Journal of Machine Learning Research 23 (2022) 1-52 Submitted 1/20; Revised 11/21; Published 4/22

New Insights for the Multivariate Square-Root Lasso

Aaron J. Molstad AMOLSTAD@UFL.EDU
Department of Statistics and Genetics Institute
University of Florida
Gainesville, FL 32611, USA

Editor: David Wipf

Abstract

We study the multivariate square-root lasso, a method for fitting the multivariate response linear regression model with dependent errors. This estimator minimizes the nuclear norm of the residual matrix plus a convex penalty. Unlike existing methods that require explicit estimates of the error precision (inverse covariance) matrix, the multivariate square-root lasso implicitly accounts for error dependence and is the solution to a convex optimization problem. We establish error bounds which reveal that like the univariate square-root lasso, the multivariate square-root lasso is pivotal with respect to the unknown error covariance matrix. In addition, we propose a variation of the alternating direction method of multipliers algorithm to compute the estimator and discuss an accelerated first order algorithm that can be applied in certain cases. In both simulation studies and a genomic data application, we show that the multivariate square-root lasso can outperform more computationally intensive methods that require explicit estimation of the error precision matrix.

Keywords: pivotal estimation, multivariate response linear regression, convex optimization, covariance matrix estimation

1. Introduction

Modeling the linear relationship between a p-variate vector of predictors and a q-variate vector of responses is a central task in multivariate analysis. In this article, we will assume that the observed response vectors for the n subjects in the study, $\boldsymbol{y}_1, \dots, \boldsymbol{y}_n$, are realizations of the random vectors

$$\boldsymbol{\beta}_{*0} + \boldsymbol{\beta}_*^\top \boldsymbol{x}_i + \boldsymbol{\epsilon}_i \tag{1}$$

for $i \in \{1, \dots, n\}$, where $\boldsymbol{x}_i \in \mathbb{R}^p$ is the predictor for the ith subject, $\boldsymbol{\beta}_{*0} \in \mathbb{R}^q$ is the unknown intercept vector, and $\boldsymbol{\beta}_* \in \mathbb{R}^{p \times q}$ is the unknown regression coefficient matrix. We assume that the $\boldsymbol{\epsilon}_i$ are independent and identically distributed q-variate random vectors with mean zero and unknown error covariance matrix $\boldsymbol{\Sigma}_* \in \mathbb{S}_+^q$, where $\mathbb{S}_+^q$ is the set of $q \times q$ symmetric positive definite matrices. Let $\boldsymbol{\Omega}_* = \boldsymbol{\Sigma}_*^{-1}$ be the unknown error precision matrix. For notational convenience, let $\boldsymbol{Y} = (\boldsymbol{y}_1 - \bar{\boldsymbol{y}}, \dots, \boldsymbol{y}_n - \bar{\boldsymbol{y}})^\top \in \mathbb{R}^{n \times q}$ and $\boldsymbol{X} = (\boldsymbol{x}_1 - \bar{\boldsymbol{x}}, \dots, \boldsymbol{x}_n - \bar{\boldsymbol{x}})^\top \in \mathbb{R}^{n \times p}$, where $\bar{\boldsymbol{y}} = n^{-1} \sum_{i=1}^n \boldsymbol{y}_i$ and $\bar{\boldsymbol{x}} = n^{-1} \sum_{i=1}^n \boldsymbol{x}_i$.

Many methods exist for fitting the multivariate response linear regression model in (1). When $n > p$ and the $\boldsymbol{\epsilon}_i$ are multivariate normal, the maximum likelihood estimator (and equivalently, least squares estimator) of $\boldsymbol{\beta}_*$ does not require knowledge of nor an estimate of $\boldsymbol{\Omega}_*$. When $p \geq n$ the least squares estimator is not unique, so a natural alternative is to estimate $\boldsymbol{\beta}_*$ by minimizing a penalized least squares criterion (i.e., penalized squared

Frobenius norm of the residual matrix) using penalties that exploit the matrix structure of the unknown regression coefficients (Turlach et al., 2005; Yuan et al., 2007; Obozinski et al., 2011; Negahban and Wainwright, 2011). However, the penalized least squares criterion implicitly assumes $\boldsymbol{\Sigma}_* \propto \boldsymbol{I}_q$: the penalized least squares estimator is equivalent to the penalized normal maximum likelihood estimator under the assumption that $\boldsymbol{\Sigma}_* \propto \boldsymbol{I}_q$.

This limitation of penalized least squares has motivated numerous methods which incorporate an estimate of $\boldsymbol{\Omega}_*$ into the estimation procedure for $\boldsymbol{\beta}_*$. One class of methods jointly estimates $\boldsymbol{\Omega}_*$ and $\boldsymbol{\beta}_*$ by maximizing a penalized normal log-likelihood (Rothman et al., 2010; Yin and Li, 2011) using L_1-norm penalties—as defined in (3)—on the optimization variable corresponding to $\boldsymbol{\beta}_*$ and on off-diagonal entries of the optimization variable corresponding to $\boldsymbol{\Omega}_*$. Wang (2015) proposed an alternative approach which performs estimation column-by-column, estimating the kth columns of $\boldsymbol{\beta}_*$ and $\boldsymbol{\Omega}_*$ jointly for $k \in \{1, \dots, q\}$. While these methods can perform well in certain settings, an estimate of $\boldsymbol{\Omega}_*$ is often not needed by the practitioner. Regardless, the methods of Rothman et al. (2010), Yin and Li (2011), and Wang (2015) require estimating $O(q^2)$ precision matrix parameters, and in the case of Rothman et al. (2010) and Yin and Li (2011), require solving a computationally burdensome nonconvex optimization problem.

An ideal estimation criterion for $\boldsymbol{\beta}_*$ is convex and can account for error dependence without requiring an explicit estimate of $\boldsymbol{\Omega}_*$ or $\boldsymbol{\Sigma}_*$. To this end, we study the class of estimators

$$\operatorname*{arg\,min}_{\boldsymbol{\beta} \in \mathbb{R}^{p \times q}} \left\{ \frac{1}{\sqrt{n}} \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_* + \lambda g(\boldsymbol{\beta}) \right\}, \tag{2}$$

where $\|\boldsymbol{A}\|_* = \operatorname{tr}\{(\boldsymbol{A}^\top \boldsymbol{A})^{1/2}\}$ denotes the nuclear norm of a matrix $\boldsymbol{A}$ (i.e., the norm which sums the singular values of its matrix-valued argument), g is a nonnegative penalty function, and $\lambda > 0$ is a user-specified tuning parameter. When g is a norm, which we will assume throughout, the objective function in (2) is convex. The estimator in (2) with L_1-norm penalty was originally proposed by Van de Geer and Stucky (2016). Their focus was on using (2) to construct confidence sets for high-dimensional regression coefficient vectors in univariate response linear regression. In this article, we study (2) as a method for fitting (1) in high-dimensional settings.

Of course, the class of estimators defined by (2) is applicable with penalties beyond the L_1-norm. We focus on three versions of (2), each defined by their choice of penalty g:

$$\text{Lasso (L)} \qquad \|\boldsymbol{\beta}\|_1 = \sum_{j=1}^{p} \sum_{k=1}^{q} |\boldsymbol{\beta}_{j,k}|, \tag{3}$$

$$\text{Group lasso (GL)} \qquad \|\boldsymbol{\beta}\|_{1,2} = \sum_{j=1}^{p} \left(\sum_{k=1}^{q} \boldsymbol{\beta}_{j,k}^2 \right)^{1/2}, \tag{4}$$

$$\text{Nuclear norm (LR)} \qquad \|\boldsymbol{\beta}\|_* = \operatorname{tr}\left\{(\boldsymbol{\beta}^\top \boldsymbol{\beta})^{1/2}\right\} = \sum_{j=1}^{\min(p,q)} \sigma_j(\boldsymbol{\beta}), \tag{5}$$

where $\sigma_j(\boldsymbol{A})$ and $\boldsymbol{A}_{j,k}$ denote the jth largest singular value and (j,k)th entry of the matrix $\boldsymbol{A}$, respectively. When referring to (2) with the penalties (3), (4), and (5), we use $\hat{\boldsymbol{\beta}}_{\mathrm{L}}$,

$\hat{\boldsymbol{\beta}}_{\rm GL}$, and $\hat{\boldsymbol{\beta}}_{\rm LR}$, respectively. For simplicity, we refer to the class of estimators (2) as the *multivariate square-root lasso* regardless of the penalty g.

Relative to the L_1-norm penalty, which encourages estimates of $\boldsymbol{\beta}_*$ with unstructured sparsity, the group lasso and nuclear norm penalties are especially well-suited for multivariate response linear regression. The group lasso penalty exploits the assumption that many predictors are irrelevant for all q responses by encouraging estimates of $\boldsymbol{\beta}_*$ with some rows entirely equal to zero (Yuan and Lin, 2006; Obozinski et al., 2011; Lounici et al., 2011). The nuclear norm penalty, in contrast, acts as a lasso penalty on the singular values of the optimization variable $\boldsymbol{\beta}$ and thus promotes estimates of $\boldsymbol{\beta}_*$ with low rank (Yuan et al., 2007; Negahban and Wainwright, 2011; Chen et al., 2013), hence the shorthand LR. Low rankness of $\boldsymbol{\beta}_*$ is assumed in reduced rank regression (Reinsel and Velu, 1998), a classical method for dimension reduction in (1).

Computing (2) is nontrivial because the nuclear norm of residuals, though convex, is nondifferentiable. To date, there are no specialized algorithms to compute (2) with convergence guarantees. Van de Geer and Stucky (2016) suggested an iterative procedure for computing $\hat{\boldsymbol{\beta}}_{\rm L}$, but unfortunately, we found their algorithm cannot be used to solve the optimization in general. In a later version of Van de Geer and Stucky (2016) appearing in a PhD thesis, Stucky (2017) computed (2) using the general purpose convex solver `CVX` (Grant and Boyd, 2014), which can be slow in high-dimensional settings.

In addition to the computational challenges, little is known about (2) in terms of its statistical properties. While Van de Geer and Stucky (2016) and van de Geer (2016) pointed out the connection between (2) and the univariate ($q = 1$) square-root lasso (Belloni et al., 2011; Sun and Zhang, 2012; Bunea et al., 2014; Derumigny, 2018), their focus was on (2) as a means for constructing confidence intervals. They did not establish any statistical properties of (2) nor did they explore the empirical performance of (2) in the context of fitting (1).

In this article, we study (2) from theoretical, computational, and empirical perspectives. We prove that like the univariate square-root lasso, (2) is pivotal in the sense that the value of the tuning parameter λ leading to near-oracle performance is determined by a random quantity whose distribution does not depend on the unknown error covariance $\boldsymbol{\Sigma}_*$. In so doing, we establish error bounds for (2) with arbitrary g, then specialize these results to the penalties in (3), (4), and (5). We also argue that (2), like the univariate square-root lasso, can be interpreted as implicitly incorporating an estimate of the error precision matrix into the criterion for estimating $\boldsymbol{\beta}_*$. Through simulation studies, we show that (2) can perform as well or better than methods that estimate $\boldsymbol{\beta}_*$ and $\boldsymbol{\Omega}_*$ jointly, both of which outperform penalized least squares estimators when $\boldsymbol{\Omega}_*$ has many nonzero off-diagonals. Based on our theory, we also study a tuning procedure that requires computing (2) for only a single value of the tuning parameter, i.e., does not require cross-validation. Finally, we propose two algorithms to compute (2) efficiently: one algorithm that can be used in any setting and has convergence guarantees, and a second algorithm that can be applied when $n > q$ and the tuning parameter is sufficiently large. Our algorithms are often 100 or more times faster than `CVX` in the simulation settings we consider. An R package implementing our method is available for download at `https://github.com/ajmolstad/MSRL`.

Before we proceed, we define notation which will be used throughout the article. Let $\boldsymbol{I}_s$ be the $s \times s$ identity matrix. When we write $(\boldsymbol{U}, \boldsymbol{D}, \boldsymbol{V}) = {\rm svd}(\boldsymbol{A})$, we refer to the components of the singular value decomposition $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^\top \in \mathbb{R}^{a \times b}$, where letting $s = \min(a, b)$,

$\boldsymbol{U} \in \mathbb{R}^{a \times s}$ and $\boldsymbol{V} \in \mathbb{R}^{b \times s}$ with $\boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{V}^\top \boldsymbol{V} = \boldsymbol{I}_s$, and $\boldsymbol{D} \in \mathbb{R}^{s \times s}$ is a diagonal matrix with $\boldsymbol{D}_{k,k} = \sigma_k(\boldsymbol{A}) \geq 0$ for $k \in \{1, \dots, s\}$. Define the norms $\|\boldsymbol{A}\|_F = (\sum_{j,k} \boldsymbol{A}_{j,k}^2)^{1/2}$, $\|\boldsymbol{A}\|_\infty = \max_{j,k} |\boldsymbol{A}_{j,k}|$, $\|\boldsymbol{A}\| = \sigma_1(\boldsymbol{A})$, and let $\varphi_j(\boldsymbol{A})$ denote the jth largest eigenvalue of square matrix $\boldsymbol{A}$. For a symmetric matrix $\boldsymbol{M}$, define $\|\boldsymbol{A}\|_{\boldsymbol{M}}^2 = \mathrm{tr}(\boldsymbol{A}^\top \boldsymbol{M} \boldsymbol{A})$. Let $\|\boldsymbol{A}\|_{\infty,2} = \max_j \|\boldsymbol{A}_{j,\cdot}\|_2$, where $\|\boldsymbol{a}\|_2$ denotes the Euclidean norm of a vector $\boldsymbol{a}$ and $\boldsymbol{A}_{j,\cdot}$ denotes the jth row of $\boldsymbol{A}$. Similarly, let $\boldsymbol{A}_{\cdot,k}$ denote the kth column of $\boldsymbol{A}$. For a subspace $\mathcal{R} \subseteq \mathbb{R}^d$, define the orthogonal complement of $\mathcal{R}$ as $\mathcal{R}^\perp = \{\boldsymbol{v} \in \mathbb{R}^d : \boldsymbol{v}^\top \boldsymbol{u} = 0 \text{ for all } \boldsymbol{u} \in \mathcal{R}\}$. For a set $\mathcal{T}$, let $|\mathcal{T}|$ be its cardinality. For sequences a_n and b_n, we use the notation $a_n \lesssim b_n$ to mean that there exists a constant $K > 0$ such that $a_n \leq K b_n$ for all n sufficiently large. Finally, let $[n] = \{1, 2, \dots, n\}$ for all positive integers n.

2. The Multivariate Square-root Lasso

2.1 Implicit Covariance Estimation

If the $\boldsymbol{\epsilon}_i$ were multivariate normal and the precision matrix $\boldsymbol{\Omega}_*$ were known, the penalized maximum likelihood estimator of $\boldsymbol{\beta}_*$ would be

$$\operatorname*{arg\,min}_{\boldsymbol{\beta} \in \mathbb{R}^{p \times q}} \left[\frac{1}{n} \mathrm{tr}\big\{ (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}) \boldsymbol{\Omega}_* (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})^\top \big\} + \lambda g(\boldsymbol{\beta}) \right], \tag{6}$$

which can be interpreted as a penalized weighted least squares estimator. Based on the first order conditions for (6), it can be verified that the solution depends on the error precision $\boldsymbol{\Omega}_*$. Of course, (6) cannot be used in practice because $\boldsymbol{\Omega}_*$ is generally unknown. Instead, a popular alternative proposed by Rothman et al. (2010) is the jointly penalized maximum likelihood estimator

$$\operatorname*{arg\,min}_{\boldsymbol{\beta} \in \mathbb{R}^{p \times q}, \boldsymbol{\Omega} \in \mathbb{S}_+^q} \left[\frac{1}{n} \mathrm{tr}\big\{ (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}) \boldsymbol{\Omega} (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})^\top \big\} - \log\det(\boldsymbol{\Omega}) + \lambda g(\boldsymbol{\beta}) + \gamma \sum_{j \neq k} |\boldsymbol{\Omega}_{j,k}| \right], \tag{7}$$

where $\gamma > 0$ is a user-specified tuning parameter. Unlike (6), the optimization problem in (7) is nonconvex. Solving (7) requires iteratively updating $\boldsymbol{\beta}$ with $\boldsymbol{\Omega}$ held fixed and vice versa (Rothman et al., 2010), which can be time-consuming in high-dimensional settings.

As an alternative to the computationally intensive task of solving (7), one could instead plug an estimate of $\boldsymbol{\Omega}_* = \boldsymbol{\Sigma}_*^{-1}$ into (6). However, standard estimators of $\boldsymbol{\Sigma}_*$ are themselves functions of $\boldsymbol{\beta}_*$, e.g., $n^{-1}(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}_*)^\top (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}_*)$. This naturally raises the question of whether one could construct a weighted least squares criterion like (6) wherein the weight is itself a function of the optimization variable $\boldsymbol{\beta}$. In fact, the nuclear norm can be interpreted in exactly this way because

$$\frac{1}{\sqrt{n}} \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_* = \frac{1}{n} \mathrm{tr}\big\{ (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}) \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^\dagger (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})^\top \big\},$$

where the weight matrix $\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$ is given by

$$\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \frac{1}{\sqrt{n}} \big\{ (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})^\top (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}) \big\}^{1/2},$$

and $\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{\dagger}$ is its Moore-Penrose pseudoinverse. That is, the nuclear norm of residuals can be expressed as a weighted least squares criterion where the weight matrix is an estimate of the square-root error precision matrix $\boldsymbol{\Omega}_*^{1/2} = \boldsymbol{\Sigma}_*^{-1/2}$. Furthermore, the multivariate square-root lasso can, in some situations, be interpreted as jointly estimating the error covariance and regression coefficient matrix, like (7).

Lemma 1 *(Van de Geer and Stucky, 2016, Lemma 1) Define $\hat{\boldsymbol{\beta}}_g$ as the solution to* (2)*, and define $(\bar{\boldsymbol{\beta}}_g, \bar{\boldsymbol{\Sigma}}_g^{1/2})$ as*

$$\underset{\boldsymbol{\beta}\in\mathbb{R}^{p\times q}, \boldsymbol{\Sigma}^{1/2}\in\mathbb{S}_+^q}{\arg\min} \left[\frac{1}{2n}\mathrm{tr}\big\{(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})\boldsymbol{\Sigma}^{-1/2}(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})^\top\big\} + \frac{\mathrm{tr}(\boldsymbol{\Sigma}^{1/2})}{2} + \lambda g(\boldsymbol{\beta})\right], \tag{8}$$

assuming the minimum is obtained for some $\boldsymbol{\Sigma}^{1/2} \in \mathbb{S}_+^q$. If $\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_g$ has q nonzero singular values, then the estimator in (8) *satisfies*

$$\bar{\boldsymbol{\Sigma}}_g = \frac{1}{n}(\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_g)^\top(\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_g) \quad \textit{and} \quad \bar{\boldsymbol{\beta}}_g = \hat{\boldsymbol{\beta}}_g.$$

Lemma 1 suggests that we can solve the joint optimization problem (8) by solving (2) directly: we need not explicitly estimate $\boldsymbol{\Sigma}_*$ or $\boldsymbol{\Omega}_*$. It is in this sense that we argue (2) implicitly estimates the error covariance. This is in contrast to (7), which requires an explicit estimate of $\boldsymbol{\Omega}_*$. In our simulation studies, we demonstrate that this implicit covariance estimation yields an estimator of $\boldsymbol{\beta}_*$ which performs similarly to methods that use $\boldsymbol{\Omega}_*$, or an estimate thereof, in their estimation criterion.

The relationship between (2) and (8) established in Lemma 1 holds only when $\hat{\boldsymbol{\beta}}_g$ leads to a residual matrix with q nonzero singular values. However, we emphasize that Lemma 1 simply provides one way to characterize $\hat{\boldsymbol{\beta}}_g$ in this special setting. We do not require or assume that $\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_g$ has rank q. Our theory (Section 3) and algorithm (Section 4.2) apply to (2) even when $\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_g$ has fewer than q nonzero singular values. In these settings, the interpretation of the nuclear norm of residuals as a weighted residual sum of squares with weight matrix $\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{\dagger}$ still applies, but $\hat{\boldsymbol{\beta}}_g$ cannot be interpreted as the solution to a joint optimization problem as neatly as in (8).

2.2 Relationship to Existing Methods

The univariate square-root lasso (Belloni et al., 2011; Sun and Zhang, 2012; Bunea et al., 2014; Derumigny, 2018) is a special case of (2) when $q = 1$ and g is the L_1-norm. However, there is an important difference between the univariate and multivariate square-root lasso estimators in terms of how they relate to their penalized least squares analogs.

Remark 2 *Suppose g is the L_1-norm. When $q = 1$, the univariate square-root lasso estimator has a solution path equivalent to that of the L_1-penalized least squares estimator (Tian et al., 2018). When $q \geq 2$,* (2) *and the L_1-penalized least squares estimator do not have equivalent solution paths in general.*

Mathematically, Remark 2 follows from the fact that the unpenalized objective function for the multivariate square-root lasso (with $q \geq 2$), unlike the univariate square-root lasso, cannot be expressed as the square-root of its least squares analog, i.e., $\|\boldsymbol{A}\|_F \neq \operatorname{tr}\{(\boldsymbol{A}^\top \boldsymbol{A})^{1/2}\}$ in general. Thus with $q \geq 2$, (2) defines a class of estimators distinct from its least squares analog in the sense that their solution paths are distinct. Our simulation results show that the multivariate square-root lasso performs more like the normal penalized maximum likelihood estimator of Rothman et al. (2010), which explicitly estimates the error precision matrix, than the penalized least squares estimator.

Our estimator (2) is not the only multivariate generalization of the univariate square-root lasso. Liu et al. (2015) proposed an estimator which minimizes the sum of the Euclidean norm of residuals for each response plus a penalty on the optimization variable corresponding to $\boldsymbol{\beta}_*$. However, the method of Liu et al. (2015) assumes that $\boldsymbol{\Sigma}_*$ is diagonal. In addition, when the penalty is separable across the columns of its matrix argument (e.g., when using (3)), the method of Liu et al. (2015) is equivalent to performing q separate univariate square-root lasso regressions with the same tuning parameter used for each response. In our simulation studies, the method of Liu et al. (2015) outperforms penalized least squares estimators, but tends to be outperformed by (2) when $\boldsymbol{\Sigma}_*$ is not diagonal. For more details, see our description of their method in Section 5.2.

3. Statistical Properties

3.1 Overview

In this section, we establish Frobenius norm error bounds for (2). We first provide a general error bound, then specialize this result to penalties (3), (4), and (5).

For each of the following results, we assume that $\boldsymbol{\beta}_*$ belongs to a subspace $\mathcal{M}$, and choose the penalty g according to $\mathcal{M}$. To make matters concrete, we define $\mathcal{M}$ as the *model subspace* and assume throughout that $\boldsymbol{\beta}_* \in \mathcal{M}$. Let $\mathcal{M}^\perp$ denote the orthogonal complement of the model subspace $\mathcal{M}$. Define the subspace $\mathcal{N}^\perp$ as the *perturbation subspace* and let $\mathcal{N}$ be its orthogonal complement. We will give concrete examples of $\mathcal{M}$ and $\mathcal{N}^\perp$ under the three different model assumptions momentarily. For subspace pairs $(\mathcal{M}, \mathcal{N}^\perp)$ for which $\mathcal{M} \subseteq \mathcal{N}$, a penalty function g is said to be *decomposable* with respect to the subspace pair $(\mathcal{M}, \mathcal{N}^\perp)$ if $g(\boldsymbol{A} + \boldsymbol{B}) = g(\boldsymbol{A}) + g(\boldsymbol{B})$ for all $\boldsymbol{A} \in \mathcal{M}$ and $\boldsymbol{B} \in \mathcal{N}^\perp$. See Negahban et al. (2012) for a further discussion of model subspaces, perturbation subspaces, and decomposability.

Throughout, let $\tilde{g}$ denote the dual norm of g, and define the *subspace compatibility constant* (Negahban et al., 2012) with respect to g as

$$\Psi_g(\mathcal{N}) = \sup_{\boldsymbol{A} \in \mathcal{N} \setminus \{0\}} \frac{g(\boldsymbol{A})}{\|\boldsymbol{A}\|_F}.$$

In the following, we consider three model subspaces for $\boldsymbol{\beta}_*$ (**M1**–**M3**): each corresponds to a distinct subspace pair and decomposable penalty function g.

M1. (Elementwise sparsity) We assume that many entries of $\boldsymbol{\beta}_*$ are zero. Letting $\mathcal{S} = \{(j,k) : \boldsymbol{\beta}_{*j,k} \neq 0 \ , (j,k) \in [p] \times [q]\}$, define the subspace pair

$$\mathcal{M}_{\mathrm{L}} = \{\boldsymbol{\beta} \in \mathbb{R}^{p \times q} : \boldsymbol{\beta}_{j,k} = 0, (j,k) \notin \mathcal{S}\}, \ \ \mathcal{N}_{\mathrm{L}}^\perp = \{\boldsymbol{\beta} \in \mathbb{R}^{p \times q} : \boldsymbol{\beta}_{j,k} = 0, (j,k) \in \mathcal{S}\}.$$

The penalty function $g(\cdot) = \|\cdot\|_1$ is decomposable with respect to $(\mathcal{M}_{\mathrm{L}}, \mathcal{N}_{\mathrm{L}}^\perp)$ and $\Psi_{\|\cdot\|_1}(\mathcal{N}_{\mathrm{L}}) \leq \sqrt{|\mathcal{S}|}$ (Negahban et al., 2012).

M2. (Row-wise sparsity) We assume that many rows of $\boldsymbol{\beta}_*$ are entirely zero. Letting $\mathcal{G} = \{j : \boldsymbol{\beta}_{*j,\cdot} \neq 0,\ j \in [p]\}$, define the subspace pair

$$\mathcal{M}_{\rm GL} = \{\boldsymbol{\beta} \in \mathbb{R}^{p\times q} : \boldsymbol{\beta}_{j,\cdot} = 0, j \notin \mathcal{G}\},\ \ \mathcal{N}_{\rm GL}^{\perp} = \{\boldsymbol{\beta} \in \mathbb{R}^{p\times q} : \boldsymbol{\beta}_{j,\cdot} = 0, j \in \mathcal{G}\}.$$

The penalty function $g(\cdot) = \|\cdot\|_{1,2}$ is decomposable with respect to $(\mathcal{M}_{\rm GL}, \mathcal{N}_{\rm GL}^{\perp})$ and $\Psi_{\|\cdot\|_{1,2}}(\mathcal{N}_{\rm GL}) \leq \sqrt{|\mathcal{G}|}$ (Liu et al., 2015).

M3. (Low-rankness) We assume that $\text{rank}(\boldsymbol{\beta}_*) = r$ where $r \ll \min(p, q)$. Letting $(\boldsymbol{U}_*, \boldsymbol{D}_*, \boldsymbol{V}_*) = \text{svd}(\boldsymbol{\beta}_*)$, define $\mathcal{U} = \text{span}(\boldsymbol{u}_{*1}, \dots, \boldsymbol{u}_{*r})$ and $\mathcal{V} = \text{span}(\boldsymbol{v}_{*1}, \dots, \boldsymbol{v}_{*r})$ where $\boldsymbol{u}_{*k}$ and $\boldsymbol{v}_{*k}$ are the kth columns of $\boldsymbol{U}_*$ and $\boldsymbol{V}_*$, respectively, for $k \in [r]$. Let $\mathcal{U}^{\perp}$ and $\mathcal{V}^{\perp}$ denote the orthogonal complements of $\mathcal{U}$ and $\mathcal{V}$, respectively. Define the subspace pair

$$\mathcal{M}_{\rm LR} = \{\boldsymbol{\beta} \in \mathbb{R}^{p\times q} : \text{row}(\boldsymbol{\beta}) \subseteq \mathcal{V}, \text{col}(\boldsymbol{\beta}) \subseteq \mathcal{U}\},$$
$$\mathcal{N}_{\rm LR}^{\perp} = \{\boldsymbol{\beta} \in \mathbb{R}^{p\times q} : \text{row}(\boldsymbol{\beta}) \subseteq \mathcal{V}^{\perp}, \text{col}(\boldsymbol{\beta}) \subseteq \mathcal{U}^{\perp}\},$$

where $\text{row}(\boldsymbol{A})$ and $\text{col}(\boldsymbol{A})$ are the row and column spaces of a matrix $\boldsymbol{A}$, respectively. The penalty function $g(\cdot) = \|\cdot\|_*$ is decomposable with respect to $(\mathcal{M}_{\rm LR}, \mathcal{N}_{\rm LR}^{\perp})$ and $\Psi_{\|\cdot\|_*}(\mathcal{N}_{\rm LR}) \leq \sqrt{2r}$ (Negahban and Wainwright, 2011).

In the following subsection, we establish error bounds for $\hat{\boldsymbol{\beta}}_g$, the solution to (2) for decomposable g. These bounds allow both p and q to grow with the sample size n. In Section 3.3, we then specialize our result to the case that the errors are multivariate normal.

3.2 Pivotal Estimation

Throughout the remainder of this section, we treat $\boldsymbol{X}$ as nonrandom. For ease of display, we let $\check{c} = (c+1)/(c-1)$ and $\tilde{c} = c(c+1)/(c-1)$ for constant $c > 1$. We will require the following condition and assumptions.

C1. The columns of $\boldsymbol{X}$ are scaled so that $\|\boldsymbol{X}_{\cdot,j}\|_2 = \sqrt{n}$ for $j \in [p]$.

A1. The $n \times q$ error matrix $\boldsymbol{\mathcal{E}} = (\boldsymbol{\epsilon}_1, \dots, \boldsymbol{\epsilon}_n)^\top$ has q nonzero singular values almost surely.

A2. The distribution of the error matrix $\boldsymbol{\mathcal{E}}$ is left-spherical, i.e., for any $n \times n$ orthogonal matrix $\boldsymbol{O}$, $\boldsymbol{O}\boldsymbol{\mathcal{E}}$ has the same matrix-variate distribution as $\boldsymbol{\mathcal{E}}$.

Assumption **A1** requires that the sample size n is at least as large as the number of responses q. Given $n \geq q$, assumptions **A1** and **A2** would hold if, for example, the rows of $\boldsymbol{\mathcal{E}}$ were independent and each row followed a mean zero multivariate normal distribution with covariance $\boldsymbol{\Sigma}_* \in \mathbb{S}_+^q$. Condition **C1** is simply a matter of rescaling the columns of $\boldsymbol{X}$.

In addition to Assumptions **A1** and **A2**, our bounds will depend on the quantity

$$\phi_{\mathcal{E},g}(\mathcal{M}, \mathcal{N}, c) = \inf_{\boldsymbol{\Delta} \in \mathcal{C}_g(\mathcal{M},\mathcal{N},c)} \left\{ \frac{\sup_{\|\boldsymbol{Q}\| \leq 1} \text{tr}\left\{(\boldsymbol{Q} - \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top)^\top (\boldsymbol{\mathcal{E}} - \boldsymbol{X}\Delta)\right\}}{\sqrt{n}\|\boldsymbol{\Delta}\|_F^2} \right\},$$

$$\mathcal{C}_g(\mathcal{M}, \mathcal{N}, c) = \{\boldsymbol{\Delta} \in \mathbb{R}^{p\times q} \colon \boldsymbol{\Delta} \neq 0,\ g(\boldsymbol{\Delta}_{\mathcal{N}^\perp}) \leq \check{c} g(\boldsymbol{\Delta}_{\mathcal{N}})\}, \quad (\boldsymbol{U}_\epsilon, \boldsymbol{D}_\epsilon, \boldsymbol{V}_\epsilon) = \text{svd}(\boldsymbol{\mathcal{E}}),$$

where $\boldsymbol{\Delta}_{\mathcal{N}}$ denotes the projection of $\boldsymbol{\Delta}$ onto $\mathcal{N}$, i.e., $\boldsymbol{\Delta}_{\mathcal{N}} = \arg\min_{\boldsymbol{M}\in\mathcal{N}} \|\boldsymbol{\Delta} - \boldsymbol{M}\|_F^2$. Using the dual characterization of the nuclear norm, it is immediate that the $\boldsymbol{Q}$ which maximizes the numerator of $\phi_{\mathcal{E},g}(\mathcal{M},\mathcal{N},c)$ is $\boldsymbol{Q} = \tilde{\boldsymbol{U}}\tilde{\boldsymbol{V}}^\top$, where $(\tilde{\boldsymbol{U}}, \tilde{\boldsymbol{D}}, \tilde{\boldsymbol{V}}) = \mathrm{svd}(\boldsymbol{\mathcal{E}} - \boldsymbol{X}\boldsymbol{\Delta})$.

The quantity $\phi_{\mathcal{E},g}(\mathcal{M},\mathcal{N},c)$ is needed to ensure the restricted strong convexity (Negahban et al., 2012) of the nuclear norm of residuals. For this, we have another assumption.

A3. There exists a constant v such that $\phi_{\mathcal{E},g}(\mathcal{M},\mathcal{N},c) \geq v > 0$ almost surely.

The quantity $\phi_{\mathcal{E},g}(\mathcal{M},\mathcal{N},c)$ is closely related to the restricted eigenvalue of $\boldsymbol{X}$ (Raskutti et al., 2010), but also depends on $\boldsymbol{\mathcal{E}}$. As we will show in the next section, under some additional assumptions on the error matrix $\boldsymbol{\mathcal{E}}$ and the matrix $\boldsymbol{X}$, $\phi_{\mathcal{E},g}(\mathcal{M},\mathcal{N},c)$ can be replaced with a restricted eigenvalue-type quantity which does not depend on $\boldsymbol{\mathcal{E}}$.

We are now ready to state our first error bound. The proof of this and all subsequent results can be found in Appendix B.

Theorem 3 *For any fixed constant $c > 1$, define the event $\mathcal{A}_c = \{\lambda \geq (c/\sqrt{n})\tilde{g}(\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top)\}$. If **C1**, **A1**, and **A3** hold, as long as g is decomposable with respect to the subspace pair $(\mathcal{M}, \mathcal{N}^\perp)$, then*

$$\|\hat{\boldsymbol{\beta}}_g - \boldsymbol{\beta}_*\|_F \leq \frac{\check{c}\Psi_g(\mathcal{N})\,\lambda}{\phi_{\mathcal{E},g}(\mathcal{M},\mathcal{N},c)}$$

*with probability at least $P(\mathcal{A}_c)$. If **A2** also holds, then the distribution of $\boldsymbol{X}^\top\boldsymbol{U}_\epsilon\boldsymbol{V}_\epsilon^\top$ does not depend on $\boldsymbol{\Sigma}_*$, i.e., $(c/\sqrt{n})\tilde{g}(\boldsymbol{X}^\top\boldsymbol{U}_\epsilon\boldsymbol{V}_\epsilon^\top)$ is pivotal with respect to the unknown error covariance.*

Theorem 3 reveals that optimal value of the tuning parameter λ depends on the random quantity $\tilde{g}(\boldsymbol{X}^\top\boldsymbol{U}_\epsilon\boldsymbol{V}_\epsilon^\top)$. Under **A1** and **A2**, $\boldsymbol{U}_\epsilon\boldsymbol{V}_\epsilon^\top$ is a random matrix uniformly distributed on the set of matrices $V_q(n) = \{\boldsymbol{S} \in \mathbb{R}^{n\times q} : \boldsymbol{S}^\top\boldsymbol{S} = \boldsymbol{I}_q\}$ (Eaton, 1989; Meckes, 2019) regardless of $\boldsymbol{\Sigma}_*$. This result suggests that the tuning parameter λ could be selected according to the quantiles of $\tilde{g}(\boldsymbol{X}^\top\boldsymbol{S})$ where $\boldsymbol{S}$ is uniformly distributed on $V_q(n)$. For example, the result of Theorem 3 would hold with probability $1-\alpha$ if we selected λ equal to the $(1-\alpha)$th quantile of $(c/\sqrt{n})\tilde{g}(\boldsymbol{X}^\top\boldsymbol{S})$. Fortunately, we can easily sample from the distribution of $(c/\sqrt{n})\tilde{g}(\boldsymbol{X}^\top\boldsymbol{S})$, so we can approximate its quantiles using Monte Carlo. We study this tuning approach empirically in Section 5.

We can use the distribution of $\boldsymbol{X}^\top\boldsymbol{U}_\epsilon\boldsymbol{V}_\epsilon^\top$ to establish explicit choices of λ which yield more insightful error bounds under the three penalties discussed in Section 1.

Corollary 4 *Suppose **C1** and **A1**–**A3** hold.*

*(i) Under **M1**, if $\lambda = c\{2\log(2pq^k)/(n-1)\}^{1/2}$ and $n > 2\log(2pq^k)+1$ for fixed constants $c > 1$ and $k > 1$, then with probability at least $1 - q^{1-k}$,*

$$\|\hat{\boldsymbol{\beta}}_{\mathrm{L}} - \boldsymbol{\beta}_*\|_F \leq \frac{\tilde{c}}{\phi_{\mathcal{E},\|\cdot\|_1}(\mathcal{M}_{\mathrm{L}},\mathcal{N}_{\mathrm{L}},c)}\sqrt{\frac{2|\mathcal{S}|\log(2pq^k)}{n-1}}.$$

*(ii) Under **M2**, if $\lambda = c\{4k\log p/(n-2)\}^{1/2} + c(q/n)^{1/2}$ and $k\log p > 4\pi$ for fixed constants $c > 1$ and $k > 1$, then with probability at least $1 - p^{1-k}$,*

$$\|\hat{\boldsymbol{\beta}}_{\mathrm{GL}} - \boldsymbol{\beta}_*\|_F \leq \frac{2\tilde{c}}{\phi_{\mathcal{E},\|\cdot\|_{1,2}}(\mathcal{M}_{\mathrm{GL}},\mathcal{N}_{\mathrm{GL}},c)}\left(\sqrt{\frac{k|\mathcal{G}|\log p}{n-2}} + \sqrt{\frac{|\mathcal{G}|q}{4n}}\right).$$

(iii) Under $\boldsymbol{M3}$*, if* $\lambda = 4c\|\boldsymbol{X}\|[k_2(p+q)/\{n(n-2)\}]^{1/2} + 4c\|\boldsymbol{X}\|/n$ *and* $k_2\|\boldsymbol{X}\|^2(p+q) > 16n\pi$ *for fixed constants* $c > 1$*,* $k_1 > 1$*, and* $k_2 = 4\log(7+k_1)$*, then with probability at least* $1 - \{8/(7+k_1)\}^{p+q}$*,*

$$\|\hat{\boldsymbol{\beta}}_{\rm LR} - \boldsymbol{\beta}_*\|_F \leq \frac{4\tilde{c}}{\phi_{\mathcal{E},\|\cdot\|_*}(\mathcal{M}_{\rm LR}, \mathcal{N}_{\rm LR}, c)} \left(\frac{\|\boldsymbol{X}\|}{\sqrt{n}}\right)\left(\sqrt{\frac{2k_2 r(p+q)}{n-2}} + \sqrt{\frac{2r}{n}}\right).$$

Corollary 4 demonstrates that we can set λ equal to explicit quantities which will satisfy the condition of Theorem 3 with high probability and do not depend on any unknown parameters.

Before concluding this section, we emphasize that assumptions **A1** and **A2** are not assumptions on the residual matrix $\hat{\boldsymbol{\mathcal{E}}}_g = \boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_g$, but on the error matrix $\boldsymbol{\mathcal{E}}$. After a version of this article had appeared on arXiv, Massias et al. (2020) derived bounds for $\|\hat{\boldsymbol{\beta}}_{\rm GL} - \boldsymbol{\beta}_*\|_{\infty,2}$. However, they required the assumption that $\hat{\boldsymbol{\mathcal{E}}}_g$ was rank q. Of course, $\hat{\boldsymbol{\mathcal{E}}}_g$ depends on both the random error matrix $\boldsymbol{\mathcal{E}}$ and the estimator $\hat{\boldsymbol{\beta}}_g$ itself, so it not clear when their required choice of λ would lead to a violation of this assumption.

3.3 Asymptotics with Normal Errors

While Theorem 3 and Corollary 4 verify that λ can be chosen according to the quantiles of a pivotal quantity, we have not made any particular distributional assumptions on $\boldsymbol{\mathcal{E}}$, which $\phi_{\mathcal{E},g}(\mathcal{M}, \mathcal{N}, c)$—itself a random quantity—depends upon. In this section, we establish asymptotic error bounds for (2) under normality assumptions on $\boldsymbol{\mathcal{E}}$. To do so, we drop assumptions **A1** and **A2**, and add restricted eigenvalue-type conditions on the matrix $\boldsymbol{X}$ to replace **A3**. The assumptions we require are as follows.

A4. The rows of $\boldsymbol{\mathcal{E}}$ are independent and identically distributed from $\mathrm{N}_q(0, \boldsymbol{\Sigma}_*)$. Moreover, there exists a constant v such that

$$0 < v^{-1} \leq \varphi_q(\boldsymbol{\Sigma}_*) \leq \varphi_1(\boldsymbol{\Sigma}_*) \leq v < \infty.$$

A5. There exists a constant $\bar{v}$ such that

$$\sup_{\boldsymbol{\Delta} \in \mathcal{C}_g(\mathcal{M}, \mathcal{N}, c)} \frac{\|\boldsymbol{X}\boldsymbol{\Delta}\|_F^2}{n\|\boldsymbol{\Delta}\|_F^2} \leq \bar{v} < \infty.$$

In addition, there exists a constant $\underline{v}$ such that $0 < \underline{v} \leq \nu_g(\mathcal{M}, \mathcal{N}, c)$ where

$$\nu_g(\mathcal{M}, \mathcal{N}, c) = \inf_{\substack{\boldsymbol{\Delta} \in \mathcal{C}_g(\mathcal{M}, \mathcal{N}, c) \\ \boldsymbol{S} \in V_q(n)}} \left\{ \frac{\sum_{i=1}^q \sum_{j=1}^q (\boldsymbol{u}_j^\top \boldsymbol{X}\boldsymbol{\Delta}\boldsymbol{v}_i - \boldsymbol{u}_i^\top \boldsymbol{X}\boldsymbol{\Delta}\boldsymbol{v}_j)^2 + 4\sum_{k=q+1}^n \sum_{j=1}^q (\boldsymbol{u}_k^\top \boldsymbol{X}\boldsymbol{\Delta}\boldsymbol{v}_j)^2}{4n\|\boldsymbol{\Delta}\|_F^2} \right\}$$

with $(\boldsymbol{U}, \boldsymbol{I}_q, \boldsymbol{V}) = \mathrm{svd}(\boldsymbol{S})$, where $\boldsymbol{u}_j$ denotes the jth column of $\boldsymbol{U} \in \mathbb{R}^{n \times q}$ for $j \in [q]$, $\boldsymbol{v}_l$ denotes the lth column of $\boldsymbol{V} \in \mathbb{R}^{q \times q}$ for $l \in [q]$, and $\boldsymbol{u}_k$ denotes the $(k-q)$th column of $\boldsymbol{U}_0 \in \mathbb{R}^{n \times (n-q)}$ where $\boldsymbol{U}_0^\top \boldsymbol{U} = 0$ and $\boldsymbol{U}_0^\top \boldsymbol{U}_0 = \boldsymbol{I}_{n-q}$ for $k \in \{q+1, q+2, \ldots, n\}$.

A6. As $n \to \infty$, $\sqrt{q/n} \to t$ for some $t \in [0, 1)$.

Assumption **A4** is standard in the multivariate response linear regression and precision matrix estimation literature. Assumptions **A4** and **A6** together would imply assumptions **A1** and **A2** asymptotically. Assumption **A5** consists of restricted eigenvalue-type conditions. The latter assumption made in **A5**, while tailored specifically to apply to our problem, can be seen as analogous to the standard restricted eigenvalue condition in penalized least squares (Raskutti et al., 2010). For example, we can write the spectral norm of $\boldsymbol{X\Delta}$ in variational form as $\|\boldsymbol{X\Delta}\| = \sup_{\boldsymbol{u} \in S^{n-1}} \sup_{\boldsymbol{v} \in S^{q-1}} \boldsymbol{u}^\top \boldsymbol{X\Delta v}$, where $S^{n-1} = \{\boldsymbol{u} \in \mathbb{R}^n : \|\boldsymbol{u}\|_2 = 1\}$. Both parts of **A5** are needed to establish the restricted strong convexity of the nuclear norm of residuals.

With **A4**–**A6**, we are now ready to state a version of Theorem 3 which applies to normally distributed error matrix $\boldsymbol{\mathcal{E}}$.

Theorem 5 *For fixed constants $c > 1$ and $d > 1$, define the events $\mathcal{A}_c = \{\lambda \geq (c/\sqrt{n})\tilde{g}(\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top)\}$ and $\mathcal{B}_d = \{\sigma_1^2(\boldsymbol{\mathcal{E}})/n \geq (t + d)\varphi_1(\boldsymbol{\Sigma}_*)\}$. If **C1** and **A4**–**A6** hold, g is decomposable with respect to the subspace pair $(\mathcal{M}, \mathcal{N}^\perp)$, and $\Psi_g(\mathcal{N})\lambda \to 0$ as $n \to \infty$, then*

$$\|\hat{\boldsymbol{\beta}}_g - \boldsymbol{\beta}_*\|_F \lesssim \frac{(t + d)\,\check{c}\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)\Psi_g(\mathcal{N})\lambda}{\nu_g(\mathcal{M}, \mathcal{N}, c)}$$

with probability at least $P(\mathcal{A}_c \cap \mathcal{B}_d)$ for n sufficiently large.

In Theorem 5, we have essentially replaced the random quantity $\phi_{\mathcal{E},g}(\mathcal{M}, \mathcal{N}, c)$ from Theorem 3 with a constant times $\nu_g(\mathcal{M}, \mathcal{N}, c)/\{(t + d){\varphi_1}^{1/2}(\boldsymbol{\Sigma}_*)\}$, which is nonrandom.

Applying the same concentration inequalities used to obtain the bounds in Corollary 4—along with a concentration inequality on the largest singular value of the matrix $\boldsymbol{\mathcal{E}}$—we arrive at the following set of asymptotic results concerning (2) with penalties (3), (4), and (5).

Corollary 6 *Suppose **C1** and **A4**–**A6** hold.*

*(i) Under **M1**, if $\lambda = c\{2\log(2pq^k)/(n-1)\}^{1/2}$, $n > 2\log(2pq^k) + 1$, and $|\mathcal{S}|\log(pq^k) = o(n)$ for fixed constants $c > 1$, $k > 1$, and $d > 1$, then*

$$\|\hat{\boldsymbol{\beta}}_{\mathrm{L}} - \boldsymbol{\beta}_*\|_F \lesssim \frac{(t + d)\,\tilde{c}\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)}{\nu_{\|\cdot\|_1}(\mathcal{M}_{\mathrm{L}}, \mathcal{N}_{\mathrm{L}}, c)}\sqrt{\frac{|\mathcal{S}|\log(2pq^k)}{n-1}} \tag{9}$$

with probability at least $1 - q^{1-k} - 2e^{-(d-1)^2 n/4}$ for n sufficiently large.

*(ii) Under **M2**, if $\lambda = c\{4k\log p/(n-2)\}^{1/2} + c(q/n)^{1/2}$, $k \log p > 4\pi$, and $|\mathcal{G}|\max(\log p, q) = o(n)$ for fixed constants $c > 1$, $k > 1$, and $d > 1$, then*

$$\|\hat{\boldsymbol{\beta}}_{\mathrm{GL}} - \boldsymbol{\beta}_*\|_F \lesssim \frac{(t + d)\,\tilde{c}\,\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)}{\nu_{\|\cdot\|_{1,2}}(\mathcal{M}_{\mathrm{GL}}, \mathcal{N}_{\mathrm{GL}}, c)}\left(\sqrt{\frac{k|\mathcal{G}|\log p}{n-2}} + \sqrt{\frac{|\mathcal{G}|q}{4n}}\right) \tag{10}$$

with probability at least $1 - p^{1-k} - 2e^{-(d-1)^2 n/4}$ for n sufficiently large.

(iii) Under ***M3****, if* $\lambda = 4c\|\boldsymbol{X}\|[k_2(p+q)/\{n(n-2)\}]^{1/2} + 4c\|\boldsymbol{X}\|/n$*,* $k_2\|\boldsymbol{X}\|^2(p+q) > 16n\pi$*, and* $\|\boldsymbol{X}\|^2 r(p+q) = o(n^2)$ *for fixed constants* $c > 1$*,* $k_1 > 1$*,* $k_2 = 4\log(7+k_1)$*, and* $d > 1$*, then*

$$\|\hat{\boldsymbol{\beta}}_{\rm LR} - \boldsymbol{\beta}_*\|_F \lesssim \frac{(t+d)\tilde{c}\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)}{\nu_{\|\cdot\|_*}(\mathcal{M}_{\rm LR}, \mathcal{N}_{\rm LR}, c)}\left(\frac{\|\boldsymbol{X}\|}{\sqrt{n}}\right)\left(\sqrt{\frac{k_2 r(p+q)}{n-2}} + \sqrt{\frac{r}{n}}\right) \tag{11}$$

with probability at least $1 - \{8/(7+k_1)\}^{p+q} - 2e^{-(d-1)^2 n/4}$ *for* n *sufficiently large.*

The error bounds in Corollary 6 agree with those in the existing literature on penalized least squares estimators. For example, the bound in (9) is asymptotically equivalent to the bound of Price and Sherwood (2017), who studied a version of the L_1-penalized least squares estimator. Similarly, the bound in (10) coheres with the bound for the group lasso-penalized least squares estimator from Lounici et al. (2011). Finally, our bounds for the nuclear norm penalized version of (2) asymptotically agree with their least squares analog from Negahban and Wainwright (2011).

In Section 7, we discuss the challenges in establishing the conditions for (2) to consistently estimate the support of $\boldsymbol{\beta}_*$ under **M1** or **M2**, or the true rank of $\boldsymbol{\beta}_*$ under **M3**. In brief, because the nuclear norm of residuals is nondifferentiable in general, applying standard proof techniques (e.g., see the proof of Theorem 3.4 of Lee et al. (2015)) is difficult.

To conclude this section, we discuss a potential limitation of our theory. Assumptions **A1** and **A6** require that the number of subjects, n, is at least as large as the number of responses, q. However, we emphasize that (2) can still be applied and perform well in finite sample settings where $q > n$, as we show in Section A.2 of Appendix A. We discuss possible ways to relax this requirement in Section 7.

4. Computation

4.1 Properties of the Solution

In the low-dimensional setting, the minimizer of the unpenalized nuclear norm of residuals is equivalent to the minimizer of the unpenalized squared Frobenius norm of residuals. That is, the least squares estimator $(\boldsymbol{X}^\top\boldsymbol{X})^{-1}\boldsymbol{X}^\top\boldsymbol{Y}$, when it exists, is a minimizer of (2) when $\lambda = 0$. Of course, the penalized solution to (2) does not coincide with the penalized least squares estimator. This can be seen by examining the first order conditions for (2) which we characterize in the following remark.

Remark 7 *When* $\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_g$ *has* q *nonzero singular values, the first order conditions for* (2)*, which are necessary and sufficient for optimality, are*

$$\frac{1}{\sqrt{n}}\boldsymbol{X}^\top(\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_g)[(\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_g)^\top(\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_g)]^{-1/2} \in \lambda\partial g(\hat{\boldsymbol{\beta}}_g) \tag{12}$$

where $\partial g(\hat{\boldsymbol{\beta}}_g)$ *is the subdifferential of* g *at* $\hat{\boldsymbol{\beta}}_g$*. If* $\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_g$ *has fewer than* q *nonzero singular values, the first order conditions for* (2) *are*

$$\frac{1}{\sqrt{n}}\boldsymbol{X}^\top(\boldsymbol{U}_{\hat{\epsilon}}\boldsymbol{V}_{\hat{\epsilon}}^\top + \boldsymbol{Z}_1) = \lambda\boldsymbol{Z}_2,$$

for some $\boldsymbol{Z}_2 \in \partial g(\hat{\boldsymbol{\beta}}_g)$ *and* $\boldsymbol{Z}_1 \in \{\boldsymbol{Z} \in \mathbb{R}^{n \times q} : \|\boldsymbol{Z}\| \leq 1, \boldsymbol{U}_{\hat{\epsilon}}^\top \boldsymbol{Z} = 0, \boldsymbol{Z}\boldsymbol{V}_{\hat{\epsilon}} = 0, (\boldsymbol{U}_{\hat{\epsilon}}, \boldsymbol{D}_{\hat{\epsilon}}, \boldsymbol{V}_{\hat{\epsilon}}) = \mathrm{svd}(\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_g)\}$.

The residual matrix $\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_g$ can only have q nonzero singular values when $n > q$ and when λ is sufficiently large (see Section 4.3). In these cases, we could use (12) as a termination criterion.

4.2 Prox-Linear ADMM Algorithm

To compute (2), we must address that the nuclear norm of residuals is nondifferentiable in general. To do so, we employ a variation of the alternating direction method of multipliers (ADMM) algorithm which decouples the nuclear norm of residuals and penalty g (Boyd et al., 2011). Throughout this and the subsequent section, we will refer to a proximal operator of a function f, defined as

$$\mathrm{Prox}_f(\boldsymbol{B}) = \arg\min_{\boldsymbol{A}} \left\{ \frac{1}{2}\|\boldsymbol{A} - \boldsymbol{B}\|_F^2 + f(\boldsymbol{A}) \right\}.$$

When f is a proper and lower semi-continuous convex function, its proximal operator is unique (Parikh and Boyd, 2014b; Polson et al., 2015). The proximal operators corresponding to (3), (4), and (5) all have closed forms and can be computed efficiently (see Table 1 of the Supplementary Materials to Molstad et al. (2021b)).

To apply the ADMM algorithm, following Boyd et al. (2011), we first introduce an additional variable $\boldsymbol{\Phi} \in \mathbb{R}^{n \times q}$ so that we may rewrite the problem in (2) as the constrained optimization problem

$$\operatorname*{minimize}_{\boldsymbol{\beta} \in \mathbb{R}^{p \times q}, \boldsymbol{\Phi} \in \mathbb{R}^{n \times q}} \left\{ \|\boldsymbol{\Phi}\|_* + \tilde{\lambda} g(\boldsymbol{\beta}) \right\} \quad \text{subject to} \quad \boldsymbol{\Phi} = \boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}, \tag{13}$$

where $\tilde{\lambda} = \sqrt{n}\lambda$. Then, we define the augmented Lagrangian for the constrained problem in (13) as

$$\mathcal{F}_\rho(\boldsymbol{\beta}, \boldsymbol{\Phi}, \boldsymbol{\Gamma}) = \|\boldsymbol{\Phi}\|_* + \tilde{\lambda} g(\boldsymbol{\beta}) + \mathrm{tr}\{\boldsymbol{\Gamma}^\top(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{\Phi})\} + \frac{\rho}{2}\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{\Phi}\|_F^2,$$

where $\rho > 0$ is fixed and $\boldsymbol{\Gamma} \in \mathbb{R}^{n \times q}$ is the Lagrangian dual variable. The updating equations for the $(k+1)$th iterate of the standard ADMM algorithm are

$$\boldsymbol{\beta}^{(k+1)} = \arg\min_{\boldsymbol{\beta} \in \mathbb{R}^{p \times q}} \mathcal{F}_\rho(\boldsymbol{\beta}, \boldsymbol{\Phi}^{(k)}, \boldsymbol{\Gamma}^{(k)}) \tag{14}$$

$$\boldsymbol{\Phi}^{(k+1)} = \arg\min_{\boldsymbol{\Phi} \in \mathbb{R}^{n \times q}} \mathcal{F}_\rho(\boldsymbol{\beta}^{(k+1)}, \boldsymbol{\Phi}, \boldsymbol{\Gamma}^{(k)}) \tag{15}$$

$$\boldsymbol{\Gamma}^{(k+1)} = \boldsymbol{\Gamma}^{(k)} + \tau\rho(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}^{(k+1)} - \boldsymbol{\Phi}^{(k+1)}), \tag{16}$$

where $\tau > 0$ modifies the step size for the dual variable update. The $\boldsymbol{\Phi}$ updating equation of the ADMM algorithm, (15), can be expressed in terms of the proximal operator of the nuclear norm

$$\boldsymbol{\Phi}^{(k+1)} = \mathrm{Prox}_{\rho^{-1}\|\cdot\|_*}\left(\boldsymbol{Y} + \rho^{-1}\boldsymbol{\Gamma}^{(k)} - \boldsymbol{X}\boldsymbol{\beta}^{(k+1)}\right),$$

which can be solved efficiently in closed form by computing the singular value decomposition of $\boldsymbol{Y} + \rho^{-1}\boldsymbol{\Gamma}^{(k)} - \boldsymbol{X}\boldsymbol{\beta}^{(k+1)}$ and soft thresholding its singular values (e.g., see 3 and 4 of Algorithm 1).

When p is large, the first step of the ADMM algorithm, (14), is more computationally burdensome since it requires solving the penalized least squares optimization problem

$$\operatorname*{arg\,min}_{\boldsymbol{\beta}\in\mathbb{R}^{p\times q}} \mathcal{F}_\rho(\boldsymbol{\beta}, \boldsymbol{\Phi}^{(k)}, \boldsymbol{\Gamma}^{(k)}) = \operatorname*{arg\,min}_{\boldsymbol{\beta}\in\mathbb{R}^{p\times q}} \left\{ \frac{1}{2}\|\boldsymbol{Y} + \rho^{-1}\boldsymbol{\Gamma}^{(k)} - \boldsymbol{\Phi}^{(k)} - \boldsymbol{X}\boldsymbol{\beta}\|_F^2 + \frac{\tilde{\lambda}}{\rho} g(\boldsymbol{\beta}) \right\}. \quad (17)$$

To avoid solving (17) at every iteration, we instead approximate (14) by minimizing a majorizing function of $\mathcal{F}_\rho(\boldsymbol{\beta}, \boldsymbol{\Phi}^{(k+1)}, \boldsymbol{\Gamma}^{(k)})$ constructed at the previous iterate $\boldsymbol{\beta}^{(k)}$. Specifically, we majorize $\mathcal{F}_\rho(\boldsymbol{\beta}, \boldsymbol{\Phi}^{(k)}, \boldsymbol{\Gamma}^{(k)})$ in (14) with

$$\mathcal{M}_{\rho,\eta}(\boldsymbol{\beta}, \boldsymbol{\Phi}^{(k)}, \boldsymbol{\Gamma}^{(k)}; \boldsymbol{\beta}^{(k)}) = \mathcal{F}_\rho(\boldsymbol{\beta}, \boldsymbol{\Phi}^{(k)}, \boldsymbol{\Gamma}^{(k)}) + \frac{\rho}{2}\operatorname{tr}\big\{(\boldsymbol{\beta} - \boldsymbol{\beta}^{(k)})^\top \boldsymbol{Q}_\eta (\boldsymbol{\beta} - \boldsymbol{\beta}^{(k)})\big\},$$

where $\boldsymbol{Q}_\eta = \eta \boldsymbol{I}_p - \boldsymbol{X}^\top\boldsymbol{X}$ with $\eta > 0$ fixed and chosen so that $\boldsymbol{Q}_\eta$ is nonnegative definite. Thus, we replace (14) with

$$\begin{aligned}\boldsymbol{\beta}^{(k+1)} &= \operatorname*{arg\,min}_{\boldsymbol{\beta}\in\mathbb{R}^{p\times q}} \mathcal{M}_{\rho,\eta}(\boldsymbol{\beta}, \boldsymbol{\Phi}^{(k)}, \boldsymbol{\Gamma}^{(k)}; \boldsymbol{\beta}^{(k)}) \\ &= \operatorname{Prox}_{(\rho\eta)^{-1}\tilde{\lambda} g}\left\{ \boldsymbol{\beta}^{(k)} + \eta^{-1}\boldsymbol{X}^\top \big(\boldsymbol{Y} + \rho^{-1}\boldsymbol{\Gamma}^{(k)} - \boldsymbol{\Phi}^{(k)} - \boldsymbol{X}\boldsymbol{\beta}^{(k)}\big)\right\}. \quad (18)\end{aligned}$$

It follows that using (18), $\mathcal{F}_\rho(\boldsymbol{\beta}^{(k+1)}, \boldsymbol{\Phi}^{(k)}, \boldsymbol{\Gamma}^{(k)}) \leq \mathcal{F}_\rho(\boldsymbol{\beta}^{(k)}, \boldsymbol{\Phi}^{(k)}, \boldsymbol{\Gamma}^{(k)})$ by the majorize-minimize principle (Lange, 2016). This approximation can improve efficiency because for many g, (18) can be computed efficiently in closed form. For example, in the case that g is the L_1-norm, (18) can be solved by soft thresholding $\boldsymbol{\beta}^{(k)} + \eta^{-1}\boldsymbol{X}^\top(\boldsymbol{Y} + \rho^{-1}\boldsymbol{\Gamma}^{(k)} - \boldsymbol{\Phi}^{(k)} - \boldsymbol{X}\boldsymbol{\beta}^{(k)})$.

The complete prox-linear ADMM algorithm we implement is stated formally in Algorithm 1. In the algorithm statement, we use $(\cdot)_+$ to denote the elementwise positive part function, i.e., $(\boldsymbol{A}_{j,k})_+ = \max(\boldsymbol{A}_{j,k}, 0)$. This variation of the ADMM algorithm—which replaces the objective function in (14) with a quadratic majorization constructed at the previous iterate—was studied by Deng and Yin (2016), who called it the prox-linear ADMM algorithm. Fortunately, we can show that the iterates of our prox-linear ADMM algorithm converge to their optimal values.

Lemma 8 *Suppose $0 < 2\tau < 1 + \sqrt{5}$, $\rho > 0$, and $\eta \geq \|\boldsymbol{X}^\top\boldsymbol{X}\|$ are fixed. Then, as $k \to \infty$, the sequence of iterates $(\boldsymbol{\Phi}^{(k)}, \boldsymbol{\beta}^{(k)}, \boldsymbol{\Gamma}^{(k)})$ generated from Algorithm 1 converge to $(\boldsymbol{\Phi}^\star, \boldsymbol{\beta}^\star, \boldsymbol{\Gamma}^\star)$, where $(\boldsymbol{\Phi}^\star, \boldsymbol{\beta}^\star)$ are optimal solutions to (13) and $\boldsymbol{\Gamma}^\star$ is an optimal solution to the dual of (13). In addition, if $\tau = 1$, then the sequence $\{\theta_k, k = 0, 1, 2, \dots\}$ defined by $\theta_k = \rho\|\boldsymbol{\beta}^{(k)} - \boldsymbol{\beta}^\star\|_{\boldsymbol{Q}_\eta}^2 + \rho\|\boldsymbol{\Phi}^{(k)} - \boldsymbol{\Phi}^\star\|_F^2 + \rho^{-1}\|\boldsymbol{\Gamma}^{(k)} - \boldsymbol{\Gamma}^\star\|_F^2$ is nonincreasing and $\theta_k = O(k^{-1})$ as $k \to \infty$.*

The arguments used to prove Lemma 8 are essentially identical to those from Gu et al. (2018), who proposed a prox-linear ADMM algorithm to compute a penalized (univariate response) quantile regression estimator. In our implementation, we found that $\tau = 1$ generally worked well, although setting τ closer to $(1 + \sqrt{5})/2$ could lead to faster convergence in

Algorithm 1: Prox-linear ADMM algorithm for (2)

```
1. Given $\rho > 0$, $\eta \geq \|\boldsymbol{X}^\top \boldsymbol{X}\|$, $\tilde{\lambda} = \sqrt{n}\lambda$, $\tau \in (0, \frac{1+\sqrt{5}}{2})$, and $s = \min(p, q)$, initialize
   $(\boldsymbol{\beta}^{(0)}, \boldsymbol{\Phi}^{(0)}, \boldsymbol{\Gamma}^{(0)}) \in \mathbb{R}^{p \times q} \times \mathbb{R}^{n \times q} \times \mathbb{R}^{n \times q}$ and set $k = 0$
2. $\boldsymbol{\beta}^{(k+1)} \leftarrow \mathrm{Prox}_{(\rho\eta)^{-1}\tilde{\lambda}g}\{\boldsymbol{\beta}^{(k)} + \eta^{-1}\boldsymbol{X}^\top\left(\boldsymbol{Y} + \rho^{-1}\boldsymbol{\Gamma}^{(k)} - \boldsymbol{\Phi}^{(k)} - \boldsymbol{X}\boldsymbol{\beta}^{(k)}\right)\}$
3. $(\boldsymbol{U}, \boldsymbol{D}, \boldsymbol{V}) \leftarrow \mathrm{svd}(\boldsymbol{Y} + \rho^{-1}\boldsymbol{\Gamma}^{(k)} - \boldsymbol{X}\boldsymbol{\beta}^{(k+1)})$
3. $\boldsymbol{\Phi}^{(k+1)} \leftarrow \boldsymbol{U}(\boldsymbol{D} - \rho^{-1}\boldsymbol{I}_s)_+\boldsymbol{V}^\top$
4. $\boldsymbol{\Gamma}^{(k+1)} \leftarrow \boldsymbol{\Gamma}^{(k)} + \tau\rho(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}^{(k+1)} - \boldsymbol{\Phi}^{(k+1)})$
5. If not converged, set $k \leftarrow k + 1$ and return to 2
```

certain scenarios. Similarly, we set $\eta = (1 + 10^{-5})\|\boldsymbol{X}^\top\boldsymbol{X}\|$, which we found was fastest among the values we considered.

The convergence criteria we use are based on the primal and dual residuals suggested by Boyd et al. (2011). At each iteration we compute

$$r^{(k+1)} = \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}^{(k+1)} - \boldsymbol{\Phi}^{(k+1)}\|_F^2, \quad s^{(k+1)} = \rho^2\|\boldsymbol{X}^\top(\boldsymbol{\Phi}^{(k+1)} - \boldsymbol{\Phi}^{(k)})\|_F^2.$$

We also compute $\mathrm{e}_{\mathrm{primal}}^{(k+1)} = \epsilon_{\mathrm{abs}}\sqrt{n} + \epsilon_{\mathrm{rel}}\max\{\|\boldsymbol{X}\boldsymbol{\beta}^{(k+1)}\|_F, \|\boldsymbol{\Phi}^{(k+1)}\|_F, \|\boldsymbol{Y}\|_F\}$ and $\mathrm{e}_{\mathrm{dual}}^{(k+1)} = \epsilon_{\mathrm{abs}}\sqrt{p} + \epsilon_{\mathrm{rel}}\|\boldsymbol{X}^\top\boldsymbol{\Gamma}^{(k+1)}\|_F$ where ϵ_{abs} and ϵ_{rel} are the absolute and relative convergence tolerances, respectively. Then, we terminate Algorithm 1 when $r^{(k+1)} \leq \mathrm{e}_{\mathrm{primal}}^{(k+1)}$ and $s^{(k+1)} \leq \mathrm{e}_{\mathrm{dual}}^{(k+1)}$. Our default implementation sets $\epsilon_{\mathrm{rel}} = 10^{-4}$ and $\epsilon_{\mathrm{abs}} = 10^{-10}$. We also adaptively update the step size ρ. Unlike the scheme originally proposed in Boyd et al. (2011), we update ρ every κth iteration using $\rho \leftarrow \rho\left\{\mathbf{1}(r^{(k+1)} > 10s^{(k+1)}) - 0.5 \cdot \mathbf{1}(s^{(k+1)} > 10r^{(k+1)}) + 1\right\}$. In our default implementation, we use $\kappa = 10$.

An R package implementing Algorithm 1, Algorithm 2 (see Section 4.3 and Section A.1 of Appendix A), and a number of auxiliary functions are available for download at `https://github.com/ajmolstad/MSRL`.

4.3 Alternative Computational Approaches

Numerous other computational approaches could be applied to solve (2). One class of methods are those that, like the prox-linear ADMM algorithm, are designed to handle optimization problems where the objective function is the sum of two nondifferentiable, convex, and proximable functions. These include, for example, the accelerated primal-dual algorithm of Chambolle and Pock (2011) and the graph projection ADMM algorithm (Parikh and Boyd, 2014a; Fougner and Boyd, 2018).

Another (arguably simpler) class of algorithms can be applied only in special settings. In particular, when $n > q$ and λ is sufficiently large, we can treat the nuclear norm of residual as differentiable. This is because the subdifferential of the nuclear norm of the residual matrix with respect to $\boldsymbol{\beta}$ is the set

$$\Big\{\boldsymbol{W} \in \mathbb{R}^{p\times q} : \boldsymbol{W} = -\boldsymbol{X}^\top(\boldsymbol{U}\boldsymbol{V}^\top + \boldsymbol{Q}), \|\boldsymbol{Q}\| \leq 1, \\ \boldsymbol{U}^\top\boldsymbol{Q} = \boldsymbol{Q}\boldsymbol{V} = 0, (\boldsymbol{U}, \boldsymbol{D}, \boldsymbol{V}) = \mathrm{svd}(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})\Big\},$$

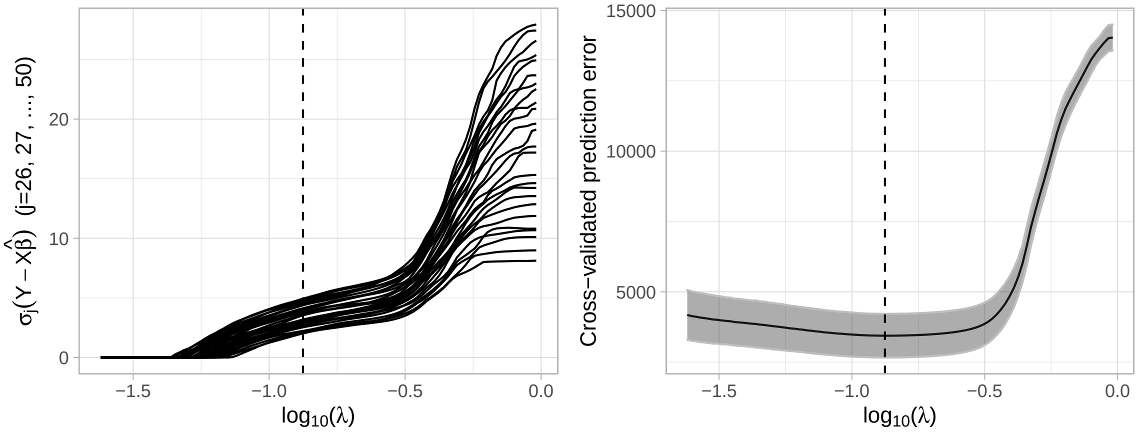

Figure 1: (Left) The solution path for the 25 smallest singular values of $\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_{\mathrm{L}}$ as a function of λ for data generated under Model 1 and **M1** (see Section 5.2 and 5.3) with $n = 200$, $p = 500$, $q = 50$, and normal errors with $\xi = 0.9$. (Right) Average five-fold cross-validation squared prediction error (and standard errors) for $\hat{\boldsymbol{\beta}}_{\mathrm{L}}$ on the same data set. In both panels the vertical dotted line denotes the tuning parameter value minimizing the average cross-validated squared prediction error.

for example, see Watson (1992). Thus, when $\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}$ has q nonzero singular values, the subdifferential of $\boldsymbol{\beta} \mapsto \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_*$ is the singleton

$$-\boldsymbol{X}^\top(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})\{(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})^\top(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})\}^{-1/2} \tag{19}$$

so that $\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_*$ can effectively be treated as differentiable over the set of $\boldsymbol{\beta}$ such that $\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}$ has q nonzero singular values.

This simple fact suggests that in these special settings, we can use first order algorithms to solve (2). To illustrate that this represents a range of interesting fitted models, we generated data from Model 1 of Section 5.2 with $\boldsymbol{\beta}_*$ constructed according to **M1**, g being the L_1-norm penalty, and $(n, p, q) = (200, 500, 50)$. In the left panel of Figure 1, we display the path of the 25 smallest singular values of $\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_{\mathrm{L}}$ as a function of the tuning parameter λ; in the right panel, we display the cross-validated squared prediction errors. We see that for λ sufficiently large, all q singular values of the residual matrix are nonzero. In addition, we see that the cross-validated squared prediction error indicates that the best model fits are those occurring at points on the solution path where $\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_{\mathrm{L}}$ has q nonzero singular values. As λ approaches zero, we see that many singular values of $\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_{\mathrm{L}}$ become zero. This is because the nuclear norm acts like a lasso-type penalty on the singular values of its matrix argument, so reducing λ is analogous to increasing the relative contribution of the nuclear norm of residuals to the overall objective function.

Hence, to solve (2) when $n > q$ and λ is sufficiently large, we consider using an accelerated proximal gradient descent algorithm (Beck and Teboulle, 2009; Combettes and Pesquet, 2011). Letting $\mathcal{D}_{\underline{\kappa}} = \left\{\boldsymbol{\beta} \in \mathbb{R}^{p \times q} : \underline{\kappa} \leq \sigma_q(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}) \leq \sigma_1(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}) \leq \underline{\kappa}^{-1}\right\}$ and letting

$(\boldsymbol{U}_{\epsilon^{(k)}}, \boldsymbol{V}_{\epsilon^{(k)}})$ denote the left and right singular vectors of $\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}^{(k)}$ respectively, it follows from (19) that if we iteratively update $\boldsymbol{\beta}$ from kth to $(k+1)$th iterate using

$$\begin{aligned}\boldsymbol{\beta}^{(k+1)} &= \underset{\boldsymbol{\beta}\in\mathbb{R}^{p\times q}}{\arg\min}\left[\frac{1}{2\rho_k}\|\boldsymbol{\beta}-\boldsymbol{\beta}^{(k)}\|_F^2 - \frac{1}{\sqrt{n}}\operatorname{tr}\left\{\boldsymbol{V}_{\epsilon^{(k)}}\boldsymbol{U}_{\epsilon^{(k)}}^\top\boldsymbol{X}(\boldsymbol{\beta}-\boldsymbol{\beta}^{(k)})\right\} + \lambda g(\boldsymbol{\beta})\right] \\ &= \operatorname{Prox}_{\rho_k\lambda g}\left(\boldsymbol{\beta}^{(k)} + \frac{\rho_k}{\sqrt{n}}\boldsymbol{X}^\top\boldsymbol{U}_{\epsilon^{(k)}}\boldsymbol{V}_{\epsilon^{(k)}}^\top\right)\end{aligned}$$

for step size ρ_k sufficiently small, $\boldsymbol{\beta}^{(k+1)} \to \hat{\boldsymbol{\beta}}_g$ as $k \to \infty$ provided that $\hat{\boldsymbol{\beta}}_g$ and each $\boldsymbol{\beta}^{(k+1)}$ belong to $\mathcal{D}_{\underline{\kappa}}$ for some positive $\underline{\kappa}$ bounded away from zero. A similar computational approach was proposed and studied theoretically in Li et al. (2020) for solving the univariate square-root lasso optimization problem.

In contrast with Algorithm 1, accelerated versions of the proximal gradient descent algorithm are known to converge at a quadratic rate (Beck and Teboulle, 2009), so this approach may be preferred in the settings where it can be applied. Of course, if the solution $\hat{\boldsymbol{\beta}}_g$ leads to residual matrix $\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_g$ with fewer than q nonzero singular values, this algorithm cannot be used. In practice, we use an accelerated proximal gradient descent algorithm to compute $\hat{\boldsymbol{\beta}}_g$ for large values of λ, but when an iterate of this algorithm leads to (nearly) rank deficient residuals, we then revert to using Algorithm 1 for that and all smaller values of λ. For example, in the setting displayed in Figure 1, accelerated proximal gradient descent could be used to compute (2) for all λ such that $\log_{10}(\lambda) > -1$. A formal statement of the accelerated proximal gradient descent algorithm we implement (Algorithm 2), along with details about our implementation, can be found in Section A.1 of Appendix A.

5. Simulation Studies

5.1 Overview

In this section, we compare (2) to alternative methods for fitting the multivariate response linear regression model in high-dimensional settings. We consider three data generating models under **M1**, **M2**, and **M3** as defined in Section 3.1. In addition to comparing methods which use cross-validation for tuning parameter selection, we also consider versions of (2) with tuning parameters chosen according to the theoretical results from Section 3.2.

5.2 Data Generating Models and Competing Methods

In each setting we consider, for one hundred independent replications, we generate $\boldsymbol{X} \in \mathbb{R}^{n\times p}$ to have rows being independent realizations of $\mathrm{N}_p(0, \boldsymbol{\Sigma}_{*\boldsymbol{X}})$ with $[\boldsymbol{\Sigma}_{*\boldsymbol{X}}]_{j,k} = 0.5^{|j-k|}$ for $(j,k) \in [p]\times[p]$. Then, given $\boldsymbol{X}$, we generate $\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta}_* + \boldsymbol{\mathcal{E}}$ where rows of $\boldsymbol{\mathcal{E}} \in \mathbb{R}^{n\times q}$ are independent and identically distributed with mean zero and covariance $\boldsymbol{\Sigma}_* \in \mathbb{S}^q_+$. We consider three distinct data generating models.

Model 1 (Compound symmetry): $\boldsymbol{\Sigma}_* = 3\tilde{\boldsymbol{\Sigma}}_*$, where $\tilde{\boldsymbol{\Sigma}}_{*j,k} = \xi\mathbf{1}(j\neq k) + \mathbf{1}(j = k)$ for $(j,k) \in [q]\times[q]$ and $\mathbf{1}(\cdot)$ is the indicator function.

Model 2 (Varying condition number): $\boldsymbol{\Sigma}_* = 2\tilde{\boldsymbol{\Sigma}}_*$, where $\tilde{\boldsymbol{\Sigma}}_{*j,k} = \boldsymbol{O}\boldsymbol{\Gamma}\boldsymbol{O}^\top$, $\boldsymbol{O}$ is a randomly generated $q\times q$ orthogonal matrix, and $\boldsymbol{\Gamma}$ is diagonal with equally spaced entries from 1 to the inverse condition number.

Model 3 (Factor model): $\boldsymbol{\Sigma}_* = \boldsymbol{R}^\top \boldsymbol{R} + 0.05 \boldsymbol{I}_q$, where $\boldsymbol{R}$ is obtained by first generating $\tilde{\boldsymbol{R}} \in \mathbb{R}^{m \times q}$ with $m \leq q$ to have independent standard normal entries and setting $\boldsymbol{R} = \tilde{\boldsymbol{R}}\boldsymbol{K}$ where $\boldsymbol{K} \in \mathbb{R}^{q \times q}$ is diagonal with entries chosen so that $\boldsymbol{R}^\top \boldsymbol{R}$ has diagonal entries equal to 1.45.

Throughout our simulations, we set $n = 200$, $p = 500$, $q = 50$, and let ξ, the condition number, and the number of factors (m) vary, under Models 1, 2, and 3, respectively. In addition to Models 1–3 with normally distributed errors, we also consider Models 1–3 with errors following a multivariate t-distribution with five degrees of freedom (henceforth, t_5).

To select tuning parameters, we also generate a validation set of size n from the same data generating model. For each method we consider, tuning parameters are chosen to minimize the squared prediction error averaged across all q responses on the validation set. In a slight abuse of terminology, we refer to this as "cross-validation" for the remainder of this section.

We will describe the construction of $\boldsymbol{\beta}_*$ separately in subsequent sections. Given a training data set, we estimate $\boldsymbol{\beta}_*$ using the following methods.

`MSR-CV`: Our proposed estimator from (2).

`Calibrated`: A variation of the calibrated multivariate response linear regression method proposed by Liu et al. (2015):

$$\operatorname*{arg\,min}_{\boldsymbol{\beta} \in \mathbb{R}^{p \times q}} \left\{ \frac{1}{\sqrt{n}} \sum_{k=1}^{q} \|\boldsymbol{Y}_{\cdot,k} - \boldsymbol{X}\boldsymbol{\beta}_{\cdot,k}\|_2 + \lambda g(\boldsymbol{\beta}) \right\}.$$

Note that when g is the L_1-norm, this estimator is equivalent to q separate univariate square-root lasso estimators (Belloni et al., 2011) with the same tuning parameter λ used for each response.

`PLS`: The penalized least squares estimator of $\boldsymbol{\beta}_*$, i.e.,

$$\operatorname*{arg\,min}_{\boldsymbol{\beta} \in \mathbb{R}^{p \times q}} \left\{ \frac{1}{n} \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_F^2 + \lambda g(\boldsymbol{\beta}) \right\}. \tag{20}$$

`MRCE-Approx`: The approximate version of the multivariate regression with covariance estimation (MRCE) method proposed by Rothman et al. (2010). This estimator is computed in three steps:

1. Obtain $\boldsymbol{\beta}^{(0)}$, the `PLS` estimator.
2. Set $\hat{\boldsymbol{\Sigma}} = n^{-1}(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}^{(0)})^\top(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}^{(0)})$ and compute

$$\boldsymbol{\Omega}_\gamma^{(1)} = \operatorname*{arg\,min}_{\boldsymbol{\Omega} \in \mathbb{S}_+^q} \left\{ \operatorname{tr}(\hat{\boldsymbol{\Sigma}}\boldsymbol{\Omega}) - \log\det(\boldsymbol{\Omega}) + \gamma \sum_{j \neq k} |\boldsymbol{\Omega}_{j,k}| \right\}.$$

3. With $\boldsymbol{\Omega}_\gamma^{(1)}$ fixed, compute the `MRCE-Approx` estimator of $\boldsymbol{\beta}_*$

$$\operatorname*{arg\,min}_{\boldsymbol{\beta} \in \mathbb{R}^{p \times q}} \left[\frac{1}{n} \operatorname{tr}\big\{ (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})\boldsymbol{\Omega}_\gamma^{(1)}(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})^\top \big\} + \lambda g(\boldsymbol{\beta}) \right]. \tag{21}$$

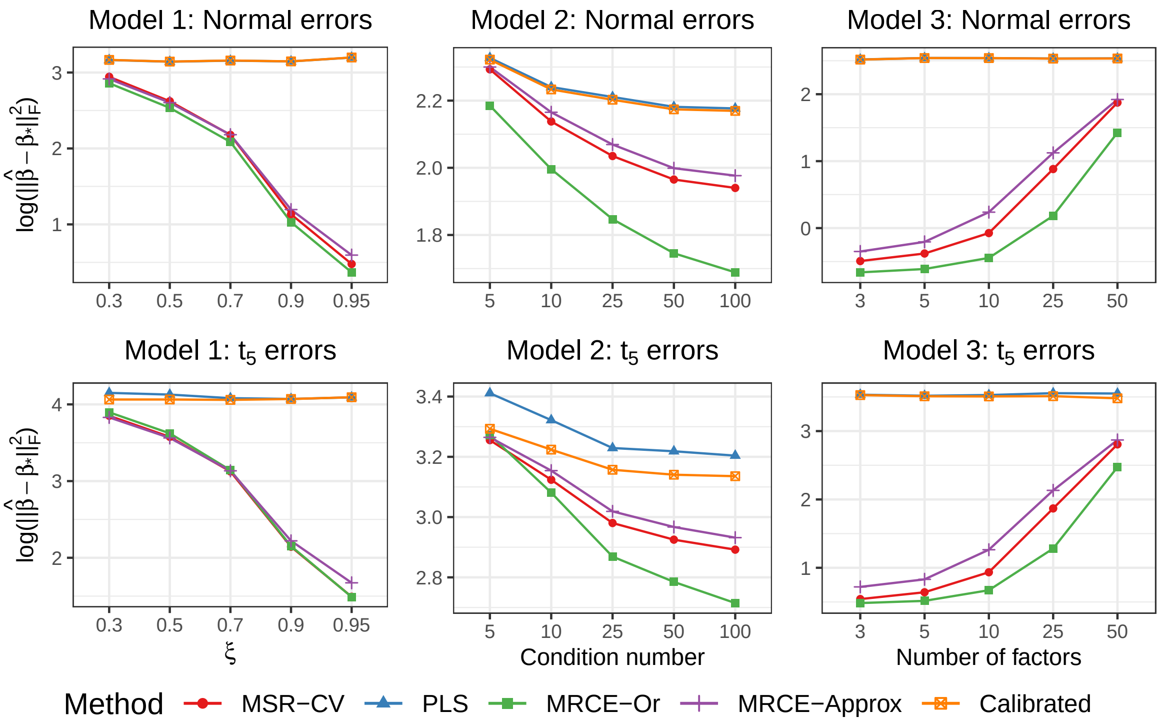

Figure 2: Average log squared Frobenius norm error over one hundred independent replications under Model 1–3 with (top row) normal errors or (bottom row) t_5 errors and ξ, the condition number, and the number of factors varying. In these simulations, $\boldsymbol{\beta}_*$ is constructed according to **M1** and g is the L_1-norm.

.

`MRCE-Or`: The "oracle" penalized normal maximum likelihood estimator of $\boldsymbol{\beta}_*$ with $\boldsymbol{\Omega}_*$ known, i.e., (21) with $\boldsymbol{\Omega}_\gamma^{(1)}$ replaced with $\boldsymbol{\Omega}_*$.

We found that computing times for the exact version of the method proposed by Rothman et al. (2010) could be prohibitively long for our data generating models, so we only compare to the approximate version described above.

5.3 Results under M1 using Cross-Validation

In our first set of simulation studies, independently for each replication we generate the regression coefficient matrix $\boldsymbol{\beta}_* \in \mathbb{R}^{p\times q}$ such that $\boldsymbol{\beta}_* = \boldsymbol{A} \circ \boldsymbol{G}$, where $\boldsymbol{A} \in \mathbb{R}^{p\times q}$, $\boldsymbol{G} \in \mathbb{R}^{p\times q}$, and $\circ$ denotes the elementwise product. The matrix $\boldsymbol{A}$, which encodes the sparsity of $\boldsymbol{\beta}_*$, has five randomly selected entries equal to one per column and all other entries equal to zero. The matrix $\boldsymbol{G}$ has independent and identically distributed standard normal entries. Thus, the matrix $\boldsymbol{\beta}_*$ has proportion of nonzero entries equal to $(5/p)$. As this $\boldsymbol{\beta}_*$ corresponds to the model subspace under **M1**, for each method, we set g to be the L_1-norm penalty.

In the top row of Figure 2, we display the average log squared Frobenius norm errors, $\log(\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}_*\|_F^2)$, for the five methods we considered under Models 1–3 with normally distributed errors. In every setting, MRCE-Or, which uses the true value of $\boldsymbol{\Omega}_*$, performs best. Among the methods which could be used in practice, MSR-CV (our method) and MRCE-Approx tend to perform similarly. Under Model 1 with normal errors, when $\xi = 0.3$, MRCE-Approx slightly outperforms the MSR-CV. As ξ increases, MSR-CV only slightly outperforms MRCE-Approx. The fact that MRCE-Approx performs well under Model 1 is not surprising: this method assumes that $\boldsymbol{\Omega}_*$ is sparse and under Model 1, $\boldsymbol{\Omega}_*$ is tri-diagonal. Under Models 2 and 3, however, MSR-CV outperforms MRCE-Approx in nearly every considered setting. Unlike Model 1, under Models 2 and 3, $\boldsymbol{\Omega}_*$ is nonsparse. Notably, MRCE-Or still outperforms both estimators, which suggests that the relatively worse performance of MRCE-Approx is due to a poor estimate of the precision matrix being used in the criterion (21).

Similar results hold when errors are generated from the t_5-distribution, although the difference between MRCE-Or and MSR-CV is slightly less apparent than under normal errors. Overall, it appears that heavy tailed errors lead to worse estimation accuracy across all the methods. Interestingly, when comparing Calibrated and PLS, we notice a difference in performance only under Model 2. This can be explained by the fact that the diagonals of $\boldsymbol{\Sigma}_*$ are different only under Model 2. Calibrated can exploit this fact, whereas PLS cannot. In fact, with condition number equal to five under Model 2, the covariance is nearly diagonal, which corresponds to the modeling assumptions of Calibrated. This partly explains why it performs similarly to MSR-CV and MRCE-Approx in this setting.

A reviewer suggested that it is counterintuitive that the performance of MRCE-Or, MRCE-Approx, and MSR-CV improves as errors become more correlated. To understand why this occurs, consider that if the errors were perfectly correlated, observing q responses for the ith subject would be like observing realizations of $\boldsymbol{\beta}_{*0} + \boldsymbol{\beta}_*^\top \boldsymbol{x}_i + e_i \mathbf{1}_q$ for $i \in [n]$, where $e_i \in \mathbb{R}$ is random and $\mathbf{1}_q = (1, 1, \dots, 1)^\top \in \mathbb{R}^q$ is a vector of ones. Of course, if we knew this were the case, we could estimate $\boldsymbol{\beta}_*$ much more efficiently than if we (incorrectly) assumed errors were independent (e.g., using least squares). The methods which improve as errors become more correlated (MRCE-Or, MRCE-Approx, and MSR-CV) are all able to exploit this situation through implicit or explicit covariance matrix estimation and thus estimate $\boldsymbol{\beta}_*$ more efficiently than the competitors. This phenomenon has been observed in numerous other works focused on multivariate response linear regression with correlated errors (Rothman et al., 2010; Molstad et al., 2021a).

In Figure 3, we display the implementation times for MSR-CV, Calibrated, and MRCE-Approx. Focusing on Model 1 under normal errors, on average, MSR-CV never takes more than a minute to compute the entire solution path (for 100 candidate tuning parameter values). Average implementation times for MRCE-Approx in the same settings are all greater than 250 seconds. Note that MRCE-Approx requires the selection of two tuning parameters—and requires estimating $\boldsymbol{\Omega}_*$—which explains the longer implementation times. Here, we consider 100 × 25 candidate tuning parameters for MRCE-Approx, but implement a rule wherein the solution path computation is terminated if the estimate of $\boldsymbol{\beta}_*$ leads to sufficiently poor prediction on the validation set. Thus, we generally compute the solution for less than half of the tuning parameter pairs under consideration. We implement no such rule for MSR-CV or Calibrated, so these results are somewhat biased in favor of MRCE-Approx. The estimator Calibrated, which we fit using the flare package in R, takes substantially longer than both other methods. However, it should be noted that the comparison to Calibrated is not

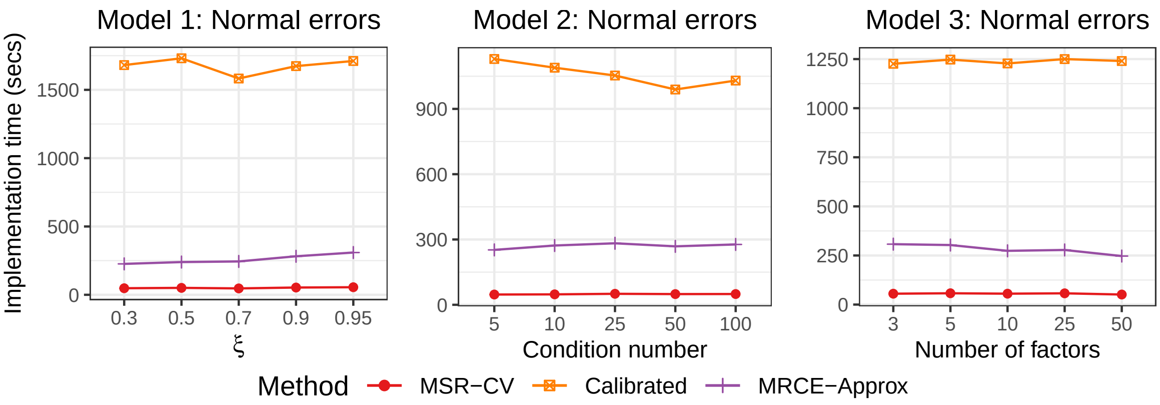

Figure 3: Average implementation times over one hundred independent replications under Model 1–3 with normal errors, $\boldsymbol{\beta}_*$ constructed according to **M1**, and g taken to be the L_1-norm.

entirely fair because the publicly available software we use requires fitting the solution path for each univariate square-root lasso estimator separately. Nonetheless, we see that `MSR-CV` is both the best performing method and can be obtained in the shortest amount of time given the existing software.

5.4 Results under M1 using Theoretical Tuning

In addition to the methods discussed in Section 5.2, we also consider multiple versions of (2) using tuning parameters suggested by the theoretical results in Section 3.2. Specifically, we also study selecting tuning parameters for (2) based on quantiles of the distribution of the random variable $(c/\sqrt{n})\|\boldsymbol{X}^\top \boldsymbol{S}\|_\infty$ where $\boldsymbol{S}$ is uniformly distributed on $V_q(n)$ and $c > 1$. In our implementation, we set $c = 1.01$. We tried multiple quantiles: 0.95, 0.85, 0.75, and 0.50. We denote the corresponding estimators `MSR-q95,` `MSR-q85,` `MSR-q75,` and `MSR-q50`, respectively. For the sake of comparison, we also used the theoretically optimal tuning parameter $(c/\sqrt{n})\|\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top\|_\infty$ where $(\boldsymbol{U}_\epsilon, \boldsymbol{D}_\epsilon, \boldsymbol{V}_\epsilon) = \text{svd}(\boldsymbol{\mathcal{E}})$: we call this estimator `MSR-Or` since it uses oracle information.

In Figure 4, we display the average squared Frobenius norm errors of `MSR-q95,` `MSR-q85,` `MSR-q75,` `MSR-q50`, and `MRCE-Or` relative to `MSR-CV`. That is, an estimator with a relative error of 1.2 has a 20% larger average squared Frobenius norm error than `MSR-CV`. Based on the results in the top row of Figure 4, it seems that in general, all directly tuned estimators tend to perform substantially worse than `MSR-CV`, including `MSR-Or`—the estimator with theoretically optimal tuning parameter. In Table 3 of the Appendix, we display average true positive and false positive variable selection rates for each of the methods displayed in Figures 2 and 4. In Table 3 we see why the directly tuned versions of (2) tended to perform worse than `MSR-CV` in terms of average squared Frobenius norm error: the false positive rates for these estimators are extremely low, but true positive rates are often much lower than those of the estimators whose tuning parameters were chosen by cross-validation. A similar result was observed in Belloni et al. (2011), who found that the direct choices of λ

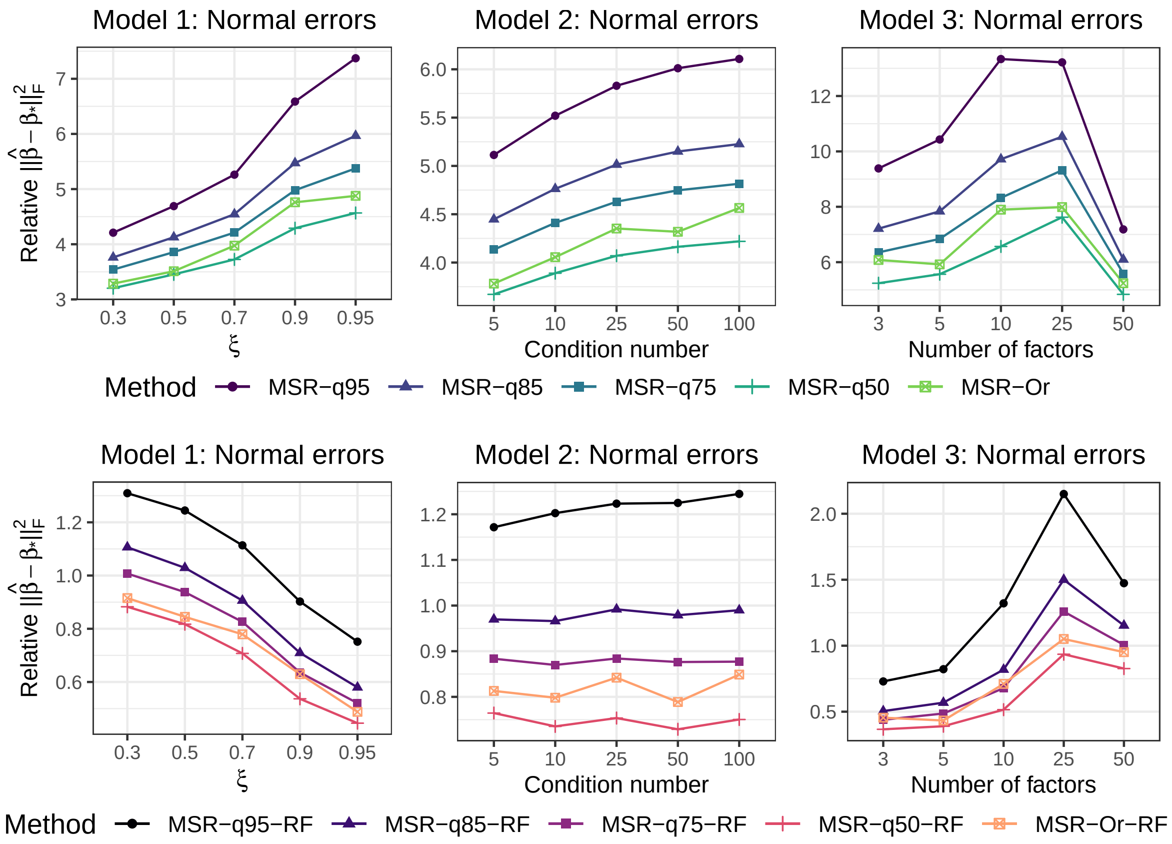

Figure 4: Relative (to `MSR-CV`) average squared Frobenius norm errors over one hundred independent replications under Model 1–3 with normal errors, $\boldsymbol{\beta}_*$ constructed according to **M1**, and g taken to be the L_1-norm both (top row) without refitting and (bottom row) with refitting.

based on theory often led to substantial bias. To alleviate this issue, we follow Belloni et al. (2011) who used a refitting procedure: we re-estimate the coefficients using a likelihood-based seemingly unrelated regression estimator described in Section A.3 of Appendix A. We refer to all refitted estimators by appending `-RF` to their names (e.g., the refitted version of `MSR-q95` is `MSR-q95-RF`). Results for refitted estimators are displayed in the bottom row of Figure 4. In this figure, we see that the performance relative to `MSR-CV` (the non-refitted version) is much improved when using refitting.

To conclude, it seems that when taking g to be the L_1-norm, direct tuning may be most useful for obtaining very sparse models with few false positives, but cross-validation may be preferred for prediction accuracy. Refitting appears to alleviate some extra bias observed when using the theory-based tuning procedures. However, in a subsequent section, we will show that under **M2**, theory-based tuning can perform as well as cross-validation-based tuning in terms of squared Frobenius norm error even without refitting.

	Model 1: ξ				
	0.3	0.5	0.7	0.9	0.95
ADMM	0.61	0.62	0.63	0.68	0.76
AccPGD	0.49	0.64	0.68	0.94	1.06
CVX	82.89	92.71	80.23	93.48	87.74

Table 1: Average computing times (in seconds) for (2) using Algorithm 1 (ADMM), and Algorithm 2 (AccPGD), and CVXR with g taken to be the L_1-norm. Averages are taken over one hundred independent replications under Model 1 with normal errors and $\boldsymbol{\beta}_*$ constructed according to **M1**. In each replication, the tuning parameter λ is that which minimizes average squared prediction error on the validation set.

5.5 Computing Time Comparisons under M1

We also compare the computing time of our algorithms to the computing time using CVX (Grant and Boyd, 2014), the off-the-shelf convex solver used to compute (2) by Stucky (2017). In Table 1, we display the average computing times for (2) with the tuning parameter selected by minimizing the average squared prediction error on the validation set under Model 1 and **M1** with normal errors. Convergence tolerances for ADMM and AccPGD are discussed in Section 4.2 and Section A.1 of Appendix A, respectively. Convergence tolerances for CVX are left at their defaults in the CVXR R package.

Briefly, the prox-linear ADMM algorithm takes less than one second on average, whereas CVX takes more than 80 seconds on average in every setting. In terms of solution accuracy, the objective function value at convergence of CVX is on average 1.000617, 1.000688, 1.000745, 1.000917, and 1.000988 (for ξ from 0.3 to 0.95) times larger than that obtained by ADMM. The solution using AccPGD is very similar to ADMM: on average their differences are even smaller than those between CVX and ADMM.

We attempted to compare the computing time of our algorithms to the iterative procedure suggested by Van de Geer and Stucky (2016). In the settings we consider, however, we found that using their algorithm, the objective function value never converged to a value near that obtained by our algorithm or CVX. In personal communication with the authors, they suggested we use CVX, citing a lack of convergence guarantees for their approach.

5.6 Results under M2 using Cross-Validation

In this section, we consider the estimation of $\boldsymbol{\beta}_*$ under **M2** by setting g to be the group lasso penalty for each of the methods discussed in Section 5.2. Specifically, for each replication under Models 1–3 as described in Section 5.2, we randomly generated $\boldsymbol{\beta}_*$ to be entirely zero except for five randomly chosen rows which have components drawn independently from a normal distribution with mean zero and standard deviation 0.1. Under this construction, only five predictors affect the q responses and the same set of predictors is important for all q responses.

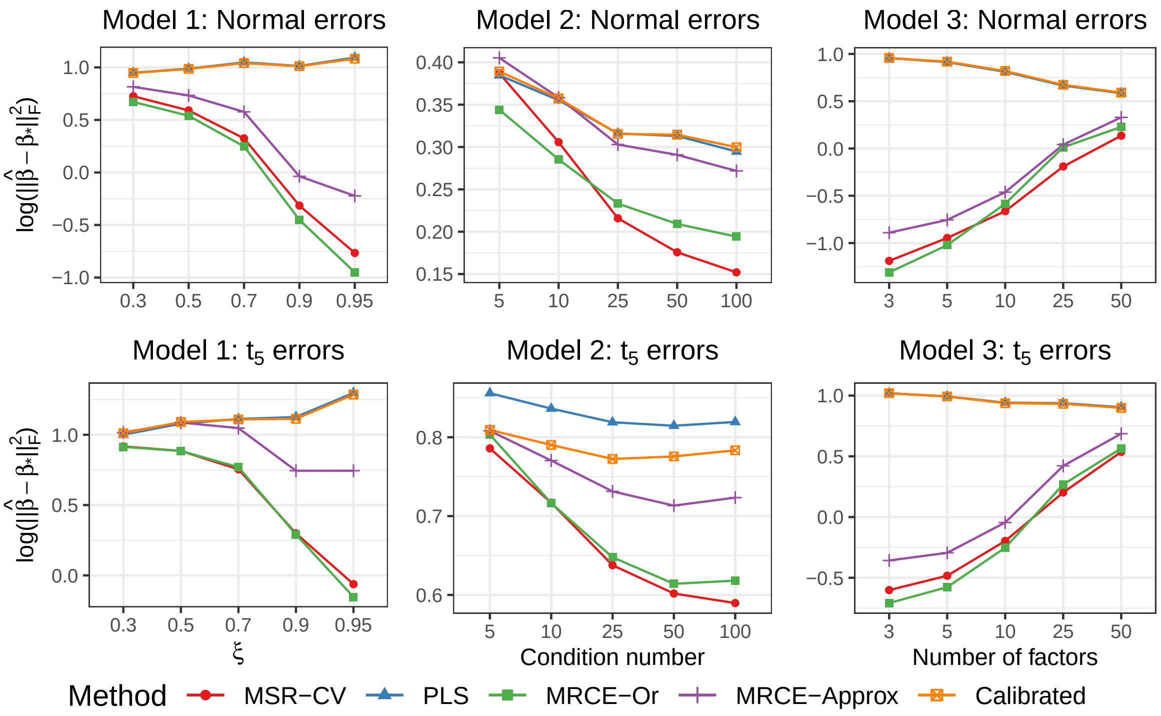

Figure 5: Average log squared Frobenius norm errors over one hundred independent replications under Model 1–3 with (top row) normal errors or (bottom row) t_5 errors and ξ, the condition number, and number of factors varying. In these simulations, $\boldsymbol{\beta}_*$ is constructed according to **M2** and g is the group lasso penalty.

.

We display average squared Frobenius norm error results in Figure 5. We see that unlike under **M1**, MSR-CV outperforms MRCE-Approx in every setting we considered. We also see that MSR-CV performs similarly to MRCE-Or. This can be partly attributed to the fact that under **M2**, variable selection is a significantly easier task than under **M1**. Because predictors are either important for all q responses or none, under **M2**, having a relatively large number of responses is helpful. Thus, since all estimators—MSR-CV included—more efficiently estimate the set of important predictors, the differences can more likely be attributed to the role of $\boldsymbol{\Omega}_*$. Evidently, using $\boldsymbol{\Omega}_*$ or an estimate thereof in (21) does not necessarily lead to better estimation than does using (2).

In Figure 6, we display implementation times of MSR-CV, Calibrated, and MRCE-Approx. To compute the solution path for Calibrated, we used the R package camel. To compute (21) with group lasso penalty, we wrote our own proximal gradient descent algorithm in R. We see that both MSR-CV and Calibrated take around a minute or less to implement in every setting. MRCE-Approx, on the other hand, can take anywhere between two and seven minutes in the settings we considered. It is important to note that here, we are using a validation set to select tuning parameters. If instead one had to perform K-fold cross-validation, MRCE-Approx may become prohibitively time-consuming to implement.

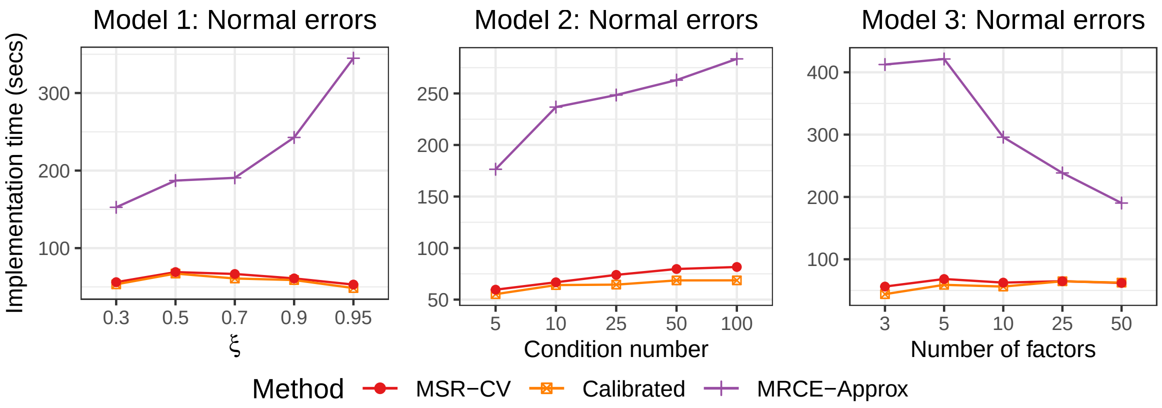

Figure 6: Average implementation times for `MSR-CV`, `Calibrated`, and `MRCE-Approx` over one hundred independent replications under Model 1–3 with normal errors, $\boldsymbol{\beta}_*$ constructed according to **M2**, and g taken to be the group lasso penalty.

.

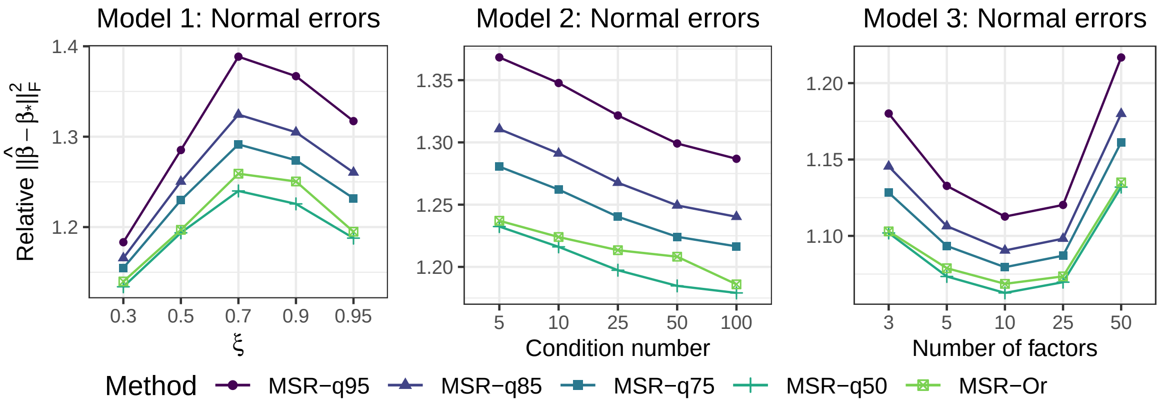

Figure 7: Relative (to `MSR-CV`) average squared Frobenius norm errors over one hundred independent replications under Model 1–3 with normal errors, $\boldsymbol{\beta}_*$ constructed according to **M2**, and g taken to be the group lasso penalty.

.

5.7 Results under M2 using Theoretical Tuning

We again consider (2) using tuning parameters chosen according to our results in Section 3.2. As mentioned in the previous subsection, variable selection in this context is substantially easier than under **M1**, and as we will see, this leads to theoretically tuned versions of (2) which perform nearly as well as those tuned using the validation set—even without refitting.

Results for the same variations of (2) (`MSR-q95`, `MSR-q85`, `MSR-q75`, `MSR-q50`, and `MRCE-Or`), except with g as the group lasso penalty and quantiles based on the distribution of $(c/\sqrt{n})\|\boldsymbol{X}^\top \boldsymbol{S}\|_{\infty,2}$, are displayed in Figure 7. In this context, we see that `MSR-q50` and `MRCE-Or` almost always have an average squared Frobenius norm error less than 1.25 that of MSR-CV. Examining the variable selection results displayed in Table 4 of the Appendix, we see that in general, the directly tuned estimators tend to have nearly perfect variable selection accuracy. The difference between the strong variable selection performance and the slight increase in squared Frobenius norm error (relative to MSR-CV) can be attributed to the bias induced from using the nuclear norm as a loss function. Refitting did slightly improve the Frobenius norm estimation error, but less so than under **M1**, so we omit these results. Finally, it is important to highlight that these estimators often take less than a single second to compute.

5.8 Results under M3 using Cross-Validation

Lastly, we consider the estimation of $\boldsymbol{\beta}_*$ under **M3** by setting g to be the nuclear norm penalty for a subset of the methods discussed in Section 5.2. The R package camel does not include an implementation of the nuclear norm penalized version of Calibrated, so this competitor is omitted from these comparisons. We focus on the setting that $(n, p, q) = (200, 50, 40)$. We adjusted dimensions because when even when $\boldsymbol{\beta}^*$ is rank r, there are a large number of parameters, $r(p + q - r)$, to be estimated.

For 100 independent replications under the data generating Models 1–3, we construct $\boldsymbol{\beta}_*$ by first computing $\boldsymbol{U}_*$ and $\boldsymbol{V}_*$, the left and right singular vectors of a randomly generated $p \times q$ matrix with independent and identically distributed standard normal entries. Then, we set $\boldsymbol{\beta}_* = \sum_{k=1}^{5} \boldsymbol{d}_k \boldsymbol{u}_{*k} \boldsymbol{v}_{*k}^\top$ where the $\boldsymbol{d} = (3, 2.5, 2, 1.5, 1)^\top$ and $\boldsymbol{u}_{*k}$ is the kth column of $\boldsymbol{U}_*$ and $\boldsymbol{v}_{*k}$ is the kth column of $\boldsymbol{V}_*$. This way, $\text{rank}(\boldsymbol{\beta}_*) = 5$ almost surely. As before, we first consider the performance of the various methods using the validation set to select tuning parameters.

Results are displayed in Figure 8. We see that like under **M1** and **M2**, in general, MRCE-Or performs best under **M3**. Interestingly, MSR-CV tends to outperform MRCE-Approx in the majority of settings considered. For example, under Model 1 and 2, when errors are more highly correlated, there is a more clear separation between MSR-CV and MRCE-Approx than under **M1**. Errors are larger overall for each method relative to **M1** or **M2** because in this setting $\boldsymbol{\beta}^*$ has $pq = 2000$ nonzero coefficients.

5.9 Results under M3 using Theoretical Tuning

Finally, we try selecting tuning parameters based on our theory. In general, however, these tuning parameters work about as poorly as under **M1** (e.g., performance was similar to that in the top row of Figure 4). For this reason, we again consider refitted versions of these estimators. To refit $\hat{\boldsymbol{\beta}}_{\text{LR}}$, we use the joint penalized maximum likelihood estimator from (23) of the Appendix, except we constrain the optimization variable $\boldsymbol{\beta}$ to belong to the set of matrices which have rank less than or equal to that of $\hat{\boldsymbol{\beta}}_{\text{LR}}$.

We display results relative to MSR-CV in Figure 9. Here, we see that theoretical tuning combined with refitting can outperform MSR-CV. In Table 5 of the Appendix, we see that the theoretically tuned versions tend to estimate the rank more accurately, but it seems that the combination of rank reduction and shrinkage of MSR-CV leads to improved performance in terms of squared Frobenius norm error compared to the refitted version which only imposes low-rankness.

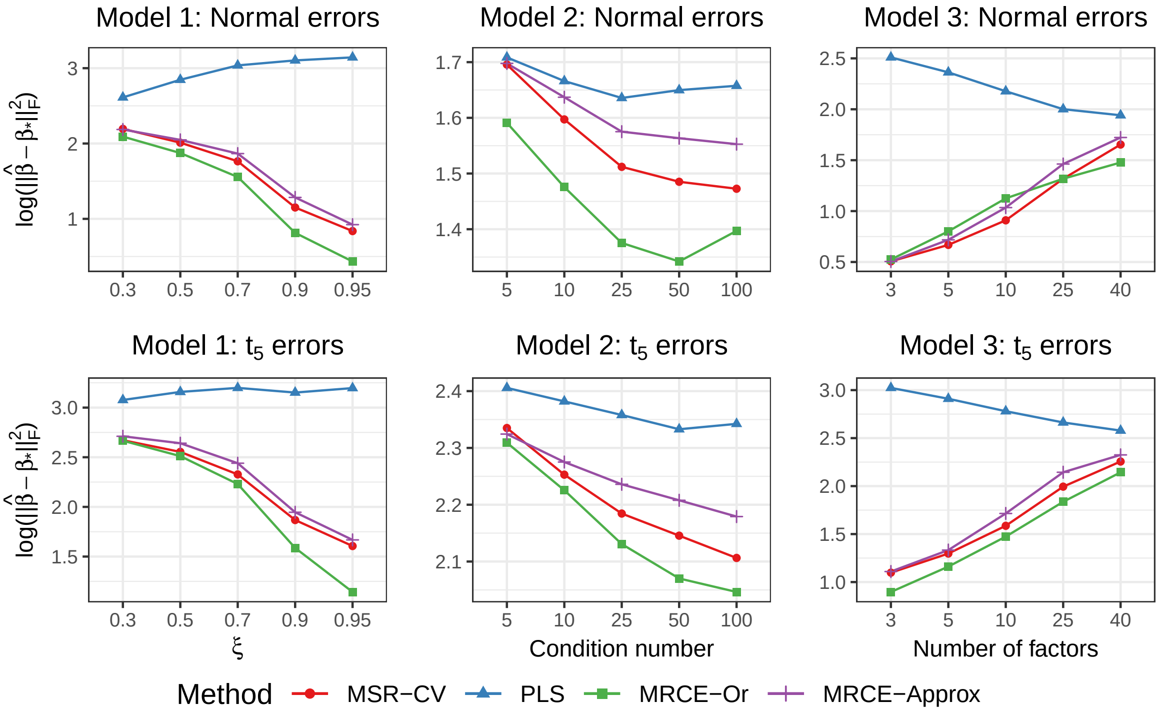

Figure 8: Average log squared Frobenius norm errors over one hundred independent replications under Model 1–3 with (top row) normal errors or (bottom row) t_5 errors and ξ, the condition number, and number of factors varying. In these simulations, $\boldsymbol{\beta}_*$ is constructed according to **M3** and g is the nuclear norm.

.

5.10 Conclusions

In these simulation studies, we saw that (2) can outperform `MRCE-Approx`, a method that requires an explicit estimate of the error precision matrix. In addition, in all of the settings we considered, `MSR-CV` required significantly less time to implement. While the tuning parameters suggested by our theory did not perform as well as those selected by cross-validation, under both **M1** and **M2**, these tuning parameters led to reasonable variable selection accuracy. Namely, the directly tuned versions of (2) rarely included predictors which were not truly important, and could be computed in around one second on average. A similar result, although related to the rank of $\boldsymbol{\beta}_*$, was observed under **M3**. In practice, we advise practitioners to use cross-validation if computing time is not an issue. Otherwise, directly tuned versions of (2) may be useful if short implementation times and model parsimony are of primary concern.

The simulation settings considered here all have $n > q$. However, (2) can be applied in settings where $q \geq n$. To demonstrate that (2) can still perform well in these settings, we provide additional simulation results in Section A.2 of Appendix A in the case that $q = 60$, $n = 50$, and $p = 500$. To summarize briefly, with $\boldsymbol{\beta}_*$ constructed according to **M1** and data generated under Models 1–3 with normal errors, (2) outperformed `MRCE-Approx` and

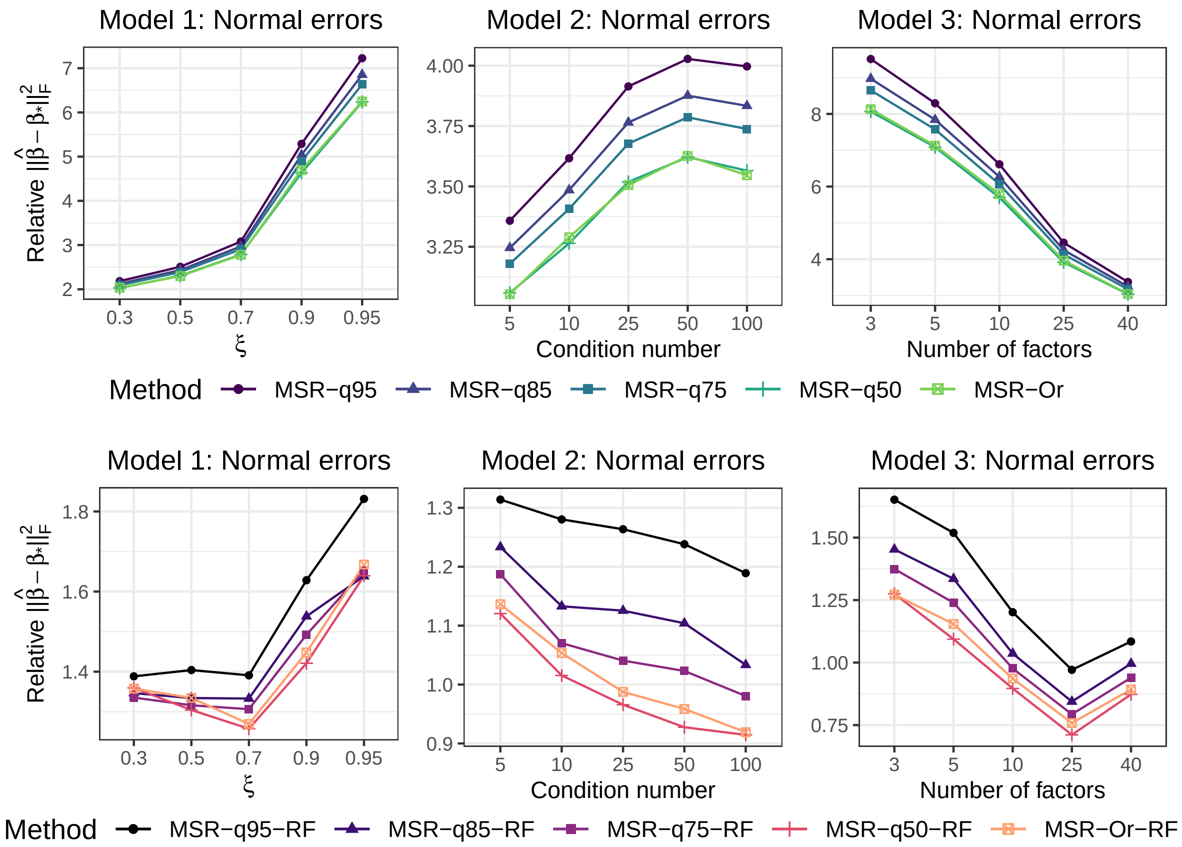

Figure 9: Relative (to `MSR-CV`) average squared Frobenius norm errors under Model 1–3 with normal errors, $\boldsymbol{\beta}^*$ constructed according to **M3**, and g taken to be the nuclear norm both (top row) without refitting and (bottom row) with refitting.

other competitors (except `MRCE-Or`) under both Models 1 and 3, but both `MRCE-Approx` and `MSR-CV` performed very poorly under Model 2. This can be attributed to the difficulties in estimating (implicitly or explicitly) the $q \times q$ error covariance with such a small sample size.

6. Glioblastoma Multiforme Application

We used our method to model the linear relationship between microRNA expression and gene expression in patients with glioblastoma multiforme—an aggressive brain cancer—collected by The Cancer Genome Atlas program (TCGA, Weinstein et al. (2013)). Earlier versions of this data set were analyzed by Wang (2015) and Lee and Liu (2012), both of whom proposed new methods for multivariate response linear regression which explicitly estimate the error precision matrix. Following both Wang (2015) and Lee and Liu (2012), microRNA expression profiles were treated as the response and gene expression profiles were treated as predictors.

Similar to Wang (2015), we reduce the dimension of both predictors and responses by retaining only the p genes with largest median absolute deviation and the q microRNAs with

	Weighted prediction error				Nuclear norm prediction error			
q	20		40		20		40	
p	500	1000	500	1000	500	1000	500	1000
`MSR-CV`	0.6411	0.6161	0.6694	0.6510	0.2126	0.2077	0.3385	0.3328
`PLS`	0.6506	0.6198	0.6740	0.6488	0.2145	0.2090	0.3399	0.3333
`PLS-q`	0.6511	0.6200	0.6754	0.6496	0.2147	0.2091	0.3414	0.3348
`MSR`*	0.6395	0.6117	0.6689	0.6460	0.2124	0.2071	0.3382	0.3319
`MRCE-Approx`*	0.6386	0.6091	0.6656	0.6399	0.2123	0.2070	0.3380	0.3313

Table 2: Weighted prediction errors and nuclear norm prediction errors averaged over 100 training/testing splits for the five considered methods from Section 6 with g taken to be the L_1-norm. The superscript $*$ denotes a method which uses best-case tuning. Methods without the $*$ uses tuning parameters chosen by five-fold cross-validation.

largest median absolute deviation. We then removed 93 subjects whose first two principal components for gene expression were substantially different than the majority of subjects. After removing these patients, there were 397 subjects in our complete data set.

For one hundred independent replications, we randomly split the data into training and testing sets of size 250 and 147, respectively. We fit the multivariate response linear regression model using multiple methods described in Section 5.2 with g taken to be the L_1-norm: `MSR-CV`, `PLS`, and a version of `PLS` with different tuning parameters λ for each of the q responses (`PLS-q`). For `MSR-CV` and `PLS`, tuning parameters are selected by five-fold cross-validation minimizing squared prediction error averaged over all responses. Unfortunately, computing times for `MRCE-Approx` could be extremely long, so we tried "best-case" tuning, i.e., we select the tuning parameters which gave the minimum squared prediction error averaged over all responses on the test set. This approach is not applicable in practice, but is included to demonstrate that (2) performs similarly to the much more computationally intensive approach. For comparison, we also include the best-case tuning version of (2). We denote both of these versions with a superscript $*$ in Table 2.

We compared the five methods in terms of two prediction metrics: nuclear norm prediction error, $\|\boldsymbol{Y}_{\text{test}} - \hat{\boldsymbol{Y}}\|_*/1000$, and weighted prediction error, $\|(\boldsymbol{Y}_{\text{test}} - \hat{\boldsymbol{Y}})\boldsymbol{\Lambda}^{-1}\|_F^2/147q$, where $\boldsymbol{\Lambda}$ is a diagonal matrix with the complete data response standard deviations along its diagonal.

Among the methods which could be used in practice, `MSR-CV` substantially outperformed both versions of `PLS` in terms of weighted prediction error when $p = 500$. When $p = 1000$, `MSR-CV` performed only similarly to `PLS`. Both best-case methods performed slightly better than `MSR-CV`, with the more computationally intensive method of Rothman et al. (2010), `MRCE-Approx`, slightly outperforming (2) with best-case tuning in the higher-dimensional settings. In terms of nuclear norm prediction error, `MSR-CV` outperformed both versions of `PLS` in every setting, and performed almost identically to the best-case version of `MRCE-Approx` in most settings.

7. Discussion

In this article, we studied multiple versions of (2), the multivariate square-root lasso. There are numerous interesting directions for future research. First, the extension of (2) to settings with matrix or tensor-valued responses may be of particular interest. In these situations, there is often a high degree of dependence across entries in the tensor-valued error (e.g., when the data are spatial and/or temporal). Implicit covariance estimation may be helpful as the dimension of the response often makes explicit covariance estimation computational infeasible. Second, it is also of interest to establish conditions under which (2) estimates exactly the set of nonzero elements of $\boldsymbol{\beta}_*$ (for **M1** and **M2**) or consistently estimates the rank of $\boldsymbol{\beta}_*$ (for **M3**). However, the nondifferentiability of the nuclear norm of residuals makes the application of the standard proof techniques (e.g., the *primal-dual witness* of Wainwright (2009) and Lee et al. (2015)) nontrivial without requiring unreasonable assumptions. For example, to establish a bound for $\|\hat{\boldsymbol{\beta}}_{\rm GL} - \boldsymbol{\beta}_*\|_{\infty,2}$, Massias et al. (2020) required that $\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_{\rm GL}$ was rank q, which as discussed in Section 3.2, is problematic. Thus, we leave the conditions necessary for support recovery and rank estimation consistency—as well as the development of a proof technique for establishing such conditions—as future work.

A reviewer pointed out a connection between (2) and a smoothed variation of (8) proposed by Massias et al. (2018, 2020). The method of Massias et al. (2018) assumes that columns of the error matrix are independent and identically distributed with covariance $\boldsymbol{\Theta}_* \in \mathbb{S}^n_+$, which they estimate explicitly. However, their estimation criterion could be modified to accommodate our assumption that rows of $\boldsymbol{\mathcal{E}}$ are independent and columns are correlated. The analog of their estimator conforming to our model assumptions in (1) is

$$\operatorname*{arg\,min}_{\boldsymbol{\beta}\in\mathbb{R}^{p\times q},\boldsymbol{\Sigma}^{1/2}\succeq\underline{\sigma}\boldsymbol{I}_q} \left[\frac{1}{2n}\mathrm{tr}\big\{(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta})\boldsymbol{\Sigma}^{-1/2}(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta})^\top\big\} + \frac{\mathrm{tr}(\boldsymbol{\Sigma}^{1/2})}{2} + \lambda g(\boldsymbol{\beta})\right], \qquad (22)$$

where the notation $\boldsymbol{\Sigma}^{1/2} \succeq \underline{\sigma}\boldsymbol{I}_q$ means $\boldsymbol{\Sigma}^{1/2} - \underline{\sigma}\boldsymbol{I}_q$ is positive semidefinite and $\underline{\sigma} > 0$ is a tuning parameter lower bounding the eigenvalues of $\boldsymbol{\Sigma}^{1/2} \in \mathbb{S}^q_+$. Thus, we can view both the method of Massias et al. (2018) and (22) as smooth approximations to (2). As future work, it would be interesting to study whether the additional constraint on $\boldsymbol{\Sigma}^{1/2}$ in (22) would allow one to relax the assumption that $n > q$. However, (22) does have a potential drawback: (22) can sometimes require explicit estimation of $\boldsymbol{\Sigma}_*^{1/2}$, so it is not clear when this estimator would be any easier to compute than the method of Rothman et al. (2010).

Acknowledgments

The author thanks three anonymous referees and the action editor for their many helpful comments. The author also thanks Benjamin Stucky and Sara van de Geer for sharing their code and their responses to inquiries; thanks Rohit K. Patra for a helpful conservation; and thanks Daniel J. Eck, Karl Oskar Ekvall, Keshav Motwani, Bradley S. Price, Adam J. Rothman, and Ben Sherwood for their feedback on earlier drafts of this article. This work was supported in part by National Science Foundation grant DMS-2113589.

Appendix A. Additional Details

A.1 Additional Computational Details

In this section, we discuss our implementation of the accelerated proximal gradient descent algorithm in Algorithm 2. As mentioned in Section 4.3, this algorithm can be used in situations where $\hat{\boldsymbol{\beta}}_g$ belongs to $\mathcal{D}_{\underline{\kappa}}$ for some positive $\underline{\kappa}$ bounded away from zero. Since we do not know, in general, whether $\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_g$ will be rank q before computing $\hat{\boldsymbol{\beta}}_g$, we can attempt to use Algorithm 2, and if any iterates do not belong to $\mathcal{D}_{\underline{\kappa}}$, we may instead revert to using Algorithm 1. In our implementation, if $n > q$, we start computing the solution path for $\hat{\boldsymbol{\beta}}_g$ using Algorithm 2, but if at any iterate, the diagonal elements of $\bar{\boldsymbol{D}}$ or $\tilde{\boldsymbol{D}}$ (see 3 and 5 of Algorithm 2) are smaller than 10^{-3}, we revert to Algorithm 1 and compute the rest of the solution path using Algorithm 1.

To claim convergence, we check the first order conditions as described in Remark 7. For concreteness, we discuss the version we use with g being the L_1-norm. Specifically, we let $(\boldsymbol{U}_{\epsilon^{(k+1)}}, \boldsymbol{D}_{\epsilon^{(k+1)}}, \boldsymbol{V}_{\epsilon^{(k+1)}}) = \text{svd}(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}^{(k+1)})$. Then, we terminate the algorithm if (i) $\|\boldsymbol{X}^\top \boldsymbol{U}_{\epsilon^{(k+1)}} \boldsymbol{V}_{\epsilon^{(k+1)}}^\top\|_\infty \leq \sqrt{n}\lambda$, (ii) $\max_{(l,m):[\hat{\boldsymbol{\beta}}^{(k+1)}]_{l,m} \neq 0} ||[\boldsymbol{X}^\top \boldsymbol{U}_{\epsilon^{(k+1)}} \boldsymbol{V}_{\epsilon^{(k+1)}}^\top - \sqrt{n}\lambda \text{sign}(\hat{\boldsymbol{\beta}}^{(k+1)})]_{l,m}| < \tau$, and (iii) $\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}^{(k+1)}$ is rank q. For the timing results in Table 1, we set $\tau = 10^{-10}$. We found that compared to the default implementation of the prox-linear ADMM (Algorithm 1), Algorithm 2 led to very slightly more accurate solutions.

```
Algorithm 2: Accelerated proximal gradient descent algorithm for (2)
  1. Given $\rho_0 > 0$ and $\gamma_{\text{decr}} \in (0,1)$, initialize $\boldsymbol{\beta}^{(-1)} = \boldsymbol{\beta}^{(0)} \in \mathbb{R}^{p \times q}$, $\alpha^{(0)} = \alpha^{(-1)} = 1$,
     $(\dot{\boldsymbol{U}}, \dot{\boldsymbol{D}}, \dot{\boldsymbol{V}}) = \text{svd}(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}^{(0)})$ and set $k = 0$
  2. $\boldsymbol{\Gamma}^{(k)} \leftarrow \boldsymbol{\beta}^{(k)} + \left(\frac{\alpha^{(k-1)}-1}{\alpha^{(k)}}\right)\left(\boldsymbol{\beta}^{(k)} - \boldsymbol{\beta}^{(k-1)}\right)$
  3. $(\tilde{\boldsymbol{U}}, \tilde{\boldsymbol{D}}, \tilde{\boldsymbol{V}}) \leftarrow \text{svd}(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\Gamma}^{(k)})$
  4. $\bar{\boldsymbol{\beta}} \leftarrow \text{Prox}_{\rho_k \lambda g}(\boldsymbol{\Gamma}^{(k)} + \frac{\rho_k}{\sqrt{n}}\boldsymbol{X}^\top \tilde{\boldsymbol{U}}\tilde{\boldsymbol{V}}^\top)$
  5. $(\bar{\boldsymbol{U}}, \bar{\boldsymbol{D}}, \bar{\boldsymbol{V}}) \leftarrow \text{svd}(\boldsymbol{Y} - \boldsymbol{X}\bar{\boldsymbol{\beta}})$
  6. If $\text{tr}(\bar{\boldsymbol{D}}) < \text{tr}(\tilde{\boldsymbol{D}}) + \text{tr}\{\tilde{\boldsymbol{V}}\tilde{\boldsymbol{U}}^\top \boldsymbol{X}(\boldsymbol{\Gamma}^{(k)} - \bar{\boldsymbol{\beta}})\} + \frac{\sqrt{n}}{2\rho_k}\|\boldsymbol{\Gamma}^{(k)} - \bar{\boldsymbol{\beta}}\|_F^2$, go to 7
         Else, update $\rho_k \leftarrow \rho_k \gamma_{\text{decr}}$ and return to 4
  7. If $\text{tr}(\bar{\boldsymbol{D}}) + \sqrt{n}\lambda g(\bar{\boldsymbol{\beta}}) \leq \text{tr}(\dot{\boldsymbol{D}}) + \sqrt{n}\lambda g(\boldsymbol{\beta}^{(k)})$, set $\boldsymbol{\beta}^{(k+1)} \leftarrow \bar{\boldsymbol{\beta}}$ and $\dot{\boldsymbol{D}} \leftarrow \bar{\boldsymbol{D}}$
         Else, set $\boldsymbol{\beta}^{(k+1)} \leftarrow \boldsymbol{\beta}^{(k)}$
  8. $\alpha^{(k+1)} \leftarrow (1 + \sqrt{1 + 4\{\alpha^{(k)}\}^2})/2$
  8. If not converged, set $\rho_{k+1} \leftarrow \rho_k$, update $k \leftarrow k + 1$, and return to 2
```

A.2 Additional Simulation Results

In this section, we display additional simulation results with $\boldsymbol{\beta}_*$ constructed according to **M1** and $(n, p, q) = (50, 60, 500)$. The only difference between these data generating models and those from Section 5.3 is that entries of $\boldsymbol{G}$ (from $\boldsymbol{\beta}_* = \boldsymbol{A} \circ \boldsymbol{G}$) are independent and identically distributed from a mean zero normal distribution with standard deviation two. Results from these simulations are displayed in Figure 10. We observe that `MSR-CV` performs

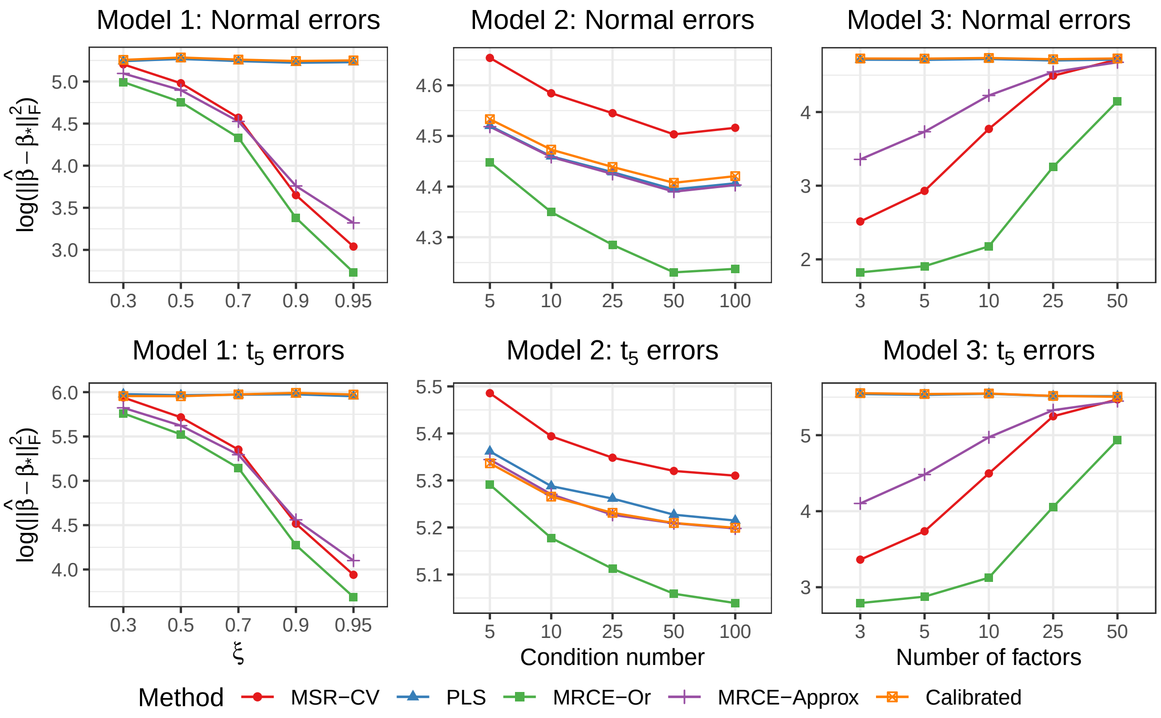

Figure 10: Average log squared Frobenius norm errors over one hundred independent replications under Model 1–3 with $(n, p, q) = (50, 500, 60)$ and (top row) normal errors or (bottom row) t_5 errors and ξ, the condition number, and the number of factors varying. In these simulations, $\boldsymbol{\beta}_*$ is constructed according to **M1** and g is the L_1-norm.

.

relatively well under both Model 1 and Model 3. Of course, compared to the results in Section 5.3, all estimators perform worse, which is expected given the smaller sample size and larger q. Notably, under Model 2, both `MSR-CV` and `MRCE-Approx` perform worse than `PLS` and `Calibrated`. However, we see that the oracle penalized maximum likelihood estimator, `MRCE-Or`, still performs well here. This suggests that the covariance structure under Model 2 is much more difficult to estimate than under Models 1 and 3 when the sample size is small relative to q. Together these results suggest that `MSR-CV` can work well in settings with $n > q$, although one may also consider `Calibrated` which makes the simplifying assumption that $\boldsymbol{\Sigma}_*$ is diagonal.

A.3 Method for Refitting

To refit the estimators as described in Section 5.4, we use a seemingly unrelated regressions-type (Zellner, 1962) penalized normal maximum likelihood estimator. Suppose we are given $\hat{\boldsymbol{\beta}}_g$, an estimate of $\boldsymbol{\beta}_*$ from which we want to obtain a refitted version with, for example, an identical sparsity pattern as $\hat{\boldsymbol{\beta}}_{\rm L}$ (when g is the L_1-norm), or a rank less than or equal to

	Model 1: ξ									
	0.3		0.5		0.7		0.9		0.95	
PLS	0.781	0.040	0.789	0.041	0.784	0.042	0.785	0.044	0.785	0.045
MRCE-Or	0.820	0.042	0.848	0.044	0.881	0.045	0.931	0.047	0.952	0.049
MRCE-Approx	0.819	0.046	0.848	0.049	0.884	0.052	0.932	0.056	0.954	0.062
Calibrated	0.782	0.041	0.789	0.041	0.785	0.041	0.784	0.043	0.784	0.045
MSR-CV	0.812	0.043	0.843	0.043	0.876	0.046	0.928	0.049	0.949	0.052
MSR-q95	0.487	0.000	0.551	0.000	0.623	0.000	0.757	0.000	0.824	0.000
MSR-q85	0.519	0.000	0.583	0.000	0.653	0.000	0.780	0.000	0.843	0.000
MSR-q75	0.537	0.000	0.599	0.000	0.666	0.000	0.791	0.000	0.851	0.000
MSR-q50	0.562	0.000	0.622	0.000	0.689	0.000	0.807	0.000	0.863	0.000
MSR-Or	0.558	0.000	0.621	0.000	0.679	0.000	0.798	0.000	0.860	0.000
	Model 2: Condition number									
	5		10		25		50		100	
PLS	0.847	0.045	0.846	0.045	0.847	0.044	0.847	0.043	0.848	0.045
MRCE-Or	0.972	0.052	0.970	0.050	0.968	0.048	0.957	0.052	0.916	0.050
MRCE-Approx	0.972	0.061	0.968	0.060	0.964	0.059	0.940	0.054	0.899	0.052
Calibrated	0.847	0.044	0.847	0.045	0.849	0.044	0.847	0.043	0.848	0.044
MSR-CV	0.970	0.051	0.966	0.050	0.962	0.048	0.936	0.045	0.892	0.046
MSR-q95	0.879	0.000	0.864	0.000	0.821	0.000	0.696	0.000	0.626	0.000
MSR-q85	0.895	0.000	0.882	0.000	0.849	0.000	0.732	0.000	0.658	0.000
MSR-q75	0.903	0.000	0.890	0.000	0.860	0.000	0.748	0.000	0.675	0.000
MSR-q50	0.911	0.000	0.901	0.000	0.876	0.000	0.775	0.000	0.700	0.000
MSR-Or	0.907	0.000	0.899	0.000	0.866	0.000	0.772	0.000	0.688	0.000
	Model 3: Number of factors									
	2		5		10		25		50	
PLS	0.862	0.044	0.870	0.044	0.875	0.045	0.876	0.044	0.874	0.044
MRCE-Or	0.874	0.046	0.886	0.047	0.899	0.049	0.903	0.050	0.905	0.050
MRCE-Approx	0.867	0.048	0.878	0.051	0.889	0.054	0.893	0.054	0.892	0.054
Calibrated	0.864	0.045	0.871	0.045	0.876	0.046	0.877	0.045	0.875	0.045
MSR-CV	0.866	0.046	0.876	0.045	0.886	0.046	0.891	0.046	0.890	0.046
MSR-q95	0.601	0.000	0.622	0.000	0.637	0.000	0.638	0.000	0.639	0.000
MSR-q85	0.631	0.000	0.652	0.000	0.665	0.000	0.668	0.000	0.669	0.000
MSR-q75	0.645	0.000	0.666	0.000	0.681	0.000	0.683	0.000	0.684	0.000
MSR-q50	0.668	0.000	0.690	0.000	0.703	0.000	0.706	0.000	0.704	0.000
MSR-Or	0.663	0.000	0.684	0.000	0.693	0.000	0.701	0.000	0.695	0.000

Table 3: Average true positive and false positive variable selection rates for Models 1–3 under normal errors with $\boldsymbol{\beta}_*$ constructed according to **M1** and g taken to be the L_1-norm.

that of $\hat{\boldsymbol{\beta}}_{\rm LR}$ (when g is the nuclear norm). Define the set

$$\mathrm{C_L}(\hat{\boldsymbol{\beta}}_{\rm L}) = \left\{\boldsymbol{\beta} \in \mathbb{R}^{p \times q} : \boldsymbol{\beta}_{j,k} = 0, \text{ for all } (j,k) \text{ such that } [\hat{\boldsymbol{\beta}}_{\rm L}]_{j,k} = 0\right\},$$

and define

$$\mathrm{C_{LR}}(\hat{\boldsymbol{\beta}}_{\rm LR}) = \left\{\boldsymbol{\beta} \in \mathbb{R}^{p \times q} : \mathrm{rank}(\boldsymbol{\beta}) \leq \mathrm{rank}(\hat{\boldsymbol{\beta}}_{\rm LR})\right\}.$$

	Model 1: ξ									
	0.3		0.5		0.7		0.9		0.95	
PLS	0.068	0.010	0.028	0.008	0.014	0.009	0.008	0.006	0.018	0.009
MRCE-Or	0.910	0.093	0.976	0.110	1.000	0.120	1.000	0.149	1.000	0.156
MRCE-Approx	0.778	0.116	0.888	0.148	0.922	0.165	0.984	0.222	0.992	0.246
Calibrated	0.058	0.009	0.028	0.007	0.014	0.008	0.008	0.006	0.014	0.008
MSR-CV	0.900	0.092	0.980	0.105	1.000	0.117	1.000	0.165	1.000	0.215
MSR-q95	0.204	0.000	0.454	0.000	0.900	0.000	1.000	0.000	1.000	0.000
MSR-q85	0.288	0.000	0.572	0.000	0.942	0.001	1.000	0.000	1.000	0.000
MSR-q75	0.340	0.001	0.614	0.001	0.952	0.001	1.000	0.001	1.000	0.000
MSR-q50	0.420	0.001	0.706	0.001	0.972	0.001	1.000	0.002	1.000	0.001
MSR-Or	0.388	0.001	0.674	0.001	0.962	0.001	1.000	0.001	1.000	0.001
	Model 2: Condition number									
	5		10		25		50		100	
PLS	0.152	0.018	0.242	0.020	0.676	0.050	0.910	0.071	0.972	0.080
MRCE-Or	1.000	0.172	1.000	0.178	1.000	0.222	1.000	0.550	1.000	0.448
MRCE-Approx	1.000	0.261	1.000	0.298	1.000	0.318	1.000	0.231	1.000	0.180
Calibrated	0.144	0.017	0.228	0.020	0.672	0.050	0.904	0.070	0.966	0.071
MSR-CV	1.000	0.192	1.000	0.152	1.000	0.154	1.000	0.144	1.000	0.100
MSR-q95	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	0.998	0.000
MSR-q85	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
MSR-q75	1.000	0.000	1.000	0.001	1.000	0.000	1.000	0.000	1.000	0.000
MSR-q50	1.000	0.001	1.000	0.001	1.000	0.001	1.000	0.001	1.000	0.001
MSR-Or	1.000	0.001	1.000	0.001	1.000	0.001	1.000	0.001	1.000	0.001
	Model 3: Number of factors									
	2		5		10		25		50	
PLS	1.000	0.100	0.998	0.103	1.000	0.102	1.000	0.098	1.000	0.102
MRCE-Or	0.998	0.149	1.000	0.195	1.000	0.280	1.000	0.350	1.000	0.397
MRCE-Approx	0.994	0.130	0.998	0.138	0.998	0.144	1.000	0.159	1.000	0.147
Calibrated	1.000	0.103	0.998	0.102	1.000	0.100	1.000	0.096	1.000	0.107
MSR-CV	1.000	0.117	1.000	0.114	1.000	0.117	1.000	0.120	1.000	0.118
MSR-q95	0.850	0.000	0.928	0.000	0.982	0.000	0.988	0.000	0.992	0.000
MSR-q85	0.894	0.000	0.960	0.000	0.990	0.000	0.994	0.000	0.996	0.000
MSR-q75	0.920	0.000	0.970	0.000	0.992	0.000	0.994	0.001	0.998	0.001
MSR-q50	0.942	0.001	0.982	0.001	0.998	0.002	0.996	0.001	1.000	0.001
MSR-Or	0.928	0.001	0.972	0.001	0.996	0.001	0.990	0.001	0.998	0.001

Table 4: Average true positive and false positive variable selection rates for Models 1–3 under normal errors with $\boldsymbol{\beta}_*$ constructed according to **M2** and g taken to the be group lasso penalty.

To obtain the refitted version of $\hat{\boldsymbol{\beta}}_g$, we solve

$$\underset{\boldsymbol{\beta} \in \mathrm{C}_g(\hat{\boldsymbol{\beta}}_g), \boldsymbol{\Omega} \in \mathbb{S}^q_+}{\arg\min} \left[\frac{1}{n} \mathrm{tr}\left\{ (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}) \boldsymbol{\Omega} (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})^\top \right\} - \log\det(\boldsymbol{\Omega}) + \frac{\alpha}{2} \|\boldsymbol{\Omega}\|_F^2 \right], \tag{23}$$

	Model 1: ξ									
	0.3		0.5		0.7		0.9		0.95	
`PLS`	9.190	0.194	6.550	0.224	4.570	0.201	3.130	0.180	3.230	0.191
`MRCE-Or`	15.060	0.134	16.360	0.131	17.960	0.141	20.610	0.138	21.990	0.165
`MRCE-Approx`	16.830	0.132	19.590	0.174	23.010	0.190	27.730	0.212	28.920	0.366
`MSR-CV`	14.860	0.146	15.590	0.198	16.260	0.275	17.060	0.331	17.330	0.403
`MSR-q95`	2.590	0.059	2.810	0.054	3.180	0.066	3.680	0.055	3.900	0.059
`MSR-q85`	2.900	0.050	3.090	0.045	3.440	0.062	3.950	0.056	4.160	0.053
`MSR-q75`	3.080	0.051	3.250	0.054	3.630	0.061	4.110	0.058	4.290	0.056
`MSR-q50`	3.380	0.053	3.570	0.059	3.930	0.059	4.400	0.055	4.530	0.054
`MSR-Or`	3.390	0.062	3.590	0.067	3.960	0.063	4.350	0.059	4.430	0.066
	Model 2: Condition number									
	5		10		25		50		100	
`PLS`	7.470	0.070	8.980	0.080	10.810	0.101	12.900	0.117	14.130	0.110
`MRCE-Or`	27.950	0.225	28.310	0.173	26.750	0.141	20.240	0.161	20.820	0.134
`MRCE-Approx`	23.390	0.282	23.020	0.295	23.400	0.204	25.900	0.259	21.220	0.203
`MSR-CV`	16.290	0.215	15.890	0.175	16.070	0.161	16.530	0.149	17.270	0.120
`MSR-q95`	4.110	0.055	3.950	0.063	4.030	0.061	3.860	0.060	3.450	0.063
`MSR-q85`	4.340	0.054	4.180	0.056	4.330	0.059	4.160	0.060	3.700	0.061
`MSR-q75`	4.510	0.050	4.360	0.054	4.470	0.054	4.300	0.058	3.850	0.059
`MSR-q50`	4.690	0.046	4.660	0.048	4.710	0.046	4.580	0.052	4.140	0.055
`MSR-Or`	4.690	0.046	4.570	0.056	4.610	0.055	4.460	0.058	4.110	0.060
	Model 3: Number of factors									
	2		5		10		25		40	
`PLS`	16.190	0.144	15.650	0.140	15.660	0.167	15.470	0.156	15.120	0.153
`MRCE-Or`	18.000	0.133	18.650	0.124	19.370	0.141	19.630	0.143	17.500	0.326
`MRCE-Approx`	17.110	0.131	17.600	0.169	18.610	0.239	19.630	0.260	19.970	0.242
`MSR-CV`	18.260	0.143	18.410	0.126	18.580	0.146	18.730	0.134	18.310	0.126
`MSR-q95`	3.040	0.060	3.220	0.060	3.300	0.059	3.380	0.055	3.480	0.059
`MSR-q85`	3.260	0.063	3.560	0.056	3.610	0.060	3.650	0.054	3.790	0.057
`MSR-q75`	3.440	0.062	3.740	0.050	3.780	0.054	3.820	0.048	3.950	0.059
`MSR-q50`	3.700	0.058	4.010	0.054	4.040	0.057	4.130	0.054	4.200	0.064
`MSR-Or`	3.700	0.067	3.960	0.065	4.010	0.064	4.100	0.063	4.210	0.064

Table 5: Average estimated rank and standard errors for Models 1–3 under normal errors with $\boldsymbol{\beta}_*$ constructed according to **M3** and g taken to be the nuclear norm.

where we fix $\alpha = 10^{-4}$. To solve (23), we use blockwise coordinate descent. Specifically, for $k = 1, 2, 3, \dots,$ until convergence, we iterate between the following two steps:

1. $\boldsymbol{\Omega}^{(k+1)} = \boldsymbol{U}\{-\boldsymbol{D}+(\boldsymbol{D}^2+4\alpha\boldsymbol{I}_q)^{1/2}\}\boldsymbol{U}^\top/(2\alpha)$ where $(\boldsymbol{U},\boldsymbol{D},\boldsymbol{U}) = \text{svd}\{(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta}^{(k)})^\top(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta}^{(k)})/n\}$.

2. $\boldsymbol{\beta}^{(k+1)} = \arg\min_{\boldsymbol{\beta}\in \mathrm{C}_g(\hat{\boldsymbol{\beta}}_g)} \text{tr}\{(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta})\boldsymbol{\Omega}^{(k+1)}(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta})^\top\}$

Step 1 is the well-known solution for the ridge-penalized normal likelihood precision matrix estimation problem (Witten and Tibshirani, 2009). When g is the L_1-norm, Step 2 can be solved efficiently using an accelerated projected gradient descent algorithm. When g is the nuclear norm, Step 2 has a closed form (e.g., see Chapter 2 of Reinsel and Velu (1998)). We terminate the algorithm when the objective function value converges.

Appendix B. Proofs

B.1 Notation and Preliminaries

First, we clarify some of the notation that will be used in later sections. Recall that we assume that $\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta}_* + \boldsymbol{\mathcal{E}}$ where $\boldsymbol{\mathcal{E}} \in \mathbb{R}^{n\times q}$ is a random matrix of errors which is assumed to have mean zero and be rank q almost surely. For the remainder, define $(\boldsymbol{U}_\epsilon, \boldsymbol{D}_\epsilon, \boldsymbol{V}_\epsilon) = \text{svd}(\boldsymbol{\mathcal{E}})$.

For ease of display, we use ϕ and ν in place of $\phi_{\mathcal{E},g}(\mathcal{M}, \mathcal{N}, c)$ and $\nu_g(\mathcal{M}, \mathcal{N}, c)$, respectively. As before, define the quantities $\check{c} = (c + 1)/(c - 1)$ and $\tilde{c} = c(c + 1)/(c - 1)$ for constant $c > 1$. Also, for a symmetric matrix $\boldsymbol{A}$, let $\varphi_1(\boldsymbol{A})$ denote the largest eigenvalue of $\boldsymbol{A}$, and for an arbitrary matrix $\boldsymbol{A}$, let $\sigma_j(\boldsymbol{A})$ denote the jth largest singular value of $\boldsymbol{A}$. For an $a \times b$ matrix $\boldsymbol{A}$, we let $\boldsymbol{A}_{j,\cdot} \in \mathbb{R}^b$ denote the jth row of $\boldsymbol{A}$, $\boldsymbol{A}_{\cdot,k} \in \mathbb{R}^a$ the kth column of $\boldsymbol{A}$, and $\boldsymbol{A}_{j,k}$ the (j, k)th entry of $\boldsymbol{A}$. Let $\boldsymbol{I}_s$ be the $s \times s$ identity matrix. Finally, for sequences a_n and b_n, let the notation $a_n \asymp b_n$ mean that $a_n = O(b_n)$ and $b_n = O(a_n)$.

Throughout, let $\tilde{g}$ denote the dual norm of g, i.e., $\tilde{g}(x) = \sup_z\{z^\top x : g(z) \leq 1\}$. For examples of penalty functions g and their dual norms, see Table 1 of Wainwright (2014). Following the notation of Negahban et al. (2012), let $\mathcal{M}$ and $\mathcal{N}^\perp$ denote the model subspace and the perturbation subspace, respectively. Under the various model assumptions, **M1**–**M3**, we assume that $\boldsymbol{\beta}_* \in \mathcal{M}$, where the specific form of $\mathcal{M}$ depends on the particular model. Stated in another way, we assume $\boldsymbol{\beta}_* = \boldsymbol{\beta}_{*\mathcal{M}} + \boldsymbol{\beta}_{*\mathcal{M}^\perp} = \boldsymbol{\beta}_{*\mathcal{M}}$, where $\boldsymbol{\beta}_{*\mathcal{M}}$ denotes the projection of $\boldsymbol{\beta}_*$ onto $\mathcal{M}$, $\boldsymbol{\beta}_{*\mathcal{M}} = \arg\min_{\boldsymbol{A}\in\mathcal{M}} \|\boldsymbol{A} - \boldsymbol{\beta}_*\|_F^2$, and $\mathcal{M}^\perp$ is the orthogonal complement of $\mathcal{M}$. Under each of the model assumptions, **M1**–**M3**, we will assume the use of a penalty which is decomposable with respect to the pair $(\mathcal{M}, \mathcal{N}^\perp)$ in the sense that $g(\boldsymbol{A} + \boldsymbol{B}) = g(\boldsymbol{A}) + g(\boldsymbol{B})$ for all $\boldsymbol{A} \in \mathcal{M}$ and $\boldsymbol{B} \in \mathcal{N}^\perp$. Under **M1**, the norm $\|\boldsymbol{A}\|_1 = \sum_{j,k} |\boldsymbol{A}_{j,k}|$ is decomposable (with respect to the subspace pair defined in **M1**); under **M2**, the norm $\|\boldsymbol{A}\|_{1,2} = \sum_j(\sum_k \boldsymbol{A}_{j,k}^2)^{1/2}$ is decomposable; and under **M3**, the norm $\|\boldsymbol{A}\|_* = \sum \varphi_j(\boldsymbol{A})$ is decomposable. Further, define the norms $\|\boldsymbol{A}\|_\infty = \max_{j,k} |\boldsymbol{A}_{j,k}|$, $\|\boldsymbol{A}\| = \sigma_1(\boldsymbol{A})$, and $\|\boldsymbol{A}\|_{\infty,2} = \max_j \|\boldsymbol{A}_{j,\cdot}\|_2$, where $\|\cdot\|_2$ denotes the Euclidean norm of a vector.

For random quantities $\boldsymbol{u}$ and $\boldsymbol{v}$, we write $\boldsymbol{u} \sim \boldsymbol{v}$ to mean that $\boldsymbol{u}$ and $\boldsymbol{v}$ have the same distribution. Throughout, let $O(n)$ denote the set of $n \times n$ matrices $\boldsymbol{O}$ such that $\boldsymbol{O}^\top\boldsymbol{O} = \boldsymbol{I}_n$ and let $V_q(n)$, a Stiefel manifold, denote the set of $n \times q$ matrices $\boldsymbol{S}$ such that $\boldsymbol{S}^\top\boldsymbol{S} = \boldsymbol{I}_q$. Let $S^{n-1} = \{\boldsymbol{u} \in \mathbb{R}^n : \|\boldsymbol{u}\|_2 = 1\}$. In the following subsections, we refer to random matrices as having the uniform distribution on $O(n)$ and $V_q(n)$. Following Eaton (1989), by uniform distribution we mean the unique translation-invariant probability measure. For a thorough treatment of the unique translation-invariant probability measure (called Haar measure) on $O(n)$, see Chapter 1 of Meckes (2019). For additional details on the uniform distribution on $V_q(n)$, see Camano-Garcia (2006) or Chapter 7 of Eaton (1989).

B.2 Preliminary Lemmas

To begin, we first provide a preliminary lemma which will be used throughout our proofs.

Lemma 9 *(i) (Mattila, 1995, Section 3.5) If $\boldsymbol{O}$ is a random matrix having the uniform distribution on $O(n)$, then for any fixed vector $\boldsymbol{a} \in S^{n-1}$, $\boldsymbol{Oa}$ has a uniform distribution on S^{n-1}. (ii) If $\boldsymbol{S}$ is a random matrix having the uniform distribution on $V_q(n)$, then for any fixed unit vector $\boldsymbol{b} \in S^{q-1}$, $\boldsymbol{Sb}$ has the uniform distribution on S^{n-1}.*

The second part of Lemma 9, (ii), follows almost immediately from (i). For example, since $\boldsymbol{S} \sim \boldsymbol{O}\boldsymbol{\mathcal{P}}_q$ where $\boldsymbol{O}$ is uniformly distributed on $O(n)$ and $\boldsymbol{\mathcal{P}}_q \in \mathbb{R}^{n \times q}$ is the first q columns of $\boldsymbol{I}_n$ (see (IV.1) of Lyubarskii and Vershynin (2010)), it follows that $\boldsymbol{Sv} \sim \boldsymbol{O}\boldsymbol{\mathcal{P}}_q\boldsymbol{v}$, so that because because $\boldsymbol{\mathcal{P}}_q\boldsymbol{v} \in S^{n-1}$, an application of (i) yields (ii).

The next lemma follows immediately from the proof of Proposition 7.1 in Eaton (1989).

Lemma 10 *Suppose $\boldsymbol{R} \in \mathbb{R}^{n \times q}$ is a random matrix which has q non-zero singular values almost surely and suppose $\boldsymbol{R}$ is left-spherical, i.e., for any $\boldsymbol{O} \in O(n)$, $\boldsymbol{OR} \sim \boldsymbol{R}$. Let $(\boldsymbol{U}_{\boldsymbol{R}}, \boldsymbol{D}_{\boldsymbol{R}}, \boldsymbol{V}_{\boldsymbol{R}}) = \mathrm{svd}(\boldsymbol{R})$. Then, the random matrix $\boldsymbol{R}(\boldsymbol{R}^\top\boldsymbol{R})^{-1/2} = \boldsymbol{U}_{\boldsymbol{R}}\boldsymbol{V}_{\boldsymbol{R}}^\top$ follows a uniform distribution on $V_q(n)$.*

The next lemma is a well-known result about the subdifferential of the nuclear norm. A proof sketch can be found in Section B.5.

Lemma 11 *Assume **A1** is true. Then, the subdifferential of $\boldsymbol{\beta} \mapsto \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_*$ at $\boldsymbol{\beta}_*$ is the singleton*

$$-\boldsymbol{X}^\top\boldsymbol{U}_\epsilon\boldsymbol{V}_\epsilon^\top = -\boldsymbol{X}^\top(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}_*)\{(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}_*)^\top(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}_*)\}^{-1/2}.$$

B.3 Proof of Theorem 3 and Corollary 4

We now focus our attention on the proofs of Theorem 3 and Corollary 4. We begin with a preliminary lemma.

Lemma 12 *Assume **A1** is true. Define the event $\mathcal{A}_c = \{\lambda \geq (c/\sqrt{n})\tilde{g}(\boldsymbol{X}^\top\boldsymbol{U}_\epsilon\boldsymbol{V}_\epsilon^\top)\}$ for a fixed constant $c > 1$. Then, on $\mathcal{A}_c$, $\hat{\boldsymbol{\Delta}} = \hat{\boldsymbol{\beta}}_g - \boldsymbol{\beta}_*$ belongs to the set*

$$\mathcal{C}_g(\mathcal{M}, \mathcal{N}, c) = \left\{\boldsymbol{\Delta} \in \mathbb{R}^{p \times q} : g(\boldsymbol{\Delta}_{\mathcal{N}^\perp}) \leq \check{c}g(\boldsymbol{\Delta}_{\mathcal{N}})\right\}.$$

We omit the proof of Lemma 12 as it follows directly from the proof of Lemma 1 from Negahban et al. (2012) using the fact that under **A1**, the gradient of nuclear norm of residuals with respect to $\boldsymbol{\beta}$ evaluated at $\boldsymbol{\beta}_*$ is $-\boldsymbol{X}^\top\boldsymbol{U}_\epsilon\boldsymbol{V}_\epsilon^\top$ (e.g., see Lemma 11). Note that in the main text, we exclude the singleton $\boldsymbol{\Delta} = 0$ from $\mathcal{C}_g(\mathcal{M}, \mathcal{N}, c)$.

Next, we give a lower bound on the difference between the nuclear norm of residuals evaluated at $\boldsymbol{\beta}_* + \boldsymbol{\Delta}$ and $\boldsymbol{\beta}_*$ for $\boldsymbol{\Delta} \in \mathcal{C}_g(\mathcal{M}, \mathcal{N}, c)$. The proof can be found in Section B.5, but this follows straightforwardly from the convexity of the nuclear norm of residuals and definition of ϕ.

Lemma 13 *Assume **A1** is true. Then, for all $\boldsymbol{\Delta} \in \mathcal{C}_g(\mathcal{M}, \mathcal{N}, c)$,*

$$\frac{1}{\sqrt{n}}\|\boldsymbol{Y} - \boldsymbol{X}(\boldsymbol{\beta}_* + \boldsymbol{\Delta})\|_* - \frac{1}{\sqrt{n}}\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}_*\|_* \geq \phi\|\boldsymbol{\Delta}\|_F^2 - \frac{1}{\sqrt{n}}|\mathrm{tr}(\boldsymbol{\Delta}^\top\boldsymbol{X}^\top\boldsymbol{U}_\epsilon\boldsymbol{V}_\epsilon^\top)|.$$

We are now ready to prove Theorem 3.

Proof of Theorem 3. To prove Theorem 3, we follow the proof technique detailed in Negahban et al. (2012). For $\delta > 0$, define the set $\mathsf{B}_{\delta,c} = \{\boldsymbol{\Delta} \in \mathbb{R}^{p \times q} : \|\boldsymbol{\Delta}\|_F = \delta, g(\boldsymbol{\Delta}_{\mathcal{N}^\perp}) \leq \check{c} g(\boldsymbol{\Delta}_{\mathcal{N}})\}$ and let $\mathcal{L}(\boldsymbol{\beta}) = \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_* / \sqrt{n} + \lambda g(\boldsymbol{\beta})$. Because $\mathcal{L}$ is convex and $\hat{\boldsymbol{\beta}}_g$ is its minimizer, on $\mathcal{A}_c$,

$$\inf_{\boldsymbol{\Delta} \in \mathsf{B}_{\delta,c}} \{\mathcal{L}(\boldsymbol{\beta}_* + \boldsymbol{\Delta}) - \mathcal{L}(\boldsymbol{\beta}_*)\} > 0 \implies \|\hat{\boldsymbol{\beta}}_g - \boldsymbol{\beta}_*\|_F \leq \delta.$$

For a proof of this fact, see the proof of Lemma 4 of the Supplementary Material to Negahban et al. (2012). To simplify notation, let $D(\boldsymbol{\Delta}) = \mathcal{L}(\boldsymbol{\beta}_* + \boldsymbol{\Delta}) - \mathcal{L}(\boldsymbol{\beta}_*)$ so that we need only show that $\inf_{\Delta \in \mathsf{B}_{\delta,c}} D(\boldsymbol{\Delta}) > 0$ for $\delta = \lambda \check{c} \Psi_g(\mathcal{N})/\phi$. Notice first that

$$\begin{aligned} D(\boldsymbol{\Delta}) &= \frac{1}{\sqrt{n}}\|\boldsymbol{Y} - \boldsymbol{X}(\boldsymbol{\beta}_* + \boldsymbol{\Delta})\|_* - \frac{1}{\sqrt{n}}\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}_*\|_* + \lambda g(\boldsymbol{\beta}_* + \boldsymbol{\Delta}) - \lambda g(\boldsymbol{\beta}_*) \\ &\geq \phi \|\boldsymbol{\Delta}\|_F^2 - \frac{1}{\sqrt{n}} |\mathrm{tr}(\boldsymbol{\Delta}^\top \boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top)| + \lambda g(\boldsymbol{\beta}_* + \boldsymbol{\Delta}) - \lambda g(\boldsymbol{\beta}_*) \qquad (24) \\ &\geq \phi \|\boldsymbol{\Delta}\|_F^2 - \frac{1}{\sqrt{n}} |\mathrm{tr}(\boldsymbol{\Delta}^\top \boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top)| + \lambda g(\boldsymbol{\Delta}_{\mathcal{N}^\perp}) - \lambda g(\boldsymbol{\Delta}_{\mathcal{N}}) \qquad (25) \end{aligned}$$

where (24) follows from the fact that $\boldsymbol{\Delta} \in \mathsf{B}_{\delta,c}$ implies $\boldsymbol{\Delta} \in \mathcal{C}_g(\mathcal{M}, \mathcal{N}, c)$ and Lemma 13; and (25) follows from the triangle inequality

$$g(\boldsymbol{\beta}_* + \boldsymbol{\Delta}) - g(\boldsymbol{\beta}_*) = g(\boldsymbol{\beta}_{*\mathcal{M}} + \boldsymbol{\Delta}_{\mathcal{N}} + \boldsymbol{\Delta}_{\mathcal{N}^\perp}) - g(\boldsymbol{\beta}_{*\mathcal{M}}) \geq g(\boldsymbol{\beta}_{*\mathcal{M}} + \boldsymbol{\Delta}_{\mathcal{N}^\perp}) - g(\boldsymbol{\Delta}_{\mathcal{N}}) - g(\boldsymbol{\beta}_{*\mathcal{M}}),$$

and decomposability of the penalty function g with respect to the pair $(\mathcal{M}, \mathcal{N}^\perp)$

$$g(\boldsymbol{\beta}_{*\mathcal{M}} + \boldsymbol{\Delta}_{\mathcal{N}^\perp}) - g(\boldsymbol{\Delta}_{\mathcal{N}}) - g(\boldsymbol{\beta}_{*\mathcal{M}}) = g(\boldsymbol{\beta}_{*\mathcal{M}}) + g(\boldsymbol{\Delta}_{\mathcal{N}^\perp}) - g(\boldsymbol{\Delta}_{\mathcal{N}}) - g(\boldsymbol{\beta}_{*\mathcal{M}}) = g(\boldsymbol{\Delta}_{\mathcal{N}^\perp}) - g(\boldsymbol{\Delta}_{\mathcal{N}}).$$

Thus, applying Hölder's inequality to the second term in (25), we have

$$D(\boldsymbol{\Delta}) \geq \phi \|\boldsymbol{\Delta}\|_F^2 - \frac{1}{\sqrt{n}} g(\boldsymbol{\Delta}) \tilde{g}(\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top) + \lambda g(\boldsymbol{\Delta}_{\mathcal{N}^\perp}) - \lambda g(\boldsymbol{\Delta}_{\mathcal{N}}).$$

It then follows that on event $\mathcal{A}_c = \{\lambda \geq (c/\sqrt{n}) \tilde{g}(\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top)\}$,

$$\begin{aligned} D(\boldsymbol{\Delta}) &\geq \phi \|\boldsymbol{\Delta}\|_F^2 - \frac{\lambda}{c} g(\boldsymbol{\Delta}) + \lambda g(\boldsymbol{\Delta}_{\mathcal{N}^\perp}) - \lambda g(\boldsymbol{\Delta}_{\mathcal{N}}), \\ &\geq \phi \|\boldsymbol{\Delta}\|_F^2 - \frac{\lambda}{c} g(\boldsymbol{\Delta}_{\mathcal{N}}) - \frac{\lambda}{c} g(\boldsymbol{\Delta}_{\mathcal{N}^\perp}) + \lambda g(\boldsymbol{\Delta}_{\mathcal{N}^\perp}) - \lambda g(\boldsymbol{\Delta}_{\mathcal{N}}), \qquad (26) \\ &= \phi \|\boldsymbol{\Delta}\|_F^2 - \lambda \left(\frac{c+1}{c}\right) g(\boldsymbol{\Delta}_{\mathcal{N}}) + \lambda \left(\frac{c-1}{c}\right) g(\boldsymbol{\Delta}_{\mathcal{N}^\perp}), \end{aligned}$$

where (26) follows from the triangle equality $g(\boldsymbol{\Delta}) \leq g(\boldsymbol{\Delta}_{\mathcal{N}}) + g(\boldsymbol{\Delta}_{\mathcal{N}^\perp})$. Then, because

$$g(\boldsymbol{\Delta}_{\mathcal{N}}) \leq \Psi_g(\mathcal{N}) \|\boldsymbol{\Delta}\|_F \qquad \text{and} \qquad \lambda \left(\frac{c-1}{c}\right) g(\boldsymbol{\Delta}_{\mathcal{N}^\perp}) \geq 0,$$

it follows that

$$D(\boldsymbol{\Delta}) \geq \phi \|\boldsymbol{\Delta}\|_F^2 - \lambda \left(\frac{c+1}{c}\right) \Psi_g(\mathcal{N}) \|\boldsymbol{\Delta}\|_F.$$

Finally, since $\|\boldsymbol{\Delta}\|_F = \delta$ for $\boldsymbol{\Delta} \in \mathsf{B}_{\delta,c}$, setting $\lambda = \phi\delta/\{\check{c}\Psi_g(\mathcal{N})\}$, or equivalently $\delta = \check{c}\lambda\Psi_g(\mathcal{N})/\phi$, yields

$$D(\boldsymbol{\Delta}) \geq \phi\delta^2 \left\{1 - \lambda\left(\frac{c+1}{c}\right)\frac{\Psi_g(\mathcal{N})}{\phi\delta}\right\} = \phi\delta^2\left(1 - \frac{c-1}{c}\right) > 0,$$

from which the first conclusion follows since **A3** implies $\phi > 0$. The second claim follows immediately from Lemma 10 under assumption **A2**. ■

We now turn our attention to the proof of Corollary 4. By the result of Theorem 3, we need only select a λ such $\mathcal{A}_c$ occurs with high probability. To do so, we will need the following three concentration inequalities: proofs can be found in Section B.5.

Lemma 14 *Suppose $\boldsymbol{S}$ is a random matrix having the uniform distribution on $V_q(n)$. Let $k > 1$ be a fixed constant. If **C1**, **A1**, and **A2** hold, then*

$$P\left(\frac{1}{\sqrt{n}}\|\boldsymbol{X}^\top \boldsymbol{S}\|_\infty \geq \sqrt{\frac{2\log(2pq^k)}{n-1}}\right) \leq q^{1-k}$$

*as long as $n > 2\log(2pq^k) + 1$. Hence, under **M1**, setting $\lambda = c\{2\log(2pq^k)/(n-1)\}^{1/2}$,*

$$P(\mathcal{A}_c) \geq 1 - q^{1-k}.$$

Lemma 15 *Suppose $\boldsymbol{S}$ is a random matrix having the uniform distribution on $V_q(n)$. Let $k > 1$ be a fixed constant such that $k\log p > 4\pi$. If **C1**, **A1**, and **A2** hold, then*

$$P\left(\frac{1}{\sqrt{n}}\|\boldsymbol{X}^\top \boldsymbol{S}\|_{\infty,2} \geq \sqrt{\frac{4k\log p}{n-2}} + \sqrt{\frac{q}{n}}\right) \leq p^{1-k}.$$

*Hence, under **M2**, setting $\lambda = c\{4k\log p/(n-2)\}^{1/2} + c(q/n)^{1/2}$,*

$$P(\mathcal{A}_c) \geq 1 - p^{1-k}.$$

Lemma 16 *Suppose $\boldsymbol{S}$ is a random matrix having the uniform distribution on $V_q(n)$. Let $k_1 > 1$ be a fixed constant such that $k_2 = 4\log(7 + k_1)$ and $k_2\|\boldsymbol{X}\|^2(p+q) > 16n\pi$. If **C1**, **A1**, and **A2** hold, then*

$$P\left\{\frac{1}{\sqrt{n}}\|\boldsymbol{X}^\top \boldsymbol{S}\| \geq \frac{4\|\boldsymbol{X}\|}{\sqrt{n}}\left(\sqrt{\frac{k_2(p+q)}{n-2}} + \sqrt{\frac{1}{n}}\right)\right\} \leq \left(\frac{8}{7+k_1}\right)^{p+q}.$$

*Hence, under **M3**, setting $\lambda = 4c\|\boldsymbol{X}\|[k_2(p+q)/\{n(n-2)\}]^{1/2} + 4c\|\boldsymbol{X}\|/n$,*

$$P(\mathcal{A}_c) \geq 1 - \left(\frac{8}{7+k_1}\right)^{p+q}.$$

Proof of Corollary 4. Under the conditions of Corollary 4, **C1**, **A1**–**A3** hold so that we can apply the result of Theorem 3.

(i) Under **M1** with $g(\cdot) = \|\cdot\|_1$, $\tilde{g}(\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top) = \|\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top\|_\infty$ and $\Psi_{\|\cdot\|_1}(\mathcal{N}) \leq \sqrt{|\mathcal{S}|}$ (Negahban et al., 2012). If $\lambda = c\{2\log(2pq^k)/(n-1)\}^{1/2}$ and $n > 2\log(2pq^k) + 1$ for fixed constants $c > 1$ and $k > 1$, an application of Lemma 14 implies $P(\mathcal{A}_c) \geq 1 - q^{1-k}$. Hence, applying Theorem 3, it follows that

$$P\left(\|\hat{\boldsymbol{\beta}}_{\mathrm{L}} - \boldsymbol{\beta}_*\|_F \leq \frac{\tilde{c}}{\phi}\sqrt{\frac{2|\mathcal{S}|\log(2pq^k)}{(n-1)}}\right) \geq 1 - q^{1-k}.$$

(ii) Under **M2** with $g(\cdot) = \|\cdot\|_{1,2}$, $\tilde{g}(\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top) = \|\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top\|_{\infty,2}$ and $\Psi_{\|\cdot\|_{1,2}}(\mathcal{N}) \leq \sqrt{|\mathcal{G}|}$ (Liu et al., 2015). If $\lambda = c\{4k\log p/(n-2)\}^{1/2} + c(q/n)^{1/2}$ for constants $c > 1$ and $k > 1$ such that $k\log p > 4\pi$, Lemma 15 implies $P(\mathcal{A}_c) \geq 1 - p^{1-k}$. Hence, applying Theorem 3, it follows that

$$P\left\{\|\hat{\boldsymbol{\beta}}_{\mathrm{GL}} - \boldsymbol{\beta}_*\|_F \leq \frac{2\tilde{c}}{\phi}\left(\sqrt{\frac{k|\mathcal{G}|\log p}{n-2}} + \sqrt{\frac{|\mathcal{G}|q}{4n}}\right)\right\} \geq 1 - p^{1-k}.$$

(iii) Under **M3** with $g(\cdot) = \|\cdot\|_*$, $\tilde{g}(\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top) = \|\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top\|$ and $\Psi_{\|\cdot\|_*}(\mathcal{N}) \leq \sqrt{2r}$ (Negahban and Wainwright, 2011). With $\lambda = 4c\|\boldsymbol{X}\|[k_2(p+q)/\{n(n-2)\}]^{1/2} + 4c\|\boldsymbol{X}\|/n$ for constants $c > 1$ and $k_1 > 1$ such that $k_2 = 4\log(7+k_1)$ and $k_2\|\boldsymbol{X}\|^2(p+q) > 16n\pi$, Lemma 16 implies $P(\mathcal{A}_c) \geq 1 - \{8/(7+k_1)\}^{p+q}$. Hence, applying Theorem 3, it follows that

$$P\left\{\|\hat{\boldsymbol{\beta}}_{\mathrm{LR}} - \boldsymbol{\beta}_*\|_F \leq \frac{4\tilde{c}}{\phi}\left(\frac{\|\boldsymbol{X}\|}{\sqrt{n}}\right)\left(\sqrt{\frac{2k_2 r(p+q)}{n}} + \sqrt{\frac{2r}{n}}\right)\right\} \geq 1 - \left(\frac{8}{7+k_1}\right)^{p+q}. \quad \blacksquare$$

B.4 Proof of Theorem 5 and Corollary 6

We now focus on the proofs of Theorem 5 and Corollary 6. To prove Theorem 5, we require the following lemma, which we prove in Section B.5.

Lemma 17 *Under* ***C1*** *and* ***A4–A6****, if n is sufficiently large, on the event $\mathcal{A}_c \cap \mathcal{B}_d$*

$$\frac{1}{\sqrt{n}}\|\boldsymbol{\mathcal{E}} - \boldsymbol{X}\boldsymbol{\Delta}\|_* - \frac{1}{\sqrt{n}}\|\boldsymbol{\mathcal{E}}\|_* + \frac{1}{\sqrt{n}}\mathrm{tr}(\boldsymbol{\Delta}^\top \boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top) \geq \frac{\nu}{4(t+d)\,\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)}\|\boldsymbol{\Delta}\|_F^2,$$

provided $\|\boldsymbol{\Delta}\|_F^2 \to 0$ as $n \to \infty$.

Proof of Theorem 5. By the same arguments used to prove Theorem 3, for sequence $\delta_n \to 0$ as $n \to \infty$, defining $\mathsf{B}_{\delta_n,c} = \{\boldsymbol{\Delta} \in \mathbb{R}^{p\times q} : \|\boldsymbol{\Delta}\|_F = \delta_n, g(\boldsymbol{\Delta}_{\mathcal{N}^\perp}) \leq \check{c}\, g(\boldsymbol{\Delta}_{\mathcal{N}})\}$, we have that on $\mathcal{A}_c \cap \mathcal{B}_d$,

$$D(\boldsymbol{\Delta}) > 0 \text{ for all } \Delta \in \mathsf{B}_{\delta_n,c} \implies \|\hat{\boldsymbol{\beta}}_g - \boldsymbol{\beta}_*\|_F \leq \delta_n.$$

Recall that

$$D(\boldsymbol{\Delta}) = \underbrace{\frac{1}{\sqrt{n}}\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}_* - \boldsymbol{X}\boldsymbol{\Delta}\|_* - \frac{1}{\sqrt{n}}\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}_*\|_*}_{T_1} + \underbrace{\lambda g(\boldsymbol{\beta}_* + \boldsymbol{\Delta}) - \lambda g(\boldsymbol{\beta}_*)}_{T_2}.$$

For n sufficiently large, on $\mathcal{A}_c \cap \mathcal{B}_d$, we can bound T_1 using Lemma 17 and Hölder's inequality, i.e., $|\mathrm{tr}(\boldsymbol{\Delta}^\top \boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top)| \leq g(\boldsymbol{\Delta})\tilde{g}(\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top)$, to see

$$T_1 \geq \frac{\nu}{4(t+d)\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)}\|\boldsymbol{\Delta}\|_F^2 - \frac{1}{\sqrt{n}} g(\boldsymbol{\Delta})\tilde{g}(\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top).$$

Similarly, we can bound T_2 by the same arguments as those used to obtain (25). Hence,

$$D(\boldsymbol{\Delta}) \geq \frac{\nu}{4(t+d)\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)}\|\boldsymbol{\Delta}\|_F^2 - \frac{1}{\sqrt{n}} g(\boldsymbol{\Delta})\tilde{g}(\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top) + \lambda g(\boldsymbol{\Delta}_{\mathcal{N}^\perp}) - \lambda g(\boldsymbol{\Delta}_{\mathcal{N}}),$$

so that on $\mathcal{A}_c = \{\lambda \geq (c/\sqrt{n})\tilde{g}(\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top)\}$, an application of the triangle inequality $g(\boldsymbol{\Delta}) \leq g(\boldsymbol{\Delta}_{\mathcal{N}^\perp}) + g(\boldsymbol{\Delta}_{\mathcal{N}})$ and the fact that $\lambda(c-1)g(\boldsymbol{\Delta}_{\mathcal{N}^\perp})/c \geq 0$ yields

$$D(\boldsymbol{\Delta}) \geq \frac{\nu}{4(t+d)\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)}\|\boldsymbol{\Delta}\|_F^2 - \lambda\left(\frac{c+1}{c}\right) g(\boldsymbol{\Delta}_{\mathcal{N}}).$$

By definition of $\Psi_g(\mathcal{N})$, this implies

$$D(\boldsymbol{\Delta}) \geq \frac{\nu}{4(t+d)\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)}\|\boldsymbol{\Delta}\|_F^2 - \lambda\left(\frac{c+1}{c}\right) \Psi_g(\mathcal{N})\|\boldsymbol{\Delta}\|_F$$

so that finally, since $\|\boldsymbol{\Delta}\|_F = \delta_n$ for $\boldsymbol{\Delta} \in \mathsf{B}_{\delta_n, c}$, we have

$$D(\boldsymbol{\Delta}) \geq \nu\delta_n^2 \left(\frac{1}{4(t+d)\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)} - \lambda\left(\frac{c+1}{c}\right)\frac{\Psi_g(\mathcal{N})}{\nu\delta_n}\right)$$

which is positive if $\lambda = \delta_n\nu/\{\check{c}\Psi_g(\mathcal{N})4(t+d)\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)\}$, or equivalently, $\delta_n = 4\lambda\check{c}\Psi_g(\mathcal{N})(t+d)\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)/\nu$. Hence, because $\delta_n \asymp \lambda\Psi_g(\mathcal{N})$ under **A4**–**A6**, the result follows by requiring that $\lambda\Psi_g(\mathcal{N}) \to 0$ as $n \to \infty$. ■

Finally, to prove Corollary 6, in addition to Lemmas 14–16, we need the following concentration inequality.

Lemma 18 *Let $\boldsymbol{\mathcal{E}} \in \mathbb{R}^{n\times q}$ be a matrix with rows independent and identically distributed from $\mathrm{N}_q(0, \boldsymbol{\Sigma}_*)$. Let $\varphi_1(\boldsymbol{\Sigma}_*)$ denote the the largest eigenvalue of $\boldsymbol{\Sigma}_*$, and let $\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)$ be its square-root. Given fixed constant $d > 1$, if $\sqrt{q/n} \to t \in [0, 1)$ as $n \to \infty$, then for n sufficiently large*

$$P\left(\frac{1}{\sqrt{n}}\|\boldsymbol{\mathcal{E}}\| \leq (t+d)\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)\right) \geq 1 - 2e^{-(d-1)^2 n/4}.$$

The proof of Lemma 18 can be found in Section B.5. With Lemma 18 in hand, we can use Theorem 5 and Lemmas 14–16 to prove Corollary 6.

Proof of Corollary 6. Under the conditions of Theorem 5, **C1** and **A4**–**A6** hold so that we can apply the result of Theorem 5.

(i) Under **M1** with $g(\cdot) = \|\cdot\|_1$, $\tilde{g}(\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top) = \|\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top\|_\infty$ and $\Psi_{\|\cdot\|_1}(\mathcal{N}) \leq \sqrt{|\mathcal{S}|}$. If $n > 2\log(2pq^k) + 1$ and $\lambda = c\{2\log(2pq^k)/(n-1)\}^{1/2}$ for fixed constants $c > 1$, $k > 1$, and $d > 1$, then applications of Lemma 14 and 18 imply that for n sufficiently large, $P(\mathcal{A}_c \cap \mathcal{B}_d) \geq 1 - q^{1-k} - 2e^{-(d-1)^2 n/4}$. Therefore, Theorem 5 implies that under the same conditions,

$$P\left\{\|\hat{\boldsymbol{\beta}}_{\mathrm{L}} - \boldsymbol{\beta}_*\|_F \leq \varphi_1^{1/2}(\boldsymbol{\Sigma}_*)\left(\frac{4(t+d)\tilde{c}}{\nu}\right)\sqrt{\frac{2c_1|\mathcal{S}|\log(2pq^k)}{(n-1)}}\right\}$$

with probability at least $1 - q^{1-k} - 2e^{-(d-1)^2 n/4}$ as long as $\lambda\sqrt{|\mathcal{S}|} \to 0$ as $n \to \infty$.

(ii) Under **M2** with $g(\cdot) = \|\cdot\|_{1,2}$, $\tilde{g}(\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top) = \|\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top\|_{\infty,2}$ and $\Psi_{\|\cdot\|_{1,2}}(\mathcal{N}) \leq \sqrt{|\mathcal{G}|}$. If $\lambda = c\{4k\log p/(n-2)\}^{1/2} + c(q/n)^{1/2}$ and $k\log p > 4\pi$ for fixed constants $c > 1$, $k > 1$, and $d > 1$, then applications of Lemma 15 and 18 imply that for n sufficiently large, $P(\mathcal{A}_c \cap \mathcal{B}_d) \geq 1 - p^{1-k} - 2e^{-(d-1)^2 n/4}$. Therefore, Theorem 5 implies that under the same conditions,

$$\|\hat{\boldsymbol{\beta}}_{\mathrm{GL}} - \boldsymbol{\beta}_*\|_F \leq \varphi_1^{1/2}(\boldsymbol{\Sigma}_*)\left(\frac{8(t+d)\tilde{c}}{\nu}\right)\left(\sqrt{\frac{k|\mathcal{G}|\log p}{n-2}} + \sqrt{\frac{|\mathcal{G}|q}{4n}}\right)$$

with probability as least $1 - p^{1-c_2} - 2e^{-(d-1)^2 n/4}$ as long as $\lambda\sqrt{|\mathcal{G}|} \to 0$ as $n \to \infty$.

(iii) Under **M3** with $g(\cdot) = \|\cdot\|_*$, $\tilde{g}(\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top) = \|\boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top\|$ and $\Psi_{\|\cdot\|_*}(\mathcal{N}) \leq \sqrt{2r}$. If $\lambda = 4c\|\boldsymbol{X}\|[k_2(p+q)/\{n(n-2)\}]^{1/2} + 4c\|\boldsymbol{X}\|/n$ for fixed constants $c > 1$, $k_1 > 1$, and $d > 1$ such that $k_2 = 4\log(7 + k_1)$ and $k_2\|\boldsymbol{X}\|^2(p+q) > 16n\pi$, then applications of Lemma 16 and Lemma 18 imply that for n sufficiently large, $P(\mathcal{A}_c \cap \mathcal{B}_d) \geq 1 - \{8/(7 + k_1)\}^{p+q} - 2e^{-(d-1)^2 n/4}$. Therefore, Theorem 5 implies that under the same conditions,

$$\|\hat{\boldsymbol{\beta}}_{\mathrm{LR}} - \boldsymbol{\beta}_*\|_F \leq \varphi_1^{1/2}(\boldsymbol{\Sigma}_*)\left(\frac{16(t+d)\tilde{c}\|\boldsymbol{X}\|}{\nu\sqrt{n}}\right)\left(\sqrt{\frac{2k_2 r(p+q)}{n-2}} + \sqrt{\frac{2r}{n}}\right)$$

with probability at least $1 - \{8/(7+k_1)\}^{p+q} - 2e^{-(d-1)^2 n/4}$ as long as $\lambda\sqrt{r} \to 0$ as $n \to \infty$. ■

In our statement of Corollary 6(i), for example, we use that under **C1** and **A4**–**A6**, $\lambda\sqrt{|\mathcal{S}|} \to 0$ as $n \to \infty$ is implied by $|\mathcal{S}|\log(pq^k) = o(n)$.

B.5 Proofs of Lemmas

Proof of Lemma 11. We proceed with the following steps: first, we derive the subdifferential of $\boldsymbol{\beta} \mapsto \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_*$, then we show that this set is a singleton at $\boldsymbol{\beta}$ such that $\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}$ has q non-zero singular values. Let $\partial f(\boldsymbol{A})$ denote the subdifferential of a function f at $\boldsymbol{A}$.

To establish the subdifferential, we first apply the chain rule for subdifferentials:

$$\partial\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_* = \left\{-\boldsymbol{X}^\top \boldsymbol{H} : \boldsymbol{H} \in \partial\|\boldsymbol{B}\|_* \,|_{\boldsymbol{B}=\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta}}\right\}. \tag{27}$$

From Watson (1992), letting $(\boldsymbol{U}_B, \boldsymbol{D}_B, \boldsymbol{V}_B) = \mathrm{svd}(\boldsymbol{B})$, we have

$$\partial\|\boldsymbol{B}\|_* = \left\{\boldsymbol{U}_B\boldsymbol{V}_B^\top + \boldsymbol{W}_B : \boldsymbol{W}_B \in \mathbb{R}^{p\times q}, \|\boldsymbol{W}_B\| \leq 1, \boldsymbol{U}_B^\top\boldsymbol{W}_B = 0, \boldsymbol{W}_B\boldsymbol{V}_B = 0\right\}. \tag{28}$$

Hence, combining (27) and (28), with $(\boldsymbol{U}, \boldsymbol{D}, \boldsymbol{V}) = \mathrm{svd}(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})$,

$$\partial\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_* = \left\{-\boldsymbol{X}^\top\boldsymbol{U}\boldsymbol{V}^\top - \boldsymbol{X}^\top\boldsymbol{W} : \|\boldsymbol{W}\| \leq 1, \boldsymbol{U}^\top\boldsymbol{W} = 0, \boldsymbol{W}\boldsymbol{V} = 0\right\}.$$

However, when $\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}$ has q non-zero singular values, the only such $\boldsymbol{W}$ which can satisfy both $\boldsymbol{U}^\top\boldsymbol{W} = 0$ and $\boldsymbol{W}\boldsymbol{V} = 0$ is $\boldsymbol{W} = 0$. Thus, in this case, the subdifferential is the singleton $-\boldsymbol{X}^\top\boldsymbol{U}\boldsymbol{V}$. ■

Proof of Lemma 13. First, recall that the nuclear norm can be expressed

$$\|\boldsymbol{A}\|_* = \sup_{\|\boldsymbol{Q}\|\leq 1} \mathrm{tr}(\boldsymbol{Q}^\top\boldsymbol{A}).$$

By Lemma 11, under **A1** it suffices to show that

$$\frac{1}{\sqrt{n}}\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}_* - \boldsymbol{X}\boldsymbol{\Delta}\|_* - \frac{1}{\sqrt{n}}\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}_*\|_* + \frac{1}{\sqrt{n}}\mathrm{tr}(\boldsymbol{\Delta}^\top\boldsymbol{X}\boldsymbol{U}_\epsilon\boldsymbol{V}_\epsilon^\top) \geq \phi\|\boldsymbol{\Delta}\|_F^2. \tag{29}$$

Clearly,

$$\begin{aligned}
&\frac{1}{\sqrt{n}}\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}_* - \boldsymbol{X}\boldsymbol{\Delta}\|_* - \frac{1}{\sqrt{n}}\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}_*\|_* + \frac{1}{\sqrt{n}}\mathrm{tr}(\boldsymbol{\Delta}^\top\boldsymbol{X}^\top\boldsymbol{U}_\epsilon\boldsymbol{V}_\epsilon^\top)\\
&\qquad = \frac{1}{\sqrt{n}}\|\boldsymbol{\mathcal{E}} - \boldsymbol{X}\boldsymbol{\Delta}\|_* - \frac{1}{\sqrt{n}}\|\boldsymbol{\mathcal{E}}\|_* + \frac{1}{\sqrt{n}}\mathrm{tr}(\boldsymbol{\Delta}^\top\boldsymbol{X}^\top\boldsymbol{U}_\epsilon\boldsymbol{V}_\epsilon^\top)\\
&\qquad = \frac{1}{\sqrt{n}}\|\boldsymbol{U}_\epsilon\boldsymbol{D}_\epsilon\boldsymbol{V}_\epsilon^\top - \boldsymbol{X}\boldsymbol{\Delta}\|_* - \frac{1}{\sqrt{n}}\|\boldsymbol{\mathcal{E}}\|_* + \frac{1}{\sqrt{n}}\mathrm{tr}(\boldsymbol{\Delta}^\top\boldsymbol{X}^\top\boldsymbol{U}_\epsilon\boldsymbol{V}_\epsilon^\top)\\
&\qquad = \left[\sup_{\|\boldsymbol{Q}\|\leq 1}\frac{1}{\sqrt{n}}\mathrm{tr}\left\{\boldsymbol{Q}^\top(\boldsymbol{U}_\epsilon\boldsymbol{D}_\epsilon\boldsymbol{V}_\epsilon^\top - \boldsymbol{X}\boldsymbol{\Delta})\right\}\right] - \frac{1}{\sqrt{n}}\|\boldsymbol{\mathcal{E}}\|_* + \frac{1}{\sqrt{n}}\mathrm{tr}(\boldsymbol{\Delta}^\top\boldsymbol{X}^\top\boldsymbol{U}_\epsilon\boldsymbol{V}_\epsilon^\top)
\end{aligned}$$

and because $\|\boldsymbol{\mathcal{E}}\|_* = \mathrm{tr}(\boldsymbol{D}_\epsilon) = \mathrm{tr}(\boldsymbol{U}_\epsilon\boldsymbol{V}_\epsilon^\top\boldsymbol{V}_\epsilon\boldsymbol{D}_\epsilon\boldsymbol{U}_\epsilon^\top)$, the previous equality can be expressed

$$= \sup_{\|\boldsymbol{Q}\|\leq 1}\frac{1}{\sqrt{n}}\mathrm{tr}\left\{(\boldsymbol{Q} - \boldsymbol{U}_\epsilon\boldsymbol{V}_\epsilon^\top)^\top(\underbrace{\boldsymbol{U}_\epsilon\boldsymbol{D}_\epsilon\boldsymbol{V}_\epsilon^\top}_{=\boldsymbol{\mathcal{E}}} - \boldsymbol{X}\boldsymbol{\Delta})\right\}.$$

Thus, the desired inequality must hold by the definition of ϕ. ■

Proof of Lemma 14. For any $\delta \geq 0$, by the union bound we have

$$P\left(\frac{1}{\sqrt{n}}\|\boldsymbol{X}^\top\boldsymbol{S}\|_\infty \geq \delta\right) \leq \sum_{l=1}^{q} P\left(\frac{1}{\sqrt{n}}\|\boldsymbol{X}^\top\boldsymbol{S}_{\cdot,l}\|_\infty \geq \delta\right)$$

where $\boldsymbol{S}_{\cdot,l} \in S^{n-1}$ is the lth column of $\boldsymbol{S} \in V_q(n)$. Under the conditions of Lemma 14, Lemma 9(ii) suggests $\boldsymbol{S}_{\cdot,l}$ is uniformly distributed on S^{n-1}. Thus, we know that $\boldsymbol{S}_{\cdot,l} \sim \boldsymbol{g}/\|\boldsymbol{g}\|_2$, where $\boldsymbol{g} \sim \mathrm{N}_n(0, \boldsymbol{I}_n)$, so the above inequality implies

$$P\left(\frac{1}{\sqrt{n}}\|\boldsymbol{X}^\top \boldsymbol{S}\|_\infty \geq \delta\right) \leq q\, P\left(\frac{\|\boldsymbol{X}^\top \boldsymbol{g}\|_\infty}{\sqrt{n}\|\boldsymbol{g}\|_2} \geq \delta\right). \tag{30}$$

It remains only to bound the right hand side of (30). By an application of Lemma 19,

$$P\left(\frac{\|\boldsymbol{X}^\top \boldsymbol{g}\|_\infty}{\sqrt{n}\|\boldsymbol{g}\|_2} \geq \sqrt{\frac{2\log(2p/\alpha)}{n-1}}\right) \leq \alpha,$$

so that setting $\alpha = q^{-k}$ for fixed constant $k > 1$, as long as $2\log(2pq^k) < n-1$, we have

$$P\left(\frac{1}{\sqrt{n}}\|\boldsymbol{X}^\top \boldsymbol{S}\|_\infty \geq \sqrt{\frac{2\log(2pq^k)}{n-1}}\right) \leq q^{1-k},$$

from which our conclusion follows. ■

(31)

Proof of Lemma 15. Recall that under **C1**, $\|\boldsymbol{X}_{\cdot,j}\|_2 = \sqrt{n}$, so that each $\boldsymbol{X}_{\cdot,j}/\sqrt{n}$ is an element of S^{n-1}. By the union bound

$$P\left(\frac{1}{\sqrt{n}}\|\boldsymbol{X}^\top \boldsymbol{S}\|_{\infty,2} \geq \delta\right) \leq \sum_{j=1}^{p} P\left(\frac{1}{\sqrt{n}}\left\|\boldsymbol{S}^\top \boldsymbol{X}_{\cdot,j}\right\|_2 \geq \delta\right)$$

for all $\delta \geq 0$. Hence, we need to establish a concentration inequality for the random quantity $\|\boldsymbol{S}^\top \boldsymbol{a}\|_2$ where $\boldsymbol{a} \in S^{n-1}$ is a fixed unit vector and $\boldsymbol{S}$ is a random matrix uniformly distributed on $V_q(n)$. Applying Lemma 24, as long as $\alpha > 4\{\pi/(n-2)\}^{1/2}$,

$$P\left(\frac{1}{\sqrt{n}}\left\|\boldsymbol{S}^\top \boldsymbol{X}_{\cdot,j}\right\|_2 \geq \alpha + \sqrt{\frac{q}{n}}\right) \leq \exp\left(-\frac{(n-2)\alpha^2}{4}\right).$$

Thus, setting $\alpha = \{4k\log p/(n-2)\}^{1/2}$ for constant $k > 4\pi/\log p$, we have

$$P\left(\frac{1}{\sqrt{n}}\left\|\boldsymbol{S}^\top \boldsymbol{X}_{\cdot,j}\right\|_2 \geq \sqrt{\frac{4k\log p}{n-2}} + \sqrt{\frac{q}{n}}\right) \leq \exp\left(-k\log p\right).$$

Therefore, with $\delta = \{4k\log p/(n-2)\}^{1/2} + (q/n)^{1/2}$, we can conclude

$$P\left(\frac{1}{\sqrt{n}}\|\boldsymbol{X}^\top \boldsymbol{S}\|_{\infty,2} \geq \sqrt{\frac{4k\log p}{n-2}} + \sqrt{\frac{q}{n}}\right) \leq p\exp\left(-k\log p\right) = p^{1-k}. \quad ■$$

Proof of Lemma 16. By identical arguments used to derive (F.3) of the Supplementary Material to Negahban and Wainwright (2011), we have that for $\delta \geq 0$

$$P\left(\frac{1}{\sqrt{n}}\|\boldsymbol{X}^\top \boldsymbol{S}\| \geq 4\delta\right) \leq 8^{p+q} \max_{\boldsymbol{v}_a, \boldsymbol{t}_b} P\left(\frac{1}{\sqrt{n}}|\boldsymbol{t}_b^\top \boldsymbol{X}^\top \boldsymbol{S}\boldsymbol{v}_a| \geq \delta\right) \tag{32}$$

where $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \dots, \boldsymbol{v}_A\}$ and $\{\boldsymbol{t}_1, \boldsymbol{t}_2, \dots, \boldsymbol{t}_B\}$ are 1/4 coverings of S^{q-1} and S^{p-1}, respectively. Thus, following Negahban and Wainwright (2011), we need to bound the random scalar $|\boldsymbol{t}^\top \boldsymbol{X}^\top \boldsymbol{S}\boldsymbol{v}|/\sqrt{n}$ for arbitrary (but fixed) vectors $\boldsymbol{t} \in S^{p-1}$, $\boldsymbol{v} \in S^{q-1}$. First, note that the random vector $\boldsymbol{z} = \boldsymbol{S}\boldsymbol{v}$ has a uniform distribution on S^{n-1} (see Lemma 9). Hence, we need only concern ourselves with $|\boldsymbol{t}^\top \boldsymbol{X}^\top \boldsymbol{z}|$ where $\boldsymbol{z}$ is uniformly distributed on S^{n-1}. Because $\mathrm{E}|\boldsymbol{t}^\top \boldsymbol{X}^\top \boldsymbol{z}| \leq \|\boldsymbol{X}\boldsymbol{t}\|_2/\sqrt{n}$ (e.g., see Lemma 5.3.2(a) of Vershynin (2018) and apply Jensen's inequality), and because $\boldsymbol{z} \mapsto \|\boldsymbol{t}^\top \boldsymbol{X}^\top \boldsymbol{z}\|_2/\sqrt{n}$ is Lipschitz with constant $\|\boldsymbol{X}\boldsymbol{t}\|_2/\sqrt{n}$, applying Lemma 23 we have

$$P\left(\frac{1}{\sqrt{n}}|\boldsymbol{t}^\top \boldsymbol{X}^\top \boldsymbol{z}| \geq \alpha + \frac{1}{n}\|\boldsymbol{X}\boldsymbol{t}\|_2\right) \leq \exp\left(-\frac{(n-2)n\alpha^2}{4\|\boldsymbol{X}\boldsymbol{t}\|_2^2}\right),$$

as long as $\alpha > 4\{\pi/(n-2)\}^{1/2}$. Since $\|\boldsymbol{X}\boldsymbol{t}\|_2 \leq \|\boldsymbol{X}\|$ for all vectors $\boldsymbol{t} \in S^{p-1}$, the above inequality implies

$$P\left(\frac{1}{\sqrt{n}}|\boldsymbol{t}^\top \boldsymbol{X}^\top \boldsymbol{z}| \geq \alpha + \frac{1}{n}\|\boldsymbol{X}\|\right) \leq \exp\left(-\frac{(n-2)n\alpha^2}{4\|\boldsymbol{X}\|^2}\right).$$

By setting $\alpha = \|\boldsymbol{X}\|[k_2(p+q)/\{n(n-2)\}]^{1/2}$ for constant $k_2 > 0$, we have

$$P\left\{\frac{1}{\sqrt{n}}|\boldsymbol{t}^\top \boldsymbol{X}^\top \boldsymbol{z}| \geq \frac{\|\boldsymbol{X}\|}{\sqrt{n}}\left(\sqrt{\frac{k_2(p+q)}{n-2}} + \sqrt{\frac{1}{n}}\right)\right\} \leq \exp\left(-\frac{k_2(p+q)}{4}\right). \tag{33}$$

Hence, using (32) and (33),

$$\begin{aligned} P\left\{\frac{1}{\sqrt{n}}\|\boldsymbol{X}^\top \boldsymbol{S}\| \geq 4\frac{\|\boldsymbol{X}\|}{\sqrt{n}}\left(\sqrt{\frac{k_2(p+q)}{n-2}} + \sqrt{\frac{1}{n}}\right)\right\}& \\ \leq 8^{p+q} P\left\{\frac{1}{\sqrt{n}}|\boldsymbol{t}^\top \boldsymbol{X}^\top \boldsymbol{z}| \geq \frac{\|\boldsymbol{X}\|}{\sqrt{n}}\left(\sqrt{\frac{k_2(p+q)}{n-2}} + \sqrt{\frac{1}{n}}\right)\right\}& \\ \leq 8^{p+q}\left\{\exp\left(-\frac{k_2(p+q)}{4}\right)\right\}&. \end{aligned}$$

The conclusion follows by taking $k_2 = 4\log(7+k_1)$ for $k_1 > 1$ large enough that $k_2\|\boldsymbol{X}\|^2(p+q) > 16n\pi$, in which case

$$\begin{aligned} P\left\{\frac{1}{\sqrt{n}}\|\boldsymbol{X}^\top \boldsymbol{S}\| \geq 4\frac{\|\boldsymbol{X}\|}{\sqrt{n}}\left(\sqrt{\frac{k_2(p+q)}{n-2}} + \sqrt{\frac{1}{n}}\right)\right\}& \\ \leq 8^{p+q}\left\{\exp\left(-\frac{k_2(p+q)}{4}\right)\right\} = \left(\frac{8}{7+k_1}\right)^{p+q}&. \quad \blacksquare \end{aligned}$$

Proof of Lemma 17. To simplify notation, let

$$\mathcal{H}(\boldsymbol{\Delta}) = \frac{1}{\sqrt{n}}\|\boldsymbol{\mathcal{E}} - \boldsymbol{X}\boldsymbol{\Delta}\|_* - \frac{1}{\sqrt{n}}\|\boldsymbol{\mathcal{E}}\|_* + \frac{1}{\sqrt{n}}\mathrm{tr}(\boldsymbol{\Delta}^\top \boldsymbol{X}^\top \boldsymbol{U}_\epsilon \boldsymbol{V}_\epsilon^\top).$$

First, letting $\boldsymbol{U}$ and $\boldsymbol{V}$ denote the left and right singular vectors of $\boldsymbol{\mathcal{E}}$ (momentarily omitting the subscript ϵ for ease of display), Lemma 20 and **A6** imply that for n sufficiently large (so that $n \geq q$),

$$\mathcal{H}(\boldsymbol{\Delta}) = \frac{1}{2}\sum_{i=1}^{q}\sum_{j=1}^{q}\frac{(\boldsymbol{u}_j^\top \boldsymbol{X}\boldsymbol{\Delta}\boldsymbol{v}_i - \boldsymbol{u}_i^\top \boldsymbol{X}\boldsymbol{\Delta}\boldsymbol{v}_j)^2}{2\sqrt{n}\{\sigma_i(\boldsymbol{\mathcal{E}}) + \sigma_j(\boldsymbol{\mathcal{E}})\}} + \frac{1}{2}\sum_{k=q+1}^{n}\sum_{j=1}^{q}\frac{(\boldsymbol{u}_k^\top \boldsymbol{X}\boldsymbol{\Delta}\boldsymbol{v}_j)^2}{\sqrt{n}\sigma_j(\boldsymbol{\mathcal{E}})} + o\left(\frac{\|\boldsymbol{X}\boldsymbol{\Delta}\|_F^2}{n}\right)$$

where $\boldsymbol{u}_j$ denotes the jth column of $\boldsymbol{U} \in \mathbb{R}^{n\times q}$ for $j \in [q]$, $\boldsymbol{v}_k$ denotes the kth column of $\boldsymbol{V} \in \mathbb{R}^{q\times q}$ for $k \in [q]$ and $\boldsymbol{u}_l$ denotes the $(l-q)$th column of $\boldsymbol{U}_0 \in \mathbb{R}^{n\times(n-q)}$ for $l \in \{q+1, q+2, \dots, n\}$ where $\boldsymbol{U}_0^\top \boldsymbol{U} = 0$ and $\boldsymbol{U}_0^\top \boldsymbol{U}_0 = \boldsymbol{I}_{n-q}$. On $\mathcal{B}_d$, we have that for each $j \in [q]$,

$$\sigma_j(\boldsymbol{\mathcal{E}}) \leq \sigma_1(\boldsymbol{\mathcal{E}}) \leq \sqrt{n}(t+d)\varphi_1^{1/2}(\boldsymbol{\Sigma}_*),$$

from which it follows that

$$\mathcal{H}(\boldsymbol{\Delta}) \geq \underbrace{\frac{1}{2}\sum_{i=1}^{q}\sum_{j=1}^{q}\frac{(\boldsymbol{u}_j^\top \boldsymbol{X}\boldsymbol{\Delta}\boldsymbol{v}_i - \boldsymbol{u}_i^\top \boldsymbol{X}\boldsymbol{\Delta}\boldsymbol{v}_j)^2}{4n(t+d)\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)} + \frac{1}{2}\sum_{k=q+1}^{n}\sum_{j=1}^{q}\frac{(\boldsymbol{u}_k^\top \boldsymbol{X}\boldsymbol{\Delta}\boldsymbol{v}_j)^2}{n(t+d)\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)}}_{T_1} + \underbrace{o\left(\frac{\|\boldsymbol{X}\boldsymbol{\Delta}\|_F^2}{n}\right)}_{T_2}.$$

By assumption **A5** and Lemma 12, $\|\boldsymbol{X}\boldsymbol{\Delta}\|_F^2/n \leq \bar{v}\|\boldsymbol{\Delta}\|_F^2$, where $\bar{v}$ is a finite constant. Thus, because $\|\boldsymbol{\Delta}\|_F^2 \to 0$ by assumption, there exists an N such that for all $n > N$, $T_2 \geq -\nu\|\boldsymbol{\Delta}\|_F^2/\{4(t+d)\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)\}$. Also by assumption **A5** and Lemma 12, $T_1 \geq \nu\|\boldsymbol{\Delta}\|_F^2/\{2(t+d)\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)\}$. Hence, for n sufficiently large,

$$\mathcal{H}(\boldsymbol{\Delta}) \geq T_1 + T_2 \geq \|\boldsymbol{\Delta}\|_F^2 \frac{\nu}{4(t+d)\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)},$$

which completes the proof. ■

Proof of Lemma 18. Let $\boldsymbol{G} \in \mathbb{R}^{n\times q}$ be a matrix with independent and identically distributed standard normal entries. Hence, for $\delta \geq 0$, we can write

$$\begin{aligned} P\left(\frac{1}{\sqrt{n}}\|\boldsymbol{\mathcal{E}}\| \leq \delta\right) &= P\left(\frac{1}{\sqrt{n}}\|\boldsymbol{G}\boldsymbol{\Sigma}_*^{1/2}\| \leq \delta\right) \\ &\geq P\left(\frac{1}{\sqrt{n}}\|\boldsymbol{G}\|\|\boldsymbol{\Sigma}_*^{1/2}\| \leq \delta\right) = P\left(\frac{1}{\sqrt{n}}\|\boldsymbol{G}\| \leq \frac{\delta}{\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)}\right) \qquad (34) \end{aligned}$$

Then, applying Lemma 21, for $\alpha \geq 0$,

$$P\left(\frac{1}{\sqrt{n}}\|\boldsymbol{G}\| > \sqrt{\frac{q}{n}} + 1 + \frac{\alpha}{\sqrt{n}}\right) \leq 2e^{-\alpha^2/2}.$$

so that taking $\alpha = d\sqrt{\frac{n}{2}}$ for $d > 0$, and recalling that $\sqrt{q/n} \to t$ for $t \in [0, 1)$, for n sufficiently large,

$$P\left(\frac{1}{\sqrt{n}}\|\boldsymbol{G}\| > (t + d + 1)\right) \leq 2e^{-d^2 n/4}.$$

Finally, setting $\delta = (t + d + 1)\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)$, applying the previous inequailty to (34), for n sufficiently large

$$P\left(\frac{1}{\sqrt{n}}\|\boldsymbol{\mathcal{E}}\| \leq (t + d + 1)\varphi_1^{1/2}(\boldsymbol{\Sigma}_*)\right) \geq 1 - 2e^{-d^2 n/4}.$$

The result follows by replacing d with $\tilde{d} - 1$ for $\tilde{d} > 1$. ■

B.6 Technical Lemmas

In this section, we provide several technical lemmas which were used in the previous sections.

Lemma 19 (Lemma 8.2, van de Geer (2016)) *Let $\boldsymbol{g} \sim \mathrm{N}_n(0, \boldsymbol{I}_n)$ and suppose $\boldsymbol{X} \in \mathbb{R}^{n \times q}$ has columns $\boldsymbol{X}_{\cdot,j}$ such that $\|\boldsymbol{X}_{\cdot,j}\|_2 = \sqrt{n}$ for $j \in [p]$. If $0 < \delta < 1$ and $2\log(2p/\delta) < n - 1$, then*

$$P\left(\frac{\|\boldsymbol{X}^\top \boldsymbol{g}\|_\infty}{\sqrt{n}\|\boldsymbol{g}\|_2} \geq \sqrt{\frac{2\log(2p/\delta)}{n-1}}\right) \leq \delta.$$

Lemma 20 (Proposition 66, Dubois et al. (2019)) *Let $n \geq q$ and let $\boldsymbol{A} \in \mathbb{R}^{n \times q}$ be a full rank matrix with $(\boldsymbol{U}, \boldsymbol{D}, \boldsymbol{V}) = \mathrm{svd}(\boldsymbol{A})$. Let $\boldsymbol{U}_0 \in \mathbb{R}^{n \times (n-q)}$ such that $\boldsymbol{U}_0^\top \boldsymbol{U}_0 = \boldsymbol{I}_{n-q}$ and $\boldsymbol{U}^\top \boldsymbol{U}_0 = 0$. Let $\boldsymbol{u}_i$ denote the ith column of $\boldsymbol{U}$ for $i \in [q]$, $\boldsymbol{v}_j$ the jth column of $\boldsymbol{V}$ for $j \in [q]$, ρ_j the jth diagonal entry of $\boldsymbol{D}$ (i.e., jth largest singular value of $\boldsymbol{A}$) for $j \in [q]$, and $\boldsymbol{u}_k$ the $(k-q)$th the column of $\boldsymbol{U}_0$ for $k \in \{q+1, q+2, \dots, n\}$. Then, for any $\boldsymbol{C} \in \mathbb{R}^{n \times q}$, it follows that*

$$\begin{aligned}\|\boldsymbol{A} - \boldsymbol{C}\|_* = \|\boldsymbol{A}\|_* - \mathrm{tr}(\boldsymbol{C}^\top \boldsymbol{U}\boldsymbol{V}^\top) + \frac{1}{2}\sum_{i=1}^{q}\sum_{j=1}^{q}\frac{(\boldsymbol{u}_j^\top \boldsymbol{C}\boldsymbol{v}_i - \boldsymbol{u}_i^\top \boldsymbol{C}\boldsymbol{v}_j)^2}{2(\rho_i + \rho_j)} \\ + \frac{1}{2}\sum_{k=q+1}^{n}\sum_{j=1}^{q}\frac{(\boldsymbol{u}_k^\top \boldsymbol{C}\boldsymbol{v}_j)^2}{\rho_j} + o(\|\boldsymbol{C}\|_F^2).\end{aligned}$$

Lemma 21 (Corollary 5.35, Vershynin (2012)) *Let $\boldsymbol{G} \in \mathbb{R}^{n \times q}$ be a matrix with independent and identically distributed standard normal entries. If $\delta \geq 0$, then*

$$P\left(\|\boldsymbol{G}\| \leq \sqrt{q} + \sqrt{n} + \delta\right) \geq 1 - 2\mathrm{exp}\left(-\frac{\delta^2}{2}\right).$$

Next, we provide a general result on the concentration of Lipschitz functions $f : S^{n-1} \to \mathbb{R}$. In order to establish this result, we need a preliminary lemma regarding the concentration of a function f near its median on S^{n-1}.

Lemma 22 (Theorem 3.4.1, Raginsky and Sason (2013)) *Let $f : S^{n-1} \to \mathbb{R}$ be an η-Lipschitz function and let $\boldsymbol{z}$ be a random vector having the uniform distribution on S^{n-1}. If $\delta \geq 0$, then*

$$\text{(i)} P(f(\boldsymbol{z}) \geq M_f + \delta) \leq \exp\left(-\frac{(n-2)\delta^2}{2\eta^2}\right),$$

where M_f is the median of f with respect to the uniform probability measure on S^{n-1}. Moreover,

$$\text{(ii)} |M_f - \mathrm{E}f(\boldsymbol{z})| \leq \eta\sqrt{\frac{\pi}{n-2}}.$$

For (ii), see the proof of Corollary 5.4 of Meckes (2019). This leads to our main lemma, which we use throughout the remainder of this section.

Lemma 23 *Let $f : S^{n-1} \to \mathbb{R}$ be an η-Lipschitz function and let $\boldsymbol{z}$ be a random vector having the uniform distribution on S^{n-1}. If $\delta > 4\eta\{\pi/(n-2)\}^{1/2}$, then*

$$P\left(f(\boldsymbol{z}) - \mathrm{E}f(\boldsymbol{z}) \geq \delta\right) \leq \exp\left(\frac{-(n-2)\delta^2}{4\eta^2}\right).$$

Note that Lemma 23 would hold if we had $\delta > \eta\left\{\sqrt{2}/(\sqrt{2}-1)\right\}\{\pi/(n-2)\}^{1/2}$. We use $\delta > 4\eta\{\pi/(n-2)\}^{1/2}$ in Lemma 23 for ease of display.

Proof of Lemma 23. We combine the two results from Lemma 22 to obtain a bound on $P(f(\boldsymbol{z}) - \mathrm{E}f(\boldsymbol{z}) \geq \delta)$. First,

$$\begin{aligned} P(f(\boldsymbol{z}) - \mathrm{E}f(\boldsymbol{z}) \geq \delta) &= P(f(\boldsymbol{z}) - M_f \geq \delta + \mathrm{E}f(\boldsymbol{z}) - M_f) \\ &\leq P(f(\boldsymbol{z}) - M_f \geq \delta - |\mathrm{E}f(\boldsymbol{z}) - M_f|) \end{aligned}$$

so that an application of Lemma (22)(ii), $\delta > \eta\left\{\sqrt{2}/(\sqrt{2}-1)\right\}\{\pi/(n-2)\}^{1/2}$—which is implied by $\delta > 4\eta\{\pi/(n-2)\}^{1/2}$—and Lemma (22)(i), respectively, yield

$$\begin{aligned} P(f(\boldsymbol{z}) - \mathrm{E}f(\boldsymbol{z}) \geq \delta) &\leq P\left(f(\boldsymbol{z}) - M_f \geq \delta - \eta\sqrt{\frac{\pi}{n-2}}\right) \\ &\leq P\left(f(\boldsymbol{z}) - M_f \geq \frac{\delta}{\sqrt{2}}\right) \\ &\leq \exp\left(-\frac{(n-2)\delta^2}{4\eta^2}\right). \quad \blacksquare \end{aligned}$$

This result can be generalized to quantities of the form $\|\boldsymbol{S}^\top \boldsymbol{a}\|_2$ where $\boldsymbol{S}$ is uniformly distributed on $V_q(n)$.

Lemma 24 *Let $\boldsymbol{a} \in S^{n-1}$ be fixed and let $\boldsymbol{S}$ be a random matrix having the uniform distribution on $V_q(n)$. If $\delta > 4\{\pi/(n-2)\}^{1/2}$, then*

$$P\left(\|\boldsymbol{S}^\top \boldsymbol{a}\|_2 \geq \delta + \sqrt{\frac{q}{n}}\right) \leq \exp\left(-\frac{(n-2)\delta^2}{4}\right).$$

Proof of Lemma 24. We apply the same arguments as in the proof of Lemma 4.2 of Lyubarskii and Vershynin (2010). Specifically, let $\boldsymbol{O}$ be a random matrix uniformly distributed on $O(n)$. We know then that $\boldsymbol{S} \sim \boldsymbol{O}\boldsymbol{\mathcal{P}}_q$ where $\boldsymbol{\mathcal{P}}_q \in \mathbb{R}^{n \times q}$ is the first q columns of $\boldsymbol{I}_n$. Thus, $\boldsymbol{S}^\top \boldsymbol{a} \sim \boldsymbol{\mathcal{P}}_q^\top \boldsymbol{O}^\top \boldsymbol{a}$ and consequently, because $\boldsymbol{O}^\top \boldsymbol{a}$ is uniformly distributed on S^{n-1} (Lemma 9), for all $\delta \geq 0$ it follows that

$$P(\|\boldsymbol{S}^\top \boldsymbol{a}\|_2 \geq \delta) = P(\|\boldsymbol{\mathcal{P}}_q^\top \boldsymbol{z}\|_2 \geq \delta)$$

for random vector $\boldsymbol{z}$ having the uniform distribution on S^{n-1}. Hence, applying Lemma 23 and using that $\boldsymbol{z} \mapsto \|\boldsymbol{\mathcal{P}}_q^\top \boldsymbol{z}\|_2$ is 1-Lipschitz,

$$P\left\{\|\boldsymbol{\mathcal{P}}_q^\top \boldsymbol{z}\|_2 \geq \alpha + \mathrm{E}(\|\boldsymbol{\mathcal{P}}_q^\top \boldsymbol{z}\|_2)\right\} \leq \exp\left(-\frac{(n-2)\alpha^2}{4}\right).$$

Then, again applying a result from the proof of Lemma 4.2 of Lyubarskii and Vershynin (2010), $\mathrm{E}(\|\boldsymbol{\mathcal{P}}_q^\top \boldsymbol{z}\|_2) \leq (q/n)^{1/2}$, so that finally, applying Lemma 23, we conclude

$$P\left(\|\boldsymbol{S}^\top \boldsymbol{a}\|_2 \geq \alpha + \sqrt{\frac{q}{n}}\right) \leq \exp\left(-\frac{(n-2)\alpha^2}{4}\right),$$

as long as $\alpha > 4\{\pi/(n-2)\}^{1/2}$. ■

References

Amir Beck and Marc Teboulle. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE Transactions on Image Processing*, 18 (11):2419–2434, 2009.

Alexandre Belloni, Victor Chernozhukov, and Lie Wang. Square-root lasso: pivotal recovery of sparse signals via conic programming. *Biometrika*, 98(4):791–806, 2011.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011.

Florentina Bunea, Johannes Lederer, and Yiyuan She. The group square-root lasso: Theoretical properties and fast algorithms. *IEEE Transactions on Information Theory*, 60(2): 1313–1325, 2014.

Gabriel Camano-Garcia. *Statistics on Stiefel manifolds.* PhD thesis, Iowa State University, 2006.

Antonin Chambolle and Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40 (1):120–145, 2011.

Kun Chen, Hongbo Dong, and Kung-Sik Chan. Reduced rank regression via adaptive nuclear norm penalization. *Biometrika*, 100(4):901–920, 2013.

Patrick L. Combettes and Jean-Christophe Pesquet. *Proximal Splitting Methods in Signal Processing*, pages 185–212. Springer New York, New York, NY, 2011. ISBN 978-1-4419-9569-8. doi: 10.1007/978-1-4419-9569-8_10. URL https://doi.org/10.1007/978-1-4419-9569-8_10.

Wei Deng and Wotao Yin. On the global and linear convergence of the generalized alternating direction method of multipliers. *Journal of Scientific Computing*, 66(3):889–916, 2016.

Alexis Derumigny. Improved bounds for square-root lasso and square-root slope. *Electronic Journal of Statistics*, 12(1):741–766, 2018.

Benjamin Dubois, Jean-François Delmas, and Guillaume Obozinski. Fast algorithms for sparse reduced-rank regression. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2415–2424. PMLR, 2019.

Morris L. Eaton. Group invariance applications in statistics. *Regional Conference Series in Probability and Statistics*, 1:i–133, 1989. ISSN 19355912. URL http://www.jstor.org/stable/4153172.

Christopher Fougner and Stephen Boyd. *Parameter Selection and Preconditioning for a Graph Form Solver*, pages 41–61. Springer International Publishing, Cham, 2018. ISBN 978-3-319-67068-3. doi: 10.1007/978-3-319-67068-3_4. URL https://doi.org/10.1007/978-3-319-67068-3_4.

Michael Grant and Stephen Boyd. CVX: Matlab software for disciplined convex programming, version 2.1, 2014.

Yuwen Gu, Jun Fan, Lingchen Kong, Shiqian Ma, and Hui Zou. ADMM for high-dimensional sparse penalized quantile regression. *Technometrics*, 60(3):319–331, 2018.

Kenneth Lange. *MM Optimization Algorithms.* SIAM, Philadelphia PA, 2016.

Jason D. Lee, Yuekai Sun, and Jonathan E. Taylor. On model selection consistency of regularized M-estimators. *Electronic Journal of Statistics*, 9(1):608–642, 2015.

Wonyul Lee and Yufeng Liu. Simultaneous multiple response regression and inverse covariance matrix estimation via penalized Gaussian maximum likelihood. *Journal of Multivariate Analysis*, 111:241–255, 2012.

Xinguo Li, Haoming Jiang, Jarvis Haupt, Raman Arora, Han Liu, Mingyi Hong, and Tuo Zhao. On fast convergence of proximal algorithms for sqrt-lasso optimization: Don't worry about its nonsmooth loss function. In *Uncertainty in Artificial Intelligence*, pages 49–59. PMLR, 2020.

Han Liu, Lie Wang, and Tuo Zhao. Calibrated multivariate regression with application to neural semantic basis discovery. *Journal of Machine Learning Research*, 16:1579–1606, 2015.

Karim Lounici, Massimiliano Pontil, Sara Van De Geer, and Alexandre B. Tsybakov. Oracle inequalities and optimal inference under group sparsity. *The Annals of Statistics*, 39(4): 2164–2204, 2011.

Yurii Lyubarskii and Roman Vershynin. Uncertainty principles and vector quantization. *IEEE Transactions on Information Theory*, 56(7):3491–3501, 2010.

Mathurin Massias, Olivier Fercoq, Alexandre Gramfort, and Joseph Salmon. Generalized concomitant multi-task lasso for sparse multimodal regression. In *International Conference on Artificial Intelligence and Statistics*, pages 998–1007. PMLR, 2018.

Mathurin Massias, Quentin Bertrand, Alexandre Gramfort, and Joseph Salmon. Support recovery and sup-norm convergence rates for sparse pivotal estimation. In *International Conference on Artificial Intelligence and Statistics*, pages 2655–2665. PMLR, 2020.

Pertti Mattila. *Geometry of Sets and Measures in Euclidean Spaces: Fractals and Rectifiability*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 1995. doi: 10.1017/CBO9780511623813.

Elizabeth S. Meckes. *The Random Matrix Theory of the Classical Compact Groups*. Cambridge Tracts in Mathematics. Cambridge University Press, 2019. doi: 10.1017/9781108303453.

Aaron J. Molstad, Wei Sun, and Li Hsu. A covariance-enhanced approach to multitissue joint eqtl mapping with application to transcriptome-wide association studies. *The Annals of Applied Statistics*, 15(2):998–1016, 2021a.

Aaron J. Molstad, Guangwei Weng, Charles R. Doss, and Adam J. Rothman. An explicit mean-covariance parameterization for multivariate response linear regression. *Journal of Computational and Graphical Statistics*, 30(3):612–621, 2021b.

Sahand Negahban and Martin J. Wainwright. Estimation of (near) low-rank matrices with noise and high-dimensional scaling. *The Annals of Statistics*, 39(2):1069–1097, 2011.

Sahand N. Negahban, Pradeep Ravikumar, Martin J. Wainwright, and Bin Yu. A unified framework for high-dimensional analysis of M-estimators with decomposable regularizers. *Statistical Science*, 27(4):538–557, 2012.

Guillaume Obozinski, Martin J. Wainwright, and Michael I. Jordan. Support union recovery in high-dimensional multivariate regression. *The Annals of Statistics*, 39(1):1–47, 2011.

Neal Parikh and Stephen Boyd. Block splitting for distributed optimization. *Mathematical Programming Computation*, 6(1):77–102, 2014a.

Neal Parikh and Stephen Boyd. Proximal algorithms. *Foundations and Trends in Optimization*, 1(3):127–239, 2014b.

Nicholas G. Polson, James G. Scott, and Brandon T. Willard. Proximal algorithms in statistics and machine learning. *Statistical Science*, 30(4):559–581, 2015.

Bradley S. Price and Ben Sherwood. A cluster elastic net for multivariate regression. *Journal of Machine Learning Research*, 18(1):8685–8723, 2017.

Maxim Raginsky and Igal Sason. Concentration of measure inequalities in information theory, communications, and coding. *Foundations and Trends in Communications and Information Theory*, 10(1-2):1–246, 2013. ISSN 1567-2190. doi: 10.1561/0100000064. URL `http://dx.doi.org/10.1561/0100000064`.

Garvesh Raskutti, Martin J. Wainwright, and Bin Yu. Restricted eigenvalue properties for correlated gaussian designs. *Journal of Machine Learning Research*, 11:2241–2259, 2010.

Gregory C. Reinsel and Raja P. Velu. *Multivariate reduced-rank regression*, volume 136 of *Lecture Notes in Statistics*. Springer-Verlag, New York, 1998. ISBN 0-387-98601-4. doi: 10.1007/978-1-4757-2853-8. URL `https://doi.org/10.1007/978-1-4757-2853-8`. Theory and applications.

Adam J. Rothman, Elizaveta Levina, and Ji Zhu. Sparse multivariate regression with covariance estimation. *Journal of Computational and Graphical Statistics*, 19(4):947–962, 2010.

Benjamin Stucky. *Asymptotic confidence regions and sharp oracle results under structured sparsity*. PhD thesis, ETH Zurich, 2017.

Tingni Sun and Cun-Hui Zhang. Scaled sparse linear regression. *Biometrika*, 99(4):879–898, 2012.

Xiaoying Tian, Joshua R. Loftus, and Jonathan E. Taylor. Selective inference with unknown variance via the square-root lasso. *Biometrika*, 105(4):755–768, 2018.

Berwin A. Turlach, William N. Venables, and Stephen J. Wright. Simultaneous variable selection. *Technometrics*, 47(3):349–363, 2005.

Sara van de Geer. *Estimation and testing under sparsity*, volume 2159 of *Lecture Notes in Mathematics*. Springer, [Cham], 2016. ISBN 978-3-319-32773-0; 978-3-319-32774-7. doi: 10.1007/978-3-319-32774-7. URL `https://doi.org/10.1007/978-3-319-32774-7`. Lecture notes from the 45th Probability Summer School held in Saint-Four, 2015, École d'Été de Probabilités de Saint-Flour. [Saint-Flour Probability Summer School].

Sara Van de Geer and Benjamin Stucky. χ2-confidence sets in high-dimensional regression. In *Statistical Analysis for High-Dimensional Data*, pages 279–306. Springer, 2016.

Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. In *Compressed sensing*, pages 210–268. Cambridge Univ. Press, Cambridge, 2012.

Roman Vershynin. *High-dimensional probability*, volume 47 of *Cambridge Series in Statistical and Probabilistic Mathematics*. Cambridge University Press, Cambridge, 2018. ISBN 978-1-108-41519-4. doi: 10.1017/9781108231596. URL `https://doi.org/10.1017/9781108231596`. An introduction with applications in data science, With a foreword by Sara van de Geer.

Martin J. Wainwright. Sharp thresholds for high-dimensional and noisy sparsity recovery using l1-constrained quadratic programming (lasso). *IEEE Transactions on Information Theory*, 55(5):2183–2202, 2009.

Martin J. Wainwright. Structured regularizers for high-dimensional problems: Statistical and computational issues. *Annual Review of Statistics and Its Application*, 1:233–253, 2014.

Junhui Wang. Joint estimation of sparse multivariate regression and conditional graphical models. *Statistica Sinica*, 25(3):831–851, 2015.

G. Alistair Watson. Characterization of the subdifferential of some matrix norms. *Linear Algebra and its Applications*, 170:33–45, 1992.

John N. Weinstein, Eric A. Collisson, Gordon B. Mills, Kenna R. Mills Shaw, Brad A. Ozenberger, Kyle Ellrott, Ilya Shmulevich, Chris Sander, Joshua M. Stuart, and Cancer Genome Atlas Research Network. The cancer genome atlas pan-cancer analysis project. *Nature Genetics*, 45(10):1113, 2013.

Daniela M. Witten and Robert Tibshirani. Covariance-regularized regression and classification for high dimensional problems. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71(3):615–636, 2009.

Jianxin Yin and Hongzhe Li. A sparse conditional gaussian graphical model for analysis of genetical genomics data. *The Annals of Applied Statistics*, 5(4):2630, 2011.

Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.

Ming Yuan, Ali Ekici, Zhaosong Lu, and Renato Monteiro. Dimension reduction and coefficient estimation in multivariate linear regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(3):329–346, 2007.

Arnold Zellner. An efficient method of estimating seemingly unrelated regressions and tests for aggregation bias. *Journal of the American Statistical Association*, 57(298):348–368, 1962.