DETECTION OF MULTIPLE OUTLIERS IN LINEAR REGRESSION USING NONPARAMETRIC METHODS

BY

ROBIAH ADNAN

MAIZAH HURA AHMAD

SITI ZANARIAH SATARI

RESEARCH VOT NO:

75021

Universiti Teknologi Malaysia

2005

ABSTRACT

.There has been considerable interest in recent years in the detection and accommodation of multiple outliers in linear regression. However, most of them are complicated and unappealing to users with no mathematical background. The clustering algorithm from Sebert et al. (1998) is discussed and used since it is easy to understand with interesting proposed approach and have a good performance in detecting the presence of outliers. Generally, method proposed by Sebert et al. (1998) is based on the use of a single linkage clustering algorithm with the Euclidean distances to cluster the points in the plots of standard predicted versus residuals values from a linear regression model. The predicted and residual values are obtained from an ordinary least squares fit of the data. The algorithm is described and is shown to perform well on classic multiple outlier data sets. A modification is done to the Sebert's method by replacing the least squares (LS) with two robust estimators. Method 1 is a modification of Sebert's method where the least squares (LS) fit is replaced by the least median of squares (LMS) fit while Method 2 is a modification of Sebert's method where the least squares (LS) fit is replaced by the least trimmed of squares (LTS) fit. This research also provides a comparison between these three procedures to help and give future researchers a comprehensive view about the best procedure to detect multiple outliers. A Monte Carlo simulation study was used to evaluate the effectiveness of these three procedures. All simulations and calculation were done using statistical package
S-PLUS 2000.

ABSTRAK

Kebelakangan ini, terdapat minat dan kecenderungan yang tinggi kepada pengesanan, pengecaman dan penyesuaian terhadap data terpencil berganda dalam regresi linear. Walaubagaimanapun, kebanyakan daripada kaedah yang diperkenalkan adalah rumit dan tidak dapat menarik mereka yang tidak mempunyai latar belakang matematik untuk menggunakannya. 'Algoritma Berkelompok' daripada Sebert dll. (1998) akan dibicangkan dan digunakan dalam kajian ini kerana kaedah ini mudah difahami dengan pendekatan yang menarik dan berkesan dalam pengesanan titik terpencil. Secara umumnya, kaedah yang dicadangkan oleh Sebert dll. (1998) ini adalah berdasarkan kepada penggunaan kaedah pautan tunggal berkelompok bersama-sama dengan jarak Euclidean bagi mengelompokkan titik-titik dalam plot antara nilai ramalan dan reja piawai model regresi linear. Nilai ramalan dan reja piawai ini diperolehi daripada kaedah penyesuaian kuasa dua terkecil (LS). 'Algoritma Berkelompok' ini telah dihuraikan dan dapat ditunjukkan bahawa ianya boleh digunakan dengan baik untuk data-data klasik yang mempunyai data terpencil berganda. Seterusnya, pengubahsuaian dilakukan ke atas kaedah Sebert dengan menggunakan dua penganggar teguh. 'Kaedah 1' adalah pengubahsuaian daripada kaedah Sebert dimana kaedah penyesuaian kuasa dua terkecil ditukarkan dengan kaedah penyesuaian median kuasa dua terkecil (LMS) manakala bagi 'Kaedah 2', kaedah penyesuaian trim kuasa dua terkecil (LTS) digunakan. Kajian ini juga menyediakan satu analisis perbandingan di antara ketiga-tiga kaedah yang dibincangkan bagi membantu dan memberi satu pendekatan kepada para pengkaji yang akan datang tentang pemilihan kaedah terbaik bagi mengesan data terpencil berganda. Kaedah simulasi Monte Carlo digunakan untuk menilai keberkesanan ketiga-tiga kaedah yang dibincangkan. Semua simulasi dan pengiraan dilakukan dengan menggunakan pakej statistik S-PLUS 2000.

CHAPTER 1

RESEARCH FRAMEWORK

1.1 Background and Motivation

Regression analysis is an important statistical tool that is routinely applied in most sciences. Out of many possible regression techniques, the least squares (LS) method has been generally adopted because of tradition and ease of computation. However, there is presently a widespread awareness of danger posed by the occurrence of outliers, which may be a result of keypunch errors, misplaced decimal points, recording or transmission error, exceptional phenomena such as earthquakes or strikes, or members of different population slipping into the sample.

Identifying outlying observations is an important aspect of the regression model-building process. Outliers occur very frequently in real data, and they often go unnoticed because nowadays computers, process much data without careful inspection or screening. Not only the response variable can be outliers, but also the explanatory part, leading to so-called leverage points. Both types of outliers may totally spoil an ordinary least squares (OLS) analysis.

In general, outliers are defined as observations that appear inconsistent with other observations in the data set. It is important to identify these types of outliers in linear regression modelling because when undetected, can lead to erroneous parameter estimates and inferences from the model. Additionally, these outliers may be of interest themselves to provide insight into process behaviour at certain operating condition.

If there is only a single or a few outliers, many standard LS regression diagnostic quantities and plots will reliably identify these observations. These diagnostics have been shown to fail in the presence of multiple outliers, particularly if the observations are clustered in an outlying cloud. The measures may either fail to identify the outliers (masking), identify the clean observation as outliers (swamping), or could both mask and swamp observations. To overcome the limitations of the standard LS diagnostics, numerous multiple outlier detection techniques have been proposed to identify the outlying subset of observations.

There has been considerable interest in recent years in the detection and accommodation of multiple outliers in statistical modelling. But, most of them are complicated and unappealing to users with no mathematical background to overcome it. This research briefly reviews the multiple outlier procedures chronologically for historical purposes. A detailed outline of the clustering algorithm from Sebert et al. (1998) and two modification from this procedure will be discussed further in chapters four and five since it is easy to understand with interesting proposed approach and have a good performance in detecting the presence of outliers. This research also provides a comparative analysis among these three procedures to help and give further researcher a comprehensive view about the best procedure to detect multiple outliers. A Monte Carlo simulation study was used to evaluate the effectiveness of these three procedures.

1.2 Research Objectives and Scopes

The objectives of this study are to

- Review the multiple outlier procedures chronologically

- Study and characterize the performance of the procedure proposed by Sebert et al. (1998)
- Study the influence of the least median of squares (LMS) fit in Sebert et al. (1998) procedure and characterize the performance of the new procedure (Method 1)
- Study the influence of the least trimmed of squares (LTS) fit in Sebert et al. (1998) procedure and characterize the performance of the new procedure (Method 2)
- Compare the performance of the procedures proposed by Sebert et al. (1998), Method 1 and Method 2
- Choose the best procedure to detect multiple outliers.

For this research, the problem of outlier detection is only focused on the linear regression model.

1.3 Organization of the Report

This report is organized into seven chapters. Chapter 1 discusses the research framework. It begins with the introduction to the multiple outlier detection problems in linear regression and also discusses the objectives and scope of this study.

Chapter 2 reviews the relevant literature on published work done recently. Chapter 3 discusses how the proposed methods perform in the different outlier situations. A detailed study of the procedure on randomly generated data sets was performed with the simulation study planning. Chapter 4 details the procedure proposed by Sebert et al. (1998) and characterizes its performance.

Chapter 5 introduces two modifications of Sebert et al. (1998) procedure, which are Method 1 and Method 2 and characterizes its performances. Chapter 6 compares the performance of the procedures proposed by Sebert et al. (1998) and the modification of Sebert et al. (1998) procedure made by Method 1 and Method 2. The last chapter, that is, chapter 7 summarizes the whole study and also includes some suggestions for future research.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter discusses the ordinary least squares (OLS) regression and hat matrix, and includes a review on the outliers problem in linear regression and a brief discussion on the multiple outlier detection methods and procedures which are recently published chronologically. The discussion on ordinary least squares regression is presented since it is the most commonly used.

2.2 Ordinary Least Squares Regression and Hat Matrix

The general linear regression model can be written in a matrix form as the following

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon} \qquad (2.1)$$

where $\mathbf{y}$ is an $n \times 1$ response variable vector , $\mathbf{X}$ is the $n \times p$ matrix of predictor (or regressor) variables with intercept, $\boldsymbol{\beta}$ is unknown $p \times 1$ vector of regression coefficients and $\boldsymbol{\varepsilon}$ is an $n \times 1$ vector of random errors assumed to be independent normally distributed with mean 0 and variance matrix σ^2.

Let the expected value of $\mathbf{y}$, $E(\mathbf{y}) = \mathbf{X}\boldsymbol{\beta}$ where $\boldsymbol{\beta}$ is the parameter to be estimated. The method of least squares consists of finding estimators $\hat{\boldsymbol{\beta}}$ which minimize the sum of squares of the error terms

$$S = \sum_{i=1}^{n} \varepsilon_i^2 = \boldsymbol{\varepsilon}^T\boldsymbol{\varepsilon} = (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \tag{2.2}$$

where $\varepsilon_i = y_i - x_i^T\beta$ and $\boldsymbol{\varepsilon} = \mathbf{y} - \mathbf{X}\boldsymbol{\beta}$.

Then S becomes:

$$\begin{aligned} S &= \mathbf{y}^T\mathbf{y} - \boldsymbol{\beta}^T\mathbf{X}^T\mathbf{y} - \mathbf{y}^T\mathbf{X}\boldsymbol{\beta} + \boldsymbol{\beta}^T\mathbf{X}^T\mathbf{X}\boldsymbol{\beta} \\ &= \mathbf{y}^T\mathbf{y} - 2\boldsymbol{\beta}^T\mathbf{X}^T\mathbf{y} + \boldsymbol{\beta}^T\mathbf{X}^T\mathbf{X}\boldsymbol{\beta} \end{aligned} \tag{2.3}$$

where

$$(\boldsymbol{\beta}^T\mathbf{X}^T\mathbf{y})^T = \mathbf{y}^T\mathbf{X}\boldsymbol{\beta}$$

and $(\boldsymbol{\beta}^T\mathbf{X}^T\mathbf{y})$ is 1×1 matrix.

The least squares estimates for $\hat{\boldsymbol{\beta}}$ must fulfill $\left.\frac{\partial S}{\partial \boldsymbol{\beta}}\right|_{\hat{\boldsymbol{\beta}}} = 0$ i.e.

$$\begin{aligned} -2\mathbf{X}^T\mathbf{y} + \mathbf{X}^T\mathbf{X}\hat{\boldsymbol{\beta}} + \hat{\boldsymbol{\beta}}^T\mathbf{X}^T\mathbf{X} = 0 \\ \Rightarrow (\mathbf{X}^T\mathbf{X}\hat{\boldsymbol{\beta}})^T = \hat{\boldsymbol{\beta}}^T\mathbf{X}^T\mathbf{X} \\ \therefore -2\mathbf{X}^T\mathbf{y} + 2\mathbf{X}^T\mathbf{X}\hat{\boldsymbol{\beta}} = 0 \end{aligned}$$

$$\Rightarrow \quad \mathbf{X}^T\mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}^T\mathbf{y} \tag{2.4}$$

So the least squares estimators are

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \tag{2.5}$$

These least squares estimators are also maximum likelihood estimators, which is unbiased, minimum variance, consistent and sufficient.

The vector of predicted or fitted values can be expressed as

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}\left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y} = \mathbf{H}\mathbf{y} \tag{2.6}$$

This model can be used to provide important information about the relationship of the response and the explanatory (or regressor) variables. The linear regression model may also be used to identify important regressor variables and/or predict future values of the response variables. The matrix **H** is referred as the hat matrix or projection matrix. The diagonal elements of the hat matrix are used in many least squares diagnostics because they provide an indication of remoteness in X-space.

It can be easily verified that **H** is idempotent $(\mathbf{HH} = \mathbf{H})$ and symmetric $(\mathbf{H}^T = \mathbf{H})$. The diagonals of the hat matrix H, is given as

$$h_{ii} = \mathbf{x}_i\left(\mathbf{X}^T\mathbf{X}\right)\mathbf{x}_i^T \tag{2.7}$$

where

$$\mathbf{x}_i = \left(x_{i,1} \cdots x_{i,p-1}\, 1\right) \tag{2.8}$$

and

$$\mathbf{X}^T\mathbf{X} = \begin{bmatrix} x_{11} & \cdots & x_{i1} & \cdots & x_{n1} \\ \vdots & & \vdots & & \vdots \\ 1 & \cdots & 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} x_{11} & \cdots & 1 \\ \vdots & & \vdots \\ x_{i1} & \cdots & 1 \\ \vdots & & \vdots \\ x_{n1} & \cdots & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \sum_{i=1}^{n} x_{i1}x_{i1} & \cdots & \sum_{i=1}^{n} x_{i1}x_{i,p-1} & \sum_{i=1}^{n} x_{i1} \\ \vdots & & \vdots & \vdots \\ \sum_{i=1}^{n} x_{i,p-1}x_{i1} & \cdots & \sum_{i=1}^{n} x_{i,p-1}x_{i,p-1} & \sum_{i=1}^{n} x_{i,p-1} \\ \sum_{i=1}^{n} x_{i1} & \cdots & \sum_{i=1}^{n} x_{i,p-1} & n. \end{bmatrix} \tag{2.9}$$

The hat matrix is also an important component in the covariance matrices of the fitted $\hat{\mathbf{y}}$ and residuals $\mathbf{e} = \mathbf{y} - \hat{\mathbf{y}}$ since

$$\mathrm{cov}(\hat{\mathbf{y}}) = \sigma^2\mathbf{H} \tag{2.10}$$

$$\mathrm{cov}(\mathbf{e}) = \sigma^2(1 - \mathbf{H}) \tag{2.11}$$

2.3 Treating Outliers in Linear Regression

Observations that do not follow the same model as the rest of the data are typically called outliers. Clearly, the presence of such an extreme value can significantly affect the least squares fitting of a model, and so it is important to determine if the analysis should be modified in some way (such as deleting the observation in question). An outlier among a set of residuals is one that is much larger than the rest in absolute value, perhaps lying as many as three or more standard deviations from the mean of the residuals. Obviously, an outlier in the data may indicate special circumstances needing further investigation.

It is important to recognize differences among possible types of extreme values. As described above, an outlier is any rare or unusual observation appearing at one of the extremes of the data range. Generally, all regression observations, and hence outliers in particular, may be evaluated to given knowledge of the variable, response extremeness and predictor extremeness. The goal is to identify observations that are important in affecting either the choice of variables in the model or the accuracy of estimations of the regression coefficients and associated standard errors.

The observation should be checked for plausibility if it has been identified as an outlier. It is important that the data analyst be familiar with the basic characteristics of the data. More generally, one may classify any observations being impossible, highly implausible, or plausible. It is then necessary to consider the importance of an observation in determining the choice of variables in the model, coefficient estimates, and associated statistics before deciding what, if any, action to take. Important concepts include leverage and influence.

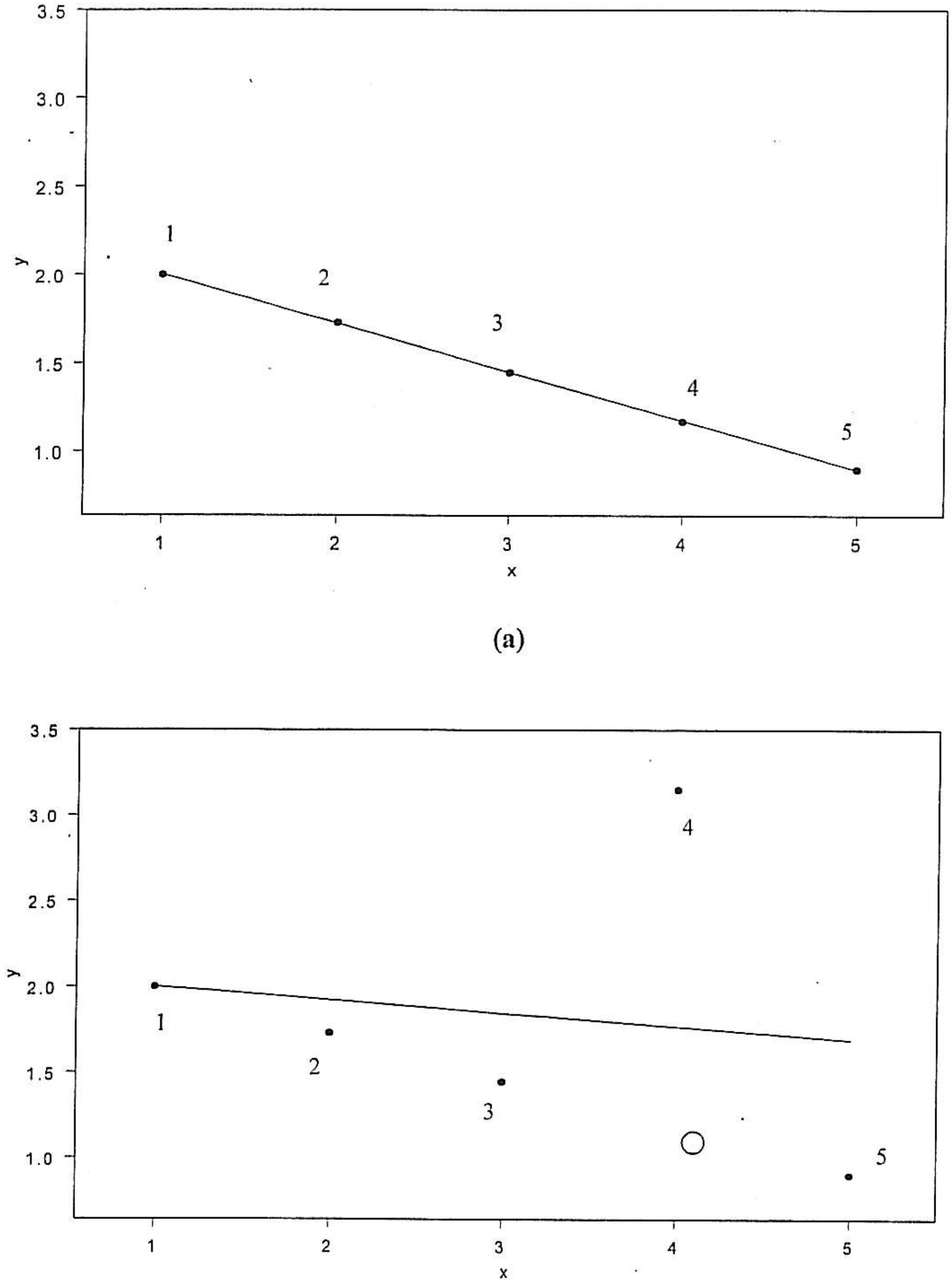
Traditionally, outliers among observations were detected by considering the residuals. It means that the least squares (LS) method is used frequently and has become the cornerstone of classical statistics. After Gauss introduced the normal (or Gaussian) distribution for which LS are optimal, the combination of Gaussian assumptions and LS has become a standard mechanism for the generation of statistical techniques. More recently, some people began to realize that real data usually do not completely satisfy the classical assumption, often with dramatic effects on the quality of the statistical analysis.

As an illustration, let us look at the effect of outliers in the simple regression model

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i \tag{2.12}$$

in which the intercept β_0 and the slope β_1 are to be estimated. In the simple regression model, one can make a plot of the (x_i, y_i), which is sometimes called a scatter plot, in order to visualize the data structure. In the general multiple regression model (2.1) with large number of independent variables, this would no longer be possible, so it is better to use simple regression for illustrative purposes.

Figure 2.1(a) is the scatter plot of five points, $(x_1, y_1), \ldots, (x_5, y_5)$, which almost lie on a straight line. Therefore, the LS solution fits the data very well, as can be seen from the LS line $\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$ in the plot. However, suppose that someone gets a wrong value of y_4 because of a copying or transmission error, thus affecting, for instance, the place of the decimal point. Then (x_4, y_4) may be rather far away from the 'ideal' line.



(a)

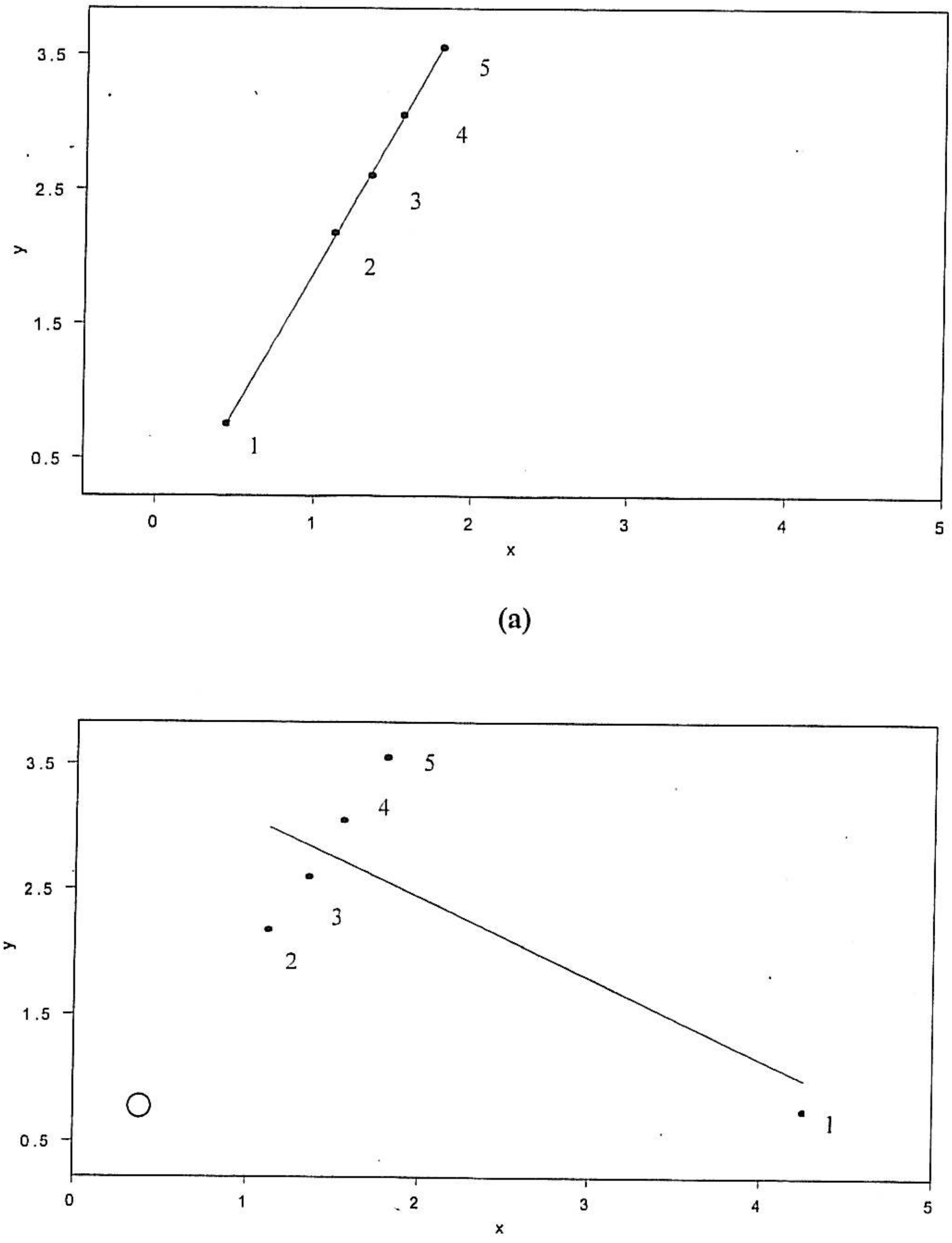
(b)

Figure 2.1: **(a)** Original data with five points and their least squares regression line. **(b)** Same data as in part (a), but with one outlier in the *y*-direction.

Figure 2.1. (b) displays such situation, where the fourth point has moved up and away from its original position (indicated by the circle). This point is called an outlier in the y-direction, and it has a rather large influence on the LS line, which is quite different from the LS line in Figure 2.1. (a). This phenomenon has received some attention in the literature because y_i usually considered as observations and the $x_{i1},\ldots,x_{ip}$ as fixed numbers (which is only true when the design has been given in advance) and because such 'vertical' outliers often posses large positive or large negative residuals. Indeed, in this example the fourth point lies farthest away from the straight line, so its e_i is suspiciously large. Even in general multiple regressions with large p, where one cannot visualize the data, such outliers can often be discovered from the list of residuals or from residual plots.

For the effect of such an outlier, let us look at an example of simple regression in Figure 2.2. Figure 2.2. (a) contains five points, $(x_1, y_1),\ldots,(x_5, y_5)$, with a well fitting LS line. If we now make an error in recording x_1, we obtain Figure 2.2. (b). The resulting point is called an outlier in the x-direction, and its effect on the least squares estimator is very severe because it actually tilts the LS line.

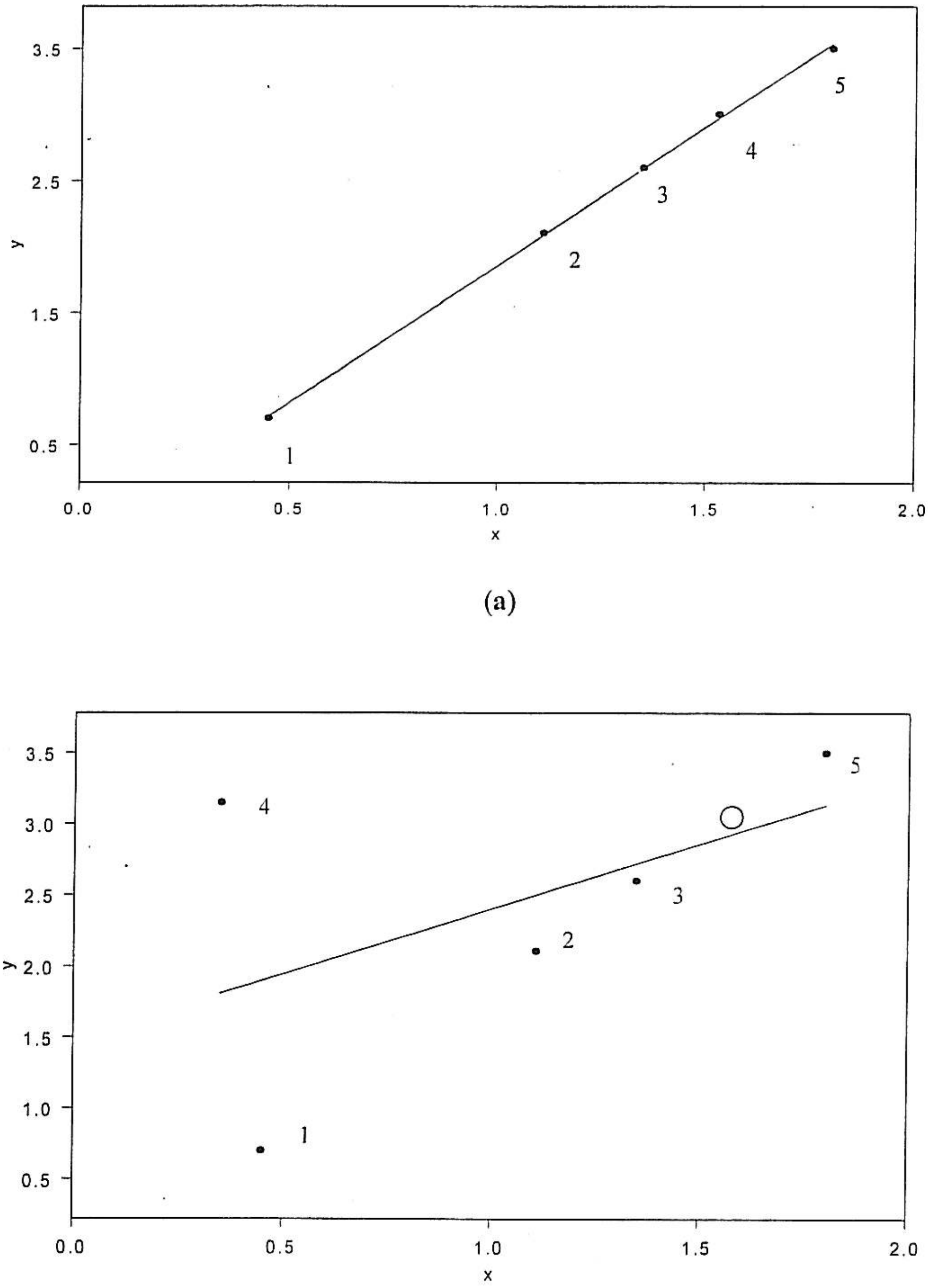
Therefore the point (x_1, y_1) in Figure 2.2. (b) is called a leverage point. This large 'pull' on the LS estimator can be obtained as follows. Because x_1 lies far away, the residual e_i from the original line (as shown in Figure2.2. (a)) becomes a very large (negative) value, contributing enormous amount to $\sum_{i=1}^{5} e_i^2$ for that line. Therefore the original line cannot be selected from a least squares perspective, and indeed the line of Figure 2.2. (b) possesses the smallest $\sum_{i=1}^{5} e_i^2$ because it has tilted to reduce that large e_1^2 even if the other four terms, $e_2^2,\ldots,e_5^2$ have increased quite a bit.



(a)

(b)

Figure 2.2: **(a)** Original data with five points and their least squares regression line. **(b)** Same data as in part (a), but with one outlier in the x-direction. (leverage point)



(a)

(b)

Figure 2.3: **(a)** Original data with five points and their least squares regression line. **(b)** Same data as in part (a), but with one outlier in the *xy*-direction. (high leverage point)

Further, if we make error in recording both x_4 and y_4, the resulting point is now called an outlier in the *xy*-direction (space). Again, the effect on the least squares estimator is large and the point $(x_4, y_4)$is called a high leverage point. Figure 2.3 (b) illustrates the problem.

In general, an observation (x_k, y_k) is called a leverage point whenever x_k lies far away from the majority of the observed x_i in the sample. Note that this does not take y_k into account, so the point (x_k, y_k) does not necessarily have to be a regression outlier. When (x_k, y_k) lies close to the regression line determined by the majority of the data, then it can be considered a 'good' leverage point as in Figure 2.4. Therefore to say that (x_k, y_k) is leverage point refers only to its potential for strongly affecting the regression coefficients $\hat{\beta}$, but it does not necessarily mean that (x_k, y_k) will actually have a large influence on $\hat{\beta}$, because it may be perfectly in line with the trend set by the other data. In such situation, a leverage point is even quite beneficial because it will shrink certain confidence regions.

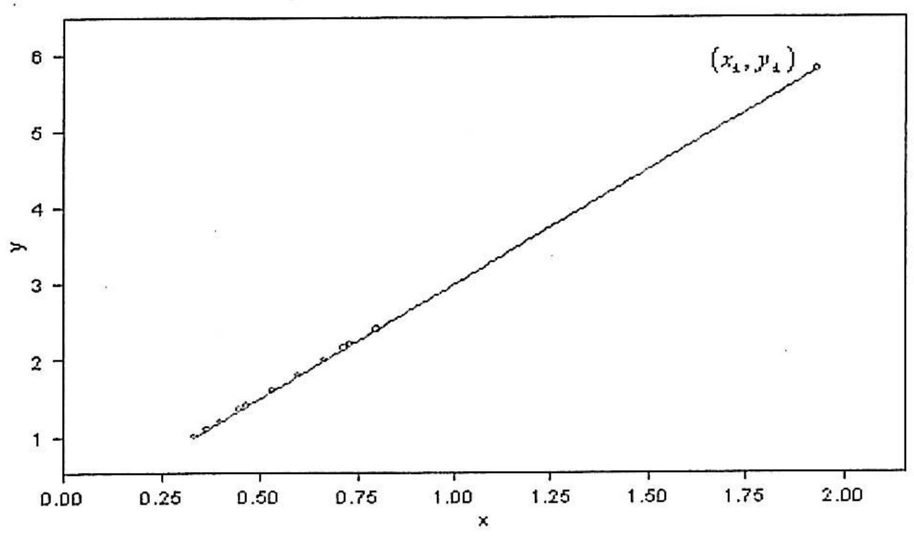


Figure 2.4: The point (x_k, y_k) is a leverage point because x_k is outlying. However (x_k, y_k) is not a regression outlier because it matches the linear pattern set by the other data points.

In multiple regression, the $(x_{i1},\ldots,x_{ip})$ lie in a space with p dimensions. A leverage point is then still defined as a point $(x_{k1},\ldots,x_{kp},y_k)$ for which $(x_{k1},\ldots,x_{kp})$ is outlying with respect to the $(x_{i1},\ldots,x_{ip})$ in the data set. As before, such leverage points have a potentially large influence on the LS regression coefficients, depending on the actual value of y_k. However, in this situation it is much more difficult to identify leverage points, because of the higher dimensionality.

A simple illustration of this problem is given in Figure 2.5, which plots x_{i2} versus x_{i1} for some data set. In this plot, we easily see two leverage points, which are, however, invisible when the variables x_{i1} and x_{i2} are considered separately. In general, it is not sufficient to look at each variable separately or even at all plots of pairs of variables. Clearly, the identification of outlying $(x_{i1},\ldots,x_{ip})$ is a difficult problem.

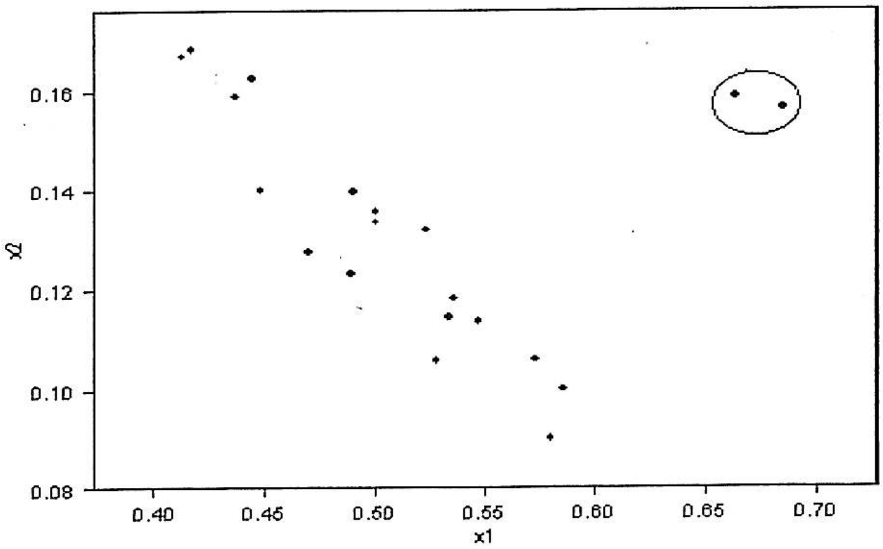


Figure 2.5: Plot of the explanatory variables (x_{i1},x_{i2}) of a regression data set. There are two leverage points (indicated by the circle), which are not outlying in either of the coordinates.

Many people will argue that by looking at the least squares residuals we can detect regression outliers. Unfortunately, this is not true when the outliers are leverage points. For example, consider again Figure 2.2. (b). Point x_1, being a leverage point, has tilted the LS line so much that it is now quite close to that line. Consequently, the residual $e_1 = y_1 - \hat{y}_1$ is a small (negative) number. On the other hand, the residuals e_2 and e_5 have much larger absolute values, although they correspond to 'good' points.

If one would apply a rule like 'delete the points with largest LS residuals', then the 'good' points would have to be deleted first. In a bivariate data set, there is really no problem at all because one can actually look at the data, but there are many multivariate data sets where the outliers remain invisible even through a careful analysis of the LS residual. To conclude, regression outliers (either in *x*, in *y*, *or* in *x* and *y*) pose serious threats to standard least squares analysis. Thus, to overcome this problem many researchers over recent years have suggested numerous procedures and strategies. A survey of these new techniques and approach is provided in Section 2.4.

2.4 Multiple Outlier Detection Method and Procedures

The multiple outliers problem has been considered by many writers over recent years. As a result of the need to identify outliers, numerous outlying measures such as residuals and influence diagnostics such as *Cook's Distance* or *COVRATIO* have been developed. These outlying measures and influence diagnostics work well when a regression data set contains only a single outlying point. However, it is well established that regression data set may have multiple outlying observation that individually are not easily identified by the same measures.

Researchers have suggested numerous strategies to solve the multiple outlier identification problem. Normally, multiple outlier identification techniques suffer from two identification errors that is masking and swamping. Masking is the inability of a detection method to correctly classify a true outlier. That is, the detection method falsely indicates that the outlier is an inlier. Swamping results when a detection method classifies an inlier as being outlier. Masking is more serious than swamping.

Generally, there are two broad classes of multiple outlier detection procedures. Hadi and Simonoff (1993) defined these procedures as direct method and indirect method. The direct method used algorithms to isolate outliers and the indirect methods use the results from robust regression estimates. The description of both the direct and indirect procedures below considers the standard linear model $\mathbf{y} = \mathbf{X\beta} + \boldsymbol{\varepsilon}$ where $\mathbf{y}$ is the response vector of dimension n, the number of observation; $\mathbf{X}$ is the $n \times p$ matrix of regressor variables with intercept; and $\boldsymbol{\varepsilon}$ is the column vector of n random errors assumed to have mean 0 and covariance matrix $\sigma^2\mathbf{I}$. Figure 2.6 shows the historical flowchart of multiple outlier detection methods for both direct and undirect procedures. Of course this list is not exhaustive.

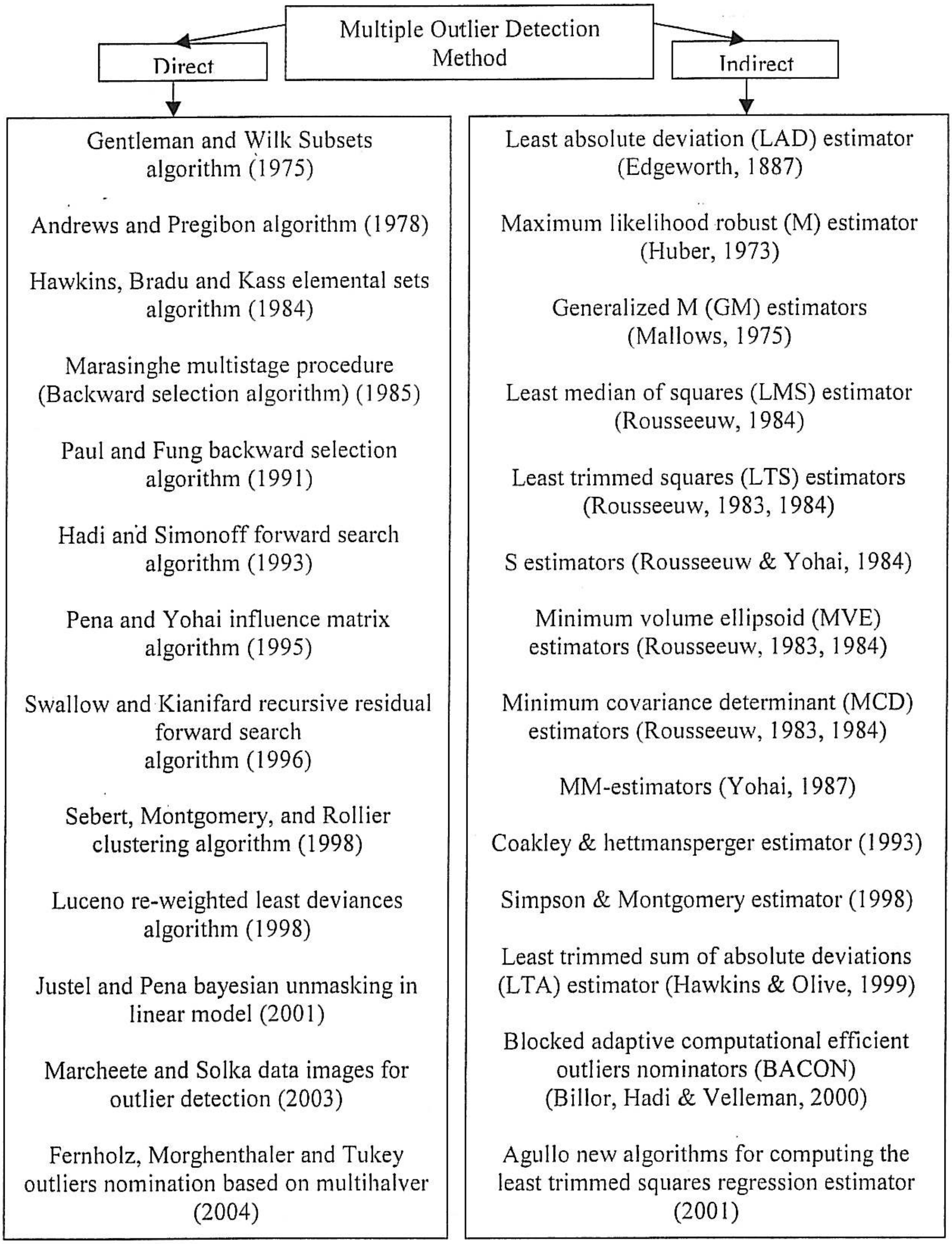

Multiple Outlier Detection Method
Direct
Indirect
Gentleman and Wilk Subsets algorithm (1975)
Andrews and Pregibon algorithm (1978)
Hawkins, Bradu and Kass elemental sets algorithm (1984)
Marasinghe multistage procedure (Backward selection algorithm) (1985)
Paul and Fung backward selection algorithm (1991)
Hadi and Simonoff forward search algorithm (1993)
Pena and Yohai influence matrix algorithm (1995)
Swallow and Kianifard recursive residual forward search algorithm (1996)
Sebert, Montgomery, and Rollier clustering algorithm (1998)
Luceno re-weighted least deviances algorithm (1998)
Justel and Pena bayesian unmasking in linear model (2001)
Marcheete and Solka data images for outlier detection (2003)
Fernholz, Morghenthaler and Tukey outliers nomination based on multihalver (2004)
Least absolute deviation (LAD) estimator (Edgeworth, 1887)
Maximum likelihood robust (M) estimator (Huber, 1973)
Generalized M (GM) estimators (Mallows, 1975)
Least median of squares (LMS) estimator (Rousseeuw, 1984)
Least trimmed squares (LTS) estimators (Rousseeuw, 1983, 1984)
S estimators (Rousseeuw & Yohai, 1984)
Minimum volume ellipsoid (MVE) estimators (Rousseeuw, 1983, 1984)
Minimum covariance determinant (MCD) estimators (Rousseeuw, 1983, 1984)
MM-estimators (Yohai, 1987)
Coakley & hettmansperger estimator (1993)
Simpson & Montgomery estimator (1998)
Least trimmed sum of absolute deviations (LTA) estimator (Hawkins & Olive, 1999)
Blocked adaptive computational efficient outliers nominators (BACON) (Billor, Hadi & Velleman, 2000)
Agullo new algorithms for computing the least trimmed squares regression estimator (2001)

Figure 2.6: The historical flowchart of multiple outlier detection methods for direct and indirect procedures

2.4.1 Direct Procedures

Direct methods are procedures based on least squares and specifically designed algorithm to detect multiple outliers. Many of the direct procedures in the literature are based on either sequential deletion (backward search) of outlying observations or sequential addition (forward search) of clean observations. In a backward search, the entire set of observations is initially considered and the outliers are sequentially removed by a criterion such as the largest absolute value of a transformed residual. The forward search works similarly. A small subset of the data is selected as the initial clean basis and clean observations are sequentially added to this basis. Wisnowski et al. (2001) pointed out that methods using forward search generally outperform backward search methods.

The following discussed briefly the relevant direct procedures chronologically from 1996 but only a detailed outline of the clustering algorithm from Sebert et al. (1998) will be discussed fully in chapter 4. The general steps of these algorithms and specific issues related to this research are outlined below.

Swallow and Kianifard Recursive Residual Forward Search Algorithm

Swallow and Kianifard (1996) suggest recursive residuals standardized by a robust estimate of scale as the test statistic to classify multiple outliers. The algorithm first orders the magnitudes of the studentized residual values from a least squares fit to form the basis of *p* clean observations. The *j*th recursive residual, w_j, is computed by

$$w_j = \frac{y_j - \mathbf{x}^{\mathrm{T}}\boldsymbol{\beta}_{j-1}}{\left(1 - \mathbf{x}_j^{\mathrm{T}}\left(\mathbf{X}_{j-1}^{\mathrm{T}}\mathbf{X}_{j-1}\right)^{-1}\mathbf{x}_j\right)^{1/2}} \quad , j = p+1, \ldots, n.$$

$\boldsymbol{\beta}_{j-1}$ is the vector of parameter estimates determined by using the subset of size *j*-1 from the ordered observations (by ordinary least square (OLS) studentized residuals). Thus, parameter estimates must be recalculated many times (n-p) to find the recursive residuals for a single data set. Similarly, $\mathbf{X}_{j-1}$ is the subset matrix of explanatory variables for the first j-1 ordered observations. Recursive residuals are scaled by the median absolute deviation from the median (MAD) estimate of scale $\hat{\sigma}$. The MAD is $\{|e_i - \text{median}\{e_i\}|\}$ where e_i is the OLS residual, not the studentized residual. The test statistic $|w_j/\hat{\sigma}|$ for each observation is compared to a cutoff value to identify the outliers. The required MAD scale estimate correction factor and the cutoff value come from the quantiles of 1000 simulations of n observations with p parameters under the null hypothesis of no outliers.

Sebert, Montgomery, and Rollier Clustering Algorithm

Sebert et al. (1998) proposed an approach for identifying a reasonable subset of potential outliers without the complexities associated with most competing procedures. This approach uses a single linkage clustering algorithm with the Euclidean distances for the standardized predicted and standardized residual values from a least squares fit. The crux of the algorithm is to find the single largest cluster, or the bulk of the data to classify as the inliers. Mojena's stopping rule forms the final clusters by splitting a cluster tree at the average of the n-1 tree cluster heights (a measure of cluster separation) plus 1.25 times the standard deviation of the tree cluster heights. The Mojena's stopping rule defined as $\bar{h} + 1.25s_h$ where $\bar{h}$ is the average height of the tree and s_h is the sample standard deviation of the heights. Minowski (1999) showed that this procedure generally perform well in classical challenging data sets. The performance of this method improves with the increase of outlying distance, number of observations, and the number of regressors as well as a decrease in the percentage of outliers.

Luceno Re-weighted Least Deviances Algorithm

Luceno (1998) proposed a procedure to detect multiple outliers in the generalized linear models (GLIM) using the weights from a re-weighted least square. The mean of the deviance (sum of squared deviance residuals) is replaced by a weighted mean of deviances. The weights used are related to the deviance residuals through Huber or redescending type function. The parameter estimates come from the minimization of the quantity $n^{-1}\sum w_i D_i(\mu;\phi;y)$ where D_i is the squared deviance residual for the i-th observation, μ is the mean, ϕ is the nuisances parameter, and w_i is the weight from the influence function. If the Huber's function is used to find w_i, then

$$\begin{aligned} w_i &= \frac{1.5}{\left|D_i^{1/2}\right|} && \text{, if } \left|D_i^{1/2}\right| > 1.5 \\ &= 1.0 && \text{, otherwise} \end{aligned}$$

Observations with unusually low values for weights are considered outliers. The procedure successfully detected outliers in several examples from McCullagh and Neldeer (1989) and also identified outliers in the stackloss data set. The method appears to be effective at detecting X-space outliers (leverage outliers).

Bayesian Unmasking in Linear Model

Justel and Pena (2001) proposed a Bayesian procedure for multiple outlier detection in linear models, which avoids the masking problem. The posterior probabilities of each data point being an outlier are estimated by using an adaptive learning Gibbs sampling method. The idea is to modify the initial condition of the Gibbs sampler in order to visit the posterior distribution space in a reasonable number of iterations. To find an appropriate vector of initial values, the information is extracted from the eigenstructure of the covariance matrix of a vector of latent variables. These variables are introduced in the model to capture the heterogeneity in the data. This procedure also overcomes the false convergence of the Gibbs sampling in problems with strong masking.

Using Data Images for Outlier Detection

Proposed by Marcheete and Solka (2003), the data image is a method for visualizing high-dimensional data. The idea is to map the data into an image, by using gray-scale (or color) values to indicate the magnitude of each variable. Thus, the image for a data set of size n and dimension d is an $d \times n$ image, where the columns correspond to observations and the rows to variables. They consider the application of this idea to the detection of outliers providing a simple visualization technique that highlights outliers and clusters within the data.

Outliers Nomination Based on Multihalver by Fernholz, Morghenthaler and Tukey

Fernholz et al. (2004) proposed a new method for detecting observations with a pronounced influence on a given estimator T. This method based on a careful selection of a set of half samples H and on a detailed study of the differences of the estimator computed on the complementary halves. After the outlier detection has been performed, the flagged observations can be removed or winsorized. Then, compute an improved estimate and a confidence interval based on the modified sample and again using half samples. An important advantage of this method is its generality. The method can easily be applied to any real-valued estimator, even if the data is multivariate.

2.4.2 Indirect Procedures from Robust Regression Estimators

Robust regression techniques accommodate outliers by down weighting or ignoring the unusual observations to ensure they are not too influential on the regression parameter estimates. It is possible to detect unusual observations from either the final weights assigned to the observations or by the magnitude of the residuals. Robust regression techniques are important since they provide similar estimates of the parameter compared to the estimates given by least squares methods when the data are free of outliers, but differ significantly when there exist outliers.

Robust regression models are useful when the random error (variation) in the data is not normally distributed or when there are outliers present. Robust regression devises estimators, which are not strongly effected by outliers. If an estimate x_R (R for robust) is relatively unaffected by outliers, the residuals $e_R = y - \hat{y}_R$ from the robust fit should be useful to flag cases off the regression line. In robust analysis, it wants to fit a regression to the majority of the data and discover the outliers at those points with large residuals from the robust solution. Clearly, robust regression methods are designed to fit data with outliers by minimizing some function of the residuals that is less sensitive than least of squares to outliers.

The three most important properties of robust regression estimators are high breakdown point, high efficiency, and bounded influence. The breakdown point is the percentage of outliers present in the data when the technique's parameter estimates become unreliable or fail to provide useful information regarding the bulk of the data. This measure used to determine an estimator's insensitivity to multiple outliers. Generally, the breakdown point gives the limiting fraction of outliers the estimator can cope with. For instance, least squares have a breakdown of $1/n$, indicating that only a single outlier can make the estimate useless. Some robust techniques have the highest possible breakdown point of $n/2n$ or 50%. Various high breakdown estimators have been used as starting points for bounded influence estimation.

Efficiency is defined as the performance of robust estimator relative to least squares estimator under the assumption of no outliers, that is, the random error ε is normally and independently distributed with mean 0 and variance covariance $\sigma^2 I$. Since the least squares estimate is the unbiased minimum variance estimator (UMVE), typically the efficiency is defined as the ratio of the mean square error, that is

$$\text{Efficiency} = \frac{MSE_R}{MSE_{LS}} \tag{2.13}$$

The bounded influence property makes sure the estimator is resistant from being affected by the extreme observation in x-space. Krasker and Welsh (1982) discussed the importance of bounded influence specifically its ability to down weight observations that are both high residual and high leverage observations. The least squares estimate is not bounded influence and therefore the observation, which is more remote; exert a greater influence on the parameter estimates. That is why the techniques based on least squares are unable to detect high leverage points.

The following discussed briefly the relevant direct procedures chronologically for historical purposes.

Maximum Likelihood Robust (M)-Estimator

M-estimator proposed by Huber (1973) as pointed out in Huber (1981), is based on the idea of replacing the squared residuals e_i^2 in least squares method by another function of the residuals, yielding

$$\underset{\hat{\beta}}{Minimize} \sum_{i=1}^{n} \rho(e_i) \tag{2.14}$$

where ρ is symmetric function (i.e., $\rho(-t) = \rho(t)$ for all t) with a unique minimum at zero. Differentiating this expression with respect to the regression coefficients $\hat{\beta}_j$ yields

$$\sum_{i=1}^{n} \varphi(e_i)\mathbf{x}_i = 0\,, \tag{2.15}$$

where ψ is the derivative of ρ, and $\mathbf{x}_i$ is the row vector of explanatory variables of the ith case. One has to standardize the residuals by means of some estimate of σ, yielding

$$\sum_{i=1}^{n} \varphi\left(\frac{e_i}{\hat{\sigma}}\right)\mathbf{x}_i = 0\,, \tag{2.16}$$

where $\hat{\sigma}$ must be estimated simultaneously. Huber proposed to use the function

$$\varphi(t) = \min(c, \max(t, -c))\,. \tag{2.17}$$

M-estimators are still robust with respect to outliers y_i. However, their breakdown point is again $1/n$ because of the effect of outliers $\mathbf{x}_i$.

Generalized M (GM)-Estimators

GM-estimators were introduced, with the basic purpose of bounding the influence of outlying $\mathbf{x}_i$ by means of some weight function w. Mallows (1975) proposed to replace (2.16) by

$$\sum_{i=1}^{n} w(\mathbf{x}_i)\varphi\left(\frac{e_i}{\hat{\sigma}}\right)\mathbf{x}_i = 0\,, \tag{2.18}$$

whereas Schweppe (1977) suggested using

$$\sum_{i=1}^{n} w(\mathbf{x}_i)\varphi\left(\frac{e_i}{w(\mathbf{x}_i)\hat{\sigma}}\right)\mathbf{x}_i = 0\,. \tag{2.19}$$

These estimators were constructed in the hope of bounding the influence of a single outlying observation, the effect of which can be measured by means of so-called influence function. Generally, the corresponding GM-estimators are called bounded influence estimator.

Least Median of Squares (LMS) Estimator

Rousseeuw (1984) introduced the high breakdown (as much as 50%) LMS estimators. LMS is obtained by minimizing the *h*th ordered squared residual where h is defined as the integer portions of $[(n/2)+(p+1)/2]$. Note h is not the median of n. In other words, LMS estimator $\hat{\beta}$ is obtained from minimizing the median of squared errors, that is, it solves

$$\underset{\hat{\beta}}{Minimize}\left[\mathrm{med}\left(e_i^{\,2}\right)\right] \tag{2.20}$$

LMS fits just over half the data and minimizes the residual for a single observation. The LMS has a high breakdown but due to its $n^{-1/3}$ convergence rate, it has zero efficiency under the central Gaussian model.

Least Trimmed Squares (LTS) Estimator

Rousseeuw (1983,1984) proposed the high breakdown LTS estimator as an efficient alternative to LMS. The LTS estimator is formed by minimizing the h out of n ordered squared residuals, given by

$$\underset{\hat{\beta}}{Minimize}\sum_{i=1}^{h}\left(e^2\right)_{i;n} \tag{2.21}$$

where $\left(e^2\right)_{1;n} \leq \ldots \leq \left(e^2\right)_{n;n}$ are the ordered squared residual. Rousseeuw and Leroy (1987) recommended $h = n(1-\alpha)+1$ where α is the trimmed percentage. This estimator is attractive because α can be selected to prevent some of the poor results (efficiency) that other 50% breakdown estimators show. LTS estimator has 7.12% asymptotic efficiency.

S-Estimator

Both the LMS and LTS are defined by minimizing a robust measure of the scatter of the residuals. Generalizing this, Rousseeuw and Yohai (1984) introduced so-called S-estimator, corresponding to

$$\underset{\hat{\beta}}{Minimize}\ S(e_1(\boldsymbol{\beta}),\ldots,e_n(\boldsymbol{\beta})), \tag{2.22}$$

where the dispersion function $S(e_1(\boldsymbol{\beta}),\ldots,e_n(\boldsymbol{\beta}))$ is found implicitly as the solution to

$$\left(\frac{1}{n-p}\right)\sum_{i=1}^{n}\rho\left[\frac{y_i-\mathbf{x}_i^T\hat{\beta}}{s}\right]=K. \tag{2.23}$$

The constant K may be defined as $E_\Phi[\rho]$, where Φ represents the standard normal distribution. S-estimator is also asymptotically normal and has 28.7% efficiency.

Minimum Volume Ellipsoid (MVE) and Minimum Covariance Determinant (MCD) Estimators

The traditional regression leverage measure, h_{ii} and the Mahalanobis distance are not robust. Robust measures of leverage include the Minimum Volume Ellipsoid (MVE) estimator, the Minimum Covariance Determinant (MCD) estimator, and M-estimates of covariance. The MVE and MCD estimators (Rousseeuw, 1983,1984) have breakdown as high as 50% but suffer from computational problems and have some outlier detection vulnerabilities. The MVE and MCD estimators computed using a global optimization search routine that considers subsamples of $p+1$ points. For moderate to large sized problems, random subsampling is required. Unfortunately, random subsampling does not guarantee locating the true minimum. Several algorithms have been developed that calculate the approximate and exact solutions require excessive processing time and the approximate solutions have considerable variability. A study was performed on the high breakdown MVE and MCD estimators using datasets with outliers (Simpson, 1995). In several instances these methods not only masked the points in the outlier cluster, but they also identified some of the inliers as outliers (swamping).

MM-estimator

The MM-estimator is a high-breakdown and high-efficiency estimator with three stages, which was proposed by Yohai (1987) as pointed out in Yohai et al. (1991). The initial estimate is a high-breakdown estimate using an S-estimate. The influence function given by

$$\begin{aligned} &\rho(x) = 3\left(x/c\right)^2 - 3\left(x/c\right)^4 + \left(x/c\right)^6 && \text{if } |x| \le c \\ &\rho(x) = 1 && \text{otherwise} \end{aligned}$$

The value of the tuning constant is set as 1.548. The second stage computes an MM parameters that minimize $\sum_{i=1}^{n} \rho\left[\frac{y_i - \mathbf{x}_i^T \hat{\beta}_{MM}}{\hat{\sigma}_0}\right]$ where $\rho(x)$ is the influence function used in the first stage with tuning constant selected as 4.687 and $\hat{\sigma}_0$ is the estimate of scale from the first stage that is the standard deviation of the residuals. In the last step, the procedure computes the MM estimate of scale as the solution to

$$\left(\frac{1}{n-p}\right)\sum_{i=1}^{n} \rho\left[\frac{y_i - \mathbf{x}_i^T \hat{\beta}}{s}\right] = 0.5 .$$

Coakley and Hettmansperger Estimator

Proposed by Coakley and Hettmansperger (1993), this compound estimator uses LTS as the initial estimate and adjusts the estimates with empirically determined weights. The weights given to the leverage come from the minimum volume ellipsoid (MVE) scaled by percentiles of the chi-square distribution. Other components include a Scheppe-type GM objective function, an estimate of scale from the scaled median of the LTS residuals, the Huber psi function and a one-step Newton–Raphson convergence approach.

Simpson and Montgomery Estimator

Proposed by Simpson and Montgomery (1998), this compound estimator uses a high-breakdown S-estimate for the initial estimate that minimizes the dispersion of the residuals. The estimate of scale also comes from this initial fit with the S-estimator. A leverage measure for each observation is based on distances using the M-estimates of covariance procedure. High-leverage points are downweighted using a Schweppe-type GM objective function if the observations do not conform to the regression surface. Regression outliers are downweighted by a Tukey-biweight psi function. This function allows outliers to exert an increasing amount of influence on parameter estimates to a certain point (e.g. 3σ off the regression plane) and then decreases the outliers' influence beyond this point until eventually reaching no influence. These weights are then used in only a single step of reweighted least squares to preserve the high-breakdown property from the S-estimator.

Least Trimmed Sum of Absolute Deviations (LTA) Estimator

Hawkins and Olive (1999) proposed the use of least trimmed sum of absolute deviations (LTA) as an alternative to LMS and the LTS. The computational complexity is of lower order than the LMS or the LTS. The use of high breakdown estimates for example, LMS and LTS, lead to a portioning of the data set into two halves, that is the covered and the uncovered 'half'. The covered half of cases are accommodated by the fit, while the uncovered half, which might include the outliers, are ignored. In LMS, the criterion is the Chebyshev norm of the residuals of the covered cases while in LTS; the criterion is the sum of squared residuals of the covered cases. The criterion of the LTA is found by minimizing the sum of squared residuals of the covered cases. The LTA is particularly attractive for large data sets. The modification of the Hawkins-Simonoff elemental set code has been used for the exact computation of the LTA, which involves enumeration of all elemental subsets of the data. It is also suggested that LTA be used as a tool foe modeling data sets with missing observations on predictors.

Blocked Adaptive Computational Efficient Outliers Nominators (BACON)

Billor, Hadi and Velleman (2000), proposed a method that is based on the methods of Hadi (1992, 1994), that is by finding a small subset of data that can be presumed free of outliers and then allowing the subset to grow rapidly, testing against a criterion and incorporating blocks of observations at each step. The following is the general BACON algorithm:

- *Step 1*: Identify an initial basic subsets of $m > p$ observations that can safely be assumed free for outliers, where p is the dimension (or number of regressors) of the data and m is an integer chosen by the analyst.
- *Step 2*: Fit an appropriate model to the basic subset, and from that model compute discrepancies for each of the observations.
- *Step 3*: Find a larger basic subset consisting of observations known (by their discrepancies) to be homogenous with the basic subset. Generally, these are the observations with smallest discrepancies. This new basic subset may omit some of the previous basic subset observations, but the size must be as large as the previous basic subset.

- *Step 4*: Iterate *Steps 2* and *3* to refine the basic subset, using stopping rule that determines when the basic subset can no longer grow safely.
- *Step 5*: Nominate the observations excluded by the final basic subset as outliers.

The BACON algorithms reliably detect multiple outliers can be as slow as four repetitions of the underlying fitting method. This algorithm can be applied to non-linear models provided the analyst is willing to assume an error distribution to use as a basis for determining the cutoff value for discrepancies. It is also easy to implement in statistics packages that have programming or macro languages.

New Algorithms for Computing the Least Trimmed Squares Regression Estimator

Agullo (2001) proposed two new algorithms to compute the LTS estimator. The first algorithm is probabilistic and based on an exchange procedure. The second algorithm is exact and based on a branch and bound (BAB) technique that guarantees global optimality without exhaustive evaluation. This BAB technique avoids the exhaustive enumeration of all *h*-subsets and the algorithm is computationally feasible for data sets with $n \leq 50$ and $p \leq 5$. However in practice, unless for these very small data sets, the BAB algorithm will be computationally prohibitive and an approximate algorithm should be used. In order to approximate the LTS and LTD estimates, Agullo also proposed a minimum-maximum exchange algorithm. Agullo recommended executing many repetitions of the minimum-maximum exchange algorithm, with each repetition starting from a partially random *h*-subset containing the *h* observations with smallest squared residuals evaluated at the LS fit for a random *p*-subset.

CHAPTER 3

MONTE CARLO SIMULATION STUDY PLANNING

3.1 Introduction

This chapter discusses how the proposed methods perform in the different outlier situations. However to further understand the performance of the methods, a detailed study of the procedure on randomly generated data sets was performed. The following discusses the details of the simulation study planning.

3.2 Outlier Scenarios and Regression Conditions

Each of the multiple outliers detection method was tested in various outlier scenarios and regression conditions. An outlier scenario refers to the placement of the outlying observations relative to the inlying observations. A regression condition refers to the number of observations in the data set, the number of regressor variables, and the percentage of outlying observations. There are 6 outlier scenarios and a total of 36 regression conditions for each scenario considered in this research. These 36 regression conditions consist of 3 levels of regressor variables, 3 levels of sample size, 2 levels of outlying percentage and 2 levels of outlier distances. Figure 3.1 illustrates the six-outlier scenarios chosen for this research for the case of one regressor variable. In the figure, the black circles represent outlying groups of observations.

These scenarios were chosen because they are situations in which multiple outliers are highly influential but typical least squares outlying measures and influence diagnostics fail to identify them. Specifically, these scenarios contain groups of high leverage outliers, which are most difficult to identify. It should be noted that these scenarios closely resemble those of Kianifard and Swallow (1990) and Hadi and Simonoff (1993).

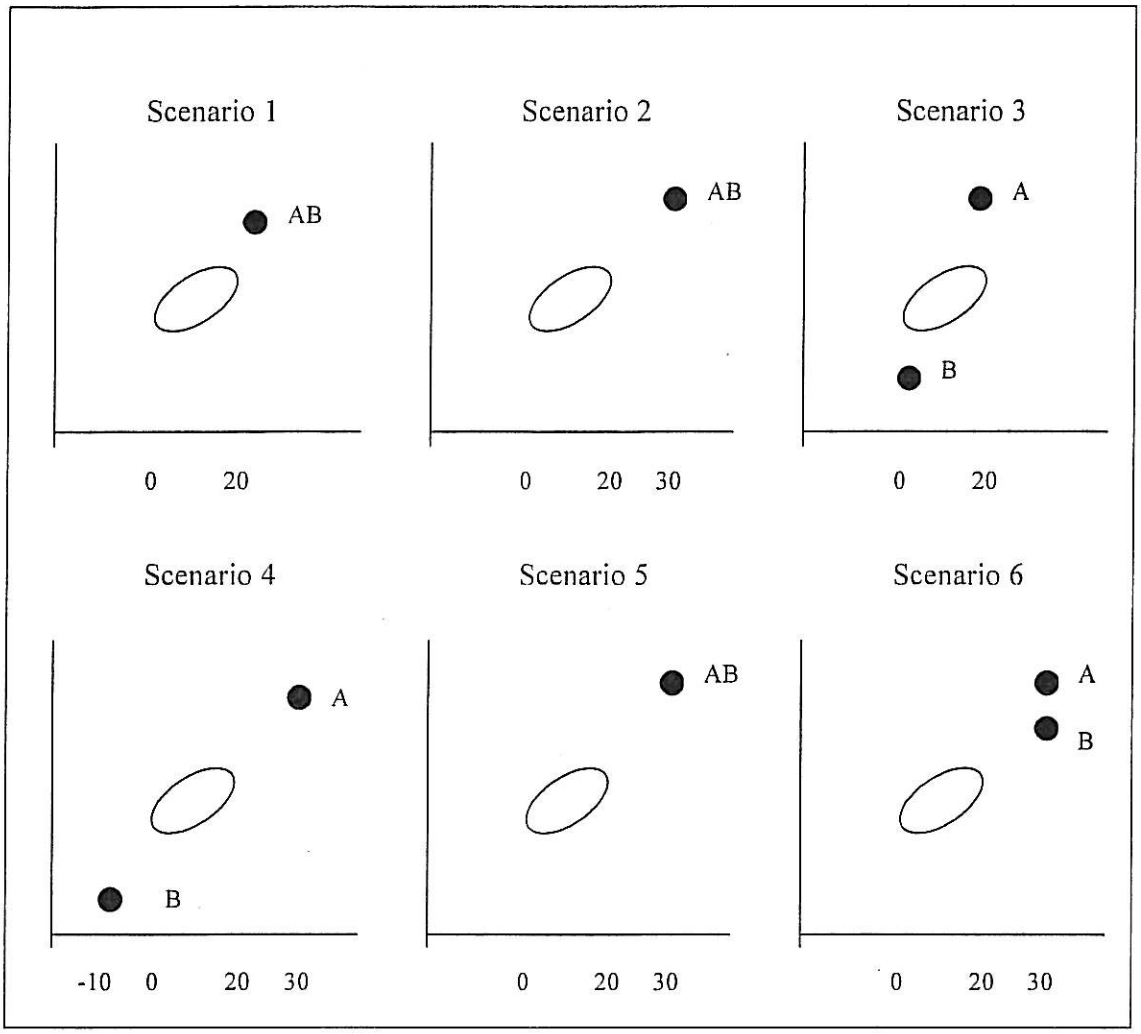


Figure 3.1: Simple regression picture of the six-outlier scenarios tested

Every outlier scenario tested had either one or two outlying groups of observations. Scenarios 1, 2, 3, and 4 have groups of *xy*-space outliers. The *xy*-space outlier is an observation with an unusual value in both the regressor variable(s) and response variable. In scenario 1, there is one group of *xy*-space outlying observations with the regressor variable values (x) approximately 20. There is also one group of *xy*-space outlying observations in scenario 2 but with the x values approximately 30.

While, in scenario 3 there are two groups of *xy*-space outlying observations with the x values approximately 20 and 0. There are also two groups of *xy*-space outlying observations in scenario 4 but with the x values approximately 30 and -10. On the other hand, scenario 5 has a group of *x*-space outlying observations with the x values approximately 30. The *x*-space outlying observations meaning that, observation is outlying in the regressor variable values only. In the last scenario, which is scenario 6, there are one *x*-space outlying observations and one *xy*-space outlying observations both with the x values approximately 30.

Table 3.1 shows various regression conditions for the six scenarios. The factors and corresponding levels were chosen so that the performance of the identification method could be tested in a wide variety of regression conditions for each outlying scenario. Since there are an infinite number of regression conditions, priority was given to regression conditions that are "typical".

Table 3.1: Factors and levels for the simulated data sets

Factor	Levels
Number of regressor variables (p)	1, 2, 6
Number of observations in data set (n)	20, 40, 60
Percentage of outlying observations	10%, 20%
Outlier distances	5σ, 10σ

In each scenario, the outliers were placed away from the inliers by a specified distance. The "outlier distances" were measured in standard deviations of inlying observations ($\sigma = 1$). Two outlying distances were considered which at 5 standard deviations and 10 standard deviations.

For each of the simulations, the value of all regression coefficients was set to equal 5 to guarantee there was a statistically significant slope in the regression line (plane). The values of the inlying or "clean" observations regressor variables were selected at random from the $U(0,20)$ distribution. The distribution of the random error for both the clean and outlying observations were $N(0,1)$.

3.3 Creating Simulated Data Sets

The approach to creating data sets to test the methodology was to randomly generate n regression observations. Of these n observations, n_c "clean" observations were generated and represent the inlying observations, while n_o observations were generated and represent the outlying observations where $n = n_c + n_o$. Next, we will show how the data sets used in this simulation study were created for the case of one regressor variable. However, the same methodology can be extended to the case of multiple regression $(p > 1)$ data sets. Every outlier scenario has either one or two groups of outlying observations. The following will illustrate how these groups of outliers were formed.

3.3.1 One Group of Outlying Observations

The n_c clean observations were generated according to the model

$$y_{ic} = \beta_0 + \beta_1 x_{ic} + \varepsilon_i, \quad i = 1,\ldots,n_c \tag{3.1}$$

where x_{ic} is $U(0,20)$ and ε_i is $N(0,1)$ with $\beta_0 = 0$ and $\beta_1 = 5$. The n_o outlying observations were generated according to the model

$$y_{io} = \beta_0 + \beta_1 (\bar{x}_c + x\text{shift}) + y\text{shift} + \varepsilon_i, \quad i = 1,\ldots,n_o \tag{3.2}$$

where ε_i is $N(0,1)$. The term $(\bar{x}_c + x\text{shift})$ allows the outliers to be placed at a specified location in the x-space. The $y$shift term allows the outliers to be placed at a specified distance away from the inliers in the y-space. Note that $y$shift is the number of standard deviations that the outliers are placed away from the clean observations.

The average of a sample of clean variable values created from the $U(0,20)$ will be approximately 10. For example, to create outlying observations with x values that approximately 20, $x$shift was set to 10. Then, to create outlying observations with y values that are approximately 10 standard deviations away from the clean observations, $y$shift was set to 10. If the outlying observation was in the x-space only, the $y$shift was set to zero. An example of this data set (scenario 1, $n = 20$, and 10% outlying) is given in Table 3.2 and the regression graph in Figure 3.2. It should be noted that $x$shift is in absolute units while $y$shift is in standard deviation units.

Table 3.2: Example of one group of *xy*-space outliers (19, 20)

Obs.	y	x
1	5.446120	25.745060
2	14.979626	58.761800
3	1.970022	15.384090
4	4.511679	19.981200
5	17.091321	82.968340
6	10.750279	53.063810
7	15.631164	86.254420
8	17.828718	110.085210
9	11.628712	61.031740
10	16.314657	88.381990
11	5.695668	14.885710
12	14.575002	92.919310
13	19.909402	97.987210
14	13.556771	71.464600
15	16.405463	77.679850
16	12.631480	66.270780
17	13.206961	74.310600
18	8.443224	31.924210
19	22.525134	228.186350
20	22.444198	225.512960

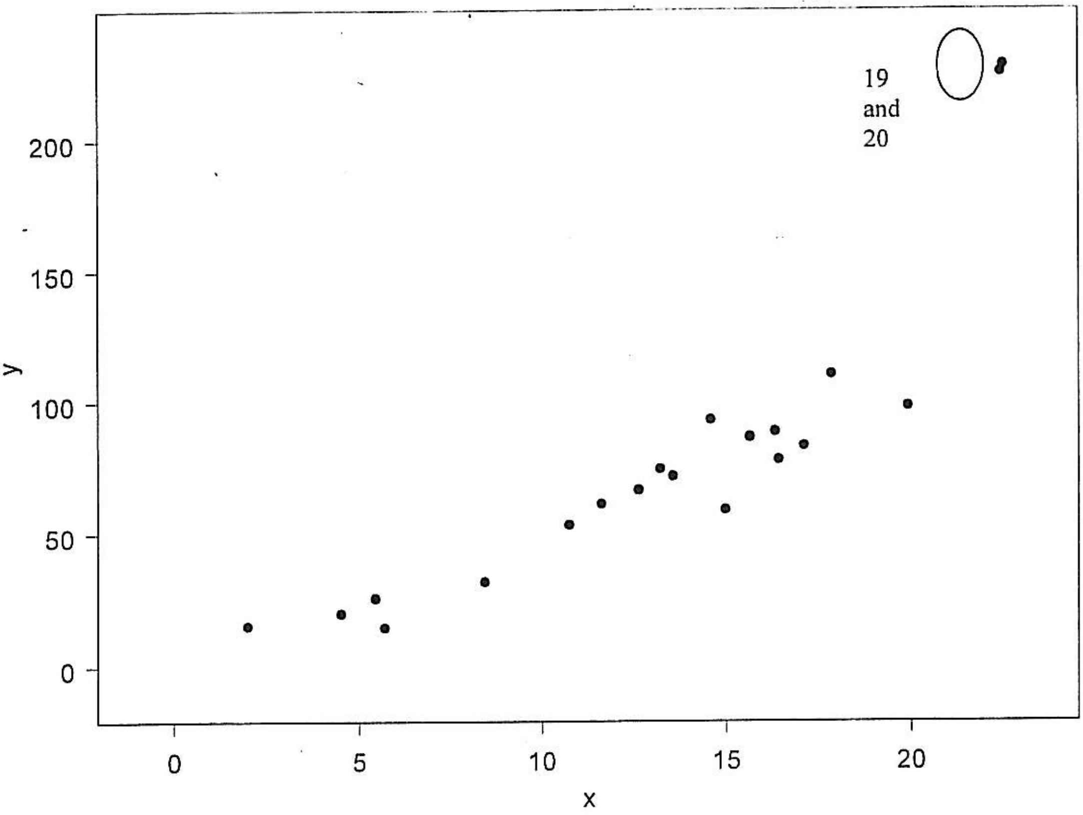


Figure 3.2: Example of one group of *xy*-space outliers (19, 20) for data in Table 3.2

3.3.2 Two Groups of Outlying Observations

The data sets containing two outlying groups were created in a similar manner. Again, the clean observations were created according to model (3.1). The two groups of outlying observations were created using the following models

$$y_{io1} = \beta_0 + \beta_1(\bar{x}_c + x\text{shift1}) + y\text{shift1} + \varepsilon_{io1}, \quad i = 1,\ldots,n_{o1} \qquad (3.3)$$

$$y_{io2} = \beta_0 + \beta_1(\bar{x}_c + x\text{shift2}) + y\text{shift2} + \varepsilon_{io2}, \quad i = 1,\ldots,n_{o2} \qquad (3.4)$$

where n_{o1} and n_{o2} are the sizes for outlying group 1 and outlying group 2 respectively.

For example consider scenario 3. To create outlying observations with x values that approximately 20, $x$shift was set to 10. To create outlying observations with y values that are approximately 10 standard deviations greater than the clean observations, $y$shift was set to 10. To create outlying observations with x values that approximately 0, $x$shift was set to -10. To create outlying observations with y values that are approximately 10 standard deviations less than the clean observations, $y$shift was set to -10. An example of this two outlying group data set with n = 20 and 10% outlying is shown in Table 3.3 and the regression graph in Figure 3.3.

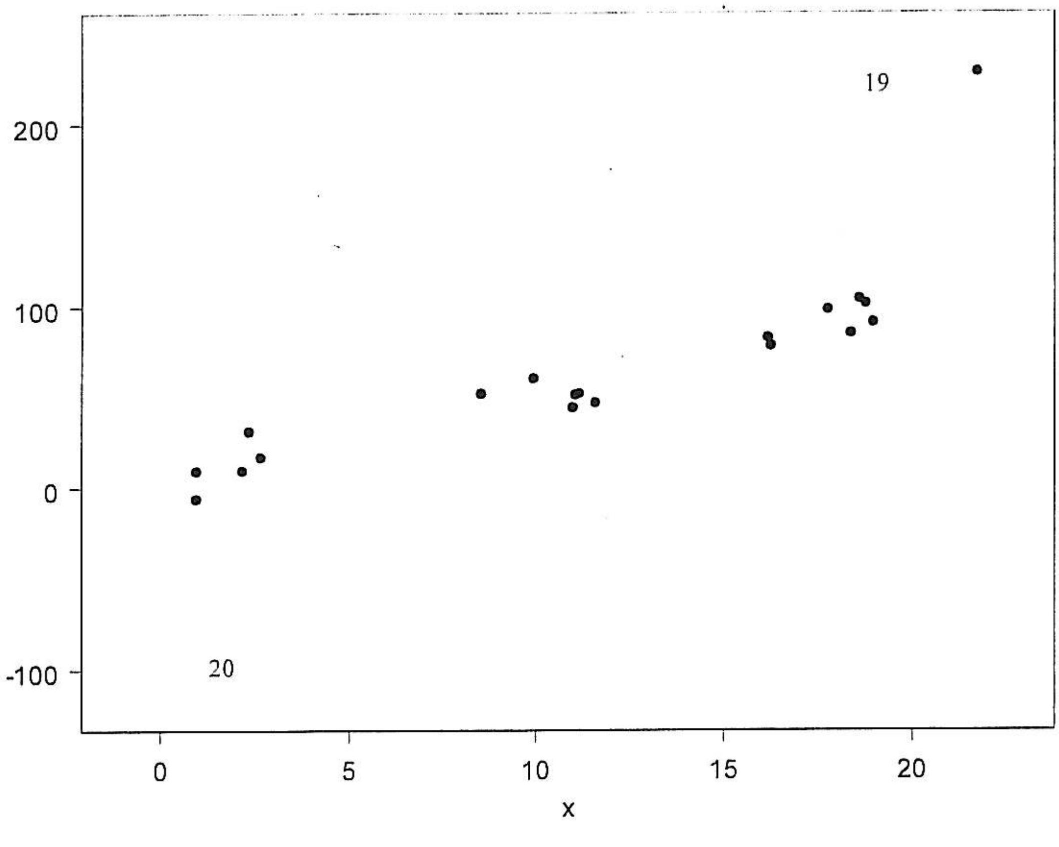


Figure 3.3: Example of two groups of xy-space outliers (19, 20) for data in Table 3.3

Table 4.3: Example of two groups of *xy*-space outliers (19, 20)

Obs.	y	x
1	16.162502	82.531400
2	2.647700	16.709300
3	10.994522	43.841952
4	2.329006	31.021920
5	11.168700	51.698600
6	18.582858	104.269000
7	0.944700	-6.283140
8	9.951488	60.173599
9	18.358800	84.994312
10	8.549300	51.629500
11	18.743107	101.755200
12	11.074729	50.865764
13	0.952399	8.977400
14	11.596495	46.480400
15	17.739295	98.288252
16	18.947655	91.137074
17	2.159500	9.349378
18	16.236994	78.118742
19	21.739419	228.569682
20	1.253974	-100.611872

3.4 Performance Measure

Recall that, the two fundamental problems with outlier identification techniques, in the presence of multiple outliers are masking and swamping. If the purpose of the analyst is to identify influential subsets of observations, masking is more of a problem than swamping. That is, if the candidate set of outlying observations does not contain true outliers, this is more "costly" than the additional cost of computing the influence inliers.

For that reason, in this research "success" will mean that the method successfully identified all of the outlying observations (no masking occurred). If the method is successful but also includes inlying observations in the candidate set of outliers (swamping occurs), this will be noted as "false alarm". Both the detection capability and false alarm rate are reported for 1000 sets of data (or replications). Figure 4.4 illustrates how the performance of this method was assessed. All code development and simulations done using S-PLUS 2000 were shown in Appendix A, C, E and G.

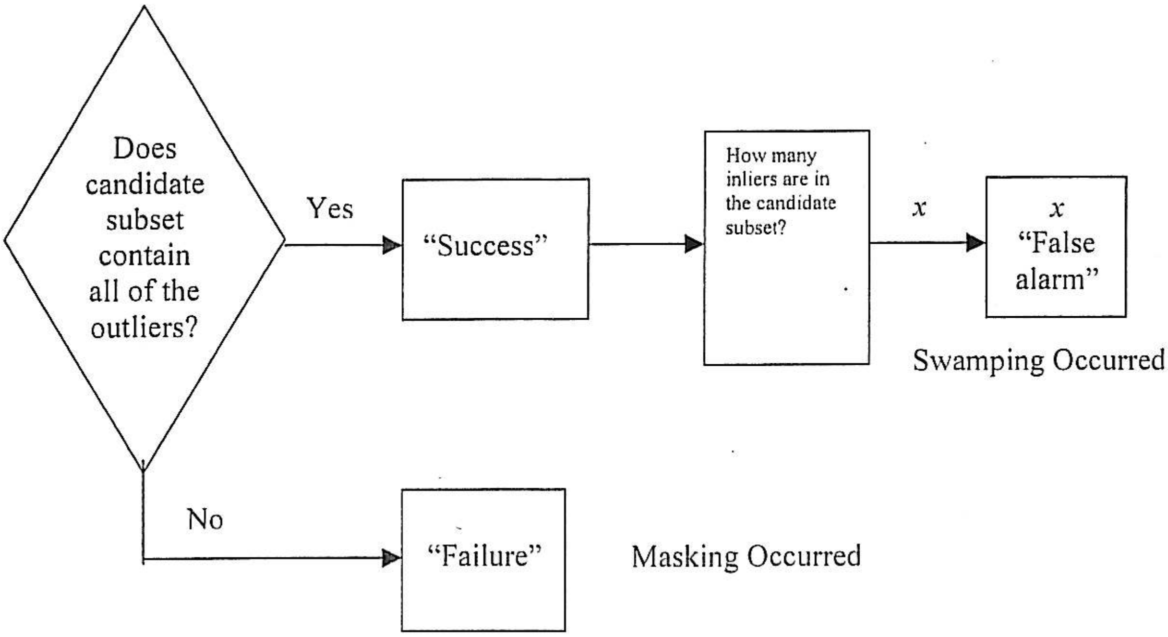

Does candidate subset contain all of the outliers?
Yes
"Success"
How many inliers are in the candidate subset?
x
x
"False alarm"
Swamping Occurred
No
"Failure"
Masking Occurred

Figure 3.4: Flowchart summarizing performance assessment of methodology

CHAPTER 4

SEBERT, MONTGOMERY AND ROLLIER CLUSTERING ALGORITHM

4.1 Introduction

This chapter discusses the Sebert et al. (1998) clustering algorithm for identifying multiple outliers in linear regression. This methodology is based on clustering the points in the plots of residuals versus predicted values, in building linear regression models. Although these plots are primarily used for assessing model adequacy, it will be shown that they can be equally valuable tool for identifying multiple outlying observations. The algorithm is described and is shown to perform well on classical multiple outlier data sets. Also, the performance characteristics of the proposed methodology are demonstrated and explored by applying the procedure to simulated data sets that have various outlier scenarios.

4.2 Clustering Overview

"Cluster analysis" is the generic name for a wide variety of procedures that can be used to create a classification. More specifically, clustering method is a multivariate statistical procedure that starts with a data set containing information about a sample of entities and attempts to reorganize these entities into relatively homogenous groups. In other words, cluster analysis discovers the natural groupings of the entities (items or variables). Similarly, Hartigan (1975) refered the word clustering as "the grouping of similar object" and noted "clustering techniques were first developed in an applied field (biological taxonomy) and significance tests, probability models, loss functions, or optimal procedure".

Cluster analysis begins by taking a set of *n* observations on *p* variables. Next a measure of similarity between observations is obtained. Then a set of rules is employed that group the observations based on their inter-observation similarities. There are three primary decisions the analyst has to make before clustering multivariate data. First, one must decide what point or variables to use. Second, the measure of similarity to use and third, the clustering algorithm to use.

4.2.1 Similarity Measure

In order to group the items (or variables) into their natural groupings, it is necessary to have a measure of "closeness" or "similarity" or a measure of dissimilarity between the items (or variables). Aldenderfer and Blashfield (1984) described four types of similarity measure: correlation coefficients, distances measures, association coefficients and probabilistic similarity coefficients.

Each of these methods has advantages and disadvantages that must be considered before a decision is made to use one. Most commonly used is to compute the measure of distance. Among the more popular representations of distance is Euclidean distance, defined as

$$d_{ij} = \sqrt{\sum_{k=1}^{p} \left(x_{ik} - x_{jk}\right)^2} \tag{4.1}$$

where d_{ij} is the distance between observation *i* and *j*, and x_{ik} is the value of the *k*th variable for the *i*th observation. Other types of distance is the Manhattan distance, or city-block metric, which is defined as

$$d_{ij} = \sum_{k=1}^{p} \left| x_{ik} - x_{ij} \right| . \tag{4.2}$$

Other metrics can be defined, but most are specifics forms of the special class of metric distance functions known as Minkowski metrics, defined in a general form as

$$d_{ij} = \left(\sum_{k=1}^{p} \left| x_{ik} - x_{ij} \right|^{r} \right)^{1/r} \tag{4.3}$$

The other distance is called generalized distance (Mahalanobis), which is defined as

$$d_{ij} = \left(\mathrm{X}_{\mathrm{i}} - \mathrm{X}_{\mathrm{j}} \right) \sum {}^{-1} \left(\mathrm{X}_{\mathrm{i}} - \mathrm{X}_{\mathrm{j}} \right) \tag{4.4}$$

where Σ is the pooled within-groups variance-covariance matrix, and X_{i} and X_{j} are vectors of the values of the variables for observation *i* and *j*.

Johnson and Wichern (1982) and Everitt (1993) noted that the Euclidean distance is the most widely accepted and commonly used measure of similarity when trying to find groups among multivariate observations. The Euclidean distance is popular because of its intuitive appeal as a similarity measure. That is, a relatively small distance should separate similar observations; while dissimilar a relatively large distance should separate observations.

4.3 A Review on Clustering Method

The primary reason for the use of cluster analysis is to find groups of similar entities in a sample data. These groups are conveniently referred to as clusters. Clusters have certain properties. Sneath and Sokal (1975) as pointed out in Aldenderfer and Blashfield (1984) have described a number of these properties, the most important of which are density, variance, dimension, shape and separation.

Density is a property of cluster that defines it as a relatively thick swarm of data points in a space when compared to other areas of the space that may have comparatively few or no points. Variance is the degree of the dispersion of the points in this space from the center of the cluster. This property can simply describe the relative nearness of points to one another in the data space. Cluster can be said to be 'tight' when all points are near the centroid, or they may be 'loose' when the data points are dispersed from the center.

Dimension is a property closely related to variance; if a cluster can be identified, it is then possible to measure its 'radius'. Shape is simply the arrangement of points in the space. The typical conception of the shape of clusters is that they are hyperspheres or ellipsoids; many different kinds of shapes, such as elongated clusters are possible.

The followings are some of the clustering methods that have been developed according to Aldenderfer and Blashfield (1984):

1. Hierarchical
2. Iterative partitioning
3. Density search
4. Factor analytic
5. Clumping
6. Graph theoretic

Each of these methods represent a different perspective on the formation of groups, and the results obtained can be very different when different methods are applied on the same data. What is important to remember when faced with the difficult choice of which clustering method to use is that the method must be compatible with the desired nature of the classification, the variables to be used, and the similarity measure used to estimate the resemblance between cases if one is required.

Sebert et al. (1998) have shown that clustering methods are sensitive to outliers. The most popular method used in order to detect the presence of outliers is hierarchical agglomerative method. This method will be discussed briefly in this project.

4.3.1 Hierarchical Clustering Methods

Hierarchical clustering is useful if the analyst has no prior ideas about how many clusters he expects or might like to have. These methods operate on similarity matrix to construct a tree depicting specified relationships among entities. These methods are divided into two groups, that is the agglomerative and the divisive. The agglomerative methods build tree from branches to root, while the divisive methods begin at the root and work toward the branches.

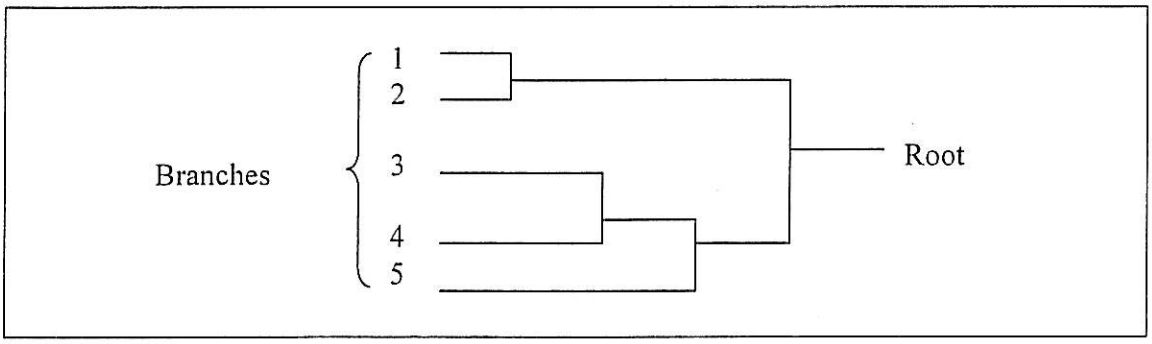


Figure 4.1: Branches and root in hierarchical clustering methods

The agglomerative hierarchical methods proceed with a series of successive mergers with the individual items as clusters. The most similar objects are first grouped and these initial groups are merged according to their similarity measures. Eventually, as the similarity decreases, all subgroups are fused into a single cluster. These cluster are nested, that is the merging are permanent. The divisive hierarchical methods are just the opposite of agglomerative. The initial group consist of all the objects and then divided into two subgroups such that the objects in one subgroup are far from the objects in the other. The process of division will continue until there are many subgroups as objects.

These clustering methods produce nonoverlapping clusters. The results of both agglomerative and divisive may be displayed in the form of two-dimensional diagram known as dendogram or tree diagram. The dendogram illustrates the merges or divisions, which have been made at successive level. It seems that all the hierarchical clustering methods treated in the literature are alternative forms or minor alterations of three major clustering concepts:

1. Linkage methods:
   - Single linkage – use the smallest dissimilarity between a point in the first cluster and a point in the second cluster.
   - Average linkage – use the average of the dissimilarities between the points in one cluster and the points in the other cluster.
   - Complete linkage – use the largest dissimilarity between a point in the first cluster and a point in the second cluster.

2. Centroid methods
   - use the Euclidean distances as the dissimilarity between two centroids of the clusters.

3. Error sum of squares or variance methods: Ward's methods
   - the mergers at each stage are chosen so as to minimize the error sum of squares between two clusters summed over all the variables.

All of these methods are suitable in clustering data units, but the linkage methods are suitable in clustering both data units and variables. There are at least 12 different linkage forms that have been proposed, three have become widely popular; single linkage, average linkage and complete linkage. Figure 4.2 shows the different representations of the linkage clustering methods.

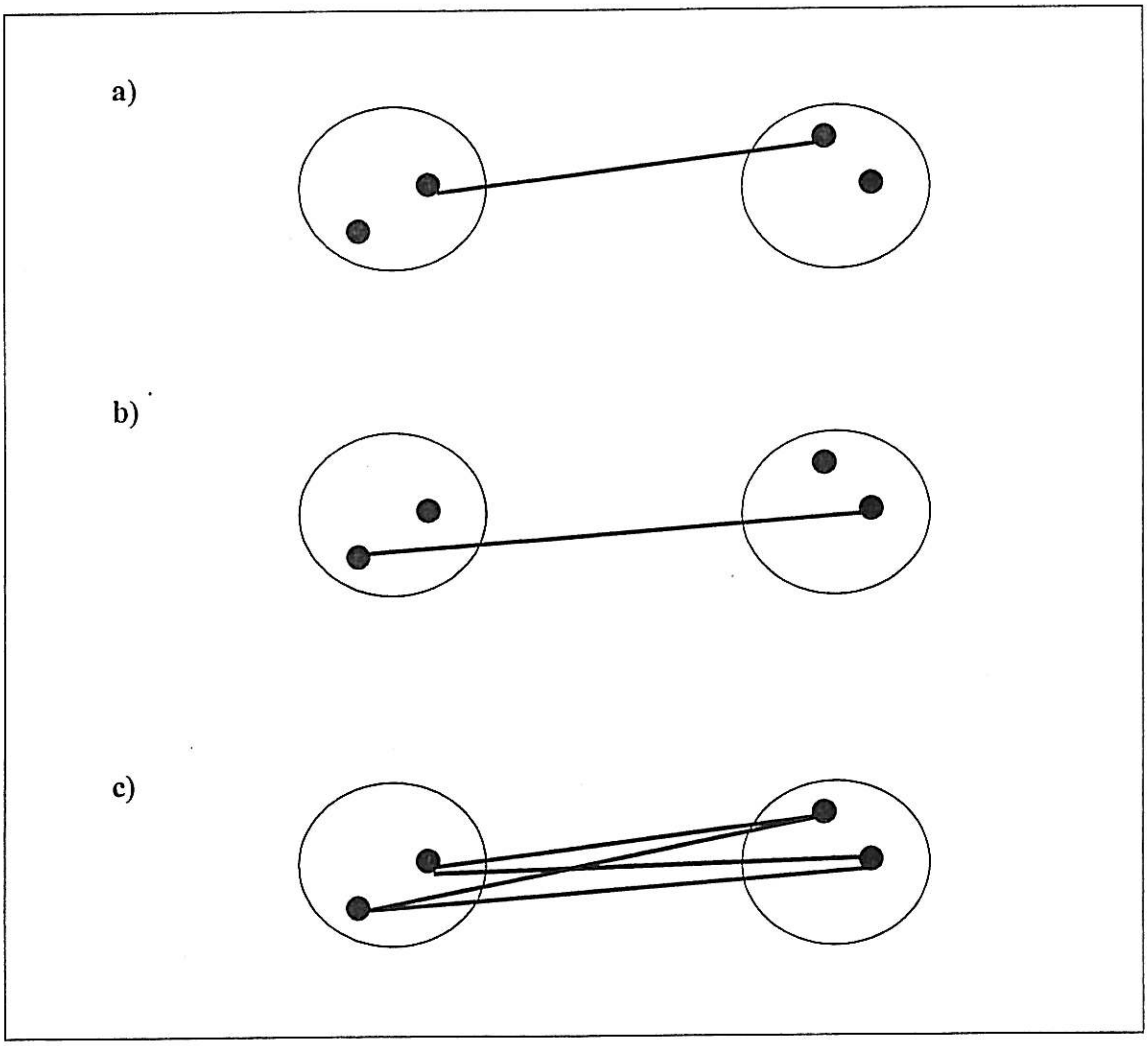


Figure 4.2: Representation of a) Single linkage b) Complete linkage c) Average linkage

The following are the general agglomerative hierarchical clustering algorithm for grouping N objects (items or variables):

1. Start with N cluster, each containing a single entity and an $N \times N$ symmetric matrix of distances (or similarities), $D = \{d_{ik}\}$

2. Search the distance matrix for the nearest (most similar) pair of clusters. Let the distance between "most similar" clusters U and V be d_{UV}.

3. Merge clusters U and V. Label the newly formed cluster (UV). Update the entries in the distance matrix by
   - Deleting the rows and columns corresponding to clusters U and V and,
   - Adding a row and column giving the distances between cluster (UV) and the remaining clusters.

4. Repeat steps 2 and 3 for $N-1$ times. All objects will be in one cluster at termination of the algorithm. Record the identity of clusters that are merged and the levels (distances or similarities) at which the merges take place.

Different agglomerative methods are implemented by varying the procedures used for defining the most similar pair in step 2 and for updating the revised similarity matrix at step 3. The stability of hierarchical solutions can be checked by applying the clustering algorithms before and after small errors (perturbation) have been added to the data units. The solution is stable if the clustering before and after perturbation agrees. Almost all hierarchical agglomerative methods are vary to monotonic transformations except for single linkage method.

The best clustering algorithm for a certain situation largely depends upon the type of clusters that are in the data set. There are many types of clusters. Figure 4.3 shows the different types of clusters, as described by Kaufmann and Rousseouw (1990).

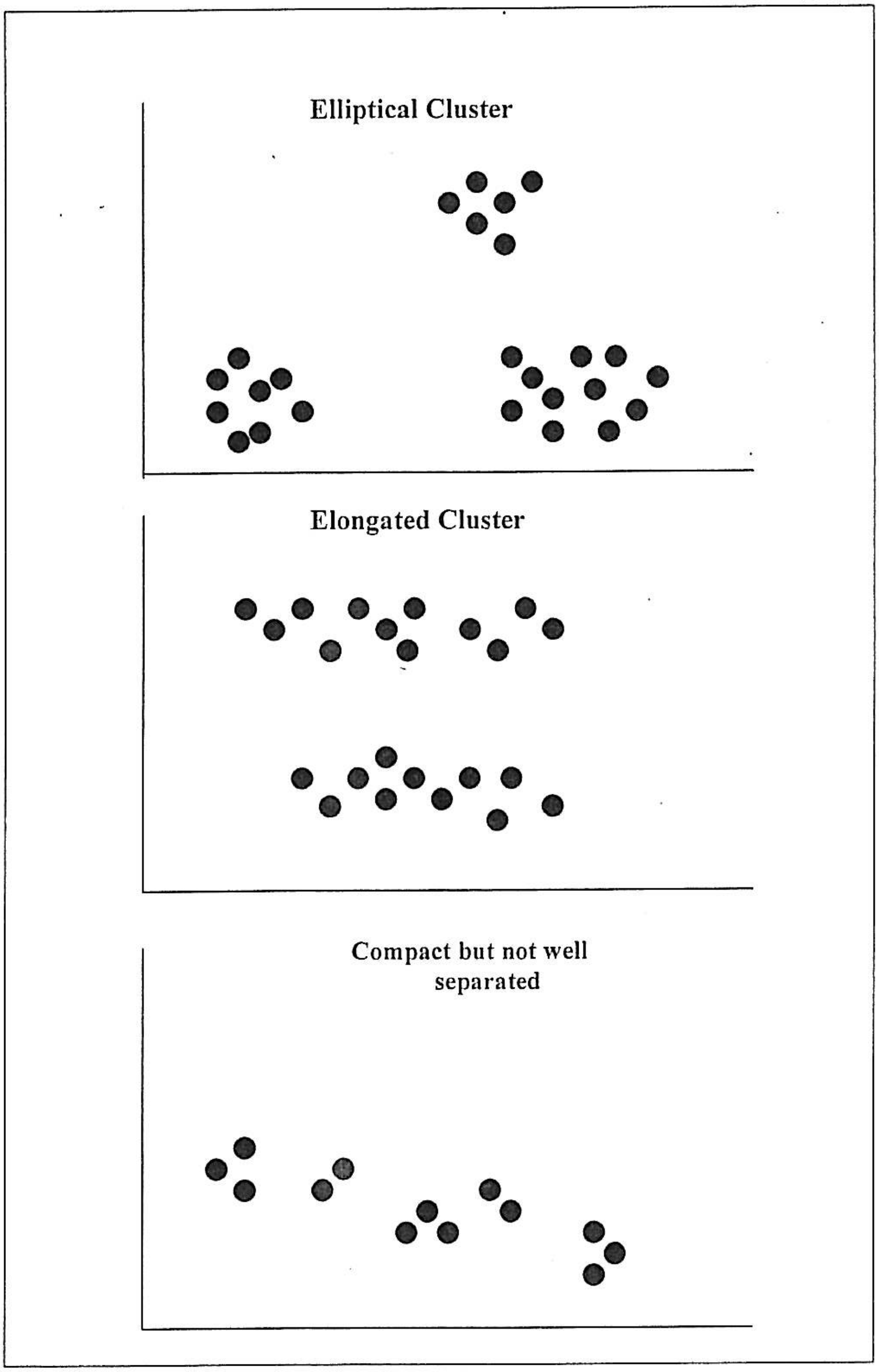


Figure 4.3: Examples of types of clusters

If one really wants to find elongated clusters then the single linkage is the clustering methods to use. The average linkage is aimed at finding elliptical clusters. Meanwhile, the complete linkage tends to produce a very compact cluster that is the clusters have small diàmeters. Sebert et al. (1998) proposed to use single linkage clustering method in other to detect the presence of multiple outliers in linear regression.

4.3.2 Single Linkage

This method is the easiest and most fruitful mathematically in constructing clusters and has been widely used since it was first introduced by Florek et al. (1951) and Sneath (1957) as pointed out in Aldenderfer and Blashfield (1984). Single linkage method operates on a matrix distance (or similarity) coefficients between groups, which is revised as each successive level of the hierarchical is generated. The term single linkage is used because two clusters are joined if any of the distances between the objects in different clusters is sufficiently small, that is if there is a single link between the clusters. The inputs to a single linkage algorithm can be distances or similarities between pairs of objects. Groups are formed from individual entities by merging nearest neighbors, where the term nearest denotes smallest distance or largest similarities.

Single linkage is incapable of delineating poorly separated clusters. The single linkage method is one of the very few clustering techniques, which can outline non-ellipsoidal clusters. The presence of scattered intermediate points lying between denser clusters of points tends to cause these clusters to link together prematurely and is sometimes called “chaining”.

It is well established (refer to Johnson and Winchern, 1982; Kaufman and Rousseouw, 1990; Everitt, 1993) that, single linkage is a good clustering algorithm for identifying elongated or "chain-like" clusters. In most regression situations one would assume that an approximate linear relationship exists between the regressor variables and the response variable and that these observations form a chain-like cluster. Observations that do not follow this linear pattern can be thought of as "outliers".

Many researches fault the single linkage approach because of its "chaining" tendency. Everitt (1993) noted, however, that "to call chaining a defect is rather misleading, since chaining is simply a description of what single linkage does. In some cases it may lead to a more accurate picture of the structure in the data than other methods". This thought is also consistent with Johnson and Winchern (1982) who noted that if in fact one expects long chain-like clusters, single linkage is one of the only clustering techniques that can find them. Therefore, in regression situation, it is the chain-like structure of the inlying observations that single linkage clustering is well suited to identify.

4.3.2.1 Single Linkage Clustering Algorithm

Initially, we must find the smallest distance in $D = \{d_{ik}\}$ and merge the corresponding objects, say *U* and *V*, to get (*UV*). For step 3 of the general algorithm, the distances between (*UV*) and other cluster *W* are computed by

$$d_{(UV)W} = \min\{d_{UW}, d_{VW}\} \qquad \text{or} \qquad S_{(UV)W} = \max\{S_{UW}, S_{VW}\}$$

Here quantities d_{UW} and d_{VW} are distances between the nearest neighbors of clusters U and W and clusters V and W, respectively. Figure 4.4 shows the general steps in the single linkage clustering algorithm.

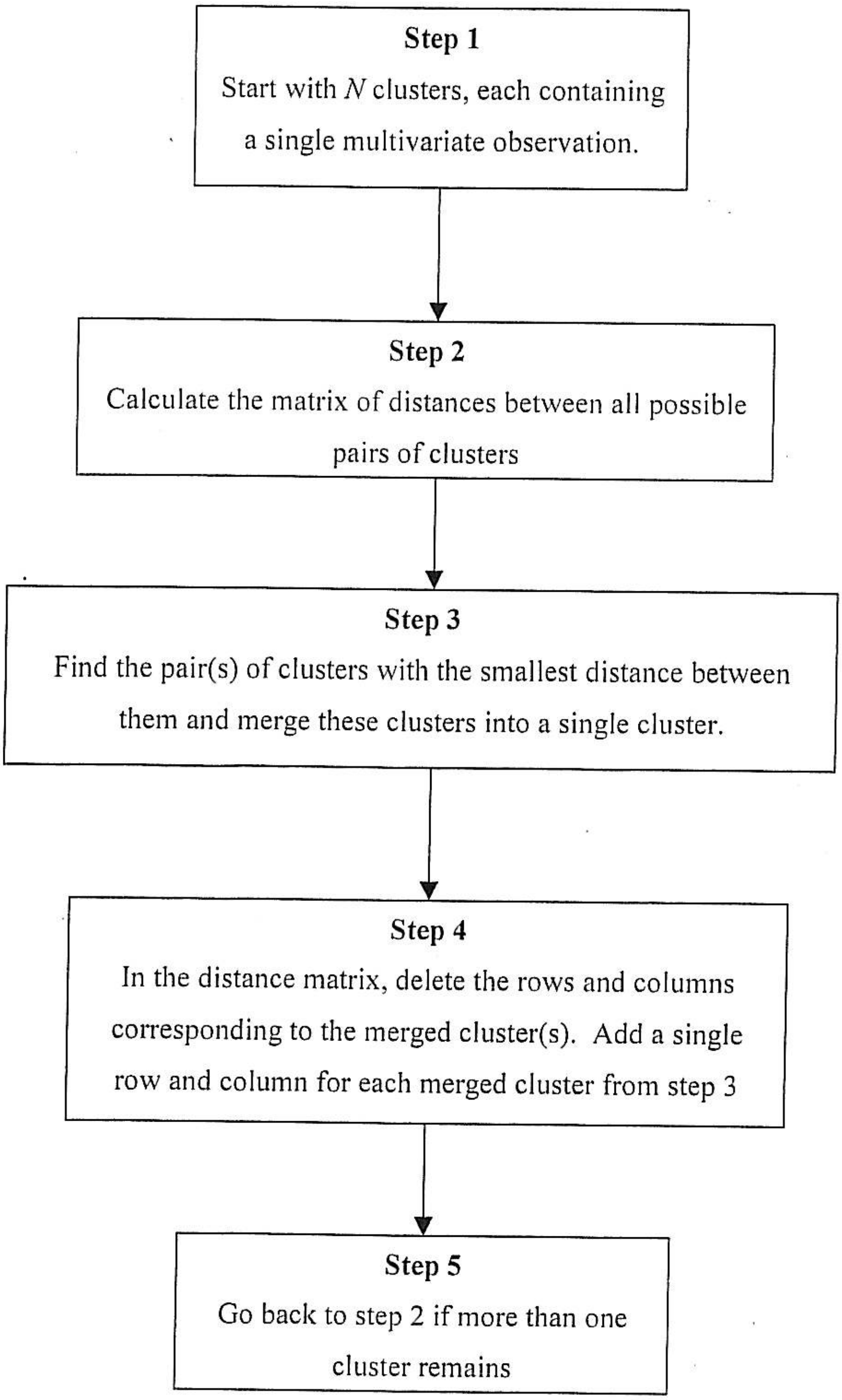

Step 1
Start with N clusters, each containing a single multivariate observation.
Step 2
Calculate the matrix of distances between all possible pairs of clusters
Step 3
Find the pair(s) of clusters with the smallest distance between them and merge these clusters into a single cluster.
Step 4
In the distance matrix, delete the rows and columns corresponding to the merged cluster(s). Add a single row and column for each merged cluster from step 3
Step 5
Go back to step 2 if more than one cluster remains

Figure 4.4: Steps in single linkage clustering algorithm

The results of single linkage clustering algorithm can be seen on a dendogram, or what is commonly referred to as "cluster tree" diagram. The branches in the tree represent clusters. The branches come together (merge) at nodes whose positions along a distance (or similarity) axis indicate the level at which the fusions occur. The following is an example illustrating the single linkage algorithm using the Euclidean distance as the similarity measure. The observations are shown in Table 4.1 and the scatter plot in Figure 4.5.

Table 4.1: Observation used to illustrate the single linkage clustering algorithm

Observation	X1	X2
1	0.651	0.573
2	-0.535	1.651
3	-0.170	0.606
4	1.450	-2.437
5	0.548	-0.547

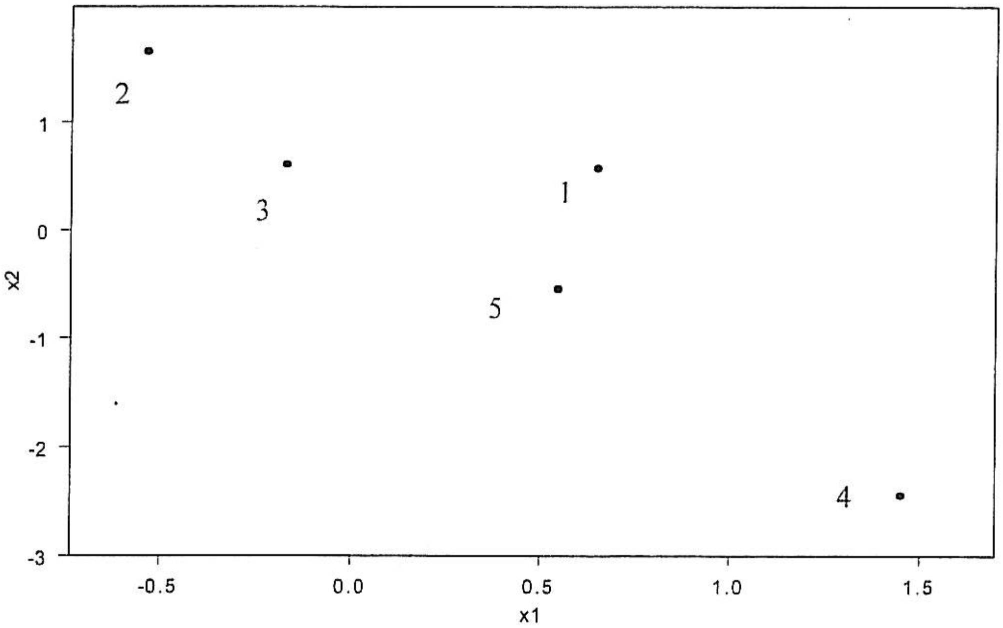


Figure 4.5: Plot of Observations used to illustrate the single linkage clustering algorithm

Since there are 5 observations, initially there are also 5 clusters with one element in each cluster. First, we need to calculate the matrix of distances (similarity matrix) between all possible pairs of clusters. As an example, the distance between observation 1 and 2 is calculated as

$$\begin{aligned} d_{12} &= \sqrt{(0.651-(-0.535))^2+(0.573-1.651)^2} \\ &= 1.602710 \end{aligned}$$

The distance for $d_{ij} = d_{ji}$, then the similarity matrix is written as an upper triangular matrix. Table 4.2 (a) shows the similarity matrix.

Table 4.2 (a)

	1	**2**	**3**	**4**	**5**
1	0	1.602710	**0.821663**	3.114242	1.124726
2		0	1.106910	4.544375	2.450325
3			0	3.447354	1.358283
4				0	2.094207
5					0

In single linkage method, the pair of clusters with the smallest distance is merged first. From Table 4.2 (a), the smallest distance is within cluster 1 and 3, which is 0.821663. In this case, cluster 1 and 3 is merged first while the row 1 and column 3 in similarity matrix are deleted. Table 4.2 (b) shows the new similarity matrix with the new row and column added for cluster (1,3).

Table 4.2 (b)

	2	4	5	(1,3)
2	0	4.544375	2.450325	**1.106910**
4		0	2.094207	3.114242
5			0	1.124726
(1,3)				0

Since the single linkage clustering algorithm computes similarity measures based on the "nearest neighbors" therefore the distance between cluster 2 and cluster (1,3) is measured based on observation 2 and observation 3. The distance between cluster 4 and cluster (1,3) is measured based on observation 1 and 4 while the distance between cluster 5 and cluster (1,3) is measured based on observation 1 and 5. According to Table 4.2 (b), cluster 2 and (1,3) is merged since this pair has the smallest distance value of 1.106910. Then, the corresponding column and row of cluster 2 and (1,3) are deleted. Table 4.2 (c) shows the new similarity matrix with the new row and column for cluster (2, (1,3)) added.

Table 4.2 (c)

	4	5	(2, (1,3))
4	0	2.094207	3.114242
5		0	**1.124726**
(2, (1,3))			0

In Table 4.2 (c), the shortest distance is between clusters 5 and (2, (1,3)) (cluster 5 and (1,3)) with value of 1.124726. The next merging is thus between cluster 5 and (2, (1,3)). Table 4.2 (d) shows the new similarity matrix with the new row and column for cluster (5, (2, (1,3))) added.

Table 4.2 (d)

	4	(5, (2, (1,3)))
4	0	**2.094207**
(5, (2, (1,3)))		0

Thus, the final merging is between cluster 4 and cluster (5, (2, (1,3))) where all the observations are put into one cluster. The distance between cluster 4 and cluster (5, (2, (1,3))) is calculated using the distance between cluster 4 and cluster 5 which is 2.094207. Below are the tree diagram illustrating the merges.

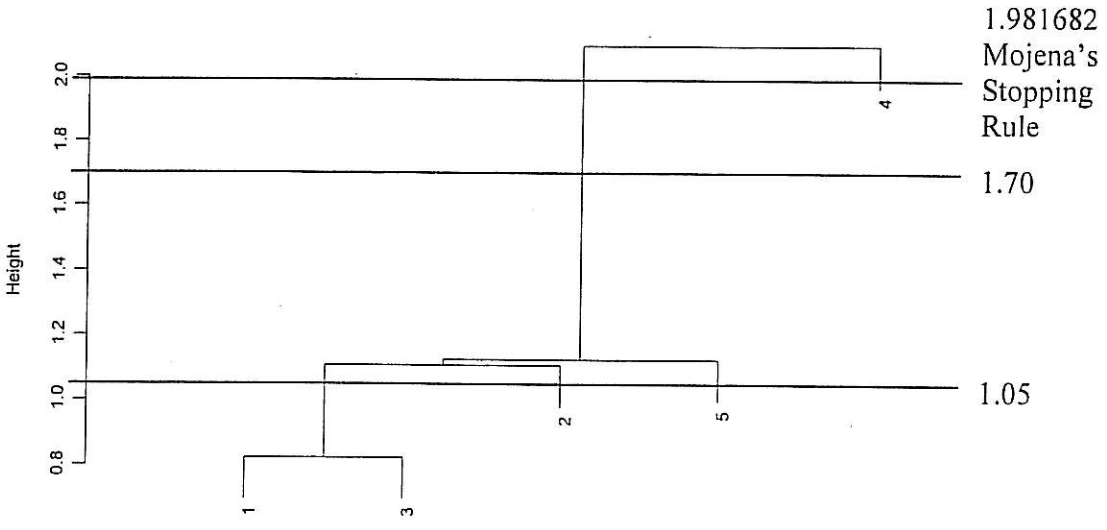


Figure 4.6: Tree diagram to illustrate the single linkage algorithm

One can also use the S-PLUS 2000 computer package to illustrate the single linkage algorithm with the Euclidean distance as the similarity measure. Figure 4.7 shows the output from S-PLUS agglomerative hierarchical clustering and explanation on some of the arguments is given below.

```
        *** Agglomerative Hierarchical Clustering ***
Call:
agnes(x = menuModelFrame(data = SDF41, variables =
        "<ALL>", subset = NULL, na.rm = T), diss = F,
        metric = "euclidean", stand = F, method =
        "single", save.x = T, save.diss = T)
Merge:
      [,1] [,2]
[1,]    -1   -3  → Step 1: Merge between observation 1 and observation 3
                   (Negative sign denote the merging is between individual
                    observations and not clusters at previous stage)
[2,]     1   -2  → Step 2: Merge between cluster from step 1 and observation 2
[3,]     2   -5  → Step 3: Merge between cluster from step 2 and observation 5
[4,]     3   -4  → Step 4: Merge between cluster from step 3 and observation 4

Order of objects: → order of objects appeared on the dendogram read from left
                    to right
[1] 1 3 2 5 4
Height: → distances between merging clusters read from left to right of the
          dendogram
[1] 0.8216629 1.1069101 1.1247262 2.0942072

Agglomerative coefficient: → dimensionless quantity between 0 and 1
[1] 0.4299352
Available arguments:
[1] "order"     "height"    "ac"        "merge"
[5] "order.lab" "diss"      "data"      "call"
```

Figure 4.7: The output from S-PLUS agglomerative hierarchical clustering

The agglomerative coefficient is given by $AC = \frac{1}{n}\sum_{i=1}^{n} l(i)$ where $l(i)$ is the length of the ith observation on the 0 - 1 scales above or below the graphical display. Generally, the AC describes how strong the clustering structure is but the value tends to be larger with the increase of the sample sizes. When the value of AC is very small, the corresponding method has not found the natural structure. But a high value of AC that is close to 1 implies very strong clustering structure has been found but not necessarily the "right" clustering. Inclusion of outlier can make the value of AC very close to 1. But with the help of graphical display such as dendogram can pinpoint the outlier.

4.3.3 Stopping Rule

After a clustering algorithm is used on a data set, the user must decide how many groups (if any) are in the data set. Specifically, the cluster tree must be portioned or "cut" at a certain height. The number of groups depends upon where the tree is cut. Again, consider Figure 4.6. Notice that the number of groups in this data set would be two if the tree is cut at a height 1.70. On the other hand, if the tree is cut at 1.05 the data will be divided into four groups. The "number of groups" problem is practical issue that any user of clustering procedures must deal with.

An extensive research on the different stopping rules was done by Milligan and Cooper (1985). Most of these stopping rules have difficulty in a two clusters scenarios that is, it would seem that a two-cluster case is the most difficult structure for the stopping rules to detect. The stopping rule used in this research is the Mojena's stopping rule (1977). The Mojena's stopping rule is widely known and has been the subject of some limited validation research (Mojena, 1977; and Blashfield and Morey, 1980).

The rule resembles a one-sided confidence interval based on the $N-1$ heights (joining distances) of the cluster tree. Formally, Mojena's stopping rule or "cut height" is $\overline{h}+\alpha s_h$ where $\overline{h}$ is the average of the heights for all $N-1$ clusters, and s_h is the unbiased standard deviation of the heights, and is a specified constant. Mojena initially suggested that α should be specified in the range of 2.75-3.50. However, Milligan and Cooper (1985) in a more comprehensive study, concluded that the best overall performance of Mojena's stopping rule occurs when the value of α is 1.25. Returning to the example in Figure 4.6, $\overline{h}=1.286877$ and $s_h=0.555844$, using $\alpha=1.25$, Mojena's cut height is 1.981682, which result in a two groups solution for the data set. The groups are cluster 4 and cluster (5, (2, (1,3))).

4.4 Sebert et al. Clustering Methodology

It is well known that residual plotted against the corresponding predicted values is a useful tool for judging the adequacy of a regression model. These plots are useful for detecting departures from normality, inequality of variance, the wrong functional specification of the regressor(s), and outliers. The regression user is generally instructed to look for an approximate horizontal band on the plot. If this is found then there are no obvious problems with the fitted regression model. Observations that are not outlying will generally have a linear relationship that can be visualized in the plot of predicted and residual values. That is why one looks for a horizontal band to determine the adequacy of the fitted model. Again, from a single linkage clustering point of view, one is looking for a long, horizontal, chain like cluster. If this type of cluster is seen then one assumes that there are no major model inadequacies.

The methodology is based on using the single linkage algorithm to cluster the points in the predicted values versus residual values plot. Specifically, single linkage will be used because it is the best technique for identifying elongated clusters, which for the purpose of this research, will be the inliers. The steps of the methodology will now be discussed in detail and illustrated with the "Modified Wood Gravity" data set given by Rousseeuw and Leroy (1987) and shown in Table 4.3. The observations 4, 6, 8, and 19 in this data were outliers.

Step 1: Standardize the predicted values and residuals obtained from an ordinary least squares (OLS) fit of the data.

Table 4.4 shows these values for the Modified Wood Gravity data.

Figure 4.8 shows a plot of the standardized predicted values and residuals for the wood data.

Step 2: Using the Euclidean distance between pairs of standardized predicted values and residuals as the similarity measure, cluster the observations using the single linkage clustering algorithm and obtain the cluster tree.

Figure 4.9 shows the output from S-PLUS agglomerative hierarchical clustering and explanation on some of the arguments.

Step 3: Based on Mojena's stopping rule cut the tree and form groups at a height of $\bar{h}+1.25s_h$ where $\bar{h}$ is the average of the tree cluster heights for all $N-1$ clusters, and s_h is the unbiased standard deviation of the heights of the $N-1$ clusters.

For the example data set, $\bar{h}$ is 0.593138 and s_h is 0.291703. Therefore, the cut height on the cluster tree is 0.593138 + 1.25*0.291703 = 0.957767.

The cluster tree and corresponding cut height is shown in Figure 4.10.

Step 4: Identify the group with a majority of the observations in it as the inlying observations. All other observations are outlying observations.

The summary of this methodology is shown in Figure 4.11.

Table 4.3: Modified Wood Gravity Data, Rousseeuw and Leroy (1987)

Obs.	y	x1	x2	x3	x4	x5
1	0.5340	0.5730	0.1059	0.4650	0.5380	0.8410
2	0.5350	0.6510	0.1356	0.5270	0.5450	0.8870
3	0.5700	0.6060	0.1273	0.4940	0.5210	0.9200
4	**0.4500**	**0.4370**	**0.1591**	**0.4460**	**0.4230**	**0.9920**
5	0.5480	0.5470	0.1135	0.5310	0.5190	0.9150
6	**0.4310**	**0.4440**	**0.1628**	**0.4290**	**0.4110**	**0.9840**
7	0.4810	0.4890	0.1231	0.5620	0.4550	0.8240
8	**0.4230**	**0.4130**	**0.1673**	**0.4180**	**0.4300**	**0.9780**
9	0.4750	0.5360	0.1182	0.5920	0.4640	0.8540
10	0.4860	0.6850	0.1564	0.6310	0.5640	0.9140
11	0.5540	0.6640	0.1588	0.5060	0.4810	0.8670
12	0.5190	0.7030	0.1335	0.5190	0.4840	0.8120
13	0.4920	0.6530	0.1395	0.6250	0.5190	0.8920
14	0.5170	0.5860	0.1114	0.5050	0.5650	0.8890
15	0.5020	0.5340	0.1143	0.5210	0.5700	0.8890
16	0.5080	0.5230	0.1320	0.5050	0.6120	0.9190
17	0.5200	0.5800	0.1249	0.5460	0.6080	0.9540
18	0.5060	0.4480	0.1028	0.5220	0.5340	0.9180
19	**0.4010**	**0.4170**	**0.1687**	**0.4050**	**0.4150**	**0.9810**
20	0.5680	0.5280	0.1057	0.4240	0.5660	0.9090

Table 4.4: Standardized least squares predicted values and residuals for Modified Wood Gravity data.

Obs.	Predicted Values	Standardized Predicted Values	Residual	Standardized Residual
1	0.5515	1.1873	-0.0175	-0.8447
2	0.5339	0.7728	0.0011	0.0550
3	0.5400	0.9179	0.0300	1.4478
4	0.4414	-1.4006	0.0086	0.4132
5	0.5238	0.5368	0.0242	1.1678
6	0.4419	-1.3899	-0.0109	-0.5264
7	0.4591	-0.9846	0.0219	1.0563
8	0.4238	-1.8146	-0.0008	-0.0407
9	0.4845	-0.3869	-0.0095	-0.4613
10	0.4960	-0.1174	-0.0100	-0.4835
11	0.5061	0.1201	0.0479	2.3135
12	0.5479	1.1026	-0.0289	-1.3953
13	0.5037	0.0637	-0.0117	-0.5656
14	0.5474	1.0916	-0.0304	-1.4693
15	0.5161	0.3562	-0.0141	-0.6834
16	0.4954	-0.1324	0.0126	0.6102
17	0.5261	0.5914	-0.0061	-0.2969
18	0.4992	-0.0434	0.0068	0.3307
19	0.4271	-1.7371	-0.0261	-1.2625
20	0.5549	1.2665	0.0131	0.6355

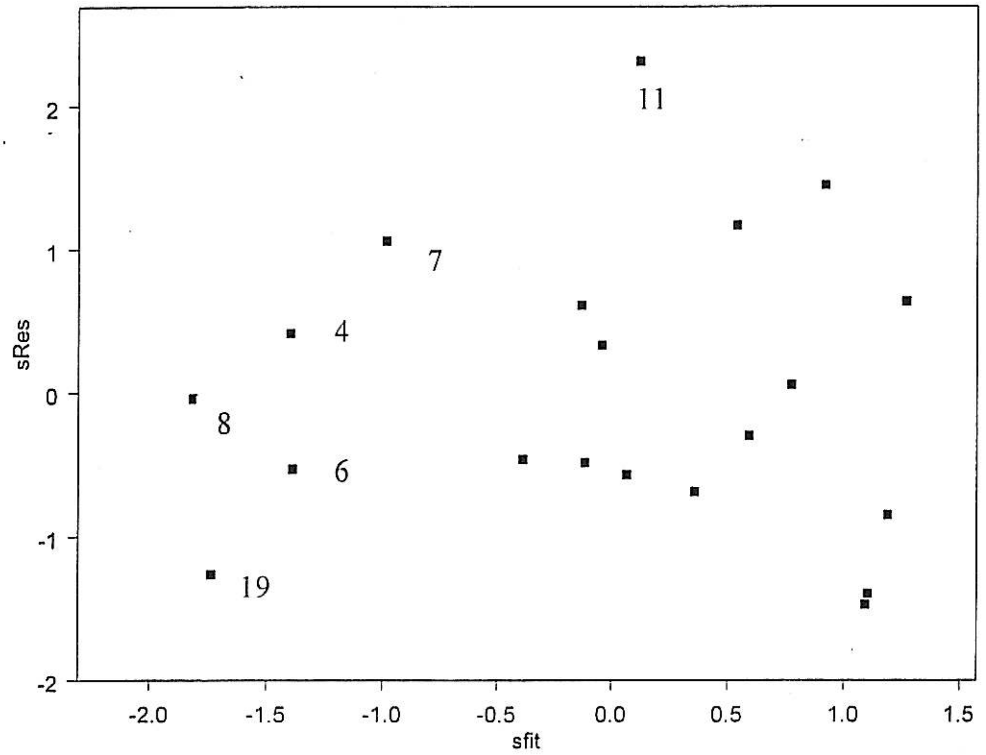
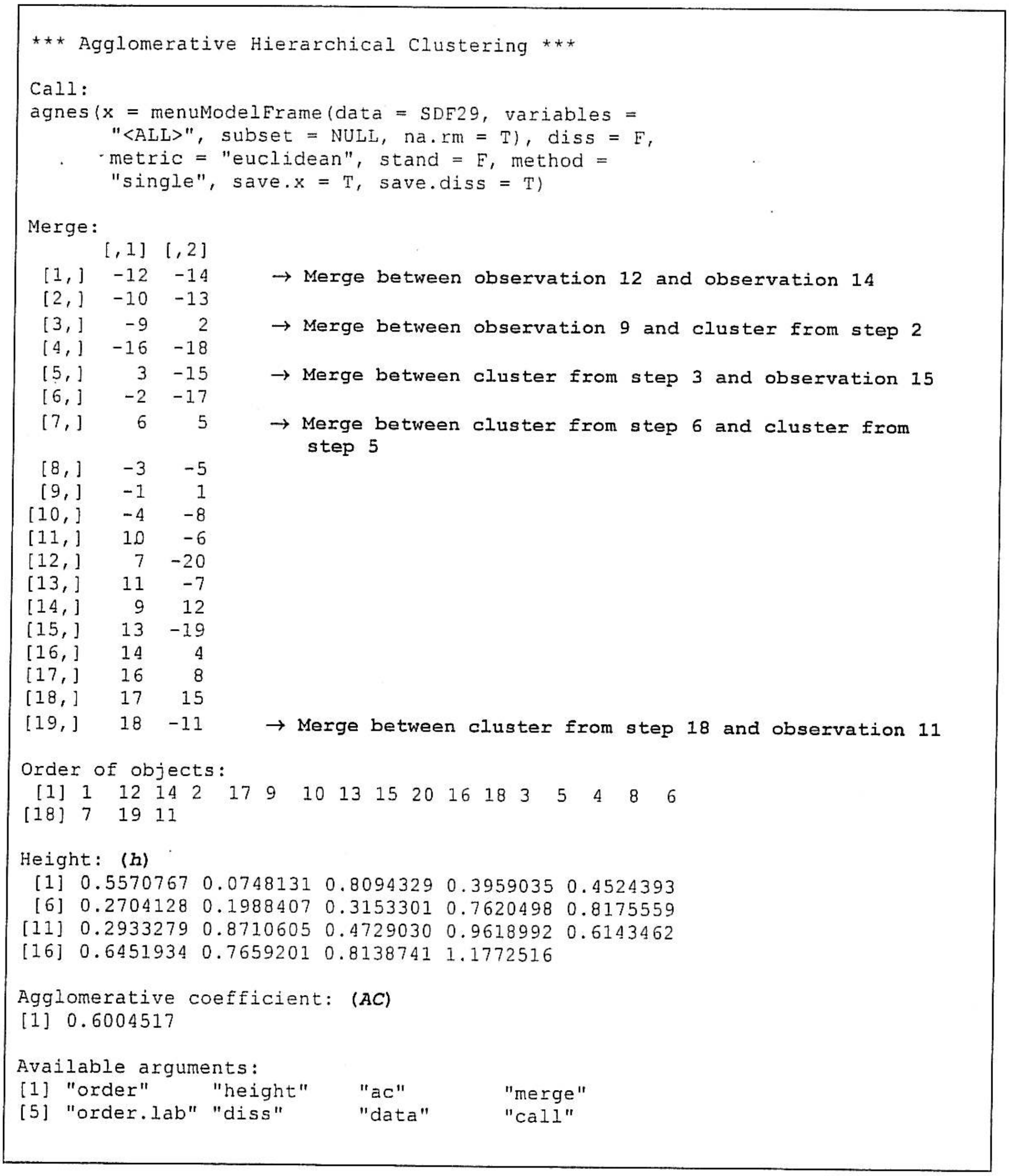


Figure 4.8: Plot of standardized predicted (sfit) and residual (sRes) values for the Modified Wood Gravity data



```
*** Agglomerative Hierarchical Clustering ***

Call:
agnes(x = menuModelFrame(data = SDF29, variables =
       "<ALL>", subset = NULL, na.rm = T), diss = F,
       metric = "euclidean", stand = F, method =
       "single", save.x = T, save.diss = T)

Merge:
       [,1] [,2]
 [1,]   -12  -14      → Merge between observation 12 and observation 14
 [2,]   -10  -13
 [3,]    -9    2      → Merge between observation 9 and cluster from step 2
 [4,]   -16  -18
 [5,]     3  -15      → Merge between cluster from step 3 and observation 15
 [6,]    -2  -17
 [7,]     6    5      → Merge between cluster from step 6 and cluster from
                        step 5
 [8,]    -3   -5
 [9,]    -1    1
[10,]    -4   -8
[11,]    10   -6
[12,]     7  -20
[13,]    11   -7
[14,]     9   12
[15,]    13  -19
[16,]    14    4
[17,]    16    8
[18,]    17   15
[19,]    18  -11      → Merge between cluster from step 18 and observation 11

Order of objects:
 [1] 1  12 14 2  17 9  10 13 15 20 16 18 3  5  4  8  6
[18] 7  19 11

Height: (h)
 [1] 0.5570767 0.0748131 0.8094329 0.3959035 0.4524393
 [6] 0.2704128 0.1988407 0.3153301 0.7620498 0.8175559
[11] 0.2933279 0.8710605 0.4729030 0.9618992 0.6143462
[16] 0.6451934 0.7659201 0.8138741 1.1772516

Agglomerative coefficient: (AC)
[1] 0.6004517

Available arguments:
[1] "order"     "height"     "ac"        "merge"
[5] "order.lab" "diss"       "data"      "call"
```

Figure 4.9: The output from S-PLUS agglomerative hierarchical clustering for Modified Wood Gravity data.

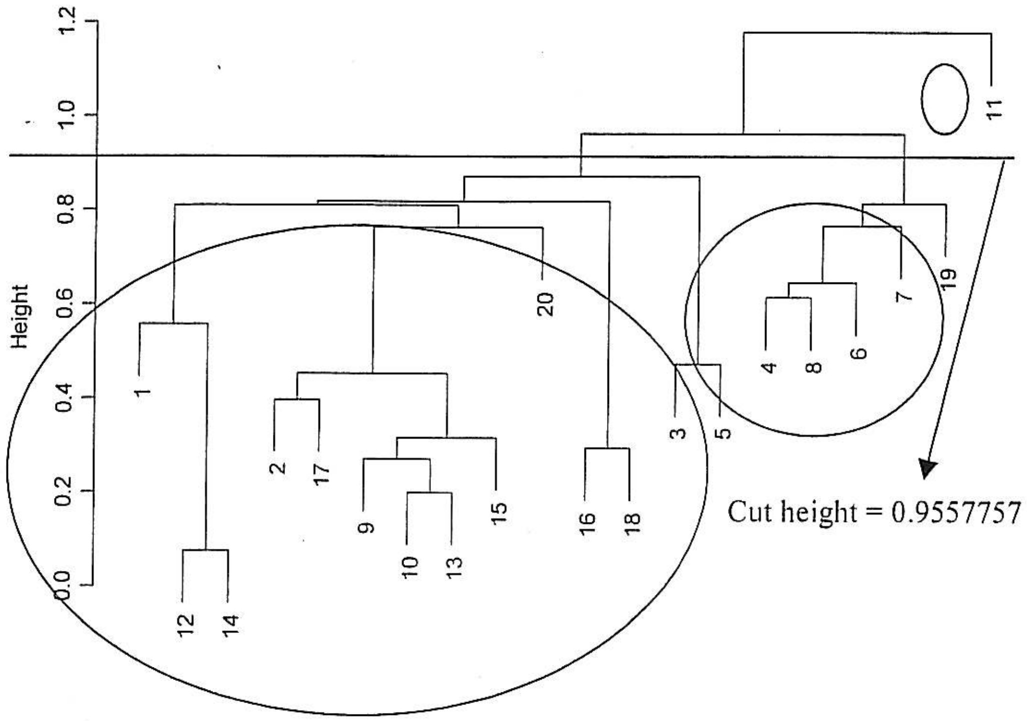
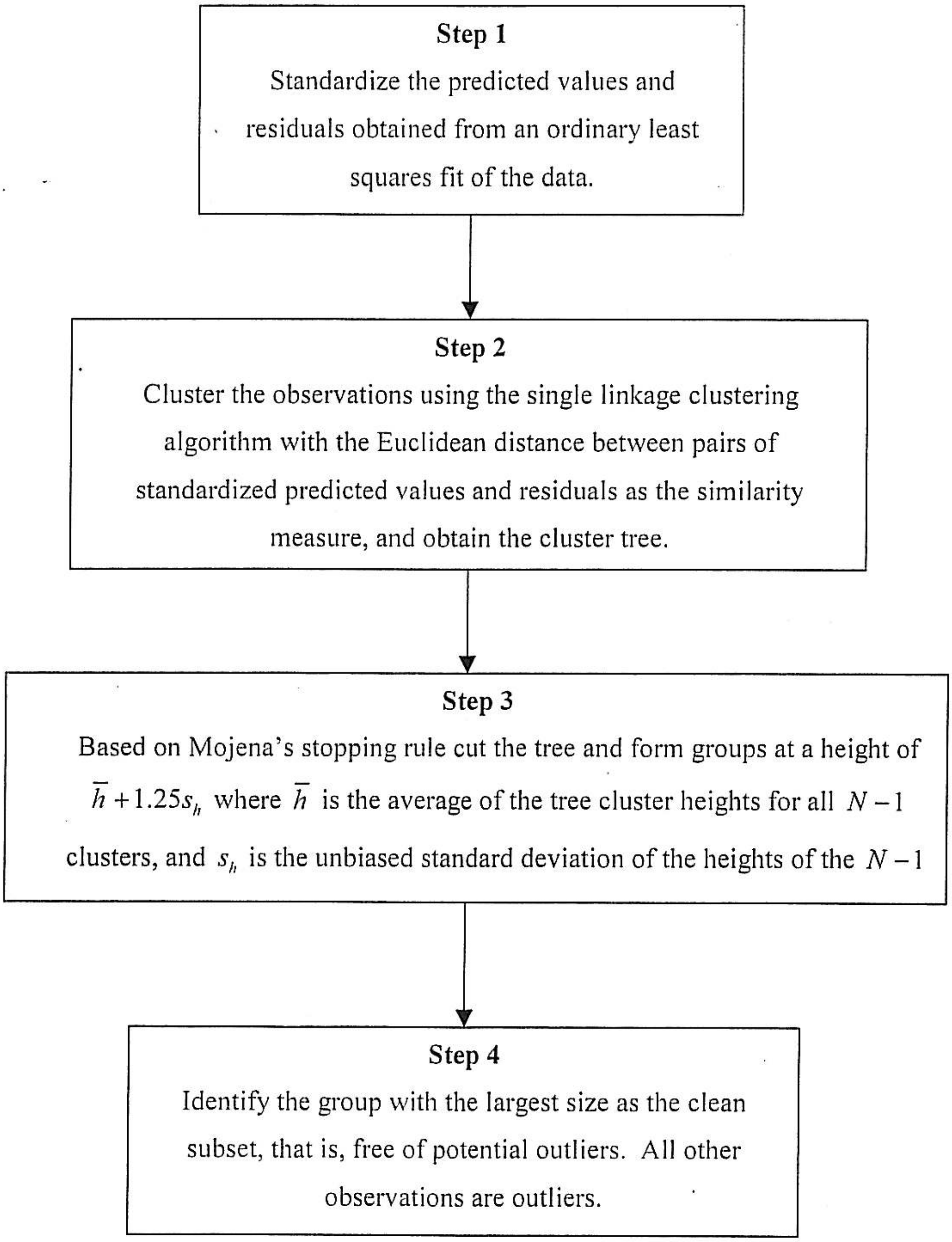


Figure 4.10: Cluster tree and Mojena's cut height for Modified Wood Gravity data.

Again referring to Figure 4.10. It can be seen that after the cut there are three groups formed. Going across the tree from left to right, Group 1 consists of observation 11 and Group 2 consists of observation 19, 7, 6, 4, and 8. Group three consists of observations 3, 5, 16, 18, 1, 12, 14, 20, 15, 9, 10, 13, 2, and 17. Group 3 contains the majority of the observations and thus this set will be the inlying observations. Observations 4, 6, 7, 8, 11, and 19 are identified as the outlying observations. The outlying observations identified by this methodology are noted also in Figure 4.8.



Step 1

Standardize the predicted values and residuals obtained from an ordinary least squares fit of the data.

Step 2

Cluster the observations using the single linkage clustering algorithm with the Euclidean distance between pairs of standardized predicted values and residuals as the similarity measure, and obtain the cluster tree.

Step 3

Based on Mojena's stopping rule cut the tree and form groups at a height of $\bar{h} + 1.25s_h$ where $\bar{h}$ is the average of the tree cluster heights for all $N-1$ clusters, and s_h is the unbiased standard deviation of the heights of the $N-1$

Step 4

Identify the group with the largest size as the clean subset, that is, free of potential outliers. All other observations are outliers.

Figure 4.11: Steps in Sebert et al. clustering algorithm

4.5 Performance of the Clustering Methodology on Classic Data Sets

Sebert et al. have used many data sets to illustrate the multiple outlier problem in linear regression. These data sets will be referred to as "classic" multiple outlier data sets and are shown in Table 4.5. In the table, p represents the number of regressor variables and n is the total number of observations in the data set. The last column lists the outlying observations. The performance of the methodology on the classic data sets is summarized in Table 4.6. It can be seen that the methodology successfully identified all the outliers for all of the data sets. The method performed perfectly for 3 out of the 5 data sets in the sense that there was no masking or swamping. When there was swamping, the number of observations swamped is small. For example, in the Modified Wood Gravity data, observations 7 and 11 are included in the outlying set of observations. Appendix B shows the full computation and results for the other 4 classic data sets using this clustering methodology.

Table 4.5: Classic multiple outlier data sets

No	Data sets	p	n	Outlying observation
1	Telephone Data (Rousseuw and Leroy, 1987)	1	24	15-24
2	Hertzsprung-Russell StarsData (Rousseuw and Leroy, 1987)	1	47	11, 20, 30, 34
3	Hawkins, Bradu, and Kass Data (Hawkins et al., 1984)	3	75	1-14
4	Modified Wood Gravity Data (Rousseuw and Leroy, 1987)	5	20	4, 6, 8, 19
5	Stackloss Data (Brownlee,1965)	3	21	1-4, 21

Table 4.6: Sebert's methodology performance on classic multiple outlier data sets

No	Data sets	Outlying observation	Outlying observations identified	Number of observations swamped (False alarms)
1	Telephone Data (Rousseuw and Leroy, 1987)	15-24	15-24	0
2	Hertzsprung-Russell StarsData (Rousseuw and Leroy, 1987)	11, 20, 30, 34	11, 20, 30, 34, 7, 14	2
3	Hawkins, Bradu, and Kass Data (Hawkins et al., 1984)	1-14	1-14	0
4	Modified Wood Gravity Data (Rousseuw and Leroy, 1987)	4, 6, 8, 19	4, 6, 7, 8, 11, 19	2
5	Stackloss Data (Brownlee,1965)	1-4, 21	1-4, 21	0

4.6 Discussion of Methodology

The following discussion is provided to highlight the important point of clustering methodology.

Standardizing predicted and residual values

Generally, it is advisable to standardize the observations of a data set before a cluster analysis. The reason for this is that the observations with the most variability will dominate the similarity measure. This is especially necessary when using Euclidean distance as similarity measure. In many regression data sets the variation of the predicted values will be much different than the variation in the residuals. Thus to properly differentiate the outlying observations the predicted values and residuals must be standardized.

Predicted values versus residual plots

One may argue that the predicted value versus residual plot is all that is required for identifying multiple outliers in linear regression. To be sure, in some instances the regression analyst will be able to identify multiple outliers by a simple examination of this plot. In fact, the methodology of this research was motivated by being able to "see" multiple outlying observations in the plots of many of the classic multiple outlier data sets. The problem of looking only at the predicted value versus residual value plot is that there will be many instances where multiple outliers are not as extreme and ones ability to see them is greatly reduced.

For example, consider again the plot of the predicted versus residual values for the "Modified Wood Gravity" data set in Figure 4.8. It is our argument that the outlying observations (4, 6, 8, 19) are not easily identifiable by visual inspection only. Lastly it seems like it would be less than satisfactory to propose that the regression user only examine the predicted versus residual plot for multiple outlying observations. Therefore, a formal method of grouping the observations and classifying them as inlying or outlying, as proposed in this research, is beneficial.

Mojena's stopping rule

One may wonder why Mojena's stopping rule for grouping the observations was chosen over numerous other rules that have been proposed. Milligan and Cooper (1985) empirically evaluated the performance of thirty proposed stopping rules and Mojena's rule was among the top performers. Milligan and Cooper pointed out that all stopping rules are "heuristic, ad hoc procedures". They discussed that, although many have recommended the development of more formal statistical methods for determining the number of clusters in a data set, progress in this area has been slow because of the "immense distributional problems" associated with clustering. Therefore, the practioner has only heuristic rules to use as a guide.

The reason that Mojena's rule was chosen for the clustering methodology of this research is because it is simple to calculate and it performs excellently on the data sets in which it was tested. It must be emphasized that Mojena's criteria is very straightforward when compared with other stopping rules. One should not select a stopping rule base on ease of computation only, however based on Milligan and Cooper's study, there is no evidence to suggest that the overall performance of Mojena's rule is worse than other stopping rules, and this research has shown that using this simple rule provides excellent solutions in the regression or multiple outlier context.

Masking and Swamping

It can be seen from the methodology's performance on the classic data sets that there was no masking. However, there were a few data sets that contained swamped observations. Masking is a more serious problem than swamping. If an outlier is missed because of masking, the outlier, if influential, can degrade the performance of the regression model. On the other hand, if swamping occurs, these "clean" or inlying observations should be identified though they are not influential.

4.6 Testing the Sebert et al. Cluster Methodology on Simulated Data Set

Sebert et al. clustering methodology discussed in this research has been shown to perform well on the classic data set. However to further understand the performance of the methods, a detailed study of the procedure on randomly generated data sets was performed. According to Figure 3.1 in chapter three, 6-outlier scenarios have been considered. Each outlier scenario has 36 regression conditions. The results are based upon applying the proposed methodology to 1000 random sets created according to a specific regression condition.

The results showing the performance of the proposed methodology for each scenario is provided in Table 4.7 – Table 4.12. Figures 4.12 - 4.17 shows the Sebert's method performance based on the number of observation n and the number of regressor variable p. Scenario 1 consists of situations 1-4, while scenario 2 consists of situations 5-8 and so on. Table 4.16 summarizes Sebert's method performance for each scenario. The following provides the general conclusions about the procedure's performance characteristics.

The tppo (total probability a planted outlier is detected) value increases as the outlying distance increases. From a clustering perspective, this is an intuitive result. The more separated outlying groups of observations are from inlying observations, the easier it is to obtain a proper clustering solution. However, it must be emphasized that this is not the case when using typical least squares diagnostics and many robust methods. In high leverage scenarios, the more separated a group of outlying observations is the more difficult it is to identify the group as outlying. The tppo (total probability a planted outlier is detected) value increases as the number of regressor variables increases. Figures 4.12- 4.14 show this situation. From Figure 4.12 where the sample size is 20, the detection probability decreases significantly with the increase in the number of regressors particularly in situations 2, 4, 10 and 12 but increases significantly in situations 19 through 24.

Table 4.7: Scenario 1 result for the Sebert et al. methodology

			$n = 20$		$n = 40$		$n = 60$	
No of regressor (*p*)	**outlier %**	**outlier distance**	**tppo**	**tpswamp**	**tppo**	**tpswamp**	**tppo**	**tpswamp**
1	10	5	0.9730	0.0528	0.9930	0.0893	0.9957	0.1120
	10	10	1	0.0069	1	0.0072	1	0.0109
	20	5	0.9128	0.1629	0.9567	0.2666	0.9723	0.2926
	20	10	0.9920	0.0672	0.9892	0.0899	0.9890	0.0889
2	10	5	0.9190	0.0000	0.9825	0.0007	0.9877	0.0026
	10	10	0.5565	0.0012	0.6333	0.0040	0.6920	0.0086
	20	5	0.9263	0.0008	0.9819	0.0028	0.9943	0.0061
	20	10	0.5515	0.0073	0.6670	0.0158	0.7229	0.0254
6	10	5	0.9990	0.0238	1	0.0396	1	0.0488
	10	10	1	0.0168	1	0.0224	1	0.0266
	20	5	0.9893	0.0541	0.9980	0.0714	0.9992	0.0800
	20	10	1	0.0508	0.9993	0.0621	0.9990	0.0631

Table 4.8: Scenario 2 result for the Sebert et al. methodology

			$n = 20$		$n = 40$		$n = 60$	
No of regressor (*p*)	**outlier %**	**outlier distance**	**tppo**	**tpswamp**	**tppo**	**tpswamp**	**tppo**	**tpswamp**
1	10	5	0.9995	0.0283	0.9983	0.0413	1	0.0466
	10	10	1	0.0076	0.9990	0.0086	1	0.0118
	20	5	0.9928	0.0787	0.9923	0.1068	0.9947	0.1083
	20	10	0.994	0.0538	0.9894	0.0678	0.9882	0.0736
2	10	5	0.9995	0.0000	1	0.0000	1	0.0000
	10	10	0.992	0.0000	1	0.0004	1	0.0012
	20	5	1	0.0000	1	0.0000	1	0.0000
	20	10	0.9943	0.0000	1	0.0005	1	0.0014
6	10	5	1	0.0107	1	0.0154	1	0.0193
	10	10	1	0.0101	1	0.0143	1	0.0174
	20	5	1	0.0228	1	0.0317	1	0.0350
	20	10	1	0.0271	1	0.0350	1	0.0371

Table 4.9: Scenario 3 result for the Sebert et al. methodology

			$n = 20$		$n = 40$		$n = 60$	
No of regressor (p)	outlier %	outlier distance	tppo	tpswamp	tppo	tpswamp	tppo	tpswamp
1	10	5	0.9260	0.0113	0.9770	0.0210	0.9850	0.0281
	10	10	1	0.0002	1	0.0003	1	0.0011
	20	5	0.8560	0.0492	0.9591	0.0976	0.9797	0.1255
	20	10	0.9995	0.0124	0.9990	0.0197	0.9980	0.0195
2	10	5	0.9015	0.0001	0.9698	0.0010	0.9807	0.0026
	10	10	0.5365	0.0006	0.6095	0.0027	0.6617	0.0066
	20	5	0.8923	0.0006	0.9653	0.0027	0.9863	0.0064
	20	10	0.5340	0.0068	0.6243	0.0133	0.6765	0.0198
6	10	5	0.9895	0.0100	0.9985	0.0159	1	0.0214
	10	10	1	0.0045	1	0.0080	1	0.0102
	20	5	0.9298	0.0208	0.9951	0.0334	0.9999	0.0404
	20	10	0.9943	0.0193	1	0.0281	1	0.0313

Table 4.10: Scenario 4 result for the Sebert et al. methodology

			$n = 20$		$n = 40$		$n = 60$	
No of regressor (p)	outlier %	outlier distance	tppo	tpswamp	tppo	tpswamp	tppo	tpswamp
1	10	5	0.9885	0.0057	0.9998	0.0108	1	0.0148
	10	10	1	0.0003	1	0.0012	1	0.0019
	20	5	0.9350	0.0276	0.9991	0.0323	0.9995	0.0380
	20	10	0.9990	0.0120	1	0.0126	0.9995	0.0161
2	10	5	0.9995	0.0000	1	0.0000	1	0.0000
	10	10	0.9925	0.0001	1	0.0004	1	0.0012
	20	5	1	0.0000	1	0.0000	1	0.0000
	20	10	0.9908	0.0001	1	0.0011	1	0.0017
6	10	5	1	0.0038	1	0.0059	1	0.0073
	10	10	1	0.0035	1	0.0056	1	0.0068
	20	5	0.9973	0.0088	1	0.0135	1	0.0156
	20	10	0.9990	0.0106	1	0.0162	1	0.0168

Table 4.11: Scenario 5 result for the Sebert et al. methodology

			$n = 20$		$n = 40$		$n = 60$	
No of regressor (p)	outlier %	outlier distance	tppo	tpswamp	tppo	tpswamp	tppo	tpswamp
1	10	5	0.9885	0.0673	1	0.0682	1	0.0729
	10	10	0.9885	0.0673	1	0.0682	1	0.0729
	20	5	0.9178	0.0860	0.9943	0.1012	0.9991	0.1017
	20	10	0.9178	0.0860	0.9943	0.1012	0.9991	0.1017
2	10	5	0.9690	0.0000	1	0.0000	1	0.0000
	10	10	0.9700	0.0000	1	0.0001	1	0.0010
	20	5	0.9930	0.0000	1	0.0000	1	0.0000
	20	10	0.9893	0.0001	1	0.0007	1	0.0015
6	10	5	1	0.0099	1	0.0152	1	0.0193
	10	10	1	0.0097	1	0.0151	1	0.0353
	20	5	1	0.0184	1	0.0296	1	0.0194
	20	10	1	0.0186	1	0.0300	1	0.0352

Table 4.12: Scenario 6 result for the Sebert et al. methodology

			$n = 20$		$n = 40$		$n = 60$	
No of regressor (p)	outlier %	outlier distance	tppo	tpswamp	tppo	tpswamp	tppo	tpswamp
1	10	5	0.8980	0.0061	0.9998	0.0102	1	0.0160
	10	10	0.7310	0.0001	0.9995	0.0008	1	0.0017
	20	5	0.8988	0.0086	1	0.0158	1	0.0259
	20	10	0.8430	0.0001	1	0.0012	1	0.0032
2	10	5	0.9995	0.0000	1	0.0000	1	0.0000
	10	10	1	0.0000	1	0.0000	1	0.0000
	20	5	1	0.0000	1	0.0000	1	0.0000
	20	10	1	0.0000	1	0.0000	1	0.0000
6	10	5	1	0.0008	1	0.0023	1	0.0039
	10	10	1	0.0000	1	0.0002	1	0.0006
	20	5	1	0.0005	1	0.0033	1	0.0068
	20	10	1	0.0000	1	0.0003	1	0.0013

Figures 4.13 and 4.14 indicate that for large n, ($n = 40$ and $n = 60$), the detection probability remains high and quite the same for different values of p. The outlying observations were generated to be outlying in all regression variables. The methodology is based on clustering the elements in the predicted versus residual value plots. As a result, as the number of regressor variables increases, observations that are outlying in all regressor variables will tend to have larger predicted values with similar residual values. This allows the proposed identification method to more easily identify the outlying observations as a cluster.

The tppo (total probability a planted outlier is detected) value increases as the percentage of outliers decreases. Recall, the outlying and inlying observations were created randomly, both having $N(0,1)$ error distribution. For a given outlier distance, as the percentage of outliers increases the probability that an outlier will be close to the inlying observations also increases. Therefore, from a clustering perspective, the possibility of falsely classifying an outlier as an inlier (failure) increases

The tppo (total probability a planted outlier is detected) value increases as the number of observations in the data set increases. Figures 4.15 – 4.17 shows this situation. In all simulations, the regressor values were generated from the $U(0,20)$ distributions. From a clustering point of view it is easier to distinguish groups of data when there are more data in a given range. In other words, clusters are easier to identify as the density of the cluster increases. This is why this method performs better as the number of observations increases.

To verify this phenomenon, consider scenario 6 for the 1 regressor, 20 observation regression condition. Overall this was the regression condition in which this methodology had the worst success rate. However, the performance of the methodology is improved dramatically when the density of the clean cluster is increased. This was done by generating the clean observations from the $U(0,10)$ distribution. The results are shown in Table 4.13.

The tpswamp (total probability a clean observation is classified as an outlier) value decreases as the outlying distance increases. Again, this is an intuitive result. The more separated the inlying and outlying observations are, the easier it is for the clustering algorithm to correctly differentiate them. The tpswamp (total probability a clean observation is classified as an outlier) value decreases as the number of regressor variables increases. Figures 4.12-4.14 shows this situation. Because the predicted values are "magnified" in these situations, they will be more separated on the predicted versus residuals plots. From a clustering perspective, the more separated groups are the easier it is to correctly identify them.

The tpswamp (total probability a clean observation is classified as an outlier) value decreases as the percentage of outliers decreases. This is consistent with the improved of tppo value at lower outlier percentages. In general, it is easier to differentiate the outliers from the inliers when there are fewer outliers.

Table 4.13: Comparison of Sebert et al. method performance with higher density clusters

Scenario 6 with $n = 20$			**Inlying observations** $U(0,10)$		**Inlying observations** $U(0,20)$	
No of regressor	**Outlier %**	**Outlier distance**	**tppo**	**tpswamp**	**tppo**	**tpswamp**
1	10	5	1	0.0008	0.8980	0.0061
	10	10	1	0.0000	0.7310	0.0001
	20	5	1	0.0018	0.8988	0.0086
	20	10	1	0.0000	0.8430	0.0001

The simulation result for Sebert et al. method performs well for most of regression conditions tested. In general, this method performs best (high tppo value with low tpswamp value) at lower outliers percentages. Table 4.14 summarizes the tppo value in percentage for the six scenarios. For example, in scenario 6 this method was successful at least 95% of the time in 32 of 36 regression conditions. Recall that each scenario has 36 regression conditions, so for scenario 6, this method was successful at least 950 out of 1000 times for 32 of the 36 regression conditions. Similarly, Table 4.15 summarizes the tpswamp value in percentage for the six scenarios. For example in scenario 6, the percentage of clean observations is classified as an outlier is between 0 to 5% in all the regression conditions.

Table 4.14: Total probability a planted outlier is detected (in percentage) of the Sebert et al. methodology in all regression conditions tested

%	**100-95**	**94.9-90**	**89.9-85**	**84.9-80**	**79.9-75**	**74.9-70**	**<70**
Scenario 1	27/36	3/36	0/36	0/36	0/36	1/36	5/36
Scenario 2	36/36	0/36	0/36	0/36	0/36	0/36	0/36
Scenario 3	25/36	3/36	0/36	0/36	0/36	0/36	6/36
Scenario 4	35/36	1/36	0/36	0/36	0/36	0/36	0/36
Scenario 5	34/36	2/36	0/36	0/36	0/36	0/36	0/36
Scenario 6	32/36	0/36	2/36	1/36	0/36	1/36	0/36

Table 4.15: Total probability a clean observation is classified as an outlier (in percentage) of the Sebert et al. methodology in all regression conditions tested

%	**0-4.9**	**5-9.9**	**10-14.9**	**15-19.9**	**20-24.9**	**>25**
Scenario 1	23/36	9/36	1/36	1/36	0/36	2/36
Scenario 2	30/36	4/36	2/36	0/36	0/36	0/36
Scenario 3	34/36	1/36	1/36	0/36	0/36	0/36
Scenario 4	36/36	0/36	0/36	0/36	0/36	0/36
Scenario 5	28/36	8/36	0/36	0/36	0/36	0/36
Scenario 6	36/36	0/36	0/36	1/36	0/36	0/36

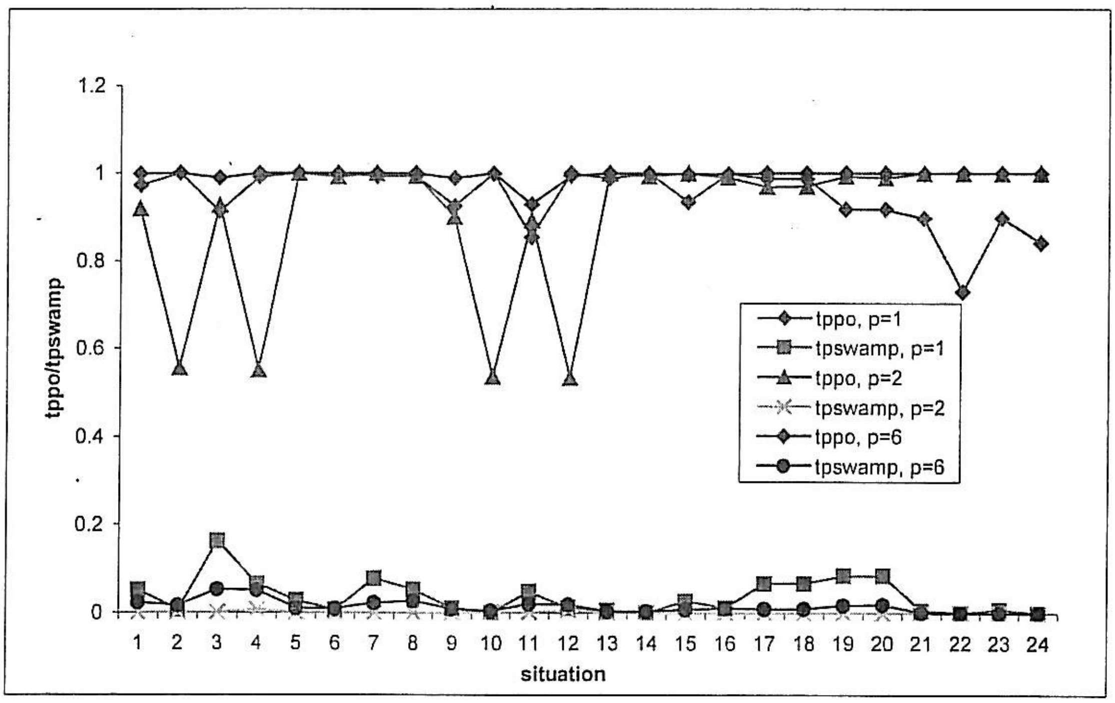


Figure 4.12: Performance of Sebert's method for $n = 20$ and all values of p

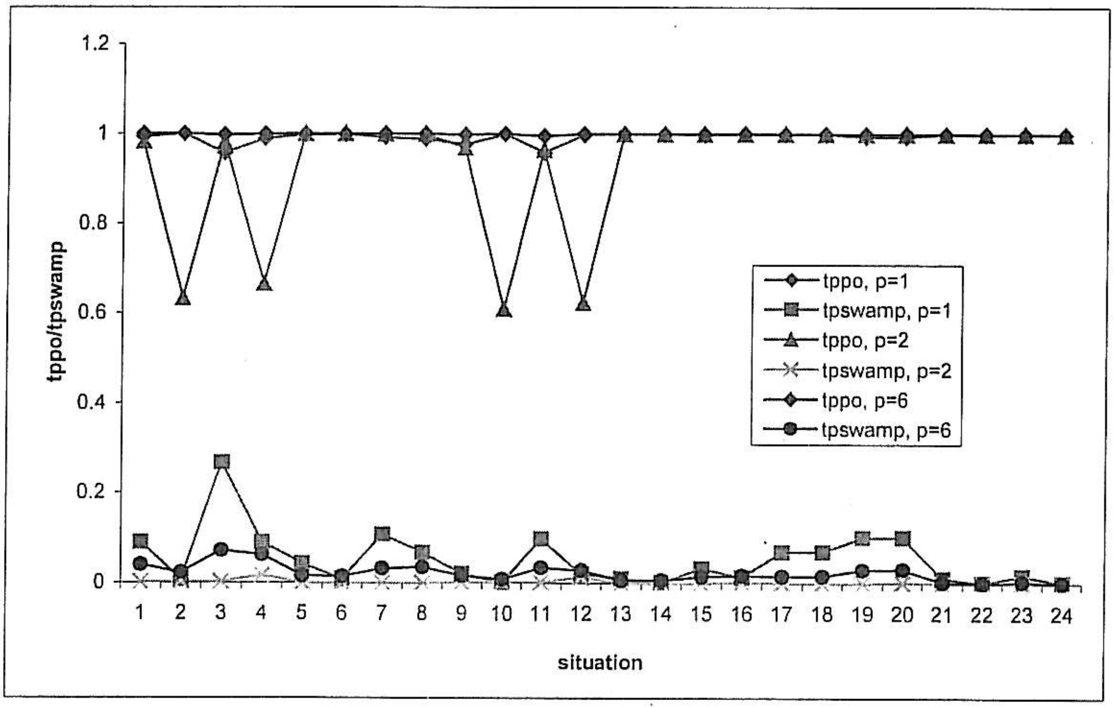


Figure 4.13: Performance of Sebert's method for $n = 40$ and all values of p

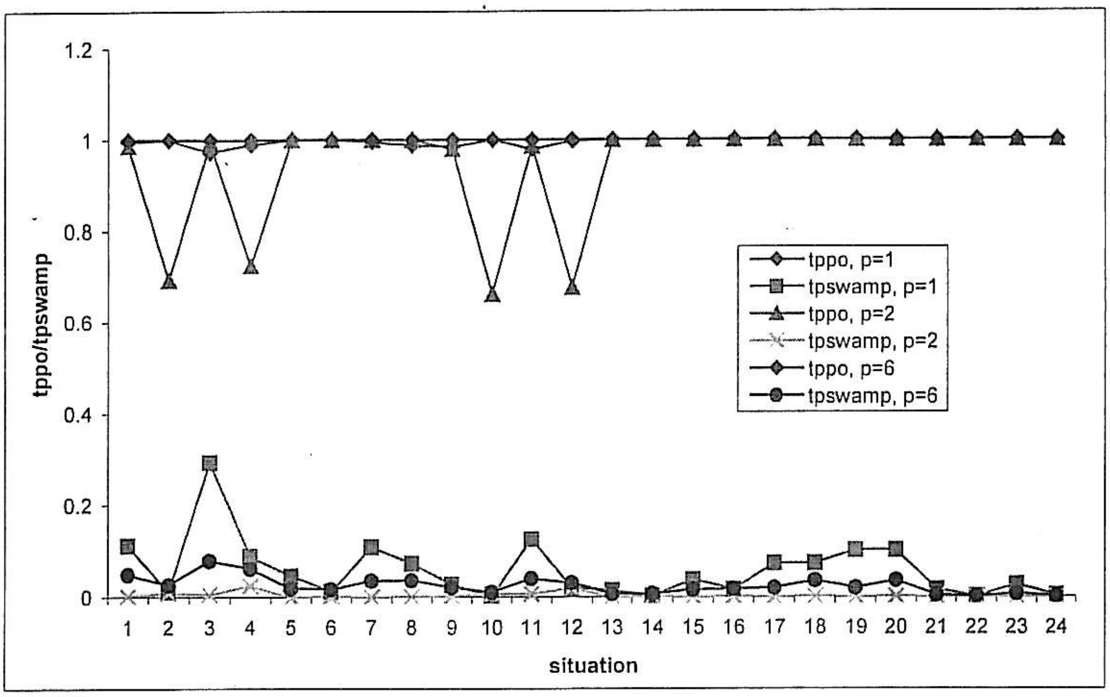


Figure 4.14: Performance of Sebert's method for $n = 60$ and all values of p

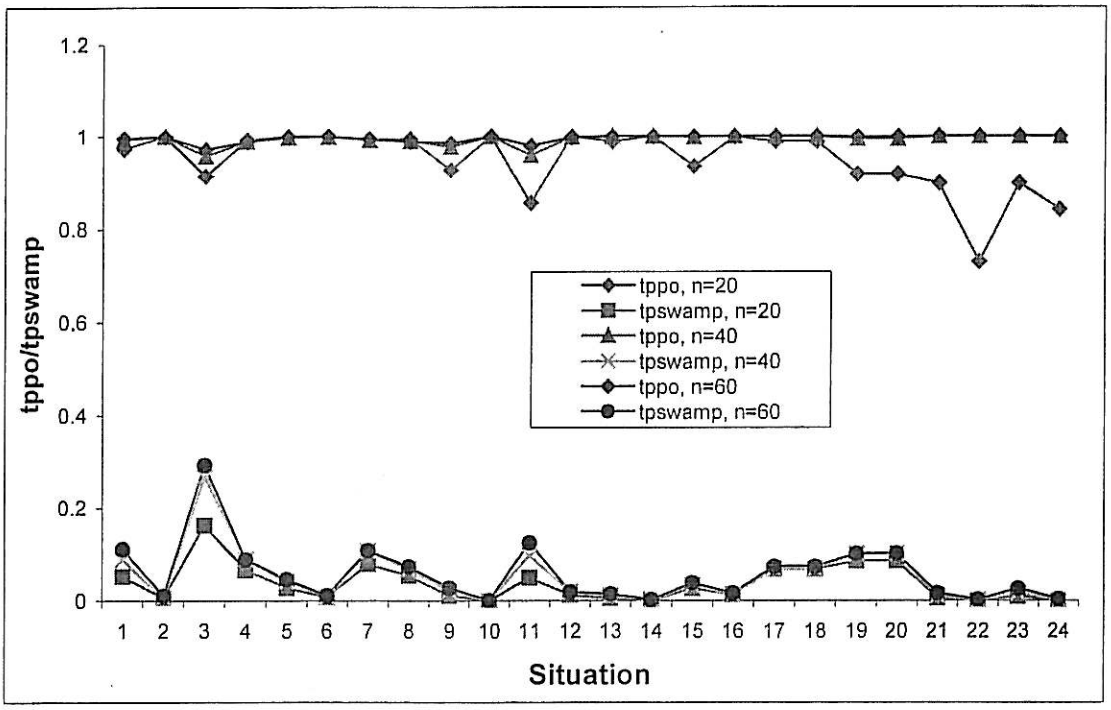


Figure 4.15: Performance of Sebert's method for p =1 and all values of n

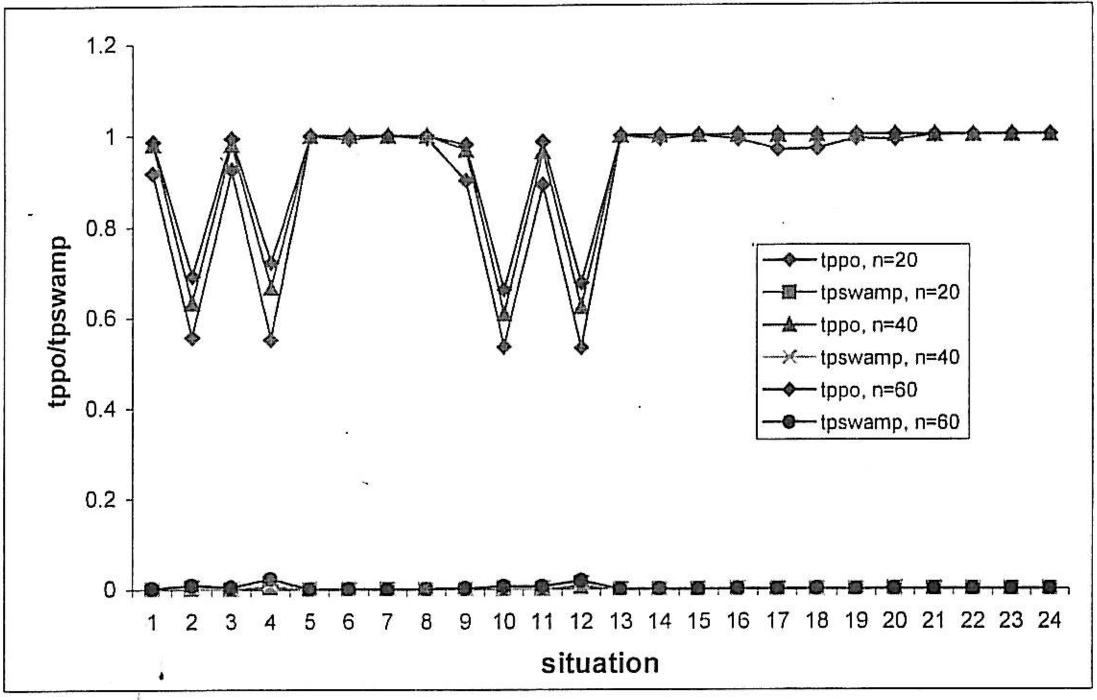


Figure 4.16: Performance of Sebert's method for $p = 2$ and all values of n

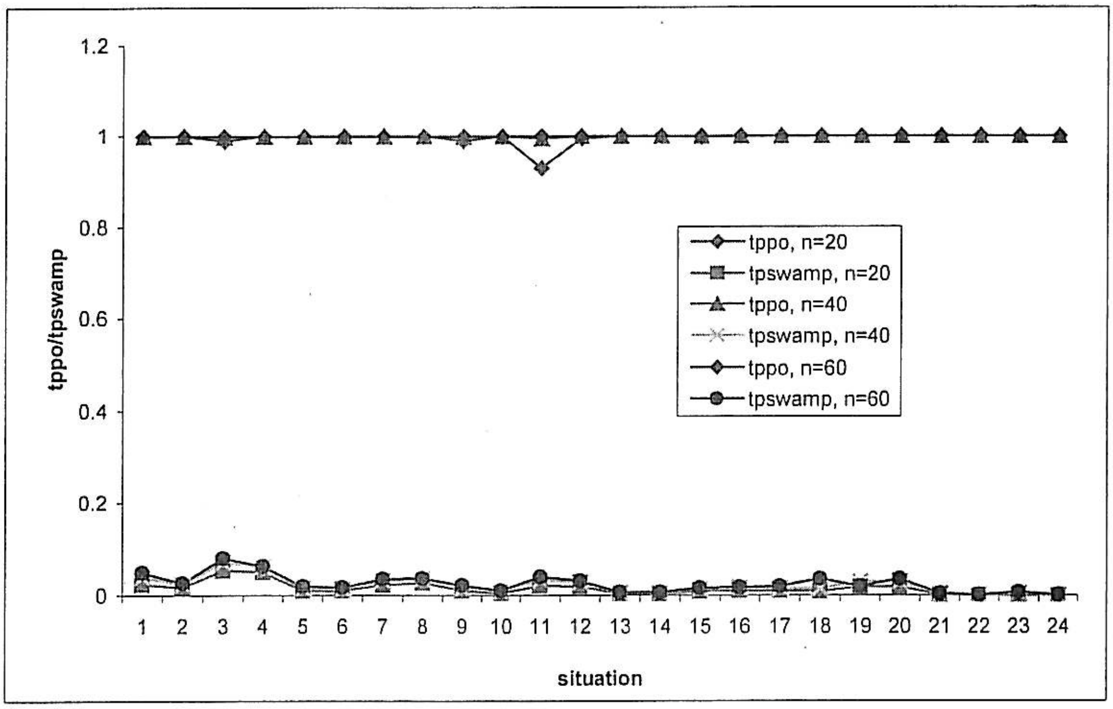


Figure 4.17: Performance of Sebert's method for $p = 6$ and all values of n

Table 4.16: Summary of Sebert'ṡ method performance for each scenario

		No of regressor increase	**No of observation increase**	**Outlier % increase**	**Outlier distance increase**
Scenario 1	tppo	increase	increase	decrease	increase
	tpswamp	decrease	increase	increase	decrease
Scenario 2	tppo	increase	increase	decrease	increase
	tpswamp	decrease	increase	increase	decrease
Scenario 3	tppo	increase	increase	decrease	increase
	tpswamp	decrease	increase	increase	decrease
Scenario 4	tppo	increase	increase	decrease	increase
	tpswamp	decrease	increase	increase	decrease
Scenario 5	tppo	increase	increase	decrease	increase
	tpswamp	decrease	increase	increase	decrease
Scenario 6	tppo	increase	increase	decrease	decrease
	tpswamp	decrease	increase	increase	decrease

4.8 Summary and Conclusion

This chapter discussed the Sebert et al. clustering method in identifying multiple outliers in linear regression and was shown to perform excellently on the classic data sets found in the literature and a wide variety of simulated data sets. This methodology is not the first to suggest a clustering based approach. For example, Gray and Ling (1984) proposed to use a clustering algorithm to identify potentially influential subsets. They used the hat matrix with the vector of response values appended as the basis for their clustering algorithm. Hadi and Simonoff (1993) also proposed a clustering strategy based on the use of single linkage clustering on $\mathbf{Z} = (\mathbf{X}|\mathbf{y})$.

There are several advantages in using this approach. First, recall that it is the residual and associated predicted values that are clustered in this methodology. This always reduces the multiple outlier problem to two dimensions, where as clustering $\mathbf{Z} = (\mathbf{X}|\mathbf{y})$ maintains the dimensions $k+1$. Having fewer dimensions will make a clustering algorithm more efficient and, as was discussed earlier, in extreme outlying cases, multiple outliers can be easily identified in a plot of the residuals versus predicted values. Next, instead of relying on hypothesis testing for determining multiple outliers, this approach uses the cluster tree itself, with an appropriate cutting rule, to separate inliers from outliers. This methodology relies on a more data analytic approach rather than having to determine appropriate hypothesis testing distributions or formal cut off values. This method is also beneficial because predicted and residual values are standard regression outputs for most common statistical software packages. Furthermore, single linkage clustering is also found on many commonly used statistical software packages such as SAS, S-PLUS and Minitab.

In general, this procedure performs best (no masking and swamping) in those situations in which the outlying observations are located at a large distance from the inliers. One could argue that this is a somewhat intuitive result, and from a clustering perspective it is. However, it must be stressed that many of the current outlier identification strategies, such as least squares type diagnostics or robust methods fail on similar types of data sets. The common characteristic of these data sets that makes them problematic for typical least squares and robust identification strategies is the presence of high leverage groups of outliers. That is, groups of observations that are located a relatively large distance away from the inliers in the regressor space. However, this separation is not problematic for clustering algorithm, but is exactly what the algorithm attempts to show.

The disadvantage of this clustering methodology is that it requires the analyst to use a diagnostic approach that is not in typical regression analyst's toolbox. However, clustering algorithm in general (including single linkage) is now found in commonly used data analysis software packages. Therefore, Sebert et al. hoped this clustering methodology is effective for identifying multiple outliers and will become a routine part of the regression model building process.

CHAPTER 5

MODIFICATION ON SEBERT'S METHOD

5.1 Introduction

This chapter discusses the modification of Sebert et al. (1998) clustering algorithm for identifying multiple outliers in linear regression. The modification is done using the robust estimator and two modifications will be discussed in this chapter. Method 1 is a modification of Sebert's method where the least squares (LS) fit is replaced by the least median of squares (LMS) fit while Method 2 is a modification of Sebert's method where the least squares (LS) fit is replaced by the least trimmed of squares (LTS) fit. The LTS and LMS estimators are chosen since they have a high breakdown (as much as 50%), efficient and bounded influence. The classical data sets will be used to illustrate the methods. Also, the performance characteristics of the proposed methodology are demonstrated and explored by applying the procedure to simulated data sets that have various outlier scenarios.

5.2 The Difference between LS, LMS and LTS Estimator.

The least median of squares (LMS) and the least trimmed of squares (LTS) estimators are the most popular and good robust estimators. These estimators have a high breakdown (as much as 50%), efficient and bounded influence. A high breakdown estimator can fit a model to the bulk of the data even if a large percentage of outliers are present. The ordinary least squares (LS) estimator has a breakdown of 0% because a single outlying observation can make the estimates and inference from the remaining (n-1) observation meaningless. An efficient estimator provides parameter estimates close to those of the ordinary least squares estimates of a data set with NID error terms in the absence of outliers. Bounded influence estimators protect the regression surface from being pulled toward extreme observation in x-space. The ordinary least squares estimators do not have bounded influence and the more extreme the outlier is in x-space, the greater is the impact it has on the estimates of the parameter.

Figure 5.1 shows the least squares fit (LS fit), the least median of squares fit (LMS fit) and the least trimmed of squares fit (LTS fit) for a data set with outliers. From this graph it is obvious that the fit from the LTS and LMS is better compared to the LS in the presence of outliers.

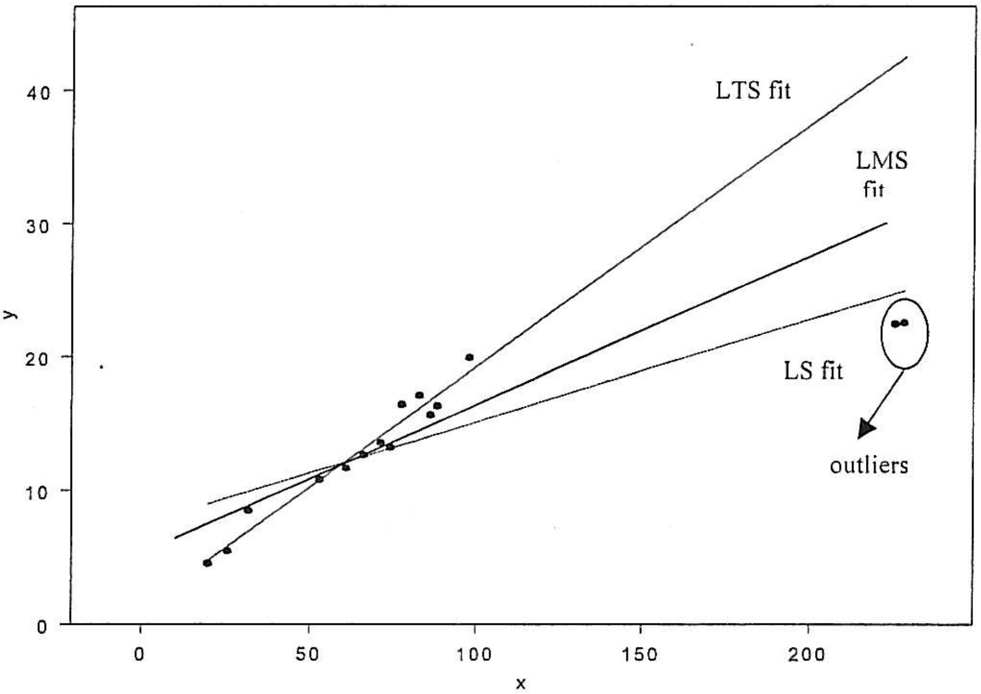


Figure 5.1: The LS, LMS and LTS regression for a data set with two xy-space outlying observations

5.3 A Review on the Least Median of Squares (LMS) Regression

The least median of squares (LMS) regression was proposed by Rousseuw in 1984. LMS estimator is obtained from minimizing the median of squared errors, that is, it solves

$$\underset{\hat{\beta}}{Minimize}\left[med\left(e_i^{\,2}\right)\right] \tag{5.1}$$

In other words, LMS is obtained by minimizing the hth ordered squared residual where h is defined as the integer portions of $[(n/2)+(p+1)/2]$. Note h is not the median of n. LMS fits just over half the data and minimizes the residual for a single observation. The LMS estimator is regression equivariant, scale equivariant, and affine equivariant. A note and proof on these three characteristics are shown in Appendix I. The LMS has a high breakdown (as much as 50%) but due to its $n^{-1/3}$ convergence rate, it has zero efficiency under the central Gaussian model. If $p > 1$ and the observation are in general position, then the breakdown point of the LMS method is $([n/2]-p+2)/n$.

5.4 A Review on the Least Trimmed of Squares (LTS) Regression

The least trimmed of squares (LTS) regression was proposed by Rousseuw in 1984. This method is similar to the least squares but instead of using all the n ordered squared residuals, it minimizes the sum of the h smallest squared residuals, given by

$$\underset{\hat{\beta}}{Minimize}\sum_{i=1}^{h}\left(e^2\right)_{i;n} \tag{5.1}$$

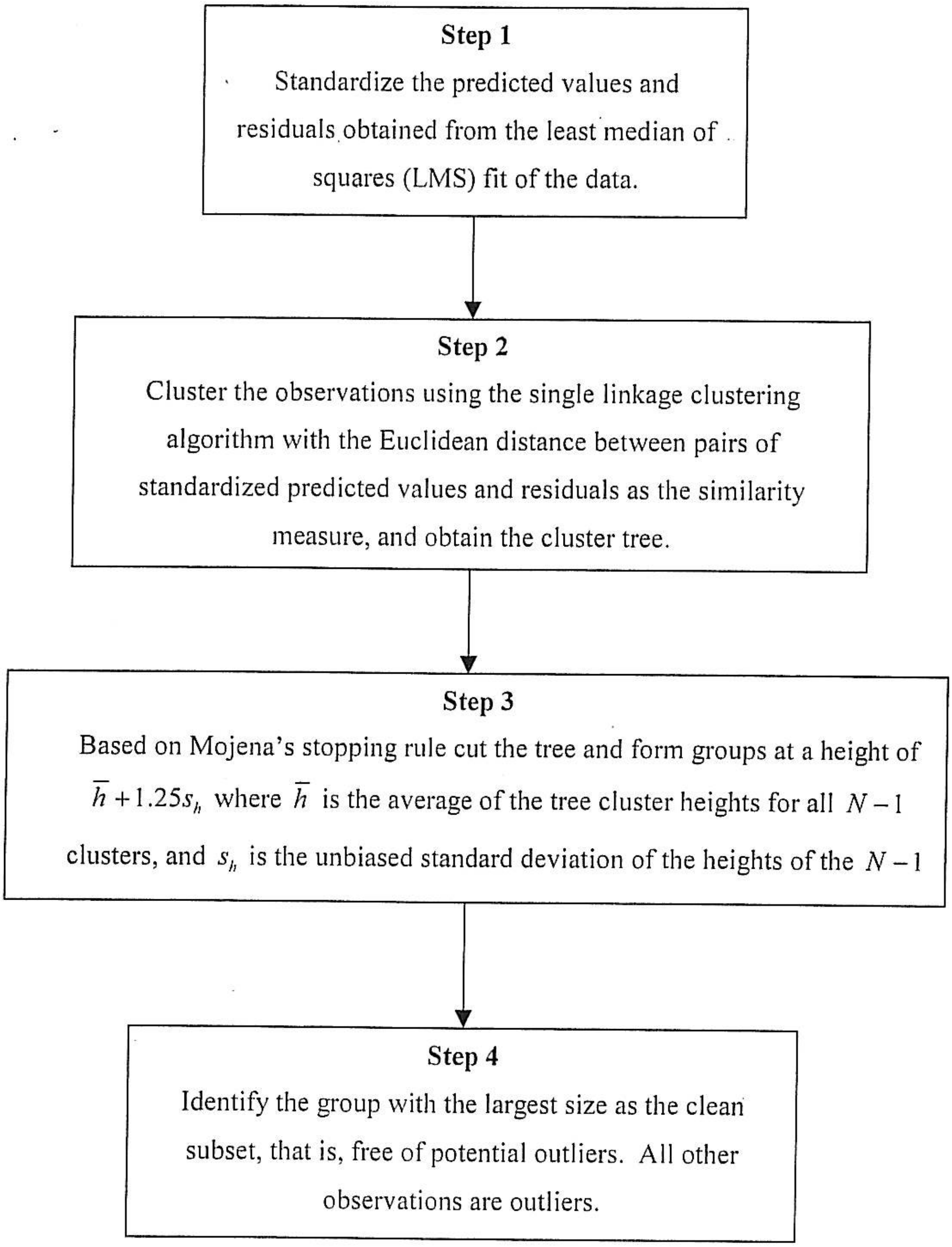
where $\left(e^2\right)_{1;n} \leq \ldots \leq \left(e^2\right)_{n;n}$ are the ordered squared residual. Rousseeuw and Leroy (1987) recommended $h = n(1-\alpha)+1$ where α is the trimmed percentage and n equal to the number of observations.

This estimator is attractive because α can be selected to prevent some of the poor results (efficiency) that other 50% breakdown estimators show. LTS attains the same breakdown point as LMS. The breakdown value for LTS will reach its maximal value, $([(n-p)/2]+1)/n$ when $h=(n+p+1)/2$ where p is the number of parameters. When h becomes close to n, the breakdown point approaches zero. The number of observations must be at least twice the number of parameters for the breakdown point to be high. LTS estimator has 7.12% asymptotic efficiency. LTS can be fairly efficient if the number of trimmed observations is close to the number of outliers because ordinary least squares is used to estimate the parameter from the remaining h observation.

5.5 Method 1

Method 1 is a modification of Sebert's method where the least squares (LS) fit is replaced by the least median of squares (LMS) fit. It is a well-known fact that the least median of squares (LMS) is a high breakdown estimator, therefore the proposed method uses the standardized predicted and residual values from the least median of squares (LMS) fit rather than the ordinary least squares (LS) fit. The flowchart for this method is presented in Figure 5.2.

The steps of the methodology will now be discussed in detail and illustrated with the "Modified Wood Gravity" data set given by Rousseuw and Leroy (1987), which is shown in Table 4.3. Table 5.1 shows the standardized predicted and residual values from the least median of squares (LMS) for the Modified Wood Gravity data. Figure 5.3 shows the output from S-PLUS agglomerative hierarchical clustering and explanation on some of the arguments. Scatter plot of the standardized predicted values and residuals for the wood data using the least median of squares (LMS) fit is shown in Figure 5.4. The cluster tree and corresponding cut height is shown in Figure 5.5.



Step 1

Standardize the predicted values and residuals obtained from the least median of squares (LMS) fit of the data.

Step 2

Cluster the observations using the single linkage clustering algorithm with the Euclidean distance between pairs of standardized predicted values and residuals as the similarity measure, and obtain the cluster tree.

Step 3

Based on Mojena's stopping rule cut the tree and form groups at a height of $\bar{h} + 1.25s_h$ where $\bar{h}$ is the average of the tree cluster heights for all $N-1$ clusters, and s_h is the unbiased standard deviation of the heights of the $N-1$

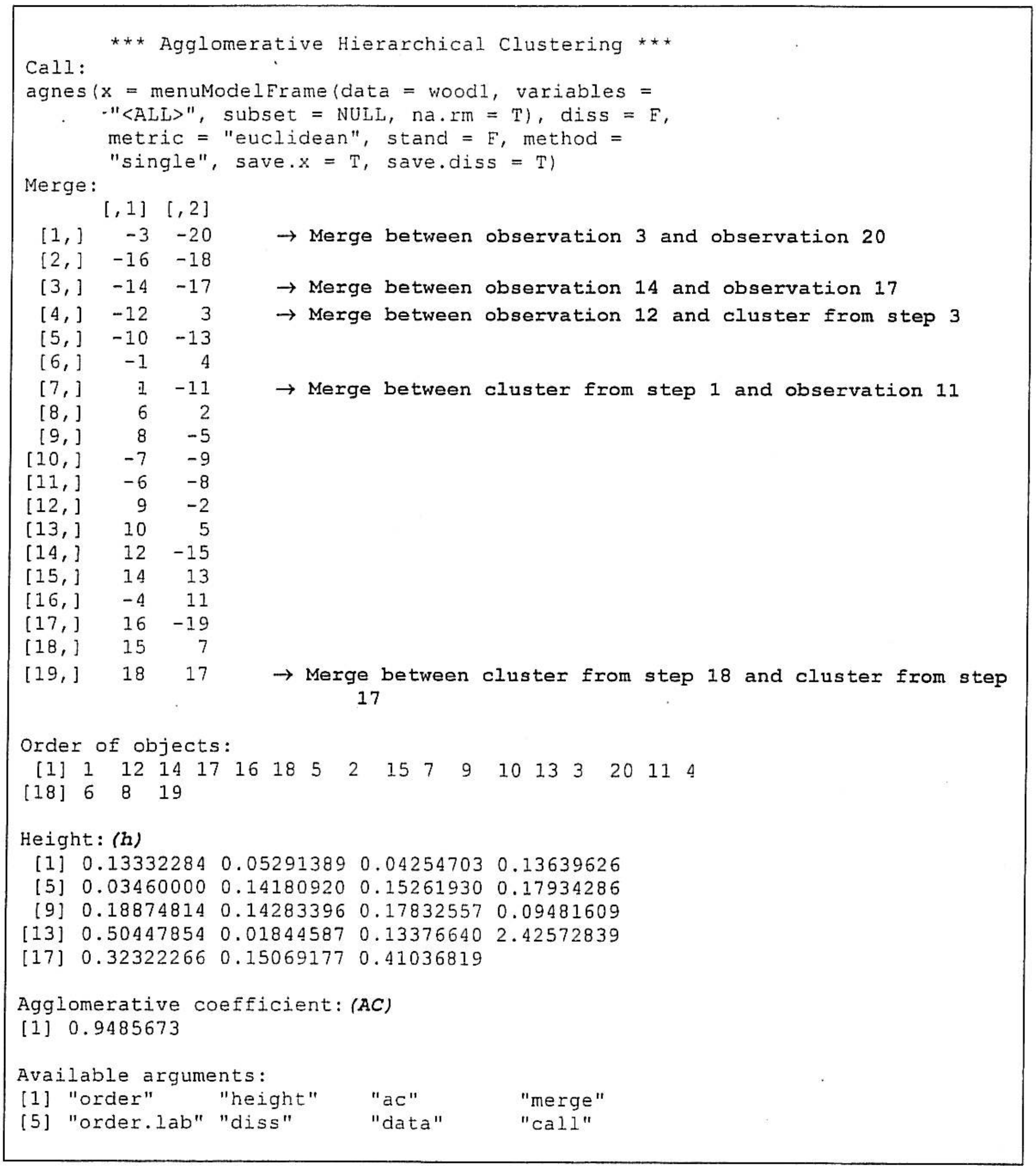
Step 4

Identify the group with the largest size as the clean subset, that is, free of potential outliers. All other observations are outliers.

Figure 5.2: Steps in Method 1

Table 5.1: Standardized least median of squares predicted values and residuals for Modified Wood Gravity data.

Obs.	Predicted Values	Standardized Predicted Values	Residual	Standardized Residual
1	0.5227	-0.3674	0.0113	0.5939
2	0.5304	-0.2337	0.0046	0.5203
3	0.5651	0.3671	0.0049	0.5235
4	0.6379	1.6267	-0.1879	-1.5879
5	0.5295	-0.2495	0.0185	0.6727
6	0.6488	1.8167	-0.2178	-1.9163
7	0.4752	-1.1891	0.0058	0.5334
8	0.6401	1.6662	-0.2171	-1.9087
9	0.4799	-1.1076	-0.0049	0.4161
10	0.4909	-0.9172	-0.0049	0.4161
11	0.5589	0.2599	-0.0049	0.4161
12	0.5239	-0.3459	-0.0049	0.4161
13	0.4889	-0.9525	0.0031	0.5041
14	0.5209	-0.3977	-0.0039	0.4269
15	0.5012	-0.7391	0.0008	0.4786
16	0.5129	-0.5364	-0.0049	0.4161
17	0.5202	-0.4101	-0.0002	0.4676
18	0.5109	-0.5710	-0.0049	0.4161
19	0.6550	1.9236	-0.2540	-2.3125
20	0.5645	0.3571	0.0035	0.5080



```
       *** Agglomerative Hierarchical Clustering ***
Call:
agnes(x = menuModelFrame(data = woodl, variables =
       "<ALL>", subset = NULL, na.rm = T), diss = F,
       metric = "euclidean", stand = F, method =
       "single", save.x = T, save.diss = T)
Merge:
      [,1] [,2]
 [1,]   -3  -20      → Merge between observation 3 and observation 20
 [2,]  -16  -18
 [3,]  -14  -17      → Merge between observation 14 and observation 17
 [4,]  -12    3      → Merge between observation 12 and cluster from step 3
 [5,]  -10  -13
 [6,]   -1    4
 [7,]    1  -11      → Merge between cluster from step 1 and observation 11
 [8,]    6    2
 [9,]    8   -5
[10,]   -7   -9
[11,]   -6   -8
[12,]    9   -2
[13,]   10    5
[14,]   12  -15
[15,]   14   13
[16,]   -4   11
[17,]   16  -19
[18,]   15    7
[19,]   18   17      → Merge between cluster from step 18 and cluster from step
                          17

Order of objects:
 [1] 1  12 14 17 16 18 5  2  15 7  9  10 13 3  20 11 4
[18] 6  8  19

Height: (h)
 [1] 0.13332284 0.05291389 0.04254703 0.13639626
 [5] 0.03460000 0.14180920 0.15261930 0.17934286
 [9] 0.18874814 0.14283396 0.17832557 0.09481609
[13] 0.50447854 0.01844587 0.13376640 2.42572839
[17] 0.32322266 0.15069177 0.41036819

Agglomerative coefficient: (AC)
[1] 0.9485673

Available arguments:
[1] "order"     "height"    "ac"        "merge"
[5] "order.lab" "diss"      "data"      "call"
```

Figure 5.3: The output from S-PLUS agglomerative hierarchical clustering for Modified Wood gravity data using the least median of squares (LMS) fit.

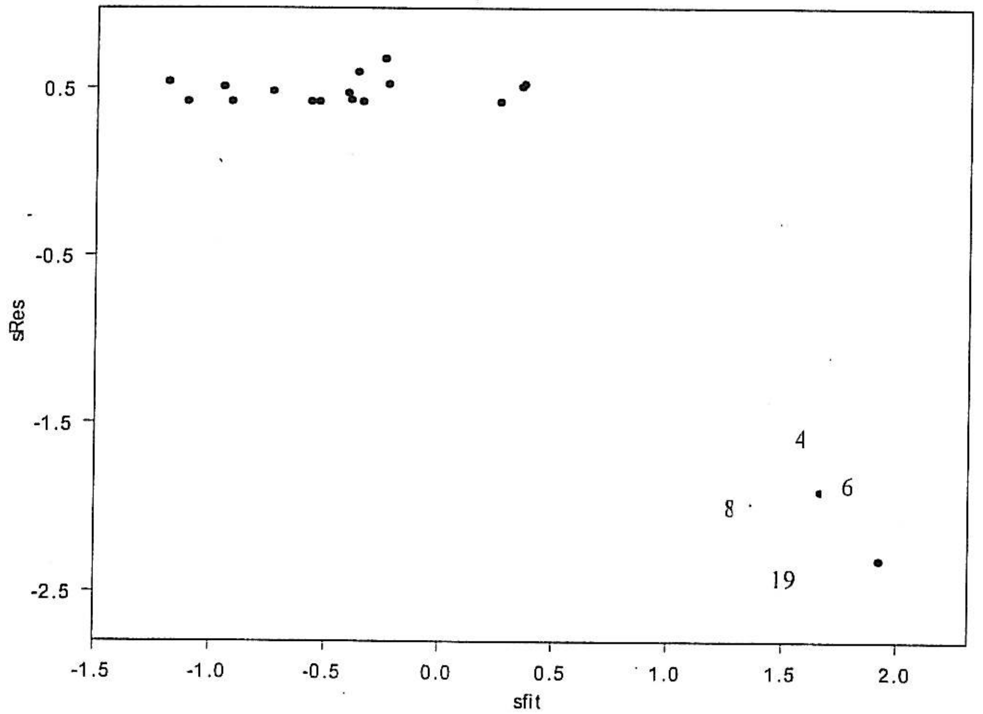


Figure 5.4: Plot of the standardized predicted (sfit) and residuals (sRes) values for the Modified Wood Gravity data using the least median of squares (LMS) fit.

Based on Mojena's stopping rule, the tree will be cut and formed groups at a height of $\overline{h}+1.25s_h$. For this example data set, $\overline{h}=0.286578$ and $s_h=0.532637$. Therefore the cut height on the cluster tree is $0.286578+1.25*0.532637=0.952374$. Referring to Figure 5.5, it can be seen that after the cut, there are two groups formed. Going across the tree from right to left, Group 1 consists of observations 4, 6, 8, and 19. Group 2 consists of observations 11, 20, 3, 13, 10, 9, 7, 15, 2, 5, 18, 16, 17, 14, 12, and 1. Group 2 contains the majority of the observations and thus this set will be the inlying observations. Observations 4, 6, 8, and 19 are identified as the outlying observations. The outlying observations identified by this methodology are also noted in Figure 5.4.

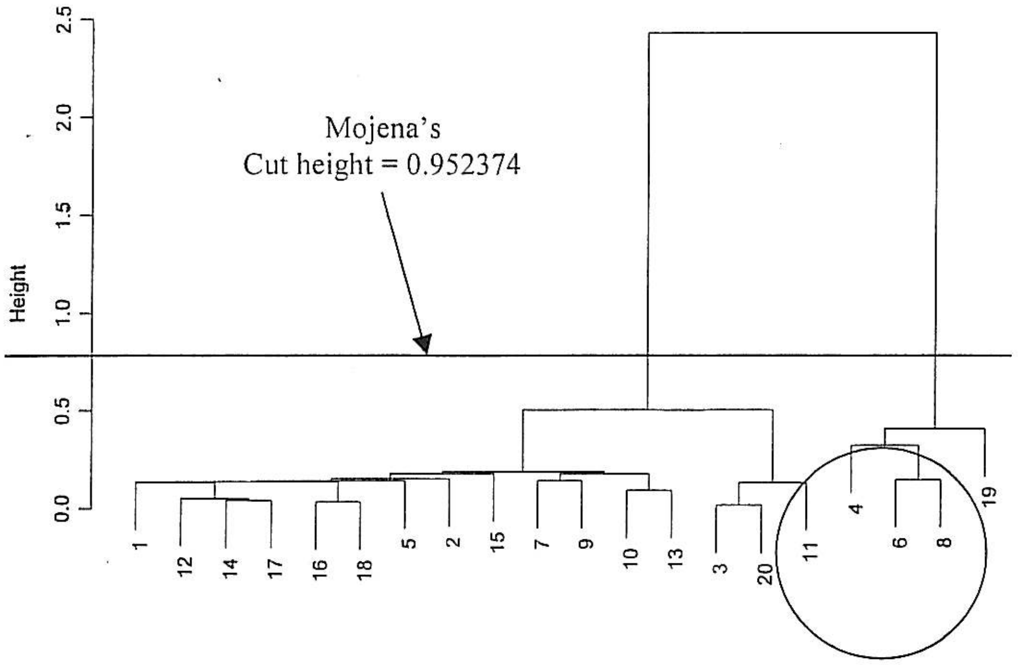


Figure 5.5: Cluster tree and Mojena's cut height for the Modified Wood Gravity data using the least median of squares (LMS) fit.

The performance of the methodology on the classic data sets is summarized in Table 5.2. It can be seen that the methodology successfully identified all the outliers for all of the data sets. The method performed perfectly for 3 out of the 5 data sets in the sense that there was no masking or swamping. When there was swamping or masking, the number of observations swamped or masked is small. For example, in the Hertzsprung-Russell StarsData, observations 7 and 14 are included in the outlying set of observations. Appendix D shows the full computation and results for the other 4 classic data sets using this clustering methodology.

Table 5.2: Method 1's performance on classic multiple outlier data sets

No	Data sets	Outlying observation	Outlying observations identified	Number of observations swamped	Number of observations masked
1	Telephone Data (Rousseuw and Leroy, 1987)	15-24	15-24	0	0
2	Hertzsprung-Russell StarsData (Rousseuw and Leroy, 1987)	11, 20, 30, 34	11, 20, 30, 34, 7, 14	2	0
3	Hawkins, Bradu, and Kass Data (Hawkins et al., 1984)	1-14	1- 10, 13, 14	0	2
4	Modified Wood Gravity Data (Rousseuw and Leroy, 1987)	4, 6, 8, 19	4, 6, 8, 19	0	0
5	Stackloss Data (Brownlee,1965)	1-4, 21	1-4, 21	0	0

"Method 1" clustering methodology discussed in this research has been shown to perform well on the classic data set. However to further understand the performance of the methods, a detailed study of the procedures on randomly generated data sets was performed. The results showing the performance of the Method 1 for each scenario is provided in Table 5.3 – Table 5.8 and Figures 5.6 – 5.11. Again, scenario 1 consists of situations 1-4, while scenario 2 consists of situations 5-8 and so on. Appendix E shows the simulation code for the Method 1.

Table 5.3: Scenario 1 result for the Method 1

			$n = 20$		$n = 40$		$n = 60$	
No of regressor (p)	**outlier %**	**outlier distance**	**tppo**	**tpswamp**	**tppo**	**tpswamp**	**tppo**	**tpswamp**
1	10	5	0.9815	0.0364	0.9943	0.0722	0.9958	0.0903
	10	10	1	0.0021	1	0.0034	1	0.0066
	20	5	0.9505	0.1184	0.9799	0.1911	0.9803	0.2097
	20	10	0.9990	0.0176	0.9980	0.0278	0.9940	0.0352
2	10	5	0.8220	0.0002	0.955	0.0009	0.9753	0.0024
	10	10	0.6350	0.0007	0.7283	0.0048	0.7787	0.0092
	20	5	0.8100	0.0012	0.9455	0.0028	0.9744	0.0057
	20	10	0.6490	0.0056	0.7599	0.0145	0.8128	0.0261
6	10	5	0.9900	0.0678	1	0.0377	1	0.0423
	10	10	1	0.0089	1	0.0156	1	0.0155
	20	5	0.9798	0.0823	1	0.0889	0.9990	0.0953
	20	10	0.9990	0.0660	1	0.0798	1	0.0751

Table 5.4: Scenario 2 result for the Method 1

			$n = 20$		$n = 40$		$n = 60$	
No of regressor (p)	**outlier %**	**outlier distance**	**tppo**	**tpswamp**	**tppo**	**tpswamp**	**tppo**	**tpswamp**
1	10	5	0.9995	0.0141	1	0.0192	1	0.0233
	10	10	1	0.0007	1	0.0002	1	0.0013
	20	5	0.9948	0.0555	0.9971	0.0577	0.9983	0.0526
	20	10	0.998	0.0103	1	0.0019	1	0.0023
2	10	5	0.9405	0	1	0	1	0
	10	10	0.9120	0	1	0.0001	1	0.0003
	20	5	0.9750	0	1	0	1	0
	20	10	0.9610	0	1	0.0002	1	0.0005
6	10	5	1	0.0251	1	0.0274	1	0.0273
	10	10	1	0.0169	1	0.0135	1	0.0123
	20	5	1	0.0473	1	0.0502	1	0.0442
	20	10	1	0.0507	1	0.0498	1	0.0471

Table 5.5: Scenario 3 result for the Method 1

			$n = 20$		$n = 40$		$n = 60$	
No of regressor (*p*)	outlier %	outlier distance	tppo	tpswamp	tppo	tpswamp	tppo	tpswamp
1	10	5	0.9395	0.0083	0.9825	0.0139	0.9898	0.0221
	10	10	1	0	1	0	1	0.0004
	20	5	0.8958	0.0220	0.9769	0.0474	0.9903	0.0700
	20	10	1	0.0017	1	0.0005	0.9995	0.0036
2	10	5	0.7935	0.0002	0.9325	0.0011	0.9580	0.0024
	10	10	0.5695	0.0007	0.6428	0.0019	0.6898	0.0038
	20	5	0.7503	0.0011	0.9051	0.0025	0.9534	0.0044
	20	10	0.5463	0.0014	0.6426	0.0043	0.6804	0.0068
6	10	5	0.9875	0.0177	0.9990	0.0165	1	0.0199
	10	10	0.9990	0.0037	1	0.0035	1	0.0056
	20	5	0.8915	0.0434	0.9905	0.0415	0.9999	0.0552
	20	10	0.9530	0.0553	1	0.0396	1	0.0444

Table 5.6: Scenario 4 result for the Method 1

			$n = 20$		$n = 40$		$n = 60$	
No of regressor (*p*)	outlier %	outlier distance	tppo	tpswamp	tppo	tpswamp	tppo	tpswamp
1	10	5	0.994	0.0032	0.9998	0.0046	1	0.0057
	10	10	1	0.0002	1	0	1	0.0001
	20	5	0.9613	0.0206	0.9996	0.0166	1	0.0167
	20	10	1	0.0023	1	0.0003	1	0.0001
2	10	5	0.8515	0	1	0	1	0
	10	10	0.8925	0	1	0.0001	1	0.0003
	20	5	0.8460	0	1	0	1	0
	20	10	0.8888	0.0002	1	0.0002	1	0.0005
6	10	5	1	0.0111	1	0.0113	1	0.0101
	10	10	1	0.0079	1	0.0057	1	0.0055
	20	5	1	0.0277	1	0.0222	1	0.0223
	20	10	0.9989	0.0362	1	0.0258	1	0.0253

Table 5.7: Scenario 5 result for the Method 1

			$n = 20$		$n = 40$		$n = 60$	
No of regressor (p)	**outlier %**	**outlier distance**	**tppo**	**tpswamp**	**tppo**	**tpswamp**	**tppo**	**tpswamp**
1	10	5	0.9880	0.0688	1	0.0707	1	0.0717
	10	10	0.9880	0.0688	1	0.0707	1	0.0717
	20	5	0.9310	0.0868	0.9925	0.1021	0.9991	0.1036
	20	10	0.9310	0.0868	0.9925	0.1021	0.9991	0.1036
2	10	5	0.807	0	1	0	1	0
	10	10	0.8845	0.0001	1	0	1	0
	20	5	0.9375	0	1	0	1	0
	20	10	0.958	0	1	0	1	0
6	10	5	1	0.0198	1	0.0204	1	0.0252
	10	10	1	0.0236	1	0.0222	1	0.0361
	20	5	1	0.0399	1	0.0432	1	0.0333
	20	10	1	0.0374	1	0.0501	1	0.0421

Table 5.8: Scenario 6 result for the Method 1

			$n = 20$		$n = 40$		$n = 60$	
No of regressor (p)	**outlier %**	**outlier distance**	**tppo**	**tpswamp**	**tppo**	**tpswamp**	**tppo**	**tpswamp**
1	10	5	0.8510	0.0048	1	0.0090	1	0.0132
	10	10	0.6295	0	0.996	0.0002	1	0.0008
	20	5	0.8918	0.0058	0.9995	0.0152	1	0.0234
	20	10	0.7525	0.0005	1	0.0004	1	0.0014
2	10	5	0.9060	0	1	0	1	0
	10	10	0.9995	0	1	0	1	0
	20	5	0.962	0	1	0	1	0
	20	10	0.999	0	1	0	1	0
6	10	5	1	0.0029	1	0.0039	1	0.0045
	10	10	1	0	1	0.0003	1	0.0005
	20	5	1	0.0017	1	0.0035	1	0.0068
	20	10	1	0	1	0.0001	1	0.0008

From the simulation result of Method 1 it shows that the method performs well for most of regression condition tested except in scenario 1 and 3 for $p = 2$. Table 5.9 summarizes the tppo value in percentage for the six scenarios.

Table 5.9: Total probability a planted outlier is detected (in percentage) of the Method 1 in all regression conditions tested

%	100-95	94.9-90	89.9-85	84.9-80	79.9-75	74.9-70	<70
Scenario 1	27/36	1/36	0/36	3/36	2/36	1/36	2/36
Scenario 2	34/36	2/36	0/36	0/36	0/36	0/36	0/36
Scenario 3	23/36	3/36	2/36	0/36	0/36	0/36	6/36
Scenario 4	32/36	0/36	3/36	1/36	0/36	0/36	0/36
Scenario 5	31/36	3/36	1/36	1/36	0/36	0/36	0/36
Scenario 6	31/36	1/36	2/36	0/36	1/36	0/36	1/36

Table 5.10: Total probability a clean observation is classified as an outlier (in percentage) of the Method 1 in all regression conditions tested

%	0-4.9	5-9.9	10-14.9	15-19.9	20-24.9	>25
Scenario 1	24/36	9/36	1/36	1/36	1/36	0/36
Scenario 2	31/36	5/36	0/36	0/36	0/36	0/36
Scenario 3	33/36	3/36	0/36	0/36	0/36	0/36
Scenario 4	36/36	0/36	0/36	0/36	0/36	0/36
Scenario 5	23/36	9/36	4/36	0/36	0/36	0/36
Scenario 6	36/36	0/36	0/36	0/36	0/36	0/36

For example, in scenario 2, this method was successful at least 95% of the time in 34 of 36 regression conditions. Recall that each scenario has 36 regression conditions, so for scenario 6, this method was successful at least 950 out of 1000 times for 34 of the 36 regression conditions. Similarly, Table 5.10 summarizes the tpswamp value in percentage for the six scenarios. For example in scenario 2, the percentage of clean observations is classified as an outlier is between 0 to 5% in 34 out of 36 conditions.

From Figure 5.6, where the sample size is 20, the detection probability is decreased significantly with the increase in the number of regressor from $p = 1$ to $p = 2$ in situations 1, 2 through 18. The detection probability also increase significantly with the increase in the number of regressor from $p = 1$ to $p = 6$. However, the detection probability increases significantly with the increase in the number of regressor from $p = 1, p =2$ to $p = 6$ in situations 19, 20 through 24. Figures 5.7 and 5.8 show that for large n ($n = 40$ and $n = 60$ for this case) the detection probability is high and quite the same for every situation and condition except for situation 2, 4, 10 and 12 in condition $p = 2$. These situations come from the data with 10σ outlier distance and xy-space outlier scenario. Figures 5.6 – 5.9 also indicate that the probability of swamping decreases as the number of regressor variable increases.

From Figure 5.9, where the number of regressor is one, the detection probability is increased significantly with the increase in the sample size particularly in situation 3, 9, 11, 15, 19, 21 through 24. For the number of regressor equals to two (Figures 5.10), the detection probability increases significantly with the increase in the sample size in every situation. However, for the number of regressor six (Figures 5.11), the detection probability increases significantly with the increase in the sample size particularly in situations 3, 9, 11, and 12. Figures 5.9 – 5.11 shows that Method 1 has difficulty in detecting the presence of outliers in situations 3, 9, and 11. Situations 3, 9, and 11 are those with outliers that are 5σ away from the rest of the data. Figures 5.9 – 5.11 also indicate that the probability of swamping decreases as the sample size increase.

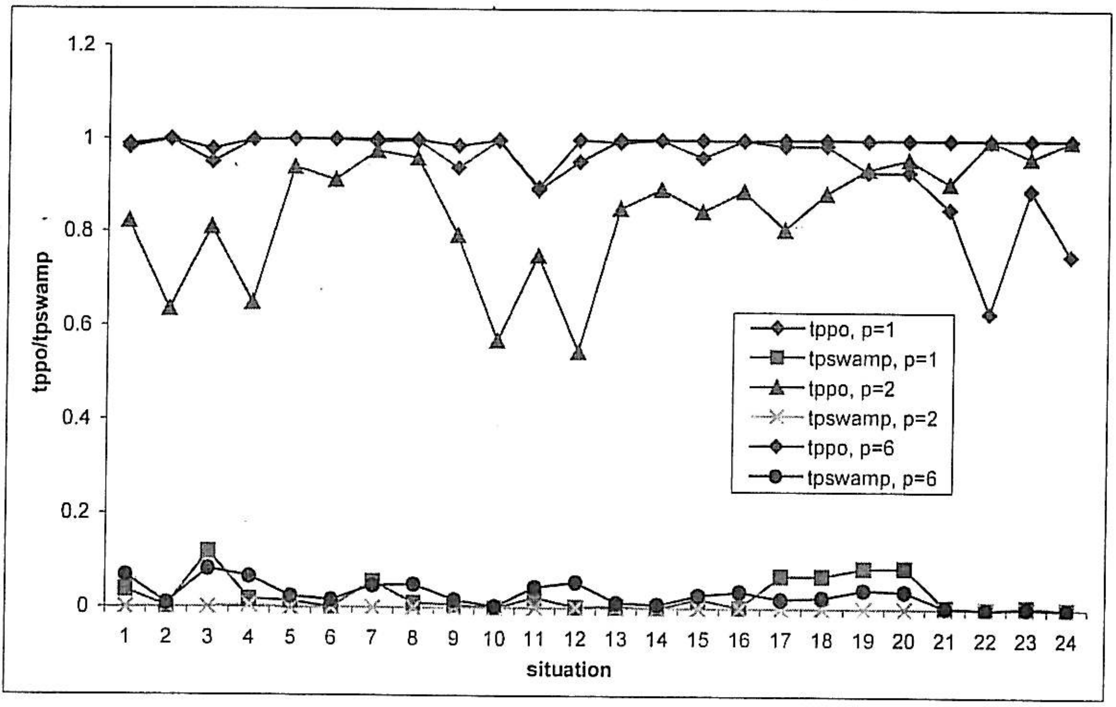


Figure 5.6: Performance of Method 1 for $n = 20$ and all values of p

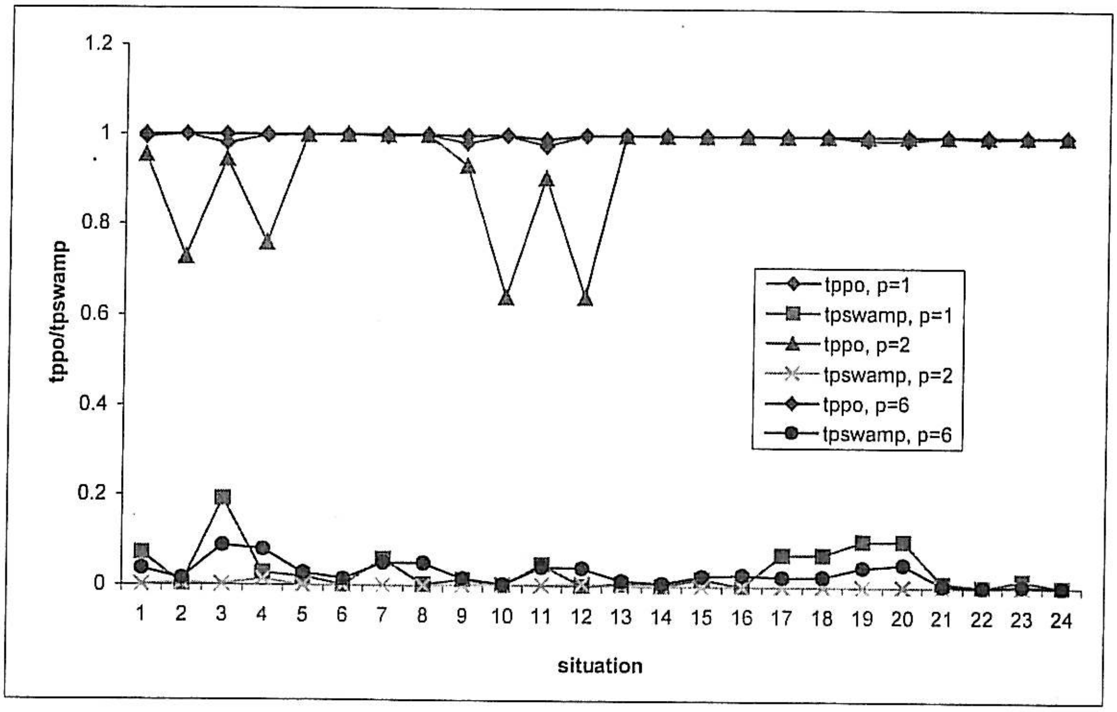


Figure 5.7: Performance of Method 1 for $n = 40$ and all values of p

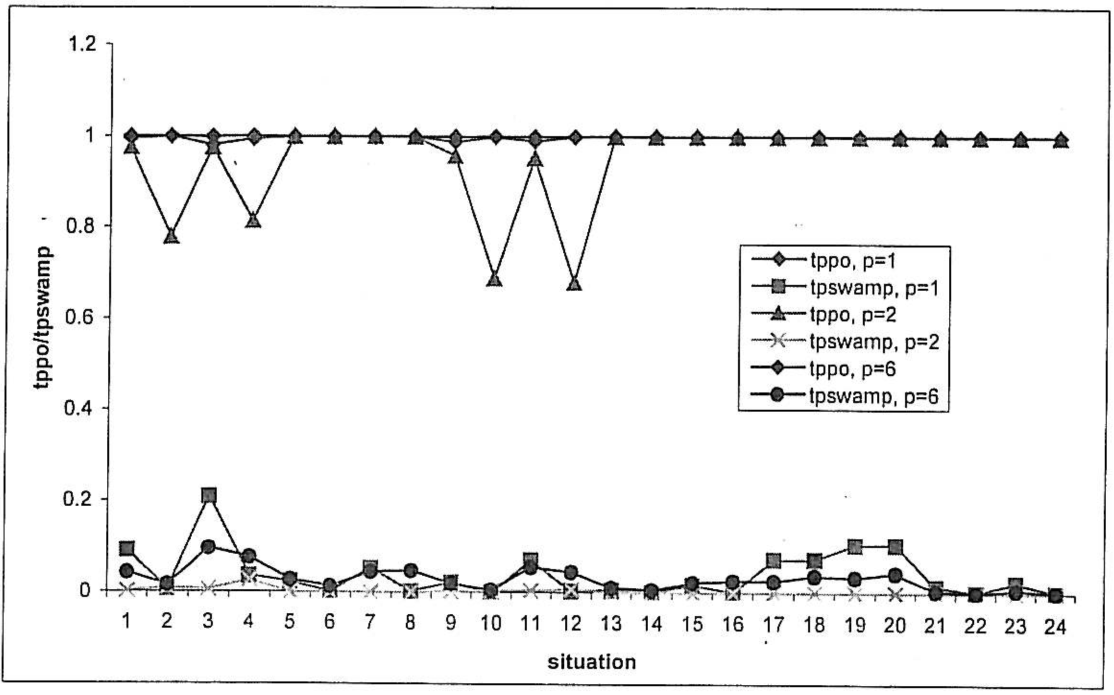


Figure 5.8: Performance of Method 1 for n = 60 and all values of p

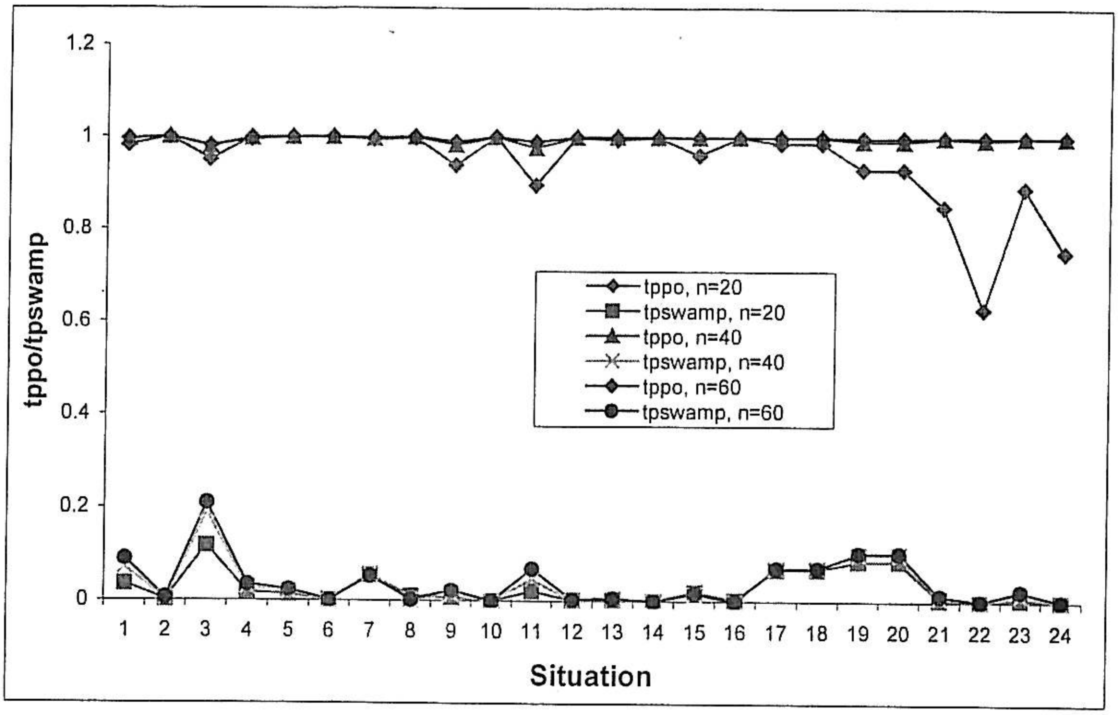


Figure 5.9: Performance of Method 1 for p =1 and all values of n

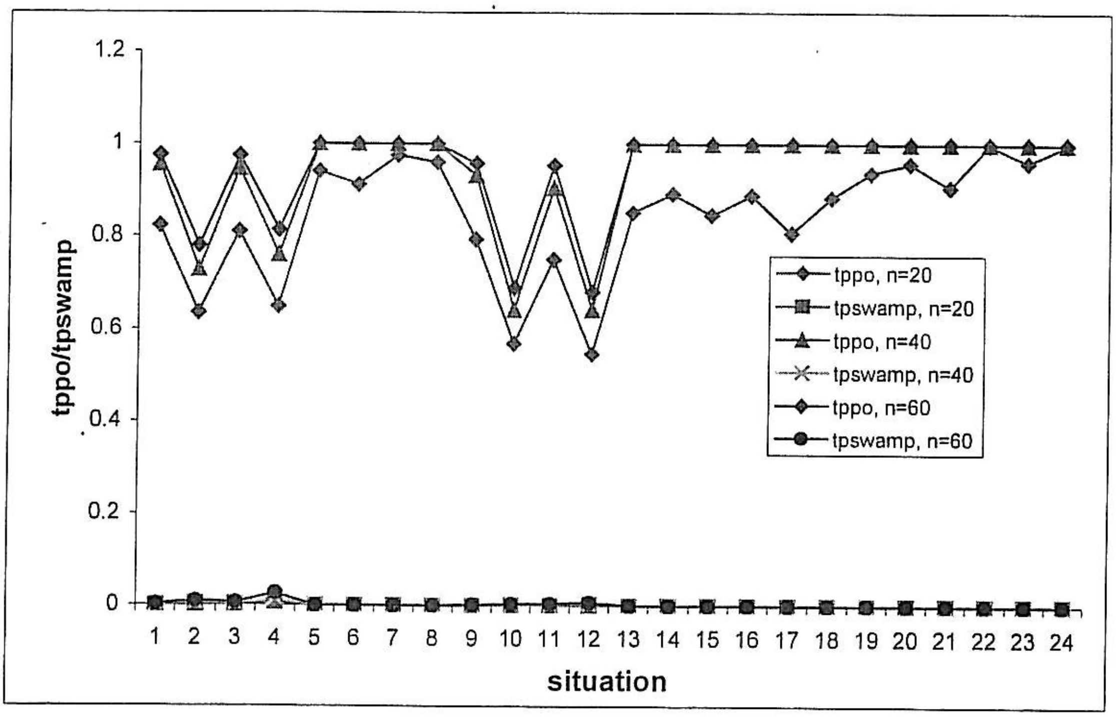


Figure 5.10: Performance of Method 1 for $p = 2$ and all values of n

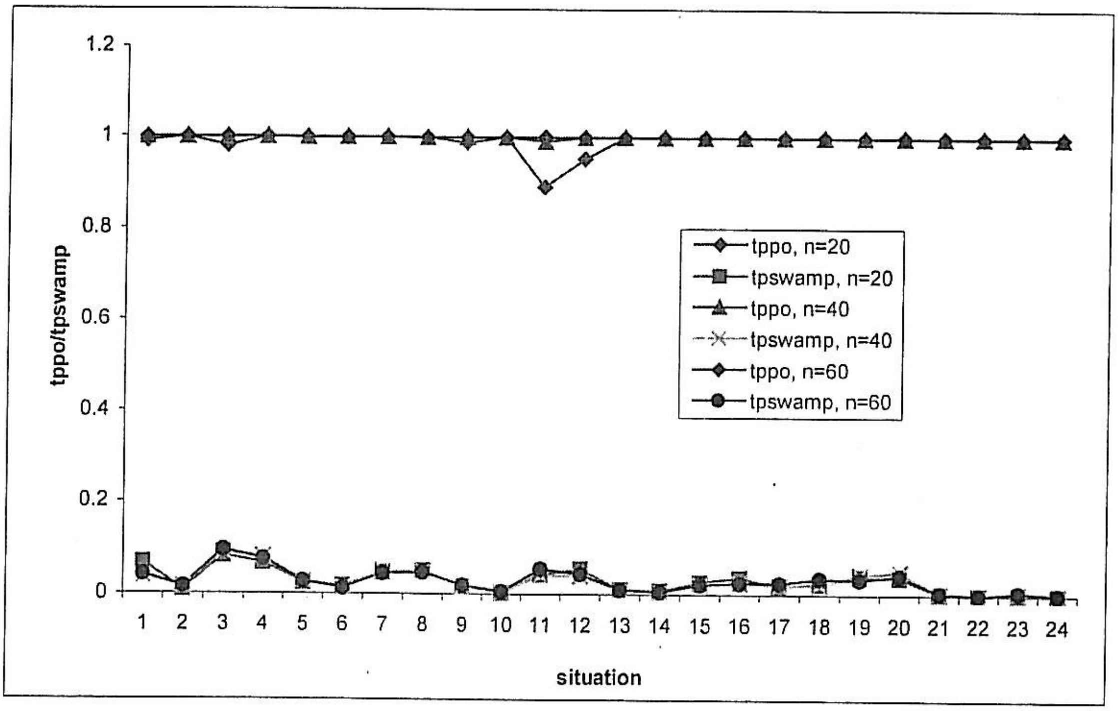


Figure 5.11: Performance of Method 1 for $p = 6$ and all values of n

Table 5.11: Summary of Method 1 performance for each scenario

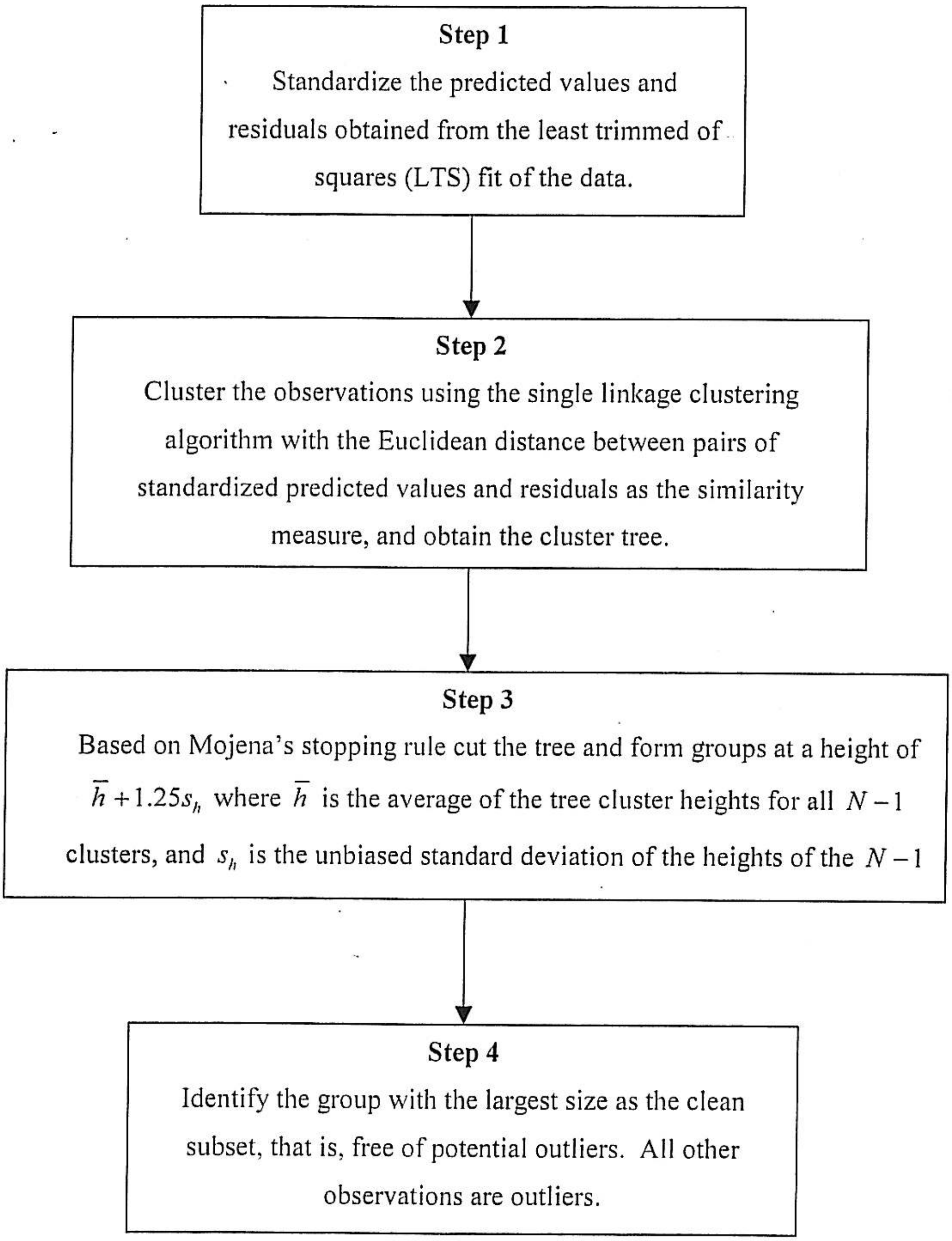
		No of regressor increase	**No of observation increase**	**Outlier % increase**	**Outlier distance increase**
Scenario 1	tppo	increase	increase	decrease	increase
	tpswamp	decrease	increase	increase	decrease
Scenario 2	tppo	increase	increase	decrease	increase
	tpswamp	decrease	increase	increase	decrease
Scenario 3	tppo	increase	increase	decrease	increase
	tpswamp	decrease	increase	increase	decrease
Scenario 4	tppo	increase	increase	decrease	increase
	tpswamp	decrease	increase	increase	decrease
Scenario 5	tppo	increase	increase	decrease	increase
	tpswamp	decrease	increase	increase	decrease
Scenario 6	tppo	increase	increase	decrease	increase
	tpswamp	decrease	increase	increase	decrease

Table 5.11 summarizes the performance of Method 1 for each scenario. The following provides the general observations and conclusions concerning the performance of Method 1. Since this method is the modification of Sebert et al. method, the tppo (total probability a planted outlier is detected) value also increases as the outlying distance and the number of regressor variables increases. Besides, the tppo value also increases as the number of observations in the data set increases and as the percentage of outliers decreases. Further, the tpswamp (total probability a clean observation is classified as an outlier) value decreases as the outlying distance and the number of regressor variables increases. However, the tpswamp value decreases as the percentage of outliers decreases. In general, this method also performs best (high tppo value with low tpswamp value) at lower outliers percentages.

5.6 Method 2

Method 2 is a modification of Sebert's method where the least squares (LS) fit is replaced by the least trimmed of squares (LTS) fit. It is a well-known fact that the least trimmed of squares (LTS) is a high breakdown estimator, therefore the proposed method uses the standardized predicted and residual values from the least trimmed of squares (LTS) fit rather than the ordinary least squares (LS) fit. The flowchart for this method is presented in Figure 5.12.

The steps of the methodology will now be discussed in detailed and illustrated with the "Modified Wood Gravity" data set given by Rousseuw and Leroy (1987) which shown in Table 4.3. Table 5.12 shows the standardized predicted and residual values from the least trimmed of squares (LTS) for the Modified Wood Gravity data. Figure 5.13 shows the output from S-PLUS agglomerative hierarchical clustering. Further, Figure 5.14 shows a plot of the standardized predicted values and residuals for the wood data using the least trimmed of squares (LTS) fit. The cluster tree and corresponding cut height is shown in Figure 5.15.



Step 1

Standardize the predicted values and residuals obtained from the least trimmed of squares (LTS) fit of the data.

Step 2

Cluster the observations using the single linkage clustering algorithm with the Euclidean distance between pairs of standardized predicted values and residuals as the similarity measure, and obtain the cluster tree.

Step 3

Based on Mojena's stopping rule cut the tree and form groups at a height of $\bar{h} + 1.25s_h$ where $\bar{h}$ is the average of the tree cluster heights for all $N-1$ clusters, and s_h is the unbiased standard deviation of the heights of the $N-1$

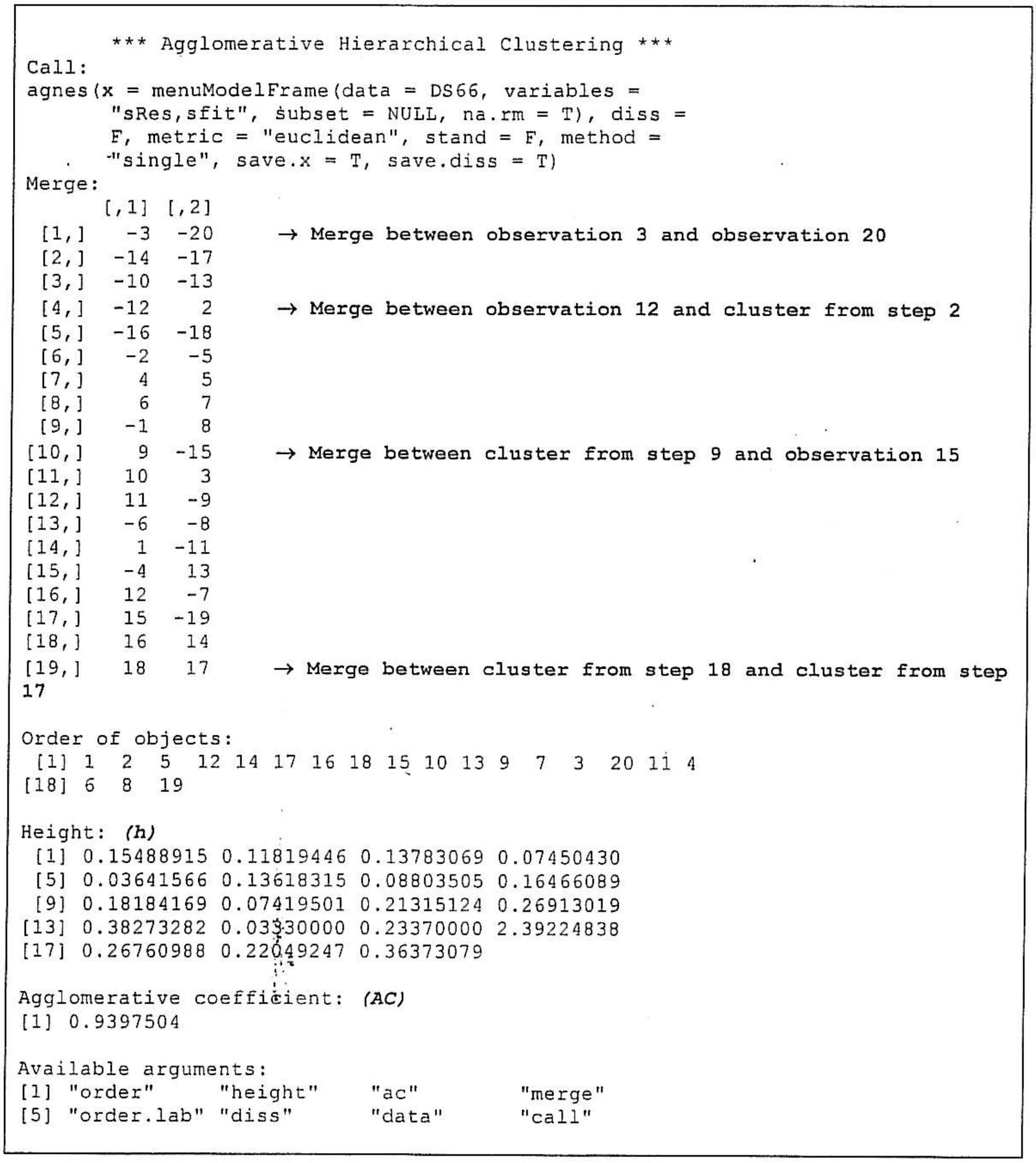
Step 4

Identify the group with the largest size as the clean subset, that is, free of potential outliers. All other observations are outliers.

Figure 5.12: Steps in Method 2

Table 5.12: Standardized least trimmed of squares (LTS) predicted values and residuals for Modified Wood Gravity data.

Obs.	Predicted Values	Standardized Predicted Values	Residual	Standardized Residual
1	0.5197	-0.3878	0.0143	0.6085
2	0.5296	-0.2225	0.0054	0.5122
3	0.5700	0.4512	0.0000	0.4541
4	0.6415	1.6455	-0.1915	-1.6187
5	0.5332	-0.1633	0.0148	0.6145
6	0.6518	1.8165	-0.2208	-1.9351
7	0.4635	-1.3260	0.0175	0.6432
8	0.6390	1.6023	-0.2160	-1.8828
9	0.4750	-1.1345	0.0000	0.4541
10	0.4882	-0.9145	-0.0022	0.4306
11	0.5540	0.1842	0.0000	0.4541
12	0.5190	-0.4001	0.0000	0.4541
13	0.4874	-0.9272	0.0046	0.5037
14	0.5220	-0.3493	-0.0050	0.3996
15	0.4984	-0.7437	0.0036	0.4930
16	0.5080	-0.5837	0.0000	0.4541
17	0.5241	-0.3142	-0.0041	0.4093
18	0.5117	-0.5213	-0.0057	0.3920
19	0.6550	1.8705	-0.2540	-2.2948
20	0.5680	0.4179	0.0000	0.4541



```
        *** Agglomerative Hierarchical Clustering ***
Call:
agnes(x = menuModelFrame(data = DS66, variables =
       "sRes,sfit", subset = NULL, na.rm = T), diss =
       F, metric = "euclidean", stand = F, method =
       "single", save.x = T, save.diss = T)
Merge:
       [,1] [,2]
 [1,]   -3  -20       → Merge between observation 3 and observation 20
 [2,]  -14  -17
 [3,]  -10  -13
 [4,]  -12    2       → Merge between observation 12 and cluster from step 2
 [5,]  -16  -18
 [6,]   -2   -5
 [7,]    4    5
 [8,]    6    7
 [9,]   -1    8
[10,]    9  -15       → Merge between cluster from step 9 and observation 15
[11,]   10    3
[12,]   11   -9
[13,]   -6   -8
[14,]    1  -11
[15,]   -4   13
[16,]   12   -7
[17,]   15  -19
[18,]   16   14
[19,]   18   17       → Merge between cluster from step 18 and cluster from step
17

Order of objects:
 [1] 1  2  5  12 14 17 16 18 15 10 13 9  7  3  20 11 4
[18] 6  8  19

Height: (h)
 [1] 0.15488915 0.11819446 0.13783069 0.07450430
 [5] 0.03641566 0.13618315 0.08803505 0.16466089
 [9] 0.18184169 0.07419501 0.21315124 0.26913019
[13] 0.38273282 0.03330000 0.23370000 2.39224838
[17] 0.26760988 0.22049247 0.36373079

Agglomerative coefficient: (AC)
[1] 0.9397504

Available arguments:
[1] "order"     "height"    "ac"        "merge"
[5] "order.lab" "diss"      "data"      "call"
```

Figure 5.13: The output from S-PLUS agglomerative hierarchical clustering for Modified Wood gravity data using the least trimmed of squares (LTS) fit.

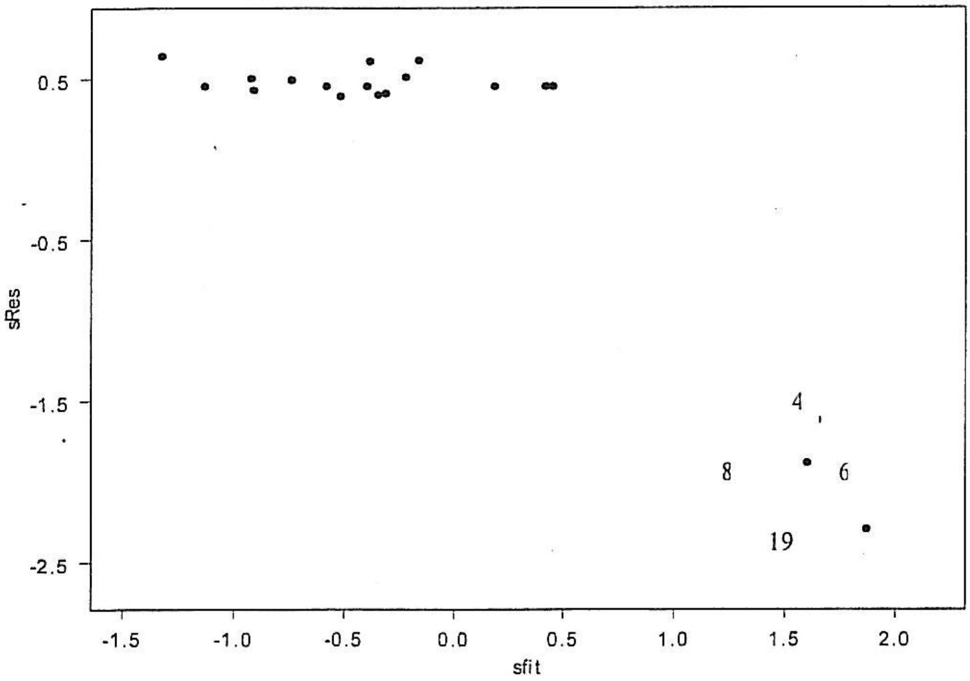


Figure 5.14: Plot of the standardized predicted (sfit) and residuals (sRes) values for the Modified Wood Gravity data using the least trimmed of squares (LTS) fit.

Again, based on Mojena's stopping rule, the tree will be cut and formed groups at a height of $\bar{h}+1.25s_h$. For this data set, $\bar{h}=0.291729$ and $s_h=0.518238$. Therefore the cut height on the cluster tree is $0.291729+1.25*0.518238=0.939526$. Referring to Figure 5.9, it can be seen that after the cut there are two groups formed. Going across the tree from right to left, Group 1 consists of observations 19, 8, 6, and 4. Group two consists of observations 11, 20, 3, 13, 10, 9, 7, 15, 2, 5, 18, 16, 17, 14, 12, and 1. Group 2 contains the majority of the observations and thus this set will be the inlying observations. Observations 4, 6, 8, and 19 are identified as the outlying observations. The outlying observations identified by this methodology are also noted in Figure 5.7.

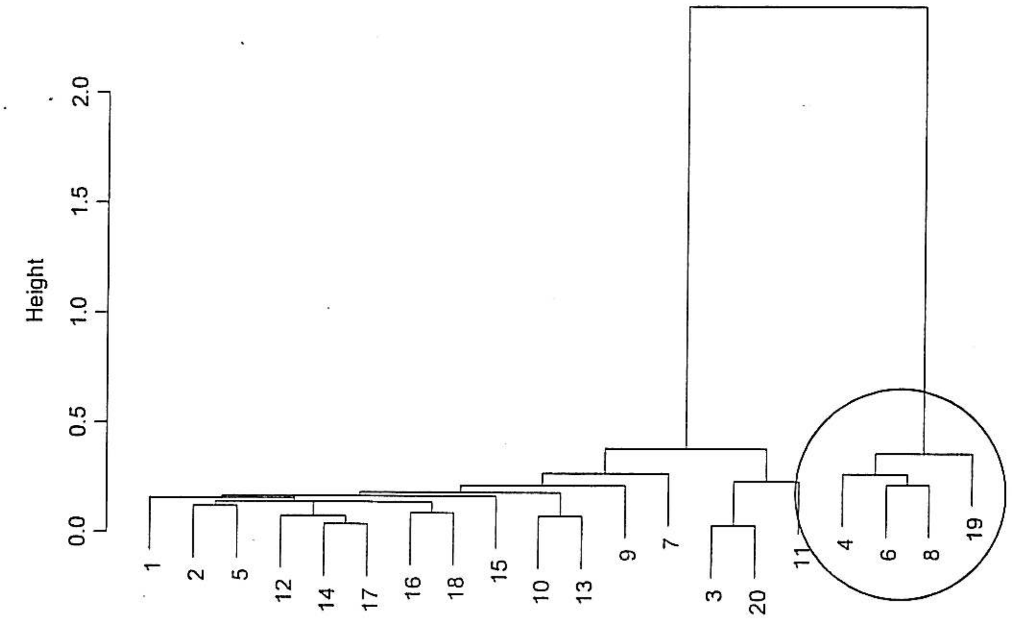


Figure 5.15: Cluster tree and Mojena's cut height for the Modified Wood Gravity data using the least trimmed of squares (LTS) fit.

The performance of the methodology on the classic data sets is summarized in Table 5.13.. It can be seen that the methodology successfully identified all the outliers for all of the data sets. The method performed perfectly for 3 out of the 5 data sets in the sense that there was no masking or swamping. When there was swamping or masking, the number of observations swamped or masked is small. Appendix F shows the full computation and results for the other 4 classic data sets using Method 2.

Table 5.13: Method 2's performance on classic multiple outlier data sets

No	Data sets	Outlying observation	Outlying observations identified	Number of observations swamped	Number of observations masked
1	Telephone Data (Rousseuw and Leroy, 1987)	15-24	15-24	0	0
2	Hertzsprung-Russell StarsData (Rousseuw and Leroy, 1987)	11, 20, 30, 34	11, 20, 30, 34, 7, 14	2	0
3	Hawkins, Bradu, and Kass Data (Hawkins et al., 1984)	1-14	1- 10	0	4
4	Modified Wood Gravity Data (Rousseuw and Leroy, 1987)	4, 6, 8, 19	4, 6, 8, 19	0	0
5	Stackloss Data (Brownlee,1965)	1-4, 21	1-4, 21	0	0

"Method 2" clustering methodology discussed in this research has been shown to perform well on the classic data set. However to further understand the performance of the methods, a detailed study of the procedure on randomly generated data sets was performed. The results showing the performance of the Method 2 for each scenario is provided in Table 5.14 – Table 5.19 and Figures 5.16 – 5.21. Again, scenario 1 consists of situations 1-4, while scenario 2 consists of situations 5-8 and so on. Appendix G shows the simulation code for the Method 2.

Table 5.14: Scenario 1 result for the Method 2

			$n = 20$		$n = 40$		$n = 60$	
No of regressor (p)	outlier %	outlier distance	tppo	tpswamp	tppo	tpswamp	tppo	tpswamp
1	10	5	0.9860	0.0293	0.9970	0.0638	0.9955	0.0865
	10	10	1	0.0018	1	0.0035	1	0.0060
	20	5	0.9313	0.1447	0.9753	0.2090	0.9743	0.2547
	20	10	0.9970	0.0371	0.9970	0.0448	0.9960	0.0451
2	10	5	0.8205	0.0001	0.9550	0.0006	0.9707	0.0019
	10	10	0.6250	0.0006	0.7118	0.0032	0.7687	0.0081
	20	5	0.6918	0.0009	0.8973	0.0033	0.9475	0.0057
	20	10	0.6163	0.0018	0.7373	0.0106	0.7995	0.0232
6	10	5	0.9920	0.0410	1	0.0363	1	0.0396
	10	10	1	0.0046	1	0.0018	1	0.0043
	20	5	0.9558	0.0946	1	0.0905	0.999	0.1003
	20	10	0.9890	0.1066	1	0.0926	1	0.0904

Table 5.15: Scenario 2 result for the Method 2

			$n = 20$		$n = 40$		$n = 60$	
No of regressor (p)	outlier %	outlier distance	tppo	tpswamp	tppo	tpswamp	tppo	tpswamp
1	10	5	0.9995	0.0138	1	0.0156	1	0.0226
	10	10	1	0.0001	1	0.0003	1	0.0012
	20	5	0.9928	0.0821	0.9884	0.1166	0.9908	0.1111
	20	10	0.996	0.061	0.997	0.0766	0.9882	0.0852
2	10	5	0.9370	0	1	0	1	0
	10	10	0.889	0	1	0	1	0.0002
	20	5	0.9600	0	1	0	1	0
	20	10	0.9565	0	1	0.0001	1	0.0001
6	10	5	1	0.0366	1	0.0358	1	0.0347
	10	10	1	0.0228	1	0.0124	1	0.0088
	20	5	1	0.0686	1	0.0614	1	0.0586
	20	10	1	0.0787	1	0.0658	1	0.0642

Table 5.16: Scenario 3 result for the Method 2

			$n = 20$		$n = 40$		$n = 60$	
No of regressor (p)	outlier %	outlier distance	tppo	tpswamp	tppo	tpswamp	tppo	tpswamp
1	10	5	0.9385	0.006	0.9828	0.0142	0.9895	0.0210
	10	10	1	0	1	0.0001	1	0.0005
	20	5	0.8685	0.0263	0.971	0.0558	0.9847	0.0749
	20	10	0.9985	0.0062	0.9995	0.0029	1	0.0041
2	10	5	0.8035	0.0001	0.9300	0.0006	0.9553	0.0021
	10	10	0.5565	0.0002	0.626	0.0013	0.6757	0.0035
	20	5	0.6223	0.0008	0.8248	0.0028	0.9008	0.0052
	20	10	0.5293	0.0004	0.6236	0.0032	0.6691	0.0061
6	10	5	0.9725	0.0267	0.9993	0.0185	1	0.0170
	10	10	0.9985	0.0033	1	0.0001	1	0.0007
	20	5	0.7805	0.0723	0.9898	0.0641	0.9999	0.0638
	20	10	0.9270	0.0769	1	0.0671	1	0.0593

Table 5.17: Scenario 4 result for the Method 2

			$n = 20$		$n = 40$		$n = 60$	
No of regressor (p)	outlier %	outlier distance	tppo	tpswamp	tppo	tpswamp	tppo	tpswamp
1	10	5	0.9920	0.0044	1	0.0045	1	0.0065
	10	10	1	0	1	0	1	0
	20	5	0.9428	0.0318	0.9995	0.0342	0.9995	0.0398
	20	10	0.9993	0.0137	0.9995	0.0172	0.9995	0.0198
2	10	5	0.8465	0	1	0	1	0
	10	10	0.8660	0	1	0.0001	1	0.0003
	20	5	0.7560	0	1	0	1	0
	20	10	0.8785	0	1	0.0001	1	0.0003
6	10	5	1	0.0221	1	0.0172	1	0.0140
	10	10	1	0.0137	1	0.0060	1	0.0035
	20	5	1	0.0494	1	0.0367	1	0.0334
	20	10	0.9978	0.0517	1	0.0416	1	0.0362

Table 5.18: Scenario 5 result for the Method 2

			$n = 20$		$n = 40$		$n = 60$	
No of regressor **(*p*)**	**outlier** **%**	**outlier** **distance**	**tppo**	**tpswamp**	**tppo**	**tpswamp**	**tppo**	**tpswamp**
1	10	5	0.9865	0.0672	1	0.0673	1	0.0743
	10	10	0.9865	0.0672	1	0.0673	1	0.0743
	20	5	0.9178	0.0891	0.9953	0.0999	0.9982	0.1029
	20	10	0.9178	0.0891	0.9953	0.0999	0.9982	0.1029
2	10	5	0.7935	0	1	0	1	0
	10	10	0.8590	0	1	0	1	0.0002
	20	5	0.9395	0	1	0	1	0
	20	10	0.9615	0	1	0.0001	1	0.0003
6	10	5	1	0.0446	1	0.0390	1	0.0369
	10	10	1	0.0448	1	0.0390	1	0.0368
	20	5	1	0.0626	1	0.0611	1	0.0598
	20	10	1	0.0620	1	0.0608	1	0.0599

Table 5.19: Scenario 6 result for the Method 2

			$n = 20$		$n = 40$		$n = 60$	
No of regressor **(*p*)**	**outlier** **%**	**outlier** **distance**	**tppo**	**tpswamp**	**tppo**	**tpswamp**	**tppo**	**tpswamp**
1	10	5	0.8190	0.0039	1	0.0081	1	0.0127
	10	10	0.595	0	0.995	0.0002	1	0.0008
	20	5	0.8750	0.0046	0.9995	0.0127	1	0.0226
	20	10	0.739	0	1	0.0002	1	0.0013
2	10	5	0.899	0	1	0	1	0
	10	10	1	0	1	0	1	0
	20	5	0.9465	0	1	0	1	0
	20	10	0.999	0	1	0	1	0.0001
6	10	5	1	0.0058	1	0.0062	1	0.0067
	10	10	1	0.0001	1	0.0002	1	0.0004
	20	5	1	0.0028	1	0.0039	1	0.0069
	20	10	1	0	1	0	1	0.0002

The Method 2 simulation result also performs well for most of regression condition tested except in scenario 1 and 3 for $p = 2$. Table 5.20 summarizes the tppo value in percentage for the six scenarios. Similarly, Table 5.21 summarizes the tpswamp value in percentage for the six scenarios. It shows that, the probability of swamping in this method is quite high.

Table 5.20: Total probability a planted outlier is detected (in percentage) of the Method 2 in all regression conditions tested

%	100-95	94.9-90	89.9-85	84.9-80	79.9-75	74.9-70	<70
Scenario 1	25/36	2/36	1/36	1/36	2/36	2/36	3/36
Scenario 2	34/36	1/36	1/36	0/36	0/36	0/36	0/36
Scenario 3	21/36	4/36	1/36	2/36	1/36	0/36	7/36
Scenario 4	31/36	1/36	2/36	1/36	1/36	0/36	0/36
Scenario 5	31/36	3/36	1/36	0/36	1/36	0/36	0/36
Scenario 6	30/36	1/36	2/36	1/36	0/36	1/36	1/36

Table 5.21: Total probability a clean observation is classified as an outlier (in percentage) of the Method 2 in all regression conditions tested

%	0-4.9	5-9.9	10-14.9	15-19.9	20-24.9	>25
Scenario 1	25/36	6/36	3/36	0/36	1/36	1/36
Scenario 2	24/36	10/36	2/36	0/36	0/36	0/36
Scenario 3	28/36	8/36	0/36	0/36	0/36	0/36
Scenario 4	35/36	1/36	0/36	0/36	0/36	0/36
Scenario 5	18/36	16/36	2/36	0/36	0/36	0/36
Scenario 6	36/36	0/36	0/36	0/36	0/36	0/36

From Figure 5.16, where the sample size is 20, the detection probability decreases significantly with the increase in the number of regressor from $p = 1$ to $p = 2$ in situations 1, 2 through 18. The detection probability also increase significantly with the increase in the number of regressor from $p = 1$ to $p = 6$. However, the detection probability increases significantly with the increase in the number of regressor from $p = 1, p = 2$ to $p = 6$ in situations 19, 20 through 24.

Figures 5.17 and 5.18 show that for large n ($n = 40$ and $n = 60$ for this case) the detection probability is high and quite the same for every situation and condition except for situation 2, 4, 10 and 12 in condition $p = 2$. These situations come from the data with 10σ outlier distance and *xy*-space outlier scenario. Figures 5.16 – 5.18 also indicate that the probability of swamping decreases as the number of regressor variable increases except for $p = 2$.

From Figure 5.19, where the number of regressor is one, the detection probability is increase significantly with the increase in the sample size particularly in situation 3, 9, 11, 15, 19, 21 through 21. For the number of regressor equals to two (Figures 5.20), the detection probability increases significantly with the increase in the sample size in every situations.

However, for the number of regressor six (Figures 5.21), the detection probability is increase significantly with the increase in the sample size particularly in situations 3, 9, 11, and 12. Figures 5.19 – 5.21 shows that Method 1 has difficulty to detect the presence of outliers in situations 3, 9, and 11. Situations 3, 9, and 11 are those with outliers that are 5σ away from the rest of the data. Figures 5.9 – 5.11 also indicate that the probability of swamping is decreases as the sample size increase.

From Figure 5.16 – 5.21, the graph for Method 1 looks almost the same as the graph for Method 2. Thus, the performance result and explanation of the graph is also same.

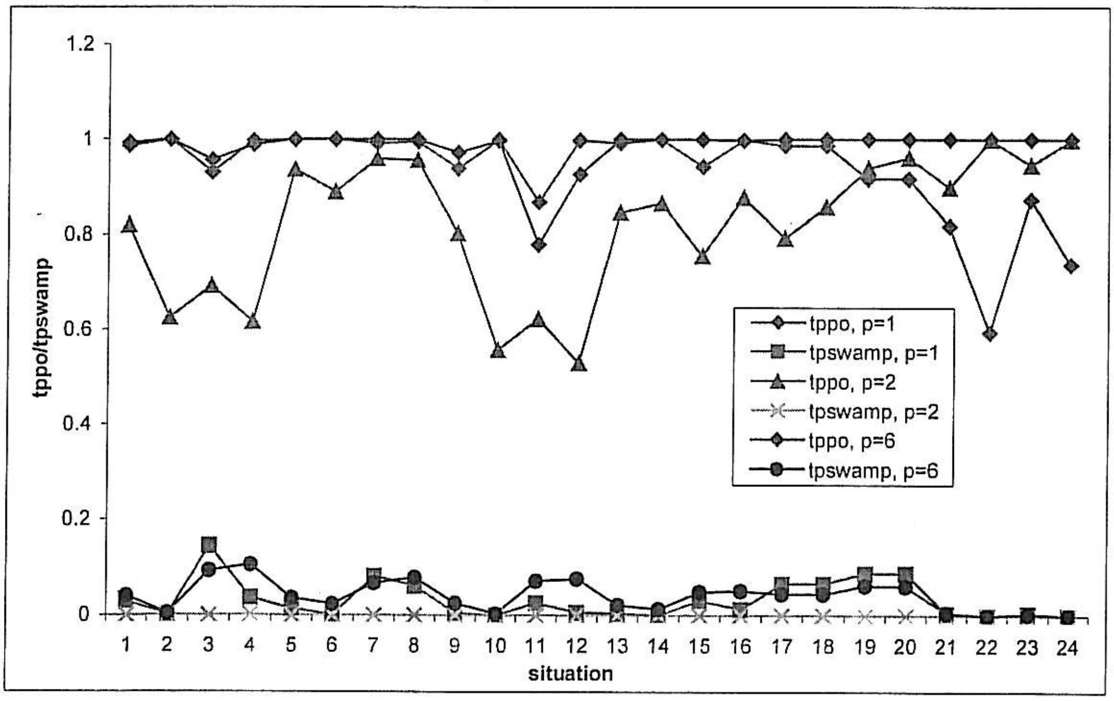


Figure 5.16: Performance of Method 2 for $n = 20$ and all values of p

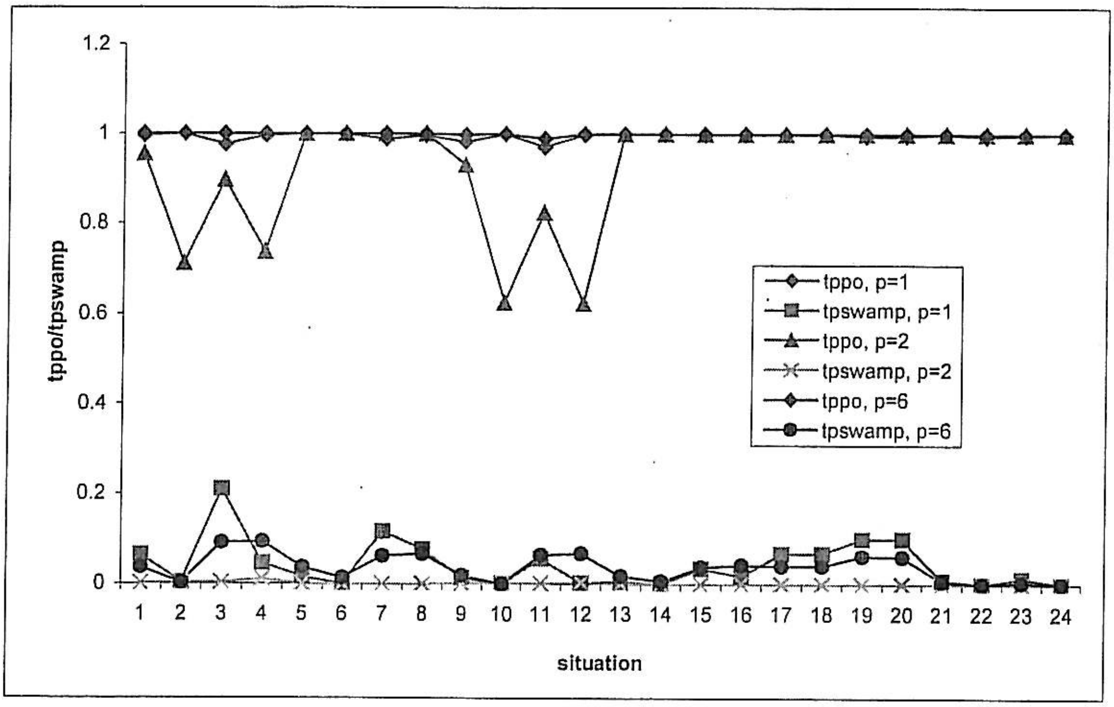


Figure 5.17: Performance of Method 2 for $n = 40$ and all values of p

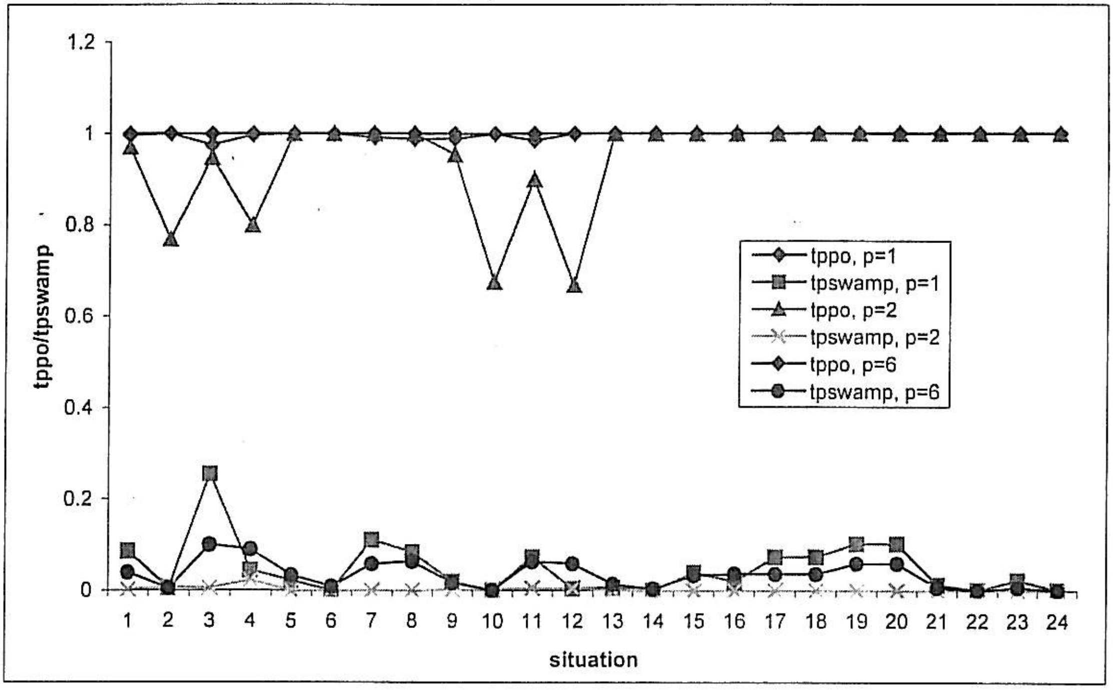


Figure 5.18: Performance of Method 1 for $n = 60$ and all values of p

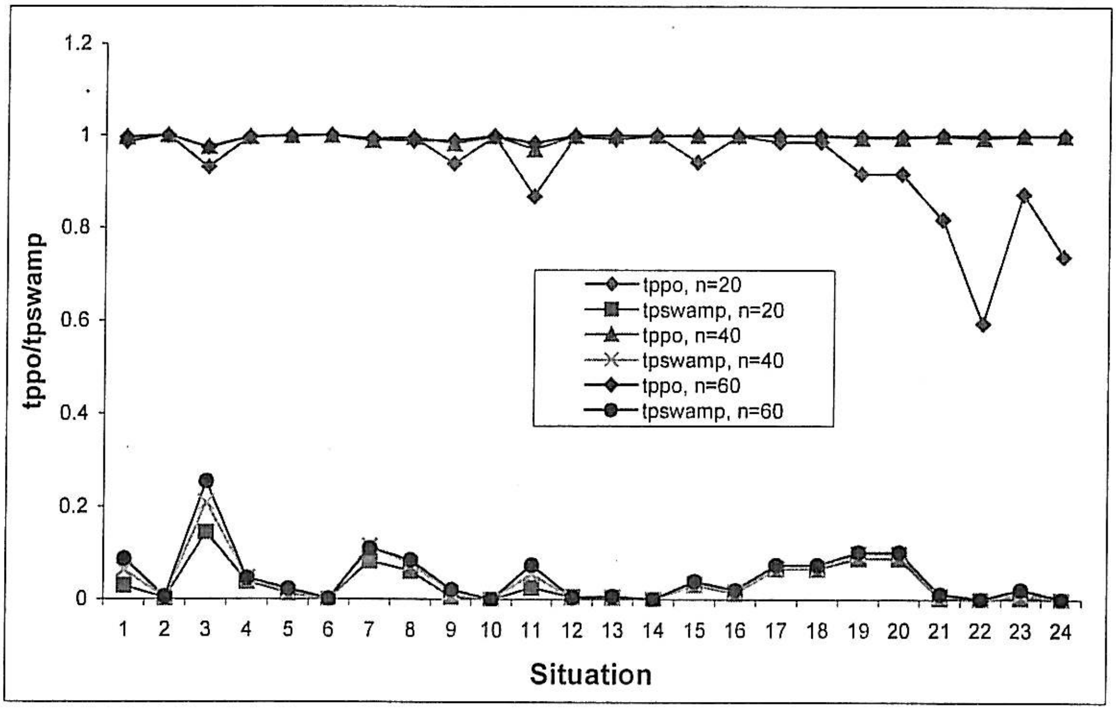


Figure 5.19: Performance of Method 1 for $p = 1$ and all values of n

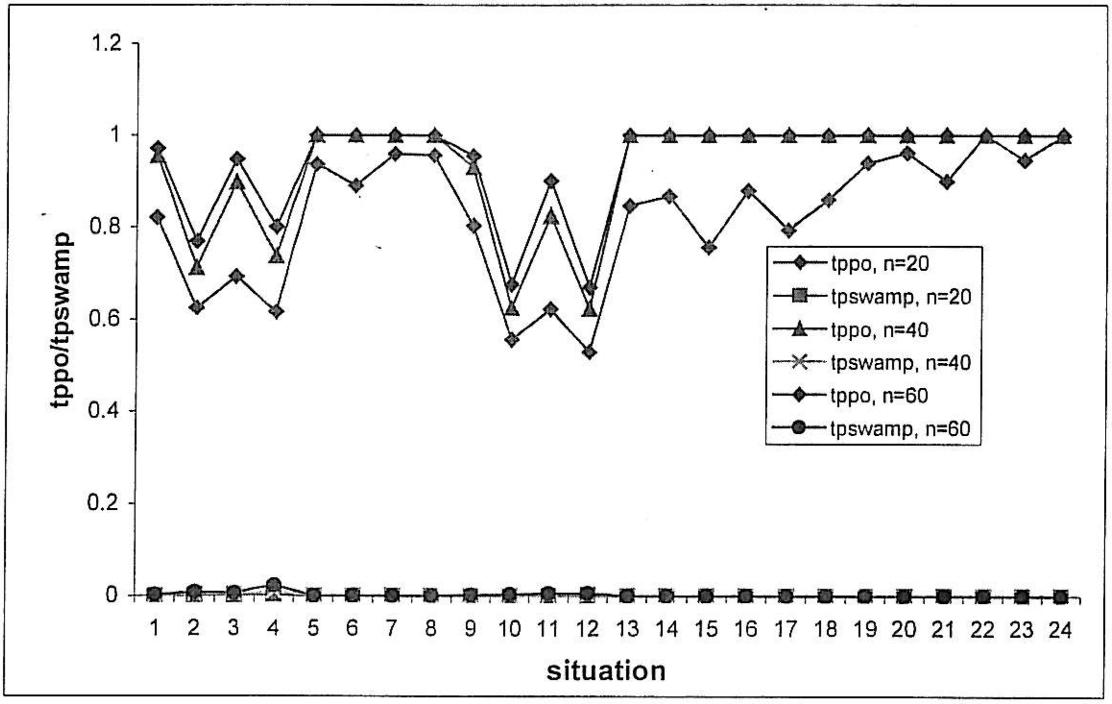


Figure 5.20: Performance of Method 2 for $p = 2$ and all values of n

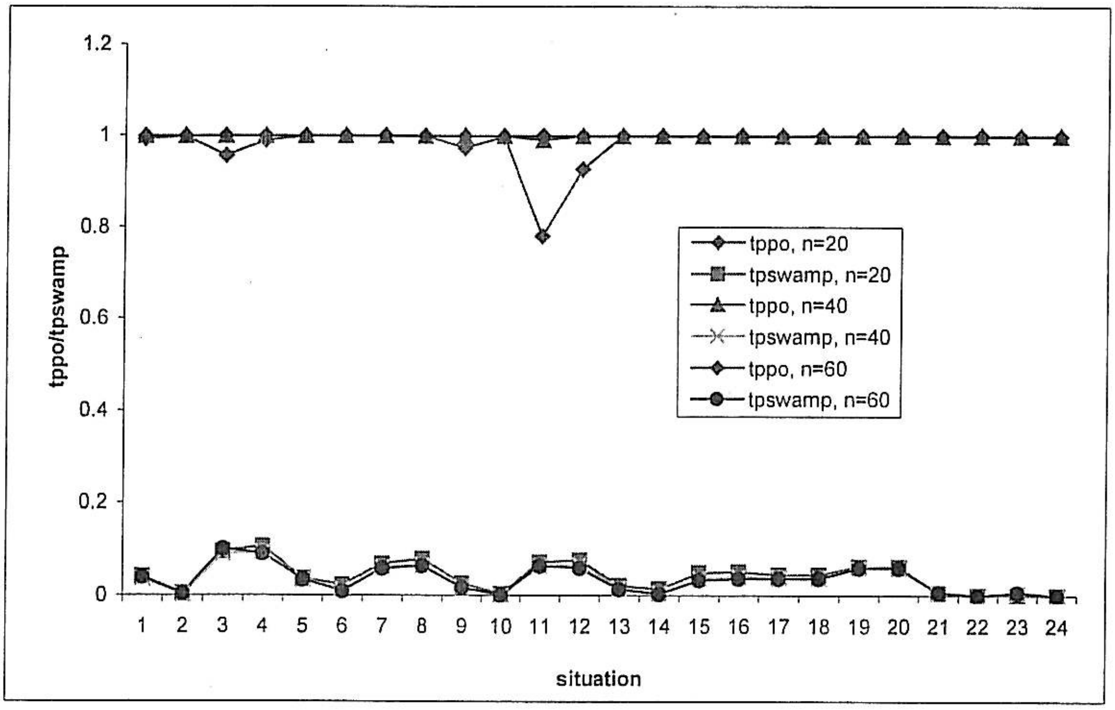


Figure 5.21: Performance of Method 2 for $p = 6$ and all values of n

Table 5.22: Summary of Method 2 performance for each scenario

		No of regressor increase	**No of observation increase**	**Outlier % increase**	**Outlier distance increase**
Scenario 1	tppo	increase	increase	decrease	increase
	tpswamp	decrease	increase	increase	decrease
Scenario 2	tppo	increase	increase	decrease	increase
	tpswamp	decrease	increase	increase	decrease
Scenario 3	tppo	increase	increase	decrease	increase
	tpswamp	decrease	increase	increase	decrease
Scenario 4	tppo	increase	increase	decrease	increase
	tpswamp	decrease	increase	increase	decrease
Scenario 5	tppo	increase	increase	decrease	increase
	tpswamp	decrease	increase	increase	decrease
Scenario 6	tppo	increase	increase	decrease	increase
	tpswamp	decrease	increase	increase	decrease

Table 5.22 summarizes the performance of Method 2 for each scenario. The following provides the general observations and conclusions concerning the performance of Method 2. Same as the Method 1, the tppo (total probability a planted outlier is detected) value also increases as the outlying distance and the number of regressor variables increases. Besides, the tppo value also increases as the number of observations in the data set increases and as the percentage of outliers decreases. Further, the tpswamp (total probability a clean observation is classified as an outlier) value decreases as the outlying distance and the number of regressor variables increases. However, the tpswamp value decreases as the percentage of outliers decreases. In general, this method also performs best (high tppo value with low tpswamp value) at lower outliers percentages.

5.7 Summary and Discussion

Two procedures known as Method 1 and Method 2, which use the robust fit and clustering technique, are discussed in this chapter. The Least Median of Squares (LMS) and Least Trimmed of Squares (LTS) fits are used to obtain the predicted and residuals from the data set. Then, the Euclidean distance is used with the single linkage clustering algorithm to cluster the points in the plot of standard predicted versus residuals values. A cluster tree is obtained and the Mojena's stopping rule is used to choose the outliers.

The conclusion given for the Method 1 and Method 2 is quite the same since the LMS and LTS estimator have the same high breakdown, efficient and bounded influence.

CHAPTER 6

COMPARISON ANALYSIS

6.1 Introduction

This chapter discusses the performance among Sebert et al. (1998) clustering algorithm, Method 1 and Method 2 for identifying multiple outliers in linear regression. Comparison analysis was done using the result from classical data and simulation performance.

6.2 Performance on Classical Data

Each method discussed in this research was tested using 5 classical multiple outlier data sets. These data sets are the most popular and widely used in any multiple outlier papers. The performances of Sebert's method, Method 1 and Method 2 on the classical data sets are summarized in Table 6.1. It can be seen that Sebert's method successfully identified all the outliers for all the data sets. The method performed perfectly for 3 out of the 5 data sets in the sense that there was no swamping. Method 1 and 2 only successfully identified all the outliers for the data sets, which the number of observation is less than 60. Masking occurred when the number of observation is greater than 60 for the Hawkins, Bradu, and Kass (Hawkins et al., 1984) data set.

Table 6.1: Performance of Sebert's Method, Method 1and Method 2 on classic multiple outlier data sets

				Sebert's Method			Method 1			Method 2		
	p	*n*	O	I	S	M	I	S	M	I	S	M
1	1	24	15-24	15-24	0	0	15-24	0	0	15-24	0	0
2	1	47	11, 20, 30, 34	11, 20, 30, 34, 7, 14	2	0	11, 20, 30, 34, 7, 14	2	0	11, 20, 30, 34, 7, 14	2	0
3	3	75	1-14	1-14	0	0	1- 10, 13, 14	0	2	1- 10	0	4
4	5	20	4, 6, 8, 19	4, 6, 7, 8,17, 19	2	0	4, 6, 8, 19	0	0	4, 6, 8, 19	0	0
5	3	21	1-4, 21	1-4, 21	0	0	1-4, 21	0	0	1-4, 21	0	0

Where

1: Telephone Data (Rousseuw and Leroy, 1987)

2: Hertzsprung-Russell StarsData (Rousseuw and Leroy, 1987)

3: Hawkins, Bradu, and Kass Data (Hawkins et al., 1984)

4: Modified Wood Gravity Data (Rousseuw and Leroy, 1987)

5: Stackloss Data (Brownlee, 1965)

p: Number of regressor variables

n: Total number of observations

O: Outlying observation

I: Outlying observations identified

S: Number of observations swamped

M: Number of observations masked

According to the performance on the classical data, Sebert's Method is better than Method 1 and Method 2 when the number of observation is greater than 60. Besides, Method 1 and Method 2 are better than Sebert's Method when the number of observation is less than 60.

6.3 Overall Performance

From the simulation study, several pattern appeared. Since Method 1 and Method 2 are the modification of Sebert's method, the pattern of the graph is quite the same for every condition, situation and scenario. Appendix H shows the full result of the simulation study for every method. Figures 6.1 – 6.9 show the detection probabilities and the probability of swamping for Sebert's method, Method 1 and Method 2 for every condition. The detailed illustration of the performances by the different methods in the six scenarios are shown in Figures 6.10(a, b, c) – 6.15(a, b, c). Besides, Tables 6.2 – 6.4 summarizes the performances of the Sebert method, Method 1 and Method 2 for different scenario. Table 6.5 show the rate of every method's performance according to the tppo value.

From Figure 6.1 where $n = 20$ and $p = 1$, the detection probability for Method 2 is better than Method 1 and Sebert's method except in situations 21 through 24. These situations come from scenario 6 where there are two outlying groups with one of them an x-space outlier and the other is an xy-space outlying observations. However, the swamping probability for the Method 2 is smaller than Method 1 and Sebert's method in almost scenarios. From Figure 6.2 where $n = 20$ and $p = 2$, the detection probability for Sebert's method is better than Method 1 and Method 2 except for situations 2, 4, 10 and 12. These situations come from scenario 1 and 3, which the outlier distance is 10σ. However, the swamping probability for the Method 2 is also smaller than Method 1 and Sebert's method in almost scenarios.

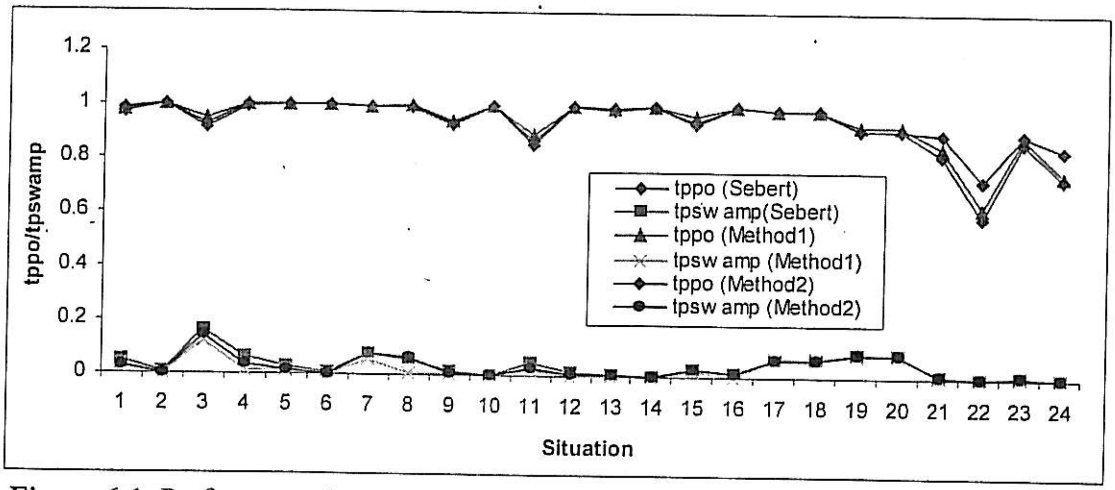


Figure 6.1: Performance between Sebert, Method 1 and Method 2 for $n = 20$ and $p = 1$

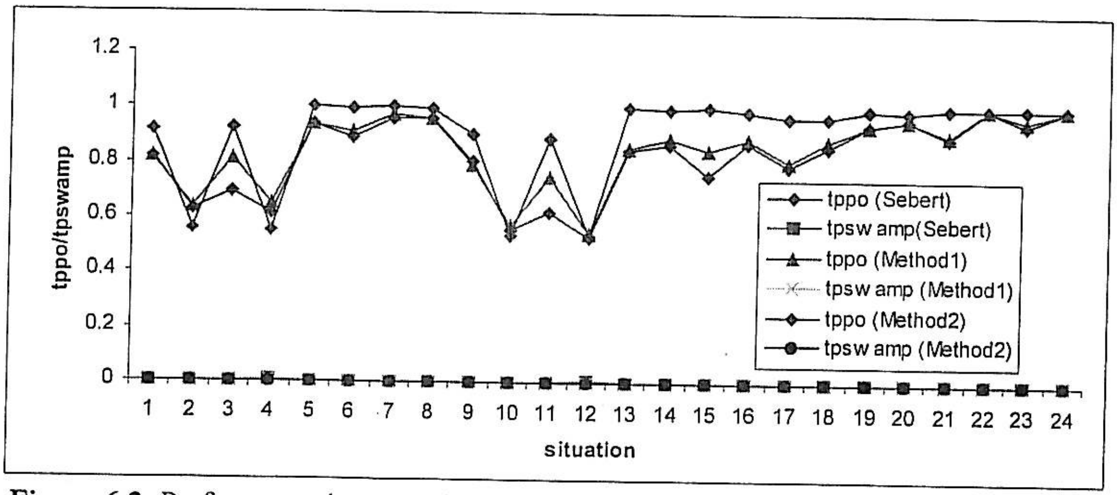


Figure 6.2: Performance between Sebert, Method 1 and Method 2 for $n = 20$ and $p = 2$

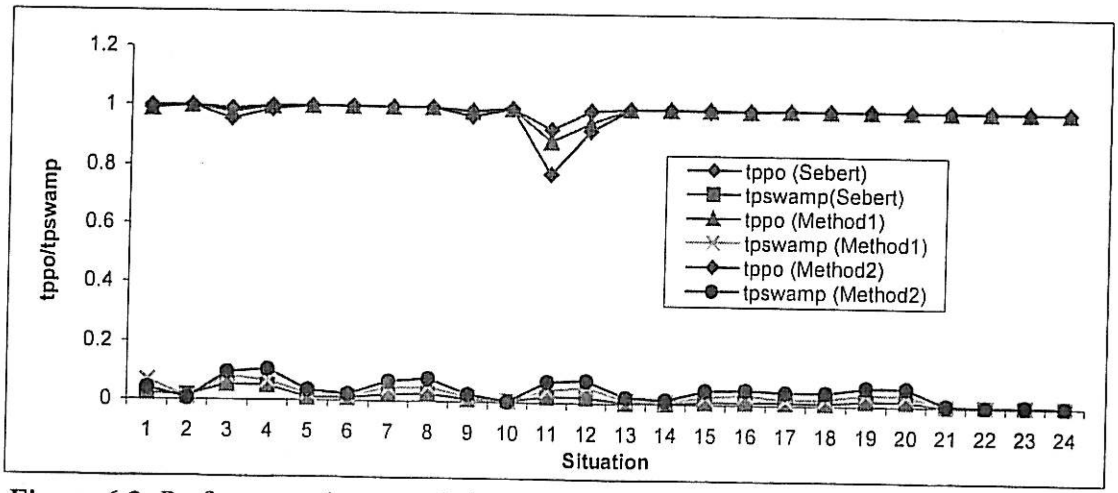


Figure 6.3: Performance between Sebert, Method 1 and Method 2 for $n = 20$ and $p = 6$

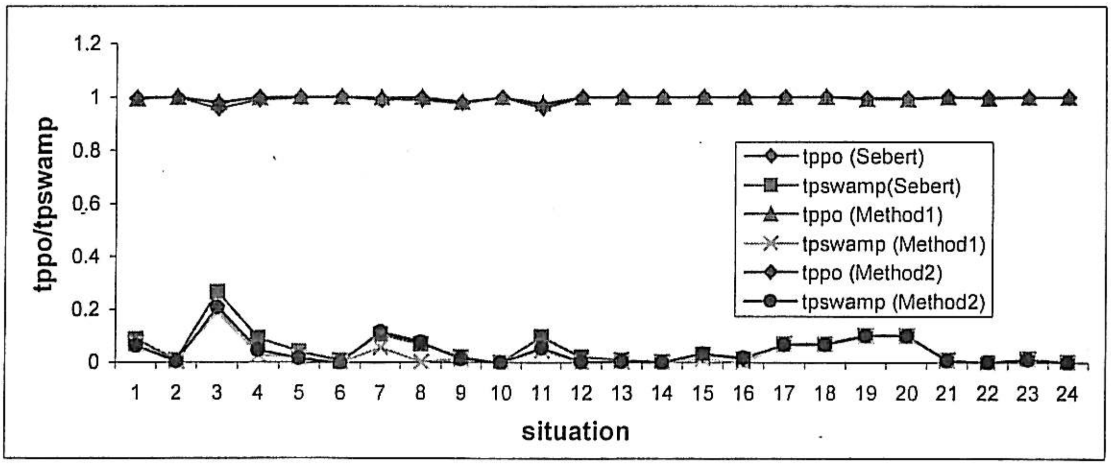


Figure 6.4: Performance between Sebert, Method 1 and Method 2 for $n = 40$ and $p = 1$

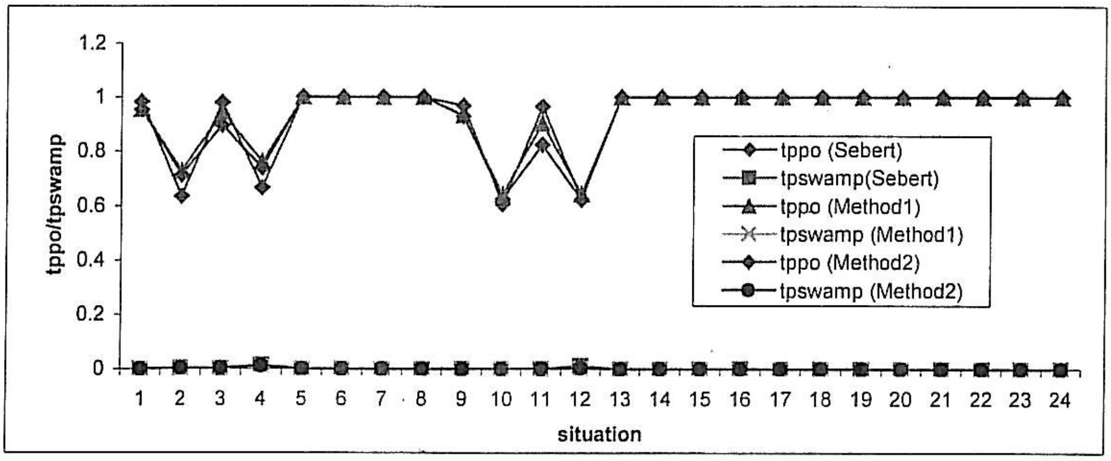


Figure 6.5: Performance between Sebert, Method 1 and Method 2 for $n = 40$ and $p = 2$

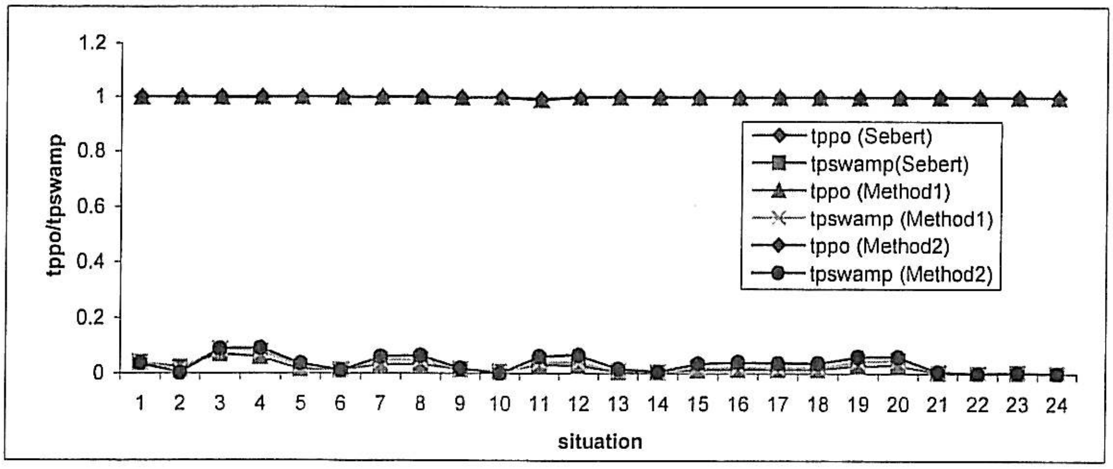


Figure 6.6: Performance between Sebert, Method 1 and Method 2 for $n = 40$ and $p = 6$

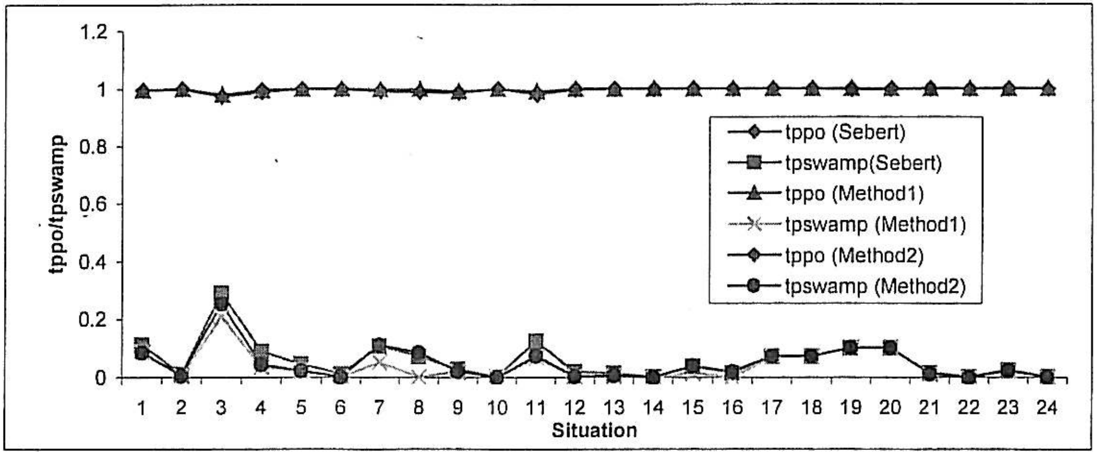


Figure 6.7: Performance between Sebert, Method 1 and Method 2 for $n = 60$ and $p = 1$

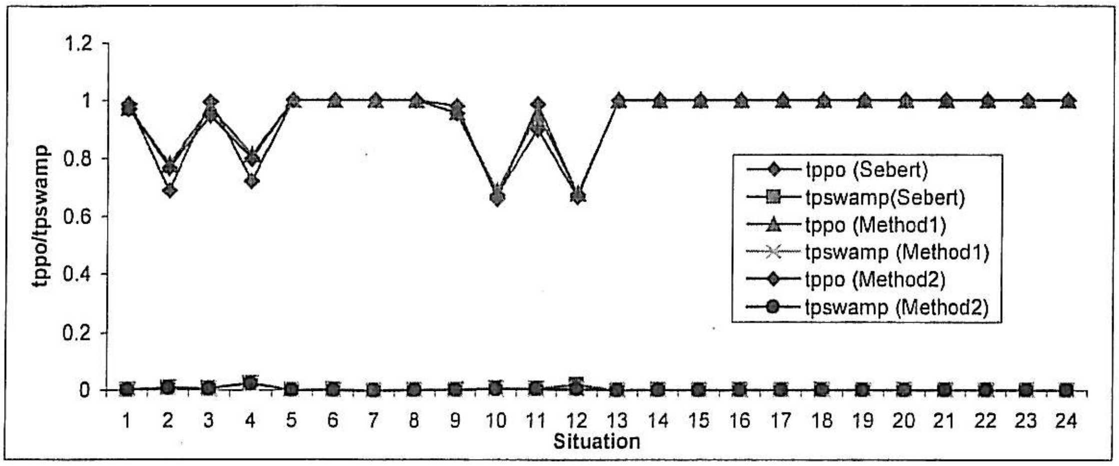


Figure 6.8: Performance between Sebert, Method 1 and Method 2 for $n = 60$ and $p = 2$

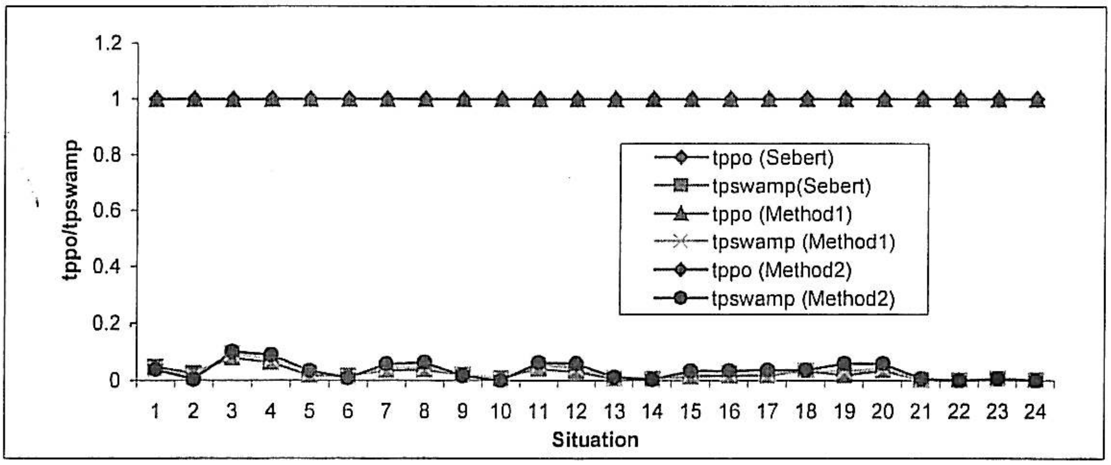


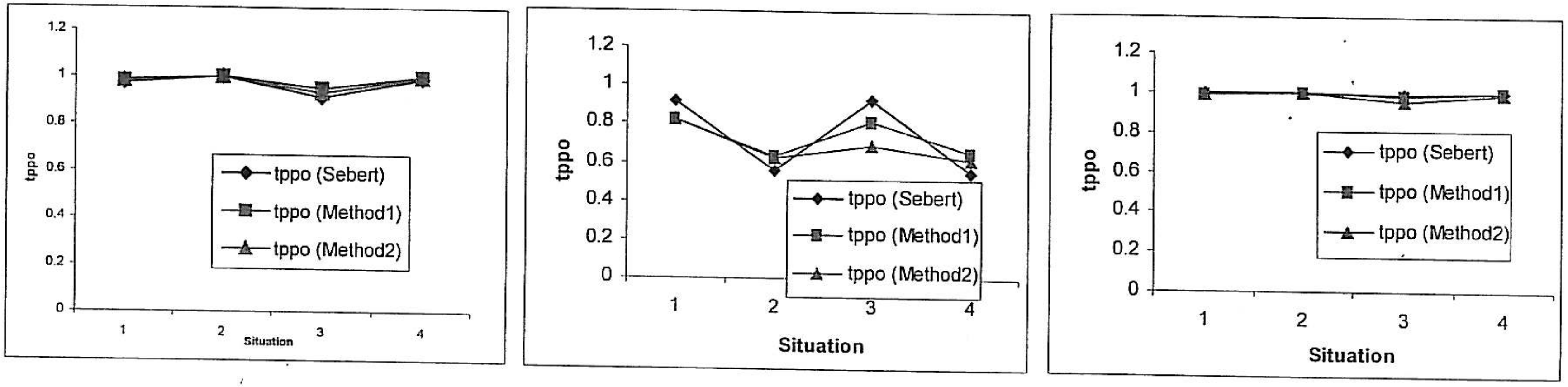
Figure 6.9: Performance between Sebert, Method 1 and Method 2 for $n = 60$ and $p = 6$

From Figure 6.3 where $n = 20$ and $p = 6$, every method are very effective in detecting the presence of outliers except in situations 3, 11 and 12. The swamping probability for the Sebert's method is smaller than Method 1 and Method 2 in almost all scenarios. Besides, Figure 6.4 where $n = 40$ and $p = 1$, shows that all methods are very effective. The detection probabilities are almost one for every situation and method. However, the swamping probability for the Method 2 is smaller than Method 1 and Sebert's method in almost all scenarios.

From Figure 6.5 where $n = 40$ and $p = 2$, the detection probabilities for all methods are high except for situation 2, 4, 10 and 12. The swamping probabilities are also small for every method in each situation. For Figure 6.6 where $n = 40$ and $p = 6$, again all methods are very effective in detecting the presence of outliers. However, the swamping probability for the Sebert's method is smaller than Method 1 and Method 2.

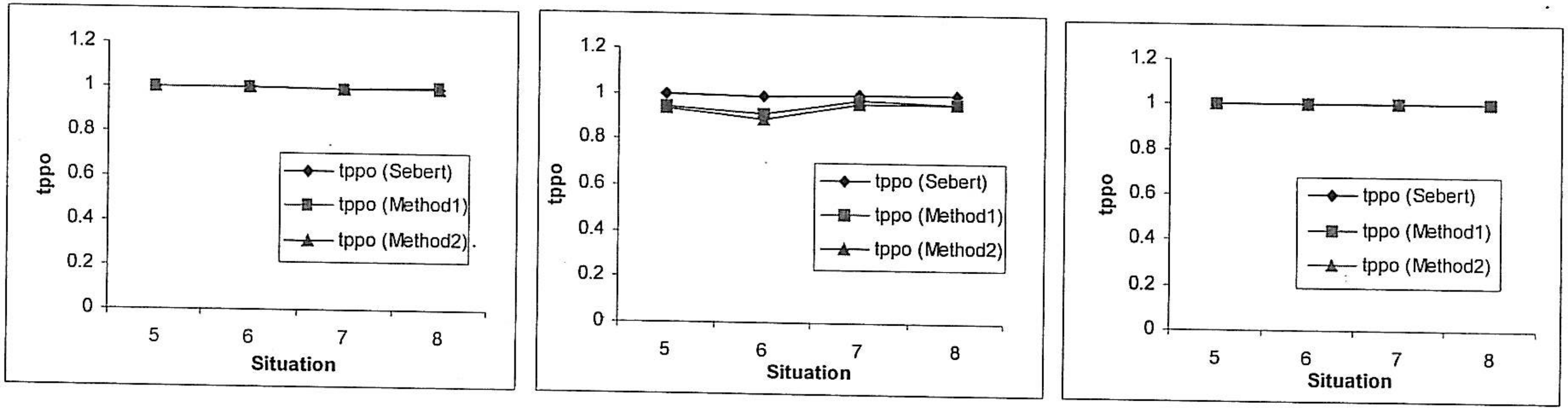
The pattern of graph in Figure 6.7 is the same as the one in Figure 6.4. For $n = 60$ and $p = 1$, all methods are also very effective and the swamping probability for the Sebert's method is smaller than Method 1 and Method 2 in almost scenarios. Further more, from Figure 6.8, where $n = 60$ and $p = 2$, the detection probabilities are also high for every method except for situations 2, 4, 10 and 12 and the swamping probabilities are small for every method. The pattern of the graph is the same as Figure 6.5. Lastly, for Figure 6.9, where $n = 60$ and $p = 6$, every method shows a good performance, but Method 2 and Method 1 is better than Sebert's method because they have small values of swamping probabilities.

Generally, Method 2 has a higher detection probability and a lower swamping probability than Method 1 and Sebert's method when the number of regressor variables $p \leq 2$. Method 2 is also better than Method 1 and Sebert's method when the sample size is bigger but the swamping probability is quite high. Sebert's method is better than Method 1 and Method 2 when the number of regressor variables is high.



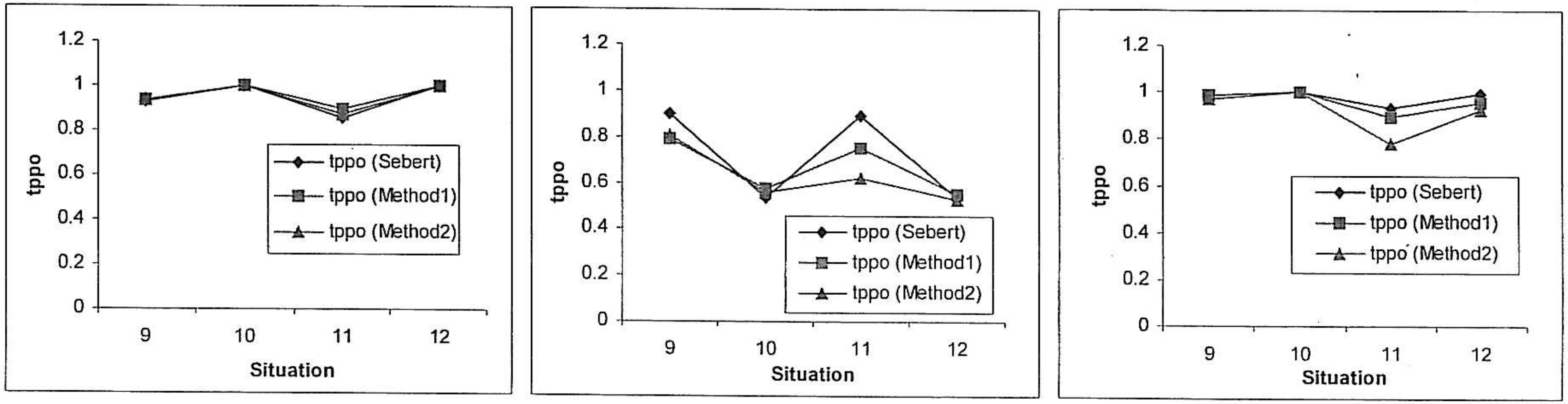
$p = 1$ $p = 2$ $p = 6$

Figure 6.10a: Performance of all methods in scenario 1 for $n = 20$ and all p respectively



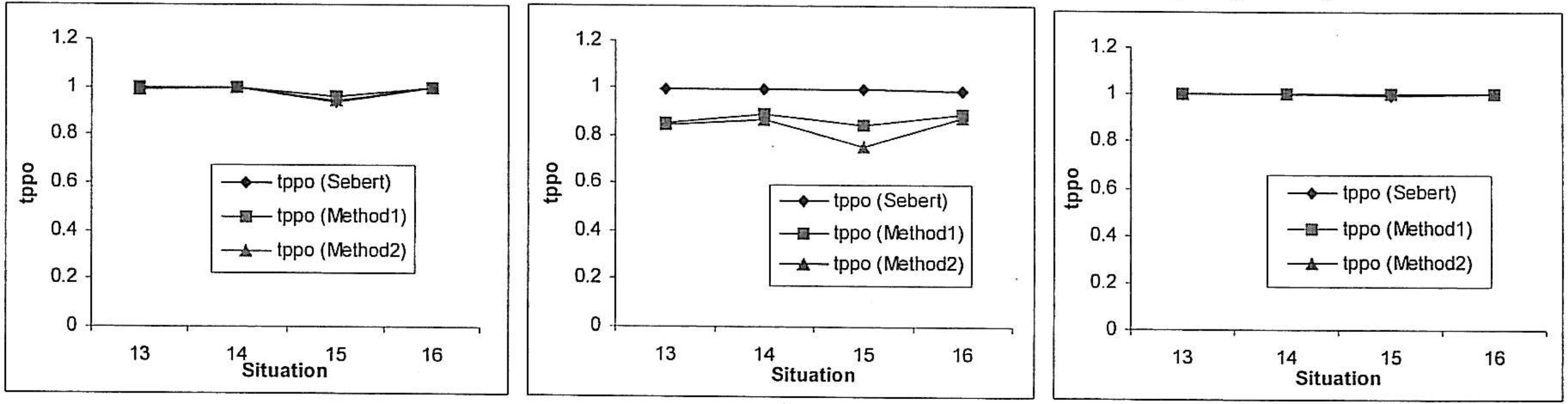
$p = 1$ $p = 2$ $p = 6$

Figure 6.11a: Performance of all methods in scenario 2 for $n = 20$ and all p respectively



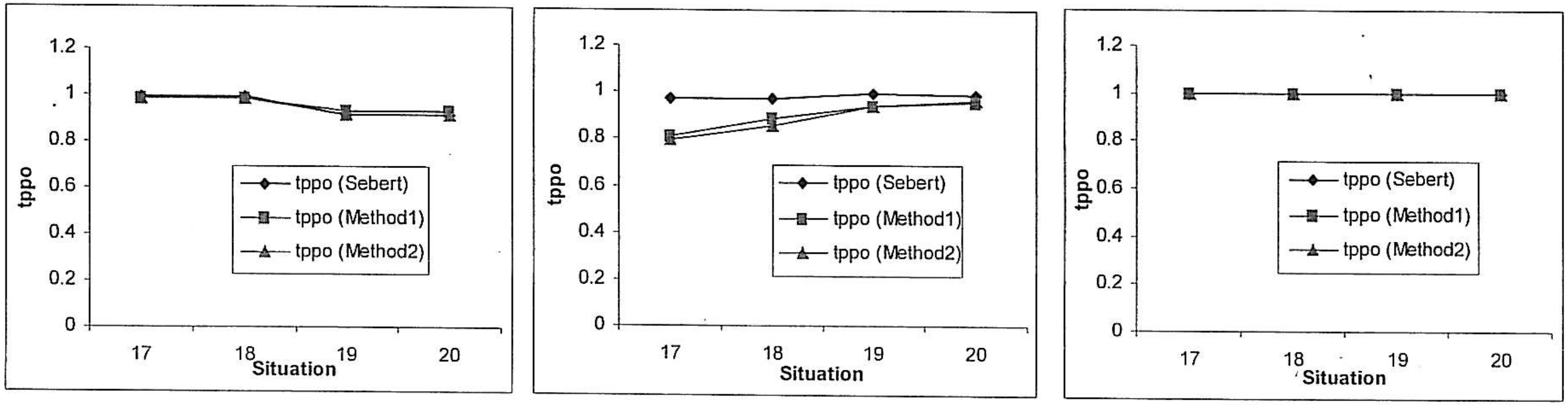
$p = 1$ $p = 2$ $p = 6$

Figure 6.12a: Performance of all methods in scenario 3 for $n = 20$ and all p respectively



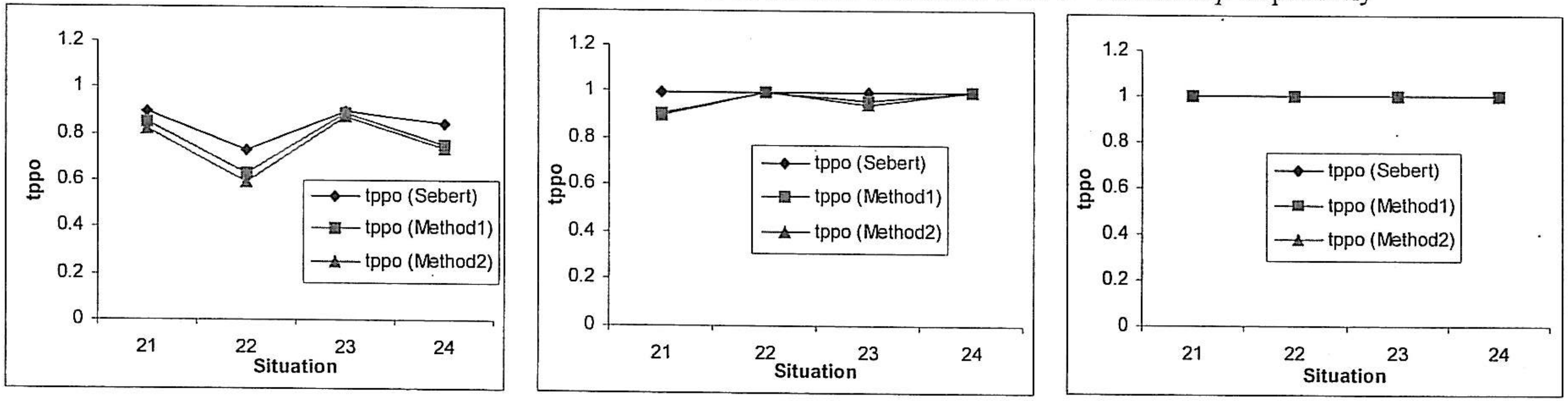
$p = 1$ $p = 2$ $p = 6$

Figure 6.13a: Performance of all methods in scenario 4 for $n = 20$ and all p respectively



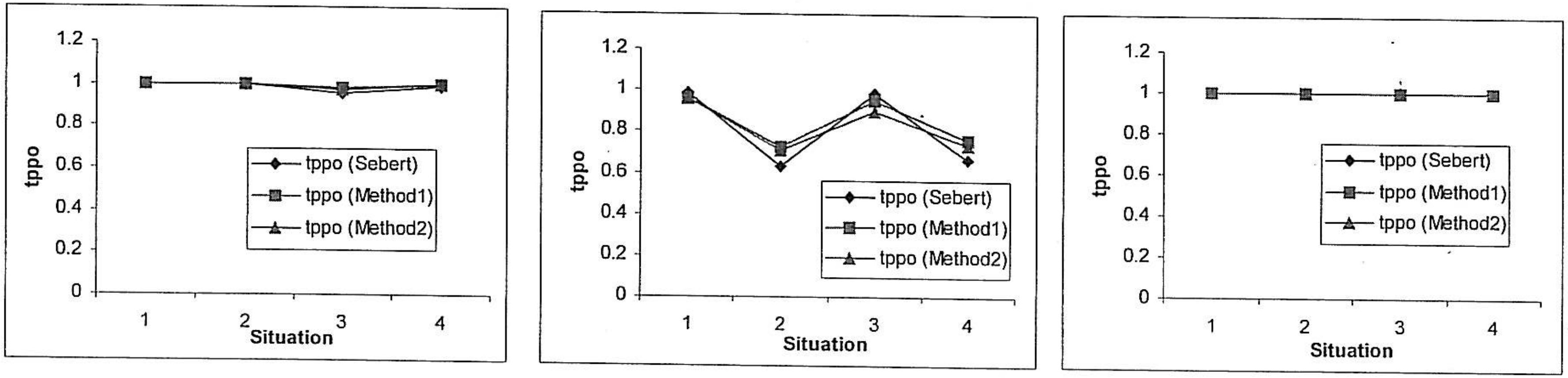
$p = 1$ $p = 2$ $p = 6$

Figure 6.14a: Performance of all methods in scenario 5 for $n = 20$ and all p respectively



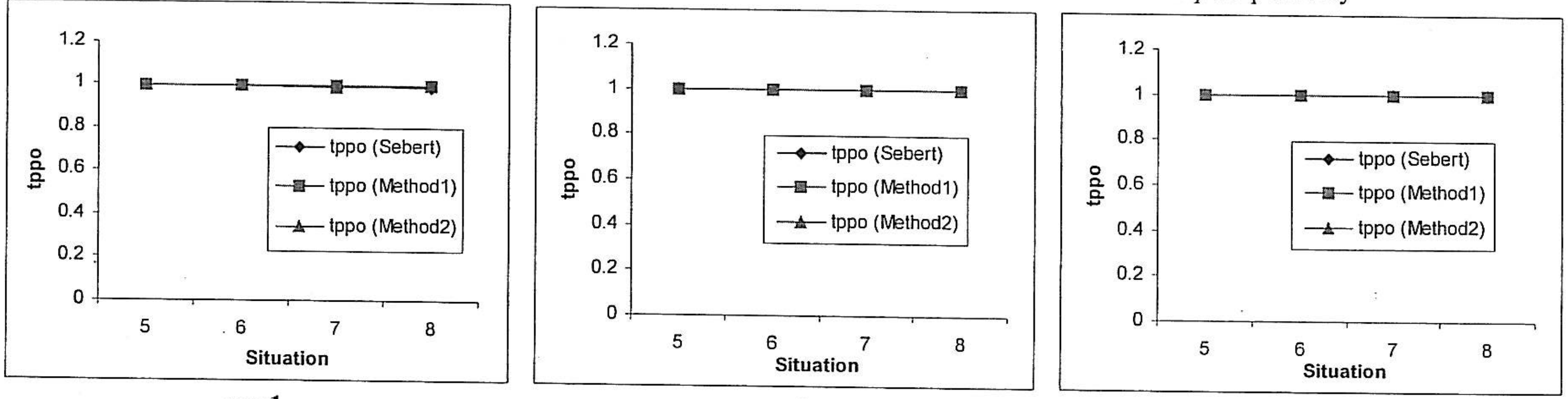
$p = 1$ $p = 2$ $p = 6$

Figure 6.15a: Performance of all methods in scenario 6 for $n = 20$ and all p respectively



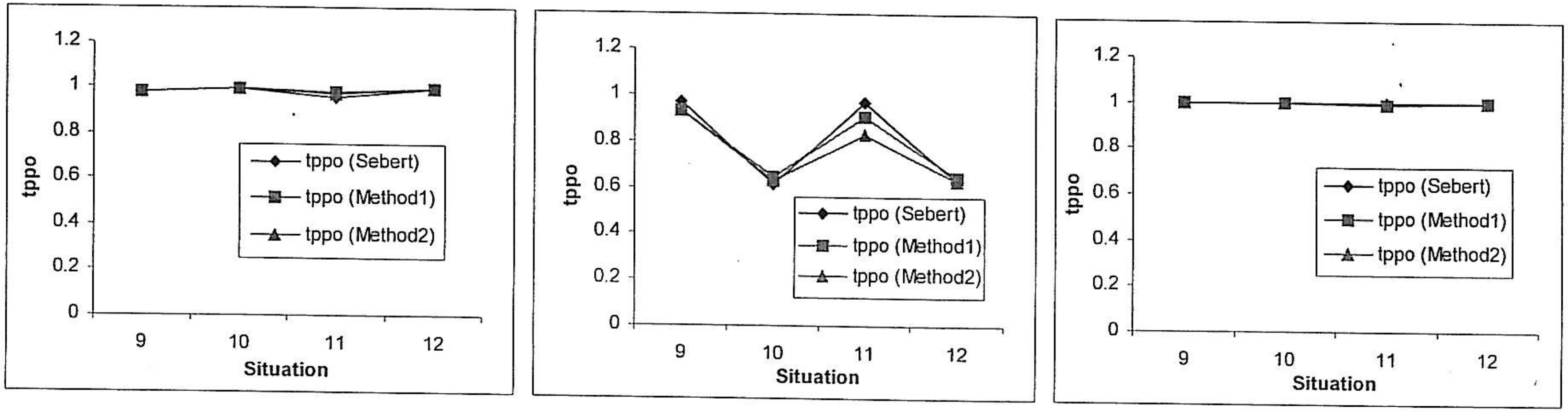
$p = 1$ $p = 2$ $p = 6$

Figure 6.10b: Performance of all methods in scenario 1 for $n = 40$ and all p respectively



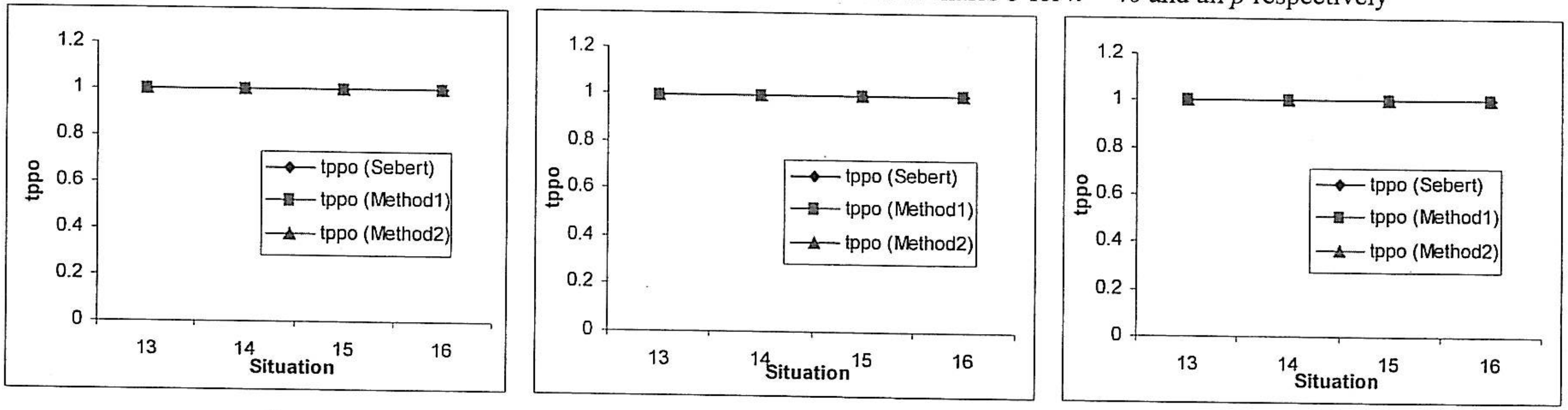
$p = 1$ $p = 2$ $p = 6$

Figure 6.11b: Performance of all methods in scenario 2 for $n = 40$ and all p respectively



$p = 1$ $p = 2$ $p = 6$

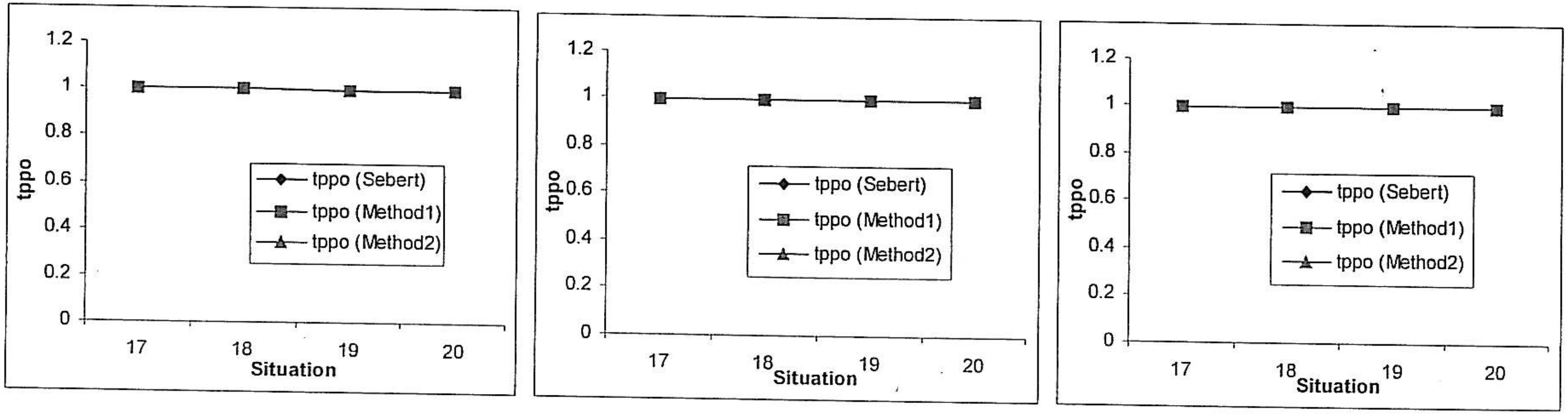
Figure 6.12b: Performance of all methods in scenario 3 for $n = 40$ and all p respectively



$p =$

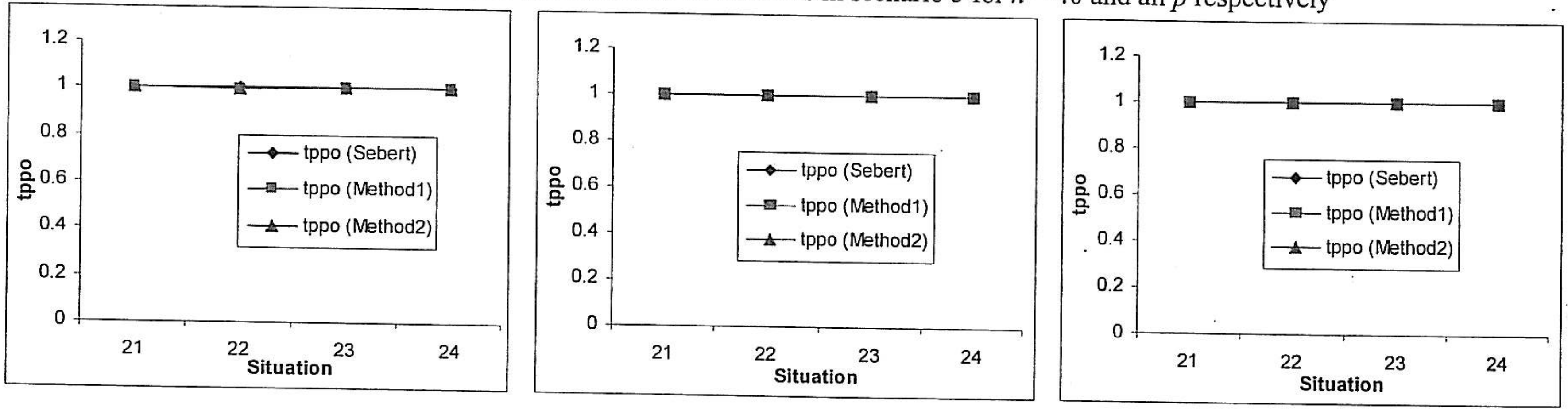
1 $p = 2$ $p = 6$

Figure 6.13b: Performance of all methods in scenario 4 for $n = 40$ and all p respectively



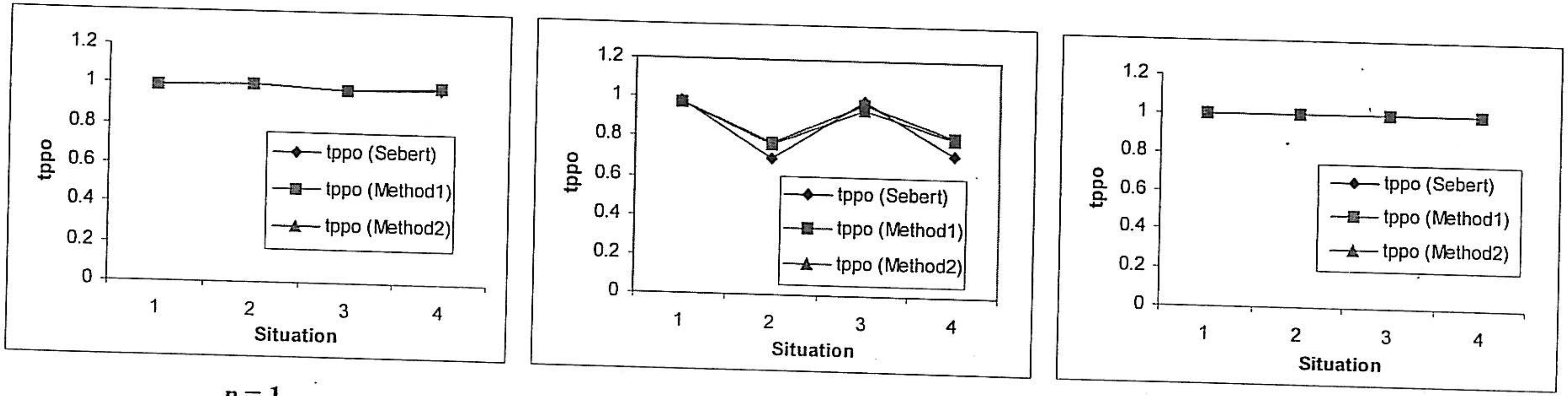
$p = 1$ $p = 2$ $p = 6$

Figure 6.14b: Performance of all methods in scenario 5 for $n = 40$ and all p respectively



$p = 1$ $p = 2$ $p = 6$

Figure 6.15b: Performance of all methods in scenario 6 for $n = 40$ and all p respectively

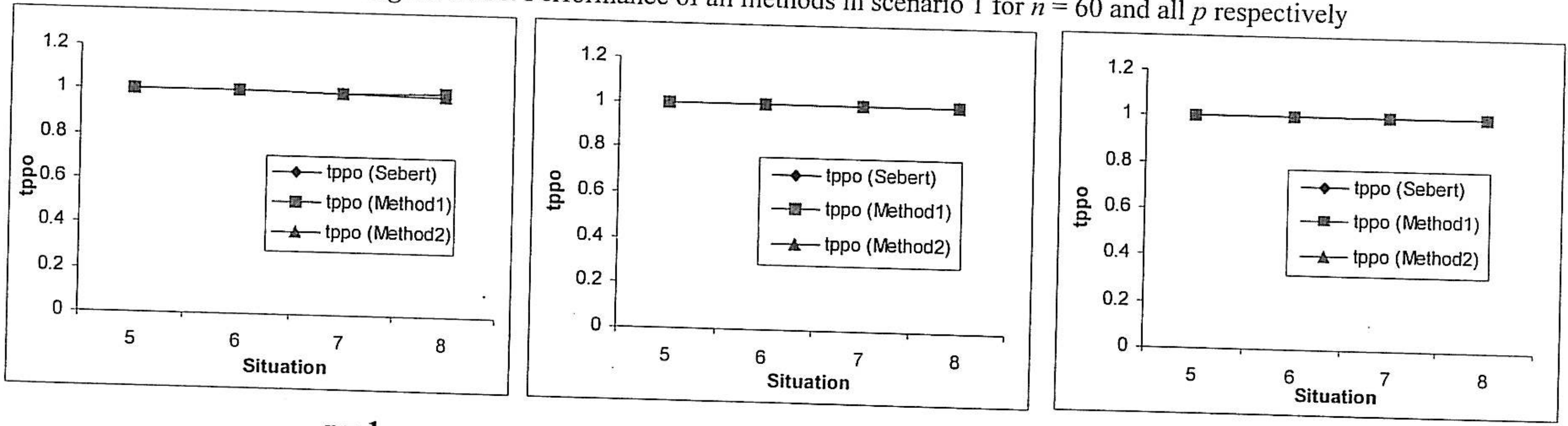


p = 1

p = 2

p = 6

Figure 6.10c: Performance of all methods in scenario 1 for $n = 60$ and all p respectively

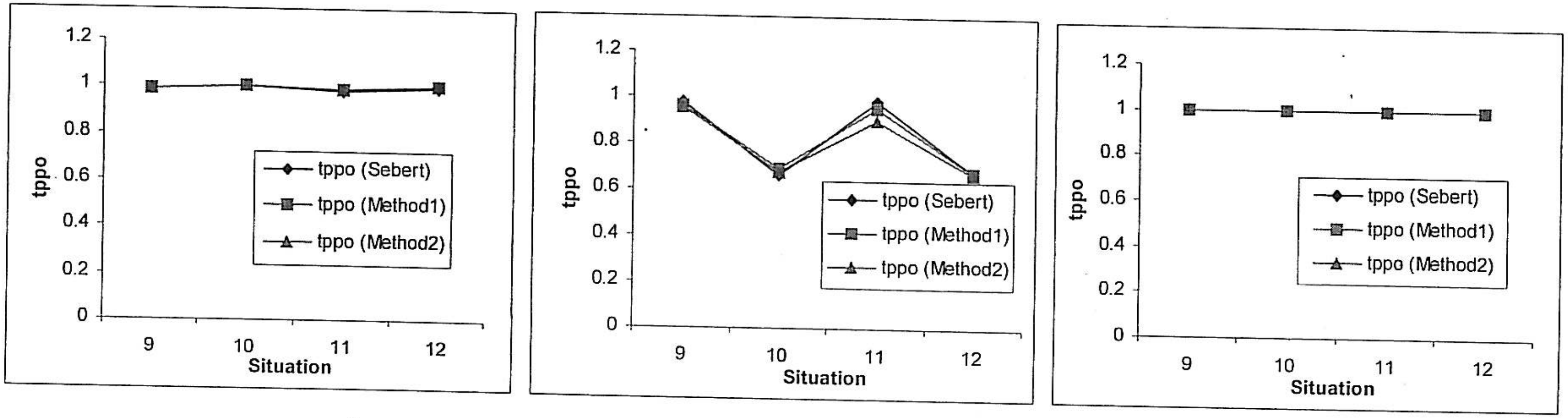


p = 1

p = 2

p = 6

Figure 6.11c: Performance of all methods in scenario 2 for $n = 60$ and all p respectively

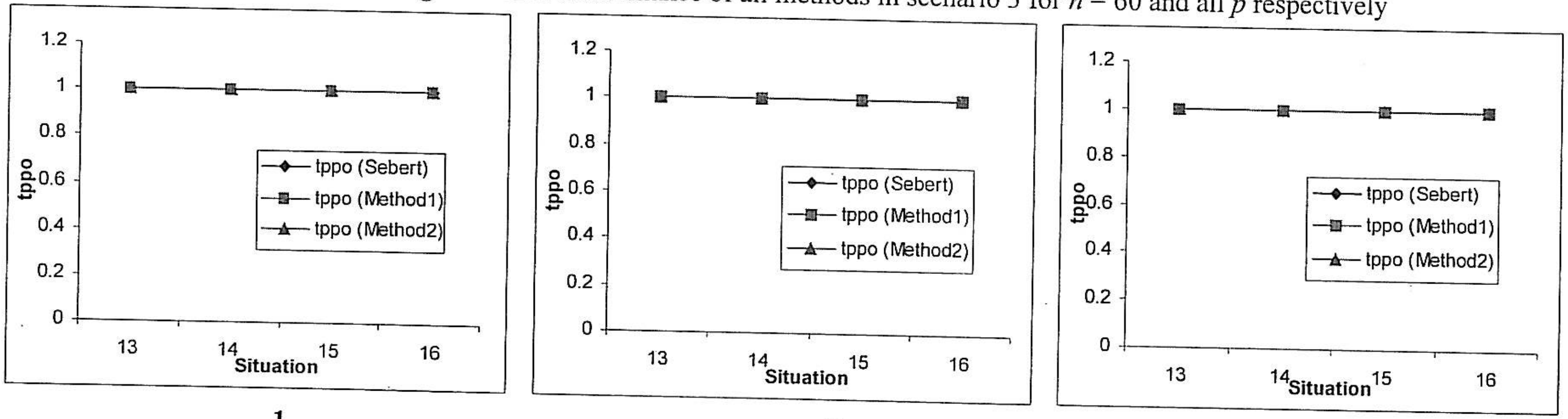


$p = 1$

$p = 2$

$p = 6$

Figure 6.12c: Performance of all methods in scenario 3 for $n = 60$ and all p respectively



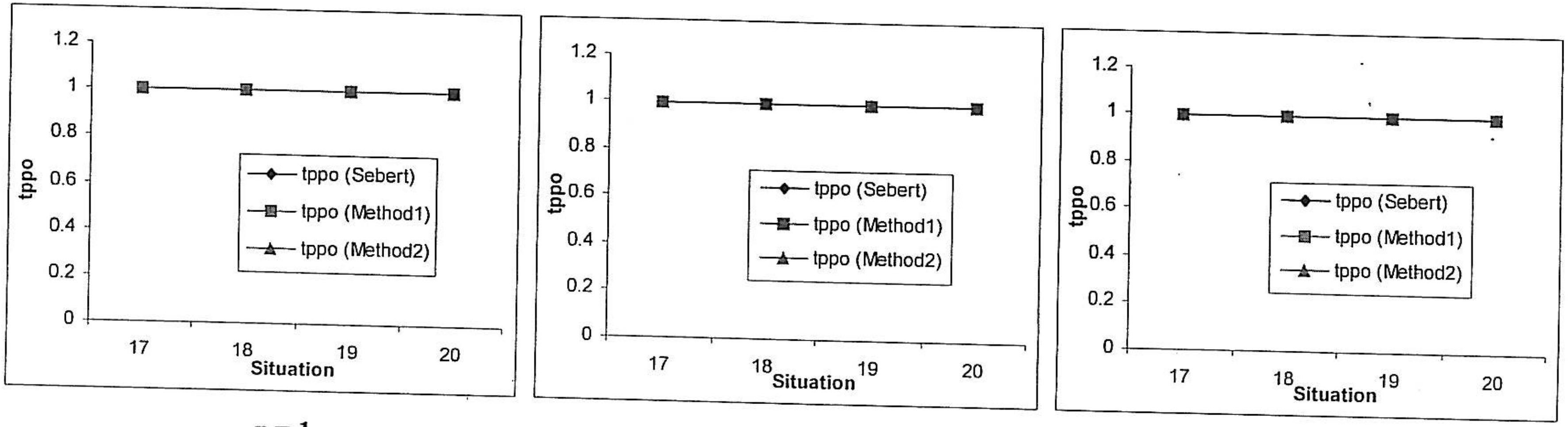
1

$p =$

$p = 2$

$p = 6$

Figure 6.13c: Performance of all methods in scenario 4 for $n = 60$ and all p respectively

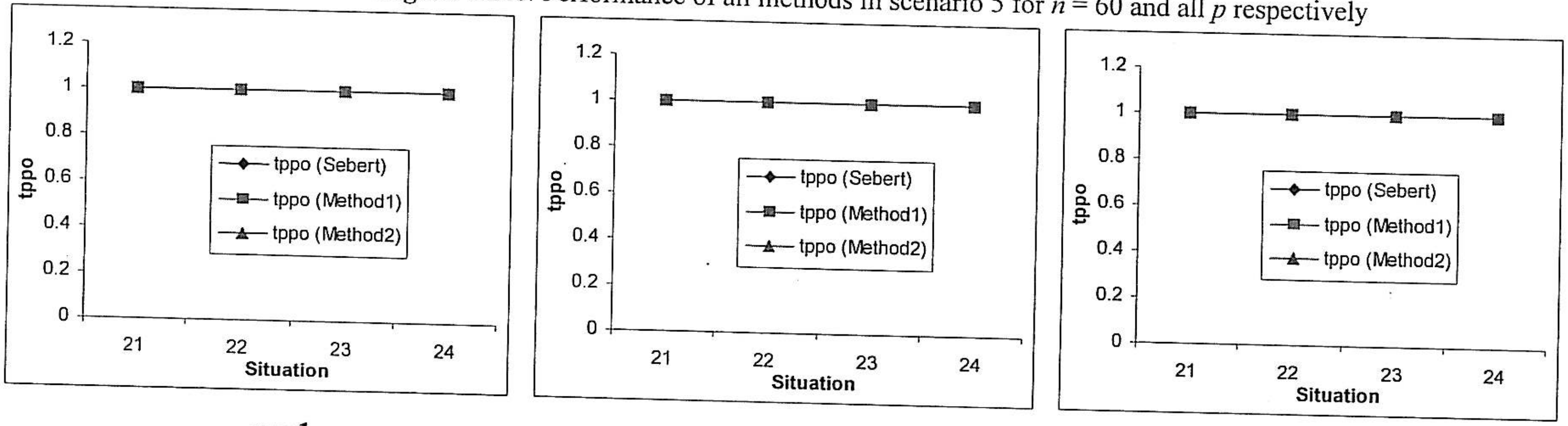


$p = 1$

$p = 2$

$p = 6$

Figure 6.14c: Performance of all methods in scenario 5 for $n = 60$ and all p respectively



$p = 1$

$p = 2$

$p = 6$

Figure 6.15c: Performance of all methods in scenario 6 for $n = 60$ and all p respectively

Table 6.2: Summary of the performances of Sebert's method, Method 1 and Method 2 for different scenarios with $p = 1$

	Scenario 1	**Scenario 2**	**Scenario 3**	**Scenario 4**	**Scenario 5**	**Scenario 6**
$n = 20$	All methods are good but Method 1 is better	All methods have effective performance	All methods are good but Method 1 is better	All methods are good but Method 1 is better	All methods are good but Method 1 is better	Only Sebert's method performed well
$n = 40$	All methods are effective but Method 1 is better	All methods have effective performance	All methods are good	All methods have effective performance	All methods have effective performance	All methods have effective performance
$n = 60$	All methods are effective but Method 1 is better	All methods are effective but Method 1 is better	All methods are effective but Method 1 is better	All methods are effective but Method 1 is better	All methods have effective performance	All methods have very effective performance

Table 6.3: Summary of the performances of Sebert's method, Method 1 and Method 2 for different scenarios with $p = 2$

	Scenario 1	**Scenario 2**	**Scenario 3**	**Scenario 4**	**Scenario 5**	**Scenario 6**
$n = 20$	Only Sebert's method performed well	All methods are good but Method 1 is better	Only Sebert's method performed well	Sebert's method has effective performance, Method 1 is good and Method 2 is worse	All methods are good but Method 1 is better	All methods are good but Method 1 is better
$n = 40$	All methods not perform well but Method 1 is better	All methods have very effective performance	All methods not perform well but Sebert's method is better	All methods have very effective performance	All methods have very effective performance	All methods have very effective performance
$n = 60$	All methods not perform well but Sebert's method is better	All methods have very effective performance	All methods not perform well but Sebert's method is better	All methods have very effective performance	All methods have very effective performance	All methods have very effective performance

Table 6.4: Summary of the performances of Sebert's method, Method 1 and Method 2 for different scenarios with $p = 6$

	Scenario 1	**Scenario 2**	**Scenario 3**	**Scenario 4**	**Scenario 5**	**Scenario 6**
$n = 20$	All methods have effective performance but sebert's method is better	All methods have very effective performance	Only Sebert's method performed well effective	All methods have effective performance	All methods have very effective performance	All methods have very effective performance
$n = 40$	All methods have effective performance	All methods have very effective performance	All methods have effective performance	All methods have very effective performance	All methods have very effective performance	All methods have very effective performance
$n = 60$	All methods have effective performance	All methods have effective performance	All methods have effective performance	All methods have very effective performance	All methods have very effective performance	All methods have very effective performance

Table 6.5: The rate of method's performance according to the tppo value

Method's performance	**tppo value**
Very effective	1
Effective	0.9-0.9999
Good	0.7-0.8999
Worse	<0.7

The following provides the general findings and conclusions for every scenario given by Figures 6.10 – 6.15 and Tables 6.2 – 6.4. It appears that all methods generally performed effectively in detecting single group outliers in the *xy*-space (Scenario 2) and the *x*-space (Scenario 5). This pattern holds in every sample size and number of regressors. For a group of outliers in *xy*-space (Scenario 1), the performances of Sebert's method and Method 1 were effective when $p \leq 2$. All methods was very effective when $p > 2$.

For two groups of outliers in *xy*-space (Scenario 3), Sebert's method performed well and approximately effective in all sample size and number of regressors. For the other two groups of outliers in *xy*-space (Scenario 4), all methods were effective when the sample size is large that is for $n = 40$ and $n = 60$ in this case. For the two groups of outliers, where one is an *x*-space outlier and the other is an *xy*-space outlier (Scenario 6), all methods were also effective when the sample size is large. Generally, Method 2 is approximately effective for every scenario.

CHAPTER 7

SUMMARY, CONCLUSION AND SUGGESTIONS

7.1 Introduction

This chapter summarizes the materials presented in the previous six chapters and discusses in further detail some of the results and findings. Some conclusions and suggestions are presented based on the results and findings given.

7.2 Summary and Conclusions

Generally, this research provided a review on the multiple outlier problems in linear regression and the limited uses of Least Squares (LS) fit to overcome these problems. As pointed out in Chapter 2, researchers have suggested numerous strategies and procedures to solve the multiple outlier identification problems.

The outliers identification procedures based on clustering algorithm proposed by Sebert et al. (1998) was chosen to be discussed and it showed to perform well on the classical multiple outliers data set and simulated random data. This method used the single linkage clustering algorithm with the Euclidean distances to cluster the points in the plots of standard predicted versus residuals values. The predicted and residuals values are obtained from an ordinary least squares fit of the data. The Mojena's stopping rule was finally used to choose the outliers. The discussion on this method was presented in Chapter 4.

This research also studied the influence of the least median of squares (LMS) and the least trimmed of squares (LTS) fit as opposed to the least squares (LS) used in Sebert et al. (1998) and characterized the performance of the new procedures, Method 1 and Method 2 as pointed out in Chapter 5. These two robust estimators were chosen because of a high breakdown, efficient and bounded influence. The new methods also performed well on the classical data set and provided a better result in some data.

The Monte Carlo simulation presented in Chapter 3 compared the performance of the procedures proposed by Sebert et al. (1998) and the modifications by Method 1 and Method 2. The comparison is discussed in Chapter 6 and the simulation was done using S-PLUS 2000 statistical package. Generally, Method 2 has a higher detection probability and a lower swamping probability than Method 1 and Sebert's method when the number of regressor variables $p \leq 2$. Method 2 is also better than Method 1 and Sebert's method when the sample size is bigger but the swamping probability is quite high. Sebert's method is better than Method 1 and Method 2 when the number of regressor variables is high. Method 2 is also approximately effective for every scenario. All methods generally performed effectively only in detecting single group outliers in the *xy*-space (Scenario 2) and the *x*-space (Scenario 5). All methods also have problems in detecting outliers in scenario 1 and 3 when the number of regressor variable is 2.

7.3 Suggestion

The following provides several suggestions and recommendations for future research

- Improve the performance of Sebert's method, Method 1 and Method 2 by other robust fit such as Least Trimmed Sum of Absolute Deviations (LTA) and generalized M (GM) robust fit.
- Use other stopping rules and compare the performances
- Compare Sebert's method with other multiple outlier detection procedures, for example, the outlier nomination method based on multihalver by Fernholz et al. (2004).
- Improve the quality of the simulation results by adding new outliers scenario.
- Study the outliers scenario used in the classical data
- Study the reason why all methods have problem in detecting outliers in scenario 1 and 3 when the number of regressor variable is equal to 2.

REFERENCES

Agullo, J. (2001). New Algorithms for Computing the Least Trimmed Squares Regression Estimator. *Computational Statistics and Data Analysis 36.* 425-439.

Aldenderfer M. S. and Blashfield R. K. (1984). *Cluster Analysis.* USA: Sage Publications.

Atkinson, A.C. (1986). Comment on 'Influential Observations, High Leverage Points, and Outliers in Linear regression'. *Statistical Science 1.* 397-402.

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

Barnett, V. and Lewis, T. (1994). *Outliers in Statistical Data. 3rd Edition.* England: John Wiley and Sons.

Billor, N., Hadi, A.S. and Velleman, P.F. (2000). BACON: Blocked Adaptive Computationally Efficient Outlier Nominators. *Computational Statistics and Data Analysis 34.* 279-298.

Blashfield, R.K. and Morey, L.C. (1980). A Comparison of Four Clustering Methods using MMPI Monte Carlo Data. *Applied Psychological Measurement 4.* 57-64.

Bradu, D. and Hawkins, D.M. (1995). An Anscombe Type Robust Regression Statistic. *Computational Statistics and Data Analysis 20.* 355-386.

Brant, R. (1986). Comment on 'Influential Observations, High Leverage Points, and Outliers in Linear regression'. *Statistical Science 1.* 405-407.

Chatterjee, S. and Hadi, A. S. (1986). Influential Observations, High Leverage Points, and Outliers in Linear regression. *Statistical Science 3.* 379-416.

Coakley, C.W. and Hettmansperger, T.P. (1993). A Bounded Influence, High Breakdown, Efficient Regression Estimator. *Journal of the American Statistical Association 88.* 872-880.

Data Analysis Products Division. (1999). *S-PLUS 2000 User's Guide.* USA: Mathsoft.

Data Analysis Products Division. (1999). *S-PLUS 2000 Programmer's Guide.* USA: Mathsoft.

Everitt, B.S. (1993). *Cluster Analysis.* 3rd edition. Halsted Press.

Fernholz, L. T., Morgenthaler, S. and Tukey, J.W. (2004). An Outlier Nomination

Method Based on Multihalver. *Journal of statistical planning and Inference 122.* 125-139.

Gray, J.B. and Ling, R.F. (1984). K-Clustering as a Detection Tool for influential Subsets in Regression. *Technometrics 26.* 305-330.

Hadi, A.S. (1992). A New Measure of Overall Potential Influence in Linear Regression. *Computational Statistics and Data Analysis 14.* 1-27.

Hadi, A.S. (1992b). Identifying Multiple Outliers in Multivariate Data. *Journal of Royal Statistical Society 54.* 761-771.

Hadi, A.S. (1994). A Modification of a method for the Detection of Outliers in Multivariate Samples. *Journal of Royal Statistical Society 56.* 393-396.

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

Hardy, A. (1996). On the Number of Clusters. *Computational Statistics and Data Analysis 23.* 83-96.

Hartingan, J.A (1975). *Clustering Algorithm.* New York: Wiley.

Hawkins, D.M. and Olive, D. (1999). Improved Feasible Solution Algorithms for High Breakdown Estimation. *Computational Statistics and Data Analysis 30.* 1-11.

Hawkins, D.M. and Olive, D. (1999b). Applications and Algorithms for Least Trimmed Sum of Absolute Deviations Regression. *Computational Statistics and Data Analysis 32.* 119-134.

Hoaglin, D.C., and Kempthorne, P.J.(1986). Comment on 'Influential Observations, High Leverage Points, and Outliers in Linear regression'. *Statistical Science 1.* 408-412.

Hoeting, J., Raftery, A.E., and Madigan, D. (1996). A Method for Simultaneous Variable Selection and Outlier Identification in Linear Regression. *Computational Statistics and Data Analysis 22.* 251-270.

Huber, P.J. (1981). *Robust Statistics.* New York: John Wiley.

Johnson, R.A. and Wichern, D.W. (1982). *Applied Multivariate statistical Analysis.* 3rd Edition. Prentice Hall.

Justel, A. and Pena, D. (2001). Bayesian unmasking in Linear Model. *Computational*

Statistics and Data Analysis 36. 69-84.

Kaufmann, L. and Rousseouw, P.J. (1990). *Finding Groups in Data.* John Wiley & Sons. New York

Kleinbaunm, Kupper, Muller. (1998). *Applied Regression Analysis and other Multivariate Methods.* USA: PWS-Kent Publishing Company.

Kianifard, F. and Swallow, W. (1990). A Monte Carlo Comparison for five Procedures for identifying Outliers in Linear Regression. *Communication in Statistics, Part A-Theory and Method 19.* 1913-1938.

Kosinski, A.S. (1999). A Procedure for the Detection of Multivariate Outliers. *Computational Statistics and Data Analysis 29.* 145-161.

Krasker, W.S. and Welsh, R.E. (1982). Efficient Bounded-Influence Regression Estimation. *Journal of American Statistical Association* 77. 595-604.

Luceno, A. (1998). Multiple Outliers Detection through Reweighted Least Deviance. *Computational Statistics and Data Analysis 26.* 313-326.

Milligan, G.W. and Cooper, M.C. (1985). An examination of Procedures for Determining the Number of Clusters in a Data Set. *Psychometrika 50.* 159-179.

Mallows, C.L. (1975). *On Some Topics in Robustness.* Unpublished Memorendum, Bell Telephone Laboratories. Murray Hill, NJ.

McCullagh, P. and Neldeer, J.A. (1989). *Generalized Linear Model.* 2nd ed. New Jersey: Chapman and Hall.

Minowski, J.W. (1999). *Multiple Outliers in Linear Regression: Advances in Detection Methods, Robust estimation, and Variable Selection.* Arizona State University. Unpublised Dissertation.

Marasinghe, M.G. (1985). A Multistage Procedure for Detecting Several Outliers in Linear Regression. *Technometrics 27.* 395-399.

Marchette, D.J., and Solka, J.L. (2003). Using Data Images for Outlier Detection. *Computational Statistics and Data Analysis 43.* 541-552.

Mojena, R. (1977). Hierarchical Grouping Methods and Stopping Rules: An Evaluation. *The Computer Journal 20.* 359-363.

Pena, D., and Yohai, V. (1999). A Fast Procedure for Outlier Diagnostics in Large Regression Problems. *Journal of the American Statistical Association 94.*

434-445.

Robiah Adnan, Mohd Nor Mohamad and Halim Setan. (1999). *Using Cluster Analysis and Least Trimmed Squares to Detect Outliers (Laporan Teknik).* Skudai: Jabatan Matematik UTM.

Robiah Adnan. (2001). *Identifying Multiple Outliers in Linear Regression: Robust Fit, Clustering and Inner-Outer Fences.* Skudai: UTM. PHD Thesis.

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

Rousseeuw, P.J. and Leroy A.M. (1987). *Robust Regression and Outlier Detection.* Canada: John Wiley & Sons.

Rousseeuw, P.J. and Yohai, V. (1984). Robust Regression by means S-estimators. *Robust and Nonlinear Time Series Analysis.* Eds. J. Franke, W. Hardle, and D. Martin. Heidelberg, Germany: Springer-Verlag.

Rousseeuw, P.J., and Zomeren, B.C.V. (1990). Unmasking Multivariate Outliers and Leverage Points. *Journal of the American Statistical Association 85.* 633-639.

Ruppert, D, and Simpson, D.G. (1990). Comment on 'Unmasking Multivariate Outliers and Leverage Points'. *Journal of the American Statistical Association 85.* 645-646.

Sebert, D.M., Montgomery, D.C., and Rollier, D.A. (1998). A Clustering Algorithm for Identifying Multiple Outliers in Linear Regression. *Computational Statistics and Data Analysis 27.* 461-484.

Sebert, D.M. (1996). *Identifying Multiple Outliers and Influence Subsets: A Clustering Approach.* Arizona state University. Unpublished Disertation.

Simpson, J.R. (1995). Unpublished dissertation. Arizona State University.

Simpson, J.R. and Montgomery, D.C. (1996). A Biased-Robust Regression Technique for the Combined Outlier-Multicollinearity Problem. *Journal Statistics Computational and Simulation 56.* 1-22.

Simpson, J.R. and Montgomery, D.C. (1998). A Performance-Based Assessment of Robust Regression Methods. *Communication in statistics- Simulation and Computation 27.* 1031-1049.

Siti Zanariah Satari. (2003). *Multiple Regression Analysis.* Skudai : UTM.

Projek Sarjana Muda.

Velleman, P.F. (1986). Comment on 'Influential Observations, High Leverage Points, and Outliers in Linear regression'. *Statistical Science 1.* 412-413.

Weisberg, S. (1986). Comment on 'Influential Observations, High Leverage Points, and Outliers in Linear regression'. *Statistical Science 1.* 414-415.

Welsch, R.E. (1986). Comment on 'Influential Observations, High Leverage Points, and Outliers in Linear regression'. *Statistical Science 1.* 403-405.

Wisnowski, J.W., Montgomery, D.C., and Simpson, J.R. (2001). A Comparative Analysis of Multiple Outlier Detection Procedures in the Linear Regression Model. *Computational Statistics and Data Analysis 36.* 351-382.

Yohai, V.J., et al. (1991). A Procedure for Robust Estimation and Inference in Linear Regression. Direction in Robust Statistics and Diagnostics, Part II. Heidenlberg, Germany: Springer-Verlag.

You. J., (1999). A Monte Carlo Comparison of Several High Breakdown and efficient Estimators. *Computational Statistics and Data Analysis 30.* 205-219.

Zani, S., Riani, M.,and Corbellani, A. (1999). Robust Bivariate Boxplots and Multiple Outlier Detection. *Computational Statistics and Data Analysis 28.* 257-270.