Spring 5-19-2019

Advances in Measurement Error Modeling

Linh Nghiem
Southern Methodist University, lnghiem@smu.edu

Advances in Measurement Error Modeling

Approved by:

Cornelis Potgieter
Assistant Professor of Statistical Science

Lynne Stokes
Professor of Statistical Science

Daniel Heitjan
Professor of Statistical Science

Jing Cao
Associate Professor of Statistical Science

Pankaj Choudmary
Professor of Statistics, UT Dallas

Advances in Measurement Error Modeling

A Dissertation Presented to the Graduate Faculty of

Dedman College

Southern Methodist University

in

Partial Fulfillment of the Requirements

of the degree of

Doctor of Philosophy

with a

Major in Statistical Science

by

Linh Nghiem

B.S.B.A., Finance and Mathematics, University of Miami

May 18, 2019

Nghiem, Linh B.S in Business Administration, University of Miami

Advances in Measurement Error Modeling

Advisor: Dr. Cornelis Potgieter
Doctor of Philosophy conferred May 18, 2019
Dissertation completed April 30, 2019

Measurement error in observations is widely known to cause bias and a loss of power when fitting statistical models, particularly when studying distribution shape or the relationship between an outcome and a variable of interest. Most existing correction methods in the literature require strong assumptions about the distribution of the measurement error, or rely on ancillary data which is not always available. This limits the applicability of these methods in many situations. Furthermore, new correction approaches are also needed for high-dimensional settings, where the presence of measurement error in the covariates adds another level of complexity to the desirable structure of the models, such as sparsity. This dissertation presents new correction methods for measurement error in two important statistical problems: density deconvolution and errors-in-variables models.

For both density deconvolution and linear errors-in-variables regression, new estimators based on the empirical phase function are proposed. Compared to the existing methods, phase function-based estimators require only mild assumptions about the measurement error distribution. For high-dimensional errors-in-variables models, a new estimator that extends the flexible Simulation-Extrapolation (SIMEX) correction procedure is proposed in order to achieve sparsity of the solution. All the new estimators have been shown to have strong theoretical support and good finite sample performance. Data examples are provided to illustrate the practical use of each estimator in reality.

Contents

List of Figures

List of Tables

Acknowledgements

I would like to express my deepest gratitude to many people who have taught, helped, supported, and mentored me during my time at Southern Methodist University (SMU). My past four years at SMU was full of both joyful and challenging moments; together, they have made a long-lasting impact in my career and my life.

I am indebted to the generous support of my advisor, Cornelis Potgieter. Not only he mentored me tirelessly on how to do research and helped me considerably in my job application, but his great dedication to student is also an excellent example for me to follow in my career. In 2018, although having many life and career incidents, he did not let them prevent him from being positive and energetic in every conversation with me, other students and colleagues. Such an attitude deeply influenced me and will continue to shape me in the future. Thanks again, Nelis!

In my first year of the PhD program, my beloved mother passed away during an accident in a minor surgery. I couldnt overcome this tremendous loss without the support of the faculty and my classmates; they were always with me when I needed help, both academically and emotionally. Even after many months later, they still asked about how my family was doing and shared my sorrows on the first Christmas after my mothers death. These supports meant a lot to me and motivated me not to give up at that difficult time.

During the PhD program, I also had the pleasure of collaborating with many other Statistics faculty and classmates on the projects that may or may not be directly related to the thesis. Particularly, I am thankful to collaborating and writing papers with Michael Byrd in three research projects. Our tons of ad-hoc discussions in the office and through the Google Hangout clarified many issues and helped us learn from each other very well.

Furthermore, I am very grateful to the experience of working in diverse roles that the

department offered me. In addition to doing research on statistical methodology, I had opportunities to work as a statistical consultant, a teaching assistant, and a tutor working with student on one-to-one basis. These experience improved my statistical communication to a wide range of audiences and opened up many interdisciplinary collaborations that bring statistics into real lives.

During my stay at SMU and Dallas, I made some great friends and my life would not have been the same without them. I would like to thank Michael Byrd, Thomas Crutcher and Emily Boak for our meals at a ton of places in Dallas, especially when you let me go to Asian places most of the time. Thank you Chinh Nguyen, Duyen Le, Duc Truong, Kathy Le, Thao Nguyen, Ngoan Tran, Luan Vu, Tu-Anh Tran, Ly Nguyen, Hiep Tran, Duc Vu, Minh Nguyen, for delicious Vietnamese food and countless parties. Thank you many Vietnamese friends around the US who hosted and traveled with me around the United States. Any many others who have interacted and supported me in this wonderful adventure.

Finally, I would like to thank my parents and my family for their endless and unconditional love and support. When my mother was alive, she strongly supported me to pursue the PhD and an academic career. Although my career journey is still far ahead with many uncertainties, I hope at this point she is smiling proudly from the heaven.

Chapter 1
Introduction

Statistics is the science of collecting, visualizing, and analyzing data. However, data are obtained from measurement processes that are subject to errors. The sources of measurement error can range from the lack of accuracy in the instruments used to measure variables to the inadequacy of short-term measurements for long-term variables. As a result, it is common that the obtained data are not samples of the variables of interests, but consist of contaminated versions of these variables. Broadly speaking, measurement error modeling refers to statistical models that correct for measurement errors in such scenarios.

Measurement errors are well-known to have a substantial impact on statistical models. Particularly, the impacts are most serious when trying to understand the effect of the variable of interest on a specific measured outcome, or when trying to understand the shape of the population distribution of the variable of interest. In general, measurement errors cause bias and loss of power in statistical models. For example, consider the simple linear regression model, $Y = \beta_0 + \beta_1 X + \varepsilon$, and the data consists of pairs (Y_i, W_i) with $W_i = X_i + U_i,\ i = 1, \ldots, n$, with U_i being the measurement error for observation i. If measurement error is ignored, regression of Y on W results in an inconsistent estimator of both the intercept β_0 and the slope β_1, see Carroll et al. (2006). Therefore, measurement error should be accounted for to understand the true relationship between Y and X.

The above example represents a typical *errors-in-variables* (EIV) models. In such models, the general interest is to model an outcome of interest Y as a function of p_1-dimensional error-prone covariates $\boldsymbol{X}$ and p_2-dimensional error-free covariates $\boldsymbol{Z}$. The function is usually involved some some parameters Θ. However, the observed sample consists of measurements $(\boldsymbol{W}_1, \boldsymbol{Z}_1, Y_1), \ldots, (\boldsymbol{W}_n, \boldsymbol{Z}_n, Y_n)$, with $\boldsymbol{W}_i = \boldsymbol{X}_i + \boldsymbol{U}_i$, $i = 1, \ldots, n$

where the measurement errors vector $\boldsymbol{U}_i$ are assumed to have mean zero and covariance matrix $\boldsymbol{\Sigma}_u$. The parameter Θ can be finite dimensional, such as in the case of linear and generalized linear models, Cox survival models, or infinite dimensional, such as in the case of nonparametric regression models.

Most popular correction methods for measurement errors in EIV models require strong distributional assumptions or potentially unavailable auxiliary data for model estimation. Specifically, it is generally assumed that the covariance matrix Σ_u is known. In the simple linear regression case ($p_1 = 1$ and $p_2 = 0$), a consistent and unbiased estimator of the slope β_1 is obtained as $\hat{\beta}_1 = \hat{\beta}_1^W \sigma_X^2/(\sigma_X^2 + \sigma_U^2)$, where $\hat{\beta}_1^W$ is the estimated slope obtained by regressing Y on W. Calculation of this estimator requires that the variance of measurement error σ_U^2 be known or estimable. In the multiple predictor setting, there is generally no closed-form solution for the corrected estimator. Instead, simulation-extrapolation (SIMEX), first proposed by Stefanski and Cook (1995) and Küchenhoff et al. (2006), is frequently used. The SIMEX procedure evaluates the effects of measurement error on the estimator by increasing the level of measurement error through a simulation step, and then extrapolating to the setting of no measurement error. SIMEX also requires that Σ_u be known or estimable. Another approach to correcting for measurement error is regression calibration, see Carroll and Stefanski (1990). Here, a regression of $\boldsymbol{X}$ on $\boldsymbol{W}$ is used to estimate $\boldsymbol{X}$, say $\hat{\boldsymbol{X}}$, and then the linear model parameters are estimated by regressing Y on $\hat{\boldsymbol{X}}$. The regression of $\boldsymbol{X}$ on $\boldsymbol{W}$ is assumed to be available through an either validation data or an instrumental variable $\boldsymbol{T}$.

When the distributions of $\boldsymbol{X}$ and $\boldsymbol{U}$ are fully specified, likelihood methods can also be used to estimate parameters, see Schafer and Purdy (1996) and Higdon and Schafer (2001). Implementation of these likelihood methods generally requires the use of numerical methods such as Gaussian quadrature or Monte Carlo integration. The EM algorithm of Dempster et al. (1977) can also be used. An approach that does not require the distribution of $\boldsymbol{X}$ to be known is the conditional score method of Stefanski and Carroll (1987). However, this method does require that both parametric models for $Y|\boldsymbol{X}$ and $\boldsymbol{W}|\boldsymbol{X}$ be specified.

For the linear EIV models where $Y = \boldsymbol{X}\beta + \varepsilon$, $\boldsymbol{W} = \boldsymbol{X} + \boldsymbol{U}$, another approach to estimating coefficients β is based on the method of moments, dating back to the work of Reiersøl (1941), who estimated the slope of the simple EIV model through third-order moments. Gillard (2014) considered slope estimators based on third and fourth moments, and finds these to have large variances. More recently, methods based on the matching of higher-order moments (or variants such as cumulants) have been explored with renewed interest. Erickson and Whited (2002) expressed high-order residual moments as nonlinear functions of both coefficients and nuisance parameters, while Erickson et al. (2014) expressed the third and fourth residual cumulants as a linear function of the coefficients. The latter also established that the two methods were asymptotically equivalent. The method of moments approach is nonparametric, in that it does not require parametric distributions to be specified for any of the components. However, an implementation based on the first M sample moments generally requires $2M$ finite population moments.

In high dimensional setting, the presence of measurement error introduces an added layer of complexity and can have severe consequences on the lasso estimator: the number of non-zero estimates can be inflated, sometimes dramatically, and as such the true sparsity pattern of the model is not recovered, see Rosenbaum et al. (2010). Several methods have been proposed that correct for measurement error in high-dimensional setting. Rosenbaum et al. (2010) proposed a matrix uncertainty selector (MU) for additive measurement error in the linear model. Rosenbaum et al. (2013) proposed an improved version of the MU selector, and Belloni et al. (2017) proved its near-optimal minimax properties and developed a conic programming estimator that can achieve the minimax bound. The conic estimators require selection of three tuning parameters, a difficult task in practice. Another approach for handling measurement error is to modify the loss and conditional score functions used with the lasso, see Sørensen et al. (2015) and Datta et al. (2017). Additionally, Sørensen et al. (2018) developed the generalized matrix uncertainty selector (GMUS) for the errors-in-variables generalized linear models. Both the conditional score approach and GMUS require the subjective choice of tuning parameters.

The second problem where it is important to correct for measurement errors is in

density estimation. This problem is often referred to as *density deconvolution*. When the noise-to-signal ratio is large, implementing a correction becomes crucial as the density of the observed data can deviate substantially from the true density of interest. Let $f_X(x)$ denote the density function of a random variable X, and assume that it is of interest to estimate $f_X(x)$ when X is not directly observable. Specifically, we are only able to observe contaminated versions of X, say $W = X + U$, where U represents measurement error. Thus, we are interested in estimating the density function of X based on an observed sample $W_1, W_2, ..., W_n$ with $W_i = X_i + U_i, i = 1, \ldots, n$. Here, the X_i are an *iid* sample from a distribution with density f_X, with U_i representing the measurement error of the i^{th} observation. The U_i are assumed both mutually independent and independent of the X_i.

The nonparametric density deconvolution problem when first considered assumed that the distribution of the measurement error was fully known, see Carroll and Hall (1988) and Stefanski and Carroll (1990). The development that followed in the literature mostly considered the case of known measurement error, and generally treated the measurement error as homoscedastic, including Fan (1991a), Fan (1991b), Fan and Truong (1993), Hall and Qiu (2005), Lee et al. (2010). The case of heteroscedastic measurement error was considered by Fan (1992) and Delaigle and Meister (2008). The problem of the measurement error having an unknown distribution was considered by Diggle and Hall (1993) and Neumann and Hössjer (1997) who assumed that samples of error data are available, and by Delaigle et al. (2008) who used replicate data to estimate the entire characteristic function of the measurement error. McIntyre and Stefanski (2011) considered the heteroscedastic case with replicate observations. Their work assumed the measurement errors all follow a normal distribution with unknown variances only. The phase function deconvolution approach developed by Delaigle and Hall (2016) is groundbreaking in that they estimate the density function f_X with both the measurement error distribution and variance unknown, and without the need for replicate data. Their method is based on minimal assumptions: The measurement error terms U_i are only assumed to be mutually independent and independent of the X_i and to have a strictly positive characteristic

function. However, Delaigle and Hall (2016) only considered the case where the U_i are homoscedastic, while heteroscedastic data is a reality often encountered in practice. In fact, the variance of measurement error often increases with the true underlying value, see Guo and Little (2011).

This thesis proposes new estimators based on the empirical phase functions and a new estimation procedure for errors-in-variables models in high dimensional settings. Specifically, in chapter 2, we develop a new density deconvolution estimator when the measurement errors are heteroscedastic of unknown type. In chapter 3, we apply the phase function method to linear errors-in-variables (EIV) models. In chapter 4, we propose a new estimation procedure that augments the traditional SIMEX for EIV models in high dimensional settings. Chapter 5 concludes the thesis with a brief summary and future direction.

Chapter 2
Density Deconvolution with Heteroscedastic Measurement Error of Unknown Type

2.1. Overview

This chapter considers the problem of density estimation when the measurement error is present. The density estimators that adjust for measurement error are broadly referred to as density deconvolution estimators. While most methods in the literature assume the distribution of the measurement error to be fully known, a recently proposed method based on the empirical phase function (EPF) can deal with the situation when the measurement error distribution is unknown. The EPF density estimator has only been considered in the context of additive and homoscedastic measurement error; however, the measurement error of many biomedical variables is heteroscedastic in nature. In this chapter, we developed a phase function approach for density deconvolution when the measurement error has unknown distribution and is heteroscedastic. A weighted empirical phase function (WEPF) is proposed where the weights are used to adjust for heteroscedasticity of measurement error. The asymptotic properties of the WEPF estimator are evaluated. Simulation results show that the weighting can result in large decreases in mean integrated squared error (MISE) when estimating the phase function. The estimation of the weights from replicate observations is also discussed. Finally, the construction of a deconvolution density estimator using the WEPF is compared to an existing deconvolution estimator that adjusts for heteroscedasticity, but assumes the measurement error distribution to be fully known. The WEPF estimator proves to be competitive, especially when considering that it relies on minimal assumption of the distribution of measurement error.

2.2. Introduction

Many biomedical variables cannot be measured with great accuracy, leading to observations contaminated by measurement error. Examples of such variables have been suggested in numerous epidemiological and clinical settings, including the measurement of blood pressure, radiation exposure, and dietary patterns, see Carroll et al. (2006). The sources of measurement error range from the instruments used to measure the variables of interest to the inadequacy of short-term measurements for long-term variables; as such, the observed measurements have larger variance than the true underlying quantity of interest. The presence of measurement error can have a substantive impact on statistical inference. For example, not correcting for measurement error can result in biased parameter estimates, and loss of power in detecting relationships among variables, see Carroll et al. (2006). Appropriate corrections need to be implemented when performing any data analysis with measurement error present to avoid making erroneous inferences.

In this chapter, we develop the phase function approach for density deconvolution when the measurement error has unknown distribution and is heteroscedastic. The model considered in this chapter assumes the observed data are of the form $W_i = X_i + \sigma_i \varepsilon_i$ where the X_i are an *iid* sample from f_X, the measurement error terms ε_i are independent and each ε_i has a positive characteristic function and satisfies $\mathrm{E}(\varepsilon_i) = 0$ and $\mathrm{Var}(\varepsilon_i) = 1$. The σ_i are non-negative constants and represent measurement error heteroscedasticity. Specifically, $\mathrm{Var}(W_i) = \sigma_X^2 + \sigma_i^2$ where σ_X^2 denotes the variance of X. Additionally, it is assumed that the random variable X is asymmetric. This assumption is fundamental to the identifiability of the phase function of X, which forms the basis of estimation. A more detailed discussion of the model assumptions is presented in Section 2.3.1, see also Delaigle and Hall (2016).

Note that the heteroscedasticity of the measurement error will require either that the constants σ_i be known, or that there are replicate data so that the σ_i can be estimated from the data. To illustrate the use of this estimator in a biomedical setting, a real-data example is included in Section 2.5. This example uses data from the Framingham

Heart Study, which collected several variables related to coronary heart disease for study subset of $n = 1615$ patients. For each patient, two measurements of long-term systolic blood pressure (SBP) were collected at each of two examinations. The distribution of true long-term SBP is estimated using the empirical phase function (EPF) and weighted empirical phase function (WEPF) density deconvolution estimator. These estimators are compared to a naive density estimator that makes no correction for measurement error, as well as the estimator of Delaigle and Meister (2008) assuming the measurement error follows a Laplace distribution.

The remainder of the chapter is organized as follows. Section 2.3 discusses the model assumptions, considers estimation of the phase function and introduces a weighted empirical phase function (WEPF) which adjusts for heteroscedasticity in the data. A small simulation study compares two different weighting schemes. Section 2.4 shows how the WEPF can be inverted to estimate the density function f_X and presents an approximation of the asymptotic mean integrated squared error for selecting the bandwidth. The WEPF deconvolution estimator is compared to that of Delaigle and Meister (2008), who treat the heteroscedastic case with known measurement error distribution. Section 2.5 illustrates the method using data from the Framingham Heart Study and Section 2.6 contains some concluding remarks.

2.3. Phase Function Estimation

2.3.1. Model and Main Assumptions

The model considered in the chapter assumes the observed data are of the form $W_i = X_i + \sigma_i \varepsilon_i$ where the X_i are an an *iid* sample from f_X, the measurement error terms ε_i are mutually independent and independent from X_i, and that each ε_i has a strictly positive characteristic function. Note that the model does not require that the ε_i have the same type of distribution, but only that each ε_i has a characteristic function satisfying the above requirement. The assumption of a strictly positive characteristic function is equivalent to ε_i being symmetric about zero with support on the entire real line. Many commonly used continuous distributions, including the Gaussian, Laplace, and Student's

t distributions, satisfy this assumption. In general, the only symmetric distributions excluded are those defined on bounded intervals (such as the uniform). For convenience, it is assumed that $\text{Var}[\varepsilon_i] = 1$, so that the constant σ_i^2 represents the heteroscedastic measurement error variance of the i^{th} observation. Specifically, $\text{Var}(W_i) = \sigma_X^2 + \sigma_i^2$ where σ_X^2 denotes the variance of X. The density function f_X is assumed to be asymmetric. More specifically, it is assumed that the random variable X does not have a symmetric component. This means that there is *no* symmetric random variable S for which X can be decomposed as $X = X_0 + S$ for arbitrary random variable X_0. This asymmetry is crucial to the ability to estimate the true density function of X. As discussed in Delaigle and Hall (2016), if one were to assumed that the density function f_X were sampled from a random universe of distributions, then the assumption of indecomposability is satisfied with probability 1. Practically, the indecomposability assumption is not unreasonable as data are rarely observed from a perfectly symmetric distribution. There is a special type of distribution for X that cannot be recovered by this method, namely when X is itself a convolution (sum) of a skew distribution and a symmetric distribution. The result from Delaigle and Hall (2016) indicates that this need not be a concern for the general practitioner implementing this method. While the exposition in this chapter assumes that the measurement error components are independent, the methodology could be generalized to a setting where $\text{Cov}[\varepsilon_i, \varepsilon_j] = \sigma_{ij} \neq 0$ for some pairs $i \neq j$. This would not affect the proposed estimator directly, but would have consequences for how the bandwidth is chosen. The latter question is beyond the scope of the present chapter.

2.3.2. The Weighted Empirical Phase Function (WEPF)

The phase function of a random variable X, denoted $\rho_X(t)$, is defined as the characteristic function of X standardized by its norm,

$$\rho_X(t) = \frac{\phi_X(t)}{|\phi_X(t)|} \tag{2.1}$$

with $\phi_Z(t)$ the characteristic function of a random variable Z and $|z| = (z\bar{z})^{1/2}$ denoting the norm function with $\bar{z}$ the complex conjugate of z. Let $W = X + \sigma\varepsilon$ with ε having

characteristic function $\phi_\varepsilon(t) \geq 0$ for all t. It is easy to verify that the random variables W and X have the same phase function, $\rho_W(t) = \rho_X(t)$. Delaigle and Hall (2016) used this relation and an empirical estimate of $\phi_W(t)$ in equation (2.1) to estimate the phase function, see their paper for details on implementation.

In the case of heteroscedastic errors, we propose to use a weighted empirical phase function (WEPF) to adjust for heteroscedasticity. Define function

$$\hat{\phi}_W(t|\boldsymbol{q}) = \sum_{j=1}^{n} q_j \exp(itW_j) \tag{2.2}$$

where $\boldsymbol{q} = \{q_1, \ldots, q_n\}$ denotes a set of non-negative constants that sum to 1. Function (2.2) is a weighted empirical characteristic function and noting random variable $W_i = X_i + \sigma_i\varepsilon_i$ has characteristic function $\phi_{W_i}(t) = \phi_X(t)\phi_{\varepsilon_i}(\sigma_i t)$, $i = 1, \ldots, n$, it follows that

$$\mathrm{E}[\hat{\phi}_W(t|\boldsymbol{q})] = \phi_X(t) \sum_{j=1}^{n} q_j \phi_{\varepsilon_j}(\sigma_j t).$$

The WEPF is defined as

$$\hat{\rho}_W(t|\boldsymbol{q}) = \frac{\hat{\phi}_W(t|\boldsymbol{q})}{|\hat{\phi}_W(t|\boldsymbol{q})|} = \frac{\sum_j q_j \exp(itW_j)}{\left\{\sum_j \sum_k q_j q_k \exp[it(W_j - W_k)]\right\}^{1/2}}. \tag{2.3}$$

For $\boldsymbol{q}_{eq} = \{1/n, \ldots, 1/n\}$, $\hat{\rho}_W(t|\boldsymbol{q}_{eq})$ essentially reduces to the phase function proposed by Delaigle and Hall (2016). Use of weights choice $\boldsymbol{q}_{eq}$ will be referred to as the empirical phase function (EPF) estimator. Other choices of weights can serve as an adjustment for heteroscedasticity – observations with large measurement error variance can be down-weighted to have smaller contribution to the phase function estimate.

The asymptotic properties of the WEPF are given in the Theorem 2.1 below.

Theorem 2.1. *Assume that* $\max_j q_j = \mathcal{O}(n^{-1})$ *and that each measurement error component* ε_j *has a strictly positive characteristic function. It then follows that the WEPF as defined in* (2.3) *is a consistent estimator of the phase function of* W*, and hence of the phase function of* X*. Also, the asymptotic variance of the WEPF is given by*

$$\text{AVar}[\hat{\rho}_W(t|\boldsymbol{q}) - \rho_W(t)] = \frac{1}{2|\phi_X(t)|^2 \psi_\varepsilon(t|\boldsymbol{q})} \sum_{k=1}^{n} q_k^2 \left[1 - |\phi_X(t)|^2 \phi_{\varepsilon_k}^2(\sigma_k t) + \phi_{\varepsilon_k}^2(\sigma_k t)\right] - \frac{\text{Re}\{\phi_X^2(t)\,\phi_X(-2t)\}}{2|\phi_X(t)|^4 \psi_\varepsilon(t|\boldsymbol{q})} \sum_{k=1}^{n} q_k^2 \phi_{\varepsilon_k}(2\sigma_k t) \tag{2.4}$$

where $\psi_\varepsilon(t|\boldsymbol{q}) = \left[\sum_k q_k \phi_{\varepsilon_k}(\sigma_k t)\right]^2$.

The proof of Theorem 2.1 can be found in the Appendix 2.7.1. Equation (2.4) shows that the asymptotic variance of $\hat{\rho}_W(t|\boldsymbol{q})$ depends on $\phi_{\varepsilon_j}(t)$ $j = 1, \ldots, n$, the characteristic functions of the measurement error components. While one would ideally like to choose weights $\boldsymbol{q}$ that minimize said asymptotic variance, this is unrealistic as the method proposed in this chapter makes no parametric assumptions about the measurement error, meaning the ϕ_{ε_j} are unknown. A much simpler weighting scheme is proposed here, relying only on knowledge of the measurement error variances.

Note that $\text{E}(W_i) = \text{E}(X) = \mu$. As such, for weights $\boldsymbol{q}$, the estimator $\hat{\mu}_{\boldsymbol{q}} = \sum_{j=1}^{n} q_j W_j$ is an unbiased estimator of μ. The weights

$$q_i^* = \sigma_{W_i}^{-2} \Big[\sum_{j=1}^{n} \sigma_{W_j}^{-2}\Big]^{-1} = (\sigma_X^2 + \sigma_i^2)^{-1} \Big[\sum_{j=1}^{n} (\sigma_X^2 + \sigma_j^2)^{-1}\Big]^{-1} \tag{2.5}$$

result in a minimum variance estimator of μ. This does have a connection to the phase function, as $\rho_X'(0) = \mu$; see the supplemental material of Delaigle and Hall (2016) for the connection between the phase function and the odd moments of the underlying distribution. Let $\boldsymbol{q}_{opt} = \{q_1^*, \ldots, q_n^*\}$ denote the vector of mean-optimal weights and let WEPF_{opt} denote the weighted empirical phase function estimator calculated using the mean-optimal weights. Both the performance of the EPF and the WEPF_{opt} will be considered for estimating the phase function and density function.

2.3.3. Estimating the Variance Components

In practice, it is often the case that neither the measurement error variances $\sigma_1^2, \ldots, \sigma_n^2$ nor σ_X^2 is known. These quantities can be easily estimated from replicate observations.

This section describes how to estimate the variance components for a heteroscedastic measurement error variance model. In a setting where the underlying measurement error variance structure is unknown, the procedure outlined in this section can be used to estimate the mean-optimal weights in (2.5) used for estimating the WEPF.

Consider replicate observations, $W_{ij} = X_i + \tau_i e_{ij}$, $j = 1, \ldots, n_i$, $i = 1, \ldots, n$ with $\min_i n_i \geq 2$, $E(e_{ij}) = 0$, $\mathrm{Var}(e_{ij}) = 1$, and τ_i^2 representing heteroscedastic measurement error variance at the observation level. Note that $W_{ij} - W_{ij'} = \tau_i (e_{ij} - e_{ij'})$ and thus $\mathrm{E}\left[(W_{ij} - W_{ij'})^2\right] = 2\tau_i^2$ for $j \neq j'$. Define grand mean

$$\bar{W} = \frac{1}{n} \sum_{i=1}^{n} \left[\frac{1}{n_i} \sum_{j=1}^{n_i} W_{ij} \right] = \frac{1}{n} \sum_{i=1}^{n} X_i + \frac{1}{n} \sum_{i=1}^{n} \left[\frac{\tau_i}{n_i} \sum_{j=1}^{n_i} e_{ij} \right]$$

and note that $\mathrm{E}(\bar{W}) = \mu$ and

$$\mathrm{Var}(\bar{W}) = \frac{\sigma_X^2}{n} + \frac{1}{n^2} \sum_{i=1}^{n} \frac{\tau_i^2}{n_i}.$$

It can also be shown that

$$E\left[\left(W_{ij} - \bar{W} \right)^2 \right] = \sigma_X^2 + \tau_i^2 + \mathcal{O}(n^{-1}). \tag{2.6}$$

Subsequently, the variance components can be estimated by

$$\hat{\tau}_i^2 = \frac{1}{n_i \left(n_i - 1 \right)} \sum_{j=1}^{n_i - 1} \sum_{j'=j+1}^{n_i} \left(W_{ij} - W_{ij'} \right)^2, \quad i = 1, \ldots, n,$$

and, motivated by (2.6), $\hat{\sigma}_X^2 = \dfrac{1}{N} \sum_{i=1}^{n} \sum_{j=1}^{n_i} (W_{ij} - \bar{W})^2 - \dfrac{1}{n} \sum_{i=1}^{n} \hat{\tau}_i^2$ with $N = \sum_i n_i$. The analysis then proceeds by defining individual-level averages $W_i = (n_i^{-1}) \sum_{j=1}^{n_i} W_{ij}$ and noting that $W_i = X_i + \sigma_i \varepsilon_i$ where $\sigma_i = \tau_i / \sqrt{n_i}$ and ε_i has a distribution with a positive characteristic function whenever the same is true for all elements of the set $\{e_{i1}, \ldots, e_{in_i}\}$. The estimate of σ_i is given by $\hat{\sigma}_i = \hat{\tau}_i / \sqrt{n_i}$.

2.3.4. Simulation Study

A small simulation study was conducted to compare the performance of the EPF and WEPF_{opt} estimators. The true X_i data were sampled from the following three distributions: (1) $X \sim \chi^2_3/\sqrt{6}$ (Scaled χ^2_3), (2) $X \sim (0.5\text{N}(1,1) + 0.5\chi^2(5))/\sqrt{9.5}$ (Mixture 1), and (3) $X \sim (0.5\text{N}(5, 0.6^2) + 0.5\text{N}(2.5, 1))/\sqrt{2.2425}$ (Mixture 2). The first two distributions are right-skewed while the third distribution is bimodal. All three distributions were scaled to have unit variance. The phase functions of these distributions are shown in Figure 2.4 of the Appendix 2.7.3. The measurement error terms $\varepsilon_{ij} = \tau_i e_{ij}$ were sampled from a normal distribution with mean 0 and variance structure $\tau_i^2 = J\sigma_i^2$ with $\sigma_i^2 = 0.025\sigma_X^2, i = 1, \ldots, n/2$ and $\sigma_i^2 = 0.975\sigma_X^2, i = n/2+1, \ldots, n$. For each candidate distribution of X, a total of $N = 1000$ samples $W_{ij} = X_i + \tau_i e_{ij}$, $i = 1, \ldots, n$ and $j = 1, \ldots, J$ were generated for sample sizes $n = 250, 500,$ and 1000. Scenarios with no replicates ($J = 1$) and also with replicates ($J = 2$ and 3) were considered in the simulation. Under the scenario with no replication, the measurement error variance was treated as known. In settings with $J = 2$ and 3 replicates, the measurement error variances were estimated from the replicate data using the procedure outlined in Section 2.3.3. The choice of observation-level measurement error variance $\tau_i^2 = J\sigma_i^2$ results in the combined replicate values $W_i = J^{-1}\sum_j W_{ij}$ having measurement error variance σ_i^2. This was done to make the simulation results with and without replicates easily comparable. For each simulated dataset, the mean-optimal weight vector $\boldsymbol{q}_{opt}$ was calculated (or estimated in the case of replicate data) using equation (2.5). The WEPF_{opt} estimator was then calculated using these weights. Additionally, the EPF estimator was calculated using equal weights for all observations. As the quality of the empirical characteristic function decreases with increasing t, the suggestion of Delaigle & Hall Delaigle and Hall (2016) was followed and the estimated phase functions were only computed on the interval $[-t^*, t^*]$, where t^* is the smallest $t > 0$ such that $|\hat{\phi}_W(t|\boldsymbol{q})| < n^{-1/4}$. The EPF and WEPF are compared using (estimated) mean integrated squared error (MISE) ratios, $\text{MISE}_{eq}/\text{MISE}_{opt}$, where MISE_{eq} and MISE_{opt} denote the MISEs of the EPF and WEPF_{opt} estimators respectively. The results are summarized in Table 2.1.

Replicates	Distribution	$n = 250$	$n = 500$	$n = 1000$
No replicate	$X \sim \chi^2_3/\sqrt{6}$	1.220 (0.021)	1.280 (0.020)	1.277 (0.023)
	$X \sim$ Mixture 1	1.298 (0.023)	1.321 (0.022)	1.303 (0.022)
	$X \sim$ Mixture 2	1.065 (0.017)	1.085 (0.018)	1.109 (0.019)
2 replicates	$X \sim \chi^2_3/\sqrt{6}$	1.075 (0.016)	1.155 (0.018)	1.139 (0.018)
	$X \sim$ Mixture 1	1.044 (0.007)	1.021 (0.006)	1.005 (0.004)
	$X \sim$ Mixture 2	1.003 (0.004)	1.007 (0.003)	1.007 (0.002)
3 replicates	$X \sim \chi^2_3/\sqrt{6}$	1.150 (0.019)	1.177 (0.019)	1.150 (0.020)
	$X \sim$ Mixture 1	1.020 (0.008)	1.017 (0.006)	1.001 (0.004)
	$X \sim$ Mixture 2	1.001 (0.004)	1.005 (0.003)	1.008 (0.002)

Table 2.1: The ratio $\text{MSE}_{eq}/\text{MSE}_{opt}$ and the corresponding jackknife standard error (in parentheses) when estimating the phase function of X with normal measurement error and variance structure given in Case 1 of Table 2.2, based on $N = 1000$ samples, when there are no replicate (assuming the true variances of measurement errors are known), 2 replicates, and 3 replicates per observation.

In Table 2.1, an MISE ratio greater than 1 indicates better performance of the WEPF_{opt} estimator compared to the EPF estimator. The table also reports estimated standard errors for the MISE ratios. The standard errors were estimated using the following jackknife procedure. For the j^{th} simulated sample, let $(\text{ISE}_{eq,j}, \text{ISE}_{opt,j})$ denote the integrated squared error for the EPF and the WEPF_{opt} respectively, $j = 1, \ldots, N$. Let $R_{(-j)}$ denote the MISE ratio calculated after deleting the j^{th} ISE pair. Then, the jackknife standard error for the MISE ratio is given by

$$\text{SE}_{jack} = \sqrt{\frac{1}{N}\sum_{j=1}^{N}\left(R_{(-j)} - \bar{R}\right)^2}$$

where $\bar{R} = N^{-1}\sum_{j=1}^{N} R_{(-j)}$.

Inspection of Table 2.1 shows that the WEPF_{opt} performs better than the EPF for the measurement error configuration considered. When the measurement error variances are known, the gain from using WEPF_{opt} can be substantial. Specifically, the MISE of WEPF_{opt} is seen to between 6.5% and 30% lower than the MISE of the EPF for the distributions considered. When there are $J = 2$ and $J = 3$ replicates per observation, the WEPF_{opt} performs slightly better than the EPF for the scaled χ^2_3 distribution, while their performance is nearly identical for Mixtures 1 and 2. In this setting, the use of the suggested weighting scheme never results in poorer performance of the WEPF_{opt}

Case	Variance Structure
Case 1	$\sigma_i^2 = 0.025\sigma_X^2, i = 1, \ldots, n/2$ and $\sigma_i^2 = 0.975\sigma_X^2, i = n/2+1, \ldots, n$
Case 2	$\sigma_i^2 = (0.25 + 0.5i/n)\sigma_X^2$, $i = 1, \ldots, n$
Case 3	$\sigma_i^2 = (0.025 + 0.95i/n)\sigma_X^2$, $i = 1, \ldots, n$

Table 2.2: Three measurement error variance structures used in simulations.

Replicates	X	Case 1	Case 2	Case 3
No replicate	$X \sim \chi_3^2/\sqrt{(6)}$	1.277 (0.023)	1.030 (0.005)	1.113 (0.002)
	$X \sim$ Mixture 1	1.303 (0.022)	1.027 (0.006)	1.117 (0.012)
	$X \sim$ Mixture 2	1.109 (0.019)	1.011 (0.006)	1.039 (0.012)
2 replicates	$X \sim \chi_3^2/\sqrt{(6)}$	1.139 (0.018)	0.925 (0.014)	0.978 (0.015)
	$X \sim$ Mixture 1	1.005 (0.004)	0.992 (0.005)	0.998 (0.004)
	$X \sim$ Mixture 2	1.007 (0.002)	1.001 (0.003)	1.002 (0.002)
3 replicates	$X \sim \chi_3^2/\sqrt{(6)}$	1.150 (0.020)	0.965 (0.014)	1.034 (0.016)
	$X \sim$ Mixture 1	1.001 (0.004)	0.994 (0.004)	0.998 (0.004)
	$X \sim$ Mixture 2	1.008 (0.002)	0.999 (0.002)	1.002 (0.002)

Table 2.3: The effect of the error variance structure on the ratio $\text{MISE}_{eq}/\text{MISE}_{opt}$ and the corresponding jackknife standard error (in parentheses) based on 1000 samples of size $n = 1000$.

estimator compared to the EPF estimator.

Next, the effect of different underlying measurement error variance structures on the MISE ratio of the EPF and WEPF_{opt} was examined. The sample size was fixed at $n = 1000$ and the three different measurement error variance structures considered are outlined in Table 2.2. The ratios $\text{MSE}_{eq}/\text{MSE}_{opt}$ based on 1000 simulated datasets are reported in Table 2.3. Again, jackknife estimates of standard error are also reported.

Inspection of Table 2.3 illustrates the effect of different heterogeneity patterns of measurement error variances on the performance of the EPF and WEPF_{opt} estimators. When the measurement error variances are known ($J = 1$), the WEPF_{opt} has a lower MISE than the EPF in all the considered configurations, with the heterogeneity pattern only affecting the size of the improvement. In the case of $J = 2$ replicates per observation, there were four instances in Case 2 and Case 3 of measurement error variances where the EPF performed better than the WEPF_{opt}. This occurrence was likely because the estimated weights for WEPF_{opt} were calculated from estimated variance components based on only a small number of replicates. When the number of replicates increases from $J = 2$ to

$J = 3$, measurement error variances are estimated with higher accuracy, so the MISE ratio increase in general. Note that, although using WEPF_{opt} can sometimes lead to a worse performance, the loss tends to be small (at most 8% as seen in the Case 2 measurement error variance setting when X follows a Scaled-χ^2_3 with 2 replicates); however, using WEPF_{opt} can still result in large gains (as much as 15% in the Case 1 measurement error variance setting when X follows a Scaled-χ^2_3 with 3 replicates).

In general, the simulation study shows that weighting to adjust for heteroscedasticity in estimating the phase function never results in a much poorer estimator, but sometimes leads to a large gain in efficiency. The loss/gain depends on how accurate measurement error variances were estimated as evidenced by the improvement in going from $J = 2$ to $J = 3$ replicates. In the next section, this is explored in the context of density deconvolution.

2.4. Density Estimation

2.4.1. Constructing an Estimator of f_X

The outline here is a brief overview of how the method of Delaigle and Hall (2016) can be implemented using the WEPF to estimate the density function f_X. Let $\hat{\phi}_W(t|\boldsymbol{q})$ and $\hat{\rho}_W(t|\boldsymbol{q})$ denote the weighted empirical characteristic function and corresponding WEPF respectively. Let $w(t)$ denote a non-negative weight function. Also let x_j, $j = 1, \ldots, m$ denote a set of arbitrary values with respective probability masses p_j. Delaigle and Hall (2016) suggest a two-stage estimation method for f_X. First, one finds a characteristic function of the form $\psi(t|\mathbf{x}, \mathbf{p}) = \sum_j p_j \exp(itx_j)$ that has phase function close to the WEPF. Since this characteristic function corresponds to a discrete distribution with probability mass p_j at the point x_j for $j = 1, \ldots, m$, the second stage of estimation involves smoothing $\psi(t|\mathbf{x}, \mathbf{p})$ before applying an inverse Fourier transformation to obtain the estimated density $\hat{f}_X(x)$. Delaigle and Hall (2016) suggest sampling the x_j uniformly on the interval $[\min W_i, \max W_i]$ with $m = 5\sqrt{n}$. The goal is then to find the set $\{p_j\}_{j=1}^m$

that minimizes

$$T(\boldsymbol{p}) = \int_{-\infty}^{\infty} \left| \hat{\rho}_W(t|\boldsymbol{q}) - \frac{\psi(t|\mathbf{x}, \mathbf{p})}{|\psi(t|\mathbf{x}, \mathbf{p})|} \right|^2 w(t)dt \tag{2.7}$$

under the constraint of also minimizing the variance of the corresponding discrete distribution, $v(\mathbf{p}) = \sum_{j=1}^{m} p_j x_j^2 - (\sum_{j=1}^{m} p_j x_j)^2$. This non-convex optimization problem of finding the solution $\{\hat{p}_j\}_{j=1}^{m}$ can be solved using MATLAB. Details are given in Delaigle and Hall (2016). The present implementation differs only in that the estimated phase function is weighted to adjust for heteroscedasticity. Beyond using a different estimator of the phase function, the optimization problem remains unchanged.

Now, let $\psi(t|\mathbf{x}, \hat{\boldsymbol{p}}) = \sum_j \hat{p}_j \exp(itx_j)$ be the characteristic function with the $\hat{p}_j$s the probability masses estimated by minimizing (2.7). The deconvolution density estimator based on the WEPF is then

$$\hat{f}_X(x) = \frac{1}{2\pi} \int \exp(-itx)\, \tilde{\phi}(t)\, K^{\mathrm{ft}}(ht)\, dt \tag{2.8}$$

where

$$\tilde{\phi}(t) = \begin{cases} \psi(t|\mathbf{x}, \hat{\boldsymbol{p}}), & \text{for } t \le t^* \\ r(t), & \text{for } t > t^* \end{cases}$$

with t^* being the smallest $t > 0$ such that $|\hat{\phi}_W(t|\boldsymbol{q})| < n^{-1/4}$. Here, $K^{\mathrm{ft}}(t)$ denotes the Fourier transform of a deconvolution kernel function and $r(t)$ denotes a ridging function. The ridging function ensures that the estimator is well-behaved outside the range $[-t^*, t^*]$. The proposed choice of ridging function is $r(t) = \hat{\phi}_W(t|\boldsymbol{q})/\hat{\phi}_L(t)$, with $\hat{\phi}_L(t)$ the characteristic function of a Laplace distribution with variance equal to an estimator of $\sigma_L^2 = \sum_j q_j \sigma_j^2$, the weighted sum of the measurement error variances. In application here, the common choice $K^{\mathrm{ft}}(t) = (1-t^2)^3$ for $|t| \le 1$ is used. The weight function is chosen to be $w(t) = \omega(t)|\hat{\phi}_W(t|\mathbf{q})\psi(t|\mathbf{x}, \mathbf{p})|^2$ with $\omega(t) = 0.75(1-t^2)$ for $|t| \le 1$ (the Epanechnikov kernel) rescaled to the interval $[-t^*, t^*]$. This choice of weight function avoids numerical difficulties that can arise when dividing by very small numbers.

2.4.2. Bandwidth Selection

The proposed phase function deconvolution estimator that accounts for heteroscedasticity in (2.8) is an approximation of the estimator

$$\tilde{f}(x) = \frac{1}{2\pi}\int \exp(-itx) K^{\mathrm{ft}}(ht) \frac{\hat{\phi}_W(t|\boldsymbol{q})}{\sum_j q_j \phi_{\varepsilon_j}(\sigma_j t)} dt \tag{2.9}$$

with $\hat{\phi}_W(t|\boldsymbol{q})$ defined in (2.2). Note that (2.9) is an estimator that one could compute if the measurement error distribution were known, but that it is different from the heteroscedastic estimator proposed by Delaigle and Meister (2008). Taking expectation of the integrated squared error (ISE) of (2.15), ISE $= \int[\tilde{f}(x) - f_X(x)]^2 dx$, gives mean integrated squared error (MISE)

$$\begin{aligned}\mathrm{MISE} &= \frac{1}{2\pi}\int |\phi_X(t)|^2 \left[K^{\mathrm{ft}}(ht) - 1\right]^2 dt + \frac{1}{2\pi}\int \left[K^{\mathrm{ft}}(ht)\right]^2 \frac{\sum_j q_j^2}{\left[\sum_j q_j \phi_{\varepsilon_j}(\sigma_j t)\right]^2} dt \\ &\quad - \frac{1}{2\pi}\int |\phi_X(t)|^2 \left[K^{\mathrm{ft}}(ht)\right]^2 \frac{\sum_j q_j^2 \phi_{\varepsilon_j}^2(\sigma_j t)}{\left[\sum_j q_j \phi_{\varepsilon_j}(\sigma_j t)\right]^2} dt. \end{aligned} \tag{2.10}$$

An argument similar to that of Delaigle and Meister (2008) when evaluating the asymptotic MISE (AMISE) of their heteroscedastic estimator, one can show that the last term of (2.10) is negligible, giving

$$\mathrm{AMISE} = \frac{1}{2\pi}\int |\phi_X(t)|^2 \left[K^{\mathrm{ft}}(ht) - 1\right]^2 dt + \frac{1}{2\pi}\int \left[K^{\mathrm{ft}}(ht)\right]^2 \frac{\sum_j q_j^2}{\left[\sum_j q_j \phi_{\varepsilon_j}(\sigma_j t)\right]^2} dt$$

In the present application, both $\phi_X(t)$ and $\phi_{\varepsilon_j}(t)$, $j = 1, \ldots, n$ are unknown. However, note that $|\phi_X(t)|^2 = \phi_X(t)\,\phi_X(-t)$ is the characteristic function of the random variable $X - X'$, where X, X' are *iid* f_X. Regardless of the shape of f_X, the random variable $X - X'$ is symmetric about 0 and has variance $2\sigma_X^2$. This suggests replacing $|\phi_X(t)|^2$ with the characteristic function of a symmetric distribution with mean 0 and variance $2\hat{\sigma}_X^2$. Appropriate choices might be the normal distribution, i.e. substituting $\exp(-\hat{\sigma}_X^2 t^2)$ for $|\phi_X(t)|^2$, or the Laplace distribution, i.e. substituting $(1 + \hat{\sigma}_X^2 t^2)^{-1}$. Additionally, one

can use appropriate approximations for $\phi_{\varepsilon_j}(\sigma_j t)$. For example, the Laplace choice is a reasonable one, see Meister (2006) and Delaigle (2008). One can therefore substitute $\left(1+0.5\hat{\sigma}_j^2 t^2\right)^{-1}$ for $\phi_{\varepsilon_j}(\sigma_j t)$. This Normal-Laplace substitution gives approximate AMISE function

$$\hat{\mathrm{A}}(h) = \frac{1}{2\pi}\int \left\{ \exp\left(-\hat{\sigma}_X^2 t^2\right)\left[K^{\mathrm{ft}}(ht)-1\right]^2 + \left[K^{\mathrm{ft}}(ht)\right]^2 \frac{\sum_j q_j^2}{\left[\sum_j q_j\left(1+0.5\hat{\sigma}_j^2 t^2\right)^{-1}\right]^2} \right\} dt \tag{2.11}$$

and the value of h that minimizes the above function can then be used to evaluate the density deconvolution estimator in equation (2.8).

2.4.3. Simulation Study

Simulation studies were done to evaluate the performance of the equally-weighted and mean-optimal weighted phase function deconvolution density estimators. These correspond to the use of the EPF and WEPF_{opt} as the phase function estimate before performing the deconvolution operation as described in Section 2.4.1. Additionally, as it is already established in the literature, the Delaigle & Meister estimator, as proposed in Delaigle and Meister (2008) for heteroscedastic data, was also calculated. The three candidate distributions for X as described in Section 2.3.4 were considered. Both normal and Laplace distributions were considered for the measurement error, each in conjunction with the three measurement error variance models outlined in Table 2.2 being considered. In all cases the sample size was taken to be $n = 500$. Due to the computational cost of evaluating the phase function deconvolution estimators, a total of 500 samples were generated for each combination of X-distribution and variance model. For the phase-function estimators, the approximate AMISE bandwidth minimizing (2.11) was computed. The bandwidth of the Delaigle-Meister estimator was a two-stage plug-in bandwidth as suggested in their paper. For all the three deconvolution estimators, the integrated squared error (ISE) was computed for each sample.

Table 2.4 presents the simulation results corresponding to the setting where the mea-

surement error variances are assumed known, and Table 2.5 presents the simulation results corresponding to the case with $J = 2$ replicates per observation and the variance components are estimated as outlined in Section 2.3.3. The simulation with replicate observations contains results for the Delaigle-Meister estimator both using the estimated variances (D&M$_{\mathrm{VarE}}$) and treating the variances as known (D&M$_{\mathrm{VarK}}$). Note that the simulations with replicate observations use the individual-level average data $W_i = (W_{i1}+W_{i2})/2$ to compute the deconvolution estimators and are therefore not directly comparable to the simulation without replication and measurement error variances assumed known. Due to the presence of outliers in the ISE calculations, the median as well as the first and third quartiles of $10 \times$ ISE are reported.

True X	Error type	Error case	EPF	WEPF$_{opt}$	D&M
Scaled χ^2_3	Normal	1	0.225	0.199	0.193
			[0.189, 0.282]	[0.159, 0.240]	[0.166, 0.230]
		2	0.483	0.482	0.458
			[0.404, 0.581]	[0.392, 0.571]	[0.386, 0.547]
		3	0.419	0.366	0.315
			[0.321, 0.493]	[0.296, 0.421]	[0.264, 0.39]
	Laplace	1	0.191	0.172	0.181
			[0.167, 0.245]	[0.147, 0.210]	[0.145, 0.213]
		2	0.311	0.306	0.299
			[0.243, 0.392]	[0.236, 0.371]	[0.229, 0.367]
		3	0.27	0.268	0.266
			[0.224, 0.352]	[0.205, 0.339]	[0.222, 0.325]
Mixture 1	Normal	1	0.184	0.140	0.117
			[0.128, 0.248]	[0.085, 0.194]	[0.082, 0.155]
		2	0.605	0.555	0.527
			[0.452, 0.723]	[0.433, 0.715]	[0.416, 0.63]
		3	0.436	0.385	0.304
			[0.319, 0.566]	[0.271, 0.503]	[0.182, 0.401]
	Laplace	1	0.142	0.107	0.105
			[0.078, 0.201]	[0.060, 0.160]	[0.073, 0.141]
		2	0.265	0.258	0.242
			[0.19, 0.384]	[0.182, 0.354]	[0.156, 0.326]
		3	0.254	0.232	0.212
			[0.178, 0.339]	[0.173, 0.293]	[0.142, 0.271]
Mixture 2	Normal	1	0.098	0.090	0.073
			[0.063, 0.175]	[0.051, 0.136]	[0.053, 0.105]
		2	0.296	0.296	0.274
			[0.224, 0.387]	[0.21, 0.391]	[0.201, 0.343]
		3	0.223	0.2	0.172
			[0.152, 0.286]	[0.132, 0.26]	[0.118, 0.217]
	Laplace	1	0.073	0.073	0.070
			[0.049, 0.128]	[0.044, 0.107]	[0.041, 0.104]
		2	0.154	0.146	0.164
			[0.1, 0.22]	[0.1, 0.23]	[0.103, 0.239]
		3	0.139	0.125	0.141
			[0.096, 0.189]	[0.081, 0.174]	[0.101, 0.192]

Table 2.4: Density estimation for $n = 500$ with no replicates and measurement error variances are assumed to be known. The median, as well as first and third quartiles, $[Q_1, Q_3]$, of 10 $\times$ ISE of density estimators under 500 simulations.

True X	Error type	Error case	EPF	WEPF$_{opt}$	D&M$_{\text{VarK}}$	D&M$_{\text{VarE}}$
Scaled χ^2_3	Normal	1	0.204	0.192	0.178	0.274
			[0.164, 0.259]	[0.156, 0.241]	[0.154, 0.205]	[0.233, 0.319]
		2	0.321	0.322	0.336	0.423
			[0.252, 0.387]	[0.267, 0.385]	[0.28, 0.405]	[0.384, 0.474]
		3	0.29	0.285	0.249	0.384
			[0.234, 0.327]	[0.237, 0.33]	[0.21, 0.298]	[0.335, 0.419]
	Laplace	1	0.176	0.165	0.148	0.209
			[0.142, 0.216]	[0.140, 0.207]	[0.123, 0.180]	[0.176, 0.246]
		2	0.277	0.273	0.281	0.343
			[0.223, 0.349]	[0.222, 0.337]	[0.234, 0.338]	[0.301, 0.378]
		3	0.219	0.218	0.23	0.298
			[0.18, 0.266]	[0.176, 0.267]	[0.184, 0.276]	[0.249, 0.325]
Mixture 1	Normal	1	0.128	0.120	0.097	0.206
			[0.088, 0.182]	[0.077, 0.166]	[0.062, 0.145]	[0.162, 0.277]
		2	0.31	0.309	0.308	0.464
			[0.214, 0.387]	[0.217, 0.4]	[0.232, 0.401]	[0.404, 0.534]
		3	0.257	0.242	0.195	0.374
			[0.175, 0.345]	[0.182, 0.339]	[0.12, 0.266]	[0.309, 0.451]
	Laplace	1	0.102	0.105	0.082	0.147
			[0.066, 0.156]	[0.074, 0.159]	[0.058, 0.117]	[0.106, 0.199]
		2	0.216	0.21	0.223	0.308
			[0.151, 0.271]	[0.14, 0.267]	[0.154, 0.272]	[0.255, 0.355]
		3	0.193	0.176	0.161	0.267
			[0.13, 0.283]	[0.119, 0.242]	[0.114, 0.244]	[0.229, 0.333]
Mixture 2	Normal	1	0.081	0.084	0.064	0.123
			[0.055, 0.111]	[0.051, 0.110]	[0.049, 0.088]	[0.098, 0.150]
		2	0.189	0.185	0.164	0.247
			[0.112, 0.251]	[0.118, 0.243]	[0.126, 0.227]	[0.204, 0.285]
		3	0.132	0.125	0.118	0.201
			[0.096, 0.193]	[0.082, 0.194]	[0.077, 0.144]	[0.172, 0.239]
	Laplace	1	0.070	0.070	0.056	0.087
			[0.049, 0.101]	[0.046, 0.099]	[0.037, 0.082]	[0.059, 0.122]
		2	0.136	0.117	0.15	0.181
			[0.086, 0.187]	[0.077, 0.163]	[0.106, 0.186]	[0.156, 0.214]
		3	0.117	0.103	0.125	0.169
			[0.076, 0.175]	[0.073, 0.165]	[0.086, 0.168]	[0.138, 0.208]

Table 2.5: Density estimation for $n = 500$ with $J = 2$ replicates for each observation. The median, as well as first and third quartiles, $[Q_1, Q_3]$, of 10 $\times$ ISE of density estimators under 500 simulations.

Inspection of Table 2.4 reveals that the Delaigle-Meister (D&M) estimator tends to have the smallest median ISE, although there are a few instances in which the phase function estimators outperform the D&M estimator, notably for Mixture 2 and Laplace

measurement error. It is also clear that calculating the mean-optimal weights is very advantageous in this setting, with the mean-optimally weighted estimator having smaller median ISE than the equally weighted estimator in all but one instance. Overall, one can conclude that the WEPF estimator performs very well and compares favorably to the D&M estimator, the latter requiring knowledge of the measurement error distribution to be useful in practice.

Inspection of the simulation results in Table 2.5 is very insightful. Note that the measurement error variances here are estimated based on only $J = 2$ replicates for each observation. As such, one might not expect good performance. However, the two phase function estimators perform very favorable when compared to the D&M estimator with known measurement error variances. The mean-optimally weighted estimator generally performs better than the equally weighted estimators in terms of median ISE, although there are two exceptions. It is interesting that weights estimated based on only two replicates give such good performance. Also revealing is that the WEPF estimator performs significantly better than the D&M estimator with estimated variances, with the median ISE of the mean-optimally weighted estimator often reflecting more than a 50% reduction in median ISE when comapared to the D&M counterpart.

Figures 2.1 and 2.2 show plots of the density estimators corresponding to the first, second, and third quantiles (Q_1, Q_2, and Q_3) of ISE for each of the methods EPF, WEPF_{opt}, and the D&M estimators corresponding to X having scaled χ^2_3 and Mixture 1 distribution. In all three instances, the estimators were calculated with estimated measurement error variances based on $J = 2$ replicates per observation. Observation-level measurement error was taken to be Case 1 of Table 2.2. Both normal and Laplace distributions were considered for the measurement error. The sample size was fixed at $n = 500$. The figures also show the true density curve for comparison. Although all three estimators considered are able to capture the shape of the true density, the D&M estimators with estimated variance do the worst among the three: For X having a scaled χ^2_3 distribution, it puts much more density in negative support than the EPF and WEPF_{opt} and tends to underestimate the modal height. Both the EPF with WEPF_{opt},

perform well for the scaled χ^2_3 distribution, with the WEPF$_{opt}$ seemingly capturing the shape around the mode a little better than the EPF. When evaluating Figure 2.2 showing the same plots for X having the distribution Mixture 1, the general observations are very similar. The EPF and WEPF$_{opt}$ have visually similar performance, while the D&M estimator underestimates the density around the mode. The Appendix 2.7.3 also contains a set of plots corresponding to X having Mixture 2 distribution. Similar observations apply there.

Additional simulation results are presented in the Appendix 2.7.3. There, the EPF, WEPF and D&M estimators are compared under the assumption that one can find an optimal bandwidth (a bandwidth minimizing ISE) for any observed sample. When no replicate data is available and the measurement error variances are assumed known, the D&M estimator has the best performance, and the WEPF outperforms the EPF in all but one case considered. However, once the measurement error variance needs to be estimated (for both $J = 2$ and $J = 3$ replicates per case), the WEPF estimator tends to have the best performance, with the D&M estimator faring worse than the EPF estimator. Finally, a simulation with plug-in bandwidth and $J = 3$ replicates is also presented. Here, the EPF and WEPF both outperform the D&M estimator.

2.5. Analysis of Framingham Data

In this section, the EPF and WEPF$_{opt}$ density deconvolution estimators are illustrated using a classical dataset in the deconvolution literature, a subset of the Framingham Heart Study. The data consists of several variables related to coronary heart disease for $n = 1615$ patients. For each patient, two measurements of long-term systolic blood pressure (SBP) were collected at each of two examination. As per Carroll et al., Carroll et al. (2006) let M_{ij} be the average of the two measurements at exam j for $j = 1, 2$, and let $W_{ij} = \log(M_{ij} - 50)$. The W_{ij} are assumed to be related to true long-term SBP, X_i according to $W_{ij} = Y_i + \sigma_i \varepsilon_{ij}$ with $Y_i = \log(X_i - 50)$. Density deconvolution is therefore used to estimate the density on the Y-scale, $\hat{f}_Y(y)$, after which it follows that $\hat{f}_X(x) = (x - 50)^{-1} \hat{f}_Y[\log(x - 50)]$, $x > 50$.

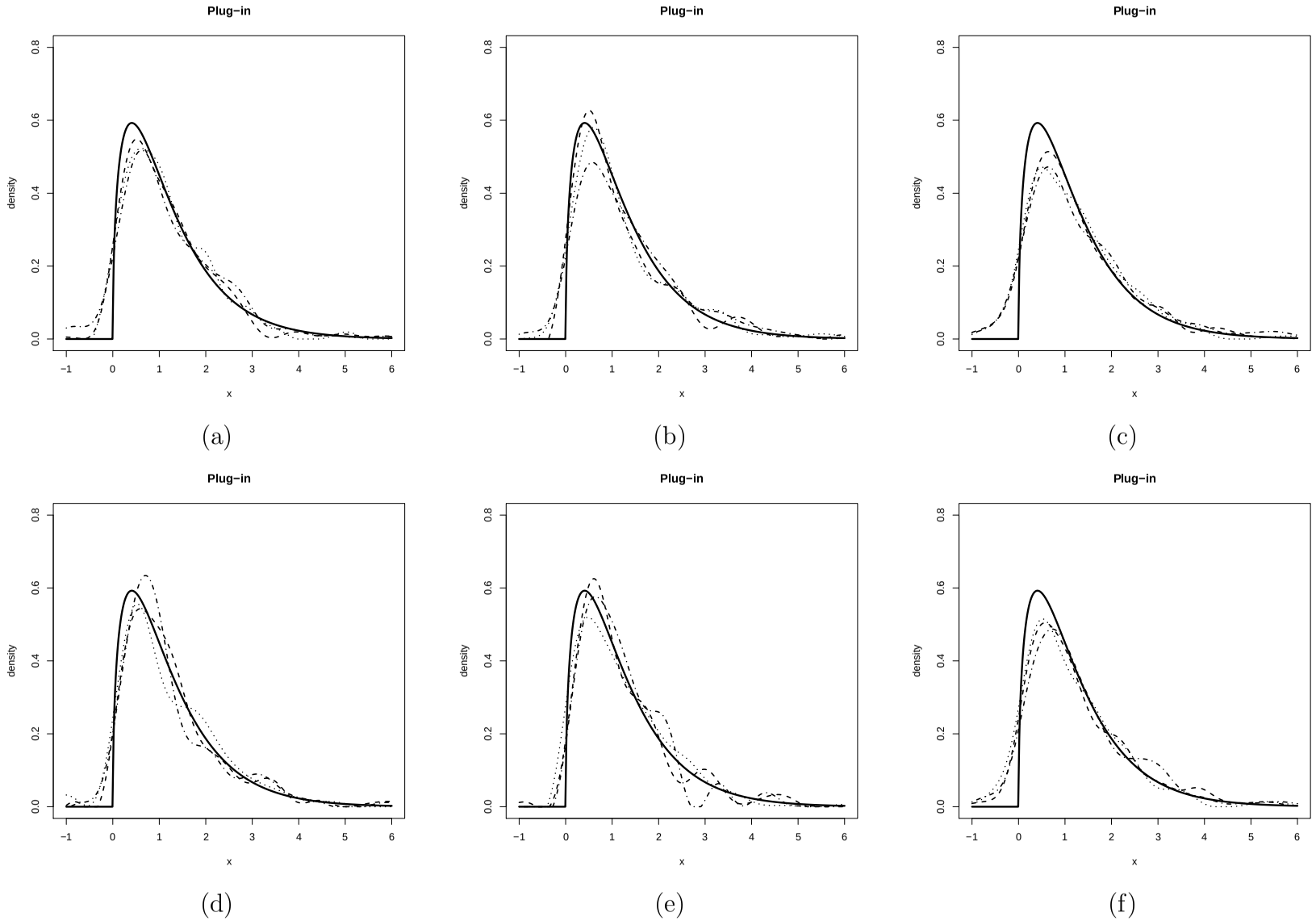


Figure 2.1: Curves Q_1 (- - - - -), Q_2 (.....), Q_3 (-.-.-.-), and true curve (——) for $X \sim$ Scaled-χ^2_3, $n = 500$, $J = 2$ replicates per observation when the errors are Normal (a)-(c), and Laplace (d)-(f), with case 1 of measurement error variances. For (a),(d): EPF estimator; (b),(e): WEPF_{opt} estimator; (c),(f): D&M estimator with estimated variances. All estimators are computed using plug-in bandwidth discussed in Section 2.4.2.

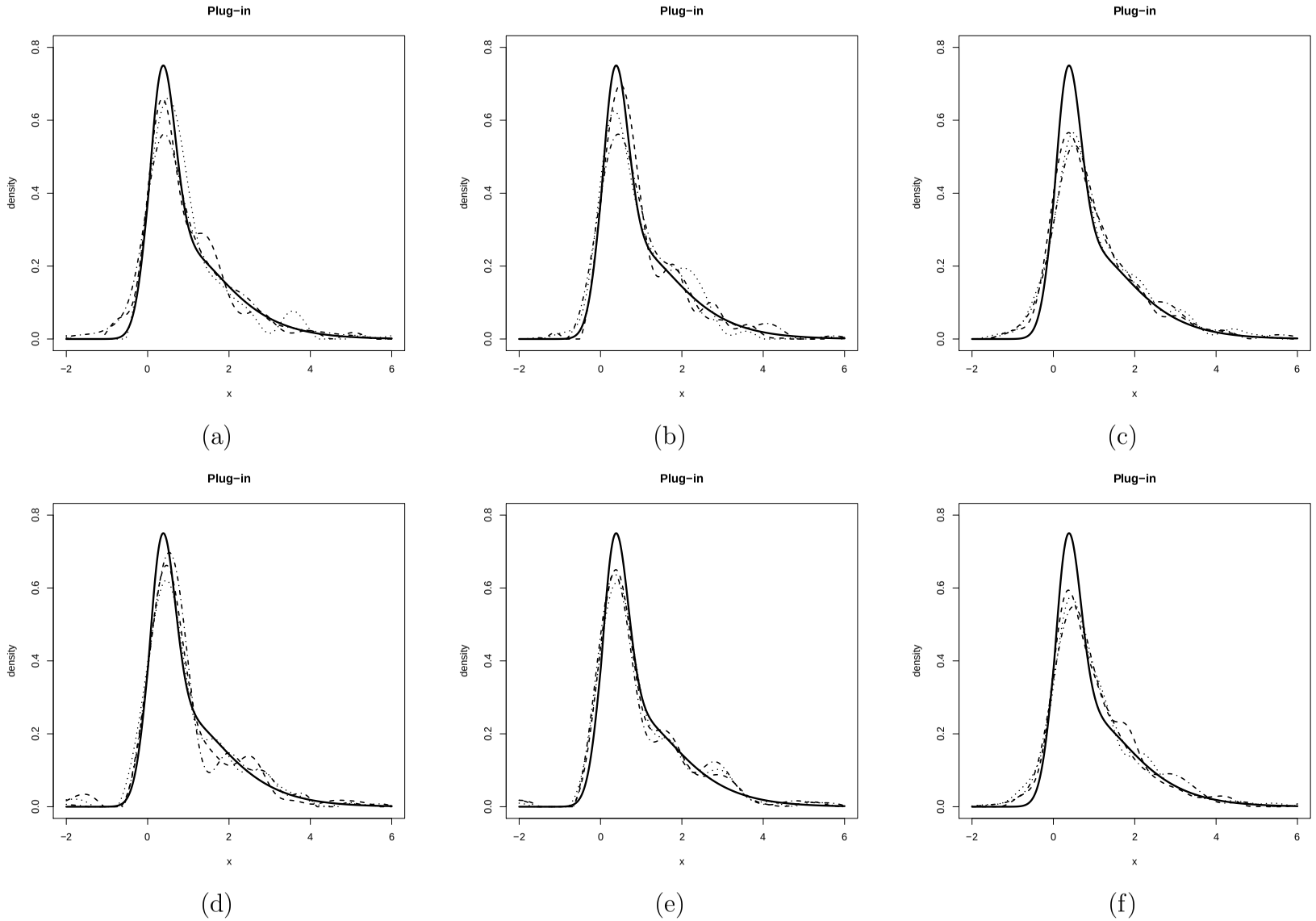


Figure 2.2: Curves Q_1 (- - - - -), Q_2 (·····), Q_3 (-·-·-·), and true curve (——) for $X \sim$ Mixture 1, $n = 500$, $J = 2$ replicates per observation, when the errors are Normal (a)-(c), and Laplace (d)-(f), with case 1 of measurement error variances. For (a),(d): EPF estimator; (b),(e): WEPF_{opt} estimator; (c),(f): D&M estimator with estimated variances. All estimators are computed using plug-in bandwidth discussed in Section 2.4.2.

For the SBP data, the EPF and WEPF$_{opt}$ were estimated, the latter with mean-optimal weights $\boldsymbol{q}_{opt}$ using variance components estimated as described in Section 2.3.3. For both the EPF and WEPF$_{opt}$, deconvolution bandwidths were estimated using (2.11). These two estimators are shown in Figure 2.3, together with the Delaigle & Meister (2008) estimator using the same estimated variances and Laplace measurement error. (The D&M estimator was also calculated for normal measurement error and was nearly identical.) A naive kernel estimator of the data using a normal references bandwidth is also shown for comparative purposes. Other bandwidth selection approaches for the naive kernel estimator were also considered with very similar results. The naive kernel estimator is much flatter around the mode and fatter in the tails. This is expected, as the kernel estimator makes no correction for the measurement error present in the data. Furthermore, it can be seen that the WEPF$_{opt}$ and EPF deconvolution density estimators are similar. The two density estimators based on phase functions suggest that the distribution of X may be multi-modal, while the D&M estimator is unimodal and positive skew.

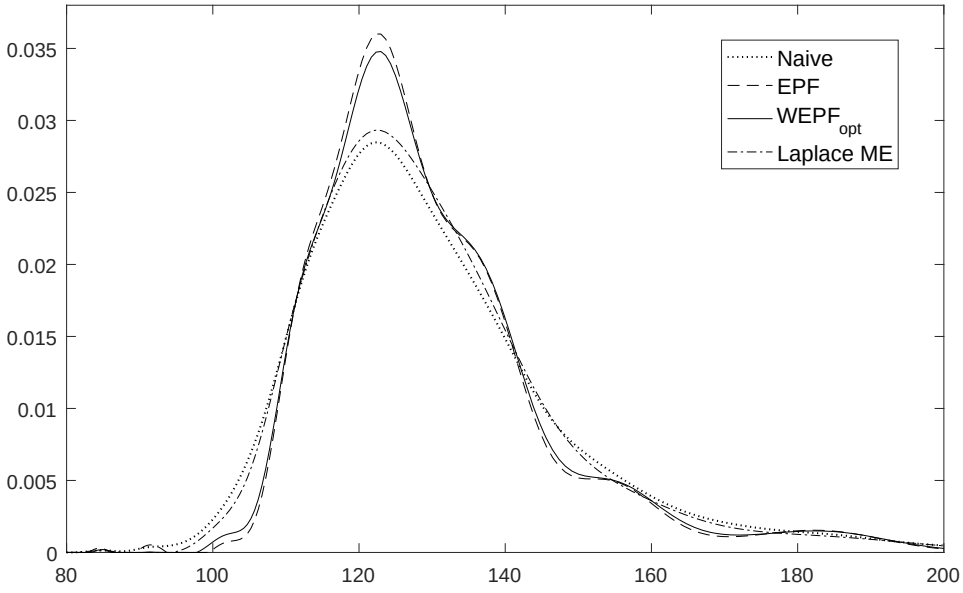


Figure 2.3: Estimation of the density f_X in the Framingham data. Four density estimates are shown: a naive kernel estimator (measurement error is ignored), the EPF estimator, the WEPF$_{opt}$ estimator, and the Delaigle & Meister estimator assuming Laplace measurement error.

2.6. Conclusions

This chapter presents a method for phase density deconvolution with heteroscedastic measurement error of unknown type and builds on the work of Delaigle and Hall (2016) who considered the homoscedastic case. Two estimators are proposed, one using equally weighted observations and the other using mean-optimal weights to adjust for heteroscedasticity of the measurement error. A method based on approximating the AMISE is proposed for bandwidth selection in both instances. In the simulation settings considered, the WEPF$_{opt}$ estimator generally performed better than the EPF estimator, although there were instances where their performance was comparable. The simulation results suggest that mean-optimal weighting of observations will not have a detrimental effect on estimating the density function, and big gains are sometimes possible. The practitioner cautious about estimaging weights from a small number of replicates could always opt for a hybrid type of estimator, calculating WEPF$_{hybrid}$ using weights $\boldsymbol{q}_{\text{hybrid}} = \alpha \boldsymbol{q}_{\text{opt}} + (1-\alpha)/n$ where α indicates their degree of confidence in using the estimated weights. The performance of this hybrid estimator is a future avenue of research. In the setting where the measurement error variances are known, the method of Delaigle and Meister (2008) will outperform both phase function estimators, although the latter are still competitive in this setting. Also recall that the Delaigle & Meister estimator requires knowledge of the measurement error distribution — an assumption not made by the EPF and WEPF estimators. When there are only 2 replicates per individual from which to estimate the measurement error variances, the phase function methods performed substantially better than the Delaigle & Meister estimator. This suggests that the phase function methods have some inherent robustness against variance estimate deviation from the true values, and that the phase function density estimators can generally do the same as Delaigle & Meister estimator with much less assumption on measurement error.

2.7. Appendix

2.7.1. Asymptotic Properties of the Weighted Empirical Phase Function (WEPF)

Assume the observed data are of the form $W_i = X_i + \sigma_i \varepsilon_i$, where the X_i are an *iid* sample from f_X, the measurement error terms ε_i are independent of one another and of the X_i, and each ε_i has a symmetric distribution with strictly positive characteristic function and satisfies $\mathrm{E}(\varepsilon_i) = 0$ and $\mathrm{Var}(\varepsilon_i) = 1$. The σ_i are non-negative constants that account for measurement error heteroscedasticity. For any random variable Z and a complex number z, denote $\phi_Z(t)$ as the characteristic function of a random variable Z and $|z| = (z\bar{z})^{1/2}$ as the norm function with $\bar{z}$ the complex conjugate of z. Define a weighted empirical characteristic function for the W_i,

$$\hat{\phi}_W(t|\boldsymbol{q}) = \sum_{j=1}^{n} q_j \exp(itW_j) \tag{2.12}$$

where $\boldsymbol{q} = \{q_1, \ldots, q_n\}$ denotes a set of non-negative constants that sum to 1. Let $\hat{\psi}_W(t|\mathbf{q})$ denote the squared norm of that function,

$$\hat{\psi}_W(t|\mathbf{q}) = \sum_j \sum_k q_j q_k \exp\left[it(W_j - W_k)\right].$$

The WEPF is defined as

$$\hat{\rho}_W(t|\boldsymbol{q}) = \frac{\hat{\phi}_W(t|\boldsymbol{q})}{\hat{\psi}_W^{1/2}(t|\boldsymbol{q})}. \tag{2.13}$$

The asymptotic properties of the WEPF are given in Theorem 2.1, which is restated here for completeness.

Theorem 2.1. *Assume that* $\max_j q_j = \mathcal{O}(n^{-1})$ *and that each measurement error component* ε_j *has a strictly positive characteristic function. It then follows that the WEPF as defined in* (2.13) *is a consistent estimator of the phase function of* W*, and hence of the*

phase function of X. Also, the asymptotic variance of the WEPF is given by

$$\begin{aligned}\mathrm{AVar}[\hat{\rho}_W(t|\boldsymbol{q}) - \rho_W(t)] = \frac{1}{2|\phi_X(t)|^2 \psi_\varepsilon(t|\boldsymbol{q})} \sum_{k=1}^{n} q_k^2 \left[1 - |\phi_X(t)|^2 \phi_{\varepsilon_k}^2(\sigma_k t) + \phi_{\varepsilon_k}^2(\sigma_k t)\right] \\ - \frac{\mathrm{Re}\left\{\phi_X^2(t)\,\phi_X(-2t)\right\}}{2|\phi_X(t)|^4 \psi_\varepsilon(t|\boldsymbol{q})} \sum_{k=1}^{n} q_k^2 \phi_{\varepsilon_k}(2\sigma_k t). \qquad (2.14)\end{aligned}$$

where $\psi_\varepsilon(t|\boldsymbol{q}) = \left[\sum_k q_k \phi_{\varepsilon_k}(\sigma_k t)\right]^2$.

Proof. Let $\phi_j(t) = \phi_{W_j}(t) = \phi_X(t)\phi_{\varepsilon_j}(\sigma_j t)$. Note that

$$\mathrm{E}\left[\hat{\phi}_W(t|\mathbf{q})\right] = \phi_X(t) \sum_j q_j \phi_{\varepsilon_j}(\sigma_j t)$$

and

$$\begin{aligned}\mathrm{E}\left[\left|\hat{\phi}_W(t|\mathbf{q})\right|^2\right] &= \sum_j q_j^2 + |\phi_X(t)|^2 \sum_j \sum_{k \neq j} q_j q_k \phi_{\varepsilon_j}(\sigma_j t)\phi_{\varepsilon_k}(-\sigma_k t) \\ &= |\phi_X(t)|^2 \left[\sum_j q_j \phi_{\varepsilon_j}(\sigma_j t)\right]^2 + \mathcal{O}(n^{-1})\end{aligned}$$

where the second expression relies on error distribution having a positive characteristic function, and also on the assumption that $\max_j q_j = \mathcal{O}(n^{-1})$. Then, by application of the generalized (non-*iid*) weak law of large numbers and Slutsky's theorem,

$$\hat{\rho}_W(t|\mathbf{q}) \to \frac{\mathrm{E}\left[\hat{\phi}_W(t|\mathbf{q})\right]}{\left\{\mathrm{E}\left[\left|\hat{\phi}_W(t|\mathbf{q})\right|^2\right]\right\}^{1/2}} = \rho_X(t) + \mathcal{O}_P(n^{-1}),$$

and thus $\hat{\rho}_W(t|\mathbf{q})$ is a consistent estimator of $\rho_X(t)$.

Next, note that $\hat{\psi}_W(t|\mathbf{q})$ is a weighted U-statistic with second-order kernel. As such, we can consider the projection of $\hat{\psi}_W(t|\mathbf{q})$ onto the space of linear statistics when evaluating its asymptotic variance. It is easily verified that the projection is given by

$$\hat{\psi}_{\mathrm{proj}}(t|\mathbf{q}) = \sum_{k=1}^{n} q_k \left[\phi(-t|\mathbf{q})\, e^{itW_k} + \phi(t|\mathbf{q})\, e^{-itW_k}\right]$$

where $\phi(t|\mathbf{q}) = \sum_{k=1}^{n} q_k \phi_k(t)$. This, combined with application of the functional delta method, gives

$$\hat{\rho}_W(t|\mathbf{q}) - \rho_W(t) = \frac{1}{\psi^{1/2}(t|\mathbf{q})}\left[\hat{\phi}_W(t|\mathbf{q}) - \phi(t|\mathbf{q})\right] - \frac{\phi(t|\mathbf{q})}{2\psi^{3/2}(t|\mathbf{q})}\left[\hat{\psi}_{\text{proj}}(t|\mathbf{q}) - \psi(t|\mathbf{q})\right] + \mathcal{O}_p\left(n^{-1}\right)$$

and

$$\begin{aligned}\text{AVar}\left[\hat{\rho}_W(t|\mathbf{q})\right] = {} & \frac{1}{\psi(t|\mathbf{q})}\text{Var}\left[\hat{\phi}_W(t|\mathbf{q})\right] + \frac{|\phi(t|\mathbf{q})|^2}{4\psi^3(t|\mathbf{q})}\text{Var}\left[\hat{\psi}_{\text{proj}}(t|\mathbf{q})\right] \\ & - \frac{\phi(-t|\mathbf{q})}{2\psi^2(t|\mathbf{q})}\text{Cov}\left[\hat{\phi}_W(t|\mathbf{q}), \hat{\psi}_{\text{proj}}(t|\mathbf{q})\right] - \frac{\phi(t|\mathbf{q})}{2\psi^2(t|\mathbf{q})}\overline{\text{Cov}\left[\hat{\phi}_W(t|\mathbf{q}), \hat{\psi}_{\text{proj}}(t|\mathbf{q})\right]}\end{aligned}$$

where $\psi(t|\mathbf{q}) = \sum_j \sum_k q_j q_k \phi_j(t)\phi_k(-t)$. Note that $\psi(t|\mathbf{q}) = |\phi(t|\mathbf{q})|^2 + \mathcal{O}_p(n^{-1})$. Some calculation now gives

$$\text{Var}\left[\hat{\phi}_W(t|\mathbf{q})\right] = \sum_j q_j^2\left[1 - |\phi_k(t)|^2\right],$$

$$\begin{aligned}\text{Var}\left[\hat{\psi}_{\text{proj}}(t|\mathbf{q})\right] = {} & 2|\phi(t|\mathbf{q})|^2 \sum_{k=1}^{n} q_k^2\left[1 - |\phi_k(t)|^2\right] + \phi^2(t|\mathbf{q}) \sum_{k=1}^{n} q_k^2\left[\phi_k(-2t) - \phi_k^2(-t)\right] \\ & + \phi^2(-t|\mathbf{q}) \sum_{k=1}^{n} q_k^2\left[\phi_k(2t) - \phi_k^2(t)\right],\end{aligned}$$

and

$$\begin{aligned}\text{Cov}\left[\hat{\phi}_W(t|\mathbf{q}), \hat{\psi}_{\text{proj}}(t|\mathbf{q})\right] = {} & \phi(t|\mathbf{q})\left[\sum_{k=1}^{n} q_k^2\left(1 - |\phi_k(t)|^2\right)\right] \\ & + \phi(-t|\mathbf{q})\left[\sum_{k=1}^{n}\left(\phi_k(2t) - \phi_k^2(t)\right)\right].\end{aligned}$$

Combining these expressions gives

$$\begin{aligned}\mathrm{AVar}\left[\hat{\rho}_W\left(t|\mathbf{q}\right)\right] = & \frac{1}{2\psi\left(t\right)}\sum_{k=1}^{n} q_k^2\left[1-\left|\phi_k\left(t\right)\right|^2\right] \\ & -\frac{1}{2\psi^2\left(t\right)}\mathrm{Re}\left\{\phi^2\left(t\right)\sum_{k=1}^{n} q_k^2\left[\phi_k\left(-2t\right)-\phi_k^2\left(-t\right)\right]\right\}.\end{aligned}$$

The desired expression is followed from noting that

$$\psi\left(t|\mathbf{q}\right) = \left|\phi_X\left(t\right)\right|^2\psi_\varepsilon\left(t|\mathbf{q}\right)$$

and

$$\phi\left(t|\mathbf{q}\right) = \phi_X\left(t\right)\phi_\varepsilon\left(t|\mathbf{q}\right),$$

where $\phi_\varepsilon\left(t|\mathbf{q}\right) = \sum_k q_k\phi_{\varepsilon_k}(\sigma_k t)$, and $\phi_\varepsilon^2\left(t|\mathbf{q}\right) = \psi_\varepsilon\left(t|\mathbf{q}\right)$. □

2.7.2. Properties of the Phase Function Density Estimator

Note that the quantity $\tilde{\phi}(t)$ in equation (2.8) of the section 2.4.1 is an approximation of the quantity $\hat{\phi}_W(t|\mathbf{q})/\left(\sum_j q_j\phi_{\varepsilon_j}(t)\right)$, where the latter cannot be calculated since the measurement error distributions are assumed unknown. An argument along the lines of one contained in the online supplemental material of Delaigle and Hall (2016) shows that it is informative to consider the estimator

$$\tilde{f}\left(x\right) = \frac{1}{2\pi}\int \exp\left(-itx\right)K^{\mathrm{ft}}\left(ht\right)\frac{\hat{\phi}_W(t|\boldsymbol{q})}{\sum_j q_j\phi_{\varepsilon_j}(\sigma_j t)}dt. \tag{2.15}$$

Estimator (2.15) cannot be used to estimate the density in practice, but it is a useful tool for investigating bandwidth selection for the estimator $\hat{f}_X(x)$ in the equation (2.8).

Estimator (2.15) has integrated squared error (ISE) given by

$$\begin{aligned}
\text{ISE} &= \int \left(\tilde{f}\left(x\right) - f_X\left(x\right)\right)^2 dt \\
&= \frac{1}{2\pi}\int \left| K^{\text{ft}}\left(ht\right) \frac{\hat{\phi}_W(t|\boldsymbol{q})}{\sum q_j \phi_{\varepsilon_j}\left(\sigma_j t\right)} - \phi_X\left(t\right)\right|^2 dt \\
&= \frac{1}{2\pi}\int \left(K^{\text{ft}}\left(ht\right)\right)^2 \frac{\left|\hat{\phi}_W(t|\boldsymbol{q})\right|^2}{\left[\sum q_j \phi_{\varepsilon_j}\left(\sigma_j t\right)\right]^2} + \frac{1}{2\pi}\int \left|\phi_X\left(t\right)\right|^2 dt \\
&\quad -\frac{1}{2\pi}\int K^{\text{ft}}\left(ht\right) \frac{\hat{\phi}_W(t|\boldsymbol{q})}{\sum q_j \phi_{\varepsilon_j}\left(\sigma_j t\right)} \phi_X\left(-t\right) dt - \frac{1}{2\pi}\int \phi_X\left(t\right) K^{\text{ft}}\left(ht\right) \frac{\hat{\phi}_W(-t|\boldsymbol{q})}{\sum q_j \phi_{\varepsilon_j}\left(\sigma_j t\right)} dt.
\end{aligned}$$

Taking expectation of ISE and substitution of the mean and variance of $\hat{\phi}_W(t|\boldsymbol{q})$ as calculated in Section 2.7.1 gives corresponding mean integrated squared error (MISE):

$$\begin{aligned}
\text{MISE} &= \frac{1}{2\pi}\int \frac{\left(K^{\text{ft}}\left(ht\right)\right)^2}{\left[\sum q_j \phi_{\varepsilon_j}\left(\sigma_j t\right)\right]^2} \left[\sum_{j=1}^n q_j^2 + \left|\phi_X\left(t\right)\right|^2 \left(\sum_{j=1}^n q_j \phi_{\varepsilon_j}\left(\sigma_j t\right)\right)^2\right] dt \\
&\quad - \frac{1}{2\pi}\int \frac{\left(K^{\text{ft}}\left(ht\right)\right)^2}{\left[\sum q_j \phi_{\varepsilon_j}\left(\sigma_j t\right)\right]^2} \left|\phi_X\left(t\right)\right|^2 \left[\sum_{j=1}^n q_j^2 \phi_{\varepsilon_j}^2\left(\sigma_j t\right)\right] dt + \frac{1}{2\pi}\int \left|\phi_X\left(t\right)\right|^2 \left[1 - 2K^{\text{ft}}\left(ht\right)\right] dt \\
&= \frac{1}{2\pi}\left(\sum_{j=1}^n q_j^2\right)\int \left(K^{\text{ft}}\left(ht\right)\right)^2 \frac{1}{\left[\sum q_j \phi_{\varepsilon_j}\left(\sigma_j t\right)\right]^2} dt \\
&\quad - \frac{1}{2\pi}\int \left|\phi_X\left(t\right)\right|^2 \left(K^{\text{ft}}\left(ht\right)\right)^2 \frac{\left[\sum_{j=1}^n q_j^2 \phi_{\varepsilon_j}^2\left(\sigma_j t\right)\right]}{\left[\sum_{j=1}^n q_j \phi_{\varepsilon_j}\left(\sigma_j t\right)\right]^2} dt + \frac{1}{2\pi}\int \left|\phi_X\left(t\right)\right|^2 \left[K^{\text{ft}}\left(ht\right) - 1\right]^2 dt \\
&= \frac{1}{2\pi}\int \left|\phi_X\left(t\right)\right|^2 \left[K^{\text{ft}}\left(ht\right) - 1\right]^2 dt + \frac{1}{2\pi}\int \left(K^{\text{ft}}\left(ht\right)\right)^2 \frac{\left[\sum_{j=1}^n q_j^2\right]}{\left[\sum_{j=1}^n q_j \phi_{\varepsilon_j}\left(\sigma_j t\right)\right]^2} dt \\
&\quad - \frac{1}{2\pi}\int \left|\phi_X\left(t\right)\right|^2 \left(K^{\text{ft}}\left(ht\right)\right)^2 \frac{\left[\sum_{j=1}^n q_j^2 \phi_{\varepsilon_j}^2\left(\sigma_j t\right)\right]}{\left[\sum_{j=1}^n q_j \phi_{\varepsilon_j}\left(\sigma_j t\right)\right]^2} dt.
\end{aligned}$$

The MISE above is an approximation to the MISE of the phase function density estimaton. The use of this approximate MISE to do bandwidth selection is discussed in section 2.4.2.

2.7.3. Additional Illustrations and Simulation Results

The simulation results comparing the EPF and WEPF$_{opt}$ estimators in Section 2.4 indicate that weighting doesn't always lead to a large improvement. While the measurement error distribution (and measurement error variance) has an impact on the quality of the estimators, the actual shape of the phase function also determines how well it can be estimated. For the three distributions considered in the simulation settings, the phase functions are plotted below in Figure 2.4.

Considering Figure 2.4, it is clear that the distribution "*Mixture 2*" has a phase function with more curvature when compared to "*Scaled* χ^2_3" and "*Mixture 1*". This also corresponds to the distribution where the WEPF$_{opt}$ has its worst performance compared to the EPF.

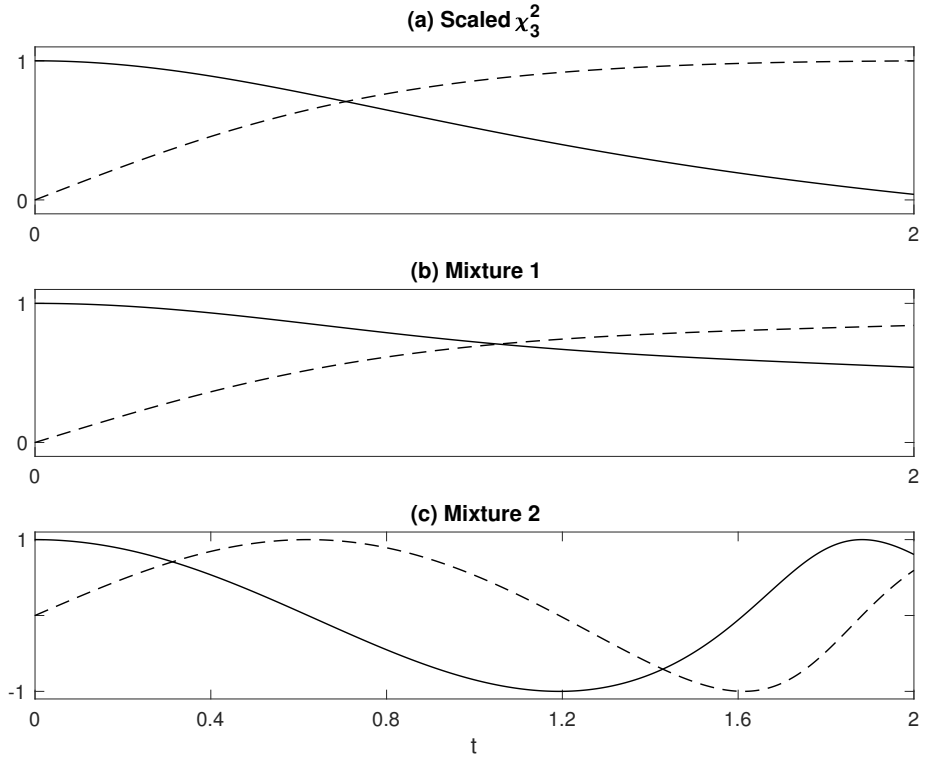


Figure 2.4: Phase functions of the three distributions considered in the simulation studies in Sections 2 and 3, real component (———) and imaginary component (- - - - -).

Next, Figure 2.5 show plots of the density estimators corresponding to the first, second,

and third quantiles (Q_1, Q_2, and Q_3) of ISE for each of the methods EPF, WEPF$_{opt}$, and the D&M estimators corresponding to X having Mixture 2 distribution. In all three instances, the estimators were calculated with estimated measurement error variances based on $J = 2$ replicates per observation. Observation-level measurement error was taken to be Case 1 of Table 2.2 in the section 2.3.4. Both normal and Laplace distributions were considered for the measurement error. The sample size was fixed at $n = 500$. The figures also show the true density curve for comparison. Although all three estimators considered are able to capture the shape of the true density, the D&M estimators with estimated variance tend to underestimate the density around the two modes and overestimate the density around the local antimode in between. Both the EPF with WEPF$_{\text{opt}}$, perform well for the Mixture 2 distribution, with the WEPF$_{opt}$ seemingly capturing the shape around the mode a little better than the EPF.

This remainder of this section compares the performance of the EPF density estimator, the mean-optimal WEPF density estimator, and the D&M density estimators computed under optimal bandwidth. The optimal bandwidth is defined as the bandwidth value that minimizes the true integrated squared error (ISE) of the corresponding estimator. For a sample $W_1, \ldots, W_n$, let $\hat{f}_{\text{est}}(x)$ denote a density estimator, where est $\in \{\text{EPF}, \text{WEPF}_{opt}, \text{D\&M}\}$ and $f_X(x)$ denote the true density of X. The true ISE is defined as:

$$\text{ISE}_{\text{est}}(h) = \int_{\mathbf{R}} (\hat{f}_{\text{est}}(x) - f_X(x))^2 dx.$$

All the settings for true distribution of X, measurement error distributions, and sample size remain the same as given in the Section 2.4.3.

Table 2.6 presents simulation results for the EPF, mean-optimal WEPF, and D&M density estimators with optimal bandwidth when measurement error variances are known, while Table 2.7 presents results for the case where measurement error variances are estimated from $J = 2$ replicates per observation. Table 2.7 presents results for the D&M estimators assuming known variances of measurement errors, as well as using the estimated variances of measurement errors. The former is included as a reference for the performance of the estimators. When using the true measurement error variances, the

D&M estimator has lower median ISE than the two phase function density estimators. However, with replicate data, although the D&M estimator computed using true variance has the lowest ISE, the D&M estimator computed using the estimated variances from replicate data has the highest median ISE. It is also clear that calculating mean-optimal weight is advantageous because the WEPF_{opt} density estimator has lower median ISE than the EPF density estimator in all but three instances.

These results are similar to the simulation results given in the Table 2.5. In all the cases, the density estimators with optimal bandwidth has the ISE close to the density estimators with plug-in bandwidth in Table 2.5, showing the reliability of the for selecting the bandwidth.

A simulation was done to compare density estimators with both plug-in and optimal bandwidth when $J = 3$ replicates are present for each observation. Table 2.8 shows that with plug-in bandwidth, the two phase function density estimators have lower median ISE than both the D&M estimators computed using true variances and the D&M estimators computed using the estimated variances for measurement errors. Also, the mean optimal WEPF_{opt} density estimator has lower median ISE than the EPF density estimator in all but five instances. Table 2.9 shows that with optimal bandwidth, the two phase function density estimators have similar median ISE to the D&M estimator computed using the true variance of measurement errors. This median ISE is much lower median ISE than the D&M estimator computed using the estimated variance of measurement errors. In other words, the simulation results with $J = 3$ replicate reinforces the fact that the phase function density estimators using mean-optimal has performance comparable to the D&M estimator, while making fewer assumptions about the distribution of measurement error.

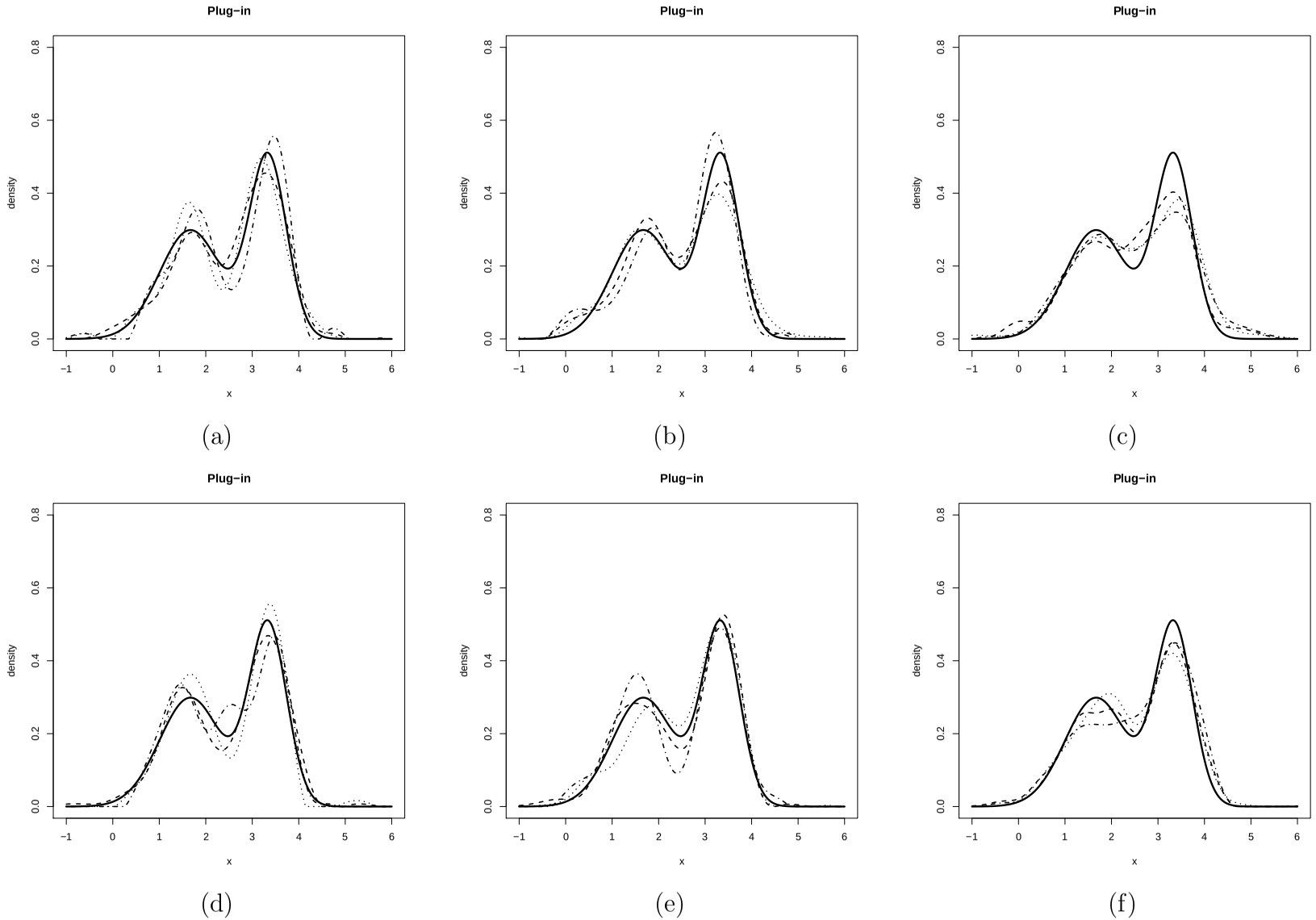


Figure 2.5: Curves Q_1 (- - - - -), Q_2 (.....), Q_3 (-·-·-·-), and true curve (——) for $X \sim$ Mixture 2, $n = 500$, with $J = 2$ replicates per observation, when the errors are Normal (a)-(c), and Laplace (d)-(f), with case 1 of measurement error variances. For (a),(d): EPF estimator; (b),(e): WEPF_{opt} estimator; (c),(f): D&M estimator with estimated variances. All estimators are computed using plug-in bandwidth.

True X	Error type	Error case	EPF	WEPF$_{opt}$	D&M
Scaled $\chi^2(3)$	Normal	1	0.219	0.197	0.164
			[0.189, 0.279]	[0.158, 0.240]	[0.134, 0.193]
		2	0.474	0.466	0.421
			[0.401, 0.549]	[0.384, 0.536]	[0.33, 0.521]
		3	0.407	0.363	0.278
			[0.321, 0.461]	[0.294, 0.417]	[0.228, 0.347]
	Laplace	1	0.186	0.171	0.144
			[0.167, 0.241]	[0.147, 0.202]	[0.120, 0.177]
		2	0.311	0.292	0.272
			[0.243, 0.367]	[0.236, 0.355]	[0.2, 0.332]
		3	0.268	0.26	0.231
			[0.221, 0.352]	[0.205, 0.339]	[0.17, 0.289]
Mixture 1	Normal	1	0.180	0.138	0.100
			[0.128, 0.248]	[0.085, 0.192]	[0.067, 0.137]
		2	0.588	0.519	0.454
			[0.452, 0.658]	[0.433, 0.667]	[0.334, 0.602]
		3	0.43	0.385	0.239
			[0.319, 0.566]	[0.271, 0.503]	[0.153, 0.349]
	Laplace	1	0.142	0.107	0.090
			[0.078, 0.201]	[0.060, 0.160]	[0.062, 0.111]
		2	0.259	0.254	0.18
			[0.19, 0.382]	[0.182, 0.351]	[0.129, 0.281]
		3	0.254	0.232	0.168
			[0.178, 0.339]	[0.173, 0.293]	[0.108, 0.217]
Mixture 2	Normal	1	0.085	0.066	0.068
			[0.059, 0.131]	[0.040, 0.101]	[0.043, 0.094]
		2	0.277	0.284	0.254
			[0.205, 0.348]	[0.204, 0.356]	[0.157, 0.326]
		3	0.218	0.193	0.129
			[0.144, 0.272]	[0.129, 0.252]	[0.089, 0.187]
	Laplace	1	0.056	0.045	0.060
			[0.039, 0.096]	[0.029, 0.077]	[0.035, 0.094]
		2	0.146	0.143	0.135
			[0.097, 0.21]	[0.09, 0.216]	[0.085, 0.202]
		3	0.133	0.117	0.125
			[0.091, 0.176]	[0.075, 0.161]	[0.081, 0.154]

Table 2.6: The median and [Q_1, Q_3] of $10 \times$ ISE of the density estimators with optimal bandwidth based on 500 simulations. Each simulation has sample size $n = 500$ with no replicate (measurement error variances are known).

True X	Error type	Error case	EPF	WEPF$_{opt}$	D&M (variance estimated)	D&M (variance known)
Scaled $\chi^2(3)$	Normal	1	0.198	0.192	0.209	0.140
			[0.159, 0.259]	[0.156, 0.235]	[0.169, 0.245]	[0.121, 0.172]
		2	0.309	0.313	0.382	0.306
			[0.248, 0.363]	[0.258, 0.366]	[0.316, 0.461]	[0.237, 0.362]
		3	0.277	0.276	0.329	0.216
			[0.234, 0.313]	[0.236, 0.314]	[0.275, 0.376]	[0.169, 0.262]
	Laplace	1	0.176	0.165	0.154	0.121
			[0.142, 0.207]	[0.137, 0.204]	[0.127, 0.190]	[0.104, 0.154]
		2	0.272	0.266	0.302	0.266
			[0.221, 0.331]	[0.222, 0.323]	[0.257, 0.339]	[0.203, 0.321]
		3	0.219	0.218	0.243	0.197
			[0.178, 0.26]	[0.176, 0.255]	[0.205, 0.288]	[0.149, 0.257]
Mixture 1	Normal	1	0.128	0.120	0.159	0.075
			[0.086, 0.182]	[0.077, 0.166]	[0.081, 0.155]	[0.050, 0.097]
		2	0.305	0.296	0.414	0.271
			[0.206, 0.382]	[0.217, 0.393]	[0.356, 0.513]	[0.184, 0.38]
		3	0.243	0.238	0.32	0.158
			[0.175, 0.333]	[0.182, 0.332]	[0.245, 0.396]	[0.096, 0.219]
	Laplace	1	0.102	0.105	0.119	0.067
			[0.066, 0.156]	[0.073, 0.157]	[0.081, 0.155]	[0.050, 0.097]
		2	0.212	0.205	0.251	0.178
			[0.145, 0.269]	[0.14, 0.26]	[0.199, 0.306]	[0.115, 0.246]
		3	0.19	0.173	0.217	0.135
			[0.13, 0.279]	[0.119, 0.241]	[0.17, 0.285]	[0.087, 0.213]
Mixture 2	Normal	1	0.063	0.062	0.104	0.059
			[0.043, 0.097]	[0.036, 0.093]	[0.077, 0.117]	[0.043, 0.074]
		2	0.164	0.162	0.206	0.155
			[0.096, 0.226]	[0.107, 0.224]	[0.165, 0.266]	[0.103, 0.202]
		3	0.125	0.11	0.166	0.097
			[0.089, 0.18]	[0.081, 0.179]	[0.137, 0.215]	[0.066, 0.133]
	Laplace	1	0.051	0.050	0.074	0.049
			[0.035, 0.083]	[0.031, 0.080]	[0.051, 0.097]	[0.029, 0.073]
		2	0.13	0.114	0.153	0.131
			[0.08, 0.171]	[0.069, 0.151]	[0.119, 0.185]	[0.091, 0.164]
		3	0.104	0.095	0.136	0.108
			[0.063, 0.154]	[0.066, 0.146]	[0.113, 0.184]	[0.07, 0.149]

Table 2.7: The median and [Q_1, Q_3] of $10 \times$ ISE of the density estimators with optimal bandwidth based on 500 simulations. Each simulation has sample size $n = 500$ and $J = 2$ replicates per observation.

True X	Error type	Error case	EPF	WEPF$_{opt}$	D&M (variance estimated)	D&M (variance known)
Scaled $\chi^2(3)$	Normal	1	0.231	0.227	0.316	0.251
			[0.194, 0.296]	[0.188, 0.279]	[0.281, 0.362]	[0.212, 0.291]
		2	0.246	0.249	0.324	0.272
			[0.199, 0.298]	[0.206, 0.303]	[0.283, 0.371]	[0.226, 0.314]
		3	0.22	0.215	0.294	0.216
			[0.182, 0.26]	[0.183, 0.261]	[0.249, 0.325]	[0.185, 0.252]
	Laplace	1	0.211	0.208	0.268	0.227
			[0.177, 0.265]	[0.177, 0.253]	[0.228, 0.308]	[0.193, 0.265]
		2	0.232	0.229	0.293	0.251
			[0.189, 0.285]	[0.189, 0.283]	[0.243, 0.32]	[0.204, 0.299]
		3	0.207	0.208	0.256	0.21
			[0.184, 0.243]	[0.175, 0.235]	[0.228, 0.299]	[0.186, 0.257]
Mixture 1	Normal	1	0.189	0.182	0.29	0.18
			[0.148, 0.252]	[0.138, 0.253]	[0.238, 0.347]	[0.134, 0.253]
		2	0.202	0.197	0.321	0.218
			[0.139, 0.262]	[0.142, 0.265]	[0.265, 0.364]	[0.159, 0.271]
		3	0.202	0.192	0.289	0.175
			[0.136, 0.267]	[0.122, 0.267]	[0.225, 0.351]	[0.121, 0.246]
	Laplace	1	0.151	0.151	0.228	0.161
			[0.102, 0.21]	[0.1, 0.207]	[0.188, 0.277]	[0.119, 0.214]
		2	0.163	0.158	0.238	0.179
			[0.114, 0.227]	[0.11, 0.234]	[0.187, 0.31]	[0.127, 0.262]
		3	0.141	0.129	0.193	0.133
			[0.094, 0.192]	[0.085, 0.194]	[0.152, 0.258]	[0.088, 0.177]
Mixture 2	Normal	1	0.095	0.097	0.16	0.106
			[0.062, 0.142]	[0.064, 0.137]	[0.12, 0.195]	[0.074, 0.135]
		2	0.127	0.125	0.187	0.136
			[0.09, 0.164]	[0.085, 0.164]	[0.147, 0.214]	[0.1, 0.176]
		3	0.094	0.092	0.149	0.091
			[0.061, 0.145]	[0.062, 0.138]	[0.107, 0.185]	[0.057, 0.132]
	Laplace	1	0.095	0.084	0.128	0.101
			[0.069, 0.125]	[0.052, 0.112]	[0.096, 0.152]	[0.069, 0.129]
		2	0.097	0.098	0.144	0.113
			[0.071, 0.141]	[0.07, 0.138]	[0.117, 0.171]	[0.082, 0.159]
		3	0.097	0.084	0.128	0.097
			[0.057, 0.128]	[0.057, 0.118]	[0.1, 0.171]	[0.073, 0.136]

Table 2.8: The median and [Q_1, Q_3] of $10 \times$ ISE of the density estimators with plug-in bandwidth based on 500 simulations. Each simulation has sample size $n = 500$ and $J = 3$ replicates per observation.

True X	Error type	Error case	EPF	WEPF$_{opt}$	D&M (variance estimated)	D&M (variance known)
Scaled $\chi^2(3)$	Normal	1	0.225	0.224	0.276	0.219
			[0.192, 0.294]	[0.187, 0.276]	[0.233, 0.321]	[0.186, 0.257]
		2	0.239	0.24	0.289	0.24
			[0.197, 0.282]	[0.203, 0.289]	[0.231, 0.348]	[0.195, 0.298]
		3	0.215	0.213	0.239	0.187
			[0.182, 0.259]	[0.183, 0.26]	[0.209, 0.288]	[0.144, 0.227]
	Laplace	1	0.209	0.205	0.22	0.201
			[0.177, 0.256]	[0.176, 0.246]	[0.188, 0.269]	[0.166, 0.239]
		2	0.228	0.227	0.243	0.216
			[0.189, 0.278]	[0.187, 0.274]	[0.198, 0.29]	[0.183, 0.273]
		3	0.204	0.205	0.216	0.194
			[0.183, 0.238]	[0.175, 0.228]	[0.188, 0.253]	[0.162, 0.231]
Mixture 1	Normal	1	0.186	0.179	0.249	0.159
			[0.145, 0.246]	[0.132, 0.244]	[0.195, 0.304]	[0.101, 0.237]
		2	0.202	0.197	0.321	0.218
			[0.138, 0.251]	[0.14, 0.264]	[0.219, 0.336]	[0.126, 0.25]
		3	0.199	0.189	0.239	0.138
			[0.136, 0.264]	[0.122, 0.264]	[0.182, 0.305]	[0.091, 0.218]
	Laplace	1	0.147	0.149	0.182	0.137
			[0.102, 0.21]	[0.1, 0.203]	[0.147, 0.226]	[0.092, 0.184]
		2	0.163	0.156	0.195	0.154
			[0.114, 0.227]	[0.11, 0.234]	[0.153, 0.268]	[0.111, 0.236]
		3	0.141	0.129	0.193	0.133
			[0.092, 0.192]	[0.085, 0.182]	[0.119, 0.207]	[0.069, 0.16]
Mixture 2	Normal	1	0.083	0.09	0.134	0.089
			[0.06, 0.123]	[0.063, 0.122]	[0.101, 0.164]	[0.065, 0.121]
		2	0.12	0.115	0.162	0.119
			[0.087, 0.152]	[0.081, 0.152]	[0.128, 0.19]	[0.088, 0.155]
		3	0.082	0.083	0.122	0.078
			[0.053, 0.126]	[0.049, 0.124]	[0.085, 0.156]	[0.044, 0.109]
	Laplace	1	0.083	0.075	0.11	0.09
			[0.059, 0.111]	[0.046, 0.097]	[0.08, 0.134]	[0.063, 0.116]
		2	0.084	0.084	0.119	0.097
			[0.059, 0.12]	[0.057, 0.118]	[0.095, 0.152]	[0.071, 0.139]
		3	0.097	0.084	0.128	0.097
			[0.054, 0.111]	[0.05, 0.102]	[0.088, 0.139]	[0.062, 0.114]

Table 2.9: The median and [Q_1, Q_3] of $10 \times$ ISE of the density estimators with optimal bandwidth based on 500 simulations. Each simulation has sample size $n = 500$ and $J = 3$ replicates per observation.

Chapter 3
Linear Errors-in-Variables Estimation with Unknown Error Distribution

3.1. Overview

Parameter estimation in linear errors-in-variables models typically requires that the measurement error distribution be known (or estimable from replicate data). A generalized method of moments approach can be used to estimate model parameters in the absence of knowledge of the error distributions, but requires the existence of a large number of model moments. In this paper, parameter estimation based on the phase function, a normalized version of the characteristic function, is considered. This approach requires the model covariates to have asymmetric distributions, while the error distributions are symmetric. Parameter estimation is then based on minimizing a distance function between the empirical phase functions of the noisy covariates and the outcome variable. No knowledge of the measurement error distribution is required to calculate this estimator. Both the asymptotic and finite sample properties of the estimator are considered. The connection between the phase function approach and method of moments is also discussed. The estimation of standard errors is also considered and a modified bootstrap algorithm is proposed for fast computation. The newly proposed estimator is competitive when compared to generalized method of moments, even while making fewer model assumptions on the measurement error. Finally, the proposed method is applied to several real datasets concerning the measurement of air pollution.

3.2. Introduction

Errors-in-variables models arise when some covariates cannot be measured accurately. Sources of measurement error include the instruments used to measure the variables

of interest and the inadequacy of measurements taken over the short term being used as proxies for long-term variables. In the classic measurement error framework, this results in observed covariates having larger variance than the true predictors. Let $X = (X^{(1)}, \ldots, X^{(p)})^\top \in \mathbb{R}^p$ denote the true model covariates and let $Y \in \mathbb{R}$ denote the outcome of interest. For $\beta_1 \in \mathbb{R}^p$, the relationship between X and Y is assumed to be $Y = \beta_0 + X^\top \beta_1 + \varepsilon$ with intercept $\beta_0 \in \mathbb{R}$ and error $\varepsilon \in \mathbb{R}$. In an errors-in-variables model, X is not directly observed. Rather, $W = (W^{(1)}, \ldots, W^{(p)})^\top \in \mathbb{R}^p$ is observed with $W = X + U$ denoting the covariates contaminated by additive measurement error, and $U \in \mathbb{R}^p$ denoting the measurement error. This model represents the classic formulation of the errors-in-variables model and the estimation of $\beta = (\beta_0, \beta_1^\top)^\top$ is of interest.

Above, the model error ε is assumed to be symmetric about 0 with scale parameter σ^2 and the measurement error U is assumed to be symmetric about $0 \in \mathbb{R}^p$ with scale matrix Σ_u. Generally, σ^2 and Σ_u represent, respectively, the variance of ε and covariance matrix of U when these quantities are well-defined. The covariates X, measurement error U and model error ε are furthermore assumed mutually independent. Given a sample $(W_1, Y_1), \ldots, (W_n, Y_n)$, it is well known that regression of the Y_i on the W_i using traditional methods such as ordinary least squares leads to an inconsistent and biased estimate of β, see Carroll et al. (2006). Hence, adjusting for the presence of measurement error is important for accurately describing the relationship between the true covariates and the outcome of interest.

This chapter proposes a method of estimation that is fully nonparametric, in that implementation does not require parametric specifications of any model components, nor does it require the existence of model moments. Furthermore, the method does not require that the measurement error variance be known, if it exists, and replication data is not needed. The estimator makes use of the empirical phase function, a normalized version of the empirical characteristic function. The empirical phase function was considered in the context of density deconvolution by Delaigle and Hall (2016) and Nghiem and Potgieter (2018). The method has two assumptions: the measurement error U is symmetric around 0 with strictly positive characteristic function, and the distribution of X is *a*symmetric.

These assumptions are fundamental for the identifiability of the phase function of X, which forms the basis of the estimation procedure. The assumptions are discussed in greater detail in Section 3.3; see also Delaigle and Hall (2016) for an in-depth discussion.

The remainder of this chapter is organized as below. In Section 2, we introduce the phase function-based estimator, develop its asymptotic properties, and establishes a connection to the method of moments approach. Section 3 considers some computational aspects relating to the estimator, including estimating standard errors in practice. Section 4 presents a simulation study to illustrate the performance of the phase function estimator and compare it with existing methods. Section 5 applies the phase function estimator to a real dataset, and Section 6 contains some concluding remarks.

3.3. Phase Function Minimum Distance Estimation

3.3.1. Phase Function-Based Estimation

Consider the simple linear errors-in-variables model with observed sample (W_i, Y_i), $i = 1, \ldots, n$ where

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i \quad \text{and} \quad W_i = X_i + U_i. \tag{3.1}$$

Here, the $X_i \in \mathbb{R}$ are independent and identically distributed with asymmetric density function f_X, the $U_i \in \mathbb{R}$ and $\varepsilon_i \in \mathbb{R}$ are independent and identically distributed with respective density functions f_U and f_ε, both symmetric about 0 and having strictly positive characteristic functions. Furthermore,the X_i, U_i and ε_i are assumed mutually independent. It should be noted that the method developed here can also be used in the more general setting where each error term U_i and ε_i has a unique density function, say $f_{U,i}$ and $f_{\varepsilon,i}$, as long as these are all independent, symmetric about 0, and have strictly positive characteristic functions. However, for simplicity of exposition the scenario with common error densities f_U and f_ε is presented. As to the assumed positivity of the characteristic functions, we note that many commonly used continuous distributions in the application of regression and measurement error satisfy this condition. This includes the Gaussian, Laplace, and Student's t distributions. In general, the only symmetric distributions ex-

cluded are those defined on bounded intervals, such as the uniform. In the context of density deconvolution, Delaigle and Hall (2016) assumed that the random variable X does not have a symmetric component, i.e. there is *no* symmetric random variable S for which X can be decomposed as $X = X_0 + S$ for arbitrary random variable X_0. In the present setting, this strict assumption is not required. More specifically, we only require that the covariate X not be symmetric.

Now, let $\phi_X(t) = \mathrm{E}\left[\exp\left(itX\right)\right]$ denote the characteristic function of a random variable X. The phase function of X is then defined as the normalized characteristic function,

$$\rho_X(t) = \frac{\phi_X(t)}{|\phi_X(t)|}, \tag{3.2}$$

where $|z| = (z\bar{z})^{1/2}$ is the complex norm with $\bar{z}$ denoting the complex conjugate of z. We now present our first result that establishes a relationship between the phase functions of W and Y.

Lemma 3.1. *Consider univariate random variables $W = X + U$ and $Y = \beta_0 + \beta_1 X + \varepsilon$. Assume that X asymmetric with phase function $\rho_X(t)$, and that U and ε are symmetric about 0 with strictly positive characteristic functions. The phase function for Y is then given by*

$$\rho_Y(t) = \exp\left(it\beta_0\right)\rho_X(\beta_1 t) = \exp\left(it\beta_0\right)\rho_W(\beta_1 t). \tag{3.3}$$

Hence, the phase function of Y can be fully specified in terms of $\rho_W(t)$, the phase function of W, and parameters (β_0, β_1).

Proof. By independence of X and U, the characteristic function of W is given by

$$\phi_W(t) = \mathbb{E}(e^{itW}) = \mathbb{E}(e^{it(X+U)}) = \mathbb{E}(e^{itX})\mathbb{E}(e^{itU}) = \phi_X(t)\phi_U(t).$$

By assumption, the characteristic function of U satisfies $\phi_U(t) = |\phi_U(t)|$ for t. Thus, the phase function for W is

$$\rho_W(t) = \frac{\phi_W(t)}{|\phi_W(t)|} = \frac{\phi_X(t)\phi_U(t)}{|\phi_X(t)||\phi_U(t)|} = \frac{\phi_X(t)}{|\phi_X(t)|} = \rho_X(t).$$

Subsequently, the random variables W and X have the same phase function. $\square$

Empirical estimates of the phase functions of W and Y can be obtained from a random sample $(W_j, Y_j),\ j = 1, \ldots, n$. Define

$$\hat{\rho}_W(t) = \frac{\hat{\phi}_W(t)}{|\hat{\phi}_W(t)|} = \frac{\sum_{j=1}^{n} \exp(itW_j)}{\left[\sum_{j=1}^{n}\sum_{k=1}^{n} \exp\left\{it\left(W_j - W_k\right)\right\}\right]^{1/2}},$$

with a similar definition holding for $\hat{\rho}_Y(t)$. The empirical phase functions can now be used to construct minimum distance estimators of the model parameters (β_0, β_1). Define statistic

$$D(b_0, b_1) = \int_{-\infty}^{\infty} \left|\hat{\rho}_Y(t) - \exp\left(itb_0\right)\hat{\rho}_W(b_1 t)\right|^2 w(t)dt, \tag{3.4}$$

where the weight function $w(t)$ is chosen to ensure that the integral is well-defined. The estimator $(\hat{\beta}_0, \hat{\beta}_1)$ is then computed as the global minimizer of the function $D(b_0, b_1)$.

The above idea can be easily extended to the case of multivariate regression with both error-prone and error-free covariates. Consider the model $Y = \beta_0 + X^\top\beta_1 + Z^\top\beta_2 + \varepsilon$ where $X, \beta_1 \in \mathbb{R}^{p_1}$ and $Z, \beta_2 \in \mathbb{R}^{p_2}$. Here, Z represents the covariates measured without error. As before, let $W = X + U$ denote the contaminated version of X where U is p_1-dimensional symmetric measurement error. Let $V = \beta_0 + X^\top\beta_1 + Z^\top\beta_2$ so that $Y = V + \varepsilon$. It then follows that $\rho_Y(t) = \rho_V(t)$.

Similarly, consider the linear combination in terms of the contaminated W, say

$$\widetilde{V} = \beta_0 + W^\top\beta_1 + Z^\top\beta_2 = V + U^\top\beta_1 = V + \widetilde{U}$$

with $\widetilde{U} = U^\top\beta_1 \in \mathbb{R}$ having distribution symmetric about zero with strictly positive characteristic function. It then also follows that $\rho_{\widetilde{V}}(t) = \rho_V(t)$. Hence, the variables Y, V and $\widetilde{V}$ all have the same phase function. To estimate $\beta = (\beta_0, \beta_1^\top, \beta_2^\top)^\top$, it is possible to construct a distance metric equivalent to (3.4),

$$D(b_0, b_1, b_2) = \int_{-\infty}^{\infty} \left|\hat{\rho}_Y(t) - \exp\left(itb_0\right)\hat{\rho}_{\widetilde{V}}(t|b_1, b_2)\right|^2 w(t)dt \tag{3.5}$$

where, given n random observations (W_j, Z_j, Y_j), the empirical phase function corresponding to $\widetilde{V}$ is

$$\hat{\rho}_{\widetilde{V}}(t|\beta_1, \beta_2) = \frac{\sum_{j=1}^{n} \exp\left\{it\left(W_j^\top \beta_1 + Z_j^\top \beta_2\right)\right\}}{\left(\sum_{j=1}^{n}\sum_{k=1}^{n} \exp\left[it\left\{(W_j - W_k)^\top \beta_1 + (Z_j - Z_k)^\top \beta_2\right\}\right]\right)^{1/2}}. \tag{3.6}$$

Note that the statistic (3.5) does not treat the variables measured with and without error any differently. As such, the phase function approach could be implemented without knowledge of which variables are subject to measurement error. The estimate of $\beta = (\beta_0, \beta_1^\top, \beta_2^\top)^\top$ is found by minimizing $D(b_0, b_1, b_2)$.

3.3.2. Asymptotic Properties of Phase Function Estimators

In this section, we verify that the estimators obtained by minimizing statistic D in (3.4) satisfy the conditions required of M-estimators, and are therefore asymptotically normal. To this end, we first establish the almost sure convergence of D to an appropriate limit. Note that, while the asymptotic properties of the phase function-based estimator are considered in the context of a simple linear errors-in-variables model, the results easily extend to the multivariate case.

Lemma 3.2. *Assume that independent pairs $(W_1, Y_1), \ldots, (W_n, Y_n)$ are observed with $W_i = X_i + U_i$ and $Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$, with the distribution of X_i asymmetric, and with U_i and ε_i having distributions symmetric about 0 and with strictly positive characteristic functions. Furthermore, let $w(t)$ be a non-negative weight function with bounded support, taken without loss of generality to be $[-c, c]$.*

For this choice of weight function, the statistic $D(b_0, b_1)$ defined in (3.4) converges almost surely to a limit $D_{\text{true}}(b_0, b_1)$ with

$$D_{\text{true}}(b_0, b_1) = \int_{-\infty}^{\infty} \left|\rho_Y(t) - \exp(itb_0)\rho_W(b_1 t)\right|^2 w(t)dt.$$

The limit has unique global minimum $D_{\text{true}}(\beta_0, \beta_1) = 0$.

The proof of this lemma follows upon noting the empirical characteristic functions

$\hat{\phi}_W(t)$ is an unbiased estimator of the true characteristic function $\phi_W(t)$ and converges almost surely to $\phi_W(t)$ on any bounded interval $[-c, c]$, see Theorem 2.1 of Feuerverger et al. (1977). Applying the continuous mapping theorem, the empirical phase function $\hat{\rho}_W(t)$ also converges almost surely to the true phase function $\rho_W(t)$ on $[-c, c]$, and is an asymptotically unbiased estimator thereof. The convergence of D to D_{true} follows from this. Next, noting that a phase function is uniquely identified by the asymmetric part of the corresponding distribution, the function D_{true} has a global minimum of 0 at the true parameter values (β_0, β_1).

Theorem 3.1. *Assume that conditions (i) and (ii) from Lemma 1 hold. Let* $\hat{\beta} = (\hat{\beta}_0, \hat{\beta}_1)^\top$ *denote the minimizer of* D *in (3.4). This estimator is consistent for the true* $\beta = (\beta_0, \beta_1)^\top$*, and is asymptotically normal,*

$$\sqrt{n}\left(\hat{\beta} - \beta\right) \to N\left(0, B^{-1}AB^{-1}\right) \tag{3.7}$$

where

$$A = E\left(\lambda\lambda^\top\right) \text{ and } B = E\left(\frac{\partial\lambda}{\partial\beta^\top}\right) \tag{3.8}$$

with $\lambda = \dfrac{\partial D}{\partial\beta}$.

The consistency of $\hat{\beta}$ follows from Lemma 1 above and Theorem 5.7 in van der Vaart (2000). Having established consistency, and noting that D has infinitely many bounded and continuous derivatives, asymptotic normality follows from Theorem 5.21 in van der Vaart (2000).

3.3.3. Connection to Method of Moments Estimation

Delaigle and Hall (2016) show that for any random variable X with infinite number of moments, the phase function of X can be expressed as

$$\rho_X(t) = \exp\left\{\sum_{j=1}^{\infty} \frac{(-1)^{j+1} t^{2j-1} \kappa_{2j-1}^X}{(2j-1)!}\right\},$$

where κ_j^X denotes the jth cumulant of X. In other words, if the infinite series above

converges, the phase function is determined uniquely by the odd-order cumulants of X. In this context, consider the model (3.1). If X, U, and ε have an infinite number of finite moments, the same holds true for W and Y. Specifically, for (W, Y) following the linear errors-in-variables model, it follows that

$$\exp\left\{i\sum_{j=1}^{\infty}\frac{(-1)^{j+1}t^{2j-1}\kappa_{2j-1}^{Y}}{(2j-1)!}\right\} = \exp\left[i\left\{t\beta_0 + \sum_{j=1}^{\infty}\frac{(-1)^{j+1}(\beta_1 t)^{2j-1}\kappa_{2j-1}^{W}}{(2j-1)!}\right\}\right]. \quad (3.9)$$

One can use (3.9) and match the coefficients of t^{2j-1} to determine the relationship between the jth odd cumulants of W and Y. For example, considering the coefficients of t and t^3 gives $\kappa_1^Y = \beta_0 + \beta_1\kappa_1^W$ and $\kappa_3^Y = \beta_1^3\kappa_3^W$.

Now, using properties of the complex norm, it follows that

$$\frac{1}{4}\left|\rho_Y(t) - \exp(it\beta_0)\rho_W(\beta_1 t)\right|^2 = \sin^2\left\{\sum_{j=1}^{\infty}\frac{(-1)^{j+1}t^{2j-1}(\kappa_{2j-1}^{Y} - \beta_1^{2j-1}\kappa_{2j-1}^{W})}{(2j-1)!} - t\beta_0\right\}.$$

When inference is based on the sample phase functions, the population cumulants above are replaced by their sample counterparts, and minimizing (3.4) is equivalent to choosing the parameters β_0 and β_1 such that a function of the difference of all odd cumulants is minimized. As such, when the underlying distributions have an infinite number of moments, the phase function approach can be thought of as a method of moments-type approach that makes use of all odd cumulants of the variables of interest.

3.4. Computational Considerations

3.4.1. Computing the Estimators

Direct minimization of statistics (3.4) and (3.5) is generally computationally expensive. In this section, a computational method is proposed that leads to faster calculation of the estimators. The idea is presented for the univariate errors-in-variables model, but can easily be extended to the multivariate model setting.

Lemma 3.3. *Consider the statistics (3.4) with weight function*

$$w(t) = K_{t^*}(t)\left[\sum_{j=1}^{n}\cos(tY_j)\sum_{j=1}^{n}\cos\left\{t\left(b_0 + b_1 W_j\right)\right\}\right]^2$$

where $K_{t^*}(t) = K(t/t^*)$ and $K(t)$ is a non-negative kernel function with bounded support on some interval $[-1, 1]$. Minimization of (3.4) is then equivalent to minimization of

$$D(b_0, b_1) = \int_0^{t^*}\left[\sum_{i=1}^{n}\sum_{j=1}^{n}\sin\left\{t(Y_i - b_0 - b_1 W_j)\right\}\right]^2 K_{t^*}(t)dt \tag{3.10}$$

Proof. For any complex number z, let $R(z) = \mathrm{Im}(z)/\mathrm{Re}(z)$ denote the ratio of the imaginary and real parts of z. Now, consider the relationship that exists between the phase functions of Y and W as given in (3.3), and recall that any phase function has norm equal to 1 for all t. It follows that (3.3) is equivalent to

$$R[\rho_Y(t)] = R[\exp(it\beta_0)]R[\rho_W(\beta_1 t)].$$

Furthermore, as the phase function is a scaled version of the characteristic function, $R[\rho_Y(t)] = R[\phi_Y(t)]$ and (3.4.1) is equivalent to

$$R[\phi_Y(t)] = R[\exp(it\beta_0)]R[\phi_W(\beta_1 t)].$$

By Euler's formula, this can be written as

$$\frac{E\left[\sin\left(tY\right)\right]}{E\left[\cos\left(tY\right)\right]} = \frac{E\left[\sin\left(t\left(\beta_0 + \beta_1 W\right)\right)\right]}{E\left[\cos\left(t\left(\beta_0 + \beta_1 W\right)\right)\right]}.$$

Therefore, minimizing (3.4) is equivalent to minimizing

$$D(b_0, b_1) = \int_{-\infty}^{\infty}\left(\frac{\sum_{j=1}^{n}\sin(tY_j)}{\sum_{j=1}^{n}\cos(tY_j)} - \frac{\sum_{j=1}^{n}\sin\left(t\left(b_0 + b_1 W_j\right)\right)}{\sum_{j=1}^{n}\cos\left(t\left(b_0 + b_1 W_j\right)\right)}\right)^2 w(t)dt.$$

If choosing the weight function as stated in the lemma, the integrand is an even function with respect to t. Then the result follows from simplifying the resulting trigonometric

products. □

Formula (3.10) has computational complexity $\mathcal{O}(n^2)$. However, by some algebra it can be re-expressed as

$$D(b_0, b_1) = \int_0^{t^*} \left[S_y \sum_{j=1}^{n} \cos\{t(b_0 + b_1 W_j)\} - C_y \sum_{j=1}^{n} \sin\{t(b_0 + b_1 W_j)\} \right]^2 K_{t^*}(t)dt, \tag{3.11}$$

with $C_y = \sum_{j=1}^{n} \cos(tY_j)$ and $S_y = \sum_{j=1}^{n} \sin(tY_j)$. Evaluating (3.11) has computational complexity $\mathcal{O}(n)$. The particular choice of weight function avoids instabilities that can occur in (3.4) as a result of dividing by numbers close to 0. With regards to choosing an appropriate constant t^*, we follow the suggestion in Delaigle and Hall (2016) who let t^* be the smallest $t > 0$ such that $|\hat{\phi}_Y(t)| \leq n^{-1/4}$.

When considering simplification of statistic (3.10), It is also possible to eliminate the integral in the equation. To this end, let $\phi_{K,h}(\alpha) = \int_{-h}^{h} \cos(\alpha t) K_h(t)dt$. It then follows that

$$D(b_0, b_1) \propto \sum_{i,j,k,l} \Big[\phi_{K,t^*}\{Y_i - Y_k - b_1(W_j - W_l)\} - \phi_{K,t^*}\{Y_i + Y_k + 2b_0 + b_1(W_j + W_l)\} \Big]. \tag{3.12}$$

Note that while expression (3.12) eliminates the need to numerically evaluate an integral as in (3.11), we generally found that the form in (3.11) was much faster to compute than the expression involving the quadruple sum in (3.12).

Now, considering again the recommended computational form in (3.11). By an application of the Leibniz rule, the first partial derivatives of D with respect to b_0 and b_1, denoted here $\lambda(b_0, b_1) = \{\lambda_0(b_0, b_1), \lambda_1(b_0, b_1)\}^\top$, are

$$\lambda_0 = \frac{\partial D}{\partial b_0} = -2 \int_0^{t^*} \left[\sum_{i,j} \sin\{t(Y_i - b_0 - b_1 W_j)\} \right] \left[\sum_{i,j} \cos\{t(Y_i - b_0 - b_1 W_j)\} \right] tK_{t^*}(t)dt \tag{3.13}$$

$$\lambda_1 = \frac{\partial D}{\partial b_1} = -2 \int_0^{t^*} \left[\sum_{i,j} \sin\{t(Y_i - b_0 - b_1 W_j)\} \right] \left[\sum_{i,j} W_j \cos\{t(Y_i - b_0 - b_1 W_j)\} \right] tK_{t^*}(t)dt. \tag{3.14}$$

The expressions for λ_0 and λ_1 can be used as estimating equations to solve for $(\hat{\beta}_0, \hat{\beta}_1)$.

These expressions will also be useful in the next section when considering the estimation of standard errors for the estimators.

3.4.2. Standard Error Estimation

We now consider estimation of the covariance matrix of $\hat{\beta}$. Using the asymptotic variance as given in Theorem 3.1 would be reasonable, but direct evaluation of matrices A and B in (3.8) is not possible as this requires knowledge of the distributions of X, U and ε. If these distributions were known, a likelihood approach could be used for parameter estimation rather than the proposed phase function approach.

The bootstrap is a popular method for estimating the covariance matrix of estimated parameters in a nonparametric setting such as this is the bootstrap. This requires repeated calculation of bootstrap estimators $\hat{\beta}_b^*$ based on bootstrap samples $(W_{b,i}^*, Y_{b,i}^*)$, $i = 1, \ldots, n$ for $b = 1, \ldots, B$ drawn with replacement from the observed sample. The estimated covariance matrix is then taken to be the sample covariance matrix of the bootstrap replicates $\hat{\beta}_b^*$. The procedure can be slow due to the repeated evaluation of a computationally expensive loss function for each bootstrap sample. Implementation is described in Algorithm 1.

We propose here a modified bootstrap algorithm for estimating the standard errors that combines bootstrap methodology with approximation of matrices A and B in (3.8). To this end, note that matrix A is the covariance matrix of λ, the first partial derivatives of (3.10) given by (3.13) and (3.14) in the univariate setting. As such, bootstrap methodology can be used to estimate matrix A, while B can estimated by evaluating the second derivatives of D at the parameter estimates $\hat{\beta}_0$ and $\hat{\beta}_1$. Expressions for these second derivatives are unwieldy, but are easily evaluated numerically; see Section A.3 in the supplemental material. We refer to this approach as the *plug-in bootstrap* approach and outline implementation in Algorithm 2. Note that the plug-in covariance matrix is orders of magnitude faster to compute that the boostrap estimator as it does not require repeated minimization of a statistic involving numerical integration.

- For $b = 1, \ldots, B$
  - Sample n pairs with replacement from the observed data to obtain bootstrap sample $(W^*_{i,b}, Y^*_{i,b})$, $i = 1, \ldots, n$.
  - Calculate the bootstrap estimators $\hat{\beta}^*_b$ by minimizing (3.10) using the bootstrap sample.
- Calculate the empirical covariance matrix of the bootstrap statistics

$$\hat{\mathbf{\Sigma}}_{\text{boot}} = \frac{1}{B} \sum_{b=1}^{B} \left(\hat{\beta}^*_b - \bar{\beta}^* \right) \left(\hat{\beta}^*_b - \bar{\beta}^* \right)^\top$$

  where $\bar{\beta}^* = B^{-1} \sum_b \hat{\beta}^*_b$ is the mean of the bootstrap replicates.

Algorithm 1: Full Bootstrap Algorithm

- For $b = 1, \ldots, B$
  - Sample n pairs with replacement from the observed data to obtain bootstrap sample $(W^*_{i,b}, Y^*_{i,b})$, $i = 1, \ldots, n$.
  - Calculate $\lambda^*_b = \lambda^*_b(\hat{\beta}_0, \hat{\beta}_1)$ as in (3.13) and (3.14) using the bth bootstrap sample.
- Calculate

$$\hat{A}_{\text{boot}} = \frac{1}{B} \sum_{b=1}^{B} \lambda^*_b \lambda^{*\top}_b \quad \text{and} \quad \hat{B} = \left[\frac{\partial \lambda}{\partial [b_0, b_1]^\top} \right]_{(b_0, b_1) = (\hat{\beta}_0, \hat{\beta}_1)}.$$

- Calculate plug-in covariance matrix $\hat{\Sigma}_{\text{plug}} = \hat{B}^{-1} \hat{A}_{\text{boot}} \hat{B}^{-1}$.

Algorithm 2: Plug-in Bootstrap Algorithm

3.5. Simulation Study

An extensive simulation study was conducted to evaluate the performance of the phase function-based estimators for various underlying distributions. In this section, we report and discuss a representative selection of these simulation results.

First, parameter estimation was explored in the univariate setting. Data were generated according to the model $Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$ and $W_i = X_i + U_i$, $i = 1, \ldots, n$ with true parameters $(\beta_0, \beta_1) = (1, 3)$. Three asymmetric distributions were used to simulate X, namely (1) a half-normal distribution, $X \sim |N(0, 1)|$, (2) an exponential distribution, $X \sim \exp(1)$, and (3) a bimodal mixture distribution, $X \sim 0.5N(5, 1^2) + 0.5N(2.5, 0.6^2)$. Three different distributions were considered for error components U and ε, namely the normal, t-distribution with 2.5 degrees of freedom, and the Cauchy distribution. For the Normal and $t_{2.5}$ distributions, the error components were simulated to have mean 0 and respective variances σ_U^2 and σ_ε^2. For the Cauchy distribution, the error components were simulated to be symmetric about 0 and have respective interquartile range (IQR) σ_U and σ_ε. The variance and IQR parameters were chosen to achieve specific noise-to-signal ratios, $p_W = \sigma_U^2/\sigma_X^2$ and $p_Y = \sigma_\varepsilon^2/(\beta_1\sigma_X)^2$. The noise-to-signal ratios pairs reported here are $(p_W, p_Y) = (0.25, 0.40)$. Results are reported for sample sizes $n \in \{500, 1000\}$. Simulation with other noise-to-signal ratios were carried out, and these results are reported in the Section C.2 of the Supplement Material. For each possible configuration of simulation specifications, $N = 2000$ samples were generated.

For the Normal and $t_{2.5}$ error cases, four different estimators were calculated for each simulated dataset. First, the naive estimators ignoring measurement error were obtained by regressing the contaminated W on Y. Second, the generalized method of moments estimators using $M = 3$ moments were computed. Three different choices of weight function were considered for the phase function estimator. Table 3.5 in the supplemental material compares the resulting estimators. As the weight function $K(t) = (1 - |t|)^2\mathbb{I}(|t| \leq 1)$ was found to have consistently good performance,the corresponding results are reported here. Finally, the disattenuated regression estimators were also calculated. For disattenuation,

the parameters (σ_U^2, σ_X^2) were treated as known quantities, and would not be computable in practice under the minimal model assumptions for the phase function method. For the Cauchy error case with infinite variance, no analog for disattenuation is known. However, even though there is no theoretical justification for doing so, the generalized method of moments estimators were computed to compare to the phase function estimators.

Now, letting $\hat{\beta}_{m,j}^{(\text{method})}$ denote the estimator of β_j calculated for the mth sample with the superscript "method" a placeholder for a specific method from those listed above, define squared error $\text{SE}_{m,j}^{(\text{method})} = [\hat{\beta}_{m,j}^{(\text{method})} - \beta_j]^2$. As both the generalized method of moments and phase function estimators are very prone to outliers in small samples, the median square errors is used rather than mean square error, as the former is more robust against these outliers. For the Normal and $t_{2.5}$ error case, we report in Table 3.1 the *ratios* of median square errors for the naive, generalized method of moments, and phase function estimators relative to the disattenuated estimators. An entry in the table larger than 1 indicates superior performance of the disattenuated estimators, while an entry smaller than 1 indicates superior performance of the associated method. Entries can also be compared across methods, with a larger entry indicating worse performance of a method for a given set of simulation specifications. The full simulation results, including the median squared error and a robust estimate of standard error based in the interquartile range, are given in the Section 3.8.3 of the Appendix.

Error type	True X	n	Naive		GMM		Phase	
			β_0	β_1	β_0	β_1	β_0	β_1
Normal	$\lvert N(0,1)\rvert$	500	73.0	98.7	2.35	2.81	**2.15**	**2.42**
		1000	131.3	189.8	2.55	3.16	**1.83**	**2.32**
	exp(1)	500	44.3	67.5	**1.18**	**1.15**	**1.18**	1.59
		1000	89.0	129.4	**1.27**	**1.24**	1.36	1.77
	Bimodal	500	100.7	118.4	10.1	11.5	**5.71**	**6.48**
		1000	200.2	235.0	9.74	11.2	**4.02**	**4.74**
$t_{2.5}$	$\lvert N(0,1)\rvert$	500	5.89	6.18	0.30	0.29	**0.16**	**0.18**
		1000	8.75	9.32	0.29	0.29	**0.09**	**0.12**
	exp(1)	500	6.64	5.88	0.17	**0.12**	**0.14**	0.19
		1000	9.09	8.75	0.16	**0.11**	**0.10**	0.12
	Bimodal	500	6.29	6.10	1.26	1.21	**0.27**	**0.24**
		1000	9.29	9.12	1.46	1.42	**0.12**	**0.11**

Table 3.1: Ratio of median square error of estimators relative to the disattenuated regression estimators in the univariate model simulation with model errors being Normal and $t_{2.5}$ distributions. Note GMM stands for generalized method of moments.

Considering the results in Table 3.1, we note that the naive estimator performs the worst among all the considered estimators across all simulation configurations. This is to be expected due to the known bias when not correcting for measurement error. For normally distributed errors, the phase function estimator performs better than generalized method of moments for both the cases X distributed as half-normal and as a bimodal mixture of normals. The improvement of the phase method is especially dramatic in the bimodal X case considered. On the other hand, for X having an exponential distribution, generalized method of moments performs better than the phase function method.

We reach similar conclusions when considering the case of a $t_{2.5}$ distribution for the error. Overall, the phase function method has superior performance for the cases X half-normal and X bimodal. In the case of X having an exponential distribution, generalized method of moments does better at estimating the slope β_1, while the phase function method does better at estimating the intercept β_0. We initially found the good perfor-

mance of generalized method of moments surprising, as its implementation here makes use of the third sample moments, whereas third moments do not exist for the error distribution used. However, generalized method of moments downweights the second and third sample moments using the fourth through sixth sample moments. Intuitively, the latter quantities will be really large, to some extent regulating the effect of using the former on estimating parameters. Still, the performance of the phase function method is generally far superior in this setting. In fact, noting that most of the median square error ratios are much smaller than 1, the phase function method is seen to be superior to correcting for attenuation using known error variances.

True X	n	GMM		Phase	
		β_0	β_1	β_0	β_1
$\lvert N(0,1)\rvert$	500	4.44	8.99	0.05	0.10
	1000	4.42	9.00	0.02	0.04
exp(1)	500	5.43	8.99	0.03	0.05
	1000	6.29	9.00	0.01	0.03
Bimodal	500	52.39	8.94	2.32	0.15
	1000	53.60	8.98	1.85	0.12

Table 3.2: Median square error of the generalized method of moments estimators, denoted GMM in the table, and the phase function estimators when the model errors are Cauchy.

Table 3.2 presents the simulation results for the generalized method of moments and the phase function estimators when the model errors follow a Cauchy distribution. In all the considered settings, the phase function estimator has a much smaller median square error than the generalized method of moments. The poor performance of generalized method of moments is expected because no moments exists for the Cauchy distribution. The phase function method, however, still performs well as it does not rely on the existence of error moments.

A second simulation study was done using two predictors, one measured with error and one without. Data were simulated according to the model $Y_i = \beta_0 + \beta_X X_i + \beta_Z Z_i + \epsilon_i$, $W_i = X_i + U_i$, $i = 1, \ldots, n$ with parameters $\beta_0 = 0, \beta_X = 3$, and $\beta_Z = 2$. Here,

True X	n	Naive		Phase	
		β_X	β_Z	β_X	β_Z
$\lvert N(0,1)\rvert$	1000	107.1	35.4	15.0	40.2
	2000	219.1	81.2	2.19	6.47
Bimodal	1000	152.8	51.9	10.1	24.5
	2000	343.3	104.7	9.02	19.9

Table 3.3: Ratio of median square error of estimators relative to the simulation-extrapolation regression estimators in the bivariate model simulation.

X is the error-prone covariate while Z is error-free. Samples sizes $n \in \{1000, 2000\}$ were considered. We include here results for the two cases X half-normal and X having the bimodal normal mixture defined at the start of this section. The covariate Z was generated from the same distribution as X, and a normal copula with $\rho = 0.5$ was used to generate X and Z to be correlated. The error distributions were taken to be normal with noise-to-signal ratios as in the univariate simulation. For each simulation configuration, 2000 replications were run. For each run, the phase function estimators and the naive estimators for both β_X and β_Z were computed. Furthermore, simulation-extrapolation of Stefanski and Cook (1995) was also implemented using the known measurement error variance. When the measurement error variance is unknown or not estimable, simulation-extrapolation cannot be used. It is therefore included for comparative purposes. Table 3.3 reports again the ratio of median square error for the naive and phase function methods relative to the simulation-extrapolation estimators. As before, see Section 3.8.3 in the Appendix for the full simulation results.

Again, the poor performance of the naive method in Table 3.3 is not surprising. The phase function method holds up well against simulation-extrapolation. It is clear that the method improves (in a relative sense) as the sample size increases from 1000 to 2000. Furthermore, the phase function approach has large relative median squared errors when $(p_W, p_Y) = (0.25, 0.4)$, corresponding to large measurement error contamination. However, these scenarios also improve, sometimes dramatically so, when the same size increases.

Finally, we performed a simulation study to examine the performance of the (full) bootstrap and plug-in bootstrap methods for estimating standard errors of the parame-

ters. Data were simulated from the univariate model used to generate Table 3.1. For each simulated sample, both bootstrap methods were used to estimate the standard errors of the model coefficients. Reported here are the results for X half-normal and X bimodal, and two sample sizes $n \in \{1000, 2000\}$. For each simulation configuration, 2000 samples were generated. For each, the phase function estimates were computed. A total of $B = 100$ bootstrap samples were generated for each of the methods described in Section 3.4.2 to estimate standard errors. The true standard error was also estimated using the 1000 pairs $\hat{\beta}_0, \hat{\beta}_1$ estimated from the simulated data using the phase function methods. The median of $\sqrt{n} \times \widehat{\text{se}}$, with $\widehat{\text{se}}$ denoting estimated standard error, is reported in Table 3.4 for each method.

We note in Table 3.4 that the full bootstrap generally gives estimated standard errors very close to the true (Monte Carlo) values. The plug-in method has a tendency to over-estimate the standard error, especially for sample size 1000. However, the plug-in method is superior in terms of computation speed. These computational time comparisons are based on running simulations on a distributed computing system with 80 nodes consisting of 36 cores each with 256GB memory and with an Intel Xeon E5-2695 v4 CPU. For sample size $n = 1000$, the average computation time for the full bootstrap around 34 minutes, while the plug-in bootstrap had an average computation time of around 1 minute. Similarly, for sample size $n = 2000$ the full and plug-in average computation times were around 49 minutes and 2 minutes, respectively. In many instances, one might be willing to use a method that over-estimates the size of the standard error for this type of speed-up in computation.

True X	n	Monte Carlo		$\text{Bootstrap}_{\text{full}}$		$\text{Bootstrap}_{\text{plug}}$	
		β_0	β_1	β_0	β_1	β_0	β_1
$\lvert N(0,1)\rvert$	1000	0.48	0.56	0.48	0.55	0.56	0.71
	2000	0.39	0.46	0.40	0.46	0.42	0.53
Bimodal	1000	2.82	0.75	2.95	0.79	5.27	1.36
	2000	2.26	0.60	2.22	0.59	2.90	0.75

Table 3.4: True standard error (Monte Carlo) and median of estimated standard error, scaled by the square root of the sample size, using two different bootstrap approaches.

3.6. Air Quality Data Examples

Here, we consider a dataset analyzed by De Vito et al. (2008) considering the measurement of carbon monoxide (CO) levels in present in an urban environment over time. The dataset is publicly available online at the UCI Machine Learning Repository (https://archive.ics.uci.edu/ml/datasets.html) and is labeled *Air Quality*. In the experiment reported, a low-cost gas multi-censor device, also known as an electronic nose, was used to monitor atmospheric pollutants in an urban environment. Carbon monoxide was one of the pollutants being monitored and is of primary interest in our analysis. Measurements obtained by electronic noses use tin oxide as a proxy for carbon monoxide. These devices are also subject to measurement error, especially when compared to networks of spatially distributed fixed stations using industrial spectromoters. The latter are commonly used to monitor air pollution in urban environments, but use is restricted by cost and size considerations. The sources of measurement error for electronic noses range from known device stability issues to local atmospheric dynamics. Even so, it is desirable to consider the proper calibration of electronic noses for supplemental use in monitoring air pollution in urban areas. Specifically, we consider estimating the true relationship between tin oxide (subject to measurement error) and carbon monoxide.

The experiment, which lasted 13 months, was performed at a main road with heavy traffic in an Italian city. During this period, hourly observations were collected from both an electronic nose (W data) and a distributed network of seven fixed stations (Y data).

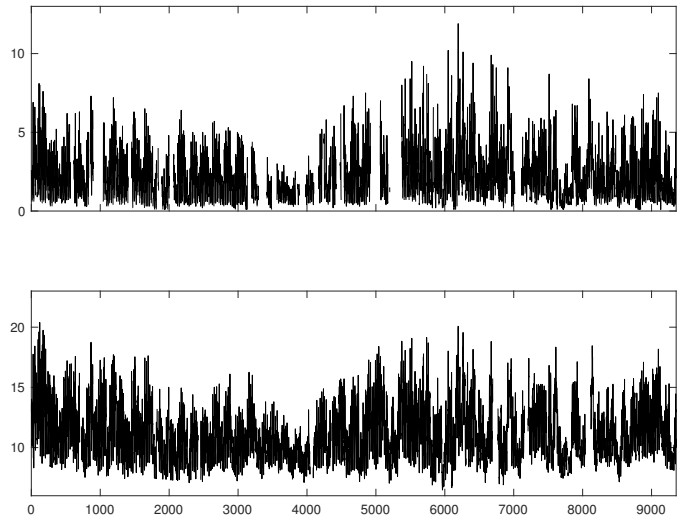


Figure 3.1: Time series plots for carbon monoxide Y_t (top) and the average sensor output W_t (bottom).

The measurements represent hourly averages of data collected at 8 second increments. The data are denoted $(W_t, Y_t),\ t = 1, \ldots, T$, with $T = 9357$ hourly periods transpiring during the experiment. However, 2013 of these have partially or completely missing data, leaving 7344 complete observations for the analysis. Time series plots of the measurements are shown in Figure 3.1. Note that the W measurements in the figure and throughout the analysis are equal to the original data divided by 100.

To account for time-of-day effects on pollution levels, the data were de-trended. To this end, let $\mathcal{I}_k = \{t : t = k + 24(j-1),\ j = 1, 2, \ldots \text{ and } t \leq T\}$ with $k = 1, \ldots, 24$ denote the collection of indices corresponding to measurements at hour k. Define observed hourly mean $\hat{\mu}_k = |\mathcal{I}_k|^{-1} \sum_{t \in \mathcal{I}_k} W_t$ for $k = 1, \ldots, 24$. The expression for $\hat{\mu}_k$ makes use of a slight abuse of notation, as the sum is only taken over indices corresponding to non-missing observations. The de-trended data are calculated as

$$\tilde{W}_t = W_t - \sum_{k=1}^{24} \mathbb{I}\{t \in \mathcal{I}_k\} \hat{\mu}_k,\ t = 1, \ldots, T.$$

The de-trended $\tilde{Y}_t$ are defined in an analogous manner, resulting in pairs $(\tilde{W}_t, \tilde{Y}_t)$, $t = 1, \ldots, T$. It is now assumed that $\tilde{Y}_t = \beta_1 X_t + \varepsilon_t$ and $\tilde{W}_t = X_t + U_t$ with X_t denoting the true CO level at time t. Note the lack of intercept term β_0 in the model. This is

a result of de-trending the data. We assume that the error components U_t and ε_t are independent, and that these error components are independent of stationary time series X_t. All variables are assumed to have finite variance. The stationarity of X_t is important as this ensures that the empirical phase functions still a consistent estimate ρ_X.

The generalized method of moments and phase function estimators of slope β_1 were calculated. To account for the correlation structure in X_t, the block bootstrap with block length $L = 192$ was used to estimate the associated standard errors, see Kunsch (1989) for details on this technique. The generalized method of moments estimator is $\hat{\beta}_1^{(\text{GMM})} = 0.73$ with estimated standard error 0.07. The phase function estimator is $\hat{\beta}_1^{(\text{phase})} = 0.71$ with estimated standard error 0.02. The naive estimator of slope is 0.52, indicating the strong effect of measurement error here. Comparing the phase function and naive estimators of slope using the known attenuation relationship suggests the proportion of error variance is 0.36. Moreover, the generalized method of moments and phase function estimates seemingly correct for the exogenous contamination present in the electronic nose measurements. While the two estimators are close to one another, the standard error of generalized method of moments is substantially larger than that of the phase function estimator.

3.7. Conclusion

The proposed phase function methodology is a new solution to the linear errors-in-variables problem where replicate data and/or prior knowledge of measurement error variance are not available. Contamination of the observed features should not be ignored when making an inference, but strong model requirements can make it difficult to appropriately correct the error and leave the practitioner with a biased model. To our knowledge, the only solution not making such strict assumptions is the generalized method of moments. Our proposed method is seen to be competitive with generalized method of moments, and often has much smaller median squared error. Furthermore, the phase function-based method does not rely on the existence of an arbitrary number of moments. Future work will consider combining the strengths of the generalized method

of moments and phase function methods: generalized method of moments can be implemented when the underlying variable has a symmetric distribution, whereas the phase function method requires asymmetry of the underlying distribution.

3.8. Appendix

3.8.1. Expressions for the second derivative of $D(b_0, b_1)$

In Section 3.4, a plug-in bootstrap method is proposed for estimating the standard errors of the phase function estimators. Evaluation there requires calculation of the second derivatives of the distance metric D evaluated at the estimators $\hat{\beta}_0$ and $\hat{\beta}_1$. These functions are reported here.

Specifically,

$$\frac{\partial^2 D}{\partial b_0^2} = \sum_{i,j,k,l} \int_0^{t^*} \Big[\cos\{t(Y_i - b_0 - b_1 W_j)\} \cos\{t(Y_l - b_0 - b_1 W_k)\} \\ - \sin\{t(Y_i - b_0 - b_1 W_j)\} \sin\{t(Y_l - b_0 - b_1 W_k)\} \Big] 2t^2 K_{t^*}(t) dt,$$

$$\frac{\partial^2 D}{\partial b_0 \partial b_1} = \sum_{i,j,k,l} \int_0^{t^*} \Big[W_j \cos\{t(Y_i - b_0 - b_1 W_j)\} \cos\{t(Y_l - b_0 - b_1 W_k)\} \\ - W_k \sin\{t(Y_i - b_0 - b_1 W_j)\} \sin\{t(Y_l - b_0 - b_1 W_k)\} \Big] 2t^2 K_{t^*}(t) dt,$$

and

$$\frac{\partial^2 D}{\partial b_1^2} = \sum_{i,j,k,l} \int_0^{t^*} \Big[W_j W_k \cos\{t(Y_i - b_0 - b_1 W_j)\} \cos\{t(Y_l - b_0 - b_1 W_k)\} \\ - W_k^2 \sin\{t(Y_i - b_0 - b_1 W_j)\} \sin\{t(Y_l - b_0 - b_1 W_k)\} \Big] 2t^2 K_{t^*}(t) dt.$$

The quadruple sums can be eliminated using some simple but tedious algebra, giving expressions that are computationally convenient,

$$\frac{\partial^2 D}{\partial b_0^2} = \int_0^{t^*} 2t^2 K_{t^*}(t) \left[\sum_{i,j} \cos\left\{ t(Y_i - b_0 - b_1 W_j) \right\} \right]^2 dt$$
$$- \int_0^{t^*} 2t^2 K_{t^*}(t) \left[\sum_{i,j} \sin\left\{ t(Y_i - b_0 - b_1 W_j) \right\} \right]^2 dt,$$

$$\frac{\partial^2 D}{\partial b_0 \partial b_1} = \int_0^{t^*} 2t^2 K_{t^*}(t) \left[\sum_{i,j} \cos\left\{ t(Y_i - b_0 - b_1 W_j) \right\} \right] \left[\sum_{i,j} W_j \cos\left\{ t(Y_i - b_0 - b_1 W_j) \right\} \right] dt$$
$$- \int_0^{t^*} 2t^2 K_{t^*}(t) \left[\sum_{i,j} \sin\left\{ t(Y_i - b_0 - b_1 W_j) \right\} \right] \left[\sum_{i,j} W_j \sin\left\{ t(Y_i - b_0 - b_1 W_j) \right\} \right] dt,$$

and

$$\frac{\partial^2 D}{\partial b_1^2} = \int_0^{t^*} 2t^2 K_{t^*}(t) \left[\sum_{i,j} \cos\left\{ t(Y_i - b_0 - b_1 W_j) \right\} \right]^2 dt$$
$$- \int_0^{t^*} 2t^2 K_{t^*}(t) \left[\sum_{i,j} \sin\left\{ t(Y_i - b_0 - b_1 W_j) \right\} \right] \left[\sum_{i,j} W_j^2 \sin\left\{ t(Y_i - b_0 - b_1 W_j) \right\} \right] dt.$$

These expressions can be used to calculate the matrix $\hat{B}$ required for the bootstrap plug-in method for standard error estimation.

3.8.2. A brief review of the Generalized Method of Moments

In this section, we provide a brief overview of the generalized method of moments (GMM) approach to linear errors-in-variables models. GMM is a popular approach to estimating the parameters of linear EIV models. Recall the model

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i \quad \text{and} \quad W_i = X_i + U_i,$$

$i = 1, \ldots, n$. In this model, the parameters β_0 and β_1 are identifiable using moments of W and Y up to order 3, provided $E[(X - \mu_X)^3] \neq 0$. Similarly, the parameters are identifiable using moments up to order 4 provided the distributions of X, U, and ε are not all Gaussian. We briefly review implementation of GMM here. Our approach is similar to that proposed by Erickson and Whited (2002). GMM is a viable nonparametric

alternative to the phase function approach, in that no parametric model assumptions are required for implementation.

For GMM using sample moments up to order K, it is assumed that each of the variables X, U, and ε has at least $2K$ finite moments. Furthermore, it is assumed that U and ε have distributions symmetric about 0, $\mathrm{E}\left[U^{2k-1}\right] = \mathrm{E}\left[\varepsilon^{2k-1}\right] = 0$ for $k = 1, 2, \ldots, K$. Note that the use of the first K moments requires that the underlying distributions have $2K$ moments for the estimators derived here to be asymptotically normally distributed with finite variance.

Let μ_X denote the mean of X, and let σ_X^2, σ_U^2, and σ_ε^2 denote the respective variances of X, U, and ε. Additionally, let $\mu_{X,j} = \mathrm{E}\left[(X - \mu_X)^j\right]$ denote the jth centered moment of X, $j = 3, \ldots, 2K$, with equivalent notation holding for $\mu_{U,j}$ and $\mu_{\varepsilon,j}$. Finally, for the pair of random variables (W, Y), let $\nu_{j,k}$ denote the joint centered moments,

$$\nu_{j,k} = \mathrm{E}\left[(W - \mu_X)^j \left(Y - \beta_0 - \beta_1 \mu_X\right)^k\right]. \tag{3.15}$$

Due to the independence of X, U, and ε, the joint moment $\nu_{j,k}$ can be expressed in terms of the marginal moments of X, U, and ε up to order $j + k$. Making a few special cases explicit, note that $\nu_{2,0} = \sigma_X^2 + \sigma_U^2$, $\nu_{1,1} = \beta_1 \sigma_X^2$, and $\nu_{0,2} = \beta_1^2 \sigma_X^2 + \sigma_\varepsilon^2$.

Now, let $\boldsymbol{\theta}_{(1)} = \{\mu_X, \beta_0, \beta_1\}$ and $\boldsymbol{\theta}_{(2)} = \{\sigma_X^2, \sigma_U^2, \sigma_\varepsilon^2\}$, and let $\boldsymbol{\theta}_{(2j-1)} = \{\mu_{X,2j-1}\}$ and $\boldsymbol{\theta}_{(2j)} = \{\mu_{X,2j}, \mu_{U,2j}, \mu_{\varepsilon,2j}\}$, $j = 2, \ldots, \lfloor K/2 \rfloor$, denote the higher-order moments. Finally, let $\boldsymbol{\theta}_K = \left\{\boldsymbol{\theta}_{(1)}, \ldots, \boldsymbol{\theta}_{(K)}\right\}$ denote the collection of unknown parameters required to specify a model up to order K. The random variables

$$A_{jk}(\boldsymbol{\theta}) = n^{-1/2} \sum_{i=1}^{n} \left\{(W_i - \mu_X)^j \left(Y_i - \beta_0 - \beta_1 \mu_X\right)^k - \nu_{jk}\right\}$$

have $\mathrm{E}\left[A_{jk}\right] = 0$ and $\mathrm{Cov}\left[A_{jk}, A_{j',k'}\right] = \nu_{j+j',k+k'} - \nu_{jk}\nu_{j',k'}$ for $j + j' + k + k' \leq 2K$. A such, the $A_{j,k}$ can be used to construct GMM estimators of the parameters. Specifically, let $\boldsymbol{A}_K(\boldsymbol{\theta}_K)$ denote the vector consisting of all terms A_{jk} with $j, k = 0, \ldots, K$ and $1 \leq j + k \leq K$. Now, define $\boldsymbol{\Sigma}_K$ to be the covariance matrix corresponding to vector $\boldsymbol{A}_K$.

This covariance matrix can be estimated empirically by defining joint sample moments

$$\hat{\nu}_{j,k} = \frac{1}{n}\sum_{i=1}^{n}\left(W_i - \bar{W}\right)^j\left(Y_i - \bar{Y}\right)^k$$

and subsequently letting

$$\widehat{\mathrm{Cov}}\left[A_{jk}, A_{j',k'}\right] = \hat{\nu}_{j+j',k+k'} - \hat{\nu}_{jk}\hat{\nu}_{j',k'}, \quad j + j' + k + k' \leq 2K.$$

Let $\hat{\boldsymbol{\Sigma}}_K$ denote this estimated covariance matrix. The GMM parameter estimates are then found by minimizing the quadratic form

$$G_K(\boldsymbol{\theta}_K) = \boldsymbol{A}_K(\boldsymbol{\theta}_K)^\top \hat{\boldsymbol{\Sigma}}_K^{-1} \boldsymbol{A}_K(\boldsymbol{\theta}_K). \tag{3.16}$$

Note that the implementation of the GMM approach requires the use of $K \geq 3$, as the choices $K = 1, 2$ result in an overidentified system in terms of the parameters in $\boldsymbol{\theta}_K$.

3.8.3. Additional Simulation Results for Univariate Simulation

The effect of weighting function

The simulation study in Section 5.1 (univariate EIV model) of the main paper explore three different choice of weighting function in calculating the phase function estimator: $K_1(t) = (1 - |t|)^2\mathbb{I}(|t| \leq 1)$, $K_2(t) = (1 - |t|)\mathbb{I}(|t| \leq 1)$, $K_3(t) = (1 - t^2)\mathbb{I}(|t| \leq 1)$. Table 3.5 presents the median SE of phase function estimates for these three weight function choices for a subset of simulation settings with $X \sim |N(0,1)|$ or $X \sim \exp(1)$, and $(p_W, p_Y) = (0.25, 0.40)$. The results for simulation configurations not reported in Table 3.5 follow the same general patterns.

True X	n	Error	$K_1(t)$		$K_2(t)$		$K_3(t)$	
			β_0	β_1	β_0	β_1	β_0	β_1
$\lvert N(0,1)\rvert$	500	Normal	3.37	4.41	3.47	4.79	3.4	4.74
			(0.14)	(0.18)	(0.14)	(0.19)	(0.14)	(0.19)
		Laplace	1.9	3.19	1.83	3.07	1.84	3.04
			(0.09)	(0.14)	(0.08)	(0.13)	(0.08)	(0.13)
	1000	Normal	3.21	4.38	3.31	4.6	3.26	4.44
			(0.14)	(0.18)	(0.14)	(0.2)	(0.14)	(0.19)
		Laplace	1.65	3.01	1.62	2.88	1.62	2.89
			(0.08)	(0.13)	(0.08)	(0.12)	(0.08)	(0.12)
exp(1)	500	Normal	4.75	4.28	5.08	5.07	4.94	4.72
			(0.21)	(0.2)	(0.23)	(0.24)	(0.23)	(0.23)
		Laplace	2.5	3.63	2.59	3.89	2.52	3.63
			(0.11)	(0.16)	(0.12)	(0.18)	(0.11)	(0.17)
	1000	Normal	5.55	4.97	5.92	6.04	5.86	5.49
			(0.24)	(0.23)	(0.26)	(0.27)	(0.25)	(0.25)
		Laplace	2.62	3.2	2.56	3.68	2.56	3.29
			(0.11)	(0.15)	(0.12)	(0.17)	(0.11)	(0.15)

Table 3.5: $n \times$ median{SE} and the corresponding interquartile range for the phase function estimators with weighting functions $K_1(t)$, $K_2(t)$, and $K_3(t)$.

As can be seen in Table 3.5, the choice of weights function does not have a major impact on the quality of the estimators when using medianSE as criterion. However, the choice of weight function $K_1(t) = (1 - |t|)^2\mathbb{I}(|t| \leq 1)$ most often results in the lowest median square error for both β_0 and β_1. As such, the phase function-based estimators are compared to the other methods of estimation for this choice of weight function.

Full Simulation Results for Univariate EIV model

In this section, we present the full results for the simulation studies in the simple EIV setting in Section 3.5. Data were generated according to the model $Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$ and $W_i = X_i + U_i$, $i = 1, \ldots, n$ with true parameters $(\beta_0, \beta_1) = (1, 3)$. Three asymmetric distributions were used to simulate X, namely (1) a half-normal distribution, $X \sim |N(0,1)|$, (2) an exponential distribution, $X \sim \exp(1)$, and (3) a bimodal mixture distribution, $X \sim 0.5N(5, 1^2) + 0.5N(2.5, 0.6^2)$. Three different distributions were considered for error components U and ε, namely the normal, t-distribution with 2.5 degrees

n	Error	(p_W, p_Y)	Naive		GMM		Disattenuation		Phase	
			β_0	β_1	β_0	β_1	β_0	β_1	β_0	β_1
500	Normal	(0.075,0.15)	14.16	21.94	1.17	1.58	0.5	0.53	0.95	1.27
			(0.16)	(0.21)	(0.05)	(0.07)	(0.02)	(0.02)	(0.04)	(0.05)
		(0.25,0.40)	114.89	180.01	3.69	5.12	1.57	1.82	3.37	4.41
			(0.73)	(0.97)	(0.18)	(0.24)	(0.07)	(0.09)	(0.14)	(0.18)
	$t_{2.5}$	(0.075,0.15)	8.45	13.18	0.78	1.11	1.45	2.16	0.59	1.01
			(0.16)	(0.22)	(0.03)	(0.05)	(0.06)	(0.07)	(0.03)	(0.04)
		(0.25,0.40)	71.86	112.03	3.67	5.34	12.2	18.13	1.9	3.19
			(0.95)	(1.39)	(0.2)	(0.28)	(0.38)	(0.55)	(0.09)	(0.14)
1000	Normal	(0.075,0.15)	28.39	44.16	1.17	1.62	0.53	0.56	0.94	1.21
			(0.24)	(0.3)	(0.06)	(0.08)	(0.02)	(0.02)	(0.04)	(0.06)
		(0.25,0.40)	229.6	358.97	4.45	5.97	1.75	1.89	3.21	4.38
			(1.1)	(1.39)	(0.19)	(0.24)	(0.08)	(0.08)	(0.14)	(0.18)
	$t_{2.5}$	(0.075,0.15)	18.54	29.12	0.98	1.31	2.05	3.08	0.63	0.99
			(0.27)	(0.39)	(0.05)	(0.06)	(0.08)	(0.1)	(0.03)	(0.04)
		(0.25,0.40)	154.67	243.3	5.04	7.44	17.67	26.1	1.65	3.01
			(1.67)	(2.52)	(0.3)	(0.47)	(0.52)	(0.79)	(0.08)	(0.13)

Table 3.6: Median square errors of estimators and the corresponding interquartile range (in parentheses), scaled by the sample size, in the univariate regression simulation when the true distribution of X is half-normal.

of freedom, and the Cauchy distribution. For the Normal and $t_{2.5}$ distributions, the error components were simulated to have mean 0 and respective variances σ_U^2 and σ_ε^2. For the Cauchy distribution, the error components were simulated to be symmetric about 0 and have respective interquartile range (IQR) σ_U and σ_ε. The variance and IQR parameters were chosen to achieve specific noise-to-signal ratios, $p_W = \sigma_U^2/\sigma_X^2$ and $p_Y = \sigma_\varepsilon^2/(\beta_1\sigma_X)^2$. The noise-to-signal ratios pairs reported here are $(p_W, p_Y) \in \{(0.075, 0.15), (0.25, 0.40)\}$. Results are reported for sample sizes $n \in \{500, 1000\}$. For each configuration, the median square error of each estimator is reported with the corresponding interquartile range.

n	Error	NSR	Naive		GMM		Disattenuation		Phase	
			β_0	β_1	β_0	β_1	β_0	β_1	β_0	β_1
500	Normal	(0.075,0.15)	22.25	22.4	1.46	0.95	1.2	0.67	1.62	1.69
			(0.31)	(0.24)	(0.07)	(0.05)	(0.05)	(0.03)	(0.07)	(0.07)
		(0.25,0.40)	178.85	181.39	4.77	3.08	4.04	2.69	4.75	4.28
			(1.33)	(1.17)	(0.21)	(0.15)	(0.17)	(0.11)	(0.21)	(0.2)
	$t_{2.5}$	(0.075,0.15)	13.19	13.39	0.95	0.58	2.57	2.3	0.88	1.07
			(0.29)	(0.24)	(0.04)	(0.03)	(0.1)	(0.08)	(0.04)	(0.05)
		(0.25,0.40)	117.47	113.1	2.97	2.26	17.69	19.25	2.5	3.63
			(1.69)	(1.59)	(0.14)	(0.11)	(0.65)	(0.58)	(0.11)	(0.16)
1000	Normal	(0.075,0.15)	43.83	44.13	1.55	1.07	1.16	0.67	1.74	1.72
			(0.44)	(0.34)	(0.07)	(0.05)	(0.05)	(0.03)	(0.08)	(0.07)
		(0.25,0.40)	363.62	362.91	5.18	3.48	4.08	2.8	5.55	4.97
			(1.99)	(1.68)	(0.23)	(0.15)	(0.19)	(0.12)	(0.24)	(0.23)
	$t_{2.5}$	(0.075,0.15)	29.74	28.94	1.05	0.65	3.28	2.96	0.92	1.27
			(0.47)	(0.41)	(0.05)	(0.04)	(0.13)	(0.11)	(0.04)	(0.06)
		(0.25,0.40)	249.07	242.99	4.28	3.06	27.39	27.79	2.62	3.2
			(2.89)	(2.67)	(0.19)	(0.15)	(0.93)	(0.86)	(0.11)	(0.15)

Table 3.7: Median square errors of estimators and the corresponding interquartile range (in parentheses), scaled by the sample size, in the univariate regression simulation when the true distribution of X is exponential.

n	Error	NSR	Naive		GMM		Disattenuation		Phase	
			β_0	β_1	β_0	β_1	β_0	β_1	β_0	β_1
500	Normal	(0.075,0.15)	306.68	21.62	63.48	4.74	9.16	0.53	23.56	1.61
			(3.31)	(0.21)	(2.52)	(0.17)	(0.39)	(0.02)	(1.14)	(0.08)
		(0.25,0.40)	2528.79	178.52	254.05	17.4	25.12	1.51	143.37	9.76
			(13.87)	(0.87)	(11.5)	(0.8)	(1.14)	(0.07)	(6.73)	(0.48)
	$t_{2.5}$	(0.075,0.15)	192.04	13.71	43.9	2.97	28.71	2.09	15.48	0.96
			(3.21)	(0.22)	(1.45)	(0.1)	(1.05)	(0.08)	(0.71)	(0.05)
		(0.25,0.40)	1572.13	111.86	314.14	22.16	249.82	18.35	67.16	4.39
			(20.39)	(1.37)	(10.69)	(0.74)	(7.76)	(0.55)	(3.43)	(0.23)
1000	Normal	(0.075,0.15)	616.32	43.87	71.47	4.99	9.1	0.55	20.96	1.41
			(4.67)	(0.3)	(3.29)	(0.23)	(0.38)	(0.02)	(0.97)	(0.06)
		(0.25,0.40)	5047.12	361.45	245.51	17.2	25.21	1.54	101.42	7.29
			(19.57)	(1.24)	(10.93)	(0.75)	(1.1)	(0.07)	(4.91)	(0.35)
	$t_{2.5}$	(0.075,0.15)	403.52	28.6	77.56	5.38	44.06	3.11	14.57	0.94
			(5.49)	(0.39)	(2.61)	(0.18)	(1.57)	(0.11)	(0.64)	(0.04)
		(0.25,0.40)	3474.56	246.26	546.55	38.33	373.97	27.02	45.81	3.05
			(35.62)	(2.5)	(19.73)	(1.37)	(11.8)	(0.82)	(2.52)	(0.16)

Table 3.8: Median square errors of estimators and the corresponding interquartile range (in parentheses), scaled by the sample size,in the univariate regression simulation when the true distribution of X is a mixture of normal distributions.

True X	n	(p_W, p_Y)	GMM		Phase function	
			β_0	β_1	β_0	β_1
$\|N(0,1)\|$	500	(0.075,0.15)	4.04	8.97	0.02	0.04
			(0.07)	(0.07)	(0.00)	(0.00)
		(0.25,0.40)	4.44	8.99	0.05	0.1
			(0.07)	(0.02)	(0.00)	(0.00)
	1000	(0.075,0.15)	4.5	9	0.01	0.02
			(0.07)	(0.03)	(0.00)	(0.00)
		(0.25,0.40)	4.42	9	0.02	0.04
			(0.07)	(0.01)	(0.00)	(0.00)
exp(1)	500	(0.075,0.15)	4.09	8.92	0.01	0.02
			(0.11)	(0.12)	(0.00)	(0.00)
		(0.25,0.40)	5.43	8.99	0.02	0.05
			(0.12)	(0.05)	(0.00)	(0.00)
	1000	(0.075,0.15)	5.2	8.98	0	0.01
			(0.12)	(0.08)	(0.00)	(0.00)
		(0.25,0.40)	6.29	9	0.01	0.02
			(0.12)	(0.02)	(0.00)	(0.00)
Bimodal	500	(0.075,0.15)	19.71	8.75	2.12	0.13
			(1.8)	(0.13)	(0.09)	(0.01)
		(0.25,0.40)	52.39	8.94	2.32	0.15
			(1.89)	(0.07)	(0.11)	(0.01)
	1000	(0.075,0.15)	29.02	8.93	1.34	0.08
			(1.89)	(0.08)	(0.06)	(0.00)
		(0.25,0.40)	53.6	8.98	1.85	0.12
			(1.89)	(0.02)	(0.08)	(0.01)

Table 3.9: Median square error and interquartile range of the GMM and phase function estimators in the univariate regression simulation when model errors are Cauchy

Full Simulation Results for Multiple Regression

In this section, we present the full results for the simulation study in the multiple EIV linear model setting in the section 3.5. Data were simulated according to the model $Y_i = \beta_0 + \beta_X X_i + \beta_Z Z_i + \epsilon_i$, $W_i = X_i + U_i$, $i = 1, \ldots, n$ with parameters $\beta_0 = 0, \beta_X = 3$, and $\beta_Z = 2$. Here, X is the error-prone covariate while Z is error-free. Samples sizes $n \in \{1000, 2000\}$ were considered. We include here results for the two cases X half-normal and X having the bimodal normal mixture defined in the simple EIV setting. The covariate Z was generated from the same distribution as X, and a normal copula

n	Error	(p_W, p_Y)	Naive		Phase function		SIMEX	
			β_X	β_Z	β_X	β_Z	β_X	β_Z
1000	Normal	(0.075,0.15)	715.69	167.7	7.34	9.96	4.19	2.78
			(1.92)	(1.01)	(0.32)	(0.46)	(0.18)	(0.13)
		(0.25,0.40)	2349.55	540.65	328.67	613.13	21.94	15.26
			(4.31)	(2.57)	(148.75)	(71.23)	(0.97)	(0.66)
	Laplace	(0.075,0.15)	705.18	164.94	7.75	12.76	5.79	3.75
			(2.15)	(1.05)	(0.38)	(0.58)	(0.27)	(0.16)
		(0.25,0.40)	2329.48	539.89	331.13	658.7	32.09	16.73
			(4.73)	(2.55)	(148.71)	(71.53)	(1.48)	(0.74)
2000	Normal	(0.075,0.15)	1417.18	326.47	7.13	9.26	4.19	3.1
			(2.73)	(1.44)	(0.3)	(0.43)	(0.19)	(0.15)
		(0.25,0.40)	4691.1	1089.42	46.83	86.77	21.41	13.41
			(5.95)	(3.82)	(2.86)	(5.69)	(0.96)	(0.61)
	Laplace	(0.075,0.15)	1419.04	326.72	8.34	12.18	6.12	3.32
			(3.48)	(1.55)	(0.36)	(0.57)	(0.27)	(0.16)
		(0.25,0.40)	4679.33	1073.46	51.75	109.01	36.64	17.17
			(7.36)	(3.86)	(3.22)	(7.79)	(1.51)	(0.8)

Table 3.10: Median square error and interquartile range (in parentheses), scaled by the sample size for the estimators in the multivariate regression simulation when X and Z are half-normal and correlated with correlation $\rho = .5$.

with $\rho = 0.5$ was used to generate X and Z to be correlated. The error distributions were taken to be normal and Laplace with noise-to-signal ratios as in the simple EIV model. For each simulation configuration, 2000 replications were run. Table 3.3 and 3.11 presents the median square error for the naive, phase function, and SIMEX estimator with their corresponding interquartile ranges.

3.8.4. Additional Data Examples

Abrasiveness Index Data

The data analyzed here was originally considered by Lombard (2005) in the context of estimating a quantile comparison function from paired data. Observations are pairs (W_j, Y_j), $j = 1, \ldots, 98$, where both W_j and Y_j represent measures of the abrasiveness index (AI) of a batch of coal. The AI is considered a proxy for the quality of coal, and is used to determine the price of a batch of coal. The Y_j measurements were obtained

n	Error	(p_W, p_Y)	Naive		Phase function		SIMEX	
			β_X	β_Z	β_X	β_Z	β_X	β_Z
1000	Normal	(0.075,0.15)	241.69	56.84	13.44	18.65	1.55	1.29
			(0.83)	(0.45)	(0.56)	(0.85)	(0.07)	(0.06)
		(0.25,0.40)	1056.68	249.27	69.64	117.72	6.91	4.81
			(2.22)	(1.47)	(5.55)	(9.45)	(0.29)	(0.23)
	Laplace	(0.075,0.15)	239.77	56.15	14.74	24.13	2.04	1.62
			(0.96)	(0.47)	(0.74)	(1.23)	(0.09)	(0.07)
		(0.25,0.40)	1056.6	248.34	149.4	250.81	9.49	5.65
			(2.86)	(1.45)	(19.69)	(65.43)	(0.46)	(0.26)
2000	Normal	(0.075,0.15)	483.2	113.82	10.74	14.71	1.77	1.5
			(1.13)	(0.63)	(0.46)	(0.71)	(0.07)	(0.06)
		(0.25,0.40)	2127.88	498.96	55.95	94.55	6.2	4.77
			(3.16)	(2.02)	(2.87)	(4.63)	(0.26)	(0.22)
	Laplace	(0.075,0.15)	486.34	113.48	11.72	21.03	2.13	1.59
			(1.4)	(0.66)	(0.62)	(0.93)	(0.1)	(0.07)
		(0.25,0.40)	2125.8	499.99	80.09	144.17	9.69	5.44
			(4.18)	(2.11)	(4.47)	(7.74)	(0.45)	(0.25)

Table 3.11: Median square error and interquartile range (in parentheses), scaled by the sample size, for the estimators in the multivariate regression simulation when X and Z are mixtures of normal distribution and correlated with correlation $\rho = .5$.

using the YGP method, see Yancey et al. (1951). This method is widely used, but is costly to implement. The W_j measurements were obtained using a similar method that is less involved and cheaper to implement. Contracts are typically written in terms of the YGP measurements, and it is of interest to determine the relationship between the new method and the YGP method. Here, we treat both the W_j and Y_j data as contaminated versions of the true quality of a batch of coal, denoted X_j. Assume that the linear errors-in-variables structure holds, i.e. $W_j = X_j + U_j$ and $Y_j = \beta_0 + \beta_1 X_j + \varepsilon_j$.

In Figure 3.2, we show kernel density estimates using normal reference plug-in bandwidths for both W and Y. Bandwidths selected using unbiased cross-validation were also considered, but did not alter the estimates in a visually discernible way. For the given data, the naive regression estimators, GMM estimators, and phase function-based estimators were calculated; the results are reported in Table 3.12. Also reported are the estimated variance components based on the second sample moments. Specifically,

$\hat{\sigma}_X^2 = s_{WY}/\hat{\beta}_1$, $\hat{\sigma}_U^2 = \max\{0, s_W^2 - \hat{\sigma}_X^2\}$, and $\hat{\sigma}_\varepsilon^2 = \max\{0, s_Y^2 - \hat{\beta}_1^2\hat{\sigma}_X^2\}$, where s_{WY} denotes the sample covariance, and s_W^2 and s_Y^2 denote the sample variances.

Method	$\hat{\beta}_0$	$\hat{\beta}_1$	$\hat{\sigma}_X^2$	$\hat{\sigma}_U^2$	$\hat{\sigma}_\varepsilon^2$
Naive	94.959	0.511	709.478	0	242.123
GMM	14.619	0.895	398.862	279.289	0
Phase	-40.776	1.157	313.066	396.411	0

Table 3.12: Naive, GMM, and phase function-based estimators of the linear errors-in-variables for the abrasiveness index data.

The results in Table 3.12 are striking. As one would expect, the naive estimator of slope is shrunk towards 0 when compared to the GMM and phase function estimators of slope. Both GMM and the phase function approach suggest that, as seen by the estimates of σ_U^2, the new method introduces a large amount of measurement error. On the other hand, the established YGP method has estimated measurement error 0. Due to the small sample size, we are hesitant to conclude that the YGP method is error free. However, the results do suggest that if the YGP method does introduce measurement error, it is small relative to the measurement error introduced by the new method. Any company considering adoption of the new method for measuring the abrasiveness index should whether the increased measurement error is worth the cost savings of the new method.

To assess the variability of the computed estimators, pairwise bootstrap resampling was used. A total of $B = 2000$ bootstrap samples were taken. Both GMM and the phase function method is prone to outliers in small samples. Subsequently, the interquartile ranges (IQR) of the respective bootstrap distributions were used as robust measures of spread. For GMM, $\text{IQR}^*(\beta_0) = 71.167$ and $\text{IQR}^*(\beta_1) = 0.335$. For the phase function method, $\text{IQR}^*(\beta_0) = 56.031$ and $\text{IQR}^*(\beta_1) = 0.271$. While this suggests that the phase function method gives less variable results, we should note that it is possible to choose a different measure of spread that contradicts this conclusion. Specifically, the difference between the 10th and 90th percentiles of the bootstrap distributions gives estimated spread 0.450 and 0.624 for the slope estimators using GMM and the phase function method respectively. Ultimately, for the data at hand, it is not possible to conclude that one method is superior to the other.

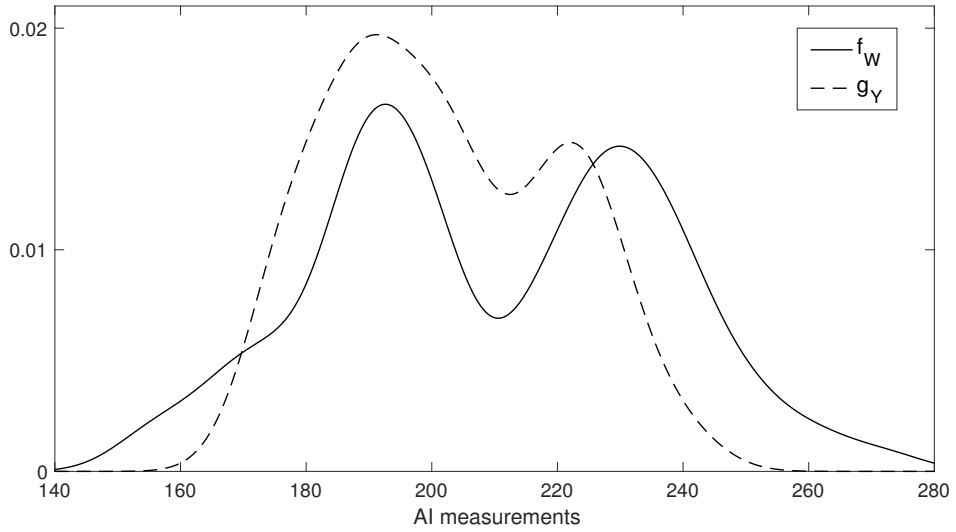


Figure 3.2: Kernel Density Estimators for W (new method) and Y (YGP method) data

Analysis of OPEN study

In this section, the relationships between true dietary intakes and various measurements like biomarkers, diary, and self-report instruments are studied. In the National Cancer's Institute OPEN study, two indicators of dietary intakes of interest include protein intake and energy intake. For each indicator, each intake was measured by a food frequency questionnaire (FFQ), a 24-hour recall interview, and a biomarker. Each measurement is replicated twice. The dataset is used to illustrate several examples of measurement error modeling in Carroll et al. (2006). The data made available on the website of the cited monograph is not the actual data from the OPEN Study, but has been simulated to have similar properties to the true data. These are $n = 223$ observations in this dataset.

For each indicator, the fitted model is of the form $Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \beta_3 X_{3i} + \varepsilon_i$, and $W_{jik} = X_{jik} + U_{jik}$, $j = 1, 2, 3$, $i = 1, \ldots, n$, $k = 1, 2$, where Y_i is the true amount of the indicator, X_{1i}, X_{2i} and X_{3i} represent the (unobserved) amount of the indicator from biomarker measurement, FFQ, and interview of the i^{th} subject respectively. If there is no measurement error exists, all the values X_{1i}, X_{2i} and X_{3i} would be equal to the value of Y_i. However, the observed data W_{jik} are all different from Y_i, showing measurement error exists in all of the measurements.

The estimators that are computed include the naive estimator, the simulation - extrapolation (SIMEX) estimator, and the phase function estimator. All the estimators are computed based on Y_i and $W_{ji} = \frac{1}{2}(W_{ji1} + W_{ji2})$. Note that the SIMEX estimator requires knowledge of the variance of the measurement errors, which is possible to estimate in this situation because replication data for each measurement is available. The variance of measurement error associated with W_{ji} was computed as

$$\hat{\sigma}_j^2 = \frac{1}{2n(n-1)} \sum_{i=1}^{n} (W_{ji1} - W_{ji2})^2.$$

The phase function estimator was computed by minimizing the statistic

$$D = \int_0^{t^*} \left(\sum_{i=1}^{n} \sum_{j=1}^{n} \sin \left[t \left(Y_j - W_{1i}\beta_{1i} - W_{2i}\beta_{2i} - W_{3i}\beta_{3i} \right) \right] \right)^2 K_{t^*}(t) dt.$$

with $K(t) = (1 - |t|)^2$ and t^* being the smallest $t > 0$ such that $|\hat{\phi}_Y(t)| \leq n^{-1/4}$. This minimization problem is nonconvex, so the numerical algorithm was started at numerous points around the naive estimate. The estimates and its estimated standard error (in parentheses) for both protein and energy intake were given in the Table 3.13. The standard error for the phase function and the SIMEX estimates was computed to be the interquartile range (IQR) of the corresponding estimates from $B = 100$ bootstrap samples, while the standard error for the naive was computed using the traditional Fisher information matrix.

	Measurement	Naive	SIMEX	Phase Function
Protein	FFQ	0.041 (0.022)	0.194 (0.068)	-0.072 (0.306)
	24-hour recall	0.041 (0.022)	0.051 (0.036)	0.127 (0.243)
	Biomarker	0.587 (0.037)	1.018 (0.138)	0.836 (0.400)
Energy	FFQ	0.006 (0.008)	0.003 (0.022)	0.133 (0.148)
	24-hour recall	0.006 (0.010)	0.007 (0.037)	0.226 (0.256)
	Biomarker	0.932 (0.017)	0.986 (0.028)	0.859 (0.268)

Table 3.13: Analysis of different measurements in the OPEN study

The results from Table 3.13 show that for both protein and energy intake, only biomarker measurements have significant effect on the true amount. In the case of pro-

tein intake, the naive estimates attentuates the effect of the biomarker considerably, while the SIMEX and phase function estimates are able to correct it. In the case of energy intake, the phase function estimate reduces the magnitude of the relationship between biomarker measurement and the true amount. Compared to the SIMEX estimate, the phase function estimator has a much higher standard error. This is expected because the SIMEX estimator uses knowledge of the measurement error variances, while the phase function estimator does not.

Chapter 4
Simulation-Selection-Extrapolation: Estimation in High-Dimensional Errors-in-Variables Model

4.1. Overview

Errors-in-variables models in a high-dimensional setting pose two challenges that need to be addressed. Firstly, the number of observed covariates is larger than the sample size, and only a small number of covariates are true predictors. Secondly, the presence of measurement error can result in severely biased parameter estimates, and also affects the ability of penalized methods such as the lasso to recover the true sparsity pattern. A new estimation procedure called SIMSELEX (SIMulation-SELection-EXtrapolation) is proposed. This procedure makes double use of lasso methodology. Firstly, the lasso is used to estimate sparse solutions in the simulation step, after which a variable selection step based on the group lasso is implemented. The SIMSELEX estimator is shown to perform well in variable selection, and has significantly lower estimation error than naive estimators that ignore measurement error. SIMSELEX can be applied in a variety of errors-in-variables settings, including linear models, generalized linear models, and Cox survival models. It is furthermore shown how SIMSELEX can be applied to spline-based regression models. A simulation study is conducted to compare the SIMSELEX estimators to existing methods in the linear and logistic model settings. Finally, the method is used to analyze a microarray dataset that contains gene expression measurements of favorable histology Wilms tumors.

4.2. Introduction

Errors-in-variables models arise in settings where some covariates cannot be measured with great accuracy. As such, the observed covariates tend to have larger variance than

the true underlying variables, obscuring the relationship between true covariates and outcome. We consider the problem in the classic additive measurement error framework. The work is motivated by microarray studies in which measurements are taken for a large number of genes, and it is of interest to identify genes related to some outcome of interest. Analysis after applying a log-transformation to the strictly positive gene expression measurements makes the assumption of additive measurement error more realistic. Microarray studies tend to have both noisy measurements and small sample sizes (relative to the number of genes measured). Biological variation in the data is usually of primary interest to investigators, but is obscured by technical variation resulting from sources such as sample preparation, labeling, and hybridization, see Zakharkin et al. (2005). As such, methodology dealing with measurement error in a large-dimensional setting is needed to identify genes related to the outcome of interest. Assuming that only a small number of genes are related to the outcome of interest further requires sparsity of the solution. One example of such a dataset is the favorable histology Wilms tumors analyzed by Sørensen et al. (2015). In this study, Affymetric microarray gene expression measurements are used to identify genes associated with relapse within three years of successful treatment.

Formalizing the problem, let a response variable $Y \in \mathbb{R}$ be related to a function of covariates $\boldsymbol{X} \in \mathbb{R}^p$. However, the observed sample consists of measurements $(\boldsymbol{W}_1, Y_1), \ldots, (\boldsymbol{W}_n, Y_n)$, with $\boldsymbol{W}_i = \boldsymbol{X}_i + \boldsymbol{U}_i$, $i = 1, \ldots, n$ where the measurement error components $\boldsymbol{U}_i \in \mathbb{R}^p$ are *i.i.d.* Gaussian with mean zero and covariance matrix $\boldsymbol{\Sigma}_u$. The $\boldsymbol{U}_i$ are assumed independent of the true covariates $\boldsymbol{X}_i$, and the matrix $\boldsymbol{\Sigma}_u$ is assumed known or estimable from auxiliary data. This paper will consider models that specify (at least partially) a distribution for Y conditional on $\boldsymbol{X}$ involving unknown parameters $\boldsymbol{\theta}$. Such models include generalized linear models, Cox survival models, and spline-based regression models. Not accounting for measurement error when fitting these models can result in biased parameter estimates as well as a loss of power when detecting relationships between variables, see Carroll et al. (2006). The effects of measurement error have mostly been studied in the low-dimensional setting where the sample size n is larger than the number of covariates p, see Armstrong (1985) for generalized linear models and Prentice

(1982) for Cox survival models. Ma and Li (2010) also studied variable selection in the measurement error context using penalized estimating equations.

We consider these models in the high-dimensional setting where p can be much larger than n. The true $\boldsymbol{\theta}$ is assumed sparse, having only $d < \min(n, p)$ non-zero components. Of interest is both recovery of the true sparsity pattern as well as estimating the non-zero components of $\boldsymbol{\theta}$. When the covariates $\boldsymbol{X}$ are observed without error, the lasso and its generalizations as proposed by Tibshirani (1996) can be employed for estimating a sparse $\boldsymbol{\theta}$. The lasso adds an ℓ_1 constraints on $\boldsymbol{\theta}$ to a loss function $\mathcal{L}(\boldsymbol{\theta}; Y, \boldsymbol{X})$. The estimator $\widehat{\boldsymbol{\theta}}$ is defined to be

$$\widehat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \left[\mathcal{L}(\boldsymbol{\theta}; Y, \boldsymbol{X}) + \xi_1 \left\|\boldsymbol{\theta}\right\|_1\right] \tag{4.1}$$

where ξ_1 is a tuning parameter and $\left\|\boldsymbol{\theta}\right\|_1 = \sum_{j=1}^{p} |\theta_j|$ is the ℓ_1 norm, with θ_j being the jth component of $\boldsymbol{\theta}$. For the generalized linear model, $\mathcal{L}(\boldsymbol{\theta}; Y, \boldsymbol{X})$ is often chosen as the negative log-likelihood function, while for the Cox survival model, $\mathcal{L}(\boldsymbol{\theta}; Y, \boldsymbol{X})$ is the negative log of the partial likelihood function, see Hastie et al. (2015) for details.

In high dimensional settings, the presence of measurement error can have severe consequences on the lasso estimator: the number of non-zero estimates can be inflated, sometimes dramatically, and as such the true sparsity pattern is not recovered Rosenbaum et al. (2010); see Appendix 4.8.1 for an illustration. To correct for measurement error in the high-dimensional setting, Rosenbaum et al. (2010) proposed a matrix uncertainty selector (MU) for linear models. Rosenbaum et al. (2013) proposed an improved version of the MU selector, while Belloni et al. (2017) proved its near-optimal minimax properties and developed a conic programming estimator that can achieve the minimax bound. The conic estimators require selection of three tuning parameters, a difficult task in practice. Another approach for handling measurement error is to modify the loss and conditional score functions used with the lasso, see Loh and Wainwright (2011), Sørensen et al. (2015) and Datta et al. (2017). Additionally, Sørensen et al. (2018) developed the generalized matrix uncertainty selector (GMUS) for generalized linear models. Both the conditional score approach and GMUS require subjective choices of tuning parameters.

This chapter proposes a new method of estimation called Simulation - Selection -

Extrapolation (SIMSELEX). This method is based on the SIMEX procedure of Cook and Stefanski (1994) which has been well-studied for correcting Normal measurement error in low-dimensional models, see for example Stefanski and Cook (1995), Küchenhoff et al. (2006) and Apanasovich et al. (2009). A SIMEX procedure for Laplace measurement error was studied by Koul et al. (2014) who considered a single covariate measured with error. Yi et al. (2015) combined SIMEX with a generalized estimating equation approach for variable selection on longitudinal data with covariate measurement error. Their variable selection step is carried out after the extrapolation step and requires a weight matrix to be prespecified.

To achieve model sparsity, the SIMSELEX approach augments SIMEX with a variable selection step (based on the group lasso) performed after the simulation step and before the extrapolation step. This means that lasso-type methodology is applied twice in SIMSELEX, once to obtain a sparse solution in the simulation step, and then again in the variable selection step. The procedure inherits the flexibility of SIMEX and can be applied to a variety of different high-dimensional errors-in-variables models.

The remainder of this paper is organized as follows. In Section 4.3, the SIMSELEX procedure for the high-dimensional setting is developed. In Section 4.4, application of SIMSELEX is illustrated for linear, logistic, and Cox regression models. Section 4.5 demonstrates the application of SIMSELEX in spline nonparametric regression. In Section 4.6, the methodology is illustrated with the favorable histology Wilms tumor data. Section 4.7 contains concluding remarks.

4.3. The SIMSELEX Estimator

Let $\boldsymbol{X}_i$ denote a vector of covariates, let $\boldsymbol{W}_i = \boldsymbol{X}_i + \boldsymbol{U}_i$ denote the covariates contaminated by measurement error $\boldsymbol{U}_i$ independent of $\boldsymbol{X}_i$, and let Y_i denote an outcome variable depending on $\boldsymbol{X}_i$ in a known way through parameter vector $\boldsymbol{\theta}$. The measurement error $\boldsymbol{U}_i$ is assumed to be multivariate Gaussian with mean zero and known covariance matrix $\boldsymbol{\Sigma}_u$. The observed data are pairs $(\boldsymbol{W}_i, Y_i)$, $i = 1, \ldots, n$. While the outcomes Y_i depend on the true covariates $\boldsymbol{X}_i$, only the observed $\boldsymbol{W}_i$ are available for model estimation. Now,

let S denote a method for estimating $\boldsymbol{\theta}$. If the uncontaminated $\boldsymbol{X}_i$ had been observed, we could calculate the *true estimator* $\widehat{\boldsymbol{\theta}}_{\text{true}} = S(\{\boldsymbol{X}_i, Y_i\}_{i=1,\ldots,n})$. The *naive estimator* of $\boldsymbol{\theta}$ based on the observed sample is $\widehat{\boldsymbol{\theta}}_{\text{naive}} = S(\{\boldsymbol{W}_i, Y_i\}_{i=1,\ldots,n})$ and treats the $\boldsymbol{W}_i$ as if no measurement error is present. Generally, the naive estimator is neither consistent nor unbiased for $\boldsymbol{\theta}$.

A SIMEX estimator of $\boldsymbol{\theta}$ was proposed by Stefanski and Cook (1995). In the *simulation step*, a grid of values $0 < \lambda_1 < \ldots < \lambda_M$ is chosen. For each λ_m, B sets of pseudodata are generated by adding simulated random noise, $\boldsymbol{W}_i^{(b)}(\lambda_m) = \boldsymbol{W}_i + \lambda_m^{1/2}\boldsymbol{U}_i^{(b)}, \quad b = 1, \ldots, B$, with $\boldsymbol{U}^{(b)}$ having the same multivariate Gaussian distribution as $\boldsymbol{U}$. For each set of pseudodata, the naive estimator is calculated, $\widehat{\boldsymbol{\theta}}^{(b)}(\lambda_m) = S(\{\boldsymbol{W}_i^{(b)}(\lambda_m), Y_i\}_{i=1,\ldots,n})$. These naive estimators are then averaged, $\widehat{\boldsymbol{\theta}}(\lambda_m) = B^{-1}\sum_{b=1}^{B}\widehat{\boldsymbol{\theta}}^{(b)}(\lambda_m)$. In the *extrapolation step* $\widehat{\boldsymbol{\theta}}(\lambda)$ is modeled as a function of λ using a suitable function and extrapolated to $\lambda = -1$, which corresponds to the error-free case and gives estimator $\widehat{\boldsymbol{\theta}}_{\text{simex}}$.

Unfortunately SIMEX as described above cannot be applied to the high-dimensional setting without some adjustments. Even if method S enforces sparsity of $\widehat{\boldsymbol{\theta}}^{(b)}(\lambda_m)$ for a given set of pseudodata, this does not guarantee sparsity of the average $\widehat{\boldsymbol{\theta}}(\lambda_m)$, or a consistent sparsity pattern across values of λ_m. Let $(\lambda_m, \widehat{\theta}_j(\lambda_m))$, $m = 1, \ldots, M$, denote the solution path for the θ_j, the jth component of $\boldsymbol{\theta}$, and assume $\theta_j = 0$. If $\widehat{\theta}_j(\lambda_i) \neq 0$ for even a single λ_i, it will result in an extrapolated value $\widehat{\theta}_j(-1) \neq 0$. In this way, many components of the extrapolated solution could be non-zero. The SIMSELEX (SIMulation-SELection-EXtrapolation) algorithm, presented below, addresses solution sparsity. Fundamental to the SIMSELEX approach is a *double-use* of the lasso: it is used for parameter estimation in the simulation step to ensure solution sparsity for a given set of pseudodata, and in the selection step to determine which covariates to include in the model.

4.3.1. Simulation step

The simulation step of SIMSELEX is identical to the simulation step of SIMEX. However, the criterion function being minimized for each set of pseudodata now incorporates

a lasso-type penalty on the model parameters. For given value of λ and corresponding pseudodata $(\boldsymbol{W}_i^{(b)}(\lambda), Y_i)$, $i = 1, \ldots, n$, the estimator $\widehat{\boldsymbol{\theta}}^{(b)}(\lambda)$ is calculated according to a criterion of the form in (4.1) with the tuning parameter $\xi_1^{(\lambda,b)}$. Note that cross-validation is implemented separately for each set of pseudodata. Two popular choices for the tuning parameter are $\xi_{\min}$, the value that minimizes the estimated prediction risk, and $\xi_{1\text{se}}$, the value that makes the estimated prediction risk fall within one standard error of the minimum (one-se-rule), see Friedman et al. (2001). The simulation step results in pairs $(\lambda_m, \widehat{\boldsymbol{\theta}}(\lambda_m))$, $m = 1, \ldots, M$, which are then used in the selection and extrapolation steps described next.

4.3.2. Selection step

Variable selection is performed by applying a version of the group lasso of Yuan and Lin (2006) to the pairs $(\lambda_m, \widehat{\boldsymbol{\theta}}(\lambda_m))$. It is assumed that the quadratic function serves as a good approximation to this relationship. Now, letting $\widehat{\theta}_{mj} = \widehat{\theta}_j(\lambda_m)$, it follows that

$$\widehat{\theta}_{mj} = \gamma_{0j} + \gamma_{1j}\lambda_m + \gamma_{2j}\lambda_m^2 + e_{mj}, \quad m = 1, \ldots, M, \quad j = 1, \ldots, p, \tag{4.2}$$

with e_{mj} denoting zero-mean error terms. To achieve model sparsity, it is desirable to shrink (as a group) the parameters $(\gamma_{0j}, \gamma_{1j}, \gamma_{2j})$ to the vector $(0, 0, 0)$ for many of the components θ_j. Extrapolation will then only be applied to the variables with non-zero solutions $(\widehat{\gamma}_{0j}, \widehat{\gamma}_{1j}, \widehat{\gamma}_{2j})$, with all other coefficients being set equal to 0. If the true model is sparse, many of the solutions $(\widehat{\gamma}_{0j}, \widehat{\gamma}_{1j}, \widehat{\gamma}_{2j})$ will be shrunk to the zero vector.

The p equations in (4.2) can be written in matrix form, $\boldsymbol{\Theta} = \boldsymbol{\Lambda}\boldsymbol{\Gamma} + \boldsymbol{E}$, where

$$\boldsymbol{\Lambda} = \begin{bmatrix} 1 & \lambda_1 & \lambda_1^2 \\ \vdots & \vdots & \vdots \\ 1 & \lambda_M & \lambda_M^2 \end{bmatrix}, \quad \boldsymbol{\Theta} = \begin{bmatrix} \widehat{\theta}_{11} & \ldots & \widehat{\theta}_{1p} \\ \vdots & & \vdots \\ \widehat{\theta}_{M1} & \ldots & \widehat{\theta}_{Mp} \end{bmatrix},$$

$$\boldsymbol{\Gamma} = \begin{bmatrix} \gamma_{01} & \dots & \gamma_{0p} \\ \gamma_{11} & \dots & \gamma_{1p} \\ \gamma_{21} & \dots & \gamma_{2p} \end{bmatrix} \quad \text{and} \quad \boldsymbol{E} = \begin{bmatrix} e_{11} & \dots & e_{1p} \\ \vdots & & \vdots \\ e_{M1} & \dots & e_{Mp}. \end{bmatrix}.$$

When the kth column of the estimated matrix $\widehat{\boldsymbol{\Gamma}}$ is a zero vector, the corresponding kth column of $\widehat{\boldsymbol{\Theta}} = \boldsymbol{\Lambda}\widehat{\boldsymbol{\Gamma}}$ will also be a zero vector and the kth variable is not selected for inclusion in the model. In the present context, the group lasso penalized discrepancy function

$$D(\boldsymbol{\Gamma}) = \frac{1}{2}\sum_{m=1}^{M}\sum_{j=1}^{p}\left(\widehat{\theta}_{mj} - \gamma_{0j} - \gamma_{1j}\lambda_m - \gamma_{2j}\lambda_m^2\right)^2 + \xi_2\left(\sum_{j=1}^{p}\sqrt{\gamma_{0j}^2 + \gamma_{1j}^2 + \gamma_{2j}^2}\right)$$

is used with ξ_2 a tuning parameter. This function can be written in matrix form,

$$D(\boldsymbol{\Gamma}) = \frac{1}{2}\sum_{j=1}^{p}\left(\|\boldsymbol{\Theta}_j - \boldsymbol{\Lambda}\boldsymbol{\Gamma}_j\|_2^2 + \xi_2\|\boldsymbol{\Gamma}_j\|_2\right) \tag{4.3}$$

where $\boldsymbol{\Theta}_j$ and $\boldsymbol{\Gamma}_j$ denote the jth columns of $\boldsymbol{\Theta}$ and $\boldsymbol{\Gamma}$ respectively, and $\|.\|_2$ denotes the ℓ_2 norm.

Group lasso variable selection is illustrated in the left plot of Figure 4.1 where each path represents the ℓ_2 norm of a column of $\widehat{\Gamma}$ as a function of ξ_2 in the Wilms tumor data example. Note that only eight of 2074 paths are shown. A larger value of ξ_2 sets more coefficients to zero. The cross-validation (one-se rule) value of ξ_2 is also shown.

To find $\widehat{\boldsymbol{\Gamma}}$ that minimizes D, standard numerical subgradient methods can be used. As equation (4.3) is block-separable and convex, subgradient methods will converge to the global minimum. The subgradient equations (Hastie et al., 2015, Section 5.2.2) are

$$-\boldsymbol{\Lambda}^T\left(\boldsymbol{\Theta}_j - \boldsymbol{\Lambda}\widehat{\boldsymbol{\Gamma}}_j\right) + \xi_2\widehat{\boldsymbol{s}}_j = 0, \quad j = 1, \dots, p, \tag{4.4}$$

where $\widehat{\boldsymbol{s}}_j \in \mathbb{R}^3$ is an element of the subdifferential of the norm $\left\|\widehat{\boldsymbol{\Gamma}}_j\right\|_2$. As a result, if $\widehat{\boldsymbol{\Gamma}}_j \neq \boldsymbol{0}$, then $\widehat{\boldsymbol{s}}_j = \widehat{\boldsymbol{\Gamma}}_j / \left\|\widehat{\boldsymbol{\Gamma}}_j\right\|_2$. On the other hand, if $\widehat{\boldsymbol{\Gamma}}_j = \boldsymbol{0}$, then $\widehat{\boldsymbol{s}}_j$ is any vector with

$\|\widehat{\boldsymbol{s}}_j\|_2 \leq 1$. Therefore, $\widehat{\boldsymbol{\Gamma}}_j$ must satisfy

$$\widehat{\boldsymbol{\Gamma}}_j = \begin{cases} \mathbf{0} & \text{if } \left\|\boldsymbol{\Lambda}^\top \boldsymbol{\Theta}_j\right\|_2 \leq \xi_2 \\ \left[\boldsymbol{\Lambda}^\top \boldsymbol{\Lambda} + \dfrac{\xi_2}{\left\|\widehat{\boldsymbol{\Gamma}}_j\right\|_2} \boldsymbol{I}\right]^{-1} \boldsymbol{\Lambda}^\top \boldsymbol{\Theta}_j & \text{otherwise.} \end{cases} \tag{4.5}$$

The first equation of (4.5) gives a simple rule for when to set the all elements of a specific column of $\widehat{\boldsymbol{\Gamma}}$ equal to 0 for a specific value of ξ_2. Therefore $\widehat{\boldsymbol{\Gamma}}$ can be computed using proximal gradient descent (Hastie et al., 2015, Section 5.3). At the kth iteration, each column of $\widehat{\boldsymbol{\Gamma}}_j$ can be updated by first calculating $\omega_j^{(k)} = \widehat{\boldsymbol{\Gamma}}_j^{(k-1)} + \nu \boldsymbol{\Lambda}^\top (\boldsymbol{\Theta}_j - \boldsymbol{\Lambda} \widehat{\boldsymbol{\Gamma}}_j^{(k-1)})$ and then using this quantity to update

$$\widehat{\boldsymbol{\Gamma}}_j^{(k)} = \left(1 - \frac{\nu \xi_2}{\left\|\omega_j^{(k)}\right\|_2}\right)_+ \omega_j^{(k)}$$

for all $j = 1, \ldots, p$. Here, ν is the step size that needs to be specified for the algorithm and $(z)_+ = \max(z, 0)$. The convergence of the algorithm is guaranteed if the step size $\nu \in (0, 1/L)$, where L is the maximum eigenvalue of the matrix $\boldsymbol{\Lambda}^{\boldsymbol{T}} \boldsymbol{\Lambda}/M$. The parameter ξ_2 can be chosen using cross-validation. The algorithm stops when the distance between the current estimate $\widehat{\boldsymbol{\Gamma}}^{(k)}$ and the previous estimate $\widehat{\boldsymbol{\Gamma}}^{(k-1)}$ is smaller than some tolerance level, say 10^{-4}.

Variable selection here assumes a quadratic approximation holds in (4.2). The method as described could also be used assuming a linear relationship. However, the nonlinear means function often used in SIMEX is unsuitable for selection as described here as would result in a non-convex loss function which would be very expensive computationally when paired with a lasso-type penalty.

4.3.3. Extrapolation step

The extrapolation step of SIMSELEX is identical to that of SIMEX, but with extrapolation only applied to the selected variables. Thus, if the jth variable has been selected for inclusion in the model, an extrapolation function $\Gamma_{\text{ex}}(\lambda)$ is fit to the simulation-step

pairs $(\lambda_m, \widehat{\theta}_j(\lambda_m))$. Let $\hat{\Gamma}_{\text{ex},j}(\lambda)$ denote the extrapolation function fit obtained for the coefficient path of variable j. The SIMSELEX estimate is then given by $\hat{\theta}_j = \hat{\Gamma}_{\text{ex},j}(-1)$. Two common extrapolation functions are the quadratic and nonlinear means models, respectively $\Gamma_{\text{quad}}(\lambda) = \gamma_0 + \gamma_1\lambda + \gamma_2\lambda^2$ and $\Gamma_{\text{nonlin}}(\lambda) = \gamma_0 + \gamma_1/(\gamma_2 + \lambda)$. Note that the extrapolation step does not directly incorporate any model penality, but the coefficient paths being used for extrapolation did result from fitting a penalized model in the simulation step.

The right plot in the Figure 4.1 illustrates the simulation and extrapolation steps of SIMSELEX. For four genes selected in the Wilms tumor example, the plotted points represents the coefficients resulting from added measurement error level λ, and the dotted lines illustrate quadratic extrapolation to $\lambda = -1$.

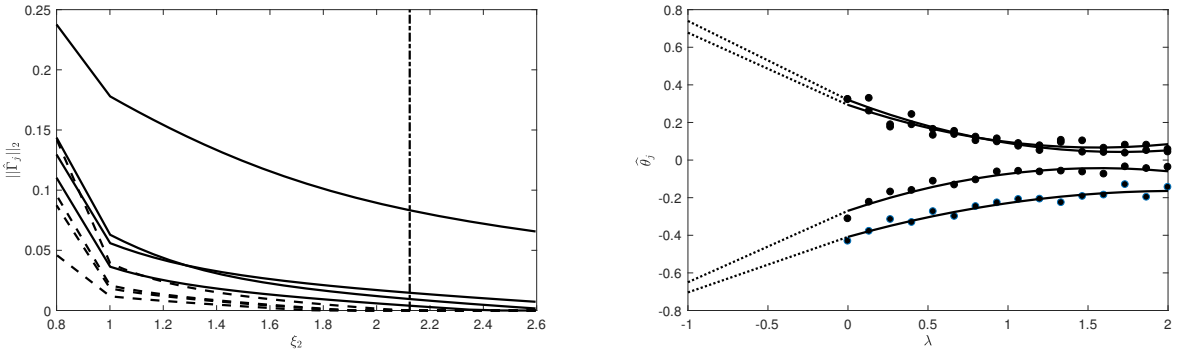


Figure 4.1: SIMSELEX illustration using microarray data (Section 5). Left figure: solid and dashed lines represent the norms $||\widehat{\Gamma}_j||_2$ of, respectively, the selected and (some) unselected genes; the vertical dash-dot line is the one-se cross-validation tuning parameter. Right figure: coefficients of selected genes are modeled quadratically in λ and then extrapolated to $\lambda = -1$.

4.4. Model Illustration and Simulation Results

The performance of SIMSELEX in high-dimensional errors-in-variables models is discussed in this section for linear, logistic, and Cox regression models. Where applicable, the performance of competitor estimators is also included. The results of extensive simulation studies are also reported. Three performance metrics related to the recovery of the sparsity pattern and also the estimation error associated with parameter recovery were considered. Simulations assumed a known measurement error covariance matrix.

4.4.1. Linear Regression

For observed data $(\boldsymbol{W}_i, Y_i)$, $i = 1, \ldots, n$ let $Y_i = \boldsymbol{X}_i^\top \boldsymbol{\theta} + \varepsilon_i$ and $\boldsymbol{W}_i = \boldsymbol{X}_i + \boldsymbol{U}_i$. For this linear models with high-dimensional covariates subject to measurement error, three solutions have been proposed in the literature. Rosenbaum et al. (2010) proposed the Matrix Uncertainty Selection (MUS), which does not require knowledge of the measurement error covariance matrix $\boldsymbol{\Sigma}_u$. The other two approaches that do make use of $\boldsymbol{\Sigma}_u$; Sørensen et al. (2015) developed a corrected scores lasso, while Belloni et al. (2017) proposed a conic programming estimator. The corrected scores lasso requires the selection of one tuning parameter, while conic programming estimator requires three tuning parameters. A brief overview of the latter two approaches is given in Appendix 4.8.2.

For the simulation study, data pairs $(\boldsymbol{W}_i, Y_i)$ were generated from above linear model. The true covariates $\boldsymbol{X}_i$ were generated to be *i.i.d.* p-variate Gaussian with mean $\mathbf{0}$ and covariance matrix $\boldsymbol{\Sigma}$), the latter having entries $\Sigma_{ij} = \rho^{|i-j|}$ with $\rho = 0.25$. The p components of each measurement error vector $\boldsymbol{U}_i$ were generated to be either *i.i.d.* Gaussian or Laplace with mean 0 and variance σ_u^2, so that $\boldsymbol{U}_i$ has mean $\mathbf{0}$ and covariance matrix $\boldsymbol{\Sigma}_u = \sigma_u^2 I_{p\times p}$. Two values were considered for the measurement error variance, $\sigma_u^2 \in \{0.15, 0.30\}$. As SIMSELEX assumes normality of the measurement error, the Laplace distribution setting was chosen in part to evaluate model robustness. The error components ε_i were simulated to be *i.i.d.* univariate normal, $\varepsilon \sim N(\mathbf{0}, \sigma_\varepsilon^2)$ with $\sigma_\varepsilon^2 = 0.256^2$. The sample size was fixed at $n = 300$, and simulations were done for the number of covariates $p \in \{500, 1000, 2000\}$. Two choice of the true $\boldsymbol{\theta}$ were considered, namely $\boldsymbol{\theta}_1 = (2, 1.75, 1.5, 1.25, 1.0, 0, \ldots, 0)^\top$ and $\boldsymbol{\theta}_2 = (1, 1, 1, 1, 1, 0, \ldots, 0)^\top$. Both cases have $d = 5$ non-zero coefficients. Under each simulation configuration considered, $N = 500$ samples were generated.

Above simulation settings correspond to noise-to-signal ratios of approximately 15% and 30% for each individual covariate. However, in multivariate space a metric such as the proportional increase in total variability, $\Delta V = (\det(\boldsymbol{\Sigma}_W) - \det(\boldsymbol{\Sigma})) \,/\, \det(\boldsymbol{\Sigma})$, is more informative. When $\sigma_u^2 = 0.15$, if one were to only observe the $d = 5$ non-zero covariates, $\Delta V = 1.145$, while for $p = 500$, this metric becomes $\Delta V = 6.79 \times 10^{33}$. When $\sigma_u^2 = 0.3$,

the equivalent values are $\Delta V = 3.132$ for $d = 5$ and $\Delta V = 5.79 \times 10^{62}$ for $p = 500$. The dramatic increase of ΔV emphasizes the severe consequences of measurement error in high-dimensional space.

In the simulation study, five different estimators were computed: the true lasso using the uncontaminated $\boldsymbol{X}$-data, the naive lasso treating the $\boldsymbol{W}$-data as if it were uncontaminated, the conic estimator with tuning parameters as implemented in Belloni et al. (2017), the corrected lasso with the tuning parameter chosen based on 10-fold cross-validation, and SIMSELEX.

SIMSELEX used $M = 5$ equi-spaced λ values from 0.01 to 2. For each λ, $B = 100$ sets of pseudodata were generated. The tuning parameter of the lasso was chosen using the one-se rule and 10-fold cross-validation. For group lasso selection, $\nu = (20L)^{-1}$ was used as step size with L the maximum eigenvalue of matrix $\boldsymbol{\Lambda}^\top \boldsymbol{\Lambda}/M$. The lasso was implemented using the `glmnet` function in MATLAB, see Qian et al. (2013). The group lasso was implementing using our own code, available online with this paper. Both quadratic and nonlinear means extrapolation functions were fitted. Only the quadratic extrapolation results are reported in this section as it has smaller ℓ_2 error that nonlinear extrapolation. The nonlinear means extrapolation results can be found in Appendix 4.8.4.

The five estimators were compared using the average estimation error

$$\ell_2 = \sqrt{\sum_{j=1}^{p} (\widehat{\theta}_j - \theta_j)^2}.$$

Furthermore, each method's ability to recover the true sparsity pattern was evaluated using the average number of false positive (FP) and false negative (FN) estimates per simulated dataset. Note that the conic estimator does not set any estimates exactly equal to 0 and cannot be used for variable selection. The simulation results for parameter vector $\boldsymbol{\theta}_1$ are presented in Table 4.1, while the results for $\boldsymbol{\theta}_2$ are presented in Table 4.7.

p	Est	$\sigma_u^2 = 0.15$						$\sigma_u^2 = 0.30$					
		Normal			Laplace			Normal			Laplace		
		ℓ_2	FP	FN	ℓ_2	FP	FN	ℓ_2	FP	FN	ℓ_2	FP	FN
500	True	0.09	0.98	0.00	0.09	0.81	0.00	0.09	0.82	0.00	0.09	0.83	0.00
		(0.02)	(2.06)	(0.00)	(0.02)	(1.56)	(0.00)	(0.02)	(1.86)	(0.00)	(0.02)	(1.65)	(0.00)
	Naive	0.73	1.36	0.00	0.74	0.99	0.00	1.11	1.48	0.00	1.12	1.12	0.00
		(0.08)	(3.3)	(0.00)	(0.08)	(2.21)	(0.00)	(0.1)	(3.29)	(0.00)	(0.1)	(2.24)	(0.00)
	SIMSELEX	0.32	0.00	0.00	0.34	0.00	0.00	0.5	0.00	0.00	0.52	0.00	0.00
		(0.1)	(0.00)	(0.00)	(0.11)	(0.00)	(0.00)	(0.14)	(0.00)	(0.00)	(0.16)	(0.04)	(0.00)
	Conic	0.37	-	-	0.38	-	-	0.52	-	-	0.53	-	-
		(0.07)	-	-	(0.06)	-	-	(0.1)	-	-	(0.1)	-	-
	Corrected	0.43	2.3	0.00	0.44	1.76	0.00	0.62	2.74	0.00	0.63	2.1	0.00
		(0.08)	(5.27)	(0.00)	(0.08)	(3.51)	(0.00)	(0.11)	(4.93)	(0.00)	(0.11)	(3.88)	(0.00)
1000	True	0.09	1.27	0.00	0.09	1.06	0.00	0.09	1.01	0.00	0.09	1.06	0.00
		(0.02)	(2.55)	(0.00)	(0.02)	(2.18)	(0.00)	(0.02)	(2.04)	(0.00)	(0.02)	(2.22)	(0.00)
	Naive	0.75	1.69	0.00	0.76	1.18	0.00	1.14	1.38	0.00	1.15	1.39	0.00
		(0.08)	(3.29)	(0.00)	(0.08)	(2.72)	(0.00)	(0.1)	(3.03)	(0.00)	(0.11)	(3.16)	(0.00)
	SIMSELEX	0.33	0.00	0.00	0.35	0.00	0.00	0.51	0.00	0.00	0.53	0.00	0.00
		(0.11)	(0.00)	(0.00)	(0.12)	(0.00)	(0.00)	(0.15)	(0.00)	(0.04)	(0.16)	(0.00)	(0.04)
	Conic	0.39	-	-	0.4	-	-	0.55	-	-	0.56	-	-
		(0.07)	-	-	(0.07)	-	-	(0.1)	-	-	(0.1)	-	-
	Corrected	0.44	3.48	0.00	0.46	3.11	0.00	0.63	3.57	0.00	0.65	3.14	0.00
		(0.09)	(6.37)	(0.00)	(0.08)	(6.26)	(0.00)	(0.12)	(5.97)	(0.00)	(0.13)	(5.26)	(0.00)
2000	True	0.1	1.29	0.00	0.1	1.45	0.00	0.1	1.56	0.00	0.1	1.32	0.00
		(0.02)	(2.68)	(0.00)	(0.02)	(3)	(0.00)	(0.02)	(3.41)	(0.00)	(0.02)	(2.62)	(0.00)
	Naive	0.77	1.76	0.00	0.78	1.59	0.00	1.17	1.89	0.00	1.17	2.06	0.00
		(0.08)	(3.52)	(0.00)	(0.09)	(5.06)	(0.00)	(0.1)	(4.57)	(0.00)	(0.11)	(5.72)	(0.00)
	SIMSELEX	0.34	0.00	0.00	0.36	0.00	0.00	0.53	0.00	0.00	0.55	0.00	0.01
		(0.1)	(0.00)	(0.00)	(0.11)	(0.00)	(0.00)	(0.15)	(0.04)	(0.00)	(0.17)	(0.00)	(0.09)
	Conic	0.41	-	-	0.41	-	-	0.59	-	-	0.59	-	-
		(0.07)	-	-	(0.07)	-	-	(0.1)	-	-	(0.11)	-	-
	Corrected	0.45	4.91	0.00	0.47	3.88	0.00	0.64	5.42	0.00	0.66	3.83	0.00
		(0.08)	(7.66)	(0.00)	(0.09)	(7.12)	(0.00)	(0.12)	(8.11)	(0.00)	(0.13)	(5.99)	(0.00)

Table 4.1: Comparison of estimators for linear regression with with the case of $\boldsymbol{\theta_1}$ based on ℓ_2 estimation error, average number of false positives (FP) and false negatives (FN). Standard errors in parentheses.

As seen in Table 4.1, the naive estimator performs worst — it has ℓ_2 error often twice that of either the conic or SIMSELEX methods. The conic estimator has comparable performance to the SIMSELEX estimators, with SIMSELEX having slightly smaller ℓ_2 error for $\boldsymbol{\theta}_1$, and the conic estimator having slightly smaller ℓ_2 error for $\boldsymbol{\theta_2}$. Both the

conic and SIMSELEX estimators have smaller ℓ_2 error than the corrected scores lasso. Interestingly, the ℓ_2 error corresponding to the Normal and Laplace measurement error settings is quite similar. This suggests that SIMSELEX is robust to at least moderate departures from normality (for the simulation settings considered).

When considering the recovery of true sparsity pattern, the average number of false negatives are negligible for all methods. For the average number of false positives, the corrected lasso generally performs the worst, while SIMSELEX does not result in any false positive for the parameter specifications considered. Overall, Table 4.7 demonstrates that SIMSELEX can have performance superior to existing methods in the literature with regards to the performance metrics considered.

4.4.2. Logistic Regression

Assume the observed data $(\boldsymbol{W}_i, Y_i)$, $i = 1, \ldots, n$ are generated by $Y_i \sim \text{Bern}\left[F(\boldsymbol{X}_i^\top \boldsymbol{\theta})\right]$ and $\boldsymbol{W}_i = \boldsymbol{X}_i + \boldsymbol{U}_i$. The logistic regression model follows when $F(x) = \text{logit}^{-1}(x)$. Two solutions for performing logistic regression in a high-dimensional errors-in-variables setting exist in the literature. The conditional scores lasso approach of Sørensen et al. (2015) can be applied to GLMs. This method requires the covariance matrix $\boldsymbol{\Sigma}_u$ be known or estimable. Sørensen et al. (2018) proposed a Generalized Matrix Uncertainty Selector (GMUS) for sparse high-dimensional models with measurement error. The GMUS estimator does not make use of $\boldsymbol{\Sigma}_u$. These methods are reviewed in Appendix 4.8.2.

For the simulation study, data pairs $(\boldsymbol{W}_i, Y_i)$ were generated using $Y_i|\boldsymbol{X}_i \sim \text{Bern}(p_i)$ with $\text{logit}(p_i) = \boldsymbol{X}_i^\top \boldsymbol{\theta}$. The true covariates $\boldsymbol{X}_i$, measurement error components $\boldsymbol{U}_i$, coefficient vectors $\boldsymbol{\theta}$, and sample size were exactly as outlined for the linear model simulation, see Section 4.4.1. The true estimator, naive estimator, conditional scores lasso, and SIMSELEX estimator using both quadratic and nonlinear extrapolation were computed for each simulated dataset for $p \in \{500, 1000, 2000\}$. The GMUS estimator was only computed for the case $p = 500$; Sørensen et al. (2018) note that GMUS becomes too computationally expensive for large p. We attempted implementation for $p = 1000$ using the `hdme` package in R, but a run time exceeding 12 hours for one sample demonstrated

the impracticality of this method. For the conditional scores lasso, Sørensen et al. (2015) recommend using an elbow method to choose the tuning parameter. For the simulation study, an adapted elbow described in Appendix 4.8.2 was used to select the tuning parameter. This adapted method isn't usable in practice and does tend to give over-optimistic results for the corrected scores approach than one is likely to otherwise obtain. The performance metrics ℓ_2 error, and average number of false positives (FP) and false negatives (FN) were calculated to compare the estimators. The results for $\boldsymbol{\theta}_1$ are presented in Table 4.2, while the results for $\boldsymbol{\theta}_2$ are presented in Table 4.8.

p	Est	$\sigma_u^2 = 0.15$						$\sigma_u^2 = 0.30$					
		Normal			Laplace			Normal			Laplace		
		ℓ_2	FP	FN	ℓ_2	FP	FN	ℓ_2	FP	FN	ℓ_2	FP	FN
500	True	2.62	0.39	0.23	2.61	0.39	0.23	2.61	0.54	0.25	2.59	0.54	0.25
		(0.21)	(2.54)	(0.43)	(0.21)	(2.54)	(0.43)	(0.23)	(3)	(0.59)	(0.2)	(3)	(0.59)
	Naive	2.83	0.59	0.57	2.83	0.59	0.57	2.99	0.42	1.08	2.99	0.42	1.08
		(0.22)	(3.79)	(1.03)	(0.22)	(3.79)	(1.03)	(0.23)	(1.59)	(1.6)	(0.24)	(1.59)	(1.6)
	SIMSELEX	2.65	0.01	0.65	2.63	0.01	0.68	2.73	0.00	1.38	2.74	0.00	1.57
		(0.43)	(0.09)	(0.58)	(0.39)	(0.09)	(0.61)	(0.46)	(0.00)	(1.09)	(0.45)	(0.06)	(1.19)
	Cond	2.36	7.02	1.15	2.33	7.02	1.15	2.58	5.67	1.7	2.53	5.67	1.7
		(0.65)	(9.77)	(0.95)	(0.61)	(9.77)	(0.95)	(0.56)	(7.94)	(1.09)	(0.57)	(7.94)	(1.09)
	GMUS	2.67	0.21	0.21	2.87	0.21	0.21	2.75	0.66	0.22	2.74	0.66	0.22
		(0.08)	(0.52)	(0.41)	(0.07)	(0.52)	(0.41)	(0.08)	(0.98)	(0.42)	(0.08)	(0.98)	(0.42)
1000	True	2.65	0.36	0.26	2.64	0.36	0.26	2.64	0.61	0.3	2.64	0.61	0.3
		(0.19)	(1.38)	(0.48)	(0.21)	(1.38)	(0.48)	(0.21)	(3.38)	(0.55)	(0.21)	(3.38)	(0.55)
	Naive	2.86	0.7	0.63	2.85	0.7	0.63	3.01	0.78	1.25	3.01	0.78	1.25
		(0.22)	(3.62)	(1.11)	(0.22)	(3.62)	(1.11)	(0.24)	(4.53)	(1.74)	(0.23)	(4.53)	(1.74)
	SIMSELEX	2.67	0.00	0.72	2.65	0.00	0.71	2.76	0.00	1.59	2.77	0.00	1.61
		(0.44)	(0.06)	(0.64)	(0.41)	(0.06)	(0.64)	(0.46)	(0.00)	(1.14)	(0.42)	(0.00)	(1.21)
	Cond	2.44	8.82	1.18	2.46	8.82	1.18	2.62	7.53	1.75	2.64	7.53	1.75
		(0.63)	(11.22)	(0.99)	(0.66)	(11.22)	(0.99)	(0.59)	(11.2)	(1.05)	(0.57)	(11.2)	(1.05)
2000	True	2.66	0.75	0.33	2.65	0.75	0.33	2.65	0.89	0.35	2.65	0.89	0.35
		(0.22)	(3.7)	(0.66)	(0.21)	(3.7)	(0.66)	(0.22)	(3.39)	(0.6)	(0.23)	(3.39)	(0.6)
	Naive	2.88	0.56	0.68	2.87	0.56	0.68	3.02	0.84	1.23	3.03	0.84	1.23
		(0.2)	(2.68)	(1.1)	(0.22)	(2.68)	(1.1)	(0.23)	(4.63)	(1.71)	(0.23)	(4.63)	(1.71)
	SIMSELEX	2.70	0.01	0.78	2.68	0.01	0.80	2.77	0.00	1.76	2.80	0.00	1.79
		(0.41)	(0.08)	(0.67)	(0.42)	(0.06)	(0.68)	(0.44)	(0.04)	(1.25)	(0.44)	(0.00)	(1.20)
	Cond	2.52	12.04	1.29	2.52	12.04	1.29	2.75	10.58	1.85	2.71	10.58	1.85
		(0.63)	(14.4)	(0.94)	(0.65)	(14.4)	(0.94)	(0.62)	(15.6)	(1.11)	(0.61)	(15.6)	(1.11)

Table 4.2: Comparison of estimators for logistic regression with with the case of $\boldsymbol{\theta_1}$ based on ℓ_2 estimation error, average number of false positives (FP) and false negatives (FN). Standard errors in parentheses.

Table 4.2 shows that in terms of ℓ_2 estimation error, the SIMSELEX estimator always performs better than the naive estimator and in many configurations, SIMSELEX has performance close to the true estimator. The conditional scores lasso has the smallest ℓ_2 error of the methods that control for measurement error, sometimes even outperforming the true estimator. We believe this to be an artifact of how the tuning parameter is selected in the simulation study, and does not correspond to "real world" performance. Furthermore, in terms of variable selection, the conditional scores lasso has both the high-

est average number of false positives and false negatives in all the considered settings. On the other hand, the SIMSELEX estimator performs variable selection well. SIMSELEX has the lowest average number of false positives in all the cases considered, and has only slightly higher average number of false negatives than the true and naive estimator. In the case of $p = 500$, GMUS has larger ℓ_2 error than SIMSELEX and the conditional scores lasso. However, it has smallest average number of false negatives among all the estimators and a slightly larger number of average false positive than SIMSELEX. As in the linear model, performance of the estimators do not differ markedly for the Normal and Laplace measurement error settings. Again, this suggests some robustness to departure from the assumed normality of measurement error in SIMSELEX.

4.4.3. Cox Proportional Hazard Model

The Cox proportional hazard model is commonly used for the analysis of survival data. It is assumed that the random failure time T has conditional hazard function $h(t|\boldsymbol{X}) = h_0(t)\exp(\boldsymbol{X}^\top\boldsymbol{\theta})$ where $h_0(t)$ is the baseline hazard function. Survival data is frequently subject to censoring in practice. It is therefore assumed that the observed data are of the form $(\boldsymbol{W}_i, Y_i, I_i)$, $i = 1, \ldots, n$ where $Y_i = \min(T_i, C_i)$, C_i being the censoring time for observation i, and $I_i = \mathcal{I}(T_i < C_i)$ being an indicator of whether failure occurred in subject i before the censoring time.

For the simulation study, the true covariates $\boldsymbol{X}_i$ and the measurement error $\boldsymbol{U}_i$ were simulated as in the linear model simulation (see Section 4.4.1). The survival times T_i were simulated using the Weibull hazard as baseline, $h_0(t) = \lambda_T \rho t^{\rho-1}$ with shape parameter $\rho = 1$ and scale parameter $\lambda_T = 0.01$. The censoring times C_i were randomly drawn from an exponential distribution with rate $\lambda_C = 0.001$. Two choice of the true $\boldsymbol{\theta}$ were considered, $\boldsymbol{\theta}_1 = (1, 1, 1, 1, 1, 0, \ldots, 0)^\top$ and $\boldsymbol{\theta}_2 = (2, 1.75, 1.50, 1.25, 1, 0, \ldots, 0)^\top$. For $\boldsymbol{\theta}_1$, the model configuration resulted in samples with between 20% and 25% of the observations being censored, while for $\boldsymbol{\theta}_2$, between 25% and 30% of the observations were censored. The sample size was fixed at $n = 300$, and simulations were done for number of covariates $p \in \{500, 1000, 2000\}$.

σ_u^2	p	ℓ_2			FP			FN		
		True	Naive	SIM-SELEX	True	Naive	SIM-SELEX	True	Naive	SIM-SELEX
0.15	500	1.36	2.25	1.8	8.59	3.66	0.00	0.00	0.00	0.03
		(0.15)	(0.11)	(0.23)	(6.34)	(3.99)	(0.00)	(0.00)	(0.00)	(0.18)
	1000	1.41	2.27	1.82	10.8	4.77	0.00	0.00	0.00	0.05
		(0.15)	(0.11)	(0.23)	(7.63)	(5.18)	(0.00)	(0.00)	(0.00)	(0.22)
	2000	1.47	2.31	1.89	12.92	5.68	0.00	0.00	0.00	0.07
		(0.15)	(0.12)	(0.24)	(8.97)	(6.32)	(0.00)	(0.00)	(0.00)	(0.26)
0.30	500	1.37	2.58	2.19	8.03	2.33	0.00	0.00	0.00	0.52
		(0.16)	(0.1)	(0.21)	(6.02)	(3.09)	(0.00)	(0.00)	(0.06)	(0.51)
	1000	1.43	2.6	2.22	10.31	3.24	0.00	0.00	0.00	0.55
		(0.15)	(0.09)	(0.2)	(7.6)	(4.1)	(0.00)	(0.00)	(0.00)	(0.53)
	2000	1.46	2.63	2.26	13.71	3.84	0.00	0.00	0.00	0.7
		(0.16)	(0.09)	(0.19)	(9.5)	(4.97)	(0.00)	(0.00)	(0.04)	(0.5)

Table 4.3: Comparison of estimators for Cox survival models for the case $\boldsymbol{\theta}_1$ based on ℓ_2 estimation error, average number of false positives (FP) and false negatives (FN). Standard errors in parentheses.

For the Cox model, implementation of SIMSELEX is much more computationally intensive than the linear and logistic models. This can be attributed to computation of the generalized lasso for the Cox model, see (Hastie et al., 2015, Section 3.5). As such, only $B = 40$ replicates were used for each λ value in the extrapolation step of the SIMSELEX algorithm. It should further be noted that, to the best of our knowledge, the Cox model with high-dimensional data subject to measurement error has not been considered by any other authors. As such, there is no competitor method for use in the simulation study. However, the model using the true covariates not subject to measurement error can be viewed as a gold standard measure of performance. The naive model was also implemented. The simulation results for the case of $\boldsymbol{\theta}_1$ are reported in Table 4.3, while the results for the case of $\boldsymbol{\theta}_2$ are presented in Table 4.9.

Table 4.3 shows that the SIMSELEX has a significantly lower ℓ_2 error than the naive estimator. With regards to recovery of the sparsity pattern, SIMSELEX has negligible average number of false positives in all the considered settings, while the naive estimator and the true estimator respectively result in the selection of more than 10 and 2 false

positives on average. Neither of the true and the naive estimators result in false negatives, while the SIMSELEX estimator has average number of false negatives around 0.05 for the case of $\sigma_u^2 = 0.15$ and around 0.6 for the case of $\sigma_u^2 = 0.3$.

4.4.4. Computational Time

The nature of SIMSELEX may lead to the suspicion that it is a computationally inefficient method that does not scale well with increasing sample size. We have investigated this possibility and have compared the proposed SIMSELEX method with existing methods for linear and logistic regression models.

For the SIMSELEX procedure, the bulk of computational time is taken up by the simulation step, i.e. the simulation of pseudo-data and the fitting of the proposed sparse model to each such set of data. In general, if algorithms exist for fast computation of the naive estimator, then implementation for the pseudo-data is equally fast. Furthermore, the generation of the psuedo-data only requires the simulation of normal random vectors, which can also be done fast. Consider therefore the linear model as an example. Here, in the simulation study the median time to implement the simulation step with $p = 500, 1000, 2000$ with 5 values of the λ and $B = 100$ replicates per λ was approximately 350, 480, and 760 seconds respectively. When considering the logistic regression model, these numbers were 510, 680, and 1010 seconds. The simulation step for the Cox survival model takes much longer time: Even with only $B = 40$ replicates in the case of $p = 500$, the median time is approximately 5380 seconds.

The computation time for selection step in the SIMSELEX procedure only depends on the number of λ-values and is generally fast to implement. With 5 values for λ and with $p = 2000$, selection takes about 250 seconds. The extrapolation step take least amount of time, with the median time less than 20 seconds in all the settings.

When compared to the other methods considered, SIMSELEX scales well with the dimension of the problem. In the linear model setting, the conic estimator takes very long when the number of covariates is large; for $p = 2000$, the median time to compute the conic estimator was around 6600 seconds. This is six times larger than the median

computation time for SIMSELEX for the same dimension size. The corrected lasso tends to be faster than SIMSELEX for $p = 500$ and $p = 1000$ but takes roughly the same amount of time in the case of $p = 2000$.

In the logistic model setting, the conditional scores lasso takes less time to compute than the SIMSELEX procedure; however, its tuning parameter is selected through a subjective rule and not in a data-driven way. The GMUS estimator is generally not scalable with the current implementation in the `hdme` package. Computation times for all methods was tabulated in Table 4.12.

4.5. SIMSELEX for Spline-Based Regression

This section provides implementation of SIMSELEX in the high-dimensional nonparametric regression setting and further demonstrates the flexibility of the procedure.

4.5.1. Spline Model Estimation

The proposed SIMSELEX algorithm can also be adapted for used for more flexible models such as regression using splines. Assume that the data $(\boldsymbol{W}_i, Y_i)$ are generated by an additive model $Y_i = \sum_{j=1}^{p} f_j(X_{ij}) + \epsilon_i$ with $\boldsymbol{W}_i = \boldsymbol{X}_i + \boldsymbol{U}_i$ and $\boldsymbol{U}_i$ having known covariance matrix $\boldsymbol{\Sigma}_U$. Also assume that $E[Y_i] = 0$, $i = 1, \ldots, n$. In practice, this can be achieved by centering the observed outcome variable. Furthermore, each of the functions $f_j(x)$ is assumed sufficiently smooth so that it can be well-approximated by an appropriately chosen set of basis functions. In this paper, the focus will be on an approximation using cubic B-splines with K knots. This model will have $p(K + 3)$ regression coefficients that need to be estimated.

Now, assume that the true covariates $\boldsymbol{X}_i$ have been observed without measurement error. Let $\phi_{jk}(x)$, $j = 1, \ldots, p$, $k = 1, \ldots, K + 3$ denote the resulting set of cubic B-spline basis functions where the knots for the jth covariate have been chosen as the $(100k)/(K+1)$th percentiles, $k = 1, \ldots, K$, of said covariate. The model to be estimated is then of the form $Y_i = \sum_{j=1}^{p} \sum_{k=1}^{K+3} \beta_{jk}\phi_{jk}(X_{ij}) + \epsilon_i$. In this setting, the jth covariate is selected if at least one of the coefficients $\beta_{jk}, k = 1, \ldots, K$ is nonzero. Therefore, it is

natural to delineate all the coefficients β_{jk} into p groups, each corresponding to a covariate and containing $K+3$ parameters. The model parameters are estimated by minimizing the penalized loss function

$$R(\boldsymbol{\beta}) = \sum_{i=1}^{n}\left[Y_i - \sum_{j=1}^{p}\sum_{k=1}^{K+3}\beta_{jk}\phi_{jk}(X_{ij})\right]^2 + (1-\alpha)\kappa\sum_{j=1}^{p}\sqrt{\sum_{k=1}^{K+3}\beta_{jk}^2} + \alpha\kappa\sum_{j=1}^{p}\sum_{k=1}^{K+3}\|\beta_{jk}\|\,. \tag{4.6}$$

This loss function has been considered in Simon et al. (2013) for the sparse group lasso estimator. Let $\widehat{\boldsymbol{\beta}}^{\text{true}}$ denote the estimated coefficients from this model. The loss function (4.6) combines the lasso and group lasso penalties. The tuning parameter $\alpha \in [0,1]$ balances overall parameter sparsity and within-group sparsity. While it is expected that only a few covariates will be selected, the nonlinear effect of each selected covariate may require a large number of basis functions to be accurately modeled. Therefore, strong overall sparsity but only mild within-group sparsity is expected. As per Simon et al. (2013), $\alpha = 0.05$ is used. The estimator of each function f_j is $\widehat{f}_j^{\text{true}}(x) = \sum_{k=1}^{K+3}\widehat{\beta}_{jk}^{\text{true}}\phi_{jk}(x)$ for all $j = 1,\ldots,p$.

Now, using the contaminated data $\boldsymbol{W}_i$, a similar procedure can be followed to obtain the naive estimator. Again, evaluate the knots of the model as equally spaced percentiles, this time of the covariates contaminated by measurement error. The corresponding cubic B-spline basis functions are denoted $\phi_{jk}^{W}(x)$. The naive estimator $\widehat{\boldsymbol{\beta}}^{\text{naive}}$ can be obtained by minimizing a function analogous to (4.6), but with true data X_{ij} replaced by contaminated data W_{ij} in the loss function. The naive estimator for function f_j is $\widehat{f}_j^{\text{naive}}(x) = \sum_{k=1}^{K+3}\widehat{\beta}_{jk}^{\text{naive}}\phi_{jk}^{W}(x)$ for all $j = 1,\ldots,p$.

To compute the SIMSELEX estimator, for each of the added noise level λ_m, generate B pseudodata $\tilde{\boldsymbol{W}}^{(b)}(\lambda_m)$, $b = 1,\ldots,B$ as before. Note that the same set of basis functions obtained for the naive estimate is used. Then, the estimate $\widehat{\beta}_{jk}^{(b)}(\lambda_m)$ for each set of pseudodata is obtained by minimizing a function analogous to (4.6), but with true data X_{ij} replaced by pseudodata $\tilde{W}_{ij}^{(b)}(\lambda_m)$ in the loss function. The estimates $\widehat{\beta}_{jk}^{(b)}(\lambda_m)$ are averaged across B samples to obtain $\widehat{\beta}_{jk}(\lambda_m)$ for each λ_m in the grid.

After the simulation step of SIMSELEX, the jth covariate is associated with $K+3$

"paths" $\Big\{(\lambda_i, \hat{\beta}_{j1}(\lambda_i)), \ldots, ...(\lambda_i, \hat{\beta}_{j,K+3}(\lambda_i))\Big\}$, each of which needs to be extrapolated to $\lambda = -1$. This is different from the parametric model setting considered in Section 4.4, where each covariate j is associated with only one parameter path $\theta_j(\lambda_i)$ that needs to be extrapolated to $\lambda = -1$. Therefore, the selection step for spline-based regression needs to be approached with some care. Here, two different approaches for selection step are considered.

The first approach for selection considered applies a variation of the group lasso to all $p(K+3)$ coefficients β_{jk}. This is done using a quadratic extrapolation function. Specifically, it is assumed that

$$\hat{\beta}_{jk}(\lambda_i) = \Gamma_{0jk} + \Gamma_{1jk}\lambda_i + \Gamma_{2jk}\lambda_i^2 + \varepsilon_{ijk}, \quad i = 1, \ldots, M, \quad j = 1, \ldots, p, \quad k = 1, \ldots, K+3$$

with ε_{ijk} zero-mean error terms. With this approach, the jth covariate is zeroed out if all the parameter estimates $\{\hat{\Gamma}_{ijk}\}_{i=0,1,2,\ k=1,\ldots,K}$ equal zero. Applying the group lasso, the loss function to be minimized is

$$R = \sum_{j=1}^{p} \left(\|\boldsymbol{\Theta}_j - \boldsymbol{\Lambda}\boldsymbol{\Gamma}_j\|_2^2 + \xi_3 \|\boldsymbol{\Gamma}_j\|_2 \right) \tag{4.7}$$

where

$$\boldsymbol{\Gamma}_j = \begin{bmatrix} \Gamma_{0j1} & \ldots & \Gamma_{0jK} \\ \Gamma_{1j1} & \ldots & \Gamma_{1jK} \\ \Gamma_{2j1} & \ldots & \Gamma_{2jK} \end{bmatrix}, \ \boldsymbol{\Theta}_j = \begin{bmatrix} \hat{\beta}_{j1}(\lambda_1) & \ldots & \hat{\beta}_{jK}(\lambda_1) \\ \vdots & & \vdots \\ \hat{\beta}_{j1}(\lambda_M) & \ldots & \hat{\beta}_{jK}(\lambda_M) \end{bmatrix}, \ \Lambda = \begin{bmatrix} 1 & \lambda_1 & \lambda_1^2 \\ \vdots & \vdots & \vdots \\ 1 & \lambda_M & \lambda_M^2 \end{bmatrix},$$

and $\|.\|_2$ denotes the Frobenius norm (matrix version of the ℓ_2 norm). This is a very natural extension of the approach considered in Section 4.4. The tuning parameter ξ_3 can be chosen through cross-validation. Even though (4.7) is convex and block-separable, the minimization is computationally very expensive due to the number of model parameters. As such, an alternative approach intended to speed up computation was also considered.

The alternative approach considered for selection applies the group lasso not to each individual coefficient, but to the *norm* of each group of coefficients $\beta_{jk},\ k = 1, \ldots, K+3$

corresponding to the jth covariate. This is motivated by noting that the norm of a group of coefficients will only equal 0 if all the coefficients in said group are equal to 0. More specifically, let $\hat{\beta}_j(\lambda_i) = [\hat{\beta}_{j1}(\lambda_i), \ldots, \hat{\beta}_{jK}(\lambda_i)]^\top$, $i = 1, \ldots, M$, $j = 1, \ldots, p$, and let $\hat{\eta}_{ij} = \left\| \hat{\beta}_j(\lambda_i) \right\|_q$ denote the corresponding ℓ_q norm. The two scenarios considered are $q = 1$ and 2. The norm is modeled quadratically as

$$\hat{\eta}_{ij} = \Gamma_{0j} + \Gamma_{1j}\lambda_i + \Gamma_{2j}\lambda_i^2 + \varepsilon_{ij}, \quad i = 1, \ldots, M,$$

with ε_{ij} zero-mean error terms. The jth covariate is not selected if all the elements of the estimated vector $(\hat{\Gamma}_{0j}, \hat{\Gamma}_{1j}, \hat{\Gamma}_{2j})$ are equal to zero. The group lasso loss function to be minimized is

$$\tilde{R} = \frac{1}{2} \sum_{i=1}^{M} \sum_{j=1}^{p} \left(\hat{\eta}_{ij} - \Gamma_{0j} - \Gamma_{1j}\lambda_i - \Gamma_{2j}\lambda_i^2 \right)^2 + \xi_4 \sum_{j=1}^{p} \sqrt{\Gamma_{0j}^2 + \Gamma_{1j}^2 + \Gamma_{2j}^2}. \tag{4.8}$$

Equation (4.8) is convex and block-separable, and can be minimized efficiently through proximal gradient descent methods. The tuning parameter ξ_4 can be chosen through cross-validation.

Finally, if the jth covariate is chosen in the selection step, extrapolation is performed separately on each β_{jk} to get the SIMSELEX estimate for each coefficient, denoted by $\widehat{\beta}_{jk}^{\text{ssx}}$. Then, the SIMSELEX estimate for each function f_j is computed as $\widehat{f}_j^s(x) = \sum_{k=1}^{K+3} \widehat{\beta}_{jk}^{\text{ssx}} \phi_{jk}^W(x)$.

4.5.2. Simulation

Data pairs $(\boldsymbol{W}_i, Y_i)$ were generated according to the additive model $Y_i = \sum_{j=1}^{p} f_j(X_{ij}) + \epsilon_i$, and $\boldsymbol{W}_i = \boldsymbol{X}_i + \boldsymbol{U}_i$ with $f_1(t) = 3\sin(2t) + \sin(t)$, $f_2(t) = 3\cos(2\pi/3t) + t$, $f_3(t) = (1-t)^2 - 4$, $f_4(t) = 3t$, and $f_j(t) = 0$, $j = 5, \ldots, p$. The $s = 4$ non-zero functions have all been centered at 0. The true covariates X_{ij} were generated from a Gaussian copula model with correlation structure $\Sigma_{ij} = 0.25^{|i-j|}$, see Xue-Kun Song (2000) for more details. The covariates marginal were then rescaled to have a uniform distribution on $[-3, 3]$. The measurement errors $\boldsymbol{U}_i$ were generated to be *i.i.d.* p-variate normal,

$\boldsymbol{U}_i \sim N_p(\boldsymbol{0}, \sigma_u^2 \boldsymbol{I}_p)$, with $\boldsymbol{I}_p$ the $p \times p$ identity matrix. Two values of σ_u^2 were considered, $\sigma_u^2 = 0.15$ and $\sigma_u^2 = 0.3$, corresponding to 5% and 10% noise-to-signal ratios for each individual covariate. Simulations were also done for number of covariates $p \in \{100, 500\}$. Although the NSR look small in each covariate, recall from Section 4.4.1 that the change in total proportion of variability ΔV increases rapidly in multivariate space. For each configuration, $N = 500$ samples were generated.

For each simulated dataset, the true, naive, and SIMSELEX estimators were computed. We are unaware of any other method in the literature dealing with spline-based regression in the high-dimensional setting when covariates are subject to measurement error. For each covariate, the number of knots was chosen to be $K = 6$. As such, each function f_j is modeled by $K + 3 = 9$ basis functions. In the simulation step of SIMSELEX, $B = 40$ sets of pseudodata are generated for each level of added measurement error. The function estimators are evaluated using integrated squared error, ISE $= \sum_{j=1}^{p} \int \left(\widehat{f}_{ij}(x) - f_{ij}(x) \right)^2 dx$, as well as the number of false positive (FP) and false negative (FN) covariates selected.

Table 4.4 compares the performance of the SIMSELEX estimator with alternative methods of doing variable selection in the case of $p = 100$ and with $\sigma_u^2 = 0.15$. Firstly, selection approach (4.7) using individual models for all the coefficients β_{jk} was implemented. Secondly, approach (4.8) was applied both for the ℓ_1 norm and for the ℓ_2 norm, calculated based on the groups of parameters corresponding to specific variables. The table reports the MISE, the number of false positives (FP) and false negatives, and also the average time (in seconds), all calculated for 500 simulated samples. The average time was recorded based on running the simulations on one node (memory 7GB) of ManeFrame II (M2), the high-performance computing cluster of Southern Methodist University in Dallas, TX.

Considering the results in Table 4.4, selection based on the ℓ_2 norm gives the best result, while selection based on individually considering all the coefficients gives the worst results. The latter also takes more than 14 times longer to compute (on average) than the ℓ_2 approach. The ℓ_1 approach is comparable to ℓ_2 in terms of MISE and average

Selection	MISE	FP	FN	Time (second)
All coefficients	17.32	21.50	0.00	819.00
ℓ_1 norm	17.17	10.06	0.00	59.70
ℓ_2 norm	16.76	4.62	0.00	56.68

Table 4.4: Comparison of SIMSELEX variable selection methods for spline regression with $p = 100$.

σ_u^2	Estimator	$p = 100$			$p = 500$		
		MISE	FP	FN	MISE	FP	FN
0.15	True	15.96	3.68	0.00	18.05	12.11	0.00
		(2.99)	(2.75)	(0.00)	(3.28)	(6.47)	(0.00)
	Naive	37.19	9.67	0.00	47.62	16	0.00
		(7.17)	(5.51)	(0.00)	(8.41)	(10.16)	(0.00)
	SIMSELEX	16.95	5.48	0.00	21.94	6.5	0.00
		(4.63)	(3.14)	(0.00)	(6.3)	(3.84)	(0.00)
0.30	True	15.96	3.68	0.00	18.05	12.11	0.00
		(2.99)	(2.75)	(0.00)	(3.28)	(6.47)	(0.00)
	Naive	69.89	9.28	0.01	87.73	13.26	0.08
		(12.31)	(6.42)	(0.12)	(13.2)	(10.84)	(0.28)
	SIMSELEX	38.51	3.74	0.03	54.41	4.06	0.17
		(11.37)	(2.77)	(0.18)	(14.15)	(3.27)	(0.39)

Table 4.5: Comparison of estimators for high-dimensional spline regression model based on estimation error (MISE), average number of false positives (FP) and false negatives (FN). Standard errors in parentheses.

computation time, but has a much higher average number of false positive selections. Therefore, the SIMSELEX estimator with selection using ℓ_2 norm for parameter groups is compared with the naive estimator. The results are summarized in Table 4.5.

Table 4.5 demonstrates that SIMSELEX has a significantly lower estimation error (MISE) than the naive estimator in all the configurations considered. Particularly, in the case of $\sigma_u^2 = 0.15$, the SIMSELEX estimator has MISE close to the true estimator. In the case of $\sigma_u^2 = 0.3$, compared to the naive estimator, the SIMSELEX estimator reduces MISE significantly. For example, in the case of $p = 500$, the reduction in MISE resulting from using the SIMSELEX over the naive estimator is more than 38%. Even so, it is clear that measurement error has a significant effect on the recovery of the functions f_j for the case $\sigma_u^2 = 0.3$.

Regarding variable selection, the SIMSELEX estimator performs very well in the case of $\sigma_u^2 = 0.15$. In this case, SIMSELEX is always able to select the true non-zero functions

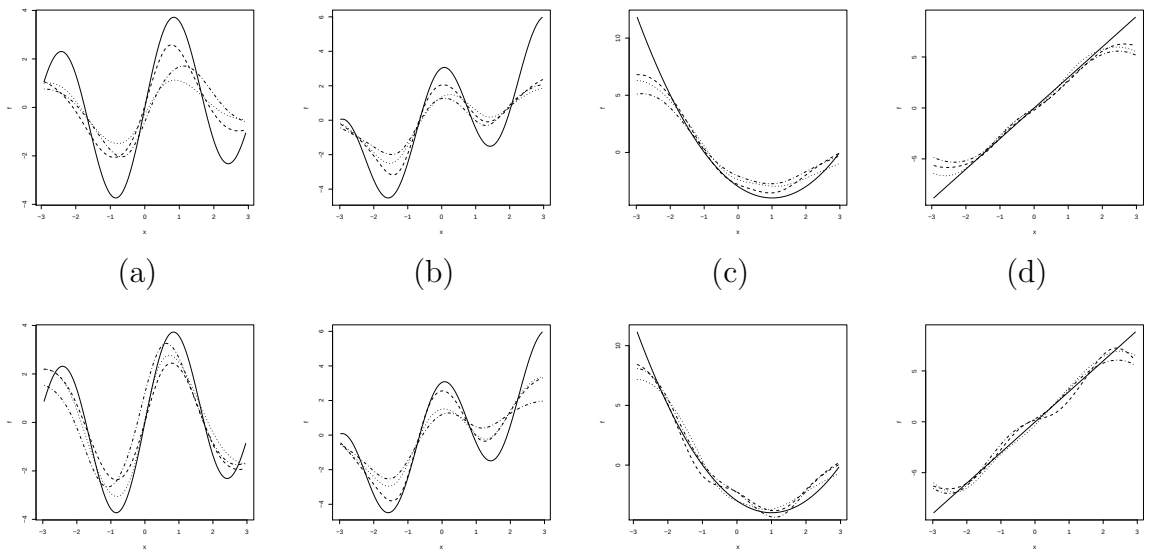


Figure 4.2: Curves Q_1 (- - - - -), Q_2 (.....), Q_3 (-.-.-.-), and true function (———) for the esimated functions from the naive estimators (top) and the SIMSELEX estimators (bottom) corresponding to $p = 600$ and $\sigma_u^2 = 0.15$. For (a),(e): $f_1(x) = 3\sin(2x) + \sin(x)$; for (b),(f): $f_2(x) = 3\cos(2\pi x/3) + x$; for (c), (g): $f_3(x) = (1-x)^2 - 4$; for (d), (h): $f_4(x) = 3x$.

by having false negatives equal 0 in all samples, while having only a slightly higher average number of false positives than the true estimator with $p = 100$ and lowest average number of of false positives with $p = 500$. In the case of $\sigma_u^2 = 0.3$, SIMSELEX gives considerably fewer false positives on averages than both the true and naive estimators. SIMSELEX does have the highest average number of false negatives for this setting, but this is still below 0.5 in all the cases considered.

Figure 4.2 shows plots of the estimators corresponding to the first, second, and third quantiles (Q_1, Q_2, and Q_3) of ISE for the naive estimator and the SIMSELEX estimator in the case of $\sigma_u^2 = 0.15$ and $p = 500$. The SIMSELEX estimator captures the shape of the functions considerably better, especially around the peaks of f_1 and f_2. Particularly, in the case of $\sigma_u^2 = 0.15$, the SIMSELEX estimator is able to capture the shape of all the nonzero functions very well. Comparable figures for the case $\sigma_u^2 = 0.3$ and $p = 500$ are given in Figure 4.3. As one would anticipate there, the increase in measurement error variance results in poorer recovery of the underlying functions. Even so, SIMSELEX has notably better performance than the naive approach.

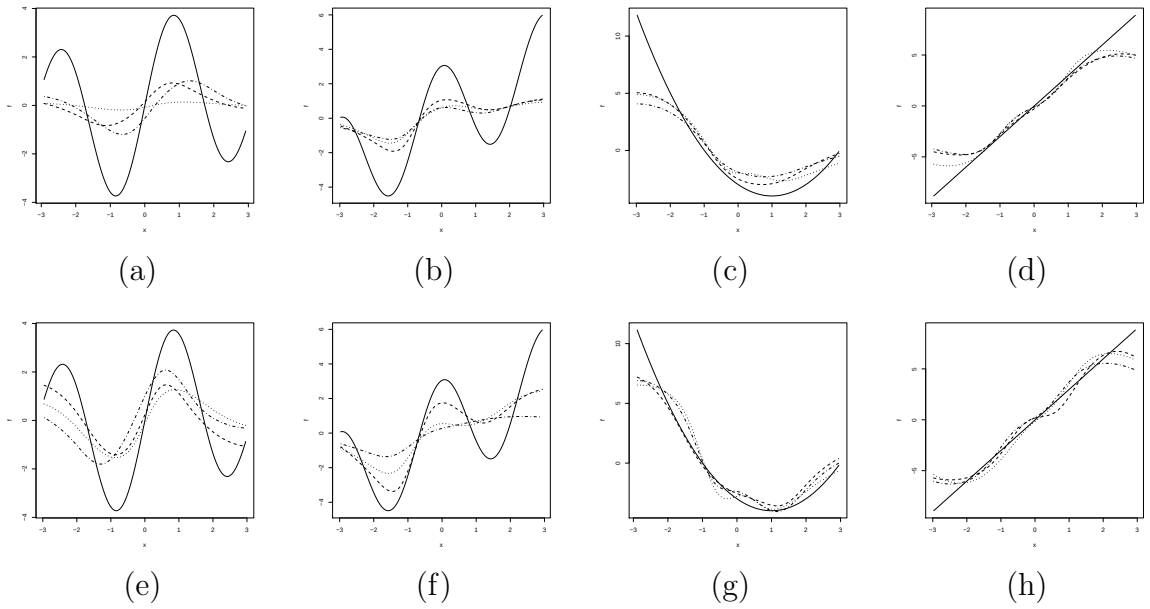


Figure 4.3: Curves Q_1 (- - - - -), Q_2 (.....), Q_3 (-·-·-·-), and true function (———) for the esimated functions from the naive estimators (top) and the SIMSELEX estimators (bottom) corresponding to $p = 600$ and $\sigma_u^2 = 0.30$. For (a),(e): $f_1(x) = 3\sin(2x) + \sin(x)$; for (b),(f): $f_2(x) = 3\cos(2\pi x/3) + x$; for (c), (g): $f_3(x) = (1-x)^2 - 4$; for (d), (h): $f_4(x) = 3x$.

4.6. Microarray Analysis

In this data application, we analyze an Affymetrix microarray dataset containing gene expression measurements of 144 favorable histology Wilms tumors. The data is publicly available on the ArrayExpress website under access number E-GEOD-10320. In these Wilms tumors, the cancer cell's nuclei is not very large or distorted, so a high proportion of patients are successfully treated. However, relapse is a possibility after treatment. It is of interest to identify any genes associated with relapse. A total of 53 patients experienced a relapse, while 91 patients had no relapse over a three year follow-up. Replicate data is available for each patient as multiple probes were collected per patient. This allows for the estimation of gene-specific measurement error variances. The analysis is performing after applying a logarithmic transformation.

To make our analysis comparable with that previously done by Sørensen et al. (2015), data preprocessing was done as described by them. The raw data were processed using the Bayesian Gene Expression (BGX) Bioconductor of Hein et al. (2005) creating a posterior

distribution for the log-scale expression level of each gene in each sample. For gene j in patient i, the posterior mean $\widehat{\mu}_{ij}$ was then taken as an estimates of the true gene expression level.

Now, let $\widehat{\boldsymbol{\mu}}_j = (\widehat{\mu}_{1j}, \ldots, \widehat{\mu}_{nj})^\top$ denote the estimated vector of gene expression levels for gene $j = 1, \ldots, p$ for the n patients. Furthermore, let $\bar{\mu}_j = (1/n)\sum_{j=1}^{n} \widehat{\mu}_{ij}$ and $\widehat{\sigma}_j^2 = (1/n)\sum_{j=1}^{n}(\widehat{\mu}_{ij} - \bar{\mu}_j)^2$ denote the estimated mean and variance of gene j. Standardized measurements $\boldsymbol{W_i} = (W_{i1}, \ldots, W_{ip})$, $i = 1, \ldots, n$ were then calculated as $W_{ij} = (\widehat{\mu}_{ij} - \bar{\mu}_j)/\widehat{\sigma}_j,\ i = 1, \ldots, n,\ j = 1, \ldots, p$. To estimate $\boldsymbol{\Sigma}_u$, it was assumed that the measurement error is independent of the patient and that the associated variance is constant across patients for a given gene. Let $\text{var}(\widehat{\mu}_{ij})$ denote the posterior variance of the estimated distribution of gene j in patient i. These estimates were then combined, $\widehat{\sigma}_{u,j}^2 = (1/n)\sum_{i=1}^{n} \text{var}(\widehat{\mu}_{ij})$, and the measurement error covariance matrix associated with $\boldsymbol{W}$ was estimated by the diagonal matrix with elements $(\widehat{\boldsymbol{\Sigma}}_u)_{j,j} = \widehat{\sigma}_{uj}^2/\widehat{\sigma}_j^2, j = 1, \ldots, p$. Only the $p = 2074$ genes with $\widehat{\sigma}_{u,j}^2 < (1/2)\widehat{\sigma}_j^2$, i.e. estimated noise-to-signal ratio less than 1, were retained for analysis.

Using the data $(\boldsymbol{W}_i, Y_i)$, $i = 1, \ldots, n$, with Y_i an indicator of relapse, four different procedures were used to fit a logistic regression model to the data. These procedures are a naive model with lasso penalty, the conditional scores lasso of Sørensen et al. (2015), the SIMSELEX model, and a SIMEX model without variable selection. For the naive, SIMSELEX and SIMEX models, 10-fold cross-validations using the one-standard-error rule was used to select the tuning parameter. For SIMEX and SIMSELEX, a grid of 16 equally spaced λ-values from 0.01 to 2 and $B = 100$ replicates were used in the simulation step. The elbow method was used for tuning parameters selection in the conditional scores lasso. SIMEX without selection identified 1699 out of 2074 genes. Though many of the estimated coefficients are close to zero, 17 estimated coefficients exceed 0.1, and a further 41 exceed 0.01. This analysis is a far departure from the required sparse model. Results of the other three analyses are in Table 4.6.

The naive approach identified 26 non-zero genes, while conditional scores identified 13 non-zero genes. SIMSELEX identified only 4 non-zero genes. Note that one of the

Table 4.6: Gene symbols and estimated coefficients from the naive lasso, the conditional scores lasso, and the SIMSELEX estimator applied to the Wilms tumors data. Genes selected by SIMSELEX are printed in bold.

Gene	Naive	Conditional scores	SIMSELEX
202016_at	**-0.2216**	**-0.0348**	**-0.7038**
205132_at	**-0.1997**	**-0.2127**	**-0.6498**
213089_at	**0.2096**	**0.0575**	**0.6775**
207761_at	**0.0691**	-	**0.7399**
209466_x_at	-0.0310	-0.2425	
218678_at	-0.1256	-0.1600	
209259_s_at	-0.1038	-0.1599	
209281_s_at	-0.0511	-0.1054	
204710_s_at	-0.2004	-0.0958	
202766_s_at	-	-0.0740	
208905_at	-	-0.0463	
201194_at	-	-0.0448	
211737_x_at	-	-0.0279	
203156_at	-0.1090	-0.0128	
213779_at	0.1142		
201859_at	-0.1087		
208965_s_at	0.1388		
205933_at	0.0913		
(11 more non-zero genes)	$\lvert\cdot\rvert < 0.06$		

genes chosen by SIMSELEX was not chosen by the conditional scores method (although it was chosen by the naive estimator). However, the magnitude of the estimated coefficients were much larger for SIMSELEX compared to the naive and conditional scores estimators. The large number of genes selected by the naive and conditional scores approaches are potentially a consequence of the false positive rates seen in the simulation studies. While SIMSELEX does suffer from the occasional false negative, this rate was lower in our simulation studies than the equivalent rate for the conditional scores lasso.

4.7. Conclusion

The chapter presents a modified SIMEX algorithm with a selection step for sparse models estimation in high-dimensional models with covariate measurement error . This SIMSELEX algorithm is explored in linear and logistic regression models, the Cox propor-

tional hazards model, as well as spline-based regression. In the linear model, SIMSELEX has performance comparable to the corrected lasso. In the logistic model, it has much better performance than the corrected scores lasso. In the Cox model and spline-model settings, no other estimators have been proposed in the literature. For these, it is shown that the method leads to much better performance than a naive approach that ignores measurement error, and compares favorably to estimators obtained using uncontaminated data.

It was noted that SIMSELEX requires the measurement error covariance matrix be known or estimable. In our data application, an estimation method based on the BGX Bioconductor of Hein et al. (2005) was used. The development and comparison of other methods for estimating measurement error covariance matrices will be explored in future work. Further work around reducing the number of false negatives in SIMSELEX will also be conducted. For example, the group lasso used for variable selection provides an ordering for the inclusion/exclusion of variables in the model (see, for example, Figure 4.1). As such, a decision can be made beforehand to include an additional number of variables, say q, after selection. Thus, if selection recommends keeping $\hat{p}$ variables, then the practitioner keeps $\hat{p}+q$ variables for extrapolation. The performance of this idea was not explored here.

4.8. Appendix

4.8.1. Illustrating SIMEX performance for a high-dimensional setting

In both Sections 4.2 and 4.3, it was mentioned that SIMEX did not perform well when applied to high-dimensional errors-in-variables models without suitably modifying the procedure. Specifically, standard SIMEX inflates the number of estimated nonzero components considerably, even when combined with a procedure such as the lasso. Here, a simulated example is illustrated.

For the example, data pairs $(\boldsymbol{W}_i, Y_i)$ were generated according to the linear model $Y_i = \boldsymbol{X}_i^\top \boldsymbol{\theta} + \varepsilon_i$ with additive measurement error $\boldsymbol{W}_i = \boldsymbol{X}_i + \boldsymbol{U}_i$. Both the true covariates $\boldsymbol{X}_i$ and the measurement error components $\boldsymbol{U}_i$ were generated to be *i.i.d.* p-variate

normal. Specifically, $\boldsymbol{X}_i \sim \mathrm{N}_p(\mathbf{0}, \boldsymbol{\Sigma})$, with $\boldsymbol{\Sigma}$ having entries $\Sigma_{ij} = \rho^{|i-j|}$ with $\rho = 0.25$, and $\boldsymbol{U}_i \sim \mathrm{N}_p(\mathbf{0}, \boldsymbol{\Sigma}_u)$ with $\boldsymbol{\Sigma}_u = \sigma_u^2 I_{p\times p}$ with $\sigma_u^2 = 0.45$. The error components ε_i were simulated to be *i.i.d.* univariate normal, $\varepsilon \sim N(\mathbf{0}, \sigma_\varepsilon^2)$ with $\sigma_\varepsilon = 0.128$. The sample sizes was fixed at $n = 300$, and the number of covariates was $p = 500$. The parameter vector was taken to be $\boldsymbol{\theta} = \{1, 1, 1, 1, 1, 0, \ldots, 0\}$ with $s = 5$ nonzero coefficients and $p - s = 495$ zero coefficients.

For the simulation step of SIMEX, a grid of $M = 13$ equally spaced λ-values ranging from 0.2 to 2 were used. For each value of λ, a total of $B = 100$ sets of pseudo-data were generated. In applying the lasso, the tuning parameter was chosen based on the one-standard-error rule based on 10-fold cross-validation. The lasso was implemented using the `glmnet` package in R. For the extrapolation step, a quadratic function was used.

The analysis of the simulated data shows that SIMEX applied to the lasso results in 174 nonzero parameter estimates. Of the 169 false positives, 156 are fairly small (less than 0.001 in absolute value), with 13 false positives being larger (greater than 0.001 in absolute value). Comparatively, a naive application of the lasso (not correcting for measurement error) gives only 5 non-zero parameter estimates. Implementing SIMEX, even when using a method such as the lasso that enforces sparsity, can result in an inflated number of variables in the model.

4.8.2. A brief review of existing methodology

In Section 4.4, the SIMSELEX estimator is compared to several existing methods for fitting errors-in-variables models in high-dimensional settings. For the linear model, SIMSELEX is compared with the corrected lasso estimator of Sørensen et al. (2015) and the conic estimator of Belloni et al. (2017). For the logistic model, the SIMSELEX estimator is compared with the conditional scores lasso of Sørensen et al. (2015). These approaches are briefly reviewed in this section.

Linear Model

The corrected lasso estimator of Sørensen et al. (2015) is the solution to the optimization problem

$$\begin{aligned} \min_{\boldsymbol{\theta}} \; & L(\boldsymbol{\theta}) = \|Y - \boldsymbol{W}\boldsymbol{\theta}\|_2^2 - \boldsymbol{\theta}^\top \boldsymbol{\Sigma}_u \boldsymbol{\theta} \\ \text{s.t. } & \|\boldsymbol{\theta}\|_1 \leq R \end{aligned}$$

where for p-dimensional vector $\boldsymbol{x}$, $\|\boldsymbol{x}\|_1 = \sum_{j=1}^p |x_j|$ and $\|\boldsymbol{x}\|_2^2 = \sum_{j=1}^p x_j^2$. Here, R is a tuning parameter that can be chosen based on cross-validation using an estimate of the unbiased loss function. Specifically, if the data are partitioned into random subset $\mathcal{P}_1, \ldots, \mathcal{P}_J$, each subset having size n/J, let $(\boldsymbol{W}_{(\mathcal{P}_j)}, Y_{(\mathcal{P}_j)})$ denote the data in the jth partition and let $(\boldsymbol{W}_{(-\mathcal{P}_j)}, Y_{(-\mathcal{P}_j)})$ denote the data excluding the jth partition. Also let $\hat{\boldsymbol{\theta}}_j$ denote the estimated parameter vector based on $(\boldsymbol{W}_{(-\mathcal{P}_j)}, Y_{(-\mathcal{P}_j)})$. Then the tuning parameter R can be chosen using cross-validation loss function

$$L_{CV}(R) = \sum_{j=1}^{J} \left\| Y_{\mathcal{P}_j} - \boldsymbol{W}_{\mathcal{P}_j} \hat{\boldsymbol{\theta}}_j \right\|_2^2 - \sum_{j=1}^{J} \hat{\boldsymbol{\theta}}_j^\top \boldsymbol{\Sigma}_u \hat{\boldsymbol{\theta}}_j.$$

The optimal tuning parameter R can be chosen either to minimize L_{CV}, or according to the one standard error rule (see Friedman et al. (2001)). Sørensen et al. (2015) prove that the corrected lasso performs sign-consistent covariate selection in large samples.

The conic estimator of Belloni et al. (2017) is also the solution to an optimization problem,

$$\begin{aligned} \min_{\boldsymbol{\theta}, t} \; & \|\boldsymbol{\theta}\|_1 + \lambda t \\ \text{s.t } & \left\| \frac{1}{n} \boldsymbol{W}^\top (Y - \boldsymbol{W}\boldsymbol{\theta} + \boldsymbol{\Sigma}_u \boldsymbol{\theta}) \right\|_\infty \leq \mu t + \tau, \quad t \geq 0, \quad \|\boldsymbol{\theta}\|_2 \leq t. \end{aligned}$$

where for p-dimensional vector $\boldsymbol{x}$, $\|\boldsymbol{x}\|_\infty = \max_{j=1,\ldots,p} |x_j|$. This method requires the selection of three tuning parameters, here denoted μ, τ and λ. The optimal choices of these tuning parameters depend on the underlying model structure, including the rate at which the number of nonzero model coefficients increases with sample size. Belloni

et al. (2017) do suggest tuning parameter values for application. Furthermore, these authors also proved that under suitable sparsity conditions, their conic estimator has smaller minimax efficiency bound than the Matrix Uncertainty Selection estimator of Rosenbaum et al. (2010). We are not aware of any comparison, numerical or otherwise, of the corrected lasso estimator and the conic estimator. This comparison is presented as part of our simulation study in Section 4.4.1 .

Logistic Regression

For the logistic regression model, the SIMSELEX estimator is compared with the conditional scores lasso estimator developed by Sørensen et al. (2015) and the Generalized Matrix Uncertainty Selector (GMUS) developed by Sørensen et al. (2018). The conditional scores lasso estimator is computed by solving the set of estimating equations

$$\sum_{i=1}^{n}\left(Y_i - F\left\{\eta_i - \frac{1}{2}\boldsymbol{\theta}^\top\boldsymbol{\Sigma}_u\boldsymbol{\theta}\right\}\right)\begin{pmatrix}1\\ \boldsymbol{W}_i + Y_i\boldsymbol{\Sigma}_u\boldsymbol{\theta}\end{pmatrix} = \mathbf{0} \text{ subject to } \|\boldsymbol{\theta}\|_1 \leq R$$

where $\eta_i = \mu + \boldsymbol{\theta}^\top(\boldsymbol{W}_i + Y_i\boldsymbol{\Sigma}_u\boldsymbol{\theta})$. Note that this is a system of $p+1$ estimating equations. Sørensen et al. (2015) also illustrate how the conditional scores lasso can be applied to other GLMs. For the simulation studies in section 4.4.2, the tuning parameter R is chosen to be $1.5\left\|\hat{\beta}_{\text{naive}}\right\|_1$, where $\hat{\beta}_{\text{naive}}$ denotes the naive lasso.

The GMUS estimator is defined as

$$\hat{\beta}_{MU} = \arg\min\{\|\beta\|_1 : \beta \in \Theta\}, \text{ where}$$

$$\Theta = \left[\beta \in \mathbb{R}^p : \left\|\frac{1}{n}\sum_{i=1}^{n} w_{ij}(y_i - \mu(\boldsymbol{w}_i^\top\beta))\right\|_\infty \leq \lambda + \frac{\delta}{\sqrt{n}}\|\beta\|_1 \|\mu'(\boldsymbol{W}\beta)\|_2\right]$$

where $\mu(.)$ denotes the Logistic function, $\mu'(\boldsymbol{W}\beta) = \{\mu'(\boldsymbol{w}_1^\top\beta), \ldots, \mu'(\boldsymbol{w}_n^\top\beta)\}^\top$, with $\mu'(.)$ denotes the first derivative of $\mu(.)$. The tuning parameter λ is chosen to be equal to the tuning parameter when computing the naive lasso, while the tuning parameter δ was chosen following the elbow rule. More specifically, a grid of δ-values is chosen. For each value of δ in the grid, the GMUS is computed. Finally, the number of non-zero coefficients

is plotted as a function of R, and the optimal R is chosen as the point at which the plot elbows i.e. starts to become flat. Note that finding this elbow for the GMUS is somewhat subjective and the authors do not provide an automated way of performing this selection.

For the simulation study in Section 4.4.2, the tuning parameter δ was chosen in a manner identical to the simulation study presented in Sørensen et al. (2015). First, $N_0 = 100$ samples were simulated using the data generation mechanism outlined. For the jth simulated dataset, let $R = \delta \left\| \hat{\boldsymbol{\theta}}_{\text{naive}} \right\|_1$, where $\left\| \hat{\boldsymbol{\theta}}_{\text{naive}} \right\|_1$ denotes the ℓ_1 norm of the naive lasso estimator. Let $(\delta, \text{NZ}_j(\delta))$ denote the curve of the number of non-zero coefficients as a function of λ. These curves were then averaged, resulting in curve $(\delta, \overline{\text{NZ}}(\delta))$ where $\overline{\text{NZ}}(\delta) = N_0^{-1} \sum_j \text{NZ}_j(\delta)$. The value of δ used subsequently to evaluate the conditional scores lasso estimators in the simulation study was the point at which the curve $\overline{\text{NZ}}(\delta)$ elbows. For each given simulation configuration, a different value of δ was calculated.

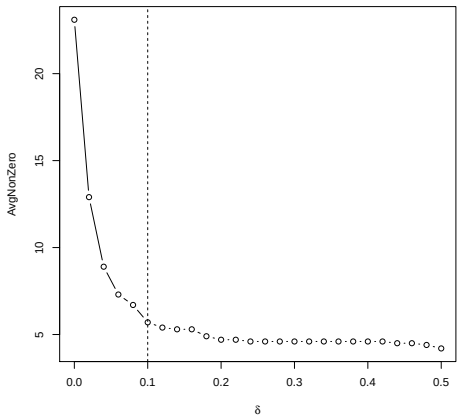
In the simulation study in Section 4.4.2, the GMUS estimator was computed only for the case of $p = 500$. The elbow plots for the settings associated with Normal measurement error were presented below. The tuning parameters in the simulation study with Laplace measurement error were chosen to be the same as the chosen value in the similar setting with Normal measurement error.

4.8.3. Additional Simulation Results for Linear Regression, Logistic Regression, and Cox Survival Model

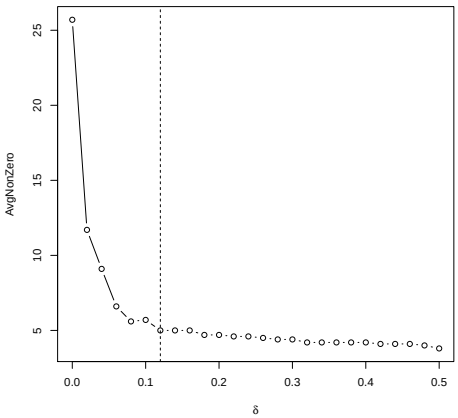
This section presents the simulation results corresponding to the case of $\boldsymbol{\theta}_2 = (1, 1, 1, 1, 1, 0, \ldots, 0)$. All the other simulation configurations are the same as outlined in the Section 4.4. The tabulated summaries included here for completeness.

4.8.4. Comparison of extrapolation functions for SIMSELEX

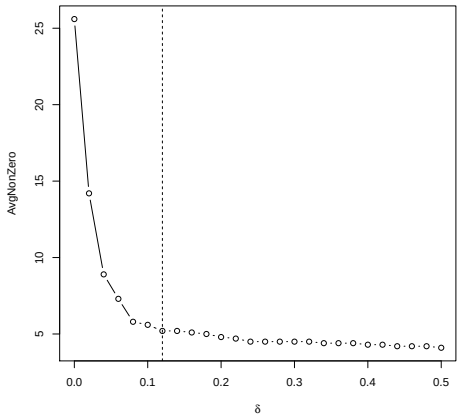
Several extrapolation functions for the SIMEX procedure have been proposed in the literature. The quadratic function and nonlinear means function are used most frequently. In this section, the performance of SIMSELEX when using either the quadratic or non-



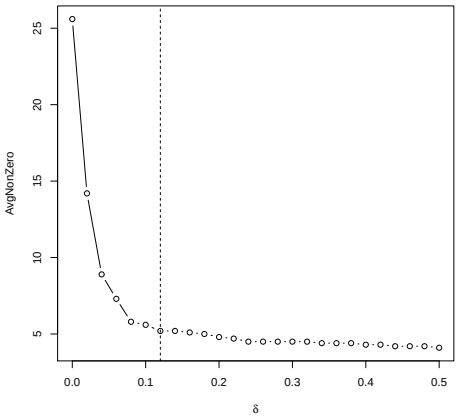
(a) Case $\boldsymbol{\theta}_1$ and $\sigma_u^2 = 0.15$



(b) Case $\boldsymbol{\theta}_2$ and $\sigma_u^2 = 0.15$



(c) Case $\boldsymbol{\theta}_1$ and $\sigma_u^2 = 0.30$



(d) Case $\boldsymbol{\theta}_2$ and $\sigma_u^2 = 0.30$

Figure 4.4: Elbow plots choosing tuning parameters in implementation of conditional scores lasso estimator in the logistic regression simulation.

p	Est	$\sigma_u^2 = 0.15$						$\sigma_u^2 = 0.30$					
		Normal			Laplace			Normal			Laplace		
		ℓ_2	FP	FN	ℓ_2	FP	FN	ℓ_2	FP	FN	ℓ_2	FP	FN
500	True	0.09	1.11	0.00	0.09	1.1	0.00	0.09	1.24	0.00	0.09	1.19	0.00
		(0.02)	(2.36)	(0.00)	(0.02)	(2.55)	(0.00)	(0.02)	(2.75)	(0.00)	(0.02)	(2.62)	(0.00)
	Naive	0.48	1.38	0.00	0.73	1.35	0.00	0.48	1.1	0.00	0.73	1.36	0.00
		(0.05)	(2.92)	(0.00)	(0.07)	(2.9)	(0.00)	(0.05)	(2.3)	(0.00)	(0.07)	(3.3)	(0.00)
	SIMSELEX	0.21	0.00	0.00	0.23	0.00	0.00	0.21	0.00	0.00	0.34	0.00	0.00
		(0.07)	(0.00)	(0.00)	(0.07)	(0.00)	(0.00)	(0.07)	(0.00)	(0.00)	(0.11)	(0.00)	(0.00)
	Conic	0.27	-	-	0.34	-	-	0.27	-	-	0.34	-	-
		(0.04)	-	-	(0.07)	-	-	(0.04)	-	-	(0.07)	-	-
	Corrected	0.29	2.48	0.00	0.4	2.19	0.00	0.29	2.55	0.00	0.4	2.32	0.00
		(0.05)	(4.5)	(0.00)	(0.08)	(3.85)	(0.00)	(0.05)	(4.18)	(0.00)	(0.08)	(4.09)	(0.00)
1000	True	0.09	1.04	0.00	0.09	1.29	0.00	0.09	1.79	0.00	0.09	1.33	0.00
		(0.02)	(2.36)	(0.00)	(0.02)	(2.74)	(0.00)	(0.02)	(4.35)	(0.00)	(0.02)	(3.33)	(0.00)
	Naive	0.5	1.78	0.00	0.75	1.24	0.00	0.5	1.79	0.00	0.75	1.63	0.00
		(0.06)	(5.09)	(0.00)	(0.07)	(2.75)	(0.00)	(0.06)	(4.47)	(0.00)	(0.07)	(3.56)	(0.00)
	SIMSELEX	0.23	0.00	0.00	0.24	0.00	0.00	0.23	0.00	0.00	0.35	0.00	0.00
		(0.07)	(0.00)	(0.00)	(0.07)	(0.00)	(0.00)	(0.07)	(0.00)	(0.00)	(0.1)	(0.00)	(0.00)
	Conic	0.27	-	-	0.37	-	-	0.27	-	-	0.37	-	-
		(0.04)	-	-	(0.07)	-	-	(0.04)	-	-	(0.07)	-	-
	Corrected	0.3	3.78	0.00	0.42	2.94	0.00	0.3	4.2	0.00	0.42	3.54	0.00
		(0.06)	(6.6)	(0.00)	(0.08)	(5.53)	(0.00)	(0.06)	(6.29)	(0.00)	(0.08)	(5.93)	(0.00)
2000	True	0.1	2.12	0.00	0.1	6.32	0.00	0.1	2.19	0.00	0.1	1.57	0.00
		(0.02)	(5.68)	(0.00)	(0.02)	(10.9)	(0.00)	(0.02)	(5.57)	(0.00)	(0.02)	(3.61)	(0.00)
	Naive	0.51	1.87	0.00	0.77	6.12	0.00	0.51	2.01	0.00	0.77	1.64	0.00
		(0.05)	(4.7)	(0.00)	(0.07)	(10.9)	(0.00)	(0.05)	(4.52)	(0.00)	(0.07)	(3.39)	(0.00)
	SIMSELEX	0.23	0.00	0.00	0.23	0.00	0.00	0.23	0.00	0.00	0.36	0.00	0.00
		(0.07)	(0.00)	(0.00)	(0.07)	(0.00)	(0.00)	(0.07)	(0.00)	(0.00)	(0.11)	(0.04)	(0.00)
	Conic	0.28	-	-	0.38	-	-	0.28	-	-	0.38	-	-
		(0.04)	-	-	(0.07)	-	-	(0.04)	-	-	(0.07)	-	-
	Corrected	0.3	5.66	0.00	0.43	4.76	0.00	0.3	5.64	0.00	0.43	4.36	0.00
		(0.05)	(9.41)	(0.00)	(0.08)	(9.62)	(0.00)	(0.05)	(8.18)	(0.00)	(0.08)	(6.5)	(0.00)

Table 4.7: Comparison of estimators for linear regression with with the case of $\boldsymbol{\theta}_2$ based on ℓ_2 estimation error, average number of false positive (FP), and average number of false negative (FN) across 500 simulations.

p	Estimator	$\sigma_u^2 = 0.15$						$\sigma_u^2 = 0.30$					
		Normal			Laplace			Normal			Laplace		
		ℓ_2	FP	FN	ℓ_2	FP	FN	ℓ_2	FP	FN	ℓ_2	FP	FN
500	True	1.75	0.32	0.56	1.75	0.32	0.56	1.75	0.28	0.53	1.75	0.28	0.53
		(0.21)	(2.17)	(1.48)	(0.21)	(2.17)	(1.48)	(0.21)	(1.49)	(1.42)	(0.21)	(1.49)	(1.42)
	Naive	1.87	0.48	1.13	1.98	0.48	1.13	1.87	0.36	1.64	1.98	0.36	1.64
		(0.22)	(2.57)	(1.98)	(0.21)	(2.57)	(1.98)	(0.22)	(1.63)	(2.21)	(0.21)	(1.63)	(2.21)
	SIMSELEX	1.77	0.01	0.93	1.81	0.01	0.92	1.77	0.00	2.25	1.90	0.00	2.39
		(0.42)	(0.11)	(1.27)	(0.43)	(0.08)	(1.23)	(0.42)	(0.00)	(1.73)	(0.34)	(0.04)	(1.76)
	Cond	2.32	3.5	1.57	2.4	3.5	1.57	2.32	3.63	2.05	2.4	3.63	2.05
		(0.67)	(6.52)	(1.19)	(0.67)	(6.52)	(1.19)	(0.67)	(6.65)	(1.24)	(0.67)	(6.65)	(1.24)
	GMUS	1.61	0.91	0.02	1.77	0.91	0.02	1.61	0.41	0.1	1.77	0.41	0.1
		(0.08)	(1.17)	(0.13)	(0.07)	(1.17)	(0.13)	(0.08)	(0.73)	(0.3)	(0.07)	(0.73)	(0.3)
1000	True	1.75	0.35	0.47	1.77	0.35	0.47	1.75	0.32	0.62	1.77	0.32	0.62
		(0.18)	(1.71)	(1.32)	(0.21)	(1.71)	(1.32)	(0.18)	(1.47)	(1.56)	(0.21)	(1.47)	(1.56)
	Naive	1.89	0.52	1.23	1.99	0.52	1.23	1.89	0.46	2.05	1.99	0.46	2.05
		(0.21)	(2.29)	(2.03)	(0.2)	(2.29)	(2.03)	(0.21)	(3.18)	(2.34)	(0.2)	(3.18)	(2.34)
	SIMSELEX	1.8	0.01	1.06	1.81	0.01	1.08	1.8	0.00	2.79	1.92	0.00	2.80
		(0.4)	(0.12)	(1.35)	(0.41)	(0.13)	(1.41)	(0.4)	(0.04)	(1.76)	(0.34)	(0.00)	(1.80)
	Cond	2.46	4.83	1.7	2.43	4.83	1.7	2.46	3.99	2.19	2.43	3.99	2.19
		(0.66)	(8.76)	(1.19)	(0.68)	(8.76)	(1.19)	(0.66)	(7.22)	(1.19)	(0.68)	(7.22)	(1.19)
2000	True	1.78	0.56	0.57	1.76	0.56	0.57	1.78	0.52	0.66	1.76	0.52	0.66
		(0.19)	(3.02)	(1.46)	(0.21)	(3.02)	(1.46)	(0.19)	(3.25)	(1.56)	(0.21)	(3.25)	(1.56)
	Naive	1.91	0.84	1.36	2.02	0.84	1.36	1.91	0.48	2.08	2.02	0.48	2.08
		(0.21)	(4.69)	(2.09)	(0.19)	(4.69)	(2.09)	(0.21)	(2.08)	(2.33)	(0.19)	(2.08)	(2.33)
	SIMSELEX	1.83	0.00	1.19	1.83	0.00	1.35	1.83	0.00	3.03	1.96	0.00	3.07
		(0.41)	(0.00)	(1.34)	(0.37)	(0.04)	(1.56)	(0.41)	(0.00)	(1.72)	(0.30)	(0.04)	(1.75)
	Cond	2.46	5.76	1.78	2.43	5.76	1.78	2.46	5.82	2.36	2.43	5.82	2.36
		(0.65)	(10.1)	(1.22)	(0.63)	(10.1)	(1.22)	(0.65)	(10.2)	(1.22)	(0.63)	(10.2)	(1.22)

Table 4.8: Comparison of estimators for logistic regression with with the case of $\boldsymbol{\theta}_2$ based on ℓ_2 estimation error, average number of false positive (FP), and average number of false negative (FN) across 500 simulations.

σ_u^2	p	ℓ_2			FP			FN		
		True	Naive	SIM-SELEX	True	Naive	SIM-SELEX	True	Naive	SIM-SELEX
0.15	500	0.88	1.32	1.03	3.92	2.65	0.00	0.00	0.00	0.00
		(0.11)	(0.09)	(0.17)	(3.93)	(3.44)	(0.00)	(0.00)	(0.00)	(0.00)
	1000	0.92	1.34	1.04	4.95	3.23	0.00	0.00	0.00	0.00
		(0.11)	(0.09)	(0.17)	(4.95)	(3.76)	(0.00)	(0.00)	(0.00)	(0.00)
	2000	0.95	1.37	1.08	5.23	3.63	0.00	0.00	0.00	0.00
		(0.1)	(0.09)	(0.17)	(5.15)	(4.47)	(0.00)	(0.00)	(0.00)	(0.00)
0.30	500	0.89	1.54	1.22	3.64	2.03	0.00	0.00	0.00	0.08
		(0.11)	(0.08)	(0.18)	(3.89)	(2.81)	(0.00)	(0.00)	(0.00)	(0.27)
	1000	0.92	1.56	1.25	4.78	2.47	0.00	0.00	0.00	0.11
		(0.11)	(0.09)	(0.19)	(5.31)	(3.61)	(0.00)	(0.00)	(0.00)	(0.31)
	2000	0.96	1.58	1.27	5.29	3.13	0.00	0.00	0.00	0.17
		(0.11)	(0.08)	(0.18)	(5.65)	(4.16)	(0.00)	(0.00)	(0.00)	(0.4)

Table 4.9: Comparison of estimators for Cox survival models for the case $\boldsymbol{\theta}_2$ based on ℓ_2 estimation error, average number of false positive (FP), average number of false negative (FN) across 500 simulations.

linear means function in the extrapolation step are compared. Table 4.10 presents the mean and median ℓ_2 error across 500 simulations for both linear and logistic regression — the simulation configurations are as described in Section 4.4.1 (linear regression) and Section 4.4.2 (logistic regression).

In the case of linear regression, the nonlinear extrapolation function results in a SIMSELEX estimator with a smaller median ℓ_2 error, but a higher mean ℓ_2 error when compared to the quadratic extrapolation function. Specifically, for small measurement error variance ($\sigma_u^2 = 0.15$), the extrapolation methods give very consistent results as measured by mean and median ℓ_2 error. However, for large measurement error variance ($\sigma_u^2 = 0.3$), there are some instances where the mean ℓ_2 error for nonlinear extrapolation is much larger than for quadratic extrapolation.

In the case of logistic regression, the quadratic extrapolation function consistently outperforms the nonlinear means function regardless of whether mean or median ℓ_2 error is used as criterion. A closer inspection of the simulation results suggest one possible explanation for the superiority of quadratic extrapolation: in many of the simulated datasets,

Model	p	ME type	$\sigma_u^2 = 0.15$				$\sigma_u^2 = 0.30$			
			Mean ℓ_2		Median ℓ_2		Mean ℓ_2		Median ℓ_2	
			NL	Quad	NL	Quad	NL	Quad	NL	Quad
Linear	500	Normal	0.34	0.32	0.31	0.32	0.5	0.5	0.44	0.5
		Laplace	0.37	0.33	0.31	0.32	0.55	0.51	0.47	0.52
	1000	Normal	0.34	0.34	0.32	0.34	1.14	0.53	0.48	0.53
		Laplace	0.33	0.34	0.32	0.34	0.52	0.51	0.46	0.51
	2000	Normal	0.35	0.35	0.32	0.34	0.79	0.54	0.5	0.55
		Laplace	0.92	0.36	0.34	0.36	0.58	0.55	0.5	0.54
Logistic	500	Normal	3.82	2.65	2.81	2.65	21.66	2.73	3.3	2.69
		Laplace	8.28	2.67	2.82	2.64	6.02	2.76	3.31	2.69
	1000	Normal	7.99	2.7	2.84	2.67	7.46	2.77	3.37	2.72
		Laplace	18.46	2.63	2.81	2.64	5.63	2.72	3.33	2.68
	2000	Normal	5.92	2.67	2.84	2.65	5.84	2.75	3.34	2.69
		Laplace	4.28	2.69	2.84	2.65	5.97	2.79	3.38	2.74

Table 4.10: Monte Carlo mean and median ℓ_2 error of SIMSELEX estimator using nonlinear means (NL) and quadratic (Quad) extrapolation function for linear and logistic regression.

the nonlinear means function results in extrapolants very far from the true values. This results in the large mean and median ℓ_2 error values. We attempted increasing the value of B, the number of pseudo-datasets used for the simulation step, but this did not alleviate the problem. It might be possible that an increase in both the number of λ values and the value of B can improve performance of the nonlinear extrapolation function, but this becomes computationally demanding and seems unnecessary given the good performance of quadratic extrapolation.

4.8.5. Post-Selection SIMEX Estimator

When implementing SIMSELEX, a natural question is whether the performance of the method can be improved by implementing standard SIMEX methodology after the variable selection step. That is, a method of simulation–selection–simulation–extrapolation could be implemented. The second simulation step is therefore implemented using only the selected variables, and no penalty method is used since the number of variables in the model has already been reduced. This estimator is referred as the post-selection SIMEX estimator. The section compares the performance of the SIMSELEX and the post-selection SIMEX estimator in the linear and logistic regression settings.

The data were generated as outlined in Section 4.4.1 and Section 4.4.2. Only the simulation configurations with Normal measurement error and the coefficients $\boldsymbol{\theta}_1$ were considered. For the post-selection SIMEX estimator, the grid of added measurement error level λ in the simulation step consists of 5 equally spaced values from 0.01 to 2 and $B = 100$ sets of pseudo-data were generated for each value of λ (this corresponds to implementation of SIMSELEX). In the extrapolation step, both the nonlinear means function and quadratic function were considered. The estimators are compared based on ℓ_2 estimation error. The simulation results are presented below in Table 4.11.

	σ_u^2	p	SIMSELEX		Post-sel. SIMEX	
			Nonlin	Quad	Nonlin	Quad
Linear	0.15	500	0.34	0.32	0.20	0.20
			(0.24)	(0.1)	(0.07)	(0.06)
		1000	0.37	0.33	0.20	0.19
			(0.65)	(0.11)	(0.07)	(0.06)
		2000	0.34	0.34	0.20	0.20
			(0.31)	(0.1)	(0.07)	(0.07)
	0.30	500	0.50	0.50	0.30	0.28
			(0.35)	(0.14)	(0.10)	(0.09)
		1000	0.55	0.51	0.30	0.28
			(0.60)	(0.15)	(0.11)	(0.10)
		2000	1.14	0.53	0.3	0.28
			(8.12)	(0.15)	(0.11)	(0.10)
Logistic	0.15	500	2.64	3.20	1.05	0.90
			(2.32)	(0.47)	(0.58)	(0.39)
		1000	6.42	3.20	0.99	0.88
			(86.1)	(0.47)	(0.52)	(0.38)
		2000	2.61	3.21	1.07	0.95
			(0.25)	(0.44)	(0.45)	(0.36)
	0.30	500	2.73	3.21	1.34	1.15
			(0.42)	(0.50)	(1.16)	(0.39)
		1000	2.75	3.21	1.37	1.24
			(0.28)	(0.52)	(0.54)	(0.37)
		2000	2.76	3.20	1.36	1.25
			(0.22)	(0.49)	(0.54)	(0.43)

Table 4.11: Comparison of SIMSELEX and post-selection SIMEX estimators using mean ℓ_2 error for linear and logistic model. Nonlinear (Nonlin) and quadratic (Quad) extrapolation were considered.

It can be seen that the post-selection SIMEX estimator gives smaller ℓ_2 estimation error than the SIMSELEX estimator in all the considered settings. The gain is most

considerable in the case of logistic regression, especially when large measurement error exists. The nonlinear and the quadratic extrapolation function have roughly the same performance in the linear model, while the quadratic function has better performance in the logistic model.

4.8.6. Computation Time

Table 4.12 presents the median computation times for the different estimators in the linear and logistic models as considered in the simulation studies of Section 4.4. In the case of the linear model, the median computation time for SIMSELEX increased by approximately 150% when going from 500 to 2000 variables, whereas the corrected scores lasso increased by around 1500% and the conic estimator increased by around 1800%. For logistic regression, the median computation time for SIMSELEX increased by 120%, while GMUS computation time increased by over 5000%. As noted in Sørensen et al. (2018), GMUS is not feasible for implementation with a large number of variables. The computation times for the conditional scores lasso for logistic regression are misleading and appear overly optimistic; the computation time here is very low as there is no sample-specific selection of tuning parameter taking place in the simulation study. In practice, this will be done using the elbow method as discussed in Appendix 4.8.2.

Model	p	SIMSELEX	Corrected / Conditional	Conic	GMUS
Linear	500	428	58	349	-
	1000	631	264	888	-
	2000	1064	1016	6597	-
Logistic	500	572	7	-	330
	1000	798	15	-	>4.5 hours
	2000	1248	43	-	>4.5 hours
Survival	500	5435	-	-	-
	1000	7924	-	-	-
	2000	10461	-	-	-

Table 4.12: Median computation time (in second) for different estimators. For the conditional score lasso and GMUS it is the median time to generate a coefficient path with 25 values of the tuning parameter.

Chapter 5
Summary and Future Directions

5.1. Summary

The thesis proposes new estimators that correct for measurement errors in the contexts of density estimation and errors-in-variables models.

First, the phase function-based estimators are established for heteroscedastic density deconvolution and linear errors-in-variables model. Compared to the existing estimators in the literature, these new estimators have a primary advantage of putting minimal assumptions on the distribution of measurement errors while still having competitive performance. Therefore, the phase function-based estimators are useful for practitioners in a wide range of situations when correcting for measurement errors is necessary but the knowledge about measurement errors on the data is limited.

Additionally, the thesis proposes SIMSELEX that both achieve sparsity and accommodate for measurement errors in high-dimensional statistical models. As an extension of the traditional simulation-extrapolation approach, the SIMSELEX makes double use of lasso methodology and can be applied in many errors-in-variables settings. As a result, the SIMSELEX provides practitioners with a flexible tool to address additional complexity caused by measurement errors to high dimensional settings.

5.2. Future Directions

5.2.1. Phase Function Method

In chapter 3, the phase function method is used to estimate the coefficients of the linear errors-in-variables. It can be seen that the key relationship between the phase function of the outcome and the linear combination of covariates holds even when the errors are

heteroscedastic, i.e, the model error and measurement error for each observation may have different variances/scale. A possible way to adjust for such heteroscedasticity is to use the weighted empirical phase function, as defined in chapter 2, where the weights are adaptive to the error variances.

Furthermore, chapter 3 also suggests that the phase function method can be used with arbitrary number of error-free and error-prone covariates in the model. Hence, it can be incorporated into more complicated linear models with additional structure on the coefficients. In such situation, the phase function estimator can be computed by minimizing a similar discrepancy function with the corresponding constraints. For example, in high-dimensional linear regression setting, such a desirable structure is sparsity. In this case, an ℓ_1 regularization term can be added into the discrepancy function (3.5) to achieve sparsity.

5.2.2. Measurement errors on high-dimensional settings

There are still many open questions on the effect of measurement errors on high dimensional statistical models that can be explored. For example, it is often of interest to model the conditional dependence structure among a large set of variables, such as a set of genes regulating a biological process. Also, because the number of variables can be much greater than the sample size, it is essential to perform dimension reduction before conducting any analysis. Many new statistical methods have been proposed for these tasks in the case of clean data, but not many of them account for measurement errors that can exist in the observations. Therefore, future research will continue to develop new correction methods for high dimensional models, so that practitioners can make proper inference when dealing with complex and noisy data.

Bibliography

Apanasovich, T. V., Carroll, R. J., and Maity, A. (2009). Simex and standard error estimation in semiparametric measurement error models. Electronic journal of statistics, 3:318.

Armstrong, B. (1985). Measurement error in the generalised linear model. Communications in Statistics-Simulation and Computation, 14(3):529–544.

Belloni, A., Rosenbaum, M., and Tsybakov, A. B. (2017). Linear and conic programming estimators in high dimensional errors-in-variables models. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 79(3):939–956.

Carroll, R. J. and Hall, P. (1988). Optimal rates of convergence for deconvolving a density. Journal of the American Statistical Association, 83(404):1184–1186.

Carroll, R. J., Ruppert, D., Stefanski, L. A., and Crainiceanu, C. M. (2006). Measurement error in nonlinear models: a modern perspective. CRC press.

Carroll, R. J. and Stefanski, L. A. (1990). Approximate quasi-likelihood estimation in models with surrogate predictors. Journal of the American Statistical Association, 85(411):652–663.

Cook, J. R. and Stefanski, L. A. (1994). Simulation-extrapolation estimation in parametric measurement error models. Journal of the American Statistical association, 89(428):1314–1328.

Datta, A., Zou, H., et al. (2017). Cocolasso for high-dimensional error-in-variables regression. The Annals of Statistics, 45(6):2400–2426.

De Vito, S., Massera, E., Piga, M., Martinotto, L., and Di Francia, G. (2008). On field calibration of an electronic nose for benzene estimation in an urban pollution monitoring scenario. Sensors and Actuators B: Chemical, 129(2):750–757.

Delaigle, A. (2008). An alternative view of the deconvolution problem. Statistica Sinica, pages 1025–1045.

Delaigle, A. and Hall, P. (2016). Methodology for non-parametric deconvolution when the error distribution is unknown. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 78(1):231–252.

Delaigle, A., Hall, P., and Meister, A. (2008). On deconvolution with repeated measurements. The Annals of Statistics, pages 665–685.

Delaigle, A. and Meister, A. (2008). Density estimation with heteroscedastic error. Bernoulli, pages 562–579.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. Journal of the royal statistical society. Series B (methodological), pages 1–38.

Diggle, P. J. and Hall, P. (1993). A fourier approach to nonparametric deconvolution of a density estimate. Journal of the Royal Statistical Society. Series B (Methodological), pages 523–531.

Erickson, T., Jiang, C. H., and Whited, T. M. (2014). Minimum distance estimation of the errors-in-variables model using linear cumulant equations. Journal of Econometrics, 183(2):211–221.

Erickson, T. and Whited, T. M. (2002). Two-step gmm estimation of the errors-in-variables model using high-order moments. Econometric Theory, 18(3):776–799.

Fan, J. (1991a). Asymptotic normality for deconvolution kernel density estimators. Sankhyā: The Indian Journal of Statistics, Series A, pages 97–110.

Fan, J. (1991b). On the optimal rates of convergence for nonparametric deconvolution problems. The Annals of Statistics, pages 1257–1272.

Fan, J. (1992). Deconvolution with supersmooth distributions. Canadian Journal of Statistics, 20(2):155–169.

Fan, J. and Truong, Y. K. (1993). Nonparametric regression with errors in variables. The Annals of Statistics, pages 1900–1925.

Feuerverger, A., Mureika, R. A., et al. (1977). The empirical characteristic function and its applications. The annals of Statistics, 5(1):88–97.

Friedman, J., Hastie, T., and Tibshirani, R. (2001). The elements of statistical learning, volume 1. Springer series in statistics New York.

Gillard, J. (2014). Method of moments estimation in linear regression with errors in both variables. Communications in Statistics-Theory and Methods, 43(15):3208–3222.

Guo, Y. and Little, R. J. (2011). Regression analysis with covariates that have heteroscedastic measurement error. Statistics in medicine, 30(18):2278–2294.

Hall, P. and Qiu, P. (2005). Discrete-transform approach to deconvolution problems. Biometrika, pages 135–148.

Hastie, T., Tibshirani, R., and Wainwright, M. (2015). Statistical learning with sparsity: the lasso and generalizations. CRC press.

Hein, A.-M. K., Richardson, S., Causton, H. C., Ambler, G. K., and Green, P. J. (2005). Bgx: a fully bayesian integrated approach to the analysis of affymetrix genechip data. Biostatistics, 6(3):349–373.

Higdon, R. and Schafer, D. W. (2001). Maximum likelihood computations for regression with measurement error. Computational statistics & data analysis, 35(3):283–299.

Koul, H. L., Song, W., et al. (2014). Simulation extrapolation estimation in parametric models with laplace measurement error. Electronic Journal of Statistics, 8(2):1973–1995.

Küchenhoff, H., Mwalili, S. M., and Lesaffre, E. (2006). A general method for dealing with misclassification in regression: The misclassification simex. Biometrics, 62(1):85–96.

Kunsch, H. R. (1989). The jackknife and the bootstrap for general stationary observations. The annals of Statistics, pages 1217–1241.

Lederer, J. (2013). Trust, but verify: benefits and pitfalls of least-squares refitting in high dimensions. arXiv preprint arXiv:1306.0113.

Lee, M., Shen, H., Burch, C., and Marron, J. (2010). Direct deconvolution density estimation of a mixture distribution motivated by mutation effects distribution. Journal of Nonparametric Statistics, 22(1):1–22.

Loh, P.-L. and Wainwright, M. J. (2011). High-dimensional regression with noisy and missing data: Provable guarantees with non-convexity. In Advances in Neural Information Processing Systems, pages 2726–2734.

Lombard, F. (2005). Nonparametric confidence bands for a quantile comparison function. Technometrics, 47(3):364–371.

Ma, Y. and Li, R. (2010). Variable selection in measurement error models. Bernoulli: official journal of the Bernoulli Society for Mathematical Statistics and Probability, 16(1):274.

McIntyre, J. and Stefanski, L. A. (2011). Density estimation with replicate heteroscedastic measurements. Annals of the Institute of Statistical Mathematics, 63(1):81–99.

Meister, A. (2006). Density estimation with normal measurement error with unknown variance. Statistica Sinica, pages 195–211.

Neumann, M. H. and Hössjer, O. (1997). On the effect of estimating the error density in nonparametric deconvolution. Journal of Nonparametric Statistics, 7(4):307–330.

Nghiem, L. and Potgieter, C. J. (2018). Density estimation in the presence of heteroscedastic measurement error of unknown type using phase function deconvolution. Statistics in medicine.

Prentice, R. (1982). Covariate measurement errors and parameter estimation in a failure time regression model. Biometrika, 69(2):331–342.

Qian, J., Hastie, T., Friedman, J., Tibshirani, R., and Simon, N. (2013). Glmnet for matlab 2013. URL http://www. stanford. edu/~ hastie/glmnet_matlab.

Reiersøl, O. (1941). Confluence analysis by means of lag moments and other methods of confluence analysis. Econometrica: Journal of the Econometric Society, pages 1–24.

Rosenbaum, M., Tsybakov, A. B., et al. (2010). Sparse recovery under matrix uncertainty. The Annals of Statistics, 38(5):2620–2651.

Rosenbaum, M., Tsybakov, A. B., et al. (2013). Improved matrix uncertainty selector. In From Probability to Statistics and Back: High-Dimensional Models and Processes–A Festschrift in Honor of Jon A. Wellner, pages 276–290. Institute of Mathematical Statistics.

Schafer, D. W. and Purdy, K. G. (1996). Likelihood analysis for errors-in-variables regression with replicate measurements. Biometrika, 83(4):813–824.

Simon, N., Friedman, J., Hastie, T., and Tibshirani, R. (2013). A sparse-group lasso. Journal of Computational and Graphical Statistics, 22(2):231–245.

Sørensen, Ø., Frigessi, A., and Thoresen, M. (2015). Measurement error in lasso: Impact and likelihood bias correction. Statistica Sinica, pages 809–829.

Sørensen, Ø., Hellton, K. H., Frigessi, A., and Thoresen, M. (2018). Covariate selection in high-dimensional generalized linear models with measurement error. Journal of Computational and Graphical Statistics, (just-accepted).

Stefanski, L. A. and Carroll, R. J. (1987). Conditional scores and optimal scores for generalized linear measurement-error models. Biometrika, 74(4):703–716.

Stefanski, L. A. and Carroll, R. J. (1990). Deconvolving kernel density estimators. Statistics, 21(2):169–184.

Stefanski, L. A. and Cook, J. R. (1995). Simulation-extrapolation: the measurement error jackknife. Journal of the American Statistical Association, 90(432):1247–1256.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society. Series B (Methodological), pages 267–288.

van der Vaart, A. W. (2000). Asymptotic statistics, volume 3. Cambridge university press.

Xue-Kun Song, P. (2000). Multivariate dispersion models generated from gaussian copula. Scandinavian Journal of Statistics, 27(2):305–320.

Yancey, H., Geer, M., and Price, J. (1951). An investigation of the abrasiveness of coal and its associated impurities. Mining Engineering, 3:262–268.

Yi, G. Y., Tan, X., and Li, R. (2015). Variable selection and inference procedures for marginal analysis of longitudinal data with missing observations and covariate measurement error. Canadian Journal of Statistics, 43(4):498–518.

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

Zakharkin, S. O., Kim, K., Mehta, T., Chen, L., Barnes, S., Scheirer, K. E., Parrish, R. S., Allison, D. B., and Page, G. P. (2005). Sources of variation in affymetrix microarray experiments. BMC bioinformatics, 6(1):214.