A REDESCENDING M-ESTIMATOR FOR DETECTION AND DELETION OF OUTLIERS IN REGRESSION ANALYSIS

BY

ANEKWE, STELLA EBELE

2012557001F

A DISSERTATION SUBMITTED TO THE DEPARTMENT OF STATISTICS NNAMDI AZIKIWE UNIVERSITY, AWKA, ANAMBRA STATE, IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF DOCTOR OF PHILOSOPY (Ph.D.) IN STATISTICS

DECEMBER, 2019.

CERTIFICATION

I, STELLA EBELE ANEKWE, hereby certify that I am responsible for the research in this dissertation and that is an original work which has not been submitted to this University or any other institution for the award of a degree or diploma.

....

Signature of Candidate Date

APPROVAL

This dissertation written by Stella Ebele Anekwe has been examined and approved for the award of degree of Doctor of Philosophy in Statistics from Nnamdi Azikiwe University, Awka.

....

Prof S.I. Onyeagu   Date

Supervisor

....

Dr. G. A. Osuji   Date

Head of Department

....

External Examiner   Date

....

Prof Sylvanus Anigbogu   Date

Dean of Faculty

....

Prof. Philomena Igbokwe   Date

Dean, SPGS

DEDICATION

This research is dedicated to God Almighty and our Blessed Virgin Mary.

ACKNOWLEDGEMENTS

I am particularly and sincerely grateful to my supervisor; Prof Onyeagu S.I. whose continuous guidance and encouragement made this dissertation a fruitful one.

I am very thankful to all the Staff at the Department of Statistics, for their endless help throughout my stay in the department.

I want to express my deepest gratitude to my husband, Mr Chike Anekwe, my children, Uche Anekwe and Chibike Anekwe and my parents, Engr and Mrs N.I. Arachie, my siblings, Mr Emeka Arachie, Mr Onyeka Arachie, Mrs Chikaodili Akunne-Nduka and Mr Chimezie Arachie and the rest of my family for their love, care and support.

Ialso thank all my friends who assisted me in one way or the other during this research.

May the good Lord bless and reward you all.

ABSTRACT

M-estimators are robust estimators that give less weight to the observations that are outliers while Redescending M-estimators are those estimators that are built such that extreme outliers are completely rejected. Several researchers proposed different methods of M-estimator and Redescending M-estimators for detection and deletion of outliers as discussed in the literature. However, there is still need to have a Redescending M-estimator that will be more efficient and robust when outliers are in both two-dimensional space compared with the existing ones. In view of this, a Redescending M-estimator is proposed while its objective, influence and weight functions are established.The proposed method is applied to different examples (real-life data) to verify its effectiveness in detecting and deleting outliers. The Monte Carlo simulation method is used to investigate the performance of the newly proposed method. The results from the simulation study and the real life data indicate that the proposed method is very good for detecting and deleting outliers. Furthermore, the proposed method is particularly more efficient and robust when outliers are in both x- and y-directions compared to the existing ones.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICIES

List of Acronyms

Ordinary Least Squares	OLS
Sum of Squared Errors	SSE
Maximum Likelihood Estimators	MLE
Median Absolute Deviation	MAD
Median Square Error	MSE
Inter-Quartile Range	IQR
Generalized Lambda Distribution	GLD
Extreme Studentized Deviate	ESD
M-estimator Correlation Coefficient	MCC
Least Absolute Deviation	LAD
Least Trimmed Squares	LTS
Residual Standard Error	RSE

CHAPTER ONE

INTRODUCTION

1.1 Background of the Study

When a regression model is fitted to a dataset containing some observations which are outlying; that is, observations which are well separated from the remainder of the data, the outlying observations may involve large residuals and often have effects on the fitted least squares regression function. Outliers are extreme observations that do not fall in the same pattern with the majority of data involved in a regression analysis problem. Hawkins (1980) defined outlier as observation which deviates so much from the other observations as to the suspicion that it was generated by a different mechanism. Furthermore, Barnett and Lewis (1994) defined outlier as an observation (or a set of observations) which appears to be inconsistent with the remainder of the set of data. The lower and upper data points in a dataset are known as extreme observations. The declaration of one or more extreme observations to be outliers depends on how they appear in relation to the rest of the data point; but on the other hand, an outlier should always be an extreme observation in a dataset which may be as a result of data entry errors, experimental errors, sampling errors, measurement errors, etc.

However, the presence of even a single outlying observation may greatly affect the performance of ordinary least squares estimation. These outliers violate the assumption of normally distributed residual in least squares regression. Such outlying observations need careful attention and should be detected while extreme outliers may be eliminated.

In the context of outlier detection, many researchers developed various methods. Aggarwal and Yu (2001) discovered a new technique for detecting outliers associated to very high dimensional datasets in which the data can contain hundreds of dimensions.Nguyena and Welch (2010) studied outlier detection and proposed a new trimmed square approximation for identifying extreme outliers. Hadi and Simonoff (1993) introduced two test procedures for the detection of multiple outliers in a linear model.They illustrated and compared those procedures to various existing methods, using several datasets containing multiple outliers. They also investigated the performances of both procedures by a Monte Carlo study.The results from both the MonteCarlo study and the datasets indicated that both procedures are effective in the detection of multiple outliers in a linear model.Zhang et al. (2015) proposed an enhanced Monte Carlo outlier detection method by establishing cross-prediction models based on normal samples and analyzing the distribution of prediction errors for dubious samples. Three real datasets and a simulation study were used to illustrate the performances of its method. The results indicated that the enhanced Monte Carlo outlier detection method out performed the Monte Carlo outlier detection method in outlier diagnosis. Other authors who studied detection of outliers include: Tukey (1977), Atkinson (1994), Becker and Gather (1999) and Carling (2000).

1.2 Regression Analysis

Regression analysis technique is used to measure the relationship between two or more variables. The technique measures an appropriate value of the dependent variable in response to a change in the independent variable(s) of a function. Let $y$denote the response that is linearly related to k independent variables, x_1,$x_2, \dots, x_{k,}$ the parameters,β_0 (slope),$\beta_1, \dots, \beta_k$, andε is the random error, then, the multiple linear regression model is

$$y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_k X_k + \varepsilon (1.1)$$

Illustrating equation (1.1) inTable 1.1 as shown below:

Assuming an experiment is conducted n times and the data is obtained as follows.

Observation number	**Response Y**	**Explanatory variables $X_1, X_2, \dots, X_K$**
1	y_1	$X_{11}, X_{12}, \dots, X_{1K}$
2	y_2	$X_{21}, X_{22}, \dots, X_{2K}$
.	.	. . .
.	.	. . .
.	.	. . .
n	y_n	$X_{n1}, X_{n2}, \dots, X_{nK}$

Table 1.1: Array of Data Consisting of n Observations of a Response Variable on k Explanatory Variables

The standard multiple regression model in matrix notation is given as

$$Y = X\beta + \varepsilon \tag{1.2}$$

$$Y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ . \\ y_n \end{pmatrix}, \quad X = \begin{bmatrix} 1 & X_{11} & \dots & X_{1k} \\ 1 & X_{21} & \dots & X_{2K} \\ . & . & . & . \\ . & . & . & . \\ . & . & . & . \\ 1 & X_{n1} & \dots & X_{nk} \end{bmatrix}, \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ . \\ \beta_k \end{pmatrix}, \quad \varepsilon = \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ . \\ \varepsilon_n \end{pmatrix}$$

where $Y = (y_1, y_2, \dots, y_n)'$ is $n \times 1$ vector of n observations, X is $n \times k$ matrix of n observations on each of the k explanatory variables, $\beta = (\beta_0, \beta_1, \dots, \beta_k)'$ is a $k \times 1$ vector of regression coefficients and $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n)'$ is a $n \times 1$ vector of random error components.

The following assumptions are made:

(i) $E\ (\varepsilon) = 0;$

(ii) $E\ (\varepsilon\varepsilon^{'}) = \sigma^2 1_n;$

(iii) $Rank\ (X) = k;$

(iv) X is a non–stochastic matrix;

(v) $\varepsilon \sim N(0, \sigma^2 1_n).$

These assumptions are made in studying the statistical properties of the estimates of regression coefficients.

According to Sokal and Rohlf (2012), Ordinary Least Squares (OLS) regression fit a line to bivariate data such that the (squared) vertical distance from each data point to the line is minimized across all data points.

OLS estimates are obtained by minimizing the sum of squared error (SSE) given as

$$SSE = \sum_{i=1}^{n} \varepsilon_i^2 = \varepsilon^{'}\varepsilon = (Y - X\beta)^{'}(Y - X\beta) \quad (1.3)$$

Expanding (1.3), we obtain:

$$SSE = Y^{'}Y + \beta^{'}X^{'}X - 2\beta^{'}X^{'}Y \quad (1.4)$$

Obtaining the derivative of SSE with respect to β and equating to 0 gives

$$\frac{dSSE}{d\beta} = 2X^{'}X\beta - 2X^{'} = 0 \quad (1.5)$$

This yields normal equation

$$X^{'}X\hat{\beta} = X^{'}Y \quad (1.6)$$

If the rank of $X^{'}X$ is k, then $X^{'}X$ is non–singular and the normal equation has a unique solution given as

$$\hat{\beta} = (X'X)^{-1}X'Y \qquad (1.7)$$

which is termed as Ordinary Least Squares estimator of β.OLS technique is unbiased linear estimation technique of any set of data that is linearly related. It also has the smallest variance among all other unbiased estimators (BLUE).

The error estimation of $\hat{\beta}$ is

$$\hat{\beta} - \beta = (X'X)^{-1}X'Y - \beta$$

$$= (X'X)^{-1}X'(X\beta + \varepsilon) - \beta$$

$$= (X'X)^{-1}X'X\beta + (X'X)^{-1}X'\varepsilon - \beta$$

$$= \beta - (X'X)^{-1}X'\varepsilon - \beta$$

$$= (X'X)^{-1}X'\varepsilon (1.8)$$

Since X is assumed as non-stochastic and $E(\varepsilon) = 0$

$$E(\hat{\beta} - \beta) = (X'X)^{-1}X'E(\varepsilon) = 0 (1.9)$$

Thus, the OLS estimator is the unbiased estimator of β.

The Covariance matrix of $\hat{\beta}$ is $E(\hat{\beta} - \beta)(\hat{\beta} - \beta)'$

$$=E[(X'X)^{-1}X'\varepsilon\varepsilon'(X'X)^{-1}X']$$

$$= (X'X)^{-1}X'E(\varepsilon\varepsilon')(X'X)^{-1}X'$$

$$=\sigma^2(X'X)^{-1}X'1X(X'X)^{-1}$$

$$=\sigma^2(X'X)^{-1}(1.10)$$

When the error term is not constant for all observations, weighted least squares procedure is used.

The weighted least squares normal equations can be expressed as:

$$X'WX\hat{\beta} = X'WY \qquad (1.11)$$

and the weighted least squares estimators are

$$\hat{\beta} = (X'WX)^{-1}X'WY \qquad (1.12)$$

If $W = 1$, then (1.12) reduces to the unweighted estimators (1.7). Draper and Smith (1998) stated that robust regression aims at assigning different weights to data, such that, outlying data is given smaller weights. Thus, observations whose error terms are subject to large variation receive less weight while those that are subject to small variation receive more weights.

Robust regression is an important tool for analyzing data contaminated with outliers. It has been developed for the purpose of improving the results of the least squares estimates in the presence of outliers. Some methods of robust regression discussed in the literature include those of: Huber (1964) who discovered M-estimators which are the generalization of the Maximum Likelihood Estimators (MLE). Rousseeuw (1982) who discovered the Least Median of Squares estimators (LMS) and Rousseeuw (1983) also proposed the Least Trimmed Squares (LTS) estimators. Some Redescending M-estimators for detection and deletion of outliers are also given in: Andrew et al. (1972), Beaton and Tukey (1974), Hampel et al. (1986) and Alamgir et al. (2013).

1.3 M-estimators

M-estimators are robust estimators introduced by Huber (1964) and can be regarded as a generalization of Maximum Likelihood Estimation; hence, the "M". The Maximum Likelihood Estimator (MLE) is a method of estimating the parameters of a model by maximizing the model's likelihood function.

We consider the linear model in equation (1.1)

The fitted model is

$$\hat{y}_i = \beta_0 + \beta_i x_{i1} + \beta_2 x_{i2} + \ldots . + \beta_p x_{ip} = X_i' \hat{\beta} \quad (1.13)$$

and,

$$\hat{y}_i = \mathrm{y_i} - r_i \quad (1.14)$$

where $\hat{y}_i$ is the vector of predicted or estimated value of y and p is the number of explanatory variables and r_i are the residuals.

This implies that residuals, r_i is given as

$$r_i = y_i - \hat{y}_i$$

$$= \mathrm{y_i} - \mathrm{X_i'} \hat{\beta} \quad (1.15)$$

To obtain the parameter β in MLE, we minimize the negative log function given as

$$\hat{\beta}_{MLE} = minimize \sum_{i=1}^{n} [-\log f\ (y_i\ ; \beta)] \quad (1.16)$$

while Ordinary Least Squares (OLS) minimizes the residual sum of squares, that is,

$$\hat{\beta}_{OLS} = minimize \sum_{i=1}^{n} r_i^2 \quad (1.17)$$

Replacing the squared error term in equation (1.17) by $\rho(\mathrm{r})$, M-estimator is given as

$$\hat{\beta}_{M-estimator} = minimize \sum_{i=1}^{n} \rho(\mathrm{r}) \quad (1.18)$$

where $\rho(r)$ is the objective function of an M-estimator

Standardizing the residuals, $\mathrm{r_i}$, equation (1.18) can also be written as

$\hat{\beta}_{M-estimator} = minimize \sum_{i=1}^{n} \rho \left(\frac{\mathrm{r_i}}{\hat{\sigma}}\right)$(1.19)

where $\hat{\sigma}$ is the scale parameter given as

$\hat{\sigma} = \frac{MAD}{k}$(1.20)

k is a constant given as 0.674 and MAD is the Median Absolute Deviation also given as

MAD= $median(|r_i - \ median(r_i)|)$(1.21)

Since standard deviation is not resistantto outliers (not unduly influenced by a few number of outliers), the Median Absolute Deviation (MAD) is used as a measure of spread in robust regression.

Objective Function,$\rho(r)$

The objective function of an M-estimator,$\rho(r)$, defines the probability distribution of the M-estimator.

The properties of the objective function include;

(1) $\rho(0) = 0$.

(2) $\rho(r_i) \geq \ 0$.

(3) $\rho(r_i) = \rho(-r_i)$.

(4) $\rho(r_i) \leq \rho(r_j)$ for 0< $r_i < r_j$

(5)The objective function is continuous and diffentiable.

We illustrate the above five properties using the objective function, $\rho(r)$ of the Alarm M-estimator **(**Alamgir et al. (2013)

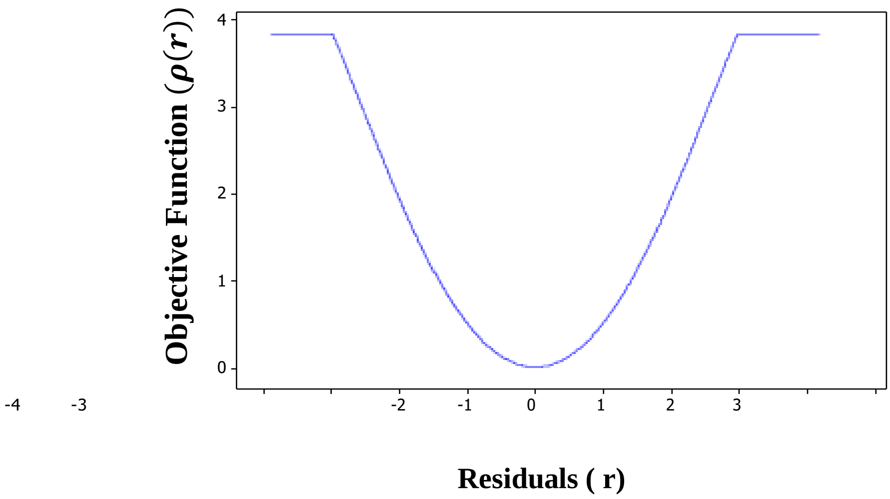

Figure 1.1: Graph of the Alarm's Objective Function, $\rho(r)$ (Alamgir et al. (2013)

From Figure 1.1, the$\rho(0) = 0$.

Secondly, the values of the objective function are non-negative, that is, from 0 to 3.8. Thirdly, the objective function is symmetric, that is,

$$\rho(1) = \rho(-1)$$

$$\rho(2) = \rho(-2)$$

$$\rho(3) = \rho(-3)$$

$$\rho(4) = \rho(-4)$$

Lastly, the graph is smooth, have no breaks and its derivative exist at all points in its domain,which implies that, it is adifferentiable and continuousfunction.

Influence Function, $\psi(r)$

The influence function describes the sensitivity of the overall estimate on the outlying data. It shows the effect of outliers on the value of the estimator. Hampel (1974) disclosed that the robustness of an estimator is measured by its influence function. The derivativeof the Objective function,$\rho(r)$with respect to the regression coefficient β gives rise to the influence function,$\psi(r)$, that is,

$$\psi(r) = \frac{d[\sum_{i=1}^{n} \rho(\mathrm{r_i})]}{d\beta} = \Sigma\psi(r)X_i = \Sigma\psi(\frac{\mathrm{y_i - X_i'}\hat{\beta}}{\hat{\sigma}})X_i \quad (1.22)$$

where$\hat{\sigma}$ is the scale parameter.

Weight Function

Draper and Smith (1998) defined the weighted function, w_i, as

$$w_i = \frac{\psi\left(\frac{\mathrm{y_i - X_i'}\hat{\beta}}{\hat{\sigma}}\right)}{\left(\frac{\mathrm{y_i - X_i'}\hat{\beta}}{\hat{\sigma}}\right)} \quad (1.23)$$

To derive the weighted least squares, we multiply the influence function, that is, equation (1.22) by $\frac{\left(\frac{y_i - X_i'\hat{\beta}}{\hat{\sigma}}\right)}{\left(\frac{y_i - X_i'\hat{\beta}}{\hat{\sigma}}\right)}$ and equate to zero.

$$\Sigma\psi\left(\frac{y_i - X_i'\hat{\beta}}{\hat{\sigma}}\right)X_i \cdot \frac{\left(\frac{y_i - X_i'\hat{\beta}}{\hat{\sigma}}\right)}{\left(\frac{y_i - X_i'\hat{\beta}}{\hat{\sigma}}\right)} = 0$$

Therefore, the weighted least squares, is given by

$$\sum_{i=1}^{n} w_i \left(y_i - X_i'\hat{\beta}\right)X_i = 0 \quad (1.24)$$

where

$$w_i = \frac{\psi\left(\frac{y_i - X_i'\hat{\beta}}{\hat{\sigma}}\right)}{\left(\frac{y_i - X_i'\hat{\beta}}{\hat{\sigma}}\right)}$$

1.3.1 Huber M-estimator

Huber (1964) proposed the Huber M-estimator and its influence function, $\psi(r)$, is

$$\psi(r) = \begin{cases} -c & ; r < -c \\ r & ; -c \leq r \leq c \\ c & ; r > c \end{cases} \quad (1.25)$$

where c is arbitrary value known as tuning constant and r are the residuals scaled over Median Absolute Deviation (MAD).Huber estimator is not robust when the outliers present in the data are in x-direction (leverage points). Leverage points are when outliers are in the explanatory variables. The Huber influence function is non-decreasing function with a tuning constant c = 1.345 which yields 95% efficiency on a normal distribution (the tuning constant$c$,determines the degree of robustness in M-estimators).

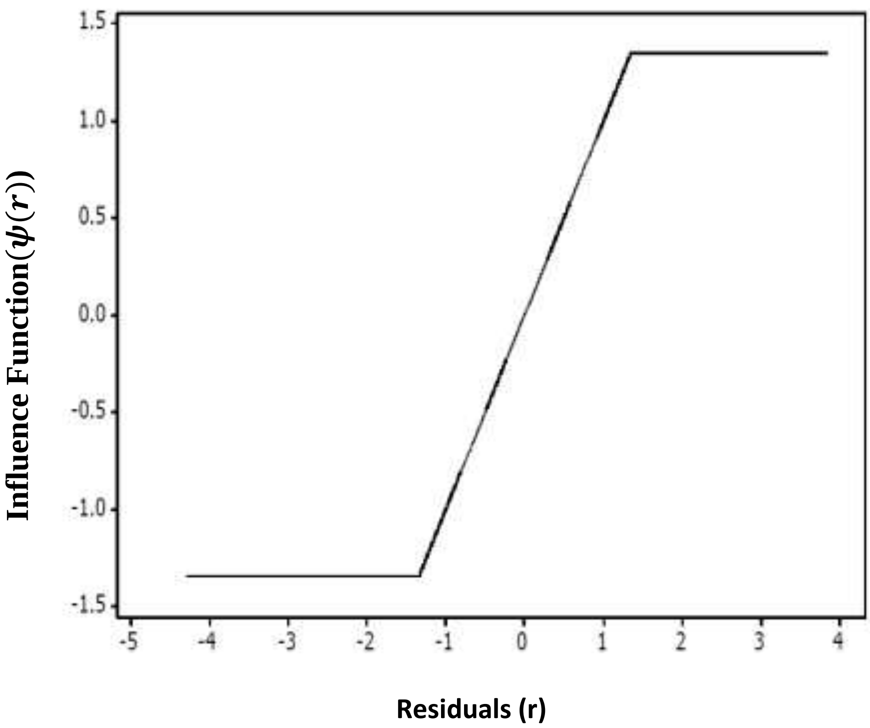

Figure 1.2: Graph of the Huber Influence Function(Huber, 1964)

From Figure 1.2, the residuals are on the x-axis while their corresponding $\psi(r)$ are shown in the y-axis. It could be shown that the extreme residuals, that is -4 and 4, are given influence values of -1.3 and 1.3 respectively, which implies that, Huber estimator does not delete outliers in a robust fit.

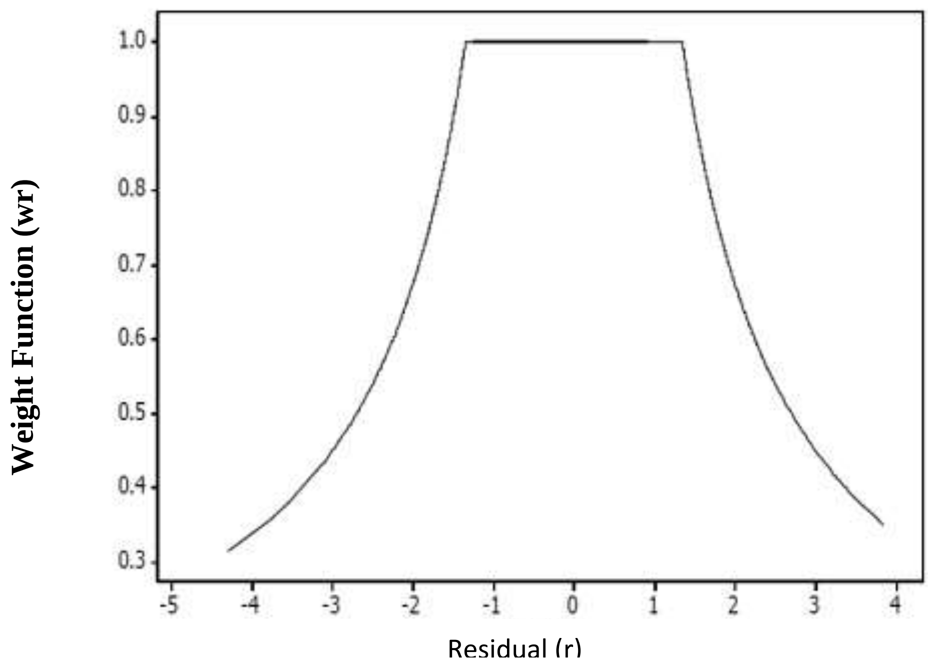

Figure 1.3: Graph of the Huber Weight Function(Huber, 1964)

From the above figure, the observations (residuals) are in x-axis while their corresponding weights are in y-axis. The very good observations, that is, -1, 0, 1, were assigned very good weights, that is 1, while extreme residuals or outliers, that is, -4 and 4, were assigned weights of 0.3 each. This proves that Huber estimator does not delete extreme outliers rather smaller weights were assigned to them.

1.4 Redescending M-estimators

Redescending M-estimators are estimators with ψ-functions redescending to zero, that is, the influence functions of the extreme outliers are zero, which implies that, extreme outliers are rejected. Some of these estimators discussed in the literature are:

1.4.1 Hampel M-estimator

Hampel's three–part Redescending M-estimator was proposed by Hampel et al. (1986) in the Princeton Robustness study. Princeton Robustness study is an extensive theoretical and Monte Carlo study of different robust estimators published in, Andrew et al. (1972). Its estimator has three tuning constants a, b and c.

Itsψ-function is given as

$$\psi(r) = \begin{cases} r & ; \text{if } |r| \leq a \\ a \text{ sign } (r) & ; \text{ if } a < |r| \leq b \\ \frac{(c-|r|)}{(c-b)} a \text{ sign } (r) & ; \text{ if } b < |r| \leq c \\ 0 & ; \text{if } |r| > c \end{cases} \quad (1.26)$$

where a, b, c are positive constants and $0 < a \leq b < c < \infty$and r are the residuals scaledover Median Absolute Deviation, MAD.

The drawback of this estimator is that, it is non-differentiable, so there is still need to propose an estimator that will be differentiable.

1.4.2 Tukey's Biweight M-estimator

Beaten and Tukey (1974) proposed Tukey's Biweight M-estimator and its ψ-function is given as

$$\psi(r) = \begin{cases} r\{1 - (\frac{r}{c})^2\}^2 & ; |r| > c \\ 0 & ; \text{otherwise} \end{cases} \quad (1.27)$$

where c is arbitrary value known as tuning constant and r are the residuals scaled over Median Absolute Deviation(MAD). For Tukey's biweight, c = 4.685 gives 95% efficiency on normal distribution.

The performance of Tukey's Biweight estimator was good, that is, its influence function is differentiable and smooth when compared to the methods proposed by Huber (1964) and Hampel et al. (1986).

1.4.3 Alarm M-estimator

Alamgir et al. (2013) proposed the Alarm's Redescending M-estimator for robust regression and outlier detection. Its ψ -function is given as

$$\psi(r) = \begin{cases} \frac{16r\,(e^{-(r/c)^2}}{(1+e^{-(r/c)^4}} & ; |r| \leq c \\ 0 & ; |r| > c \end{cases} \quad (1.28)$$

where c is the tuning constant and r are the residuals scaled over Median Absolute Deviation (MAD).

The Alarm estimator was based on the modified tangent hyperbolic-type (tan h) weight function. The Mean Square Errors (MSE) of the Alarm estimator are the smallest when compared with those of Huber (1964), Beaton and Tukey (1974) and Hampel et al. (1986) estimators, yielding efficient results. For its empirical study, c = 3 gives approximately 95% efficiency at normal distribution.

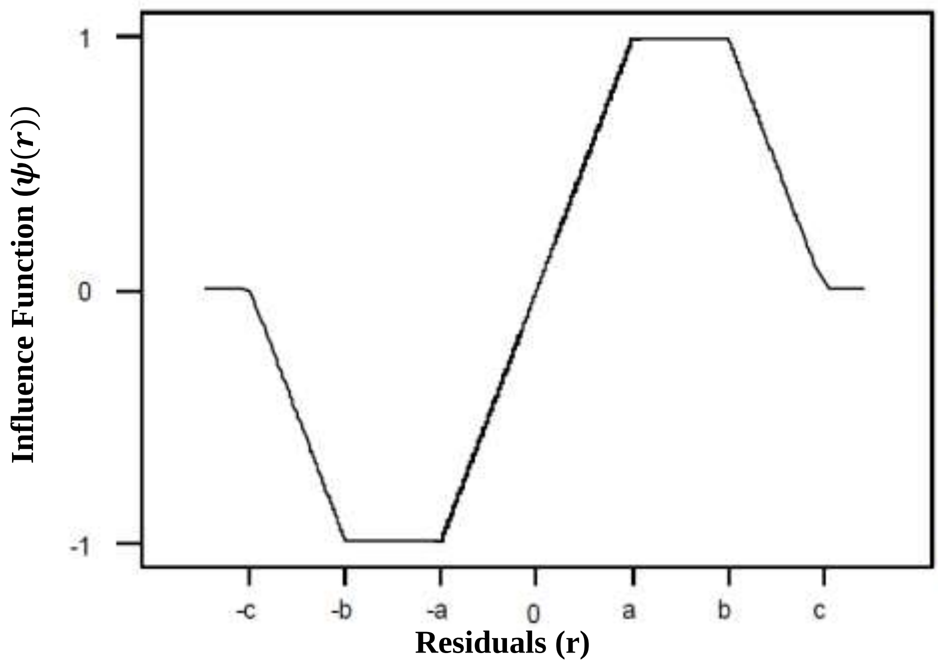

Figure 1.4: Graph of Hampel's three part Influence Function(Hampel et al. (1986))

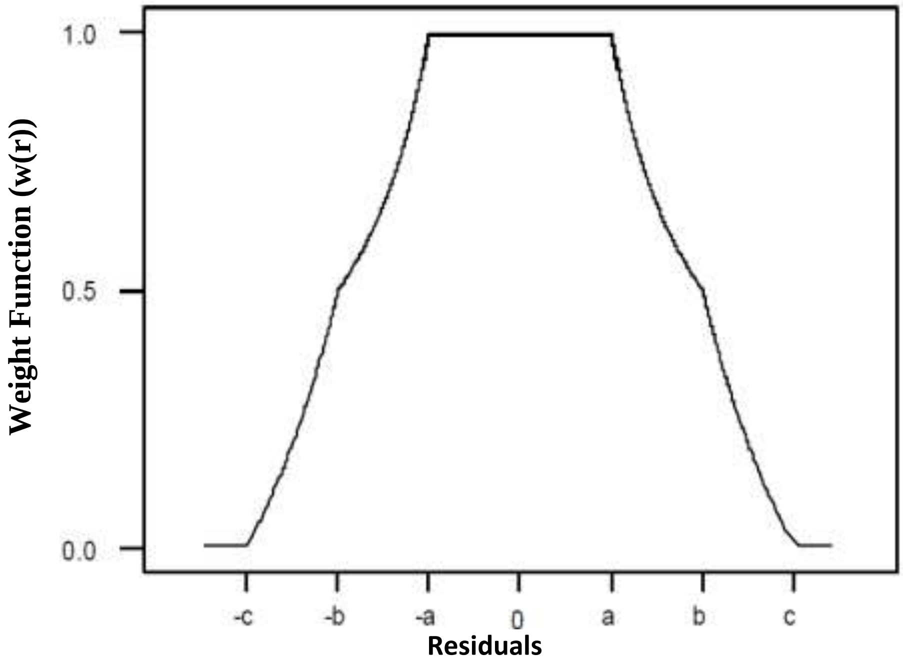

Figure 1.5: Graph of Hampel's three part Weight Function(Hampel et al. (1986))

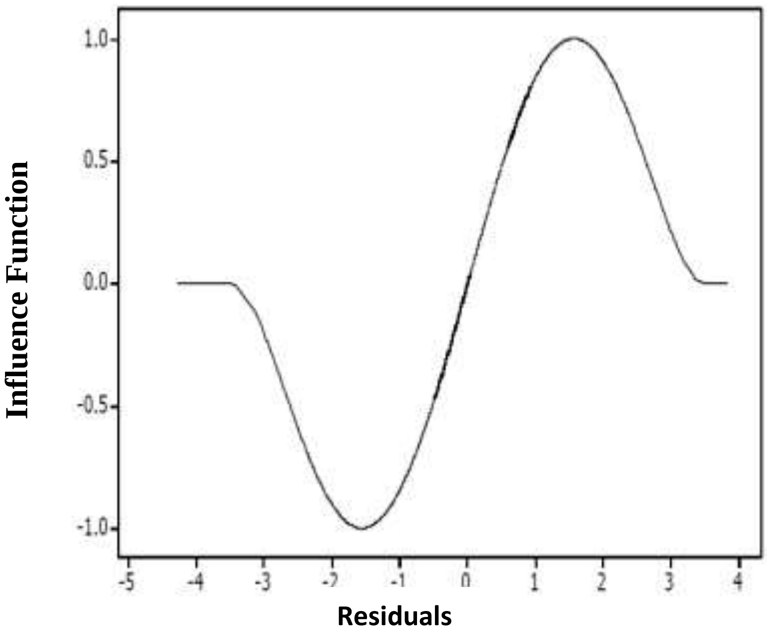

Figure 1.6: Graph of Tukey's Biweight Influence Function(Beaton and Tukey (1974))

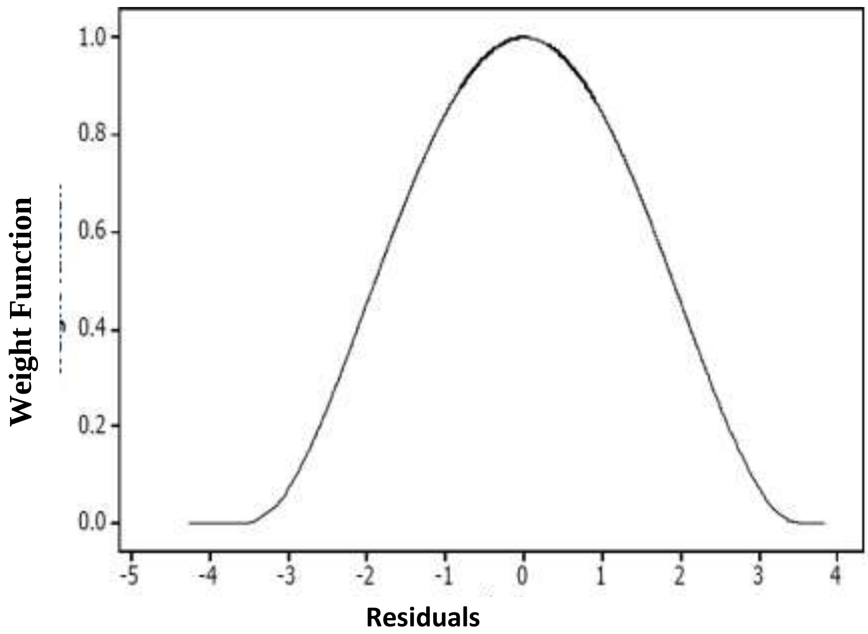

Figure1.7: Graph of Tukey's Biweight Weight Function(Beaton and Tukey (1974))

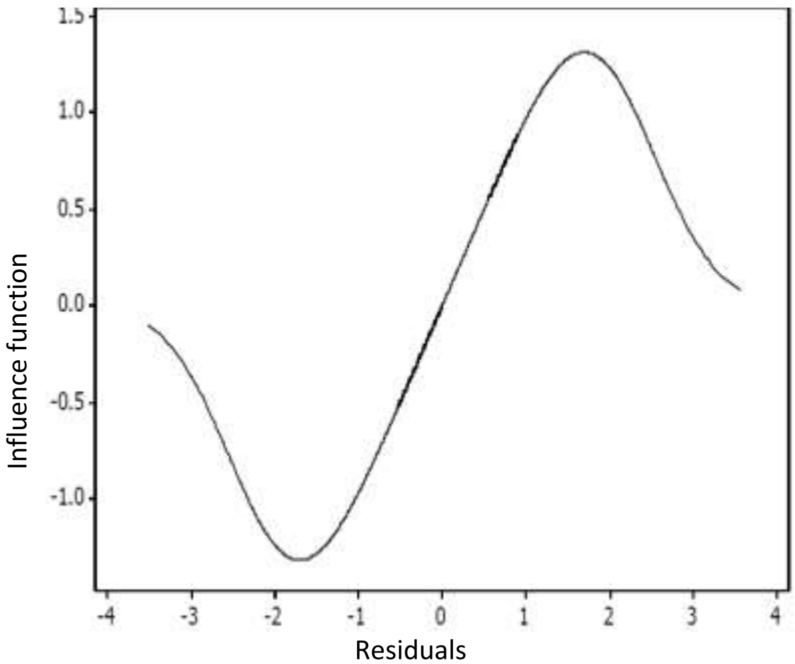

Figure 1.8: Graph of Alarm Influence Function(Alamgir et al. (2013))

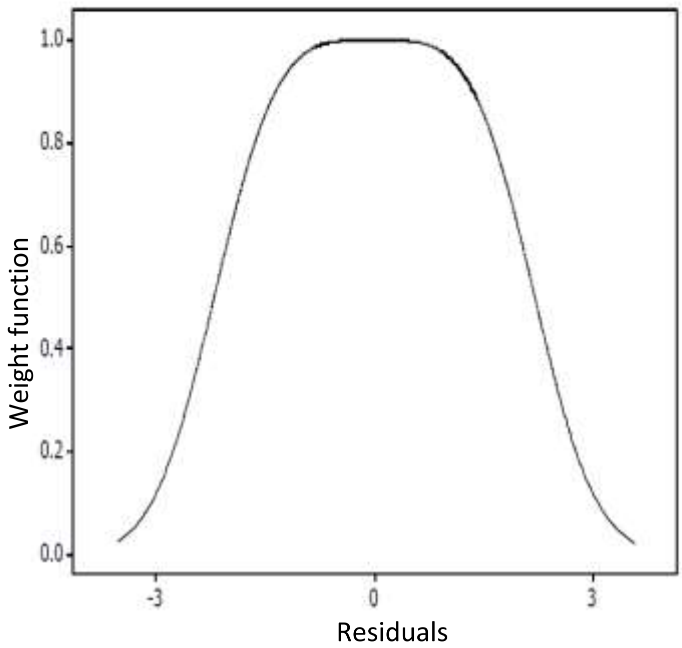

Figure 1.9: Graph of Alarm Weight Function(Alamgir et al. (2013))

From Figures 1.4, 1.6, and 1.8, the graphs of the three influence functions all redescend to zero, that is, extreme outliers were assigned zero in the robust fit.

Example for Illustration Purpose

Using the graph of Tukey's Biweight influence function (Figure 1.6), both extreme outliers (-4 and 4), redescend to zero, that is, their influence functions are zero ($\psi(4) = \psi(-4) = 0$). It also showed that outliers have zero or no influence in Redescending M-estimators and are not to be included in the robust fit.

In addition, the graphs of the three weight functions (Figures 1.5, 1.7, and 1.9) showed that the extreme outliers were rejected by assigning zero weights to them.

Example for Illustration Purpose

Using the graph of Alarm weight function (Figure 1.9), the values of the residuals (-3 to 3) are in the x-axis while their corresponding weights (0 to 1) in the y-axis. The outliers (-3 and

3) were assigned weights of zero while non-outlying values were assigned higher weights. This implies that Redescending M-estimators detect and delete outliers in a robust fit.

1.5 Statement of the Problem

Huber estimator (Huber, 1964) does not delete large residuals, and this brings about the Redescending M-estimators. The Hampel's three-part function (Hampel et al. (1986)) is non-differentiable. A differential function is a continuous function whose derivative exists at all points in its domain. Wang and Opsomer (2011) stated that the theoreticalproperties of non-differentiable estimators are substantially more complicated to derive than those of differentiable estimators. The Huber (1964) and Beaton and Tukey (1974) estimators are not robust to outliers in the leverage points. The methods of estimation produced by Huber (1964), Beaton and Tukey (1974), Hampel et al. (1986) and Alamgir et al. (2013) need an improvement in handling outliers in both the x and y axes. Therefore, to handle these problems, there is a need to propose a Redescending M-estimator that is smooth, differentiable and also could handle outliers in both x and y directions.

1.6 Aim and Objectives of the Study

The aim of the research work is to propose a robust estimator for the detection and deletion of extreme outliers in regression analysis.

The objectives are:

(i) to propose a Redescending M-estimator (that will be differentiable and continuous) which includes the objective function (ρ-function), the corresponding influence function (ψ-function) and weight function (w-function);

(ii) to determine the various properties and shapes of the objective, influence and weightfunctions in view of achieving the qualities of a good Redescending M-estimator.

(iii) to compare the proposed Redescending M-estimator with some existing M-estimators and Redescending M-estimators in terms of efficiency and robustness.

1.7 Scope of the Study

The study covers robust estimators for detection and deletion of outliers in a regression model. Monte-Carlosimulationstudies as well as real life application were considered in the study to enable us examine the performance of the proposed method, and for comparison of the proposed method with some existing methods in the literature.

1.8 Significance of the Study

The main purpose of this research is to propose a Redescending M-estimator for detecting and deleting of outliers in regression analysis. The proposed Redescending M-estimator should be of great importance to researchers whenever outliers were discovered in their data.

CHAPTER TWO

LITERATURE REVIEW

2.1 Methods of Outlier Detection

Aggarwal and Yu (2001) discovered a new technique for outlier detection associated with very high dimensional datasets, in which the data can contain hundreds of dimensions. They implemented the technique effectively for high-dimensional applications using an evolutionary search technique. They also discussed the application of outlier detection method to high dimensional problems such as data mining (data mining is the practice of examining large pre-existing databases to generate new information).

Chrominski (2010) used various methods of outlier detection in medical diagnoses. The methods of outlier detection used were; Grubb's test, Dixon's test, Hampel's test and Quartile method. They discussed the detection speed and performances of those outlier methods. Results from the analysis showed that Hampel's test and Quartile method are easier and faster in outlier detection than the Grubb's and Dixon's tests.

Tukey (1977) developed the boxplot which was very helpful in the detection of outliers. This method does not require any distributional assumptions neither does it depend on mean or standard deviation and also suggested that the lower quartile (q_1) is the 25^{th} percentile, and the upper quartile (q_3) is the 75^{th} percentile of the data. The inter-quartile range (IQR) is defined as the interval between q_1 and q_3. The boxplot has q_1 - (1.5 * IQR) and q_3 + (1.5 * IQR) as "Inner fences", q_1 – (3 * IQR) and q_3 + (3 * IQR) as "Outer fences" such that, the observations between an inner fence and its nearby outer fence are regarded as "outside", and anything beyond outer fences as "far out".

Carling (2000) introduced the median rule for identification of outliers through studying the relationship between target outlier percentage and Generalized Lambda Distributions (GLDs). GLDs with different parameters were used for various moderately skewed distributions. The median substitutes for the quartiles of Tukey's method and a different scale of the IQR is employed in the proposed method. The proposed method is good compared to Tukey's method in the sense that it is more resistant and its target outlier percentage is less affected by its sample size in the non-Gaussian case.

Manoj and Kaliyaperumal (2013) compared the performances of five outlier detection methods (Grubbs test, Dixon test, Hampel method, Quartile method and generalized ESD (generalized Extreme Studentized Deviate). Their aim was to find amongst the five methods the one that would strongly detect outlier in a dataset. They carried out an experimentusing R software, and the result showed that the three methods (Hampel method, Quartile method and Generalized ESD) were better than Grubbs and Dixon test.

Zhang et al. (2015) proposed an enhanced Monte Carlo outlier detection method by establishing cross-prediction models based on determinate normal samples and analyzing the distribution of prediction errors for dubious samples. One simulated and three real datasets were used to illustrate and validate the performance of the proposed method. The results showed that the proposed method outperformed Monte Carlo outlier detection in outlier diagnosis.

2.2 Robust Methods for Outlier Detection

Fruhwirth and Waltenberger (2010) constructed a new type of redescending M-estimators based on data augmentation with an unspecified outlier model. They also derived the necessary conditions for the convergence of the resulting estimators to the Huber-type skipped mean. They developed two applications of the annealing M-estimators. The results showed that annealing was instrumental in identifying and suppressing the outliers, and also used for regression diagnostics in the context of estimation of the tail index of a distribution from a sample.

Muller (2004) reviewed the properties and applications of M-estimators with redescending score functions and also stated that the redescending M-estimators can be used to detect sub-structures in the data; that is, they can be used in cluster analysis.

Galimberti et al. (2007) addressed the problem of robustness of regression trees with respect to outlying values in the dependent variable. They also proposed new robust tree-based procedures which were obtained using the Huber and Tukey's objective functions. The performance of the procedure was evaluated through a Monte Carlo experiment. The results showed the usefulness and efficacy of the procedure with respect to the outlying values in the dependent variable.

Arya et al. (2007) proposed a method for robust image registration based on M-estimator correlation coefficient (MCC). A real value correlation mask function was computed using Huber and Tukey's robust statistics and used as similarity measure for registering image windows. The mask function suppresses the influence of outlier points and makes the registration algorithm robust to noisy pixels, brightness fluctuations and presence of occluding objects. The superiority of the proposed algorithm in terms of registration performance and computation time was demonstrated through experimental studies on different types of real world images.

Muthukrishnan and Radha (2010) compared the performances of the robust estimators and that of ordinary least squares estimators in regression study. The study established the fact that the performances of M-estimators were almost the same as the ordinary least squares in normal situations. They further stated that when outliers were present in the data, the least square estimators do not provide useful information for the majority of the data, but not in the case of robust estimators. That is, it was observed that M-estimators were not affected by outliers.

Turkan et al. (2012) proposed alternative robust versions of Cook's distance, Welsch-Kuh distance and the Hadi measure in the detection of outliers. Simulation study was performed, using the ROBUSTREG procedure in SAS version 9, to compare the performance of the classical diagnostics with the proposed versions. The results indicated that, the proposed alternative versions of detection diagnostics seem to be reasonably well and should be considered as worthy robust alternatives to the least squares estimation.

Perez et al. (2014) discussed the outlier detection and robust estimation with data that is naturally distributed into groups which followed approximately a linear regression model with fixed group effects. The result from the simulation method showed that the final regression estimator preserved good efficiency under normality while keeping good robustness properties.

Khan et al. (2016) compared three robust regression techniques (Trimmed Square, the Least Absolute Deviation and a redescending M-estimator) in terms of efficiency and robustness in simple and multiple regressions.

Rousseeuw and Yohai (1984) discovered the least absolute deviation (LAD) estimates, and it's by minimizing the sum of the absolute values of the residuals.

$$\hat{\beta} = \underset{\beta}{argmin} \sum_{i=1}^{n} |y_i - X_i^T \beta| \quad (2.1)$$

The LTS (Least trimmed squares) estimate (Rousseeuw, 1984) is defined as

$$\hat{\beta} = \underset{\beta}{argmin} \sum_{i=1}^{q} r_{(i)}(\beta)^2 \quad (2.2)$$

Where $r_{(i)}(\beta)^2 = (y_i - X_{(i)}^T\beta)^2$,

$r_{(1)}(\beta)^2 \leq r_{(2)}(\beta)^2 \leq \cdots \leq r_{(q)}(\beta)^2$ are squared residuals,

q = $[n(1-\propto) + 1]$, and $\propto$ is the proportion of trimming.

Huber (1964) replaced the least squares criterion with a robust criterion, and is given by $\hat{\beta} = \underset{\beta}{argmin} \sum_{i=1}^{n} \rho\left(\frac{y_i - X_i'\hat{\beta}}{\hat{\sigma}}\right)$(2.3)

where $\rho(r)$is the objective functionand $\hat{\sigma}$ error scale estimate.

The derivative of $\rho(r)$, denoted by $\psi(r) = \rho'(.)$, is called the influence function. Huber (1964) further compared the performance of least squares method with his method using the Monte Carlo simulation. The result showed that his method was more efficient than that of the least squares.

Hampel et al. (1986) proposed the Hampel estimator with ρ-function bounded and ψ −function becoming zero for large $|t|$. The Hampel's three-part function is non-differentiable. Hampel et al. (1986) further demonstrated the good performance of his estimator in the Princeton Robustness study. Princeton Robustness study is an extensive theoretical and Monte Carlo study of different robust estimators published in 1972.

Andrew et al. (1972) and Beaton and Tukey (1974)proposed Redescending M-estimators for detection and deletion of outliers named Andrew sine wave and Tukey biweight estimators respectively. Both Andrew's wave and Tukey's Biweight estimators have smoothly redescending ψ-functions. The performances of Tukey's Biweight and that of Andrew's sine estimator were good, that is, they are differentiable and smooth compared to the methods proposed by Huber (1964) and Hampel et al. (1986).

Alamgir et al. (2013) proposed the Alarm's Redescending M-estimator for robust regression and outlier detection.

The Alarm's estimator was based on the modified tangent hyperbolic (tan h) type weight function. Its estimator was compared with OLS, Huber (1964), Hampel et al. (1986), Andrew et al. (1972) and Beaton and Tukey (1974) in a simulation study as well as real life data. The results obtained showed that the Alarm estimator is more robust and efficient compared to OLS, Huber (1964), Hampel et al. (1986), Andrew et al. (1972) and Beaton and Tukey (1974).

In conclusion, Huber (1964) introduced the M-estimator for detection of outliers while Hampel et al. (1986) improved on Huber (1964) by inventing the Redescending M-estimators for detection and deletion of extreme outliers. Andrew et al. (1972) and Beaton and Tukey (1974) improved on Huber (1964) and Hampel et al. (1986) by producing a smooth and differentiable function. Alamgir et al. (2013) proposed an estimator (based on modified tangent (tan h) type weight function) that is more robust and efficient compared with other existing M-estimators and Redescending M-estimators.

2.3 Literature Gap

Huber estimator (Huber, 1964) does not delete large residuals while The Hampel's three-part function (Hampel et al. (1986)) is non- differentiable. The Huber (1964) and Beaton and Tukey (1974) estimators are not robust to outliers in the leverage points. The methods of estimation produced by Huber (1964), Beaton and Tukey (1974), Hampel et al. (1986) and Alamgir et al. (2013) need an improvement in handling outliers in both the x and y axes. Therefore, to handle these problems, there is a need to propose a Redescending M-estimator that is smooth, differentiable and also could handle outliers in both x and y directions.

CHAPTER THREE

MATERIALS AND METHOD

3.1The Redescending M-Estimator for Detection and Deletion of Outliers in Regression Analysis

We propose a Redescending M-estimator that will be smooth, differentiable, more robust and efficient (to outliers in both x and y directions) compared to Huber (1964), Hampel et al. (1986), Beaton and Tukey (1974) and Alamgir et al. (2013).

To propose the new influence function, $\psi(r)$, (based on modified Tukey's biweight ψ-function) with 95% efficiency at normal distribution, using a tuning constant, c = 3, we introduced a function $g(r)$.

$$g(r) = \left(1 + \left(\frac{r}{c}\right)^2\right)^2 \quad (3.1)$$

$g(r)$ is a smooth and differentiable function for all r, where r are the residuals scaled over Median Absolute Deviation (MAD).

In addition, we multiply the function, $g(r) = \left(\left(1 + \left(\frac{r}{c}\right)^2\right)^2\right)$, by the Tukey's biweight ψ-function =$\left(r\left(1 - \left(\frac{r}{c}\right)^2\right)^2\right)$ resulting in the proposed influence function, $\psi(r)$,given as;

$$\psi(r)=\begin{cases} r\left(1 - \left(\frac{r}{c}\right)^2\right)^2\left(1 + \left(\frac{r}{c}\right)^2\right)^2; & |r| < c \\ 0 & ; \quad |r| \geq c \end{cases} \quad (3.2)$$

where c is the tuning constant for the ith observation and the variable r are the residuals scaled over Median Absolute Deviation (MAD).

By integrating the $\psi(r)$ with respect to r, we obtain the corresponding objective function, $\rho(r)$, given as

$$\rho(r) = \begin{cases} \frac{r^6}{c^4} + \frac{r^{10}}{2c^8} - \frac{2r^6}{c^4} + \frac{r^2}{2} - \frac{2r^6(3r^4 - 5c^4)}{15c^8}; & |r| \leq c \\ \frac{4c^2}{15} \ ; & |r| > c \end{cases} \quad (3.3)$$

where c is the tuning constant for the ith observation and the variable, r, are the residuals scaled over MAD.

Derivation of equation (3.3)

$$\rho(r) = \int_{-\infty}^{\infty} \psi(r) dr$$

(3.4)

where $\psi(r)$ and $\rho(r)$ areinfluence and objectivefunctions, respectively, r are the residuals scaled over MAD (Median Absolute Deviation) .

Given: $\psi(r) = r\left(1 - \left(\frac{r}{c}\right)^2\right)^2 \left(1 + \left(\frac{r}{c}\right)^2\right)^2$

Using the identity;

$$a^2 - b^2 = (a + b)(a - b)$$

Squaring both sides;

$$(a^2 - b^2)^2 = \{(a + b)(a - b)\}^2$$

Where a = 1 and b = $\left(\frac{r}{c}\right)^2$

$$\Rightarrow \psi(r) = r\left(1 - \left(\frac{r}{c}\right)^4\right)^2$$

and

$$\rho(r) = \int \psi(r)dr$$

$$= \int r\left(1 - \left(\frac{r}{c}\right)^4\right)^2 dr$$

Using integration by parts;

Let

u = $\left(1 - \left(\frac{r}{c}\right)^4\right)^2$, $du = -\frac{8r^3(c^4 - r^4)}{c^8}\, dr$

and

$dv = rdr, \quad v = \frac{r^2}{2}$

$$\int \left(1 - \left(\frac{r}{c}\right)^4\right)^2 rdr = \left(1 - \left(\frac{r}{c}\right)^4\right)^2 \left(\frac{r^2}{2}\right) + \int \left(\frac{r^2}{2}\right)\left(\frac{-8r^3(c^4 - r^4)}{c^8}\right)dr$$

$$= \frac{r^2(c^4 - r^4)}{2c^8} + 2\,\frac{(5c^4r^6) - 3r^{10}}{15c^8}$$

$$= \frac{r^{10}}{2c^8} - \frac{r^6}{c^4} + \frac{r^2}{2} - \frac{2r^6(3r^4 - 5c^4)}{15c^8} \quad (3.5)$$

For the second part of $\rho(r)$, we use the same argument in Beaton and Tukey (1974) by substituting r for c in equation (3.5).

$$\int \left(1 - \left(\frac{r}{c}\right)^4\right)^2 rdr = \frac{c^{10}}{2c^8} - \frac{c^6}{c^4} + \frac{c^2}{2} - \frac{2c^6(3c^4 - 5c^4)}{15c^8}$$

$$= \frac{4c^2}{15} \quad (3.6)$$

The proposed $\rho(r)$ satisfies the standard properties of the objective function of an M-estimator as stated in section 1.3.

Dividing the proposed $\psi(r)$by r gives the weight function, $w(r)$, as follows:

$$w(r) = \begin{cases} \left(1 - \left(\frac{r}{c}\right)^2\right)^2 \left(1 + \left(\frac{r}{c}\right)^2\right)^2; & |r| < c \\ 0 & ; |r| \geq c \end{cases} \quad (3.7)$$

Graphs of the proposed objective, influence and weight functions are shown below:

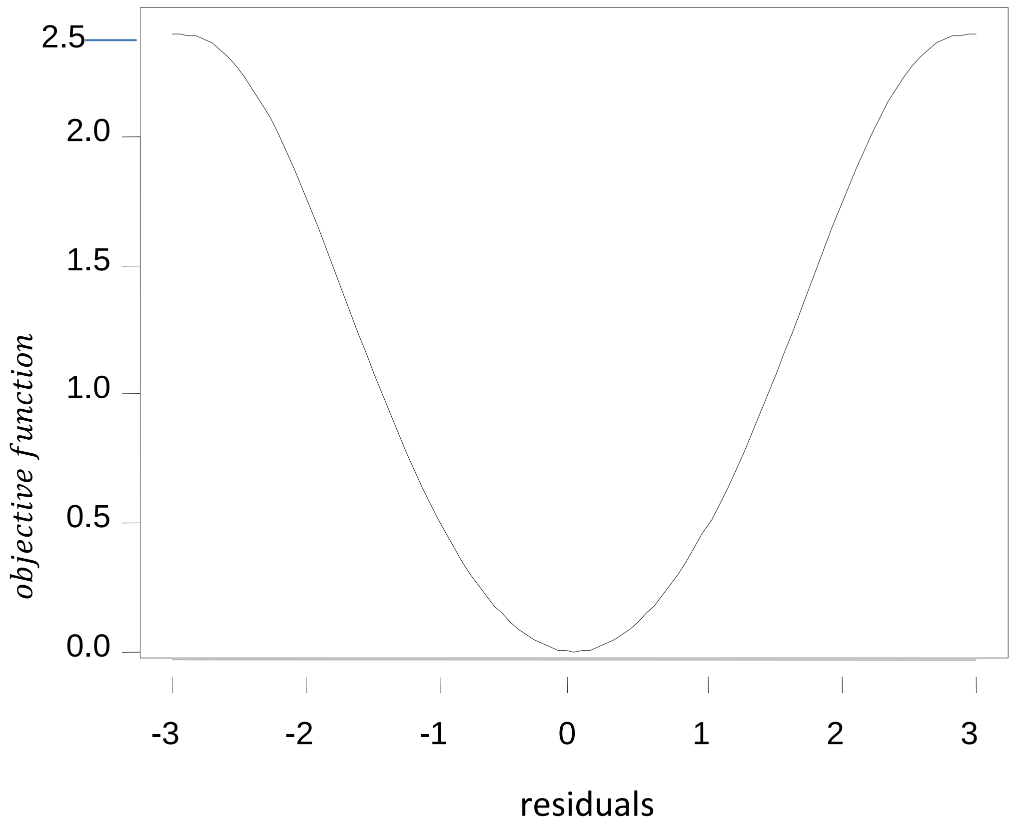

Figure 3.1: Graph of the Proposed Objective Function

From Figure 3.1, the residuals are given from -3 to 3while their corresponding values for the proposed objective function runs from 0 to 2.5.

ToIllustrate the Properties of theObjective Function in Figure 3.1

1. The proposed objective function at zero is equal zero, that is, $\rho(0) = 0$

2. Its values are non- negative (from 0 to 2.5).

3. It is symmetric, which implies that, $\rho(1) = \rho(-1)$

$$\rho(2) = \rho(-2)$$

$$\rho(3) = \rho(-3)$$

Lastly, the graph is smooth and its derivative exists at all points in its domain, which showed that the proposed objective function isa differentiable and continuous function.

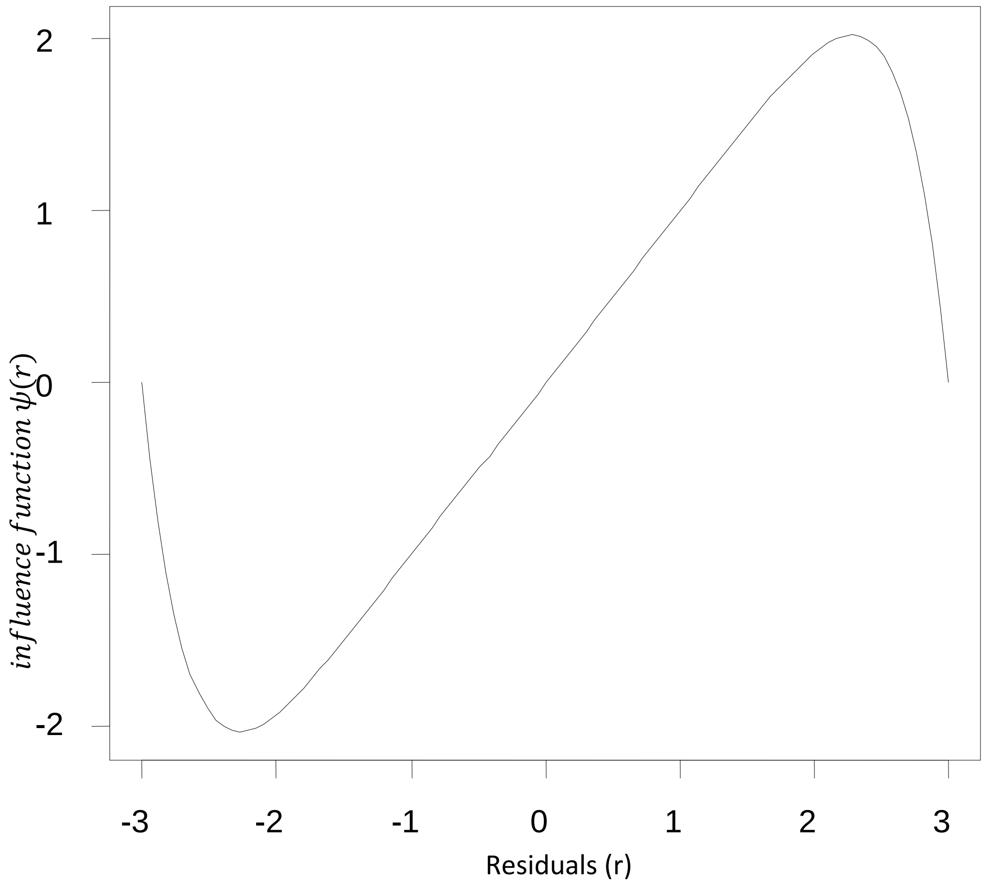

Figure 3.2 : Graph of the Proposed Influence Function

In Figure 3.2, the extreme outliers are -3 and 3 and their corresponding influence functions are 0, that is, $\psi(3) = \psi(-3) = 0$. This implied that the proposed Redescending M-estimator assigned zero or no influence to the extreme outliers.

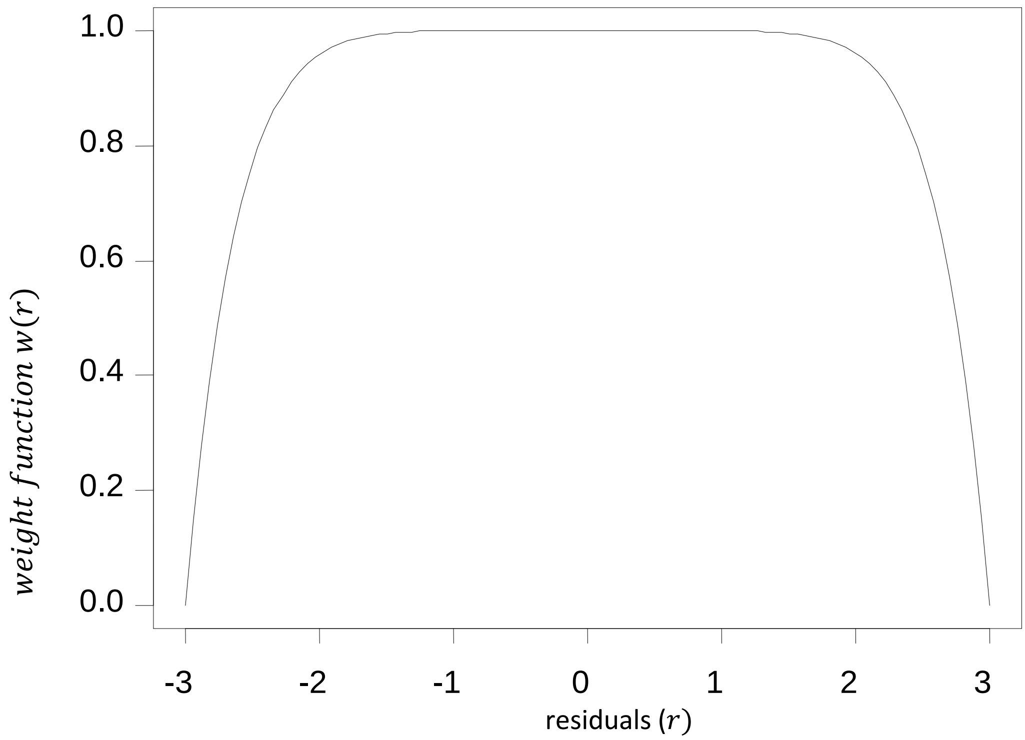

Figure 3.3 : Graph of the Proposed Weight Function

From Figure 3.3, the proposed weight function assigned the very good observations or residuals(-2,-1, 0, 1, and 2) higher weights while the extremer residuals or outliers (-3 and 3) were assigned zero or no weights, which showed that those observations with zero weights were rejected or deleted from the robust fit. This implied that, the proposed weight function detects and deletes outliers in a robust regression analysis.

3.2 Monte Carlo Simulation Method

Monte Carlo simulation method is used to generate random data from different probability distributions. The purpose of our simulation study is to determine the extent our estimates differ from their true values (robustness). Secondly, to compare the proposed estimator in terms of robustness and efficiency with some existing M-estimator and Redescending M-estimators ((Huber (1964), Hampel et al. (1986), Beaton and Tukey (1974) and Alamgir et al. (2013)).

We took the true parameters to be 1, 2, and 5 for β_0, β_1, and β_2, respectively. Each simulation case was replicated $M = 1000$ times. The estimates of each estimator were calculated in each of the iteration and the mean (average) of the M replicated estimates given by:

$$\hat{\beta}_j = \frac{\sum_{i=1}^{M} \hat{\beta}_{ji}}{M} \qquad for\ j = 0, 1, 2, \cdots, p \tag{3.8}$$

was recorded for each estimator.

For comparison, the parameters estimates of the Mean Square Error (MSE) and the absolute bias (BIAS) of the OLS, Huber (1964),Hampel et al. (1986), Beaton and Tukey (1974) and Alamgir et al. (2013) alongside the proposed Redescending M-estimators are computed.

Robustness of an Estimator is measured using absolute bias given as

$$AbsBias(\hat{\beta}_j) = |\beta_j - \hat{\beta}_j| \qquad for\ j = 0, 1, 2, \cdots, p \tag{3.9}$$

A robust estimator has an estimate that is close to the actual parameter irrespective of the distortion in the distribution of the error terms. The lower the bias, the more robust is the estimator.

Efficiency of an Estimator is measured using the MSE (mean square error) defined as

$$MSE(\hat{\beta}_j) = \frac{\sum_{i=1}^{M} (\beta_j - \hat{\beta}_{ji})^2}{M} \qquad for\ j = 0, 1, 2, \cdots, p \tag{3.10}$$

and the variance of the estimator is defined as

$$Var(\hat{\beta}_j) = MSE(\hat{\beta}_j) - \left(Bias(\hat{\beta}_j)\right)^2 \qquad for\ j = 0, 1, 2, \cdots, p \tag{3.11}$$

The estimator with lowest MSE is the most efficient; the smaller the MSE the more efficient is the estimator.

Algorithm for the Simulation Studies

1. Compute the initial estimates using the Least Median Squares (LMS).

2. Obtain the corresponding residuals from our initial estimates.

3. Compute the corresponding weights based on the proposed weight function.

4. Calculate the new estimates of the regression coefficients using weighted least squares.

5. Repeat step 2 to 4 until convergence.

6. Stop when the difference between the two consecutive values is less than the error tolerance, where error tolerance is a specified value less than 10^{-n}, where n is a small positive integer.

3.3Simulation Study

Simulated data were generated (including percentage mixtures of contaminated and uncontaminated data)in simple and multiple regressions, using five sample sizes, $n =$ 20, 50, 100, 150 and 200. The percentages of outliers considered in the simulation study were as follows:

For the y- axis, we chose contamination at 10%, 20%, 30% and 40%.

For the x-axis, we chose outliers at 10%, 20% and 30%

Forboth thex- and y-axes, we chose outliers at 5%, 10%, 15% and 20%.

The choices of the distributions used and the range choices for each distribution were chosen to use the idea of Rousseeuw and Leroy (1987).

3.4 Different Scenarios of Simulated Data for Simple and Multiple Regression Analyses

3.4.1 Data without outlier

The uncontaminated data were generated from a normal distribution, $\varepsilon \sim N(0,1)$.

3.4.2 Data with outliers in the x- direction (leverage points)

- 90% of the uncontaminated x-variates were generated from a uniform distribution, $x_1 \sim U(10,20)$, $x_2 \sim U(10,20)$ and 10% of the contaminated x-variates were also generated from a uniform distribution, $x_1 \sim U(-1,1), x_2 \sim U(-2,2)$.
- 80% of the uncontaminated x-variates were generated from a uniform distribution, $x_1 \sim U(10,20)$, $x_2 \sim U(10,20)$ and 20% of the contaminated x-variates were also generated from a uniform distribution, $x_1 \sim U(-1,1), x_2 \sim U(-2,2)$.
- 70% of the uncontaminated x-variates were generated from a uniform distribution, $x_1 \sim U(10,20)$, $x_2 \sim U(10,20)$ and 30% of contaminated x-variates were also generated from a uniform distribution, $x_1 \sim U(-1,1), x_2 \sim U(-2,2)$.

3.4.3 Data with outliers at the response, that is, in the y-direction

- 90% of non-outlying data were generated from a normal distribution, $\varepsilon \sim N(0,1)$ and 10% of the outlying data were also generated from a normal distribution, $\varepsilon \sim N(10,9)$.
- 80% of non-outlying data were generated from a normal distribution, $\varepsilon \sim N(0,1)$ and 20% of the outlying data were also generated from a normal distribution, $\varepsilon \sim N(10,9)$
- 70% of non-outlying data were generated from a normal distribution, $\varepsilon \sim N(0,1)$ and 30% of the outlying data were also generated from a normal distribution, $\varepsilon \sim N(10,9)$.
- 60% of non-outlying data were generated from a normal distribution, $\varepsilon \sim N(0,1)$ and 40% of the outlying data were also generated from a normal distribution, $\varepsilon \sim N(10,9)$.

3.4.4 Data with outliers in both x and y directions

- 95% of the uncontaminated x-variates were generated using a uniform distribution,$x_1 \sim U(10,20)$,$x_2 \sim U(10,20)$ and 5% of contaminated x-variates were also generated from a uniform distribution, $x_1 \sim U(-1,1), x_2 \sim U(-2,2)$: 95% of non-outlying data in the y axis were generated from a normal distribution, $\varepsilon \sim N(0,1)$ and 5% of the outlying data in the y axis were also generated from a normal distribution,$\varepsilon \sim N(10,9)$.
- 90% of the uncontaminated x-variates were generated using a uniform distribution,$x_1 \sim U(10,20)$,$x_2 \sim U(10,20)$ and 10% of contaminated x-variates were also generated from a uniform distribution, $x_1 \sim U(-1,1), x_2 \sim U(-2,2)$: 90% of non-outlying data in the y axis were generated from a normal distribution, $\varepsilon \sim N(0,1)$ and 10% of the outlying data in the y axis were also generated from a normal distribution,$\varepsilon \sim N(10,9)$.
- 85% of the uncontaminated x-variates were generated using a uniform distribution,$x_1 \sim U(10,20)$,$x_2 \sim U(10,20)$ and 15% of contaminated x-variates were also generated from a uniform distribution, $x_1 \sim U(-1,1), x_2 \sim U(-2,2)$: 85% of non-outlying data in the y axis were generated from a normal distribution, $\varepsilon \sim N(0,1)$ and 15% of the outlying data in the y axis were also generated from a normal distribution,$\varepsilon \sim N(10,9)$.
- 80% of the uncontaminated x-variates were generated using a uniform distribution,$x_1 \sim U(10,20)$,$x_2 \sim U(10,20)$ and 20% of contaminated x-variates were also generated from a uniform distribution, $x_1 \sim U(-1,1), x_2 \sim U(-2,2)$: 80% of non-outlying data in the y axis were generated from a normal distribution, $\varepsilon \sim N(0,1)$ and

20% of the outlying data in the y axis were also generated from a normal distribution,$\varepsilon \sim N(10,9)$.

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Results and Discussions of the Simulation Study from the Different Scenarios of Data

The simulation results for the proposed estimator and that of OLS, Huber, Hampel, Bisquare (Biweight) and Alarm estimators are discussed as follows with the of values for MSE and BIAS given from n = 20 to n = 200:

4.1.1 Discussion of simulation results for data without outlier

Appendix III presents detailed simulation result for uncontaminated data from a simple regression model. The OLS having the least MSE, that is, 0.0509, 0.1526, ... ,0.0147,outperformed the Huber(MSE from 0.0547 to 0.0155), Hampel (MSE from 0.0521 to 0.0148), Bisquare(MSE from 0.0565 to 0.0156), Alarm(MSE from 0.0540 to 0.0149) and the proposed (MSE from 0.0604 to 0.0156) estimators, that is, the most efficient estimator. Similarly, the OLS, Huber, Hampel, Bisquare, Alarms and the proposed estimators are all closer to their true parameter's estimates (robustness). The values for the BIAS are as follows: OLS from 0.0033 to 0.0047, Huber estimator from 0.0020 to 0.0050, Hampel estimator from 0.0031 to 0.0052, Bisquare estimator from 0.0017 to 0.0051, Alarm estimator from 0.0018 to 0.0052 and the proposed estimator from 0.0031 to 0.0062.

Similarly, Appendix XVII presents detailed simulation result for uncontaminated datafrom a multiple regression model. The OLS having the least MSE, that is, 0.0520, 0.1537, . . . ,0.0033, out performed the Huber with MSE from 0.0556 to 0.0035, Hampel (MSE from 0.0529 to 0.0034), Bisquare (MSE from 0.0576 to 0.0035), Alarm (MSE from 0.0549 to 0.0034) and the proposed (MSE from 0.0677 to 0.0035) estimators, that is, the most efficient estimator. Similarly, the OLS, Huber, Hampel, Bisquare, Alarms and the proposed estimators are all closer to their true parameter's estimates (robustness).The values for the BIAS are as follows: OLS from 0.0100 to 0.0016, Huber estimator (from 0.0100 to 0.0007), Hampel estimator (from 0.0096 to 0.0013), Bisquare estimator (from 0.0098 to 0.0008), Alarm estimator (from 0.0092 to 0.0013) and the proposed estimator (from 0.0151 to 0.0009).

4.1.2 Discussion of simulation results for data with outliers in the x- direction (leverage points)

Appendix V presents the result when 10% of the data are outlying in the$x$- direction in a simple regression model. The MSE of Alarm (from 0.0667 to 0.0164) and the proposed (from 0.0751 to 0.0165) estimatorsare smaller compared to that of OLS (from 0.1165 to 3.3378),Huber (from 0.1303 to 3.8812), Hampel (from 0.1026 to 3.8867) and Bisquare (from 0.1348 to 3.8820) estimators. The proposed estimator also has the least BIAS that is, from 0.0121 to 0.0003 alongside the Alarms estimator (from 0.0099 to 0.0006) compared to OLS (from 0.0247 to 1.9716), Huber (from 0.0288 to 1.9701), Hampel (from 0.0282 to 1.9713) and Bisquare (from 0.0289 to 1.9701) estimators. This proves that the Alarm and proposed estimators are more efficient and robust compared to OLS, Huber, Hampel, and Bisquare estimators. In addition, outliers strongly affect the slopes of OLS, Huber, Hampel, and Bisquare estimators.

Appendix VII presents the simulated result for 20% outliers in the x-direction in a simple regression model. The Alarm estimator having smaller MSE(from 0.1052 to 0.0699)and that of the proposed estimator (MSE from 0.1120 to 0.0932) are more efficient compared to OLS (MSE from 0.1391 to 3.9370, Huber (MSE from 0.1585 to 3.9329), Hampel (MSE from 0.1436 to 3.9360) and Bisquare (MSE from 0.1625 to 3.9328) estimators.In addition, the proposed estimator with the least BIAS (from 0.0002 to 0.0391) and that of Alarm estimator (BIAS from 0.0006 to 0.0285) are more robust compared to OLS (BIAS from 0.0107 to 1.9841), Huber (BIAS from 0.0112 to 1.9831), Hampel (BIAS from 0.0103 to 1.9839) and Bisquare (BIAS from 0.0117 to 1.9831) estimators. Moreover, the parameter estimates of the slopes of OLS, Huber, Hampel, and Bisquare estimators are very high, which implies that, these estimators are strongly affected by outliers.

With the increase of the percentage of outliers in the x-direction in simple regression to 30% as shown in Appendix IX, all the estimators performed badly for both MSE and BIAS. The values for their MSE's are given by: OLS from 0.1563 to 3.9646, Huber estimator from 0.1756 to 3.9628, Hampel estimator from 0.1633 to 3.9642, Bisquare estimator from 0.1840 to 3.9628, Alarm estimator from 0.1462 to 1.3739 and the proposed estimator from 0.1653 to 1.9144,while that for the BIAS are given by: OLS from 0.0300 to 1.9911, Huber estimator from 0.0302 to 1.9906, Hampel estimator from 0.0286 to 1.9910, Bisquare estimator from 0.0288 to 1.9906, Alarm estimator from 0.0066 to 0.6651 and the proposed estimator from 0.0090 to 0.9326, For comparison, the proposed and Alarms estimators are more efficient and robust compared to OLS, Huber, Hampel, and Bisquare estimators.

Appendix XXIX presents the result for 10% outliers in the x-direction in a multiple regression model. The proposed estimator having least MSE from 0.1012 to0.0044 alongsidethe Hampel (from 0.0748 to 0.0044) and Alarm (from 0.0804 to 0.0044) estimators, are more efficient compared to OLS (from 0.1357 to 0.0090), Huber (from 0.1516 to 0.0102) and Bisquare (1.1587 to 0.0101) estimators. In addition, the proposed estimator with the least BIAS (from 0.0081 to 0.0038) and that of Alarm (from 0.0084 to 0.0039) and the Hampel (from 0.0050 to 0.0039) estimators are more robust compared to OLS (from 0.0438 to 0.0038), Huber (from 0.0468 to 0.0035) and Bisquare (from 0.0456 to 0.0035) estimators.

Appendix XXXI presents the result for 20% outliers in the x-direction in a multiple regression model. The proposed estimator having least MSE from 0.1484 to 0.0049 alongside the Hampel (from 0.1150 to 0.0049) and Alarm (from 0.1266 to 0.0049) estimators, are more efficient compared to OLS (from 0.1631 to 0.0097), Huber (from 0.1814 to 0.0106) and Bisquare (0.1900 to 0.0106) estimators. In addition, the proposed estimator with the least BIAS (from 0.0082 to 0.0846) and that of Alarm (from 0.0095 to 0.0896) and the Hampel (from 0.0078 to 0.1722) estimators are more robust compared to OLS (from 0.0412 to 2.4397), Huber (from 0.0400 to 0.0485) and Bisquare (from 0.0377 to 0.0623) estimators.

Based on the data generated from 30% outliers in x- direction in a multiple regression, shown in Appendix XXXIII, the result indicated that the Hampel estimator is the most efficient having the smallest MSE(0.1667 to 0.0071) among others while Alarm estimator is the second(MSE from 0.1809 to 0.0071).The third most efficient is the proposed estimator (MSE from 0.2113 to 0.0075) followed by the three remaining estimators (OLS, Huber, and Bisquare estimators). With regards to robustness,the proposed estimatoris the best, having the least BIAS (0.0364 to 0.0008)while the Alarm and Hampel estimators followed closely (BIAS from 0.0439 to 0.00160 and from 0.0471 to 0.0023, respectively). Furthermore, outliers strongly affect the slopesof all the estimators (the proposed, Hampel, Alarm, OLS, Huber, and Bisquare estimators). All the estimators performed badly in this category.

4.1.3 Discussion of simulation results for data with outliers at the response, that is, in the y-direction

Appendix XI presents the result for 10% outliers in the y-direction for simple regression analysis. All the estimators except, the OLS performed very well, but having the least values of MSE, the Bisquare (from 0.0622 to 0.0184), proposed (from 0.0649 to 0.0182), Hampel (from 0.0640 to 0.0196) and Alarm (from 0.0612 to 0.0185) estimators are more efficient compared to OLS (from 1.0170 to 0.1922) and Huber (from 0.0791 to 0.0204) estimators. In addition, regarding how the estimators differ from their true parameter's estimates (robustness), the proposed method, with the least value of the BIAS from 0.0079 to 0.0087, takes the lead, followed by the Bisquare (from 0.0081 to 0.0081), then, the Alarm (from 0.0138 to 0.0072) and the Hampel's (from 0.0278 to 0.0042) estimators.

Furthermore, the result from Appendix XIII (20% outliers in the y-direction for simple regression), indicates that the proposed estimator (MSE from 0.0772 to 0.0216 and BIAS from 0.0315 to0.0160) is the most efficient and robust, followed closely by the Bisquareestimator (MSE from 0.0752 to 0.0218 and BIAS from 0.0336 to 0.0173). The Alarm(MSE from 0.0819 to 0.0245 and BIAS from 0.0527 to 0.0198), Hampel (MSE from 0.1340 to 0.0259 and BIAS from 0.1299 to 0.0259)and Huber (MSE from 0.2057 to 0.0308 and BIAS from 0.3405 to 0.0483) estimators follow thereafter, while OLS(MSE from 4.1726 to 0.3063 and BIAS from 1.8531 to 0.1453) is the least.

Appendix XV presents the result for 30% outliers in the y-direction for simple regression. The proposed estimator (MSE from 0.1172 to 0.0331) competes favourably with the Bisquare estimator (MSE from 0.1452 to 0.0304) as the most efficient estimator. The least efficient is the OLS (MSE from 9.5809 to 0.4556) followed by the Huber estimator (MSE from 1.2896 to 0.0623), then, the Hampel's estimator (MSE from 3.6198 to 0.0702). Nevertheless, the proposed (BIAS from 0.0860 to 0.0216) and Bisquare (BIAS from 0.1008 to 0.0218) estimators, with least BIAS are more robust compared to the Hampel(BIAS from 1.2486 to 0.0702), Alarm(BIAS from 0.1619 to 0.0304), OLS(BIAS from 2.9036 to 0.2589) and Huber (BIAS from 0.9170 to 0.1076) estimators.

Nevertheless, in Appendix XVII(results for40% outliers in the y-direction for simple regression), the proposed (MSE from 0.4626 to 0.0934 and BIAS from 0.3102 to 0.0268) estimator is the most efficient and robust compared to theBisquare (MSE from 4.4558 to 0.0606 and BIAS from 1.2854 to 0.0241), Hampel,(MSE from 11.8804 to 0.4004 and BIAS from 2.9863 to 0.0852), Alarm(MSE from 1.14118 to 0.1229 and BIAS from 0.7094 to 0.0302), OLS(MSE from 15.8511 to 0.5570 and BIAS from 3.7895 to 0.0871) and Huber (MSE from 2.3770 to 0.2131 and BIAS from 2.3770 to 0.0731) estimators.

Appendix XXXV presents the result for 10% outliers in the y-direction for multiple regression. All the estimators except, the OLS(MSE from 1.2113 to 0.0378) performed very well, but the Bisquare (MSE from 0.0602 to 0.0038), proposed (MSE from 0.0678 to 0.0038), Hampel (MSE from 0.0615 to 0.00.0039) and Alarm (MSE from 0.0596 to 0.0037)estimators are more efficient compared to OLS(MSE from 1.2113 to 0.0378) and Huber (MSE from 0.0783 to 0.0043) estimators. In addition, regarding how the estimators differ from their true parameters estimates (robustness), the proposed method (BIAS from 0.0061 to 0.0010) takes the lead, followed by the Bisquare (BIAS from 0.0075 to 0.0006), then, the Alarm (BIAS from 0.0028 to 0.0006) and the Hampel's (BIAS from 0.0128 to 0.0013) estimators.

Furthermore, the result from Appendix XXXVII (20% outliers in the y-direction for multiple regression), indicated that the proposed estimator (MSE from 0.0758 to 0.0047 and BIAS from 0.0123 to 0.0039) is the most efficient and robust, followed closely by the Bisquare estimator (MSE from 0.0709 to 0.0047 and BIAS from 0.0149 to 0.0034). The Alarm (MSE from 0.0754 to 0.0050 and BIAS from 0.0365 to 0.0059), Hampel (MSE from 0.0985 to 0.0058 and BIAS from 0.0948 to 0.0101) and Huber (MSE from 0.1805 to 0.0063 and BIAS from 0.0.3018 to

0.0286) estimators followed thereafter, while OLS (MSE from 3.7599 to 0.0681 and BIAS from 1.7370 to 0.1287) is the least.

Appendix XXXIX presents the result for 30% outliers in the y-direction for multiple regression. The proposed estimator (MSE from 0.1200 to 0.0076) competes favourably with the Bisquare estimator (MSE from 0.4679 to 0.0068) as the most efficient estimator. The least efficient is the OLS (MSE from 8.3023 to 0.1234)) followed by the Huber estimator (MSE from 1.6545 to 0.0158), then, the Hampel's estimator (MSE from 1.2216 to 0.0129). Nevertheless, the proposed(BIAS from 0.0764 to 0.0044) and Bisquare(BIAS from 0.2159 to 0.0059) estimators are more robust compared to the Hampel (BIAS from 0.5721 to 0.0282), Alarm (BIAS from 0.1781 to 0.0119), OLS (BIAS from 2.6733 to 0.1929) and Huber (BIAS from 1.0070 to 0.0689) estimators.

In Appendix XLI (results for40% outliers in the y-direction for multiple regression), the proposed (MSE from 0.8166 to 0.0186 and BIAS from 0.4042 to 0.0340) and Bisquare (MSE from 6.6230 to 0.0148 and BIAS from 1.8229 to 0.0313) estimators are more efficient and robust compared toHampel,(MSE from 9.5266 to 0.1646 and BIAS from 2.3304 to 0.2937), Alarm (MSE from 2.3911 to 0.0297 and BIAS from 0.9315 to 0.0695), OLS (MSE from 17.3861to 0.2172 and BIAS from 3.9455 to 0.3261and Huber (MSE from 9.3040 to 0.0999 and BIAS from 2.6372 to 0.2437) estimators.

4.1.4 Discussion of simulation results for data with outliers in both x and y directions.

Appendix XIX presents the result for 5% outliers for both x and $y$axes in a simple regression model. The proposed(MSE from 0.0765 to 0.0171 and BIAS from 0.0086 to 0.0017) and Alarm estimators (MSE from 0.0716 to 0.0170 and BIAS from 0.0.0103 to 0.0021) are more efficient and robust compared to OLS(MSE from 0.5943 to 3.9175 and BIAS from 0.5147 to 1.9789), Huber(MSE from 0.1484 to 3.7981 and BIAS from 0.0793 to 1.9485), Hampel (MSE from

0.1359 to 3.7943 and BIAS from 0.0.0211 to 1.9476) and Bisquare (MSE from 0.1433 to 3.7751 and BIAS from 0.0.0131 to 1.9426) estimators. Also, the slopes of the Hampel, OLS, Huber, and Bisquare estimators are affected by the outliers.

As the outliers in both axes are increased, that is, 10% outliers for both x and y axes in a simple regression model, the result from Appendix XXI indicated that the proposed estimator (MSE from 0.0870 to 0.0242 and BIAS from 0.0130 to 0.0064) takes the lead as the most efficient and robust method but followed closely by the Alarm estimator (MSE from 0.0903 to 0.0252 and BIAS from 0.0297 to 0.0072). The Hampel, OLS, Huber, and Bisquare estimators performed badly in this category (having higher estimates of the MSE and BIAS).

With higher estimates of the MSE and BIAS, the results of Hampel(MSE from 0.2917 to 3.9939 and BIAS from 0.1965 to 1.9983), OLS(MSE from 4.1452 to 4.3904 and BIAS from 1.7716 to 2.0951), Huber(MSE from 0.3756 to 4.0417 and BIAS from 0.3922 to 2.0103) and Bisquare (MSE from 0.1939 to 3.9504 and BIAS from 0.0344 to 1.9875) estimators got worse by the increase of outliers at both axes, that is, 15% outliers for both x and y axes in a simple regression as shown in Appendix XXIII. The proposed estimator (MSE from 0.1302 to 0.0816 and BIAS from 0.0203 to 0.0270) is still the best with respect to efficiency and robustness but followed closely by the Alarm estimator (MSE from 0.1469 to 0.0649 and BIAS from 0.0539 to 0.0163).

At 20% outliers in both axes in a simple regression model as shown in Appendix XXV, the proposed (MSE from 0.2245 to 1.4156 and BIAS from 0.0571 to 0.6731) and Alarm (MSE from 0.2870 to 0.8216 and BIAS from 0.1471 to 0.3748) estimators are more efficient and robust compared to Hampel (MSE from 0.9700 to 4.0727 and BIAS from 0.5218 to 2.0180), OLS(MSE from 7.4081 to 4.6178 and BIAS from 2.4627 to 2.1487), Huber(MSE from 0.7428 to 4.1375 and BIAS from 0.6629 to 2.0340) and Bisquare (MSE from 0.2606 to 3.9885 and

BIAS from 0.0809 to 1.9970) estimators. All the estimators do not perform very well in this category with Hampel, OLS, Huber, and Bisquare estimators on the lead.

The result from Appendix XLIIII, that is, 5% outliers for both x and y axes in a multiple regression, showed that the proposed (MSE from 0.0951 to 0.0044 and BIAS from 0.0.0144 to 0.0023), Hampel (MSE from 0.0784 to 0.0046 and BIAS from 0.0254 to 0.0025), and Alarm (MSE from 0.0774 to 0.0044 and BIAS from 0.0.0189 to 0.0020)estimators are more efficient and robust compared to OLS(MSE from 0.7824 to 0.3028 and BIAS from 0.0.4827 to 0.4878),Huber(MSE from 0.1666 to 0.0198 and BIAS from 0.0654 to 0.0950) and Bisquare (MSE from 0.1660 to 0.0073 and BIAS from 0.0120 to 0.0016)estimators.

Appendix XLV presents the result for 10% outliers in both axes in a multiple regression model. The proposed (MSE from 0.1072 to 0.0049 and BIAS from 0.0.0051 to 0.0043) and Alarm (MSE from 0.1054 to 0.0052 and BIAS from 0..0131 to 0.0036) estimators are more efficient and robust compared to Hampel (MSE from 0.1096 to 0.0120 and BIAS from 0.0384 to 0.0274), OLS(MSE from 2.3390 to 1.1228 and BIAS from 1.0970 to 1.0033), Huber(MSE from 0.2547 to 0.0778 and BIAS from 0.2026 to 0.2467) and Bisquare (MSE from 0.1750 to 0.0246 and BIAS from 0.0007 to 0.0131) estimators.

Furthermore, the proposed estimator (MSE from 0.1462 to 0.0063 and BIAS from 0.0182 to 0.0020) takes the lead as the most efficient and robust estimator as shown in Appendix XLVII for 15% outliers for both x and $y$axes, in a multiple regression analysis. Alarm estimator (MSE from 0.1569 to 0.0104 and BIAS from 0.0407 to 0.0148) came second while Hampel's estimator (MSE from 0.2455 to 0.1325 and BIAS from 0.1109 to 0.2670) was the third most efficient and robust estimator. The OLS(MSE from 4.7700 to 2.3358 and BIAS from 1.6927 to 1.4771), Huber(MSEfrom 0.7943 to 0.2616 and BIAS from 0.0.4370 to 0.4797) and Bisquare(MSE from

0.5347 to 0.0294 and BIAS from 0.0680 to 0.0999) estimators were also the least efficient and robust estimators in this category.

Lastly, Appendix XLIX presents the result for 20% outliers in both axes in a multiple regression model. The proposed estimator(MSE from 0. 0.2479 to 0.0161 and BIAS from 0.0603 to 0.0161) outperformed other estimators as the most efficient and robust estimator. The second most efficient and robust estimator is the Alarm estimator(MSE from 0.4146 to 0.1884 and BIAS from 0.1465 to 0.2496) which performs better than Hampel(MSE from 1.9598 to 1.4194 and BIAS from 0.6816 to 1.0724), OLS(MSE from 8.9306 to 4.1653 and BIAS from 2.4819 to 1.9901), Huber(MSE from 2.2063 to 0.9828 and BIAS from 1.0034 to 0.9414) and Bisquare (MSE from 1.0812 to 0.1216 and BIAS from 0.3308 to 0.2784) estimators.

4.2Kruskal-Wallis Test

The Kruskal-Wallis Test was used to determine the average rank of the six estimators (OLS, Huber, Bisquare, Hampel, Alarm and the proposed estimators). The estimator with the least aveage rank is regarded as the best estimator. The detailed result for the Kruskal-Wallis Test are shown in the Appendix.

	OLS	Huber	Bisquare	Hampel	Alarm	Proposed
No Outlier	55.3	61.5	62.3	58.7	61.4	63.7
10% Outliers in x	74.7	74.9	73.7	74.8	33.6	31.3
20% Outliers in x	70.2	69.6	69.5	70.2	41.4	42.0
30% Outliers in x	67.2	66.7	64.5	66.4	47.1	52.5
10% Outliers in y	109.8	77.9	39.0	56.3	43.5	37.7
20% Outliers in y	109.0	84.3	35.4	61.1	44.0	30.7
30% Outliers in y	103.5	80.1	32.6	80.5	41.8	28.2
40% Outliers in y	86.0	74.0	46.3	80.8	48.0	30.3
5% Outliers in x and y	90.9	76.8	68.8	63.2	34.5	29.7
10% Outliers in x and y	91.8	79.1	57.4	71.3	36.6	28.9
15% Outliers in x and y	96.3	78.2	54.6	69.3	36.5	30.5

20% Outliers in x and y	102.3	71.2	49.4	63.6	40.4	38.0

Table 4.1: Summary of the Average Rank (Kruskal-Wallis Test) on Simple Regression of the Simulation Study from the Different Scenarios of Data

	OLS	**Huber**	**Bisquare**	**Hampel**	**Alarm**	**Proposed**
No Outlier	85.4	90.8	93.5	87.6	89.6	96.1
10% Outliers in x	112.0	113.2	113.3	67.9	68.9	67.6
20% Outliers in x	105.0	104.3	104.3	80.1	75.8	73.8
30% Outliers in x	96.2	97.0	97.8	83.1	81.9	87.0
10% Outliers in y	159.5	112.0	62.0	83.2	65.5	60.7
20% Outliers in y	157.8	118.8	57.9	84.9	67.9	55.8
30% Outliers in y	147.1	119.1	59.4	100.4	68.1	48.8
40% Outliers in y	121.3	106.6	72.9	108.6	77.8	55.7
5% Outliers in x and y	148.8	115.7	92.3	67.5	59.8	58.9
10% Outliers in x and y	150.7	122.1	90.5	78.8	54.9	46.0
15% Outliers in x and y	152.4	120.7	84.4	90.0	55.6	39.8

20% Outliers in x and y	155.7	113.3	76.3	106.6	60.2	31.0

Table 4.2: Summary of the Average Rank (Kruskal-Wallis Test) on Multiple Regression of the Simulation Study from the Different Scenarios of Data

The Average Rank (Kruskal-Wallis Test) on Simple and Multiple Regressionsof the Simulation Study from the Different Scenarios of Data as shown in Tables 4.1 and 4.2 indicated the following:

- In a clean data (non-outlying data), OLS having the least average rank, is the best estimator, followed by the Hampel, Alarm, Huber, Bisquare and the proposed estimators.
- When outliers are in x-axis, Alarm's estimator is the best estimator, followed closely by the proposed estimator.
- When outliers are in y-axis, the proposed method takes the lead, followed by the Bisquare and the Alarm estimators.
- When outliers are in both x and y-axes, the proposed estimator having the least average rank,is the best estimator.

4.3 Real-Life Data

We applied the proposed estimator to real-life data to verify its effectiveness in detecting and deleting of outliers. These datasets had been extensively used by other researchers in the area of robust regression.

4.3.1 Example 1: Telephone-call data (simple regression case)

This is a real regression data with a few outliers in y-direction. The data set is taken from Belgium Statistical Survey (Rousseeuw and Leroy, 1987). The data contains 24 data points and 2 variables. The dependent variable is the number of telephone calls made from Belgium (in ten of millions) and the independent variable is the year. The dataset was executed and analyzed by many researchers. The data are shown in Appendix LI.

Parameter	OLS	Huber	Hampel	Biweight	Alarm	Proposed
β_0	-260.059	-99.905	-52.389	-52.348	-52.454	-52.456
β_1	5.041	1.987	1.101	1.100	1.102	1.102
Data points used	24	24	18	17	17	17
Residual Standard Error	56.22	19.51	1.62	1.24	1.38	1.39

Table 4.3: Estimates of the Model Parameters for Telephone Calls Data

The summary of the results for estimates of the model parameters for Telephone Calls Data for the estimators are presented in Table 4.3. The Biweight, Alarm, Hampel and the proposed estimators with Residual Standard Error (RSE) of 1.24, 1.38, 1.62 and 1.39, respectively, performed betterthan OLS and Huber estimators (RSE of 56.22 and 19.51, respectively). In addition, OLS and Huber estimators used all the data in the analysis while Alarm, Biweight and the proposed method detected and deleted 7 outliers in the robust fit. The detailed results of the analysis are shown in Appendix LV.

4.3.2 Example 2: The Hawkins, Bradu, and Kass data (Multiple Regression Case)

The Hawkins et al. (1984) (Rousseeuw and Leroy, 1987) generated artificial data for testing the performance of robust estimators. The data contains **75** observations in four dimensions (one response and three explanatory variables). The first 10 observations are bad leverage points, and the next four points are good leverage points (i.e., their $\boldsymbol{x_i}$ are outlying, but the corresponding $\boldsymbol{y_i}$ fit the model quite well). The data are shown in Appendix LII.

Parameter	OLS	Huber	Hampel	Biweight	Alarm	Proposed
β_0	-0.388	-0.776	-0.181	-0.946	-0.181	-0.181
β_1	0.239	0.167	0.081	0.145	0.082	0.081
β_2	-0.335	0.007	0.040	0.197	0.040	0.040
β_3	0.383	0.274	-0.052	0.180	-0.052	-0.052
Data points used	75	75	65	71	65	65
Residual Standard Error	2.25	1.13	0.77	0.63	0.56	0.56

Table 4.4: Estimates of the model parameters for Hawkins, Bradu and Kass data

The summary of the results for estimates of the model parameters for Hawkins, Bradu and Kass data for the estimators are presented in Table 4.4. With smaller Residual Standard Error (RSE), the Alarm, Hampel, Biweight and the proposed estimators (with RSE of 0.56, 0.77, 0.63 and 0.56, respectively),performed better than OLS and Huber estimators (with RSE of 2.25 and 1.13, respectively). In addition, OLS and Huber used all the data in the analysis while Alarm, Hampel and the proposed method detected and deleted 10 outliers in the robust fit. The Biweight estimator detected and deleted 4 outliers in the analysis. The detailed results of the analysis are shown in Appendix LVI

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATION

5.1 Summary

A Redescending M-estimator was proposed and the graphs of its objective, influence and weight functions satisfied the various properties of these functions. The graph of the objective function satisfied the five properties of a good objective function of an M-estimator while the extreme outliers on the graph of the proposed influence function redescends to zero, which implied that the proposed influence function is a Redescending M-estimator. Lastly, the extreme residuals (outliers) on the graph of the proposed weight function are zero. This implied that the proposed weight function detects and deletes outliers in the robust fit.

Simulation studies were done to ascertain the effectiveness of the proposed Redescending M-estimator and for comparison with other existing methods. Simple and multiple regression analyses were considered in the simulation studies using four scenarios of data with different probability distributions / percentages of outliers in the data. Mean square error (MSE) and absolute bias (BIAS) were used for comparison under five different sample sizes.

From the simulation results andKruskal-Wallis Test, it was obvious that Ordinary least squares estimator having the least MSE and BIAS, outperformed other estimators in an uncontaminated data (clean data). Consequently, when outliers are in the leverage points, the proposed and Alarm estimators take the lead as the most efficient and robust estimators among others. On the other hand, all the estimators performed very well when outliers are in the y-direction but the proposed estimator tops the list as the most efficient and robust estimator while the Biweight, Alarm and Hampel estimators followed closely. Lastly, when outliers are both in the leverage points and the response, the proposed estimator is the most efficient and robust estimator compared to the Hampel, OLS, Huber, Alarm and Bisquare estimators.

In addition, robust regression analysis was fitted using the Telephone call data and the Hawkins, Bradu and Kass data to illustrate the ability of the proposed estimator to detect and delete outliers and to compare with the existing ones. The results from the two robust fits showed that the proposed method can successively detect and delete outliers and for comparison, the proposed estimator alongside the Alarm, Hampel and Biweight (only when outliers arse in the response direction) estimators showed great resistance to outliers.

5.2 Conclusion

The proposed Redescending estimator is the most efficient and robust method and should be used extensively when outliers are in both x-and-y axes.

5.3 Recommendations for Further Study

This work can be extended in future to handle outliers effectively on x-axis with higher percentages, that is, 30% and 40%.

5.4 Contribution to Knowledge

Based on the research, the following improvements were made on the literature:

- The influence function of the proposed M-estimator redescends to zero contrary to that of Huber (1964).
- The proposed objective function is differentiable contrary to that of Hampel et al (1974).
- The proposed Redescending M-estimator improved on the Beaton and Tukey (174) for being robust to outliers in the leverage points.
- The proposed Redescending M-estimator is more efficient and robust when outliers are in both x-and y-axes compared with Huber (1964), Hampel et al. (1986), Beaton and Tukey (1974) and Alamgir et al. (2013

REFERENCES

Aggarwal, C. and Yu, P. (2001).*Outlier Detection for High Dimensional Data.* In Proceedings of the ACM SIGMOD International Conference on Management of Data. ACM Press, 37-46.

Alamgir, A.A., Khan, S.A, Khan, D.M. and Khalil, U. (2013). *A New Efficient Redescending M-estimator, Alamgir Redesending M-estimator*. Research Journal of Recent Sciences, 2(8). 79 - 91.

Andrew, D.F., Bickel,. Hampel, F.R., Huber P.J., Tukey J.W. and Rogers W.H. (1972).*Robust Estimates of Location Survey and Advances,* Princeton University Press, Princeton, NJ.

Arya, K.V., Gupta, P., Kalra, P.K. and Mitra, P. (2007).*Image Registration using Robust M-estimate*. Journal of Mathematics and Statistics, Vol. 41(1), 147 – 155.

Atkinson, A. (1994).*Fast Very Robust Methods for the Detection of Multiple Outliers*, Journal of American Statistical Association, 89, 1329-1339.

Barnett, V. and Lewis, T. (1994). Outliers in Statistical Data. Wiley & Sons, 3rd edition.

Beaton, A.E. and Tukey, J.W. (1974).*The Fitting of Power Series, Meaning Polynomials, Illustrated on Band-Spectroscopic Data*, Technometrics, 16 (2), 147-185.

Becker, C. and Gather, U. (1999).*The Masking Breakdown Point of Multivariate Outlier Identification Rule*, Journal of the American Statistical Association, 94, 949-955.

Carling, K. (2000).*Resistant Outlier Rules and Non-Gaussian Case*, Computational Statistics and Analysis, Vol. 33, 249 – 258.

Chrominski K. M. TKACZ (2010). *Comparison of outlier detection methods in biomedial data*, Journal of Medical Informatics and Technologies, Vol. 16, ISSN 1642-6037.

Draper, N.R. and Smith H. (1998).*Applied Regression Analysis*, Third Edition, John Wiley and Sons, New York.

Fruhwirth, R. and Waltenberger W. (2010).*Redescending M-estimators and Deterministic Annealing, With Application to Robust Regression and Tail Index Estimation*, arxiv:1006:3707v1 [stat.M E].

Galimberti, G., Pillati, M. and Soffritti, G. (2007).*Robust Regreesion Trees Based On M-Estimators*, Statistica, Anno Lxvii, n.2.

Hadi, A.S. and Simonoff, J.S. (1993).*Procedures for Identification of Multiple Outliers in Linear Models*, Journal of the American Statistical Association, 88(424), 1264-1272.

Hampel, F.R. (1974). *The Influence Curve and Its Role in Robust Estimation*, Journal of the American Statistical Association, Vol. 69, No 346, pp. 383-393.

Hampel, F.R., Ronchetti, E.M., Rousseeuw, P.J. and Stahel, W.A. (1986).*Robust Statistic: the Approach Based on Influence Functions*. New York: John Wiley.

Hawkins, D.M. (1980).*The Identification of Outliers*, Chapman and Hall, London.

Hawkins, D.M., Bradu, D. and Kass, G. V. (1984).*Location of several outliers in multiple regression data using elemental sets*. Technometrics 26, 197-208.

Huber, P.J. (1964).*Robust Estimation of Location Parameter*, The Annals of Mathematical Statistics, 35, 73 – 101.

Khan, D.M., Khan, S.A., Alamgir, Khalil, U. and Ali, A. (2016).*A Comparative Study of three Improved Robust Regression Procedures*, Pak. J.Statist, Vol. 32(6), 425-441.

Manoj, k. and Kaliyaperumal, S.K. (2013).*Comparison of Methods for Detecting Outliers*, International Journal of Scientific and Engineering Research, Vol 4, Issue 9, ISSN 2229-5518.

Muller, C. H. (2004).*Redescending M-estimators in Regression Analysis*, Cluster Analysis and Image Analysis, Discussiones Mathematicae - Probability and Statistics, 24 (2004), 59-75.

Muthukrishnan, R. and Radha, M., (2010).*M-estimators in Regression Models*, Journal of Mathematics Research, Vol 2, 23-27.

Nguyena T.D. and Welch, R. (2010).*Outlier Detection and Least Trimmed Squares Approximation using Semi-definite Programming*, Comput Stat Data, 54: 3212-3226.

Perez Betsabe, Molina Isabel and Pena Daniel (2014).*Outlier detection and robust estimation in linear regression models with fixed group effects*, Journal of Statistical Computation and Simulation, 84:12,2652-2669, DOI:1080/00949655.2013.

811669.

Rousseeuw, P.J. (1982).*Least Median of Squares Regression*, Research Report No. 178, Centre for Statistics and Operations Research, VUB Brussels.

Rousseeuw, P.J. (1983).*Multivariate Estimation with High Breakdown point*, Research Report No. 192, Centre for Statistics and Operations Research, VUB Brussels.

Rousseeuw, P.J. (1984).*Least Median of Squares Regression*, Journal of American Statistical Association, 79, 871-880.

Rousseeuw, P.J. and Leroy, A.M. (1987).*Robust Regression and Outlier Detection*, Wiley-Interscience, New York.

Rousseeuw, P.J. and Yohai, V. (1984).*"Robust Regression by Means of S-Estimators", in Robust and Nonlinear Time Series*, edited by J. Franke, W. Hardle, and R.D. Martin, Lecture Notes in Statistics 26, Springer Verlag, New York, 256-274.

Sokal, R.R and Rohlf, F. J. (2012).*Biometry*, 4th Edition. W.H. Freeman and Co, New York.

Tukey, J. W. (1977).*Exploratory Data Analysis*. Reading, MA: Addison–Wesley.

Turkan, S., Cetin, M. C. and Toktamis, O. (2012).*Outlier Detection by Regression Diagnostic Based on Robust Parameter Estimate*. Journal of Mathematics and Statistic, 41(1), 147 – 155.

Wang J. C. and Opsomer J.D. (2011).*On Asymptotic Normality and Variance Estimation for Non- Differential Survey Estimators*. Biometrika (2010) pp. 1-16.

Zhang, L., Li, P., Mao, J., Ma, F., Ding, X. and Zhang, Q. (2015).*An Enhanced Monte Carlo Outlier Detection Method*, J Comput Chem. (2015) 36(25), DOI: 10.1002/jcc.24026. Epub 2015 Jul 31.

APPENDIX I

R Codes for Plotting Graphs of the Proposed Objective, Influence and Weight Functions

R code for the graph of the objective function

```
fun= function(r){

(((r^6)/(3^4))+((r^10)/(2*(3^8)))-((2*(r^6))/(3^4))+((r^2)/2)-(((2*(r^6))*((3*(r^4))-
(5*(^4))))/(15*(3^8))))

 }

 plot(fun)

 plot(fun, -3, 3)
```

R code for the graph of the influence function

```
fun= function(r){
```

```
(r*(1+((r/3)^2)^2)*(1-((r/3)^2)^2))

 }

 plot(fun)

 plot(fun, -3, 3)
```

R code for the graph of the weight function

```
fun= function(r){

((1+((r/3)^2)^2)*(1-((r/3)^2)^2))

}

 plot(fun)

 plot(fun, -3, 3)
```

APPENDIX II

R PROGRAM FOR CALCULATING THE MSE AND BIAS OF M-ESTIMATORS

```
# Weight equal to zero is used to trim large residuals observations.
sink("Stella1Case1.1Results.txt") #   Write   output   in   the   file
"Stella1Case1.2Results.txt" inside my document
SampleSize<-20
Error<-"100% e~(0,1), No Outlier"
Errorlabel<-"Error Distribution";Errorlabel;Error
SampleSizelabel<-"Sample Size";SampleSizelabel;SampleSize
print("------------------------------------------------------------")
set.seed(13)                 # Set the random number generator starting
point to enable regeneration of the same sequence of random number
M<-1000                      # Monte Carlo Replication
n<-20                  # Sample Size
YErrorMean<-0
YErrorStd<-1
Min<--1
Max<-1
```

```
a <- 1                                # True value for the intercept
b <- 2                 # True value for the slope
BetaLSE.0 <- numeric(M)      # Empty vector for storing the simulated intercepts
BetaLSE.1 <- numeric(M)      # Empty vector for storing the simulated slopes
BetaHuberM.0 <- numeric(M)   # Empty vector for storing the simulated intercepts
BetaHuberM.1 <- numeric(M)   # Empty vector for storing the simulated slopes
BetaBisquareM.0 <- numeric(M)      # Empty vector for storing the simulated intercepts
BetaBisquareM.1 <- numeric(M)      # Empty vector for storing the simulated slopes
BetaHampelM.0 <- numeric(M)  # Empty vector for storing the simulated intercepts
BetaHampelM.1 <- numeric(M)  # Empty vector for storing the simulated slopes
BetaAlamgirM.0 <- numeric(M)       # Empty vector for storing the simulated intercepts
BetaAlamgirM.1 <- numeric(M)       # Empty vector for storing the simulated slopes
BetaStellaM.0 <- numeric(M)  # Empty vector for storing the simulated intercepts
BetaStellaM.1 <- numeric(M)  # Empty vector for storing the simulated slopes
BetaLSEAbsDev.0 <- numeric(M)      # Empty vector for storing the simulated intercepts
BetaLSEAbsDev.1 <- numeric(M)      # Empty vector for storing the simulated slopes
BetaHuberMAbsDev.0 <- numeric(M)  # Empty vector for storing the simulated intercepts
BetaHuberMAbsDev.1 <- numeric(M)  # Empty vector for storing the simulated slopes
BetaBisquareMAbsDev.0 <- numeric(M)     # Empty vector for storing the simulated intercepts
BetaBisquareMAbsDev.1 <- numeric(M)     # Empty vector for storing the simulated slopes
BetaHampelMAbsDev.0 <- numeric(M) # Empty vector for storing the simulated intercepts
BetaHampelMAbsDev.1 <- numeric(M) # Empty vector for storing the simulated slopes
BetaAlamgirMAbsDev.0 <- numeric(M)      # Empty vector for storing the simulated intercepts
BetaAlamgirMAbsDev.1 <- numeric(M)      # Empty vector for storing the simulated slopes
BetaStellaMAbsDev.0 <- numeric(M) # Empty vector for storing the simulated intercepts
BetaStellaMAbsDev.1 <- numeric(M) # Empty vector for storing the simulated slopes
library(MASS)
library(robustbase)

AlamgirM<-function(Y,X){ # ALAMGIR using LTS as initial estimate
library(MASS)
library(robustbase)
n <- length(Y)
w <- rep(1,n)
```

```
irwls.1 <- ltsReg(x=X,y=Y)
res1 <- residuals(irwls.1)
b.old <- coef(irwls.1)
MAD <- mad(res1)
u <- res1/MAD
for(i in 1:n){
  if(abs(u[i])< 4.685){
      w[i]<-(16*exp(-2*(u[i]/3)^2))/(1+exp(-(u[i]/3)^2))^4
  }else{
      w[i]<-0
  }
}
delta_b <- 100.0
num.iter=0
while (delta_b > 0.000001) {
num.iter <- num.iter + 1
irwls.2 <- lm(Y ~ X,weights=w)
res2 <- residuals(irwls.2)
b.new <- coef(irwls.2)
MAD <- mad(res2)
u <- res2/MAD
for(i in 1:n){
  if(abs(u[i])< 4.685){
      w[i]<-(16*exp(-2*(u[i]/3)^2))/(1+exp(-(u[i]/3)^2))^4
  }else{
      w[i]<-0
  }
}
delta_b <- max(abs((b.new-b.old)/b.old))
b.old <- b.new
if(num.iter>100){break}
}
Out<-list(num.iter=num.iter,w=w,irwls.2=irwls.2)
}

StellaM<-function(Y,X){
############### Improved brute-force IRWLS - Stella Method
library(MASS)
library(robustbase)
n <- length(Y)
w <- rep(1,n)
irwls.1 <- lmsreg(x=X,y=Y)
res1 <- residuals(irwls.1)
b.old <- coef(irwls.1)
MAD <- mad(res1)
u <- res1/MAD
for(i in 1:n){
  if(abs(u[i])< 3){
      w[i]<-((1+((u[i]/3)^2)^2)*(1-((u[i]/3)^2)^2))
  }else{
      w[i]<-0
  }
}
delta_b <- 100.0
num.iter=0
while (delta_b > 0.000001) {
num.iter <- num.iter + 1
irwls.2 <- lm(Y ~ X,weights=w)
```

```
res2 <- residuals(irwls.2)
b.new <- coef(irwls.2)
MAD <- mad(res2)
u <- res2/MAD
for(i in 1:n){
  if(abs(u[i])< 3){
      w[i]<-((1+((u[i]/3)^2)^2)*(1-((u[i]/3)^2)^2))
  }else{
      w[i]<-0
  }
}
delta_b <- max(abs((b.new-b.old)/b.old))
b.old <- b.new
if(num.iter>100){break}
}
Out<-list(num.iter=num.iter,w=w,irwls.2=irwls.2)
}

X <- runif(n, Min, Max)      # Create a sample of n uniform observations
on the variable X.
                       # should be fixed in repeated samples.
for(m in 1:M){                  # Start the loop
Y <- a + b*X + rnorm(n, YErrorMean, YErrorStd)     # The true DGP, with
N(0, 1) error

HuberM<-rlm(Y~X, psi = psi.huber, init = "ls",maxit=100)
BetaHuberM.0[m] <- HuberM$coef[1] # Put the estimate for the intercept
in the vector BetaLSE.0
BetaHuberM.1[m] <- HuberM$coef[2] # Put the estimate for X in the
vector BetaLSE.1
BetaHuberMAbsDev.0[m]<-abs(a-BetaHuberM.0[m])
BetaHuberMAbsDev.1[m]<-abs(b-BetaHuberM.1[m])

BisquareM<-rlm(Y~X, psi = psi.bisquare,maxit=100)
BetaBisquareM.0[m] <- BisquareM$coef[1] # Put the estimate for the
intercept in the vector BetaLSE.0
BetaBisquareM.1[m] <- BisquareM$coef[2] # Put the estimate for X in the
vector BetaLSE.1
BetaBisquareMAbsDev.0[m]<-abs(a-BetaBisquareM.0[m])
BetaBisquareMAbsDev.1[m]<-abs(b-BetaBisquareM.1[m])

HampelM<-rlm(Y~X, psi = psi.hampel,maxit=100)
BetaHampelM.0[m] <- HampelM$coef[1] # Put the estimate for the
intercept in the vector BetaLSE.0
BetaHampelM.1[m] <- HampelM$coef[2] # Put the estimate for X in the
vector BetaLSE.1
BetaHampelMAbsDev.0[m]<-abs(a-BetaHampelM.0[m])
BetaHampelMAbsDev.1[m]<-abs(b-BetaHampelM.1[m])

AlamgirMModel<-AlamgirM(Y,X)
AlamgirModel<-AlamgirMModel$irwls.2
BetaAlamgirM.0[m] <- AlamgirModel$coef[1] # Put the estimate for the
intercept in the vector BetaLSE.0
BetaAlamgirM.1[m] <- AlamgirModel$coef[2] # Put the estimate for X in
the vector BetaLSE.1
BetaAlamgirMAbsDev.0[m]<-abs(a-BetaAlamgirM.0[m])
BetaAlamgirMAbsDev.1[m]<-abs(b-BetaAlamgirM.1[m])
```

```
StellaMModel<-StellaM(Y,X)
StellaModel<-StellaMModel$irwls.2
BetaStellaM.0[m] <- StellaModel$coef[1] # Put the estimate for the intercept in the vector BetaLSE.0
BetaStellaM.1[m] <- StellaModel$coef[2] # Put the estimate for X in the vector BetaLSE.1
BetaStellaMAbsDev.0[m]<-abs(a-BetaStellaM.0[m])
BetaStellaMAbsDev.1[m]<-abs(b-BetaStellaM.1[m])

LSEmodel<-lm(formula=Y~X)
BetaLSE.0[m]<-LSEmodel$coef[1] # Put the estimate for the intercept in the vector BetaLTS.0
BetaLSE.1[m]<-LSEmodel$coef[2] # Put the estimate for the Slope1 in the vector BetaLTS.1
BetaLSEAbsDev.0[m]<-abs(a-BetaLSE.0[m])
BetaLSEAbsDev.1[m]<-abs(b-BetaLSE.1[m])
}

LSE.B0<-mean(BetaLSE.0)
LSE.B1<-mean(BetaLSE.1)
LSEMed.B0<-median(BetaLSE.0)
LSEMed.B1<-median(BetaLSE.1)
LSEMSE.B0<-mean((BetaLSEAbsDev.0)^2)
LSEMSE.B1<-mean((BetaLSEAbsDev.1)^2)
LSEMAE.B0<-mean(BetaLSEAbsDev.0)
LSEMAE.B1<-mean(BetaLSEAbsDev.1)
LSEMedAE.B0<-median(BetaLSEAbsDev.0)
LSEMedAE.B1<-median(BetaLSEAbsDev.1)
HuberM.B0<-mean(BetaHuberM.0)
HuberM.B1<-mean(BetaHuberM.1)
HuberMMed.B0<-median(BetaHuberM.0)
HuberMMed.B1<-median(BetaHuberM.1)
HuberMMSE.B0<-mean((BetaHuberMAbsDev.0)^2)
HuberMMSE.B1<-mean((BetaHuberMAbsDev.1)^2)
HuberMMAE.B0<-mean(BetaHuberMAbsDev.0)
HuberMMAE.B1<-mean(BetaHuberMAbsDev.1)
HuberMMedAE.B0<-median(BetaHuberMAbsDev.0)
HuberMMedAE.B1<-median(BetaHuberMAbsDev.1)
BisquareM.B0<-mean(BetaBisquareM.0)
BisquareM.B1<-mean(BetaBisquareM.1)
BisquareMMed.B0<-median(BetaBisquareM.0)
BisquareMMed.B1<-median(BetaBisquareM.1)
BisquareMMSE.B0<-mean((BetaBisquareMAbsDev.0)^2)
BisquareMMSE.B1<-mean((BetaBisquareMAbsDev.1)^2)
BisquareMMAE.B0<-mean(BetaBisquareMAbsDev.0)
BisquareMMAE.B1<-mean(BetaBisquareMAbsDev.1)
BisquareMMedAE.B0<-median(BetaBisquareMAbsDev.0)
BisquareMMedAE.B1<-median(BetaBisquareMAbsDev.1)
HampelM.B0<-mean(BetaHampelM.0)
HampelM.B1<-mean(BetaHampelM.1)
HampelMMed.B0<-median(BetaHampelM.0)
HampelMMed.B1<-median(BetaHampelM.1)
HampelMMSE.B0<-mean((BetaHampelMAbsDev.0)^2)
HampelMMSE.B1<-mean((BetaHampelMAbsDev.1)^2)
HampelMMAE.B0<-mean(BetaHampelMAbsDev.0)
HampelMMAE.B1<-mean(BetaHampelMAbsDev.1)
HampelMMedAE.B0<-median(BetaHampelMAbsDev.0)
HampelMMedAE.B1<-median(BetaHampelMAbsDev.1)
```

```
AlamgirM.B0<-mean(BetaAlamgirM.0)
AlamgirM.B1<-mean(BetaAlamgirM.1)
AlamgirMMed.B0<-median(BetaAlamgirM.0)
AlamgirMMed.B1<-median(BetaAlamgirM.1)
AlamgirMMSE.B0<-mean((BetaAlamgirMAbsDev.0)^2)
AlamgirMMSE.B1<-mean((BetaAlamgirMAbsDev.1)^2)
AlamgirMMAE.B0<-mean(BetaAlamgirMAbsDev.0)
AlamgirMMAE.B1<-mean(BetaAlamgirMAbsDev.1)
AlamgirMMedAE.B0<-median(BetaAlamgirMAbsDev.0)
AlamgirMMedAE.B1<-median(BetaAlamgirMAbsDev.1)
StellaM.B0<-mean(BetaStellaM.0)
StellaM.B1<-mean(BetaStellaM.1)
StellaMMed.B0<-median(BetaStellaM.0)
StellaMMed.B1<-median(BetaStellaM.1)
StellaMMSE.B0<-mean((BetaStellaMAbsDev.0)^2)
StellaMMSE.B1<-mean((BetaStellaMAbsDev.1)^2)
StellaMMAE.B0<-mean(BetaStellaMAbsDev.0)
StellaMMAE.B1<-mean(BetaStellaMAbsDev.1)
StellaMMedAE.B0<-median(BetaStellaMAbsDev.0)
StellaMMedAE.B1<-median(BetaStellaMAbsDev.1)
VecBiasB0<-c(abs(1-LSE.B0),abs(1-HuberM.B0),  abs(1-BisquareM.B0),
      abs(1-HampelM.B0),     abs(1-AlamgirM.B0),    abs(1-StellaM.B0))
BiasLocB0<-sort.int(VecBiasB0,index.return=TRUE)
RobustB0<-BiasLocB0$ix[1]
RobustB0
VecBiasB1<-c(abs(2-LSE.B1),abs(2-HuberM.B1),  abs(2-BisquareM.B1),
      abs(2-HampelM.B1), abs(2-AlamgirM.B1), abs(2-StellaM.B1))
BiasLocB1<-sort.int(VecBiasB1,index.return=TRUE)
RobustB1<-BiasLocB1$ix[1]
RobustB1
VecMSEB0<-c(LSEMSE.B0, HuberMMSE.B0,    BisquareMMSE.B0, HampelMMSE.B0,
AlamgirMMSE.B0,  StellaMMSE.B0)
MSELocB0<-sort.int(VecMSEB0,index.return=TRUE)
EfficiencyB0<-MSELocB0$ix[1]
EfficiencyB0
VecMSEB1<-c(LSEMSE.B1, HuberMMSE.B1,    BisquareMMSE.B1, HampelMMSE.B1,
AlamgirMMSE.B1,  StellaMMSE.B1)
MSELocB1<-sort.int(VecMSEB1,index.return=TRUE)
EfficiencyB1<-MSELocB1$ix[1]
EfficiencyB1
LADVec1<-c("OLS","Huber","Bisquare","Hampel","Alamgir","Stella")
as.table(matrix(c(LADVec1[RobustB0],              LADVec1[EfficiencyB0],
LADVec1[RobustB1], LADVec1[EfficiencyB1]), nrow=2, byrow=TRUE,
dimnames=list(Beta= c("Beta0", "Beta1"),Criteria = c("Bias", "MSE"))))
Vec.Bias  <-  c(abs(1-LSE.B0),  abs(1-HuberM.B0),  abs(1-BisquareM.B0),
abs(1-HampelM.B0),   abs(1-AlamgirM.B0),   abs(1-StellaM.B0),   abs(2-
LSE.B1),   abs(2-HuberM.B1),   abs(2-BisquareM.B1),   abs(2-HampelM.B1),
abs(2-AlamgirM.B1), abs(2-StellaM.B1))
VecBias<- round(Vec.Bias,4)
as.table(matrix(VecBias, nrow=2, byrow=TRUE,
dimnames=list(Beta=       c("Beta0",        "Beta1"),Estimator        =
c("OLS","Huber","Bisquare","Hampel","Alamgir","Stella"))))
Vec.MSE <- c(LSEMSE.B0,      HuberMMSE.B0,    BisquareMMSE.B0,
HampelMMSE.B0,   AlamgirMMSE.B0,  StellaMMSE.B0,   LSEMSE.B1,
      HuberMMSE.B1,    BisquareMMSE.B1, HampelMMSE.B1,   AlamgirMMSE.B1,
      StellaMMSE.B1)
VecMSE<- round(Vec.MSE,4)
as.table(matrix(VecMSE, nrow=2, byrow=TRUE,
```

```
dimnames=list(Beta=        c("Beta0",        "Beta1"),Estimator        =
c("OLS","Huber","Bisquare","Hampel","Alamgir","Stella"))))
MatB0 <- cbind(BetaLSE.0,    BetaHuberM.0,    BetaBisquareM.0,
BetaHampelM.0,   BetaAlamgirM.0,  BetaStellaM.0)
MatB1 <- cbind(BetaLSE.1,    BetaHuberM.1,    BetaBisquareM.1,
BetaHampelM.1,   BetaAlamgirM.1,  BetaStellaM.1)
print(round(MatB0,4))
print(round(MatB1,4))
plot.new()
plot(X,Y,pch=20)
abline(a=LSE.B0,b=LSE.B1,lty=1)
abline(a=HuberM.B0,b=HuberM.B1,lty=2)
abline(a=BisquareM.B0,b=BisquareM.B1,lty=3)
abline(a=HampelM.B0,b=HampelM.B1,lty=4)
abline(a=AlamgirM.B0,b=AlamgirM.B1,lty=5)
abline(a=StellaM.B0,b=StellaM.B1,lty=6)
legend("bottomright",c("OLS","Huber","Bisquare","Hampel","Alamgir","Ste
lla"),lty=c(1,2,3,4,5,6))
Vec1<-c(abs(1-LSE.B0),abs(1-HuberM.B0), abs(1-BisquareM.B0),   abs(1-
HampelM.B0),      abs(1-AlamgirM.B0),    abs(1-StellaM.B0),
      LSEMSE.B0, HuberMMSE.B0,    BisquareMMSE.B0, HampelMMSE.B0,
AlamgirMMSE.B0,  StellaMMSE.B0)
Vec2<-c(abs(2-LSE.B1),abs(2-HuberM.B1), abs(2-BisquareM.B1),   abs(2-
HampelM.B1), abs(2-AlamgirM.B1), abs(2-StellaM.B1),LSEMSE.B1,
      HuberMMSE.B1,    BisquareMMSE.B1, HampelMMSE.B1,   AlamgirMMSE.B1,
      StellaMMSE.B1)
Reg1B01.1<-
matrix(Vec1,nrow=6,ncol=2,dimnames=list(c("OLS","Huber","Bisquare","Ham
pel","Alamgir","Stella"), c("Bias","MSE")))
Reg1B11.1<-
matrix(Vec2,nrow=6,ncol=2,dimnames=list(c("OLS","Huber","Bisquare","Ham
pel","Alamgir","Stella"), c("Bias","MSE")))
Ymax<-max(c(Vec1,Vec2))
Ylim<-Ymax
win.graph(width=10,height=5)
plot.new()
par(mfrow=c(1,2),ps=8)
barplot(Reg1B01.1,beside=TRUE,legend=TRUE,ylim=c(0,Ylim),main="Intercep
t, n = 20, No Outlier")
barplot(Reg1B11.1,beside=TRUE,legend=FALSE,ylim=c(0,Ylim),main="Predict
or, n = 20, No Outlier")
sink()
```

APPENDIX III

Simulated MSE and BIAS on Simple Regression for data with no outlier

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED
20	β_0	BIAS	0.0033	0.0020	0.0017	0.0031	0.0018	0.0031
	β_0	MSE	0.0509	0.0547	0.0565	0.0521	0.0540	0.0604
20	β_1	BIAS	0.0071	0.0063	0.0060	0.0072	0.0086	0.0173
	β_1	MSE	0.1526	0.1586	0.1630	0.1542	0.1585	0.1866
50	β_0	BIAS	0.0027	0.0039	0.0039	0.0032	0.0034	0.0035
	β_0	MSE	0.0020	0.0210	0.0213	0.0202	0.0204	0.0217
50	β_1	BIAS	0.0033	0.0016	0.0017	0.0024	0.0024	0.0004
	β_1	MSE	0.0665	0.0695	0.0700	0.0667	0.0672	0.0714
100	β_0	BIAS	0.0030	0.0034	0.0036	0.0031	0.0033	0.0035
	β_0	MSE	0.0097	0.0101	0.0102	0.0098	0.0098	0.0102
100	β_1	BIAS	0.0013	0.0023	0.0026	0.0017	0.0019	0.0024
	β_1	MSE	0.0292	0.0312	0.0313	0.0296	0.0298	0.0314

150	$\boldsymbol{\beta_0}$	**BIAS**	0.0015	0.0016	0.0016	0.0015	0.0014	0.0012
	$\boldsymbol{\beta_0}$	**MSE**	0.0070	0.0074	0.0074	0.0071	0.0071	0.0074
150	$\boldsymbol{\beta_1}$	**BIAS**	0.0008	0.0005	0.0000	0.0007	0.0006	0.0000
	$\boldsymbol{\beta_1}$	**MSE**	0.0193	0.0209	0.0210	0.0197	0.0198	0.0212
200	$\boldsymbol{\beta_0}$	**BIAS**	0.0027	0.0036	0.0038	0.0032	0.0033	0.0040
	$\boldsymbol{\beta_0}$	**MSE**	0.0049	0.0051	0.0051	0.0049	0.0049	0.0052
200	$\boldsymbol{\beta_1}$	**BIAS**	0.0047	0.0050	0.0051	0.0052	0.0052	0.0062
	$\boldsymbol{\beta_1}$	**MSE**	0.0147	0.0155	0.0156	0.0148	0.0149	0.0156

APPENDIX IV

Kruskal-Wallis Teston Simple Regression for data with no outlier: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment     N    Median  Ave Rank      Z
1            20  0.004800      55.3  -0.74
2            20  0.005700      61.5   0.14
3            20  0.005550      62.3   0.25
4            19  0.005200      58.7  -0.24
5            21  0.007100      61.4   0.13
6            20  0.006800      63.7   0.45
Overall     120                60.5

H = 0.76  DF = 5  P = 0.980
H = 0.76  DF = 5  P = 0.980  (adjusted for ties)
```

Where
Treatment 1 is OLS estimator
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX V

Simulated MSE and BIAS on Simple Regression for 10% outliers in x-axis

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED
20	β_0	BIAS	0.0274	0.0288	0.0289	0.0282	0.0099	0.0121
	β_0	MSE	0.1165	0.1303	0.1348	0.1206	0.0667	0.0751
20	β_1	BIAS	1.9679	1.9685	1.9674	1.9682	0.2303	0.1894
	β_1	MSE	3.8786	3.8812	3.8771	3.8799	0.7187	0.6571
50	β_0	BIAS	0.0336	0.0364	0.0363	0.0344	0.0053	0.0042
	β_0	MSE	0.0506	0.0570	0.0572	0.0522	0.0252	0.0248
50	β_1	BIAS	1.9704	1.9687	1.9685	1.9700	0.0737	0.0350
	β_1	MSE	3.8845	3.8783	3.8775	3.8831	0.2547	0.1765
100	β_0	BIAS	0.0421	0.0441	0.0441	0.0427	0.0009	0.0008
	β_0	MSE	0.0306	0.0337	0.0336	0.0313	0.0119	0.0121
100	β_1	BIAS	1.9734	1.9721	1.9719	1.9730	0.0131	0.0062
	β_1	MSE	3.8953	3.8903	3.8895	3.8939	0.0647	0.0495

150	$\boldsymbol{\beta_0}$	**BIAS**	0.0377	0.0406	0.0403	0.0386	0.0012	0.0015
	$\boldsymbol{\beta_0}$	**MSE**	0.0171	0.0189	0.0188	0.0175	0.0071	0.0071
150	$\boldsymbol{\beta_1}$	**BIAS**	1.9724	1.9706	1.9706	1.9720	0.0027	0.0007
	$\boldsymbol{\beta_1}$	**MSE**	3.8911	3.8842	3.8842	3.8896	0.0291	0.0220
200	$\boldsymbol{\beta_0}$	**BIAS**	0.0399	0.0426	0.0426	0.0404	0.0038	0.0039
	$\boldsymbol{\beta_0}$	**MSE**	0.0141	0.0156	0.0156	0.0143	0.0056	0.0057
200	$\boldsymbol{\beta_1}$	**BIAS**	1.9716	1.9701	1.9701	1.9713	0.0006	0.0003
	$\boldsymbol{\beta_1}$	**MSE**	3.8878	3.8882	3.8820	3.8867	0.0164	0.0165

APPENDIX VI

Kruskal-Wallis Test on Simple Regression for 10% outliers in x-axis: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment     N   Median  Ave Rank      Z
1            20  1.04220      74.7   2.00
2            20  1.04940      74.9   2.02
3            20  1.05110      73.7   1.85
4            20  1.04440      74.8   2.02
5            20  0.01250      33.6  -3.79
6            20  0.01210      31.3  -4.11
Overall     120               60.5

H = 39.07  DF = 5  P = 0.000
H = 39.08  DF = 5  P = 0.000  (adjusted for ties)
```

Where
Treatment 1 is OLS estimator
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX VII

Simulated MSE and BIAS on Simple Regression for 20% outliers in x-axis

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED
20	β_0	BIAS	0.0107	0.0112	0.0117	0.0103	0.0006	0.0002
	β_0	MSE	0.1391	0.1585	0.1625	0.1436	0.1052	0.1120
20	β_1	BIAS	1.9871	1.9867	1.9860	1.9869	0.0702	0.6573
	β_1	MSE	3.9516	3.9504	3.9479	3.9511	1.6876	1.5633
50	β_0	BIAS	0.0295	0.0306	0.0313	0.0296	0.0069	0.0086
	β_0	MSE	0.0550	0.0618	0.0619	0.0564	0.0326	0.0350
50	β_1	BIAS	1.9863	1.9853	1.9853	1.9860	0.4022	0.3963
	β_1	MSE	3.9465	3.9427	3.9427	3.9455	0.8831	0.8738
100	β_0	BIAS	0.0381	0.0416	0.0414	0.0389	0.0022	0.0006
	β_0	MSE	0.0322	0.0359	0.0357	0.0329	0.0151	0.0149
100	β_1	BIAS	1.9858	1.9850	1.9849	1.9856	0.1660	0.1108
	β_1	MSE	3.9441	3.9047	3.9405	3.9433	0.3962	0.2835

150	$\boldsymbol{\beta_0}$	**BIAS**	0.0445	0.0477	0.0475	0.0450	0.0027	0.0020
	$\boldsymbol{\beta_0}$	**MSE**	0.0218	0.0238	0.0237	0.0221	0.0098	0.0099
150	$\boldsymbol{\beta_1}$	**BIAS**	1.9853	1.9845	1.9845	1.9851	0.0999	0.0680
	$\boldsymbol{\beta_1}$	**MSE**	3.9417	3.9388	3.9387	3.9410	0.2404	0.1783
200	$\boldsymbol{\beta_0}$	**BIAS**	0.0377	0.0403	0.0400	0.0382	0.0024	0.0021
	$\boldsymbol{\beta_0}$	**MSE**	0.0150	0.0169	0.0168	0.0153	0.0061	0.0062
200	$\boldsymbol{\beta_1}$	**BIAS**	1.9841	1.9831	1.9831	1.9839	0.0285	0.0391
	$\boldsymbol{\beta_1}$	**MSE**	3.9370	3.9329	3.9328	3.9360	0.0699	0.0932

APPENDIX VIII

Kruskal-Wallis Test on Simple Regression for 20% outliers in x-axis: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment     N   Median  Ave Rank      Z
1            20  1.06160      70.2   1.37
2            20  1.07080      69.6   1.29
3            20  1.07280      69.5   1.27
4            20  1.06375      70.2   1.37
5            20  0.05125      41.4  -2.69
6            20  0.05355      42.0  -2.60
Overall     120               60.5

H = 17.54  DF = 5  P = 0.004
```

```
H = 17.54  DF = 5  P = 0.004  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX IX

Simulated MSE and BIAS on Simple Regression for 30% outliers in x-axis

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED
20	$\boldsymbol{\beta_0}$	BIAS	0.0300	0.0302	0.0288	0.0286	0.0066	0.0090
	$\boldsymbol{\beta_0}$	MSE	0.1563	0.1756	0.1840	0.1633	0.1462	0.1653
20	$\boldsymbol{\beta_1}$	BIAS	1.9899	1.9898	1.9896	1.9899	1.2578	1.2324
	$\boldsymbol{\beta_1}$	MSE	3.9620	3.9619	3.9613	3.9621	2.7002	2.6389
50	$\boldsymbol{\beta_0}$	BIAS	0.0374	0.0382	0.0385	0.0379	0.0225	0.0244
	$\boldsymbol{\beta_0}$	MSE	0.0656	0.0721	0.0723	0.0674	0.0529	0.0566
50	$\boldsymbol{\beta_1}$	BIAS	1.9902	1.9897	1.9895	1.9900	1.0633	1.1104
	$\boldsymbol{\beta_1}$	MSE	3.9619	3.9600	3.9594	3.9612	2.2442	2.3366
100	$\boldsymbol{\beta_0}$	BIAS	0.0374	0.0408	0.0407	0.0383	0.0184	0.0229
	$\boldsymbol{\beta_0}$	MSE	0.0334	0.0364	0.0365	0.0341	0.0252	0.0272
100	$\boldsymbol{\beta_1}$	BIAS	1.9912	1.9906	1.9906	1.9910	0.9588	1.0792

	β_1	**MSE**	3.9652	3.9630	3.9630	3.9647	1.9792	2.2256
150	β_0	**BIAS**	0.0366	0.0386	0.0390	0.0371	0.0142	0.0167
	β_0	**MSE**	0.0210	0.0234	0.0233	0.0214	0.0143	0.0158
150	β_1	**BIAS**	1.9909	1.9904	1.9904	1.9908	0.7560	0.9698
	β_1	**MSE**	3.9642	3.9621	3.9618	3.9635	1.5629	1.9978
200	β_0	**BIAS**	0.0334	0.0353	0.0354	0.0339	0.0102	0.0150
	β_0	**MSE**	0.0168	0.0189	0.0188	0.0172	0.0140	0.0122
200	β_1	**BIAS**	1.9911	1.9906	1.9906	1.9910	0.6651	0.9326
	β_1	**MSE**	3.9646	3.9628	3.9628	3.9642	1.3739	1.9144

APPENDIX X

Kruskal-Wallis Test on Simple Regression for 30% outliers in x-axis: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment     N  Median  Ave Rank      Z
1            20  1.0731      67.2   0.94
2            20  1.0827      66.7   0.87
3            21  0.1840      64.5   0.59
4            18  1.0766      66.4   0.78
5            22  0.4057      47.1  -2.00
6            19  0.9326      52.5  -1.10
Overall     120              60.5
```

```
H = 6.46  DF = 5  P = 0.264
H = 6.46  DF = 5  P = 0.264  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XI

Simulated MSE and BIAS on Simple Regression for 10% Outliers in y-axis

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED
20	β_0	BIAS	0.8322	0.1233	0.0081	0.0278	0.0138	0.0079
	β_0	MSE	1.0170	0.0791	0.0622	0.0640	0.0612	0.0649
20	β_1	BIAS	1.2935	0.1869	0.0069	0.0391	0.0113	0.0012
	β_1	MSE	2.6604	0.2183	0.1743	0.1844	0.1773	0.1884
50	β_0	BIAS	0.9913	0.1429	0.0134	0.0346	0.0175	0.0117
	β_0	MSE	1.1653	0.0453	0.0234	0.0251	0.0234	0.0234
50	β_1	BIAS	0.2903	0.0504	0.0079	0.0153	0.0087	0.0064
	β_1	MSE	0.6970	0.0892	0.0811	0.0877	0.0832	0.0801
100	β_0	BIAS	1.0240	0.1455	0.0109	0.0332	0.0160	0.0086
	β_0	MSE	1.1340	0.0340	0.0116	0.0137	0.0119	0.0114
100	β_1	BIAS	0.5421	0.0815	0.0005	0.0128	0.0038	0.0002

	$\boldsymbol{\beta_1}$	**MSE**	0.6941	0.0491	0.0395	0.0426	0.0398	0.0394
150	$\boldsymbol{\beta_0}$	**BIAS**	0.9985	0.1399	0.0079	0.0308	0.0132	0.0056
	$\boldsymbol{\beta_0}$	**MSE**	1.0579	0.0287	0.0083	0.0097	0.0084	0.0081
150	$\boldsymbol{\beta_1}$	**BIAS**	0.0899	0.0122	0.0012	0.0006	0.0006	0.0001
	$\boldsymbol{\beta_1}$	**MSE**	0.1905	0.0263	0.0240	0.0252	0.0238	0.0235
200	$\boldsymbol{\beta_0}$	**BIAS**	1.0028	0.1447	0.0112	0.0339	0.0157	0.0086
	$\boldsymbol{\beta_0}$	**MSE**	1.0469	0.0274	0.0061	0.0076	0.0064	0.0061
200	$\boldsymbol{\beta_1}$	**BIAS**	0.1924	0.0196	0.0081	0.0042	0.0072	0.0087
	$\boldsymbol{\beta_1}$	**MSE**	0.1922	0.0204	0.0184	0.0196	0.0185	0.0182

APPENDIX XII

Kruskal-Wallis Test on Simple Regression for 10% Outliers in y-axis: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment     N    Median  Ave Rank      Z
1            20  0.994900     109.8   6.94
2            20  0.064750      77.9   2.45
3            21  0.011200      39.0  -3.11
4            18  0.029300      56.3  -0.56
5            22  0.014750      43.5  -2.54
6            19  0.008700      37.7  -3.12
Overall     120                60.5

H = 66.88  DF = 5  P = 0.000
```

```
H = 66.88  DF = 5  P = 0.000  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XIII

Simulated MSE and BIAS on Simple Regression for 20% Outliers in y-axis

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED
20	β_0	BIAS	1.8531	0.3405	0.0336	0.1299	0.0527	0.0315
	β_0	MSE	4.1726	0.2057	0.0752	0.1340	0.0819	0.0772
20	β_1	BIAS	0.8329	0.1530	0.0035	0.0436	0.0079	0.0020
	β_1	MSE	2.0266	0.2214	0.1813	0.2103	0.1942	0.1883
50	β_0	BIAS	1.9870	0.3622	0.0326	0.1120	0.0527	0.0272
	β_0	MSE	4.2870	0.1676	0.0285	0.0478	0.0324	0.0283
50	β_1	BIAS	0.9154	0.2038	0.0201	0.0665	0.0321	0.0156
	β_1	MSE	1.8429	0.1538	0.0984	0.1230	0.1083	0.0986
100	β_0	BIAS	2.0325	0.3824	0.0301	0.1119	0.0518	0.0249
	β_0	MSE	4.2979	0.1672	0.0153	0.0331	0.0191	0.0151
100	β_1	BIAS	1.4137	0.3319	0.0206	0.0939	0.0396	0.0168

	β_1	**MSE**	2.6118	0.1733	0.0504	0.0767	0.0580	0.0502
150	β_0	**BIAS**	1.9845	0.3528	0.0287	0.1043	0.0494	0.0233
	β_0	**MSE**	4.0447	0.1374	0.0110	0.0239	0.0136	0.0104
150	β_1	**BIAS**	0.3146	0.0723	0.0093	0.0232	0.0127	0.0074
	β_1	**MSE**	0.4940	0.0428	0.0315	0.0391	0.0342	0.0314
200	β_0	**BIAS**	2.0044	0.3597	0.0319	0.1083	0.0523	0.0250
	β_0	**MSE**	4.0958	0.1393	0.0084	0.0218	0.0110	0.0081
200	β_1	**BIAS**	0.1453	0.0483	0.0173	0.0259	0.0198	0.0160
	β_1	**MSE**	0.3063	0.0308	0.0218	0.0292	0.0245	0.0216

APPENDIX XIV

Kruskal-Wallis Test on Simple Regression for 20% Outliers in y-axis: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment     N   Median  Ave Rank       Z
1            20  1.98575     109.0    6.83
2            20  0.17045      84.3    3.36
3            21  0.02870      35.4   -3.65
4            18  0.08530      61.1    0.08
5            22  0.03330      44.0   -2.47
6            19  0.02330      30.7   -4.07
Overall     120               60.5

H = 78.18  DF = 5  P = 0.000
```

```
H = 78.18  DF = 5  P = 0.000  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XV

Simulated MSE and BIAS on Simple Regression for 30% Outliers in y-axis

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED
20	β_0	BIAS	2.9036	0.9170	0.1008	1.2486	0.1619	0.0860
	β_0	MSE	9.5809	1.2896	0.1452	3.6198	0.1700	0.1172
20	β_1	BIAS	0.3581	0.1170	0.0045	0.1599	0.0091	0.0027
	β_1	MSE	1.7931	0.2551	0.1970	0.4807	0.2225	0.2072
50	β_0	BIAS	3.0141	0.8969	0.0801	0.9085	0.1665	0.0773
	β_0	MSE	9.5918	1.0044	0.0470	2.0136	0.0883	0.0513
50	β_1	BIAS	0.4552	0.2002	0.0185	0.2104	0.0381	0.0135
	β_1	MSE	2.0669	0.2867	0.1472	0.5208	0.2106	0.1677
100	β_0	BIAS	3.0260	0.8592	0.0756	0.6523	0.1549	0.0726

	$\boldsymbol{\beta_0}$	**MSE**	9.4091	0.8271	0.0258	1.1164	0.0546	0.0276
100	$\boldsymbol{\beta_1}$	**BIAS**	0.7668	0.3106	0.0281	0.2412	0.0601	0.0247
	$\boldsymbol{\beta_1}$	**MSE**	1.3959	0.2074	0.0646	0.2852	0.0891	0.0712
150	$\boldsymbol{\beta_0}$	**BIAS**	2.9874	0.8167	0.0747	0.4621	0.1511	0.0718
	$\boldsymbol{\beta_0}$	**MSE**	9.0830	0.7154	0.0189	0.5055	0.0420	0.0201
150	$\boldsymbol{\beta_1}$	**BIAS**	0.4858	0.1906	0.0241	0.1061	0.0442	0.0266
	$\boldsymbol{\beta_1}$	**MSE**	0.8080	0.1078	0.0435	0.1120	0.0605	0.0493
200	$\boldsymbol{\beta_0}$	**BIAS**	3.0142	0.8234	0.0773	0.4167	0.1534	0.0743
	$\boldsymbol{\beta_0}$	**MSE**	9.2003	0.7147	0.0164	0.3854	0.0387	0.0172
200	$\boldsymbol{\beta_1}$	**BIAS**	0.2589	0.1076	0.0218	0.0702	0.0304	0.0216
	$\boldsymbol{\beta_1}$	**MSE**	0.4556	0.0623	0.0304	0.0702	0.0434	0.0331

APPENDIX XVI

Kruskal-Wallis Test on Simple Regression for 30% Outliers in y-axis: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment     N   Median  Ave Rank       Z
1            20  2.48525     103.5    6.05
2            20  0.51265      80.1    2.76
3            21  0.04700      32.6   -4.05
4            18  0.43940      80.5    2.65
5            22  0.07810      41.8   -2.79
6            19  0.04930      28.2   -4.41
```

```
Overall    120              60.5

H = 79.07  DF = 5  P = 0.000
H = 79.07  DF = 5  P = 0.000  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XVII

Simulated MSE and BIAS on Simple Regression for 40% Outliers in y-axis

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED
20	β_0	BIAS	3.7895	2.3770	1.2854	2.9863	0.7094	0.3102
	β_0	MSE	15.8511	7.5439	4.4558	11.8804	1.4118	0.4626
20	β_1	BIAS	1.3977	1.0138	0.6583	1.1816	0.3185	0.1331
	β_1	MSE	4.4199	2.1769	1.6292	3.2272	0.7103	0.3782
50	β_0	BIAS	3.9902	2.2975	0.6596	3.1591	0.6304	0.3195
	β_0	MSE	16.5562	6.1603	1.6311	11.5706	0.7261	0.2682
50	β_1	BIAS	0.2024	0.1706	0.0559	0.1954	0.0597	0.0373
	β_1	MSE	2.2637	0.8666	0.4549	1.6002	0.4953	0.3659

100	$\boldsymbol{\beta_0}$	**BIAS**	4.0362	2.2869	0.3455	3.2253	0.5561	0.2743
	$\boldsymbol{\beta_0}$	**MSE**	16.6201	5.7262	0.4993	11.3777	0.4514	0.1463
100	$\boldsymbol{\beta_1}$	**BIAS**	0.4988	0.3864	0.0699	0.4591	0.1143	0.0606
	$\boldsymbol{\beta_1}$	**MSE**	1.1618	0.4859	0.1295	0.8498	0.2025	0.1425
150	$\boldsymbol{\beta_0}$	**BIAS**	3.9894	2.1852	0.2535	3.1707	0.5460	0.2727
	$\boldsymbol{\beta_0}$	**MSE**	16.1345	5.0842	0.1486	10.6955	0.3898	0.1214
150	$\boldsymbol{\beta_1}$	**BIAS**	0.0931	0.0866	0.0080	0.0950	0.0267	0.0150
	$\boldsymbol{\beta_1}$	**MSE**	0.7666	0.2830	0.0928	0.5534	0.1787	0.1364
200	$\boldsymbol{\beta_0}$	**BIAS**	4.0119	2.2093	0.2309	3.2384	0.5412	0.2731
	$\boldsymbol{\beta_0}$	**MSE**	16.2480	5.1134	0.0851	10.9361	0.3561	0.1095
200	$\boldsymbol{\beta_1}$	**BIAS**	0.0871	0.0731	0.0241	0.0852	0.0302	0.0268
	$\boldsymbol{\beta_1}$	**MSE**	0.5570	0.2131	0.0606	0.4004	0.1229	0.0934

APPENDIX XVIII

Kruskal-Wallis Test on Simple Regression for 40% Outliers in y-axis: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment     N  Median  Ave Rank      Z
1            20  3.8895      86.0   3.59
2            20  2.1810      74.0   1.90
3            21  0.2535      46.3  -2.06
4            18  3.0727      80.8   2.68
5            22  0.3951      48.0  -1.86
```

```
6            19  0.1425      30.3  -4.12
Overall     120              60.5

H = 40.47  DF = 5  P = 0.000
H = 40.47  DF = 5  P = 0.000  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XIX

Simulated MSE and BIAS on Simple Regression for 5% Outliers in x and y-axes

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED
20	β_0	**BIAS**	0.5147	0.0793	0.0131	0.0211	0.0103	0.0086
	β_0	**MSE**	0.5943	0.1484	0.1433	0.1359	0.0716	0.0765
20	β_1	**BIAS**	1.9727	1.9508	1.9475	1.9413	0.1601	0.1237
	β_1	**MSE**	3.9057	3.8177	3.8048	3.7810	0.5616	0.4988
50	β_0	**BIAS**	0.3822	0.0448	0.0258	0.0009	0.0127	0.0111
	β_0	**MSE**	0.2588	0.0520	0.0496	0.0468	0.0127	0.0216
50	β_1	**BIAS**	1.9542	1.9309	1.9123	1.9297	0.0184	0.0103

	$\boldsymbol{\beta_1}$	**MSE**	3.8252	3.7345	3.6922	3.7296	0.1278	0.1136
100	$\boldsymbol{\beta_0}$	**BIAS**	0.4856	0.0635	0.0254	0.0061	0.0091	0.0061
	$\boldsymbol{\beta_0}$	**MSE**	0.3079	0.0344	0.0301	0.0292	0.0117	0.0117
100	$\boldsymbol{\beta_1}$	**BIAS**	1.9790	1.9493	1.9437	1.9483	0.0121	0.0085
	$\boldsymbol{\beta_1}$	**MSE**	3.9190	3.8024	3.7805	3.7983	0.0460	0.0393
150	$\boldsymbol{\beta_0}$	**BIAS**	0.4538	0.0615	0.0227	0.0080	0.0097	0.0063
	$\boldsymbol{\beta_0}$	**MSE**	0.2509	0.0242	0.0207	0.0199	0.0081	0.0081
150	$\boldsymbol{\beta_1}$	**BIAS**	1.9712	1.9427	1.9372	1.9423	0.0031	0.0036
	$\boldsymbol{\beta_1}$	**MSE**	3.8875	3.7760	3.7546	3.7742	0.0213	0.0219
200	$\boldsymbol{\beta_0}$	**BIAS**	0.4967	0.0713	0.0193	0.0140	0.0113	0.0081
	$\boldsymbol{\beta_0}$	**MSE**	0.2825	0.0195	0.0144	0.0136	0.0057	0.0057
200	$\boldsymbol{\beta_1}$	**BIAS**	1.9789	1.9485	1.9426	1.9476	0.0021	0.0017
	$\boldsymbol{\beta_1}$	**MSE**	3.9175	3.7981	3.7751	3.7943	0.0170	0.0171

APPENDIX XX

Kruskal-Wallis Test on Simple Regression for 5% Outliers in x and y-axes: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment     N   Median  Ave Rank      Z
1            20  1.27425      90.9   4.28
2            20  1.03965      76.8   2.30
3            21  0.14330      68.8   1.21
4            18  1.03280      63.2   0.36
5            22  0.01240      34.5  -3.88
```

```
6            19  0.01110      29.7  -4.20
Overall     120               60.5

H = 48.15  DF = 5  P = 0.000
H = 48.15  DF = 5  P = 0.000  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XXI

Simulated MSE and BIAS on Simple Regression for 10% Outliers in x and y-axes

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED
20	β_0	BIAS	1.0771	0.2102	0.0084	0.0913	0.0297	0.0130
	β_0	MSE	1.7889	0.2200	0.1651	0.1832	0.0903	0.0870
20	β_1	BIAS	2.0478	1.9920	1.9782	1.9831	0.3536	0.3026
	β_1	MSE	4.2028	3.9744	3.9194	3.9387	0.9525	0.8435
50	β_0	BIAS	1.0909	0.2160	0.0100	0.0951	0.0213	0.0082
	β_0	MSE	1.4509	0.1175	0.0657	0.0801	0.0312	0.0286
50	β_1	BIAS	2.0465	1.9892	1.9756	1.9817	0.0997	0.0587

	β_1	**MSE**	4.1917	3.9593	3.9054	3.9294	0.3241	0.2338
100	β_0	**BIAS**	1.0715	0.1998	0.0072	0.0794	0.0172	0.0063
	β_0	**MSE**	1.2860	0.0741	0.0333	0.0414	0.0140	0.0132
100	β_1	**BIAS**	2.0470	1.9904	1.9769	1.9834	0.0375	0.0307
	β_1	**MSE**	4.1916	3.9628	3.9091	3.9348	0.0947	0.0828
150	β_0	**BIAS**	1.0718	0.1999	0.0058	0.0779	0.0236	0.0103
	β_0	**MSE**	1.2269	0.0638	0.0235	0.0312	0.0107	0.0098
150	β_1	**BIAS**	2.0447	1.9884	1.9751	1.9814	0.0006	0.0055
	β_1	**MSE**	4.1821	3.9545	3.9019	3.9265	0.0391	0.0310
200	β_0	**BIAS**	1.0673	0.1939	0.0138	0.0733	0.0204	0.0092
	β_0	**MSE**	1.1943	0.0538	0.0161	0.0224	0.0079	0.0072
200	β_1	**BIAS**	2.0445	1.9881	1.9747	1.9812	0.0072	0.0064
	β_1	**MSE**	4.1807	3.9530	3.9000	3.9256	0.0252	0.0242

APPENDIX XXII

Kruskal-Wallis Test on Simple Regression for 10% Outliers in x and y-axes: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment     N   Median  Ave Rank       Z
1            20  1.91670      91.8    4.40
2            20  1.10405      79.1    2.62
3            21  0.16510      57.4   -0.45
4            18  1.08220      71.3    1.42
```

```
5            22  0.02745      36.6  -3.56
6            19  0.02420      28.9  -4.32
Overall     120               60.5

H = 49.80  DF = 5  P = 0.000
H = 49.80  DF = 5  P = 0.000  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XXIII

Simulated MSE and BIAS on Simple Regression for 15% Outliers in x and$y$-axes

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED
20	β_0	BIAS	1.7716	0.3922	0.0344	0.1965	0.0539	0.0203
	β_0	MSE	4.1452	0.3756	0.1939	0.2917	0.1469	0.1302
20	β_1	BIAS	2.0993	2.0092	1.9850	1.9965	0.5903	0.5300
	β_1	MSE	4.4161	4.0420	3.9452	3.9912	1.5426	1.3790
50	β_0	BIAS	1.6118	0.3591	0.0319	0.1839	0.0474	0.0170

	β_0	MSE	2.9758	0.2170	0.0809	0.1322	0.0465	0.0407
50	β_1	**BIAS**	2.0892	2.0088	1.9876	1.9977	0.2339	0.2014
	β_1	**MSE**	4.3681	4.0371	3.9522	3.9927	0.5995	0.5274
100	β_0	**BIAS**	1.7144	0.3978	0.0430	0.2136	0.0720	0.0372
	β_0	**MSE**	3.1317	0.2005	0.0397	0.0946	0.0265	0.0197
100	β_1	**BIAS**	2.0971	2.0126	1.9897	2.0011	0.1397	0.1189
	β_1	**MSE**	4.3992	4.0516	3.9599	4.0054	0.3571	0.3114
150	β_0	**BIAS**	1.6771	0.3791	0.0312	0.1970	0.0614	0.0321
	β_0	**MSE**	2.9411	0.1755	0.0292	0.0748	0.0168	0.0127
150	β_1	**BIAS**	2.0936	2.0104	1.9879	1.9990	0.0481	0.0346
	β_1	**MSE**	4.3844	4.0425	3.9525	3.9967	0.1404	0.1094
200	β_0	**BIAS**	1.7138	0.3846	0.0310	0.1949	0.0634	0.0290
	β_0	**MSE**	3.0311	0.1699	0.0216	0.0636	0.0138	0.0100
200	β_1	**BIAS**	2.0951	2.0103	1.9875	1.9983	0.0163	0.0270
	β_1	**MSE**	4.3904	4.0417	3.9504	3.9936	0.0649	0.0816

APPENDIX XXIV

Kruskal-Wallis Test on Simple Regression for 15% Outliers in x and y-axes: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment     N   Median  Ave Rank       Z
1            20  2.52020      96.3    5.03
2            20  1.20330      78.2    2.49
```

```
3            21  0.19390      54.6  -0.86
4            18  1.14410      69.3   1.17
5            22  0.06415      36.5  -3.58
6            19  0.04070      30.3  -4.13
Overall     120               60.5

H = 52.87  DF = 5  P = 0.000
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XXV

Simulated MSE and BIAS on Simple Regression for 20% Outliers in x and y-axes

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED
20	β_0	BIAS	2.4627	0.6629	0.0809	0.5218	0.1471	0.0571
	β_0	MSE	7.4081	0.7428	0.2606	0.9700	0.2870	0.2245
20	β_1	BIAS	2.1490	2.0321	1.9939	2.0231	1.0304	0.9843

	β_1	**MSE**	4.6271	4.1335	3.9797	4.0984	2.5685	2.3292
50	β_0	**BIAS**	2.4650	0.6690	0.0933	0.4281	0.1656	0.0675
	β_0	**MSE**	6.6139	0.5740	0.1117	0.3823	0.1201	0.0792
50	β_1	**BIAS**	2.1510	2.0354	1.9980	2.0197	0.7556	0.8385
	β_1	**MSE**	4.6302	4.1445	3.9936	4.0812	1.7552	1.8677
100	β_0	**BIAS**	2.4548	0.6753	0.0992	0.4365	0.1779	0.0751
	β_0	**MSE**	6.2936	0.5208	0.0626	0.2899	0.0773	0.0432
100	β_1	**BIAS**	2.1489	2.0351	1.9979	2.0198	0.4878	0.6681
	β_1	**MSE**	4.6195	4.1425	3.9924	4.0804	1.1267	1.4791
150	β_0	**BIAS**	2.4617	0.6621	0.0890	0.4087	0.1689	0.0669
	β_0	**MSE**	6.2293	0.4833	0.0454	0.2330	0.0566	0.0304
150	β_1	**BIAS**	2.1492	2.0341	1.9971	2.0179	0.4060	0.6666
	β_1	**MSE**	4.6201	4.1381	3.9890	4.0725	0.9038	1.4144
200	β_0	**BIAS**	2.4534	0.6630	0.0916	0.4125	0.1772	0.0743
	β_0	**MSE**	6.1568	0.4724	0.0354	0.2180	0.0547	0.0247
200	β_1	**BIAS**	2.1487	2.0340	1.9970	2.0180	0.3748	0.6731
	β_1	**MSE**	4.6178	4.1375	3.9885	4.0727	0.8216	1.4156

APPENDIX XXVI

Kruskal-Wallis Teston Simple Regression for 20% Outliers in x and$y$-axes: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment     N  Median  Ave Rank       Z
1            20  3.5414     102.3    5.88
```

```
2           20  1.3874      71.2   1.50
3           21  0.2899      49.4  -1.61
4           18  1.4939      63.6   0.41
5           22  0.3309      40.4  -3.00
6           19  0.6666      38.0  -3.07
Overall    120              60.5

H = 48.30  DF = 5  P = 0.000
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XXVII

Simulated MSE and BIAS on Multiple Regression for Data with no Outliers

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED
20	β_0	BIAS	0.0100	0.0100	0.0098	0.0096	0.0092	0.0151
	β_0	MSE	0.0520	0.0556	0.0576	0.0529	0.0549	0.0677
20	β_1	BIAS	0.0006	0.0021	0.0048	0.0006	0.0021	0.0041
	β_1	MSE	0.1537	0.1660	0.1771	0.1576	0.1670	0.2225

20	$\boldsymbol{\beta_2}$	**BIAS**	0.0085	0.0105	0.0113	0.0093	0.0111	0.0130
	$\boldsymbol{\beta_2}$	**MSE**	0.0462	0.0471	0.0499	0.0460	0.0487	0.0622
50	$\boldsymbol{\beta_0}$	**BIAS**	0.0022	0.0029	0.0028	0.0023	0.0023	0.0020
	$\boldsymbol{\beta_0}$	**MSE**	0.0198	0.0210	0.0217	0.0204	0.0207	0.0229
50	$\boldsymbol{\beta_1}$	**BIAS**	0.0116	0.0113	0.0114	0.0121	0.0122	0.0155
	$\boldsymbol{\beta_1}$	**MSE**	0.0606	0.0640	0.0651	0.0611	0.0611	0.0683
50	$\boldsymbol{\beta_2}$	**BIAS**	0.0010	0.0022	0.0021	0.0015	0.0015	0.0014
	$\boldsymbol{\beta_2}$	**MSE**	0.0160	0.0164	0.0165	0.0159	0.0160	0.0174
100	$\boldsymbol{\beta_0}$	**BIAS**	0.0022	0.0020	0.0023	0.0025	0.0025	0.0028
	$\boldsymbol{\beta_0}$	**MSE**	0.0101	0.0106	0.0107	0.0102	0.0102	0.0108
100	$\boldsymbol{\beta_1}$	**BIAS**	0.0030	0.0055	0.0051	0.0036	0.0036	0.0036
	$\boldsymbol{\beta_1}$	**MSE**	0.0309	0.0325	0.0327	0.0312	0.0312	0.0328
100	$\boldsymbol{\beta_2}$	**BIAS**	0.0005	0.0009	0.0010	0.0009	0.0009	0.0016
	$\boldsymbol{\beta_2}$	**MSE**	0.0067	0.0071	0.0073	0.0068	0.0068	0.0074
150	$\boldsymbol{\beta_0}$	**BIAS**	0.0006	0.0013	0.0014	0.0009	0.0009	0.0015
	$\boldsymbol{\beta_0}$	**MSE**	0.0064	0.0068	0.0068	0.0064	0.0065	0.0069
150	$\boldsymbol{\beta_1}$	**BIAS**	0.0030	0.0036	0.0040	0.0033	0.0033	0.0039
	$\boldsymbol{\beta_1}$	**MSE**	0.0200	0.0212	0.0211	0.0203	0.0203	0.0212
150	$\boldsymbol{\beta_2}$	**BIAS**	0.0011	0.0012	0.0012	0.0012	0.0012	0.0013
	$\boldsymbol{\beta_2}$	**MSE**	0.0045	0.0050	0.0050	0.0049	0.0049	0.0051
200	$\boldsymbol{\beta_0}$	**BIAS**	0.0014	0.0011	0.0014	0.0016	0.0016	0.0016
	$\boldsymbol{\beta_0}$	**MSE**	0.0050	0.0052	0.0053	0.0050	0.0050	0.0052
200	$\boldsymbol{\beta_1}$	**BIAS**	0.0037	0.0027	0.0028	0.0034	0.0034	0.0031
	$\boldsymbol{\beta_1}$	**MSE**	0.0144	0.0149	0.0150	0.0145	0.0145	0.0151
200	$\boldsymbol{\beta_2}$	**BIAS**	0.0016	0.0007	0.0008	0.0013	0.0013	0.0009
	$\boldsymbol{\beta_2}$	**MSE**	0.0033	0.0035	0.0035	0.0034	0.0034	0.0035

APPENDIX XXVIII

Kruskal-Wallis Test on Multiple Regression for Data with no Outliers: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment    N    Median  Ave Rank      Z
```

```
1           30  0.005700      85.4  -0.59
2           30  0.006150      90.8   0.03
3           30  0.006050      93.5   0.35
4           30  0.005700      87.6  -0.33
5           30  0.005750      89.6  -0.11
6           30  0.006050      96.1   0.64

Overall    180                90.5

H = 0.83  DF = 5  P = 0.975
H = 0.83  DF = 5  P = 0.975  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XXIX

Simulated MSE and BIAS on Multiple Regression for 10% Outliers in x-axis

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED
20	$\boldsymbol{\beta_0}$	BIAS	0.0438	0.0468	0.0456	0.0050	0.0084	0.0081
	$\boldsymbol{\beta_0}$	MSE	0.1357	0.1516	1.1587	0.0748	0.0804	0.1012
20	$\boldsymbol{\beta_1}$	BIAS	1.9684	1.9693	1.9691	0.2817	0.2941	0.2490
	$\boldsymbol{\beta_1}$	MSE	3.8813	3.8853	3.8847	0.8307	0.8411	0.8002

20	$\boldsymbol{\beta_2}$	**BIAS**	0.0103	0.0113	0.0125	0.0097	0.0115	0.0141
	$\boldsymbol{\beta_2}$	**MSE**	0.0919	0.1029	0.1102	0.0591	0.0622	0.0789
50	$\boldsymbol{\beta_0}$	**BIAS**	0.0399	0.0425	0.0415	0.0068	0.0083	0.0104
	$\boldsymbol{\beta_0}$	**MSE**	0.0531	0.0584	0.0591	0.0246	0.0244	0.0255
50	$\boldsymbol{\beta_1}$	**BIAS**	1.9731	1.9720	1.9719	0.0900	0.0718	0.0791
	$\boldsymbol{\beta_1}$	**MSE**	3.8957	3.8913	3.8910	0.2868	0.2485	0.2630
50	$\boldsymbol{\beta_2}$	**BIAS**	0.0019	0.0017	0.0016	0.0021	0.0017	0.0030
	$\boldsymbol{\beta_2}$	**MSE**	0.0334	0.0371	0.0379	0.0190	0.0190	0.0205
100	$\boldsymbol{\beta_0}$	**BIAS**	0.0396	0.0439	0.0439	0.0010	0.0005	0.0012
	$\boldsymbol{\beta_0}$	**MSE**	0.0280	0.0310	0.0307	0.0121	0.0122	0.0123
100	$\boldsymbol{\beta_1}$	**BIAS**	1.9719	1.9702	1.9702	0.0260	0.0183	0.0059
	$\boldsymbol{\beta_1}$	**MSE**	3.8893	3.8828	3.8829	0.0945	0.0791	0.0529
100	$\boldsymbol{\beta_2}$	**BIAS**	0.0019	0.0036	0.0036	0.0013	0.0081	0.0013
	$\boldsymbol{\beta_2}$	**MSE**	0.0185	0.0202	0.0203	0.0088	0.0088	0.0090
150	$\boldsymbol{\beta_0}$	**BIAS**	0.0414	0.0444	0.0441	0.0023	0.0024	0.0019
	$\boldsymbol{\beta_0}$	**MSE**	0.0185	0.0210	0.0208	0.0075	0.0075	0.0077
150	$\boldsymbol{\beta_1}$	**BIAS**	1.9709	1.9691	1.9691	0.0049	0.0030	0.0028
	$\boldsymbol{\beta_1}$	**MSE**	3.8850	3.8781	3.8782	0.0227	0.0265	0.0266
150	$\boldsymbol{\beta_2}$	**BIAS**	0.0050	0.0042	0.0040	0.0034	0.0034	0.0037
	$\boldsymbol{\beta_2}$	**MSE**	0.0120	0.0128	0.0128	0.0061	0.0061	0.0062
200	$\boldsymbol{\beta_0}$	**BIAS**	0.0389	0.0424	0.0420	0.0019	0.0018	0.0019
	$\boldsymbol{\beta_0}$	**MSE**	0.0132	0.0147	0.0146	0.0056	0.0056	0.0057
200	$\boldsymbol{\beta_1}$	**BIAS**	1.9727	1.9709	1.9709	0.0009	0.0009	0.0008
	$\boldsymbol{\beta_1}$	**MSE**	3.8920	3.8849	3.8851	0.0167	0.0167	0.0170
200	$\boldsymbol{\beta_2}$	**BIAS**	0.0038	0.0035	0.0035	0.0039	0.0039	0.0038
	$\boldsymbol{\beta_2}$	**MSE**	0.0090	0.0102	0.0101	0.0044	0.0044	0.0044

APPENDIX XXX

Kruskal-Wallis Test on Multiple Regression for 10% Outliers in x-axis: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment    N    Median  Ave Rank      Z
```

```
1           30  0.040650     112.0   2.48
2           30  0.044150     113.2   2.62
3           30  0.044000     113.3   2.63
4           30  0.008150      67.9  -2.60
5           30  0.008600      68.9  -2.49
6           30  0.008550      67.6  -2.64
Overall    180                90.5

H = 33.21  DF = 5  P = 0.000
H = 33.21  DF = 5  P = 0.000  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XXXI

Simulated MSE and BIAS on Multiple Regression for 20% Outliers in x-axis

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED
20	$\boldsymbol{\beta_0}$	**BIAS**	0.0412	0.0400	0.0377	0.0078	0.0095	0.0082
	$\boldsymbol{\beta_0}$	**MSE**	0.1631	0.1814	0.1900	0.1150	0.1266	0.1484
20	$\boldsymbol{\beta_1}$	**BIAS**	1.9871	1.9872	1.9871	0.8225	0.8270	0.8325

	β_1	**MSE**	3.9521	3.9528	3.9527	1.9240	1.9260	1.9626
20	β_2	**BIAS**	0.0206	0.0205	0.0197	0.0175	0.0174	0.0180
	β_2	**MSE**	0.1053	0.1145	0.1198	0.0778	0.0826	0.1033
50	β_0	**BIAS**	0.0442	0.0465	0.0471	0.0093	0.0072	0.0091
	β_0	**MSE**	0.0607	0.0673	0.0677	0.0372	0.0381	0.0396
50	β_1	**BIAS**	1.9862	1.9857	1.9856	0.4655	0.4448	0.4396
	β_1	**MSE**	3.9463	3.9445	3.9441	1.0260	0.9862	0.9816
50	β_2	**BIAS**	0.0017	0.0018	0.0019	0.0055	0.0049	0.0056
	β_2	**MSE**	0.0370	0.0409	0.0414	0.0237	0.0234	0.0237
100	β_0	**BIAS**	0.5413	0.5955	0.5819	0.1494	0.0946	0.0135
	β_0	**MSE**	0.0302	0.0338	0.0337	0.0163	0.0157	0.0156
100	β_1	**BIAS**	2.0000	2.0000	2.0000	0.8829	0.8461	0.8093
	β_1	**MSE**	3.9360	3.9429	3.9429	0.5358	0.4978	0.4788
100	β_2	**BIAS**	0.1344	0.1792	0.1642	0.1465	0.0472	0.0256
	β_2	**MSE**	0.0172	0.0184	0.0183	0.0104	0.0104	0.0107
150	β_0	**BIAS**	0.0371	0.0399	0.0400	0.0018	0.0017	0.0010
	β_0	**MSE**	0.0207	0.0231	0.0230	0.0094	0.0093	0.0098
150	β_1	**BIAS**	1.9860	1.9850	1.9850	0.1491	0.1140	0.1065
	β_1	**MSE**	3.9448	3.9408	3.9408	0.3226	0.2523	0.2359
150	β_2	**BIAS**	0.0012	0.0014	0.0014	0.0012	0.0005	0.0010
	β_2	**MSE**	0.0129	0.0142	0.0142	0.0071	0.0071	0.0073
200	β_0	**BIAS**	0.5143	0.5684	0.5684	0.1693	0.1349	0.1593
	β_0	**MSE**	0.0162	0.0182	0.0180	0.0073	0.0067	0.0068
200	β_1	**BIAS**	2.0000	2.0000	2.0000	0.9744	0.5278	0.4601
	β_1	**MSE**	3.9467	3.9433	3.9434	0.1702	0.1042	0.0961
200	β_2	**BIAS**	2.4397	0.0485	0.0623	0.1722	0.0896	0.0846
	β_2	**MSE**	0.0097	0.0106	0.0106	0.0049	0.0049	0.0049

APPENDIX XXXII

Kruskal-Wallis Test on Multiple Regression for 20% Outliers in x-axis: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment    N   Median  Ave Rank      Z
```

```
1           30  0.11985     105.0   1.67
2           30  0.09090     104.3   1.58
3           30  0.09375     104.0   1.56
4           30  0.09640      80.1  -1.19
5           30  0.06490      75.8  -1.70
6           30  0.03260      73.8  -1.92
Overall    180               90.5

H = 13.09  DF = 5  P = 0.023
H = 13.09  DF = 5  P = 0.023  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XXXIII

Simulated MSE and BIAS on Multiple Regression for 30% Outliers in x-axis

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED
20	$\boldsymbol{\beta_0}$	**BIAS**	0.0430	0.0461	0.0479	0.0364	0.0439	0.0471

	β_0	**MSE**	0.1773	0.1983	0.2099	0.1667	0.1809	0.2113
20	β_1	**BIAS**	1.9904	1.9903	1.9901	1.4550	1.4457	1.3603
	β_1	**MSE**	3.9642	3.9640	3.9636	3.0673	3.0473	2.8826
20	β_2	**BIAS**	0.0081	0.0082	0.0088	0.0079	0.0055	0.0098
	β_2	**MSE**	0.1017	0.1116	0.1168	0.0970	0.1056	0.1303
50	β_0	**BIAS**	0.0493	0.0500	0.0508	0.0285	0.0281	0.0278
	β_0	**MSE**	0.0709	0.0780	0.0786	0.0564	0.0570	0.0603
50	β_1	**BIAS**	1.9890	1.9885	1.9884	1.2666	1.2327	1.2732
	β_1	**MSE**	3.9571	3.9553	3.9546	2.6345	2.5726	2.6475
50	β_2	**BIAS**	0.0024	0.0015	0.0013	0.0009	0.0003	0.0022
	β_2	**MSE**	0.0377	0.0419	0.0423	0.0340	0.0341	0.0380
100	β_0	**BIAS**	0.0356	0.0393	0.0394	0.0200	0.0173	0.0219
	β_0	**MSE**	0.0330	0.0368	0.0365	0.0261	0.0257	0.0281
100	β_1	**BIAS**	1.9913	1.9907	1.9907	1.1025	1.0521	1.1904
	β_1	**MSE**	3.9656	3.9633	3.9633	2.2759	2.1822	2.4529
100	β_2	**BIAS**	0.0068	0.0063	0.0059	0.0055	0.0068	0.0027
	β_2	**MSE**	0.0185	0.0204	0.0203	0.0162	0.0162	0.0172
150	β_0	**BIAS**	0.0384	0.0400	0.0402	0.0181	0.0171	0.0230
	β_0	**MSE**	0.0228	0.0253	0.0251	0.0169	0.0167	0.0186
150	β_1	**BIAS**	1.9906	1.9901	1.9901	0.9897	0.8934	1.1016
	β_1	**MSE**	3.9629	3.9609	3.9610	2.0272	1.8372	2.2525
150	β_2	**BIAS**	0.0002	0.0003	0.0062	0.0009	0.0007	0.0009
	β_2	**MSE**	0.0117	0.0131	0.0130	0.0098	0.0101	0.0106
200	β_0	**BIAS**	0.0453	0.0480	0.0482	0.0265	0.0233	0.0299
	β_0	**MSE**	0.0183	0.0197	0.0198	0.0129	0.0122	0.0144
200	β_1	**BIAS**	1.9910	1.9895	1.9895	0.9721	0.8087	1.0685
	β_1	**MSE**	3.9609	3.9583	3.9584	1.9867	1.6560	2.1805
200	β_2	**BIAS**	0.0031	0.0030	0.0033	0.0008	0.0016	0.0023
	β_2	**MSE**	0.0087	0.0095	0.0095	0.0071	0.0071	0.0075

APPENDIX XXXIV

Kruskal-Wallis Test on Multiple Regression for 30% Outliers in x-axis: Response versus Treatment

```
Kruskal-Wallis Test on Response
```

```
Treatment     N   Median  Ave Rank       Z
1            30  0.04415      96.2    0.66
2            30  0.04705      97.0    0.74
3            30  0.04805      97.8    0.84
4            30  0.03125      83.1   -0.85
5            30  0.03110      81.9   -0.99
6            30  0.03395      87.0   -0.41
Overall     180               90.5

H = 2.97  DF = 5  P = 0.705
H = 2.97  DF = 5  P = 0.705  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XXXV

Simulated MSE and BIAS on Multiple Regression for 10% Outliers in y-axis

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED

20	$\boldsymbol{\beta_0}$	**BIAS**	0.8800	0.1106	0.0075	0.0128	0.0028	0.0061
	$\boldsymbol{\beta_0}$	**MSE**	1.2113	0.0783	0.0602	0.0615	0.0596	0.0678
20	$\boldsymbol{\beta_1}$	**BIAS**	1.3779	0.1834	0.0019	0.0308	0.0011	0.0040
	$\boldsymbol{\beta_1}$	**MSE**	3.2926	0.2405	0.2004	0.2016	0.1957	0.2331
20	$\boldsymbol{\beta_2}$	**BIAS**	0.1146	0.0050	0.0158	0.0139	0.0158	0.0155
	$\boldsymbol{\beta_2}$	**MSE**	0.7776	0.0699	0.0650	0.0701	0.0655	0.0875
50	$\boldsymbol{\beta_0}$	**BIAS**	1.0628	0.1610	0.0056	0.0355	0.0128	0.0030
	$\boldsymbol{\beta_0}$	**MSE**	1.3223	0.0542	0.0258	0.0294	0.0263	0.0261
50	$\boldsymbol{\beta_1}$	**BIAS**	0.4382	0.0810	0.0153	0.0278	0.0176	0.0163
	$\boldsymbol{\beta_1}$	**MSE**	0.7506	0.0853	0.0746	0.0772	0.0739	0.0742
50	$\boldsymbol{\beta_2}$	**BIAS**	0.7271	0.1176	0.0077	0.0282	0.0115	0.0040
	$\boldsymbol{\beta_2}$	**MSE**	0.6958	0.0365	0.0201	0.0230	0.0208	0.0208
100	$\boldsymbol{\beta_0}$	**BIAS**	1.0190	0.1443	0.0099	0.0330	0.0151	0.0082
	$\boldsymbol{\beta_0}$	**MSE**	1.1268	0.0341	0.0120	0.0140	0.0123	0.0119
100	$\boldsymbol{\beta_1}$	**BIAS**	0.5021	0.0791	0.0025	0.0156	0.0050	0.0023
	$\boldsymbol{\beta_1}$	**MSE**	0.6711	0.0510	0.0404	0.0437	0.0411	0.0403
100	$\boldsymbol{\beta_2}$	**BIAS**	0.0482	0.0083	0.0019	0.0025	0.0018	0.0021
	$\boldsymbol{\beta_2}$	**MSE**	0.0513	0.0082	0.0078	0.0078	0.0076	0.0077
150	$\boldsymbol{\beta_0}$	**BIAS**	1.0020	0.1428	0.0085	0.0321	0.0135	0.0064
	$\boldsymbol{\beta_0}$	**MSE**	1.0639	0.0288	0.0078	0.0091	0.0079	0.0078
150	$\boldsymbol{\beta_1}$	**BIAS**	0.0447	0.0092	0.0029	0.0026	0.0023	0.0024
	$\boldsymbol{\beta_1}$	**MSE**	0.1729	0.0256	0.0241	0.0251	0.0242	0.0241
150	$\boldsymbol{\beta_2}$	**BIAS**	0.2897	0.0485	0.0033	0.0118	0.0047	0.0033
	$\boldsymbol{\beta_2}$	**MSE**	0.1423	0.0092	0.0059	0.0066	0.0060	0.0060
200	$\boldsymbol{\beta_0}$	**BIAS**	1.0106	0.1438	0.0102	0.0338	0.0154	0.0080
	$\boldsymbol{\beta_0}$	**MSE**	1.0675	0.0274	0.0060	0.0075	0.0062	0.0058
200	$\boldsymbol{\beta_1}$	**BIAS**	0.1755	0.0295	0.0040	0.0093	0.0050	0.0043
	$\boldsymbol{\beta_1}$	**MSE**	0.1813	0.0201	0.0185	0.0194	0.0188	0.0183
200	$\boldsymbol{\beta_2}$	**BIAS**	0.1110	0.0154	0.0006	0.0013	0.0006	0.0010
	$\boldsymbol{\beta_2}$	**MSE**	0.0378	0.0043	0.0038	0.0039	0.0037	0.0038

APPENDIX XXXVI

Kruskal-Wallis Test on Multiple Regression for 10% Outliers in y-axis: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment    N    Median  Ave Rank      Z
1           30  0.711450     159.5   7.95
2           30  0.049750     112.0   2.48
3           30  0.008150      62.0  -3.28
4           30  0.021200      83.2  -0.84
5           30  0.012550      65.5  -2.88
6           30  0.007750      60.7  -3.43
Overall    180                90.5

H = 83.99  DF = 5  P = 0.000
H = 83.99  DF = 5  P = 0.000  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XXXVII

Simulated MSE and BIAS on Multiple Regression for 20% Outliers in y-axis

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED

20	$\boldsymbol{\beta_0}$	**BIAS**	1.7370	0.3018	0.0149	0.0948	0.0365	0.0123
	$\boldsymbol{\beta_0}$	**MSE**	3.7599	0.1805	0.0709	0.0985	0.0754	0.0758
20	$\boldsymbol{\beta_1}$	**BIAS**	0.6647	0.0782	0.0109	0.0083	0.0084	0.0108
	$\boldsymbol{\beta_1}$	**MSE**	2.1465	0.2271	0.2026	0.2269	0.0280	0.2201
20	$\boldsymbol{\beta_2}$	**BIAS**	0.5552	0.1733	0.0425	0.0880	0.0507	0.0329
	$\boldsymbol{\beta_2}$	**MSE**	1.4471	0.1305	0.0852	0.1316	0.1000	0.1000
50	$\boldsymbol{\beta_0}$	**BIAS**	2.0505	0.3999	0.0253	0.1257	0.0525	0.0197
	$\boldsymbol{\beta_0}$	**MSE**	4.5510	0.2068	0.0335	0.0673	0.0411	0.0339
50	$\boldsymbol{\beta_1}$	**BIAS**	1.0370	0.2229	0.0292	0.0790	0.0417	0.0278
	$\boldsymbol{\beta_1}$	**MSE**	1.9479	0.1534	0.0870	0.1135	0.0933	0.0883
50	$\boldsymbol{\beta_2}$	**BIAS**	0.6703	0.1689	0.0149	0.0566	0.0273	0.0111
	$\boldsymbol{\beta_2}$	**MSE**	0.8455	0.0702	0.0308	0.0507	0.0370	0.0320
100	$\boldsymbol{\beta_0}$	**BIAS**	2.0171	0.3829	0.0317	0.1159	0.0530	0.0264
	$\boldsymbol{\beta_0}$	**MSE**	4.2427	0.1684	0.0162	0.0353	0.0206	0.0160
100	$\boldsymbol{\beta_1}$	**BIAS**	1.3637	0.3232	0.0213	0.0967	0.0375	0.0177
	$\boldsymbol{\beta_1}$	**MSE**	2.5008	0.1734	0.0519	0.0797	0.0601	0.0532
100	$\boldsymbol{\beta_2}$	**BIAS**	0.1156	0.0304	0.0055	0.0132	0.0071	0.0051
	$\boldsymbol{\beta_2}$	**MSE**	0.1504	0.0130	0.0100	0.0124	0.0110	0.0103
150	$\boldsymbol{\beta_0}$	**BIAS**	1.9964	0.3567	0.0271	0.0157	0.0481	0.0211
	$\boldsymbol{\beta_0}$	**MSE**	4.0920	0.1394	0.0100	0.0233	0.0124	0.0097
150	$\boldsymbol{\beta_1}$	**BIAS**	0.3770	0.0748	0.0027	0.0209	0.0073	0.0008
	$\boldsymbol{\beta_1}$	**MSE**	0.4981	0.0443	0.0324	0.0421	0.0360	0.0325
150	$\boldsymbol{\beta_2}$	**BIAS**	0.3026	0.0703	0.0069	0.0234	0.0102	0.0058
	$\boldsymbol{\beta_2}$	**MSE**	0.1858	0.0143	0.0073	0.0100	0.0083	0.0074
200	$\boldsymbol{\beta_0}$	**BIAS**	2.0080	0.3556	0.0279	0.1044	0.0486	0.0218
	$\boldsymbol{\beta_0}$	**MSE**	4.1176	0.1366	0.0079	0.0202	0.0103	0.0077
200	$\boldsymbol{\beta_1}$	**BIAS**	0.0188	0.0301	0.0026	0.0021	0.0071	0.0033
	$\boldsymbol{\beta_1}$	**MSE**	0.3028	0.0281	0.0233	0.0282	0.0257	0.0238
200	$\boldsymbol{\beta_2}$	**BIAS**	0.1287	0.0286	0.0034	0.0101	0.0059	0.0039
	$\boldsymbol{\beta_2}$	**MSE**	0.0681	0.0063	0.0047	0.0058	0.0050	0.0047

APPENDIX XXXVIII

Kruskal-Wallis Teston Multiple Regression for 20% Outliers in y-axis: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment    N   Median  Ave Rank       Z
1           30  1.20035     157.8   7.75
2           30  0.14640     118.8   3.26
3           30  0.02230      57.9  -3.75
4           30  0.04640      84.9  -0.65
5           30  0.03200      67.9  -2.61
6           30  0.01870      55.8  -4.00
Overall    180               90.5

H = 89.92  DF = 5  P = 0.000
H = 89.92  DF = 5  P = 0.000  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XXXIX

Simulated MSE and BIAS on Multiple Regression for 30% Outliers in y-axis

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED

20	β_0	**BIAS**	2.6733	1.0070	0.2159	0.5721	0.1781	0.0764
	β_0	**MSE**	8.3023	1.6545	0.4679	1.2216	0.2081	0.1200
20	β_1	**BIAS**	0.0159	0.1784	0.0952	0.0754	0.0576	0.0345
	β_1	**MSE**	2.2758	0.4384	0.3291	0.4289	0.2970	0.2803
20	β_2	**BIAS**	1.0262	0.6365	0.2144	0.3277	0.1498	0.0804
	β_2	**MSE**	2.5571	0.8740	0.4153	0.5396	0.2389	0.1888
50	β_0	**BIAS**	3.1262	1.1827	0.1349	0.6772	0.2227	0.0890
	β_0	**MSE**	10.2968	1.7730	0.1639	1.1934	0.1775	0.0757
50	β_1	**BIAS**	0.2966	0.3323	0.0375	0.1579	0.0489	0.0027
	β_1	**MSE**	1.6722	0.4759	0.1773	0.4070	0.2531	0.1630
50	β_2	**BIAS**	1.1783	0.6918	0.0986	0.3070	0.1406	0.0594
	β_2	**MSE**	1.9328	0.7343	0.1312	0.4541	0.1538	0.0786
100	β_0	**BIAS**	2.9958	0.8616	0.0807	0.4267	0.1602	0.0800
	β_0	**MSE**	9.2274	0.8322	0.0288	0.4267	0.0600	0.0308
100	β_1	**BIAS**	0.7223	0.3024	0.0244	0.1579	0.0600	0.0238
	β_1	**MSE**	1.3055	0.2089	0S.0621	0.1653	0.0883	0.0683
100	β_2	**BIAS**	0.1949	0.0925	0.0126	0.0526	0.0222	0.0134
	β_2	**MSE**	0.2221	0.0311	0.0129	0.0289	0.0176	0.0147
150	β_0	**BIAS**	2.9829	0.8329	0.0756	0.3719	0.1549	0.0718
	β_0	**MSE**	9.0538	0.7476	0.0186	0.2589	0.0434	0.0197
150	β_1	**BIAS**	0.5423	0.2129	0.0182	0.1033	0.0415	0.0187
	β_1	**MSE**	0.8525	0.1243	0.0473	0.1119	0.0687	0.0530
150	β_2	**BIAS**	0.3044	0.1267	0.0143	0.0632	0.0295	0.0145
	β_2	**MSE**	0.2307	0.0351	0.0107	0.0256	0.0160	0.0124
200	β_0	**BIAS**	3.0213	0.8221	0.0765	0.3375	0.1528	0.0729
	β_0	**MSE**	9.2560	0.7156	0.0157	0.1739	0.0381	0.0166
200	β_1	**BIAS**	0.2577	0.0871	0.0046	0.0304	0.0107	0.0057
	β_1	**MSE**	0.4520	0.0563	0.0314	0.0526	0.0409	0.0355
200	β_2	**BIAS**	0.1929	0.0689	0.0059	0.0282	0.0119	0.0044
	β_2	**MSE**	0.1234	0.0158	0.0068	0.0129	0.0091	0.0076

APPENDIX XL

Kruskal-Wallis Test on Multiple Regression for 30% Outliers in y-axis: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment    N   Median  Ave Rank       Z
1           30  1.24190     147.1    6.52
2           30  0.45715     119.1    3.29
3           30  0.05470      59.4   -3.58
4           30  0.21640     100.4    1.14
5           30  0.06435      68.1   -2.58
6           30  0.04425      48.8   -4.80
Overall    180               90.5

H = 80.93  DF = 5  P = 0.000
H = 80.93  DF = 5  P = 0.000  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XLI

Simulated MSE and BIAS on Multiple Regression for 40% Outliers in y-axis

SAMPLE	BETA	CRITERI	OLS	HUBER	BISQUA	HAMPEL	ALARM	PROPOSE

SIZE		A			RE			D
20	β_0	BIAS	3.9455	2.6372	1.8229	2.3304	0.9315	0.4042
	β_0	MSE	17.3861	9.3040	6.6230	9.5266	2.3911	0.8166
20	β_1	BIAS	1.6181	1.5173	1.4029	1.1184	0.5720	0.2760
	β_1	MSE	6.5166	5.7200	5.8235	4.4222	3.0762	1.7525
20	β_2	BIAS	0.6502	0.8344	0.8935	0.5206	0.3212	0.1689
	β_2	MSE	3.0760	2.8364	3.6828	2.3742	2.1760	1.4693
50	β_0	BIAS	4.1457	2.6964	1.2994	2.8781	1.2522	0.4830
	β_0	MSE	17.8895	8.2459	3.6370	10.6878	2.9975	0.6740
50	β_1	BIAS	0.4531	0.1883	0.0231	0.3019	0.0755	0.0453
	β_1	MSE	2.0876	0.9115	0.6388	1.2974	0.7046	0.4015
50	β_2	BIAS	1.6332	1.4494	0.8593	1.3434	0.7551	0.3017
	β_2	MSE	3.2959	2.6364	1.6588	2.5810	1.3282	0.4140
100	β_0	BIAS	3.9923	2.2667	0.3977	2.8736	0.6103	0.2931
	β_0	MSE	16.2661	5.6214	0.6066	9.9614	0.5620	0.1756
100	β_1	BIAS	0.5147	0.3865	0.0812	0.4283	0.1346	0.0543
	β_1	MSE	1.1650	0.4648	0.1327	0.7253	0.2135	0.1353
100	β_2	BIAS	0.1580	0.1148	0.0297	0.1293	0.0366	0.0128
	β_2	MSE	0.2695	0.1074	0.0403	0.1778	0.0641	0.0421
150	β_0	BIAS	3.9972	2.2677	0.2973	0.0356	0.5971	0.2869
	β_0	MSE	16.1967	5.4705	0.2773	10.3128	0.4678	0.1349
150	β_1	BIAS	0.0327	0.0570	0.0127	0.0646	0.0195	0.0085
	β_1	MSE	0.7449	0.2984	0.1010	0.5063	0.2048	0.1417
150	β_2	BIAS	0.4032	0.3238	0.0625	0.3688	0.1200	0.0591
	β_2	MSE	0.3315	0.1741	0.0343	0.2602	0.0564	0.0344
200	β_0	BIAS	4.0250	2.2330	0.2428	3.0746	0.5587	0.2715
	β_0	MSE	16.3720	5.2455	0.1187	10.3992	0.3834	0.1057
200	β_1	BIAS	0.0841	0.0420	0.0039	0.0646	0.0018	0.0060
	β_1	MSE	0.5241	0.2049	0.0617	0.3634	0.1239	0.0871
200	β_2	BIAS	0.3261	0.2437	0.0313	0.2937	0.0695	0.0340
	β_2	MSE	0.2172	0.0999	0.0148	0.1646	0.0297	0.0186

APPENDIX XLII

Kruskal-Wallis Test on Multiple Regression for 40% Outliers in y-axis: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment     N  Median  Ave Rank      Z
1            30  1.6257     121.3   3.55
2            30  1.1805     106.6   1.85
3            30  0.2601      72.9  -2.03
4            30  0.9219     108.6   2.09
5            30  0.4256      77.8  -1.46
6            30  0.1553      55.7  -4.00
Overall     180              90.5

H = 35.57  DF = 5  P = 0.000
H = 35.57  DF = 5  P = 0.000  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XLIII

Simulated MSE and BIAS on Multiple Regression for 5% Outliers in x and y-axes

SAMPLE	BETA	CRITERI	OLS	HUBER	BISQUA	HAMPEL	ALARM	PROPOSE

SIZE		A			RE			D
20	β_0	BIAS	0.4827	0.0654	0.0120	0.0254	0.0189	0.0144
	β_0	MSE	0.7872	0.1666	0.1660	0.0784	0.0774	0.0951
20	β_1	BIAS	1.9552	1.9364	1.9342	0.1719	0.1660	0.1547
	β_1	MSE	4.1399	4.0644	4.0563	0.5339	0.5281	0.5677
20	β_2	BIAS	0.2888	0.0497	0.0145	0.0056	0.0115	0.0143
	β_2	MSE	0.6827	0.1375	0.1307	0.0555	0.0603	0.0823
50	β_0	BIAS	0.3594	0.0214	0.0287	0.0031	0.0000	0.0001
	β_0	MSE	0.3053	0.0510	0.0446	0.0250	0.0249	0.0257
50	β_1	BIAS	1.9367	1.7006	1.4066	0.0786	0.0542	0.0400
	β_1	MSE	3.9011	3.0906	2.5550	0.1799	0.1500	0.1334
50	β_2	BIAS	0.2984	0.0536	0.0006	0.0016	0.0017	0.0028
	β_2	MSE	0.2789	0.0437	0.0364	0.0195	0.0191	0.0199
100	β_0	BIAS	0.4832	0.0418	0.0268	0.0015	0.0011	0.0015
	β_0	MSE	0.3337	0.0289	0.0226	0.0113	0.0111	0.0113
100	β_1	BIAS	1.9925	1.8154	1.3881	0.1089	0.0442	0.0333
	β_1	MSE	4.0273	3.3796	2.4411	0.1348	0.0706	0.0614
100	β_2	BIAS	0.4368	0.0894	0.0056	0.0033	0.0032	0.0033
	β_2	MSE	0.3123	0.0314	0.0183	0.0091	0.0088	0.0088
150	β_0	BIAS	0.4538	0.0413	0.0127	0.0115	0.0089	0.0075
	β_0	MSE	0.2721	0.0199	0.0135	0.0080	0.0076	0.0077
150	β_1	BIAS	1.9662	1.7781	1.0941	0.1066	0.0306	0.0260
	β_1	MSE	3.9026	3.2206	1.8345	0.1061	0.0398	0.0377
150	β_2	BIAS	0.4412	0.0794	0.0048	0.0018	0.0020	0.0025
	β_2	MSE	0.2748	0.0219	0.0107	0.0064	0.0059	0.0060
200	β_0	BIAS	0.4935	0.0457	0.0153	0.0061	0.0033	0.0017
	β_0	MSE	0.2953	0.0149	0.0087	0.0060	0.0057	0.0057
200	β_1	BIAS	1.9724	1.7915	0.9256	0.1178	0.0244	0.0137
	β_1	MSE	3.9199	3.2576	1.5327	0.1155	0.0351	0.0279
200	β_2	BIAS	0.4878	0.0950	0.0016	0.0025	0.0020	0.0023
	β_2	MSE	0.3028	0.0198	0.0073	0.0046	0.0044	0.0044

APPENDIX XLIV

Kruskal-Wallis Test on Multiple Regression for 5% Outliers in x and y-axes: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment     N   Median  Ave Rank       Z
1            30  0.48295     148.8   6.71
2            30  0.07240     115.7   2.90
3            30  0.02775      92.3   0.21
4            30  0.01550      67.5  -2.64
5            30  0.01520      59.8  -3.54
6            30  0.01400      58.9  -3.64
Overall     180               90.5

H = 71.90  DF = 5  P = 0.000
H = 71.90  DF = 5  P = 0.000  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XLV

Simulated MSE and BIAS on Multiple Regression for 10% Outliers in x and y-axes

SAMPLE	BETA	CRITERI	OLS	HUBER	BISQUA	HAMPEL	ALARM	PROPOSE

SIZE		A			RE			D
20	β_0	BIAS	1.0970	0.2026	0.0007	0.0384	0.0131	0.0051
	β_0	MSE	2.3390	0.2547	0.1750	0.1096	0.1054	0.1072
20	β_1	BIAS	2.0409	1.8881	1.7762	0.3924	0.3628	0.3328
	β_1	MSE	4.3281	3.7749	3.5352	1.0484	0.9481	0.9278
20	β_2	BIAS	0.7530	0.1900	0.0162	0.0138	0.0021	0.0044
	β_2	MSE	1.8666	0.2634	0.1574	0.1004	0.0893	0.0100
50	β_0	BIAS	1.1030	0.1890	0.0058	0.0288	0.0149	0.0101
	β_0	MSE	1.6288	0.1145	0.0624	0.0334	0.0289	0.0272
50	β_1	BIAS	2.0445	1.9344	1.8526	0.2326	0.1317	0.0920
	β_1	MSE	4.2386	3.8231	3.5850	0.4501	0.2762	0.2273
50	β_2	BIAS	0.9505	0.2460	0.0423	0.0204	0.0104	0.0082
	β_2	MSE	1.3820	0.1378	0.0592	0.0307	0.0246	0.0234
100	β_0	BIAS	1.0773	0.1768	0.0132	0.0320	0.0137	0.0092
	β_0	MSE	1.3743	0.0725	0.0325	0.0198	0.0148	0.0142
100	β_1	BIAS	2.0398	1.9223	1.7810	0.3749	0.0767	0.0586
	β_1	MSE	4.1869	3.7358	3.2977	0.6077	0.1268	0.1012
100	β_2	BIAS	0.9463	0.2422	0.0373	0.0215	0.0031	0.0019
	β_2	MSE	1.1211	0.0928	0.0268	0.0171	0.0105	0.0104
150	β_0	BIAS	1.0607	0.1692	0.0159	0.0290	0.0121	0.0058
	β_0	MSE	1.2703	0.0542	0.0192	0.0126	0.0089	0.0083
150	β_1	BIAS	2.0456	1.9237	1.7540	0.4106	0.0537	0.0239
	β_1	MSE	4.2030	3.7309	3.1954	0.6464	0.0775	0.0485
150	β_2	BIAS	0.9807	0.2517	0.0345	0.0251	0.0044	0.0035
	β_2	MSE	1.1244	0.0878	0.0185	0.0130	0.0070	0.0066
200	β_0	BIAS	1.0670	0.1699	0.0154	0.0286	0.0107	0.0049
	β_0	MSE	1.2373	0.0474	0.0146	0.0109	0.0069	0.0064
200	β_1	BIAS	2.0449	1.9252	1.7170	0.5577	0.0656	0.0343
	β_1	MSE	4.1940	3.7264	3.0553	0.9027	0.0718	0.0473
200	β_2	BIAS	1.0033	0.2467	0.0246	0.0274	0.0036	0.0043
	β_2	MSE	1.1228	0.0778	0.0131	0.0120	0.0052	0.0049

APPENDIX XLVI

Kruskal-Wallis Test on Multiple Regression for 10% Outliers in x and$y$-axes: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment     N   Median  Ave Rank       Z
1            30  1.32230     150.7    6.93
2            30  0.24410     122.1    3.63
3            30  0.03980      90.5    0.01
4            30  0.03135      78.8   -1.35
5            30  0.01485      54.9   -4.10
6            30  0.01025      46.0   -5.12
Overall     180               90.5

H = 88.46  DF = 5  P = 0.000
H = 88.46  DF = 5  P = 0.000  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XLVII

Simulated MSE and BIAS on Multiple Regression for 15% Outliers in x and y-axes

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED
20	β_0	**BIAS**	1.6927	0.4370	0.0680	0.1109	0.0407	0.0182
	β_0	**MSE**	4.7700	0.7943	0.5347	0.2455	0.1569	0.1462
20	β_1	**BIAS**	2.0982	2.0047	1.9679	0.5967	0.4631	0.4363
	β_1	**MSE**	4.5136	4.1521	4.0595	1.6517	1.2286	1.1852
20	β_2	**BIAS**	1.2932	0.4980	0.1498	0.1103	0.0369	0.0132
	β_2	**MSE**	3.6658	0.9011	0.4868	0.3000	0.1460	0.1378
50	β_0	**BIAS**	1.5423	0.3292	0.0184	0.0894	0.0323	0.0127
	β_0	**MSE**	2.9695	0.2228	0.0815	0.0815	0.0453	0.0400
50	β_1	**BIAS**	2.0759	1.9665	1.8903	0.5917	0.2292	0.1736
	β_1	**MSE**	4.3481	3.9230	3.7009	1.1675	0.4640	0.3513
50	β_2	**BIAS**	1.3342	0.4281	0.0861	0.0869	0.1557	0.0064
	β_2	**MSE**	2.3671	0.3053	0.0759	0.0878	0.0330	0.0283
100	β_0	**BIAS**	1.7380	0.3989	0.0280	0.1490	0.0390	0.0157
	β_0	**MSE**	3.3509	0.2257	0.0429	0.0788	0.0223	0.0185
100	β_1	**BIAS**	2.0970	1.9837	1.8958	0.9459	0.1829	0.1019
	β_1	**MSE**	4.4154	3.9615	3.6785	1.8691	0.2994	0.1639
100	β_2	**BIAS**	1.4936	0.4936	0.0976	0.1970	0.0107	0.0020
	β_2	**MSE**	2.5625	0.3181	0.0492	0.1412	0.0190	0.0154
150	β_0	**BIAS**	1.6754	0.3534	0.0009	0.1257	0.0194	0.0035
	β_0	**MSE**	3.0133	0.1654	0.0276	0.0529	0.0139	0.0117
150	β_1	**BIAS**	2.0900	1.9776	1.8625	1.1476	0.1702	0.0774
	β_1	**MSE**	4.3797	3.9285	3.5430	2.2395	0.2564	0.1007
150	β_2	**BIAS**	1.4349	0.4681	0.0981	0.2238	0.0127	0.0012
	β_2	**MSE**	2.2640	0.2611	0.0364	0.1221	0.0111	0.0085
200	β_0	**BIAS**	1.7078	0.3735	0.0122	0.1560	0.0297	0.0117
	β_0	**MSE**	3.0991	0.1719	0.0208	0.0553	0.0116	0.0091
200	β_1	**BIAS**	2.0967	1.9831	1.8639	1.3730	0.2419	0.0741
	β_1	**MSE**	4.0584	3.9470	3.5381	2.7039	0.3560	0.0787
200	β_2	**BIAS**	1.4771	0.4797	0.0999	0.2670	0.0148	0.0020
	β_2	**MSE**	2.3358	0.2616	0.0294	0.1325	0.0104	0.0063

APPENDIX XLVIII

Kruskal-Wallis Test on Multiple Regression for 15% Outliers in x and y-axes: Response versus Treatment

Kruskal-Wallis Test on Response

Treatment	N	Median	Ave Rank	Z
1	30	2.18110	152.4	7.13
2	30	0.47390	120.7	3.48
3	30	0.09785	84.4	-0.70
4	30	0.17650	90.0	-0.05
5	30	0.03985	55.6	-4.02
6	30	0.01835	39.8	-5.84
Overall	180		90.5	

H = 94.76 DF = 5 P = 0.000
H = 94.76 DF = 5 P = 0.000 (adjusted for ties)

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX XLIX

Simulated MSE and BIAS on Multiple Regression for 20% Outliers in x and$y$-axes

SAMPLE SIZE	BETA	CRITERIA	OLS	HUBER	BISQUARE	HAMPEL	ALARM	PROPOSED
20	β_0	BIAS	2.4819	1.0034	0.3308	0.6816	0.1465	0.0603
	β_0	MSE	8.9306	2.2063	1.0812	1.9598	0.4146	0.2479
20	β_1	BIAS	2.1647	2.0564	1.9990	1.0853	0.8032	0.6188
	β_1	MSE	4.7698	4.3245	4.1447	3.3236	1.9788	1.6481
20	β_2	BIAS	1.8517	1.0465	0.5267	0.5560	0.1276	0.0440
	β_2	MSE	5.8901	2.5138	1.5443	1.6220	0.4391	0.2376
50	β_0	BIAS	2.4429	0.8453	0.1307	0.6919	0.1177	0.0327
	β_0	MSE	6.9271	1.0715	0.2047	1.2201	0.1274	0.0695
50	β_1	BIAS	2.1578	2.0445	0.1964	1.3283	0.5140	0.3497
	β_1	MSE	4.6850	4.2185	3.9534	2.9971	1.1126	0.6673
50	β_2	BIAS	1.9403	0.9764	0.3145	0.7505	0.1257	0.0268
	β_2	MSE	4.5939	1.3577	0.3575	1.2879	1.7451	0.0585
100	β_0	BIAS	2.4469	0.8381	0.1270	0.8348	0.1558	0.0451
	β_0	MSE	6.4537	0.8616	0.0902	1.1565	0.0989	0.0332
100	β_1	BIAS	2.1513	2.0375	1.9557	1.7026	0.5956	0.2231
	β_1	MSE	4.6429	4.1708	3.8830	3.5808	1.2228	0.3780
100	β_2	BIAS	1.9925	0.9823	0.3128	0.1000	0.1786	0.0231
	β_2	MSE	4.3943	1.1847	0.2232	1.5692	0.2000	0.0281
150	β_0	BIAS	2.4333	0.8100	0.1091	0.8422	0.1704	0.0444
	β_0	MSE	6.2249	0.7661	0.0626	1.0472	0.0962	0.0241
150	β_1	BIAS	2.1452	2.0300	1.9409	1.8516	0.8033	0.2552
	β_1	MSE	4.6115	4.1337	3.8142	3.8500	1.6127	0.3681
150	β_2	BIAS	1.9942	0.9553	0.2887	1.0482	0.2273	0.0281
	β_2	MSE	4.2501	1.0431	0.1442	1.4656	0.1958	0.0200
200	β_0	BIAS	2.4556	0.8260	0.1171	0.8758	0.1856	0.0510
	β_0	MSE	6.2478	0.7619	0.0476	1.0360	0.0847	0.0192
200	β_1	BIAS	2.1487	2.0313	1.9319	1.9476	0.8974	0.2640
	β_1	MSE	4.6239	4.1365	3.7705	4.0105	1.8030	0.3753
200	β_2	BIAS	1.9901	0.9414	0.2784	1.0724	0.2496	0.0205
	β_2	MSE	4.1653	0.9828	0.1216	1.4194	0.1884	0.0161

APPENDIX L

Kruskal-Wallis Test on Multiple Regression for 20% Outliers in x and y-axes: Response versus Treatment

```
Kruskal-Wallis Test on Response

Treatment     N   Median  Ave Rank      Z
1            30  3.32360     155.7   7.50
2            30  1.05900     113.3   2.63
3            30  0.31365      76.3  -1.64
4            30  1.25400     106.6   1.85
5            30  0.21365      60.2  -3.49
6            30  0.05475      31.0  -6.86
Overall     180               90.5

H = 107.08  DF = 5  P = 0.000
H = 107.08  DF = 5  P = 0.000  (adjusted for ties)
```

Where
Treatment 1 is OLS
Treatment 2 is Huber estimator
Treatment 3 is Hampel estimator
Treatment 4 is Biweight estimator
Treatment 5 is Alarm estimator
Treatment 6 is the proposed estimator

and the responses are the different values of the MSE and BIAS

APPENDIX LI

Telephone-Call data forNumber of International Calls from Belgium (Rousseeuw and Leroy 1987)

Year x_i	**Number of calls** y_i
50	0.44
51	0.47
52	0.47
53	0.59
54	0.66
55	0.73
56	0.81
57	0.88
58	1.06
59	1.20
60	1.35
61	1.49
62	1.61
63	2.12
64	11.90
65	12.40

66	14.20
67	15.90
68	18.20
69	21.20
70	4.30
71	2.40
72	2.70
73	2.90

APPENDIX LII

Artificial Data Set of Hawkins et al. (1984) (Rousseeuw and Leroy 1987)

Index	x_1	x_2	x_3	Y
1	10.1	19.6	28.3	9.7
2	9.5	20.51	28.9	10.1
3	10.7	20.2	31.0	10.3
4	9.9	21.5	31.7	9.5
5	10.3	21.1	31.1	10.0
6	10.8	20.4	29.2	10.0
7	10.5	20.9	29.1	10.8
8	9.9	19.6	28.8	10.3
9	9.7	20.7	31.0	9.6
10	9.3	19.7	30.3	9.9
11	11.0	24.0	35.0	-0.2
12	12.0	23.0	37.0	-0.4
13	12.0	26.0	34.0	0.7
14	11.0	34.0	34.0	0.1
15	3.4	2.9	2.1	-0.4
16	3.1	2.2	0.3	0.6
17	0.0	1.6	0.2	-0.2
18	2.3	1.6	2.0	0.0
19	0.8	2.9	1.6	0.1
20	3.1	3.4	2.2	0.4
21	2.6	2.2	1.9	0.9
22	0.4	3.2	1.9	0.3
23	2.0	2.3	0.8	-0.8
24	1.3	2.3	0.5	0.7
25	1.0	0.0	0.4	-0.3
26	0.9	3.3	2.5	-0.8
27	3.3	2.5	2.9	-0.7
28	1.8	0.8	2.0	0.3
29	1.2	0.9	0.8	0.3
30	1.2	0.7	3.4	-0.3
31	3.1	1.4	1.0	0.0
32	0.5	2.4	0.3	-0.4
33	1.5	3.1	1.5	-0.6
34	0.4	0.0	0.7	-0.7
35	3.1	2.4	3.0	0.3
36	1.1	2.2	2.7	-1.0
37	0.1	3.0	2.6	-0.6
38	1.5	1.2	0.2	0.9
39	2.1	0.0	1.2	-0.7
40	0.5	2.0	1.2	-0.5
41	3.4	1.6	2.9	-0.1
42	0.3	1.0	2.7	-0.7
43	0.1	3.3	0.9	0.6
44	1.8	0.5	3.2	-0.7
45	1.9	0.1	0.6	-0.5
46	1.8	0.5	3.0	-0.4
47	3.0	0.1	0.8	-0.9

48	3.1	1.6	3.0	0.1
49	3.1	2.5	1.9	0.9
50	2.1	2.8	2.9	-0.4
51	2.3	1.5	0.4	0.7
52	3.3	0.6	1.2	-0.5
53	0.3	0.4	3.3	0.7
54	1.1	3.0	0.3	0.7
55	0.5	2.4	0.9	0.0
56	1.8	3.2	0.9	0.1
57	1.8	0.7	0.7	0.7
58	2.4	3.4	1.5	-0.1
59	1.6	2.1	3.0	-0.3
60	0.3	1.5	3.3	-0.9
61	0.4	3.4	3.0	-0.3
62	0.9	0.1	0.3	0.6
63	1.1	2.7	0.2	-0.3
64	2.8	3.0	2.9	-0.5
65	2.0	0.7	2.7	0.6
66	0.2	1.8	0.8	-0.9
67	1.6	2.0	1.2	-0.7
68	0.1	0.0	1.1	0.6
69	2.0	0.6	0.3	0.2
70	1.0	2.2	2.9	0.7
71	2.2	2.5	2.3	0.2
72	0.6	2.0	1.5	-0.2
73	0.3	1.7	2.2	0.4
74	0.0	2.2	1.6	-0.9
75	0.3	0.4	2.6	0.2

APPENDIX LIII

R PROGRAM FOR THE ESTIMATION OF PARAMETERS FOR ROBUST REGRESSION ON BELGIUM PHONE DATA

```
# Belgian Phone Data
sink("Phone-Data.txt") #     Write     output     in     the     file
"DWMCase1.1Results.txt" inside my document
library(MASS)
data(phones) # Belgium Phone data
attach(phones)
Y<-calls
X<-year
Y
X
HuberM<-function(Y,X){
############### Improved brute-force IRWLS - Huber Method
# Brownlee's Stack Loss Data
# Robust Regression and Outlier Detection p. 76, Rousseeuw & Leroy,
1989.
# Robust Estimation and Testing p. 216, Staudte & Sheather, 1990.
library(MASS)
library(robustbase)
n <- length(Y)
w <- rep(1,n)
irwls.1 <- lm(Y ~ X)
res1 <- residuals(irwls.1)
b.old <- coef(irwls.1)
MAD <- mad(res1)
u <- res1/MAD
for(i in 1:n)  w[i] <- min(1,1.345/abs(u[i]))
delta_b <- 100.0
num.iter=0
while (delta_b > 0.000001) {
num.iter <- num.iter + 1
irwls.2 <- lm(Y ~ X,weights=w)
res2 <- residuals(irwls.2)
b.new <- coef(irwls.2)
MAD <- mad(res2)
u <- res2/MAD
for(i in 1:n)  w[i] <- min(1,1.345/abs(u[i]))
delta_b <- max(abs((b.new-b.old)/b.old))
b.old <- b.new
if(num.iter>100){break}
}
Out<-list(num.iter=num.iter,w=w,irwls.2=irwls.2)
}

BisquareM<-function(Y,X){
library(MASS)
library(robustbase)
n <- length(Y)
w <- rep(1,n)
M.Huber<-HuberM(Y,X)
M.H.Model<-M.Huber$irwls.2
res1 <- residuals(M.H.Model)
b.old <- M.H.Model$coef
MAD <- mad(res1)
u <- res1/MAD
for(i in 1:n){
  if(abs(u[i])< 4.685){
      w[i]<-(1-(u[i]/4.685)^2)^2
  }else{
```

```
      w[i]<-0
    }
  }
  delta_b <- 100.0
  num.iter=0
  while (delta_b > 0.000001) {
  num.iter <- num.iter + 1
  irwls.2 <- lm(Y ~ X,weights=w)
  res2 <- residuals(irwls.2)
  b.new <- coef(irwls.2)
  MAD <- mad(res2)
  u <- res2/MAD
  for(i in 1:n){
    if(abs(u[i])< 4.685){
        w[i]<-(1-(u[i]/4.685)^2)^2
    }else{
        w[i]<-0
    }
  }
  delta_b <- max(abs((b.new-b.old)/b.old))
  b.old <- b.new
  if(num.iter>100){break}
  }
  Out<-list(num.iter=num.iter,w=w,irwls.2=irwls.2)
  }

AlamgirM<-function(Y,X){  #  ALAMGIR  Bisquare  using  LTS  as  initial
estimate
library(MASS)
library(robustbase)
n <- length(Y)
w <- rep(1,n)
irwls.1 <- ltsReg(x=X,y=Y)
res1 <- residuals(irwls.1)
b.old <- coef(irwls.1)
MAD <- mad(res1)
u <- res1/MAD
for(i in 1:n){
  if(abs(u[i])< 4.685){
      w[i]<-(16*exp(-2*(u[i]/3)^2))/(1+exp(-(u[i]/3)^2))^4
  }else{
      w[i]<-0
  }
}
delta_b <- 100.0
num.iter=0
while (delta_b > 0.000001) {
num.iter <- num.iter + 1
irwls.2 <- lm(Y ~ X,weights=w)
res2 <- residuals(irwls.2)
b.new <- coef(irwls.2)
MAD <- mad(res2)
u <- res2/MAD
for(i in 1:n){
  if(abs(u[i])< 4.685){
      w[i]<-(16*exp(-2*(u[i]/3)^2))/(1+exp(-(u[i]/3)^2))^4
  }else{
      w[i]<-0
```

```
  }
}
delta_b <- max(abs((b.new-b.old)/b.old))
b.old <- b.new
if(num.iter>100){break}
}
Out<-list(num.iter=num.iter,w=w,irwls.2=irwls.2)
}

StellaM<-function(Y,X){
############### Improved brute-force IRWLS - Huber Method
library(MASS)
library(robustbase)
n <- length(Y)
w <- rep(1,n)
irwls.1 <- lmsreg(Y ~ X)
res1 <- residuals(irwls.1)
b.old <- coef(irwls.1)
MAD <- mad(res1)
u <- res1/MAD
for(i in 1:n){
  if(abs(u[i])< 3){
      w[i]<-((1+((u[i]/3)^2)^2)*(1-((u[i]/3)^2)^2))
  }else{
      w[i]<-0
  }
}
delta_b <- 100.0
num.iter=0
while (delta_b > 0.000001) {
num.iter <- num.iter + 1
irwls.2 <- lm(Y ~ X,weights=w)
res2 <- residuals(irwls.2)
b.new <- coef(irwls.2)
MAD <- mad(res2)
u <- res2/MAD
for(i in 1:n){
  if(abs(u[i])< 3){
      w[i]<-((1+((u[i]/3)^2)^2)*(1-((u[i]/3)^2)^2))
  }else{
      w[i]<-0
  }
}
delta_b <- max(abs((b.new-b.old)/b.old))
b.old <- b.new
if(num.iter>100){break}
}
Out<-list(num.iter=num.iter,w=w,irwls.2=irwls.2)
}

M.OLS<-lm(Y~X)
summary(M.OLS)

M.Huber<-HuberM(Y,X)
num.iter1<-M.Huber$num.iter
num.iter1
W1<-M.Huber$w
W1
```

```
summary(M.Huber$irwls.2)

M.Bisquare<-BisquareM(Y,X)
num.iter2<-M.Bisquare$num.iter
num.iter2
W2<-M.Bisquare$w
W2
summary(M.Bisquare$irwls.2)

M.Hampel<-rlm(Y~X, psi = psi.hampel, init = "lts",maxit=100)
W<-M.Hampel$w
W
summary(M.Hampel)

M.Alamgir<-AlamgirM(Y,X)
num.iter3<-M.Alamgir$num.iter
num.iter3
W3<-M.Alamgir$w
W3
summary(M.Alamgir$irwls.2)

M.Stella<-StellaM(Y,X)
num.iter4<-M.Stella$num.iter
num.iter4
W4<-M.Stella$w
W4
summary(M.Stella$irwls.2)
sink()
```

APPENDIX LIV

R PROGRAM FOR THE ESTIMATION OF PARAMETERS FOR ROBUST REGRESSION ON HAWKINS-BRADU-KASS DATA

```
# Hawkins-Bradu-Kass data (Rousseeuw & Leroy, 1987, p. 94)
sink("Hawkins-Bradu-Kass data.txt")
ImportDataY<-read.table("J:/RData/MphilData5YImport.txt",header=T)
ImportDataX1<-read.table("J:/RData/MphilData5X1Import.txt",header=T)
ImportDataX2<-read.table("J:/RData/MphilData5X2Import.txt",header=T)
ImportDataX3<-read.table("J:/RData/MphilData5X3Import.txt",header=T)
Y<-ImportDataY$Y
X1<-ImportDataX1$X1
X2<-ImportDataX2$X2
X3<-ImportDataX3$X3
Y
X1
X2
X3

HuberM<-function(Y,X1,X2,X3){
############### Improved brute-force IRWLS - Huber Method
library(MASS)
library(robustbase)
library(quantreg)
n <- length(Y)
w <- rep(1,n)
irwls.1 <- lm(Y ~ X1+X2+X3)
res1 <- residuals(irwls.1)
b.old <- coef(irwls.1)
MAD <- mad(res1)
u <- res1/MAD
for(i in 1:n)  w[i] <- ifelse(abs(u[i])<=1.345,1,1.345/abs(u[i]))
delta_b <- 100.0
num.iter=0
while (delta_b > 0.000001) {
num.iter <- num.iter + 1
irwls.2 <- lm(Y ~ X1+X2+X3,weights=w)
res2 <- residuals(irwls.2)
b.new <- coef(irwls.2)
MAD <- mad(res2)
u <- res2/MAD
for(i in 1:n)  w[i] <- min(1,1.345/abs(u[i]))
delta_b <- max(abs((b.new-b.old)/b.old))
b.old <- b.new
if(num.iter>100){break}
}
Out<-list(num.iter=num.iter,w=w,irwls.2=irwls.2)
}

BisquareM<-function(Y,X1,X2,X3){  #Bisquare  using  Huber  as  the  Initial
estimator
library(MASS)
library(robustbase)
library(quantreg)
n <- length(Y)
w <- rep(1,n)
M.Huber<-HuberM(Y,X1,X2,X3)
M.H.Model<-M.Huber$irwls.2
res1 <- residuals(M.H.Model)
b.old <- M.H.Model$coef
```

```
MAD <- mad(res1)
u <- res1/MAD
for(i in 1:n){
  if(abs(u[i])< 4.685){
      w[i]<-(1-(u[i]/4.685)^2)^2
  }else{
      w[i]<-0
  }
}
delta_b <- 100.0
num.iter=0
while (delta_b > 0.000001) {
num.iter <- num.iter + 1
irwls.2 <- lm(Y ~ X1+X2+X3,weights=w)
res2 <- residuals(irwls.2)
b.new <- coef(irwls.2)
MAD <- mad(res2)
u <- res2/MAD
for(i in 1:n){
  if(abs(u[i])< 4.685){
      w[i]<-(1-(u[i]/4.685)^2)^2
  }else{
      w[i]<-0
  }
}
delta_b <- max(abs((b.new-b.old)/b.old))
b.old <- b.new
if(num.iter>100){break}
}
Out<-list(num.iter=num.iter,w=w,irwls.2=irwls.2)
}

AlamgirM<-function(Y,X1,X2,X3){ # ALAMGIR Bisquare using LTS as initial
estimate
library(MASS)
library(robustbase)
n <- length(Y)
w <- rep(1,n)
Boundx<-cbind(X1,X2,X3)
irwls.1 <- ltsReg(x=Boundx,y=Y)
res1 <- residuals(irwls.1)
b.old <- coef(irwls.1)
MAD <- mad(res1)
u <- res1/MAD
for(i in 1:n){
  if(abs(u[i])< 4.685){
      w[i]<-(16*exp(-2*(u[i]/3)^2))/(1+exp(-(u[i]/3)^2))^4
  }else{
      w[i]<-0
  }
}
delta_b <- 100.0
num.iter=0
while (delta_b > 0.000001) {
num.iter <- num.iter + 1
irwls.2 <- lm(Y ~ X1+X2+X3,weights=w)
res2 <- residuals(irwls.2)
b.new <- coef(irwls.2)
```

```
MAD <- mad(res2)
u <- res2/MAD
for(i in 1:n){
  if(abs(u[i])< 4.685){
      w[i]<-(16*exp(-2*(u[i]/3)^2))/(1+exp(-(u[i]/3)^2))^4
  }else{
      w[i]<-0
  }
}
delta_b <- max(abs((b.new-b.old)/b.old))
b.old <- b.new
if(num.iter>100){break}
}
Out<-list(num.iter=num.iter,w=w,irwls.2=irwls.2)
}

StellaM<-function(Y,X1,X2,X3){ # LMS as the initial estimator
############### Improved brute-force IRWLS - Stella Method
library(MASS)
library(robustbase)
n <- length(Y)
w <- rep(1,n)
Boundx<-cbind(X1,X2,X3)
irwls.1 <- lmsreg(x=Boundx,y=Y)
res1 <- residuals(irwls.1)
b.old <- coef(irwls.1)
MAD <- mad(res1)
u <- res1/MAD
for(i in 1:n){
  if(abs(u[i])< 3){
      w[i]<-((1+((u[i]/3)^2)^2)*(1-((u[i]/3)^2)^2))
  }else{
      w[i]<-0
  }
}
delta_b <- 100.0
num.iter=0
while (delta_b > 0.000001) {
num.iter <- num.iter + 1
irwls.2 <- lm(Y ~ X1+X2+X3,weights=w)
res2 <- residuals(irwls.2)
b.new <- coef(irwls.2)
MAD <- mad(res2)
u <- res2/MAD
for(i in 1:n){
  if(abs(u[i])< 3){
      w[i]<-((1+((u[i]/3)^2)^2)*(1-((u[i]/3)^2)^2))
  }else{
      w[i]<-0
  }
}
delta_b <- max(abs((b.new-b.old)/b.old))
b.old <- b.new
if(num.iter>100){break}
}
Out<-list(num.iter=num.iter,w=w,irwls.2=irwls.2)
}
```

```
M.OLS<-lm(Y~X1+X2+X3)
summary(M.OLS)

M.Huber<-HuberM(Y,X1,X2,X3)
num.iter1<-M.Huber$num.iter
num.iter1
W1<-M.Huber$w
W1
summary(M.Huber$irwls.2)

M.Bisquare<-BisquareM(Y,X1,X2,X3)
num.iter2<-M.Bisquare$num.iter
num.iter2
W2<-M.Bisquare$w
W2
summary(M.Bisquare$irwls.2)

M.Hampel<-rlm(Y~X1+X2+X3, psi = psi.hampel, init = "lts",maxit=100)
W<-M.Hampel$w
W
summary(M.Hampel)

M.Alamgir<-AlamgirM(Y,X1,X2,X3)
num.iter3<-M.Alamgir$num.iter
num.iter3
W3<-M.Alamgir$w
W3
summary(M.Alamgir$irwls.2)

M.Stella<-StellaM(Y,X1,X2,X3)
num.iter4<-M.Stella$num.iter
num.iter4
W4<-M.Stella$w
W4
summary(M.Stella$irwls.2)
sink()
```

APPENDIX LV

RESULT FOR ESTIMATION OF PARAMETERS FOR ROBUST REGRESSION ON BELGIUM PHONE DATA

```
> library(MASS)

> data(phones) # Belgium Phone data

> attach(phones)

> Y<-calls

> X<-year

> Y
 [1]   4.4   4.7   4.7   5.9   6.6   7.3   8.1   8.8  10.6  12.0  13.5
14.9
[13]  16.1  21.2 119.0 124.0 142.0 159.0 182.0 212.0  43.0  24.0  27.0
29.0

> X
 [1] 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71
72 73

> HuberM<-function(Y,X){
+ library(MASS)
+ library(robustbase)
+ n <- length(Y)
+ w <- rep(1,n)
+ irwls.1 <- lm(Y ~ X)
+ res1 <- residuals(irwls.1)
+  .... [TRUNCATED]

> BisquareM<-function(Y,X){
+ library(MASS)
+ library(robustbase)
+ n <- length(Y)
+ w <- rep(1,n)
+ M.Huber<-HuberM(Y,X)
+ M.H.Model<-M.Huber$irwls.2 .... [TRUNCATED]

>  AlamgirM<-function(Y,X){  #  ALAMGIR  Bisquare  using  LTS  as  initial
estimate
+ library(MASS)
+ library(robustbase)
+ n <- length(Y)
+ w <- rep(1,n)
+ i .... [TRUNCATED]

> StellaM<-function(Y,X){
+ library(MASS)
+ library(robustbase)
+ n <- length(Y)
+ w <- rep(1,n)
+ irwls.1 <- lmsreg(Y ~ X)
+ res1 <- residuals(irwls. .... [TRUNCATED]

> M.OLS<-lm(Y~X)
```

```
> summary(M.OLS)

Call:
lm(formula = Y ~ X)

Residuals:
   Min     1Q Median     3Q    Max 
-78.97 -33.52 -12.04  23.38 124.20 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)   
(Intercept) -260.059    102.607  -2.535   0.0189 * 
X              5.041      1.658   3.041   0.0060 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 56.22 on 22 degrees of freedom
Multiple R-squared:  0.2959, Adjusted R-squared:  0.2639 
F-statistic: 9.247 on 1 and 22 DF,  p-value: 0.005998


> M.Huber<-HuberM(Y,X)

> num.iter1<-M.Huber$num.iter

> num.iter1
[1] 24

> W1<-M.Huber$w

> W1
 [1] 1.00000000 1.00000000 1.00000000 1.00000000 1.00000000 1.00000000
 [7] 1.00000000 1.00000000 1.00000000 1.00000000 1.00000000 1.00000000
[13] 1.00000000 1.00000000 0.11104192 0.10751065 0.09196664 0.08098849
[19] 0.06939420 0.05827282 1.00000000 0.59280225 0.62993609 0.63043767

> summary(M.Huber$irwls.2)

Call:
lm(formula = Y ~ X, weights = w)

Weighted Residuals:
    Min      1Q  Median      3Q     Max 
-13.229  -5.458  -1.444  11.352  42.194 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)   
(Intercept) -99.9045    41.6690  -2.398  0.02543 * 
X             1.9871     0.6992   2.842  0.00948 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 19.51 on 22 degrees of freedom
Multiple R-squared:  0.2686, Adjusted R-squared:  0.2353 
F-statistic: 8.078 on 1 and 22 DF,  p-value: 0.009482
> M.Bisquare<-BisquareM(Y,X)

> num.iter2<-M.Bisquare$num.iter
```

```
> num.iter2
[1] 7

> W2<-M.Bisquare$w

> W2
 [1]  0.9112256  0.9723318  0.9996748  0.9999989  0.9953509  0.9817543
0.9658043
 [8]  0.9370709  0.9818714  0.9929023  0.9997155  0.9988607  0.9974279
0.5445502
[15]  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000
0.0000000
[22] 0.9206967 0.9987500 0.9653544

> summary(M.Bisquare$irwls.2)

Call:
lm(formula = Y ~ X, weights = w)

Weighted Residuals:
    Min      1Q  Median      3Q     Max
-1.6223 -0.4282  0.0000  0.2029  3.1749

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -52.34782    2.62364  -19.95 3.27e-12 ***
X             1.09913    0.04407   24.94 1.26e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.237 on 15 degrees of freedom
Multiple R-squared:  0.9765, Adjusted R-squared:  0.9749
F-statistic: 622.2 on 1 and 15 DF,  p-value: 1.259e-13


> M.Hampel<-rlm(Y~X, psi = psi.hampel, init = "lts",maxit=100)

> W<-M.Hampel$w

> W
        1         2         3         4         5         6         7
8
1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
1.0000000
        9        10        11        12        13        14        15
16
1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 0.7640701 0.0000000
0.0000000
       17        18        19        20        21        22        23
24
0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 1.0000000 1.0000000
1.0000000

> summary(M.Hampel)

Call: rlm(formula = Y ~ X, psi = psi.hampel, init = "lts", maxit = 100)
Residuals:
```

```
     Min       1Q   Median       3Q      Max
 -1.7612  -0.4750   0.1955  38.9906 188.4402

Coefficients:
            Value    Std. Error t value
(Intercept) -52.3891   3.0149   -17.3766
X             1.1007   0.0487    22.5946

Residual standard error: 1.621 on 22 degrees of freedom

> M.Alamgir<-AlamgirM(Y,X)

> num.iter3<-M.Alamgir$num.iter

> num.iter3
[1] 3

> W3<-M.Alamgir$w

> W3
 [1]  0.9940388  0.9994835  0.9999997  1.0000000  0.9999724  0.9996546
0.9988473
 [8]  0.9962180  0.9996446  0.9999344  0.9999995  0.9999998  0.9999983
0.8194307
[15]  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000
0.0000000
[22] 0.9934787 0.9999999 0.9993536

> summary(M.Alamgir$irwls.2)

Call:
lm(formula = Y ~ X, weights = w)

Weighted Residuals:
    Min      1Q  Median      3Q     Max
-1.7857 -0.4842  0.0000  0.1121  3.8245

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -52.45432    2.88126   -18.2 1.23e-11 ***
X             1.10205    0.04834    22.8 4.70e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.383 on 15 degrees of freedom
Multiple R-squared:  0.9719, Adjusted R-squared:  0.9701
F-statistic: 519.8 on 1 and 15 DF,  p-value: 4.695e-13


> M.Stella<-StellaM(Y,X)

> num.iter4<-M.Stella$num.iter

> num.iter4
[1] 4

> W4<-M.Stella$w
```

```
> W4
 [1]  0.9998585  0.9999990  1.0000000  1.0000000  1.0000000  0.9999995
0.9999945
 [8]  0.9999413  0.9999995  1.0000000  1.0000000  1.0000000  1.0000000
0.8368835
[15]  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000
0.0000000
[22] 0.9998242 1.0000000 0.9999984

> summary(M.Stella$irwls.2)

Call:
lm(formula = Y ~ X, weights = w)

Weighted Residuals:
    Min      1Q  Median      3Q     Max
-1.7964 -0.4877  0.0000  0.1072  3.8611

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -52.45588    2.89637  -18.11 1.32e-11 ***
X             1.10215    0.04859   22.68 5.06e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.392 on 15 degrees of freedom
Multiple R-squared:  0.9717, Adjusted R-squared:  0.9698
F-statistic: 514.5 on 1 and 15 DF,  p-value: 5.056e-13


> sink()
```

APPENDIX LVI

RESULT FOR ESTIMATION OF PARAMETERS FOR ROBUST REGRESSION ON HAWKINS-BRADU-KASS DATA

```
> ImportDataY<-read.table("J:/RData/MphilData5YImport.txt",header=T)

> ImportDataX1<-read.table("J:/RData/MphilData5X1Import.txt",header=T)

> ImportDataX2<-read.table("J:/RData/MphilData5X2Import.txt",header=T)

> ImportDataX3<-read.table("J:/RData/MphilData5X3Import.txt",header=T)

> Y<-ImportDataY$Y

> X1<-ImportDataX1$X1


> X2<-ImportDataX2$X2

> X3<-ImportDataX3$X3

> Y
[1]   9.7 10.1 10.3  9.5 10.0 10.0 10.8 10.3  9.6  9.9 -0.2 -0.4  0.7
0.1 -0.4
[16]  0.6 -0.2  0.0  0.1  0.4  0.9  0.3 -0.8  0.7 -0.3 -0.8 -0.7  0.3
0.3 -0.3
[31]  0.0 -0.4 -0.6 -0.7  0.3 -1.0 -0.6  0.9 -0.7 -0.5 -0.1 -0.7  0.6 -
0.7 -0.5
[46] -0.4 -0.9  0.1  0.9 -0.4  0.7 -0.5  0.7  0.7  0.0  0.1  0.7 -0.1 -
0.3 -0.9
[61] -0.3  0.6 -0.3 -0.5  0.6 -0.9 -0.7  0.6  0.2  0.7  0.2 -0.2  0.4 -
0.9  0.2

> X1
 [1] 10.1  9.5 10.7  9.9 10.3 10.8 10.5  9.9  9.7  9.3 11.0 12.0 12.0
11.0  3.4
[16]  3.1  0.0  2.3  0.8  3.1  2.6  0.4  2.0  1.3  1.0  0.9  3.3  1.8
1.2  1.2
[31]  3.1  0.5  1.5  0.4  3.1  1.1  0.1  1.5  2.1  0.5  3.4  0.3  0.1
1.8  1.9
[46]  1.8  3.0  3.1  3.1  2.1  2.3  3.3  0.3  1.1  0.5  1.8  1.8  2.4
1.6  0.3
[61]  0.4  0.9  1.1  2.8  2.0  0.2  1.6  0.1  2.0  1.0  2.2  0.6  0.3
0.0  0.3

> X2
 [1] 19.6 20.5 20.2 21.5 21.1 20.4 20.9 19.6 20.7 19.7 24.0 23.0 26.0
34.0  2.9
[16]  2.2  1.6  1.6  2.9  3.4  2.2  3.2  2.3  2.3  0.0  3.3  2.5  0.8
0.9  0.7
[31]  1.4  2.4  3.1  0.0  2.4  2.2  3.0  1.2  0.0  2.0  1.6  1.0  3.3
0.5  0.1
[46]  0.5  0.1  1.6  2.5  2.8  1.5  0.6  0.4  3.0  2.4  3.2  0.7  3.4
2.1  1.5
[61]  3.4  0.1  2.7  3.0  0.7  1.8  2.0  0.0  0.6  2.2  2.5  2.0  1.7
2.2  0.4

> X3
```

```
 [1] 28.3 28.9 31.0 31.7 31.1 29.2 29.1 28.8 31.0 30.3 35.0 37.0 34.0
34.0  2.1
[16]  0.3  0.2  2.0  1.6  2.2  1.9  1.9  0.8  0.5  0.4  2.5  2.9  2.0
0.8  3.4
[31]  1.0  0.3  1.5  0.7  3.0  2.7  2.6  0.2  1.2  1.2  2.9  2.7  0.9
3.2  0.6
[46]  3.0  0.8  3.0  1.9  2.9  0.4  1.2  3.3  0.3  0.9  0.9  0.7  1.5
3.0  3.3
[61]  3.0  0.3  0.2  2.9  2.7  0.8  1.2  1.1  0.3  2.9  2.3  1.5  2.2
1.6  2.6

> HuberM<-function(Y,X1,X2,X3){
+ library(MASS)
+ library(robustbase)
+ library(quantreg)
+ n <- length(Y)
+ w <- rep(1,n)
+ irwls.1 <- lm(Y ~ X1+X2+X .... [TRUNCATED]

> BisquareM<-function(Y,X1,X2,X3){ #Bisquare using Huber as the Initial
estimator
+ library(MASS)
+ library(robustbase)
+ library(quantreg)
+ n <- len .... [TRUNCATED]

>  AlamgirM<-function(Y,X1,X2,X3){  #  ALAMGIR  Bisquare  using  LTS  as
initial estimate
+ library(MASS)
+ library(robustbase)
+ n <- length(Y)
+ w <- rep(1 .... [TRUNCATED]

> StellaM<-function(Y,X1,X2,X3){ # LMS as the initial estimator
+ library(MASS)
+ library(robustbase)
+ n <- length(Y)
+ w <- rep(1,n)
+ Boundx<-cbind .... [TRUNCATED]

> M.OLS<-lm(Y~X1+X2+X3)

> summary(M.OLS)

Call:
lm(formula = Y ~ X1 + X2 + X3)

Residuals:
    Min      1Q  Median      3Q     Max
-9.3717 -0.7162 -0.0230  0.7083  4.5130

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.3875     0.4165  -0.930  0.35527
X1            0.2392     0.2625   0.911  0.36521
X2           -0.3345     0.1551  -2.158  0.03434 *
X3            0.3833     0.1288   2.976  0.00399 **
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.25 on 71 degrees of freedom
Multiple R-squared:  0.6018, Adjusted R-squared:  0.585
F-statistic: 35.77 on 3 and 71 DF,  p-value: 3.382e-14


> M.Huber<-HuberM(Y,X1,X2,X3)

> num.iter1<-M.Huber$num.iter

> num.iter1
[1] 13

> W1<-M.Huber$w

> W1
 [1]  1.0000000  1.0000000  1.0000000  1.0000000  1.0000000  1.0000000
0.7351727
 [8]  0.8999258  1.0000000  1.0000000  0.1119302  0.1034247  0.1229838
0.1172345
[15]  1.0000000  1.0000000  1.0000000  1.0000000  1.0000000  1.0000000
1.0000000
[22]  1.0000000  1.0000000  1.0000000  1.0000000  1.0000000  0.9589455
1.0000000
[29]  1.0000000  1.0000000  1.0000000  1.0000000  1.0000000  1.0000000
1.0000000
[36]  1.0000000  1.0000000  0.9076738  1.0000000  1.0000000  1.0000000
1.0000000
[43]  1.0000000  1.0000000  1.0000000  1.0000000  1.0000000  1.0000000
1.0000000
[50]  1.0000000  1.0000000  1.0000000  1.0000000  1.0000000  1.0000000
1.0000000
[57]  1.0000000  1.0000000  1.0000000  1.0000000  1.0000000  1.0000000
1.0000000
[64]  1.0000000  1.0000000  1.0000000  1.0000000  1.0000000  1.0000000
1.0000000
[71] 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000

> summary(M.Huber$irwls.2)

Call:
lm(formula = Y ~ X1 + X2 + X3, weights = w)

Weighted Residuals:
    Min      1Q  Median      3Q     Max 
-3.8439 -0.6060  0.0296  0.5871  1.4418 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) -0.776070   0.214974  -3.610 0.000567 ***
X1           0.166850   0.132859   1.256 0.213290    
X2           0.007474   0.110771   0.067 0.946395    
X3           0.274448   0.080011   3.430 0.001009 ** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.127 on 71 degrees of freedom
```

```
Multiple R-squared:  0.8957, Adjusted R-squared:  0.8913
F-statistic: 203.3 on 3 and 71 DF,  p-value: < 2.2e-16


> M.Bisquare<-BisquareM(Y,X1,X2,X3)

> num.iter2<-M.Bisquare$num.iter

> num.iter2
[1] 8

> W2<-M.Bisquare$w

> W2
 [1]  0.9939720  0.9773817  0.9980465  0.8854780  0.9868551  0.9988743
0.9057194
 [8]  0.9265821  0.9629399  0.9970605  0.0000000  0.0000000  0.0000000
0.0000000
[15]  0.8964244  0.9516652  0.9796067  0.9994628  0.9993674  0.9961551
0.9376559
[22]  0.9939671  0.9285973  0.8928009  0.9758829  0.8501435  0.8047175
0.9714699
[29]  0.9270763  0.9897096  0.9997870  0.9996106  0.9263895  0.9995026
0.9937466
[36]  0.8371825  0.9309373  0.7719823  0.9900352  0.9925488  0.9691196
0.9695384
[43]  0.9330707  0.9379467  0.9997619  0.9834423  0.9600225  0.9910918
0.9588698
[50]  0.9102729  0.8856282  0.9822766  0.8894551  0.9088588  0.9925627
0.9999904
[57]  0.8410013  0.9742345  0.9616604  0.8983520  0.9492314  0.7764359
0.9991050
[64]  0.8532727  0.9481062  0.9694580  0.9534693  0.7791459  0.9392438
0.9613098
[71] 0.9991358 0.9999948 0.9575308 0.9403893 0.9598222

> summary(M.Bisquare$irwls.2)

Call:
lm(formula = Y ~ X1 + X2 + X3, weights = w)

Weighted Residuals:
    Min      1Q  Median      3Q     Max
-1.1199 -0.4765  0.0000  0.5381  1.1910

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.94553    0.12574  -7.519 1.77e-10 ***
X1           0.14482    0.07675   1.887 0.063511 .
X2           0.19722    0.06908   2.855 0.005723 **
X3           0.18034    0.04879   3.696 0.000442 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.6331 on 67 degrees of freedom
Multiple R-squared:  0.9688, Adjusted R-squared:  0.9674
F-statistic: 692.7 on 3 and 67 DF,  p-value: < 2.2e-16
```

```
> M.Hampel<-rlm(Y~X1+X2+X3, psi = psi.hampel, init = "lts",maxit=100)

> W<-M.Hampel$w

> W
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
25 26
 0  0  0  0  0  0  0  0  0  0  1  1  1  1  1  1  1  1  1  1  1  1  1  1
1  1
27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50
51 52
 1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
1  1
53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75
 1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1

> summary(M.Hampel)

Call: rlm(formula = Y ~ X1 + X2 + X3, psi = psi.hampel, init = "lts",
    maxit = 100)
Residuals:
     Min       1Q   Median       3Q      Max
-0.92633 -0.39554  0.05279  0.71373 10.79551

Coefficients:
            Value   Std. Error t value
(Intercept) -0.1805  0.1112    -1.6225
X1           0.0814  0.0701     1.1611
X2           0.0399  0.0414     0.9636
X3          -0.0517  0.0344    -1.5018

Residual standard error: 0.7719 on 71 degrees of freedom

> M.Alamgir<-AlamgirM(Y,X1,X2,X3)

> num.iter3<-M.Alamgir$num.iter

> num.iter3
[1] 3

> W3<-M.Alamgir$w

> W3
 [1]  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000
0.0000000
 [8]  0.0000000  0.0000000  0.0000000  0.9999998  0.9999801  0.9981779
0.9999740
[15]  0.9992209  0.9994795  0.9999997  1.0000000  0.9999865  0.9998940
0.9927972
[22]  0.9996270  0.9941995  0.9969492  0.9999874  0.9971852  0.9964156
0.9996717
[29]  0.9997239  0.9999997  0.9999996  0.9998381  0.9985583  0.9991481
0.9999174
[36]  0.9935060  0.9996508  0.9913357  0.9981181  0.9997544  0.9999982
0.9995328
[43]  0.9972979  0.9991161  0.9995174  0.9999660  0.9911314  0.9999975
0.9943742
```

```
[50]  0.9998141  0.9977887  0.9988916  0.9874077  0.9973108  0.9999992
0.9999999
[57]  0.9963307  0.9999891  0.9999879  0.9980691  0.9999963  0.9967672
0.9998934
[64]  0.9991341  0.9965738  0.9958509  0.9976034  0.9942725  0.9999768
0.9933486
[71] 0.9999713 0.9999997 0.9984039 0.9966841 0.9993813

> summary(M.Alamgir$irwls.2)

Call:
lm(formula = Y ~ X1 + X2 + X3, weights = w)

Weighted Residuals:
    Min      1Q  Median      3Q     Max 
-0.9223 -0.3947  0.0000  0.3972  1.0054 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)  
(Intercept) -0.18125    0.10435  -1.737   0.0874 .
X1           0.08172    0.06660   1.227   0.2246  
X2           0.04002    0.04041   0.990   0.3259  
X3          -0.05185    0.03532  -1.468   0.1473  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.556 on 61 degrees of freedom
Multiple R-squared:  0.04317,     Adjusted R-squared:  -0.003889 
F-statistic: 0.9173 on 3 and 61 DF,  p-value: 0.4379


> M.Stella<-StellaM(Y,X1,X2,X3)

> num.iter4<-M.Stella$num.iter

> num.iter4
[1] 4

> W4<-M.Stella$w

> W4
 [1]  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000
0.0000000
 [8]  0.0000000  0.0000000  0.0000000  1.0000000  1.0000000  0.9999867
1.0000000
[15]  0.9999976  0.9999989  1.0000000  1.0000000  1.0000000  1.0000000
0.9997909
[22]  0.9999995  0.9998646  0.9999627  1.0000000  0.9999680  0.9999485
0.9999996
[29]  0.9999997  1.0000000  1.0000000  0.9999999  0.9999917  0.9999971
1.0000000
[36]  0.9998292  0.9999995  0.9996973  0.9999858  0.9999998  1.0000000
0.9999991
[43]  0.9999709  0.9999968  0.9999991  1.0000000  0.9996826  1.0000000
0.9998724
[50]  0.9999999  0.9999804  0.9999951  0.9993628  0.9999710  1.0000000
1.0000000
```

```
[57]  0.9999461  1.0000000  1.0000000  0.9999849  1.0000000  0.9999583
1.0000000
[64]  0.9999970  0.9999532  0.9999304  0.9999769  0.9998692  1.0000000
0.9998229
[71] 1.0000000 1.0000000 0.9999899 0.9999555 0.9999985

> summary(M.Stella$irwls.2)

Call:
lm(formula = Y ~ X1 + X2 + X3, weights = w)

Weighted Residuals:
    Min      1Q  Median      3Q     Max
-0.9262 -0.3955  0.0000  0.3968  1.0103

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.18050    0.10444  -1.728    0.089 .
X1           0.08140    0.06667   1.221    0.227
X2           0.03991    0.04047   0.986    0.328
X3          -0.05168    0.03537  -1.461    0.149
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5572 on 61 degrees of freedom
Multiple R-squared:  0.0428, Adjusted R-squared:  -0.004273
F-statistic: 0.9092 on 3 and 61 DF,  p-value: 0.4419


> sink()
```