.01	6.53	6.35	81935	400003.8*

Table 2.9: Computational results for the Harman example ($p = 24$). Parameters as set in Table 2.2. Here we display results for the choices of $r^* \in \{1, 2, 3\}$.

Example	r^*	Upper bound	Lower bound
$A_1(100/1000)$	10	460.35	457.46
	50	198.70	197.13
	90	28.65	28.34
$A_2(1000)$	10	184.22	183.18
	30	20.25	19.50
	50	2.17	1.71
$B_2(20/90/1000)$	20	378.37	376.64
	40	235.16	233.94
	80	34.33	34.09
$B_3(20/150/1000)$	20	426.31	422.13
	70	174.00	171.70
	120	20.30	19.80

Example	r^*	Upper bound	Lower bound
$A_1(360/4000)$	100	1334.40	1327.60
	200	693.02	688.87
	350	30.75	30.50
$A_2(4000)$	10	733.41	733.09
	50	8.57	8.39
	70	0.90	0.79
$B_2(80/360/4000)$	100	1360.1	1354.04
	200	703.78	700.21
	350	31.02	30.91
$B_3(120/360/4000)$	100	1081.20	1078.74
	200	253.50	252.50
	240	7.89	7.46

Table 2.10: Computational results across several classes for larger scale instances with $p \in \{1000, 4000\}$. Results are displayed across a variety of choices of rank r^*. The "Upper bound" denotes the upper bound found by the conditional gradient method, while "Lower bound" denotes the Weyl bound at the root node (no convex envelope bounds via Algorithm 2 are shown here because of the large nature of the SDO-based convex envelope lower bounds).

decompositions in noisy factor analysis settings. We see that this phenomenon is true across all classes. The numerical results suggest that corresponding theoretical guarantees for Weyl's method are potentially of interest in their own right and warrant further consideration.

Finally, we remark that if the true convex envelope of the objective over the set of semidefinite constraints was taken, then the convex envelope objective would always be nonnegative. However, because we have taken the convex envelope of the objective over the polyhedral constraints only, this is not the case.

Performance at termination: It is particularly encouraging that the initial feasible solutions provided via Algorithm 1 remain the best feasible solution throughout the execution of Algorithm 2 for all but a few instances. (Of course, this need not be universally true.) This is an important observation to make because without a provable optimality scheme such as the one we consider here, it is difficult to quantify the performance of heuristic upper bound methods. As we demonstrate here, despite the only local guarantees of solutions obtained via a conditional gradient scheme, they tend to perform quite well in the setting of factor analysis. Indeed, even in the instances where the best feasible solution is improved, the improved solution is found very early in the branching process.

Across the different example classes, we see that in general the gap tends to decrease more when r^* is small relative to p and p is smaller. To appropriately contextualize and appreciate the number of nodes solved for a problem with $p = 100$ on the timescale of 100s, with state-of-the-art implementations of interior point methods, solving a single node in the branch and bound tree can take on the order of 40s (for the specifically structured problems of interest and on the same machine). In other words, if one were to naïvely use interior point methods, it would only be possible to solve approximately three nodes during a 100s time limit. In contrast, by using a first-order method approach which facilitates warm starts, we are able to solve hundreds of nodes in the same amount of time.

We see that Algorithm 2 performs particularly well for classes A_1 (Tables 2.2

and 2.3) and B_1 (Table 2.5), for problems of reasonable size with relatively small underlying ranks. This is highly encouraging. Class A_1 forms a set of prototypical examples for which theoretical recovery guarantees perform well; in stark contrast, problems such as those in B_1 which have highly structured underlying factors tend to not satisfy the requirements of such recovery guarantees. Indeed, if $\mathbf{\Sigma} \sim B_1(R/p)$, then there generally appears to be a rank $R/2$ matrix $\mathbf{\Theta} \succcurlyeq \mathbf{0}$ so that $\mathbf{\Sigma} - \mathbf{\Theta}$ is positive semidefinite and diagonal. In such a problem, for r^* on the order of $R/2 - 1$, we provide nearly optimal solutions within a reasonable time frame.

Further, we note that similar results to those obtained for classes A_1 and B_1 are obtained for the other classes and are detailed in Tables 2.4 (class A_2), 2.6 (class B_2), 2.7 (class B_3), and 2.8 and 2.9. In a class such as A_2, which is generated as a high rank matrix (with decaying spectrum) with added individual variances, theoretical recovery guarantees do not generally apply, so again it is encouraging to see that our approach still makes significant progress towards proving optimality. Further, as shown in Table 2.10, for a variety of problems with p on the order of 1000 or 4000, solutions can be found in seconds and optimality can be certified within minutes via Weyl bounds, with no need for convex envelope bounds as computed via Algorithm 2, so long as the rank r^* is sufficiently small (for classes A_1, B_2, and B_3 on the order of hundreds, and for A_2 on the order of tens). This strongly supports the value of such an eigenvalue-based approach. When computing lower bounds solely via Weyl's method, the only necessary computations are two eigenvalue decompositions. As Table 2.10 suggests, for sufficiently small rank, one can still quickly find certifiably optimal solutions even for very large-scale factor analysis problems.

Finally, we note that all synthetic examples we have considered have equal proportions of common and individual variances (although, of course, this is not exploited by our approach as this information is not *a priori* possible to specify without additional contextual information). If one modifies the classes so that the proportion of the common variances is higher than the proportion of individual variances (in the generative example), then Algorithm 2 is able to deliver better bounds on a smaller time scale. (Results are not included here.) This is not particularly surprising be-

cause the branch-and-bound approach we take essentially hinges on how well the products $W_{ii}\Phi_i$ can be approximated. When there is underlying factor structure with a lower proportion of individual variances, the scale of $W_{ii}\Phi_i$ is smaller and hence these products are easier to approximate well.

2.5.4 Additional considerations

We now turn our attention to assessing the benefits of various algorithmic modifications as presented in Section 2.4. We illustrate the impact of these by focusing on four representative examples from across the classes: $A_1(3/10)$, $B_1(6/50)$, $B_3(3/5/20)$, and $\mathsf{G} := \mathsf{geomorphology}$. All relevant results can be found in Table 2.11.

Performance of branching strategy: We begin by considering the impact of our branching strategy as developed in Section 2.4.4. The results across the four examples are shown in Table 2.11a. Recall that $\epsilon \in [0, 1)$ controls the extent of deviation from the canonical branching location, with $\epsilon = 0$ corresponding to no deviation. Across all examples, we see that the number of nodes considered to prove optimality is approximately convex in $\epsilon \in [0, 0.5]$. In particular, for all examples, the "optimal" choice of ϵ is not the canonical choice of $\epsilon = 0$. This contrast is stark for the examples $B_3(3/5/20)$ and G. Indeed, for these two examples, the number of nodes considered when $\epsilon = 0$ is over five times larger than for any $\epsilon \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$.

In other words, the alternative branching strategy can have a substantial impact on the number of nodes considered in the branch-and-bound tree. As a direct consequence, this strategy can drastically reduce the computation time needed to prove optimality. As the examples suggest, it is likely that ϵ should be chosen dynamically during algorithm execution, as the particular choice depends on a given problem's structure. However, we set $\epsilon = 0.4$ for all other computational experiments because this appears to offer a distinct benefit over the naïve strategy of setting $\epsilon = 0$.

Performance of node selection strategy: We now turn our attention to the node selection strategy as detailed in Section 2.4.6. Recall that node selection considers

Example	r^*	Nodes considered for $\epsilon =$					
		0.0	0.1	0.2	0.3	0.4	0.5
$A_1(3/10)$	2	103	72	59	62	78	101
$B_1(6/50)$	2	39	33	29	40	50	53
$B_3(3/5/20)$	2	8245	762	278	189	217	203
G	1	937	162	59	46	44	52

(a) Effect of branching strategy

Example	r^*	Nodes considered for	
		Naïve strategy	Modified strategy
$A_1(3/10)$	2	99	78
$B_1(6/50)$	2	135	50
$B_3(3/5/20)$	2	375	217
G	1	43	44

(b) Effect of node selection strategy

Example	r^*	Nodes considered for TOL =		
		0.10	0.05	0.025
$A_1(3/10)$	2	78	287	726
$B_1(6/50)$	2	1	50	262
$B_3(3/5/20)$	2	217	1396	5132
G	1	44	217	978

(c) Effect of numerical tolerance

Example	r^*	Upper Bound	CE LB with tightening	CE LB without tightening
$A_1(3/10)$	2	0.88	0.70	0.39
$B_1(6/50)$	2	0.11	−0.15	−0.17
$B_3(3/5/20)$	2	1.00	0.62	0.51
G	1	4.06	3.78	3.78

(d) Effect of root node bound tightening

Table 2.11: Computational results for effects of algorithmic modifications. Unless explicitly stated, all other parameters are as in Table 2.2. We consider how the number of nodes needed to prove optimality changes across different choices of the following: (a) ϵ as in Section 2.4.4; (b) node selection strategy (either naïve and modified) as described in Section 2.4.6; and (c) numerical tolerance TOL for algorithm termination. Finally, in (d) we show how the convex envelope lower bound (denoted "CE LB") compares with and without bound tightening at the root node (see Section 2.4.5). "Upper bound" is included for scale (i.e., to compare the relative impact of tightening).

how to pick which node to consider next in the current branch-and-bound tree at any iteration of Algorithm 2. We compare two strategies: the naïve strategy which selects the node with worst convex envelope bound (as explicitly written in Algorithm 2) and the modified strategy which employs randomness and Weyl bounds to consider nodes which might not be advantageous to fathom when only convex envelope bounds are considered.

The comparison is shown in Table 2.11b. We see that this strategy is advantageous overall (with only example G showing a negligible decrease in performance). The benefit is particularly strong for examples from the B classes which have highly structured underlying factors. For such examples, there is a large difference between the convex envelope bounds and the Weyl bounds at the root node (see e.g. Table 2.5). Hence, an alternative branching strategy which incorporates the eigenvalue information provided by Weyl bounds has potential to improve beyond the naïve strategy. Indeed, this appears to be the case across all examples where such behavior occurs.

Influence of optimality tolerance: Now let us consider the influence of the additive optimality tolerance for termination, TOL. In particular, we study how the number of nodes to prove optimality changes as a function of additive gap necessary for termination. The corresponding results across the four examples are shown in Table 2.11c. Not surprisingly, as the gap necessary for termination is progressively halved, the corresponding number of nodes considered increases substantially. However, it is important to note that even though the gap at termination is smaller as this tolerance decreases (by design), for these examples the best feasible solution remains unchanged. In other words, the increase in the number of nodes appears to be the price for more accurately proving optimality and not for finding better feasible solutions. Indeed, as noted earlier, the solutions found via conditional gradient methods at the outset are of remarkably high quality.

Performance of bound tightening: We close this section by considering bound tightening. All computational results employ bound tightening as developed in Section 2.4.5, but only at the root node. Bound tightening, which requires repeated eigenvalue computations, is a computationally expensive process. For this reason, we have chosen not to employ bound tightening at every node in the branch-and-bound tree. From a variety of computational experiments, we observed that the most important node for bound tightening is the root node, and therefore it is a reasonable choice to only employ bound tightening there. Consequently, we employ pruning via Weyl's method (detailed in Section 2.4.5 as well) at all nodes in the branch-and-bound tree. (Recall that bound tightening can be thought of as optimal pruning via Weyl's method.)

In Table 2.11d we show the impact of bound tightening at the root node in terms of the improvement in the lower bounds provided by convex envelopes. The results for the class A_1 are particularly distinctive. Indeed, for this class bound tightening has a substantial impact on the quality of the convex envelope bound (for the example $A_1(3/10)$ given, the improvement is from a relative gap at the root node of 56% to a gap of 20%). For the examples shown, bound tightening offers the least improvement in the real example G. In light of Table 2.8 this is not too surprising, as Weyl's method (at the root node) is not particular effective for this example. As Weyl's inequality is central to bound tightening, problems for which Weyl's inequality is not particularly effective tend to experience less benefit from bound tightening at the root node.

2.6 Conclusions

In this chapter, we analyzed the classical rank-constrained FA problem from a computational perspective. We proposed a general, flexible family of rank-constrained, nonlinear SDO-based formulations for the task of approximating an observed covariance matrix $\mathbf{\Sigma}$ as the sum of a PSD low-rank component $\mathbf{\Theta}$ and a diagonal matrix $\mathbf{\Phi}$ (with nonnegative entries) subject to $\mathbf{\Sigma} - \mathbf{\Phi}$ being PSD. Our framework enables us to estimate the underlying factors and unique variances under the restriction that

the residual covariance matrix is semidefinite—this is important for statistical interpretability and understanding the proportion of variance explained by a given number of factors. This constraint, however, seems to ignored by most other widely used methods in FA.

We introduce a novel *exact* reformulation of the rank-constrained FA problem as a smooth optimization problem with convex, compact constraints. We present a unified algorithmic framework, utilizing modern techniques in nonlinear optimization and first order methods in convex optimization to obtain high-quality solutions for the FA problem. At the same time, we use techniques from discrete and global optimization to demonstrate that these solutions are often provably optimal. We provide computational evidence demonstrating that the methods proposed herein provide high quality estimates with improved accuracy when compared to existing, popularly-used methods in FA.

In this work we have demonstrated that a previously intractable rank optimization problem can be solved to provable optimality. We envision that ideas similar to those used here can be used to address an even larger class of estimation problems with underlying matrix structure. In this way, we anticipate significant progress on such problems in the next decade, particularly in light of myriad advances throughout distinct areas of modern optimization.

Chapter 3

Equivalence of Robustification and Regularization

3.1 Introduction

The development of predictive methods that perform well in the face of uncertainty is at the core of modern machine learning and statistical practice. Indeed, the notion of *regularization*—loosely speaking, a means of controlling the ability of a statistical model to generalize to new settings by trading off with the model's complexity—is at the very heart of such work [77]. Corresponding regularized statistical methods, such as the Lasso for linear regression [156] and nuclear-norm-based approaches to matrix completion [126, 44], are now ubiquitous and have seen widespread success in practice.

In parallel to the development of such regularization methods, it has been shown in the field of robust optimization that under certain conditions these regularized problems result from the need to immunize the statistical problem against adversarial perturbations in the data [58, 162, 19, 47]. Such a *robustification* offers a different perspective on regularization methods by identifying which adversarial perturbations the model is protected against. Conversely, this can help to inform statistical modeling decisions by identifying potential choices of regularizers. Further, this connection between regularization and robustification offers the potential to use sophisticated

data-driven methods in robust optimization [29, 160] to design regularizers in a principled fashion.

With the continuing growth of the adversarial viewpoint in machine learning (e.g. the advent of new deep learning methodologies such as generative adversarial networks [70, 71, 137]), it is becoming increasingly important to better understand the connection between robustification and regularization. Our goal in this chapter is to shed new light on this relationship by focusing in particular on linear and matrix regression problems. Specifically, our contributions include:

1. In the context of linear regression, we demonstrate that in general such a robustification procedure is not equivalent to regularization (via penalization). We characterize precisely under which conditions on the model of uncertainty used and on the loss function penalties one has that robustification is equivalent to regularization.

2. We consider problems in the matrix setting, such as matrix completion and Principal Component Analysis (PCA). We show that the nuclear norm, a popular penalty function used throughout this setting, arises directly through robustification. As with the case of vector regression, we characterize under which conditions on the model of uncertainty there is equivalence of robustification and regularization in the matrix setting.

The structure of the chapter is as follows. In Section 3.2, we review background on norms and consider robustification and regularization in the context of linear regression, focusing both on their equivalence and non-equivalence. In Section 3.3, we turn our attention to regression with underlying matrix variables, considering in depth both matrix completion and PCA. In Section 3.4, we include some concluding remarks.

3.2 A robust perspective of linear regression

We begin by briefly introducing the necessary notation and background on norms which we will use to address the equivalence of robustification and regularization in the context of linear regression.

3.2.1 Norms and their duals

Given a vector space $V \subseteq \mathbb{R}^n$ we say that $\|\cdot\| : V \to \mathbb{R}$ is a *norm* if for all $\mathbf{v}, \mathbf{w} \in V$ and $\alpha \in \mathbb{R}$

1. If $\|\mathbf{v}\| = 0$, then $\mathbf{v} = 0$,
2. $\|\alpha\mathbf{v}\| = |\alpha|\|\mathbf{v}\|$ (absolute homogeneity), and
3. $\|\mathbf{v} + \mathbf{w}\| \leq \|\mathbf{v}\| + \|\mathbf{w}\|$ (triangle inequality).

If $\|\cdot\|$ satisfies conditions 2 and 3, but not 1, we call it a *seminorm*. For a norm $\|\cdot\|$ on $\mathbb{R}^n$ we define its dual, denoted $\|\cdot\|_*$, to be

$$\|\boldsymbol{\beta}\|_* := \max_{\mathbf{x} \in \mathbb{R}^n} \frac{\mathbf{x}'\boldsymbol{\beta}}{\|\mathbf{x}\|}.$$

For example, the ℓ_p norms $\|\boldsymbol{\beta}\|_p := \left(\sum_i |\beta_i|^p\right)^{1/p}$ for $p \in [1, \infty)$ and $\|\boldsymbol{\beta}\|_\infty := \max_i |\beta_i|$ satisfy a well-known duality relation: ℓ_{p^*} is dual to ℓ_p, where $p^* \in [1, \infty]$ with $1/p + 1/p^* = 1$. We call p^* the *conjugate* of p. More generally for matrix norms[1] $\|\cdot\|$ on $\mathbb{R}^{m \times n}$ the dual is defined analogously:

$$\|\boldsymbol{\Delta}\|_* := \max_{\mathbf{A} \in \mathbb{R}^{m \times n}} \frac{\langle \mathbf{A}, \boldsymbol{\Delta} \rangle}{\|\mathbf{A}\|},$$

where $\boldsymbol{\Delta} \in \mathbb{R}^{m \times n}$ and $\langle \cdot, \cdot \rangle$ denotes the trace inner product.We note that the dual of the dual norm is the original norm [38].

[1]We treat a matrix norm as any norm on $\mathbb{R}^{m \times n}$ which satisfies the three conditions of a usual vector norm, although some authors reserve the term "matrix norm" for a norm on $\mathbb{R}^{m \times n}$ which also satisfies a submultiplicativity condition (see [80, p. 341]).

Three widely used choices for matrix norms (see [80]) are Frobenius, spectral, and induced norms. The definitions for these norms are given below for $\mathbf{\Delta} \in \mathbb{R}^{m \times n}$ and summarized in Table 3.1 for convenient reference.

1. The p-Frobenius norm, denoted $\|\cdot\|_{F_p}$, is the entrywise ℓ_p norm on the entries of $\mathbf{\Delta}$:

$$\|\mathbf{\Delta}\|_{F_p} := \left(\sum_{ij} |\Delta_{ij}|^p \right)^{1/p}.$$

   Analogous to before, F_{p^*} is dual to F_p, where $1/p + 1/p^* = 1$.

2. The p-spectral (Schatten) norm, denoted $\|\cdot\|_{\sigma_p}$, is the ℓ_p norm on the singular values of the matrix $\mathbf{\Delta}$:

$$\|\mathbf{\Delta}\|_{\sigma_p} := \|\boldsymbol{\sigma}(\mathbf{\Delta})\|_p,$$

   where $\boldsymbol{\sigma}(\mathbf{\Delta})$ denotes the vector containing the singular values of $\mathbf{\Delta}$. Again, σ_{p^*} is dual to σ_p.

3. Finally we consider the class of induced norms. If $g : \mathbb{R}^m \to \mathbb{R}$ and $h : \mathbb{R}^n \to \mathbb{R}$ are norms, then we define the induced norm $\|\cdot\|_{(h,g)}$ as

$$\|\mathbf{\Delta}\|_{(h,g)} := \max_{\boldsymbol{\beta} \in \mathbb{R}^n} \frac{g(\mathbf{\Delta}\boldsymbol{\beta})}{h(\boldsymbol{\beta})}.$$

   An important special case occurs when $g = \ell_p$ and $h = \ell_q$. When such norms are used, (q, p) is used as shorthand to denote (ℓ_q, ℓ_p). Induced norms are sometimes referred to as operator norms. We reserve the term operator norm for the induced norm $(\ell_2, \ell_2) = (2, 2) = \sigma_\infty$, which measures the largest singular value.

Name	Notation	Definition	Description
p-Frobenius	F_p	$\left(\sum_{ij} \lvert\Delta_{ij}\rvert^p\right)^{1/p}$	entrywise ℓ_p norm
p-spectral (Schatten)	σ_p	$\lVert\boldsymbol{\sigma}(\boldsymbol{\Delta})\rVert_p$	ℓ_p norm on the singular values
Induced	(h, g)	$\max_{\boldsymbol{\beta}} \frac{g(\boldsymbol{\Delta}\boldsymbol{\beta})}{h(\boldsymbol{\beta})}$	induced by norms g, h

Table 3.1: Matrix norms on $\boldsymbol{\Delta} \in \mathbb{R}^{m\times n}$.

3.2.2 Uncertain regression

We now turn our attention to uncertain linear regression problems and regularization. The starting point for our discussion is the standard problem

$$\min_{\boldsymbol{\beta}\in\mathbb{R}^n} g(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}),$$

where $\mathbf{y} \in \mathbb{R}^m$ and $\mathbf{X} \in \mathbb{R}^{m\times n}$ are data and g is some convex function, typically a norm. For example, $g = \ell_2$ is least squares, while $g = \ell_1$ is known as least absolute deviation (LAD). In favor of models which mitigate the effects of overfitting these are often replaced by the *regularization* problem

$$\min_{\boldsymbol{\beta}} g(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) + h(\boldsymbol{\beta}),$$

where $h : \mathbb{R}^n \to \mathbb{R}$ is some penalty function, typically taken to be convex. This approach often aims to address overfitting by penalizing the complexity of the model, measured as $h(\boldsymbol{\beta})$. (For a more formal treatment using Hilbert space theory, see [36, 16].) For example, taking $g = \ell_2^2$ and $h = \ell_2^2$, we recover the so-called regularized least squares (RLS), also known as ridge regression [77]. The choice of $g = \ell_2^2$ and $h = \ell_1$ leads to Lasso, or least absolute shrinkage and selection operator, introduced in [156]. Lasso is often employed in scenarios where the solution $\boldsymbol{\beta}$ is desired to be sparse, i.e., $\boldsymbol{\beta}$ has very few nonzero entries. Broadly speaking, regularization can take much more general forms; for our purposes, we restrict our attention to regularization that appears in the penalized form above.

In contrast to this approach, one may alternatively wish to re-examine the nominal

regression problem $\min_{\boldsymbol{\beta}} g(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})$ and instead attempt to solve this taking into account adversarial noise in the data matrix $\mathbf{X}$. As in [58, 101, 102, 19, 162], this approach may take the form

$$\min_{\boldsymbol{\beta}} \max_{\boldsymbol{\Delta} \in \mathcal{U}} g(\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}), \tag{3.1}$$

where the set $\mathcal{U} \subseteq \mathbb{R}^{m \times n}$ characterizes the user's belief about uncertainty on the data matrix $\mathbf{X}$. This set $\mathcal{U}$ is known in the language of robust optimization [19, 25] as an uncertainty set and the inner maximization problem $\max_{\boldsymbol{\Delta} \in \mathcal{U}} g(\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta})$ takes into account the worst-case error (measured via g) over $\mathcal{U}$. We call such a procedure *robustification* because it attempts to immunize or robustify the regression problem from structural uncertainty in the data. Such an adversarial or "worst-case" procedure is one of the key tenets of the area of robust optimization [19, 25].

As noted in the introduction, the adversarial perspective offers several attractive features. Let us first focus on settings when robustification coincides with a regularization problem. In such a case, the robustification identifies the adversarial perturbations the model is protected against, which can in turn provide additional insight into the behavior of different regularizers. Further, technical machinery developed for the construction of data-driven uncertainty sets in robust optimization [29, 160] enables the potential for a principled framework for the design of regularization schemes, in turn addressing a complex modeling decision encountered in practice.

Moreover, the adversarial approach is of interest in its own right, even if robustification does not correspond directly to a regularization problem. This is evidenced in part by the burgeoning success of generative adversarial networks and other methodologies in deep learning [70, 71, 137]. Further, the worst-case approach often leads to a more straightforward analysis of properties of estimators [162] as well as algorithms for finding estimators [18].

Let us now return to the robustification problem. A natural choice of an uncertainty set which gives rise to interpretability is the set $\mathcal{U} = \{\boldsymbol{\Delta} \in \mathbb{R}^{m \times n} : \|\boldsymbol{\Delta}\| \leq \lambda\}$, where $\|\cdot\|$ is some matrix norm and $\lambda > 0$. One can then write $\max_{\boldsymbol{\Delta} \in \mathcal{U}} g(\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta})$

as

$$\begin{aligned} \max_{\widetilde{\mathbf{X}}} \quad & g(\mathbf{y} - \widetilde{\mathbf{X}}\boldsymbol{\beta}) \\ \text{s. t.} \quad & \|\mathbf{X} - \widetilde{\mathbf{X}}\| \leq \lambda, \end{aligned}$$

or the worst case error taken over all $\widetilde{\mathbf{X}}$ sufficiently close to the data matrix $\mathbf{X}$. In what follows, if $\|\cdot\|$ is a norm or seminorm, then we let $\mathcal{U}_{\|\cdot\|}$ denote the ball of radius λ in $\|\cdot\|$:

$$\mathcal{U}_{\|\cdot\|} = \{\boldsymbol{\Delta} : \|\boldsymbol{\Delta}\| \leq \lambda\}.$$

For example, $\mathcal{U}_{F_p}$, $\mathcal{U}_{\sigma_p}$, and $\mathcal{U}_{(h,g)}$ denote uncertainty sets under the norms F_p, σ_p, and (h, g), respectively. We assume $\lambda > 0$ fixed for the remainder of the chapter.

We briefly mention addressing uncertainty in $\mathbf{y}$. Suppose that we have a set $\mathcal{V} \subseteq \mathbb{R}^m$ which captures some belief about the uncertainty in $\mathbf{y}$. If again we have an uncertainty set $\mathcal{U} \subseteq \mathbb{R}^{m \times n}$, we may attempt to solve a problem of the form

$$\min_{\boldsymbol{\beta}} \max_{\substack{\boldsymbol{\delta} \in \mathcal{V} \\ \boldsymbol{\Delta} \in \mathcal{U}}} g(\mathbf{y} + \boldsymbol{\delta} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}).$$

We can instead work with a new loss function $\bar{g}$ defined as

$$\bar{g}(\mathbf{v}) := \max_{\boldsymbol{\delta} \in \mathcal{V}} g(\mathbf{v} + \boldsymbol{\delta}).$$

If g is convex, then so is $\bar{g}$. In this way, we can work with the problem in the form

$$\min_{\boldsymbol{\beta}} \max_{\boldsymbol{\Delta} \in \mathcal{U}} \bar{g}(\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}),$$

where there is only uncertainty in $\mathbf{X}$. Throughout the remainder of this chapter we will only consider such uncertainty.

Relation to robust statistics: As noted in [19], the connection between robust optimization and robust statistics is not entirely clear. We will return in more depth to the relationship in Chapter 4.

3.2.3 Equivalence of robustification and regularization

A natural question is when do the procedures of regularization and robustification coincide. This problem was first studied in [58] in the context of uncertain least squares problems and has been extended to more general settings in [162, 47] and most comprehensively in [19]. In this subsection, we present settings in which robustification is equivalent to regularization. When such an equivalence holds, tools from robust optimization can be used to analyze properties of the regularization problem (*cf.* [162, 47]).

We begin with a general result on robustification under induced seminorm uncertainty sets.

Theorem 7. *If $g : \mathbb{R}^m \to \mathbb{R}$ is a seminorm which is not identically zero and $h : \mathbb{R}^n \to \mathbb{R}$ is a norm, then for any $\mathbf{z} \in \mathbb{R}^m$ and $\boldsymbol{\beta} \in \mathbb{R}^n$*

$$\max_{\boldsymbol{\Delta} \in \mathcal{U}_{(h,g)}} g(\mathbf{z} + \boldsymbol{\Delta}\boldsymbol{\beta}) = g(\mathbf{z}) + \lambda h(\boldsymbol{\beta}),$$

where $\mathcal{U}_{(h,g)} = \{\boldsymbol{\Delta} : \|\boldsymbol{\Delta}\|_{(h,g)} \leq \lambda\}$.

Proof. From the triangle inequality $g(\mathbf{z} + \boldsymbol{\Delta}\boldsymbol{\beta}) \leq g(\mathbf{z}) + g(\boldsymbol{\Delta}\boldsymbol{\beta}) \leq g(\mathbf{z}) + \lambda h(\boldsymbol{\beta})$ for any $\boldsymbol{\Delta} \in \mathcal{U} := \mathcal{U}_{(h,g)}$. We next show that there exists some $\boldsymbol{\Delta} \in \mathcal{U}$ so that $g(\mathbf{z} + \boldsymbol{\Delta}\boldsymbol{\beta}) = g(\mathbf{z}) + \lambda h(\boldsymbol{\beta})$. Let $\mathbf{v} \in \mathbb{R}^n$ so that $\mathbf{v} \in \operatorname{argmax}_{h^*(\mathbf{v})=1} \mathbf{v}'\boldsymbol{\beta}$, where h^* is the dual norm of h. Note in particular that $\mathbf{v}'\boldsymbol{\beta} = h(\boldsymbol{\beta})$ by the definition of the dual norm h^*. For now suppose that $g(\mathbf{z}) \neq 0$. Define the rank one matrix $\widehat{\boldsymbol{\Delta}} = \frac{\lambda}{g(\mathbf{z})}\mathbf{z}\mathbf{v}'$. Observe that

$$g(\mathbf{z} + \widehat{\boldsymbol{\Delta}}\boldsymbol{\beta}) = g\left(\mathbf{z} + \frac{\lambda h(\boldsymbol{\beta})}{g(\mathbf{z})}\mathbf{z}\right) = \frac{g(\mathbf{z}) + \lambda h(\boldsymbol{\beta})}{g(\mathbf{z})} g(\mathbf{z}) = g(\mathbf{z}) + \lambda h(\boldsymbol{\beta}).$$

We next show that $\widehat{\boldsymbol{\Delta}} \in \mathcal{U}$. Observe that for any $\mathbf{x} \in \mathbb{R}^n$ that

$$g(\widehat{\boldsymbol{\Delta}}\mathbf{x}) = g\left(\frac{\lambda \mathbf{v}'\mathbf{x}}{g(\mathbf{z})}\mathbf{z}\right) = \lambda |\mathbf{v}'\mathbf{x}| \leq \lambda h(\mathbf{x}) h^*(\mathbf{v}) = \lambda h(\mathbf{x}),$$

where the final inequality follows by definition of the dual norm. Hence $\widehat{\boldsymbol{\Delta}} \in \mathcal{U}$, as

desired.

We now consider the case when $g(\mathbf{z}) = 0$. Let $\mathbf{u} \in \mathbb{R}^m$ so that $g(\mathbf{u}) = 1$ (because g is not identically zero there exists some $\mathbf{u}$ so that $g(\mathbf{u}) > 0$, and so by homogeneity of g we can take $\mathbf{u}$ so that $g(\mathbf{u}) = 1$). Let $\mathbf{v}$ be as before. Now define $\widehat{\boldsymbol{\Delta}} = \lambda \mathbf{u}\mathbf{v}'$. We observe that

$$g(\mathbf{z} + \widehat{\boldsymbol{\Delta}}\boldsymbol{\beta}) = g(\mathbf{z} + \lambda \mathbf{u}\mathbf{v}'\boldsymbol{\beta}) \leq g(\mathbf{z}) + \lambda |\mathbf{v}'\boldsymbol{\beta}| g(\mathbf{u}) = \lambda h(\boldsymbol{\beta}).$$

Now, by the reverse triangle inequality,

$$g(\mathbf{z} + \widehat{\boldsymbol{\Delta}}\boldsymbol{\beta}) \geq g(\widehat{\boldsymbol{\Delta}}\boldsymbol{\beta}) - g(\mathbf{z}) = g(\widehat{\boldsymbol{\Delta}}\boldsymbol{\beta}) = \lambda h(\boldsymbol{\beta}),$$

and therefore $g(\mathbf{z} + \widehat{\boldsymbol{\Delta}}\boldsymbol{\beta}) = \lambda h(\boldsymbol{\beta}) = g(\mathbf{z}) + \lambda h(\boldsymbol{\beta})$. The proof that $\widehat{\boldsymbol{\Delta}} \in \mathcal{U}$ is identical to the case when $g(\mathbf{z}) \neq 0$. This completes the proof. □

This result implies as a corollary known results on the connection between robustification and regularization as found in [162, 19, 47] and references therein.

Corollary 1 ([162, 19, 47]). *If $p, q \in [1, \infty]$ then*

$$\min_{\boldsymbol{\beta}} \max_{\boldsymbol{\Delta} \in \mathcal{U}_{(q,p)}} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_p = \min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_p + \lambda \|\boldsymbol{\beta}\|_q.$$

In particular, for $p = q = 2$ we recover regularized least squares as a robustification; likewise, for $p = 2$ and $q = 1$ we recover the Lasso.[2]

Theorem 8 ([162, 19, 47]). *One has the following for any $p, q \in [1, \infty]$:*

$$\min_{\boldsymbol{\beta}} \max_{\boldsymbol{\Delta} \in \mathcal{U}_{F_p}} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_p = \min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_p + \lambda \|\boldsymbol{\beta}\|_{p^*},$$

[2]Strictly speaking, we recover *equivalent* problems to regularized least squares and Lasso, respectively. We take the usual convention and overlook this technicality (see [19] for a discussion). For completeness, we note that one can work directly with the true ℓ_2^2 loss function, although at the cost of requiring more complicated uncertainty sets to recover equivalence results.

where p^* is the conjugate of p. Similarly,

$$\min_{\boldsymbol{\beta}} \max_{\boldsymbol{\Delta} \in \mathcal{U}_{\sigma_q}} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_2 = \min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2 + \lambda \|\boldsymbol{\beta}\|_2.$$

Observe that regularized least squares arises again under all uncertainty sets defined by the spectral norms σ_q when the loss function is $g = \ell_2$. Now we continue with a remark on how Lasso arises through regularization. See [162] for comprehensive work on the robustness and sparsity implications of Lasso as interpreted through such a robustification considered here.

Remark 1. *As per Corollary 1 it is known that Lasso arises as uncertain ℓ_2 regression with uncertainty set $\mathcal{U} := \mathcal{U}_{(1,2)}$ [162]. As with Theorem 7, one might argue that the ℓ_1 penalizer arises as an artifact of the model of uncertainty. We remark that one can derive the set $\mathcal{U}$ as an induced uncertainty set defined using the "true" nonconvex penalty ℓ_0, where $\|\boldsymbol{\beta}\|_0 := |\{i : \beta_i \neq 0\}|$. To be precise, for any $p \in [1, \infty]$ and for $\Gamma = \{\boldsymbol{\beta} \in \mathbb{R}^n : \|\boldsymbol{\beta}\|_p \leq 1\}$ we claim that*

$$\mathcal{U}' := \left\{ \boldsymbol{\Delta} : \max_{\boldsymbol{\beta} \in \Gamma} \frac{\|\boldsymbol{\Delta}\boldsymbol{\beta}\|_2}{\|\boldsymbol{\beta}\|_0} \leq \lambda \right\}$$

satisfies $\mathcal{U} = \mathcal{U}'$. This is summarized, with an additional representation $\mathcal{U}''$ as used in [162], in the following proposition.

Proposition 2. *If $\mathcal{U} = \mathcal{U}_{(1,2)}$, $\mathcal{U}' = \{\boldsymbol{\Delta} : \|\boldsymbol{\Delta}\boldsymbol{\beta}\|_2 \leq \lambda \|\boldsymbol{\beta}\|_0 \ \forall \|\boldsymbol{\beta}\|_p \leq 1\}$ for an arbitrary $p \in [1, \infty]$, and $\mathcal{U}'' = \{\boldsymbol{\Delta} : \|\boldsymbol{\Delta}_i\|_2 \leq \lambda \ \forall i\}$, where $\boldsymbol{\Delta}_i$ is the ith column of $\boldsymbol{\Delta}$, then $\mathcal{U} = \mathcal{U}' = \mathcal{U}''$.*

Proof. We first show that $\mathcal{U} = \mathcal{U}'$. Because $\|\boldsymbol{\beta}\|_1 \leq \|\boldsymbol{\beta}\|_0$ for all $\boldsymbol{\beta} \in \mathbb{R}^n$ with $\|\boldsymbol{\beta}\|_p \leq 1$, we have that $\mathcal{U} \subseteq \mathcal{U}'$. Now suppose that $\boldsymbol{\Delta} \in \mathcal{U}'$. Then for any $\boldsymbol{\beta} \in \mathbb{R}^n$, we have that

$$\|\boldsymbol{\Delta}\boldsymbol{\beta}\|_2 = \left\| \sum_i \beta_i \boldsymbol{\Delta} \mathbf{e}_i \right\|_2 \leq \sum_i |\beta_i| \, \|\boldsymbol{\Delta}\mathbf{e}_i\|_2 \leq \sum_i |\beta_i| \lambda = \lambda \|\boldsymbol{\beta}\|_1,$$

where $\{\mathbf{e}_i\}_{i=1}^n$ is the standard orthonormal basis for $\mathbb{R}^n$. Hence, $\boldsymbol{\Delta} \in \mathcal{U}$ and therefore

$\mathcal{U}' \subseteq \mathcal{U}$. Combining with the previous direction gives $\mathcal{U} = \mathcal{U}'$.

We now prove that $\mathcal{U} = \mathcal{U}''$. That $\mathcal{U}'' \subseteq \mathcal{U}$ is essentially obvious; $\mathcal{U} \subseteq \mathcal{U}''$ follows by considering $\boldsymbol{\beta} \in \{\mathbf{e}_i\}_{i=1}^n$. This completes the proof. □

This proposition implies that ℓ_1 arises from the robustification setting without directly appealing to standard convexity arguments for why ℓ_1 should be used to replace ℓ_0 (which use the fact that ℓ_1 is the convex envelope of ℓ_0 on $[-1,1]^n$, see e.g. [38]).

In light of the above discussion, it is not difficult to show that other Lasso-like methods can also be expressed as an adversarial robustification, supporting the flexibility and versatility of such an approach. One such example is the elastic net [167, 53, 115], a hybridized version of ridge regression and the Lasso. An equivalent representation of the elastic net is as follows:

$$\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2 + \lambda\|\boldsymbol{\beta}\|_1 + \mu\|\boldsymbol{\beta}\|_2.$$

As per Theorem 8, this can be written exactly as

$$\min_{\boldsymbol{\beta}} \max_{\substack{\boldsymbol{\Delta},\boldsymbol{\Delta}': \\ \|\boldsymbol{\Delta}\|_{F_\infty} \le \lambda \\ \|\boldsymbol{\Delta}'\|_{F_2} \le \mu}} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta} + \boldsymbol{\Delta}')\boldsymbol{\beta}\|_2.$$

Under this interpretation, we see that λ and μ directly control the tradeoff between two different types of perturbations: "feature-wise" perturbations $\boldsymbol{\Delta}$ (controlled via λ and the F_∞ norm) and "global" perturbations $\boldsymbol{\Delta}'$ (controlled via μ and the F_2 norm).

We conclude this subsection with another example of when robustification is equivalent to regularization for the case of LAD (ℓ_1) and maximum absolute deviation (ℓ_∞) regression under row-wise uncertainty.

Theorem 9 ([162]). *Fix $q \in [1,\infty]$ and let $\mathcal{U} = \{\boldsymbol{\Delta} : \|\boldsymbol{\delta}_i\|_q \le \lambda \ \forall i\}$, where $\boldsymbol{\delta}_i$ is the ith row of $\boldsymbol{\Delta} \in \mathbb{R}^{m\times n}$. Then*

$$\min_{\boldsymbol{\beta}} \max_{\boldsymbol{\Delta}\in\mathcal{U}} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_1 = \min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_1 + m\lambda\|\boldsymbol{\beta}\|_{q^*}$$

and

$$\min_{\boldsymbol{\beta}} \max_{\boldsymbol{\Delta} \in \mathcal{U}} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_\infty = \min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_\infty + \lambda \|\boldsymbol{\beta}\|_{q^*}.$$

For completeness, we note that the uncertainty set $\mathcal{U} = \{\boldsymbol{\Delta} : \|\boldsymbol{\delta}_i\|_q \leq \lambda \; \forall i\}$ considered in Theorem 9 is actually an induced uncertainty set, namely, $\mathcal{U} = \mathcal{U}_{(q^*,\infty)}$.

3.2.4 Non-equivalence of robustification and regularization

In contrast to previous work studying robustification for regression, which primarily addresses tractability of solving the new uncertain problem [19] or the implications for Lasso [162], we instead focus our attention on characterization of the equivalence between robustification and regularization. We begin with a regularization upper bound on robustification problems.

Proposition 3. *Let $\mathcal{U} \subseteq \mathbb{R}^{m \times n}$ be any non-empty, compact set and $g : \mathbb{R}^m \to \mathbb{R}$ a seminorm. Then there exists some seminorm $\overline{h} : \mathbb{R}^n \to \mathbb{R}$ so that for any $\mathbf{z} \in \mathbb{R}^m$, $\boldsymbol{\beta} \in \mathbb{R}^n$,*

$$\max_{\boldsymbol{\Delta} \in \mathcal{U}} g(\mathbf{z} + \boldsymbol{\Delta}\boldsymbol{\beta}) \leq g(\mathbf{z}) + \overline{h}(\boldsymbol{\beta}),$$

with equality when $\mathbf{z} = \mathbf{0}$.

Proof. Let $\overline{h} : \mathbb{R}^n \to \mathbb{R}$ be defined as

$$\overline{h}(\boldsymbol{\beta}) := \max_{\boldsymbol{\Delta} \in \mathcal{U}} g(\boldsymbol{\Delta}\boldsymbol{\beta}).$$

To show that $\overline{h}$ is a seminorm we must show it satisfies absolute homogeneity and the triangle inequality. For any $\boldsymbol{\beta} \in \mathbb{R}^n$ and $\alpha \in \mathbb{R}$,

$$\overline{h}(\alpha\boldsymbol{\beta}) = \max_{\boldsymbol{\Delta} \in \mathcal{U}} g(\boldsymbol{\Delta}(\alpha\boldsymbol{\beta})) = \max_{\boldsymbol{\Delta} \in \mathcal{U}} |\alpha| g(\boldsymbol{\Delta}\boldsymbol{\beta}) = |\alpha| \left(\max_{\boldsymbol{\Delta} \in \mathcal{U}} g(\boldsymbol{\Delta}\boldsymbol{\beta}) \right) = |\alpha| \overline{h}(\boldsymbol{\beta}),$$

so absolute homogeneity is satisfied. Similarly, if $\boldsymbol{\beta}, \boldsymbol{\gamma} \in \mathbb{R}^n$,

$$\begin{aligned}\overline{h}(\boldsymbol{\beta}+\boldsymbol{\gamma}) = \max_{\boldsymbol{\Delta}\in\mathcal{U}} g(\boldsymbol{\Delta}(\boldsymbol{\beta}+\boldsymbol{\gamma})) &\leq \max_{\boldsymbol{\Delta}\in\mathcal{U}} \left[g(\boldsymbol{\Delta}\boldsymbol{\beta}) + g(\boldsymbol{\Delta}\boldsymbol{\gamma})\right] \\ &\leq \left(\max_{\boldsymbol{\Delta}\in\mathcal{U}} g(\boldsymbol{\Delta}\boldsymbol{\beta})\right) + \left(\max_{\boldsymbol{\Delta}\in\mathcal{U}} g(\boldsymbol{\Delta}\boldsymbol{\gamma})\right),\end{aligned}$$

and hence the triangle inequality is satisfied. Therefore, $\overline{h}$ is a seminorm which satisfies the desired properties, completing the proof. □

When equality is attained for all pairs $(\mathbf{z}, \boldsymbol{\beta}) \in \mathbb{R}^m \times \mathbb{R}^n$, we are in the regime of the previous subsection, and we say that robustification under $\mathcal{U}$ is equivalent to regularization under $\overline{h}$. We now discuss a variety of explicit settings in which regularization only provides upper and lower bounds to the true robustified problem.

Fix $p, q \in [1, \infty]$. Consider the robust ℓ_p regression problem

$$\min_{\boldsymbol{\beta}} \max_{\boldsymbol{\Delta}\in\mathcal{U}_{F_q}} \|\mathbf{y} - (\mathbf{X}+\boldsymbol{\Delta})\boldsymbol{\beta}\|_p,$$

where $\mathcal{U}_{F_q} = \{\boldsymbol{\Delta} \in \mathbb{R}^{m\times n} : \|\boldsymbol{\Delta}\|_{F_q} \leq \lambda\}$. In the case when $p = q$ we saw earlier (Theorem 8) that one exactly recovers ℓ_p regression with an ℓ_{p^*} penalty:

$$\min_{\boldsymbol{\beta}} \max_{\boldsymbol{\Delta}\in\mathcal{U}_{F_p}} \|\mathbf{y} - (\mathbf{X}+\boldsymbol{\Delta})\boldsymbol{\beta}\|_p = \min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_p + \lambda\|\boldsymbol{\beta}\|_{p^*}.$$

Let us now consider the case when $p \neq q$. We claim that regularization (with $\overline{h}$) is no longer equivalent to robustification (with $\mathcal{U}_{F_q}$) unless $p \in \{1, \infty\}$. Applying Proposition 3, one has for any $\mathbf{z} \in \mathbb{R}^m$ that

$$\max_{\boldsymbol{\Delta}\in\mathcal{U}_{F_q}} \|\mathbf{z} + \boldsymbol{\Delta}\boldsymbol{\beta}\|_p \leq \|\mathbf{z}\|_p + \overline{h}(\boldsymbol{\beta}),$$

where $\overline{h} = \max_{\boldsymbol{\Delta}\in\mathcal{U}_{F_q}} \|\boldsymbol{\Delta}\boldsymbol{\beta}\|_p$ is a norm (when $p = q$, this is precisely the ℓ_{p^*} norm, multiplied by λ). Here we can compute $\overline{h}$. To do this we first define a discrepancy function as follows:

Definition 2. *For $a, b \in [1, \infty]$ define the discrepancy function $\delta_m(a, b)$ as*

$$\delta_m(a, b) := \max\{\|\mathbf{u}\|_a : \mathbf{u} \in \mathbb{R}^m,\ \|\mathbf{u}\|_b = 1\}.$$

This discrepancy function is computable and well-known (see e.g. [80]):

$$\delta_m(a, b) = \begin{cases} m^{1/a - 1/b}, & \text{if } a \le b \\ 1, & \text{if } a > b. \end{cases}$$

It satisfies $1 \le \delta_m(a, b) \le m$ and $\delta_m(a, b)$ is continuous in a and b. One has that $\delta_m(a, b) = \delta_m(b, a) = 1$ if and only if $a = b$ (so long as $m \ge 2$). Using this, we now proceed with the theorem. The proof applies basic tools from real analysis and is contained in Appendix B.

Theorem 10. *(a) For any $\mathbf{z} \in \mathbb{R}^m$ and $\boldsymbol{\beta} \in \mathbb{R}^n$,*

$$\max_{\boldsymbol{\Delta} \in \mathcal{U}_{F_q}} \|\mathbf{z} + \boldsymbol{\Delta}\boldsymbol{\beta}\|_p \le \|\mathbf{z}\|_p + \lambda \delta_m(p, q) \|\boldsymbol{\beta}\|_{q^*}. \tag{3.2}$$

(b) When $p \in \{1, \infty\}$, there is equality in (3.2) for all $(\mathbf{z}, \boldsymbol{\beta})$.

(c) When $p \in (1, \infty)$ and $p \ne q$, for any $\boldsymbol{\beta} \ne \mathbf{0}$ the set of $\mathbf{z} \in \mathbb{R}^m$ for which the inequality (3.2) holds at equality is a finite union of one-dimensional subspaces (so long as $m \ge 2$). Hence, for any $\boldsymbol{\beta} \ne \mathbf{0}$ the inequality in (3.2) is strict for almost all $\mathbf{z}$.

(d) For $p \in (1, \infty)$, one has for all $\mathbf{z} \in \mathbb{R}^m$ and $\boldsymbol{\beta} \in \mathbb{R}^n$ that

$$\|\mathbf{z}\|_p + \frac{\lambda}{\delta_m(q, p)} \|\boldsymbol{\beta}\|_{q^*} \le \max_{\boldsymbol{\Delta} \in \mathcal{U}_{F_q}} \|\mathbf{z} + \boldsymbol{\Delta}\boldsymbol{\beta}\|_p. \tag{3.3}$$

(e) For $p \in (1, \infty)$, the lower bound in (3.3) is best possible in the sense that the gap can be arbitrarily small, i.e., for any $\boldsymbol{\beta} \in \mathbb{R}^n$,

$$\inf_{\mathbf{z}} \left(\max_{\boldsymbol{\Delta} \in \mathcal{U}_{F_q}} \|\mathbf{z} + \boldsymbol{\Delta}\boldsymbol{\beta}\|_p - \|\mathbf{z}\|_p - \frac{\lambda}{\delta_m(q, p)} \|\boldsymbol{\beta}\|_{q^*} \right) = 0.$$

Theorem 10 characterizes precisely when robustification under $\mathcal{U}_{F_q}$ is equivalent to regularization for the case of ℓ_p regression. In particular, when $p \neq q$ and $p \in (1, \infty)$, the two are *not* equivalent, and one only has that

$$\begin{aligned}\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_p + \frac{\lambda}{\delta_m(q,p)}\|\boldsymbol{\beta}\|_{q^*} &\leq \min_{\boldsymbol{\beta}} \max_{\boldsymbol{\Delta} \in \mathcal{U}_{F_q}} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_p \\ &\leq \min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_p + \lambda\delta_m(p,q)\|\boldsymbol{\beta}\|_{q^*}.\end{aligned}$$

Further, we have shown that these upper and lower bounds are the *best possible* (Theorem 10, parts (c) and (e)). While ℓ_p regression with uncertainty set $\mathcal{U}_{F_q}$ for $p \neq q$ and $p \in (1, \infty)$ still has both upper and lower bounds which correspond to regularization (with different regularization parameters $\lambda' \in [\lambda/\delta_m(q,p), \lambda\delta_m(p,q)]$), we emphasize that in this case there is no longer the direct connection between the parameter garnering the magnitude of uncertainty (λ) and the parameter for regularization (λ').

Example 1. *As a concrete example, consider the implications of Theorem 10 when $p = 2$ and $q = \infty$. We have that*

$$\begin{aligned}\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2 + \lambda\|\boldsymbol{\beta}\|_1 &\leq \min_{\boldsymbol{\beta}} \max_{\boldsymbol{\Delta} \in \mathcal{U}_{F_\infty}} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_p \\ &\leq \min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2 + \sqrt{m}\lambda\|\boldsymbol{\beta}\|_1.\end{aligned}$$

In this case, robustification is not *equivalent to regularization. In particular, in the regime where there are many data points (i.e. m is large), the gap appearing between the different problems can be quite large.*

Before proceeding with other choices of uncertainty sets, it is important to make a further distinction about the general non-equivalence of robustification and regularization as presented in Theorem 10. In particular, it is simple to construct examples which imply the following strong existential result (see Appendix B.2):

Theorem 11. *In a setting when robustification and regularization are not equivalent,*

it is possible for the two problems to have different optimal solutions. In particular,

$$\boldsymbol{\beta}^* \in \operatorname*{argmin}_{\boldsymbol{\beta}} \max_{\boldsymbol{\Delta} \in \mathcal{U}} g(\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta})$$

is not necessarily a solution of

$$\min_{\boldsymbol{\beta}} g(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) + \widetilde{\lambda} h(\boldsymbol{\beta})$$

for any $\widetilde{\lambda} > 0$, and vice versa.

As a result, when robustification and regularization do not coincide, they can induce structurally distinct solutions. In particular, the regularization path (as $\widetilde{\lambda} \in (0, \infty)$ varies) and the robustification path (as the radius $\lambda \in (0, \infty)$ of $\mathcal{U}$ varies) can be different.

We now proceed to analyze another setting in which robustification is not equivalent to regularization. The setting, in line with Theorem 8, is ℓ_p regression under spectral uncertainty sets $\mathcal{U}_{\sigma_q}$. As per Theorem 8, one has that

$$\min_{\boldsymbol{\beta}} \max_{\boldsymbol{\Delta} \in \mathcal{U}_{\sigma_q}} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_2 = \min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2 + \lambda \|\boldsymbol{\beta}\|_2$$

for any $q \in [1, \infty]$. This result on the "universality" of RLS under a variety of uncertainty sets relies on the fact that the ℓ_2 norm underlies spectral decompositions; namely, one can write any matrix $\mathbf{X}$ as $\sum_i \mu_i \mathbf{u}_i \mathbf{v}_i'$, where $\{\mu_i\}_i$ are the singular values of $\mathbf{X}$, $\{\mathbf{u}_i\}_i$ and $\{\mathbf{v}_i\}_i$ are the left and right singular vectors of $\mathbf{X}$, respectively, and $\|\mathbf{u}_i\|_2 = \|\mathbf{v}_i\|_2 = 1$ for all i.

A natural question is what happens when the loss function ℓ_2, a modeling choice, is replaced by ℓ_p, where $p \in [1, \infty]$. We claim that for $p \notin \{1, 2, \infty\}$, robustification under $\mathcal{U}_{\sigma_q}$ is no longer equivalent to regularization. In light of Theorem 10, this is not difficult to prove. We find that the choice of $q \in [1, \infty]$, as before, is inconsequential. We summarize this in the following proposition:

Proposition 4. *For any* $\mathbf{z} \in \mathbb{R}^m$ *and* $\boldsymbol{\beta} \in \mathbb{R}^n$,

$$\max_{\boldsymbol{\Delta} \in \mathcal{U}_{\sigma_q}} \|\mathbf{z} + \boldsymbol{\Delta}\boldsymbol{\beta}\|_p \leq \|\mathbf{z}\|_p + \lambda \delta_m(p,2)\|\boldsymbol{\beta}\|_2. \tag{3.4}$$

In particular, if $p \in \{1, 2, \infty\}$, *there is equality in* (3.4) *for all* $(\mathbf{z}, \boldsymbol{\beta})$. *If* $p \notin \{1, 2, \infty\}$, *then for any* $\boldsymbol{\beta} \neq \mathbf{0}$ *the inequality in* (3.4) *is strict for almost all* $\mathbf{z}$ *(when* $m \geq 2$*). Further, for* $p \notin \{1, 2, \infty\}$ *one has the lower bound*

$$\|\mathbf{z}\|_p + \frac{\lambda}{\delta_m(2,p)}\|\boldsymbol{\beta}\|_2 \leq \max_{\boldsymbol{\Delta} \in \mathcal{U}_{\sigma_q}} \|\mathbf{z} + \boldsymbol{\Delta}\boldsymbol{\beta}\|_p,$$

whose gap is arbitrarily small for all $\boldsymbol{\beta}$.

Proof. This result is Theorem 10 in disguise. This follows by noting that

$$\max_{\boldsymbol{\Delta} \in \mathcal{U}_{\sigma_q}} \|\mathbf{z} + \boldsymbol{\Delta}\boldsymbol{\beta}\|_p = \max_{\boldsymbol{\Delta} \in \mathcal{U}_{F_2}} \|\mathbf{z} + \boldsymbol{\Delta}\boldsymbol{\beta}\|_p$$

and directly applying the preceding results. □

We now consider a third setting for ℓ_p regression, this time subject to uncertainty $\mathcal{U}_{(q,r)}$; this is a generalized version of the problems considered in Theorems 7 and 9. From Theorem 7 we know that if $p = r$, then

$$\min_{\boldsymbol{\beta}} \max_{\boldsymbol{\Delta} \in \mathcal{U}_{(q,p)}} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_p = \min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_p + \lambda\|\boldsymbol{\beta}\|_q.$$

Similarly, as per Theorem 9, when $r = \infty$ and $p \in \{1, \infty\}$,

$$\min_{\boldsymbol{\beta}} \max_{\boldsymbol{\Delta} \in \mathcal{U}_{(q,\infty)}} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_p = \min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_p + \lambda\delta_m(p,\infty)\|\boldsymbol{\beta}\|_q.$$

Given these results, it is natural to inquire what happens for more general choices of induced uncertainty set $\mathcal{U}_{(q,r)}$. As before with Theorem 10, we have a complete characterization of the equivalence of robustification and regularization for ℓ_p regression with uncertainty set $\mathcal{U}_{(q,r)}$:

Proposition 5. *For any* $\mathbf{z} \in \mathbb{R}^m$ *and* $\boldsymbol{\beta} \in \mathbb{R}^n$,

$$\max_{\boldsymbol{\Delta} \in \mathcal{U}_{(q,r)}} \|\mathbf{z} + \boldsymbol{\Delta}\boldsymbol{\beta}\|_p \leq \|\mathbf{z}\|_p + \lambda \delta_m(p, r) \|\boldsymbol{\beta}\|_q. \tag{3.5}$$

In particular, if $p \in \{1, r, \infty\}$, *there is equality in* (3.4) *for all* $(\mathbf{z}, \boldsymbol{\beta})$. *If* $p \in (1, \infty)$ *and* $p \neq r$, *then for any* $\boldsymbol{\beta} \neq \mathbf{0}$ *the inequality in* (3.5) *is strict for almost all* $\mathbf{z}$ *(when* $m \geq 2$*). Further, for* $p \in (1, \infty)$ *with* $p \neq r$ *one has the lower bound*

$$\|\mathbf{z}\|_p + \frac{\lambda}{\delta_m(r, p)} \|\boldsymbol{\beta}\|_q \leq \max_{\boldsymbol{\Delta} \in \mathcal{U}_{(q,r)}} \|\mathbf{z} + \boldsymbol{\Delta}\boldsymbol{\beta}\|_p,$$

whose gap is arbitrarily small for all $\boldsymbol{\beta}$.

Proof. The proof follows the argument given in the proof of Theorem 10. Here we simply note that now one uses the fact that

$$\max_{\boldsymbol{\Delta} \in \mathcal{U}_{(q,r)}} \|\mathbf{z} + \boldsymbol{\Delta}\boldsymbol{\beta}\|_p = \max_{\|\mathbf{u}\|_r \leq \lambda \|\boldsymbol{\beta}\|_q} \|\mathbf{z} + \mathbf{u}\|_p.$$

□

We summarize all of the results on linear regression in Table 3.2.

Loss function	Uncertainty set $\mathcal{U}$	$\overline{h}(\boldsymbol{\beta})$	Equivalence if and only if
seminorm g	$\mathcal{U}_{(h,g)}$ (h norm)	$\lambda h(\boldsymbol{\beta})$	always
ℓ_p	$\mathcal{U}_{\sigma_q}$	$\lambda \delta_m(p, 2) \|\boldsymbol{\beta}\|_2$	$p \in \{1, 2, \infty\}$
ℓ_p	$\mathcal{U}_{F_q}$	$\lambda \delta_m(p, q) \|\boldsymbol{\beta}\|_{q^*}$	$p \in \{1, q, \infty\}$
ℓ_p	$\mathcal{U}_{(q,r)}$	$\lambda \delta_m(p, r) \|\boldsymbol{\beta}\|_q$	$p \in \{1, r, \infty\}$
ℓ_p	$\{\boldsymbol{\Delta} : \|\boldsymbol{\delta}_i\|_q \leq \lambda \ \forall i\}$	$\lambda m^{1/p} \|\boldsymbol{\beta}\|_{q^*}$	$p \in \{1, \infty\}$

Table 3.2: Summary of equivalencies for robustification with uncertainty set $\mathcal{U}$ and regularization with penalty $\overline{h}$, where $\overline{h}$ is as given in Proposition 3. Here by equivalence we mean that for all $\mathbf{z} \in \mathbb{R}^m$ and $\boldsymbol{\beta} \in \mathbb{R}^n$, $\max_{\boldsymbol{\Delta} \in \mathcal{U}} g(\mathbf{z} + \boldsymbol{\beta}) = g(\mathbf{z}) + \overline{h}(\boldsymbol{\beta})$, where g is the loss function, i.e., the upper bound $\overline{h}$ is also a lower bound. Here δ_m is as in Theorem 10. Throughout $p, q \in [1, \infty]$ and $m \geq 2$. Here $\boldsymbol{\delta}_i$ denotes the ith row of $\boldsymbol{\Delta}$.

3.3 On the equivalence of robustification and regularization in matrix estimation problems

A substantial body of problems at the core of modern developments in statistical estimation involves underlying matrix variables. Two prominent examples that we consider here are matrix completion and Principal Component Analysis (PCA). In both cases we show that a common choice of the regularization problem corresponds exactly to a robustification of the nominal problem subject to uncertainty. In doing so we expand the existing knowledge of robustification for vector regression to a novel and substantial domain. We begin by reviewing these two problem classes before introducing a simple model of uncertainty analogous to the vector model of uncertainty.

3.3.1 Problem classes

In matrix completion problems one is given data $Y_{ij} \in \mathbb{R}$ for $(i,j) \in E \subseteq \{1,\dots,m\} \times \{1,\dots,n\}$. One problem of interest is rank-constrained matrix completion

$$\begin{array}{ll} \min\limits_{\mathbf{X}} & \|\mathbf{Y}-\mathbf{X}\|_{P(F_2)} \\ \text{s.t.} & \operatorname{rank}(\mathbf{X}) \leq k, \end{array} \tag{3.6}$$

where $\|\cdot\|_{P(F_2)}$ denotes the projected 2−Frobenius seminorm, namely,

$$\|\mathbf{Z}\|_{P(F_2)} = \left(\sum_{(i,j)\in E} Z_{ij}^2 \right)^{1/2}.$$

Matrix completion problems appear in a wide variety of areas. One well-known application is in the Netflix challenge [142], where one wishes to predict user movie preferences based on a very limited subset of given user ratings. Here rank-constrained models are important in order to obtain parsimonious descriptions of user preferences in terms of a limited number of significant latent factors. The rank-constrained problem (3.6) is typically converted to a regularized form with rank replaced by the nuclear

norm σ_1 (the sum of singular values) to obtain the convex problem

$$\min_{\mathbf{X}} \|\mathbf{Y} - \mathbf{X}\|_{P(F_2)} + \lambda \|\mathbf{X}\|_{\sigma_1}.$$

In what follows we show that this regularized problem can be written as an uncertain version of a nominal problem $\min_{\mathbf{X}} \|\mathbf{Y} - \mathbf{X}\|_{P(F_2)}$.

Similarly to matrix completion, PCA typically takes the form

$$\begin{array}{ll} \min\limits_{\mathbf{X}} & \|\mathbf{Y} - \mathbf{X}\| \\ \text{s.t.} & \text{rank}(\mathbf{X}) \leq k, \end{array} \tag{3.7}$$

where $\|\cdot\|$ is either the usual Frobenius norm $F_2 = \sigma_2$ or the operator norm σ_∞, and $\mathbf{Y} \in \mathbb{R}^{m \times n}$. PCA arises naturally by assuming that $\mathbf{Y}$ is observed as some low-rank matrix $\mathbf{X}$ plus noise: $\mathbf{Y} = \mathbf{X} + \mathbf{E}$. The solution to (3.7) is well-known to be a truncated singular value decomposition which retains the k largest singular values [57]. PCA is popular for a variety of applications where dimension reduction is desired.

A variant of PCA known as robust PCA [46] operates under the assumption that some entries of $\mathbf{Y}$ may be grossly corrupted. Robust PCA assumes that $\mathbf{Y} = \mathbf{X} + \mathbf{E}$, where $\mathbf{X}$ is low rank and $\mathbf{E}$ is sparse (few nonzero entries). Under this model robust PCA takes the form

$$\min_{\mathbf{X}} \|\mathbf{Y} - \mathbf{X}\|_{F_1} + \lambda \|\mathbf{X}\|_{\sigma_1}. \tag{3.8}$$

Here again we can interpret $\|\mathbf{X}\|_{\sigma_1}$ as a surrogate penalty for rank. In the spirit of results from compressed sensing on exact ℓ_1 recovery, it is shown in [46] that (3.8) can exactly recover the true $\mathbf{X}_0$ and $\mathbf{E}_0$ assuming that the rank of $\mathbf{X}_0$ is small, $\mathbf{E}_0$ is sufficiently sparse, and the eigenvectors of $\mathbf{X}_0$ are well-behaved (see technical conditions contained therein). Below we derive explicit expressions for PCA subject to certain types of uncertainty; in doing so we show that robust PCA does not correspond to an adversarially robust version of $\min_{\mathbf{X}} \|\mathbf{Y} - \mathbf{X}\|_{\sigma_\infty}$ or $\min_{\mathbf{X}} \|\mathbf{Y} - \mathbf{X}\|_{F_2}$ for *any* model of additive linear uncertainty.

Finally let us note that the results we consider here on robust PCA are distinct

from considerations in the robust statistics community on robust approaches to PCA. For results and commentary on such methods, see [52, 83, 82].

3.3.2 Models of uncertainty

For these two problem classes we now detail a model of uncertainty. Our underlying problem is of the form $\min_{\mathbf{X}} \|\mathbf{Y} - \mathbf{X}\|$, where $\mathbf{Y}$ is given data (possibly with some unknown entries). As with the vector case, we do not concern ourselves with uncertainty in the observed $\mathbf{Y}$ because modeling uncertainty in $\mathbf{Y}$ simply leads to a different choice of loss function. To be precise, if $\mathcal{V} \subseteq \mathbb{R}^{m\times n}$ and g is convex loss function then

$$\bar{g}(\mathbf{Y} - \mathbf{X}) := \max_{\mathbf{\Delta}\in\mathcal{V}} g((\mathbf{Y} + \mathbf{\Delta}) - \mathbf{X})$$

is a new convex loss function $\bar{g}$ of $\mathbf{Y} - \mathbf{X}$.

As in the vector case we assume a linear model of uncertainty in the measurement of $\mathbf{X}$:

$$Y_{ij} = X_{ij} + \left(\sum_{\ell k} \Delta_{\ell k}^{(ij)} X_{\ell k}\right) + \epsilon_{ij},$$

where $\mathbf{\Delta}^{(ij)} \in \mathbb{R}^{m\times n}$; alternatively, in inner product notation, $Y_{ij} = X_{ij} + \langle \mathbf{\Delta}^{(ij)}, \mathbf{X}\rangle + \epsilon_{ij}$. This linear model is in direct analogy with the model for vector regression taken earlier; now $\boldsymbol{\beta}$ is replaced by $\mathbf{X}$, and again we consider linear perturbations of the unknown regression variable.

This linear model of uncertainty captures a variety of possible forms of uncertainty and accounts for possible interactions among different entries of the matrix $\mathbf{X}$. Note that in matrix notation, the nominal problem becomes, subject to linear uncertainty in $\mathbf{X}$,

$$\min_{\mathbf{X}} \max_{\mathbf{\Delta}\in\mathcal{U}} \|\mathbf{Y} - \mathbf{X} - \mathbf{\Delta}(\mathbf{X})\|,$$

where here $\mathcal{U}$ is some collection of linear maps and $\mathbf{\Delta} \in \mathcal{U}$ is defined as $[\mathbf{\Delta}(\mathbf{X})]_{ij} = \langle \mathbf{\Delta}^{(ij)}, \mathbf{X}\rangle$, where again $\mathbf{\Delta}^{(ij)} \in \mathbb{R}^{m\times n}$ (all linear maps can be written in such a form). Note here the direct analogy to the vector case, with the notation $\mathbf{\Delta}(\mathbf{X})$ chosen for simplicity. (For clarity, note that $\mathbf{\Delta}$ is not itself a matrix, although one could interpret

it as a matrix in $\mathbf{\Delta}^{mn\times mn}$, albeit at a notational cost; we avoid this here.)

We now outline some particular choices for uncertainty sets. As with the vector case, one natural set is an induced uncertainty set. Precisely, if $g, h : \mathbb{R}^{m\times n} \to \mathbb{R}$ are functions, then we define an induced uncertainty set

$$\mathcal{U}_{(h,g)} := \left\{\mathbf{\Delta} : \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n} \mid \mathbf{\Delta} \text{ linear}, \, g(\mathbf{\Delta}(\mathbf{X})) \leq \lambda h(\mathbf{X}) \; \forall \mathbf{X} \in \mathbb{R}^{m\times n}\right\}.$$

As before, when g and h are both norms, $\mathcal{U}_{(h,g)}$ is precisely a ball of radius λ in the induced norm

$$\|\mathbf{\Delta}\|_{(h,g)} = \max_{\mathbf{X}} \frac{g(\mathbf{\Delta}(\mathbf{X}))}{h(\mathbf{X})}.$$

There are also many other possible choices of uncertainty sets. These include the spectral uncertainty sets

$$\mathcal{U}_{\sigma_p} = \{\mathbf{\Delta} : \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n} | \mathbf{\Delta} \text{ linear}, \, \|\mathbf{\Delta}\|_{\sigma_p} \leq \lambda\},$$

where we interpret $\|\mathbf{\Delta}\|_{\sigma_p}$ as the σ_p norm of $\mathbf{\Delta}$ in any, and hence all, of its matrix representations. Other uncertainty sets are those such as $\mathcal{U} = \{\mathbf{\Delta} : \mathbf{\Delta}^{(ij)} \in \mathcal{U}^{(ij)}\}$, where $\mathcal{U}^{(ij)} \subseteq \mathbb{R}^{m\times n}$ are themselves uncertainty sets. These last two models we will not examine in depth here because they are often subsumed by the vector results (note that these two uncertainty sets do not truly involve the matrix structure of $\mathbf{X}$, and can therefore be "vectorized", reducing directly to vector results).

3.3.3 Basic results on equivalence

We now continue with some underlying theorems for our models of uncertainty. As a first step, we provide a proposition on the spectral uncertainty sets. As noted above, this result is exactly Theorem 8, and therefore we will not consider such uncertainty sets for the remainder of the chapter.

Proposition 6. *For any $q \in [1, \infty]$ and any $\mathbf{Y} \in \mathbb{R}^{m \times n}$,*

$$\min_{\mathbf{X}} \max_{\mathbf{\Delta} \in \mathcal{U}_{\sigma_q}} \|\mathbf{Y} - \mathbf{X} - \mathbf{\Delta}(\mathbf{X})\|_{F_2} = \min_{\mathbf{X}} \|\mathbf{Y} - \mathbf{X}\|_{F_2} + \lambda \|\mathbf{X}\|_{F_2}.$$

For what follows, we restrict our attention to induced uncertainty sets. We begin with an analogous result to Theorem 7. Throughout we always assume without loss of generality that if Y_{ij} is not known then $Y_{ij} = 0$ (i.e., we set it to some arbitrary value).

Theorem 12. *If $g : \mathbb{R}^{m \times n} \to \mathbb{R}$ is a seminorm which is not indentically zero and $h : \mathbb{R}^{m \times n} \to \mathbb{R}$ is a norm, then*

$$\min_{\mathbf{X}} \max_{\mathbf{\Delta} \in \mathcal{U}_{(h,g)}} g\left(\mathbf{Y} - \mathbf{X} - \mathbf{\Delta}(\mathbf{X})\right) = \min_{\mathbf{X}} g\left(\mathbf{Y} - \mathbf{X}\right) + \lambda h\left(\mathbf{X}\right).$$

This theorem leads to an immediate corollary:

Corollary 2. *For any norm $\|\cdot\| : \mathbb{R}^{m \times n} \to \mathbb{R}$ and any $p \in [1, \infty]$*

$$\min_{\mathbf{X}} \max_{\mathbf{\Delta} \in \mathcal{U}_{(\sigma_p, \|\cdot\|)}} \|\mathbf{Y} - \mathbf{X} - \mathbf{\Delta}(\mathbf{X})\| = \min_{\mathbf{X}} \|\mathbf{Y} - \mathbf{X}\| + \lambda \|\mathbf{X}\|_{\sigma_p}.$$

In the two subsections which follow we study the implications of Theorem 12 for matrix completion and PCA.

3.3.4 Robust matrix completion

We now proceed to apply Theorem 12 for the case of matrix completion. Note that the projected Frobenius "norm" $P(F_2)$ is a seminorm. Therefore, we arrive at the following corollary:

Corollary 3. *For any $p \in [1, \infty]$ one has that*

$$\min_{\mathbf{X}} \max_{\mathbf{\Delta} \in \mathcal{U}_{(\sigma_p, P(F_2))}} \|\mathbf{Y} - \mathbf{X} - \mathbf{\Delta}(\mathbf{X})\|_{P(F_2)} = \min_{\mathbf{X}} \|\mathbf{Y} - \mathbf{X}\|_{P(F_2)} + \lambda \|\mathbf{X}\|_{\sigma_p}.$$

In particular, for $p = 1$ one exactly recovers so-called nuclear norm penalized matrix completion:

$$\min_{\mathbf{X}} \|\mathbf{Y} - \mathbf{X}\|_{P(F_2)} + \lambda \|\mathbf{X}\|_{\sigma_1}.$$

It is not difficult to show by modifying the proof of Theorem 12 that even though $\mathcal{U}_{(\sigma_p, F_2)} \subsetneq \mathcal{U}_{(\sigma_p, P(F_2))}$, the following holds:

Proposition 7. *For any $p \in [1, \infty]$ one has that*

$$\min_{\mathbf{X}} \max_{\mathbf{\Delta} \in \mathcal{U}_{(\sigma_p, F_2)}} \|\mathbf{Y} - \mathbf{X} - \mathbf{\Delta}(\mathbf{X})\|_{P(F_2)} = \min_{\mathbf{X}} \|\mathbf{Y} - \mathbf{X}\|_{P(F_2)} + \lambda \|\mathbf{X}\|_{\sigma_p}.$$

In particular, for $p = 1$ one exactly recovers nuclear norm penalized matrix completion.

Let us briefly comment on the appearance of the nuclear norm in Corollary 3 and Proposition 7. In light of Remark 1, it is not surprising that such a penalty can be derived by working directly with the rank function (nuclear norm is the convex envelope of the rank function on the ball $\{\mathbf{X} : \|\mathbf{X}\|_{\sigma_\infty} \leq 1\}$, which is why the nuclear norm is typically used to replace rank [61, 126]). We detail this argument as before. For any $p \in [1, \infty]$ and $\Gamma = \{\mathbf{X} \in \mathbb{R}^{m \times n} : \|\mathbf{X}\|_{\sigma_p} \leq 1\}$, one can show that

$$\mathcal{U}_{(\sigma_1, P(F_2))} = \left\{ \mathbf{\Delta} \text{ linear} : \max_{\mathbf{X} \in \Gamma} \frac{\|\mathbf{\Delta}(\mathbf{X})\|_{P(F_2)}}{\operatorname{rank}(\mathbf{X})} \leq \lambda \right\}. \tag{3.9}$$

Therefore, similar to the vector case with an underlying ℓ_0 penalty which becomes a Lasso ℓ_1 penalty, rank leads to the nuclear norm from the robustification setting without directly invoking convexity.

3.3.5 Robust PCA

We now turn our attention to the implications of Theorem 12 for PCA. We begin by noting robust analogues of $\min_{\mathbf{X}} \|\mathbf{Y} - \mathbf{X}\|$ under the F_2 and σ_∞ norms. This is distinct from the considerations in [47] on robustness of PCA with respect to training and testing sets.

Corollary 4. *For any $p \in [1, \infty]$ one has that*

$$\min_{\mathbf{X}} \max_{\mathbf{\Delta} \in \mathcal{U}_{(\sigma_p, F_2)}} \|\mathbf{Y} - \mathbf{X} - \mathbf{\Delta}(\mathbf{X})\|_{F_2} = \min_{\mathbf{X}} \|\mathbf{Y} - \mathbf{X}\|_{F_2} + \lambda \|\mathbf{X}\|_{\sigma_p}$$

and

$$\min_{\mathbf{X}} \max_{\mathbf{\Delta} \in \mathcal{U}_{(\sigma_p, \sigma_\infty)}} \|\mathbf{Y} - \mathbf{X} - \mathbf{\Delta}(\mathbf{X})\|_{\sigma_\infty} = \min_{\mathbf{X}} \|\mathbf{Y} - \mathbf{X}\|_{\sigma_\infty} + \lambda \|\mathbf{X}\|_{\sigma_p}.$$

We continue by considering robust PCA as presented in [46]. Suppose that $\mathcal{U}$ is some collection of linear maps $\mathbf{\Delta} : \mathbb{R}^{m \times n} \to \mathbb{R}^{m \times n}$ and $\|\cdot\|$ is some norm so that for any $\mathbf{Y}, \mathbf{X} \in \mathbb{R}^{m \times n}$

$$\max_{\mathbf{\Delta} \in \mathcal{U}} \|\mathbf{Y} - \mathbf{X} - \mathbf{\Delta}(\mathbf{X})\| = \|\mathbf{Y} - \mathbf{X}\|_{F_1} + \lambda \|\mathbf{X}\|_{\sigma_1}.$$

It is easy to see that this implies $\|\cdot\| = \|\cdot\|_{F_1}$. These observations, combined with Theorem 12, imply the following:

Proposition 8. *The problem (3.8) can be written as an uncertain version of* $\min_{\mathbf{X}} \|\mathbf{Y} - \mathbf{X}\|$ *subject to additive, linear uncertainty in* $\mathbf{X}$ *if and only if* $\|\cdot\|$ *is the* 1*-Frobenius norm* F_1*. In particular, (3.8) does not arise as uncertain versions of PCA (using* F_2 *or* σ_∞*) under such a model of uncertainty.*

This result is not entirely surprising. This is because robust PCA attempts to solve, based on its model of $\mathbf{Y} = \mathbf{X} + \mathbf{E}$ where $\mathbf{X}$ is low-rank and $\mathbf{E}$ is sparse, a problem of the form

$$\min_{\mathbf{X}} \|\mathbf{Y} - \mathbf{X}\|_{F_0} + \lambda \operatorname{rank}(\mathbf{X}),$$

where $\|\mathbf{A}\|_{F_0}$ is the number of nonzero entries of $\mathbf{A}$. In the usual way, F_0 and rank are replaced with surrogates F_1 and σ_1, respectively. Hence, (3.8) appears as a convex, regularized form of the problem

$$\begin{aligned} \min_{\mathbf{X}} \quad & \|\mathbf{Y} - \mathbf{X}\|_{F_1} \\ \text{s. t.} \quad & \operatorname{rank}(\mathbf{X}) \leq k. \end{aligned}$$

Again, as with matrix completion, it is possible to show that (3.8) and uncertain

forms of PCA with a nuclear norm penalty (as appearing in Corollary 4) can be derived using the true choice of penalizer, rank, instead of imposing an *a priori* assumption of a nuclear norm penalty. We summarize this, without proof, as follows:

Proposition 9. *For any $p \in [1, \infty]$ and any norm $\|\cdot\|$,*

$$\min_{\mathbf{X}\in\Gamma} \max_{\mathbf{\Delta}\in\mathcal{U}_{\Gamma(\mathrm{rank},\|\cdot\|)}} \|\mathbf{Y} - \mathbf{X} - \mathbf{\Delta}(\mathbf{X})\| = \min_{\mathbf{X}\in\Gamma} \|\mathbf{Y} - \mathbf{X}\| + \lambda\|\mathbf{X}\|_{\sigma_1},$$

where $\Gamma = \{\mathbf{X} \in \mathbb{R}^{m\times n} : \|\mathbf{X}\|_{\sigma_p} \leq 1\}$ and

$$\mathcal{U}_{\Gamma(\mathrm{rank},\|\cdot\|)} = \left\{ \mathbf{\Delta} \text{ linear} : \max_{\mathbf{X}\in\Gamma} \frac{\|\mathbf{\Delta}(\mathbf{X})\|}{\mathrm{rank}(\mathbf{X})} \leq \lambda \right\}.$$

3.3.6 Non-equivalence of robustification and regularization

As with vector regression it is not always the case that robustification is equivalent to regularization in matrix estimation problems. For completeness we provide analogues here of the linear regression results. We begin by stating results which follow over with essentially identical proofs from the vector case; proofs are not included here. Then we characterize precisely when another plausible model of uncertainty leads to equivalence.

We begin with the analogue of Proposition 3.

Proposition 10. *Let $\mathcal{U} \subseteq \{$linear maps $\mathbf{\Delta} : \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}\}$ be any non-empty, compact set and $g : \mathbb{R}^{m\times n} \to \mathbb{R}$ a seminorm. Then there exists some seminorm $\overline{h} : \mathbb{R}^{m\times n} \to \mathbb{R}$ so that for any $\mathbf{Z}, \mathbf{X} \in \mathbb{R}^{m\times n}$,*

$$\max_{\mathbf{\Delta}\in\mathcal{U}} g(\mathbf{Z} + \mathbf{\Delta}(\mathbf{X})) \leq g(\mathbf{Z}) + \overline{h}(\mathbf{X}),$$

with equality when $\mathbf{Z} = \mathbf{0}$.

As before with Theorem 10 and Propositions 4 and 5, one can now compute $\overline{h}$ for a variety of problems.

Proposition 11. *For any* $\mathbf{Z}, \mathbf{X} \in \mathbb{R}^{m\times n}$,

$$\|\mathbf{Z}\|_{F_p} + \frac{\lambda}{\delta_{mn}(q,p)}\|\mathbf{X}\|_{F_{q^*}} \leq \max_{\mathbf{\Delta}\in\mathcal{U}_{F_q}} \|\mathbf{Z} + \mathbf{\Delta}(\mathbf{X})\|_{F_p} \tag{3.10}$$

$$\leq \|\mathbf{Z}\|_{F_p} + \lambda\delta_{mn}(p,q)\|\mathbf{X}\|_{F_{q^*}} \tag{3.11}$$

where $\|\mathbf{\Delta}\|_{F_q}$ *is interpreted as the* F_q *norm on the matrix representation of* $\mathbf{\Delta}$ *in the standard basis. In particular, if* $p \neq q$ *and* $p \in (1,\infty)$*, then for any* $\mathbf{X} \neq \mathbf{0}$ *the upper bound in* (3.11) *is strict for almost all* $\mathbf{Z}$ *(so long as* $mn \geq 2$*). Further, when* $p \neq q$ *and* $p \in (1,\infty)$*, the gap in the lower bound in* (3.10) *is arbitrarily small for all* $\mathbf{X}$*.*

Proposition 12. *For any* $\mathbf{Z}, \mathbf{X} \in \mathbb{R}^{m\times n}$,

$$\|\mathbf{Z}\|_{p} + \frac{\lambda}{\delta_{mn}(2,p)}\|\mathbf{X}\|_{F_2} \leq \max_{\mathbf{\Delta}\in\mathcal{U}_{\sigma_q}} \|\mathbf{Z} + \mathbf{\Delta}(\mathbf{X})\|_{F_p} \tag{3.12}$$

$$\leq \|\mathbf{Z}\|_{F_p} + \lambda\delta_{mn}(p,2)\|\mathbf{X}\|_{F_2}. \tag{3.13}$$

In particular, if $p \notin \{1,2,\infty\}$*, then for all* $\mathbf{X} \neq \mathbf{0}$ *the upper bound in* (3.13) *is strict for almost all* $\mathbf{Z}$ *(so long as* $mn \geq 2$*). Further, if* $p \notin \{1,2,\infty\}$*, the gap in the lower bound in* (3.12) *is arbitrarily small for all* $\mathbf{X}$*.*

We now turn our attention to non-equivalencies which may arise under different models of uncertainty instead of the general matrix model of linear uncertainty which we have included here, where

$$[\mathbf{\Delta}(\mathbf{X})]_{ij} = \sum_{\ell k} \Delta_{\ell k}^{(ij)} X_{\ell k} = \langle \mathbf{\Delta}^{(ij)}, \mathbf{X}\rangle,$$

with $\mathbf{\Delta}^{(ij)} \in \mathbb{R}^{m\times n}$. Another plausible model of uncertainty is one for which the jth column of $\mathbf{\Delta}(\mathbf{X})$ only depends on $\mathbf{X}_j$, the jth column of $\mathbf{X}$ (or, for example, with columns replaced by rows). We now examine such a model. In this setup, we now have n matrices $\mathbf{\Delta}^{(j)} \in \mathbb{R}^{m\times m}$ and we define the linear map $\mathbf{\Delta}$ so that the jth column of $\mathbf{\Delta}(\mathbf{X}) \in \mathbb{R}^{m\times n}$, denoted $[\mathbf{\Delta}(\mathbf{X})]_j$, is $[\mathbf{\Delta}(\mathbf{X})]_j := \mathbf{\Delta}^{(j)}\mathbf{X}_j$, which is simply matrix

vector multiplication. Therefore,

$$\boldsymbol{\Delta}(\mathbf{X}) = \begin{bmatrix} \boldsymbol{\Delta}^{(1)}\mathbf{X}_1 & \cdots & \boldsymbol{\Delta}^{(n)}\mathbf{X}_n \end{bmatrix}. \tag{3.14}$$

For an example of where such a model of uncertainty may arise, we consider matrix completion in the context of the Netflix problem. If one treats $\mathbf{X}_j$ as user j's true ratings, then such a model addresses uncertainty within a given user's ratings, while not allowing uncertainty to have cross-user effects. This model of uncertainty does not rely on true matrix structure and therefore reduces to earlier results on non-equivalence in vector regression. As an example of such a reduction, we state the following proposition characterizing equivalence. Again, this is a direct modification of Theorem 10 and the proof we do not include here.

Proposition 13. *For the model of uncertainty in (3.14) with $\boldsymbol{\Delta}^{(j)} \in \mathcal{U}_{F_{q_j}}$ for $j = 1, \ldots, n$, where $q_j \in [1, \infty]$, one has for the problem $\min_{\mathbf{X}} \max_{\boldsymbol{\Delta} \in \mathcal{U}} \|\mathbf{Y} - \mathbf{X} - \boldsymbol{\Delta}(\mathbf{X})\|_{F_p}$ that $\overline{h}$ is defined as*

$$\overline{h}(\mathbf{X}) = \lambda \left(\sum_j \delta_m^p(p, q_j) \|\mathbf{X}_j\|_{q_j^*}^p \right)^{1/p}. \tag{3.15}$$

Further, under such a model of uncertainty, robustification is equivalent to regularization with $\overline{h}$ if and only if $p \in \{1, \infty\}$ or $p = q_j$ for all $j = 1, \ldots, n$.

While the case of matrix regression offers a large variety of possible models of uncertainty, we see again that, as with vector regression, this variety inevitably leads to scenarios in which robustification is no longer directly equivalent to regularization. We summarize the conclusions of this section in Table 3.3.

3.4 Conclusion

In this chapter, we have considered the robustification of a variety of problems from classical and modern statistical regression as subject to data uncertainty. We have taken care to emphasize that there is a fine line between this process of robustification and the usual process of regularization, and that the two are not always directly

Loss function	Uncertainty set	$\overline{h}(\mathbf{X})$	Equivalence if and only if
seminorm g	$\mathcal{U}_{(h,g)}$ (h norm)	$\lambda h(\mathbf{X})$	always
F_p	$\mathcal{U}_{\sigma_q}$	$\lambda \delta_{mn}(p,2)\|\mathbf{X}\|_{F_2}$	$p \in \{1,2,\infty\}$
F_p	$\mathcal{U}_{F_q}$	$\lambda \delta_{mn}(p,q)\|\mathbf{X}\|_{F_{q^*}}$	$p \in \{1,q,\infty\}$
F_p	$\mathcal{U}$ in (3.14) with $\mathbf{\Delta}^{(j)} \in \mathcal{U}_{F_{q_j}}$	(3.15)	$(p = q_j\ \forall j)$ or $p \in \{1,\infty\}$

Table 3.3: Summary of equivalencies for robustification with uncertainty set $\mathcal{U}$ and regularization with penalty $\overline{h}$, where $\overline{h}$ is as given in Proposition 10. Here by equivalence we mean that for all $\mathbf{Z}, \mathbf{X} \in \mathbb{R}^{m\times n}$, $\max_{\mathbf{\Delta}\in\mathcal{U}} g(\mathbf{Z}+\mathbf{X}) = g(\mathbf{Z}) + \overline{h}(\mathbf{X})$, where g is the loss function, i.e., the upper bound $\overline{h}$ is also a lower bound. Here δ_{mn} is as in Theorem 10. Throughout $p, q \in [1,\infty]$ and $mn \geq 2$.

equivalent. While deepening this understanding we have also extended this connection to new domains, such as in matrix completion and PCA. In doing so, we have shown that the usual regularization approaches to modern statistical regression do not always coincide with an adversarial approach motivated by robust optimization.

Chapter 4

The Trimmed Lasso

4.1 Introduction

Sparse modeling in linear regression has been a topic of fervent interest in recent years [77, 41]. This interest has taken several forms, from substantial developments in the theory of the Lasso to advances in algorithms for convex optimization. Throughout there has been a strong emphasis on the increasingly high-dimensional nature of linear regression problems; in such problems, where the number of variables p can vastly exceed the number of observations n, sparse modeling techniques are critical for performing inference.

Context

One of the fundamental approaches to sparse modeling in the usual linear regression model of $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$, with $\mathbf{y} \in \mathbb{R}^n$ and $\mathbf{X} \in \mathbb{R}^{n \times p}$, is the best subset selection [112] problem:

$$\min_{\|\boldsymbol{\beta}\|_0 \leq k} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2, \tag{4.1}$$

which seeks to find the best choice of k from among p features that best explain the response in terms of the least squares loss function. The problem (4.1) has received extensive attention from a variety of statistical and optimization perspectives—see for example [30] and references therein. One can also consider the Lagrangian, or

penalized, form of (4.1), namely,

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \mu\|\boldsymbol{\beta}\|_0, \tag{4.2}$$

for a regularization parameter $\mu > 0$. One of the advantages of (4.1) over (4.2) is that it offers direct control over estimators' sparsity via the discrete parameter k, as opposed to the Lagrangian form (4.2) for which the correspondence between the continuous parameter μ and the resulting sparsity of estimators obtained is not entirely clear. For further discussion, see [141].

Another class of problems that have received considerable attention in the statistics and machine learning literature is the following:

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + R(\boldsymbol{\beta}), \tag{4.3}$$

where $R(\boldsymbol{\beta})$ is a choice of regularizer which encourages sparsity in $\boldsymbol{\beta}$. For example, the popularly used Lasso [156] takes the form of problem (4.3) with $R(\boldsymbol{\beta}) = \mu\|\boldsymbol{\beta}\|_1$, where $\|\cdot\|_1$ is the ℓ_1 norm; in doing so, the Lasso simultaneously selects variables and also performs shrinkage. The Lasso has seen widespread success across a variety of applications.

In contrast to the convex approach of the Lasso, there also has been been growing interest in considering richer classes of regularizers R which include nonconvex functions. Examples of such penalties include the L_q-penalty[1] (for $q \in [0, 1]$), minimax concave penalty (MCP) [164], and the smoothly clipped absolute deviation (SCAD) [60], among others. Many of the nonconvex penalty functions considered are *coordinate-wise separable*; in other words, R can be decomposed as

$$R(\boldsymbol{\beta}) = \sum_{i=1}^{p} \rho(|\beta_i|),$$

where $\rho(\cdot)$ is a real-valued function [165]. There has been a variety of evidence sug-

[1]We use L_q instead of ℓ_q throughout this chapter to avoid confusion with an index term ℓ that appears later.

gesting the promise of such nonconvex approaches in overcoming certain shortcomings of Lasso-like approaches.

One of the central ideas of nonconvex penalty methods used in sparse modeling is that of creating a continuum of estimation problems which bridge the gap between convex methods for sparse estimation (such as Lasso) and subset selection in the form (4.2). However, as noted above, such a connection does not necessarily offer direct control over the desired level of sparsity of estimators.

The trimmed Lasso

In contrast with coordinate-wise separable penalties as considered above, we consider a family of penalties that are not separable across coordinates. One such penalty which forms a principal object of our study herein is

$$T_k(\boldsymbol{\beta}) := \min_{\|\boldsymbol{\phi}\|_0 \leq k} \|\boldsymbol{\phi} - \boldsymbol{\beta}\|_1.$$

The penalty T_k is a measure of the distance from the set of k-sparse estimators as measured via the L_1 norm. In other words, when used in problem (4.3), the penalty $R = T_k$ controls the amount of shrinkage towards sparse models.

The penalty T_k can equivalently be written as

$$T_k(\boldsymbol{\beta}) = \sum_{i=k+1}^{p} |\beta_{(i)}|,$$

where $|\beta_{(1)}| \geq |\beta_{(2)}| \geq \cdots \geq |\beta_{(p)}|$ are the sorted entries of $\boldsymbol{\beta}$. In words, $T_k(\boldsymbol{\beta})$ is the sum of the absolute values of the $p-k$ smallest magnitude entries of $\boldsymbol{\beta}$. The penalty was first introduced in [154, 78, 72, 159]. We refer to this family of penalty functions (over choices of k) as the *trimmed Lasso*.[2] The case of $k = 0$ recovers the usual Lasso, as one would suspect. The distinction, of course, is that for general k, T_k no longer shrinks, or biases towards zero, the k largest entries of $\boldsymbol{\beta}$.

Let us consider the least squares loss regularized via the trimmed lasso penalty—

[2]The choice of name is our own and is motivated by the least trimmed squares estimator (4.5).

this leads to the following optimization criterion:

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda T_k(\boldsymbol{\beta}), \tag{4.4}$$

where $\lambda > 0$ is the regularization parameter. The penalty term shrinks the smallest $p - k$ entries of $\boldsymbol{\beta}$ and does not impose any penalty on the largest k entries of $\boldsymbol{\beta}$. If λ becomes larger, the smallest $p - k$ entries of $\boldsymbol{\beta}$ are shrunk further; after a certain threshold—as soon as $\lambda \geq \lambda_0$ for some finite λ_0—the smallest $p - k$ entries are set to zero. The existence of a finite λ_0 (as stated above) is an attractive feature of the trimmed Lasso and is known as its *exactness* property, namely, for λ sufficiently large, the problem (4.4) exactly solves constrained best subset selection as in problem (4.1) (see [72]). Note here the contrast with the separable penalty functions which correspond instead with problem (4.2); as such, the trimmed Lasso is distinctive in that it offers precise control over the desired level of sparsity via the discrete parameter k. Further, it is also notable that many algorithms developed for separable-penalty estimation problems can be directly adapted for the trimmed Lasso.

Our objective in studying the trimmed Lasso is distinctive from previous approaches. In particular, while previous work on the penalty T_k has focused primarily on its use as a tool for reformulating sparse optimization problems [154, 78] and on how such reformulations can be solved computationally [72, 159], we instead aim to explore the trimmed Lasso's structural properties and its relation to existing sparse modeling techniques.

In particular, a natural question we seek to explore is, what is the connection of the trimmed Lasso penalty with existing separable penalties commonly used in sparse statistical learning? For example, the trimmed Lasso bears a close resemblance to the clipped (or capped) Lasso penalty [166], namely, $\sum_{i=1}^{p} \mu \min\{\gamma|\beta_i|, 1\}$, where $\mu, \gamma > 0$ are parameters (when γ is large, the clipped Lasso approximates $\mu\|\boldsymbol{\beta}\|_0$).

Robustness: robust statistics and robust optimization

A significant thread woven throughout the consideration of penalty methods for sparse modeling is the notion of robustness—in short, the ability of a method to perform in the face of noise. Not surprisingly, the notion of robustness has myriad distinct meanings depending on the context. Indeed, as Huber, a pioneer in the area of robust statistics, aptly noted:

> "The word 'robust' is loaded with many—sometimes inconsistent—connotations." [81, p. 2]

For this reason, we consider robustness from several perspectives—both the robust statistics [81] and robust optimization [19] viewpoints.

A common premise of the various approaches is as follows: that a robust model should perform well even under small deviations from its underlying assumptions; and that to achieve such behavior, some efficiency under the assumed model should be sacrificed. Not surprisingly in light of Huber's prescient observation, the exact manifestation of this idea can take many different forms, even if the initial premise is ostensibly the same.

Robust statistics and the "min-min" approach

One such approach is in the field of robust statistics [81, 131, 114]. In this context, the primary assumptions are often probabilistic, i.e. distributional, in nature, and the deviations to be "protected against" include possibly gross, or arbitrarily bad, errors. Put simply, robust statistics is primary focused on analyzing and mitigating the influence of outliers on estimation methods.

There have been a variety of proposals of different estimators to achieve this. One that is particularly relevant for our purposes is that of *least trimmed squares* ("LTS") [131]. For fixed $j \in \{1, \dots, n\}$, the LTS problem is defined as

$$\min_{\boldsymbol{\beta}} \sum_{i=j+1}^{n} |r_{(i)}(\boldsymbol{\beta})|^2, \tag{4.5}$$

where $r_i(\boldsymbol{\beta}) = y_i - \mathbf{x}_i'\boldsymbol{\beta}$ are the residuals and $r_{(i)}(\boldsymbol{\beta})$ are the sorted residuals given $\boldsymbol{\beta}$ with $|r_{(1)}(\boldsymbol{\beta})| \geq |r_{(2)}(\boldsymbol{\beta})| \geq \cdots \geq |r_{(n)}(\boldsymbol{\beta})|$. In words, the LTS estimator performs ordinary least squares on the $n-j$ smallest residuals (discarding the j largest or worst residuals).

Furthermore, it is particularly instructive to express (4.5) in the equivalent form (*cf.* [31])

$$\min_{\boldsymbol{\beta}} \min_{\substack{I \subseteq \{1,\ldots,n\}: \\ |I| = n-j}} \sum_{i \in I} |r_i(\boldsymbol{\beta})|^2. \tag{4.6}$$

In light of this representation, we refer to LTS as a form of "min-min" robustness. One could also interpret this min-min robustness as *optimistic* in the sense the estimation problems (4.6) and, *a fortiori*, (4.5) allow the modeler to also choose observations to discard.

Other min-min models of robustness

Another approach to robustness which also takes a min-min form like LTS is the classical technique known as *total least squares* [69, 109]. For our purposes, we consider total least squares in the form

$$\min_{\boldsymbol{\beta}} \min_{\boldsymbol{\Delta}} \frac{1}{2}\|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_2^2 + \eta\|\boldsymbol{\Delta}\|_2^2, \tag{4.7}$$

where $\|\boldsymbol{\Delta}\|_2$ is the usual Frobenius norm of the matrix $\boldsymbol{\Delta}$ and $\eta > 0$ is a scalar parameter. In this framework, one again has an optimistic view on error: find the best possible "correction" of the data matrix $\mathbf{X}$ as $\mathbf{X} + \boldsymbol{\Delta}^*$ and perform least squares using this corrected data (with η controlling the flexibility in choice of $\boldsymbol{\Delta}$).

In contrast with the penalized form of (4.7), one could also consider the problem in a constrained form such as

$$\min_{\boldsymbol{\beta}} \min_{\boldsymbol{\Delta} \in \mathcal{V}} \frac{1}{2}\|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_2^2, \tag{4.8}$$

where $\mathcal{V} \subseteq \mathbb{R}^{n \times p}$ is defined as $\mathcal{V} = \{\boldsymbol{\Delta} : \|\boldsymbol{\Delta}\|_2 \leq \eta'\}$ for some $\eta' > 0$. This problem again has the min-min form, although now with perturbations $\boldsymbol{\Delta}$ as restricted to $\mathcal{V}$.

Robust optimization and the "min-max" approach

We now turn our attention to a different approach to the notion of robustness known as robust optimization [19, 25], as seen in Chapter 3. In contrast with robust statistics, robust optimization typically replaces distributional assumptions with a new primitive, namely, the deterministic notion of an *uncertainty set*. Further, in robust optimization one considers a worst-case or pessimistic perspective and the focus is on perturbations from the nominal model (as opposed to possible gross corruptions as in robust statistics).

To be precise, one possible robust optimization model for linear regression takes form

$$\min_{\boldsymbol{\beta}} \max_{\boldsymbol{\Delta} \in \mathcal{U}} \frac{1}{2} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_2^2, \tag{4.9}$$

where $\mathcal{U} \subseteq \mathbb{R}^{n \times p}$ is a (deterministic) uncertainty set that captures the possible deviations of the model (from the nominal data $\mathbf{X}$), *cf.* Chapter 3. Note the immediate contrast with the robust models considered earlier (LTS and total least squares in (4.5) and (4.7), respectively) that take the min-min form; instead, robust optimization focuses on "min-max" robustness. For a related discussion contrasting the min-min approach with min-max, see [17, 91, 40] and references therein.

As seen in Chapter 3, one of the attractive features of the min-max formulation is that it gives a re-interpretation of several statistical regularization methods. For example, the usual Lasso (problem (4.3) with $R = \mu\ell_1$) can be expressed in the form (4.9) for a specific choice of uncertainty set:

Proposition 14 ([162])**.** *Problem* (4.9) *with uncertainty set* $\mathcal{U} = \{\boldsymbol{\Delta} : \|\boldsymbol{\Delta}_i\|_2 \leq \mu \ \forall i\}$ *is equivalent to the Lasso, i.e., problem* (4.3) *with* $R(\boldsymbol{\beta}) = \mu\|\boldsymbol{\beta}\|_1$*, where* $\boldsymbol{\Delta}_i$ *denotes the* i*th column of* $\boldsymbol{\Delta}$*.*

Other min-max models of robustness

We close our discussion of robustness by considering another example of min-max robustness that is of particular relevance to the trimmed Lasso. In particular, we

consider problem (4.3) with the SLOPE (or OWL) penalty [34, 63], namely,

$$R_{\mathrm{SLOPE}(\mathbf{w})}(\boldsymbol{\beta}) = \sum_{i=1}^{p} w_i |\beta_{(i)}|,$$

where $\mathbf{w}$ is a (fixed) vector of weights with $w_1 \geq w_2 \geq \cdots \geq w_p \geq 0$ and $w_1 > 0$. In its simplest form, the SLOPE penalty has weight vector $\tilde{\mathbf{w}}$, where $\tilde{w}_1 = \cdots = \tilde{w}_k = 1$, $\tilde{w}_{k+1} = \cdots = \tilde{w}_p = 0$, in which case we have the identity

$$R_{\mathrm{SLOPE}(\tilde{\mathbf{w}})}(\boldsymbol{\beta}) = \|\boldsymbol{\beta}\|_1 - T_k(\boldsymbol{\beta}).$$

There are some apparent similarities but also subtle differences between the SLOPE penalty and the trimmed Lasso. From a high level, while the trimmed Lasso focuses on the smallest magnitude entries of $\boldsymbol{\beta}$, the SLOPE penalty in its simplest form focuses on the *largest* magnitude entries of $\boldsymbol{\beta}$. As such, the trimmed Lasso is generally nonconvex, while the SLOPE penalty is always convex; consequently, the techniques for solving the related estimation problems will necessarily be different.

Finally, we note that the SLOPE penalty can be considered as a min-max model of robustness for a particular choice of uncertainty set:

Proposition 15. *Problem* (4.9) *with uncertainty set*

$$\mathcal{U} = \left\{ \boldsymbol{\Delta} : \begin{array}{c} \boldsymbol{\Delta} \text{ has at most } k \text{ nonzero} \\ \text{columns and } \|\boldsymbol{\Delta}_i\|_2 \leq \mu \; \forall i \end{array} \right\}$$

is equivalent to problem (4.3) *with* $R(\boldsymbol{\beta}) = \mu R_{\mathrm{SLOPE}(\tilde{\mathbf{w}})}(\boldsymbol{\beta})$, *where* $\tilde{w}_1 = \cdots = \tilde{w}_k = 1$ *and* $\tilde{w}_{k+1} = \cdots = \tilde{w}_p = 0$.

We return to this particular choice of uncertainty set later. (For completeness, we include a more general min-max representation of SLOPE in Appendix C.1.)

Computation and Algorithms

Broadly speaking, there are numerous distinct approaches to algorithms for solving problems of the form (4.1)–(4.3) for various choices of R. We do not attempt to provide a comprehensive list of such approaches for general R, but we will discuss existing approaches for the trimmed Lasso and closely related problems. Approaches typically take one of two forms: heuristic or exact.

Heuristic techniques

Heuristic approaches to solving problems (4.1)–(4.3) often use techniques from convex optimization [38], such as proximal gradient descent or coordinate descent (see [60, 110]). Typically these techniques are coupled with an analysis of local or global behavior of the algorithm. For example, global behavior is often considered under additional restrictive assumptions on the underlying data; unfortunately, verifying such assumptions can be as difficult as solving the original nonconvex problem. (For example, consider the analogy with compressed sensing [45, 55, 59] and the hardness of verifying whether underlying assumptions hold [157, 12]).

There is also extensive work studying the local behavior (e.g. stationarity) of heuristic approaches to these problems. For the specific problems (4.1) and (4.2), the behavior of augmented Lagrangian methods [7, 153] and complementarity constraint techniques [39, 42, 62, 54] have been considered. For other local approaches, see [105].

Exact techniques

One of the primary drawbacks of heuristic techniques is that it can often be difficult to verify the degree of suboptimality of the estimators obtained. For this reason, there has been an increasing interest in studying the behavior of exact algorithms for providing certifiably optimal solutions to problems of the form (4.1)–(4.3) [30, 31, 104, 111]. Often these approaches make use of techniques from *mixed integer optimization* [35] which are implemented in a variety of software, e.g. Gurobi [73]. The tradeoff with such approaches is that they typically carry a heavier computational burden than

convex approaches. For a discussion of the application of mixed integer optimization in statistics, see [30, 31, 104, 111].

What this chapter is about

In this chapter, we focus on a detailed analysis of the trimmed Lasso, especially with regard to its properties and its relation to existing methods. In particular, we explore the trimmed Lasso from two perspectives: that of sparsity as well as that of robustness. We summarize our contributions as follows:

1. We study the robustness of the trimmed Lasso penalty. In particular, we provide several min-min robustness representations of it. We first show that the same choice of uncertainty set that leads to the SLOPE penalty in the min-max robust model (4.9) gives rise to the trimmed Lasso in the corresponding min-min robust problem (4.8) (with an additional regularization term). This gives an interpretation of the SLOPE and trimmed Lasso as a complementary pair of penalties, one under a pessimistic (min-max) model and the other under an optimistic (min-min) model.

   Moreover, we show another min-min robustness interpretation of the trimmed Lasso by comparison with the ordinary Lasso. In doing so, we further highlight the nature of the trimmed Lasso and its relation to the LTS problem (4.5).

2. We provide a detailed analysis on the connection between estimation approaches using the trimmed Lasso and separable penalty functions. In doing so, we show directly how penalties such as the trimmed Lasso can be viewed as a generalization of such existing approaches in certain cases. In particular, a trimmed-Lasso-like approach always subsumes its separable analogue, and the containment is strict in general. We also focus on the specific case of the clipped (or capped) Lasso [166]; for this we precisely characterize the relationship and provide a necessary and sufficient condition for the two approaches to be equivalent. In doing so, we highlight some of the limitations of an approach using a separable penalty function.

3. Finally, we describe a variety of algorithms, both existing and new, for trimmed Lasso estimation problems. We contrast two heuristic approaches for finding locally optimal solutions with exact techniques from mixed integer optimization that can be used to produce certificates of optimality for solutions found via the convex approaches. We also show that the convex envelope [130] of the trimmed Lasso takes the form

$$(\|\boldsymbol{\beta}\|_1 - k)_+ ,$$

where $(a)_+ := \max\{0, a\}$, a "soft-thresholded" variant of the ordinary Lasso. Throughout this section, we emphasize how techniques from convex optimization can be used to find high-quality solutions to the trimmed Lasso estimation problem. An implementation of the various algorithms presented herein can be found in Appendix D.

Chapter structure

The structure of the chapter is as follows. In Section 4.2, we study several properties of the trimmed Lasso, provide a few distinct interpretations, and highlight possible generalizations. In Section 4.3, we explore the trimmed Lasso in the context of robustness. Then, in Section 4.4, we study the relationship between the trimmed Lasso and other nonconvex penalties. In Section 4.5, we study the algorithmic implications of the trimmed Lasso. Finally, in Section 4.6 we share our concluding thoughts and highlight future directions.

4.2 Structural properties and interpretations

In this section, we provide further background on the trimmed Lasso: its motivations, interpretations, and generalizations. Our remarks in this section are broadly grouped as follows: in Section 4.2.1 we summarize the trimmed Lasso's basic properties as detailed in [154, 78, 72, 159]; we then turn our attention to an interpretation

of the trimmed Lasso as a relaxation of complementarity constraints problems from optimization (Section 4.2.2) and as a variable decomposition method (Section 4.2.3); finally, in Sections 4.2.4 and 4.2.5 we highlight the key structural features of the trimmed Lasso by identifying possible generalizations of its definition and its application. These results augment the existing literature by giving a deeper understanding of the trimmed Lasso and provide a basis for further results in Sections 4.3 and 4.4.

4.2.1 Basic observations

We begin with a summary of some of the basic properties of the trimmed Lasso as studied in [154, 78, 72]. First of all, let us also include another representation of T_k:

Lemma 1. *For any $\boldsymbol{\beta}$,*

$$T_k(\boldsymbol{\beta}) = \min_{\substack{I \subseteq \{1,\dots,p\}: \\ |I| = p-k}} \sum_{i \in I} |\beta_i| = \begin{array}{ll} \min\limits_{\mathbf{z}} & \langle \mathbf{z}, |\boldsymbol{\beta}| \rangle \\ \text{s. t.} & \sum\limits_i z_i = p - k \\ & \mathbf{z} \in \{0,1\}^p, \end{array}$$

where $|\boldsymbol{\beta}|$ denotes the vector whose entries are the absolute values of the entries of $\boldsymbol{\beta}$.

In other words, the trimmed Lasso can be represented using auxiliary binary variables.

Now let us consider the problem

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda T_k(\boldsymbol{\beta}), \tag{TL$_{\lambda,k}$}$$

where $\lambda > 0$ and $k \in \{0, 1, \dots, p\}$ are parameters. Based on the definition of T_k, we have the following:

Lemma 2. *The problem* ($\mathrm{TL}_{\lambda,k}$) *can be rewritten exactly in several equivalent forms:*

$$\begin{aligned}(\mathrm{TL}_{\lambda,k}) &= \min_{\substack{\boldsymbol{\beta},\boldsymbol{\phi}:\\ \|\boldsymbol{\phi}\|_0\le k}} \frac{1}{2}\|\mathbf{y}-\mathbf{X}\boldsymbol{\beta}\|^2+\lambda\|\boldsymbol{\beta}-\boldsymbol{\phi}\|_1\\ &= \min_{\substack{\boldsymbol{\beta},\boldsymbol{\phi},\boldsymbol{\epsilon}:\\ \boldsymbol{\beta}=\boldsymbol{\phi}+\boldsymbol{\epsilon}\\ \|\boldsymbol{\phi}\|_0\le k}} \frac{1}{2}\|\mathbf{y}-\mathbf{X}\boldsymbol{\beta}\|^2+\lambda\|\boldsymbol{\epsilon}\|_1\\ &= \min_{\substack{\boldsymbol{\phi},\boldsymbol{\epsilon}:\\ \|\boldsymbol{\phi}\|_0\le k}} \frac{1}{2}\|\mathbf{y}-\mathbf{X}(\boldsymbol{\phi}+\boldsymbol{\epsilon})\|^2+\lambda\|\boldsymbol{\epsilon}\|_1\end{aligned}$$

Exact penalization

Based on the definition of T_k, it follows that $T_k(\boldsymbol{\beta}) = 0$ if and only if $\|\boldsymbol{\beta}\|_0 \le k$. Therefore, one can rewrite problem (4.1) as

$$\min_{T_k(\boldsymbol{\beta})=0} \frac{1}{2}\|\mathbf{y}-\mathbf{X}\boldsymbol{\beta}\|_2^2.$$

In Lagrangian form, this would suggest an approximation for (4.1) of the form

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y}-\mathbf{X}\boldsymbol{\beta}\|_2^2+\lambda T_k(\boldsymbol{\beta}),$$

where $\lambda > 0$. As noted in the introduction, this approximation is in fact exact (in the sense of [22, 24]), summarized in the following theorem; for completeness, we include a full proof that is distinct from that in [72].[3]

Theorem 13 (*cf.* [72])**.** *For any fixed* $k \in \{0, 1, 2, \ldots, p\}$, $\eta > 0$, *and problem data* $\mathbf{y}$ *and* $\mathbf{X}$, *there exists some* $\underline{\lambda} = \underline{\lambda}(\mathbf{y}, \mathbf{X}) > 0$ *so that for all* $\lambda > \underline{\lambda}$, *the problems*

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y}-\mathbf{X}\boldsymbol{\beta}\|_2^2+\lambda T_k\left(\boldsymbol{\beta}\right)+\eta\|\boldsymbol{\beta}\|_1$$

and

$$\begin{aligned}\min_{\boldsymbol{\beta}} \quad & \tfrac{1}{2}\|\mathbf{y}-\mathbf{X}\boldsymbol{\beta}\|_2^2+\eta\|\boldsymbol{\beta}\|_1\\ \text{s. t.} \quad & \|\boldsymbol{\beta}\|_0 \le k\end{aligned}$$

[3]The presence of the additional regularizer $\eta\|\boldsymbol{\beta}\|_1$ can be interpreted in many ways. For our purposes, it serves to make the problems well-posed.

have the same optimal objective value and the same set of optimal solutions.

Proof. Let $\underline{\lambda} = \|\mathbf{y}\|_2 \cdot (\max_j \|\mathbf{x}_j\|_2)$, where $\mathbf{x}_j$ denotes the jth row of $\mathbf{X}$. We fix $\lambda > \underline{\lambda}$, k, and $\eta > 0$ throughout the entire proof. We begin by observing that it suffices to show that any solution $\boldsymbol{\beta}$ to

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda T_k(\boldsymbol{\beta}) + \eta\|\boldsymbol{\beta}\|_1 \tag{4.10}$$

satisfies $T_k(\boldsymbol{\beta}) = 0$, or equivalently, $\|\boldsymbol{\beta}\|_0 \leq k$. As per Lemma 1, problem (4.10) can be rewritten exactly as

$$\begin{aligned} \min_{\boldsymbol{\beta},\mathbf{z}} \quad & \tfrac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda\langle \mathbf{z}, |\boldsymbol{\beta}|\rangle + \eta\|\boldsymbol{\beta}\|_1 \\ \text{s.t.} \quad & \sum_i z_i = p - k \\ & \mathbf{z} \in \{0,1\}^p. \end{aligned} \tag{4.11}$$

Let $(\boldsymbol{\beta}^*, \mathbf{z}^*)$ be any solution to (4.11). Observe that necessarily $\boldsymbol{\beta}^*$ is also a solution to the problem

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda\langle \mathbf{z}^*, |\boldsymbol{\beta}|\rangle + \eta\|\boldsymbol{\beta}\|_1. \tag{4.12}$$

Note that, unlike (4.10), the problem in (4.12) is readily amenable to an analysis using the theory of proximal gradient methods [49, 16]. In particular, we must have for any $\gamma > 0$ that

$$\boldsymbol{\beta}^* = \mathrm{prox}_{\gamma R}\left(\boldsymbol{\beta}^* - \gamma(\mathbf{X}'\mathbf{X}\boldsymbol{\beta}^* - \mathbf{X}'\mathbf{y})\right), \tag{4.13}$$

where $R(\boldsymbol{\beta}) = \eta\|\boldsymbol{\beta}\|_1 + \lambda \sum_{i\,:\,z_i^*=1} |\beta_i|$. Suppose that $T_k(\boldsymbol{\beta}^*) > 0$. In particular, for some $j \in \{1, \ldots, p\}$, we have $\beta_j^* \neq 0$ and $z_j^* = 1$. Yet, as per (4.13),[4]

$$\left|\beta_j^* - \gamma\langle \mathbf{x}_j, \mathbf{X}\boldsymbol{\beta}^* - \mathbf{y}\rangle\right| > \gamma(\eta + \lambda) \qquad \text{for all } \gamma > 0,$$

[4]This is valid for the following reason: since $\beta_j^* \neq 0$ and β_j^* satisfies (4.13), it must be the case that $\left|\beta_j^* - \gamma\mathbf{x}_j'(\mathbf{X}\boldsymbol{\beta}^* - \mathbf{y})\right| > \gamma(\eta + \lambda)$, for otherwise the soft-thresholding operator at level $\gamma(\eta + \lambda)$ would set this quantity to zero.

where $\mathbf{x}_j$ denotes the jth row of $\mathbf{X}$. This implies that

$$|\langle \mathbf{x}_j, \mathbf{X}\boldsymbol{\beta}^* - \mathbf{y}\rangle| \geq \eta + \lambda.$$

Now, using the definition of $\underline{\lambda}$, observe that

$$\begin{aligned}\eta + \lambda \leq |\langle \mathbf{x}_j, \mathbf{X}\boldsymbol{\beta}^* - \mathbf{y}\rangle| &\leq \|\mathbf{x}_j\|_2 \|\mathbf{X}\boldsymbol{\beta}^* - \mathbf{y}\|_2 \\ &\leq \|\mathbf{x}_j\|_2 \|\mathbf{y}\| \leq \underline{\lambda} < \lambda,\end{aligned}$$

which is a contradiction since $\eta > 0$. Hence, $T_k(\boldsymbol{\beta}^*) = 0$, completing the proof. □

The direct implication is that trimmed Lasso leads to a continuum (over λ) of relaxations to the best subset selection problem starting from ordinary least squares estimation; further, best subset selection lies on this continuum for λ sufficiently large.

4.2.2 A complementary constraints viewpoint

We now turn our attention to a new perspective on the trimmed Lasso as considered via mathematical programming with complementarity constraints ("MPCCs") [136, 103, 89, 79, 90, 42], sometimes also referred to as mathematical programs with equilibrium constraints [48]. By studying this connection, we will show that a penalized form of a common relaxation scheme for MPCCs leads directly to the trimmed Lasso penalty. This gives a distinctly different optimization perspective on the trimmed Lasso penalty.

As detailed in [39, 42, 62], the problem (4.1) can be exactly rewritten as

$$\begin{aligned}\min_{\boldsymbol{\beta},\mathbf{z}} \quad & \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 \\ \text{s.t.} \quad & \textstyle\sum_i z_i = p - k \\ & \mathbf{z} \in [0,1]^p \\ & z_i\beta_i = 0.\end{aligned} \tag{4.14}$$

by the inclusion of auxiliary variables $\mathbf{z} \in [0,1]^p$. In essence, the auxiliary variables

replace the combinatorial constraint $\|\boldsymbol{\beta}\|_0 \leq k$ with *complementarity* constraints of the form $z_i\beta_i = 0$. Of course, the problem as represented in (4.14) is still not directly amenable to convex optimization techniques.

As such, relaxation schemes can be applied to (4.14). One popular method from the MPCC literature is the Scholtes-type relaxation [79]; applied to (4.14) as in [42, 62], this takes the form

$$\begin{aligned} \min_{\boldsymbol{\beta},\mathbf{z}} \quad & \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 \\ \text{s.t.} \quad & \textstyle\sum_i z_i = p - k \\ & \mathbf{z} \in [0,1]^p \\ & |z_i\beta_i| \leq t, \end{aligned} \tag{4.15}$$

where $t > 0$ is some fixed numerical parameter which controls the strength of the relaxation, with $t = 0$ exactly recovering (4.14). In the traditional MPCC context, it is standard to study local optimality and stationarity behavior of solutions to (4.15) as they relate to the original problem (4.1), *cf.* [62].

Instead, let us consider a different approach. In particular, consider a penalized, or Lagrangian, form of the Scholtes relaxation (4.15), namely,

$$\begin{aligned} \min_{\boldsymbol{\beta},\mathbf{z}} \quad & \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \sum_i (|z_i\beta_i| - t) \\ \text{s.t.} \quad & \textstyle\sum_i z_i = p - k \\ & \mathbf{z} \in [0,1]^p \end{aligned} \tag{4.16}$$

for some fixed $\lambda \geq 0$.[5] Observe that we can minimize (4.16) with respect to $\mathbf{z}$ to obtain the equivalent problem

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda T_k(\boldsymbol{\beta}) - p\lambda t,$$

which is precisely problem ($\mathrm{TL}_{\lambda,k}$) (up to the fixed additive constant). In other words,

[5] To be precise, this is a *weaker* relaxation than if we had separate dual variables λ_i for each constraint $|z_i\beta_i| \leq t$, at least in theory.

the trimmed Lasso can also be viewed as arising directly from a penalized form of the MPCC relaxation, with auxiliary variables eliminated. This gives another view on Lemma 1 which gave a representation of T_k using auxiliary binary variables.

4.2.3 Variable decomposition

To better understand the relation of the trimmed Lasso to existing methods, it is also useful to consider alternative representations. Here we focus on representations which connect it to variable decomposition methods. Our discussion here is an extended form of related discussions in [78, 72, 159].

To begin, we return to the final representation of the trimmed Lasso problem as shown in Lemma 2, viz.,

$$(\mathrm{TL}_{\lambda,k}) = \min_{\substack{\boldsymbol{\phi},\boldsymbol{\epsilon}:\\ \|\boldsymbol{\phi}\|_0 \leq k}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}(\boldsymbol{\phi}+\boldsymbol{\epsilon})\|^2 + \lambda\|\boldsymbol{\epsilon}\|_1. \tag{4.17}$$

We will refer to $(\mathrm{TL}_{\lambda,k})$ in the form (4.17) as the *split* or *decomposed* representation of the problem. This is because in this form it is clear that we can think about estimators $\boldsymbol{\beta}$ found via $(\mathrm{TL}_{\lambda,k})$ as being decomposed into two different estimators: a sparse component $\boldsymbol{\phi}$ and another component $\boldsymbol{\epsilon}$ with small ℓ_1 norm (as controlled via λ).

Several remarks are in order. First, the decomposition of $\boldsymbol{\beta}$ into $\boldsymbol{\beta} = \boldsymbol{\phi} + \boldsymbol{\epsilon}$ is truly a decomposition in that if $\boldsymbol{\beta}^*$ is an optimal solution to $(\mathrm{TL}_{\lambda,k})$ with $(\boldsymbol{\phi}^*, \boldsymbol{\epsilon}^*)$ a corresponding optimal solution to the split representation of the problem (4.17), then one must have that $\phi_i^*\epsilon_i^* = 0$ for all $i \in \{1,\ldots,p\}$. In other words, the supports of $\boldsymbol{\phi}$ and $\boldsymbol{\epsilon}$ do not overlap; therefore, $\boldsymbol{\beta}^* = \boldsymbol{\phi}^* + \boldsymbol{\epsilon}^*$ is a genuine decomposition.

Secondly, the variable decomposition (4.17) suggests that the problem of finding the k largest entries of $\boldsymbol{\beta}$ (i.e., finding $\boldsymbol{\phi}$) can be solved as a best subset selection problem with a (possibly different) convex loss function (without $\boldsymbol{\epsilon}$). To see this,

observe that the problem of finding $\boldsymbol{\phi}$ in (4.17) can be written as the problem

$$\min_{\|\boldsymbol{\phi}\|_0 \leq k} \widetilde{L}(\boldsymbol{\phi}),$$

where

$$\widetilde{L}(\boldsymbol{\phi}) = \min_{\boldsymbol{\epsilon}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}(\boldsymbol{\phi} + \boldsymbol{\epsilon})\|_2^2 + \lambda\|\boldsymbol{\epsilon}\|_1.$$

Using theory on duality for the Lasso problem [120], one can argue that $\widetilde{L}$ is itself a convex loss function. Hence, the variable decomposition gives some insight into how the largest k loadings for the trimmed Lasso relates to solving a related sparse estimation problem.

A view towards matrix estimation

Finally, we contend that the variable decomposition of $\boldsymbol{\beta}$ as a sparse component $\boldsymbol{\phi}$ plus a "noise" component $\boldsymbol{\epsilon}$ with small norm is a natural and useful analogue of corresponding decompositions in the matrix estimation literature, such as in factor analysis and robust Principal Component Analysis [46]. For the purposes of the present work, we will focus on the analogy with factor analysis.

To describe the connection, we briefly review the setup as given in Chapter 2. Given a covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{p \times p}$, one is interested in describing it as the sum of two distinct components: a low-rank component $\boldsymbol{\Theta}$ (corresponding to a low-dimensional covariance structure common across the variables) and a diagonal component $\boldsymbol{\Phi}$ (corresponding to individual variances unique to each variable)—in symbols, $\boldsymbol{\Sigma} = \boldsymbol{\Theta} + \boldsymbol{\Phi}$.

In reality, this *noiseless* decomposition is often too restrictive, and therefore it is often better to focus on finding a decomposition $\boldsymbol{\Sigma} = \boldsymbol{\Theta} + \boldsymbol{\Phi} + \mathcal{N}$, where $\mathcal{N}$ is a noise component with small norm. Accordingly, as in Chapter 2, a corresponding

estimation procedure can take the form

$$\begin{aligned} \min_{\boldsymbol{\Theta},\boldsymbol{\Phi}} \quad & \|\boldsymbol{\Sigma} - (\boldsymbol{\Theta} + \boldsymbol{\Phi})\| \\ \text{s.t.} \quad & \operatorname{rank}(\boldsymbol{\Theta}) \leq k \\ & \boldsymbol{\Phi} \text{ is diagonal} \\ & \boldsymbol{\Theta}, \boldsymbol{\Phi} \succcurlyeq \mathbf{0}, \end{aligned} \tag{4.18}$$

where the constraint $\mathbf{A} \succcurlyeq \mathbf{0}$ denotes that $\mathbf{A}$ is symmetric, positive semidefinite, and $\|\cdot\|$ is some norm. One of the attractive features of the estimation criterion (4.18) is that for common choices of $\|\cdot\|$, it is possible to completely eliminate the combinatorial rank constraint and the variable $\boldsymbol{\Theta}$ to yield a smooth (nonconvex) optimization problem with compact, convex constraints.

This exact same argument can be used to motivate the appearance of the trimmed Lasso penalty. Indeed, instead of considering estimators $\boldsymbol{\beta}$ which are exactly k-sparse (i.e., $\|\boldsymbol{\beta}\|_0 \leq k$), we instead consider estimators which are approximately k-sparse, i.e., $\boldsymbol{\beta} = \boldsymbol{\phi} + \boldsymbol{\epsilon}$, where $\|\boldsymbol{\phi}\|_0 \leq k$ and $\boldsymbol{\epsilon}$ has small norm. Given fixed $\boldsymbol{\beta}$, such a procedure is precisely

$$\min_{\|\boldsymbol{\phi}\|_0 \leq k} \|\boldsymbol{\beta} - \boldsymbol{\phi}\|.$$

Just as the rank constraint is eliminated from (4.18), the sparsity constraint can be eliminated from this to yield a continuous penalty which precisely captures the quality of the approximation $\boldsymbol{\beta} \approx \boldsymbol{\phi}$. The trimmed Lasso uses the choice $\|\cdot\| = L_1$, although other choices are possible; see Section 4.2.4.

This analogy with factor analysis is also useful in highlighting additional benefits of the trimmed Lasso. One of particular note is that it enables the direct application of existing convex optimization techniques to find high-quality solutions to ($\text{TL}_{\lambda,k}$).

4.2.4 Generalizations

We close this section by considering some generalizations of the trimmed Lasso. These are particularly useful for connecting the trimmed Lasso to other penalties, as we will see later in Section 4.4.

As noted earlier, the trimmed Lasso measures the distance (in ℓ_1 norm) from the set of k-sparse vectors; therefore, it is natural to inquire what properties other measures of distance might carry. In light of this, we begin with a definition:

Definition 3. *Let $k \in \{0, 1, \ldots, p\}$ and $g : \mathbb{R}_+ \to \mathbb{R}_+$ be any unbounded, continuous, and strictly increasing function with $g(0) = 0$. Define the corresponding kth projected penalty function, denoted π_k^g, as*

$$\pi_k^g(\boldsymbol{\beta}) = \min_{\|\boldsymbol{\phi}\|_0 \leq k} \sum_i g(|\phi_i - \beta_i|).$$

It is not difficult to argue that π_k^g has as an equivalent definition

$$\pi_k^g(\boldsymbol{\beta}) = \sum_{i>k} g(|\beta_{(i)}|).$$

As an example, π_k^g is the trimmed Lasso penalty when g is the absolute value, viz. $g(x) = |x|$, and so it is a special case of the projected penalties. Alternatively, suppose $g(x) = x^2/2$. In this case, we get a trimmed version of the ridge regression penalty: $\sum_{i>k} |\beta_{(i)}|^2/2$.

This class of penalty functions has one notable feature, summarized in the following results:

Proposition 16. *If $g : \mathbb{R}_+ \to \mathbb{R}_+$ is an unbounded, continuous, and strictly increasing function with $g(0) = 0$, then for any $\boldsymbol{\beta}$, $\pi_k^g(\boldsymbol{\beta}) = 0$ if and only if $\|\boldsymbol{\beta}\|_0 \leq k$. Hence, the problem $\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \pi_k^g(\boldsymbol{\beta})$ converges in objective value to $\min_{\|\boldsymbol{\beta}\|_0 \leq k} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2$ as $\lambda \to \infty$.*

Let us set a standard notion: we say that $\boldsymbol{\beta}$ is ϵ-optimal (for $\epsilon > 0$) to an optimization problem (P) if the optimal objective value of (P) is within ϵ of the objective value of $\boldsymbol{\beta}$.

Proposition 17 (Extended form of Proposition 16)**.** *Let $g : \mathbb{R}_+ \to \mathbb{R}_+$ be an unbounded, continuous, and strictly increasing function with $g(0) = 0$. Consider the*

problems

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \pi_k^g(\boldsymbol{\beta}) + \eta\|\boldsymbol{\beta}\|_1 \tag{4.19}$$

and

$$\min_{\|\boldsymbol{\beta}\|_0 \leq k} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \eta\|\boldsymbol{\beta}\|_1. \tag{4.20}$$

For every $\epsilon > 0$, there exists some $\underline{\lambda} = \underline{\lambda}(\epsilon) > 0$ so that for all $\lambda > \underline{\lambda}$,

1. *For every optimal $\boldsymbol{\beta}^*$ to (4.19), there is some $\widehat{\boldsymbol{\beta}}$ so that $\|\boldsymbol{\beta}^* - \widehat{\boldsymbol{\beta}}\|_2 \leq \epsilon$, $\widehat{\boldsymbol{\beta}}$ is feasible to (4.20), and $\widehat{\boldsymbol{\beta}}$ is ϵ-optimal to (4.20).*

2. *Every optimal $\boldsymbol{\beta}^*$ to (4.20) is ϵ-optimal to (4.19).*

Proof. The proof follows a basic continuity argument that is simpler than the one presented below in Theorem 15. For that reason, we do not include a full proof. Observe that the assumptions on g imply that g^{-1} is well-defined on, say, $g([0,1])$. If we let $\epsilon > 0$ and suppose that $\boldsymbol{\beta}^*$ is optimal to (4.19), where $\lambda > \underline{\lambda} := \|\mathbf{y}\|_2^2/(2g(\epsilon/p))$, and if we define $\widehat{\boldsymbol{\beta}}$ to be $\boldsymbol{\beta}^*$ with all but the k largest magnitude entries truncated to zero (ties broken arbitrarily), then $\pi_k^g(\boldsymbol{\beta}^*) \leq \|\mathbf{y}\|_2^2/(2\lambda)$ and $\pi_k^g(\boldsymbol{\beta}^*) = \sum_{i=1}^p g(|\beta_i^* - \widehat{\beta}_i|)$ so that $|\beta_i^* - \widehat{\beta}_i| \leq g^{-1}(\|\mathbf{y}\|_2^2/(2\lambda)) \leq \epsilon/p$ by definition of $\underline{\lambda}$. Hence, $\|\boldsymbol{\beta}^* - \widehat{\boldsymbol{\beta}}\|_1 \leq \epsilon$, and all the other claims essentially follow from this. □

Therefore, any projected penalty π_k^g results in the best subset selection problem (4.1) asymptotically. While the choice of g as the absolute value gives the trimmed Lasso penalty and leads to exact sparsity in the non-asymptotic regime (Theorem 13), Proposition 16 suggests that the projected penalty functions have potential utility in attaining approximately sparse estimators. We will return to the penalties π_k^g again in Section 4.4 to connect the trimmed Lasso to nonconvex penalty methods.

Before concluding this section, we briefly consider a projected penalty function that is different than the trimmed Lasso. As noted above, if $g(x) = x^2/2$, then the corresponding penalty function is the trimmed ridge penalty $\sum_{i>k} |\beta_{(i)}|^2/2$. The estimation procedure is then

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \frac{\lambda}{2}\sum_{i>k} |\beta_{(i)}|^2,$$

or equivalently in decomposed form (*cf.* Section 4.2.3),[6]

$$\min_{\substack{\boldsymbol{\phi},\boldsymbol{\epsilon}:\\ \|\boldsymbol{\phi}\|_0 \leq k}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}(\boldsymbol{\phi} + \boldsymbol{\epsilon})\|_2^2 + \frac{\lambda}{2}\|\boldsymbol{\epsilon}\|_2^2.$$

It is not difficult to see that the variable $\boldsymbol{\epsilon}$ can be eliminated to yield

$$\min_{\|\boldsymbol{\phi}\|_0 \leq k} \frac{1}{2} \|\mathbf{A}(\mathbf{y} - \mathbf{X}\boldsymbol{\phi})\|_2^2, \tag{4.21}$$

where $\mathbf{A} = (\mathbf{I} - \mathbf{X}(\mathbf{X}'\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}')^{1/2}$. It follows that the largest k loadings are found via a modified best subset selection problem under a different loss function—precisely a variant of the ℓ_2 norm. This is in the same spirit of observations made in Section 4.2.3.

Observation 1. *An obvious question is whether the norm in* (4.21) *is genuinely different. Observe that this loss function is the same as the usual* ℓ_2^2 *loss if and only if* $\mathbf{A}'\mathbf{A}$ *is a nonnegative multiple of the identity matrix. It is not difficult to see that this is true iff* $\mathbf{X}'\mathbf{X}$ *is a nonnegative multiple of the identity. In other words, the loss function in* (4.21) *is the same as the usual ridge regression loss if and only if* $\mathbf{X}$ *is (a scalar multiple of) an orthogonal design matrix.*

4.2.5 Other applications of the trimmed Lasso: the (Discrete) Dantzig Selector

The above discussion which pertains to the least squares loss data-fidelity term can be generalized to other loss functions as well. For example, let us consider a data-fidelity term given by the maximal absolute inner product between the features and residuals, given by $\|\mathbf{X}'(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\|_\infty$. An L_1-penalized version of this data-fidelity term, popularly known as the Dantzig Selector [33, 85], is given by the following

[6]Interestingly, if one considers this trimmed ridge regression problem and uses convex envelope techniques [130, 38] to relax the constraint $\|\boldsymbol{\phi}\|_0 \leq k$, the resulting problem takes the form $\min_{\boldsymbol{\phi},\boldsymbol{\epsilon}} \|\mathbf{y} - \mathbf{X}(\boldsymbol{\phi} + \boldsymbol{\epsilon})\|_2^2/2 + \lambda\|\boldsymbol{\epsilon}\|_2^2 + \tau\|\boldsymbol{\phi}\|_1$, a sort of "split" variant of the usual elastic net [167], another popular convex method for sparse modeling.

linear optimization problem:

$$\min_{\boldsymbol{\beta}} \|\mathbf{X}'(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\|_\infty + \mu\|\boldsymbol{\beta}\|_1. \tag{4.22}$$

Estimators found via (4.22) have statistical properties similar to the Lasso. Further, problem (4.22) may be interpreted as an L_1-approximation to the cardinality constrained version:

$$\min_{\|\boldsymbol{\beta}\|_0 \leq k} \|\mathbf{X}'(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\|_\infty, \tag{4.23}$$

that is, the Discrete Dantzig Selector, recently proposed and studied in [111]. The statistical properties of (4.23) are similar to the best-subset selection problem (4.1), but may be more attractive from a computational viewpoint as it relies on mixed integer *linear* optimization as opposed to mixed integer *conic* optimization (see [111]).

The trimmed Lasso penalty can also be applied to the data-fidelity term $\|\mathbf{X}'(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\|_\infty$, leading to the following estimator:

$$\min_{\boldsymbol{\beta}} \|\mathbf{X}'(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\|_\infty + \lambda T_k(\boldsymbol{\beta}) + \mu\|\boldsymbol{\beta}\|_1.$$

Similar to the case of the least squares loss function, the above estimator yields k-sparse solutions for any $\mu > 0$ and for $\lambda > 0$ sufficiently large.[7] While this claim follows *a fortiori* by appealing to properties of the Dantzig selector, it nevertheless highlights how any exact penalty method with a separable penalty function can be turned into a trimmed-style problem which offers direct control over the sparsity level.

4.3 A perspective on robustness

We now turn our attention to a deeper exploration of the robustness properties of the trimmed Lasso. We begin by studying the min-min robust analogue of the min-max robust SLOPE penalty; in doing so, we show under which circumstances this

[7] For the same reason, but instead with the usual Lasso objective, the proof of Theorem 13 could be entirely omitted; yet, it is instructive to see in the proof there that the trimmed Lasso truly does set the *smallest* entries to zero, and not simply all entries (when λ is large) like the Lasso.

analogue is the trimmed Lasso problem. Indeed, in such a regime, the trimmed Lasso can be viewed as an optimistic counterpart to the robust optimization view of the SLOPE penalty. Finally, we turn our attention to an additional min-min robust interpretation of the trimmed Lasso in direct correspondence with the least trimmed squares estimator shown in (4.5), using the ordinary Lasso as our starting point.

4.3.1 The trimmed Lasso as a min-min robust analogue of SLOPE

We begin by reconsidering the uncertainty set that gave rise to the SLOPE penalty via the min-max view of robustness as considered in robust optimization:

$$\mathcal{U}_k^\lambda := \left\{ \boldsymbol{\Delta} : \begin{array}{c} \boldsymbol{\Delta} \text{ has at most } k \text{ nonzero} \\ \text{columns and } \|\boldsymbol{\Delta}_i\|_2 \leq \lambda \ \forall i \end{array} \right\}.$$

As per Proposition 15, the min-max problem (4.9), viz.,

$$\min_{\boldsymbol{\beta}} \max_{\boldsymbol{\Delta} \in \mathcal{U}_k^\lambda} \frac{1}{2} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_2^2$$

is equivalent to the SLOPE-penalized problem

$$\min_{\boldsymbol{\beta}} \frac{1}{2} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda R_{\text{SLOPE}(\tilde{\mathbf{w}})}(\boldsymbol{\beta}). \tag{4.24}$$

for the specific choice of $\tilde{\mathbf{w}}$ with $\tilde{w}_1 = \cdots = \tilde{w}_k = 1$ and $\tilde{w}_{k+1} = \cdots = \tilde{w}_p = 0$.

Let us now consider the form of the min-min robust analogue of the the problem (4.9) for this specific choice of uncertainty set. As per the discussion in Section 4.1, the min-min analogue takes the form of problem (4.8), i.e., a variant of total least squares:

$$\min_{\boldsymbol{\beta}} \min_{\boldsymbol{\Delta} \in \mathcal{U}_k^\lambda} \frac{1}{2} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_2^2,$$

or equivalently as the linearly homogenous problem[8]

$$\min_{\boldsymbol{\beta}} \min_{\boldsymbol{\Delta} \in \mathcal{U}_k^\lambda} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_2. \tag{4.25}$$

It is useful to consider problem (4.25) with an explicit penalization (or regularization) on $\boldsymbol{\beta}$:

$$\min_{\boldsymbol{\beta}} \min_{\boldsymbol{\Delta} \in \mathcal{U}_k^\lambda} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_2 + r(\boldsymbol{\beta}), \tag{4.26}$$

where $r(\cdot)$ is, say, a norm (the use of lowercase is to distinguish from the function R in Section 4.1).

As described in the following theorem, this min-min robustness problem (4.26) is equivalent to the trimmed Lasso problem for specific choices of r.

Theorem 14. *For any k, $\lambda > 0$, and norm r, the problem (4.26) can be rewritten exactly as*

$$\begin{aligned} \min_{\boldsymbol{\beta}} \quad & \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2 + r(\boldsymbol{\beta}) - \lambda \sum_{i=1}^k |\beta_{(i)}| \\ \text{s. t.} \quad & \lambda \sum_{i=1}^k |\beta_{(i)}| \leq \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2. \end{aligned}$$

Proof. We begin by showing that for any $\boldsymbol{\beta}$,

$$\min_{\boldsymbol{\Delta} \in \mathcal{U}_k^\lambda} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_2 = \left(\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2 - \lambda \sum_{i=1}^k |\beta_{(i)}| \right)_+,$$

where $(a)_+ := \max\{0, a\}$. Fix $\boldsymbol{\beta}$ and set $\mathbf{r} = \mathbf{y} - \mathbf{X}\boldsymbol{\beta}$. We assume without loss of generality that $\mathbf{r} \neq \mathbf{0}$ and that $\boldsymbol{\beta} \neq \mathbf{0}$. For any $\boldsymbol{\Delta}$, note that $\|\mathbf{r} - \boldsymbol{\Delta}\boldsymbol{\beta}\|_2 \geq 0$ and $\|\mathbf{r} - \boldsymbol{\Delta}\boldsymbol{\beta}\|_2 \geq \|\mathbf{r}\|_2 - \|\boldsymbol{\Delta}\boldsymbol{\beta}\|_2$ by the reverse triangle inequality. Now observe that for $\boldsymbol{\Delta} \in \mathcal{U}_k^\lambda$,

$$\|\boldsymbol{\Delta}\boldsymbol{\beta}\|_2 \leq \sum_i |\beta_i| \|\boldsymbol{\Delta}_i\|_2 \leq \sum_{i=1}^k \lambda |\beta_{(i)}|.$$

Therefore, $\|\mathbf{r} - \boldsymbol{\Delta}\boldsymbol{\beta}\|_2 \geq \left(\|\mathbf{r}\|_2 - \lambda \sum_{i=1}^k |\beta_{(i)}| \right)_+$. Let $I \subseteq \{1, \ldots, p\}$ be a set of k

[8] In what follows, the linear homogeneity is useful primarily for simplicity of analysis, *cf.* [19, ch. 12]. Indeed, the conversion to linear homogeneous functions is often hidden in equivalence results like Proposition 15.

indices which correspond to the k largest entries of $\boldsymbol{\beta}$ (if $|\beta_{(k)}| = |\beta_{(k+1)}|$, break ties arbitrarily). Define $\boldsymbol{\Delta} \in \mathcal{U}_k^\lambda$ as the matrix whose ith column is

$$\begin{cases} \underline{\lambda}\,\mathrm{sgn}(\beta_i)\mathbf{r}/\|\mathbf{r}\|_2, & i \in I \\ 0, & i \notin I, \end{cases}$$

where $\underline{\lambda} = \min\left\{\lambda, \|\mathbf{r}\|_2 / \left(\sum_{i=1}^k |\beta_{(i)}|\right)\right\}$. It is easy to verify that $\boldsymbol{\Delta} \in \mathcal{U}_k^\lambda$ and $\|\mathbf{r} - \boldsymbol{\Delta}\boldsymbol{\beta}\|_2 = \left(\|\mathbf{r}\|_2 - \lambda \sum_{i=1}^k |\beta_{(i)}|\right)_+$. Combined with the lower bound, we have

$$\min_{\boldsymbol{\Delta} \in \mathcal{U}_k^\lambda} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_2 = \left(\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2 - \lambda \sum_{i=1}^k |\beta_{(i)}|\right)_+,$$

which completes the first claim.

It follows that the problem (4.26) can be rewritten exactly as

$$\min_{\boldsymbol{\beta}} \left(\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2 - \lambda \sum_{i=1}^k |\beta_{(i)}|\right)_+ + r(\boldsymbol{\beta}). \tag{4.27}$$

To finish the proof of the theorem, it suffices to show that if $\boldsymbol{\beta}^*$ is a solution to (4.27), then

$$\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^*\|_2 - \lambda \sum_{i=1}^k |\beta^*_{(i)}| \geq 0.$$

If this is not true, then $\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^*\|_2 - \lambda \sum_{i=1}^k |\beta^*_{(i)}| < 0$ and so $\boldsymbol{\beta}^* \neq \mathbf{0}$. However, this implies that for $1 > \epsilon > 0$ sufficiently small, $\boldsymbol{\beta}_\epsilon := (1-\epsilon)\boldsymbol{\beta}^*$ satisfies $\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}_\epsilon\|_2 - \lambda \sum_{i=1}^k |(\beta_\epsilon)_{(i)}| < 0$. This in turn implies that

$$\begin{aligned} &\left(\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}_\epsilon\|_2 - \lambda \textstyle\sum_{i=1}^k |(\beta_\epsilon)_{(i)}|\right)_+ + r(\boldsymbol{\beta}_\epsilon) \\ &< \left(\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^*\|_2 - \lambda \textstyle\sum_{i=1}^k |\beta^*_{(i)}|\right)_+ + r(\boldsymbol{\beta}^*), \end{aligned}$$

which contradicts the optimality of $\boldsymbol{\beta}^*$. (We have used the absolute homogeneity of the norm r and that $\boldsymbol{\beta}^* \neq \mathbf{0}$.) Hence, any optimal $\boldsymbol{\beta}^*$ to (4.27) necessarily satisfies $\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^*\|_2 - \lambda \sum_{i=1}^k |\beta^*_{(i)}| \geq 0$ and so the desired results follows. □

N.B. The assumption that r is a norm can be relaxed somewhat (as is clear in the proof), although the full generality is not necessary for our purposes.

This leads in part to the following:

Theorem 15. *For the choice of $r(\boldsymbol{\beta}) = \tau\|\boldsymbol{\beta}\|_1$, where $\tau > \lambda$, the problem (4.26) is*

$$\begin{aligned} \min_{\boldsymbol{\beta}} \quad & \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2 + (\tau - \lambda)\|\boldsymbol{\beta}\|_1 + \lambda T_k(\boldsymbol{\beta}) \\ \text{s. t.} \quad & \lambda \sum_{i=1}^{k} |\beta_{(i)}| \leq \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2. \end{aligned} \tag{4.28}$$

In particular, when $\lambda > 0$ is small, it is approximately equal (in a precise sense) to the trimmed Lasso problem

$$\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2 + (\tau - \lambda)\|\boldsymbol{\beta}\|_1 + \lambda T_k(\boldsymbol{\beta}). \tag{4.29}$$

Namely, for all $\epsilon > 0$, there exists some $\underline{\lambda} = \underline{\lambda}(\epsilon) > 0$ so that whenever $\lambda \in (0, \underline{\lambda})$,

1. *Every optimal $\boldsymbol{\beta}^*$ to (4.28) is ϵ-optimal to (4.29).*
2. *For every optimal $\boldsymbol{\beta}^*$ to (4.29), there is some $\widehat{\boldsymbol{\beta}}$ so that $\|\boldsymbol{\beta}^* - \widehat{\boldsymbol{\beta}}\|_2 \leq \epsilon$, $\widehat{\boldsymbol{\beta}}$ is feasible to (4.28), and $\widehat{\boldsymbol{\beta}}$ is ϵ-optimal to (4.28).*

Proof. Fix $\tau > 0$ throughout. We assume without loss of generality that $\mathbf{y} \neq \mathbf{0}$, as otherwise the claim is obvious. We will prove the second claim first, as it essentially implies the first.

Let us consider two situations. In particular, we consider whether there exists a nonzero optimal solution to

$$\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2 + \tau\|\boldsymbol{\beta}\|_1. \tag{4.30}$$

Case 1—existence of nonzero optimal solution to (4.30):

We first consider the case when there exists a nonzero solution to problem (4.30). We show a few lemmata:

1. We first show that the norm of solutions to (4.29) are uniformly bounded away from zero, independent of λ. To proceed, let $\widehat{\boldsymbol{\beta}}$ be any nonzero optimal solution to (4.30). Observe that if $\boldsymbol{\beta}^*$ is optimal to (4.29), then

$$\begin{aligned}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^*\|_2 + (\tau - \lambda)\|\boldsymbol{\beta}^*\|_1 + \lambda T_k(\boldsymbol{\beta}^*) & \\ \leq \|\mathbf{y} - \mathbf{X}\widehat{\boldsymbol{\beta}}\|_2 + (\tau - \lambda)\|\widehat{\boldsymbol{\beta}}\|_1 + \lambda T_k(\widehat{\boldsymbol{\beta}}) & \\ \leq \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^*\|_2 + \tau\|\boldsymbol{\beta}^*\|_1 - \lambda\|\widehat{\boldsymbol{\beta}}\|_1 + \lambda T_k(\widehat{\boldsymbol{\beta}}), &\end{aligned}$$

implying that $\|\widehat{\boldsymbol{\beta}}\|_1 - T_k(\widehat{\boldsymbol{\beta}}) \leq \|\boldsymbol{\beta}^*\|_1 - T_k(\boldsymbol{\beta}^*)$. In other words, $\sum_{i=1}^k |\widehat{\beta}_{(i)}| \leq \sum_{i=1}^k |\beta^*_{(i)}| \leq \|\boldsymbol{\beta}^*\|_1$. Using the fact that $\widehat{\boldsymbol{\beta}} \neq \mathbf{0}$, we have that any solution $\boldsymbol{\beta}^*$ to (4.29) has strictly positive norm:

$$\|\boldsymbol{\beta}^*\|_1 \geq C > 0,$$

where $C := \sum_{i=1}^k |\widehat{\beta}_{(i)}|$ is a universal constant depending only on τ (and not λ).

2. We now upper bound the norm of solutions to (4.29). In particular, if $\boldsymbol{\beta}^*$ is optimal to (4.29), then

$$\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^*\|_2 + (\tau - \lambda)\|\boldsymbol{\beta}^*\|_1 + \lambda T_k(\boldsymbol{\beta}^*) \leq \|\mathbf{y}\|_2 + 0 + 0 = \|\mathbf{y}\|_2,$$

and so $\|\boldsymbol{\beta}^*\|_1 \leq \|\mathbf{y}\|_2/(\tau - \lambda)$. (This bound is not uniform in λ, but if we restrict our attention to, say $\lambda \leq \tau/2$, it is.)

3. We now lower bound the loss for scaled version of optimal solutions. In particular, if $\sigma \in [0, 1]$ and $\boldsymbol{\beta}^*$ is optimal to (4.29), then by optimality we have that

$$\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^*\|_2 + (\tau - \lambda)\|\boldsymbol{\beta}^*\|_1 + \lambda T_k(\boldsymbol{\beta}^*) \leq \|\mathbf{y} - \sigma\mathbf{X}\boldsymbol{\beta}^*\|_2 + (\tau - \lambda)\sigma\|\boldsymbol{\beta}^*\|_1 + \lambda\sigma T_k(\boldsymbol{\beta}^*),$$

which in turn implies that

$$\begin{aligned}\|\mathbf{y} - \sigma\mathbf{X}\boldsymbol{\beta}^*\|_2 &\geq \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^*\|_2 + (\tau - \lambda)(1-\sigma)\|\boldsymbol{\beta}^*\|_1 + \lambda(1-\sigma)T_k(\boldsymbol{\beta}^*) \\ &\geq \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^*\|_2 + (\tau - \lambda)(1-\sigma)C \geq (\tau - \lambda)(1-\sigma)C\end{aligned}$$

by combining with the first observation.

Using these, we are now ready to proceed. Let $\epsilon > 0$; we assume without loss of generality that $\epsilon < 2\|\mathbf{y}\|_2/\tau$. Let

$$\underline{\lambda} := \min\left\{\frac{\epsilon\tau^3 C}{4\|\mathbf{y}\|_2(2\|\mathbf{y}\|_2 - \epsilon\tau)}, \frac{\tau}{2}\right\}.$$

Fix $\lambda \in (0, \underline{\lambda})$ and let $\boldsymbol{\beta}^*$ be any optimal solution to (4.29). Define

$$\sigma := \left(1 - \frac{\epsilon\tau}{2\|\mathbf{y}\|_2}\right) \quad \text{and} \quad \widehat{\boldsymbol{\beta}} := \sigma\boldsymbol{\beta}^*.$$

We claim that $\widehat{\boldsymbol{\beta}}$ satisfies the desired requirements of the theorem:

1. We first argue that $\|\boldsymbol{\beta}^* - \widehat{\boldsymbol{\beta}}\|_2 \leq \epsilon$. Observe that

$$\|\boldsymbol{\beta}^* - \widehat{\boldsymbol{\beta}}\|_2 = \epsilon\tau\|\boldsymbol{\beta}^*\|_2/(2\|\mathbf{y}\|_2) \leq \epsilon\tau\|\boldsymbol{\beta}^*\|_1/(2\|\mathbf{y}\|_2) \leq \epsilon\tau\|\mathbf{y}\|_2/(2\|\mathbf{y}\|_2(\tau-\lambda)) \leq \epsilon.$$

2. We now show that $\widehat{\boldsymbol{\beta}}$ is feasible to (4.28). This requires us to argue that $\lambda\sum_{i=1}^k |\widehat{\beta}_{(i)}| \leq \|\mathbf{y} - \mathbf{X}\widehat{\boldsymbol{\beta}}\|_2$. Yet,

$$\begin{aligned}\lambda\sum_{i=1}^k |\widehat{\beta}_{(i)}| \leq \lambda\|\widehat{\boldsymbol{\beta}}\|_1 = \lambda\sigma\|\boldsymbol{\beta}^*\|_1 \leq 2\lambda\sigma\|\mathbf{y}\|_2/\tau \leq \frac{\tau}{2}(1-\sigma)C \\ \leq (\tau - \lambda)(1-\sigma)C \leq \|\mathbf{y} - \sigma\mathbf{X}\boldsymbol{\beta}^*\|_2 = \|\mathbf{y} - \mathbf{X}\widehat{\boldsymbol{\beta}}\|_2,\end{aligned}$$

as desired. The only non-obvious step is the inequality $2\lambda\sigma\|\mathbf{y}\|_2/\tau \leq \tau(1-\sigma)C/2$, which follows from algebraic manipulations using the definitions of σ and $\underline{\lambda}$.

3. Finally, we show that $\widehat{\boldsymbol{\beta}}$ is $(\epsilon\|\mathbf{X}\|_2)$-optimal to (4.28). Indeed, because $\boldsymbol{\beta}^*$ is optimal to (4.29) which necessarily lowers bound problem (4.28), we have that the objective value gap between $\widehat{\boldsymbol{\beta}}$ and an optimal solution to (4.28) is at most

$$\begin{aligned}
&\|\mathbf{y} - \sigma\mathbf{X}\boldsymbol{\beta}^*\|_2 - \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^*\|_2 + (\tau - \lambda)(\sigma - 1)\|\boldsymbol{\beta}^*\|_1 + \lambda(\sigma - 1)T_k(\boldsymbol{\beta}^*) \\
&\leq (1 - \sigma)\|\mathbf{X}\boldsymbol{\beta}^*\|_2 + 0 + 0 \leq (1 - \sigma)\|\mathbf{X}\|_2\|\boldsymbol{\beta}^*\|_2 \leq 2(1 - \sigma)\|\mathbf{X}\|_2\|\mathbf{y}\|_2/\tau \\
&= 2\epsilon\tau/(2\|\mathbf{y}\|_2)\|\mathbf{X}\|_2\|\mathbf{y}\|_2/\tau = \epsilon\|\mathbf{X}\|_2.
\end{aligned}$$

As the choice of $\epsilon > 0$ was arbitrary, this completes the proof of claim 2 in the theorem in the case when $\mathbf{0}$ is not a solution to (4.30).

Case 2—no nonzero optimal solution to (4.30):

In the case when there is no nonzero optimal solution to (4.30), $\mathbf{0}$ is optimal and it is the only optimal point. Our analysis will be similar to the previous approach, with the key difference being in how we lower bound the quantity $\|\mathbf{y} - \sigma\mathbf{X}\boldsymbol{\beta}^*\|_2$ where $\boldsymbol{\beta}^*$ is optimal to (4.29). Again, we have several lemmata:

1. As before, if $\boldsymbol{\beta}^*$ is optimal to (4.29), then $\|\boldsymbol{\beta}^*\|_1 \leq \|\mathbf{y}\|_2/(\tau - \lambda)$.

2. We now lower bound the quantity $\|\mathbf{y} - \sigma\mathbf{X}\boldsymbol{\beta}^*\|_2$, where $\boldsymbol{\beta}^*$ is optimal to (4.29) and $\sigma \in [0, 1]$. As such, consider the function

$$f(\sigma) := \|\mathbf{y} - \sigma\mathbf{X}\boldsymbol{\beta}^*\|_2 + \sigma\tau\|\boldsymbol{\beta}^*\|_1.$$

Because f is convex in σ and the unique optimal solution to (4.30) is $\mathbf{0}$, we have that

$$f(\sigma) \geq f(0) + \sigma f'(0) \quad \forall \sigma \in [0, 1] \quad \text{and} \quad f'(0) \geq 0.$$

(It is not difficult to argue that f is differentiable at 0.) An elementary computation shows that $f'(0) = \tau\|\boldsymbol{\beta}^*\|_1 - \langle \mathbf{y}, \mathbf{X}\boldsymbol{\beta}^* \rangle / \|\mathbf{y}\|_2$. Therefore, we have that

$$\|\mathbf{y} - \sigma\mathbf{X}\boldsymbol{\beta}^*\|_2 + \sigma\tau\|\boldsymbol{\beta}^*\|_1 \geq \|\mathbf{y}\|_2 + \sigma\left(\tau\|\boldsymbol{\beta}^*\|_1 - \langle \mathbf{y}, \mathbf{X}\boldsymbol{\beta}^* \rangle / \|\mathbf{y}\|_2\right),$$

implying that

$$\|\mathbf{y}-\sigma\mathbf{X}\boldsymbol{\beta}^*\|_2 \geq \|\mathbf{y}\|_2-\sigma\langle\mathbf{y},\mathbf{X}\boldsymbol{\beta}^*\rangle/\|\mathbf{y}\|_2 \geq \|\mathbf{y}\|_2-\sigma\tau\|\boldsymbol{\beta}^*\|_1 \geq \|\mathbf{y}\|_2-\sigma\tau\|\mathbf{y}\|_2/(\tau-\lambda),$$

with the final step following by an application of the previous lemma.

We are now ready to proceed. Let $\epsilon > 0$; we assume without loss of generality that $\epsilon < 2\|\mathbf{y}\|_2/\tau$. Let

$$\underline{\lambda} := \min\left\{\frac{\epsilon\tau^2}{4\|\mathbf{y}\|_2 - \epsilon\tau}, \frac{\tau}{2}\right\}.$$

Fix $\lambda \in (0, \underline{\lambda})$ and let $\boldsymbol{\beta}^*$ be any optimal solution to (4.29). Define

$$\sigma := \left(1 - \frac{\epsilon\tau}{2\|\mathbf{y}\|_2}\right) \quad \text{and} \quad \widehat{\boldsymbol{\beta}} := \sigma\boldsymbol{\beta}^*.$$

We claim that $\widehat{\boldsymbol{\beta}}$ satisfies the desired requirements:

1. The proof of the claim that $\|\boldsymbol{\beta}^* - \widehat{\boldsymbol{\beta}}\|_2 \leq \epsilon$ is exactly as before.

2. We now show that $\widehat{\boldsymbol{\beta}}$ is feasible to (4.28), which requires a different proof. Again this requires us to argue that $\lambda\sum_{i=1}^k |\widehat{\beta}_{(i)}| \leq \|\mathbf{y} - \mathbf{X}\widehat{\boldsymbol{\beta}}\|_2$. Yet,

$$\begin{aligned}\lambda\sum_{i=1}^k |\widehat{\beta}_{(i)}| \leq \lambda\|\widehat{\boldsymbol{\beta}}\|_1 = \lambda\sigma\|\boldsymbol{\beta}^*\|_1 \leq \lambda\sigma\|\mathbf{y}\|_2/(\tau-\lambda) \leq \|\mathbf{y}\|_2 - \sigma\tau\|\mathbf{y}\|_2/(\tau-\lambda)\\ \leq \|\mathbf{y} - \sigma\mathbf{X}\boldsymbol{\beta}^*\|_2 = \|\mathbf{y} - \mathbf{X}\widehat{\boldsymbol{\beta}}\|_2,\end{aligned}$$

   as desired. The only non-obvious step is the inequality $\lambda\sigma\|\mathbf{y}\|_2/(\tau-\lambda) \leq \|\mathbf{y}\|_2 - \sigma\tau\|\mathbf{y}\|_2/(\tau-\lambda)$, which follows from algebraic manipulations using the definitions of σ and $\underline{\lambda}$.

3. Finally, the proof that $\widehat{\boldsymbol{\beta}}$ is $(\epsilon\|\mathbf{X}\|_2)$-optimal to (4.28) follows in the same way as before.

Therefore, we conclude that in the case when $\mathbf{0}$ is the unique optimal solution to (4.30), then again we have that the claim 2 of the theorem holds.

Finally, we show that claim 1 holds: any solution $\boldsymbol{\beta}^*$ to (4.28) is ϵ-optimal to (4.29). This follows by letting $\overline{\boldsymbol{\beta}}$ be any optimal solution to (4.29). By applying the entire argument above, we know that the objective value of some $\widehat{\boldsymbol{\beta}}$, feasible to (4.28) and close to $\overline{\boldsymbol{\beta}}$, is within ϵ of the optimal objective value of (4.28), i.e., the objective value of $\boldsymbol{\beta}^*$, and within ϵ of the objective value of (4.29), i.e., the objective value of $\overline{\boldsymbol{\beta}}$. This completes the proof. □

In short, the key complication is that the quantity $\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^*\|_2$ does not need to be uniformly bounded away from zero for solutions $\boldsymbol{\beta}^*$ to problem (4.29). This is part of the complication of working with the homogeneous form of the trimmed Lasso problem. For a concrete example, if one considers the homogeneous Lasso problem with $p = n = 1$, $\mathbf{y} = (1)$, and $\mathbf{X} = (1)$, then the homogeneous Lasso problem $\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2 + \eta\|\boldsymbol{\beta}\|_1$ is

$$\min_{\beta} |1 - \beta| + \eta|\beta|.$$

For $\eta \in [0, 1]$, $\beta^* = 1$ is an optimal solution to this problem with corresponding error $\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^*\| = 0$. If we make an assumption about the behavior of $\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^*\|$, then we do not need the setup as shown above.

Interpreting Theorem 15

In words, the min-min problem (4.26) (with an L_1 regularization on $\boldsymbol{\beta}$) can be written as a variant of a trimmed Lasso problem, subject to an additional constraint. It is instructive to consider both the objective and the constraint of problem (4.28). To begin, the objective has a combined penalty on $\boldsymbol{\beta}$ of $(\tau - \lambda)\|\boldsymbol{\beta}\|_1 + \lambda T_k(\boldsymbol{\beta})$. This can be thought of as the more general form of the penalty T_k. Namely, one can consider the penalty $T_{\mathbf{x}}$ (with $0 \leq x_1 \leq x_2 \leq \cdots \leq x_p$ fixed) defined as

$$T_{\mathbf{x}}(\boldsymbol{\beta}) := \sum_{i=1}^{p} x_i |\beta_{(i)}|.$$

In this notation, the objective of (4.28) can be rewritten as $\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2 + T_{\mathbf{x}}(\boldsymbol{\beta})$, with

$$\mathbf{x} = (\underbrace{\tau - \lambda, \ldots, \tau - \lambda}_{k \text{ times}}, \underbrace{\tau, \ldots, \tau}_{p-k \text{ times}}).$$

In terms of the constraint of problem (4.28), note that it takes the form of a model-fitting constraint: namely, λ controls a trade-off between model fit $\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2$ and model complexity measured via the SLOPE norm $\sum_{i=1}^{k} |\beta_{(i)}|$.

Having described the structure of problem (4.28), a few remarks are in order. First of all, the trimmed Lasso problem (with an additional L_1 penalty on $\boldsymbol{\beta}$) can be interpreted as (a close approximation to) a min-min robust problem, at least in the regime when λ is small; this provides an interesting contrast to the sparse-modeling regime when λ is large (*cf.* Theorem 13). Moreover, the trimmed Lasso is a min-min robust problem in a way that is the *optimistic* analogue of its min-max counterpart, namely, the SLOPE-penalized problem (4.24). Finally, Theorem 14 gives a natural representation of the trimmed Lasso problem in a way that directly suggests why methods from difference-of-convex optimization [4] are relevant (see Section 4.5).

The general SLOPE penalty

Let us briefly remark upon SLOPE in its most general form (with general $\mathbf{w}$); again we will see that this leads to a more general trimmed Lasso as its (approximate) min-min counterpart. In its most general form, the SLOPE-penalized problem (4.24) can be written as the min-max robust problem (4.9) with choice of uncertainty set

$$\mathcal{U}_{\mathbf{w}}^{\lambda} = \left\{ \boldsymbol{\Delta} : \|\boldsymbol{\Delta}\boldsymbol{\phi}\|_2 \leq \lambda \sum_i w_i |\phi_{(i)}| \ \forall \boldsymbol{\phi} \right\}$$

(see Appendix C.1). In this case, the penalized, homogenized min-min robust counterpart, analogous to problem (4.26), can be written as follows:

Proposition 18. *For any k, $\lambda > 0$, and norm r, the problem*

$$\min_{\boldsymbol{\beta}} \min_{\boldsymbol{\Delta} \in \mathcal{U}_{\mathbf{w}}^{\lambda}} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_2 + r(\boldsymbol{\beta}) \tag{4.31}$$

can be rewritten exactly as

$$\begin{aligned} \min_{\boldsymbol{\beta}} \quad & \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2 + r(\boldsymbol{\beta}) - \lambda R_{\mathrm{SLOPE}(\mathbf{w})}(\boldsymbol{\beta}) \\ \text{s. t.} \quad & \lambda R_{\mathrm{SLOPE}(\mathbf{w})}(\boldsymbol{\beta}) \leq \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2. \end{aligned}$$

For the choice of $r(\boldsymbol{\beta}) = \tau\|\boldsymbol{\beta}\|_1$*, where* $\tau > \lambda w_1$*, the problem* (4.31) *is*

$$\begin{aligned} \min_{\boldsymbol{\beta}} \quad & \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2 + T_{\tau\mathbf{1}-\lambda\mathbf{w}}(\boldsymbol{\beta}) \\ \text{s. t.} \quad & \lambda R_{\mathrm{SLOPE}(\mathbf{w})}(\boldsymbol{\beta}) \leq \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2. \end{aligned}$$

In particular, when $\lambda > 0$ *is sufficiently small, problem* (4.31) *is approximately equal to the generalized trimmed Lasso problem*

$$\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2 + T_{\tau\mathbf{1}-\lambda\mathbf{w}}(\boldsymbol{\beta}).$$

Proof. The proof is entirely analogous to that of Theorems 14 and 15 and is omitted. □

Put plainly, the general form of the SLOPE penalty leads to a generalized form of the trimmed Lasso, precisely as was true for the simplified version considered in Theorem 14.

4.3.2 Another min-min interpretation

We close our discussion of robustness by considering another min-min representation of the trimmed Lasso. We use the ordinary Lasso problem as our starting point and show how a modification in the same spirit as the min-min robust least trimmed squares estimator in (4.5) leads directly to the trimmed Lasso.

To proceed, we begin with the usual Lasso problem

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda\|\boldsymbol{\beta}\|_1. \tag{4.32}$$

As per Proposition 14, this problem is equivalent to the min-max robust problem

(4.9) with uncertainty set $\mathcal{U} = \mathcal{L}^\lambda = \{\mathbf{\Delta} : \|\mathbf{\Delta}_i\|_2 \leq \lambda \ \forall i\}$:

$$\min_{\boldsymbol{\beta}} \max_{\mathbf{\Delta} \in \mathcal{L}^\lambda} \frac{1}{2}\|\mathbf{y} - (\mathbf{X} + \mathbf{\Delta})\boldsymbol{\beta}\|_2^2. \tag{4.33}$$

In this view, the usual Lasso (4.32) can be thought of as a least squares method which takes into account certain feature-wise adversarial perturbations of the matrix $\mathbf{X}$. The net result is that the adversarial approach penalizes all loadings equally (with coefficient λ).

Using this setup and Theorem 13, we can re-express the trimmed Lasso problem ($\text{TL}_{\lambda,k}$) in the equivalent min-min form

$$\min_{\boldsymbol{\beta}} \min_{\substack{I \subseteq \{1,\ldots,p\}: \\ |I| = p-k}} \max_{\mathbf{\Delta} \in \mathcal{L}_I^\lambda} \frac{1}{2}\|\mathbf{y} - (\mathbf{X} + \mathbf{\Delta})\boldsymbol{\beta}\|_2^2, \tag{4.34}$$

where $\mathcal{L}_I^\lambda \subseteq \mathcal{L}^\lambda$ requires that the columns of $\mathbf{\Delta} \in \mathcal{L}_I^\lambda$ are supported on I:

$$\mathcal{L}_I^\lambda = \{\mathbf{\Delta} : \|\mathbf{\Delta}_i\|_2 \leq \lambda \ \forall i, \ \mathbf{\Delta}_i = \mathbf{0} \ \forall i \notin I\}.$$

While the adversarial min-max approach in problem (4.33) would attempt to "corrupt" all p columns of $\mathbf{X}$, in estimating $\boldsymbol{\beta}$ we have the power to optimally discard k out of the p corruptions to the columns (corresponding to I^c). In this sense, the trimmed Lasso in the min-min robust form (4.34) acts in a similar spirit to the min-min, robust-statistical least trimmed squares estimator shown in problem (4.6).

4.4 Connection to nonconvex penalty methods

In this section, we explore the connection between the trimmed Lasso and existing, popular nonconvex (component-wise separable) penalty functions used for sparse modeling. We begin in Section 4.4.1 with a brief overview of existing approaches. In Section 4.4.2 we then highlight how these relate to the trimmed Lasso, making the connection more concrete with examples in Section 4.4.3. Then in Section 4.4.4 we exactly characterize the connection between the trimmed Lasso and the clipped

Lasso [166]. In doing so, we show that the trimmed Lasso subsumes the clipped Lasso; further, we provide a necessary and sufficient condition for when the containment is strict. Finally, in Section 4.4.5 we comment on the special case of unbounded penalty functions.

4.4.1 Setup and Overview

Our focus throughout will be the penalized M-estimation problem of the form

$$\min_{\boldsymbol{\beta}} L(\boldsymbol{\beta}) + \sum_{i=1}^{p} \rho(|\beta_i|; \mu, \gamma), \tag{4.35}$$

where μ represents a (continuous) parameter controlling the desired level of sparsity of $\boldsymbol{\beta}$ and γ is a parameter controlling the quality of the approximation of the indicator function $I\{|\beta| > 0\}$. A variety of nonconvex penalty functions and their description in this format is shown in Table 4.1 (for a general discussion, see [165]). In particular, for each of these functions we observe that

$$\lim_{\gamma \to \infty} \rho(|\beta|; \mu, \gamma) = \mu \cdot I\{|\beta| > 0\}.$$

It is particularly important to note the *separable* nature of the penalty functions appearing in (4.35)—namely, each coordinate β_i is penalized (via ρ) independently of the other coordinates.

Our primary focus will be on the bounded penalty functions (clipped Lasso, MCP, and SCAD), all of which take the form

$$\rho(|\beta|; \mu, \gamma) = \mu \min\{g(|\beta|; \mu, \gamma), 1\}, \tag{4.36}$$

where g is an increasing function of $|\beta|$. We will show that in this case, the problem (4.35) can be rewritten exactly as an estimation problem with a (non-separable)

trimmed penalty function:

$$\min_{\boldsymbol{\beta}} L(\boldsymbol{\beta}) + \mu \sum_{i=\ell+1}^{p} g(|\beta_{(i)}|) \tag{4.37}$$

for some $\ell \in \{0, 1, \ldots, p\}$ (note the appearance of the projected penalties π_k^g as considered in Section 4.2.4). In the process of doing so, we will also show that, in general, (4.37) cannot be solved via the separable-penalty estimation approach of (4.35), and so the trimmed estimation problem leads to a richer class of models. Throughout we will often refer to (4.37) (taken generically over all choices of ℓ) as the *trimmed counterpart* of the separable estimation problem (4.35).

4.4.2 Reformulating the problem (4.35)

Let us begin by considering penalty functions ρ of the form (4.36) with g a nonnegative, increasing function of $|\beta|$. Observe that for any $\boldsymbol{\beta}$ we can rewrite $\sum_{i=1}^{p} \min\{g(|\beta_i|), 1\}$ as

$$\begin{aligned}
&\min \left\{ \sum_{i=1}^{p} g(|\beta_{(i)}|), 1 + \sum_{i=2}^{p} g(|\beta_{(i)}|), \ldots, p - 1 + g(|\beta_{(p)}|), p \right\} \\
&= \min_{\ell \in \{0,\ldots,p\}} \left\{ \ell + \sum_{i>\ell} g(|\beta_{(i)}|) \right\}.
\end{aligned}$$

It follows that (4.35) can be rewritten *exactly* as

$$\min_{\substack{\boldsymbol{\beta}, \\ \ell \in \{0,\ldots,p\}}} \left(L(\boldsymbol{\beta}) + \mu \sum_{i>\ell} g(|\beta_{(i)}|) + \mu\ell \right) \tag{4.38}$$

An immediate consequence is the following theorem:

Theorem 16. *If $\boldsymbol{\beta}^*$ is an optimal solution to (4.35), where $\rho(|\beta|; \mu, \gamma) = \mu \min\{g(|\beta|; \mu, \gamma), 1\}$, then there exists some $\ell^* \in \{0, \ldots, p\}$ so that $\boldsymbol{\beta}^*$ is optimal to its trimmed counterpart*

$$\min_{\boldsymbol{\beta}} L(\boldsymbol{\beta}) + \mu \sum_{i>\ell^*} g(|\beta_{(i)}|).$$

Name	Definition
Clipped Lasso [166]	$\mu \min\{\gamma\|\beta\|, 1\}$
MCP [164]	$\mu \min\{g_1(\|\beta\|), 1\}$
SCAD [60]	$\mu \min\{g_2(\|\beta\|), 1\}$
L_q $(0 < q < 1)$ [65, 67]	$\mu\|\beta\|^{1/\gamma}$
Log [67]	$\mu\log(\gamma\|\beta\| + 1)/\log(\gamma + 1)$

(a) Penalty functions

Function	Value	Conditional on
$g_1(\|\beta\|)$	$2\gamma\|\beta\| - \gamma^2\beta^2$	$\|\beta\| \leq 1/\gamma$
	1	$\|\beta\| > 1/\gamma$
$g_2(\|\beta\|)$	$\|\beta\|/(\gamma\mu)$	$\|\beta\| \leq 1/\gamma$
	$\dfrac{\beta^2 + (2/\gamma - 4\mu\gamma)\|\beta\| + 1/\gamma^2}{4\mu - 4\mu^2\gamma^2}$	$1/\gamma < \|\beta\| \leq 2\mu\gamma - 1/\gamma$
	1	$\|\beta\| > 2\mu\gamma - 1/\gamma$

(b) Auxiliary functions

Table 4.1: Nonconvex penalty functions $\rho(|\beta|; \mu, \gamma)$ represented as in (4.35). The precise parametric representation is different than their original presentation but they are equivalent. We have taken care to normalize the different penalty functions so that μ is the sparsity parameter and γ corresponds to the approximation of the indicator $I\{|\beta| > 0\}$. For SCAD, it is usually recommended to set $2\mu > 3/\gamma^2$. For L_q, $q = 1/\gamma$.

In particular, the choice of $\ell^* = |\{i : g(|\beta_i^*|) \geq 1\}|$ *suffices. Conversely, if* $\boldsymbol{\beta}^*$ *is an optimal solution to* (4.38)*, then* $\boldsymbol{\beta}^*$ *in an optimal solution to* (4.35)*.*

It follows that the estimation problem (4.35), which decouples each loading β_i in the penalty function, can be solved using "trimmed" estimation problems of the form (4.37) with a trimmed penalty function that couples the loadings and only penalizes the $p - \ell^*$ smallest. Because the trimmed penalty function is generally nonconvex by nature, we will focus on comparing it with other nonconvex penalties for the remainder of the section.

4.4.3 Trimmed reformulation examples

We now consider the structure of the estimation problem (4.35) and the corresponding trimmed estimation problem for the clipped Lasso and MCP penalties. We use the least squares loss throughout.

Clipped Lasso

The clipped (or capped, or truncated) Lasso penalty [166, 140] takes the component-wise form

$$\rho(|\beta|; \mu, \gamma) = \mu \min\{\gamma|\beta|, 1\}.$$

Therefore, in our notation, g is a multiple of the absolute value function. A plot of ρ is shown in Figure 4-1a. In this case, the estimation problem with ℓ_2^2 loss is

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \mu \sum_i \min\{\gamma|\beta_i|, 1\}. \tag{4.39}$$

It follows that the corresponding trimmed estimation problem (*cf.* Theorem 16) is exactly the trimmed Lasso problem studied earlier, namely,

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \mu\gamma T_k(\boldsymbol{\beta}). \tag{4.40}$$

A distinct advantage of the trimmed Lasso formulation (4.40) over the traditional clipped Lasso formulation (4.39) is that it offers direct control over the desired level of sparsity vis-à-vis the discrete parameter k. We perform a deeper analysis of the two problems in Section 4.4.4.

MCP

The MCP penalty takes the component-wise form

$$\rho(|\beta|;\mu,\gamma) = \mu \min\{g(|\beta|), 1\}$$

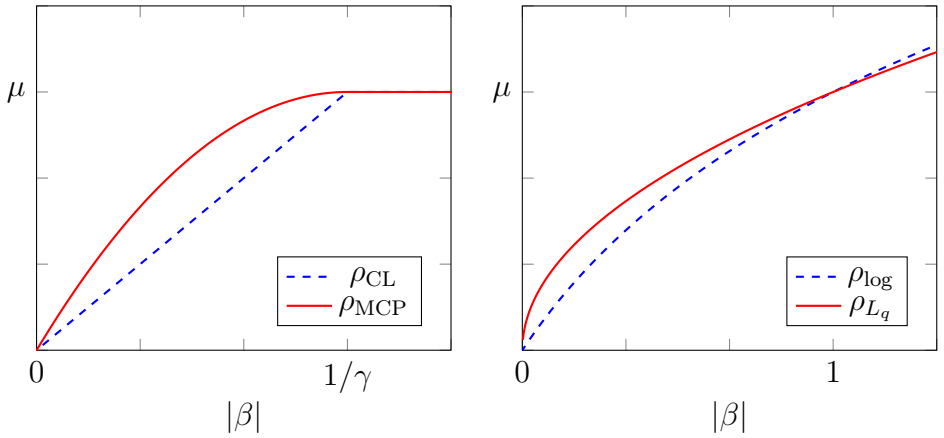
where g is any function with $g(|\beta|) = 2\gamma|\beta| - \gamma^2\beta^2$ whenever $|\beta| \leq 1/\gamma$ and $g(|\beta|) \geq 1$ whenever $|\beta| > 1/\gamma$. An example of one such g is shown in Table 4.1. A plot of ρ is shown in Figure 4-1a. Another valid choice of g is $g(|\beta|) = \max\{2\gamma|\beta| - \gamma^2\beta^2, \gamma|\beta|\}$. In this case, the trimmed counterpart is

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2 + \mu\gamma \sum_{i>\ell} \max\left\{2|\beta_{(i)}| - \gamma\beta_{(i)}^2, |\beta_{(i)}|\right\}.$$

Note that this problem is amenable to the same class of techniques as applied to the trimmed Lasso problem in the form (4.40) because of the increasing nature of g, although the subproblems with respect to $\boldsymbol{\beta}$ are no longer convex (although it is a usual MCP estimation problem which is well-suited to convex optimization approaches; see [110]). Also observe that we can separate the penalty function into a trimmed Lasso component and another component:

$$\sum_{i>\ell} |\beta_{(i)}| \quad \text{and} \quad \sum_{i>\ell} \left(|\beta_{(i)}| - \gamma\beta_{(i)}^2\right)_+.$$

Observe that the second component is uniformly bounded above by $(p-\ell)/(4\gamma)$, and so as $\gamma \to \infty$, the trimmed Lasso penalty dominates.



(a) Clipped Lasso and MCP (b) Log and L_q

Figure 4-1: Plots of $\rho(|\beta|;\mu,\gamma)$ for some of the penalty functions in Table 4.1.

4.4.4 The generality of trimmed estimation

We now turn our focus to more closely studying the relationship between the separable-penalty estimation problem (4.35) and its trimmed estimation counterpart. The central problems of interest are the clipped Lasso and its trimmed counterpart, viz., the trimmed Lasso:[9]

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y}-\mathbf{X}\boldsymbol{\beta}\|_2^2 + \mu \sum_i \min\{\gamma|\beta_i|, 1\} \tag{$\mathrm{CL}_{\mu,\gamma}$}$$

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y}-\mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda T_\ell(\boldsymbol{\beta}). \tag{$\mathrm{TL}_{\lambda,\ell}$}$$

As per Theorem 16, if $\boldsymbol{\beta}^*$ is an optimal solution to ($\mathrm{CL}_{\mu,\gamma}$), then $\boldsymbol{\beta}^*$ is an optimal solution to ($\mathrm{TL}_{\lambda,\ell}$), where $\lambda = \mu\gamma$ and $\ell = |\{i : |\beta_i^*| \geq 1/\gamma\}|$. We now consider the converse: given some $\lambda > 0$ and $\ell \in \{0,1,\ldots,p\}$ and a solution $\boldsymbol{\beta}^*$ to ($\mathrm{TL}_{\lambda,\ell}$), when does there exist some $\mu,\gamma > 0$ so that $\boldsymbol{\beta}^*$ is an optimal solution to ($\mathrm{CL}_{\mu,\gamma}$)? As the following theorem suggests, the existence of such a γ is closely connected to

[9] One may be concerned about the well-definedness of such problems (e.g. as guaranteed vis-à-vis coercivity of the objective, *cf.* [130]). In all the results of Section 4.4.4, it is possible to add a regularizer $\eta\|\boldsymbol{\beta}\|_1$ for some fixed $\eta > 0$ to both ($\mathrm{CL}_{\mu,\gamma}$) and ($\mathrm{TL}_{\lambda,\ell}$) and the results remain valid, *mutatis mutandis*. The addition of this regularizer implies coercivity of the objective functions and, consequently, that the minimum is indeed well-defined. For completeness, we note a technical reason for a choice of $\eta\|\boldsymbol{\beta}\|_1$ is its positive homogeneity; thus, the proof technique of Lemma 3 easily adapts to this modification.

an underlying discrete form of "convexity" of the sequence of problems ($\mathrm{TL}_{\lambda,k}$) for $k \in \{0, 1, \ldots, p\}$. We will focus on the case when $\lambda = \mu\gamma$, as this is the natural correspondence of parameters in light of Theorem 16.

Theorem 17. *If $\lambda > 0$, $\ell \in \{0, \ldots, p\}$, and $\boldsymbol{\beta}^*$ is an optimal solution to ($\mathrm{TL}_{\lambda,\ell}$), then there exist $\mu, \gamma > 0$ with $\mu\gamma = \lambda$ and so that $\boldsymbol{\beta}^*$ is an optimal solution to ($\mathrm{CL}_{\mu,\gamma}$) if and only if*

$$Z(\mathrm{TL}_{\lambda,\ell_e}) < \frac{j - \ell_e}{j - i} Z(\mathrm{TL}_{\lambda,i}) + \frac{\ell_e - i}{j - i} Z(\mathrm{TL}_{\lambda,j}) \tag{4.41}$$

for all $0 \leq i < \ell_e < j \leq p$, where $Z(\mathrm{P})$ denotes the optimal objective value to optimization problem (P) and $\ell_e = \min\{\ell, \|\boldsymbol{\beta}^*\|_0\}$.

Let us note why we refer to the condition in (4.41) as a discrete analogue of convexity of the sequence $\{z_k := Z(\mathrm{TL}_{\lambda,k}),\ k = 0, \ldots, p\}$. In particular, observe that this sequence satisfies the condition of Theorem 17 if and only if the function defined as the linear interpolation between the points $(0, z_0)$, $(1, z_1)$, ..., and (p, z_p) is strictly convex about the point (ℓ, z_ℓ).[10]

Before proceeding with the proof of the theorem, we state and prove a technical lemma about the structure of ($\mathrm{TL}_{\lambda,\ell}$).

Lemma 3. *Fix $\lambda > 0$ and suppose that $\boldsymbol{\beta}^*$ is optimal to ($\mathrm{TL}_{\lambda,\ell}$).*

(a) The optimal objective value of ($\mathrm{TL}_{\lambda,\ell}$) is $Z(\mathrm{TL}_{\lambda,\ell}) = (\|\mathbf{y}\|_2^2 - \|\mathbf{X}\boldsymbol{\beta}^*\|_2^2)/2$.

(b) If $\boldsymbol{\beta}^*$ is also optimal to ($\mathrm{TL}_{\lambda,\ell'}$), where $\ell < \ell'$, then $\|\boldsymbol{\beta}^*\|_0 \leq \ell$ and $\boldsymbol{\beta}^*$ is optimal to ($\mathrm{TL}_{\lambda,j}$) for all integral j with $\ell < j < \ell'$.

(c) If $\kappa := \|\boldsymbol{\beta}^*\|_0 < \ell$, then $\boldsymbol{\beta}^*$ is also optimal to ($\mathrm{TL}_{\lambda,\kappa}$), ($\mathrm{TL}_{\lambda,\kappa+1}$), ..., and ($\mathrm{TL}_{\lambda,\ell-1}$). Further, $\boldsymbol{\beta}^*$ is not *optimal to ($\mathrm{TL}_{\lambda,0}$), ($\mathrm{TL}_{\lambda,1}$), ..., nor ($\mathrm{TL}_{\lambda,\kappa-1}$).*

Proof. Suppose $\boldsymbol{\beta}^*$ is optimal to ($\mathrm{TL}_{\lambda,\ell}$). Define

$$a(\epsilon) := \|\mathbf{y} - \epsilon\mathbf{X}\boldsymbol{\beta}^*\|_2^2/2 + \epsilon\lambda T_\ell\left(\boldsymbol{\beta}^*\right).$$

[10] To be precise, we mean that the real-valued function that is a linear interpolation of the points has a subdifferential at the point (ℓ, z_ℓ) which is an interval of strictly positive width.

By the optimality of $\boldsymbol{\beta}^*$, $a(\epsilon) \geq a(1)$ for all $\epsilon \geq 0$. As a is a polynomial with degree at most two, one must have that $a'(1) = 0$. This implies that

$$a'(1) = -\langle \mathbf{y}, \mathbf{X}\boldsymbol{\beta}^* \rangle + \|\mathbf{X}\boldsymbol{\beta}^*\|_2^2 + \lambda T_\ell\left(\boldsymbol{\beta}^*\right) = 0.$$

Adding $(\|\mathbf{y}\|_2^2 - \|\mathbf{X}\boldsymbol{\beta}^*\|_2^2)/2$ to both sides, the desired result of part (a) follows.

Now suppose that $\boldsymbol{\beta}^*$ is also optimal to $(\mathrm{TL}_{\lambda,\ell'})$, where $\ell' > \ell$. By part (a), one must necessarily have that $Z(\mathrm{TL}_{\lambda,\ell}) = Z(\mathrm{TL}_{\lambda,\ell'}) = (\|\mathbf{y}\|_2^2 - \|\mathbf{X}\boldsymbol{\beta}^*\|_2^2)/2$. Inspecting $Z(\mathrm{TL}_{\lambda,\ell}) - Z(\mathrm{TL}_{\lambda,\ell'})$, we see that

$$0 = Z(\mathrm{TL}_{\lambda,\ell}) - Z(\mathrm{TL}_{\lambda,\ell'}) = \lambda \sum_{i=\ell+1}^{\ell'} |\beta^*_{(i)}|.$$

Hence, $|\beta^*_{(\ell+1)}| = 0$ and therefore $\|\boldsymbol{\beta}^*\|_0 \leq \ell$.

Finally, for any integral j with $\ell \leq j \leq \ell'$, one always has that $Z(\mathrm{TL}_{\lambda,\ell}) \geq Z(\mathrm{TL}_{\lambda,j}) \geq Z(\mathrm{TL}_{\lambda,\ell'})$. As per the preceding argument, $Z(\mathrm{TL}_{\lambda,\ell}) = Z(\mathrm{TL}_{\lambda,\ell})$ and so $Z(\mathrm{TL}_{\lambda,\ell}) = Z(\mathrm{TL}_{\lambda,j})$, and therefore $\boldsymbol{\beta}^*$ must also be optimal to $(\mathrm{TL}_{\lambda,j})$ by applying part (a). This completes part (b).

Part (c) follows from a straightforward inspection of objective functions and using the fact that $Z(\mathrm{TL}_{\lambda,j}) \geq Z(\mathrm{TL}_{\lambda,\ell})$ whenever $j \leq \ell$. □

Using this lemma, we can now proceed with the proof of the theorem.

Proof of Theorem 17. Let $z_k = Z(\mathrm{TL}_{\lambda,k})$ for $k \in \{0, 1, \ldots, p\}$. Suppose that $\mu, \gamma > 0$ is so that $\lambda = \mu\gamma$ and $\boldsymbol{\beta}^*$ is an optimal solution to $(\mathrm{CL}_{\mu,\gamma})$. Let $\ell_e = \min\{\ell, \|\boldsymbol{\beta}^*\|_0\}$. Per equation (4.38), $\boldsymbol{\beta}^*$ must be optimal to

$$\min_{\boldsymbol{\beta}} \min_{k \in \{0,\ldots,p\}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \mu k + \mu\gamma T_k\left(\boldsymbol{\beta}\right). \tag{4.42}$$

Observe that this implies that if k is such that k is a minimizer of $\min_k \mu k + \mu\gamma T_k\left(\boldsymbol{\beta}^*\right)$, then $\boldsymbol{\beta}^*$ must be optimal to $(\mathrm{TL}_{\lambda,k})$.

We claim that this observation, combined with Lemma 3, implies that

$$\ell_e = \arg\min_{k\in\{0,\dots,p\}} \mu k + \mu\gamma T_k\left(\boldsymbol{\beta}^*\right).$$

This can be shown as follows:

(a) Suppose $\ell \le \|\boldsymbol{\beta}^*\|_0$ and so $\ell_e = \min\{\ell, \|\boldsymbol{\beta}^*\|_0\} = \ell$. Therefore, by Lemma 3(b), $\boldsymbol{\beta}^*$ is not optimal to ($\mathrm{TL}_{\lambda,j}$) for any $j < \ell$, and thus

$$\min_{k\in\{0,\dots,p\}} \mu k + \mu\gamma T_k\left(\boldsymbol{\beta}^*\right) = \min_{k\in\{\ell,\dots,p\}} \mu k + \mu\gamma T_k\left(\boldsymbol{\beta}^*\right).$$

If $k > \ell$ is such that k is a minimizer of $\min_k \mu k + \mu\gamma T_k\left(\boldsymbol{\beta}^*\right)$, then $\boldsymbol{\beta}^*$ must be optimal to ($\mathrm{TL}_{\lambda,k}$) (using the observation), and hence by Lemma 3(b), $\|\boldsymbol{\beta}^*\|_0 \le \ell$. Combined with $\ell \le \|\boldsymbol{\beta}^*\|_0$, this implies that $\|\boldsymbol{\beta}^*\|_0 = \ell$. Yet then, $\mu\ell = \mu\ell + \mu\gamma T_\ell\left(\boldsymbol{\beta}^*\right) < \mu k + \mu\gamma T_k\left(\boldsymbol{\beta}^*\right)$, contradicting the optimality of k. Therefore, we conclude that $\ell_e = \ell$ is the *only* minimizer of $\min_k \mu k + \mu\gamma T_k\left(\boldsymbol{\beta}^*\right)$.

(b) Now instead suppose that $\ell_e = \|\boldsymbol{\beta}^*\|_0 < \ell$. Lemma 3(c) implies that any optimal solution k to $\min_k \mu k + \mu\gamma T_k\left(\boldsymbol{\beta}^*\right)$ must satisfy $k \ge \|\boldsymbol{\beta}^*\|_0$ (by the second part combined with the observation). As before, if $k > \|\boldsymbol{\beta}^*\|_0 = \ell_e$, then $\mu k > \mu\ell_e$, and so k cannot be optimal. As a result, $k = \ell_e = \|\boldsymbol{\beta}^*\|_0$ is the unique minimum.

In either case, we have that ℓ_e is the unique minimizer to $\min_k \mu k + \mu\gamma T_k\left(\boldsymbol{\beta}^*\right)$.

It then follows that $Z(\text{problem (4.42)}) = z_{\ell_e} + \mu\ell_e$. Further, by optimality of $\boldsymbol{\beta}^*$, $z_{\ell_e} + \mu\ell_e < z_i + \mu i$ for all $0 \le i \le p$ with $i \ne \ell_e$. For $0 \le i < \ell_e$, this implies $\mu < (z_i - z_{\ell_e})/(\ell_e - i)$ and for $j > \ell_e$, $\mu > (z_{\ell_e} - z_j)/(j - \ell_e)$. In other words, for $0 \le i < \ell_e < j \le p$,

$$\frac{z_{\ell_e} - z_j}{j - \ell_e} < \frac{z_i - z_{\ell_e}}{\ell_e - i}, \quad \text{i.e.,} \quad z_{\ell_e} < \frac{j - \ell_e}{j - i} z_i + \frac{\ell_e - i}{j - i} z_j.$$

This completes the forward direction. The reverse follows in the same way by taking any μ with

$$\mu \in \left(\max_{j>\ell_e} \frac{z_{\ell_e} - z_j}{j - \ell_e}, \min_{i<\ell_e} \frac{z_i - z_{\ell_e}}{\ell_e - i} \right).$$

□

We briefly remark upon one implication of the proof of Theorem 17. In particular, if $\boldsymbol{\beta}^*$ is a solution to ($\mathrm{TL}_{\lambda,\ell}$) and $\ell < \|\boldsymbol{\beta}^*\|_0$, then $\boldsymbol{\beta}^*$ is not the solution to ($\mathrm{TL}_{\lambda,k}$) for any $k \neq \ell$.

An immediate question is whether the convexity condition (4.41) of Theorem 17 always holds. While the sequence $\{Z(\mathrm{TL}_{\lambda,k}) : k = 0, 1, \ldots, p\}$ is always non-increasing, the following example shows that the convexity condition need not hold in general; as a result, there exist instances of the trimmed Lasso problem whose solutions *cannot* be found by solving a clipped Lasso problem.

Example 2. *Consider the case when $p = n = 2$ with*

$$\mathbf{y} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad \text{and} \quad \mathbf{X} = \begin{pmatrix} 1 & -1 \\ -1 & 2 \end{pmatrix}.$$

Let $\lambda = 1/2$ and $\ell = 1$, and consider $\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2/2 + |\beta_{(2)}|/2 = \min_{\beta_1,\beta_2}(1 - \beta_1 + \beta_2)^2/2 + (1 + \beta_1 - 2\beta_2)^2/2 + |\beta_{(2)}|/2$. This has unique optimal solution $\boldsymbol{\beta}^* = (3/2, 1)$ with corresponding objective value $z_1 = 3/4$. One can also compute $z_0 = Z(\mathrm{TL}_{1/2,0}) = 39/40$ and $z_2 = Z(\mathrm{TL}_{1/2,2}) = 0$. Note that $z_1 = 3/4 > (39/40)/2 + (0)/2 = z_0/2 + z_2/2$, and so there do not exist any $\mu, \gamma > 0$ with $\mu\gamma = 1/2$ so that $\boldsymbol{\beta}^*$ is an optimal solution to ($\mathrm{CL}_{\mu,\gamma}$) by Theorem 17. Further, it is possible to show that $\boldsymbol{\beta}^*$ is not an optimal solution to ($\mathrm{CL}_{\mu,\gamma}$) for any *choice of $\mu, \gamma \geq 0$.*

Proof of validity of Example 2. Set

$$\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + |\beta_{(2)}| = \min_{\beta_1,\beta_2}(1 - \beta_1 + \beta_2)^2 + (1 + \beta_1 - 2\beta_2)^2 + |\beta_{(2)}|. \tag{4.43}$$

We have omitted the factor of $1/2$ in order to avoid unnecessary complications.

Solving problem (4.43) and its related counterparts (for $\ell \in \{0, 2\}$) can rely on convex analysis because we can simply enumerate all possible scenarios. In particular,

the solution to (4.43) is $\boldsymbol{\beta}^* = (3/2, 1)$ based on an analysis of two related problems:

$$\min_{\beta_1,\beta_2}(1-\beta_1+\beta_2)^2+(1+\beta_1-2\beta_2)^2+|\beta_1|.$$
$$\min_{\beta_1,\beta_2}(1-\beta_1+\beta_2)^2+(1+\beta_1-2\beta_2)^2+|\beta_2|.$$

(We should be careful to impose the additional constraints $|\beta_1| \leq |\beta_2|$ and $|\beta_1| \geq |\beta_2|$, respectively, although a simple argument shows that these constraints are not required in this example.) Standard convex analysis using the Lasso (e.g. by directly using subdifferentials) shows that the problems have respective solutions $(1/2, 1/2)$ and $(3/2, 1)$, with the latter having the better objective value in (4.43). As such, $\boldsymbol{\beta}^*$ is indeed optimal. The solution in the cases of $\ell \in \{0, 2\}$ follows a similarly standard analysis.

It is perhaps more interesting to study the general case where $\mu, \gamma \geq 0$. In particular, we will show that $\boldsymbol{\beta}^* = (3/2, 1)$ is not an optimal solution to the clipped Lasso problem

$$\min_{\beta_1,\beta_2}(1-\beta_1+\beta_2)^2+(1+\beta_1-2\beta_2)^2+\mu\min\{\gamma|\beta_1|,1\}+\mu\min\{\gamma|\beta_2|,1\} \tag{4.44}$$

for any choices of μ and γ. While in general such a problem may be difficult to fully analyze, we can again rely on localized analysis using convex analysis. To proceed, let

$$f(\beta_1,\beta_2)=(1-\beta_1+\beta_2)^2+(1+\beta_1-2\beta_2)^2+\mu\min\{\gamma|\beta_1|,1\}+\mu\min\{\gamma|\beta_2|,1\},$$

with the parameters μ and γ implicit. We consider the following exhaustive cases:

1. $\boxed{\gamma > 1}$: In this case, f is convex and differentiable in a neighborhood of $\boldsymbol{\beta}^*$. Its gradient at $\boldsymbol{\beta}^*$ is $\nabla f(\boldsymbol{\beta}^*) = (0, -1)$, and therefore $\boldsymbol{\beta}^*$ is neither locally optimal nor globally optimal to problem (4.44).

2. $\boxed{\gamma < 2/3}$: In this case, f is again convex and differentiable in a neighborhood of $\boldsymbol{\beta}^*$. Its gradient at $\boldsymbol{\beta}^*$ is $\nabla f(\boldsymbol{\beta}^*) = (\mu\gamma, \mu\gamma - 1)$. Again, this cannot equal

$(0,0)$ and therefore $\boldsymbol{\beta}^*$ is neither locally nor globally optimal to problem (4.44).

3. $\boxed{2/3 < \gamma < 1}$: In this case, f is again convex and differentiable in a neighborhood of $\boldsymbol{\beta}^*$. Its gradient at $\boldsymbol{\beta}^*$ is $\nabla f(\boldsymbol{\beta}^*) = (0, \mu\gamma - 1)$. As a necessary condition for local optimality, we must have that $\mu\gamma = 1$, implying that $\mu > 1$. Further, if $\boldsymbol{\beta}^*$ is optimal to (4.44), then $f(\boldsymbol{\beta}^*) \leq f(0,0)$. Yet,

$$\begin{aligned} f(\boldsymbol{\beta}^*) &= 1/2 + \mu + \mu\gamma = 3/2 + \mu \\ f(0,0) &= 2, \end{aligned}$$

implying that $\mu \leq 1/2$, in contradiction of $\mu > 1$. Hence, $\boldsymbol{\beta}^*$ cannot be optimal to (4.44).

4. $\boxed{\gamma = 2/3}$: In this case, we make two comparisons, using the points $\boldsymbol{\beta}^*$, $(0,0)$, and $(3,2)$:

$$\begin{aligned} f(\boldsymbol{\beta}^*) &= 1/2 + \mu + 2\mu/3 = 1/2 + 5\mu/3 \\ f(0,0) &= 2 \\ f(3,2) &= 2\mu. \end{aligned}$$

Assuming optimality of $\boldsymbol{\beta}^*$, we have that $f(\boldsymbol{\beta}^*) \leq f(0,0)$, i.e., $\mu \leq 9/10$; similarly, $f(\boldsymbol{\beta}^*) \leq f(3,2)$, i.e., $\mu \geq 3/2$. Clearly both cannot hold, and so therefore $\boldsymbol{\beta}^*$ cannot be optimal.

5. $\boxed{\gamma = 1}$: Finally, we see that $f(\boldsymbol{\beta}^*) \leq f(3,2)$ would imply that $1/2 + 2\mu \leq 2\mu$, which is impossible; hence, $\boldsymbol{\beta}^*$ is not optimal to (4.44). (This argument can clearly also be used in the case when $\gamma > 1$, although it is instructive to see the argument given above in that case.)

In any case, we have that $\boldsymbol{\beta}^*$ cannot be a solution to the clipped Lasso problem (4.44). This completes the proof of validity of Example 2. □

An immediate corollary of this example, combined with Theorem 16, is that the

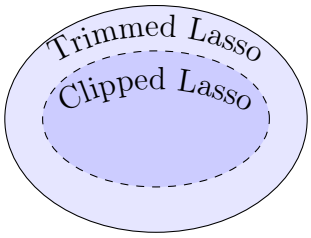


Figure 4-2: Stylized relation of clipped Lasso and trimmed Lasso models. Every clipped Lasso model can be written as a trimmed Lasso model, but the reverse does not hold in general.

class of trimmed Lasso models contains the class of clipped Lasso models as a *proper* subset, regardless of whether we restrict our attention to $\lambda = \mu\gamma$. In this sense, the trimmed Lasso models comprise a richer set of models. The relationship is depicted in stylized form in Figure 4-2.

Limit analysis

It is important to contextualize the results of this section as $\lambda \to \infty$. This corresponds to $\gamma \to \infty$ for the clipped Lasso problem, in which case ($\text{CL}_{\mu,\gamma}$) converges to the penalized form of subset selection:

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \mu\|\boldsymbol{\beta}\|_0. \tag{$\text{CL}_{\mu,\infty}$}$$

Note that penalized problems for all of the penalties listed in Table 4.1 have this as their limit as $\gamma \to \infty$. On the other hand, ($\text{TL}_{\lambda,\ell}$) converges to constrained best subset selection:

$$\min_{\|\boldsymbol{\beta}\|_0 \leq \ell} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2. \tag{$\text{TL}_{\infty,k}$}$$

Indeed, from this comparison it now becomes clear why a convexity condition of the form in Theorem 17 appears in describing when the clipped Lasso solves the trimmed Lasso problem. In particular, the conditions under ($\text{CL}_{\mu,\infty}$) solves the constrained best subset selection problem ($\text{TL}_{\infty,k}$) are precisely those in Theorem 17.

4.4.5 Unbounded penalty functions

We close this section by now considering nonconvex penalty functions which are unbounded and therefore do not take the form $\mu \min\{g(|\beta|), 1\}$. Two such examples are the L_q penalty ($0 < q < 1$) and the log family of penalties as shown in Table 4.1 and depicted in Figure 4-1. Estimation problems with these penalties can be cast in the form

$$\min_{\boldsymbol{\phi}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\phi}\|_2^2 + \mu \sum_{i=1}^{p} g(|\phi_i|; \gamma) \tag{4.45}$$

where $\mu, \gamma > 0$ are parameters, g is an unbounded and strictly increasing function, and $g(|\phi_i|; \gamma) \xrightarrow{\gamma \to \infty} I\{|\phi_i| > 0\}$. The change of variables in (4.45) is intentional and its purpose will become clear shortly.

Observe that because g is now unbounded, there exists some $\underline{\lambda} = \underline{\lambda}(\mathbf{y}, \mathbf{X}, \mu, \gamma) > 0$ so that for all $\lambda > \underline{\lambda}$ any optimal solution $(\boldsymbol{\phi}^*, \boldsymbol{\epsilon}^*)$ to the problem

$$\min_{\boldsymbol{\phi}, \boldsymbol{\epsilon}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}(\boldsymbol{\phi} + \boldsymbol{\epsilon})\|_2^2 + \lambda\|\boldsymbol{\epsilon}\|_1 + \mu \sum_{i=1}^{p} g(|\phi_i|; \gamma) \tag{4.46}$$

has $\boldsymbol{\epsilon}^* = \mathbf{0}$.[11] Therefore, (4.45) is a special case of (4.46). We claim that in the limit as $\gamma \to \infty$ (all else fixed), that (4.46) can be written exactly as a trimmed Lasso problem ($\text{TL}_{\lambda,k}$) for some choice of k and with the identification of variables $\boldsymbol{\beta} = \boldsymbol{\phi} + \boldsymbol{\epsilon}$.

We summarize this as follows:

Proposition 19. *As $\gamma \to \infty$, the penalized estimation problem (4.45) is a special case of the trimmed Lasso problem.*

Proof. This can be shown in a straightforward manner: namely, as $\gamma \to \infty$, (4.46) becomes

$$\min_{\boldsymbol{\phi}, \boldsymbol{\epsilon}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}(\boldsymbol{\phi} + \boldsymbol{\epsilon})\|_2^2 + \lambda\|\boldsymbol{\epsilon}\|_1 + \mu\|\boldsymbol{\phi}\|_0$$

[11] The proof involves a straightforward modification of an argument along the lines of that given in Theorem 13. Also note that we can choose $\underline{\lambda}$ so that it is decreasing in γ, *ceteris paribus*.

which can be in turn written as

$$\min_{\substack{\boldsymbol{\phi},\boldsymbol{\epsilon}:\\ \|\boldsymbol{\phi}\|_0 \leq k}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}(\boldsymbol{\phi}+\boldsymbol{\epsilon})\|_2^2 + \lambda\|\boldsymbol{\epsilon}\|_1$$

for some $k \in \{0, 1, \ldots, p\}$. But as per the observations of Section 4.2.3, this is exactly ($\text{TL}_{\lambda,k}$) using a change of variables $\boldsymbol{\beta} = \boldsymbol{\phi}+\boldsymbol{\epsilon}$. In the case when λ is sufficiently large, we necessarily have $\boldsymbol{\beta} = \boldsymbol{\phi}$ at optimality. □

While this result is not surprising (given that as $\gamma \to \infty$ the problem is (4.45) is precisely penalized best subset selection), it is useful for illustrating the connection between (4.45) and the trimmed Lasso problem even when the trimmed Lasso parameter λ is not necessarily large: in particular, ($\text{TL}_{\lambda,k}$) can be viewed as estimating $\boldsymbol{\beta}$ as the sum of two components—a sparse component $\boldsymbol{\phi}$ and small-norm ("noise") component $\boldsymbol{\epsilon}$. Indeed, in this setup, λ precisely controls the desirable level of allowed "noise" in $\boldsymbol{\beta}$. From this intuitive perspective, it becomes clearer why the trimmed Lasso type approach represents a continuous connection between best subset selection (λ large) and ordinary least squares (λ small).

We close this section by making the following observation regarding problem (4.46). In particular, observe that regardless of λ, we can rewrite this as

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \sum_{i=1}^{p} \widetilde{\rho}(|\beta_i|)$$

where $\widetilde{\rho}(|\beta_i|)$ is the new penalty function defined as

$$\widetilde{\rho}(|\beta_i|) = \min_{\phi+\epsilon=\beta_i} \lambda|\epsilon| + \mu g(|\phi|;\gamma).$$

For the unbounded and concave penalty functions shown in Table 4.1, this new penalty function is quasi-concave and can be rewritten easily in closed form. For example, for the L_q penalty $\rho(|\beta_i|) = \mu|\beta_i|^{1/\gamma}$ (where $\gamma > 1$), the new penalty function is

$$\widetilde{\rho}(|\beta_i|) = \min\{\mu|\beta_i|^{1/\gamma}, \lambda|\beta_i|\}.$$

4.5 Algorithmic Approaches

We now turn our attention to algorithms for estimation with the trimmed Lasso penalty. Our principle focus throughout will be the same problem considered in Theorem 13, namely

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda T_k(\boldsymbol{\beta}) + \eta\|\boldsymbol{\beta}\|_1 \tag{4.47}$$

We present three possible approaches to finding potential solutions to (4.47): a first-order-based alternating minimization scheme that has accompanying local optimality guarantees and was first studied in [72, 159]; an augmented Lagrangian approach that appears to perform noticeably better, despite lacking optimality guarantees; and a convex envelope approach. We contrast these methods with approaches for certifying global optimality of solutions to (4.47) (described in [154]) and include an illustrative computational example. Implementations of the various algorithms presented can be found in Appendix D.

4.5.1 Upper bounds via convex methods

We start by focusing on the application of convex optimization methods to finding to finding potential solutions to (4.47). Technical details are contained in Appendix C.2.

Alternating minimization scheme

We begin with a first-order-based approach for obtaining a locally optimal solution of (4.47) as described in [72, 159]. The key tool in this approach is the theory of difference of convex optimization ("DCO") [3, 145, 4]. Set the following notation:

$$\begin{aligned} f(\boldsymbol{\beta}) &= \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2/2 + \lambda T_k(\boldsymbol{\beta}) + \eta\|\boldsymbol{\beta}\|_1, \\ f_1(\boldsymbol{\beta}) &= \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2/2 + (\eta+\lambda)\|\boldsymbol{\beta}\|_1, \\ f_2(\boldsymbol{\beta}) &= \lambda \textstyle\sum_{i=1}^k |\beta_{(i)}|. \end{aligned}$$

Let us make a few simple observations:

(a) Problem (4.47) can be written as $\min_{\boldsymbol{\beta}} f(\boldsymbol{\beta})$.

(b) For all $\boldsymbol{\beta}$, $f(\boldsymbol{\beta}) = f_1(\boldsymbol{\beta}) - f_2(\boldsymbol{\beta})$.

(c) The functions f_1 and f_2 are convex.

While simple, these observations enable one to apply the theory of DCO, which focuses precisely on problems of the form

$$\min_{\boldsymbol{\beta}} f_1(\boldsymbol{\beta}) - f_2(\boldsymbol{\beta}),$$

where f_1 and f_2 are convex. In particular, the optimality conditions for such a problem have been studied extensively [4]. Let us note that while it may appear that the representation of the objective f as $f_1 - f_2$ might otherwise seem like an artificial algebraic manipulation, the min-min representation in Theorem 14 shows how such a difference-of-convex representation can arise naturally.

We now discuss an associated alternating minimization scheme (or equivalently, a sequential linearization scheme), shown in Algorithm 3, for finding local optima of (4.47). The convergence properties of Algorithm 3 can be summarized as follows:[12]

Theorem 18 (*cf.* [72], Convergence of Algorithm 3)**.** *(a) The sequence* $\{f(\boldsymbol{\beta}^\ell) : \ell = 0, 1, \ldots\}$*, where* $\boldsymbol{\beta}^\ell$ *are as found in Algorithm 3, is non-increasing.*

(b) The set $\{\boldsymbol{\gamma}^\ell : \ell = 0, 1, \ldots\}$ *is finite and eventually periodic.*

(c) Algorithm 3 converges in a finite number of iterations to local minimum of (4.47)*.*

(d) The rate of convergence of $f(\boldsymbol{\beta}^\ell)$ *is linear.*

Observation 2. *Let us return to a remark that preceded Algorithm 3. In particular, we noted that Algorithm 3 can also be viewed as a sequential linearization approach*

[12] To be entirely correct, this result holds for Algorithm 3 with a minor technical modification—see details in Appendix C.2.

Algorithm 3 An alternating scheme for computing a local optimum to (4.47)

1. Initialize with any $\boldsymbol{\beta}^0 \in \mathbb{R}^p$ ($\ell = 0$); for $\ell \geq 0$, repeat Steps 2-3 until $f(\boldsymbol{\beta}^\ell) = f(\boldsymbol{\beta}^{\ell+1})$.

2. Compute $\boldsymbol{\gamma}^\ell$ as

$$\boldsymbol{\gamma}^\ell \in \begin{array}{ll} \underset{\boldsymbol{\gamma}}{\operatorname{argmax}} & \langle \boldsymbol{\gamma}, \boldsymbol{\beta}^\ell \rangle \\ \text{s.t.} & \sum_i |\gamma_i| \leq \lambda k \\ & |\gamma_i| \leq \lambda \ \forall i. \end{array} \tag{4.48}$$

3. Compute $\boldsymbol{\beta}^{\ell+1}$ as

$$\boldsymbol{\beta}^{\ell+1} \in \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \ \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + (\eta + \lambda)\|\boldsymbol{\beta}\|_1 - \langle \boldsymbol{\beta}, \boldsymbol{\gamma}^\ell \rangle. \tag{4.49}$$

to solving (4.47). Namely, this corresponds to sequentially performing a linearization of f_2 (and leaving f_1 as is), and then solving the new convex linearized problem.

Further, let us note why we refer to Algorithm 3 as an alternating minimization scheme. In particular, in light of the reformulation (4.11) of (4.47), we can rewrite (4.47) exactly as

$$(4.47) = \begin{array}{ll} \underset{\boldsymbol{\beta},\boldsymbol{\gamma}}{\min} & f_1(\boldsymbol{\beta}) - \langle \boldsymbol{\gamma}, \boldsymbol{\beta} \rangle \\ \text{s.t.} & \sum_i |\gamma_i| \leq \lambda k \\ & |\gamma_i| \leq \lambda \ \forall i. \end{array}$$

In this sense, if one takes care in performing alternating minimization in $\boldsymbol{\beta}$ (with $\boldsymbol{\gamma}$ fixed) and in $\boldsymbol{\gamma}$ (with $\boldsymbol{\beta}$ fixed) (as in Algorithm 3), then a locally optimal solution is guaranteed.

We now turn to how to actually apply Algorithm 3. Observe that the algorithm is quite simple; in particular, it only requires solving two types of well-structured convex optimization problems. The first such problem, for a fixed $\boldsymbol{\beta}$, is shown in (4.48). This can be solved in closed form by simply sorting the entries of $|\boldsymbol{\beta}|$, i.e., by finding $|\beta_{(1)}|, \ldots, |\beta_{(p)}|$. The second subproblem, shown in (4.49) for a fixed $\boldsymbol{\gamma}$, is precisely the usual Lasso problem and is amenable to any of the possible algorithms for the Lasso.

Augmented Lagrangian approach

We briefly mention another technique for finding potential solutions to (4.47) using an Alternating Directions Method of Multipliers (ADMM) [37] approach. To our knowledge, the application of ADMM to the trimmed Lasso problem is novel, although it appears closely related to [153]. We begin by observing that (4.47) can be written exactly as

$$\begin{aligned} \min_{\boldsymbol{\beta},\boldsymbol{\gamma}} \quad & \tfrac{1}{2}\left\|\mathbf{y}-\mathbf{X}\boldsymbol{\beta}\right\|_2^2+\eta\left\|\boldsymbol{\beta}\right\|_1+\lambda T_k\left(\boldsymbol{\gamma}\right) \\ \text{s. t.} \quad & \boldsymbol{\beta}=\boldsymbol{\gamma}, \end{aligned}$$

which makes use of the canonical variable splitting. Introducing dual variable $\mathbf{q}\in\mathbb{R}^p$ and parameter $\sigma>0$, this becomes in augmented Lagrangian form

$$\begin{aligned} \min_{\boldsymbol{\beta},\boldsymbol{\gamma}}\max_{\mathbf{q}} \frac{1}{2}\left\|\mathbf{y}-\mathbf{X}\boldsymbol{\beta}\right\|_2^2+\eta\left\|\boldsymbol{\beta}\right\|_1+\lambda T_k\left(\boldsymbol{\gamma}\right)+ \\ \left\langle\mathbf{q},\boldsymbol{\beta}-\boldsymbol{\gamma}\right\rangle+\frac{\sigma}{2}\left\|\boldsymbol{\beta}-\boldsymbol{\gamma}\right\|_2^2. \end{aligned} \tag{4.50}$$

The utility of such a reformulation is that it is directly amenable to ADMM, as detailed in Algorithm 4. While the problem is nonconvex and therefore the ADMM is not guaranteed to converge, numerical experiments suggest that this approach has superior performance to the DCO-inspired method considered in Algorithm 3.

We close by commenting on the subproblems that must be solved in Algorithm 4. Step 2 can be carried out using "hot" starts. Step 3 is the solution of the trimmed Lasso in the orthogonal design case and can be solved by performed by sorting p numbers; see Appendix C.2.3.

Convexification approach

We briefly consider the convex relaxation of the problem (4.47). We begin by computing the convex envelope [130, 38] of T_k on $[-1,1]^p$ (here the choice of $[-1,1]^p$ is standard, such as in the convexification of L_0 over this set which leads to L_1). The proof follows standard techniques (e.g. computing the biconjugate [130]) and is omitted.

Algorithm 4 ADMM algorithm for (4.50)

1. Initialize with any $\boldsymbol{\beta}^0, \boldsymbol{\gamma}^0, \mathbf{q}^0 \in \mathbb{R}^p$ and $\sigma > 0$. Repeat, for $\ell \geq 0$, Steps 2, 3, and 4 until a desired numerical convergence tolerance is satisfied.

2. Set

$$\boldsymbol{\beta}^{\ell+1} \in \operatorname*{argmin}_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \eta\|\boldsymbol{\beta}\|_1 + \langle \mathbf{q}^\ell, \boldsymbol{\beta}\rangle + \frac{\sigma}{2}\|\boldsymbol{\beta} - \boldsymbol{\gamma}^\ell\|_2^2.$$

3. Set

$$\boldsymbol{\gamma}^{\ell+1} \in \operatorname*{argmin}_{\boldsymbol{\gamma}} \lambda T_k(\boldsymbol{\gamma}) + \frac{\sigma}{2}\|\boldsymbol{\beta}^{\ell+1} - \boldsymbol{\gamma}\|_2^2 - \langle \mathbf{q}^\ell, \boldsymbol{\gamma}\rangle.$$

4. Set $\mathbf{q}^{\ell+1} = \mathbf{q}^\ell + \sigma\left(\boldsymbol{\beta}^{\ell+1} - \boldsymbol{\gamma}^{\ell+1}\right)$.

Lemma 4. *The convex envelope of T_k on $[-1,1]^p$ is the function $\overline{T}_k$ defined as*

$$\overline{T}_k(\boldsymbol{\beta}) = \left(\|\boldsymbol{\beta}\|_1 - k\right)_+.$$

In words, the convex envelope of T_k is a "soft thresholded" version of the Lasso penalty (thresholded at level k). This can be thought of as an alternative way of interpreting the name "trimmed Lasso."

As a result of Lemma 4, it follows that the convex analogue of (4.47), as taken over $[-1,1]^p$, is precisely

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \eta\|\boldsymbol{\beta}\|_1 + \lambda\left(\|\boldsymbol{\beta}\|_1 - k\right)_+. \tag{4.51}$$

Problem (4.51) is amenable to a variety of convex optimization techniques such as subgradient descent [38].

4.5.2 Certificates of optimality for (4.47)

We close our discussion of the algorithmic implications of the trimmed Lasso by discussing techniques for finding certifiably optimal solutions to (4.47). All approaches

presented in the preceding section find potential candidates for solutions to (4.47), but none is necessarily globally optimal. Let us return to a representation of (4.47) that makes use Lemma 1:

$$\begin{aligned} \min_{\boldsymbol{\beta},\mathbf{z}} \quad & \tfrac{1}{2}\|\mathbf{y}-\mathbf{X}\boldsymbol{\beta}\|_2^2 + \eta\|\boldsymbol{\beta}\|_1 + \lambda\langle \mathbf{z}, |\boldsymbol{\beta}|\rangle \\ \text{s. t.} \quad & \sum_i z_i = p-k \\ & \mathbf{z} \in \{0,1\}^p. \end{aligned}$$

As noted in [72], this representation of (4.47) is amenable to mixed integer optimization ("MIO") methods [35] for finding globally optimal solutions to (4.47), in the same spirit as other MIO-based approaches to statistical problems [31, 30].

One approach, as described in [154], uses the notion of "big M." In particular, for $M > 0$ sufficiently large, problem (4.47) can be written exactly as the following linear MIO problem:

$$\begin{aligned} \min_{\boldsymbol{\beta},\mathbf{z},\mathbf{a}} \quad & \frac{1}{2}\|\mathbf{y}-\mathbf{X}\boldsymbol{\beta}\|_2^2 + \eta\|\boldsymbol{\beta}\|_1 + \lambda\sum_i a_i \\ \text{s. t.} \quad & \sum_i z_i = p-k \\ & \mathbf{z} \in \{0,1\}^p \\ & \mathbf{a} \geq \boldsymbol{\beta} + M\mathbf{z} - M\mathbf{1} \\ & \mathbf{a} \geq -\boldsymbol{\beta} + M\mathbf{z} - M\mathbf{1} \\ & \mathbf{a} \geq \mathbf{0}. \end{aligned} \tag{4.52}$$

This representation as a linear MIO problem enables the direct application of numerous existing MIO algorithms (such as [73]).[13] Also, let us note that the linear relaxation of (4.52), i.e., problem (4.52) with the constraint $\mathbf{z} \in \{0,1\}^p$ replaced with

[13]There are certainly other possible representations of (4.11), such as using special ordered set (SOS) constraints, see e.g. [30]. Without more sophisticated tuning of M as in [30], the SOS formulations appear to be vastly superior in terms of time required to prove optimality. The precise formulation essentially takes the form of problem (4.14). An SOS-based implementation is provided in the supplementary code as the default method of certifying optimality.

$\mathbf{z} \in [0,1]^p$, is the problem

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \eta\|\boldsymbol{\beta}\|_1 + \lambda\left(\|\boldsymbol{\beta}\|_1 - Mk\right)_+,$$

where we see the convex envelope penalty appear directly. As such, when M is large, the linear relaxation of (4.52) is the ordinary Lasso problem $\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{y}-\mathbf{X}\boldsymbol{\beta}\|_2^2+\eta\|\boldsymbol{\beta}\|_1$.

4.5.3 Computational example

Because a rigorous computational comparison is not the primary focus of this chapter, we provide a limited demonstration that describes the behavior of solutions to (4.47) as computed via the different approaches. Precise computational details are contained in Appendix C.2.4. We will focus on two different aspects: sparsity and approximation quality.

Sparsity properties: As the motivation for the trimmed Lasso is ostensibly sparse modeling, its sparsity properties are of particular interest. We consider a problem instance with $p = 20$, $n = 100$, $k = 2$, and signal-to-noise ratio 10 (the sparsity of the ground truth model $\boldsymbol{\beta}_{\text{true}}$ is 10). The relevant coefficient profiles as a function of λ are shown in Figure 4-3. In this example none of the convex approaches finds the optimal two variable solution computed using mixed integer optimization. Further, as one would expect *a priori*, the optimal coefficient profiles (as well as the ADMM profiles) are not continuous in λ. Finally, note that by design of the algorithms, the alternating minimization and ADMM approaches yield solutions with sparsity at most k for λ sufficiently large.

Optimality gap: Another critical question is the degree of suboptimality of solutions found via the convex approaches. We average optimality gaps across 100 problem instances with $p = 20$, $n = 100$, and $k = 2$; the relevant results are shown in Figure 4-4. The results are entirely as one might expect. When λ is small and the problem is convex or nearly convex, the heuristics perform well. However, this

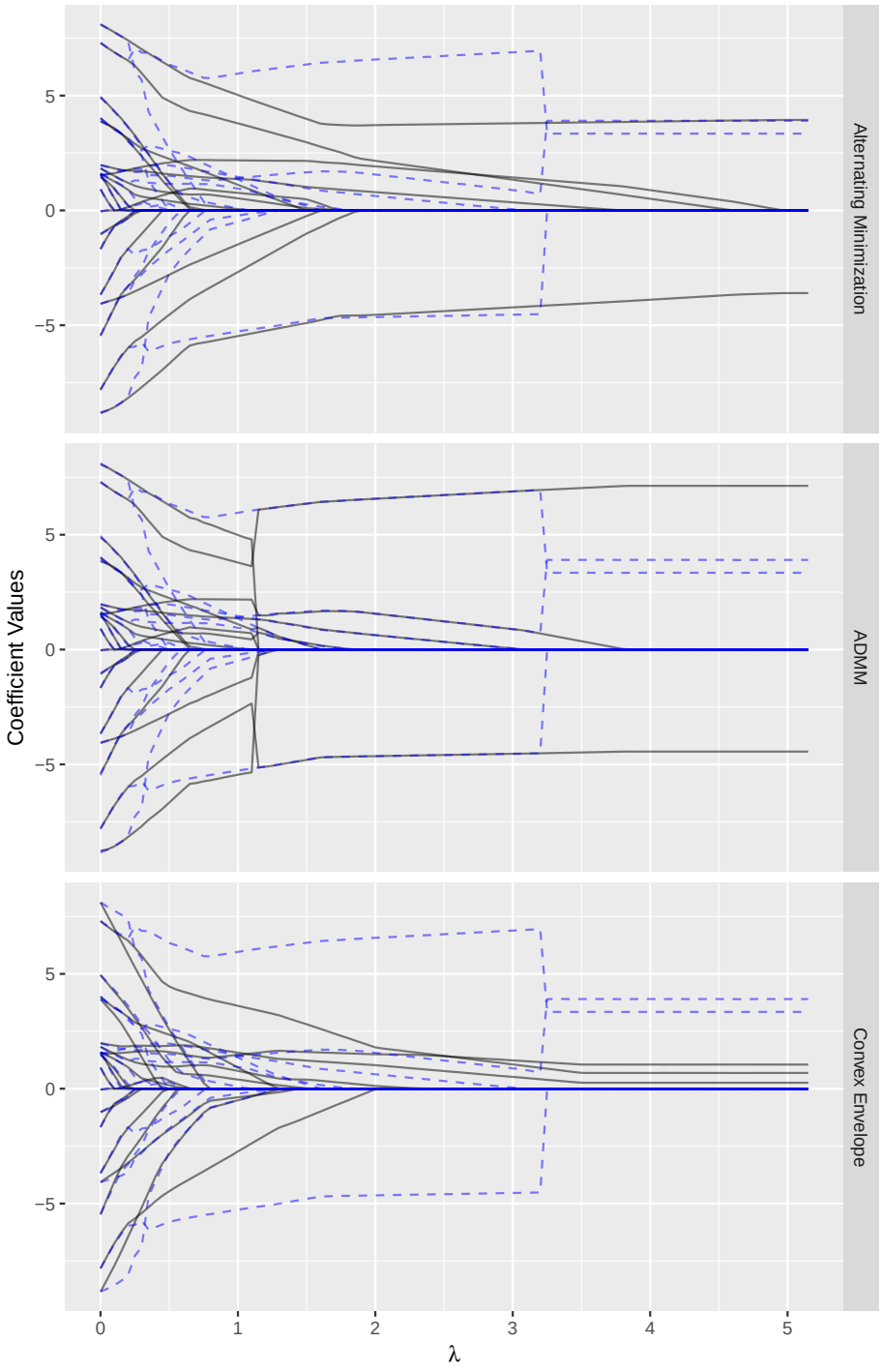


Figure 4-3: Trimmed Lasso regularization paths for heuristic algorithms as compared with optimal. Heuristic shown in solid black; optimal shown in dashed blue.

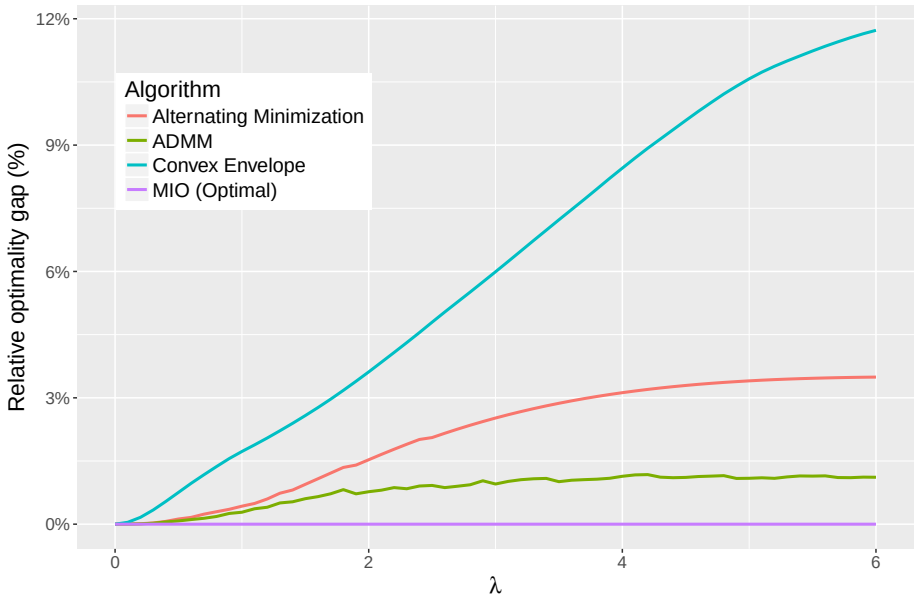


Figure 4-4: Relative optimality gaps for heuristic algorithms for trimmed Lasso. Color denotes algorithm type (purple is the optimal baseline).

breaks down as λ increases and the sparsity-inducing nature of the trimmed Lasso penalty comes into play. Further, we see that the convex envelope approach tends to perform the worst, with the ADMM performing the best of the three heuristics. This is perhaps not surprising, as any solution found via the ADMM can be guaranteed to be locally optimal by subsequently applying the alternating minimization scheme of Algorithm 3 to any solution found via Algorithm 4.

Computational burden: Loosely speaking, the heuristic approaches all carry a similar cost per iteration, namely, solving a Lasso-like problem. In contrast, the MIO approach can take significantly more computational resources. However, by design, the MIO approach maintains a suboptimality gap throughout computation and can therefore be terminated, before optimality is certified, with a certificate of suboptimality. We do not consider any empirical analysis of runtime here.

Other considerations: There are other additional computational considerations that are potentially of interest as well, but they are primarily beyond the scope of the present work. For example, instead of considering optimality purely in terms of

objective values in (4.47), there are other critical notions from a statistical perspective (e.g. ability to recover true sparse models and performance on out-of-sample data) that would also be necessary to consider across the multiple approaches.

4.6 Conclusions

In this chapter, we have studied the trimmed Lasso, a nonconvex adaptation of Lasso that acts as an exact penalty method for best subset selection. Unlike some other approaches to exact penalization which use coordinate-wise separable functions, the trimmed Lasso offers direct control of the desired sparsity k. Further, we emphasized the interpretation of the trimmed Lasso from the perspective of robustness. In doing so, we provided contrasts with the SLOPE penalty as well as comparisons with estimators from the robust statistics and total least squares literature.

We have also taken care to contextualize the trimmed Lasso within the literature on nonconvex penalized estimation approaches to sparse modeling, showing that penalties like the trimmed Lasso can be viewed as a generalization of such approaches in the case when the penalty function is bounded. In doing so, we also highlighted how precisely the problems were related, with a complete characterization given in the case of the clipped Lasso.

Finally, we have shown how modern developments in optimization can be brought to bear for the trimmed Lasso to create convex optimization optimization algorithms that can take advantage of the significant developments in algorithms for Lasso-like problems in recent years.

Our work here raises many interesting questions about further properties of the trimmed Lasso and the application of similar ideas in other settings. We see two particularly noteworthy directions of focus: algorithms and statistical properties. For the former, we anticipate that an approach like trimmed Lasso, which leads to relatively straightforward algorithms that use close analogues from convex optimization, is simple to interpret and to implement. At the same time, the heuristic approaches to the trimmed Lasso presented herein carry no more of a computational burden than

solving convex, Lasso-like problems. On the latter front, we anticipate that a deeper analysis of the statistical properties of estimators attained using the trimmed Lasso would help to illuminate it in its own right while also further connecting it to existing approaches in the statistical estimation literature.

Appendix A

Supplement for Chapter 2

A.1 Proofs

This appendix contains all omitted proofs for results presented in Chapter 2.

Proof of Proposition 1: (a) We start by observing that for any two real symmetric matrices $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{p\times p}$ and the matrix q-norm, a result (due to Mirsky and sometimes known as the q-Wielandt-Hoffman inequality [144, p. 205]) states that

$$\|\mathbf{A}-\mathbf{B}\|_q \geq \|\boldsymbol{\lambda}(\mathbf{A})-\boldsymbol{\lambda}(\mathbf{B})\|_q, \tag{A.1}$$

where $\boldsymbol{\lambda}(\mathbf{A})$ and $\boldsymbol{\lambda}(\mathbf{B})$ denote the vector of eigenvalues of $\mathbf{A}$ and $\mathbf{B}$, respectively, arranged in decreasing order, i.e., $\lambda_1(\mathbf{A}) \geq \lambda_2(\mathbf{A}) \geq \ldots \geq \lambda_p(\mathbf{A})$ and $\lambda_1(\mathbf{B}) \geq \lambda_2(\mathbf{B}) \geq \ldots \geq \lambda_p(\mathbf{B})$. Using this result for Problem (2.4), it is easy to see that for fixed $\boldsymbol{\Phi}$ one has

$$\{\boldsymbol{\Theta} : \boldsymbol{\Theta} \succcurlyeq \mathbf{0},\ \mathrm{rank}(\boldsymbol{\Theta}) \leq r\} = \{\boldsymbol{\Theta} : \boldsymbol{\lambda}(\boldsymbol{\Theta}) \geq \mathbf{0},\ \|\boldsymbol{\lambda}(\boldsymbol{\Theta})\|_0 \leq r\}, \tag{A.2}$$

where $\|\boldsymbol{\lambda}(\boldsymbol{\Theta})\|_0$ counts the number of nonzero elements of $\boldsymbol{\lambda}(\boldsymbol{\Theta})$. If we partially minimize the objective function in Problem (2.4), with respect to $\boldsymbol{\Theta}$ (with $\boldsymbol{\Phi}$

fixed), and use (A.1) along with (A.2), we have the following inequality:

$$\begin{aligned} & \inf_{\boldsymbol{\Theta}} \quad \|(\boldsymbol{\Sigma}-\boldsymbol{\Phi})-\boldsymbol{\Theta}\|_q^q \qquad \text{s.t.} \quad \boldsymbol{\Theta} \succcurlyeq \mathbf{0}, \operatorname{rank}(\boldsymbol{\Theta}) \leq r \\ \geq & \inf_{\boldsymbol{\lambda}(\boldsymbol{\Theta})} \quad \|\boldsymbol{\lambda}(\boldsymbol{\Sigma}-\boldsymbol{\Phi})-\boldsymbol{\lambda}(\boldsymbol{\Theta})\|_q^q \quad \text{s.t.} \quad \boldsymbol{\lambda}(\boldsymbol{\Theta}) \geq \mathbf{0}, \|\boldsymbol{\lambda}(\boldsymbol{\Theta})\|_0 \leq r. \end{aligned} \tag{A.3}$$

Since $\boldsymbol{\Sigma}-\boldsymbol{\Phi} \succcurlyeq \mathbf{0}$, it follows that the minimum objective value of the right hand side of (A.3) is given by $\sum_{i=r+1}^{p} \lambda_i^q(\boldsymbol{\Sigma}-\boldsymbol{\Phi})$ and is achieved for $\lambda_i(\boldsymbol{\Theta}) = \lambda_i(\boldsymbol{\Sigma}-\boldsymbol{\Phi})$ for $i = 1, \ldots, r$. This leads to the inequality

$$\inf_{\substack{\boldsymbol{\Theta}:\boldsymbol{\Theta}\succcurlyeq\mathbf{0}, \\ \operatorname{rank}(\boldsymbol{\Theta})\leq r}} \|(\boldsymbol{\Sigma}-\boldsymbol{\Phi})-\boldsymbol{\Theta}\|_q^q \;\geq\; \sum_{i=r+1}^{p} \lambda_i^q(\boldsymbol{\Sigma}-\boldsymbol{\Phi}). \tag{A.4}$$

Furthermore, if $\mathbf{U} \in \mathbb{R}^{p\times p}$ denotes the matrix of p eigenvectors of $\boldsymbol{\Sigma}-\boldsymbol{\Phi}$, then the following choice of

$$\boldsymbol{\Theta}^* := \mathbf{U}\ \operatorname{diag}\big(\lambda_1(\boldsymbol{\Sigma}-\boldsymbol{\Phi}), \ldots, \lambda_r(\boldsymbol{\Sigma}-\boldsymbol{\Phi}), \underbrace{0, \ldots, 0}_{p-r \text{ times}}\big)\ \mathbf{U}' \tag{A.5}$$

gives equality in (A.4). This leads to the following result:

$$\begin{aligned} \inf_{\substack{\boldsymbol{\Theta}:\boldsymbol{\Theta}\succcurlyeq\mathbf{0}, \\ \operatorname{rank}(\boldsymbol{\Theta})\leq r}} \|\boldsymbol{\Sigma}-(\boldsymbol{\Theta}+\boldsymbol{\Phi})\|_q^q \;&=\; \|\boldsymbol{\Sigma}-(\boldsymbol{\Theta}^*+\boldsymbol{\Phi})\|_q^q \\ &=\; \sum_{i=r+1}^{p} \lambda_i^q(\boldsymbol{\Sigma}-\boldsymbol{\Phi}). \end{aligned} \tag{A.6}$$

(b) The minimizer $\boldsymbol{\Theta}^*$ of (A.6) is given by (A.5). In particular, if $\boldsymbol{\Phi}^*$ solves Problem (CFA$_q$) and we compute $\boldsymbol{\Theta}^*$ via (A.5) (with $\boldsymbol{\Phi} = \boldsymbol{\Phi}^*$), then the tuple $(\boldsymbol{\Phi}^*, \boldsymbol{\Theta}^*)$ solves Problem (2.4). This completes the proof of the proposition.

□

Proof of Proposition 2: We build upon the proof of Proposition 1. Note that any $\boldsymbol{\Phi}$ that is feasible for Problems (2.12) and (2.4) is PSD. Observe that $\boldsymbol{\Theta}^*$ (appearing in

the proof of Proposition 1) as given by (A.5) satisfies:

$$\mathbf{\Sigma} - \mathbf{\Phi} - \mathbf{\Theta}^* \succcurlyeq \mathbf{0} \implies \mathbf{\Sigma} - \mathbf{\Theta}^* \succcurlyeq \mathbf{0}, \tag{A.7}$$

where the right hand side of (A.7) follows because $\mathbf{\Phi} \succcurlyeq \mathbf{0}$. We have thus established that the solution $\mathbf{\Theta}^*$ to the following problem

$$\begin{aligned} \min_{\mathbf{\Theta}} \quad & \|(\mathbf{\Sigma} - \mathbf{\Phi}) - \mathbf{\Theta}\|_q^q \\ \text{s.t.} \quad & \mathbf{\Theta} \succcurlyeq \mathbf{0} \\ & \text{rank}(\mathbf{\Theta}) \leq r \end{aligned} \tag{A.8}$$

is feasible for the following optimization problem:

$$\begin{aligned} \min_{\mathbf{\Theta}} \quad & \|(\mathbf{\Sigma} - \mathbf{\Phi}) - \mathbf{\Theta}\|_q^q \\ \text{s.t.} \quad & \mathbf{\Theta} \succcurlyeq \mathbf{0} \\ & \text{rank}(\mathbf{\Theta}) \leq r \\ & \mathbf{\Sigma} - \mathbf{\Theta} \succcurlyeq \mathbf{0}. \end{aligned} \tag{A.9}$$

Since Problem (A.9) involves minimization over a subset of the feasible set of Problem (A.8), it follows that $\mathbf{\Theta}^*$ is also a minimizer for Problem (A.9). This completes the proof of the equivalence. □

Proof of Theorem 1: (a) The proof is based on ideas appearing in [121], where it was shown that the sum of the top r eigenvalues of a real symmetric matrix can be written as the solution to a linear SDO problem.

By an elegant classical result due to Fan [144], the smallest $(p-r)$ eigenvalues of a real symmetric matrix $\mathbf{A}$ can be written as

$$\begin{aligned} \sum_{i=r+1}^{p} \lambda_i(\mathbf{A}) \;=\; \inf_{\mathbf{V} \in \mathbb{R}^{p \times (p-r)}} \quad & \text{Tr}\left(\mathbf{V}'\mathbf{A}\mathbf{V}\right) \\ \text{s.t.} \quad & \mathbf{V}'\mathbf{V} = \mathbf{I}. \end{aligned} \tag{A.10}$$

We will show that the solution to the above nonconvex problem can be obtained

via the following linear (convex) SDO problem:

$$\begin{array}{ll} \min\limits_{\mathbf{W}} & \mathrm{Tr}\,(\mathbf{W}\mathbf{A}) \\ \text{s.t.} & \mathbf{I} \succcurlyeq \mathbf{W} \succcurlyeq \mathbf{0} \\ & \mathrm{Tr}(\mathbf{W}) = p - r. \end{array} \tag{A.11}$$

Clearly, Problem (A.11) is a convex relaxation of Problem (A.10)—hence its minimum value is at least smaller than $\sum_{i=r+1}^{p} \lambda_i(\mathbf{A})$. By standard results in convex analysis [130], it follows that the minimum of the above linear SDO problem (A.11) is attained at the extreme points of the feasible set of (A.11). The extreme points [121, 161] of this set are given by the set of orthonormal matrices of rank $p - r$:

$$\big\{\mathbf{V}\mathbf{V}' : \mathbf{V} \in \mathbb{R}^{p\times(p-r)} : \mathbf{V}'\mathbf{V} = \mathbf{I}\big\}.$$

It thus follows that the (global) minima of Problems (A.11) and (A.10) are the same. Applying this result to the PSD matrix $\mathbf{A} = (\mathbf{\Sigma} - \mathbf{\Phi})^q$ appearing in the objective of (CFA$_q$), we arrive at (2.14). This completes the proof of part (a).

(b) The statement follows from (A.5).

□

Proof of Proposition 3: Observe that, for every fixed $\mathbf{\Phi}$, the function $g_q(\mathbf{W}, \mathbf{\Phi})$ is concave (in fact, it is linear). Since $G_q(\mathbf{W})$ is obtained by taking a point-wise infimum with respect to $\mathbf{\Phi}$ of the concave function $g_q(\mathbf{W}, \mathbf{\Phi})$, the resulting function $G_q(\mathbf{W})$ is concave [38]. Finally, the expression of the subgradient (2.17) is an immediate consequence of Danskin's Theorem [23, 130]. □

Proof of Theorem 2. Using the concavity of the function $G_q(\mathbf{W})$ we have:

$$\begin{array}{rcl} G_q(\mathbf{W}^{(i+1)}) & \leq & G_q(\mathbf{W}^{(i)}) + \langle \nabla G_q(\mathbf{W}^{(i)}), \mathbf{W}^{(i+1)} - \mathbf{W}^{(i)} \rangle \\ & = & G_q(\mathbf{W}^{(i)}) + \Delta(\mathbf{W}^{(i)}). \end{array} \tag{A.12}$$

Note that $\Delta(\mathbf{W}^{(i)}) \leq 0$ and $G_q(\mathbf{W}^{(i+1)}) \leq G_q(\mathbf{W}^{(i)})$ for all $i \leq k$. Consequently, the (decreasing) sequence of objective values converge to some $G_q(\mathbf{W}^{(\infty)})$ and $\Delta(\mathbf{W}^{(\infty)}) \geq$

0. Adding up the terms in (A.12) from $i = 1, \ldots, k$ we have:

$$\begin{aligned} G_q(\mathbf{W}^{(\infty)}) - G_q(\mathbf{W}^{(1)}) &\leq G_q(\mathbf{W}^{(k+1)}) - G_q(\mathbf{W}^{(1)}) \\ &\leq -k \min_{i=1,\ldots,k} \left\{-\Delta(\mathbf{W}^{(i)})\right\} \end{aligned}$$

from which (2.23) follows. □

Proof of Theorem 4. By construction,

$$\max_{[\boldsymbol{\ell},\mathbf{u}]\in\texttt{Nodes}} \|\mathbf{u} - \boldsymbol{\ell}\|_1$$

decreases monotonically to zero as the iterative scheme proceeds. Therefore, to show convergence it suffices to show that the additive error at a node $\mathtt{n} = [\boldsymbol{\ell}, \mathbf{u}] \in \texttt{Nodes}$ is $O(\|\mathbf{u}-\boldsymbol{\ell}\|_1)$. Fix an arbitrary node $\mathtt{n} = [\boldsymbol{\ell}, \mathbf{u}] \in \texttt{Nodes}$. Without loss of generality, we may assume that (CFA_1) as restricted to $\operatorname{diag}(\boldsymbol{\Phi}) \in [\boldsymbol{\ell}, \mathbf{u}]$ has a feasible solution; as such, let $\boldsymbol{\Phi}^*$ be an optimal solution. Likewise, let $\boldsymbol{\Phi}^\dagger$ be an optimal solution ($\text{LS}_{\boldsymbol{\ell},\mathbf{u}}$). First note that $\sum_{i=r+1}^{p} \lambda_i(\boldsymbol{\Sigma} - \boldsymbol{\Phi}^*) \leq \sum_{i=r+1}^{p} \lambda_i(\boldsymbol{\Sigma} - \boldsymbol{\Phi}^\dagger)$. Now, by two applications of the bound in Theorem 3 and the fact that $\boldsymbol{\Phi}^\dagger$ is optimal to ($\text{LS}_{\boldsymbol{\ell},\mathbf{u}}$), we have that $\sum_{i=r+1}^{p} \lambda_i(\boldsymbol{\Sigma} - \boldsymbol{\Phi}^\dagger) \leq \sum_{i=r+1}^{p} \lambda_i(\boldsymbol{\Sigma} - \boldsymbol{\Phi}^*) + \|\mathbf{u} - \boldsymbol{\ell}\|_1/2$. Hence, the additive error satisfies

$$\left| \sum_{i=r+1}^{p} \lambda_i(\boldsymbol{\Sigma} - \boldsymbol{\Phi}^\dagger) - \sum_{i=r+1}^{p} \lambda_i(\boldsymbol{\Sigma} - \boldsymbol{\Phi}^*) \right| \leq \|\mathbf{u} - \boldsymbol{\ell}\|_1/2,$$

as was to be shown. □

A.2 Alternative conditional gradient approach

This section contains an alternative conditional gradient method for finding feasible solutions. Since Problem (2.14) involves the minimization of a smooth function over a compact convex set, the CG method requires iteratively solving the convex

optimization problem

$$\begin{aligned} \min_{\mathbf{W},\mathbf{\Phi}} \ & \left\langle \nabla g_p(\mathbf{W}^{(k)}, \mathbf{\Phi}^{(k)}), (\mathbf{W}, \mathbf{\Phi}) \right\rangle \\ \text{s.t.} \ & \mathbf{W} \in \mathcal{W}_{p-r} \\ & \mathbf{\Phi} \in \mathcal{F}_{\mathbf{\Sigma}}, \end{aligned} \tag{A.13}$$

where $\nabla g_p(\mathbf{W}^{(k)}, \mathbf{\Phi}^{(k)})$ is the gradient of $g_p(\mathbf{W}^{(k)}, \mathbf{\Phi}^{(k)})$ at the current iterate $(\mathbf{W}^{(k)}, \mathbf{\Phi}^{(k)})$. Note that due to the separability of the constraints in $\mathbf{W}$ and $\mathbf{\Phi}$, Problem (A.13) splits into two independent optimization problems with respect to $\mathbf{W}$ and $\mathbf{\Phi}$. The overall procedure is outlined in Algorithm 5.

Algorithm 5 A CG based algorithm for Problem (2.14)

1 Initialize with $(\mathbf{W}^{(1)}, \mathbf{\Phi}^{(1)})$, feasible for Problem (2.14) and repeat the following Steps 2-3 until the convergence criterion described in (A.18) is met.

2 Solve the linearized Problem (A.13), which requires solving two separate (convex) SDO problems over $\mathbf{W}$ and $\mathbf{\Phi}$:

$$\overline{\mathbf{W}}^{(k+1)} \in \arg\min_{\mathbf{W}\in\mathcal{W}_{p-r}} \langle \mathbf{W}, \nabla_{\mathbf{W}} g_p(\mathbf{W}^{(k)}, \mathbf{\Phi}^{(k)}) \rangle \tag{A.14}$$

$$\overline{\mathbf{\Phi}}^{(k+1)} \in \arg\min_{\mathbf{\Phi}\in\mathcal{F}_{\mathbf{\Sigma}}} \langle \mathbf{\Phi}, \nabla_{\mathbf{\Phi}} g_p(\mathbf{W}^{(k)}, \mathbf{\Phi}^{(k)}) \rangle \tag{A.15}$$

where $\nabla_{\mathbf{W}} g_p(\mathbf{W}, \mathbf{\Phi})$ (and $\nabla_{\mathbf{\Phi}} g_p(\mathbf{W}, \mathbf{\Phi})$) is the partial derivative with respect to $\mathbf{W}$ (respectively, $\mathbf{\Phi}$).

3 Obtain the new iterates:

$$\begin{aligned} \mathbf{W}^{(k+1)} &= \mathbf{W}^{(k)} + \eta_k(\overline{\mathbf{W}}^{(k+1)} - \mathbf{W}^{(k)}), \\ \mathbf{\Phi}^{(k+1)} &= \mathbf{\Phi}^{(k)} + \eta_k(\overline{\mathbf{\Phi}}^{(k+1)} - \mathbf{\Phi}^{(k)}). \end{aligned}$$

with $\eta_k \in [0, 1]$ chosen via an Armijo-type line-search rule [23].

Since $\nabla_{\mathbf{W}} g_p(\mathbf{W}^{(k)}, \mathbf{\Phi}^{(k)}) = (\mathbf{\Sigma} - \mathbf{\Phi}^{(k)})^q$, the update for $\mathbf{W}$ appearing in (A.14) requires solving

$$\min_{\mathbf{W}\in\mathcal{W}_{p-r}} \ \langle \mathbf{W}, (\mathbf{\Sigma} - \mathbf{\Phi}^{(k)})^q \rangle. \tag{A.16}$$

Similarly, the update for $\mathbf{\Phi}$ appearing in (A.15) requires solving:

$$\min_{\mathbf{\Phi} \in \mathcal{F}_{\mathbf{\Sigma}}} \quad \sum_{i=1}^{p} \Phi_i \ell_i, \tag{A.17}$$

where the vector $(\ell_1, \ldots, \ell_p)$ is given by $\operatorname{diag}(\ell_1, \ldots, \ell_p) = -q \ \operatorname{diag}\left(\mathbf{W}^{(k)}(\mathbf{\Sigma} - \mathbf{\Phi})^{q-1}\right)$, where $\operatorname{diag}(\mathbf{A})$ is a diagonal matrix having the same diagonal entries as $\mathbf{A}$. The sequence $(\mathbf{W}^{(k)}, \mathbf{\Phi}^{(k)})$ is recursively computed via Algorithm 5 until a convergence criterion is met:

$$g_p(\mathbf{W}^{(k)}, \mathbf{\Phi}^{(k)}) - g_p(\mathbf{W}^{(k+1)}, \mathbf{\Phi}^{(k+1)}) \leq \texttt{TOL} \cdot g_p(\mathbf{W}^{(k)}, \mathbf{\Phi}^{(k)}) \tag{A.18}$$

for some user-defined tolerance $\texttt{TOL} > 0$.

A tuple $(\mathbf{W}^*, \mathbf{\Phi}^*)$ satisfies the first order stationarity conditions [23] for Problem (2.14), if the following condition is satisfied:

$$\begin{aligned} \min_{\mathbf{W}, \mathbf{\Phi}} \quad & \langle \nabla g_p(\mathbf{W}^*, \mathbf{\Phi}^*), (\mathbf{W} - \mathbf{W}^*, \mathbf{\Phi} - \mathbf{\Phi}^*) \rangle \ \geq \qquad 0 \\ \text{s.t.} \quad & \mathbf{W} \in \mathcal{W}_{p-r} \\ & \mathbf{\Phi} \in \mathcal{F}_{\mathbf{\Sigma}}. \end{aligned}$$

Note that $\mathbf{\Phi}^*$ defined above also satisfies the first order stationarity conditions for problem (CFA_q).

The following theorem presents a global convergence guarantee for Algorithm 5:

Theorem 19 (*cf.* [23])**.** *Every limit point of a sequence* $(\mathbf{W}^{(k)}, \mathbf{\Phi}^{(k)})$ *produced by Algorithm 5 is a first order stationary point of the optimization Problem* (2.14)*.*

Numerical experiments (in line with those from Section 2.5) suggest that Algorithm 5 performs similarly to Algorithm 1, and therefore we only present the results for Algorithm 1 in the main text. Algorithm 1 has the advantage that it does not require a line search, unlike Algorithm 5. Finally, we note that for Algorithm 5 the update for $\mathbf{W}$ at iteration k for solving Problem (A.16) corresponds to $\widetilde{\mathbf{W}} = (\mathbf{\Sigma} - \mathbf{\Phi}^{(k)})^q$.

A.3 Alternative spectral inequality methods

In this section, we briefly study the landscape of eigenvalue inequalities and how they can be adapted to produce lower bounds similar to Weyl's method as detailed in Section 2.4.5. In particular, we consider computing lower bounds to

$$\min_{\boldsymbol{\Phi} \in \mathcal{F}_{\boldsymbol{\Sigma}}} \sum_{i=r+1}^{p} \lambda_i(\boldsymbol{\Sigma} - \boldsymbol{\Phi}) \tag{A.19}$$

using eigenvalue inequalities in conjunction with mixed integer semidefinite optimization ("MISDO") modeling techniques. We will conclude this section by relating this approach to that taken in Chapter 2. Throughout what follows, we abuse the notation $\boldsymbol{\Phi} \in \mathcal{F}_{\boldsymbol{\Sigma}}$: for a vector $\boldsymbol{\phi} \in \mathbb{R}^p$, we let $\boldsymbol{\phi} \in \mathcal{F}_{\boldsymbol{\Sigma}}$ denote that $\operatorname{diag}(\boldsymbol{\phi}) \in \mathcal{F}_{\boldsymbol{\Sigma}}$.

A.3.1 Ky Fan and mixed integer optimization

We begin by recalling the Ky Fan inequality [80, p. 250]: for any $\boldsymbol{\phi}, \boldsymbol{\gamma} \in \mathbb{R}^p$,

$$\sum_{i>r} \lambda_i(\boldsymbol{\Sigma} - \operatorname{diag}(\boldsymbol{\phi})) \geq \sum_{i>r} \left[\lambda_i(\boldsymbol{\Sigma} - \operatorname{diag}(\boldsymbol{\gamma})) + (\boldsymbol{\gamma} - \boldsymbol{\phi})_{(i)}\right]. \tag{KF}$$

Further, note that

$$\sum_{i>r} \lambda_i(\boldsymbol{\Sigma} - \operatorname{diag}(\boldsymbol{\phi})) = \sup_{\boldsymbol{\gamma} \in \mathbb{R}^p} \sum_{i>r} \left[\lambda_i(\boldsymbol{\Sigma} - \operatorname{diag}(\boldsymbol{\gamma})) + (\boldsymbol{\gamma} - \boldsymbol{\phi})_{(i)}\right]. \tag{A.20}$$

Observe that for a fixed $\boldsymbol{\gamma}$, the lower bound in (KF) can be written using auxiliary binary variables:

$$\begin{aligned}
\sum_{i>r} \left[\lambda_i(\boldsymbol{\Sigma} - \operatorname{diag}(\boldsymbol{\gamma})) + (\boldsymbol{\gamma} - \boldsymbol{\phi})_{(i)}\right] \quad = \quad & \min_{\mathbf{a},\mathbf{z}} \quad \sum_{i>r} \lambda_i(\boldsymbol{\Sigma} - \operatorname{diag}(\boldsymbol{\gamma})) + \mathbf{z}'\boldsymbol{\gamma} - \sum_i a_i \\
& \text{s.t.} \quad \mathbf{a} \leq \boldsymbol{\phi} \\
& \qquad\; \mathbf{a} \leq \operatorname{diag}(\mathbf{u})\mathbf{z} \\
& \qquad\; \textstyle\sum_i z_i = p - r \\
& \qquad\; \mathbf{z} \in \{0,1\}^p.
\end{aligned}$$

Here $\mathbf{u} \in \mathbb{R}^p$ is as defined in (2.37). As a result, we have the following:

Proposition 20. *For any $\boldsymbol{\gamma} \in \mathbb{R}^p$, problem (A.19) is lower bounded by the MISDO problem*

$$\begin{aligned} \min_{\boldsymbol{\phi},\mathbf{a},\mathbf{z}} \quad & \sum_{i>r} \lambda_i(\boldsymbol{\Sigma} - \operatorname{diag}(\boldsymbol{\gamma})) + \mathbf{z}'\boldsymbol{\gamma} - \sum_i a_i \\ \text{s. t.} \quad & \operatorname{diag}(\boldsymbol{\phi}) \in \mathcal{F}_{\boldsymbol{\Sigma}} \\ & \mathbf{a} \leq \boldsymbol{\phi} \\ & \mathbf{a} \leq \operatorname{diag}(\mathbf{u})\mathbf{z} \\ & \textstyle\sum_i z_i = p - r \\ & \mathbf{z} \in \{0,1\}^p. \end{aligned} \tag{A.21}$$

In words, (A.19) can be lower bounded by solving an auxiliary mixed integer optimization problem that is convex (except for the binary constraints). In what follows, we describe the quality of the lower bounds and how to obtain these bounds computationally.

Solving Problem (A.21): We briefly discuss how one might solve a MISDO problem such as (A.21). Unfortunately, there is little in terms of state-of-the-art MISDO solvers. To our knowledge, the only such available implementation is provided in an add-on to the software `SCIP` [1]; this implementation involves a rudimentary branch-and-bound scheme built on top of interior point methods. There are no commercial codes which solve MISDOs.

For this reason, we choose to instead leverage the striking power of modern linear mixed integer optimization ("MIO") solvers (such as `Gurobi` [73]) by solving the MISDOs using *linear* MIO problems with a cutting plane subroutine. In particular, we take the canonical reformulation of semidefinite constraints as a set of semi-infinite linear inequality constraints [92, 93, 95, 94]. The only nonlinear constraint present in (A.21) is $\boldsymbol{\Sigma} - \operatorname{diag}(\boldsymbol{\phi}) \succcurlyeq \mathbf{0}$. This can be equivalently written as

$$\mathbf{v}'(\boldsymbol{\Sigma} - \operatorname{diag}(\boldsymbol{\phi}))\mathbf{v} \geq 0 \quad \forall \mathbf{v} \in \mathbb{R}^p.$$

It follows that problem (A.21) can be solved as a linear MIO problem with callbacks

Problem	r	Gap (%) Ky Fan (A.21)	Gap (%) Weyl (2.41)
$A_1(2/10)$	1	6.08	7.66
$A_1(3/10)$	2	22.44	17.19
$A_1(2/20)$	1	1.26	1.89
$A_1(3/20)$	2	4.36	4.09
$A_1(5/20)$	4	47.69	21.53
$A_2(10)$	2	41.48	67.39
	3	62.75	74.21
$A_2(20)$	2	10.75	59.20
	3	17.88	63.25
	5	38.56	74.35

Table A.1: Computational results for Ky Fan approach in (A.21) with the choice of $\boldsymbol{\gamma} = \mathbf{u}^0$ as in (2.37). Problems are solved using `Gurobi` in `julia` via `JuMP`. Gap listed is in percentage relative to an incumbent found via Algorithm 1. "Weyl" refers to Weyl's method (Section 2.4.5).

[106]; this approach is easily implemented in the `julia` language using `JuMP` modeling tools [56].[1]

Quality of bounds: In Table A.1, we briefly show how the Ky Fan approach in (A.21) does as compared to Weyl's method for a particular choice of $\boldsymbol{\gamma}$ using such a cutting plane approach. If there exists an underlying very low rank solution, then it seems that the Ky Fan approach works reasonably well, although the quality of (A.21) degrades relative to Weyl's method as r increases. Of course, at the same time, the Ky Fan approach is computationally intensive, in contrast to Weyl's method. Finally, let us note that the Ky Fan bounds are highly sensitive to the choice of $\boldsymbol{\gamma}$, which is not surprising. Indeed, this raises the question of finding the best lower bound by solving over all possible $\boldsymbol{\gamma}$. We turn our attention to precisely this problem in the next section.

[1] Practically speaking, we terminate the callback procedure when $\lambda_p(\boldsymbol{\Sigma} - \operatorname{diag}(\boldsymbol{\phi}_{\text{cur}})) \geq -\texttt{TOL}$, where $\boldsymbol{\phi}_{\text{cur}}$ is the current (otherwise feasible) incumbent and `TOL` is a small numerical tolerance, e.g., $\texttt{TOL} = 10^{-4}$.

A.3.2 Optimal Ky Fan bounds

As seen above, the Ky Fan bounds (Proposition 20) can perform well, but depend on the choice of $\boldsymbol{\gamma}$. This raises the obvious question: what is the best possible choice of $\boldsymbol{\gamma}$, viz.,

$$\sup_{\boldsymbol{\gamma}} \min_{\boldsymbol{\phi} \in \mathcal{F}_{\boldsymbol{\Sigma}}} \sum_{i>r} \left[\lambda_i(\boldsymbol{\Sigma} - \operatorname{diag}(\boldsymbol{\gamma})) + (\boldsymbol{\gamma} - \boldsymbol{\phi})_{(i)} \right]. \tag{A.22}$$

We call this the *optimal* Ky Fan lower bound.

Numerical results suggest the marked strength of this approach. To proceed, we will first detail how one can solve a problem of the form (A.22), and then discuss a variety of numerical results.

Solving Problem (A.22): As per (A.20), the optimal Ky Fan lower bound (A.22) is a (weak) dual of (A.19). There are a variety of possible approaches to solving (A.22). In particular, observe that it is a concave maximization in $\boldsymbol{\gamma}$. For the present purposes, we consider a cutting plane approach, in a similar spirit to the one used above to solve (A.21) using linear semidefinite and MIO solvers instead of dedicated MISDO solvers. In particular, we proceed as follows: consider the reformulation of (A.22) as

$$\begin{aligned}
(\text{A.22}) \quad = \quad & \max_{\boldsymbol{\gamma}, \mathbf{M}, t, \kappa} \quad - \operatorname{Tr}(\mathbf{M}) + (p-r)t + \kappa \\
& \text{s.t.} \quad \mathbf{M} \succcurlyeq \mathbf{0} \\
& \qquad \boldsymbol{\Sigma} - \operatorname{diag}(\boldsymbol{\gamma}) + \mathbf{M} - \mathbf{I}t \succcurlyeq \mathbf{0} \\
& \qquad \kappa \leq \sum_{i>r} (\boldsymbol{\gamma} - \boldsymbol{\phi})_{(i)} \quad \forall\, \boldsymbol{\phi} \in \mathcal{F}_{\boldsymbol{\Sigma}} \\
\\
= \quad & \max_{\boldsymbol{\gamma}, \mathbf{M}, t, \kappa} \quad - \operatorname{Tr}(\mathbf{M}) + (p-r)t + \kappa \\
& \text{s.t.} \quad \mathbf{M} \succcurlyeq \mathbf{0} \\
& \qquad \boldsymbol{\Sigma} - \operatorname{diag}(\boldsymbol{\gamma}) + \mathbf{M} - \mathbf{I}t \succcurlyeq \mathbf{0} \\
& \qquad \kappa \leq \mathbf{z}'(\boldsymbol{\gamma} - \boldsymbol{\phi}) \quad \forall\, \boldsymbol{\phi} \in \mathcal{F}_{\boldsymbol{\Sigma}},\ \mathbf{z} \in \mathcal{Z}_{p-r},
\end{aligned}$$

where $\mathcal{Z}_{p-r} = \{\mathbf{z} \in \{0,1\}^p : \sum_i z_i = p - r\}$. Problem (A.22) is directly amenable to solution via cutting planes. In particular, the cut problem is precisely

$$\min_{\boldsymbol{\phi} \in \mathcal{F}_{\boldsymbol{\Sigma}}} \sum_{i>r} (\boldsymbol{\gamma} - \boldsymbol{\phi})_{(i)} = \min_{\substack{\boldsymbol{\phi} \in \mathcal{F}_{\boldsymbol{\Sigma}}, \\ \mathbf{z} \in \mathcal{Z}_{p-r}}} \mathbf{z}'(\boldsymbol{\gamma} - \boldsymbol{\phi}),$$

which can be solved using linear MIO solvers (again itself using cutting planes, as described in the preceding section).

In our numerical results that follow, we apply this cutting plane procedure, with the problem (A.22) formulated as

$$\begin{aligned}
&\max_{\boldsymbol{\gamma}, \mathbf{M}, t, \kappa} \; -\operatorname{Tr}(\mathbf{M}) + (p-r)t + \kappa \\
&\text{s.t.} \quad \mathbf{v}'\mathbf{M}\mathbf{v} \geq 0 \quad \forall\, \mathbf{v} \in \mathbb{R}^p && \text{(A.23a)} \\
&\qquad \mathbf{v}' \left(\boldsymbol{\Sigma} - \operatorname{diag}(\boldsymbol{\gamma}) + \mathbf{M} - \mathbf{I}t\right) \mathbf{v} \geq 0 \quad \forall\, \mathbf{v} \in \mathbb{R}^p && \text{(A.23b)} \\
&\qquad \kappa \leq \mathbf{z}'(\boldsymbol{\gamma} - \boldsymbol{\phi}) \quad \forall\, \boldsymbol{\phi} \in \mathcal{F}_{\boldsymbol{\Sigma}},\ \mathbf{z} \in \mathcal{Z}_{p-r}. && \text{(A.23c)}
\end{aligned}$$

This problem can be solving using linear optimization for the outer problem, with the three cut problems requiring eigenvalue decompositions (cuts A.23a and A.23b) and the solution to an MISDO (cut A.23c). Again, the third cut is itself solved computed using a linear MIO solver and cutting planes.[2]

Computational performance for optimal Ky Fan bounds: In Table A.1, we show how the optimal Ky Fan approach in (A.22) compares to the others. This small set of examples suggests that it performs exceptionally well as a lower bounding technique in some instances. Again, the key complication is that the method is computationally intensive, and therefore it is likely necessary to develop a more sophisticated, scalable algorithmic approach to solving (A.22).

[2] As a minor implementation detail, these cuts are not added in the relaxation of the problem, and therefore the relaxation will be unbounded from above. Therefore, we also include a tautological upper bound on the objective of the best feasible objective value found for (A.19) from some heuristic such as the CG-based methods in Chapter 2. As an even simpler approach, one could use $-\operatorname{Tr}(\mathbf{M}) + (p-r)t + \kappa \leq \sum_{i>r} \lambda_i(\boldsymbol{\Sigma})$ (the objective value in (A.19) with zero uniquenesses).

Problem	r	Gap (%)		
		OptKF (A.22)	Ky Fan (A.21)	Weyl (2.41)
$A_1(2/10)$	1	0.02	6.08	7.66
$A_1(3/10)$	2	0.32	22.44	17.19
$A_1(2/20)$	1	0.02	1.26	1.89
$A_1(3/20)$	2	0.01	4.36	4.09
$A_2(10)$	2	14.80	41.48	67.39
	3	35.37	62.75	74.21

Table A.2: Computational results for optimal Ky Fan approach in (A.22). Problems are solved using `Gurobi` in `julia` via `JuMP`. Gap listed is in percentage relative to an incumbent found via Algorithm 1. "OptKF" denotes the optimal Ky Fan method (A.22); "Ky Fan" denotes (A.21) with the choice of $\boldsymbol{\gamma} = \mathbf{u}^0$ as in (2.37); "Weyl" refers to Weyl's method (Section 2.4.5).

A.3.3 Iterative spectral methods and bilinear optimization

We close our discussion of alternative spectral approaches to lower bounds by considering an iterative approach to the Ky Fan lower bounds. In doing so, we show that this technique, appropriately modified, coincides with a bilinear optimization problem, precisely the sort of problem under consideration in Section 2.4.

To proceed, recall that the preceding results rely on a single fixed $\boldsymbol{\gamma} \in \mathbb{R}^p$ for which Proposition 20 can be applied. One could instead consider an iterative process: fix some initial choice of $\boldsymbol{\gamma} = \operatorname{diag}(\boldsymbol{\Phi}^{(0)})$, and then for $n = 1, 2, \ldots$, define

$$\boldsymbol{\Phi}^{(n)} \in \operatorname*{arg\,min}_{\boldsymbol{\Phi} \in \mathcal{F}_{\boldsymbol{\Sigma}}} \max_{0 \leq \ell < n} \sum_{i > r} \left[\lambda_i(\boldsymbol{\Sigma} - \boldsymbol{\Phi}^{(\ell)}) + (\operatorname{diag}(\boldsymbol{\Phi}^{(\ell)} - \boldsymbol{\Phi}))_{(i)} \right].$$

It is not difficult to argue that $\sum_{i>r} \lambda_i(\boldsymbol{\Sigma} - \boldsymbol{\Phi}^{(n)})$ converges to the optimal objective value of (A.19). The nth iteration requires the solution of a MISDO with np binary variables, and thus the problems become prohibitively large rather quickly. At the same time, the number of iterations necessary to produce an ϵ-optimal solution to (A.19) can also be exorbitant. (We neglect to include numerical results here.)

Finally, we conclude this section by observing that this iterative method can actually be strengthened further by exploiting concavity properties. In doing so, we see that a MISDO method such as the Ky Fan approach is substantively the same

as solving a bilinear optimization problem—precisely the approach taken to lower bounds in Chapter 2. The proof follows standard duality techniques and is omitted.

Proposition 21. *If* $\boldsymbol{\Phi}, \boldsymbol{\Phi}^{(0)}, \dots, \boldsymbol{\Phi}^{(n-1)} \in \mathcal{F}_{\boldsymbol{\Sigma}}$*, then*

$$\begin{aligned}\sum_{i>r} \lambda_i(\boldsymbol{\Sigma} - \boldsymbol{\Phi}) &\geq \min_{\substack{\mathbf{b}\in\mathbb{R}^p \\ \mathbf{0}\leq\mathbf{b}\leq\mathbf{1} \\ \sum_i b_i = p-k}} \max_{0\leq\ell<n} \left\{\mathbf{b}' \operatorname{diag}(\boldsymbol{\Phi}^{(\ell)}) + t_\ell\right\} - \mathbf{b}' \operatorname{diag}(\boldsymbol{\Phi}) \\ &\geq \max_{0\leq\ell<n} \sum_{i>r} \left[\lambda_i(\boldsymbol{\Sigma} - \boldsymbol{\Phi}^{(\ell)}) + (\operatorname{diag}(\boldsymbol{\Phi}^{(\ell)} - \boldsymbol{\Phi}))_{(i)}\right],\end{aligned}$$

where $t_\ell = \sum_{i>r} \lambda_i(\boldsymbol{\Sigma} - \boldsymbol{\Phi}^{(\ell)})$ *for* $\ell = 0, \dots, n-1$.

Of course, this should not be too surprising because the bilinear problem

$$\min_{\substack{\mathbf{b},\boldsymbol{\Phi}: \\ \boldsymbol{\Phi}\in\mathcal{F}_{\boldsymbol{\Sigma}}, \\ \mathbf{b}\in\operatorname{conv}(\mathcal{Z}_{p-r})}} \max_{0\leq\ell<n} \left\{\mathbf{b}' \operatorname{diag}(\boldsymbol{\Phi}^{(\ell)}) + \sum_{i=r+1}^{p} \lambda_i(\boldsymbol{\Sigma} - \boldsymbol{\Phi}^{(\ell)})\right\} - \mathbf{b}' \operatorname{diag}(\boldsymbol{\Phi})$$

is a (cutting-plane-based) relaxation of

$$\min_{\substack{\mathbf{b},\boldsymbol{\Phi}: \\ \boldsymbol{\Phi}\in\mathcal{F}_{\boldsymbol{\Sigma}}, \\ \mathbf{b}\in\operatorname{conv}(\mathcal{Z}_{p-r})}} \sup_{\boldsymbol{\gamma}\in\mathbb{R}^p} \left\{\mathbf{b}'\boldsymbol{\gamma} + \sum_{i=r+1}^{p} \lambda_i(\boldsymbol{\Sigma} - \operatorname{diag}(\boldsymbol{\gamma}))\right\} - \mathbf{b}' \operatorname{diag}(\boldsymbol{\Phi}),$$

which itself equals $\min_{\substack{\boldsymbol{\Phi}\in\mathcal{F}_{\boldsymbol{\Sigma}}, \\ \mathbf{W}\in\mathcal{W}_{p-r}}} \langle \mathbf{W}, \boldsymbol{\Sigma} - \boldsymbol{\Phi}\rangle$—precisely Problem (A.19).

Appendix B

Supplement for Chapter 3

This appendix contains proofs and additional technical results for the vector regression setting. We prove our results in the vector setting, from which the results on matrices follow as a direct corollary.

B.1 Proof of Theorem 10

Proof of Theorem 10. (a) We begin by proving the upper bound. Here we proceed by showing that $\overline{h}$ is precisely $\overline{h}(\boldsymbol{\beta}) = \lambda \delta_m(p,q)\|\boldsymbol{\beta}\|_{q^*}$. Now observe that for any $\boldsymbol{\Delta} \in \mathcal{U}_{F_q}$,

$$\|\boldsymbol{\Delta}\boldsymbol{\beta}\|_p \leq \delta_m(p,q)\|\boldsymbol{\Delta}\boldsymbol{\beta}\|_q \leq \delta_m(p,q)\|\boldsymbol{\Delta}\|_{F_q}\|\boldsymbol{\beta}\|_{q^*} \leq \delta_m(p,q)\lambda\|\boldsymbol{\beta}\|_{q^*}. \tag{B.1}$$

The first inequality follows by the definition of the discrepancy function δ_m. The second inequality follows from a well-known matrix inequality: $\|\boldsymbol{\Delta}\boldsymbol{\beta}\|_q \leq \|\boldsymbol{\Delta}\|_{F_q}\|\boldsymbol{\beta}\|_{q^*}$ (this follows from a simple application of Hölder's inequality). Now observe that in the chain of inequalities in (B.1), if one takes any $\mathbf{u} \in \arg\max \delta_m(p,q)$ and any $\mathbf{v} \in \arg\max_{\|\mathbf{v}\|_q=1} \mathbf{v}'\boldsymbol{\beta}$, then $\widehat{\boldsymbol{\Delta}} := \lambda \mathbf{u}\mathbf{v}' \in \mathcal{U}_{F_q}$ and $\|\widehat{\boldsymbol{\Delta}}\boldsymbol{\beta}\|_p = \delta_m(p,q)\lambda\|\boldsymbol{\beta}\|_{q^*}$. Hence, $\overline{h}(\boldsymbol{\beta}) = \delta_m(p,q)\lambda\|\boldsymbol{\beta}\|_{q^*}$. This proves the upper bound.

(b) We now prove that for $p \in \{1, \infty\}$ that one has equality for all $(\mathbf{z}, \boldsymbol{\beta}) \in \mathbb{R}^m \times \mathbb{R}^n$. First consider the case when $p = 1$. Fix $\mathbf{z} \in \mathbb{R}^m$. Again let $\mathbf{u} \in \arg\max \delta_m(1,q)$

and $\mathbf{v} \in \arg\max_{\|\mathbf{v}\|_q=1} \mathbf{v}'\boldsymbol{\beta}$. Without loss of generality we may assume that $\operatorname{sgn}(z_i) = \operatorname{sgn}(u_i)$ for $i = 1, \ldots, m$ (one may change the sign of entries of $\mathbf{u}$ and it is still in $\arg\max \delta_m(1, q)$). Then again we have $\widehat{\boldsymbol{\Delta}} := \lambda \mathbf{u}\mathbf{v}' \in \mathcal{U}_{F_q}$ and

$$\begin{aligned}\|\mathbf{z} + \widehat{\boldsymbol{\Delta}}\boldsymbol{\beta}\|_1 &= \|\mathbf{z} + \lambda \mathbf{u}\mathbf{v}'\boldsymbol{\beta}\|_1 = \|\mathbf{z} + \lambda \|\boldsymbol{\beta}\|_{q^*}\mathbf{u}\|_1 \\ &= \|\mathbf{z}\|_1 + \lambda\|\boldsymbol{\beta}\|_{q^*}\|\mathbf{u}\|_1 = \|\mathbf{z}\|_1 + \lambda\|\boldsymbol{\beta}\|_{q^*}\delta_m(1, q).\end{aligned}$$

Hence, one has equality in the upper bound for $p = 1$, as claimed.

We now turn our attention to the case $p = \infty$. Note that $\delta_m(\infty, q) = 1$ because $\|\mathbf{z}\|_\infty \le \|\mathbf{z}\|_q$ for all $\mathbf{z} \in \mathbb{R}^m$. Fix $\mathbf{z} \in \mathbb{R}^m$, and again let $\mathbf{v} \in \arg\max_{\|\mathbf{v}\|_q=1} \mathbf{v}'\boldsymbol{\beta}$. Let $\ell \in \{1, \ldots, m\}$ so that $|z_\ell| = \|\mathbf{z}\|_\infty$. Define $\mathbf{u} = \operatorname{sgn}(z_\ell)\mathbf{e}_\ell \in \mathbb{R}^m$, where $\mathbf{e}_\ell$ is the vector whose only nonzero entry is a 1 in the ℓth position. Now observe that $\widehat{\boldsymbol{\Delta}} := \lambda \mathbf{u}\mathbf{v}' \in \mathcal{U}_{F_q}$ and

$$\begin{aligned}\|\mathbf{z} + \widehat{\boldsymbol{\Delta}}\boldsymbol{\beta}\|_\infty &= \|\mathbf{z} + \operatorname{sgn}(z_\ell)\lambda\|\boldsymbol{\beta}\|_{q^*}\mathbf{e}_\ell\|_\infty \\ &= \|\mathbf{z}\|_\infty + \lambda\|\boldsymbol{\beta}\|_{q^*}\|\mathbf{e}_\ell\|_\infty = \|\mathbf{z}\|_\infty + \lambda\|\boldsymbol{\beta}\|_{q^*},\end{aligned}$$

which proves equality in (3.2), as was to be shown.

(c) To proceed, we examine the case where $p \in (1, \infty)$ and consider for which $(\mathbf{z}, \boldsymbol{\beta})$ the inequality in (3.2) is strict. Fix $\boldsymbol{\beta} \neq \mathbf{0}$. For $p \in (1, \infty)$ and $\mathbf{y}, \mathbf{z} \in \mathbb{R}^m$, one has by Minkowski's inequality that $\|\mathbf{y} + \mathbf{z}\|_p = \|\mathbf{y}\|_p + \|\mathbf{z}\|_p$ if and only if one of $\mathbf{y}$ or $\mathbf{z}$ is a nonnegative scalar multiple of the other. To have equality in (3.2), it must be that there exists some $\boldsymbol{\Delta} \in \arg\max_{\boldsymbol{\Delta} \in \mathcal{U}_{F_q}} \|\boldsymbol{\Delta}\boldsymbol{\beta}\|_p$ for which $\|\mathbf{z} + \boldsymbol{\Delta}\boldsymbol{\beta}\|_p = \|\mathbf{z}\|_p + \|\boldsymbol{\Delta}\boldsymbol{\beta}\|_p$. For any $\mathbf{z} \neq \mathbf{0}$ this observation, combined with Minkowski's inequality, implies that

$$\|\boldsymbol{\Delta}\|_{F_q} = \lambda, \ \boldsymbol{\Delta}\boldsymbol{\beta} = \mu\mathbf{z} \ \text{ for some } \mu \ge 0, \ \text{ and } \|\boldsymbol{\Delta}\boldsymbol{\beta}\|_p = \lambda\delta_m(p, q)\|\boldsymbol{\beta}\|_{q^*}.$$

The first and last equalities imply that $\boldsymbol{\Delta}\boldsymbol{\beta} \in \lambda\|\boldsymbol{\beta}\|_{q^*} \arg\max \delta_m(p, q)$. Note that $\arg\max \delta_m(p, q)$ is finite whenever $p \neq q$ and $m \ge 2$, a geometric property

of ℓ_p balls. Hence, taking any $\mathbf{z}$ which is not a scalar multiple of a point in $\arg\max \delta_m(p,q)$ implies by Minkowski's inequality that

$$\max_{\mathbf{\Delta}\in\mathcal{U}_{F_q}} \|\mathbf{z}+\mathbf{\Delta}\boldsymbol{\beta}\|_p < \|\mathbf{z}\|_p + \lambda\delta_m(p,q)\|\boldsymbol{\beta}\|_{q^*}.$$

Hence, for any $\boldsymbol{\beta}\neq\mathbf{0}$, the inequality in (3.2) is strict for all $\mathbf{z}$ not in a finite union of one-dimensional subspaces, so long as $p\in(1,\infty)$, $p\neq q$, and $m\geq 2$.

(d) We now prove the lower bound in (3.3). If $\mathbf{z}=\mathbf{0}$ then there is nothing to show, and therefore we assume $\mathbf{z}\neq\mathbf{0}$. Let $\mathbf{v}\in\mathbb{R}^n$ so that

$$\mathbf{v}\in\arg\max_{\|\mathbf{v}\|_q=1} \mathbf{v}'\boldsymbol{\beta}.$$

Hence $\mathbf{v}'\boldsymbol{\beta}=\|\boldsymbol{\beta}\|_{q^*}$ by the definition of the dual norm. Define $\widehat{\mathbf{\Delta}}=\frac{\lambda}{\|\mathbf{z}\|_q}\mathbf{z}\mathbf{v}'$. Observe that $\widehat{\mathbf{\Delta}}\in\mathcal{U}_{F_q}$. Further, note that $\|\mathbf{z}\|_q\leq\delta_m(q,p)\|\mathbf{z}\|_p$ by definition of δ_m and therefore $1/\delta_m(q,p)\leq\|\mathbf{z}\|_p/\|\mathbf{z}\|_q$. Putting things together,

$$\begin{aligned}
\|\mathbf{z}\|_p + \frac{\lambda\|\boldsymbol{\beta}\|_{q^*}}{\delta_m(q,p)} &\leq \|\mathbf{z}\|_p + \frac{\lambda\|\mathbf{z}\|_p\|\boldsymbol{\beta}\|_{q^*}}{\|\mathbf{z}\|_q} \\
&= \|\mathbf{z}\|_p\left(1+\frac{\lambda\|\boldsymbol{\beta}\|_{q^*}}{\|\mathbf{z}\|_q}\right) \\
&= \|\mathbf{z}+\widehat{\mathbf{\Delta}}\boldsymbol{\beta}\|_p \\
&\leq \max_{\mathbf{\Delta}\in\mathcal{U}_{F_q}} \|\mathbf{z}+\mathbf{\Delta}\boldsymbol{\beta}\|_p.
\end{aligned}$$

This completes the proof of the lower bound.

(e) To conclude we prove that the gap in (3.3) can be made arbitrarily small for $p\in(1,\infty)$. We proceed in several steps. We first prove that for any $\mathbf{z}\neq\mathbf{0}$ that

$$\lim_{\alpha\to\infty}\left(\max_{\mathbf{\Delta}\in\mathcal{U}_{F_q}} \|\alpha\mathbf{z}+\mathbf{\Delta}\boldsymbol{\beta}\|_p - \|\alpha\mathbf{z}\|_p\right) = \frac{\lambda\|\boldsymbol{\beta}\|_{q^*}\|\mathbf{z}^{p-1}\|_{q^*}}{\|\mathbf{z}\|_p^{p-1}}, \tag{B.2}$$

where we use the shorthand $\mathbf{z}^{p-1}$ to denote the vector in $\mathbb{R}^m$ whose ith entry is

$|z_i|^{p-1}$. Observe that

$$\max_{\boldsymbol{\Delta} \in \mathcal{U}_{F_q}} \|\alpha \mathbf{z} + \boldsymbol{\Delta}\boldsymbol{\beta}\|_p = \max_{\|\mathbf{u}\|_q \leq \lambda \|\boldsymbol{\beta}\|_{q^*}} \|\alpha \mathbf{z} + \mathbf{u}\|_p.$$

It is easy to argue that we may assume without any loss of generality that $\mathbf{u} \in \arg\max_{\|\mathbf{u}\|_q \leq \lambda \|\boldsymbol{\beta}\|_{q^*}} \|\alpha \mathbf{z} + \mathbf{u}\|_p$ has $\operatorname{sgn}(u_i) = \operatorname{sgn}(\alpha z_i)$, where

$$\operatorname{sgn}(a) = \begin{cases} 1, & a \geq 0 \\ -1, & a < 0. \end{cases}$$

Therefore, we restrict our attention to $\mathbf{z} \geq \mathbf{0}$, $\mathbf{z} \neq \mathbf{0}$, and $\mathbf{u} \geq \mathbf{0}$. For any $\mathbf{u}$ such that $\|\mathbf{u}\|_q \leq \lambda \|\boldsymbol{\beta}\|_{q^*}$ and $\mathbf{u} \geq \mathbf{0}$, note that

$$\begin{aligned} \lim_{\alpha \to \infty} \|\alpha \mathbf{z} + \mathbf{u}\|_p - \|\alpha \mathbf{z}\|_p &= \lim_{\alpha \to \infty} \frac{\|\mathbf{z} + \mathbf{u}/\alpha\|_p - \|\mathbf{z}\|_p}{1/\alpha} \\ &= \lim_{\bar{\alpha} \to 0^+} \frac{\|\mathbf{z} + \bar{\alpha}\mathbf{u}\|_p - \|\mathbf{z}\|_p}{\bar{\alpha}} \\ &= \frac{d}{d\bar{\alpha}}\bigg|_{\bar{\alpha}=0} \|\mathbf{z} + \bar{\alpha}\mathbf{u}\|_p \\ &= \frac{\mathbf{u}'\mathbf{z}^{p-1}}{\|\mathbf{z}\|_p^{p-1}}. \end{aligned}$$

We can now proceed to finish the claim in (B.2) (still restricting attention to $\mathbf{z} \geq \mathbf{0}$ without loss of generality). By the above arguments, for any $\mathbf{u} \geq \mathbf{0}$ and any $\epsilon > 0$ there exists some $\hat{\alpha} = \hat{\alpha}(\mathbf{u}) > 0$ sufficiently large so that for all $\alpha > \hat{\alpha}$,

$$\left| \|\alpha \mathbf{z} + \mathbf{u}\|_p - \|\alpha \mathbf{z}\|_p - \frac{\mathbf{u}'\mathbf{z}^{p-1}}{\|\mathbf{z}\|_p^{p-1}} \right| \leq \epsilon.$$

It remains to be shown that for any $\epsilon > 0$ there exists some $\hat{\alpha}$ so that for all $\alpha > \hat{\alpha}$,

$$\left| \left(\max_{\|\mathbf{u}\|_q \leq \lambda \|\boldsymbol{\beta}\|_{q^*}} \|\alpha \mathbf{z} + \mathbf{u}\|_p - \|\alpha \mathbf{z}\|_p \right) - \left(\max_{\|\mathbf{u}\|_q \leq \lambda \|\boldsymbol{\beta}\|_{q^*}} \frac{\mathbf{u}'\mathbf{z}^{p-1}}{\|\mathbf{z}\|_p^{p-1}} \right) \right| \leq \epsilon.$$

We prove this as follows. Let $\epsilon > 0$. Choose points $\{\mathbf{u}_1, \ldots, \mathbf{u}_M\} \subseteq \mathbb{R}^m$ with

$\|\mathbf{u}_j\|_q = \lambda\|\boldsymbol{\beta}\|_{q^*}$ $\forall j$ so that for any $\mathbf{u} \in \mathbb{R}^m$ with $\|\mathbf{u}\|_q = \lambda\|\boldsymbol{\beta}\|_{q^*}$, there exists some j so that $\|\mathbf{u} - \mathbf{u}_j\|_p \leq \epsilon/3$ (note that our choice of ℓ_p here is intentional). Now observe that for any α,

$$\begin{aligned}
\max_j \|\alpha\mathbf{z} + \mathbf{u}_j\|_p &\leq \max_{\|\mathbf{u}\|_q \leq \lambda\|\boldsymbol{\beta}\|_{q^*}} \|\alpha\mathbf{z} + \mathbf{u}\|_p \\
&\leq \max_j \left(\max_{\|\mathbf{u}-\mathbf{u}_j\|_p \leq \epsilon/3} \|\alpha\mathbf{z} + \mathbf{u}\|_p \right) \\
&= \max_j \left(\max_{\|\bar{\mathbf{u}}\|_p \leq \epsilon/3} \|\alpha\mathbf{z} + \mathbf{u}_j + \bar{\mathbf{u}}\|_p \right) \\
&\leq \max_j \left(\max_{\|\bar{\mathbf{u}}\|_p \leq \epsilon/3} \|\alpha\mathbf{z} + \mathbf{u}_j\|_p + \|\bar{\mathbf{u}}\|_p \right) \\
&= \epsilon/3 + \max_j \|\alpha\mathbf{z} + \mathbf{u}_j\|_p.
\end{aligned}$$

Similarly, one has for $\bar{\mathbf{z}} = \mathbf{z}^{p-1}/\|\mathbf{z}\|_p^{p-1}$ that $\left|\max_j \mathbf{u}_j'\bar{\mathbf{z}} - \max_{\|\mathbf{u}\|_q \leq \lambda\|\boldsymbol{\beta}\|_{q^*}} \mathbf{u}'\bar{\mathbf{z}}\right| \leq \epsilon/3$. (This uses the fact that $\|\bar{\mathbf{z}}\|_{p^*} = 1$.) Now for each j choose $\hat{\alpha}_j$ so that for all $\alpha > \hat{\alpha}_j$,

$$\left| \|\alpha\mathbf{z} + \mathbf{u}_j\|_p - \|\alpha\mathbf{z}\|_p - \mathbf{u}_j'\bar{\mathbf{z}} \right| \leq \epsilon/3.$$

Define $\hat{\alpha} = \max_j \hat{\alpha}_j$. Now observe that by combining the above two observations, one has for any $\alpha > \hat{\alpha}$ that

$$\begin{aligned}
&\left| \left(\max_{\|\mathbf{u}\|_q \leq \lambda\|\boldsymbol{\beta}\|_{q^*}} \|\alpha\mathbf{z} + \mathbf{u}\|_p - \|\alpha\mathbf{z}\|_p \right) - \left(\max_{\|\mathbf{u}\|_q \leq \lambda\|\boldsymbol{\beta}\|_{q^*}} \mathbf{u}'\bar{\mathbf{z}} \right) \right| \leq \\
&\qquad \leq 2\epsilon/3 + \left| \left(\max_j \|\alpha\mathbf{z} + \mathbf{u}_j\|_p - \|\alpha\mathbf{z}\|_p \right) - \left(\max_\ell \mathbf{u}_\ell'\bar{\mathbf{z}} \right) \right| \\
&\qquad \leq 2\epsilon/3 + \max_j \left| \|\alpha\mathbf{z} + \mathbf{u}_j\|_p - \|\alpha\mathbf{z}\|_p - \mathbf{u}_j'\bar{\mathbf{z}} \right| \\
&\qquad \leq 2\epsilon/3 + \epsilon/3 = \epsilon.
\end{aligned}$$

Noting that $\max_{\|\mathbf{u}\|_q \leq \lambda\|\boldsymbol{\beta}\|_{q^*}} \mathbf{u}'\bar{\mathbf{z}} = \lambda\|\boldsymbol{\beta}\|_{q^*}\|\bar{\mathbf{z}}\|_{q^*}$ concludes the proof of (B.2). We now claim that

$$\min_{\mathbf{z}} \frac{\|\mathbf{z}^{p-1}\|_{q^*}}{\|\mathbf{z}\|_p^{p-1}} = \frac{1}{\delta_m(q,p)}. \tag{B.3}$$

First note that

$$\min_{\mathbf{z}} \frac{\|\mathbf{z}^{p-1}\|_{q^*}}{\|\mathbf{z}\|_p^{p-1}} = \min_{\tilde{\mathbf{z}}} \frac{\|\tilde{\mathbf{z}}\|_{q^*}}{\|\tilde{\mathbf{z}}\|_{p^*}}. \tag{B.4}$$

We prove this as follows: given $\mathbf{z}$, let $\tilde{\mathbf{z}} = \mathbf{z}^{p-1}$. Then one can show that $\|\tilde{\mathbf{z}}\|_{p^*}/\|\mathbf{z}\|_p^{p-1} = 1$, and so $\|\tilde{\mathbf{z}}\|_{p^*}/\|\tilde{\mathbf{z}}\|_{q^*} = \|\mathbf{z}\|_p^{p-1}/\|\mathbf{z}^{p-1}\|_{q^*}$. The converse is similar, proving (B.4). Finally, note that

$$\min_{\tilde{\mathbf{z}}} \frac{\|\tilde{\mathbf{z}}\|_{q^*}}{\|\tilde{\mathbf{z}}\|_{p^*}} = \frac{1}{\delta_m(p^*, q^*)}$$

which follows from an elementary analysis using the definition of δ_m. Combined with the observation that $\delta_m(p^*, q^*) = \delta_m(q, p)$, which follows by a simply duality argument (or by inspecting the formula), we have that (B.3) is proven. To finish the argument, pick any $\mathbf{z} \in \arg\min_{\mathbf{z}} \|\mathbf{z}^{p-1}\|_{q^*}/\|\mathbf{z}\|_p^{p-1}$. Per (B.3), $\|\mathbf{z}^{p-1}\|_{q^*}/\|\mathbf{z}\|_p^{p-1} = 1/\delta_m(q, p)$. Hence, now applying (B.2), given any $\epsilon > 0$, there exists some $\alpha > 0$ large enough so that

$$\left| \left(\max_{\mathbf{\Delta} \in \mathcal{U}_{F_q}} \|\alpha\mathbf{z} + \mathbf{\Delta}\boldsymbol{\beta}\|_p \right) - \left(\|\alpha\mathbf{z}\|_p + \frac{\lambda}{\delta_m(q, p)} \|\boldsymbol{\beta}\|_{q^*} \right) \right| \leq \epsilon.$$

Therefore, the gap in the lower bound in (3.3) can be made arbitrarily small for any $\boldsymbol{\beta} \in \mathbb{R}^n$. This concludes the proof.

□

B.2 Counterexample

This section includes an example of choice of loss function and uncertainty set under which (a) regularization is not equivalent to robustification in general and (b) there exist problem instances for which the regularization path and robustification path are different.

To proceed, let $m = 2$ and $n = 2$, and consider $\mathcal{U} = \mathcal{U}_{(1,1)}$ and loss function ℓ_2,

with $\mathbf{y} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$ and $\mathbf{X} = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}$. In symbols, the problem of interest is

$$\min_{\boldsymbol{\beta}} \max_{\boldsymbol{\Delta} \in \mathcal{U}_{(1,1)}} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_2. \tag{B.5}$$

For fixed $\boldsymbol{\beta}$, the objective can be rewritten exactly as

$$\begin{aligned}
&\max_{\boldsymbol{\Delta} \in \mathcal{U}_{(1,1)}} \|\mathbf{y} - (\mathbf{X} + \boldsymbol{\Delta})\boldsymbol{\beta}\|_2 \\
&= \max_{\substack{\mathbf{u}: \\ \|\mathbf{u}\|_1 \le \lambda \|\boldsymbol{\beta}\|_1}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta} + \mathbf{u}\|_2 \\
&= \max \left\{ \left\| \mathbf{y} - \mathbf{X}\boldsymbol{\beta} \pm \begin{pmatrix} \lambda \|\boldsymbol{\beta}\|_1 \\ 0 \end{pmatrix} \right\|_2, \left\| \mathbf{y} - \mathbf{X}\boldsymbol{\beta} \pm \begin{pmatrix} 0 \\ \lambda \|\boldsymbol{\beta}\|_1 \end{pmatrix} \right\|_2 \right\} \\
&= \max \left\{ \left\| \mathbf{y} - \left(\mathbf{X} + \begin{pmatrix} \pm\lambda & \pm\lambda \\ 0 & 0 \end{pmatrix} \right) \boldsymbol{\beta} \right\|_2, \left\| \mathbf{y} - \left(\mathbf{X} + \begin{pmatrix} 0 & 0 \\ \pm\lambda & \pm\lambda \end{pmatrix} \right) \boldsymbol{\beta} \right\|_2 \right\} \\
&= \max_{\mathbf{S} \in \mathcal{S}} \|\mathbf{y} - (\mathbf{X} + \mathbf{S})\boldsymbol{\beta}\|_2,
\end{aligned}$$

where $\mathcal{S}$ is the set of eight matrices $\left\{ \begin{pmatrix} \pm\lambda & \pm\lambda \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 \\ \pm\lambda & \pm\lambda \end{pmatrix} \right\}$. The first step follows by inspecting the definition of $\mathcal{U}_{(1,1)}$; the second step follows from the convexity of $\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta} + \mathbf{u}\|_2$ (in particular, the maximum of the convex function is attained at an extreme point of $\{\mathbf{u} : \|\mathbf{u}\|_1 \le \lambda \|\boldsymbol{\beta}\|_1\}$); and the third step follows from the definition of the ℓ_1 norm. Hence, the objective is the maximum of eight modified ℓ_2 losses.

Let us consider $\lambda = 1/2$. We claim that $\boldsymbol{\beta}^* = (1, 1)$ is an optimal solution to (B.5) with objective value $\sqrt{5}$. We will argue that $\boldsymbol{\beta}^*$ is optimal by exhibiting a dual feasible solution with the same objective value. It is easy to see that the dual (lower bounding) problem is

$$\max_{\substack{\boldsymbol{\mu} \in \mathbb{R}^{\mathcal{S}}: \\ \sum_{\mathbf{S}} \mu_{\mathbf{S}} = 1 \\ \boldsymbol{\mu} \ge \mathbf{0}}} \min_{\boldsymbol{\beta}} \sum_{\mathbf{S}} \mu_{\mathbf{S}} \|\mathbf{y} - (\mathbf{X} + \mathbf{S})\boldsymbol{\beta}\|_2,$$

where there are eight variables $\{\mu_{\mathbf{S}} : \mathbf{S} \in \mathcal{S}\}$, one for each $\mathbf{S} \in \mathcal{S}$. Note that weak

duality of the two problems is immediate. Let $\boldsymbol{\mu}^*$ be the dual feasible point with $\mu_{\mathbf{S}} = 0$ for $\mathbf{S} \in \mathcal{S} \setminus \{\mathbf{S}_1\}$, where $\mathbf{S}_1 = \begin{pmatrix} 0 & 0 \\ -1/2 & -1/2 \end{pmatrix}$, and $\mu_{\mathbf{S}_1} = 1$. Hence, a lower bound to (B.5) is

$$\min_{\boldsymbol{\beta}} \sum_{\mathbf{S}} \mu^*_{\mathbf{S}} \|\mathbf{y} - (\mathbf{X} + \mathbf{S})\boldsymbol{\beta}\|_2 = \min_{\boldsymbol{\beta}} \|\mathbf{y} - (\mathbf{X} + \mathbf{S}_1)\boldsymbol{\beta}\|_2 = \sqrt{5}.$$

The final step follows by calculus, using that $\mathbf{X} + \mathbf{S}_1 = \begin{pmatrix} 1 & -1 \\ -1/2 & 1/2 \end{pmatrix}$. It follows that $\boldsymbol{\beta}^* = (1, 1)$ (with objective value $\sqrt{5}$) must be optimal to (B.5), as claimed.

We now turn our attention to the central point of interest in this Appendix, namely, that $\boldsymbol{\beta}^* = (1, 1)$ is *not* a solution to the corresponding regularization problem, viz.

$$\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2 + \rho\|\boldsymbol{\beta}\|_1, \tag{B.6}$$

for any $\rho \in (0, \infty)$ (*cf.* Proposition 5). The solution path of (B.6) ranging over ρ is immediate from the proximal (soft-thresholding) analysis of the Lasso. In particular, it is the set of points $\{(3\alpha, 2\alpha) : \alpha \in [0, 1]\}$. This set does not contain $\boldsymbol{\beta}^* = (1, 1)$, and hence the regularization problem does not solve the robustification problem (B.5) with $\lambda = 1/2$ for any corresponding choice of ρ. (If one does not wish to rely on such an indirect analysis, note that one can solve the equivalent problem to (B.6) of $\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \mu\|\boldsymbol{\beta}\|_1$, ranging over $\mu \in (0, \infty)$. The objective is differentiable at the point $\boldsymbol{\beta}^* = (1, 1)$, and the gradient is $(-2 + \mu, 0 + \mu)$. As this is never $(0, 0)$, $\boldsymbol{\beta}^*$ can never be optimal to this problem, and consequently can never be optimal to (B.6). Despite the more direct analysis, the conclusion is the same.)

To show the converse, we can use the same example. In particular, consider the solution $(3/2, 1)$ to (B.6) (the choice of ρ for which this is optimal is irrelevant for our purposes). We must show that $(3/2, 1)$ is never a solution to (B.5) for any choice of λ. Let us first inspect the objective of (B.5) for $\boldsymbol{\beta}^* = (3/2, 1)$. It can be computed to be $\sqrt{1/4 + (1 + 5\lambda/2)^2}$. We make two observations:

(1) For any $0 \leq \lambda < (\sqrt{19} + 2)/15$, the point $(3, 2)$ has strictly smaller objective

(namely, 5λ) than $\boldsymbol{\beta}^*$, and so $\boldsymbol{\beta}^*$ is not optimal to (B.5) whenever $\lambda < (\sqrt{19}+2)/15 \approx 0.424$.

(2) Similarly, for any $\lambda > (\sqrt{31}-2)/9$, the point $(1,1)$ has strictly smaller objective (namely, $\sqrt{4\lambda^2+4\lambda+2}$) than $\boldsymbol{\beta}^*$, and so $\boldsymbol{\beta}^*$ is not optimal to (B.5) whenever $\lambda > (\sqrt{31}-2)/9 \approx 0.396$.

Because the intervals $[(\sqrt{19}+2)/15, \infty)$ and $[0, (\sqrt{31}-2)/9]$ have no overlap, the point $\boldsymbol{\beta}^* = (3/2, 1)$ cannot be a solution to (B.5) for any choice of λ.

Thus, the robustification and regularization solutions for the problems connected via Theorem 10 do not need to coincide. The statement of Theorem 11 follows as desired.

Appendix C

Supplement for Chapter 4

C.1 General min-max representation of SLOPE

For completeness, in this appendix we include the more general representation of the SLOPE penalty $R_{\text{SLOPE}(\mathbf{w})}$ in the same spirit of Proposition 15. Here we work with SLOPE in its most general form, namely,

$$R_{\text{SLOPE}(\mathbf{w})}(\boldsymbol{\beta}) = \sum_{i=1}^{p} w_i |\beta_{(i)}|,$$

where $\mathbf{w}$ is a (fixed) vector of weights with $w_1 \geq w_2 \geq \cdots \geq w_p \geq 0$ and $w_1 > 0$.

To describe the general min-max representation, we first set some notation. For a matrix $\boldsymbol{\Delta} \in \mathbb{R}^{n \times p}$, we let $\boldsymbol{\nu}(\boldsymbol{\Delta}) \in \mathbb{R}^p$ be the vector $(\|\boldsymbol{\Delta}_1\|_2, \ldots, \|\boldsymbol{\Delta}_p\|_2)$ with entries sorted so that $\nu_1 \geq \nu_2 \geq \cdots \geq \nu_p$. As usual, for two vectors $\mathbf{x}$ and $\mathbf{y}$, we use $\mathbf{x} \leq \mathbf{y}$ to denote that coordinate-wise inequality holds. With this notation, we have the following:

Proposition 22. *Problem (4.9) with uncertainty set*

$$\mathcal{U}_{\mathbf{w}} = \{\boldsymbol{\Delta} : \boldsymbol{\nu}(\boldsymbol{\Delta}) \leq \mathbf{w}\}$$

is equivalent to problem (4.3) with $R(\boldsymbol{\beta}) = R_{\text{SLOPE}(\mathbf{w})}(\boldsymbol{\beta})$. *Further, problem (4.9)*

with uncertainty set

$$\mathcal{U}_{\mathbf{w}} = \left\{ \boldsymbol{\Delta} : \|\boldsymbol{\Delta}\boldsymbol{\phi}\|_2 \leq R_{\mathrm{SLOPE}(\mathbf{w})}(\boldsymbol{\phi}) \; \forall \boldsymbol{\phi} \right\}$$

is equivalent to problem (4.3) *with* $R(\boldsymbol{\beta}) = R_{\mathrm{SLOPE}(\mathbf{w})}(\boldsymbol{\beta})$.

The proof, like the proof of Proposition 15, follows basic techniques and is omitted.

C.2 Supplementary details for algorithms

This section contains further details on algorithms as discussed in Section 4.5. The presentation here is primarily self-contained. Note that the alternating minimization scheme based on difference-of-convex optimization can be found in [72].

C.2.1 Alternating minimization scheme

Let us set the following notation:

$$\begin{aligned} f(\boldsymbol{\beta}) &= \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2/2 + \lambda T_k(\boldsymbol{\beta}) + \eta\|\boldsymbol{\beta}\|_1, \\ f_1(\boldsymbol{\beta}) &= \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2/2 + (\eta + \lambda)\|\boldsymbol{\beta}\|_1, \\ f_2(\boldsymbol{\beta}) &= \lambda \textstyle\sum_{i=1}^{k} |\beta_{(i)}|. \end{aligned}$$

Definition 4. *For any function* $F : \mathbb{R}^p \to \mathbb{R}$ *and* $\epsilon \geq 0$*, we define the* ϵ*-subdifferential of* F *at* $\boldsymbol{\beta}_0 \in \mathbb{R}^p$ *to be the set* $\partial_\epsilon F(\boldsymbol{\beta}_0)$ *defined as*

$$\left\{ \boldsymbol{\gamma} \in \mathbb{R}^p \; : \; F(\boldsymbol{\beta}) - F(\boldsymbol{\beta}_0) \geq \langle \boldsymbol{\gamma}, \boldsymbol{\beta} - \boldsymbol{\beta}_0 \rangle - \epsilon \; \forall \; \boldsymbol{\beta} \in \mathbb{R}^p \right\}.$$

In particular, when $\epsilon = 0$*, we refer to* $\partial_0 F(\boldsymbol{\beta}_0)$ *as the subdifferential of* F *at* $\boldsymbol{\beta}_0$*, and we will denote this as* $\partial F(\boldsymbol{\beta}_0)$.

Using this definition, we have the following result precisely characterizing local and global optima of (4.47).

Theorem 20. *(a) A point* $\boldsymbol{\beta}^*$ *is a local minimum of* f *if and only if* $\partial f_2(\boldsymbol{\beta}^*) \subseteq \partial f_1(\boldsymbol{\beta}^*)$.

(b) A point $\boldsymbol{\beta}^*$ is a global minimum of f if and only if $\partial_\epsilon f_2(\boldsymbol{\beta}^*) \subseteq \partial_\epsilon f_1(\boldsymbol{\beta}^*)$ for all $\epsilon \geq 0$.

Proof. This is a direct application of results in [145, Thm. 1]. Part (b) is immediate. The forward implication of part (a) is immediate as well; the converse implication follows by observing that f_2 is a *polyhedral* convex function [4, Thm. 1(ii)] (see definition therein). □

Let us note that ∂f_1 and ∂f_2 are both easily computable, and hence, local optimality can be verified given some candidate $\boldsymbol{\beta}^*$ per Theorem 20.[1] We now discuss the associated alternating minimization scheme (or equivalently, as a sequential linearization scheme), shown in Algorithm 3 for finding local optima of (4.47) by making use of Theorem 20. Through what follows, we make use of the standard notion of a conjugate function, defined as follows:

Definition 5. *For any function $F : \mathbb{R}^p \to \mathbb{R}$, we define its conjugate function $F^* : \mathbb{R}^p \to \mathbb{R}$ to be the function*

$$F^*(\boldsymbol{\gamma}) = \sup_{\boldsymbol{\beta}} \langle \boldsymbol{\gamma}, \boldsymbol{\beta} \rangle - F(\boldsymbol{\beta}).$$

We will make the following minor technical assumption: in step 2) of Algorithm 3, we assume without loss of generality that the $\boldsymbol{\gamma}^\ell$ so computed satisfies the additional criteria:

1. it is an extreme point of the relevant feasible region,
2. and that if $\partial f_2(\boldsymbol{\beta}^\ell) \not\subseteq \partial f_1(\boldsymbol{\beta}^\ell)$, then $\boldsymbol{\gamma}^\ell$ is chosen such that $\boldsymbol{\gamma}^\ell \in \partial f_2(\boldsymbol{\beta}^\ell) \setminus \partial f_1(\boldsymbol{\beta}^\ell)$.

Solving (4.48) with these additional assumptions can nearly be solved in closed form by simply sorting the entries of $|\boldsymbol{\beta}|$, i.e., by finding $|\beta_{(1)}|, \ldots, |\beta_{(p)}|$.We must take some

[1] For the specific functions of interest, verifying local optimality of a candidate $\boldsymbol{\beta}^*$ can be performed in $O(p \min\{n, p\} + p \log p)$ operations; the first component relates to the computation of $\mathbf{X}'\mathbf{X}\boldsymbol{\beta}^*$, while the second captures the sorting of the entries of $\boldsymbol{\beta}^*$. See Appendix C.2.2 for details.

care to ensure that the second without loss of generality condition on $\boldsymbol{\gamma}$ is satisfied. This is straightforward but tedious; the details are shown in Appendix C.2.2.

Using this modification, the convergence properties of Algorithm 3 can be proven as follows:

Proof of Theorem 18. This is an application of [145, Thms. 3-5]. The only modification is in requiring that $\boldsymbol{\gamma}^\ell$ is chosen so that $\boldsymbol{\gamma}^\ell \in \partial f_2(\boldsymbol{\beta}^*) \setminus \partial f_1(\boldsymbol{\beta}^*)$ if $\boldsymbol{\beta}^\ell$ is not a local minimum of f—see [145, §3.3] for a motivation and justification for such a modification. Finally, the correspondence between $\boldsymbol{\gamma}^\ell \in \partial f_2(\boldsymbol{\beta}^\ell)$ and (4.48), and between $\boldsymbol{\beta}^{\ell+1} \in \partial f_1^*(\boldsymbol{\gamma}^\ell)$ and (4.49), is clear from an elementary argument applied to subdifferentials of variational formulations of functions. □

C.2.2 Algorithm 3, Step 2

Here we present the details of solving (4.48) in Algorithm 3 in a way that ensures that the associated without loss of generality claims hold. In doing so, we also implicitly study how to verify the conditions for local optimality (Theorem 20). Throughout, we use the sgn function defined as

$$\operatorname{sgn}(x) = \begin{cases} 1, & x > 0 \\ -1, & x < 0 \\ 0, & x = 0. \end{cases}$$

For fixed $\boldsymbol{\beta}$, the problem of interest is

$$\begin{aligned} \max_{\boldsymbol{\gamma}} \quad & \langle \boldsymbol{\beta}, \boldsymbol{\gamma} \rangle \\ \text{s. t.} \quad & \sum_i |\gamma_i| \leq \lambda k \\ & |\gamma_i| \leq \lambda \ \forall i. \end{aligned}$$

We wish to find a maximizer $\boldsymbol{\gamma}$ for which the following hold:

1. $\boldsymbol{\gamma}$ is an extreme point of the relevant feasible region,
2. and that if $\partial f_2(\boldsymbol{\beta}) \not\subseteq \partial f_1(\boldsymbol{\beta})$, then $\boldsymbol{\gamma}$ is such that $\boldsymbol{\gamma} \in \partial f_2(\boldsymbol{\beta}) \setminus \partial f_1(\boldsymbol{\beta})$.

As the problem on its own can be solved by sorting the entries of $\boldsymbol{\beta}$, the crux of the problem is ensuring that 2) holds.

Given the highly structured nature of f_1 and f_2 in our setup, it is simple, albeit tedious, to ensure that such a condition is satisfied. Let $I = \{i : |\beta_i| = |\beta_{(k)}|\}$. If $|I| = 1$, the optimal solution is unique, and there is nothing to show. Therefore, we will assume that $|I| \geq 2$. We will construct an optimal solution $\boldsymbol{\gamma}$ which satisfies the desired conditions. First observe that we necessarily must have that 1) $\gamma_i = \lambda \operatorname{sgn}(\beta_i)$ if $|\beta_i| > |\beta_{(k)}|$ and 2) $\gamma_i = 0$ if $|\beta_i| < |\beta_{(k)}|$. We now proceed to define the rest of the entries of $\boldsymbol{\gamma}$. We consider two cases:

1. First consider the case when $|\beta_{(k)}| > 0$. We claim that $\partial f_2(\boldsymbol{\beta}) \not\subseteq \partial f_1(\boldsymbol{\beta})$. To do so, we will inspect the ith entries of $\partial f_1(\boldsymbol{\beta})$ for $i \in I$; as such, let $P_i^j = \{\delta_i : \boldsymbol{\delta} \in \partial f_j(\boldsymbol{\beta})\}$ for $j \in \{1, 2\}$ and $i \in I$ (a projection). For each $i \in I$, we have using basic convex analysis that P_i^1 is a singelton: $P_i^1 = \{\langle \mathbf{X}_i, \mathbf{X}\boldsymbol{\beta} - \mathbf{y}\rangle + (\eta + \lambda) \operatorname{sgn}(\beta_i)\}$, where $\mathbf{X}_i$ is the ith column of $\mathbf{X}$. In contrast, because $|I| \geq 2$, the set P_i^2 is an interval with strictly positive length for each $i \in I$ (it is either $[-\lambda, 0]$ or $[0, \lambda]$, depending on whether $\beta_i < 0$ or $\beta_i > 0$, respectively). Therefore, $\partial f_2(\boldsymbol{\beta}) \not\subseteq \partial f_1(\boldsymbol{\beta})$, as claimed.

   Fix an arbitrary $j \in I$. Per the above argument, we must have that $\langle \mathbf{X}_j, \mathbf{X}\boldsymbol{\beta} - \mathbf{y}\rangle + (\eta + \lambda) \operatorname{sgn}(\beta_j) \neq 0$ or $\langle \mathbf{X}_j, \mathbf{X}\boldsymbol{\beta} - \mathbf{y}\rangle + (\eta + \lambda) \operatorname{sgn}(\beta_j) \neq \lambda \operatorname{sgn}(\beta_j)$. In the former case, set $\gamma_i = 0$, while in the latter case we define $\gamma_i = \lambda \operatorname{sgn}(\beta_i)$ (if both are true, either choice suffices). It is clear that it is possible to fill in the remaining entries of γ_i for $i \in I \setminus \{j\}$ in a straightforward manner so that $\boldsymbol{\gamma} \in \partial f_2(\boldsymbol{\beta})$. Further, by construction, $\boldsymbol{\gamma} \notin \partial f_1(\boldsymbol{\beta})$, as desired.

2. Now consider the case when $|\beta_{(k)}| = 0$. Using the preceding argument, we see that P_i^1 is the interval $[\langle \mathbf{X}_i, \mathbf{X}\boldsymbol{\beta} - \mathbf{y}\rangle - (\eta + \lambda), \langle \mathbf{X}_i, \mathbf{X}\boldsymbol{\beta} - \mathbf{y}\rangle + \eta + \lambda]$ for $i \in I$. In contrast, P_i^2 is the interval $[-\lambda, \lambda]$ for $i \in I$. If for all $i \in I$ one has that $P_i^2 \subseteq P_i^1$, then the choice of γ_i for $i \in I$ is obvious: any optimal extreme point $\boldsymbol{\gamma}$ of the problem will suffice. (Note here that it may or may not be that $\partial f_2(\boldsymbol{\beta}) \subseteq \partial f_1(\boldsymbol{\beta})$. This entirely depends on β_i for $i \notin I$.)

Therefore, we may assume that there exists some $j \in I$ so that $P_j^2 \not\subseteq P_j^1$. (It follows immediately that $\partial f_2(\boldsymbol{\beta}) \not\subseteq \partial f_1(\boldsymbol{\beta})$.) We must have that $\langle \mathbf{X}_j, \mathbf{X}\boldsymbol{\beta} - \mathbf{y} \rangle - (\eta + \lambda) > -\lambda$ or $\langle \mathbf{X}_j, \mathbf{X}\boldsymbol{\beta} - \mathbf{y} \rangle + (\eta + \lambda) < \lambda$. In the former case, set $\gamma_i = -\lambda$, while in the latter case we define $\gamma_i = \lambda$ (if both are true, either choice suffices). It is clear that it is possible to fill in the remaining entries of γ_i for $i \in I \setminus \{j\}$ in a straightforward manner so that $\boldsymbol{\gamma} \in \partial f_2(\boldsymbol{\beta})$. By construction, $\boldsymbol{\gamma} \notin \partial f_1(\boldsymbol{\beta})$, as desired.

In either case, we have that one can choose $\boldsymbol{\gamma} \in \partial f_2(\boldsymbol{\beta})$ so that 1) $\boldsymbol{\gamma}$ is an extreme point of the feasible region $\{\boldsymbol{\gamma} : \sum_i |\gamma_i| \leq \lambda k,\ |\gamma_i| \leq \lambda\ \forall i\}$ and that 2) $\boldsymbol{\gamma} \in \partial f_2(\boldsymbol{\beta}) \setminus \partial f_1(\boldsymbol{\beta})$ whenever $\partial f_2(\boldsymbol{\beta}) \not\subseteq \partial f_1(\boldsymbol{\beta})$. This concludes the analysis; thus, we have shown the validity (and computational feasibility) of the without loss of generality claim present in Algorithm 3. Indeed, per our analysis, Step 2 in Algorithm 3 can be solved in $O(p \min\{n, p\} + p \log p)$ operations (sorting of $\boldsymbol{\beta}$ in $O(p \log p)$ followed by $O(p)$ conditionals and gradient evaluation in $O(np)$). In reality, if we keep track of gradients in Step 3, there is no need to recompute gradients in Step 2, and therefore in practice Step 2 is of the same complexity of sorting a list of p numbers. (We assume that $\mathbf{X}'\mathbf{y}$ has been computed offline and store throughout for simplicity.)

C.2.3 Algorithm 4, Step 3

Here we show how to solve Step 3 in Algorithm 4, namely, solving the orthogonal design trimmed Lasso problem

$$\min_{\boldsymbol{\gamma}} \lambda T_k(\boldsymbol{\gamma}) + \frac{\sigma}{2} \|\boldsymbol{\beta} - \boldsymbol{\gamma}\|_2^2 - \langle \mathbf{q}, \boldsymbol{\gamma} \rangle, \tag{C.1}$$

where $\boldsymbol{\beta}$ and $\mathbf{q}$ are fixed. This is solvable in closed form. Let $\boldsymbol{\alpha} = \boldsymbol{\beta} + \mathbf{q}/\sigma$. First observe that we can rewrite (C.1), up to an irrelevant additive constant, as

$$\begin{aligned}
\text{(C.1)} \equiv \quad & \min_{\boldsymbol{\gamma}} \quad \lambda T_k(\boldsymbol{\gamma}) + \sigma \|\boldsymbol{\gamma} - \boldsymbol{\alpha}\|_2^2/2 \\
= \quad & \min_{\substack{\boldsymbol{\gamma}, \mathbf{z}: \\ \sum_i z_i = p-k \\ \mathbf{z} \in \{0,1\}^p}} \quad \lambda \langle \mathbf{z}, |\boldsymbol{\gamma}| \rangle + \sigma \|\boldsymbol{\gamma} - \boldsymbol{\alpha}\|_2^2/2 \\
= \quad & \min_{\substack{\boldsymbol{\gamma}, \mathbf{z}: \\ \sum_i z_i = p-k \\ \mathbf{z} \in \{0,1\}^p}} \quad \sum_i \left(\lambda z_i |\gamma| + \sigma (\gamma_i - \alpha_i)^2/2 \right).
\end{aligned}$$

The penultimate step follows via Lemma 1. Per this final representation, the solution becomes clear. In particular, let I be a set of k indices of $\boldsymbol{\alpha}$ corresponding to $\alpha_{(1)}$, $\alpha_{(2)}$, ..., $\alpha_{(k)}$. (If $|\alpha_{(k)}| = |\alpha_{(k+1)}|$, we break ties arbitrarily.) Then a solution $\boldsymbol{\gamma}^*$ to (C.1) is

$$\gamma_i^* = \begin{cases} \alpha_i, & i \in I \\ \text{soft}_{\lambda/\sigma}(\alpha_i), & i \notin I, \end{cases}$$

where $\text{soft}_{\lambda/\sigma}(\alpha_i) = \text{sgn}(\alpha_i)\,|\alpha_i - \lambda/\sigma|$.

C.2.4 Computational details

For completeness and reproducibility, we also include all computational details. For Figure 4-3, the following parameters were used to generate the test instance: $n = 100$, $p = 20$, SNR $= 10$, `julia` seed $= 1$, $\eta = 0.01$, $k = 2$. The example was generated from the following true model:

1. $\boldsymbol{\beta}_{\text{true}}$ is a vector with ten entries equal to 1 and all others equal to zero. (So $\|\boldsymbol{\beta}_{\text{true}}\|_0 = 10$.)
2. covariance matrix $\boldsymbol{\Sigma}$ is generated with $\Sigma_{ij} = .8^{|i-j|}$.
3. $\mathbf{X} \sim N(\mathbf{0}, \boldsymbol{\Sigma})$.
4. $\epsilon_i \overset{\text{i.i.d.}}{\sim} N(0, \boldsymbol{\beta}_0' \boldsymbol{\Sigma} \boldsymbol{\beta}_0 / \text{SNR})$
5. $\mathbf{y}$ is then defined as $\mathbf{X}\boldsymbol{\beta}_0 + \boldsymbol{\epsilon}$

The 100 examples generated for Figure 4-4 were using the following parameters: $n = 100$, $p = 20$, SNR $= 10$, `julia` seed $\in \{1, \ldots, 100\}$, $\eta = 0.01$, $k = 2$, bigM $= 20$. MIO using Gurobi solver. Max iterations: alternating minimization—1000; ADMM (inner)—2000; ADMM (outer)—10000. ADMM parameters: $\sigma = 1$, $\tau = 0.9$. The examples themselves had the same structure as the previous example. The optimal gaps shown are relative to the objective in (4.47). The averages are computed as geometric means (relative to optimal 100%) across the 100 instances, and then displayed relative to the optimal 100%.

Appendix D

Supplemental Code

This appendix contains sample code for several of the algorithms described herein. At the time of writing, this code is also available at

github.com/copenhaver/⋆,

where ⋆ ∈ {factoranalysis, trimmedlasso}. For archival purposes, that code is included here as well. The code is as follows:

1. For Chapter 2, there are two files: `funcs.R` contains necessary functions, while `demo.R` carries out a basic demonstration of the code functionality.

2. For Chapter 4, there are three files: `funcs.jl` contains necessary functions, `demo.jl` carries out a basic demonstration of the code functionality, and `example-creator.jl` generates example instances for testing.

D.1 Factor Analysis

`funcs.R`

```
# An R implementation of Algorithm 1 from Chapter 2

FA <- function(S, factors, maxiter.inner = 1000, maxiter.outer = 1000, rho = .01,
↪  tol.inner = 1e-9, tol.outer = 1e-5){
  # Input: a covariance matrix S and the desired number of factors to perform
  ↪  factor analysis.
  # Other parameters: optimal algorithmic parameters with defaults
```

```
# Output: decomposition S = T + P + N, where T is positive-semidefinite (PSD)
↪  with rank <= factors,
#         P is diagonal and PSD, and N is PSD (N is the "noise" component)

### algorithmic parameters:
# maxiter.* : maximum number of * iterations
# rho : scaling parameter in ADMM
# tol.* : * optimality tolerance

# verify that S is indeed a matrix

if (!(is.matrix(S))) {
  stop(simpleError("Inputted covariance matrix is not in matrix format. Convert
  ↪  using as.matrix( )."));
}

# verify that S is square

if (dim(S)[1] != dim(S)[2]) {
  stop(simpleError("Inputted covariance matrix is not square."));
}

# verify (approximate symmetry of S)

if (norm(S-t(S), type="F") > 1e-10) {
  stop(simpleError("Inputted covariance matrix is not symmetric."));
}

# problem parameters
p = dim(S)[1];

# verify that number of factors is valid (between 0 and p, inclusive)

if (!(as.integer(factors) == factors | factors < 0 | factors > p)) {
  stop(simpleError("Number of factors is not valid."));
}

# key variables
phi = matrix(0, nrow = p, ncol = 1);
W = matrix(1, nrow = p, ncol = p);

# useful constants
zp = matrix(0, nrow = p, ncol = 1); #matrix of zeros
dS = diag(S); # diagonal of S
weig = diag(c(rep(0,factors),rep(1,p-factors))); #weighting used on
↪  eigenvalues

# "inner" phi, Lambda, and Nu (for inner iterations)
lphi = rep(0, p);
lLambda = matrix(0, nrow = p, ncol = p);
lNu = matrix(0, nrow = p, ncol = p);

bobj = Inf; # best objective
```

```
oerr = Inf; # "outer" error for problem wrt W
opobj = Inf; # previous (outer) objective
ocobj = 0; # current (outer) objective

ocount = 0; # outer counter
while ( (oerr > tol.outer) & (ocount < maxiter.outer) ) {
  ### solve wrt phi

  ierr = Inf; # "inner" error for problem wrt Phi (which requires an ADMM)
  pobj = Inf; # previous objective
  cobj = 0; # current objective

  dW = diag(W);
  dWs = dW/rho;

  count = 0;
  while ( (ierr > tol.inner) & (count < maxiter.inner) ) {
    ## first: update lphi -- closed form

    lphi = pmax(zp,dS - diag(lLambda) + dWs - diag(lNu)/rho);

    ## now: update lLambda -- eigendecomposition (projection on PSD)

    e = eigen(S - diag(lphi) - lNu/rho,T); # "T" for symmetric argument

    lLambda = e$vectors %*% diag(as.vector(pmax(zp,as.vector(e$values)))) %*%
    ↪  t(e$vectors) # would normally need inverse, but can just do tranpose
    ↪  since unitary.

    ## then: update lNu -- closed form (gradient update)

    lNu = lNu + rho * (lLambda - S + diag(as.vector(lphi)));

    ## finally: update ierr (error) = norm of (lLambda - (S-lph))

    pobj = cobj;
    cobj = sum(dW*lphi);
    relimp = abs(cobj-pobj)/(abs(pobj)+.01)*100;
    ierr = max( norm( lLambda - S + diag(as.vector(lphi)) , type = "F") ,
    ↪  relimp );
    # print(c(ierr,count));
    count = count + 1;
  }

  # update phi to be lphi

  phi = lphi;

  ### solve wrt W --- requires eigendecomp of S - diag(phi)

  e = eigen(S - diag(as.vector(phi)), T );
  W = e$vectors %*% weig %*% t(e$vectors); # would normally need inverse, but
  ↪  can just do tranpose since hermitian (real unitary)
```

```
    bobj = min(bobj, sum(W*(S-diag(as.vector(phi)))) ); # second term is the
    ↪  inner product of W and S-Phi

    opobj = ocobj;
    ocobj = sum(W*(S-diag(as.vector(phi))));

    oerr = abs(ocobj-opobj)/(abs(opobj)+.01)*100;
    ocount = ocount + 1;

    print(c(ocount,bobj));
  }

  e = eigen(S - diag(as.vector(phi)), T );
  Theta = e$vectors %*% diag(c(e$values[1:factors],rep(0,p-factors))) %*%
  ↪  t(e$vectors); # would normally need inverse, but can just do tranpose since
  ↪  hermitian (real unitary)

  return(list(Theta=Theta,Phi=as.vector(phi))); # return matrix Theta (T) and Phi
  ↪  (P)
}
```

demo.R

```
# A demo of R implementation of Algorithm 1 from Chapter 2

######################

# generate random Theta and Phi (class A_1 in BCM17)

p = 20
r = 2

# create matrices THETA and PHI

L = matrix(rnorm(p*r), ncol = r)
THETA = L %*% t(L)
PHI = runif(p)
PHI = (sum(diag(THETA))/sum(PHI))*PHI # normalized so that equal proportion of
↪  common and individual variances

# covariance matrix S is sum of THETA and PHI

S = THETA + diag(PHI)

####  Now perform factor analysis

res = FA(S,2) # FA function from code.R

# See if you recover true Theta and true Phi

norm(THETA - res$Theta, type="F") # in Frobenius norm
norm(PHI - as.matrix(res$Phi), type="F")
```

D.2 Trimmed Lasso

funcs.jl

```
# A julia implementation of various algorithms from Chapter 4

##########################
## Import packages    ##
##########################

using JuMP

#################################
## Auxiliary functions          ##
#################################

function aux_lassobeta(n::Int,p::Int,k::Int,mu::Float64,lambda::Float64,
XX::Array{Float64,2},loc_b_c::Array{Float64,1},
grad_rest::Array{Float64,1},max_iters=10000,tol=1e-3)
  # solve subproblem wrt beta, with (outer) beta as starting point

  MAX_ITERS = max_iters;
  TOL = tol;

  lbc = copy(loc_b_c);
  lbp = loc_b_c - ones(p);
  tcur = 1./norm(XX);
  iterl = 0;

  while (iterl < MAX_ITERS) && ( norm(lbc - lbp) > TOL )

    lbp = lbc;

    gg = lbc - tcur*(XX*lbc + grad_rest);

    lbc = sign(gg).*max(abs(gg)-tcur*(mu+lambda)*ones(p),zeros(p));

    #tcur = TAU*tcur;

    iterl = iterl + 1;

  end

  return(lbc);
end

function aux_admmwrtbeta(n::Int,p::Int,k::Int,mu::Float64,lambda::Float64,
XX::Array{Float64,2},loc_b_c::Array{Float64,1},
grad_rest::Array{Float64,1},sigma,max_iters=10000,tol=1e-3)
  # solve subproblem wrt beta, with (outer) beta as starting point

  MAX_ITERS = max_iters;
  TOL = tol;
  SIGMA = sigma;
```

```
  lbc = copy(loc_b_c);
  lbp = loc_b_c - ones(p);
  tcur = 1./norm(XX+SIGMA*eye(p));
  iterl = 0;

  while (iterl < MAX_ITERS) && ( norm(lbc - lbp) > TOL )

    lbp = lbc;

    gg = lbc - tcur*((XX+SIGMA*eye(p))*lbc + grad_rest);

    lbc = sign(gg).*max(abs(gg)-tcur*mu*ones(p),zeros(p));

    #tcur = TAU*tcur;

    iterl = iterl + 1;

  end

  return(lbc);
end

##################################
## Exact methods (MIO-based)    ##
##################################

### SOS-1 formulation

function tl_exact(p,k,y,X,mu,lambda,solver)
  #####
  # Inputs (required arguments):
  #    data matrix `X` and response `y`
  #    `p` is the number of columns of X (i.e., the number of features).
  #    `k` is the sparsity parameter on the trimmed Lasso
  #    `mu` is the multipler on the usual Lasso penalty: mu*sum_i |beta_i|
  #    `lambda` is the multipler on the trimmed Lasso penalty: lambda*sum_{i>k}
  ↪  |beta_{(i)}|
  #    `solver` is the desired mixed integer optimization solver. This should
  ↪  have SOS-1 capabilities (will return error otherwise).
  #    `bigM` is an upper bound on the largest magnitude entry of beta. if the
  ↪  constraint |beta_i|<= bigM is binding at optimality, an error will be
  ↪  thrown, as this could mean that the value of `bigM` given may have been too
  ↪  small.
  # Output: estimator beta that is optimal to the problem
  #         minimize_beta  0.5*norm(y-X*beta)^2 + mu*sum_i |beta_i| +
  ↪  lambda*T_k(beta)
  # Method: exact approach using SOS-1 constraints and mixed integer optimization
  ↪  (e.g. using commercial solver Gurobi)
  #####

  if ( p != size(X)[2] )
    println("Specified p is not equal to row dimension of X. Halting
    ↪  execution.");
```

```
    return;
  end

  m = Model(solver = solver);

  @variable(m, gamma[1:p] >= 0);
  @variable(m, beta[1:p] );
  @variable(m, z[1:p], Bin);
  @variable(m, pi[1:p] >= 0);

  @constraint(m, gamma[i=1:p] .>= beta[i] );
  @constraint(m, gamma[i=1:p] .>= -beta[i] );
  @constraint(m, sum(z) == p - k );
  @constraint(m, pi .<= gamma );

  # add SOS-1 constraints to the model; if the solver supplied does not support
  ↪  SOS-1 constraints, JuMP will throw an error; we do not catch that here so
  ↪  it will raise to the user
  for i=1:p
    addSOS1(m, [z[i],pi[i]]);
  end

  # add quadratic objective; again, if the solver cannot handle such an
  ↪  objective, an error will be raised
  @objective(m, Min, dot(beta,.5*X'*X*beta) - dot(y,X*beta)+dot(y,y)/2
   + (mu+lambda)*sum(gamma)-lambda*sum(pi) )

  solve(m);

  return getvalue(beta);

end

### big-M formulation

function tl_exact_bigM(p,k,y,X,mu,lambda,solver,bigM,throwbinding=true)
  #####
  # Inputs (required arguments):
  #    data matrix `X` and response `y`
  #    `p` is the number of columns of X (i.e., the number of features).
  #    `k` is the sparsity parameter on the trimmed Lasso
  #    `mu` is the multipler on the usual Lasso penalty: mu*sum_i |beta_i|
  #    `lambda` is the multipler on the trimmed Lasso penalty: lambda*sum_{i>k}
  ↪  |beta_{(i)}|
  #    `solver` is the desired mixed integer optimization solver. This should
  ↪  have SOS-1 capabilities (will return error otherwise).
  #    `bigM` is an upper bound on the largest magnitude entry of beta. if the
  ↪  constraint |beta_i|<= bigM is binding at optimality, an error will be
  ↪  thrown, as this could mean that the value of `bigM` given may have been too
  ↪  small.
  # Optional arguments:
  #    `throwbinding`---default value of `true`. To disable the built-in error
  ↪  functionality that occurs when the `bigM` value is potentially too small,
  ↪  set `throwbinding=false`.
```

```
# Output: estimator beta that is optimal to the problem
#         minimize_beta  0.5*norm(y-X*beta)^2 + mu*sum_i |beta_i| +
↪  lambda*T_k(beta)
# Method: exact approach using bigM constraints and mixed integer optimization
↪  (e.g. using commercial solver Gurobi)
# Because bigM formulations are more easily used by solvers, this approach is
↪  much easier to use if you have a specific preference on which solver you
↪  use. However, note that the performance of this approach, much like the
↪  performance of solvers for any big-M-based optimization problem, is highly
↪  dependent upon tuning of the value of M. Therefore, if you do not have a
↪  good sense of what value to set for M and you have access to a solver that
↪  handles SOS-1 constraints, we recommend using the SOS-1-based approach
↪  (given in function tl_exact )
#####

if ( p != size(X)[2] )
  println("Specified p is not equal to row dimension of X. Halting
  ↪  execution.");
  return;
end

if !( bigM >= 0 && bigM < Inf )
  println("Invalid big-M value supplied. Halting execution.");
end

m = Model(solver = solver);

@variable(m, gamma[1:p] >= 0);
@variable(m, a[1:p] >= 0);
@variable(m, beta[1:p] );
@variable(m, z[1:p], Bin);

@constraint(m, gamma[i=1:p] .>= beta[i] );
@constraint(m, gamma[i=1:p] .>= -beta[i] );
@constraint(m, a[i=1:p] .>= bigM*z[i] + gamma[i] - bigM );
@constraint(m, beta[1:p] .<= bigM );
@constraint(m, beta[1:p] .>= -bigM );
@constraint(m, sum(z[i] for i=1:p) == p - k );

@objective(m, Min, dot(beta,.5*X'*X*beta) - dot(y,X*beta)+dot(y,y)/2
 + sum{mu*gamma[i]+lambda*a[i], i=1:p})

solve(m);

binding = false;

for i=1:p
  if abs(getvalue(beta[i])) >= bigM - 1e-3
    binding = true
  end
end

if (binding && throwbinding)
```

```
    println("\t\tWarning: big-M constraint is binding  -- you should increase
    ↪  big-M and resolve. Otherwise, re-use same big-M and set optional argument
    ↪  `throwbinding=false`.");;
  else
    return getvalue(beta);
  end
end

#################################
## Heuristic (convex) methods   ##
#################################

### alternating minimization

function tl_apx_altmin(p,k,y,X,mu,lambda,lassosolver=aux_lassobeta,
max_iter=10000,rel_tol=1e-6,print_every=200)
  #####
  # Inputs:
  #    data matrix `X` and response `y`
  #    `p` is the number of columns of X (i.e., the number of features).
  #    `mu` is the multipler on the usual Lasso penalty: mu*sum_i |beta_i|
  #    `lambda` is the multipler on the trimmed Lasso penalty: lambda*sum_{i>k}
  ↪  |beta_{(i)}|
  # Optional arguments:
  #    `lassosolver`---default value of `aux_lassobeta`, which is a simple Lasso
  ↪  problem solver whose implementation is included above as an auxiliary
  ↪  function. If you would like to solve the Lasso subproblems using your own
  ↪  Lasso solver, you should change this argument. Note that the `lassosolver`
  ↪  values expect as function which has the following characteristics:
  ##       Input arguments will be as follows:
  ##               `n` - dimension of row size of `X`;
  ##               `p` - as in outer problem;
  ##               `k` - as in outer problem;
  ##               `mu` - as in outer problem;
  ##               `lambda` - as in outer problem;
  ##               `XX` - value of transpose(X)*X (can be precomputed and stored
  ↪  offline);
  ##               `loc_b_c` - initial value of beta from which to initial the
  ↪  algorithm;
  ##               `grad_rest` - the remaining part of the gradient term (-X'*y-
  ↪  gamma).
  ##       Output: solution beta to the Lasso problem
  ##              minimize_beta norm(y-X*beta)^2 +(mu+lambda)*sum_i |beta_i| +
  ↪  dot(beta,gamma) (gamma is the solution from the alternating problem, as
  ↪  supplied in the additional gradient information).
  #    `max_iter`---default value of 10000. Maximum number of alternating
  ↪  iterations for the algorithm.
  #    `rel_tol`---default value of 1e-6. The algorithm concludes when the
  ↪  relative improvement
  ↪  (current_objective-previous_objective)/(previous_objective + .01) is less
  ↪  than `rel_tol`. The additional `0.01` in the denominator ensures no
  ↪  numerical issues.
  #    `print_every`---default value of 200. Controls amount of amount output.
  ↪  Set `print_every=Inf` to suppress output.
```

```
# Output: estimator beta that is a *possible* solution for the problem
#         minimize_beta  0.5*norm(y-X*beta)^2 + mu*sum_i |beta_i| +
↪  lambda*T_k(beta)
# Method: alternating minimization approach which finds heuristic solutions to
↪  the trimmed Lasso problem. See details in Algorithm 3 (Chapter 4)
#####

AM_ITER = max_iter;
REL_TOL = rel_tol;
PRINT_EVERY = print_every; # AM will print output on every (PRINT_EVERY)th
↪  iteration

beta = randn(p);
gamma = zeros(p);

XpX = X'*X; # can separate computation if desired

prev_norm = 0;
prev_obj = 0;

for I=0:AM_ITER

  # solve wrt gamma (by sorting beta)

  II = zeros(p);
  sto = 0; # number set to "one" (really += lambda)

  bk = sort(abs(beta))[p-k+1];

  for i=1:p
    if (abs(beta[i]) > bk)
      gamma[i] = lambda*sign(beta[i]);
      sto = sto + 1;
    else
      if (abs(beta[i]) < bk)
        gamma[i] = 0;
      else
        II[i] = 1;
      end
    end
  end

  if sum(II) == 0
    println("ERROR!");
  else
    if sum(II) == 1
      gamma[indmax(II)] = lambda*sign(beta[indmax(II)]);
      sto = sto + 1;
    else # |II| >= 2, so need to use special cases as detailed in Appendix C
      if bk > 0
        j = indmax(II); # arbitrary one from II ---> should probably choose
        ↪  randomly amongst them
        if dot(X[:,j],X*beta-y) + (mu+lambda)*sign(beta[j]) != 0
          gamma[j] = 0;
```

```
    else
      gamma[j] = lambda*sign(beta[j]);
      sto = sto + 1;
    end
    # assign rest of gamma
    for i=randperm(p)
      if (sto < k) && (II[i] > 0.5)
        gamma[i] = sign(randn())*lambda;
        sto = sto + 1;
      end
    end

  else # so bk == 0
    # need to check interval containment over indices in II
    notcontained = false;
    corrindex = -1;
    corrdot = Inf;
    for i=randperm(p)
      if II[i] > 0.5 # i.e. == 1
        dp = dot(X[:,i],X*beta - y);
        if (abs(dp) > mu)
          notcontained = true;
          corrindex = i;
          corrdot = dp;
          break;
        end
      end
    end

    if notcontained
      j = corrindex;
      if corrdot > mu
        gamma[j] = -lambda;
        sto = sto + 1;
      else
        gamma[j] = lambda;
        sto = sto + 1;
      end
      # fill in rest of gamma
      for i=randperm(p)
        if (sto < k) && (II[i] > 0.5) && (i != j)
          gamma[i] = sign(randn())*lambda;
          sto = sto + 1;
        end
      end
    else # any extreme point will do
      for i=randperm(p)
        if (sto < k) && (II[i] > 0.5)
          gamma[i] = sign(randn())*lambda;
          sto = sto + 1;
        end
      end
    end
```

```
        end
      end
    end

    # ensure that sto == k

    if sto != k
      println("ERROR. EXTREME POINT NOT FOUND. ABORTING.");
      II(1)
    end

    # solve wrt beta

    beta = lassosolver(n,p,k,mu,lambda,XpX,beta,-X'*y- gamma);

    # perform updates as necessary

    cur_obj = .5*norm(y-X*beta)^2 + mu*norm(beta,1)
    ↪  +lambda*sum(sort(abs(beta))[1:p-k]);

    if abs(cur_obj-prev_obj)/(prev_obj+.01) < REL_TOL # .01 in denominator is for
    ↪  numerical tolerance with zero
      println(I);
      break; # end AM loops
    end

    prev_obj = cur_obj;

  end

  return copy(beta);

end

### ADMM

function tl_apx_admm(p,k,y,X,mu,lambda,
max_iter=2000,rel_tol=1e-6,sigma=1.,print_every=200)
  #####
  # Inputs:
  #    data matrix `X` and response `y`
  #    `p` is the number of columns of X (i.e., the number of features).
  #    `mu` is the multipler on the usual Lasso penalty: mu*sum_i |beta_i|
  #    `lambda` is the multipler on the trimmed Lasso penalty: lambda*sum_{i>k}
  ↪  |beta_{(i)}|
  # Optional arguments:
  #    `max_iter`---default value of 2000. Maximum number of (outer) ADMM
  ↪  iterations for the algorithm.
  #    `rel_tol`---default value of 1e-6. The algorithm concludes when the
  ↪  relative improvement
  ↪  (current_objective-previous_objective)/(previous_objective + .01) is less
  ↪  than `rel_tol`. The additional `0.01` in the denominator ensures no
  ↪  numerical issues.
```

```
#    `sigma`---default value of 1.0. This is the augmented Lagranian penalty as
↪  shown in Algorithm 4.
#    `print_every`---default value of 200. Controls amount of amount output.
↪  Set `print_every=Inf` to suppress output.
# Output: estimator beta that is a *possible* solution for the problem
#         minimize_beta  0.5*norm(y-X*beta)^2 + mu*sum_i |beta_i| +
↪  lambda*T_k(beta)
# Method: ADMM approach which finds heuristic solutions to the trimmed Lasso
↪  problem. See details in Algorithm 4 (Chapter 4)
#####

ADMM_ITER = max_iter;
REL_TOL = rel_tol;
# TAU = tau; ---> Could add the scaling parameter tau, but we will neglect to
↪  include that in our implementation
SIGMA = sigma;
PRINT_EVERY = print_every; # AM will print output on every (PRINT_EVERY)th
↪  iteration

XpX = X'*X; # can separate computation if desired

# ADMM vars
beta = zeros(p);
gamma = zeros(p);
q = zeros(p);

# <solve ADMM>

prev_norm = 0;
prev_obj = 0;

for I=0:ADMM_ITER

  beta = aux_admmwrtbeta(n,p,k,mu,lambda,XpX,beta,q-X'*y- SIGMA*gamma,SIGMA);;

  ### solve wrt gamma

  aux_sb = min(SIGMA/2*(beta.^2) + q.*beta+(1/2/SIGMA)*(q.^2) ,
  ↪  (lambda^2)/(2*SIGMA)*ones(p) +
  ↪  lambda*abs(beta+q/SIGMA+lambda/SIGMA*ones(p)),
    (lambda^2)/(2*SIGMA)*ones(p) +
    ↪  lambda*abs(beta+q/SIGMA-lambda/SIGMA*ones(p)));
  sb = sort([(aux_sb[i],i) for i=1:p]);
  zz = zeros(p);
  for i=1:(p-k)
    zz[sb[i][2]] = 1;
  end

  for i=1:p
    if zz[i] == 0
      gamma[i] = copy(beta[i]) + copy(q[i])/SIGMA;
    else # zz[i] = 1
      aar = [(SIGMA/2*(beta[i]^2) + q[i]*beta[i]+(1/2/SIGMA)*(q[i]^2) ,    0 ),
```

```
            ((lambda^2)/(2*SIGMA) + lambda*abs(beta[i]+q[i]/SIGMA+lambda/SIGMA),
            ↪  beta[i] + q[i]/SIGMA + lambda/SIGMA),
            ((lambda^2)/(2*SIGMA) + lambda*abs(beta[i]+q[i]/SIGMA-lambda/SIGMA),
            ↪  beta[i] + q[i]/SIGMA - lambda/SIGMA)];
        gamma[i] = sort(aar)[1][2];
      end
    end

    q = copy(q) + SIGMA*(beta-gamma);

    cur_norm = norm(beta-gamma);
    cur_obj = .5*norm(y-X*beta)^2 + mu*norm(beta,1)
    ↪  +lambda*sum(sort(abs(beta))[1:p-k]);

    if abs(cur_norm-prev_norm)/(prev_norm+.01) +
    ↪  abs(cur_obj-prev_obj)/(prev_obj+.01) < REL_TOL # .01 in denominator is
    ↪  for numerical tolerance with zero
      break; # end ADMM loops
    end

    prev_norm = cur_norm;
    prev_obj = cur_obj;

  end

  # </ end ADMM>

  return copy(gamma);

end

### convex envelope

function tl_apx_envelope(p,k,y,X,mu,lambda,solver)
  #####
  # Inputs:
  #    data matrix `X` and response `y`
  #    `p` is the number of columns of X (i.e., the number of features).
  #    `mu` is the multipler on the usual Lasso penalty: mu*sum_i |beta_i|
  #    `lambda` is the multipler on the trimmed Lasso penalty: lambda*sum_{i>k}
  ↪  |beta_{(i)}|
  #    `solver` is the desired linear optimization solver.
  # Optional arguments: none
  # Output: estimator beta that is a *possible* solution for the problem
  #         minimize_beta  0.5*norm(y-X*beta)^2 + mu*sum_i |beta_i| +
  ↪  lambda*T_k(beta)
  # beta is found by solving (to optimality) the following linear optimization
  ↪  problem:
  #         minimize_{e,beta}    0.5*norm(y-X*beta)^2 + mu*sum_i |beta_i| +
  ↪  beta*e
  #         subject to        e >= 0;
  #                           e >= sum_i |beta_i| - k;
  # As discussed in Chapter 4, this is the convex relaxation of the first problem
  ↪  when using convex envelopes.
```

```
  # Method: convexification approach which finds heuristic solutions to the
  ↪  trimmed Lasso problem. See details in Chapter 4.
  #####

  m = Model(solver = solver);

  @defVar(m, tau >= 0);
  @defVar(m, gamma[1:p] >= 0);
  @defVar(m, beta[1:p] );
  @defVar(m, envelope >= 0);

  @addConstraint(m, gamma[i=1:p] .>= beta[i] );
  @addConstraint(m, gamma[i=1:p] .>= -beta[i] );
  @addConstraint(m, envelope >= sum{lambda*gamma[i], i=1:p} - lambda*k); #convex
  ↪  envelope!
  #@addConstraint(m, norm2{y[i] - sum{X[i,j]*beta[j], j=1:p} , i=1:n} <= tau);
  @addConstraint(m, dot(beta,.5*X'*X*beta) - dot(y,X*beta)+dot(y,y)/2 <= tau);

  @setObjective(m, Min, tau + sum{mu*gamma[i], i=1:p} + envelope)

  solve(m);

  return getvalue(beta);

end
```

example-creator.jl

```
# Instance creator for use in demo.jl.

############################
## Import packages      ##
############################

using Distributions

function instance_creator(n,p,k,SNR,egclass,seed=1)

  srand(seed)

  SS = eye(p,p);
  beta0 = zeros(p);

  if egclass == 1
    rho = 0.8;
    ir = round(p/k);
    for i=1:p
      if i%ir == 1 # then beta0[i] = 1
        beta0[i] = 1;
      end
      for j = 1:p
        SS[i,j] = rho^abs(i-j);
      end
```

```
    end

  end
  if egclass == 2
    for i=1:5
      beta0[i] = 1;
    end
  end
  if egclass == 3
    for i=1:10
      beta0[i] = 1/2 + 10/9*.95*(i-1);
    end
  end
  if egclass == 4
    for i=1:6
      beta0[i] = -14 + 4*i;
    end

  end
  if egclass == 5
    for i=1:6
      beta0[i] = 1/2 + 10/9*.95*(i-1)^5;
    end
    beta0 = beta0/norm(beta0);
  end

  ### for all, define y = Xb+eps

  sig = sqrt(beta0'*SS*beta0/SNR)[1,1];
  eps = rand(Normal(0,sig),n);
  X = rand(MvNormal(SS),n)';

  # normalize columns of X to have ell2 norm of 1

  for i=1:p
    X[:,i] = X[:,i]/norm(X[:,i]);
  end

  y = X*beta0 + eps;

  return y, X, beta0;
end
```

demo.jl

```
# A demo of julia implementation of algorithms contained in code.jl

##############################
## Import packages and code ##
##############################

include("code.jl");
include("example-creator.jl");
```

```
########################
## Example Parameters ##
########################

n = 100;
p = 20;
k = 10;
SNR = 10.;
seed = 1;
egclass = 1;
mu = .01;
lambda = .01;
EPS = 1e-3;

### if you have the Gurobi solver use the following:

using Gurobi
SOLVER = GurobiSolver(OutputFlag=1); # possible options of interest:
↪  OutputFlag=1,TimeLimit=100000,Heuristics=.05

#### otherwise, use the free open-source solver Couenne (uncomment following two
↪  lines):

# using CoinOptServices
# SOLVER = OsilBonminSolver();

###################################
## Create example               ##
###################################

# set seed for reproducibility

srand(1);

y, X, beta0 = instance_creator(n,p,k,SNR,egclass);

#####################################
## Solve exact and heuristic models ##
#####################################

beta_hat_exact = tl_exact(p,k,y,X,mu,lambda,SOLVER);

# if solver you are using cannot handle SOS-1 constraints, you may need to use
↪  the big-M formulation: tl_exact_bigM

beta_hat_altmin = tl_apx_altmin(p,k,y,X,mu,lambda);

beta_hat_admm = tl_apx_admm(p,k,y,X,mu,lambda);

beta_hat_envelope = tl_apx_envelope(p,k,y,X,mu,lambda,SOLVER);
```

References

[1] T. Achterberg, "SCIP: solving constraint integer programs," *Mathematical Programming Computation*, vol. 1, no. 1, pp. 1–41, 2009.

[2] F. Al-Khayyal and J. Falk, "Jointly constrained biconvex programming," *Mathematics of Operations Research*, vol. 8, pp. 273–286, 1983.

[3] L. T. H. An, "Analyse numérique des algorithmes de l'optimisation DC. Approches locale et globale. Codes et simulations numériques en grande dimension. Applications," Ph.D. dissertation, Université de Rouen, 1994.

[4] L. T. H. An and P. D. Tao, "The DC (difference of convex functions) programming and DCA revisited with DC models of real world nonconvex optimization problems," *Annals of Operations Research*, vol. 133, pp. 23–46, 2005.

[5] E. Andersen and K. Andersen, *High Performance Optimization*. Springer, 2000, ch. The Mosek Interior Point Optimizer for Linear Programming: An Implementation of the Homogeneous Algorithm.

[6] T. Anderson, *An Introduction to Multivariate Statistical Analysis*, 3rd ed. Wiley, New York, 2003.

[7] R. Andreani, L. Secchin, and P. Silva, "Convergence properties of a second order augmented Lagrangian method for mathematical programs with complementarity constraints," 2017, preprint. [Online]. Available: http://www.optimization-online.org/DB_HTML/2017/04/5948.html

[8] K. Anstreicher and S. Burer, "Computable representations for convex hulls of low-dimensional quadratic forms," *Mathematical Programming, Series B*, vol. 124, pp. 33–43, 2010.

[9] J. Bai and K. Li, "Statistical analysis of factor models of high dimension," *The Annals of Statistics*, vol. 40, no. 1, pp. 436–465, 2012.

[10] J. Bai and S. Ng, "Large dimensional factor analysis," *Foundations and Trends in Econometrics*, vol. 3(2), pp. 89–163, 2008.

[11] L. Bai, J. Mitchell, and J.-S. Pang, "On conic QPCCs, conic QCQPs and completely positive programs," *Mathematical Programming, Series A*, vol. 159, no. 1-2, pp. 109–136, 2016.

[12] A. Bandeira, E. Dobriban, D. Mixon, and W. Sawin, "Certifying the Restricted Isometry Property is hard," *IEEE Transactions in Information Theory*, vol. 59, pp. 3448–3450, 2013.

[13] X. Bao, N. V. Sahinidis, and M. Tawarmalani, "Multiterm polyhedral relaxations for nonconvex, quadratically constrained quadratic programs," *Optimization Methods and Software*, vol. 24, pp. 485–504, 2009.

[14] D. Bartholomew, M. Knott, and I. Moustaki, *Latent variable models and Factor Analysis: A unified approach.* Wiley, 2011.

[15] A. Barvinok, "Problems of distance geometry and convex properties of quadratic maps," *Discrete and Computational Geometry*, vol. 12, pp. 189–202, 1995.

[16] H. Bauschke and P. Combettes, *Convex analysis and monotone operator theory in Hilbert spaces.* Springer, 2011.

[17] A. Beck and A. Ben-Tal, "Duality in robust optimization: primal worst equals dual best," *Operations Research Letters*, vol. 37, no. 1, pp. 1–6, 2009.

[18] A. Ben-Tal, E. Hazan, T. Koren, and S. Mannor, "Oracle-based robust optimization via online learning," *Operations Research*, vol. 63, no. 3, pp. 628–638, 2015.

[19] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski, *Robust Optimization.* Princeton University Press, 2009.

[20] P. Bentler and J. Woodward, "Inequalities among lower-bounds to reliability: With applications to test construction and factor analysis," *Psychometrika*, vol. 45, pp. 249–267, 1980.

[21] C. Bernaards and R. Jennrich, "Gradient projection algorithms and software for arbitrary rotation criteria in factor analysis," *Educational and Psychological Measurement*, vol. 65, pp. 676–696, 2005.

[22] D. P. Bertsekas, "Multiplier methods: a survey," *Automatica*, vol. 12, no. 2, pp. 133–145, 1976.

[23] ——, *Nonlinear programming*, 2nd ed. Athena Scientific, 1999.

[24] ——, *Constrained optimization and Lagrange multiplier methods.* Academic Press, 2014.

[25] D. Bertsimas, D. Brown, and C. Caramanis, "Theory and applications of robust optimization," *SIAM Review*, vol. 53, no. 3, pp. 464–501, 2011.

[26] D. Bertsimas and M. S. Copenhaver, "Characterization of the equivalence of robustification and regularization in linear and matrix regression," *European Journal of Operational Research*, 2017.

[27] D. Bertsimas, M. S. Copenhaver, and R. Mazumder, "Certifiably optimal low rank factor analysis," *Journal of Machine Learning Research*, vol. 18, no. 29, pp. 1–53, 2017.

[28] ——, "The Trimmed Lasso: Sparsity and robustness," 2017, preprint. [Online]. Available: http://www.optimization-online.org/DB_HTML/2017/08/6167.html

[29] D. Bertsimas, V. Gupta, and N. Kallus, "Data-driven robust optimization," *Mathematical Programming, Series A*, vol. 167, no. 2, pp. 235–292, 2018.

[30] D. Bertsimas, A. King, and R. Mazumder, "Best subset selection via a modern optimization lens," *The Annals of Statistics*, vol. 44, no. 2, pp. 813–852, 2016.

[31] D. Bertsimas and R. Mazumder, "Least quantile regression via modern optimization," *The Annals of Statistics*, vol. 42, no. 6, pp. 2494–2525, 2014.

[32] R. Bhatia, *Perturbation bounds for matrix eigenvalues*. SIAM, 2007.

[33] P. J. Bickel, Y. Ritov, and A. B. Tsybakov, "Simultaneous analysis of Lasso and Dantzig selector," *The Annals of Statistics*, pp. 1705–1732, 2009.

[34] M. Bogdan, E. van den Berg, C. Sabatti, W. Su, and E. Candès, "SLOPE: Adaptive variable selection via convex optimization," *Annals of Applied Statistics*, vol. 9, pp. 1103–1140, 2015.

[35] P. Bonami, M. Kilinc, and J. Linderoth, *Mixed integer nonlinear programming*. Springer, 2012, ch. Algorithms and software for convex mixed integer nonlinear programs.

[36] O. Bousquet, S. Boucheron, and G. Lugosi, *Advanced Lectures on Machine Learning*. Springer, 2004, ch. Introduction to statistical learning theory.

[37] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the Alternating Direction Method of Multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1042–1068, 2011.

[38] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2004.

[39] M. Branda, M. Bucher, M. Červinka, and A. Schwartz, "Convergence of a Scholtes-type regularization method for cardinality-constrained optimization problems with an application in sparse robust portfolio optimization," 2017, preprint arXiv:1703.10637.

[40] J. Brofos, R. Shu, and F. Zhang, "The optimistic method for model estimation," in *International Symposium on Intelligent Data Analysis*, 2016, pp. 146–157.

[41] P. Bühlmann and S. Van De Geer, *Statistics for high-dimensional data: Methods, theory, and applications*. Springer, 2011.

[42] O. P. Burdakov, C. Kanzow, and A. Schwartz, "Mathematical programs with cardinality constraints: Reformulation by complementarity-type conditions and a regularization method," *SIAM Journal on Optimization*, vol. 26, no. 1, pp. 397–425, 2016.

[43] S. Burer, *Handbook on Semidefinite, Conic and Polynomial Optimization*. Springer, 2012, ch. Copositive programming, pp. 201–218.

[44] E. Candès and B. Recht, "Exact matrix completion via convex optimization," *Communications of the ACM*, vol. 55, no. 6, pp. 111–119, 2012.

[45] E. Candès, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications in Pure and Applied Mathematics*, vol. 59, pp. 1207–1223, 2005.

[46] E. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *Journal of the ACM*, vol. 58, no. 3, pp. 11:1–11:37, 2011.

[47] C. Caramanis, S. Mannor, and H. Xu, *Optimization for machine learning*. MIT Press, 2011, ch. Robust optimization in machine learning.

[48] B. Colson, P. Marcotte, and G. Savard, "An overview of bilevel optimization," *Annals of Operations Research*, vol. 153, no. 1, pp. 235–256, 2007.

[49] P. Combettes and V. Wajs, "Signal recovery by proximal forward-backward splitting," *Multiscale Modeling and Simulation*, vol. 4, no. 4, pp. 1168–200, 2005.

[50] G. Connor and R. Korajczyk, "Performance measurement with the arbitrage pricing theory: A new framework for analysis," *Journal of Financial Economics*, vol. 15, no. 3, pp. 373–394, 1986.

[51] A. Costa and L. Liberti, "Relaxations of multilinear convex envelopes: dual is better than primal," in *International Symposium on Experimental Algorithms*. Springer, 2012, pp. 87–98.

[52] C. Croux and A. Ruiz-Gazen, "High breakdown estimators for principal components: the projection-pursuit approach revisited," *Journal of Multivariate Analysis*, vol. 95, pp. 206–226, 2005.

[53] C. De Mol, E. De Vito, and L. Rosasco, "Elastic-net regularization in learning theory," *Journal of Complexity*, vol. 25, no. 2, pp. 201–230, 2009.

[54] H. Dong, M. Ahn, and J.-S. Pang, "Structural properties of affine sparsity constraints," 2017, preprint. [Online]. Available: http://www.optimization-online.org/DB_HTML/2017/04/5965.html

[55] D. Donoho, "Compressed sensing," *IEEE Transactions in Information Theory*, vol. 52, pp. 1289–1306, 2006.

[56] I. Dunning, J. Huchette, and M. Lubin, "JuMP: A modeling language for mathematical optimization," *SIAM Review*, vol. 59, no. 2, pp. 295–320, 2017.

[57] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1, pp. 211–8, 1936.

[58] L. El Ghaoui and H. Lebret, "Robust solutions to least-squares problems with uncertain data," *SIAM Journal of Matrix Analysis and Applications*, vol. 18, no. 4, pp. 1035–64, 1997.

[59] Y. Eldar and G. Kutyniok, Eds., *Compressed sensing: Theory and applications*. Cambridge University Press, 2012.

[60] J. Fan and R. Li, "Variable selection via nonconcave penalized likelihood and its oracle properties," *Journal of the American Statistical Association*, vol. 96, pp. 1348–1360, 2001.

[61] M. Fazel, "Matrix rank minimization with applications," Ph.D. dissertation, Stanford University, 2002.

[62] M. Feng, J. E. Mitchell, J.-S. Pang, X. Shen, and A. Wächter, "Complementarity formulations of ℓ_0-norm optimization problems," 2013, preprint. [Online]. Available: http://www.optimization-online.org/DB_HTML/2013/09/4053.html

[63] M. Figueiredo and R. Nowak, "Sparse estimation with strongly correlated variables using ordered weighted ℓ_1 regularization," 2014, preprint arXiv:1409.4005.

[64] C. A. Floudas, *Deterministic global optimization: Theory, algorithms, and applications*. Kluwer, 1999.

[65] I. Frank and J. Friedman, "A statistical view of some chemometrics regression tools," *Technometrics*, vol. 35, pp. 109–148, 1993.

[66] M. Frank and P. Wolfe, "An algorithm for quadratic programming," *Naval Research Logistics Quarterly*, vol. 3, no. 1-2, pp. 95–110, 1956.

[67] J. Friedman, "Fast sparse regression and classification," *International Journal of Forecasting*, vol. 28, no. 3, pp. 722–738, 2012.

[68] G. Golub, "Some modified matrix eigenvalue problems," *SIAM Review*, vol. 15, no. 2, pp. 318–334, 1973.

[69] G. Golub and C. Van Loan, "An analysis of the total least squares problem," *SIAM Journal of Numerical Analysis*, vol. 17, no. 6, pp. 883–893, 1980.

[70] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems 27*, 2014, pp. 2672–2680.

[71] I. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," 2014, preprint arXiv:1412.6572.

[72] J.-Y. Gotoh, A. Takeda, and K. Tono, "DC formulations and algorithms for sparse optimization problems," *Mathematical Programming, Series B*, 2017.

[73] Gurobi Optimization, Inc., *Gurobi Optimizer Reference Manual*, 2016. [Online]. Available: http://www.gurobi.com

[74] L. Guttman, "To what extent can communalities reduce rank?" *Psychometrika*, vol. 23, no. 4, pp. 297–308, 1958.

[75] P. Hansen, B. Jaumard, M. Ruiz, and J. Xiong, "Global minimization of indefinite quadratic functions subject to box constraints," *Naval Research Logistics*, vol. 40, pp. 373–392, 1993.

[76] H. Harman and W. Jones, "Factor analysis by minimizing residuals (MINRES)," *Psychometrika*, vol. 31, pp. 351–368, 1966.

[77] T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning: Data mining, inference, and prediction*. Springer, 2009.

[78] A. B. Hempel and P. J. Goulart, "A novel method for modelling cardinality and rank constraints," in *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, 2014, pp. 4322–4327.

[79] T. Hoheisel, C. Kanzow, and A. Schwartz, "Theoretical and numerical comparison of relaxation methods for mathematical programs with complementarity constraints," *Mathematical Programming, Series A*, pp. 1–32, 2013.

[80] R. Horn and C. Johnson, *Matrix analysis*, 2nd ed. Cambridge University Press, 2013.

[81] P. Huber and E. Ronchetti, *Robust statistics*, 2nd ed. Wiley, 2009.

[82] M. Hubert, P. J. Rousseeuw, and S. Van Aelst, "High-breakdown robust multivariate methods," *Statistical Science*, vol. 23, no. 1, pp. 92–119, 2008.

[83] M. Hubert, P. J. Rousseeuw, and K. Van den Branden, "ROBPCA: a new approach to robust Principal Components Analysis," *Technometrics*, vol. 47, pp. 64–79, 2005.

[84] F. Husson, J. Josse, S. Le, and J. Mazet, *FactoMineR: Multivariate Exploratory Data Analysis and Data Mining*, 2015, R package version 1.31.3. [Online]. Available: http://CRAN.R-project.org/package=FactoMineR

[85] G. M. James, P. Radchenko, and J. Lv, "DASSO: connections between the Dantzig selector and Lasso," *Journal of the Royal Statistical Society, Series B*, vol. 71, no. 1, pp. 127–142, 2009.

[86] E. John and E. Yildirim, "Implementation of warm-start strategies in interior-point methods for linear programming in fixed dimension," *Computational Optimization and Applications*, vol. 41, pp. 151–183, 2008.

[87] K. Jöreskog, "Some contributions to maximum likelihood factor analysis," *Psychometrika*, vol. 32, no. 4, pp. 443–482, 1967.

[88] ——, "Structural analysis of covariance and correlation matrices," *Psychometrika*, vol. 43, no. 4, pp. 443–477, 1978.

[89] C. Kanzow and A. Schwartz, "A new regularization method for mathematical programs with complementarity constraints with strong convergence properties," *SIAM Journal on Optimization*, vol. 23, no. 2, pp. 770–798, 2013.

[90] ——, "The price of inexactness: convergence properties of relaxation methods for mathematical programs with complementarity constraints revisited," *Mathematics of Operations Research*, vol. 40, no. 2, pp. 253–275, 2014.

[91] K. Klamroth, E. Köbis, A. Schöbel, and C. Tammer, "A unified approach to uncertain optimization," *European Journal of Operational Research*, vol. 260, no. 2, pp. 403–420, 2017.

[92] K. Krishnan, "Linear programming approaches to semidefinite programming problems," Ph.D. dissertation, Rensselaer Polytechnic Institute, 2002.

[93] K. Krishnan and J. Mitchell, *Novel approaches to hard discrete optimization problems*, ser. Fields Institute Communications Series. American Mathematical Society, 2003, vol. 37, ch. Semi-infinite linear programming approaches to semidefinite programming (SDP) problems, pp. 121–140.

[94] ——, "Properties of a cutting plane method for semidefinite programming," Rensselaer Polytechnic Institute, Tech. Rep., 2003.

[95] ——, "A unifying framework for several cutting plane methods for semidefinite programming," *Optimization Methods and Software*, vol. 21, pp. 57–74, 2006.

[96] J. Lasserre, *Moments, positive polynomials and their applications*. Imperial College Press, 2009.

[97] D. Lawley and A. Maxwell, "Factor analysis as a statistical method," *Journal of the Royal Statistical Society, Series D*, vol. 12, no. 3, pp. 209–229, 1962.

[98] P. le Bodic and G. Nemhauser, "An abstract model for branching and its application to mixed integer programming," 2015, preprint arXiv:1511.01818.

[99] W. Ledermann, "On the rank of the reduced correlational matrix in multiple-factor analysis," *Psychometrika*, vol. 2, no. 2, pp. 85–93, 1937.

[100] A. Lewis, "Derivatives of spectral functions," *Mathematics of Operations Research*, vol. 21(3), pp. 576–588, 1996.

[101] ——, "Robust regularization," School of ORIE, Cornell University, Tech. Rep., 2002.

[102] A. Lewis and C. Pang, "Lipschitz behavior of the robust regularization," *SIAM Journal on Control and Optimization*, vol. 48, no. 5, pp. 3080–3104, 2009.

[103] G.-H. Lin and M. Fukushima, "A modified relaxation scheme for mathematical programs with complementarity constraints," *Annals of Operations Research*, vol. 133, no. 1, pp. 63–84, 2005.

[104] H. Liu, T. Yao, and R. Li, "Global solutions to folded concave penalized nonconvex learning," *Annals of Statistics*, vol. 44, no. 2, pp. 629–659, 2016.

[105] H. Liu, T. Yao, R. Li, and Y. Ye, "Folded concave penalized sparse linear regression: Sparsity, statistical performance, and algorithmic theory for local solutions," *Mathematical Programming, Series A*, pp. 1–34, 2016.

[106] A. Lodi, "Mixed integer programming computation," in *50 Years of Integer Programming 1958-2008*. Springer, 2010, pp. 619–645.

[107] J. Löfberg, "YALMIP: A toolbox for modeling and optimization in Matlab," in *International Symposium on Computer Aided Control Systems Design*. IEEE, 2004, pp. 284–289.

[108] K. Mardia, J. Kent, and J. Bibby, *Multivariate analysis*. Academic Press, 1979.

[109] I. Markovsky and S. V. Huffel, "Overview of total least-squares methods," *Signal Processing*, vol. 87, pp. 2283–2302, 2007.

[110] R. Mazumder, J. Friedman, and T. Hastie, "SparseNet: Coordinate descent with nonconvex penalties," *Journal of the American Statistical Association*, vol. 106, pp. 1125–1138, 2011.

[111] R. Mazumder and P. Radchenko, "The discrete Dantzig selector: Estimating sparse linear models via mixed integer linear optimization," *IEEE Transactions on Information Theory*, vol. 63, no. 5, pp. 3053–3075, 2017.

[112] A. Miller, *Subset selection in regression*. CRC Press, 2002.

[113] R. Misener and C. A. Floudas, "Global optimization of mixed-integer quadratically-constrained quadratic programs (MIQCQP) through piecewise-linear and edge-concave relaxations," *Mathematical Programming, Series B*, vol. 136, pp. 155–182, 2012.

[114] S. Morgenthaler, "A survey of robust statistics," *Statistical Methods and Applications*, vol. 15, pp. 271–293, 2007.

[115] S. Mosci, L. Rosasco, M. Santoro, A. Verri, and S. Villa, "Solving structured sparsity regularization with proximal methods," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2010, pp. 418–433.

[116] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*. Kluwer, 2004.

[117] ——, "Smooth minimization of non-smooth functions," *Mathematical Programming, Series A*, vol. 103, pp. 127–152, 2005.

[118] ——, "Gradient methods for minimizing composite objective function," Center for Operations Research and Econometrics (CORE), Catholic University of Louvain, Tech. Rep. 76, 2007.

[119] B. O'Donoghue, E. Chu, N. Parikh, and S. Boyd, "Conic optimization via operator splitting and homogeneous self-dual embedding," *Journal of Optimization Theory and Applications*, vol. 169, no. 3, pp. 1042–1068, 2016.

[120] M. Osborne, B. Presnell, and B. Turlach, "On the Lasso and its dual," *Journal of Computational and Graphical Statistics*, vol. 9, no. 2, pp. 319–337, 2000.

[121] M. Overton and R. Womersley, "On the sum of the largest eigenvalues of a symmetric matrix," *SIAM Journal of Matrix Analysis and Applications*, vol. 13, pp. 41–45, 1992.

[122] G. Pataki, "On the rank of extreme matrices in semidefinite programs and the multiplicity of optimal eigenvalues," *Mathematics of Operations Research*, vol. 23, pp. 339–358, 1998.

[123] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, 2014. [Online]. Available: http://www.R-project.org/

[124] G. Raiche and D. Magis, "nFactors: Parallel analysis and non graphical solutions to the Cattell Scree test," 2011. [Online]. Available: http://cran.r-project.org/web/packages/nFactors/index.html

[125] C. R. Rao, *Linear statistical inference and its applications*. Wiley, New York, 1973.

[126] B. Recht, M. Fazel, and P. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM Review*, vol. 52, no. 3, pp. 471–501, 2010.

[127] W. Revelle, *psych: Procedures for psychological, psychometric, and personality research*, 2015. [Online]. Available: http://CRAN.R-project.org/package=psych

[128] G. Riccia and A. Shapiro, "Minimum rank and minimum trace of covariance matrices," *Psychometrika*, vol. 47, pp. 443–448, 1982.

[129] D. Robertson and J. Symons, "Maximum likelihood factor analysis with rank-deficient sample covariance matrices," *Journal of Multivariate Analysis*, vol. 98, no. 4, pp. 813–828, 2007.

[130] R. Rockafeller, *Convex analysis*. Princeton University Press, 1970.

[131] P. J. Rousseeuw and A. M. Leroy, *Robust regression and outlier detection*. Wiley, 1987.

[132] S. Roweis and Z. Ghahramani, "A unifying review of linear gaussian models," *Neural Computation*, vol. 11, no. 2, pp. 305–345, 1999.

[133] D. Rubin and D. Thayer, "EM algorithms for ML factor analysis," *Psychometrika*, vol. 47, no. 1, pp. 69–76, 1982.

[134] N. V. Sahinidis, *BARON 14.3.1: Global Optimization of Mixed-Integer Nonlinear Programs,* User's Manual, 2014.

[135] J. Saunderson, V. Chandrasekaran, P. Parrilo, and A. Willsky, "Diagonal and low-rank matrix decompositions, correlation matrices, and ellipsoid fitting," *SIAM Journal on Matrix Analysis and Applications*, vol. 33, no. 4, pp. 1395–1416, 2012.

[136] S. Scholtes and M. Stöhr, "Exact penalization of mathematical programs with equilibrium constraints," *SIAM Journal on Control and Optimization*, vol. 37, no. 2, pp. 617–652, 1999.

[137] U. Shaham, Y. Yamada, and S. Negahban, "Understanding adversarial training: Increasing local stability of neural nets through robust optimization," 2015, preprint arXiv:1511.05432.

[138] A. Shapiro, "Rank-reducibility of a symmetric matrix and sampling theory of minimum trace factor analysis," *Psychometrika*, vol. 47, pp. 187–199, 1982.

[139] A. Shapiro and J. M. F. ten Berge, "Statistical inference of minimum rank factor analysis," *Psychometrika*, vol. 67, pp. 79–94, 2002.

[140] X. Shen, W. Pan, and Y. Zhu, "Likelihood-based selection and sharp parameter estimation," *Journal of the American Statistical Association*, vol. 107, no. 497, pp. 223–232, 2012.

[141] X. Shen, W. Pan, Y. Zhu, and H. Zhou, "On constrained and regularized high-dimensional regression," *Annals of the Institute of Statistical Mathematics*, vol. 65, no. 5, pp. 807–832, 2013.

[142] SIGKDD and Netflix, "Soft modelling by latent variables: the nonlinear iterative partial least squares (NIPALS) approach," *Proceedings of the KDD Cup and Workshop*, 2007.

[143] C. Spearman, "'General intelligence,' objectively determined and measured," *American Journal of Psychology*, vol. 15, pp. 201–293, 1904.

[144] G. Stewart and J.-G. Sun, *Matrix perturbation theory*. Academic Press, 1990.

[145] P. D. Tao and L. T. H. An, "Convex analysis approach to DC programming: theory, algorithms, and applications," *Acta Mathematica Vietnamica*, vol. 22, pp. 287–355, 1997.

[146] M. Tawarmalani, J.-P. P. Richard, and C. Xiong, "Explicit convex and concave envelopes through polyhedral subdivisions," *Mathematical Programming, Series A*, vol. 138, pp. 531–577, 2013.

[147] M. Tawarmalani and N. V. Sahinidis, "Convex extensions and envelopes of lower semi-continuous functions," *Mathematical Programming, Series A*, vol. 93, pp. 247–263, 2002.

[148] ——, *Convexification and global optimization in continuous and mixed-integer nonlinear programming: theory, algorithms, software, and applications*, ser. Nonconvex Optimization and its Applications. Kluwer, 2002, vol. 65.

[149] J. M. F. ten Berge, "Some recent developments in factor analysis and the search for proper communalities," in *Advances in data science and classification*. Springer, 1998, pp. 325–334.

[150] J. M. F. ten Berge and H. Kiers, "A numerical approach to the approximate and the exact minimum rank of a covariance matrix," *Psychometrika*, vol. 56, pp. 309–315, 1991.

[151] ——, "The minimum rank factor analysis program MRFA," 2003. [Online]. Available: http://www.ppsw.rug.nl/~kiers/

[152] J. M. F. ten Berge, T. A. Snijders, and F. E. Zegers, "Computational aspects of the greatest lower bound to the reliability and constrained minimum trace factor analysis," *Psychometrika*, vol. 46, no. 2, pp. 201–213, 1981.

[153] Y. Teng, L. Yang, B. Yu, and X. Song, "An augmented Lagrangian proximal alternating method for sparse discrete optimization problems," 2017, preprint. [Online]. Available: http://www.optimization-online.org/DB_HTML/2017/02/5876.html

[154] M. Thiao, P. D. Tao, and L. T. H. An, "A DC programming approach for sparse eigenvalue problem," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 2010, pp. 1063–1070.

[155] L. Thurstone, *Multiple Factor Analysis: a development and expansion of the vectors of the mind.* University of Chicago Press, 1947.

[156] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society, Series B*, vol. 58, pp. 267–288, 1996.

[157] A. Tillman and M. Pfetsch, "The computational complexity of the Restricted Isometry Property, the nullspace property, and related concepts in compressed sensing," *IEEE Transactions in Information Theory*, vol. 60, pp. 1248–1259, 2014.

[158] K. Toh, M. Todd, and R. Tutuncu, "SDPT3—a MATLAB software package for semidefinite programming," *Optimization Methods and Software*, vol. 11, pp. 545–581, 1999.

[159] K. Tono, A. Takeda, and J.-Y. Gotoh, "Efficient DC algorithm for constrained sparse optimization," 2017, preprint arXiv:1701.08498.

[160] T. Tulabandhula and C. Rudin, "Robust optimization using machine learning for uncertainty sets," 2014, preprint arXiv:1407.1097.

[161] L. Vandenberghe and S. Boyd, "Semidefinite programming," *SIAM Review*, vol. 38, no. 1, pp. 49–95, 1996.

[162] H. Xu, C. Caramanis, and S. Mannor, "Robust regression and Lasso," *IEEE Transactions in Information Theory*, vol. 56, no. 7, pp. 3561–74, 2010.

[163] E. Yildirim and S. Wright, "Warm-start strategies in interior-point methods for linear programming," *SIAM Journal on Optimization*, vol. 12, pp. 782–810, 2002.

[164] C. Zhang, "Nearly unbiased variable selection under minimax concave penalty," *The Annals of Statistics*, vol. 38, pp. 894–942, 2010.

[165] C.-H. Zhang and T. Zhang, "A general theory of concave regularization for high-dimensional sparse estimation problems," *Statistical Science*, pp. 576–593, 2012.

[166] T. Zhang, "Analysis of multi-stage convex relaxation for sparse regularization," *Journal of Machine Learning Research*, vol. 11, pp. 1081–1107, 2010.

[167] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society, Series B*, vol. 67, no. 2, pp. 301–320, 2005.