Winter 2019

Estimation and Variable Selection in High-Dimensional Settings with Mismeasured Observations

Michael Byrd
Southern Methodist University, mcbyrd@smu.edu

ESTIMATION AND VARIABLE SELECTION IN HIGH-DIMENSIONAL SETTINGS WITH MISMEASURED OBSERVATIONS

Approved by:

Monnie McGee
Associate Professor of Statistical Science

Jing Cao
Associate Professor of Statistical Science

Chul Moon
Assistant Professor of Statistical Science

Yunkai Zhou
Associate Professor of Mathematics

ESTIMATION AND VARIABLE SELECTION IN HIGH-DIMENSIONAL SETTINGS WITH MISMEASURED OBSERVATIONS

A DissertationPresented to the Graduate Faculty of the

Dedman College

Southern Methodist University

in

Partial Fulfillment of the Requirements

for the degree of

Doctor of Philosophy

with a

Major in Statistical Science

by

Michael Craig Byrd Jr.

B.S., Economics, University of North Texas

December 21, 2019

ACKNOWLEDGMENTS

First, I would like to thank my family for always supporting and encouraging me from the beginning to end of this long process. Many things have happened along the way, and I truly doubt that it would have been possible to reach the end goal without their help. In particular, I thank my parents, Mike and Helen Byrd, whom have offered support at every step along the way. I also thank my son, Lukas Byrd, for always being good for a laugh. He may not know it, but he has kept me sane these past years.

I have had the pleasure of making many friends during my time at SMU. Specifically, I would like to shout out Linh Nghiem. After having worked on three separate research projects and conversing a countless number of times, I can finally spell your name without looking. I would also like to thank Thomas Crutcher and Emily Boak for the good times and timely biology factoids.

I would also like to express my deepest gratitude to the faculty in the Statistical Sciences department. In particular, I would like to thank my advisor, Monnie McGee, for providing lively and insightful discussion. Monnie was always supportive and understanding, even when things were not working, which helped curve the unrelenting pressure over these past few years. I would also like to thank Jing Cao and Nelis Potgieter, with whom I have learned much from in collaboration with other projects.

During my time at SMU I have had the opportunity to gain experience in multiple roles. The department has given me the opportunity to not only be a teaching assistant, but also teach my own sections. I am grateful for this experience to share my knowledge and gain insight into how to best share that knowledge to unfamiliar listeners. In addition, for the past year and a half I have had the opportunity to work and be supported by Sabre Corporation's research group. My colleagues at Sabre have been very encouraging of my academic pursuits

while still directing engaging problems my direction. Specifically, I would like to thank Ben Vinod for the opportunity, Ross Darrow for never leading me astray, and Bob Newman for his continual wit.

Byrd Jr., Michael Craig B.S., Economics, University of North Texas

Estimation and Variable Selection in High-Dimensional Settings with Mismeasured Observations

Advisor: Monnie McGee
Doctor of Philosophy degree conferred December 21, 2019
Dissertation completed December 4, 2019

Understanding high-dimensional data has become essential for practitioners across many disciplines. The general increase in ability to collect large amounts of data has prompted statistical methods to adapt for the rising number of possible relationships to be uncovered. The key to this adaptation has been the notion of sparse models, or, rather, models where most relationships between variables are assumed to be negligible at best. Driving these sparse models have been constraints on the solution set, yielding regularization penalties imposed on the optimization procedure. While these penalties have found great success, they are typically formulated with strong assumptions on the variability of the observed data. We consider variables observed with some amount of measurement error in the high-dimensional setting. The common sparsity inducing models must be corrected for measurement error from a variety of sources, requiring special reformulations with nonstandard solutions.

We propose to utilize a recent methodology, the Imputation Regularization Optimization algorithm, to incorporate correction for measurement error. Focusing on the scenario where the amount of variables outnumbers the amount of observations, a scenario known to break traditional correction methods, we focus on two classes of models. The first class of model we investigate is the Gaussian graphical model, which aims to find all pair-wise dependencies from observed multivariate data. We find our method to be asymptotically consistent, and the method provides compelling numerical improvement over a model not accounting for the contaminated data. The second class of models we investigate is the well-known generalized

linear model, in which we show our correction method for contaminated covariates to be highly performant in comparison to other established techniques. To illustrate the real-world efficacy of our proposed procedures, both models are applied to a microarray data example.

TABLE OF CONTENTS

LIST OF FIGURES xi

LIST OF TABLES xii

CHAPTER

1. Introduction 1

1.1. Motivation 1

1.2. Contributions 4

2. Bayesian Regularization of Gaussian Graphical Models with Measurement Error 6

2.1. Introduction 6

2.2. Contaminated Gaussian Graphical Models 8

2.3. The IRO-BAGUS Algorithm 9

2.3.1. The Spike-and-Slab Lasso Prior Specification 11

2.3.2. The Full Model 12

2.3.3. Variable Selection 12

2.3.4. Consistency of the IRO-BAGUS algorithm 13

2.4. Computation for the IRO-BAGUS algorithm 14

2.4.1. Finding MAP estimate for $\mathbf{\Omega}_x$ 14

2.4.2. Other Computation Considerations 15

2.4.2.1. Estimating $\mathbf{\Sigma}_u$ 15

2.4.2.2. Starting Values 16

2.4.2.3. Addressing the constraint, $||\mathbf{\Omega}_x|| \leq B$ 16

2.4.2.4. Positive-Definiteness of $\mathbf{\Omega}_x$ 16

2.4.2.5. Parameter Tuning 16

2.5. Simulation Study 17

2.5.1. Simulation Setup 17

2.5.2. Simulation Results 19
2.6. Data Analysis 22
2.7. Conclusion 24
3. A Simple Correction Procedure for Contaminated High-Dimensional Generalized Linear Models 26
3.1. Introduction 26
3.1.1. Literature Review 28
3.1.2. Overview 29
3.2. The IRO-Algorithm for EIV Regression 30
3.2.1. The Imputation Regularization Optimization Algorithm 30
3.2.2. The I-Step for High-Dimensional EIV Regression 31
3.2.3. The RO-Step for High-Dimensional EIV Regression 32
3.2.4. Computational Considerations 33
3.3. IRO-Procedures for Some Contaminated GLMs 35
3.3.1. Gaussian Linear Regression 35
3.3.2. Binomial Linear Regression 37
3.3.3. Negative Binomial Linear Regression 39
3.4. Simulation 39
3.4.1. Gaussian Linear Regression 41
3.4.2. Binomial Linear Regression 43
3.5. Data Analysis 45
3.6. Conclusion 47
4. Conclusions and Future Directions 48
APPENDIX
A. Supplementary Material for Chapter 2 50

A.1. Proofs 50
A.2. Computing BAGUS with the EM-Algorithm 56
B. Supplementary Material for Chapter 3 58
B.1. Derivations 58
B.1.1. Covariate Only Imputation Distribution Derivation 58
B.1.2. Gaussian Linear Regression Imputation Distribution Derivation 59
B.1.3. Binomial Linear Regression Imputation Distribution Derivation 59
B.2. Estimating the Measurement Error Covariance with Replicates 61
B.3. Other Gaussian Simulation Results 63
B.3.1. Complete MCP Results 63
B.3.2. Scaled Lasso Results 64
B.4. Other Binomial Regression Results 66
B.5. Details for Data Analysis 67
BIBLIOGRAPHY 69

LIST OF FIGURES

Figure Page

2.1. Graphical representation for $d = 100$ of the hub (left) and random (right) structure, respectively. While the hub structure is fixed for a given d, the random graph is subject to change due to the generation process. 18

2.2. The conditional pair-wise relationships for each of the 273 genes remaining after filtering from the Wilms tumor study. Each edge represents a conditional pair-wise dependency between two nodes. The left shows the naive analysis, not correcting for measurement error, and the right shows the corrected analysis, correcting for measurement error. Green edges signify edges found on both graphs, and purple signifies analysis specific edges. 24

B.1. Outputted ELBO-plots for CLasso (left) and GMUS (right). Note that the increase of the regularization parameter has varying affect, hence the opposing trend. 68

LIST OF TABLES

Table Page

2.1. Simulation results for the hub graph structure, as specified in Section 2.5.1. For each signal-to-noise ratio and d, the true, naive, and corrected models are shown for metrics defined in Section 2.5.1. 20

2.2. Simulation results for the random graph structure, as specified in Section 2.5.1. For each signal-to-noise ratio and d, the true, naive, and corrected models are shown for metrics defined in Section 2.5.1. 21

3.1. Simulation results for Gaussian linear regression under the three specified settings with noise-to-signal ratio $\gamma = 0.5$. Ideal, Naive, and IRO use the MCP penalty for regularization. Bold numbers illustrate the best method between the correction procedures for the setting metric. 42

3.2. Simulation results for Binomial linear regression under the two specified settings with signal-to-noise ratio $\gamma = 0.5$, as described in-line. The Ideal, Naive, and IRO procedures use the MCP penalty for regularization... 45

3.3. The total number of selected genes that overlapped between the Naive, IRO-adjusted, CLasso, and GMUS procedures. Note, the diagonal shows the total number of genes found by each procedure. 47

B.1. Simulation results for Gaussian linear regression under the three specified settings with noise-to-signal ratio $\gamma = 0.5$. Ideal, Naive, and IRO use the MCP penalty for regularization. 63

B.2. Simulation results for Gaussian linear regression under the three specified settings with noise-to-signal ratio $\gamma = 1$. Ideal, Naive, and IRO use the MCP penalty for regularization. 64

B.3. Simulation results for Gaussian linear regression under the three specified settings with noise-to-signal ratio $\gamma = 0.5$. Ideal, Naive, and IRO use the Scaled Lasso penalty for regularization. Due to convergence issues, many L2 norms for the naive method are missing and denoted NA. 65

B.4. Simulation results for Gaussian linear regression under the three specified settings with noise-to-signal ratio $\gamma = 1$. Ideal, Naive, and IRO use the Scaled Lasso penalty for regularization. Due to convergence issues, many L2 norms for the naive method are missing and denoted NA. 65

B.5. Simulation results for Binomial linear regression under the two specified settings with signal-to-noise ratio $\gamma = 1$. The Ideal, Naive, and IRO procedures use the MCP penalty for regularization. 66

Dedicated to Lukas, for whom this journey would never have been possible without.

Chapter 1

Introduction

1.1. Motivation

Identifying underlying relationships among sets of random variables is a fundamental problem in traditional statistical practice. As time progresses, and our computational capabilities grow, the amount of data to analyze is increasing at a rapid pace. Intuitively, the more data available, the more relationships to be uncovered. However, in a practical sense, it is likely that the majority of these relationships are negligible at best. This thought, termed "the bet on sparsity principle," advocates for assuming most relationships for a proposed model in the high-dimensional setting to be negligible or non-existent [20]. Sparsity has been a focal point of statistical research, particularly with respect to the statistical learning literature, and has become standard practice when implementing a wide variety of statistical models to analyze data with a large number of variables [16].

We consider the high-dimensional setting where data, $\boldsymbol{X} \in \mathbb{R}^{n \times p}$, is parameterized through some distribution with unknown parameters, $\boldsymbol{\Theta}$. Of particular interest is the setting in which $n \ll p$, as is typical in, for example, high thoroghput biological assays. Models for data in this setting tend to be unidentifiable, hence some constraint is typically imposed on the solution. This constraint is then incorporated into optimization procedure used to estimate model parameters. For negative log-likelihood $\mathcal{L}(\boldsymbol{X}; \boldsymbol{\Theta})$ and penalty $P(\boldsymbol{\Theta}; \lambda)$, the estimate of $\boldsymbol{\Theta}$ is then

$$\boldsymbol{\Theta}^* = \underset{\boldsymbol{\Theta}}{\operatorname{argmin}}\, \mathcal{L}(\boldsymbol{X}; \boldsymbol{\Theta}) + P(\boldsymbol{\Theta}; \lambda). \tag{1.1}$$

For simplicity we focus only on $\boldsymbol{X}$, but this setting is easily extended to incorporate a response for each observation $\boldsymbol{y} = (y_1, \ldots, y_n)^T$ for regression models. The penalty function

enforces some constraint on the estimated parameters, termed regularization, and is tuned by the parameter λ. For an appropriately chosen penalty, $\boldsymbol{\Theta}^*$ will be sparse, which can then be used to identify the potential existing relationships.

While much progress has been made on efficient variable selection with models of the form (1.1), most of these statistical methodologies implicitly assume perfectly observed realizations from the underlying stochastic process that produces the data. This assumption, however, is often not the reality, such as in microarray experiments where the complex nature of the assays, involving multiple steps, results in a propagation of measurement error in the final gene expression measures. Data contaminated with measurement error is well known to statisticians, and has been studied in detail for a variety of traditional, low-dimensional contexts [7]. Incorporating the measurement error into an analysis generally reduces the number of false positives, but at the cost of reducing the overall power of the model [7].

As an example on the effect of measurement error, consider traditional Gaussian regression model, where $y_i \sim N(\boldsymbol{x}_i^T\boldsymbol{\beta}, \sigma_\epsilon^2)$ for $i = 1, \ldots, n$. The estimated coefficients are

$$\boldsymbol{\beta}^* = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \frac{1}{2\sigma_\epsilon^2}||\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}||_2^2, \tag{1.2}$$

which is well known to give $\boldsymbol{\beta}^* = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}\boldsymbol{y}$ when $n \geq p$. However, consider instead that we observe contaminated observations. Let the contamination be additive to the true realization, where $\boldsymbol{w}_i = \boldsymbol{x}_i + \boldsymbol{u}_i$ such that $\boldsymbol{u}_i \sim N(\boldsymbol{0}_p, \boldsymbol{\Sigma}_u)$ for $i = 1, \ldots, n$. Naively replacing $\boldsymbol{X}$ with $\boldsymbol{W} = (\boldsymbol{w}_1, \ldots, \boldsymbol{w}_n)^T$ in (1.2) would result with asymptotically inconsistent estimates [7]. Take $p = 1$, then this attenuation can be characterized as

$$\beta^* = \beta_w^* \frac{\sigma_x^2}{\sigma_u^2 + \sigma_x^2}, \tag{1.3}$$

where β_w^* is the coefficient found from the naive regression, $Var(x_i) = \sigma_x^2$, and $Var(u_i) = \sigma_u^2$. Hence, the naive estimate of the regression coefficient is systematically of smaller magnitude than optimal estimate. While such a characterization is not easily written for regularized

models with measurement error, not correcting for measurement error in regularized models is known to bias results and lead to many more false positives being introduced into the model [48].

Our focus in this work is to develop a simple correction procedure which can be applied to a wide variety of regularized problems. Of interest is the ability to leverage current methodologies developed for the ideal case of observing the uncontaminated data. We focus on the high-dimensional problem, particularly for $n < p$, and, often, $n \ll p$. We consider two such problems of the form (1.1): (1) Gaussian graphical models (GGMs) and (2) generalized linear models (GLMs). Both of these classes of models in the high-dimensional setting benefit from introducing the regulatory penalty; this benefit not only eases interpretation of the model, but enables the model to be identifiable and unique.

The first class of models, Gaussian graphical models, aim to identify all conditional pairwise dependencies from a multivariate process [58]. If $\boldsymbol{x} \sim N(\mathbf{0}_p, \boldsymbol{\Sigma}_x)$, then the precision matrix, $\boldsymbol{\Omega}_x = \boldsymbol{\Sigma}_x^{-1}$, completely identifies these conditional relationships. This fact was shown by [12], where the element $[\boldsymbol{\Omega}_x]_{ij} = 0$ implies the pair of variables (i, j) is conditionally independent; otherwise, the pair are conditionally dependent. Regularization methods are used to set most of these off-diagonal precision elements to be 0 for identifying important pairwise relationships. Moreover, a sparsity inducing model is necessary to achieve a precision matrix of full-rank, which is helpful if the estimate is to be used in downstream analysis.

The second class of models, generalized linear models, are foundational to statistical practice for quantifying the relationship between explanatory variables and response data of various types such as continuous, binary, and count. Specifically, let response $y_i \sim \mathcal{D}$, such that

$$\mathbb{E}_{\mathcal{D}} = f(\boldsymbol{x}_i^T \boldsymbol{\theta}), \tag{1.4}$$

where f is the inverse of the canonical link function [33]. The goal of introducing sparsity inducing regularization is for the purpose of selecting the relevant covariates for the model, a task long explored in the model selection literature for linear models [20]. Moreover, like

GGMs, if $n < p$, then it is necessary to impose a constraint on the model coefficients to obtain an identifiable model.

Much of the methods developed revolve specifically around the penalty imposed onto the model [20]. Most of the focus has been on convex penalty terms due to computational benefits in the optimization procedure. Arguably the most popular convex procedure, the Lasso [53], imposes an ℓ_1-norm constraint on the variables of interest, which results in negligible variables being set to 0. In the notation of (1.1), the Lasso penalty is

$$P(\boldsymbol{\Theta}; \lambda) = \lambda ||\boldsymbol{\theta}||_1,$$

where $|| \cdot ||_1$ is the ℓ_1-norm. The convexity of the Lasso's loss function allows for efficient optimization via coordinate ascent [60], which is readily available for widespread use [15]. The Lasso's popularity has resulted in many adaptations and extensions such as the the elastic net penalty [66], the group Lasso penalty [34], and the fused Lasso penalty [54]. While computational guarantees are easier to verify rigorously with convex penalties, the estimates are often biased. This has given rise to non-convex penalties, such as the Minimax Concave Penalty [64], which aim to adjust the amount of regularization to be less biased and provide better selection qualities. Such penalties have been found to be implemented efficiently, with much success in many applications [57] [44]. Naturally, many formulations for the GGM [17] [31] and GLM [37] setting have been developed, which we make use of throughout.

1.2. Contributions

The goal of the methods we propose is to correct for mismeasured observations in such a way that established regularization procedures can be used directly. To this end, we utilize a recently introduced methodology for missing data, the Imputation Regularization Optimization (IRO) algorithm [29], in the context of correcting for measurement error. The

IRO-algorithm is similar in nature to the EM-algorithm [13], in that it iterates between accounting for missing random variables with some parametric assumptions, then it optimizes the resulting objective function. However, unlike the EM-algorithm, the IRO-algorithm gives asymptotically consistent estimates of the unknown parameters, even when $n \ll p$ and regularization is a part of the optimization.

The GGM problem is not well studied in the presence of measurement error, and we provide, to the best of our knowledge, a first approach to directly estimating the precision matrix in the high-dimensional setting. In Chapter 2 we address the contaminated GGM problem by using the machinery of the IRO-algorithm to correct for the mismeasurements. As with the regression scenario described before, not correcting for the mismeasured data can lead to inconsistent estimates as

$$\boldsymbol{\Omega}_w = \boldsymbol{\Sigma}_w^{-1} = (\boldsymbol{\Sigma}_{\boldsymbol{x}} + \boldsymbol{\Sigma}_u)^{-1} = \boldsymbol{\Omega}_x - \boldsymbol{\Omega}_x(\boldsymbol{I} - \boldsymbol{\Sigma}_u\boldsymbol{\Omega}_x)^{-1}\boldsymbol{\Omega}_x \neq \boldsymbol{\Omega}_x, \tag{1.5}$$

where $\boldsymbol{\Sigma}$ and $\boldsymbol{\Omega}$ correspond to the covariance and precision matrices of the indexed random variable and $\boldsymbol{I}$ is the identity matrix. We show our procedure to be asymptotically consistent and is also able to reduce the number of false positives.

In Chapter 3, we incorporate response data into the IRO-algorithm for measurement error. This is done in context of the contaminated high-dimensional GLM, where we develop a general correction procedure and explicitly specify how to apply it to three common types of response data: continuous, binary, and count. While the procedure for implementing the IRO-algorithm for Gaussian response was briefly addressed in [29], we illustrate for the first time how to apply it to binary and count data using a data-augmentation scheme found in the Bayesian computation literature [41]. Empirically, we find the IRO-algorithm correction to improve upon the naive model. We also compare our proposed estimator against the Corrected Lasso [48] and Generalized Matrix Uncertainty Selector, and show that it outperforms them in terms of estimation quality and variable selection.

Chapter 2

Bayesian Regularization of Gaussian Graphical Models with Measurement Error

2.1. Introduction

A core problem in statistical inference is estimating the conditional relationship among random variables. Naturally, a full description of the underlying connections among the numerous random variables is valuable information across many disciplines, such as in biology where the relationships among hundreds of genes involved in a metabolic process is desired to be uncovered. In fact, under the assumption that the variables follow a multivariate Gaussian distribution, the inverse covariance matrix, known as the precision matrix, characterizes conditional dependence between two dimensions. This is accomplished by noting that if an element of the precision matrix is 0, then the two variables are conditionally independent; see [28] for a review. This setting, often referred to as a Gaussian graphical model, is where our analysis takes place.

Estimating the precision matrix is a difficult task when the number of observations n is often much less than the dimension of the features d (we use d here instead of p as before due to p being used later in this chapter). A naive approach is to estimate the precision matrix by the inverse of the empirical covariance matrix; this estimate, however, is known to perform poorly and is ill-posed when $n < d$ [25]. The common approach is to assume that the precision matrix is sparse [12]; that is, we assume the precision matrix's off-diagonal elements are mostly 0. As a result, most pairs of variables are conditionally independent. The sparsity assumption has led to different lines of research with regularized models to estimate the precision matrix. While one approach utilizes a sparse regression technique that estimates the precision by iteratively regressing each variable on the remaining

variables, for instance [26], we instead focus on the direct likelihood approach. The direct likelihood approach optimizes the full likelihood function with an element-wise penalty on the precision matrix; common examples being graphical lasso [17], CLIME [6], and TIGER [31]. We utilize a recent Bayesian optimization procedure, called BAGUS, that relies on optimization performed by the EM-algorithm, which was shown to have desirable theoretical properties, including consistent estimation of the precision matrix and selection consistency of the conditional pair-wise relationships [18].

There are many practical issues associated with Gaussian graphical models, such as hyperparameter tuning [63], missing data [29], and repeated trials [51], which practitioners need to adjust for a successful analysis. We address another practical issue involved with these models, measurement error. Measurement error occurs when the variables of interest are not observed directly; instead, the observations are the desired variables that have been additionally perturbed with noise from some measurement process. This happens when, for instance, an inaccurate device is used to measure some sort of health metric. Measurement error models have been studied extensively for classical settings such as density deconvolution and regression [7], but, to our knowledge, have not yet been well studied in the context of Gaussian graphical models, especially in high dimensional setting.

We propose a Bayesian methodology to correct for measurement error in estimating a sparse precision matrix; our new method extends the optimization procedure of [18]. While directly incorporating the estimate of the uncontaminated variable is possible, we find the incorporation of the imputation-regularization technique of [29] to provide more desirable results. Our procedure imputes the mismeasured random variables, then performs BAGUS on this imputation; these steps are performed for a small number of cycles, requiring more computation but giving better results than the naive estimator. We prove consistency of the estimated precision matrix with the imputed procedure, and illustrate the performance in a simulation study. Finally, we apply the methodology to a microarray dataset that contains gene measurement of favorable histology Wilms tumor.

2.2. Contaminated Gaussian Graphical Models

Given a d-dimensional random vector, $\boldsymbol{x} = \{x^1, \ldots, x^d\}$, we are interested in the conditional dependence of two variables x^i and x^j, for any pair (i, j) with $1 \leq i \leq j \leq d$, given all the remaining variables. This conditional dependence structure is usually represented by an undirected graph $G = (V, E)$, where $V = \{1, \ldots, d\}$ is the set of nodes and $E \subseteq V \times V = \{(1,1), (1,2), \ldots, (d,d)\}$ is the set of edges [28]. In this representation, the two variables x^i and x^j are conditionally independent if there is no edge between node i and node j.

If the vector $\boldsymbol{x}$ follows a multivariate normal distribution with mean $\mathbf{0}$ and covariance matrix $\boldsymbol{\Sigma}_x$, $\boldsymbol{x} \sim N_d(\mathbf{0}, \boldsymbol{\Sigma}_x)$, every edge corresponds to a non-zero entry in the precision matrix $\boldsymbol{\Omega}_x = \boldsymbol{\Sigma}_x^{-1}$, see [28]. The model in this scenario is often known as a Gaussian graphical model. In the high dimensional setting, the set of edges are usually assumed to be sparse, meaning that only a few pairs (x^i, x^j) are conditionally dependent. In the Gaussian case, this assumption implies only a few off-diagonal entries of $\boldsymbol{\Omega}_x$ are non-zero.

Suppose the data consist of *iid* observations $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_n$, where $\boldsymbol{w}_i = \boldsymbol{x}_i + \boldsymbol{u}_i$, $i = 1, \ldots, n$ with $\boldsymbol{x}_i \sim N_d(\mathbf{0}, \boldsymbol{\Sigma}_x)$ and $\boldsymbol{u}_i \sim N_d(\mathbf{0}, \boldsymbol{\Sigma}_u)$. Here, $\boldsymbol{w}_i = (w_i^1, \ldots, w_i^d)$, with the subscript and superscript denoting the observation and components respectively. Denote $\boldsymbol{W}$ as the $n \times d$ matrix of observed data. The model is equivalent to the following hierarchical representation. First, the latent random variables $\boldsymbol{x}_i$ are generated from a $N_d(\mathbf{0}, \boldsymbol{\Sigma}_x)$ distribution, and when conditioned on $\boldsymbol{x}_i$ and $\boldsymbol{\Sigma}_u$, we have $\boldsymbol{w}_i | \boldsymbol{x}_i, \boldsymbol{\Sigma}_u \sim N_d(\boldsymbol{x}_i, \boldsymbol{\Sigma}_u)$ for each $i = 1, \ldots, n$. This forms an intuitive generative process, where first $\boldsymbol{x}$ is realized, then contaminated by measurement error $\boldsymbol{u}$, and observed finally as $\boldsymbol{w}$. The problem of interest is to estimate the precision matrix $\boldsymbol{\Omega}_x$ in the high dimensional setting $n < d$.

Consider an additive measurement error model where $\boldsymbol{w} = \boldsymbol{x} + \boldsymbol{u}$ and $\boldsymbol{w}$ is the observed data. Denote $\boldsymbol{U} = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n)^T$ as measurement errors that are independent from data $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^T$. For $i = 1, \ldots, n$, the amount of measurement error is drawn from another multivariate normal distribution with mean $\mathbf{0}$ and covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{u}}$, $\boldsymbol{u}_i \sim N_d(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{u}})$.

We assume $\boldsymbol{\Sigma_u}$ to be diagonal, and hence the amount of measurement error on each variable is uncorrelated. We also assume that $\boldsymbol{\Sigma_u}$ is known or estimable from ancillary data, such as replicate measurements. The contaminated variables $\boldsymbol{w}$ in general have a different conditional dependence structure from that of $\boldsymbol{x}$. Indeed, the covariance and precision matrix of $\boldsymbol{w}$ is given by

$$\boldsymbol{\Sigma}_w = \boldsymbol{\Sigma_x} + \boldsymbol{\Sigma}_u$$

and

$$\boldsymbol{\Omega}_w = \boldsymbol{\Sigma}_w^{-1} = (\boldsymbol{\Sigma_x} + \boldsymbol{\Sigma}_u)^{-1} = \boldsymbol{\Omega}_x - \boldsymbol{\Omega}_x(\boldsymbol{I} + \boldsymbol{\Sigma}_u\boldsymbol{\Omega}_x)^{-1}\boldsymbol{\Sigma}_u\boldsymbol{\Omega}_x, \tag{2.1}$$

respectively; here, $\boldsymbol{I}$ denotes the $d \times d$ identity matrix. Equation (2.1) follows from the Kailath variant formula in [39]. Furthermore, equation (2.1) suggests that $\boldsymbol{\Omega}_w$ and $\boldsymbol{\Omega}_x$ are equal if the product $\boldsymbol{\Omega}_x(\boldsymbol{I} + \boldsymbol{\Sigma}_u\boldsymbol{\Omega}_x)^{-1}\boldsymbol{\Sigma}_u\boldsymbol{\Omega}_x$ is equal to a zero matrix. This is generally not the case when the matrix $\boldsymbol{\Sigma}_u$ is not zero.

When no measurement error is present, i.e the $\boldsymbol{x}_i$ are directly observed, the sample covariance matrix $\boldsymbol{S} = n^{-1}\sum_{i=1}^{n}(\boldsymbol{x}_i - \bar{\boldsymbol{x}})(\boldsymbol{x}_i - \bar{\boldsymbol{x}})^\top$, with $\bar{\boldsymbol{x}}$ being the sample mean, is a consistent estimator for $\boldsymbol{\Sigma}_x$. However it has the rank of at most $n < d$, so it is not invertible to estimate $\boldsymbol{\Omega}_x$. When measurement error is present, we assume the covariance matrix of measurement error $\boldsymbol{\Sigma}_u$ is known or estimable from replicates. A naive approach is first to estimate $\boldsymbol{\Sigma}_x$ by $\tilde{\boldsymbol{\Sigma}}_x = \boldsymbol{S}_w - \boldsymbol{\Sigma}_u$, where $\boldsymbol{S}_w$ denotes the sample covariance from contaminated data $\boldsymbol{W}$, and then to invert $\tilde{\boldsymbol{\Sigma}}_x$ to estimate $\boldsymbol{\Omega}_x$. The main issue with this approach is that $\tilde{\boldsymbol{\Sigma}}_x$ is generally not positively definite. This implies its inverse is also not positively definite, which is necessary to find a consistent estimate $\boldsymbol{\Omega}_x$. Hence, a correction procedure to estimate $\boldsymbol{\Omega_x}$ need not rely upon the sample covariance matrix $\tilde{\boldsymbol{\Sigma}}_x$ directly. Furthermore, the procedure should also be able to incorporate sparsity constraints to recover the graphical model structure. These requirements are addressed by the procedure described in the next section.

2.3. The IRO-BAGUS Algorithm

In a recent work, [29] develop a methodology to efficiently handle high dimensional problems with missing data. Their solution is an EM-algorithm variant which alternates between two steps, the imputation step and regularized optimization step; we refer to their algorithm as the IRO algorithm. Denote the missing data as Y, and observed data as X. Also denote the desired parameter to be estimated by θ, and begin with some initial guess $\theta^{(0)}$. During the t^{th} iteration, the IRO algorithm generates Y from the distribution given by the current estimate of θ, i.e. $Y \sim \pi(Y|X, \theta^{(t-1)})$. Then, using X and Y, maximizes θ, under regulation, using the full likelihood. [29] show that this procedure results in a consistent estimate of $\theta^{(t)}$, and results in a Markov chain with stationary distribution.

We make use of this framework for our current problem pertaining to mismeasured observations instead of missing values. The problems are naturally related in the sense that both are generating values of the true process from some estimated underlying distribution. We return to the hierarchical structure of the problem, i.e. $\boldsymbol{w}|\boldsymbol{x}, \boldsymbol{\Sigma}_u \sim N_d(\boldsymbol{x}, \boldsymbol{\Sigma_u})$ and $\boldsymbol{x} \sim N_d(\boldsymbol{0}, \boldsymbol{\Sigma_x})$. The IRO algorithm proceeds iteratively between the two following steps:

- *Imputation step*: At iteration t, draw $\boldsymbol{X}^{(t)} = (\boldsymbol{x}_1^{(t)}, \ldots, \boldsymbol{x}_d^{(t)})$ from the posterior full conditional of $\boldsymbol{X}$, using the current estimate of $\boldsymbol{\Omega}_{\boldsymbol{x}}^{(t-1)}$. Specifically, for $i = 1, \ldots, n$, draw $\boldsymbol{x}_i^{(t)}|\boldsymbol{w}, \boldsymbol{\Omega}_{\boldsymbol{x}}^{(\boldsymbol{t-1})} \sim N_d(\boldsymbol{\Lambda^{-1}\Omega_u w}_i, \boldsymbol{\Lambda}^{-1})$ where $\boldsymbol{\Lambda} = (\boldsymbol{\Omega}_{\boldsymbol{x}}^{(t-1)} + \boldsymbol{\Omega_u})$. Note that the posterior distribution of $\boldsymbol{x}_i$ depends only on $\boldsymbol{w}_i$ due to independence. This allows for easy generation of data from the true underlying distribution.

- *Regularization Step*: Apply a regularization to the generated $\boldsymbol{X}^{(t)}$ and obtain a new estimate of $\boldsymbol{\Omega}_{\boldsymbol{x}}^{(t)}$.

In this work, the regularization step is carried out based on a recent Bayesian methodology, called BAGUS. Hence, the whole algorithm is referred to as the IRO-BAGUS algorithm. The next subsections 2.3.1-2.3.3 outline prior specifications, the full model, and variable selection for BAGUS. After that, section 2.3.4 discusses consistency of the IRO-BAGUS estimate.

2.3.1. The Spike-and-Slab Lasso Prior Specification

Denote the elements $\mathbf{\Omega}_x$ to be ω_{ij}. Recently, a non-convex, continuous relaxation penalty for the spike-and-slab prior was created for the standard lasso problem [44]. This prior was extended to the case of graphical models by [18], and is given by

$$\pi(\omega_{ij}) = \frac{\eta}{2v_1} \exp\left\{-\frac{|\omega_{ij}|}{v_1}\right\} + \frac{1-\eta}{2v_0} \exp\left\{-\frac{|\omega_{ij}|}{v_0}\right\} \tag{2.2}$$

for the off diagonal elements ($i \neq j$), where $0 < v_0 < v_1$ and $0 < \eta < 1$. This prior can be interpreted as a mixture of the spike-and-slab prior. The first component of the mixture has prior probability η, and is associated with the slab component, i.e. $\omega_{ij} \neq 0$. Conversely, with prior probability $1 - \eta$ the element is from the spike component, suggesting $\omega_{ij} = 0$.

Traditionally, the spike-and-slab prior has a point mass component at 0 and some other continuous distribution for the slab component. This is to represent setting unwanted terms exactly to 0. Here, both the spike and the slab components are distributed according to a Laplace distribution; both are centered at 0, but the spike is more tightly centered by a smaller variance term than the slab. This relaxation of the spike-and-slab prior allows for efficient gradient based algorithms, while still being theoretically sound as shown in [43].

Shrinkage is not desired on the diagonal elements, so a different weakly informative exponential prior is given instead, $\pi(\omega_{ii}) = \tau \exp\{-\tau\omega_{ii}\}$. Another consideration for the prior of $\mathbf{\Omega}_x$ is to ensure the whole matrix to be positive definite, denoting as $\mathbf{\Omega}_x \succ 0$. Moreover, in line with [18], we require the spectral norm to be bounded above by some value B, $||\mathbf{\Omega}_x|| \leq B$. This assumption will be important for establishing consistency going forward. The full prior distribution for $\mathbf{\Omega}_x$ is then given by

$$\pi(\mathbf{\Omega}_x) = \prod_{i<j} \pi(\omega_{ij}) \prod_i \pi(\omega_{ii}) I(\mathbf{\Omega}_x \succ 0) I(||\mathbf{\Omega}_x|| \leq B). \tag{2.3}$$

2.3.2. The Full Model

Without measurement error, the posterior distribution is specified as

$$\pi(\boldsymbol{\Omega}_x|\boldsymbol{X}) \propto \prod_{i=1}^{n} \pi(\boldsymbol{x}_i|\boldsymbol{\Omega}_x)\pi(\boldsymbol{\Omega}_x). \tag{2.4}$$

The full conditionals can be derived for (2.4), but, to avoid computationally expensive MCMC sampling for this large dimensional problem, [18] opted to instead find the mode of the the posterior distribution, often referred to as the MAP. The MAP can be found by minimizing the uncontaminated (UC) objective

$$L^{\mathrm{UC}}(\boldsymbol{\Omega}_x) = \log \pi(\boldsymbol{\Omega}_x|\boldsymbol{X}) = \frac{n}{2}\left(\mathrm{tr}(\boldsymbol{X}^T\boldsymbol{\Omega}_x\boldsymbol{X}) - \mathrm{logdet}(\boldsymbol{\Omega}_x)\right) + \sum_{i<j}\pi(\omega_{ij}) + \sum_{i}\pi(\omega_{ii}) + K \tag{2.5}$$

with respect to $\boldsymbol{\Omega}_x$, where K is the normalizing constant in (2.4). To this end, [18] proved the local convexity of (2.4) when $||\boldsymbol{\Omega}_x|| \leq B < \infty$, which allows an easy optimization procedure that converges asymptotically to the correct precision matrix.

2.3.3. Variable Selection

Many practitioners use Gaussian graphical models for the purpose of identifying non-zero entries of $\boldsymbol{\Omega}_x$, which signify conditional dependencies among the two different variables. The spike-and-slab lasso formulation allows for this quite easily by viewing the optimization as an instance of the EM-algorithm and defining the hierarchical prior

$$\begin{cases} \omega_{ij}|r_{ij} = 0 \sim \mathrm{Laplace}(0, v_0) \\ \omega_{ij}|r_{ij} = 1 \sim \mathrm{Laplace}(0, v_1) \end{cases}. \tag{2.6}$$

Here, r_{ij} is the random indicator that the element of the precision matrix follows from the spike or the slab component, where $r_{ij} \sim \mathrm{Bern}(\eta)$. A further hierarchical level can be added by treating η as random instead of a fixed hyperparameter. Recent work from [14] illustrates

this and is line with the spike-and-slab Lasso setting of [44]. Given our purpose is to study the effect of the measurement error, we choose to treat it as a fixed.

The conditional posterior distribution for r_{ij} is also Bernoulli, with probability of success

$$p_{ij} = \frac{v_1}{v_0}\frac{1-\eta}{\eta}\exp\left\{|\omega_{ij}|\left(\frac{1}{v_1}-\frac{1}{v_0}\right)\right\}. \tag{2.7}$$

We will use the MAP estimate of ω_{ij} in (2.7) as the approximate probability of inclusion. A hard threshold will be specified for the inclusion probability matrix to select the final model. Denote $\boldsymbol{R}$ and $\boldsymbol{P}$ to be the matrix of indicators and conditional posterior probability of inclusion for each element of $\boldsymbol{\Omega}_x$. We note that for final inference it may be better to forgo this inclusion threshold, and instead rank-order the p_{ij} for purposes of downstream investigation; however, this will depend on the application at hand.

2.3.4. Consistency of the IRO-BAGUS algorithm

The entire data generation process for the contaminated sample is summarized below:

$$\begin{aligned}
&\boldsymbol{w}_i|\boldsymbol{x}_i, \boldsymbol{\Omega}_x \sim N_d(\boldsymbol{x}_i, \boldsymbol{\Sigma}_u),\ i = 1, \ldots, n\\
&\boldsymbol{x}_i|\boldsymbol{\Omega}_x \sim N_d(\boldsymbol{0}, \boldsymbol{\Omega}_x^{-1}),\ i = 1, \ldots, n\\
&\omega_{ij}|r_{ij} = 0, v_0 \sim \text{Laplace}(0, v_0),\ i \neq j,\ i, j = 1, \ldots, n\\
&\omega_{ij}|r_{ij} = 1, v_1 \sim \text{Laplace}(0, v_1),\ i \neq j,\ i, j = 1, \ldots, n\\
&\omega_{ii} \sim \text{Exp}(\tau),\ i = 1, \ldots, n\\
&r_{ij}|\eta \sim \text{Bern}(\eta),\ i \neq j, i, j = 1, \ldots, n.
\end{aligned}$$

Instead of approximating the posterior distribution of all the parameters, the IRO-BAGUS algorithm iteratively generates realizations of uncontaminated data, $\boldsymbol{X}$, then optimizes $\boldsymbol{\Omega}_x$ with these generated values. Under some technical conditions, the IRO algorithm is shown to produce a consistent estimate after each iteration in the context of missing data when

the regularization step results in a consistent estimate [29]. We show that these conditions are also held in the case of contaminated data, so the IRO-BAGUS algorithm results in a consistent estimate. Theorem 1 is the analogue statement of consistency as in the missing data case. The proof is given in the appendix.

Theorem 1. *Assuming* $||\mathbf{\Omega}_x|| \leq B$*, then the estimate* $\mathbf{\Omega}_x^{(t)}$ *is uniformly consistent to* $\mathbf{\Omega}_x$ *when* $\log(t) = \mathcal{O}(n)$.

It can be seen that the nature of the IRO algorithm is similar to that of MCMC. With additional mild conditions, [29] note that the IRO results in a Markov chain with a stationary distribution, and hence the average of the maximization steps are consistent estimates of the underlying parameters. Our final estimates are the averaged regularized optimization steps given by BAGUS from the imputed data at each iteration, removing a small number of the beginning iterations as burn-in. By averaging instead of taking only the final iteration, we make the analysis less variable. In this sense, the relationships that the correction procedure identifies are more likely to be true relationships, cutting down on the number of false positives.

2.4. Computation for the IRO-BAGUS algorithm

2.4.1. Finding MAP estimate for $\mathbf{\Omega}_x$

Here we consider some computational aspects of our proposed methodology. First, we focus to the optimization of $\mathbf{\Omega}_x$. In our procedure, once $\boldsymbol{X}$ is generated, the objective function to be optimized is L^{uc}, as was shown in Equation (2.7); we note this is due to the conditional independence of $\boldsymbol{W}$ and $\mathbf{\Omega}_x$ in the hierarchical structure of the contamination process. Optimizing L^{uc} is difficult to do directly; therefore, the latent factors r_{ij} from Section 2.3.3 are introduced into the process as in [18]. This allows an E-step similar to the spike-and-slab Lasso and an M-step similar to the Graphical Lasso.

The optimization seeks to find the MAP of the posterior proportional to

$$|\boldsymbol{\Omega}_x|^{\frac{1}{2}} \exp\left\{-\frac{1}{2}\boldsymbol{X}^T\boldsymbol{\Omega}_x\boldsymbol{X}\right\} \prod_{i<j} \pi(\omega_{ij}|r_{ij})\pi(r_{ij}|\eta) \prod_i \pi(\omega_{ii}|\tau) I(\boldsymbol{\Omega}_x \succ 0) I(||\boldsymbol{\Omega}_x|| \leq B). \quad (2.8)$$

The E-step takes the conditional expectation of r_{ij} in the proportional posterior. Each r_{ij} is conditionally Bernoulli with probability as given in Equation (2.7), which allows for easy calculation of the desired conditional expectations. Then, the desired Q function to maximize in the M-step is given by

$$Q(\boldsymbol{\Omega}_x|\boldsymbol{\Omega}_x^{(t)}) = \mathbb{E}_{\boldsymbol{R}|\boldsymbol{\Omega}_x^{(t)}} \log \pi(\boldsymbol{\Omega}_x, \boldsymbol{X}|\boldsymbol{W}, \boldsymbol{\Sigma}_u), \quad (2.9)$$

where the expectation is taken element wise for $\boldsymbol{R}$ by assumed independence of inclusion. Maximizing Q is done by a block coordinate descent algorithm. The algorithm cycles between column-wise updates of Ω_x. We put the details of this procedure in the Appendix.

2.4.2. Other Computation Considerations

2.4.2.1. Estimating $\boldsymbol{\Sigma}_u$

We have assumed the covariance matrix of measurement error $\boldsymbol{\Sigma}_u$ to be known before applying the IRO-BAGUS algorithm. In practice, the matrix $\boldsymbol{\Sigma}_u$ is often estimated from ancillary data, such as replicate observations. Assuming measurement error between variables to be independent is reasonable for many problems and often used in the literature [48]. In that case, only the diagonal of $\boldsymbol{\Sigma}_u$ only needs to be estimated. For the data analysis application we provide in Section 2.6, we estimate the diagonal elements with the method described in [55], assuming homogeneity of measurement error between observations. After that, we performed the IRO-BAGUS algorithm as previously described.

2.4.2.2. Starting Values

The starting value plays a significant role in the speed of optimization at each step. To begin, we perform a naive analysis on the raw contaminated data, $\boldsymbol{W}$, giving estimate $\boldsymbol{\Omega}_x^{(0)}$. This value is then used to start the IRO procedure by generating $\boldsymbol{X}$. Each optimization has a warm start from the previous iteration's estimated precision matrix.

2.4.2.3. Addressing the constraint, $||\boldsymbol{\Omega}_x|| \leq B$

The constaint that $||\boldsymbol{\Omega}_x|| \leq B$ needs to be incorporated into the optimization. [18] suggest using a threshold on the largest absolute value of the elements of the column being updated in the block coordinate descent. We use the same threshold, and find no performance issues when used with the IRO algorithm.

2.4.2.4. Positive-Definiteness of $\boldsymbol{\Omega}_x$

Many procedures to estimate a sparse precision cannot guarantee postive-definiteness, however [18] show that the output of BAGUS from the EM algorithm is always symmetric and positive definite. It is easy to show that the imputation step, with final results averaged, also results in this nice property.

Theorem 2. *The estimated precision matrix $\hat{\boldsymbol{\Omega}}_x = T^{-1} \sum_{t=1}^{T} \boldsymbol{\Omega}_x^{(t)}$ is symmetric and positive definite if the initial value of $\boldsymbol{\Omega}_x$ for BAGUS at each iteration t was also positive definite.*

Proof. By Theorem 5 in [18], if the initial value to optimize BAGUS is positive definite, then $\boldsymbol{\Omega}_x^{(t)}$ is also positive definite. The set of positive definite matrices form a cone, and hence the average will also be in this cone. □

2.4.2.5. Parameter Tuning

There are four hyperparameters in BAGUS, $\eta, \tau, v_0,$ and v_1. As with [18], we always set $\eta = 0.5$ and $\tau = v_0$, which leaves two hyperparameters to tune. Again, we follow [18], who

suggest a BIC-like criteria to select the best model from a grid of hyperparameters. This criteria is

$$\text{BIC} = n(\text{tr}(\boldsymbol{S}\hat{\boldsymbol{\Omega}}_x) - \text{logdet}(\hat{\boldsymbol{\Omega}}_x)) + \log(n) \times q,$$

where $\hat{\boldsymbol{\Omega}}_x$ is the estimated precision matrix and q is the number of non-zero elements of the estimated in the upper diagonal of the precision matrix. We use this in similar fashion for the IRO procedure, but instead we use the averaged $\boldsymbol{\Omega}_x^{(t)}$ in the BIC calculation.

2.5. Simulation Study

2.5.1. Simulation Setup

We investigate the performance of our methodology under several different settings. For each setting we generate $\boldsymbol{x}_i$ following a d-variate Gaussian distribution with mean $\mathbf{0}$ and precision matrix $\boldsymbol{\Omega}_x$ according to some graphical structure; we refer to this as the *true data*. Then, the contaminated observations $\boldsymbol{w}_i$ were generated from $\boldsymbol{w}_i = \boldsymbol{x}_i + \boldsymbol{u}_i$, where $\boldsymbol{u}_i \sim N_d(\mathbf{0}, \boldsymbol{\Sigma}_u),\ i = 1, \ldots, n$. The measurement error covariance matrix $\boldsymbol{\Sigma}_u$ is assumed to be a diagonal matrix, with element $[\boldsymbol{\Sigma}_u]_{ii} = \gamma\,[\boldsymbol{\Sigma}_x]_{ii}$, where $[\boldsymbol{\Sigma}_x]_{ii}$ is the variance of the dimension x^i. In other words, the constant γ controls the noise-to-signal ratio on each variable. For the purposes of simulation, we assume the amount of measurement error to be known.

To generate the true data we use the *huge* package [65]. We inspect two different types of graphs, referred to as *hub* and *random*; we expand on these below where ω_{ij} denotes the (i, j) element of $\boldsymbol{\Omega}_x$.

1. Hub: For $d/20$ groups, $\omega_{ij} = \omega_{ji} = 1$ if in the same group. $\omega_{ij} = 0$ otherwise.

2. Random: For $1 \leq i < j \leq d$, $\omega_{ij} = 1$ with probability $\frac{3}{d}$, 0 otherwise.

We illustrate the stuctures in Figure 2.1.

Each model was generated with $n = 100$ observations. We inspect each model for $d =$

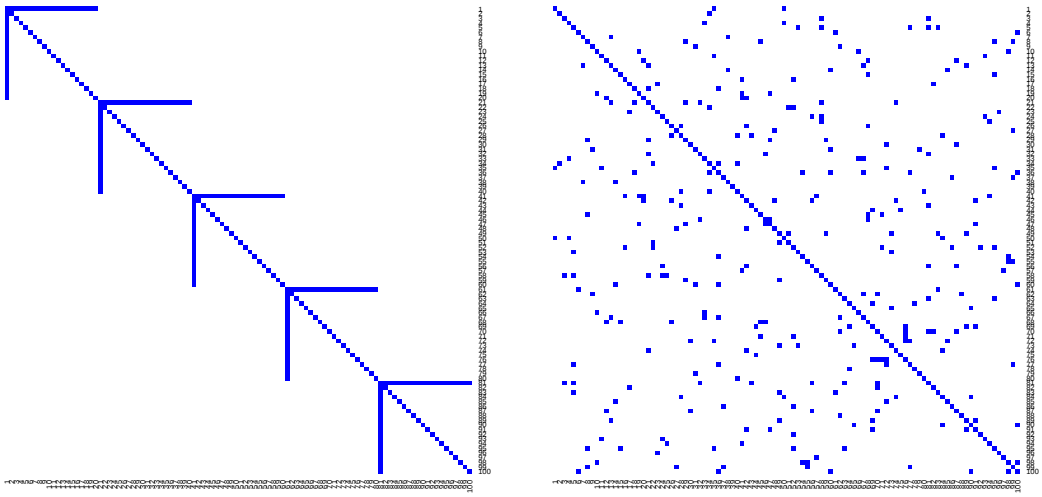

Figure 2.1: Graphical representation for $d = 100$ of the hub (left) and random (right) structure, respectively. While the hub structure is fixed for a given d, the random graph is subject to change due to the generation process.

$\{100, 200\}$ and $\gamma = \{0.1, 0.25, 0.5\}$. The amount of correction-imputations was set to be 50, with the first 20% discarded as burn-in; we note that we inspected 25 and 100 imputations with the same percentage of burn-in samples with minimal differences in output. Each setting was replicated 50 times, and the final results are the average of these replicates. Hyperparameter tuning was done as described in Section 2.4.2. Because measurement error is often ignored in the context of GGMs, our simulations also provide perspective onto the negative effect that measurement error can impose on model performance.

To inspect model performance, we examine both the estimated precision matrix and the ability to do variable selection of BAGUS on the true data (true), BAGUS on the contaminated data (naive), and our IRO-BAGUS methodology on the contaminated data (corrected). For each estimated precision matrix $\hat{\mathbf{\Omega}}_x$, estimation error is measured by $||\hat{\mathbf{\Omega}}_x - \mathbf{\Omega}_x||_F$, and variable selection is evaluated by different metrics involving the true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) are reported: specificity (SPE), sensitivity (SEN), precision (PRE), accuracy (ACC), and Matthews correlation coefficient

(MCC); these values are defined as

$$\begin{aligned}
\text{SPE} &= \frac{\text{TN}}{\text{TN} + \text{FP}}, \qquad \text{SEN} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \\
\text{PRE} &= \frac{\text{TP}}{\text{TP} + \text{FP}}, \qquad \text{ACC} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \\
\text{MCC} &= \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{(\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN})}.
\end{aligned}$$

Additionally, we also report the area under the ROC curve (AUC), which gives insight into the amount of seperation of the classification. These different metrics give insight into the tradeoffs and gains of each setting.

2.5.2. Simulation Results

Table 2.1 and Table 2.2 present the results for the hub and random structure, respectively. To begin, we note the effect of the increasing measurement error. This can be observed by examining the growing difference in the performance of the true and naive model when holding d fixed and increasing the amount of contamination. Focusing on the hub structure, a decrease in the quality of selection and estimation can be observed for each setting, which grows worse with more contamination; for example, when $d = 100$ the AUC drops from 0.95 to 0.93 for $\gamma = 0.1$, but drops from 0.95 to 0.84 for $\gamma = 0.5$. The selection accuracy metrics with respect to the prespecified 0.5 cut-off show drops in performance of around 50%. The estimated precision matrix from the naive grows worse with measurement error, and is also about 50% worse when the signal-to-noise is 0.5.

We now turn attention to the performance of the correction step. First, take note of the first five metrics which are based on the confusion matrix for the 0.5 cutoff threshold. Averaging across the IRO iterations was expected to result in an analysis that favored identifying relationships that were more certain, which can be observed by inspection of the precision (PRE). The gains from the precision are most notable as d grows larger, and more pair-wise relationships exist; when $d = 200$, we note nearly 10% and 50% performance gains in the

γ	d	Model	SEN	SPE	PRE	ACC	MCC	FROB	AUC
0.1	100	True	1.00	0.65	0.85	0.99	0.73	5.11	0.95
		Naive	1.00	0.50	0.76	0.99	0.61	6.81	0.93
		Corrected	1.00	0.51	0.78	0.99	0.62	6.17	0.97
	200	True	1.00	0.67	0.77	0.99	0.71	7.36	0.94
		Naive	1.00	0.51	0.69	0.99	0.59	9.63	0.92
		Corrected	1.00	0.51	0.71	0.99	0.60	8.68	0.96
0.25	100	True	1.00	0.66	0.84	0.99	0.74	5.09	0.95
		Naive	0.99	0.38	0.60	0.98	0.47	8.54	0.90
		Corrected	1.00	0.36	0.68	0.98	0.49	7.71	0.94
	200	True	1.00	0.67	0.78	1.00	0.72	7.29	0.94
		Naive	1.00	0.40	0.52	0.99	0.45	12.10	0.90
		Corrected	1.00	0.37	0.62	0.99	0.48	10.68	0.95
0.5	100	True	1.00	0.66	0.85	0.99	0.74	5.03	0.95
		Naive	1.00	0.20	0.50	0.98	0.31	9.67	0.84
		Corrected	1.00	0.20	0.70	0.98	0.37	8.74	0.89
	200	True	1.00	0.68	0.77	0.99	0.72	7.36	0.94
		Naive	1.00	0.20	0.37	0.99	0.27	13.70	0.83
		Corrected	1.00	0.17	0.59	0.99	0.31	12.53	0.89

Table 2.1: Simulation results for the hub graph structure, as specified in Section 2.5.1. For each signal-to-noise ratio and d, the true, naive, and corrected models are shown for metrics defined in Section 2.5.1.

precision for signal-to-noise ratios of 0.25 and 0.5, respectively. In both the hub and random structure the naive and corrected models perform similarly in terms of the sensitivity, specificity, accuracy, and MCC.

It seems at first glance that the selection performance, ignoring the precision, of the correction procedure is comparable to the naive, but these discrepencies can be attributed to the prespecified inclusion cut-off on the $\boldsymbol{P}$ matrix. In practice it can often be more reasonable to rank order the inclusion probabilities to identify relationships for further investigation in future experiments. With this in mind, we turn to the performance with respect to the AUC where consistent improvements can be seen for the hub and random structure in most all settings. The AUC helps understand the amount of seperation found in the model across all thresholds, which helps justify that the correction step is making improvements in seperating the classes for the true relationships as AUC improvements are seen in all but the random graph with $d = 200$ and signal-to-noise ratio of 0.5.

We note two items in regard to the AUC. First, the AUC of the corrected model sometimes

Amt. ME	d	Model	SEN	SPE	PRE	ACC	MCC	FROB	AUC
0.1	100	True	1.00	0.42	0.84	0.98	0.59	4.61	0.89
		Naive	1.00	0.32	0.80	0.98	0.50	5.34	0.88
		Corrected	1.00	0.32	0.80	0.98	0.50	5.04	0.91
	200	True	1.00	0.36	0.76	0.99	0.52	6.72	0.86
		Naive	1.00	0.30	0.66	0.99	0.44	7.61	0.85
		Corrected	1.00	0.28	0.68	0.99	0.43	7.25	0.90
0.25	100	True	1.00	0.45	0.86	0.98	0.61	4.66	0.90
		Naive	1.00	0.27	0.70	0.97	0.42	6.68	0.85
		Corrected	1.00	0.23	0.75	0.97	0.40	5.72	0.88
	200	True	1.00	0.37	0.75	0.99	0.52	6.77	0.86
		Naive	1.00	0.23	0.52	0.99	0.34	9.04	0.82
		Corrected	1.00	0.18	0.60	0.99	0.32	7.95	0.86
0.5	100	True	1.00	0.43	0.85	0.98	0.59	4.65	0.89
		Naive	1.00	0.14	0.55	0.97	0.26	7.71	0.79
		Corrected	1.00	0.09	0.67	0.97	0.24	6.49	0.79
	200	True	1.00	0.37	0.76	0.99	0.53	6.74	0.86
		Naive	1.00	0.12	0.39	0.98	0.21	10.42	0.77
		Corrected	1.00	0.06	0.56	0.99	0.18	8.92	0.78

Table 2.2: Simulation results for the random graph structure, as specified in Section 2.5.1. For each signal-to-noise ratio and d, the true, naive, and corrected models are shown for metrics defined in Section 2.5.1.

outperforms the true model. In particular, this happens in the hub structure when the amount of measurement error is 0.1. This can be attributed to the measurement error in models that are easily identified. Second, in the random structure with the amount of measurement error being 0.5, the corrected model does not make substantial improvements in results over the naive model. We note the difficulty of this setting, as the random structure often performs worse than other structures in identification, and now we add more noise via the contamination. With a relatively small sample size, this noise is difficult to overcome.

Finally, we note the quality of the estimated precision matrix, as measured by Frobenious norm of the difference. In every setting for both the hub and random matrices, the corrected model outperforms the naive model's estimate of the precision matrix. In the hub structure this improvement is often of the order of 15-20% better, while in the random structure a 10% improvement is generally observed. If the intent of the analysis is to use the estimated precision matrix in downstream analysis, this can result in more refined results.

2.6. Data Analysis

A common source of noise in analysis involving gene expression datasets is measurement error [42]. Gaussian graphical models have been employed to inspect the relationship of different genes in varying experiments [27]. We illustrate our methodology using an Affymetrix microarray dataset containing 144 subjects of favorable histology Wilms tumors hybridized to the Affymetrix Human Genome U133A Array [24]. The data is publicly available on the GEO website, dataset GSE10320 uploaded 1/30/2009. A feature of Affymetrix data, and many other gene expression measurement platforms, is the use of multiple probes for each gene for each patient, giving replicate measurements for each patient's gene measurement. The replicates for each patient enable an estimate of the measurement error, where we again assume the amount of contamination is independent across genes.

We follow the preprocessing steps taken in [48] and [35], which used this study in the context of measurement error in variable selection for linear models. The process begins by processing the raw data with the Bayesian Gene Expression (BGX) package [55]. BGX creates a posterior distribution for the log-scale expression level of each gene in each sample. The study recorded measurements for 22283 different genes.

To remove unnecessary computational burden, we reduced the number of genes by applying four different filters in the following order. The first filter removes expression values that do not have a corresponding Entrez gene ID in the NCBI database [36]. The second filter removes expression values with low variability by requiring at least 25% of samples to have intensities above 100 fluorescence units. The third filter removes expression values with low variability by requiring the interquartile range to be at least 0.6 on the log scale. The last filter removes expression values that have have an error to signal to noise ratio greater than 0.5, which we discuss in more depth below. After filtering, there were 273 expression values remaining for the analysis.

Now, we discuss how we estimate the measurement error of each gene. We assume that the measurement error variance is constant across patients for a given gene. We also assume

that the measurement error is independent for each gene, and need not be equal for each gene. Let $\hat{\boldsymbol{\mu}} = (\hat{\mu}_{1j}, \ldots, \hat{\mu}_{nj})^T$ denote the estimated vector of the patient's gene expression levels for gene j. Further, let $\bar{\mu} = n^{-1} \sum_{j=1}^{n} \hat{\mu}_{ij}$ and $\hat{\sigma}_j^2 = n^{-1} \sum_{j=1} (\hat{\mu}_{ij} - \bar{\mu}_j)^2$ denote the mean and variance of each gene, respectively. For patient i, standardized measurements are given by $\boldsymbol{W}_i = (W_{i1}, \ldots, W_{ip})$, calculated as $W_{ij} = \hat{\sigma}_j^{-1}(\hat{\mu}_{ij} - \bar{\mu}_j)$ for each $j = 1, \ldots, 273$.

Let $\text{var}(\hat{\mu}_{ij})$ denote the posterior variance of the estimated distribution of patient i's gene j. These estimates are then combined as $\hat{\sigma}_{u,j}^2 = n^{-1} \sum_{i=1}^{n} \text{var}(\hat{\mu}_{ij})$. The measurement error covariance matrix of the standardized data $\boldsymbol{W}$ is then estimated by diagonal matrix $\hat{\boldsymbol{\Sigma}}_u$, where $(\hat{\boldsymbol{\Sigma}}_u)_{j,j} = \hat{\sigma}_{u,j}^2 / \hat{\sigma}_j^2$ for $j = 1, \ldots, p$ and off-diagonal elements are 0. The fourth filter can be now formalized, where genes are removed if $\hat{\sigma}_{u,j}^2 \geq 0.5\hat{\sigma}_j^2$; i.e. only genes with a noise-to-signal ratio less than 1 are kept for the analysis.

The original BAGUS algorithm and the IRO-BAGUS algorithm were run for the remaining genes found after filtering. As with the simulations, the corrected BAGUS found fewer conditional pair-wise relationships; for this data set, the IRO-BAGUS and IRO-BAGUS found 1045 and 552 conditional pair-wise relationships, respectively. Of the 1045 naive pair-wise relationships, 42% were also found in the corrected pair-wise relationships; similarly, of the 552 corrected conditional pair-wise relationships, 80% were found in the naive model. The large percentage overlap of relationships in the corrected model with relationships in the naive model suggests that most relationships in the corrected model are true relationships. Conversely, the small percentage overlap of relationships in the naive model with those in the correct model suggests that the naive model is finding many false positive relationships. We illustrate the conditional pair-wise dependencies of the genes in Figure 2.2. The naive analysis is shown on the left and the corrected on the right, where the green edges signify relationships found by both procedures and purple edges signify procedure specific relationships.

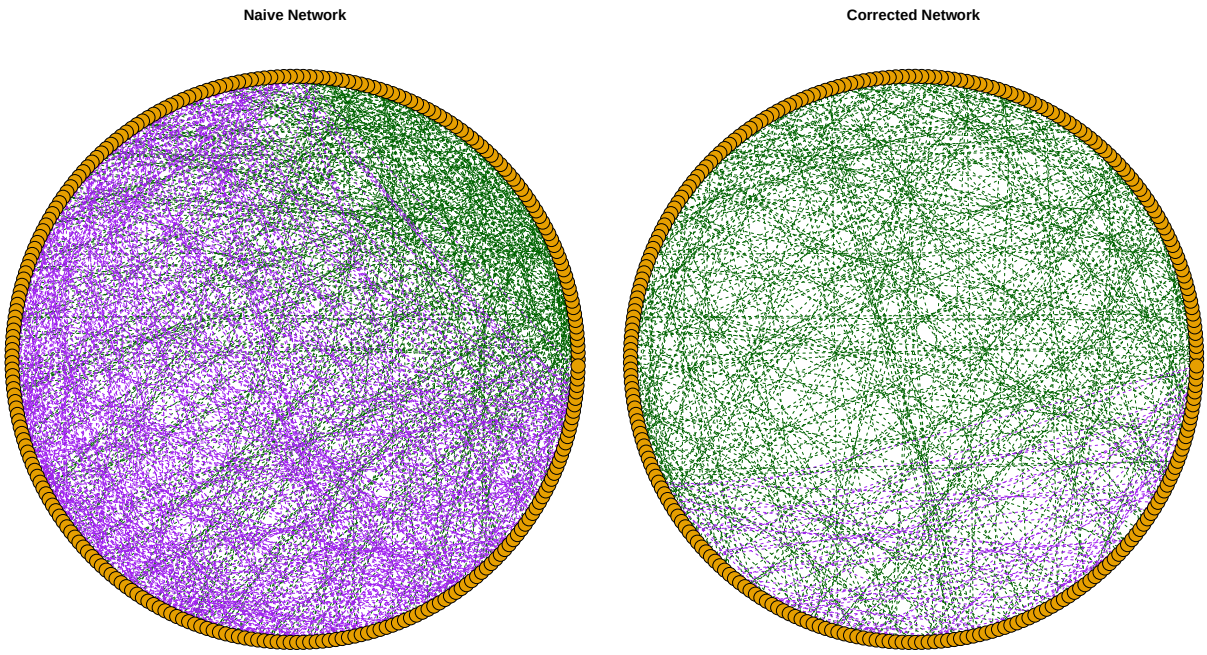

Figure 2.2: The conditional pair-wise relationships for each of the 273 genes remaining after filtering from the Wilms tumor study. Each edge represents a conditional pair-wise dependency between two nodes. The left shows the naive analysis, not correcting for measurement error, and the right shows the corrected analysis, correcting for measurement error. Green edges signify edges found on both graphs, and purple signifies analysis specific edges.

2.7. Conclusion

We proposed a correction methodology for Gaussian graphical models when contaminated with additive measurement error. The core solution to the problem involves using the imputation-regularization algorithm to generate the true values of underlying process with a consistent estimate of the precision matrix. This provides a consistent, positive-definite estimate of the true precision matrix, which, as simulations illustrate, removes many false positive pair-wise relationships. Additionally, we show marked improvements in the AUC of the threshold matrix, indicating better separation of the underlying relationships. From a practitioner's point of view, this allows for more reliable downstream analysis and a stronger set of results from which to continue research.

To our knowledge, the novel imputation-regularization algorithm has yet to be used for problems pertaining to contaminated data. This provides an avenue of future research for more a practical issue in high-dimensional problems, measurement error, which is starting to gain attention. Moreover, many practical issues still remain in the Gaussian graphical

model context, such as the tuning of hyperparameters and the interpretation of the output from the Gibbs sampler-like IRO algorithm. Another potiental avenue of research to pursue is when the amount of measurement error is unknown and not assumed independent. In this case, sparsity would need to be imposed on $\mathbf{\Omega}_u$ in conjunction with Ω_x, posing a challenging, but useful, computational procedure.

Chapter 3

A Simple Correction Procedure for Contaminated High-Dimensional Generalized Linear Models

3.1. Introduction

Complex, high-dimensional data sets have become the norm for many fields where it is often of interest to uncover underlying structures and to estimate the effect size of a given relationship between the observed variables. For instance, in a microarray experiment it may be of value to identify which genes are related to some quantitative outcome or if a particular gene influences a disease. Statistical regularization procedures have been essential to addressing these fundamental problems. In particular, when the number of variables p is larger than the sample size n, traditional methods, such as least squares regression, can no longer be used due to identifiability issues. Hence, regularization procedures, like the Lasso [53] and the Minimax Concave Penalty (MCP) [64], have become necessary tools for practitioners to identify patterns in their studies for a wide variety of problems [20].

For $i = 1, \ldots, n$, consider the generalized linear model (GLM) for independent and identically distributed pairs of responses and covariates $(y_i, \boldsymbol{x}_i)$, such that

$$\mathbb{E}(y_i) = f(\boldsymbol{x}_i^T \boldsymbol{\beta}) \tag{3.1}$$

for covariates, $\boldsymbol{\beta} \in \mathbb{R}^p$, and inverse-link function, f [33]. The regularized GLM aims to minimize the objective

$$Q(\boldsymbol{\beta}; \boldsymbol{X}, \boldsymbol{y}, \lambda) = \boldsymbol{\mathcal{L}}(y; \boldsymbol{x}, \boldsymbol{\beta}) + P(\boldsymbol{\beta}; \lambda) \tag{3.2}$$

with respect to $\boldsymbol{\beta}$, where $\mathcal{L}(y; \boldsymbol{x}, \boldsymbol{\beta})$ is the negative log-likelihood function and $P(\boldsymbol{\beta}; \lambda)$ is a penalty function on the coefficients. The regularization parameter λ determines the overall level of sparsity, and is typically tuned with cross-validation [16]. This formulation captures most types of data, including continuous, categorical, and count. Adaptations of the GLM have been well studied for many common penalties [56], and the objective in Equation (3.2) has many implemented procedures for a wide variety of problems [60] [4].

In addition to regularized procedures' well documented empirical performance, favorable theoretical properties, such as selection consistency, have been well studied [20]. These properties, however, make the assumption that the observed covariates are perfectly measured, which, in many contexts, is not a realistic assumption. Furthermore, lack of incorporation of measurement error can lead to biased estimates and misleading outcomes. For $i = 1, \ldots, n$, assume that, instead of observing true data $\boldsymbol{x}_i$, we instead observe a contaminated observation $\boldsymbol{w}_i = \boldsymbol{x}_i + \boldsymbol{u}_i$, such that added noise $\boldsymbol{u}_i$ is mean $\mathbf{0}_p$ with covariance $\boldsymbol{\Sigma}_u$. For instance, in microarray experiments there are many possible sources for random error to be incorporated naturally into the data collection process [42]. While the Affymetrix microarray itself is manufactured under controlled conditions according to precise specifications, the genetic material prepared as the microarray sample is subject to propagation of error. RNA preparation, for example, takes at least three days, and requires up 15 steps per day. At any one of these steps, error or contaminants leading to error could be introduced [38]

We consider high-dimensional variable selection and estimation for GLMs in the context of measurement error. In particular, we address the additive measurement error setting, which is known to cause a decrease in selection and estimation quality if not corrected [32] [48]. Error-in-variables (EIV) regression has been a known issue in statistics, and has been well studied in a plethera of contexts [7]. The effect of mismeasured covariates in EIV regression are biased estimates of the regression coefficients and a higher Type I error rate. An analysis that incorporates and corrects for measurement error will aim to result in consistent estimates with fewer false positives. This correction, however, will come with

decreases in power and model efficiency.

To overcome the limitation of not directly observing the variables of interest, but rather contaminated variations, we make use of the Imputation-Regularized Optimization (IRO) algorithm [29]. The IRO-algorithm was proposed as a technique for missing data in the high-dimensional setting, which notably gives a flexible framework with consistency guarantees for latent variables. Recently, the IRO-algorithm was used in the context of estimating Gaussian graphical models with mismeasured observations [5]. The procedure was shown to be asymptomatically consistent; in addition it greatly reduced the number of false positives found in the selection process and reduced the overall estimation error. We provide an extension to this process for common types of generalized linear models in the presence of measurement error. Our goal is to provide a simple framework for high-dimensional measurement error problems that can be implemented using well established tools and procedures.

3.1.1. Literature Review

High-dimensional EIV regression procedures have accumulated much attention due to the fact that the contaminated observations result in inconsistent estimates and poor variable selection [48] [35] [2]. These procedures typically correct for the contamination by incorporating the assumed known or estimable measurement error variability into the optimization [32] [48] [11], or by some pivotal estimation without a well defined likelihood [2] [49]. Notably, these procedures tend to make traditionally convex penalties into non-convex formulations, requiring special care in development of optimization routines to solve them. Moreover, even when these issues have been addressed, model tuning is known to be more difficult as standard cross-validation is not easily applied for contaminated observations [11] [10].

While many of these procedures offer nice theoretical properties for symmetric, continuous responses, few have explored a more general framework for different types of response data. We focus on two well established methods for correcting for measurement error: (1) the Corrected Lasso (CLasso) and (2) The Generalized Matrix Uncertainty Selector (GMUS).

Both CLasso and GMUS were originally established in the Gaussian error case, see [32] and [45], but have been established in the GLM framework by [48] and [49], respectively. The CLasso attempts to account for the bias introduced with the measurement error by incorporating it into the optimization problem with two hyperparameters controlling the size of coefficients [48]. GMUS takes a slightly different approach, limiting the amount of correlation between the response and covariates by bounding the score function with a Taylor series expansion of the residual [49].

In practice, both methods can be hard to tune due to not having a well defined likelihood. In the GLM case, CLasso and GMUS both use an elbow-plot, as explained in [49], to determine the amount of regularization, which requires user input and does not always give a clear amount to to be used in the final model. While GMUS is a convex optimization, the CLasso is not. Originally, [32] show favorable convergence properties in the Gaussian residual case, we find that the GLM solution from CLasso in a popular implementation does not always share these nice properties. Finally, we note that CLasso and the proposed method require some knowledge of the measurement error variability, whereas GMUS does not. However, in many applications, like gene expressions, replicates are taken with common practice, which allows for estimation of the variability of the contamination. Hence, lacking the ability to incorporate the measurement error variability could be a disadvantage for various settings.

3.1.2. Overview

The outline for the remainder of this work is as follows. In Section 3.2, we establish the additive measurement error formulation and the IRO-algorithm. We show how the IRO-algorithm can be used in solutions pertaining to the context of contaminated linear models and we give practical considerations for its usage. Section 3.3 establishes required imputation procedures for continuous, categorical, and count data. This is done by assuming the response has parametric form of Gaussian, binomial, and negative binomial distributions, respectively. A simulation study is then presented in Section 3.4, illustrating our method's

performance in Gaussian and binomial linear regression. Finally, a data analysis is presented in Section 3.5, illustrating the proposed method with two other correction procedures on an experiment using microarray gene expressions to find underlying causes of a tumor relapsing. All derivations and further results can be found in the Appendix.

3.2. The IRO-Algorithm for EIV Regression

Consider the following additive measurement error formulation that will persist for the remainder of the paper. Let $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^T \in \mathbb{R}^{n \times p}$ be n independent and identical realizations of a p-dimensional random variable, where, for covariance matrix $\boldsymbol{\Sigma}_x$, $\boldsymbol{x}_i \sim N(\mathbf{0}_p, \boldsymbol{\Sigma}_x)$. Instead of directly observing realization $\boldsymbol{x}_i$, we observe $r_i \geq 2$ contaminated replicates. Assume the contaminated observation to be related additively to the true realization, where

$$\boldsymbol{w}_{ij} = \boldsymbol{x}_i + \boldsymbol{u}_{ij} \tag{3.3}$$

such that $\boldsymbol{u}_{ij} \sim N(\mathbf{0}_p, \boldsymbol{\Sigma}_u)$ for $i = 1, \ldots, n$ and $j = 1, \ldots, r_i$; note that, by independence of $\boldsymbol{x}_i$ and $\boldsymbol{u}_{ij}$, that $\boldsymbol{w}_{ij} \sim N(\mathbf{0}_p, \boldsymbol{\Sigma}_x + \boldsymbol{\Sigma}_u)$. Denote the collection of observation i's replicates as $\boldsymbol{W}_i = (\boldsymbol{w}_1, \ldots, \boldsymbol{w}_{r_i})^T$, and let $\bar{\boldsymbol{w}}_i$ to be the average of the replicates for observation i. Without loss of generality, we assume the measurement error is centered at $\mathbf{0}$.

3.2.1. The Imputation Regularization Optimization Algorithm

The Imputation-Regularized Optimization (IRO) algorithm was recently introduced in the context of high-dimensional variable selection with missing data [29]. The IRO-algorithm provides a much needed procedure for imputation in case where $n < p$, as common methods, like the well known EM-algorithm [13], can fail due to inconsistent or non-unique likelihoods [61]. The IRO-algorithm consists of two iterative steps. At iteration $t = 1, \ldots, T$, missing values, $\boldsymbol{z}_m$, are imputed through a predictive density that is conditioned on the observed

data, $\boldsymbol{z}_o$, and the estimated model parameters from the previous iteration, $\boldsymbol{\Delta}^{(t-1)}$, namely

$$\boldsymbol{z}_m^{(t)} \sim \pi\left(\boldsymbol{z}_m | \boldsymbol{z}_o, \boldsymbol{\Delta}^{(t-1)}\right). \tag{3.4}$$

The newly generated values $\boldsymbol{z}_m^{(t)}$ are then used with observed values $\boldsymbol{z}_o$ to estimate the model parameters with a regularized objective function

$$\boldsymbol{\Delta}^{(t)} = \underset{\boldsymbol{\Delta}}{\operatorname{argmin}} \left\{ F(\boldsymbol{\Delta}; \boldsymbol{z}_m^{(t)}, \boldsymbol{z}_o) + P(\boldsymbol{\Delta}; \lambda) \right\}, \tag{3.5}$$

where F denotes the parameter's relationship to the data and P denotes the regularization function with sparsity parameter λ. The two steps are iterated, and, when the optimization in (3.5) is asymptotically consistent, the IRO-algorithm forms a valid Markov chain that provides an asymptotically consistent estimate of the high-dimensional variables under mild conditions [29].

3.2.2. The I-Step for High-Dimensional EIV Regression

Measurement error is similar in nature to missing data in that the missing values are related by some underlying density like the contaminated variable. Regardless of procedure, the conditional density for $\boldsymbol{x}_i | \boldsymbol{W}_i, \boldsymbol{\Omega}_u, \boldsymbol{\Omega}_x$ must be estimated for all $i = 1, \ldots, n$. Recently, the IRO-algorithm was used in a measurement error correction procedure for Gaussian graphical models, which estimates the precision matrix $\boldsymbol{\Omega}_x$ with an assumed known or estimable $\boldsymbol{\Omega}_u$ [5]. Going forward we will refer to a procedure using the IRO-algorithm to correct for measurement error as an IRO-adjusted procedure. Referring back to the aforementioned contaminated model in (3.3), the predictive density used to compute the imputation is the full conditional found in Normal-Normal models in Bayesian inference,

$$\pi(\boldsymbol{x}_i | \boldsymbol{W}_i, \boldsymbol{\Omega}_u, \boldsymbol{\Omega}_x) \sim N(r_i \boldsymbol{\Lambda} \boldsymbol{\Omega}_u \bar{\boldsymbol{w}}_i, \boldsymbol{\Lambda}), \tag{3.6}$$

where $\boldsymbol{\Lambda} = (\boldsymbol{\Omega}_x + \boldsymbol{\Omega}_u)^{-1}$, as shown in the Appendix B.1.1.

We consider the high-dimensional EIV regression problem for GLMs with response $y \sim \mathcal{D}$ and nuissance parameters $\boldsymbol{\Theta}$. Here, we assume a relationship exists between the expectation of the response, y_i, and a function of the linear combination of covariates, $\boldsymbol{x}_i$. Namely, we formulate the model

$$\mathbb{E}_{\mathcal{D}}(y_i; \boldsymbol{\Theta}) = f(\boldsymbol{x}_i^T \boldsymbol{\beta}) \tag{3.7}$$

for the inverse-link function f and sparse coefficients $\boldsymbol{\beta} \in \mathbb{R}^p$. The sparsity of the coefficients implies that most are 0. Denote the number of non-zero coefficients by $q = ||\boldsymbol{\beta}||_0$, where $q \leq n$ and, typically, $q \ll p$. Instead of observing pair $(y_i, \boldsymbol{x}_i)$, the covariate is observed with (replicated) contamination $(y_i, \boldsymbol{W}_i)$. To implement the IRO-algorithm for the EIV regression problem the imputation step in (3.6) must be adjusted to include the response model. The imputation distribution is altered, up to a normalizing constant, as

$$\pi(\boldsymbol{x}_i | y_i, \boldsymbol{W}_i, \boldsymbol{\Omega}_x, \boldsymbol{\Omega}_u, \boldsymbol{\beta}, \boldsymbol{\Theta}) \propto \pi(y_i | \boldsymbol{x}_i, \boldsymbol{\beta}, \boldsymbol{\Theta}) \pi(\boldsymbol{x}_i | \boldsymbol{W}_i, \boldsymbol{\Omega}_x, \boldsymbol{\Omega}_u), \tag{3.8}$$

where the distribution of $\boldsymbol{x}_i | \boldsymbol{W}_i, \boldsymbol{\Omega}_u, \boldsymbol{\Omega}_x$ is as in (3.6). For well specified densities for each function, the distribution for the imputation step in (3.8) is known and easily sampled, which will be explored in Section 3.3. Once $\boldsymbol{X}$ has been imputed, an estimate of $\boldsymbol{\Omega}_x$ and $\boldsymbol{\beta}$ is obtained, and the process repeated. The general procedure is presented in Algorithm 1.

Algorithm 1 The IRO-adjusted Procedure for Contaminated GLMs

1: Set number of IRO iterations, T
2: Input known $\boldsymbol{\Omega}_u$ or obtain $\hat{\boldsymbol{\Omega}}_u$ using replicate data
3: Obtain initial estimate for $\boldsymbol{\beta}^{(0)}$ and $\boldsymbol{\Omega}_x^{(0)}$ using $\bar{\boldsymbol{W}}$
4: **for** t = 1, . . . , T **do**
5: **for** i = 1 , . . . , n **do**
6: draw $\boldsymbol{x}_i^{(t)} \sim \pi(\boldsymbol{x}_i | y_i, \boldsymbol{W}_i, \boldsymbol{\Omega}_x^{(t-1)}, \hat{\boldsymbol{\Omega}}_u, \boldsymbol{\beta}^{(t-1)}, \boldsymbol{\Theta}^{(t-1)})$ ▷ Impute
7: Estimate $\boldsymbol{\Omega}_x^{(t)}$ with $\boldsymbol{X}^{(t)}$
8: Estimate $\boldsymbol{\beta}^{(t)}$ and $\boldsymbol{\Theta}^{(t)}$ ▷ Regularize

3.2.3. The RO-Step for High-Dimensional EIV Regression

Once the imputation step has been performed, then the remaining parameters must be estimated from the imputed realizations. Beginning with $\boldsymbol{\Omega}_x$, the precision matrix of the true underlying data, it is tempting to estimate the covariance directly and then invert the estimated covariance. However in the setting where $n < p$, the estimated covariance is likely to not be of full-rank, and hence inversion would not be possible due to $\hat{\boldsymbol{\Sigma}}_x$ being singular. Additionally, even if one could reasonably estimate $\boldsymbol{\Sigma}_x$, inversion is computationally expensive. The Gaussian graphical model literature has given several ways to estimate $\boldsymbol{\Omega}_x$ directly with a regularization term to impose sparsity, which could then estimate a full rank matrix [17].

While estimating the off-diagonal elements of $\boldsymbol{\Omega}_x$ is appealing, many regularization procedures assume independence among covariates. Even if the assumption is not strictly made, few regularization procedures make use of the dependence structure among the covariates; though some exceptions do exist [62]. Disregarding the dependency between covariates allows for estimation of only the diagonal of $\boldsymbol{\Omega}_x$. This results in computational gains in the imputation step, as explained in Section 3.3, and saves a costly optimization of $\boldsymbol{\Omega}_x$. We observe in our simulations with dependent covariates that estimating only the diagonal of $\boldsymbol{\Omega}_x$ performs well.

Many procedures have been developed to estimate coefficients in regularized general linear models [37]. Any method which is consistent will be adequate for the regularization step in estimating the coefficients at each iteration. The more accurate the regularization method, the better the imputation. Of particular note is the ability to estimate the nuisance parameter $\boldsymbol{\Theta}$, which is required for the imputation step. This is a known problem in, for instance, Gaussian linear regression, where the underlying model variability affects the selection quality [1]. The general IRO-adjusted procedure for mismeasured random variables is to alternate between imputation, as in equation (3.8), and optimizing parameters $\boldsymbol{\Omega}_x, \boldsymbol{\beta}$, and $\boldsymbol{\Theta}$.

3.2.4. Computational Considerations

Most of the regularized optimization procedures for GLMs require some hyperparameter tuning. For example, consider the Lasso penalty's Lagrangian form, then the optimization in (3.2) will be such that $P(\boldsymbol{\beta}; \lambda) = \lambda||\boldsymbol{\beta}||_1$. The hyperparameter λ directly affects the output by controlling the amount of sparsity, and hence needs to be tuned. We handle this hyperparameter tuning at each iteration of the IRO-algorithm via conventional procedures like k-fold cross-validation, which for many competing methods is unavailable. Using the standard tool set makes analysis for a practitioner easier as measurement error invalidates traditional methods, and additional methods like [10] are incorporated into the procedure. Moreover, competing methods are often in a position of tuning a grid of hyperparameters [2]. This adds to their computational costs to find an optimal solution, which may not be plausible depending on the difficultly of the optimization. Moreover, this adds to the difficultly of use for practitioners, and a higher chance of misapplication or misinterpretation.

We briefly note the similarity of the IRO-algorithm and Gibbs samplers from Bayesian literature [47]. Gibbs samplers require obtaining the distribution of each random variable conditioned on the all other random variables in the model, known as the full conditional distribution. These distributions are then used to generate values of that random variable, conditioned on the most recently generated value of the other random variables. This is similar to the IRO-algorithm, which replaces sampling of some variables with an optimization step. As such, the massive amount of literature that has been developed for Gibbs sampling is applicable to procedures using the IRO-algorithm. This was illustrated in [29], where the well-known Gelman-Rubin diagnostics [19] were used to illustrate convergence of the IRO-algorithm. We note that both samplers can take some time to reach reasonable areas of the posterior distribution. While a good starting value helps, it is still often beneficial to discard some initial amount of iterates as burn-in. We make use of this practice in our implementation of IRO-algorithm.

The IRO-algorithm estimates a set of coefficients at each iteration, and the differences in the estimated coefficients may be interpreted as the amount of variation added into the estimation process as a result of the contaminated observations [29]. With mild conditions on the regularization procedure and variability of the data, the findings of [29] show that the IRO-algorithm gives a consistent estimate of the optimized parameters in each iteration. Moreover, the results of [5] extend this result to the measurement error scenario. A typical final estimate would make use of all iterations, such as taking the average estimated parameter from each iteration. However, the average of multiple sparse vectors is not guaranteed to be sparse, which does not give an easy interpretation of the variable selection. Intuitively, spurious coefficients that appear in the model ought to do so a few number of times; therefore a trimmed mean could be used. Alternatively, we find using the median of each estimated coefficient as the final estimate to give reliable estimates, as illustrated in Section 3.4.

3.3. IRO-Procedures for Some Contaminated GLMs

In this section we explore imputation steps for three common types of response data: continuous, categorical, and count. This is done by determining the necessary form of Equation (3.8) for responses distributed as a Gaussian, binomial, and negative binomial distribution. These distributions are standard for GLMs, and cover most use cases. We illustrate that the imputation can be accomplished from known, parameterized distributions, which makes the sampling painless. Additionally, we address computational considerations of the imputation step. While we focus here on closed form distributions to be used in the imputation step, there may not always be well-known distributional forms available for every class of model. Many procedures exist for approximating distributions, such as the Integrated Nested Laplace Approximation [46]. Samples drawn from the output of these methods could be used to estimate unknown distributional forms given by other models. All derivations are deferred to the Appendix B.1.

3.3.1. Gaussian Linear Regression

The natural starting point is continuous data with Gaussian linear regression. Here, we assume the response follows the familiar model, $y_i \sim N(\boldsymbol{x}_i^T\boldsymbol{\beta}, \sigma^2)$ for all $i = 1, \ldots, n$. The imputation step can be shown to be

$$\boldsymbol{x}_i | y_i, \boldsymbol{\beta}, \boldsymbol{W}_i, \boldsymbol{\Omega}_u, \boldsymbol{\Omega}_x \sim N\left(\boldsymbol{\Lambda}_G\left(r_i\boldsymbol{\Omega}_u\bar{\boldsymbol{w}}_i + \frac{y_i}{\sigma^2}\boldsymbol{\beta}\right),\ \boldsymbol{\Lambda}_G\right), \tag{3.9}$$

where $\boldsymbol{\Lambda}_G = (\boldsymbol{\Lambda}^{-1} + \sigma^{-2}\boldsymbol{\beta}\boldsymbol{\beta}^T)^{-1}$ for $\boldsymbol{\Lambda}$ as defined in (3.6). We note the impact of quality estimates for $\boldsymbol{\beta}$ and σ^2, which will be used iteratively for the imputations. Many regularization procedures do not incorporate the residual variability into the estimation. If one is confident in the quality of the estimates directly from the regularized model, then the residual variance could be estimated as

$$\hat{\sigma}^2 = \frac{||\boldsymbol{y} - \boldsymbol{X}^{(t)}\boldsymbol{\beta}^{(t)}||_2^2}{n - \hat{q}}, \tag{3.10}$$

where $\hat{q}$ is the number of estimated, non-zero coefficients. However, many procedures' performance is known to become worse when the model error variance is not 1 [1]. Hence, if using such a method, like Lasso, it is often beneficial to instead use variants that incorporate the error variance, like the scaled Lasso [50], into the model during the optimization procedure.

We remark on the computation of the imputation step, which requires inverting the sum of a full rank and rank-1 matrix. If the features are modeled as independent, implying $\boldsymbol{\Omega}_x$ and $\boldsymbol{\Omega}_u$ are diagonal, then some computational gains can be found by noting that $\boldsymbol{\beta}\boldsymbol{\beta}^T$ is a rank-1 matrix. A typical procedure for generating p-dimensional Gaussian data is to generate p independent standard Normal variables, and then to multiply this vector by the Cholesky decomposition of the covariance matrix. The Cholesky decomposition of a diagonal matrix is simply the square root of the diagonal elements, which can then be updated by $\boldsymbol{\beta}\boldsymbol{\beta}^T$ in $\mathcal{O}(p^2)$ time instead of $\mathcal{O}(p^3)$ time if done directly. There is not an easy way to address the problem of generating the imputation step when $\boldsymbol{\Omega}_x$ or $\boldsymbol{\Omega}_u$ is not diagonal without making assumptions on its form. However, this is a well known problem in Bayesian literature, and

recent advances, such as [3], may prove applicable to our situation in the future.

3.3.2. Binomial Linear Regression

We now consider the binomial linear regression setting where covariates are contaminated with measurement error. For each observation i, let $y_i \sim Bern(p_i)$ such that

$$p_i = \frac{e^{\boldsymbol{x}_i^T \boldsymbol{\beta}}}{1 + e^{\boldsymbol{x}_i^T \boldsymbol{\beta}}}. \tag{3.11}$$

Incorporating binomial regression into the IRO-algorithm is not as immediate as in Gaussian linear regression due to the logit function, which maps the linear combination of the covariates to the success probability. The Gaussian setting is conjugate, and hence easily found as in Bayesian inference. Binomial linear regression is known to not have a closed form full conditional distribution due to the logit function, and has been an long-time area of interest in Bayesian literature [22]. Due to the overlap in the IRO-algorithm and Gibbs sampling methodologies, we are able to utilize some of these findings to incorporate into the imputation step.

Specifically, we will make use of a recent advancement in a line of research using data-augmentation to achieve a well-known distribution for the imputation step. Using a newly proposed Pólya-Gamma family of distributions, [41] have been successful in implementing a procedure that allows for a closed-form binomial regression Gibbs sampler. A random variable z is Pólya-Gamma distributed with parameters $b \in \mathbb{R}^+$ and $c \in \mathbb{R}$ if

$$z = \frac{1}{2\pi^2} \sum_{k=1}^{\infty} \frac{g_k}{(k - 1/2)^2 + c^2/(4\pi^2))}, \tag{3.12}$$

where $g_k \sim Ga(b, 1)$ are *iid* Gamma random variables; we denote the the Pólya-Gamma distribution as $z \sim PG(b, c)$. The main result in [41] is that

$$\frac{(e^{\psi})^a}{(1 + e^{\psi})^b} = 2^{-b} e^{\kappa \psi} \int_0^{\infty} e^{-z\psi^2/2} \pi(z) dz, \tag{3.13}$$

where $\kappa = a - b/2$ and $z \sim PG(b, 0)$. Note that when $\psi = \boldsymbol{x}_i^T \boldsymbol{\beta}$, the integrand of (3.13) is the kernel of a Gaussian distribution with respect to $\boldsymbol{x}_i^T \boldsymbol{\beta}$. Hence, the inverse-logit function, as in (3.11), can be expressed as an infinite convolutional mixture of normal and gamma distributions.

Exploiting the mixture representation of the logit function in (3.13), [41] showed that a Gibbs sampler was possible by exploiting the Normal-Normal conjugacy of the prior on the coefficients and Gaussian kernel. This procedure is possible by including the Pólya-Gamma random variable into the sampler. Thus, in addition to needing to impute $\boldsymbol{X}$, our imputation step must also sample $\boldsymbol{z} = (z_1, \ldots, z_n)^T$, which requires the full conditional distribution of z_i. Fortunately, sampling $z_i | \boldsymbol{x}_i, \boldsymbol{\beta}$ is an easy task. [41] showed that

$$z_i | \boldsymbol{x}_i, \boldsymbol{\beta} \sim PG(1, \boldsymbol{x}_i^T \boldsymbol{\beta}), \tag{3.14}$$

which has been illustrated to have an efficient sampling routine [41]. The full conditional to sample each observation's true realization is then Gaussian, namely

$$\boldsymbol{x}_i | y_i, z_i, \boldsymbol{\beta}, \boldsymbol{W}_i, \boldsymbol{\Omega}_u, \boldsymbol{\Omega}_x \sim N\left(\boldsymbol{\Lambda}_B \left(\kappa_i \boldsymbol{\beta} + r_i \boldsymbol{\Omega}_u \bar{\boldsymbol{w}}_i\right), \boldsymbol{\Lambda}_B\right) \tag{3.15}$$

where $\kappa_i = y_i - 1/2$ and $\boldsymbol{\Lambda}_B = (\boldsymbol{\Lambda}^{-1} + z_i \boldsymbol{\beta} \boldsymbol{\beta}^T)^{-1}$. This computation is facilitated by assuming $\boldsymbol{X}$ to be normally distributed, as in the Gaussian linear regression case.

Hence the IRO-algorithm in this context will alternate between imputing $\boldsymbol{X}$ and $\boldsymbol{z}$, then optimizing regression coefficients $\boldsymbol{\beta}$ and the covariate's precision $\boldsymbol{\Omega}_x$. While we have focused on the binomial case, the Pólya-Gamma augmentation can be extended to the multinomial linear regression case [41] [8] [30]. The inclusion of $\boldsymbol{z}$ was shown to create an uniformaly ergodic Gibbs sampler [9], and similar logic should apply to the IRO-adjusted procedure. Additionally, when $\boldsymbol{\Omega}_x$ and $\boldsymbol{\Omega}_u$ are assumed to be diagonal, a similar procedure to the Gaussian linear case can be used to quickly sample from the Normal distribution in (3.15). Unfortunately, this procedure will need to be computed n times, for each coefficient, as the

inverse requires observation specific z_i.

3.3.3. Negative Binomial Linear Regression

Finally, we briefly consider response data being observed as counts. In the GLM framework, the typical procedures for modeling count data are variants of Poisson and negative binomial regression. We opt for the more flexible of the two methods, negative binomial regression, which is less susceptible to overdispersion by not enforcing the mean and variance to be the same. Remarkably, the Pólya-Gamma augmentation works for any distribution in the binomial family, and hence the negative binomial imputations can be implemented in similar nature to Section 3.3.2. The full conditional for the imputing $\boldsymbol{x}_i$ is exactly as in (3.15). However, the augmented variable $\boldsymbol{z}$ is of slightly different form. Appealing to the additive nature of the prior distribution, as in (3.13), for y_i observed counts out of m_i trials, then

$$z_i|\boldsymbol{x}_i, \boldsymbol{\beta}, m_i \sim PG(m_i, \boldsymbol{x}_i^T\boldsymbol{\beta}) \tag{3.16}$$

as shown in [41]. While sampling the full conditional density becomes more costly as m_i grows, efficient routines have been explored to quickly generate samples [40].

3.4. Simulation

Here, we examine the numerical performance of the our proposed estimator for high-dimensional Gaussian and binomial linear regression under different settings. In each setting, five different estimates are compared in terms of estimation quality and variable selection. The first two estimates come from running the same regularization procedure used in the IRO-adjustment, the MCP penalty [64], on (1) the true realizations (Ideal) and (2) the average of the contaminated replicates for each realization (Naive). The MCP was also then used in (3) our implementation of the IRO-algorithm for measurement error (IRO). In addition to comparing the performance to the ideal and naive model, we also inspect two other competing models: (4) the Corrected Lasso (CLasso) [32] and (5) the Generalized

Matrix Uncertainty Selector (GMUS) [49], as described in Section 3.1.1.

All computations were performed in R. To illustrate the ease of incorporating established methodologies into the IRO-adjustment, we make use of a standard regularization package. The MCP penalty was implemented with the R package 'ncvreg'. This package has been developed using efficient coordinate-descent algorithms created for non-convex regularization, and is built with care to appropriately handle possible numerical issues in the optimization. For model tuning, the MCP procedure used the package default 10-fold cross-validation. The Corrected Lasso and GMUS procedures were implemented with the R package 'hdme'. For model tuning, the Corrected Lasso is able to take advantage of cross-validation, and used 10-fold cross-validation for tuning. However, the GMUS procedure requires hand-tuning for each problem by inspecting a scree-plot and choosing the point where the number of zero coefficients stabilizes. We automate this tuning for the simulation study by choosing the the first tuning parameter such that the following two points in the grid give the same number of non-zero coefficients.

For each setting, one of two different sets of coefficients are inspected. The two sets of coefficients are as follows:

1. $\boldsymbol{\beta}_1^* = (1, \ldots, 1, -1, \ldots, -1, 0, \ldots, 0)^T$ where 1 and -1 are repeated 5 times with all $p - 10$ remaining coefficients set to 0,

2. $\boldsymbol{\beta}_2^* = (1, 1/2, 1/3, \ldots, 1/10, 0, \ldots, 0)^T$ where, again, $p - 10$ coefficients are set to 0.

The measurement error was generated from a $\mathbf{0}$ mean Gaussian distribution, with diagonal covariance $\boldsymbol{\Sigma}_u$. To control for the signal-to-noise ratio, we use $\gamma \in \{0.5, 1\}$ as $\text{diag}(\boldsymbol{\Sigma}_u) = \gamma \text{diag}(\boldsymbol{\Sigma}_x)$. Each observation in every case was generated to have $r = 3$ replicates. Each setting, as described in the following sections, was implemented with $n = 400$, $\text{p} = \{100, 500, 1000\}$, and 100 random instances. Additionally, the IRO-algorithm ran for $T = 100$ imputation steps. To inspect the performance of each model, we take the average of the ℓ_2-norm differ-

ence (L2) of the estimated and true coefficients from each replicate within each setting,

$$\ell_2(\hat{\boldsymbol{\beta}}) = \frac{1}{100}\sum_{i=1}^{100} ||\hat{\boldsymbol{\beta}}_i - \boldsymbol{\beta}^*||_2^2,$$

which measures the quality of the estimated coefficients. The variable selection quality is reported by the average number of true positives (TP) and false positives (FP).

3.4.1. Gaussian Linear Regression

We begin by examining Gaussian linear regression. In addition to the MCP penalty, we also inspected the performance using the Scaled Lasso [50], for which we defer discussion and results to the Appendix B.3. Three different data generating processes were considered, where data is generated such that $\boldsymbol{X} \sim N(\mathbf{0}_p, \boldsymbol{\Sigma}_x)$ and $y_i = \boldsymbol{x}_i^T\boldsymbol{\beta} + \epsilon_i$ for $\epsilon_i \sim N(0, \sigma^2)$. The three settings inspect different values of $\boldsymbol{\Sigma}_x, \boldsymbol{\beta}, \text{and } \sigma^2$, and are given by the following:

(G1) $\boldsymbol{X} \sim N(\mathbf{0}, \boldsymbol{\Sigma}_x)$ such that the covariance is diagonal where $\boldsymbol{\Sigma}_x = \boldsymbol{I}$. We use $\boldsymbol{\beta}_2^*$ to define the relationship $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta}_2^* + \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim N(\mathbf{0}, \boldsymbol{I})$.

(G2) $\boldsymbol{X} \sim N(\mathbf{0}, \boldsymbol{\Sigma}_x)$ such that the covariance is diagonal where $\boldsymbol{\Sigma}_x = \boldsymbol{I}$. We use $\boldsymbol{\beta}_1^*$ to define the relationship $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta}_1^* + \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim N(\mathbf{0}, 3\boldsymbol{I})$.

(G3) $\boldsymbol{X} \sim N(\mathbf{0}, \boldsymbol{\Sigma}_x)$ such that $\boldsymbol{\Omega}_x = \boldsymbol{\Sigma}_x^{-1}$ is generated with a band structure so that the diagonal and super-diagonal elements are non-zero. This is generated using the 'huge' package for the default "band" setting. The final covariance has $\text{diag}(\boldsymbol{\Sigma}_x) = \mathbf{1}_p$ and a decreasing relation for variables that are further away from each other. The off-diagonal elements have a magnitude starting between 0.4 and 0.55 depending on p. We use $\boldsymbol{\beta}_1^*$ to define the relationship $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta}_1^* + \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim N(\mathbf{0}_p, \boldsymbol{I})$.

These settings give potiential situations that arise in practice.

We display the results for $p = 500$ and $p = 1000$ for settings G1, G2, and G3 when using the MCP penalty for $\gamma = 0.5$ in Table 3.1; results for $p = 100$ are similar and presented in Appendix B.3.1 To begin, we compare the results of the Ideal and Naive model to the results

Setting	p	Metric	Ideal	Naive	IRO	CLasso	GMUS
G1	500	L2	0.319	0.405	**0.373**	0.61	0.654
		TP	7.39	6.2	**5.68**	6.43	3.83
		FP	9.53	8.47	3.02	15	**0.23**
	1000	L2	0.338	0.423	**0.391**	0.62	0.676
		TP	7.15	6.32	5.5	**6.07**	3.61
		FP	11.62	11.09	3.49	18.14	**0.34**
G2	500	L2	0.363	0.68	**0.458**	1.227	1.89
		TP	10	10	**10**	**10**	9.93
		FP	3.43	6.78	1.02	19.34	**0.21**
	1000	L2	0.359	0.679	**0.426**	1.14	1.949
		TP	10	10	**10**	**10**	9.93
		FP	5.13	10.54	0.98	22.44	**0.2**
G3	500	L2	0.424	1.138	**0.916**	3.24	2.793
		TP	10	9.9	**9.79**	7.58	5.37
		FP	4.64	14.87	3.83	9.87	**0.97**
	1000	L2	0.445	1.229	**1.047**	3.239	2.836
		TP	10	9.81	**9.58**	7.27	4.94
		FP	8.07	23.65	4.5	12.1	**1.49**

Table 3.1: Simulation results for Gaussian linear regression under the three specified settings with noise-to-signal ratio $\gamma = 0.5$. Ideal, Naive, and IRO use the MCP penalty for regularization. Bold numbers illustrate the best method between the correction procedures for the setting metric.

of the our IRO-adjusted model. Focusing on variable selection, it is easy to see that the Ideal model outperforms the Naive model in every setting, as expected. When comparing the Naive and IRO-adjusted procedure, the biggest take-away is the difference in the number of false positives. In most every setting the Naive model finds about five times as many false positives. In all but one setting, the Naive model has a precision considerably less than 0.5. The IRO-adjusted procedure, however, never falls below 0.6. The corrected procedure does have more trouble identifying true positives, but the number of true positives never decreases by more than 10%. Finally, the quality of the estimated coefficients, as estimated by the norm difference, is always favorable to the IRO-adjusted procedure.

Now, comparing the IRO-adjusted model with the Corrected Lasso and GMUS gives more varied results. The Corrected Lasso seems to generally have a higher false positive rate and lower true positive rate than both IRO and GMUS. Interestingly, in setting G2, the Corrected Lasso performs worse than the Naive model, suggesting a lack of robustness to model assumptions. GMUS does not seem to have much issue at all with false positives,

having the lowest amount for every setting. However, the IRO-adjustment always has more true positives identified. This can be attributed to using the covariance structure information that GMUS does not take into account. The IRO-adjusted model and Corrected Lasso have comparable true positive identification. The IRO-adjusted model appears to have the highest quality coefficient estimates, as illustrated by the superior norm difference of the estimated coefficients in every setting.

3.4.2. Binomial Linear Regression

We now consider the case of using binomial linear regression to measure the the relationship of contaminated covariates with binary response. To this end we consider two settings for this scenario, easily described as the covariates being either independent or dependent. These settings are:

(B1) $\boldsymbol{X} \sim N(\mathbf{0}, \boldsymbol{\Sigma}_x)$ such that the covariance is diagonal where $\boldsymbol{\Sigma}_x = \boldsymbol{I}$. We use $\boldsymbol{\beta}_2^*$ to generate the relationship $y_i \sim Binom(f(\boldsymbol{x}_i^T\boldsymbol{\beta}))$.

(B2) $\boldsymbol{X} \sim N(\mathbf{0}, \boldsymbol{\Sigma}_x)$ such that dependencies exist between features, where $\boldsymbol{\Sigma}_x$ is generated as in setting G3. We use $\boldsymbol{\beta}_1^*$ to generate $y_i \sim Binom(f(\boldsymbol{x}_i^T\boldsymbol{\beta}))$.

In both instances f defines the inverse logit function.

In Table 3.2 we display the results of these two settings for $p = 100, 500, \text{and } 1000$ when signal-to-noise is specified such that $\gamma = 0.5$. Again, we begin by comparing the averaged results of the Ideal and Naive model with the IRO-adjusted procedure that is proposed. Again, it should be of no surprise that the naive implementation generally performs worse than the ideal. The effect is extreme for this case, but typically the presence of contaminated observations increases the number of false positives and decreases the number of true positives. The IRO-adjusted procedure is able to achieve nearly the same number of true positives as the naive method, while reducing the number of false positives by more than half in every case. Strangely, the IRO-adjusted procedure also has fewer false positives than the ideal model for every case. This can be attributed to the removal of spurious effects when

examining the model at each imputation iteration. Finally, the IRO-adjustment is able to either do as well or better than the naive model in terms of estimate quality, as measured by the norm difference.

Comparing the results of the IRO-adjustment and alternative correction procedures we note similar results as to the Gaussian case. Beginning with the CLasso, we first note a general poor performance for each of the settings, having a relatively low number of true positives and poor estimation quality. As the model is less powerful, less identification was done in total, as seen by the low number of false positives, too. It may be possible to achieve better performance with extensive tuning, but the defaults already search a well specified grid. A more likely reason for the performance can be attributed to the non-convex optimization that is performed to find the solution. While the Gaussian case has been specially designed for finding near-optimal solutions, the GLM case in general is much harder, and would make the elbow method used for tuning a challenge if only some tuning parameters found good solutions.

On the other hand, GMUS was able to find reasonable results for each setting. We see, again, that the IRO-adjustment performs better in identifying the true positives in the model. This effect is seen best when the covariates are correlated in Setting B2. However, GMUS does perform better in regards to the number of false positives in the model. The choice of method would be then given to the practitioner, as both methods perform better than the naive method. In terms of estimation quality, the IRO-adjustment performs better in every case. This is consistent with the results from the Gaussian setting, and establishes a general bias from the GMUS procedure. We note that, unlike CLasso and GMUS, the IRO-adjusted setting is easily established for other classification methods, like Linear Discriminant Analysis, which could be incorporated to improve the variable selection [59].

Setting	p	Metric	Ideal	Naive	IRO	CLasso	GMUS
B1	100	L2	0.754	1.204	**0.971**	3.939	2.576
		TP	10	10	**9.98**	2.71	9.53
		FP	4.16	5.9	2.45	**0.1**	0.51
	500	L2	0.898	1.537	**1.207**	4.012	2.739
		TP	10	9.97	**9.93**	2.39	9.18
		FP	8.75	11.35	4.31	**0.08**	0.55
	1000	L2	1.035	1.682	**1.413**	4.088	2.796
		TP	9.99	9.93	**9.86**	2.2	8.66
		FP	12.27	15.59	6.16	**0.08**	0.9
B2	100	L2	0.728	1.528	**1.268**	3.853	2.89
		TP	9.97	9.57	**9.45**	2.13	5.86
		FP	5	8.31	3.55	**0.14**	1.32
	500	L2	0.934	2.132	**2.185**	3.716	2.982
		TP	9.81	7.85	**7.33**	1.68	4.62
		FP	13.65	15.15	6.16	**0.11**	2.52
	1000	L2	1.206	2.455	**2.523**	3.446	3.008
		TP	9.41	6.77	**5.92**	1.52	4.28
		FP	18.82	16.96	6.71	**0.06**	2.99

Table 3.2: Simulation results for Binomial linear regression under the two specified settings with signal-to-noise ratio $\gamma = 0.5$, as described in-line. The Ideal, Naive, and IRO procedures use the MCP penalty for regularization.

3.5. Data Analysis

To show the efficacy of our proposed method, we illustrate it with an application to a microarray gene expression data set. For the sake of comparison, the data set and the pre-processing steps are the same as in [48]. This data set is comprised of $n = 144$ subjects' gene expressions for favorable histology Wilms tumors, of which 53 relapsed and 91 did not; the data set is accessible by the GEO website, dataset GSE10320 [24]. Each subject was measured with 10 or 11 probes, and hence replicates are available to estimate the measurement error variability. The Bioconductor package 'bgx' is able to incorporate the subject-level replicates in the preprocessing step to obtain the estimated measurement error covariance [21], which is assumed diagonal. To cut down the number of genes to be inspected, any gene that had estimated signal-to-noise value $\gamma > 0.5$ was discarded, the rational being that with too much noise, no discernible selection would be possible, regardless of correction.

We make use of the already processed output for the same dataset found in [35] and [5]. After removing genes with estimated signal-to-noise ratio larger than 0.5, there were a

remaining $p = 2074$ genes remaining. The goal is to determine any genes that have an impact on the tumor relapsing. We accomplish this with a binomial linear regression. Similar to the analysis done in Section 3.4, we compare the results of the Naive estimate, the IRO-adjusted estimate, the CLasso estimate, and the GMUS estimate. For the purposes of illustration, we use the Lasso procedure with 10-fold cross-validation for the Naive and IRO-adjustment. As the CLasso and GMUS procedures both lack a well-defined likelihood, we utilize the elbow-plot method as in the Binomial Regression simulation in Section 3.4.2.

We present the results of the analysis in Table 3.3, which shows the total number of genes selected for each procedure and the number of overlap between each procedure. Beginning with the results of the Naive analysis, the Lasso procedure selected a total of 35 genes in total. The IRO-adjusted Lasso procedure selected less than half that of the Naive implementation, finding a total of 14 genes when taking the median of each iteration's estimated coefficients. Additionally, all 14 variables found by the IRO-adjusted Lasso were also found by the Naive procedure. This is in line with the results found in the simulation conducted in the previous section, where the Naive and IRO implementations typically had similar amounts of true positives and a disparate amount of false positives.

Turning to the competing methods, the CLasso selected a total of 3 genes, all of which are shared with the Naive and IRO-adjustment. Given the relatively low number of true and false positives in the simulation, this seems to indicate similar behavior. Finally, the GMUS procedure selected a total of 7 genes. The behavior of the of GMUS was odd in the sense that there was only one gene in common with the CLasso and two genes in common with the Naive and IRO-adjustment. We believe that this is likely attributed to the dependencies between the genes, which had a relatively large negative impact on GMUS. The overall outcome seems to suggest the legitimacy of the simulation study, which illustrated the the IRO-adjustment to be a middle ground between true and false positives.

	Naive	IRO	CLasso	GMUS
Naive	35	14	3	2
IRO	14	14	3	2
CLasso	3	3	3	1
GMUS	2	2	1	7

Table 3.3: The total number of selected genes that overlapped between the Naive, IRO-adjusted, CLasso, and GMUS procedures. Note, the diagonal shows the total number of genes found by each procedure.

3.6. Conclusion

We have provided a new method of correction for high-dimensional generalized linear models with regularization. We employed the recent Imputation Regularization Optimization algorithm in a general correction context, and showed explicitly how to correct for the three most common data types: continuous, categorical, and count. Our proposed methodology improves on a simple naive implementation, which ignores the measurement error, and is competitive with current existing measurement error correction procedures in this context. The ease of use is the main draw of our proposal, and does not require special reformulation of existing methods. This is advantageous for practitioners who can use existing, well designed software, as well as providing an easy way to incorporate new state-of-the-art procedures.

Future work could be to establish imputation procedures for other settings, such as survival analysis or non-parametric regression. Many of these settings will not have a well-known density for the imputation step, and hence would require a way of estimating that density for sampling purposes. Such problems are well-known to Bayesian statisticians, and methods such as the Integrated Nested Laplace Approximation [46] could prove useful. An alternative direction could be towards establishing post-selection inference procedures on the estimated coefficients. This notion, termed selective inference, has become popular recently for making a valid inference with regularized models [52], and could prove insightful for rigorously providing a final set of estimated coefficients.

Chapter 4

Conclusions and Future Directions

In this dissertation we explored correction procedures with mismeasured observations for various statistical learning models in the high-dimensional setting. In particular we focused on the scenario where $n < p$, and, as traditional correction procedures fail in this setting, we appealed to the recently proposed Imputation Regularization Optimization algorithm. We showed how the IRO-algorithm could be used for mismeasured observations in the context of regularized Gaussian graphical models and generalized linear models. Our procedures can be used in conjunction with already established methods and procedures, whereas alternatively one would have to formulate model specific corrections and optimization routines. The imputation step naturally lends itself to a Bayesian flavor. Established tools from the MCMC literature can be utilized in conjunction with the optimization step without the need for a costly imputation process for all variables of interest. This in mind, we are able to forgo a computationally costly fully Bayes approach by partially using the same randomness mechanisms to improve upon frequentist approaches, which was our overarching goal. We showed that our relatively simple procedure is asymptotically consistent in the case of Gaussian graphical models, and we illustrated superior empirical performance compared to both correction-less procedures and alternative correction methods. We believe the simplicity of the IRO-correction is a valuable addition to the measurement error literature.

A first area of future interest would be establishing the IRO-procedure for more types of models, such a non-parametric regression. Such progress would be done by finding the imputation step, which may not exist in closed form. Hence, exploration of density estimation procedures that could be used for sampling would be a second area of future research. A third area of future work pertains to making an inference on the selected variables. Post

variable selection inference, termed selective inference, has recently became a popular topic in the literature due to complications in providing probabilistic bounds on selected variables [52]. It would be of interest in this regard to quantify the uncertainty about the variables selected from the IRO-procedure. This would not only require a rigorous understanding of the uncertainty at each imputation step, but also a rigorous method of combining each iterates' estimated values. This likely would lead to a better established selection criterion, too, instead of just using the median or trimmed average.

Appendix A
Supplementary Material for Chapter 2

A.1. Proofs

The proof for Theorem 1 in Section 2.3 is established here. Work done for the IRO algorithm laid the foundation for certain conditions to be met to establish consistency, see the appendix of [29]. We follow closely with their development, and prove the necessary conditions to establish consistency in our context of contaminated GGMs. These conditions include two main parts: (1) the consistency of the regularization step, specifically the BAGUS procedure in our context, and (2) some technical conditions regarding the log-likelihood $\pi(\boldsymbol{X}, \boldsymbol{W})$. To that end, Assumptions 1 and 2 below ensures the consistency of the BAGUS procedure, while Assumption 3 ensures the metric entropy of the log likelihood not to grow too fast. Discussion of Assumptions 1 and 2 can be found in [18], while Assumption 3 has been commonly used in the literature of high-dimensional statistics, see the Remark 1 in the appendix of [29].

Assumption 1. *$\lambda_{max}(\boldsymbol{\Omega}_x) \leq 1/k_1 \leq \infty$, where $\lambda_{max}(\boldsymbol{\Omega}_x)$ is the largest eigenvalue of $\boldsymbol{\Omega}_x$ and k_1 is a constant such that $k_1 > 0$.*

For Assumption 2 we need to define the following values. Let the column sparsity for $\boldsymbol{\Omega}_x$ be denoted $b = \max_{i=1,\ldots,d} \sum_{j=1}^{d} \mathbf{1}(\omega_{ij} \neq 0)$. For a $m \times q$ matrix A let $|||A|||_\infty = \max_{1\leq j\leq q} \sum_{i=1}^{d} |a_{ij}|$ be the maximum absolute row sum. Define $M_{\Sigma_x} = |||\boldsymbol{\Sigma}|||_\infty$ and $M_\Gamma = |||\boldsymbol{\Gamma}_{s,s}^{-1}|||_\infty$ where $\boldsymbol{\Gamma} = \boldsymbol{\Sigma}_x \otimes \boldsymbol{\Sigma}_x$ and $\boldsymbol{\Gamma}_{s,s}$ denotes the subset of $\boldsymbol{\Gamma}$ by indices $s = \{(i,j) : \boldsymbol{\Omega}_x \neq 0\}$. Let $a_1 > 0$ and $a_2 > 0$ be any predefined constant value. Also, let a_3 and k_2 be defined such that $\frac{\log(d)}{n} < a_3 < \frac{1}{4}$ and $\mathbb{E}(e^{tx^{(j)^2}}) \leq k_2$ for all $|t| \leq a_3$ and $j = 1, \ldots, d$. We define

$a_4 = a_1(2 + a_2 + a_3^{-1}k_2^2)$, $a_5 = (a_4 + 2M_{\Sigma_x}^2(a_1 + a_4)M_\Gamma + 6(a_1 + a_4)bM_\Gamma^2 M_\Sigma^3/M$. Finally, define constants $\epsilon_0 > 0$ and $\epsilon_1 > 0$, where ϵ_1 is small.

Assumption 2. *For the previously defined constants, the following three statements hold:*

1. *The hyperparameters v_0, v_1, η, and τ satisfy*

$$\begin{aligned}
&(a) \quad \frac{1}{nv_1} = a_1\sqrt{\frac{\log(d)}{n}}(1-\epsilon_0),\\
&(b) \quad \frac{1}{nv_0} > a_5\sqrt{\frac{\log(d)}{n}},\\
&(c) \quad \frac{v_1^2(1-\eta)}{v_0^2\eta} \leq d^{\epsilon_1},\\
&(d) \quad \tau \leq a_1\frac{n}{2}\sqrt{\frac{\log(d)}{n}}.
\end{aligned}$$

2. *For the bound $||\boldsymbol{\Omega}_x|| < B$, we have that B satisfies*

$$\frac{1}{k_1} + 2b(a_1 + a_4)M_\Gamma\sqrt{\frac{\log(d)}{n}} < B < \sqrt{2nv_0}.$$

3. *For $M = \max\{2b(a_1+a_4)M_\Gamma \max\{3M_\Sigma, 3M_\Gamma M_\Sigma^3, \frac{2}{k_1^2}\}, \frac{2a_1\epsilon_0}{k_1^2}\}$, we have $\sqrt{n} \geq M\sqrt{\log(p)}$.*

Assumption 3. *The parameter space of $\boldsymbol{\Omega}_x$, or an L_1-ball containing the space of $\boldsymbol{\Omega}_x$, grows at a rate of $\mathcal{O}(n^\alpha)$ for some $0 \leq \alpha \leq \frac{1}{2}$.*

Under these three assumptions, we show that the developed procedure to correct for measurement errors satisfy the general conditions for the consistency of the IRO estimate. We state each condition and prove it to hold with our procedure.

Condition 1. $\log \pi(\boldsymbol{X}, \boldsymbol{W}|\boldsymbol{\Omega}_x)$ *is a continuous function of $\boldsymbol{\Omega}_x$ for each $\boldsymbol{x}, \boldsymbol{w} \in \mathbb{R}^d$ and a measurable function of $(\boldsymbol{X}, \boldsymbol{W})$ for each $\boldsymbol{\Omega}_x$.*

Proof. We have the expansion

$$\log \pi(\boldsymbol{X}, \boldsymbol{W}|\boldsymbol{\Omega}_x) = \log \pi(\boldsymbol{X}|\boldsymbol{\Omega}_x) + \log \pi(\boldsymbol{W}|\boldsymbol{X}, \boldsymbol{\Omega}_u).$$

Hence, the log posterior is continuous for symmetric positive-definite $\boldsymbol{\Omega}_x$ since $\boldsymbol{x} \sim N(\mathbf{0}_d, \boldsymbol{\Omega}_x^{-1})$. The log posterior is also measurable for $(\boldsymbol{X}, \boldsymbol{W})$ due to properties of the Gaussian distribution. □

Condition 2. *Three conditions for the Glivenko-Cantelli theorem to hold.*

1. There exists a function $m_n(\boldsymbol{X}, \boldsymbol{W})$ such that $\sup_{\boldsymbol{\Omega}_x, \boldsymbol{X}} |\log \pi(\boldsymbol{X}, \boldsymbol{W}|\boldsymbol{\Omega}_x)| \leq m_n(\boldsymbol{X}, \boldsymbol{W})$.
2. There exists $m_n^*(\boldsymbol{W})$, such that:
   (a) $0 \leq \int m_n(\boldsymbol{X}, \boldsymbol{W}) \pi(\boldsymbol{X}|\boldsymbol{W}, \boldsymbol{\Omega}_x^{(t)}) d\boldsymbol{X} \leq m_n^*(\boldsymbol{W})$ for all $\boldsymbol{\Omega}_x^{(t)}$,
   (b) $\mathbb{E}[m_n^*(\boldsymbol{W})] < \infty$,
   (c) $\sup_{n \in \mathbb{Z}^+} \mathbb{E}[m_n^*(\boldsymbol{W}) \mathbb{I}(m_n^*(\boldsymbol{W}) \geq \xi)] \to 0$ as $\xi \to \infty$.

   Also, as $\xi \to \infty$,

$$\sup_{n \geq 1} \sup_{\boldsymbol{X}, \boldsymbol{\Omega}_x} \left| \int m_n(\boldsymbol{X}, \boldsymbol{W}) \mathbb{I}(m_n(\boldsymbol{X}, \boldsymbol{W}) > \xi) \pi(\boldsymbol{X}|\boldsymbol{W}, \boldsymbol{\Omega}_x) \right| \to 0.$$

3. Define $\mathcal{F}_n = \{\int \log \pi(\boldsymbol{X}, \boldsymbol{W}|\boldsymbol{\Omega}_x) \pi(\boldsymbol{X}|\boldsymbol{W}, \boldsymbol{\Omega}_x^{(t)}) d\boldsymbol{X}\}$ and $\mathcal{G}_{n,M} = \{q \mathbf{1}\{m_n^*(\boldsymbol{W}) \leq M\} | q \in \mathcal{F}_n\}$. Suppose that, for every ϵ and $M > 0$, the metric entropy $\log(N(\epsilon, \mathcal{G}_{n,M}, L_1(\mathbb{P}_n))) = \mathcal{O}(n)$, where $\mathbb{P}_n$ is the emprical measure of $\boldsymbol{W}$ and $N(\epsilon, \mathcal{G}_{n,M}, L_1(\mathbb{P}_n))$ is the covering number with respect to the $L_1(\mathbb{P})$-norm.

Proof. We begin with part (1). Note that

$$\begin{aligned}
\log \pi(\boldsymbol{X}, \boldsymbol{W}|\boldsymbol{\Omega}_x) &= \sum_{i=1}^{n} \left[\log \pi(\boldsymbol{w}_i|\boldsymbol{x}_i, \boldsymbol{\Omega}_u) + \log \pi(\boldsymbol{x}_i|\boldsymbol{\Omega}_x)\right] \\
&= -\frac{1}{2} \sum_{i=1}^{n} \left[(\boldsymbol{w}_i - \boldsymbol{x}_i)^T \boldsymbol{\Omega}_u (\boldsymbol{w}_i - \boldsymbol{x}_i) + \boldsymbol{x}_i^T \boldsymbol{\Omega}_x \boldsymbol{x}_i\right] + \frac{1}{2} \log \det(\boldsymbol{\Omega}_x) + C,
\end{aligned}$$

where C contains constants not related to $(\boldsymbol{X}, \boldsymbol{W}, \boldsymbol{\Omega}_x)$. Hence,

$$\begin{aligned} |\log \pi(\boldsymbol{X}, \boldsymbol{W}|\boldsymbol{\Omega}_x)| &\leq \frac{1}{2}\sum_{i=1}^{n}\left[(\boldsymbol{w}_i - \boldsymbol{x}_i)^T\boldsymbol{\Omega}_u(\boldsymbol{w}_i - \boldsymbol{x}_i) + K_1\boldsymbol{x}_i^T\boldsymbol{x}_i\right] + K_2 \\ &= \sum_{i=1}^{n} m(\boldsymbol{x}_i, \boldsymbol{w}_i) = m(\boldsymbol{X}, \boldsymbol{W}), \end{aligned}$$

where K_1 and K_2 are constants depending on upper bound B.

To prove part (2) note

$$\begin{aligned} \tilde{m}(\boldsymbol{W}, \boldsymbol{\Omega}_x^{(t)}) &= \int m(\boldsymbol{X}, \boldsymbol{W})\pi(\boldsymbol{X}|\boldsymbol{W}, \boldsymbol{\Omega}_x^{(t)})d\boldsymbol{X} \\ &= \int \sum_{i=1}^{n} m(\boldsymbol{x}_i, \boldsymbol{w}_i)\left[\prod_{j=1}^{n}\pi(\boldsymbol{x}_i|\boldsymbol{w}_i, \boldsymbol{\Omega}_x^{(t)})\right] d\boldsymbol{x}_1, \ldots, d\boldsymbol{x}_n \\ &= \sum_{i=1}^{n}\int m(\boldsymbol{x}_i, \boldsymbol{w}_i)\pi(\boldsymbol{x}_i|\boldsymbol{w}_i, \boldsymbol{\Omega}_x^{(t)})d\boldsymbol{x}_i, \end{aligned}$$

where the last equality follows from conditional independence of each $\boldsymbol{x}_i$. Let $\boldsymbol{\Lambda}^{(t)} = (\boldsymbol{\Omega}_x^{(t)} + \boldsymbol{\Omega}_u)^{-1}$, and notice this the sum of expectations of $m(\boldsymbol{x}_i, \boldsymbol{w}_i)$ with respect to Gaussian random variables following $N(\boldsymbol{\Lambda}^{(t)}\boldsymbol{\Omega}_u\boldsymbol{w}_i, \boldsymbol{\Lambda}^{(t)})$ for each $i = 1, \ldots, n$. Now,

$$\mathbb{E}_{\boldsymbol{x}_i|\boldsymbol{w}_i, \boldsymbol{\Omega}_x^{(t)}}\left[m(\boldsymbol{x}_i, \boldsymbol{w}_i)\right] = \frac{1}{2}\boldsymbol{w}_i^T\boldsymbol{\Omega}_u\boldsymbol{w}_i + \frac{1}{2}\text{tr}((\boldsymbol{\Omega}_u + K_1\boldsymbol{I}_d)\boldsymbol{\Lambda}^{(t)}) - \underbrace{\boldsymbol{w}_i^T\boldsymbol{\Omega}_u\boldsymbol{\Lambda}^{(t)}\boldsymbol{\Omega}_u\boldsymbol{w}_i}_{\geq 0},$$

which, since $||\boldsymbol{\Lambda}^{(t)}|| \leq K_3$, implies

$$\tilde{m}(\boldsymbol{W}, \boldsymbol{\Omega}_x^{(t)} \leq \frac{1}{2}\sum_{i=1}^{n}\boldsymbol{w}_i^T\boldsymbol{\Omega}_u\boldsymbol{w}_i + K_3 = m^*(\boldsymbol{W}).$$

Marginally $\boldsymbol{w}_i \sim N(\boldsymbol{0}_d, \boldsymbol{\Sigma}_x, \boldsymbol{\Sigma}_u)$, and hence $m^*(\boldsymbol{W})$ is the sum of scaled chi-square distributions. Conditions (b) and (c) easily follow from the properties of the chi-square distribution.

To prove part (3), we make use of Remark 1 found in the Appendix of [29]. Since all

elements in $\cup_{n\geq 1}\mathcal{F}_n$ are uniformly Lipschitz, see [23], the metric entropy can be measured on the basis of the parameter space of $\boldsymbol{\Omega}_x$. The functions in $\mathcal{G}_{n,M}$ are bounded and the parameter space can be contained by the L_1 ball due to the continuity of $\log \pi(\boldsymbol{X}, \boldsymbol{W}|\boldsymbol{\Omega}_x)$. By Assumption 3, then $\log(N(\epsilon, \mathcal{G}_{n,M}, L_1(\mathbb{P}_n))) = \mathcal{O}(n^{2\alpha}\log(d))$. □

Condition 3. *Define* $Z_{t,i} = \log \pi(\boldsymbol{x}_i, \boldsymbol{w}_i|\boldsymbol{\Omega}_x) - \int \log \pi(\boldsymbol{x}_i, \boldsymbol{w}_i|\boldsymbol{\Omega}_x)\pi(\boldsymbol{X}|\boldsymbol{w}_i, \boldsymbol{\Omega}_x^{(t)})$. *$Z_{t,i}$ are subexponential random variables.*

Proof. First, we note that

$$\begin{aligned}\log \pi(\boldsymbol{x}_i, \boldsymbol{w}_i|\boldsymbol{\Omega}_x) &= -\frac{1}{2}(\boldsymbol{w}_i - \boldsymbol{x}_i)^T\boldsymbol{\Omega}_u(\boldsymbol{w}_i - \boldsymbol{x}_i) - \frac{1}{2}\boldsymbol{x}_i\boldsymbol{\Omega}_x\boldsymbol{x}_i \\ &= -\frac{1}{2}\boldsymbol{x}_i^T(\boldsymbol{\Omega}_x + \boldsymbol{\Omega}_u)\boldsymbol{x}_i + \boldsymbol{x}_i^T\boldsymbol{\Omega}_u\boldsymbol{w}_i + C_1,\end{aligned}$$

where C_1 is a constant free of $\boldsymbol{X}$. Also note $\log \pi(\boldsymbol{w}_i, \boldsymbol{X}|\boldsymbol{\Omega}_x) = \log \pi(\boldsymbol{w}_i, \boldsymbol{x}_i|\boldsymbol{\Omega}_x) + \log \pi(\boldsymbol{X}_{-i}|\boldsymbol{\Omega}_x)$. The integral can then be shown to be

$$\begin{aligned}&\int \log \pi(\boldsymbol{x}_i, \boldsymbol{w}_i|\boldsymbol{\Omega}_x)\pi(\boldsymbol{X}|\boldsymbol{w}_i, \boldsymbol{\Omega}_x^{(t)}) \\ &= \int \left[\log \pi(\boldsymbol{w}_i, \boldsymbol{x}_i|\boldsymbol{\Omega}_x) + \log \pi(\boldsymbol{X}_{-i}|\boldsymbol{\Omega}_x)\right]\pi(\boldsymbol{x}_i|\boldsymbol{w}_i, \boldsymbol{\Omega}_x^{(t)})\pi(\boldsymbol{X}_{-i}|\boldsymbol{\Omega}_x^{(t)})d\boldsymbol{x}_i d\boldsymbol{X}_{-i} \\ &= \underbrace{\int \log \pi(\boldsymbol{w}_i, \boldsymbol{x}_i|\boldsymbol{\Omega}_x)\pi(\boldsymbol{x}_i|\boldsymbol{w}_i, \boldsymbol{\Omega}_x^{(t)})d\boldsymbol{x}_i}_{=A}\underbrace{\int \pi(\boldsymbol{X}_{-i}|\boldsymbol{\Omega}_x^{(t)})d\boldsymbol{X}_{-i}}_{=1} \\ &\quad + \underbrace{\int \log \pi(\boldsymbol{X}_{-i}|\boldsymbol{\Omega}_x\pi(\boldsymbol{X}_{-i}|\boldsymbol{\Omega}_x^{(t)})d\boldsymbol{X}_{-i}}_{=C_2}\underbrace{\int \pi(\boldsymbol{x}_i|\boldsymbol{w}_i, \boldsymbol{\Omega}_x^{(t)})d\boldsymbol{x}_i}_{=1}.\end{aligned}$$

The value of A is the expectation of $\log \pi(\boldsymbol{w}_i, \boldsymbol{x}_i|\boldsymbol{\Omega}_x)$ with respect to the full conditional of X at iteration t, $\boldsymbol{x}_i|\boldsymbol{w}_i, \boldsymbol{\Omega}_x^{(t)} \sim N_d(\boldsymbol{\Lambda}^{-1,(t)}\boldsymbol{\Omega_u}\boldsymbol{w}_i, \boldsymbol{\Lambda}^{-1,(t)})$ where $\boldsymbol{\Lambda}^{(t)} = (\boldsymbol{\Omega}_x^{(t)} + \boldsymbol{\Omega_u})$. This expectation is composed of two parts,

$$\mathbb{E}_{\boldsymbol{x}_i|\boldsymbol{w}_i,\boldsymbol{\Omega}_x}(\boldsymbol{x}_i(\boldsymbol{\Omega}_x + \boldsymbol{\Omega}_u)\boldsymbol{x}_i) = \text{tr}((\boldsymbol{\Omega}_x + \boldsymbol{\Omega}_u)\Lambda^{-1,(t)}) + \boldsymbol{w}_i\boldsymbol{\Omega}_u\boldsymbol{\Lambda}^{(t)}(\boldsymbol{\Omega}_x + \boldsymbol{\Omega}_u)\boldsymbol{\Lambda}^{(t)}\boldsymbol{\Omega}_u\boldsymbol{w}_i$$

and

$$\mathbb{E}_{\boldsymbol{x}_i|\boldsymbol{w}_i,\boldsymbol{\Omega}_x}(\boldsymbol{x}_i^T\boldsymbol{\Omega}_u\boldsymbol{w}_i) = \boldsymbol{w}_i^T\boldsymbol{\Omega}_u\boldsymbol{\Lambda}^{(t)}\boldsymbol{\Omega}_u\boldsymbol{w}_i.$$

Hence, $Z_{t,i}$ is

$$-\frac{1}{2}\boldsymbol{x}_i^T(\boldsymbol{\Omega}_x+\boldsymbol{\Omega}_u)\boldsymbol{x}_i + \boldsymbol{x}_i^T\boldsymbol{\Omega}_u\boldsymbol{w}_i - \frac{1}{2}\boldsymbol{w}_i^T\boldsymbol{\Omega}_u\boldsymbol{\Lambda}^{(t)}(\boldsymbol{\Omega}_x+\boldsymbol{\Omega}_u)\boldsymbol{\Lambda}^{(t)}\boldsymbol{\Omega}_u\boldsymbol{w}_i + \boldsymbol{w}_i^T\boldsymbol{\Omega}_u\boldsymbol{\Lambda}^{(t)}\boldsymbol{\Omega}_u\boldsymbol{w}_i + C,$$

where $C = C_1 + C_2$ is free of $\boldsymbol{x}_i$ and $\boldsymbol{w}_i$, which is the sum of scaled chi-squared distributions and thus is subexponential. □

Condition 4. *For $t = 1, \ldots, T$, $Q(\boldsymbol{\Omega}_x|\boldsymbol{\Omega}_x^{(t)})$ has a unique maximum at $\tilde{\boldsymbol{\Omega}}_x^{(t)}$; for any $\epsilon > 0$, $\sup_{\boldsymbol{\Omega}_x\setminus B_t(\epsilon)} Q(\boldsymbol{\Omega}_x|\boldsymbol{\Omega}_x^{(t)})$ exists, where $B_t(\epsilon) = \{\boldsymbol{\Omega}_x : |\boldsymbol{\Omega}_x - \tilde{\boldsymbol{\Omega}}_x^{(t)}| < \epsilon\}$.*

Proof. As noted in [29], this is satisfied if $\boldsymbol{\Omega}_x$ is restricted to a compact set. So, since BAGUS is strictly convex when restricted by the condition that $||\boldsymbol{\Omega}_x|| \leq B$, then the condition is satisfied. □

Condition 5. *The penalty function is non-negative, ensures the existence of $\boldsymbol{\Omega}_x^{(t+1)}$ for $t = 2, \ldots, T$, and converges to 0 uniformly as $n \to \infty$.*

Proof. BAGUS is a non-negative penalty that exists for any $\boldsymbol{X}$, and, due to the adaptive nature of the penalty, converges to 0 as $n \to \infty$. To see the penalty converges to 0, note Assumption 2.1a implies

$$v_1 = \frac{1}{a_1(1-\epsilon_0)\sqrt{n\log(d)}} \to 0$$

as $n \to \infty$, which, with a similar argument for v_0, results in the penalty being 0 as $n \to \infty$. □

As each of the previous conditions are true for our proposed model, by results of [29] the consistency claim holds.

A.2. Computing BAGUS with the EM-Algorithm

Here we review the optimization of the uncontaminated objective distribution. The direct optimization of L^{UC} in (2.5) is not easy due to the sum inside the logarithm. [18] use the EM-algorithm to get around this issue by introducing the latent factors r_{ij} from section 2.3.3. This allows an E-step similar to the spike-and-slab Lasso and an M-step similar to the Graphical Lasso. In this section, if not specified, $\boldsymbol{\Sigma}$ and $\boldsymbol{\Omega}$ refer to $\boldsymbol{x}$'s covariance and precision matrix, respectively.

The optimization seeks to find the MAP of the posterior proportional to

$$|\boldsymbol{\Omega}_x|^{\frac{1}{2}} \exp\left\{-\frac{1}{2}\boldsymbol{X}^T\boldsymbol{\Omega}_x\boldsymbol{X}\right\} \prod_{i<j} \pi(\omega_{ij}|r_{ij})\pi(r_{ij}|\eta) \prod_i \pi(\omega_{ii}|\tau) I(\boldsymbol{\Omega}_x \succ 0) I(||\boldsymbol{\Omega}_x|| \leq B),$$

where the latent indicator r_{ij}, as defined in Section 2.3.3, is incorporated into the off-diagonal elements in the prior specification. The E-step takes the conditional expectation of r_{ij} in the proportional posterior. Each r_{ij} is conditionally Bernoulli with probability

$$p_{ij} = \frac{v_1}{v_0}\frac{1-\eta}{\eta} \exp\left\{|\omega_{ij}^{(t)}|\left(\frac{1}{v_1} - \frac{1}{v_0}\right)\right\},$$

allowing for easy calculation of the conditional expectation. Then, the desired Q function to maximize in the M-step is given by

$$Q(\boldsymbol{\Omega}_x|\boldsymbol{\Omega}_x^{(t)}) = \mathbb{E}_{\boldsymbol{R}|\boldsymbol{\Omega}_x^{(t)}} \log \pi(\boldsymbol{\Omega}_x, \boldsymbol{X}|\boldsymbol{W}, \boldsymbol{\Sigma}_u),$$

where the expectation is taken element wise for $\boldsymbol{R}$ by assumed independence of inclusion.

The M-step optimizes each column of Q seperately with coordinate descent. The last column's update is now explained, with the other columns following in the same pattern.

Partition the covariance matrix as

$$\boldsymbol{\Sigma}_x = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\sigma}_{12} \\ \boldsymbol{\sigma}_{12}^T & \boldsymbol{\sigma}_{22} \end{bmatrix},$$

and let similar partitions be available for $\boldsymbol{\Omega}_x, \boldsymbol{P}, \boldsymbol{R}$, and $\boldsymbol{S}$. Also note that

$$\begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\sigma}_{12} \\ \cdot & \boldsymbol{\sigma}_{22} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Omega}_x^{-1} + c^{-1}\boldsymbol{\Omega}_{11}^{-1}\boldsymbol{\omega}_{12}\boldsymbol{\omega}_{12}^T\boldsymbol{\Omega}_{11}^{-1} & -c^{-1}\boldsymbol{\Omega}_{11}^{-1}\boldsymbol{\omega}_{12} \\ \cdot & c^{-1} \end{bmatrix},$$

where $c = \omega_{22} - \boldsymbol{\omega}_{12}^T\boldsymbol{\Omega}_{11}^{-1}\boldsymbol{\omega}_{12}$. The update for the last column of $\boldsymbol{\Sigma}_x$ is the solution from setting subgradient of Q with respect to $[\boldsymbol{\sigma}_{12}\ \boldsymbol{\sigma}_{22}]^T$ to 0. The update for σ_{22} can is easily attained from the setting the subgradient of ω_{22} to 0,

$$\omega_{22} = \frac{1}{\sigma_{22}} + \boldsymbol{\omega}_{12}^T\boldsymbol{\Omega}_{11}^{-1}\boldsymbol{\omega}_{12}.$$

We note that each column update requires the matrix $\boldsymbol{\Omega}_{11}^{-1}$. This can be computed as $\boldsymbol{\Sigma}_{11} - \boldsymbol{\sigma}_{12}\boldsymbol{\sigma}_{12}^T/\sigma_{22}$.

Appendix B
Supplementary Material for Chapter 3

B.1. Derivations

In this section we provide the derivations used to obtain the resulting distributions for the respective imputation steps.

B.1.1. Covariate Only Imputation Distribution Derivation

To impute missing true data $\boldsymbol{x}_i$, we wish to find the full conditional distribution of $\boldsymbol{x}_i|\bar{\boldsymbol{w}}_i, \boldsymbol{\Omega}_x, \boldsymbol{\Omega}_u$ for each $i = 1, \ldots, n$. Standard calculations find

$$\begin{aligned}
\pi(\boldsymbol{x}_i|\boldsymbol{W}_i, \boldsymbol{\Omega}_x, \boldsymbol{\Omega}_u) &\propto \exp\left\{-\frac{1}{2}\boldsymbol{x}_i^T\boldsymbol{\Omega}_x\boldsymbol{x}_i\right\}\prod_{j=1}^{r_i}\exp\left\{-\frac{1}{2}(\boldsymbol{w}_{ij}-\boldsymbol{x}_i)^T\boldsymbol{\Omega}_u(\boldsymbol{w}_{ij}-\boldsymbol{x}_i)\right\} \\
&\propto \exp\left\{-\frac{1}{2}\left[\boldsymbol{x}_i^T\left(r_i\boldsymbol{\Omega}_u+\boldsymbol{\Omega}_x\right)\boldsymbol{x}_i - 2r_i\boldsymbol{x}_i^T\boldsymbol{\Omega}_u\bar{\boldsymbol{w}}_i\right]\right\} \\
&\propto \exp\left\{-\frac{1}{2}(\boldsymbol{x}_i - r_i\boldsymbol{\Lambda}\boldsymbol{\Omega}_u\bar{\boldsymbol{w}}_i)^T\boldsymbol{\Lambda}^{-1}(\boldsymbol{x}_i - r_i\boldsymbol{\Lambda}\boldsymbol{\Omega}_u\bar{\boldsymbol{w}}_i)\right\},
\end{aligned}$$

where $\boldsymbol{\Lambda} = (\boldsymbol{\Omega}_x + r_i\boldsymbol{\Omega}_u)^{-1}$. This result is a kernel of a multivariate Gaussian distribution,

$$\pi(\boldsymbol{x}_i|\boldsymbol{W}_i, \boldsymbol{\Omega}_x, \boldsymbol{\Omega}_u) \sim N(r_i\boldsymbol{\Lambda}\boldsymbol{\Omega}_u\bar{\boldsymbol{w}}_i, \boldsymbol{\Lambda}), \tag{B.1}$$

We briefly note that so long as $r_i = r_j$, then observations i and j share the same covariance component. When generating large multivariate Gaussian distributions, most of the computation comes from the matrix inversions. By grouping observations with the same number of replicates, time can be saved by only needing compute the full-conditional distributions'

covariance once.

B.1.2. Gaussian Linear Regression Imputation Distribution Derivation

The distribution to impute missing data from a linear model with a Gaussian link, where $\boldsymbol{\Lambda}$ is as in (B.1), is

$$\begin{aligned}
&\pi(\boldsymbol{x}_i|\boldsymbol{W}_i, y_i, \boldsymbol{\Omega}_x, \boldsymbol{\Omega}_u, \boldsymbol{\beta}, \sigma^2) \propto \pi(y_i|\boldsymbol{w}_i, \boldsymbol{\beta}, \sigma^2)\pi(\boldsymbol{x}_i|\boldsymbol{W}_i, \boldsymbol{\Omega}_x, \boldsymbol{\Omega}_u) \\
&\propto \exp\left\{-\frac{1}{2\sigma^2}(y_i - \boldsymbol{x}_i^T\boldsymbol{\beta})^2\right\} \exp\left\{-\frac{1}{2}(\boldsymbol{x}_i - r_i\boldsymbol{\Lambda}\boldsymbol{\Omega}_u\bar{\boldsymbol{w}}_i)^T\boldsymbol{\Lambda}^{-1}(\boldsymbol{x}_i - r_i\boldsymbol{\Lambda}\boldsymbol{\Omega}_u\bar{\boldsymbol{w}}_i)\right\} \\
&\propto \exp\left\{-\frac{1}{2}\left[\boldsymbol{x}_i^T(r_i\boldsymbol{\Omega}_u + \boldsymbol{\Omega}_x + \frac{1}{\sigma^2}\boldsymbol{\beta}\boldsymbol{\beta}^T)\boldsymbol{x}_i - 2\boldsymbol{x}_i^T\left(\frac{y_i}{\sigma^2}\boldsymbol{\beta} + r_i\boldsymbol{\Lambda}\boldsymbol{\Omega}_u\bar{\boldsymbol{w}}_i\right)\right]\right\} \\
&= \exp\left\{-\frac{1}{2}\left(\boldsymbol{x}_i - \boldsymbol{\Lambda}_G\left(r_i\boldsymbol{\Omega}_u\bar{\boldsymbol{w}}_i + \frac{y_i}{\sigma^2}\boldsymbol{\beta}\right)\right)^T\boldsymbol{\Lambda}_G^{-1}\left(\boldsymbol{x}_i - \boldsymbol{\Lambda}_G\left(r_i\boldsymbol{\Omega}_u\bar{\boldsymbol{w}}_i + \frac{y_i}{\sigma^2}\boldsymbol{\beta}\right)\right)\right\},
\end{aligned}$$

where $\boldsymbol{\Lambda}_G = \left(r_i\boldsymbol{\Omega}_u + \boldsymbol{\Omega}_x + \sigma^{-2}\boldsymbol{\beta}\boldsymbol{\beta}^T\right)^{-1}$. This result is, again, the kernel of a multivariate Gaussian distribution,

$$\pi(\boldsymbol{x}_i|\boldsymbol{W}_i, \boldsymbol{\Omega}_x, \boldsymbol{\Omega}_u) \sim N\left(\boldsymbol{\Lambda}_G\left(r_i\boldsymbol{\Omega}_u\bar{\boldsymbol{w}}_i + \frac{y_i}{\sigma^2}\boldsymbol{\beta}\right), \boldsymbol{\Lambda}_G\right). \tag{B.2}$$

Again, by grouping observations with the same number of replicates, time can be saved by computing each matrix inverse for each unique number of replicates.

B.1.3. Binomial Linear Regression Imputation Distribution Derivation

To impute $\boldsymbol{x}_i$ when using the logit function we appeal to [41], which, as explained in Section 3.3.2, uses Pólya Gamma random variables to augment the data generating process.

For the most recently generated z_i, from [41] we note that

$$\pi(y_i|z_i, \boldsymbol{x}_i, \boldsymbol{\beta}) = \frac{\exp\{\boldsymbol{x}_i^T\boldsymbol{\beta}\}^{y_i}}{1+\exp\{\boldsymbol{x}_i^T\boldsymbol{\beta}\}} \propto \exp\left\{-\frac{z_i}{2}\left(\frac{\kappa_i}{z_i} - \boldsymbol{x}_i^T\boldsymbol{\beta}\right)^2\right\}, \tag{B.3}$$

where $\kappa_i = y_i - 1/2$. Hence, with $\boldsymbol{\Lambda}$ as in (B.1), we have

$$\begin{aligned}
&\pi(\boldsymbol{x}_i|\boldsymbol{W}_i, \boldsymbol{\Omega}_x, \boldsymbol{\Omega}_u, y_i, \boldsymbol{\beta}, z_i) \\
&\propto \exp\left\{-\frac{1}{2}(\boldsymbol{x}_i - r_i\boldsymbol{\Lambda}\boldsymbol{\Omega}_u\bar{\boldsymbol{w}}_i)^T\boldsymbol{\Lambda}^{-1}(\boldsymbol{x}_i - r_i\boldsymbol{\Lambda}\boldsymbol{\Omega}_u\bar{\boldsymbol{w}}_i) - \frac{z_i}{2}\left(\frac{\kappa_i}{z_i} - \boldsymbol{x}_i^T\boldsymbol{\beta}\right)^2\right\} \\
&\propto \exp\left\{-\frac{1}{2}\left[\boldsymbol{x}_i^T(z_i\boldsymbol{\beta}\boldsymbol{\beta}^T + r_i\boldsymbol{\Omega}_u + \boldsymbol{\Omega}_x)\boldsymbol{x}_i - 2\boldsymbol{x}_i^T(\kappa_i\boldsymbol{\beta} + r_i\boldsymbol{\Omega}_u\bar{\boldsymbol{w}}_i)\right]\right\} \\
&\propto \exp\left\{-\frac{1}{2}(\boldsymbol{x}_i - \boldsymbol{\Lambda}_B(\kappa_i\beta + r_i\boldsymbol{\Omega}_u\bar{\boldsymbol{w}}_i))^T\boldsymbol{\Lambda}_B(\boldsymbol{x}_i - \boldsymbol{\Lambda}_B(\kappa_i\beta + r_i\boldsymbol{\Omega}_u\bar{\boldsymbol{w}}_i))\right\},
\end{aligned}$$

where $\boldsymbol{\Lambda}_B = (z_i\boldsymbol{\beta}\boldsymbol{\beta}^T + r_i\boldsymbol{\Omega}_u + \boldsymbol{\Omega}_x)^{-1}$. As expected from the results of [41], we have a Gaussian kernal, where

$$\pi(\boldsymbol{x}_i|\boldsymbol{W}_i, \boldsymbol{\Omega}_x, \boldsymbol{\Omega}_u, y_i, \boldsymbol{\beta}, z_i) \sim N(\boldsymbol{\Lambda}_B(\kappa_i\beta + r_i\boldsymbol{\Omega}_u\bar{\boldsymbol{w}}_i), \boldsymbol{\Lambda}_B). \tag{B.4}$$

The derivation for the full conditional distribution of z_i is exactly the same as in [41].

B.2. Estimating the Measurement Error Covariance with Replicates

Here, we address an estimate of the measurement error's precision matrix, $\boldsymbol{\Omega}_u$, which is necessary for the imputation step. In some instances, it may be realistic to know the amount of variability in the measurement process; for instance, a machine taking measurements where the output falls within some perturbation of the truth. However, in many contexts the variability of the contamination process will not be known, and hence need to be estimated, typically with replicates. Estimating $\boldsymbol{\Sigma}_u$ is difficult due to not directly observing the amount of contamination on each observation. However, if one assumes the amount of contamination is independent between each variable, then a procedure exists to get an empirical estimate of the diagonal of $\boldsymbol{\Sigma}_x$ and, hence, $\boldsymbol{\Omega}_u$.

An estimate of $\boldsymbol{\Sigma}_u$ is also necessary for the imputation. When data is observed with replicates for each observation, then this covariance matrix is an estimable variable under independence assumptions. Consider the measurement error distribution as described in Section 3.2, where for each observation's replicates, $\boldsymbol{u}_{ij} \sim N(\mathbf{0}_p, \boldsymbol{\Sigma}_u)$. Estimating $\boldsymbol{\Sigma}_u$ is not trivial because $n < p$ and $\boldsymbol{u}_{ij}$ is not directly observed. Note for replicate j and k of observation i that

$$\boldsymbol{d}_{ijk} = \boldsymbol{w}_{ij} - \boldsymbol{w}_{ik} = \boldsymbol{x}_i - \boldsymbol{u}_{ij} - \boldsymbol{x}_i - \boldsymbol{u}_{ik} = \boldsymbol{u}_{ij} - \boldsymbol{u}_{ik}. \tag{B.5}$$

Assuming the amount of contamination is independent for each covariate, then, for covariate m,

$$Var(d_{ijk}^{(m)}) = Var(u_{ij}^{(m)}) + Var(u_{ik}^{(m)}) = 2[\boldsymbol{\Sigma}_u]_{m,m}.$$

Hence, if one were willing to assume the same distribution governing the contamination of each observation, the differences from all $i = 1, \ldots, n$ where $j < k$ could be used and

averaged for all $r_i(r_i - 1)$ possible differences per observation,

$$[\hat{\mathbf{\Omega}}_u]_{m,m} = \frac{1}{\sqrt{2}} \frac{1}{n} \sum_{i=1}^{n} \frac{1}{r_i(r_i - 1)} \sum_{j<k} d_{ijk}^{(m)}. \tag{B.6}$$

If heterogenious measurement error is believed to exist between observations, then it can easily be incorporated into the imputation step by using observation specific $\mathbf{\Omega}_{u,i}$. Here, the averaged covariance diagonal element would only be between the pair-wise replicates for the obervation.

B.3. Other Gaussian Simulation Results

B.3.1. Complete MCP Results

In Tables B.1 and B.2 we display the complete results for the results found in Chapter 3.4.1. Table B.1 displays the results for $p = 100$, and all results are displayed for $\gamma = 1$ in Table B.2. All results are similar to the discussion presented in Chapter 3.4.1.

Setting	p	Metric	Ideal	Naive	IRO	CLasso	GMUS
G1	100	L2	0.266	0.349	0.319	0.611	0.616
		TP	8.45	7.61	7.09	7.57	4.34
		FP	5.44	4.69	2.75	9.85	0.09
	500	L2	0.319	0.405	0.373	0.61	0.654
		TP	7.39	6.2	5.68	6.43	3.83
		FP	9.53	8.47	3.02	15	0.23
	1000	L2	0.338	0.423	0.391	0.62	0.676
		TP	7.15	6.32	5.5	6.07	3.61
		FP	11.62	11.09	3.49	18.14	0.34
G2	100	L2	0.33	0.602	0.422	1.293	1.6
		TP	10	10	10	10	10
		FP	1.33	3.15	0.75	11.06	0.16
	500	L2	0.363	0.68	0.458	1.227	1.89
		TP	10	10	10	10	9.93
		FP	3.43	6.78	1.02	19.34	0.21
	1000	L2	0.359	0.679	0.426	1.14	1.949
		TP	10	10	10	10	9.93
		FP	5.13	10.54	0.98	22.44	0.2
G3	100	L2	0.407	1.016	0.768	3.792	2.665
		TP	10	9.97	9.96	8.07	6.42
		FP	2.03	5.51	2.02	4.36	0.61
	500	L2	0.424	1.138	0.916	3.24	2.793
		TP	10	9.9	9.79	7.58	5.37
		FP	4.64	14.87	3.83	9.87	0.97
	1000	L2	0.445	1.229	1.047	3.239	2.836
		TP	10	9.81	9.58	7.27	4.94
		FP	8.07	23.65	4.5	12.1	1.49

Table B.1: Simulation results for Gaussian linear regression under the three specified settings with noise-to-signal ratio $\gamma = 0.5$. Ideal, Naive, and IRO use the MCP penalty for regularization.

Setting	p	Metric	Ideal	Naive	IRO	CLasso	GMUS
G1	100	L2	0.265	0.464	0.385	1.576	0.709
		TP	8.38	6.73	6	3.1	3.94
		FP	5.79	4.95	2.17	1.37	0.12
	500	L2	0.322	0.503	0.436	1.45	0.735
		TP	7.64	5.77	4.41	2.72	3.62
		FP	9.37	7.69	1.02	1.8	0.22
	1000	L2	0.342	0.524	0.456	1.408	0.762
		TP	6.95	5.45	4.23	2.41	3.43
		FP	11.22	11.06	1.09	2.38	0.4
G2	100	L2	0.351	0.913	0.535	3.943	1.937
		TP	10	10	10	7.26	9.88
		FP	1.82	4.47	0.35	0.77	0.24
	500	L2	0.371	1.026	0.606	3.367	2.195
		TP	10	10	9.99	7.28	9.62
		FP	3.9	9.56	0.56	1.13	0.15
	1000	L2	0.366	1.055	0.637	3.372	2.257
		TP	10	10	9.99	7.07	9.62
		FP	4.3	13.49	0.4	1.23	0.34
G3	100	L2	0.39	1.527	1.166	6.744	2.775
		TP	10	9.85	9.65	2.83	5.87
		FP	1.66	8.69	1.89	0.05	0.74
	500	L2	0.416	1.845	1.74	5.891	2.853
		TP	10	9.06	8.18	2.88	5.24
		FP	4.87	19.58	3.12	0.14	1.76
	1000	L2	0.42	2.1	2.105	5.461	2.885
		TP	10	8.18	6.98	3.02	4.82
		FP	6.92	22.68	3.27	0.2	1.94

Table B.2: Simulation results for Gaussian linear regression under the three specified settings with noise-to-signal ratio $\gamma = 1$. Ideal, Naive, and IRO use the MCP penalty for regularization.

B.3.2. Scaled Lasso Results

To illustrate the IRO-algorithm with another methodology, we opted to illustrate the incorporation of the Scaled Lasso penalty [50]. The Scaled Lasso penalty incorporates the residual error term into the Lasso estimation procedure, which is necessary for the imputation step. We display the results in Tables B.3 and B.4. The overall results are similar to the MCP penalty, with a few differences. One difference is that the Naive model had a difficult time finding convergence. This occurred in both results for $\gamma = 0.5$ and $\gamma = 1$. We believe this can be attributed to the residual variability being confused with the covariate mismeasurement variability. This would lead to poor estimation of σ_ϵ^2, and hence $\boldsymbol{\beta}$. The Second differnce

is slight degredation of performance for the IRO-correct. This is likely due to the bias incorporated into the estimate from the ℓ_1 penalty.

Setting	p	Metric	Ideal	Naive	IRO	CLasso	GMUS
G1	500	L2	0.422	NA	0.507	0.61	0.654
		TP	6.87	6.73	5.45	6.43	3.83
		FP	4.79	53.81	2.23	15	0.23
	1000	L2	0.448	NA	0.542	0.62	0.676
		TP	6.55	6.18	5.12	6.07	3.61
		FP	4.99	13.18	2.19	18.14	0.34
G2	500	L2	1.009	NA	1.425	1.227	1.89
		TP	10	9.99	10	10	9.93
		FP	5.25	107.28	2.34	19.34	0.21
	1000	L2	1.069	NA	1.516	1.14	1.949
		TP	10	10	10	10	9.93
		FP	4.87	13.3	2.17	22.44	0.2
G3	500	L2	2.315	NA	2.65	3.24	2.793
		TP	8.99	8.28	6.72	7.58	5.37
		FP	4.55	42.47	2.98	9.87	0.97
	1000	L2	2.548	NA	2.767	3.239	2.836
		TP	7.87	7.17	5.52	7.27	4.94
		FP	5.15	24.31	2.62	12.1	1.49

Table B.3: Simulation results for Gaussian linear regression under the three specified settings with noise-to-signal ratio $\gamma = 0.5$. Ideal, Naive, and IRO use the Scaled Lasso penalty for regularization. Due to convergence issues, many L2 norms for the naive method are missing and denoted NA.

Setting	p	Metric	Ideal	Naive	IRO	CLasso	GMUS
G1	500	L2	0.419	NA	0.574	1.45	0.735
		TP	7.07	6.67	4.42	2.72	3.62
		FP	5.01	46.3	1.42	1.8	0.22
	1000	L2	0.453	NA	0.614	1.408	0.762
		TP	6.34	5.72	4.06	2.41	3.43
		FP	4.79	17.88	1.22	2.38	0.4
G1	500	L2	0.983	NA	1.77	3.367	2.195
		TP	10	9.95	9.96	7.28	9.62
		FP	4.92	91.09	1.51	1.13	0.15
	1000	L2	1.057	NA	1.938	3.372	2.257
		TP	10	10	9.91	7.07	9.62
		FP	4.8	33	1.14	1.23	0.34
G1	500	L2	2.299	NA	2.78	5.891	2.853
		TP	9.06	7.98	5.44	2.88	5.24
		FP	5.03	65.15	1.76	0.14	1.76
	1000	L2	2.533	NA	2.839	5.461	2.885
		TP	7.81	6.79	4.63	3.02	4.82
		FP	4.63	17.58	1.58	0.2	1.94

Table B.4: Simulation results for Gaussian linear regression under the three specified settings with noise-to-signal ratio $\gamma = 1$. Ideal, Naive, and IRO use the Scaled Lasso penalty for regularization. Due to convergence issues, many L2 norms for the naive method are missing and denoted NA.

B.4. Other Binomial Regression Results

We display the results when $\gamma = 1$ for the binomial linear regression simulation found in Chapter 3.4.2. Besides Setting B2 being slightly more in favor of the IRO-adjusted procedure, the results are similar.

Setting	p	Metric	Ideal	Naive	IRO	CLasso	GMUS
B1	100	L2	0.695	1.748	1.21	3.613	2.713
		TP	10	9.98	9.91	2.3	9.22
		FP	4.07	5.56	1.73	0.05	0.53
	500	L2	0.933	1.987	1.658	3.702	2.846
		TP	10	9.86	9.62	2.03	8.53
		FP	8.51	13.07	3.07	0.07	0.89
	1000	L2	1.026	2.117	1.938	3.735	2.887
		TP	10	9.83	9.38	2.07	8.17
		FP	11.86	17.51	4.08	0.12	1.41
B2	100	L2	0.731	2.171	2.059	3.374	2.949
		TP	9.97	8.82	7.88	1.95	5.41
		FP	4.99	9.23	3.11	0.1	1.6
	500	L2	0.967	2.701	2.726	3.266	3.009
		TP	9.74	6.18	5	1.43	4.37
		FP	12.8	10.83	3.35	0.05	2.38
	1000	L2	1.216	2.825	2.859	3.276	3.026
		TP	9.38	5.29	4.1	1.36	4
		FP	17.98	12.35	3.35	0.06	3.01

Table B.5: Simulation results for Binomial linear regression under the two specified settings with signal-to-noise ratio $\gamma = 1$. The Ideal, Naive, and IRO procedures use the MCP penalty for regularization.

B.5. Details for Data Analysis

Here we illustrate the ELBO-plots for both the CLasso and GMUS as performed in the data analysis found in Section 3.5. These plots, generated by the 'hdme' package output, show the tuning parameter on the x-axis and the number of non-zero coefficients on the y-axis. The authors encourage picking where the number of non-zero coefficients stabilize. That is to say, pick the tuning parameter where the following tuning parameters give the same number of non-zero coefficients. We present the plots in Figure B.1, where the left and right plot is for CLasso and GMUS, respectively. As noted in Section 3.4.2, the optimality of the solution for CLasso only holds for the Gaussian case, hence the bumpiness. Hence, for CLasso we opt to choose the largest radius value in the grid that gives the number of non-zero coefficients to be 3 as this is the most common amount in a short succession. GMUS begins to stablize at 0.2, and this is the value used for the analysis.

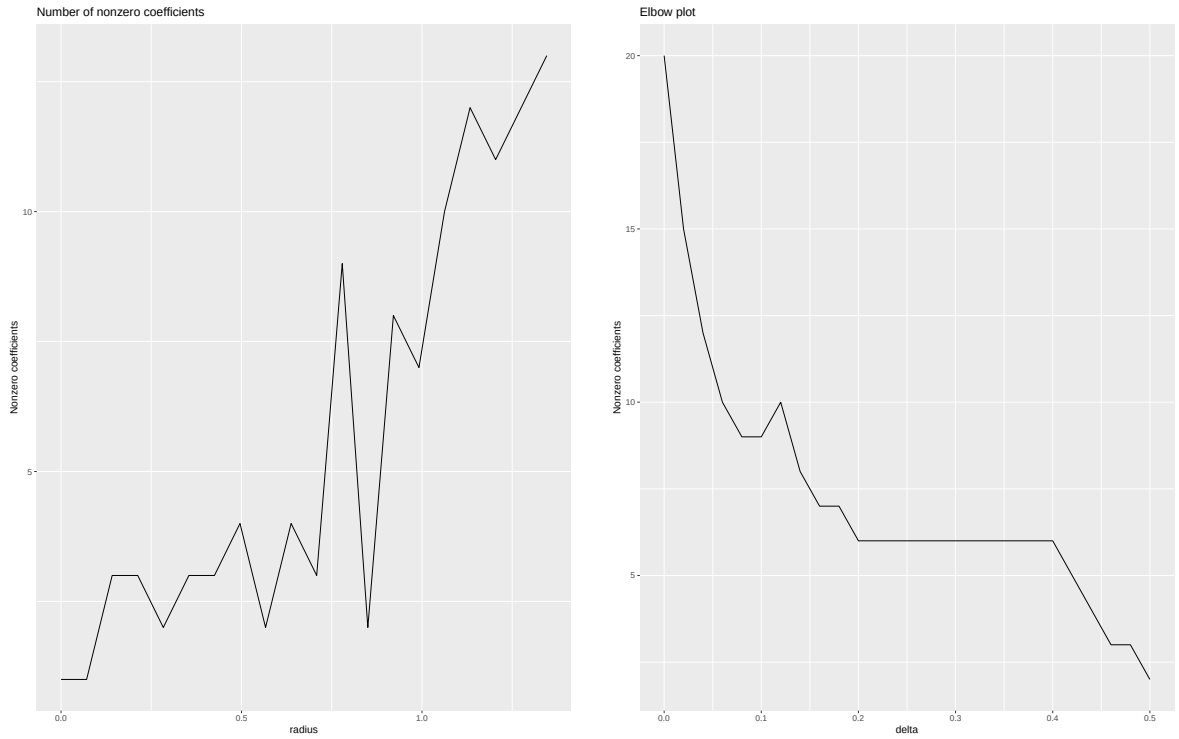

Figure B.1: Outputted ELBO-plots for CLasso (left) and GMUS (right). Note that the increase of the regularization parameter has varying affect, hence the opposing trend.

BIBLIOGRAPHY

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

[2] Alexandre Belloni, Mathieu Rosenbaum, Alexandre B Tsybakov, et al. An $\{\ell_1, \ell_2, \ell_\infty\}$-regularization approach to high-dimensional errors-in-variables models. *Electronic Journal of Statistics*, 10(2):1729–1750, 2016.

[3] Anirban Bhattacharya, Antik Chakraborty, and Bani K Mallick. Fast sampling with Gaussian scale mixture priors in high-dimensional regression. *Biometrika*, pages 985–991, 2016.

[4] Patrick Breheny and Jian Huang. Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. *The Annals of Applied Statistics*, 5(1):232, 2011.

[5] Michael Byrd, Linh Nghiem, and Monnie McGee. Bayesian regularization of Gaussian graphical models with measurement error. *arXiv preprint arXiv:1907.02241*, 2019.

[6] Tony Cai, Weidong Liu, and Xi Luo. A constrained l1 minimization approach to sparse precision matrix estimation. *Journal of the American Statistical Association*, 106(494):594–607, 2011.

[7] Raymond J Carroll, David Ruppert, Leonard A Stefanski, and Ciprian M Crainiceanu. *Measurement error in nonlinear models: a modern perspective*. Chapman and Hall/CRC, 2006.

[8] Jianfei Chen, Jun Zhu, Zi Wang, Xun Zheng, and Bo Zhang. Scalable inference for logistic-normal topic models. In *Advances in Neural Information Processing Systems*, pages 2445–2453, 2013.

[9] Hee Min Choi, James P Hobert, et al. The Polya-Gamma Gibbs sampler for Bayesian logistic regression is uniformly ergodic. *Electronic Journal of Statistics*, 7:2054–2064, 2013.

[10] Abhirup Datta and Hui Zou. A note on cross-validation for lasso under measurement errors. *Technometrics*, (just-accepted):1–13, 2019.

[11] Abhirup Datta, Hui Zou, et al. Cocolasso for high-dimensional error-in-variables regression. *The Annals of Statistics*, 45(6):2400–2426, 2017.

[12] Arthur P Dempster. Covariance selection. *Biometrics*, pages 157–175, 1972.

[13] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

[14] Sameer K Deshpande, Veronika Rockova, and Edward I George. Simultaneous variable and covariance selection with the multivariate spike-and-slab lasso. *arXiv preprint arXiv:1708.08911*, 2017.

[15] Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1, 2010.

[16] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*, volume 1. Springer series in statistics New York, 2001.

[17] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008.

[18] Lingrui Gan, Naveen N Narisetty, and Feng Liang. Bayesian regularization for graphical models with unequal shrinkage. *Journal of the American Statistical Association*, pages 1–14, 2018.

[19] Andrew Gelman, Donald B Rubin, et al. Inference from iterative simulation using multiple sequences. *Statistical Science*, 7(4):457–472, 1992.

[20] Trevor Hastie, Robert Tibshirani, and Martin Wainwright. *Statistical learning with sparsity: the lasso and generalizations*. Chapman and Hall/CRC, 2015.

[21] Anne-Mette K Hein, Sylvia Richardson, HC Causton, Graeme K Ambler, and Peter J Green. Bgx: a fully bayesian gene expression index for affymetrix genechip data. *Biostatistics*, 6(3):349–373, 2005.

[22] Chris C Holmes, Leonhard Held, et al. Bayesian auxiliary variable models for binary and multinomial regression. *Bayesian Analysis*, 1(1):145–168, 2006.

[23] Jean Honorio. Lipschitz parametrization of probabilistic graphical models. *arXiv preprint arXiv:1202.3733*, 2012.

[24] Chiang-Ching Huang, Samantha Gadd, Norman Breslow, Colleen Cutcliffe, Simone T Sredni, Irene B Helenowski, Jeffrey S Dome, Paul E Grundy, Daniel M Green, Michael K Fritsch, et al. Predicting relapse in favorable histology Wilms tumor using gene expression analysis: a report from the Renal Tumor Committee of the Children's Oncology Group. *Clinical Cancer Research*, 15(5):1770–1778, 2009.

[25] Iain M Johnstone et al. On the distribution of the largest eigenvalue in principal components analysis. *The Annals of Statistics*, 29(2):295–327, 2001.

[26] Kshitij Khare, Sang-Yun Oh, and Bala Rajaratnam. A convex pseudolikelihood framework for high dimensional partial correlation estimation with convergence guarantees. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 77(4):803–825, 2015.

[27] Nicole Krämer, Juliane Schäfer, and Anne-Laure Boulesteix. Regularized estimation of large-scale gene association networks using graphical Gaussian models. *BMC Bioinformatics*, 10(1):384, 2009.

[28] Steffen L Lauritzen. *Graphical models*, volume 17. Clarendon Press, 1996.

[29] Faming Liang, Bochao Jia, Jingnan Xue, Qizhai Li, and Ye Luo. An imputation–regularized optimization algorithm for high dimensional missing data problems and beyond. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(5):899–926, 2018.

[30] Scott Linderman, Matthew Johnson, and Ryan P Adams. Dependent multinomial models made easy: Stick-breaking with the Pólya-Gamma augmentation. In *Advances in Neural Information Processing Systems*, pages 3456–3464, 2015.

[31] Han Liu, Lie Wang, et al. Tiger: A tuning-insensitive approach for optimally estimating gaussian graphical models. *Electronic Journal of Statistics*, 11(1):241–294, 2017.

[32] Po-Ling Loh and Martin J Wainwright. High-dimensional regression with noisy and missing data: Provable guarantees with non-convexity. In *Advances in Neural Information Processing Systems*, pages 2726–2734, 2011.

[33] Peter McCullagh. *Generalized Linear Models*. Routledge, 2019.

[34] Lukas Meier, Sara Van De Geer, and Peter Bühlmann. The group Lasso for logistic regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(1):53–71, 2008.

[35] Linh Nghiem and Cornelis Potgieter. Simulation-selection-extrapolation: Estimation in high-dimensional errors-in-variables models. *arXiv preprint arXiv:1808.10477*, 2018.

[36] Nuala A O'Leary, Mathew W Wright, J Rodney Brister, Stacy Ciufo, Diana Haddad, Rich McVeigh, Bhanu Rajput, Barbara Robbertse, Brian Smith-White, Danso Ako-Adjei, et al. Reference sequence (refseq) database at NCBI: current status, taxonomic expansion, and functional annotation. *Nucleic Acids Research*, 44(D1):D733–D745, 2015.

[37] Mee Young Park and Trevor Hastie. L1-regularization path algorithm for generalized linear models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(4):659–677, 2007.

[38] J. Perez. Preparation of rna for microarray analysis. Technical report, 2006.

[39] Kaare Brandt Petersen, Michael Syskind Pedersen, et al. The matrix cookbook. *Technical University of Denmark*, 7(15):510, 2008.

[40] Nicholas G Polson, James G Scott, and Jesse Windle. Improved Pólya-Gamma sampling. Technical report.

[41] Nicholas G Polson, James G Scott, and Jesse Windle. Bayesian inference for logistic models using Pólya–Gamma latent variables. *Journal of the American Statistical Association*, 108(504):1339–1349, 2013.

[42] David M Rocke and Blythe Durbin. A model for measurement error for gene expression arrays. *Journal of Computational Biology*, 8(6):557–569, 2001.

[43] Veronika Ročková et al. Bayesian estimation of sparse signals with a continuous spike-and-slab prior. *The Annals of Statistics*, 46(1):401–437, 2018.

[44] Veronika Ročková and Edward I George. The spike-and-slab lasso. *Journal of the American Statistical Association*, 113(521):431–444, 2018.

[45] Mathieu Rosenbaum, Alexandre B Tsybakov, et al. Sparse recovery under matrix uncertainty. *The Annals of Statistics*, 38(5):2620–2651, 2010.

[46] Håvard Rue, Sara Martino, and Nicolas Chopin. Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71(2):319–392, 2009.

[47] Adrian FM Smith and Gareth O Roberts. Bayesian computation via the Gibbs sampler and related Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Methodological)*, 55(1):3–23, 1993.

[48] Øystein Sørensen, Arnoldo Frigessi, and Magne Thoresen. Measurement error in lasso: Impact and likelihood bias correction. *Statistica Sinica*, pages 809–829, 2015.

[49] Øystein Sørensen, Kristoffer Herland Hellton, Arnoldo Frigessi, and Magne Thoresen. Covariate selection in high-dimensional generalized linear models with measurement error. *Journal of Computational and Graphical Statistics*, 27(4):739–749, 2018.

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

[51] Kean Ming Tan, Yang Ning, Daniela M Witten, and Han Liu. Replicates in high dimensions, with applications to latent variable graphical models. *Biometrika*, 103(4):761–777, 2016.

[52] Jonathan Taylor and Robert J Tibshirani. Statistical learning and selective inference. *Proceedings of the National Academy of Sciences*, 112(25):7629–7634, 2015.

[53] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

[54] Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. Sparsity and smoothness via the fused Lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(1):91–108, 2005.

[55] Ernest Turro, Natalia Bochkina, Anne-Mette K Hein, and Sylvia Richardson. BGX: a Bioconductor package for the Bayesian integrated analysis of Affymetrix GeneChips. *BMC Bioinformatics*, 8(1):439, 2007.

[56] Sara A Van de Geer et al. High-dimensional generalized linear models and the Lasso. *The Annals of Statistics*, 36(2):614–645, 2008.

[57] Xue Wang, Mingcheng Wei, and Tao Yao. Minimax concave penalized multi-armed bandit model with high-dimensional covariates. In *International Conference on Machine Learning*, pages 5200–5208, 2018.

[58] Joe Whittaker. *Graphical models in applied multivariate statistics*. Wiley Publishing, 2009.

[59] Daniela M Witten and Robert Tibshirani. Penalized classification using fisher's linear discriminant. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(5):753–772, 2011.

[60] Tong Tong Wu, Kenneth Lange, et al. Coordinate descent algorithms for Lasso penalized regression. *The Annals of Applied Statistics*, 2(1):224–244, 2008.

[61] Xinyang Yi and Constantine Caramanis. Regularized em algorithms: A unified framework and statistical guarantees. In *Advances in Neural Information Processing Systems*, pages 1567–1575, 2015.

[62] Guan Yu and Yufeng Liu. Sparse regression incorporating graphical structure among predictors. *Journal of the American Statistical Association*, 111(514):707–720, 2016.

[63] Ming Yuan and Yi Lin. Model selection and estimation in the Gaussian graphical model. *Biometrika*, 94(1):19–35, 2007.

[64] Cun-Hui Zhang et al. Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 38(2):894–942, 2010.

[65] Tuo Zhao, Han Liu, Kathryn Roeder, John Lafferty, and Larry Wasserman. The huge package for high-dimensional undirected graph estimation in R. *Journal of Machine Learning Research*, 13(Apr):1059–1062, 2012.

[66] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320, 2005.