LINEAR REGRESSION WITH LAPLACE MEASUREMENT ERROR

by

CHENDI CAO

B.S.,Kansas State University, 2013

A REPORT

submitted in partial fulfillment of the
requirements for the degree

MASTER OF SCIENCE

Department of Statistics
College of Arts and Sciences

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2016

Approved by:

Major Professor
Dr. Weixing Song

Abstract

In this report, an improved estimation procedure for the regression parameter in simple linear regression models with the Laplace measurement error is proposed. The estimation procedure is made feasible by a Tweedie type equality established for $E(X|Z)$, where $Z = X + U$, X and U are independent, and U follows a Laplace distribution. When the density function of X is unknown, a kernel estimator for $E(X|Z)$ is constructed in the estimation procedure. A leave-one-out cross validation bandwidth selection method is designed. The finite sample performance of the proposed estimation procedure is evaluated by simulation studies. Comparison study is also conducted to show the superiority of the proposed estimation procedure over some existing estimation methods.

Table of Contents

List of Figures

List of Tables

Acknowledgments

Firstly, I would like to express my sincere gratitude to my major advisor Dr. Weixing Song for the continuous support for my M.S. study and related research, for his patience, motivation and immense knowledge. Dr. Song taught me how to question thoughts and express ideas. His patience and support helped me overcome many crisis situations and finish this report.

I would like to thank my committee members, Dr. Juan Du and Dr. Wei-Wen Hsu for all of their guidance, constant enthusiasm and encouragement through this process.

I would like to thank all graduate students, faculties, friends from K-state statistics department. I am very grateful to all of you.

Finally, I would especially like to thank my family for the love, support, and constant encouragement I have gotten over the years. In particular, I would like to thank my parents, my wife. They told me never give up, and I undoubtedly could not have done this without you.

Chapter 1

Introduction

1.1 Errors-in-variables(EV) models

Regression modeling is one of the most popular statistical inference tools used for fitting the relationship between a scalar quantity Y and an explanatory covariate or covariates X. The classical regression model takes the form of

$$Y = m(X) + \varepsilon,$$

where $m(x) = E(Y|X = x)$ is the regression function, and ε is the error term accounted for any other variability of Y which cannot explained by X, $E\big(\sum(X)\big) = 0$ When both Y and X are available, which is often the case in the classical regression setup, myriad of estimation procedures are proposed for the regression function whenever $m(x)$ has a parametric form or not.

However, in some practice we cannot observe the covariates X directly. Instead, a surrogate, say Z, is available. It is commonly assumed in the measurement literature that the surrogate Z and the covariate X are related in an additive way, that is $Z = X + U$, where U is called measurement error. The research of interest thus becomes how to make statistical inference on $m(x)$ based on the data (Z, Y) in the following errors-in-variables regression model

$$Y = m(X) + \varepsilon, \quad Z = X + U.$$

Examples of measurement error are abundant in real applications and literature. Most medical variables, such as blood pressure, pulse rate, temperature, and blood chemistries, are measured with non-negligible error; Variables in agricultural studies such as the precipitation, soil nitrogen content, degree of pest infestation, farm crop acreage allocation, and the like cannot be measured precisely. In management sciences, social sciences, and nearly every other field many variables can only be measured with error. For more examples, see Carroll et al. (2006).[1]

Although it is still debatable, Adcock(1877,1878)[2][3] is usually regarded as the first person specifically to consider such models. Depending on X being random or fixed, measurement error models can be classified into three subgroups.

- **Functional Models**: X_i's are fixed unknown constants.
- **Structural Models**: X_i's are $i.i.d$ and independent of the errors.
- **Ultra-Structural Models**: X_i's are independent, but not identically distributed, possibly with different means, homoscedasticity remains.

In this report, we are considering structural measurement models.

It might be tempting to consider applying the classical statistical inference procedures by simply ignoring the measurement errors, that is, replacing X with Z in all relevant statistical procedures. This is the so-called the naive procedures. According to Carroll et al. (2006),[1] the naive methods generally induces three negative consequences in statistical inferences:

- It causes bias in parameter estimation for statistical models;

- It leads to a loss of power, sometimes profound, for detecting interesting relationship among variables;
- It masks the features of the data, making graphical model analysis difficult.

So in estimation theory, how to handle the bias caused by the measurement error remains the primary research interest in the measurement error literature. Correcting for measurement error requires additional information or data. Myriad approaches to carry out the corrections for measurement errors have been proposed, including the direct bias correction, moment based approaches, likelihood based techniques, regression calibration, SIMEX and techniques based on modifying estimating equations.

In this report, we focus on the simple errors-in-variables linear regression model, that is

$$Y = \alpha + \beta X + \varepsilon, \quad Z = X + U. \tag{1.1}$$

The proposed estimation methods can be readily extended to multiple linear regression case, even to nonlinear and parametric regression models. Some typical assumptions on model (1.1) include $EU = E(\varepsilon) = 0$, $\mathrm{Var}(U) = \sigma_u^2 > 0$, $Var(\varepsilon) = \sigma_\varepsilon^2 > 0$, and X, ε, U are independent.

1.1.1 Identifiability

Identifiability present a big challenge in the early development of measurement error modeling.

For simple linear errors-in-variables regression models, if we assume that ε, X, U are independent, each being normally distributed with mean $0, \mu_x, 0$, and variances σ_ε^2, σ_x^2 and σ_u^2, respectively, then one can easily find two different sets of values for α and β such that (Y, Z) possesses the same distribution. In fact, if there is no auxiliary information available, μ_x is the only parameter that is identifiable.

Under the normality assumptions, there are six side assumptions found in the literature that make the structural model identifiable.

(a). the ratio of the error variance $\lambda = \sigma_\varepsilon^2/\sigma_u^2$ is known;

(b). the reliability ratio (attenuation coefficient) $\kappa_x = \sigma_x^2/(\sigma_x^2 + \sigma_u^2)$ is known;

(c). σ_u^2 is known;

(d). σ_ε^2 is known;

(e). both σ_u^2 and σ_ε^2 are known;

(f). β_0 is known, and $EZ \neq 0$.

These identifiable conditions are used in the different contexts. For example, (a) is the most popular of these additional assumptions and is the one with the most published theoretical results; (b) is commonly found in the social science and psychology literatures and it is often referred as heritability in genetics; (c) has gained attention recently and is a popular assumption when working with nonlinear models. In the case of σ_u^2 being unknown, an estimate of σ_u^2 can be constructed by using replications of Z at X; (d) is less useful and cannot be used to make the equation error model or the ME model with more than one explanatory variable identifiable; (e) frequently leads to the same estimates as those for (a) and also leads to an over-identified model. It worths to point out that (f) only apply to one predictor case, and it does not make the normal model, with more than one explanatory variable, identifiable.

The most important theoretical result on the identifiability of the simple linear measurement error regression models belongs to Reiersol (1950)[4]. He proved that

- if (u, ε) are jointly normal, then X is not normal if and only if β_0, β_1 are identifiable.
- when u and ε are independent, then X being nonnormally distributed is sufficient for β_0 and β_1 to be identifiable in the structural model.

- if X is normal, u and ε are independent, then β_0, β_1 are identifiable if and only if neither u nor ε has a distribution that is divisible by a normal distribution.

1.2 Estimation Procedures in Structural EV Models

A common way to estimate the regression coefficients in the normal structural errors-in-variables regression models is the maximum likelihood estimation procedure. Based on the structural relationship, we have

$$
\begin{aligned}
&EZ = EX = \mu, \quad EY = \beta_0 + \beta_1 \mu, \\
&\mathrm{Var}(Z) = \sigma_x^2 + \sigma_u^2, \quad \mathrm{Var}(Y) = \beta_1^2 \sigma_x^2 + \sigma_\varepsilon^2, \quad \mathrm{Cov}(Z, Y) = \beta_1 \sigma_x^2.
\end{aligned}
$$

The invariance properties of ML estimates implies that the solutions of the following equations are the valid MLEs of the six unknown parameters in the simple linear normal structural errors-in-variables regression model:

$$
\begin{aligned}
&\bar{Z}_n = \hat{\mu}, \quad \bar{Y}_n = \hat{\beta}_0 + \hat{\beta}_1 \hat{\mu}, \quad S_{ZZ} = \hat{\sigma}_x^2 + \hat{\sigma}_u^2, \\
&S_{YY} = \hat{\beta}_1^2 \hat{\sigma}_x^2 + \hat{\sigma}_\varepsilon^2, \quad S_{ZY} = \hat{\beta}_1 \hat{\sigma}_x^2,
\end{aligned}
\tag{1.2}
$$

if $S_{ZZ} \geq S_{ZY}/\hat{\beta}_1$, $S_{YY} \geq \hat{\beta}_1 S_{ZY}$, $S_{ZZ} \geq \hat{\sigma}_u^2$, $S_{YY} \geq \hat{\sigma}_\varepsilon^2$, $\mathrm{sign}(S_{ZY}) = \mathrm{sign}(\hat{\beta}_1)$, where S_{ZY} is the sample covariance between Z_i's and Y_i's. For any generic random variables (X, Y), S_{XY} denotes the sample covariance of X and Y based on a sample from (X, Y), that is

$$
S_{XY} = \frac{1}{n} \sum_{i=1}^{n} (X_i - \bar{X})(Y_i - \bar{Y}),
$$

with $\bar{X}$ and $\bar{Y}$ being the sample mean based on the data from X and Y. The regression line solving the above equations lies between the standard regression line of Y against Z

and the standard regression line of Z on Y. In fact, we can show that

$$|\hat{\beta}_{1R}| = \frac{|S_{ZY}|}{S_{ZZ}} \leq |\hat{\beta}_1| \leq \frac{S_{YY}}{|S_{ZY}|} = \left|\frac{1}{\hat{\beta}_{1I}}\right|,$$

where $\hat{\beta}_{1R}$ is the estimated slope from the standard regression of Y on Z, and $\hat{\beta}_{1I}$ is the estimated slope from the standard regression of Z on Y. Moran (1971)[5] illustrated the lack of uniqueness of the MLE. Let γ be a small positive quantity less than both $|\hat{\beta}_1|$ and $\hat{\sigma}_\varepsilon^2 \hat{\beta}_1^{-1} \hat{\sigma}_x^{-2}$. Replace the quantities $\hat{\beta}_0$, $\hat{\beta}_1$, $\hat{\sigma}_x^2$, $\hat{\sigma}_u^2$, and $\hat{\sigma}_\varepsilon^2$ in (refeq1.2) by $\hat{\beta}_0 - \gamma\hat{\mu}$, $\hat{\beta}_1 + \gamma$, $\hat{\beta}_1 \hat{\sigma}_x^2 (\hat{\beta}_1 + \gamma)^{-1}, \hat{\sigma}_u^2 + \gamma \hat{\sigma}_x^2 (\hat{\beta}_1 + \gamma)^{-1}$, and $\hat{\sigma}_\varepsilon^2 - \hat{\beta}_1 \gamma \hat{\sigma}_x^2$, respectively. Then the five equations remain unchanged so that if one set of estimates is an ML solution, so is the other.

1.2.1 MLE under Identifiability Conditions

To uniquely determine the MLE, we need to adopt some identifiability conditions to make sure the equations (1.2) have unique solutions. In the following, we list the MLEs of β_0, β_1 under different identifiability conditions. The MLEs of other unknown parameters in the model can be readily obtained, however, the discussion of these estimates will be omitted, since our focus is on the regression coefficients.

When $\lambda = \sigma_\varepsilon^2/\sigma_u^2$ is known. If we assume that $\lambda = \sigma_\varepsilon^2/\sigma_u^2$ is known, then the MLEs are

$$\begin{aligned}\hat{\beta}_1 &= \frac{S_{YY} - \lambda S_{ZZ} + \sqrt{(S_{YY} - \lambda S_{ZZ})^2 + 4\lambda S_{ZY}^2}}{2S_{ZY}},\\ \hat{\beta}_0 &= \bar{Y}_n - \hat{\beta}_1 \bar{Z}_n, \quad \hat{\sigma}_x^2 = S_{ZY}/\hat{\beta}_1.\end{aligned}$$

In fact, we can show that

$$(\hat{\beta}_0, \hat{\beta}_1) = \text{argmin}_{\beta_0,\beta_1} \sum_{i=1}^n \left[\frac{Y_i - \beta_0 - \beta_1 Z_i}{\sqrt{\lambda + \beta_1^2}}\right]^2.$$

For $\lambda = 1$, we can see that, geometrically, $(\hat{\beta}_0, \hat{\beta}_1)$ minimizes the squared distance from points (Z_i, Y_i)'s from a straight line with intercept β_0 and slope β_1.

When σ_u^2 is known. If σ_u^2 is assumed to be known, then solving equations (1.2) we obtain the following estimates

$$\hat{\beta}_1 = (S_{ZZ} - n\sigma_u^2)^{-1} S_{ZY}, \quad \hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{Z},$$
$$\hat{\sigma}_\varepsilon^2 = S_{YY} - \hat{\beta}_1 S_{ZY}, \quad \hat{\sigma}_x^2 = S_{ZZ} - \sigma_u^2.$$

For the quantities defined above to be a proper estimates, $\hat{\sigma}_x^2$ and $\hat{\sigma}_\varepsilon^2$ must be nonnegative. One can show that the estimates $\hat{\sigma}_x^2$, $\hat{\sigma}_\varepsilon^2$ will be positive if and only if $S_{YY}(S_{ZZ} - \sigma_u^2) - S_{ZY}^2 > 0$.

When the reliability ratio $k = \sigma_x^2/(\sigma_x^2 + \sigma_u^2)$ is known. The MLE of β_1 when k is known has the form of $\hat{\beta}_1 = k^{-1}\hat{\beta}_{\text{Naive}}$, where $\hat{\beta}_{\text{Naive}}$ is the naive estimate of β_1 which is simply the least squares estimate by regression Y on Z.

The MLE under other identifiability conditions are also easy to derive. The details are omitted here for the sake of brevity. More details on this topic can be found in Fuller (1987)[6], Cheng and van Ness (1992)[7], Buonaccorsi (2010)[8] and the references therein.

1.2.2 Bias-Corrected Estimate of the Regression Coefficients

Under the condition of σ_u^2 being known, the most popular estimate for β_1 is the bias-corrected estimator

$$\hat{\beta}_{\text{BC}} = (S_{ZZ} - \sigma_u^2)^{-1} S_{ZY} = \frac{\sum_{i=1}^n (Z_i - \bar{Z})(Y_i - \bar{Y})}{\sum_{i=1}^n (Z_i - \bar{Z})^2 - n\sigma_u^2}. \tag{1.3}$$

Although we derived $\hat{\beta}_{\text{BC}}$ as the MLE of β_1 under the normality assumptions, the other derivation of $\hat{\beta}_{\text{BC}}$ indeed does not need the distributional assumptions on the random components in linear EV regression models. It can be obtained by directly modifying the moment conditions. In fact, from $\text{Cov}(Y, Z) = \beta_1\sigma_x^2$, $\text{Var}(Z) = \sigma_x^2 + \sigma_u^2$, we can immediately get $\beta_1 = (\text{Var}(Z) - \sigma_u^2)\text{Cov}(Z, Y)$. Replacing $\text{Cov}(Z, Y)$ and $\text{Var}(Z)$ with the sample analogs leads to the bias-corrected estimate (1.3).

Because of its simple form and free of distributional assumptions, the bias-corrected estimate $\hat{\beta}_{\mathrm{BC}}$ has enjoyed its tremendous popularity in the errors-in-variables regression literature. $\hat{\beta}_{\mathrm{BC}}$ behaves very well when the sample size is moderately large, however, the MSE of $\hat{\beta}_{\mathrm{BC}}$ can be out of control when the sample size is small. For illustration purpose, we generate a sample of size $n = 30$ from the simple linear EV regression model with U, X, ε all from $N(0, 1)$, $U \sim N(0, 0.8)$, $\beta_0 = \beta_1 = 1$ are chosen to be true regression parameter values, then the bias-corrected estimate $\hat{\beta}_{\mathrm{BC}}$ of β_1 is calculated. The estimation procedure is repeated 500 times and the MSE is recorded which is shown in the following table. We also found that the MSE of $\hat{\beta}_{\mathrm{BC}}$ is affected heavily by the magnitude of σ_u^2. The general trend is that the larger the σ_u^2 values, the large the MSE values.

Table 1.1: MSE of $\hat{\beta}_{\mathrm{BC}}$

n	**MSE**
30	20.898
50	1.1728
100	0.0837

1.3 Motivation

In real applications, it appears that the normality assumptions on all random components in model (1.1) are too restrictive. Although estimating the regression coefficients is possible even if no distributional assumptions imposed on the model, as evidenced by the bias-corrected estimation procedure, the poor performance of this estimate would still make seeking for new estimates under fewer distributional assumptions a worthwhile errand.

By only assuming that $U \sim N(0, \sigma_u^2)$ and σ_u^2 is known, Song, Shi and Zhang (2016)[9] proposed an improved estimation procedure based on Tweedie's formula. To be specific, under the normality assumption on U, Tweedie's formula asserts that

$$E(X|Z) = Z + \sigma_u^2 \frac{g'(Z)}{g(Z)},$$

where $g(\cdot)$ denotes the density function of Z. In addition to this conditional expectation formula, we also have

$$\mathrm{Var}(X|Z = z) = \sigma_u^2 + \sigma_u^4 \left(\frac{g''(z)}{g(z)} - \frac{g'^2(z)}{g^2(x)} \right).$$

The extension to multivariate X is also straightforward. Efron (2011)[10] acclaimed the Tweedie's formula as an "extraordinary Bayesian estimation formula", and he employed this formula to deal with the selection bias and also applied it to genomics data analysis. The original proof of Tweedies's formula is based upon the property of the exponential family and a Bayesian argument. Using a deconvolution relationship, Song, Shi and Zhang (2016)[9] provided a much simpler proof of Tweedie's formula.

Denote $W = E(X|Z)$, we can rewrite model (1.1) as

$$Y = \alpha + \beta W + \varepsilon + \beta(X - W) = \alpha + \beta W + e. \tag{1.4}$$

It is easy to see that $e = \varepsilon + \beta(X - W)$ is uncorrelated with W by the independence assumption on ε, X and U. Now (1.4) is indeed a classical regression model. Therefore, if the density function g of Z, or the density function of X, is known, consistent estimates of the regression coefficients can be readily obtained by the least squares procedure. To be specific, suppose that $(Y_i, Z_i), i = 1, 2, \ldots, n$, constitutes a sample from model (1.1). Define $W_i = E(X_i|Z_i) = Z_i + \sigma_u^2 \frac{g'(Z_i)}{g(Z_i)}$. Then the least square estimates of β and α based on model (1.4) has the well known forms

$$\hat{\beta} = \frac{\sum_{j=1}^n (W_i - \bar{W})(Y_i - \bar{Y})}{\sum_{j=1}^n (W_i - \bar{W})^2}, \quad \hat{\alpha} = \bar{Y} - \hat{\beta}\bar{W}, \tag{1.5}$$

where $\bar{Y}$, $\bar{W}$ are the sample means of Y_i's and W_i's. The technique is indeed a special example of the regression calibration approach proposed in Carroll and Stefanski (1990)[11]. If the density function of Z is unknown, usually it is the case, then the estimates in (1.5) can be modified by replacing the true density g with its kernel density estimate.

Based on the tail behavior of the characteristic function of U, Fan and Troung (1993)[12] made a classification on the distributions of the measurement errors.

- **Super-smooth:** The distribution of a random variable u is super-smooth of order r, if its characteristic function $\phi_u(t)$ satisfies

$$d_0|t|^{r_0}\exp(-|t|^r/\gamma) \leq |\phi_u(t)| \leq d_1|t|^{r_1}\exp(-|t|^r/\gamma)$$

as $t \to \infty$, for some positive constants d_0, d_1, r, γ and constants r_0, r_1.

- **Ordinary Smooth:** The distribution of a random variable u is ordinary smooth of order r, if its characteristic function $\phi_u(t)$ satisfies

$$d_0|t|^{-r} \leq |\phi_u(t)| \leq d_1|t|^{-r}$$

as $t \to \infty$, for some positive constants d_0, d_1, r.

Examples of super-smooth distributions include $N(0,1)$ $(r = 2)$, Cauchy$(0,1)$ $(r = 1)$; and examples of ordinary smooth include Gamma with density $\alpha^p x^{p-1}\exp(-\alpha x)/\Gamma(p)$ $(r = p)$, and Laplace distribution with density $2^{-1}\exp(-|x|)$ $(r = 2)$.

Tweedie's formula has assisted us to construct an improved estimation procedure when U follows a normal distribution, which is a representative of super-smooth distributions. However, in real applications, the measurement error U may possesses heavy tailed distributions. Thus it might be interesting to investigate if a Tweedie-type formula exists for $E(X|Z)$ when U follows a heavy tailed distribution. If so, then similar to the estimation procedure developed in Song, Shi and Zhang (2016)[9], we can construct an improved estimation procedure accordingly for the regression parameters when the measurement error has a heavy tailed structure.

In the next chapter, we shall explore this possibility when U has a Laplace distribution, which is a typical example of the ordinary smooth distributions.

Chapter 2

Linear Regression with Laplace Measurement Error

In this chapter, we consider the simple linear regression model with Laplace measurement error. First, some basic facts on Laplace distribution are summarized, then a Tweedie-type formula is established based on the relationship between the density functions of X and Z when the measurement error U possesses a Laplace distribution. A new estimate for the regression coefficient thus can be constructed based the formula.

2.1 Laplace Distribution

The Laplace distribution, named after Pierre Simon Laplace, arises naturally as the distribution of the difference of two independent, identically distributed exponential variables. For this reason, it is also called the double exponential distribution.

The probability density function of a Laplace distribution with mean μ and variance σ^2 takes the form of

$$f(x;\mu,\sigma^2) = \frac{1}{\sqrt{2}\sigma} e^{-\frac{\sqrt{2}|x-\mu|}{\sigma}}.$$

For convenience, we denote Laplace(μ, σ^2) the Laplace distribution with mean μ and variance

σ^2. The characteristic function of Laplace(μ, σ^2) is given by

$$\psi(t) = \frac{\exp(\mathbf{i}\mu t)}{1 + \sigma^2 t^2/2}, \quad \mathbf{i}^2 = -1.$$

In the following discussion, we denote g the density function of Z. Since σ_u^2 is assumed to be known, so without of generality, we simply assume that $\sigma_u = \sqrt{2}$.

2.2 $E(X|Z)$ When U Follows Laplace$(0, \sqrt{2})$

In order to construct a similar estimate as in the normal measurement error case, we have to derive an expression for $E(X|Z)$ when U follows Laplace$(0, \sigma^2)$. The result is summarized in the following lemma.

Lemma 1. Suppose the density function g is twice differentiable, and for $H(z) = g(z)$, $zg(z)$, $zg'(z)$, $H(z)\exp(-|z|) \to 0$ as $z \to \pm\infty$. Then

$$E(X|Z = z) = z + \frac{\exp(z)\int_z^\infty g(x)\exp(-x)dx - \exp(-z)\int_{-\infty}^z g(x)\exp(x)dx}{g(z)}. \tag{2.1}$$

As pointed out in Efron (2011), the Tweedie's formula for normal measurement error can be applied more generally to multivariate exponential families. Since the Laplace measurement error is clearly not a member in the exponential family, so the formula proved in Lemma 1 is more interesting.

Proof. Denote f_x, $f_{x,z}$ and $f_{z|x}$ the density function of X, (X, Z) and the conditional density function of Z given X. We have

$$\begin{aligned} E(X|Z = z) &= \int xf(x|z)dx = \int \frac{xf_{x,z}(x,z)}{g(z)}dx \\ &= \frac{\int xf_{z|x}(z|x)f_x(x)dx}{g(z)} = \frac{\int x \cdot \frac{1}{2}e^{-|x-z|}\Big(g(x) - g''(x)\Big)dx}{g(z)}. \end{aligned} \tag{2.2}$$

The numerator can be further written as

$$\begin{aligned} & \int x \cdot \frac{1}{2} e^{-|x-z|}\Big(g(x)-g''(x)\Big)dx \\ = \ & \frac{1}{2}\int_z^\infty xe^{z-x}\Big(g(x)-g''(x)\Big)dx + \frac{1}{2}\int_{-\infty}^z xe^{x-z}\Big(g(x)-g''(x)\Big)dx \\ = \ & \frac{1}{2}e^z\int_z^\infty xe^{-x}\Big(g(x)-g''(x)\Big)dx + \frac{1}{2}e^{-z}\int_{-\infty}^z xe^{x}\Big(g(x)-g''(x)\Big)dx. \end{aligned}$$

One the one hand, we have

$$\begin{aligned} & \int_z^\infty xe^{-x}g''(x)dx = \int_z^\infty xe^{-x}dg'(x) = xe^{-x}g'(x)|_z^\infty - \int_z^\infty g'(x)(e^{-x}-xe^{-x})dx \\ = \ & -ze^{-z}g'(z) - \int_z^\infty g'(x)e^{-x}dx + \int_z^\infty g'(x)(xe^{-x})dx \\ = \ & -ze^{-z}g'(z) - \Big[g(x)e^{-x}|_z^\infty + \int_z^\infty g(x)e^{-x}dx\Big] + \Big[g(x)xe^{-x}|_z^\infty - \int_z^\infty g(x)(e^{-x}-xe^{-x})dx\Big] \\ = \ & -ze^{-z}g'(z) - \Big[-g(z)e^{-z} + \int_z^\infty g(x)(e^{-x})dx\Big] \\ & +\Big[-g(z)ze^{-z} - \int_z^\infty g(x)e^{-x}dx + \int_z^\infty g(x)xe^{-x}dx\Big] \\ = \ & -ze^{-z}g'(z) + g(z)e^{-z} - 2\int_z^\infty g(x)e^{-x}dx - g(z)ze^{-z} + \int_z^\infty g(x)xe^{-x}dx, \end{aligned}$$

on the other hand, we have

$$\begin{aligned} & \int_{-\infty}^z xe^{-x}g''(x)dx = g'(x)\cdot xe^x|_{-\infty}^z - \int_{-\infty}^z g'(x)[e^x+xe^x]dx \\ = \ & g'(z)\cdot ze^z - \int_{-\infty}^z g'(x)e^xdx - \int_{-\infty}^z g'(x)xe^xdx \\ = \ & g'(z)\cdot ze^z - \Big[g(x)e^x|_{-\infty}^z - \int_{-\infty}^z g(x)e^xdx\Big] - \Big[g(x)xe^x|_{-\infty}^z - \int_{-\infty}^z g(x)(e^x+xe^x)dx\Big] \\ = \ & g'(z)\cdot ze^z - g(z)\cdot e^z + \int_{-\infty}^z g(x)e^xdx - g(z)ze^z + \int_{-\infty}^z g(x)e^xdx + \int_{-\infty}^z g(x)xe^xdx \\ = \ & g'(z)ze^z - g(z)e^z + 2\int_{-\infty}^z g(x)e^xdx - g(z)ze^z + \int_{-\infty}^z g(x)xe^xdx. \end{aligned}$$

Therefore, the numerator in (2.2) can be written as

$$
\begin{aligned}
& \frac{1}{2}e^{z}\Big[\int_{z}^{\infty} xe^{-x}g(x) + ze^{-z}g'(z) - g(z)e^{-z} + 2\int_{z}^{\infty} g(x)(e^{-x})dx + g(z)ze^{-z} - \\
& \int_{z}^{\infty} g(x)xe^{-x}dx\Big] + \frac{1}{2}e^{-z}\Big[\int_{-\infty}^{z} xe^{x}g(x)dx - g'(z)ze^{z} + g(z)e^{z} \\
& -2\int_{-\infty}^{z} g(x)e^{x}dx + g(z)ze^{z} - \int_{-\infty}^{z} g(x)xe^{x}dx\Big] \\
= \; & e^{z}\int_{z}^{\infty} g(x)e^{-x}dx + \frac{1}{2}g(z)\cdot z - e^{-z}\int_{-\infty}^{z} g(x)e^{x}dx + \frac{1}{2}g(z)\cdot z \\
= \; & g(z)\cdot z + e^{z}\int_{z}^{\infty} g(x)e^{-x}dx - e^{-z}\int_{-\infty}^{z} g(x)e^{x}dx.
\end{aligned}
$$

Plugging the above result into (2.2) leads to the desired result.

2.3 Estimation When g Is Known

If the density function g is known, or equivalently, if the density function of X is known, then similar to the procedure developed in Shi, Zhang and Song (2016), we can estimate α, β in model (1.1) with the same formulae as in (1.5)

$$
\hat{\beta} = \frac{\sum_{j=1}^{n}(W_i - \bar{W})(Y_i - \bar{Y})}{\sum_{j=1}^{n}(W_i - \bar{W})^2}, \quad \hat{\alpha} = \bar{Y} - \hat{\beta}\bar{W}, \tag{2.3}
$$

where $W_i = E(X|Z = Z_i)$, and

$$
E(X|Z = z) = z + \frac{e^{z}\int_{z}^{\infty} g(x)e^{-x}dx - e^{-z}\int_{-\infty}^{z} g(x)e^{x}dx}{g(z)}.
$$

Example 1. Assume that $X \sim N(0, \tau^2)$, U follows Laplace distribution with mean 0 and $\sigma_u = \sqrt{2}$. Let $\bar{\Phi}_\tau(x) = 1 - \Phi(\tau - x/\tau)$ and $\Phi_\tau(x) = \Phi(-x/\tau - \tau)$. Then

$$
g(z) = \frac{1}{2}e^{\tau^2/2}\Big[e^{-z}\bar{\Phi}_\tau(z) + e^{z}\Phi_\tau(z)\Big],
$$

and

$$E(X|Z=z) = z + \frac{e^{z}\int_{z}^{\infty}\left\{e^{-2x}\bar{\Phi}_{\tau}(x)+\Phi_{\tau}(x)\right\}dx - e^{-z}\int_{-\infty}^{z}\left\{\bar{\Phi}_{\tau}(x)+e^{2x}\Phi_{\tau}(x)\right\}dx}{e^{-z}\bar{\Phi}_{\tau}(x)+e^{z}\Phi_{\tau}(x)}.$$

Example 2. Assume that X has a uniform distribution over $[-a, a]$ with $a > 0$, U follows Laplace distribution with mean 0 and $\sigma_u = \sqrt{2}$. Then

$$g(z) = \frac{e^{-(z-a)} - e^{-(z+a)}}{4a} I_{(a,\infty)}(z) + \frac{2 - e^{-(z+a)} - e^{z-a}}{4a} I_{[-a,a]}(z) + \frac{e^{z+a} - e^{z-a}}{4a} I_{(-\infty,-a)}(z),$$

and

$$E(X|Z=z) = z + \frac{A(z)}{g(z)},$$

where, for $z < -a$,

$$A(z) = \frac{(1-a-z)e^{a+z} - (1+a-z)e^{-a+z}}{4a};$$

for $-a \leq z \leq a$,

$$A(z) = \frac{(1+a+z)e^{-a-z} - (1+a-z)e^{-a+z}}{4a};$$

and for $z > a$,

$$A(z) = \frac{(1+a+z)e^{-a-z} - (1-a+z)e^{a-z}}{4a}.$$

2.4 Estimation When g Is Unknown

If the density function g is unknown, then $\hat{\alpha}$, $\hat{\beta}$ defined in (2.3) are not legitimate estimates of α and β. In this case, some nonparametric estimate of the density function g can be constructed based on the sample from Z, then plugging this nonparametric estimate into the expression of W_i defined in the previous section will provide estimates for α, β.

One of the most popular nonparametric density estimate is the kernel density estimate.

Let K be a symmetric density function about 0, h_n be a sequence of positive numbers satisfying $h_n \to 0$ and $nh_n \to \infty$ as $n \to \infty$. The kernel density estimate of g based on the sample $Z_1, Z_2, \ldots, Z_n$ is defined by

$$\hat{g}_n(z) = \frac{1}{nh_n} \sum_{i=1}^{n} K\left(\frac{Z_i - z}{h_n}\right).$$

Under some regularity conditions, the kernel density estimate $\hat{g}_n(z)$ is consistent and asymptotically normal.

Thus, plugging the kernel estimate $\hat{g}_n(z)$ into the formula (2.1) in Lemma 1, we get an estimate for $E(X|Z)$, that is

$$\hat{E}(X|Z = z) = z + \frac{\exp(z) \int_z^\infty \hat{g}_n(x) \exp(-x) dx - \exp(-z) \int_{-\infty}^z \hat{g}_n(x) \exp(x) dx}{\hat{g}_n(z)}. \qquad (2.4)$$

The finite sample performance of the kernel density estimate is quite sensitive to the choice of the bandwidths, however, it is not sensitive to the selection of the kernel functions. Therefore, the kernel function is mainly chosen for the sake of convenience. The commonly used kernel functions include standard normal, uniform and Epanechnikov kernel. In the following, we shall derive the formulae of $W = E(X|Z)$ when the kernel function is selected to be standard normal and Epanechnikov kernel. For the sake of brevity, the subscript n will be suppressed from h_n in the sequel.

2.4.1 When K Is Standard Normal

If K is chosen to be standard normal density function, that is $K(x) = (2\pi)^{-1/2}\exp(-x^2/2)$, then we have

$$W = Z + he^{\frac{h^2}{2}} \frac{\sum_{i=1}^n \left[e^{Z-Z_i}\Big(1 - \Phi\Big((Z-Z_i)/h + h\Big)\Big) - e^{-(Z-Z_i)}\Phi\Big((Z-Z_i)/h - h\Big)\right]}{\sum_{i=1}^n \phi\Big((Z-Z_i)/h\Big)}, \tag{2.5}$$

where ϕ and Φ denote the density function and CDF of standard normal distribution, respectively.

2.4.2 When K Is Epanechnikov kernel

The Epanechnikov kernel function is defined as

$$K(x) = \frac{3}{4}(1 - x^2)I(|x| \leq 1).$$

It is notes that Epanechnikov kernel is the optimal function in the sense that it minimizes the asymptotic MSE of the kernel density estimate among all kernel functions with finite second moment.

It is shown that when K is taken to be the Epanechnikov kernel, we obtain

$$W = Z + \frac{A(Z)}{B(Z)}, \tag{2.6}$$

where

$$\begin{aligned}
A(Z) \;=\; & \frac{3h}{4n}\sum_{i=1}^{n} e^{Z-Z_i}\Bigg[\mathbb{1}[Z < Z_i - h]\Bigg[\Bigg(\frac{2h-2}{h^3}\Bigg)e^{h} + \Bigg(\frac{2h+2}{h^3}\Bigg)e^{-h}\Bigg] \\
& + \mathbb{1}[|Z - Z_i| \leq h]\Bigg[\Bigg(\frac{1}{h} - \frac{1}{h^3}\Big((Z - Z_i + 1)^2 + 1\Big)\Bigg)e^{Z_i - Z} - \Bigg(\frac{2h+2}{h^3}\Bigg)e^{-h}\Bigg]\Bigg] \\
& - \frac{3h}{4n}\sum_{i=1}^{n} e^{-Z+Z_i}\Bigg[\mathbb{1}[Z < Z_i - h]\Bigg(\frac{2h-2}{h^3}e^{h} + \frac{2h+2}{h^3}e^{-h}\Bigg) \\
& + \mathbb{1}[|Z - Z_i| \leq h]\Bigg(\frac{2h-2}{h^3}e^{h} + \Bigg(\frac{(Z - Z_i - 1)^2 + 1}{h^3} - \frac{1}{h}\Bigg)e^{Z-Z_i}\Bigg)\Bigg],
\end{aligned}$$

and

$$B(Z) \;=\; \frac{3}{4nh}\sum_{i=1}^{n}\Bigg(1 - \Bigg(\frac{Z - Z_i}{h}\Bigg)^2\Bigg)[Z_i - h \leq Z \leq Z_i + h]$$

2.5 Bandwidth Selection

As we pointed out in the previous section, the finite sample performance of the kernel estimate is very sensitive to the choice of bandwidth. There are two common methods in practice to recommend bandwidth values. The first is to try different choices of bandwidths, and suggest a reasonable range on which the MSEs appear to be small and stable. This method indeed can help us to see how sensitive the estimation procedure is on the choice of bandwidths; the second alternative is to use some data driven methods to select the bandwidth, for example, the cross validation procedure.

Due to its objective nature, the cross validation procedures are widely used in nonparametric smoothing. In general, cross-validation is a model validation technique for assessing how the results of a statistical analysis will generalize to an independent data set. It is mainly used in settings where the goal is prediction, and one wants to estimate how accurately a predictive model will perform in practice.

In the simulation studies we conducted in Chapter 3, we will use the so called leave-one-out cross validation procedure. In this procedure, we use 1 observation as the validation set and the remaining observations as the training set. To be specific, suppose $(Z_i, Y_i), i = 1, 2, \ldots, n$ is a sample of size n from (Z, Y) in model (1.1). For each $i = 1, 2, \ldots, n$, we use the observations $(Z_j, Y_j), j = 1, 2, \ldots, n, j \neq i$ to estimate g, then W, α, β by the procedures proposed in the previous chapter, denote the resulting estimates as $W_{(-i)}$, $\hat{\alpha}_{(-i)}$, and $\hat{\beta}_{(-i)}$. For each h, define the criterion function

$$CV(h) = \sum_{i=1}^{n} \left[Y_i - \hat{\alpha}_{(-i)} - \hat{\beta}_{(-i)} W_{(-i)} \right]^2 .$$

Then the leave-one-out procedure will use the minimizer of the $CV(h)$ to be the bandwidth used in the final estimation.

Chapter 3

Simulation Studies

To evaluate the finite sample performance of the proposed methodology, several simulation studies will be conducted in this section. Comparison studies will be also made to show the superiority of the proposed estimation procedure over some other existing ones in the literature.

In measurement error literature, the distribution of X is rarely assumed to be known. However, for comparison purpose, we shall conduct some simulation studies under this strict assumptions, in particular, $X \sim N(0, \sigma_x^2)$ will be considered. See Example 1 in Section 2.3 for the explicit expression of $E(X|Z)$. For convenience, we shall call such estimate the oracle estimate.

For comparison purpose, the naive estimate will be also calculated along with other estimation procedures. The naive estimates of α and β can be obtained by simply regressing Y directly on the surrogate Z.

For each scenario, the simulation will be replicated 500 times, the MSEs based on these 500 estimates will be used as the criterion to evaluate the relative performance of the chosen estimation procedures.

3.1 The Sensitivity of The Bandwidth

In this simulation study, we generate the data from model (1.1) with $\alpha = \beta = 1$, $X \sim N(0,1)$, $U \sim$Laplace$(0, \sqrt{2})$. The sample size is taken to $n = 100, 200, 300$. To generate the random sample from Laplace distribution, we used the function `rdoublex` from the R package `smoothmest`. To see how sensitive the proposed estimation procedure to the choice of bandwidth, we chose $h = an^{-1/5}$ with a values range from 0.1 to 4 and the kernel function to be the standard normal density. The simulation results are summarized in Table 3.1.

From Table 3.1, we can see that the finite MSEs for $\hat{\beta}$ are bigger when a values are too big or too small. This observation well aligns with the fact that smaller bandwidths increase the variability of the kernel estimate and bigger bandwidths increase the bias of the kernel estimate. However, the MSE values do not vary too much, which indicates the estimation of the regression parameters in the linear errors-in-variables model is not affected too much by the selection of the bandwidth. This is also the case for the estimate $\hat{\alpha}$.

We also conduct the simulation using the Epanechnikov kernel, see Table 3.2. Surprisingly, the MSEs of both $\hat{\alpha}$ and $\hat{\beta}$ varies more than the Gaussian kernel. The MSE values for $\hat{\beta}$ is much less around $a = 1.5$ than the MSEs at two ends of chosen a values, while for $\hat{\alpha}$, the MSE values seem to increase when a gets bigger. This phenomenon worths a further investigation in the future.

3.2 Leave-1-Out Cross Validation

In nonparametric smoothing, sometimes it is more desirable to have a data-driven bandwidth selection procedure to help us to determine the bandwidth used in the estimation procedure. In Section 2.5, we introduced a data-driven bandwidth selection procedure, the leave-one-out cross validation method, which is a special case of the leave-p-out cross validation methods. In stead of using 1 observation as the validation set, the leave-p-out procedure uses p observations as the validation set and the rest observations as the training data set.

The simulation setup is the same as in the previous section except for $X \sim N(0, 4)$. To search the optimal bandwidth by minimizing the criterion function $CV(h)$ defined in Section 2.5, we consider the $CV(h)$ values for a grid of h values in $[1.5, 6]$ by a step of 0.05.

Figure 3.11 presents the $CV(h)$ graphs using K as the kernel function, and Figure 3.12 is the graph of $CV(h)$ when K is chosen to be Epanechnikov kernel, for $n = 100, 200, 300$. For illustration purpose, we also create some histograms of $\hat{\beta}$ using the cross validation bandwidths. The simulation results based on cross validation bandwidth are not very encouraging when n is small. However, the estimates are clearly improved when n gets larger.

We also create a series histograms based on 500 estimates of $\hat{\beta}$, for the purpose of illustration. See Figure 3.1 to Figure 3.10.

3.3 Comparison Studies

In this section, a simulation study is conducted to compare the proposed method with some other existing estimation procedures such as the bias-corrected estimate (BC), Stein type I (Stein1) and type II estimate (Stein II). The regression estimates using the true values of X (Oracle) and the estimates assuming the density function of X is known (True) are also calculated and serve as bench marks for comparison. Naive estimate is also calculated to see the effect of ignoring the measurement errors in the estimation procedure. The simulation setup is exactly the same as in Section 3.1 except some changes in the sample size, bandwidth, and variance of the measurement error, which is specified in the following.

The two Stein type estimates was proposed by Alice S. Whittemore (1989)[13] by using Stein estimates of the unobserved true covariates. The estimates are obtained by regressing the response variable Y on adjusted covariates based on the observed surrogates. For Stein type I, II estimates, the adjusted covariate are,

$$e_1(Z_i) = Z_i - \frac{(n-2)Z_i}{\sum_{j=1}^{n} Z_j^2}, \quad e_2(Z_i) = Z_i - \frac{(n-3)\sigma_u^2(Z_i - \bar{Z})}{\sum_{j=1}^{n}(Z_j - \bar{Z})^2},$$

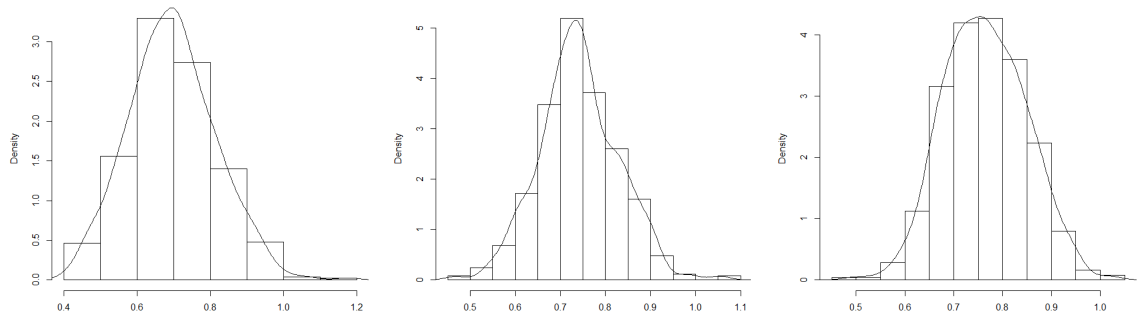
respectively.

Also, to see how the magnitude of the measurement error affects the estimation procedure, two values of σ_u^2 are tried in the simulation study. Table 3.4 shows the simulation results when the sample size are chosen to be $n = 30, 80, 100$ and $\sigma_u^2 = 1/4, 1/6$ and $h = an^{-1/5}$ with $a = 0.4, 0.8, 1.5$.

From Table 3.4, we can see that Stein type I estimate is inferior to all other estimate; clearly, the naive estimate is very biased, as expected according to the theory of linear regression model with measurement errors. The proposed estimate shows a certain degree of biasedness comparing to the bias-corrected estimate, but the bias can be partly attributed to the kernel estimate of g, which has a non-negligible bias. However, the MSE of the proposed estimate is less or at least comparable to that of the bias-corrected estimate.

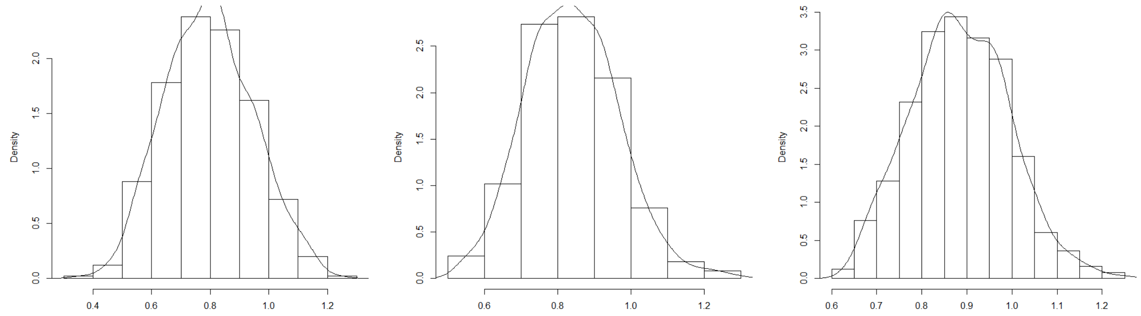
As expected, the estimate using the true values of X (Oracle) and the estimates assuming the density function of X is known (True) perform best, although the MSE of the True estimate is slightly bigger than that of the Oracle estimate.

Figure 3.1: Histogram of $\hat{\beta}$ when a=0.1 for Gaussian Kernel function N=500



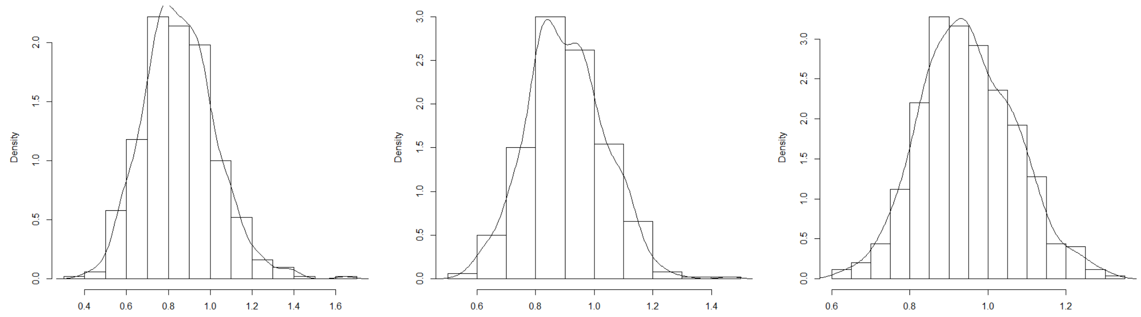
(a) a=0.1 n=100 (b) a=0.1 n=200 (c) a=0.1 n=300

Figure 3.2: Histogram of $\hat{\beta}$ when a=0.3 for Gaussian Kernel function N=500



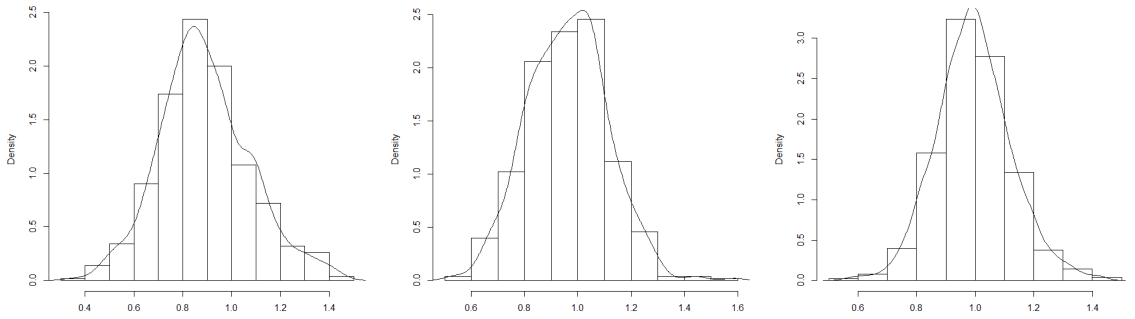
(a) a=0.3 n=100 (b) a=0.3 n=200 (c) a=0.3 n=300

Figure 3.3: Histogram of $\hat{\beta}$ when a=0.5 for Gaussian Kernel function N=500



(a) a=0.5 n=100 (b) a=0.5 n=200 (c) a=0.5 n=300

Figure 3.4: Histogram of $\hat{\beta}$ when a=0.8 for Gaussian Kernel function N=500

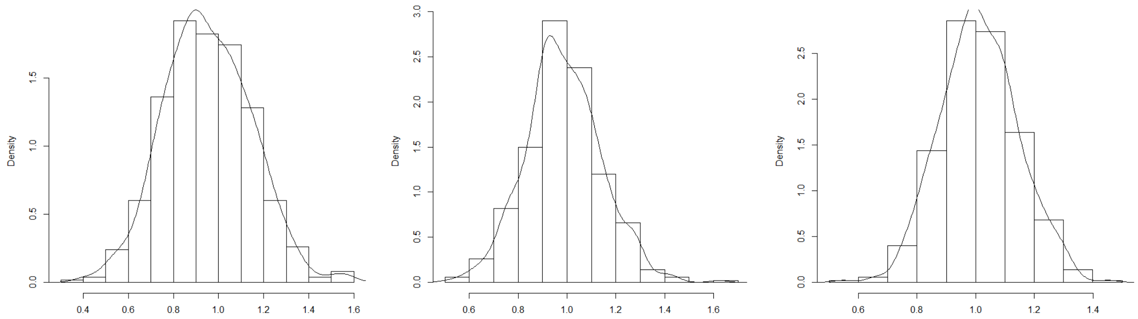


(a) a=0.8 n=100

(b) a=0.8 n=200

(c) a=0.8 n=300

Figure 3.5: Histogram of $\hat{\beta}$ when a=1 for Gaussian Kernel function N=500



(a) a=1 n=100

(b) a=1 n=200

(c) a=1 n=300

Figure 3.6: Histogram of $\hat{\beta}$ when a=0.1 for Epanechnikov Kernel function N=500

(a) a=0.1 n=100 (b) a=0.1 n=200 (c) a=0.1 n=300

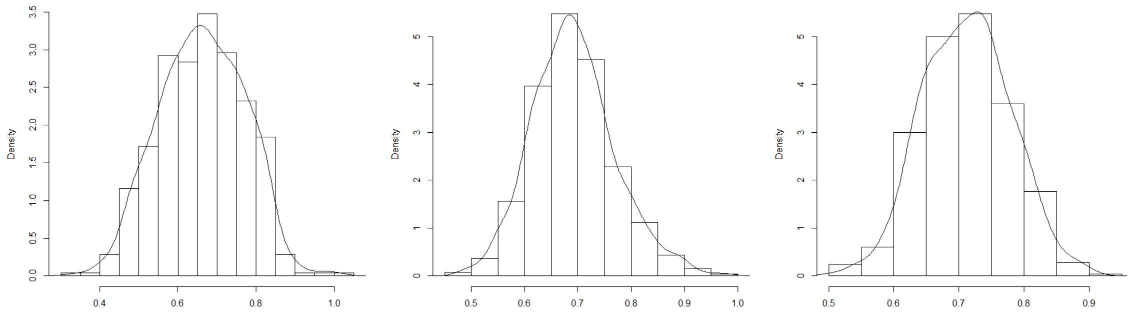


Figure 3.7: Histogram of $\hat{\beta}$ when a=0.3 for Epanechnikov Kernel function N=500

(a) a=0.3 n=100 (b) a=0.3 n=200 (c) a=0.3 n=300

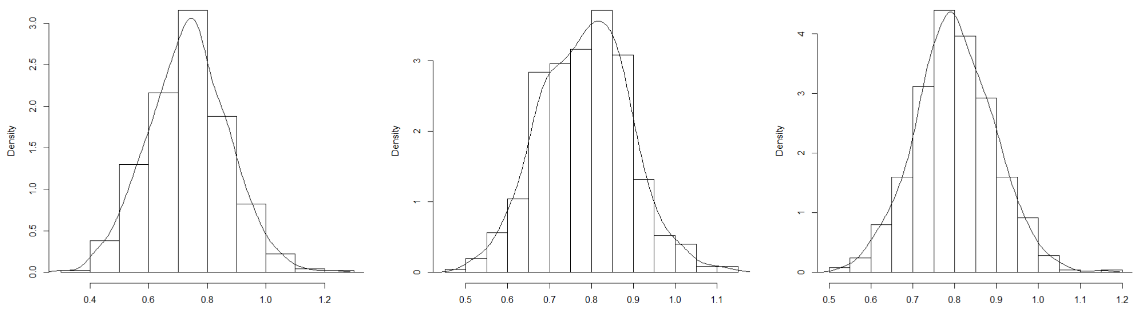


Figure 3.8: Histogram of $\hat{\beta}$ when a=0.5 for Epanechnikov Kernel function N=500

(a) a=0.5 n=100 (b) a=0.5 n=200 (c) a=0.5 n=300

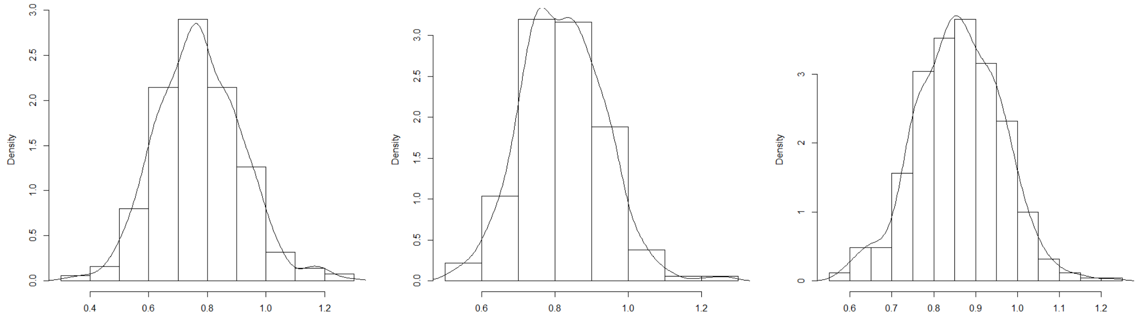
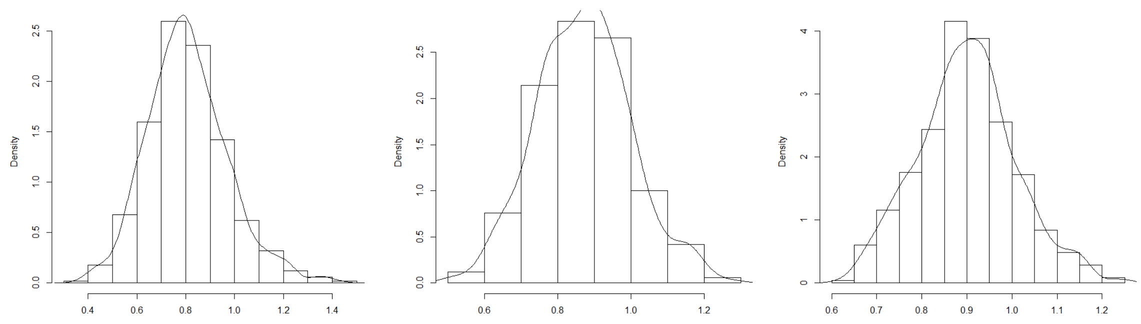


Figure 3.9: Histogram of $\hat{\beta}$ when a=0.8 for Epanechnikov Kernel function N=500

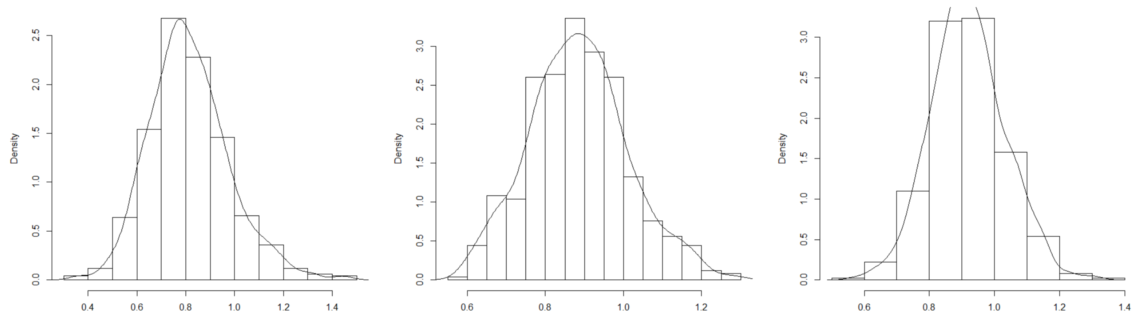


(a) a=0.8 n=100

(b) a=0.8 n=200

(c) a=0.8 n=300

Figure 3.10: Histogram of $\hat{\beta}$ when a=1 for Epanechnikov Kernel function N=500

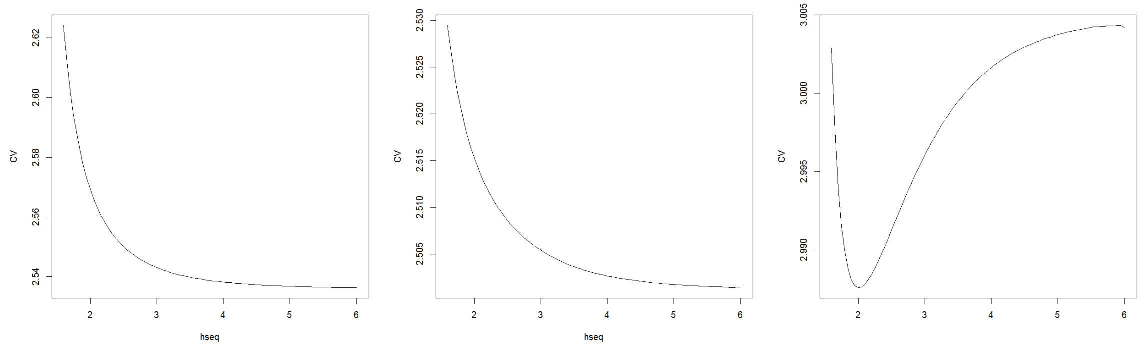


(a) a=1 n=100

(b) a=1 n=200

(c) a=1 n=300

Figure 3.11: $CV(h)$ plot for Gaussian Kernel function



(a) n=100

(b) n=200

(c) n=300

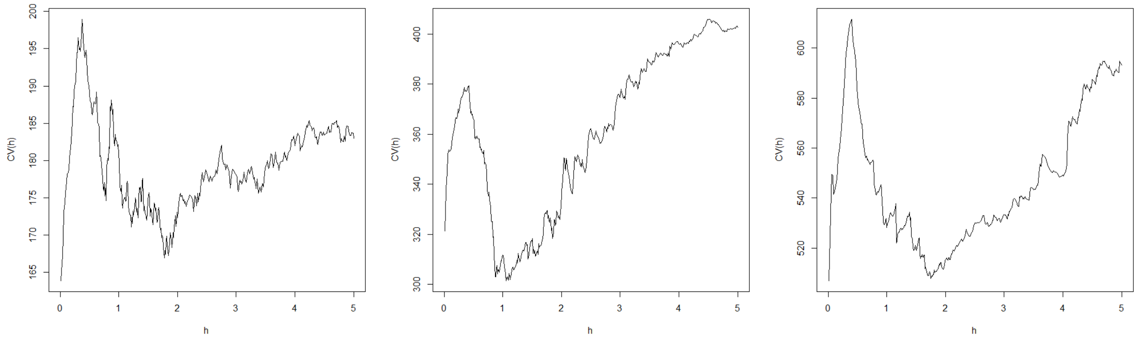
Table 3.1: Mean and MSE for Gaussian Kernel

n	a	Mean of $\hat{\alpha}$	Mean of $\hat{\beta}$	MSE of $\hat{\alpha}$	MSE of $\hat{\beta}$
100	0.1	1.0009	0.7019	0.0175	0.1026
	0.3	1.0080	0.7859	0.0203	0.0656
	0.5	1.0015	0.8535	0.0209	0.0500
	0.8	1.0086	0.9218	0.0264	0.0417
	1	0.9982	0.9382	0.0247	0.0462
	1.5	0.9975	0.9719	0.0235	0.0409
	2	0.9968	0.9488	0.0222	0.0313
	2.5	1.0004	0.9433	0.0198	0.0302
	3	1.0052	0.9020	0.0207	0.0310
	3.5	1.0041	0.8530	0.0191	0.0381
	4	0.9903	0.8355	0.0220	0.0416
200	0.1	1.0057	0.7357	0.0094	0.0784
	0.3	0.9949	0.8510	0.0119	0.0349
	0.5	0.9981	0.9102	0.0105	0.0264
	0.8	0.9996	0.9591	0.0116	0.0205
	1	1.0047	0.9954	0.0122	0.0230
	1.5	1.0026	1.0187	0.0115	0.0221
	2	0.9972	1.0100	0.0122	0.0176
	2.5	0.9987	0.9810	0.0117	0.0146
	3	1.0010	0.9568	0.0116	0.0141
	3.5	1.0043	0.9072	0.0112	0.0196
	4	1.0031	0.8662	0.0098	0.0268
300	0.1	0.9959	0.7706	0.0061	0.0593
	0.3	0.9974	0.8807	0.0078	0.0247
	0.5	1.0049	0.9406	0.0080	0.0186
	0.8	1.0061	0.9973	0.0074	0.0170
	1	0.9921	1.0130	0.0088	0.0167
	1.5	1.0006	1.0399	0.0081	0.0172
	2	1.0041	1.0391	0.0086	0.0159
	2.5	1.0002	1.0053	0.0080	0.0131
	3	1.0028	0.9732	0.0078	0.0098
	3.5	0.9977	0.9285	0.0062	0.0123
	4	0.9983	0.9000	0.0070	0.0166

Table 3.2: Mean and MSE for Epanechnikov Kernel

n	a	Mean of $\hat{\alpha}$	Mean of $\hat{\beta}$	MSE of $\hat{\alpha}$	MSE of $\hat{\beta}$
100	0.1	1.0073	0.6914	0.0313	0.1126
	0.3	1.0040	0.8090	0.0323	0.0638
	0.5	1.0521	0.8813	0.0338	0.0528
	0.8	1.0918	0.9342	0.0369	0.0393
	1	1.1789	0.9450	0.0615	0.0447
	1.5	1.3862	0.8914	0.1840	0.0425
	2	1.6251	0.7676	0.4264	0.0763
	2.5	1.8409	0.6386	0.7654	0.1479
200	0.1	1.0030	0.7405	0.0179	0.0782
	0.3	1.0067	0.8826	0.0160	0.0307
	0.5	1.0219	0.9560	0.0163	0.0261
	0.8	1.0871	1.0036	0.0230	0.0233
	1	1.1526	0.9985	0.0391	0.0215
	1.5	1.3119	0.9620	0.1105	0.0198
	2	1.5224	0.8633	0.2897	0.0315
	2.5	1.7366	0.7528	0.5656	0.0710
300	0.1	0.9975	0.7956	0.0102	0.0503
	0.3	1.0136	0.9367	0.0111	0.0182
	0.5	1.0308	0.9874	0.0122	0.0174
	0.8	1.0705	1.0320	0.0157	0.0191
	1	1.1110	1.0175	0.0230	0.0178
	1.5	1.2762	0.9900	0.0885	0.0144
	2	1.4581	0.9132	0.2213	0.0181
	2.5	1.6675	0.8111	0.4602	0.0439

Figure 3.12: $CV(h)$ plot for Epanechnikov Kernel function



(a) n=100

(b) n=200

(c) n=300

Table 3.3: Estimates using Gaussian kernel with bandwidth selected by cross validation

n	bandwidth	$\hat{\alpha}$	$\hat{\beta}$
100	5.95	0.9530	0.8347
200	5.90	0.9743	0.9132
300	2.00	1.0275	0.9743

Table 3.4: Means and MSEs of the Estimates of β

σ_u	n	a		Oracle	Naive	BC	Tweedie	Stein1	Stein2	True
1/4	30	0.4	Mean	0.9958	0.7970	1.0270	0.8988	1.7928	1.0063	0.9914
			MSE	0.0373	0.0788	0.0775	0.0626	3352.7427	0.0716	0.0578
		0.8	Mean	1.0050	0.8082	1.0329	0.9290	-0.7013	1.0128	1.0072
			MSE	0.0363	0.0742	0.0800	0.0594	4845.7029	0.0729	0.0573
		1.5	Mean	0.9963	0.7978	1.0222	0.9035	7.9331	1.0022	0.9943
			MSE	0.0038	0.0791	0.0771	0.0600	56208.1381	0.0716	0.0584
	80	0.4	Mean	0.9981	0.8003	1.0072	0.9247	4.8584	1.0006	0.9989
			MSE	0.0119	0.0531	0.0237	0.0240	513.0648	0.0232	0.0203
		0.8	Mean	0.9991	0.8018	1.0109	0.9421	0.6866	1.0042	1.0004
			MSE	0.0130	0.0526	0.0237	0.0225	42477.1090	0.0231	0.0206
		1.5	Mean	0.9937	0.7983	1.0106	0.9246	0.9421	1.0038	0.9959
			MSE	0.0129	0.0547	0.0265	0.0256	4251.7449	0.0259	0.0214
1/6	30	0.4	Mean	1.0023	0.8646	1.0271	0.9444	-3.0804	1.0137	1.0065
			MSE	0.0358	0.0570	0.0670	0.0519	11321.3502	0.0579	0.0521
		0.8	Mean	1.0035	0.8673	1.0279	0.9596	-12.5888	1.0147	1.0100
			MSE	0.0393	0.0561	0.0621	0.0517	250376.1693	0.0590	0.0518
		1.5	Mean	0.9967	0.8583	1.0194	0.9379	8.4888	1.0061	1.0000
			MSE	0.0364	0.0574	0.0588	0.0494	52011.5187	0.0562	0.0504
	80	0.4	Mean	1.0053	0.8602	1.0077	0.9555	-2.2989	1.0033	1.0036
			MSE	0.0120	0.0316	0.0180	0.0179	40193.3335	0.0177	0.0163
		0.8	Mean	0.9998	0.8593	1.0096	0.9648	9.7799	1.0052	1.0017
			MSE	0.0137	0.0336	0.0213	0.0198	5177.6765	0.0210	0.0185
		1.5	Mean	0.9930	0.8589	1.0076	0.9510	16.4470	1.0032	1.0015
			MSE	0.0129	0.0336	0.0199	0.0195	124107.3356	0.0197	0.0184

Chapter 4

Conclusions

In this report, an improved estimation procedure for the regression parameter in simple linear regression models with the Laplace measurement error is proposed. The estimation procedure is made feasible by a Tweedie type equality established for $E(X|Z)$. Both cases where the density function of Z is known and unknown are discussed. When the density function of Z is unknown, a kernel estimator for the density function of Z is constructed which in turn is used in estimating $E(X|Z)$. We provided the formulae of $E(X|Z)$ when Gaussian kernel and Epanechnikov Kernel are used. Simulation study are conducted to evaluate the finite sample performance of the proposed procedures. Bandwidth selection is also discussed in implementing the proposed estimation procedures. In particular, a trail and practice method in bandwidth selection can help us decide a reasonable range of values where the MSEs of the estimation of β keep small and stable. As a data driven bandwidth selection procedure, the eave-one-out cross validation bandwidth selection method is also discussed. Simulation studies show that the proposed estimator performs better than or at least comparable to some existing estimating procedures.

The asymptotic properties of the proposed estimator has not been investigated in this report, and this will be our future research.

Bibliography

[1] RJ Carroll, D Ruppert, LA Stefanski, and CM Crainiceanu. *Measurement Error in Nonlinear Models: A Modern Perspective*, volume 2nd Ed. CRC Press, 2006.

[2] R.J. Adcock. Note on the method of least squares. *Analyst*, 4:184–184, 1877.

[3] R.J. Adcock. A problem in least squares. *Analyst*, 5:53–54, 1878.

[4] O Reiersol. Identifiability of a linear relation between variables which are subject to error. *Econometrica*, 18:375–389, 1950.

[5] P.A.P. Moran. Estimating structural and functional relationships. *Journal of Multivariate Analysis*, 1:232–255, 1971.

[6] W.A. Fuller. *Measurement Error Models*, volume 1st Ed. Wiley, New York, 1987.

[7] Chi-Lun Cheng and John W. Van Ness. Generalized m-estimators for errors-in-variables regression. *The Annals of Statistics*, 20:385–397, 1992.

[8] John P. Buonaccorsi. *Measurement Error: Models, Methods, and Applications*, volume 1st Ed. CRC Press, 2010.

[9] Weixing Song, Jianhong Shi, and Chunxiu Zhang. Linear errors-in-variables regression and tweedie's formula. 2016.

[10] Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106:1602–1614, 2011.

[11] Raymond J. Carroll and Leonard A. Stefanski. Approximate quasi-likelihood estimation in models with surrogate predictors. *Journal of the American Statistical Association*, 85:652–663, 1990.

[12] Fan.J. and Y.K. Troung. Nonparametric regression with errors in variables. *The Annals of Statistics*, 21:1990–1925, 1993.

[13] Alice S. Whittemore. Errors-in-variables regression using stein estimates. *The American Statistician*, 43:226–228, 1989.

Appendix A

Proofs of Main Results

This appendix includes the proof of the main formulae we used in Chapter 2.

A.1 Proofs of (2.5) for Gaussian Kernel

Note that

$$e^z \int_z^\infty \hat{g}(x)e^{-x}dx = \frac{e^z}{nh}\sum_{i=1}^n \int_z^\infty K\left(\frac{Z_i-x}{h}\right)e^{-x}dx.$$

With K being standard normal density, we have

$$\begin{aligned}
& \int_z^\infty K\left(\frac{Z_i-x}{h}\right)e^{-x}dx = \int_z^\infty \frac{1}{\sqrt{2\pi}}e^{-\frac{(Z_i-x)^2}{2h^2}}e^{-x}dx \\
= & \int_z^\infty \frac{1}{\sqrt{2\pi}}e^{-\frac{Z_i^2}{2h^2}+\frac{Z_i x}{h^2}-\frac{x^2}{2h^2}-x}dx = \frac{1}{\sqrt{2\pi}}e^{-\frac{Z_i^2}{2h^2}}\int_z^\infty e^{(\frac{Z_i}{h^2}-1)x-\frac{x^2}{2h^2}}dx \\
= & \frac{1}{\sqrt{2\pi}}e^{-\frac{Z_i^2}{2h^2}}\int_z^\infty e^{-\frac{x^2-2(Z_i-h^2)x}{2h^2}}dx = \frac{1}{\sqrt{2\pi}}e^{-\frac{Z_i^2}{2h^2}}\int_z^\infty e^{-\frac{(x-(Z_i-h^2))^2}{2h^2}}e^{\frac{(Z_i-h^2)^2}{2h^2}}dx \\
= & \frac{1}{\sqrt{2\pi}}e^{-\frac{Z_i^2}{2h^2}}e^{\frac{(Z_i-h^2)^2}{2h^2}}\int_z^\infty e^{-\frac{(x-(Z_i-h^2))^2}{2h^2}}dx = he^{-\frac{Z_i^2}{2h^2}}e^{\frac{(Z_i-h^2)^2}{2h^2}}\int_z^\infty \frac{1}{\sqrt{2\pi}h}e^{-\frac{(x-(Z_i-h^2))^2}{2h^2}}dx
\end{aligned}$$

By changing variable, $\frac{x-(Z_i-h^2)}{h}=v$, we can obtain

$$\begin{aligned} & he^{-\frac{Z_i^2}{2h^2}}e^{\frac{(Z_i-h^2)^2}{2h^2}}\int_{\frac{z-(Z_i-h^2)}{h}}^{\infty}\frac{1}{\sqrt{2\pi}h}e^{-\frac{v^2}{2}}dv = he^{-\frac{Z_i^2}{2h^2}}e^{\frac{(Z_i-h^2)^2}{2h^2}}\left[1-\Phi\left(\frac{z-(Z_i-h^2)}{h}\right)\right] \\ = \; & he^{\frac{h^4-2Z_ih^2}{2h^2}}\left[1-\Phi\left(\frac{z-(Z_i-h^2)}{h}\right)\right] = he^{\frac{h^2}{2}-Z_i}\left[1-\Phi\left(\frac{z-Z_i}{h}+h\right)\right]. \end{aligned}$$

Therefore,

$$\begin{aligned} e^z\int_z^{\infty}\hat{g(x)}e^{-x}dx &= \frac{1}{nh}e^z\sum_{i=1}^n e^{\frac{h^2}{2}-Z_i}\left[1-\Phi\left(\frac{z-Z_i}{h}+h\right)\right] \\ &= \frac{1}{n}\sum_{i=1}^n e^{z-Z_i+\frac{h^2}{2}}\left[1-\Phi\left(\frac{z-Z_i}{h}+h\right)\right]. \end{aligned}$$

On the other hand, we have

$$e^{-z}\int_{-\infty}^{z}\hat{g}(x)e^xdx = \frac{e^{-z}}{nh}\sum_{i=1}^n\int_z^{\infty}K\left(\frac{Z_i-x}{h}\right)e^xdx.$$

Similarly, we have

$$\int_{-\infty}^{z}K\left(\frac{Z_i-x}{h}\right)e^xdx = \frac{1}{\sqrt{2\pi}}e^{-\frac{Z_i^2}{2h^2}}\int_{-\infty}^{z}e^{(\frac{Z_i}{h^2}+1)x-\frac{x^2}{2h^2}}dx = he^{\frac{h^2}{2}+Z_i}\left[\Phi\left(\frac{z-Z_i}{h}-h\right)\right],$$

and

$$e^{-z}\int_{-\infty}^{z}\hat{g}(x)e^xdx = e^{-z}\left[\frac{1}{nh}\sum_{i=1}^n he^{\frac{h^2}{2}+Z_i}\Phi\left(\frac{z-Z_i}{h}-h\right)\right] = \frac{1}{n}\sum_{i=1}^n e^{-(z-Z_i)+\frac{h^2}{2}}\Phi\left(\frac{z-Z_i}{h}-h\right).$$

Plugging the above result into (2.4) completes the proof of (2.5).

A.2 Proof of (2.6) for Epanechnikov Kernel

For Epanechnikov kernel function

$$K(x) = \frac{3}{4}(1 - x^2)I[|x| \le 1],$$

we have

$$\begin{aligned}
&\int_z^{\infty} K\left(\frac{Z_i - x}{h}\right) e^{-x} dx = \frac{3}{4}\int_z^{\infty} \left(1 - \left(\frac{Z_i - x}{h}\right)^2\right) e^{-x} \mathbb{1}\left[\left|\frac{x - Z_i}{h}\right| \le 1\right] dx \\
= \;& \frac{3}{4}\mathbb{1}[z < Z_i - h] \int_{Z_i - h}^{Z_i + h} \left(1 - \left(\frac{Z_i - x}{h}\right)^2\right) e^{-x} dx + \frac{3}{4}\mathbb{1}[z > Z_i - h] \cdot 0 \\
& + \frac{3}{4}\mathbb{1}[Z_i - h \le z \le Z_i + h] \int_z^{Z_i + h} \left(1 - \left(\frac{Z_i - x}{h}\right)^2\right) e^{-x} dx.
\end{aligned}$$

By changing variable and integration by parts, we obtain

$$\begin{aligned}
&\int_{Z_i - h}^{Z_i + h} \left(1 - \left(\frac{Z_i - x}{h}\right)^2\right) e^{-x} dx = \int_{-1}^{1} [1 - u^2] e^{-Z_i - uh} du \cdot h \\
= \;& e^{-Z_i} \int_{-1}^{1} [1 - u^2] e^{-uh} du \cdot h = h e^{-Z_i} \left[\int_{-1}^{1} e^{-uh} du - \int_{-1}^{1} u^2 e^{-uh} du\right].
\end{aligned}$$

Note that

$$\int_{-1}^{1} e^{-uh} du = -\frac{1}{h} e^{-uh}\Big|_{-1}^{1} = \frac{1}{h}\left[e^h - e^{-h}\right],$$

and

$$\int_{-1}^{1} u^2 e^{-uh} du = \int_{-h}^{h} \left(\frac{v}{h}\right)^2 e^{-v} \frac{1}{h} dv = \frac{1}{h^3}\left[h^2 e^h - 2he^h + 2e^h - h^2 e^{-h} - 2he^{-h} - 2e^{-h}\right],$$

we can get

$$\int_{Z_i - h}^{Z_i + h} \left(1 - \left(\frac{Z_i - x}{h}\right)^2\right) e^{-x} dx = h e^{-Z_i} \left[\left(\frac{2h - 2}{h^3}\right) e^h + \left(\frac{2h + 2}{h^3}\right) e^{-h}\right].$$

Now, let's consider

$$\int_{z}^{Z_i+h} \left(1-\left(\frac{Z_i-x}{h}\right)^2\right) e^{-x}dx.$$

By changing variable again,

$$\begin{aligned}\int_{z}^{Z_i+h} \left(1-\left(\frac{Z_i-x}{h}\right)^2\right) e^{-x}dx &= \int_{\frac{z-Z_i}{h}}^{1} (1-u^2)e^{-Z_i-uh}du\cdot h \\ = \quad e^{-Z_i}\int_{\frac{z-Z_i}{h}}^{1}(1-u^2)e^{-uh}du\cdot h &= he^{-Z_i}\left[\int_{\frac{z-Z_i}{h}}^{1} e^{-uh}du - \int_{\frac{z-Z_i}{h}}^{1} u^2e^{-uh}du\right].\end{aligned}$$

Note that

$$\int_{\frac{z-Z_i}{h}}^{1} e^{-uh}du = -\frac{1}{h}e^{-uh}|_{\frac{z-Z_i}{h}}^{1} = \frac{1}{h}(e^{Z_i-z}-e^{-h})$$

and

$$\int_{\frac{z-Z_i}{h}}^{1} u^2e^{-uh}du = \frac{1}{h^3}\left((z-Z_i)^2e^{Z_i-z}+2(z-Z_i)e^{Z_i-z}+2e^{Z_i-z}-h^2e^{-h}-2he^{-h}-2e^{-h}\right),$$

we have

$$\begin{aligned}\int_{z}^{Z_i+h} \left(1-\left(\frac{Z_i-x}{h}\right)^2\right) e^{-x}dx &= he^{-Z_i}[\frac{1}{h}(e^{Z_i-z}-e^{-h}) \\ &-\frac{1}{h^3}((z-Z_i)^2e^{Z_i-z}+2(z-Z_i)e^{Z_i-z}+2e^{Z_i-z}-h^2e^{-h}-2he^{-h}-2e^{-h})] \\ = \quad he^{-Z_i}&\left[(\frac{1}{h}-\frac{1}{h^3}((z-Z_i+1)^2+1))e^{Z_i-z}-\left(\frac{2h+2}{h^3}\right)e^{-h}\right].\end{aligned}$$

In summary, we have

$$\int_{z}^{\infty} K\left(\frac{Z_i-x}{h}\right)e^{-x}dx = \frac{3}{4}\mathbb{1}[z<Z_i-h]\int_{Z_i-h}^{Z_i+h}\left(1-\left(\frac{Z_i-x}{h}\right)^2\right)e^{-x}dx$$

$$+\frac{3}{4}\mathbb{1}[Z_i-h\le z\le Z_i+h]\int_{z}^{Z_i+h}\left(1-\left(\frac{Z_i-x}{h}\right)^2\right)e^{-x}dx$$

$$= \frac{3}{4}he^{-Z_i}[\mathbb{1}[z<Z_i-h][\left(\frac{2h-2}{h^3}\right)e^h+\left(\frac{2h+2}{h^3}\right)e^{-h}]$$

$$+\mathbb{1}[Z_i-h\le z\le Z_i+h]\left[(\frac{1}{h}-\frac{1}{h^3}\left((z-Z_i+1)^2+1\right))e^{Z_i-z}-\left(\frac{2h+2}{h^3}\right)e^{-h}\right].$$

To calculate the second half in the numerator of (2.4), first we have

$$\int_{-\infty}^{z} K\left(\frac{Z_i-x}{h}\right)e^{x}dx = \frac{3}{4}\mathbb{1}[z<Z_i-h]\cdot 0+\frac{3}{4}\mathbb{1}[z<Z_i-h]\int_{Z_i-h}^{Z_i+h}\left(1-\left(\frac{Z_i-x}{h}\right)^2\right)e^{x}dx$$

$$+\frac{3}{4}\mathbb{1}[Z_i-h\le z\le Z_i+h]\int_{Z_i-h}^{z}\left(1-\left(\frac{Z_i-x}{h}\right)^2\right)e^{x}dx.$$

Changing variable again, we have

$$\int_{Z_i-h}^{Z_i+h}\left(1-\left(\frac{Z_i-x}{h}\right)^2\right)e^{x}dx = \int_{Z_i-h}^{Z_i+h}[1-u^2]e^{uh+Z_i}du\cdot h$$

$$= he^{Z_i}\int_{-1}^{1}[1-u^2]e^{uh}du = he^{Z_i}\int_{-1}^{1}e^{uh}du-\int_{-1}^{1}u^2e^{uh}du.$$

Note that

$$\int_{-1}^{1}e^{uh}du = \frac{1}{h}\left(e^h-e^{-h}\right),$$

and

$$\int_{-1}^{1}u^2e^{uh}du = \int_{-h}^{h}\frac{v}{h}^2e^v\frac{1}{h}dv = \frac{1}{h^3}((h^2-2h+2)e^h-(h^2+2h+2)e^{-h}),$$

we obtain

$$\int_{Z_i-h}^{Z_i+h}\left(1-\left(\frac{Z_i-x}{h}\right)^2\right)e^x dx = he^{Z_i}\left(\frac{2h-2}{h^3}e^h+\frac{2h+2}{h^3}e^{-h}\right).$$

For $\int_{Z_i-h}^{z}(1-(\frac{Z_i-x}{h})^2)e^x dx$, by changing variable again, we have

$$\int_{Z_i-h}^{z}\left(1-\left(\frac{Z_i-x}{h}\right)^2\right)e^x dx = he^{Z_i}\int_{\frac{z-Z_i}{h}}^{1}(1-u^2)e^{uh}du = he^{Z_i}[\int_{\frac{z-Z_i}{h}}^{1}e^{uh}du-\int_{-1}^{\frac{z-Z_i}{h}}(u^2)e^{uh}du].$$

Since

$$\int_{\frac{z-Z_i}{h}}^{1}e^{uh}du=\frac{1}{h}e^{uh}|_{\frac{z-Z_i}{h}}^{1}=\frac{1}{h}(e^h-e^{z-Z_i}),$$

and

$$\int_{\frac{z-Z_i}{h}}^{1}u^2e^{uh}du=\frac{1}{h^3}[(h^2-2h+2)e^h-((z-Z_i)^2-2(z-Z_i)+2)e^{z-Z_i}],$$

so, we obtain

$$\int_{Z_i-h}^{z}\left(1-\left(\frac{Z_i-x}{h}\right)^2\right)e^x dx = he^{Z_i}(\frac{2h-2}{h^3}e^h+\left(\frac{(z-Z_i-1)^2+1}{h^3}-\frac{1}{h}\right)e^{z-Z_i}).$$

Finally, we can get

$$\begin{aligned}
&\int_{-\infty}^{z}K\left(\frac{Z_i-x}{h}\right)e^x dx = \frac{3}{4}\mathbb{1}[z<Z_i-h]\int_{Z_i-h}^{Z_i+h}\left(1-\left(\frac{Z_i-x}{h}\right)^2\right)e^x dx\\
&\quad+\frac{3}{4}\mathbb{1}[Z_i-h\le z\le Z_i+h]\int_{Z_i-h}^{z}\left(1-\left(\frac{Z_i-x}{h}\right)^2\right)e^x dx\\
=\;&\frac{3}{4}he^{Z_i}[\mathbb{1}[z<Z_i-h]\left(\frac{2h-2}{h^3}e^h+\frac{2h+2}{h^3}e^{-h}\right)\\
&\quad+\mathbb{1}[Z_i-h\le z\le Z_i+h]\left(\frac{2h-2}{h^3}e^h+\left(\frac{(z-Z_i-1)^2+1}{h^3}-\frac{1}{h}\right)e^{z-Z_i}\right)].
\end{aligned}$$

Plugging all the above result into (2.4), we complete the proof of (2.6).

Appendix B

R Codes

In this appendix, we list all the R-programs we used in the simulation studies.

B.1 R Codes for Table 1.1

```
# Simulation for Table 1.1
# Biased Correction estimate
   b_hat = 0
   a_hat = 0
   MSE = 0

# 500 time for simulation
   N = 500
   for (k in 1:N)
   {
      n=30
      U=rdoublex(n,mu=0,lambda=1)
      X=rnorm(n, mean = 0, sd =1)
      E=rnorm(n, mean = 0, sd = 1)
      a=1
      b=1
      Y=a+b*X+E
      Z =X+U
      VarU = sqrt(2)
      Ybar = mean(Y)
      Zbar = mean(Z)
      b_hat[k] = sum((Y - Ybar)*(Z-Zbar)) / (sum((Z-Zbar)^2)- n*VarU)
      B = sum((Y - Ybar)*(Z-Zbar)) / (sum((Z-Zbar)^2)- n*VarU)
      a_hat[k]  = Ybar - B*Zbar
   }
   mean(a_hat)
   mean(b_hat)
   1/N*sum((a_hat-1)^2)
   1/N*sum((b_hat-1)^2)
```

B.2 R Codes for Table 3.1 , Figure 3.1 - 3.5

```
# Simulation for Table 3.1 , Figure 3.1 - 3.5
# Histogram of bhat when a=0.1,0.3 ... 4 for Gaussian Kernal function

  bhat = 0
  ahat = 0
  MSE = 0

# use outside loop N times for simulation
  N=500
  for (k in 1:N)
  {
     n=100
     U=rdoublex(n,mu=0,lambda=1)

   # Generate n random number from normal disrtribution
     X=rnorm(n, mean = 0, sd = 1)

   # Generate n error term from normal disrtribution
     E=rnorm(n, mean = 0, sd = 1)

   # We can get Y and Z_original a(alpha)=1, b(beta)=1
     a=1
     b=1
     Y=a+b*X+E
     Z_original=X+U

   # A  window width
     A = 1
     h=A*n^(-1/5)
     EXZ = 0
     for (i in 1:n)
     {
         Z=rep(Z_original[i],100)
         Zi=X+U
         q_one=(Z-Zi)/h+h
         q_two=(Z-Zi)/h-h
         pnorm1=pnorm(q_one, mean = 0, sd = 1)
         pnorm2=pnorm(q_two, mean = 0, sd = 1)

         numerator_one=(exp(Z-Zi)*(1-pnorm1)-exp(Zi-Z)*(pnorm2))
         denominator_one=(1/(sqrt(2*pi))*exp(-1/2*((Z-Zi)/h)^2))
         EXZ[i]  = Z_original[i] + h*exp(h^2/2)* sum(numerator_one)
         /sum(denominator_one)
     }
     reg = lm(Y~EXZ)
     bhat[k]=reg$coefficients[2]
     ahat[k]=reg$coefficients[1]
     MSE[k] = mean(reg$residuals^2)
  }

  mean(bhat)
  1/N*sum((bhat-1)^2)
  hist(bhat, freq=FALSE,main="Histogram of bhat when a=1
  for Gaussian Kernal function")
  lines(density(bhat))
```

B.3 R Codes for Table 3.2 , Figure 3.6 - 3.10

```
# Simulation for Table 3.2 , Figure 3.6 - 3.10
# Histogram of bhat when a=0.1..2.5 for Epanechnikov Kernal function
  b_hat = 0
  a_hat = 0
  MSE = 0
# N times for simulation
  N = 500

  for (k in 1:N)
  {
     n=300
     U=rdoublex(n,mu=0,lambda=1)
     X=rnorm(n, mean = 0, sd = 1)
     E=rnorm(n, mean = 0, sd = 1)
     a=1
     b=1
     Y=a+b*X+E
     Z_original=X+U

   # A bandwidth
     A = 2.5
     h=A*n^(-1/5)
     EXZe = gzhat= 0
     for (j in 1:n)
     {
        Z=rep(Z_original[j],100)
        Zi=X+U
        numerator_1stHalf = 0.75*sum((exp(Z-Zi)*(((Z<Zi-h)*(((2*h-2)/h^3)*
        exp(h)+((2*h+2)/h^3)*exp(-h)))+(Z>Zi-h)*(Z<Zi+h)*((1/h-(1/h^3)*
        ((Z-Zi+1)^2+1))*exp(Zi-Z)+((2*h+2)/h^3)*exp(-h)))))/n

        numerator_2ndHalf = 0.75*sum((exp(-Z+Zi)*(((Z>Zi+h)*(((2*h-2)/h^3)*
        exp(h)+((2*h+2)/h^3)*exp(-h)))+(Z>Zi-h)*(Z<Zi+h)*(((2*h-2)/h^3)*
        exp(h)+((1/h^3)*((Z-Zi-1)^2+1)-1/h)*exp(Z-Zi)))))/n

        denominator_two = (0.75)*(1/(n*h))*sum((Z>Zi-h)*(Z<Zi+h)*
        (1-((Z-Zi)/h)^2))
        gzhat[j]=denominator_two;
        EXZe[j]  = Z_original[j]*denominator_two + (numerator_1stHalf-
        numerator_2ndHalf)
     }

     Ynew=Y*gzhat;
     reg1 = lm(Ynew~gzhat+EXZe-1)
     b_hat[k]=reg1$coefficients[2]
     a_hat[k]=reg1$coefficients[1]
     MSE[k] = mean(reg1$residuals^2)
  }

  mean(a_hat)
  mean(b_hat)
  1/N*sum((a_hat-1)^2)
  1/N*sum((b_hat-1)^2)

# main="Histogram of b_hat when a=1 for Epanechikov Kernal function"
```

```
   hist(b_hat, freq=FALSE)
   lines(density(b_hat))
```

B.4 R Codes for Figure 3.11

```
# Simulation for Figure 3.11
# CV(h) and hi plot for  Gaussian Kernel function

 n=100
 U=rdoublex(n,mu=0,lambda=1)

# Generate n random number from normal disrtribution
 X=rnorm(n, mean = 0, sd = 1)

# Generate n error term from normal disrtribution
 E=rnorm(n, mean = 0, sd = 1)

# We can get Y and Z_original a(alpha)=1, b(beta)=1
 a=1
 b=1
 Y=a+b*X+E
 Z_original=X+U
 start= 0.02
 end = 2
 range = 0.02
# create hi range for h (bandwidth)
 hi = seq(start,end, by= range)

# Create Rh set to 0. Rh is function for h
 Rh =0
 bvalue = 0

# start loop by setting h equal to hi range
 for ( h in hi)
 {
    bhat = 0
    ahat = 0
    k=1
    EXZ = 0
    while(k <= n)
    {
       start_insideloop= 1
       end_insideloop = n
       range_insideloop = 1
       ni = seq(start_insideloop,end_insideloop, by= range_insideloop)
       ni = ni[-k]

       for (i in ni)
       {
          Z=rep(Z_original[i],n-1)
          Zi=X+U
          Zi=Zi[-k]
          q_one=(Z-Zi)/h+h
          q_two=(Z-Zi)/h-h
          pnorm1=pnorm(q_one, mean = 0, sd = 1)
          pnorm2=pnorm(q_two, mean = 0, sd = 1)
          numerator_one=(exp(Z-Zi)*(1-pnorm1)-exp(Zi-Z)*(pnorm2))
          denominator_one=(1/(sqrt(2*pi))*exp(-1/2*((Z-Zi)/h)^2))
          EXZ[i]  = Z_original[i] + h*exp(h^2/2)* sum(numerator_one)/
          sum(denominator_one)
       }

       reg = lm(Y[-k]~EXZ[-k])
       bhat[k]=reg$coefficients[2]
       ahat[k]=reg$coefficients[1]
       k=k+1
    }
  # transform range to index and store them in vector Rh with order
    index = (h+range-start)/range
    Rh[index] = sum((Y - ahat*-bhat*EXZ)^2)
    bvalue[index] = summary(reg)$coefficients[2]
 }
 plot(hi[!is.na(Rh)],Rh[!is.na(Rh)],type="l", ylab="CV(h)",xlab="hi")
```

B.5 R Codes for Figure 3.12

```
# Simulation for Figure 3.12
# CV(h) and hi plot for Epanechnikov Kernel function
# set the sample size n
  n=100
  U=rdoublex(n,mu=0,lambda=1)
  X=rnorm(n, mean = 0, sd = 1)
```

```
# Generate n error term from normal distribution
  E=rnorm(n, mean = 0, sd = 1)

# set true value for a and b to 1 and get Y and Z_original value
  a=1
  b=1
  Y=a+b*X+E
  Z_original=X+U

# Create a range hi for h from start to end point by range
  start= 0.01
  end = 5
  range = 0.01
# create hi range for h (bandwidth)
  hi = seq(start,end, by= range)

# Create Rh set to 0. Rh is function for h
  Rh =0

# start loop by setting h equal to hi range
  for ( h in hi)
  {
   # Epanechikov Kernal function
     b_hat = 0
     a_hat = 0
     MSE = 0

   # create EXZe for expected value in Epanechikov kernal
     EXZe = 0
     for (j in 1:n)
     {
      # create Z vector with equal value from Z_original
        Z=rep(Z_original[j],n)
      # add X to measurement error which is random number from laplace U
        Zi=X+U
        numerator_1stHalf = sum((exp(Z-Zi)*(((Z<Zi-h)*(((2*h-2)/h^3)*
        exp(h)+((2*h+2)/h^3)*exp(-h)))+(Z>Zi-h)*(Z<Zi+h)*((1/h-(1/h^3)*
        ((Z-Zi+1)^2+1))* exp(Zi-Z)+((2*h+2)/h^3)*exp(-h)))))

        numerator_2ndHalf = sum((exp(-Z+Zi)*(((Z>Zi+h)*(((2*h-2)/h^3)*
        exp(h)+((2*h+2)/h^3)*exp(-h)))+ (Z>Zi-h)*(Z<Zi+h)*
        (((2*h-2)/h^3)*exp(h)+((1/h^3)*((Z-Zi-1)^2+1)-1/h)*exp(Z-Zi))))
        denominator_two = (1/h)*sum((Z>Zi-h)*(Z<Zi+h)*(1-((Z-Zi)/h)^2))
      # calculate EXZe, expected value of X give Z in Epanechikov kernal
        EXZe[j]  = Z_original[j] + (numerator_1stHalf-numerator_2ndHalf)/
        denominator_two
     }

   # create regression model and get estimate b_hat and a_hat and MSE
   # store b_hat, a_hat and MSE in vector we created before
     reg1 = lm(Y~EXZe)
     b_hat=reg1$coefficients[2]
     a_hat=reg1$coefficients[1]
     MSE = mean(reg1$residuals^2)

   # transform range to index and store them in vector Rh with order
     index = h/range
   # calculate Rh function related to Y ma_hat, mb_hat, mEXZe, store
   # by index
     Rh[index] = sum((Y - a_hat-b_hat*EXZe)^2)
  }
# plot hi vs Rh graph
  plot(hi[!is.na(Rh)],Rh[!is.na(Rh)],type="l", ylab="CV(h)",xlab="hi")
```

B.6 R Codes for Table 3.3

```
# Simulation for Table 3.3
  set.seed(66889)
  a=1;
  b=1;
  reg=matrix(0,nrow=3,ncol=3);
  kk=1
  for(n in c(100,200,300))
  {
     x=rnorm(n,0,3);
     u=rexp(n,1)-rexp(n,1);
     e=rnorm(n,0,1);
     y=a+b*x+e
     z=x+u;

     hseq=seq(1.6,6,by=0.05)
     CV=rep(0,length(hseq));
```

```
   k=1;

   for(h in hseq)
   {
      zdiff=kronecker(z,z,"-");
      Atemp=exp(zdiff)*(1-pnorm(zdiff/h+h))-
      exp(-zdiff)*pnorm(zdiff/h-h);
      At=matrix(Atemp,nrow=n);
      Bt=matrix(dnorm(zdiff/h),nrow=n);
      res=rep(0,n);

      for(i in seq(n))
      {
         yi=y[-i];
         wi=z[-i]+h*exp(h^2/2)*apply(At[-i,-i],2,sum)/
         apply(Bt[-i,-i],2,sum);
         myreg=lm(yi~wi)$coefficients;
         cat(myreg,"\n")
         res[i]=y[i]-myreg[1]-myreg[2]*(z[i]+h*exp(h^2/2)*
         apply(At[-i,],2,sum)[i]/apply(Bt[-i,],2,sum)[i]);
      }
      CV[k]=mean(res^2)
      k=k+1;
   }

   plot(hseq, CV,type="l")
   x11()
   h=hseq[CV==min(CV)]
   zdiff=kronecker(z,z,"-");
   Atemp=exp(zdiff)*(1-pnorm(zdiff/h+h))-exp(-zdiff)*
   pnorm(zdiff/h-h);
   At=matrix(Atemp,nrow=n);
   Bt=matrix(dnorm(zdiff/h),nrow=n);
   w=z+h*exp(h^2/2)*apply(At,2,sum)/apply(Bt,2,sum);
   regtemp=lm(y~w)$coefficients;
   reg[kk,]=c(h,regtemp[1],regtemp[2]);
   kk=kk+1
}
reg
```

B.7 R Codes for Table 3.4

```
 total=1000
 aTrue=bTrue=rep(0,total)
 aNaive=bNaive=rep(0,total)
 aCrect=bCrect=rep(0,total)
 aOracle=bOracle=rep(0,total)
 aTwid=bTwid=rep(0,total)
 aStein1=bStein1=rep(0,total)
 aStein2=bStein2=rep(0,total)

for(k in seq(total))
   {
   # Generating Sample
   n=100;
   h=0.8*n^(-1/5)
   su=sqrt(1/4);
   x=rnorm(n,0,1);
   u=rdoublex(n,0,su/sqrt(2));
   e=rnorm(n,0,1);
   z=x+u;
   y=1+x+e;
   su2=su^2;

#True
   bTrue[k]=cov(y,x)/cov(x,x);
   aTrue[k]=mean(y)-bTrue[k]*mean(x);

# Naive

   bNaive[k]=cov(y,z)/cov(z,z);
   aNaive[k]=mean(y)-bNaive[k]*mean(z);

# Bias-Corrected

   bCrect[k]=cov(y,z)/(cov(z,z)-su2);
   aCrect[k]=mean(y)-mean(z)*bCrect[k];

# Known x and u distribution

   Znum=Zdem=rep(0,n)
   f0=function(v)
```

```
    {
      ((1-pnorm(sqrt(2)/su-v))*exp(1/su2-v*sqrt(2)/su)+
      pnorm(-sqrt(2)/su-v)*exp(1/su2+v*sqrt(2)/su))*
      exp(-sqrt(2)*v/su)
    }
  f1=function(v)
    {
     ((1-pnorm(sqrt(2)/su-v))*exp(1/su2-v*sqrt(2)/su)+
     pnorm(-sqrt(2)/su-v)*exp(1/su2+v*sqrt(2)/su))*
     exp(sqrt(2)*v/su)
    }
  f2=function(v)
    {
      (1-pnorm(sqrt(2)/su-v))*exp(1/su2-v*sqrt(2)/su)+
      pnorm(-sqrt(2)/su-v)*exp(1/su2+v*sqrt(2)/su)
    }

  for(j in seq(n))
    {
     Znum[j]=integrate(f0,z[j],200)$value*exp(sqrt(2)*z[j]/su)/
     f2(z[j])
     Zdem[j]=-integrate(f1,-200,z[j])$value*exp(-sqrt(2)*z[j]/su)/
     f2(z[j])
    }


  ez=z+Znum+Zdem
  myreg=lm(y~ez)$coefficients;
  aOracle[k]=myreg[1]
  bOracle[k]=myreg[2]

# Nonparametric Tweedie Estimate
  for(i in seq(n))
   {
     q_one=(z[i]-z)/h+h*sqrt(2)/su;
     q_two=(z[i]-z)/h-h*sqrt(2)/su;
     pnorm1=pnorm(q_one);
     pnorm2=pnorm(q_two);
     numerator_one=(exp((z[i]-z)*sqrt(2)/su)*(1-pnorm1)
                   -exp((z-z[i])*sqrt(2)/su)*(pnorm2));
     denominator_one=(1/(sqrt(2*pi))*exp(-1/2*((z[i]-z)/h)^2));
     ez[i]=z[i]+h*exp(h^2/su2)*sum(numerator_one)/
     sum(denominator_one)
    }
   myreg=lm(y~1+ez)$coefficients;
   bTwid[k]=myreg[2]
   aTwid[k]=myreg[1]

 # Stein Estimate (Alice S. Whitemore)

   ez1=z-(n-2)*z/sum(z^2)
   ez2=z-su2*(n-3)*(z-mean(z))/(var(z)*(n-1))

   myreg=lm(y~ez1)$coefficients
   aStein1[k]=myreg[1]
   bStein1[k]=myreg[2]

   myreg=lm(y~ez2)$coefficients
   aStein2[k]=myreg[1]
   bStein2[k]=myreg[2]
   cat(k,"\n")

}
est1=c(mean(aTrue),mean((aTrue-1)^2),mean(bTrue),
mean((bTrue-1)^2))
est2=c(mean(aNaive),mean((aNaive-1)^2),mean(bNaive),
mean((bNaive-1)^2))
est3=c(mean(aCrect),mean((aCrect-1)^2),mean(bCrect),
mean((bCrect-1)^2))
est4=c(mean(aTwid),mean((aTwid-1)^2),mean(bTwid),
mean((bTwid-1)^2))
est5=c(mean(aStein1),mean((aStein1-1)^2),mean(bStein1),
mean((bStein1-1)^2))
est6=c(mean(aStein2),mean((aStein2-1)^2),mean(bStein2),
mean((bStein2-1)^2))
est7=c(mean(aOracle),mean((aOracle-1)^2),mean(bOracle),
mean((bOracle-1)^2))

result=cbind(est1,est2,est3,est4,est5,est6,est7)
dimnames(result)=list(c("alpha (Mean)","alpha (MSE)","beta (Mean)",
"beta MSE"),
      c("True","Naive","Bias-Corrected","Tweedie","Stein1","Stein2",
      "Oracle"))
round(result,4)

ymax=max(density(bTrue)$y,density(bNaive)$y,density(bOracle)$y,
         density(bCrect)$y,density(bTwid)$y)
xmin=min(bTrue,bNaive,bCrect,bOracle,bTwid);
xmax=max(bTrue,bNaive,bCrect,bOracle,bTwid);
plot(density(bTrue),type="l",lwd=2,ylim=c(0,ymax),xlim=c(xmin,xmax),
  xlab="Estimates of Slope","Density",main="")
lines(density(bNaive),lty=1,lwd=1)
lines(density(bCrect),lty=2,lwd=1)
lines(density(bOracle),lty=4,lwd=1)
lines(density(bTwid) ,lty=5,lwd=2)

legend("topright", legend = c("True", "Naive","Bias-Corrected",
           "Oracle","Tweedie"), lwd=c(2,1,1,1,2),xjust = 1, yjust=1,
      cex=0.8,lty=c(1,1,2,4,5),bty="n")
```