Robust Regression

Catherine Stuart

16th April, 2011

Abstract

An introduction to robustness in statistics, with emphasis on its relevance to regression analysis. The weaknesses of the least squares estimator are highlighted, and the idea of error in data refined. Properties such as breakdown, efficiency and equivariance are discussed and, through consideration of M, S and MM-estimators in relation to these properties, the progressive nature of robust estimator development is demonstrated. Finally, some (often overlooked) limitations of favoured 'high breakdown, high efficiency' estimators are considered, and based on results obtained from simulations, guidance is given and cautions proposed regarding the application of such techniques.

This piece of work is a result of my own work except where it forms an assessment based on group project work. In the case of a group project, the work has been prepared in collaboration with other members of the group. Material from the work of others not involved in the project has been acknowledged and quotations and paraphrases suitably indicated.

Contents

Chapter 1

Introduction

During the past five decades, regression methods have become increasingly sophisticated as computers have become able to process vast quantities of data and perform extremely complex calculations (Takeaki and Hiroshi 2004: xi). In particular, 'robust' regression techniques requiring lengthy iterative algorithms to find the parameter estimate $\hat{\boldsymbol{\beta}}$ have become feasible. This chapter begins by introducung the idea of robustness in statistics, and then introduces regression analysis. Subsequently, by discussing the limitations of the traditional regression technique, the motivation to develop robust regression techniques is explained.

1.1 Robustness

When making statistical inferences one only partly uses observations; modelling assumptions are equally important (Huber and Ronchetti 2009: 1). Often one must assume that small deviations will not significantly affect the conclusions drawn from the data. During the 20th Century, however, statisticians have become acutely aware that common statistical procedures, particularly those based on a normality assumption, are very sensitive even to slight deviations from the assumed model (Huber and Ronchetti 2009: 1-2). Robust procedures are a reaction to this.

Huber (1996: 1) describes 'robustness' as, 'insensitivity to small deviations from the assumptions made'. Huber and Ronchetti (2009: 5), reflecting this idea, highlight three properties that a 'good' robust procedure should have:

1. When the assumed model is correct, the procedure should give results with a fairly small sampling variance.

2. Small deviations from the assumed model should only have a minimal effect on performance.

3. Larger deviations from the assumed model 'do not cause catastrophe'.

These criteria mean that one does not have to trust entirely any modelling assumptions. Huber and Ronchetti emphasise that such a robust statistic has to be based on compromise: the sampling variance at the assumed model is increased, in order to protect against deviations from the model.

Andersen (2008: 3) specifies two parts of the robustness of a statistic to consider. The first, 'robustness of validity', represents its robustness to unusual observations. When an estimator has robustness of validity, a small change to the data will not substantially change the statistic.

Notably, *gross* errors in a small fraction of the observations in a sample represent a 'small' change in this context, so a primary objective of robust procedures should be protection against gross errors (Huber and Ronchetti 2009: 5). The second type of robustness is called 'robustness of efficiency'. Robustness of efficiency means that even if the observations don't match the distributional assumptions, the standard error of the statistic will hardly be affected.

So what causes these deviations, and how common are they? Hampel (1973: 88) gives three main sources of deviation from the assumed model in data:

- Rounding and grouping of data.
- The occurrence of gross errors.
- The assumed model having been conceived only as an approximation, via the central limit theorem for example.

This report will focus on the second source. Gross errors can occur as a result of mistakes in the measuring of observations, possibly due to equipment being improperly calibrated or malfunctioning, incorrectly placed decimal points, errors in copying or transmission or simply, as Hampel says, 'something went wrong' (Hampel 1973: 88; Rice 1995: 362; Rousseeuw and Leroy 1987: 25). Resultant errors cannot always be detected, so even if one initially 'cleans up' a data set it is unlikely to fit a model perfectly; robust procedures should be employed (Hampel 1973: 88; Hogg 1979: 108; Huber 1996: 3). If a normal model is assumed for data, errors can also result from the presence of a heaver-tailed distribution (Hogg 1979: 108).

The frequency of errors in data obviously depends on its quality, but Rice (1995: 362) emphasises the prevalence of unusual observations in even the most careful studies. Hampel (1973: 88) states that, '5-10% wrong values in a data set seem to be the rule rather than the exception.' He noted that even high quality samples in astronomy and geodesy consisting of thousands of observations – which should be prime examples of normally distributed data – are mildly, but definitely heavier-tailed. In fact, Huber (1996: 2) gives the following error law, perhaps a little ironically, to model the behaviour of 'good' data from the physical sciences:

$$F\left(x\right) = (1-\varepsilon)\Phi\left(x\right) + \varepsilon\Phi\left(\frac{x}{3}\right).$$

This probability distribution gives the data a standard normal distribution but contaminates it by a small fraction, ε, with a normal distribution for the data scaled by $\frac{1}{3}$, thereby lengthening the distribution's tails.

As Geary (1947: 241) once suggested, perhaps a safe attitude to statistical inferences could be maintained by printing on every statistics textbook: 'Normality is a myth; there never was, and never will be, a normal distribution'.

1.2 Regression Analysis

Regression analysis is one of the most widely employed statistical techniques (Takeaki and Hiroshi 2004: xi). Its purpose is to illuminate any underlying association between variables by fitting equations to the observed variables, according to some model (Rousseeuw and Leroy 1987: 1). The classic linear model relates the dependent, or 'response', variables y_i to independent 'explanatory' variables $x_{i1}, x_{i2}, \ldots, x_{ip}$ for $i = 1, \ldots, n$, such that

$$y_i = \boldsymbol{x}_i^T \boldsymbol{\beta} + \epsilon_i \qquad i = 1, \ldots, n,$$

where $\boldsymbol{x}_i^T = (x_{i1}, x_{i2}, \ldots, x_{ip})$, $\boldsymbol{\beta} = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}$ and ϵ_i, the 'error' term, is a random variable with expectation 0. It is necessary to fix $x_{i1} = 1$ for all i so that the first element of $\boldsymbol{\beta}$ corresponds to an intercept term. In order to fit such a model to the data, one has to use a regression estimator to estimate the unknown parameters in $\boldsymbol{\beta}$, generating:

$$\hat{\boldsymbol{\beta}} = \begin{pmatrix} \hat{\beta}_1 \\ \vdots \\ \hat{\beta}_p \end{pmatrix}.$$

The expected value of y_i, called the fitted value, is $\hat{y}_i = \boldsymbol{x}_i^T \hat{\boldsymbol{\beta}}$, and one can use this to calculate the 'residual' for the i^{th} case, $r_i = y_i - \hat{y}_i$.

The 'cornerstone of classical statistics' is the least squares regression estimator (Rousseeuw and Leroy 1987: 2). Legendre, a mathematical scientist working in theoretical and practical astronomy and geodesy, was in 1805 first to publish on the method of least squares; within 20 years the method was a standard tool in these fields in France, England, Prussia and the Italian states (Stigler 1986: 12, 15). This estimator became so popular partly because it could be calculated explicitly from data, and partly because it was derived from an easily understood criterion (Rousseeuw and Leroy 1987: vii, 2; Stigler 1986: 40).

It is first necessary to assume that the error terms are independently and identically normally distributed; $\epsilon_i \sim N\left(0, \sigma^2\right)$. Then the least squares regression estimator is the maximum likelihood estimator for $\boldsymbol{\beta}$, maximising

$$\prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{\epsilon_i^2}{2\sigma^2}\right) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{\left(y_i - \boldsymbol{x}_i^T \boldsymbol{\beta}\right)^2}{2\sigma^2}\right) \tag{1.1}$$

over $\boldsymbol{\beta}$, which is equivalent to maximising the logarithm of (1.1) over $\boldsymbol{\beta}$:

$$\sum_{i=1}^{n} \left(-\frac{1}{2} \ln\left(2\pi\sigma^2\right) - \frac{\epsilon_i^2}{2\sigma^2}\right).$$

This corresponds to minimising

$$\sum_{i=1}^{n} \epsilon_i^2$$

since σ is a constant. So, a least squares estimate is the value $\hat{\boldsymbol{\beta}}$ which results in the minimal sum of the squares of the residuals, $\sum_{i=1}^{n} r_i^2$. In simple regression this corresponds to fitting the regression line by minimising the sum of the squared vertical distances of observed points from the line, implying that the method is not symmetric in $\boldsymbol{x}$ and y (Rice 1995: 507). This has a profound effect on the way in which different types of points affect the regression estimate.

Calculation of the Least Squares Estimator

Firstly, define the design matrix $\boldsymbol{X}$, and the vectors $\boldsymbol{Y}$ and $\boldsymbol{\epsilon}$:

$$\boldsymbol{X} = \begin{pmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & & \vdots \\ x_{n1} & \cdots & x_{np} \end{pmatrix} = \begin{pmatrix} \boldsymbol{x}_1^T \\ \vdots \\ \boldsymbol{x}_n^T \end{pmatrix}, \boldsymbol{Y} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}, \text{ and } \boldsymbol{\epsilon} = \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{pmatrix}.$$

Now the classic linear model is $\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$. The least squares estimator aims to minimise

$$\begin{aligned} \sum_{i=1}^{n} \epsilon_i^2 &= \boldsymbol{\epsilon}^T \boldsymbol{\epsilon} \\ &= (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})^T (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}) \\ &= \boldsymbol{Y}^T\boldsymbol{Y} - \boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{Y} + \boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta}. \end{aligned}$$

At the minimum:

$$\begin{aligned} \frac{\partial}{\partial \boldsymbol{\beta}} \left(\sum_{i=1}^{n} \epsilon_i^2 \right) &= \frac{\partial}{\partial \boldsymbol{\beta}} \left(\boldsymbol{Y}^T\boldsymbol{Y} - \boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{Y} + \boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta} \right) \\ &= 0 - \boldsymbol{X}^T\boldsymbol{Y} - \boldsymbol{X}^T\boldsymbol{Y} + 2\left(\boldsymbol{X}^T\boldsymbol{X}\right)\boldsymbol{\beta}. \end{aligned}$$

So the least squares estimator $\hat{\boldsymbol{\beta}}$ is the solution to:

$$\boldsymbol{X}^T\boldsymbol{X}\hat{\boldsymbol{\beta}} = \boldsymbol{X}^T\boldsymbol{Y},$$

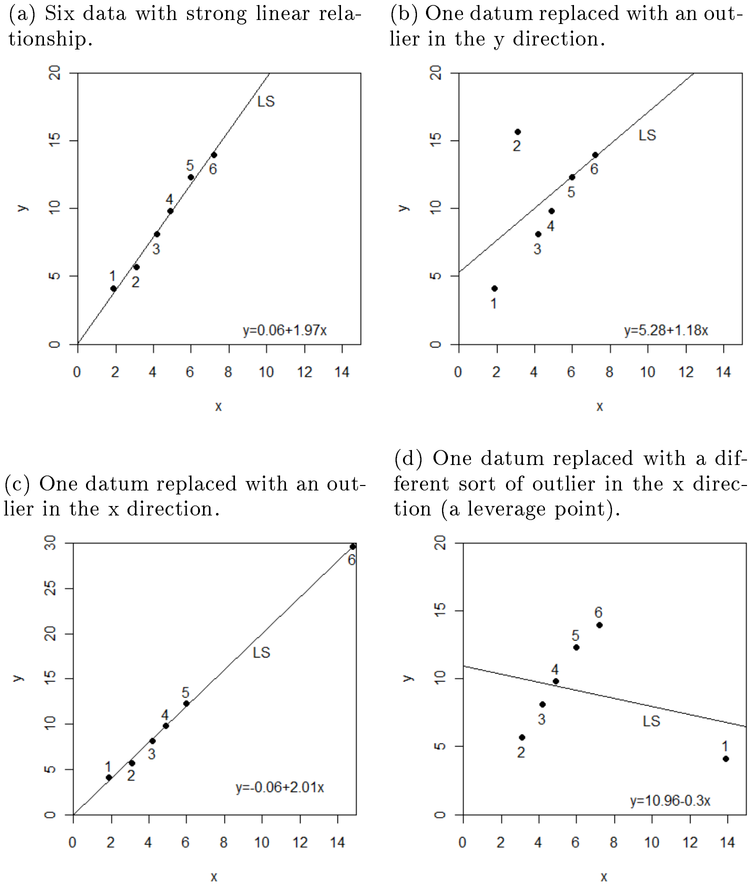
as this minimises $\hat{\boldsymbol{\epsilon}}^T\hat{\boldsymbol{\epsilon}} = \sum_{i=1}^{n} r_i^2$. Thus when $\boldsymbol{X}^T\boldsymbol{X}$ is non-singular the least squares estimator can be evaluated directly from the data:

$$\hat{\boldsymbol{\beta}} = \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{Y}.$$

1.3 Limitations of the least squares estimator

Although the least squares estimator is easy to calculate, it is also extremely sensitive to deviations from the model assumptions as a normal distribution is assumed for the errors. Hence, observations that are quite far from the majority of data can dramatically affect a least squares regression estimate. In the context of regression such gross errors are called 'outliers'. It is important to classify the different ways in which data can be outlying, as each has different repercussions on the least squares estimator due to its asymmetry . Figure 1.1 highlights this: whilst in Figure 1.1b the estimated regression line has been noticeably tilted, in Figure 1.1d, as a result of the outlier, the line is approaching the normal to the orginal estimated line.

Figure 1.1: Least squares estimates of regression lines in the presence of outliers.



(a) Six data with strong linear relationship.

(b) One datum replaced with an outlier in the y direction.

(c) One datum replaced with an outlier in the x direction.

(d) One datum replaced with a different sort of outlier in the x direction (a leverage point).

Classification of outliers

A datum which sits away from the rest of the distribution for a particular variable, either response or explanatory, is a ‘univariate’ outlier (Andersen 2008: 31). Such a point isn’t necessarily an outlier in regression though. A ‘regression’ outlier is a point that deviates from the linear relation followed by the bulk of the data, such that the y value of the datum *conditioned* on its $\boldsymbol{x}$ value is unusual (Rousseeuw and Leroy 1987: 7). Another type of outlier, called a ‘leverage point’, is one for which the explanatory variables, $(x_{k1}, \ldots, x_{kp})$, are outlying compared to the $(x_{i1}, \ldots, x_{ip})$ of the rest of the data (Rousseeuw and Leroy 1987: 5-6). Such a datum is said to ‘have leverage’ on the regression slope and the point has *high* leverage if it lies far from the means of the explanatory variables (Andersen 2008: 30).

A leverage point is not necessarily a regression outlier as it may still follow the general pattern in the data. If this is the case it is referred to as a ‘good’ leverage point, whereas if it *is* a regression outlier, it is called a ‘bad’ leverage point. A datum that is ‘influential’ is one for which the regression estimate changes considerably if it is removed. A bad leverage point is capable of tilting the least squares regression line so much that it might actually go through the point, so bad leverage points have leverage and influence on the estimate; they pose a severe threat.

With these definitions in mind the outliers in Figure 1.1 can be classified. In Figure 1.1b Point 2 is outlying only in its response variable conditioned on its explanatory variable, and so is a regression outlier. Figure 1.1c shows an example of a 'good' leverage point; although Point 6 is outlying with respect to the other x-values, the point fits the pattern of the bulk of the data. In Figure 1.1d Point 1, is only outlying in its x co-ordinate; it is a univariate outlier in terms of its x co-ordinate, and is a leverage point. The point has affected the least squares regression estimate so that it no longer represents the general trend in the bulk of the data at all; it is a 'bad' leverage point.

The least squares estimator's sensitivity to leverage points results from it minimising the sum of the *squared* residuals. Consider Figure 1.1a: there are no particularly large residuals present. However, in Figure 1.1d, Point 1 has been moved far from the original regression line horizontally and suddenly has an extremely large negative residual compared to this line, contributing an enormous amount to $\sum_{i=1}^{n} r_i^2$. In order to obtain the least squares estimate the sum of squared residuals must be re-minimised. To do this the regression line has to be tilted closer towards the outlying value; despite the fact that doing this increases the other points' residuals, they are were all so much smaller to start with that, overall, $\sum_{i=1}^{n} r_i^2$ decreases.

Deleting outliers before analysis

Although so far such outliers in data have been regarded as essentially erroneous, they could be far more interesting. Outliers could be caused by exceptional occurrences, or a group of outliers could be the results of a factor not yet considered in a study. There could even be something happening systematically that the model simply doesn't accommodate. There are circumstances in which data can be justifiably removed., but generally, since unusual observations are not necessarily 'bad' observations, it is reasonable to conclude that they should not be discounted.

It can be very difficult to spot outliers in the data without careful investigation. One cannot simply employ least squares regression and use residual plots to identify outlying data because the method is *so* sensitive to outliers; if the regression line has been pulled towards them, such data will not necessarily have large residuals at all (Rousseeuw and Leroy 1987: 7). Spotting outliers becomes particularly difficult in multivariate data sets as they cannot simply be identified on a scatter-plot (Hogg 1979: 108). It is in this situation, when outlying observations that could ruin the least squares estimate cannot be removed with justification, and such outlying data cannot necessarily even be spotted, that robust regression techniques are required.

1.4 Robust regression estimators

Concerning regression estimators, there is some confusion in the literature regarding the use of the words 'robust' and 'resistant'. Whilst Rousseeuw and Leroy (1987) make no differentiation between robust and resistant regression estimators, others, such as Andersen (2008: 3-4), make the distinction clear. To explain the division Andersen employs the definitions of the two types of robustness described in Section 1.1. *Resistant* regression estimators are primarily concerned with robustness of validity; their main concern is to prevent unusual observations from affecting the estimates produced. *Robust* regression estimators however are concerned with both robustness of efficiency *and* robustness of validity, meaning that they should also maintain a small sampling variance, even when the data does not fit the assumed distribution. This project will focus on making regression analysis robust to outlying observations, with the intention of being able to use the resulting estimates to make inferences. As such, the distinction between robust and resistant estimators will not be dwelt upon. Huber (2009: 3) states that 'distributionally robust' and 'outlier resistant' are interchangeable terms for most practical purposes.

In general, robust regression estimators aim to fit a model that describes the majority of a sample (Rousseeuw and Leroy 1987: viii). Their robustness is achieved by giving the data different weights within the calculation of the estimate, so that outlying data have a relatively smaller influence on the regression estimator (Draper and Smith 1998: 567). Comparatively, in least squares regression all data are treated equally.

Consequently, robust regression estimators can be a powerful tool for outlier detection in complicated data sets. Rousseeuw and Leroy (1987: ix, 25) describe such estimators as giving statisticians 'outliers on a "silver platter"'; by lessening the impact of outliers the estimators also expose them. However, Andersen (2008: 91-92) encourages a cautious approach to employing robust regression estimators because, unless the data is very well-behaved, different robust estimators may give very different estimates. On their own, robust regression methods do not provide a final model. Andersen suggests that it makes sense, as a primary stage in regression analysis, to employ both multiple robust methods *and* least squares to compare the results. If plotting the residuals of the least squares method against those of a robust method highlights potentially outlying data, it is necessary to explain this data if possible, before fitting a final model.

Identifying *multiple* influential observations, even using very resistant regression estimators, becomes much harder due to two effects called 'masking' and 'swamping*'* (Hadi and Simonoff 1993: 1264). Masking occurs when an outlying subset goes unnoticed because of the presence of another, whereas swamping refers to good observations being identified as outliers because of a remote subset of influential observations. Furthermore, even those robust regression estimators that are extremely resilient to outliers may be unable to detect curvature in the data, and so such techniques should be used in combination with ones which are able to detect this - nonparametric regression models for example (Andersen 2008: 91). Put simply, the results of fitting a robust model should be studied carefully.

Many robust regression estimators have been proposed, and in order to gain an appreciation of their strengths and weaknesses it is necessary to define some key properties.

Finite sample breakdown point of an estimator

Breakdown point (BDP) is a measure of the resistance of an estimator. The following description is based on Andersen (2008: 7-8) and Rousseeuw and Leroy (1987: 9-10). The BDP of a regression estimator is the smallest fraction of contamination that can cause the estimator to 'break down' and no longer represent the trend in the bulk of the data. When an estimator breaks down, the estimate it produces from the contaminated data can become *arbitrarily* far from the estimate it would give when the data was uncontaminated. In order to describe the BDP mathematically, define T as a regression estimator, Z as a sample of n data points, and $T\left(Z\right) = \hat{\boldsymbol{\beta}}$. Let Z' be a corrupted sample where m of the original data points are replaced with arbitrary values. The maximum effect that could be caused by such contamination is

$$\text{effect}(m; T, Z) = \sup_{Z'} \|T\left(Z'\right) - T\left(Z\right)\|. \tag{1.2}$$

When (1.2) is infinite, m outliers can have an arbitrarily large effect on T. The BDP of T at the sample Z is therefore defined as:

$$BDP\left(T, Z\right) = \min\left\{\frac{m}{n} : \text{effect}(m; T, Z)\,\text{is}\,\text{infinite}\right\}. \tag{1.3}$$

The least squares estimator for example has a breakdown point of $\frac{1}{n}$ because just one leverage point can cause it to break down. As the number of data increases, the breakdown point tends to

0, and so it is said that the least squares estimator has BDP 0%. Robust estimators aim to have better breakdown points than this, though some of the earliest estimators had breakdown points very close to, if not actually, 0%. Bianco *et al* (2005: 511-512) advise that one always ought to use an estimator with a BDP higher than the expected fraction of outliers. The highest BDP one can hope for is 50%, as if more than half the data is contaminated one cannot differentiate between 'good' and 'bad' data. Though an estimator with a high BDP will tend to produce reasonable estimates even when applied to data with multiple high leverage points, such an estimator becomes somewhat less appealing if the estimates produced for uncontaminated data are subject to much higher levels of uncertainty than least squares estimates. This motivates the consideration of the following property.

Relative efficiency of an estimator

Following Andersen (2008: 9), the efficiency of an estimator for a particular parameter is defined as the ratio of its minimum possible variance to its actual variance. Strictly, an estimator is considered 'efficient' when this ratio is one. An estimator that reaches an acceptable level of efficiency with larger samples is called 'asymptotically' efficient. Andersen (2008: 91) emphasises that high efficiency is crucial for an estimator if the intention is to use an estimate from sample data to make inferences about the larger population from which the sample was drawn.

Relative efficiency compares the efficiency of an estimator to that of a well known method. Given two estimators, T_1 and T_2, for a population parameter $\boldsymbol{\beta}$, where T_1 is the most efficient estimator possible and T_2 is less efficient, the relative efficiency of T_2 is calculated as the ratio of its mean squared error to the mean squared error of T_1 (Andersen 2008: 9):

$$\text{Efficiency}\,(T_1,\, T_2) = \frac{E\left[(T_1-\beta)^2\right]}{E\left[(T_2-\beta)^2\right]}.$$

In the context of regression, estimators are compared to the least squares estimator since, when its assumptions are met, it provides the minimum variance estimate, so is the most efficient estimator known. According to Andersen (2008: 10), relative efficiency is usually calculated in terms of asymptotic efficiency, so for small samples it is not necessarily a relevant property to consider. Indeed, the relative efficiency of a regression estimator is given as the ratio between its mean squared error and that of the least squares estimator, for an infinite sample with normally distributed errors.

Equivariance

If an estimator is equivariant it means that it transforms 'properly' in some sense. Rousseeuw and Leroy (1987: 116) discuss three crucial equivariance properties for a regression estimator:

- Regression equivariance
  An estimator T is regression equivariant if

  $$T\left(\left\{\left(\boldsymbol{x}_i^T,\, y_i+\boldsymbol{x}_i^T\boldsymbol{v}\right);\, i=1,\ldots,n\right\}\right) = T\left(\left\{\left(\boldsymbol{x}_i^T,\, y_i\right);\, i=1,\ldots,n\right\}\right)+\boldsymbol{v}.$$

  It means that any additional linear dependence $\boldsymbol{Y}\rightarrow\boldsymbol{Y}+\boldsymbol{X}\boldsymbol{v}$ is reflected in the regression vector accordingly $\hat{\boldsymbol{\beta}}\rightarrow\hat{\boldsymbol{\beta}}+\boldsymbol{v}$. This property is used routinely when studying regression estimators; in proofs of asymptotic properties it allows the fixing of $\boldsymbol{\beta}=\boldsymbol{0}$ without loss of generality.

- Scale equivariance
  An estimator being scale equivariant means that the fit produced by it is independent of the choice of measurement unit for the response variable. Therefore

  $$T\left(\left\{\left(\boldsymbol{x}_i^T, cy_i\right); i=1,\ldots,n\right\}\right) = cT\left(\left\{\left(\boldsymbol{x}_i^T, y_i\right); i=1,\ldots,n\right\}\right).$$

  So if $\boldsymbol{y} \rightarrow c\boldsymbol{y}$ then $\hat{\boldsymbol{\beta}} \rightarrow c\hat{\boldsymbol{\beta}}$. If an estimator is not scale equivariant it is necessary to standardise the residuals when estimating the regression parameters. This poses the difficulty of obtaining a robust estimate of the residual scale, when the model has not yet been fitted. This type of equivariance will be discussed in detail in Section 2.2.

- Affine equivariance
  An affine equivariant estimator satisfies

  $$T\left(\left\{\left(\boldsymbol{x}_i^T\boldsymbol{A}, y_i\right); i=1,\ldots,n\right\}\right) = \boldsymbol{A}^{-1}T\left(\left\{\left(\boldsymbol{x}_i^T, y_i\right); i=1,\ldots,n\right\}\right),$$

  So when the explanatory variables, $\boldsymbol{x}_i$, are linearly transformed, $\boldsymbol{X} \rightarrow \boldsymbol{X}\boldsymbol{A}$, the estimator is transformed accordingly,$\hat{\boldsymbol{\beta}} \rightarrow \boldsymbol{A}^{-1}\hat{\boldsymbol{\beta}}$. This is useful because it means that changing to a different co-ordinate system for the explanatory variables will not affect the estimate: $\hat{\boldsymbol{Y}} = \boldsymbol{X}\hat{\boldsymbol{\beta}} = (\boldsymbol{X}\boldsymbol{A})\left(\boldsymbol{A}^{-1}\hat{\boldsymbol{\beta}}\right)$.

The earliest proposed robust regression techniques had weaknesses in terms of one or more of these properties. However, many of the popular modern robust regression techniques, which perform well in terms of *all* of the properties given above, grew out of these earlier techniques. The following two chapters illustrate such a development, from the less robust M-estimator, to the highly robust and popular MM-estimator. In the fourth chapter, the weaknesses of the MM-estimator are reviewed, highlighting the limitations of the robustness qualities defined above. Finally, simulations are carried out in an attempt to ascertain the real nature of the MM-estimator.

Chapter 2

M-Estimators

Though M-estimators of location and scale had already been proposed, M-estimators of regression were first proposed by Huber (1973). They were one of the first attempts at a compromise between a resistant estimator and the efficiency of the least squares estimator (Andersen 2008: 51; Rousseeuw and Leroy 1987: 148). This chapter introduces such M-estimators of regression, considering their motivation and properties, discussing their use within modern robust regression techniques and concluding with an extended example of their application.

2.1 Maximum likelihood type estimators

The least squares estimator is obtained by minimising a function of the residuals obtained from the likelihood function of the assumed normal distribution of the errors. M-estimation is based on the idea that, whilst we still want a maximum likelihood estimator, the errors might be better represented by a different, heavier-tailed, distribution. If this probability distribution function is $f\left(\epsilon_i\right)$, then the maximum likelihood estimator for $\boldsymbol{\beta}$ is that which maximises the likelihood function

$$\prod_{i=1}^{n} f\left(\epsilon_i\right) = \prod_{i=1}^{n} f\left(y_i - \boldsymbol{x}_i^T \boldsymbol{\beta}\right).$$

This means it also maximises the log-likelihood function

$$\sum_{i=1}^{n} \ln f\left(\epsilon_i\right) = \sum_{i=1}^{n} \ln f\left(y_i - \boldsymbol{x}_i^T \boldsymbol{\beta}\right).$$

When the errors are normally distributed it has been shown that this leads to minimising the function $\sum_{i=1}^{n} r_i^2 = \sum_{i=1}^{n} \left(y_i - \boldsymbol{x}_i^T \boldsymbol{\beta}\right)^2$. Assuming that the errors are distributed differently leads to the maximum likelihood estimator minimising a different function. Using this idea, an M-estimator, or 'maximum likelihood type' estimator, $\hat{\boldsymbol{\beta}}$, minimises

$$\sum_{i=1}^{n} \rho\left(\epsilon_i\right) = \sum_{i=1}^{n} \rho\left(y_i - \boldsymbol{x}_i^T \boldsymbol{\beta}\right), \tag{2.1}$$

where $\rho\left(u\right)$ is a continuous, symmetric function called the 'objective' function, with a unique minimum at 0 (Andersen 2008: 51; Rousseeuw and Leroy 1987: 12). The least squares estimator is in fact a special less robust case of M-estimation. Of course, knowing the appropriate $\rho\left(u\right)$ to use requires knowledge of how the errors are really distributed, which is usually unclear

(Norman and Smith 1998: 569). Functions are instead chosen through consideration of how the resulting estimator down-weights the larger residuals (Norman and Smith 1998: 569). A robust M-estimator achieves this by minimising the sum of a less rapidly increasing objective function than the $\rho(u) = u^2$ of the least squares estimator (Andersen 2008: 51).

2.2 Constructing a scale equivariant estimator

The method of finding an M-estimate given a particular objective function and sample will be explored in Section 2.3. First, a slight alteration to (2.1) is needed, as an estimate found through minimisation of (2.1) would not be scale equivariant. But what does this mean? Hogg (1979: 109) gives an example of how non-scale equivariance affects an estimator. He explains that if such a $\hat{\boldsymbol{\beta}}$ were found for a sample of observations, and then the response variables replaced with ones for which the deviations from the initial estimated fit had been tripled, the new estimate for this modified sample would have changed. The $\hat{\boldsymbol{\beta}}$ of a scale equivariant estimator would have remained unchanged.

With this in mind, Example 2.1 demonstrates that the estimator defined as the minimisation of (2.1) is not scale equivariant. The procedure used to calculate the estimates is based on the procedure described in Section 2.3, using a Huber objective function as defined in Section 2.4.

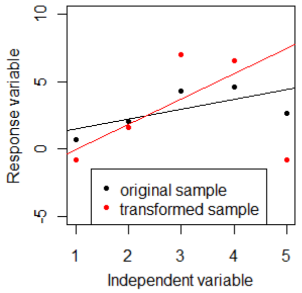
Example 2.1

i	x_i	y_i	$\hat{y}_i$	r_i	$\widetilde{y}_i$
1	1	0.696	1.464	-0.768	-0.841
2	2	2.007	2.200	-0.193	1.622
3	3	4.285	2.935	1.350	6.985
4	4	4.631	3.670	0.961	6.553
5	5	2.676	4.405	-1.729	-0.783

The black line in Figure 2.1 represents the slope $y = \hat{\beta}_1 + x\hat{\beta}_2$, for the sample of x and y shown in the table above, using the estimate $\hat{\boldsymbol{\beta}}^{(1)} = \begin{pmatrix} 0.729 \\ 0.735 \end{pmatrix}$ corresponding to (2.1). A sample of transformed response variables was constructed, labelled $\tilde{y}$, such that each of the points had been moved three times vertically further from the regression line of $\hat{\boldsymbol{\beta}}^{(1)}$. The vertical distance from the line to each observation in the original sample is its residual, $r_i = y_i - \hat{y}_i = y_i - \left(\hat{\beta}_1 + x_i\hat{\beta}_2\right)$. Hence each $\tilde{y}_i$ is the fitted value $\hat{y}_i$ summed with three times the residual for that observation: $\widetilde{y}_i = \hat{y}_i + 3r_i$. These new points are shown in red in Figure 2.1.

Since the responses have simply been moved three times further from the line given by $\hat{\boldsymbol{\beta}}^{(1)}$, a good estimator would surely give the same estimate again. However, using the sample $(x_i, \widetilde{y}_i)$ and (2.1) the new estimate for $\boldsymbol{\beta}$ is $\hat{\boldsymbol{\beta}}^{(2)} = \begin{pmatrix} -1.962 \\ 1.888 \end{pmatrix}$. The red line in Figure 2.1 represents the slope corresponding to this estimate: it is clearly very different; the estimator is not scale equivariant.

Figure 2.1: Plot demonstrating the non-scale equivariance of the estimator defined by minimisation of (2.1).



The problem of scale equivariance is solved by simply standardising the ϵ_i in (2.1) by an estimate, s, of their scale (Hogg 1979: 109). So M-estimators actually minimise

$$\sum_{i=1}^{n} \rho \left(\frac{\epsilon_i}{s} \right) \tag{2.2}$$

(Norman and Smith 1998: 569).

The standard deviation of the sample of residuals cannot be used for s because it is strongly affected by outliers; a robust estimator of scale is needed (Hogg 1979: 109). It is important to choose the scale estimator carefully because the regression estimator is not invariant to the estimator used (Norman and Smith 1998: 572). It should also be noted that one cannot estimate the scale of the errors without first estimating the errors using $\hat{\boldsymbol{\beta}}$, which, as has just been shown, requires an estimate of the scale itself. The errors and scale need to be estimated simultaneously. An iterative procedure that gradually converges on an estimate for both will be required (Andersen 2008: 52).

A popular estimator for s is 're-scaled MAD':

$$s = 1.4826 \times MAD,$$

where MAD (median absolute deviation) is calculated as follows:

$$MAD = \text{median} \mid y_i - \boldsymbol{x}_i^T \hat{\boldsymbol{\beta}} \mid = \text{median} \mid r_i \mid$$

(Andersen 2008: 51; Draper and Smith 1998: 572; Street *et al* 1988: 153). It is highly resistant to outlying observations, with BDP 50%, as it is based on the median rather than the mean (Andersen 2008: 16, 51). The estimator s rescales MAD by the factor 1.4826 so that when the sample is large and ϵ_i really are distributed as $N\left(0, \sigma^2\right)$, s estimates the standard deviation

(Hogg 1979: 109). With a large sample and $\epsilon_i \sim N\left(0, \sigma^2\right)$:

$$\begin{aligned}
P\left(\mid \epsilon_i \mid < MAD\right) &\approx 0.5 \\
\Longrightarrow P\left(\mid \frac{\epsilon_i - 0}{\sigma} \mid < \frac{MAD}{\sigma}\right) &\approx 0.5 \\
\Longrightarrow P\left(\mid Z \mid < \frac{MAD}{\sigma}\right) &\approx 0.5 \\
\Longrightarrow \frac{MAD}{\sigma} &\approx \Phi^{-1}\left(0.75\right) \\
\Longrightarrow \frac{MAD}{\Phi^{-1}\left(0.75\right)} &\approx \sigma \\
\Longrightarrow 1.4826 \times MAD &\approx \sigma
\end{aligned}$$

where $Z \sim N\left(0, 1\right)$, and it is used that $\frac{1}{\Phi^{-1}(0.75)} \approx 1.4826$.

Alternatives to this s include Huber's Proposal 2. This is defined as the solution to

$$\frac{1}{n-p} \sum_{i=1}^{n} \psi^2 \left(\frac{y_i - \boldsymbol{x}_i^T \hat{\boldsymbol{\beta}}}{s} \right) = E_\Phi \left(\psi^2 \left(\epsilon \right) \right), \tag{2.3}$$

where $\psi\left(u\right) = \frac{\partial \rho}{\partial u}$, and $E_\Phi\left(\psi^2\left(\epsilon\right)\right)$ is the expected value of $\psi^2\left(\epsilon\right)$ when the errors have a standard normal distribution (Huber 1996: 36; Street *et al* 1988: 153). To find such an estimate requires iteration. The formula for the next step in an iterative calculation of Huber's Proposal 2, given a current estimate s, is given by Street *et al* (1988: 153).

Re-scaled MAD and Huber's Proposal 2 are the two options for scale-estimation built in to the M-estimation procedure *rlm* in R's 'MASS' package. Huber's Proposal 2 is preferred by Shrader and Hettmansperger (1980: 95-96) who say that re-scaled MAD 'leads to very liberal tests' and also emphasise that, unlike re-scaled MAD, Huber's Proposal 2 'varies quite smoothly with the data'. Huber (1996: 36) favours this proposal over re-scaled MAD in the case of M-estimation of regression because he considers it easier to deal with in theory such as convergence proofs. In this report the re-scaled MAD estimator will be used, partly because many sources, such as Andersen (2008), Draper and Smith (1998), and Rousseeuw and Leroy (1987), refer only to estimators based on MAD and partly because it is easier to calculate it directly from a sample!

To conclude this section, using $s = 1.4826 \times MAD$, Example 2.2 demonstrates the scale equivariance of M-estimators.

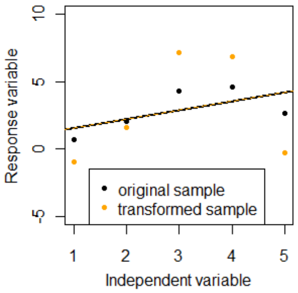
Example 2.2

i	x_i	y_i	$\hat{y}_i$	r_i	$\widetilde{y}_i$
1	1	0.696	1.542	-0.846	-0.996
2	2	2.007	2.200	-0.194	1.620
3	3	4.285	2.859	1.426	7.137
4	4	4.631	3.517	1.114	6.858
5	5	2.676	4.176	-1.500	-0.324

The black line in Figure 2.2 represents the slope $y = \hat{\beta}_1 + x\hat{\beta}_2$, for the sample of x and y shown in the table above, using the estimate $\hat{\boldsymbol{\beta}}^{(1)} = \begin{pmatrix} 0.884 \\ 0.658 \end{pmatrix}$ corresponding to (2.2). A sample of transformed y variables, $\tilde{y}$, was again constructed such that the points had been moved three times vertically further from the regression line of $\hat{\boldsymbol{\beta}}^{(1)}$: $\tilde{y}_i = \hat{y}_i + 3r_i$. The new points are shown in orange in Figure 2.2.

Using this new sample and (2.2), the estimate is $\hat{\boldsymbol{\beta}}^{(2)} = \begin{pmatrix} 0.884 \\ 0.658 \end{pmatrix}$, which corresponds to the dotted orange line in Figure 2.2. Clearly, transforming the sample of response variables has not altered the estimate for $\boldsymbol{\beta}$, hence the estimator is scale equivariant as desired.

Figure 2.2: Plot demonstrating the scale equivariance of the estimator defined by minimisation of (2.2).



2.3 Finding an M-estimate

The following is based on Draper and Smith (1998: 569, 572). Minimising (2.2) to find an M-estimate requires partial differentiation with respect to each of the p parameters in turn, resulting in a system of p equations:

$$\sum_{i=1}^{n} \boldsymbol{x}_{ij} \psi \left(\frac{y_i - \boldsymbol{x}_i^T \boldsymbol{\beta}}{s} \right) = \sum_{i=1}^{n} \boldsymbol{x}_{ij} \psi \left(\frac{\epsilon_i}{s} \right) = 0 \qquad j = 1, \ldots, p \tag{2.4}$$

where $\psi(u) = \frac{\partial \rho}{\partial u}$ is called the 'score' function. A 'weight' function is then defined as

$$w(u) = \frac{\psi(u)}{u}$$

yielding $w_i = w\left(\frac{\epsilon_i}{s}\right)$ for $i = 1, \ldots, n$, with $w_i = 1$ if $\epsilon_i = 0$. Substituting this into (2.4) results in

$$\begin{aligned} \sum_{i=1}^{n} \boldsymbol{x}_{ij} w_i \frac{\epsilon_i}{s} = \sum_{i=1}^{n} \boldsymbol{x}_{ij} w_i \left(y_i - \boldsymbol{x}_i^T \boldsymbol{\beta} \right) \frac{1}{s} &= 0 \qquad j = 1, \ldots, p \\ \Longrightarrow \sum_{i=1}^{n} \boldsymbol{x}_{ij} w_i \left(y_i - \boldsymbol{x}_i^T \boldsymbol{\beta} \right) &= 0 \qquad j = 1, \ldots, p \\ \Longrightarrow \sum_{i=1}^{n} \boldsymbol{x}_{ij} w_i \boldsymbol{x}_i \boldsymbol{\beta} &= \sum_{i=1}^{n} \boldsymbol{x}_{ij} w_i y_i \qquad j = 1, \ldots, p \end{aligned} \tag{2.5}$$

since $s \neq 0$. Defining the weight matrix $\boldsymbol{W} = \text{diag}\left(\{w_i : i = 1, \ldots, n\}\right)$ as follows:

$$\boldsymbol{W} = \begin{pmatrix} w_1 & & & 0 \\ & w_2 & & \\ & & \ddots & \\ 0 & & & w_n \end{pmatrix}$$

yields the following matrix form of (2.5):

$$\begin{aligned} \boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X}\boldsymbol{\beta} &= \boldsymbol{X}^T\boldsymbol{W}\boldsymbol{Y} \\ \Longrightarrow \qquad \hat{\boldsymbol{\beta}} &= \left(\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{Y}. \end{aligned} \tag{2.6}$$

This is very similar to the solution for the least squares estimator, but with the introduction of a weight matrix to reduce the influence of outliers. Generally, unlike least squares, (2.6) cannot be used to calculate an M-estimate directly from data, since $\boldsymbol{W}$ depends on the residuals, which depend on the estimate. In fact an initial estimate and iterations are required, to eventually converge on $\boldsymbol{W}$ and an M-estimate for $\boldsymbol{\beta}$. M-estimates of regression are found using the iterative procedure IRLS:

Iteratively Reweighted Least Squares (IRLS) (Andersen 2008: 52; Draper and Smith 1998: 572)

1. With the iteration counter I set to 0, the least squares method is used to fit an initial model to the data, yielding the initial estimates of the regression coefficients $\hat{\boldsymbol{\beta}}^{(0)}$.

2. Initial residuals $r_i^{(0)}$ are found using $\hat{\boldsymbol{\beta}}^{(0)}$ and used to calculate $s^{(0)}$.

3. A weight function $w\left(u\right)$ is chosen and applied to $\frac{r_i^{(0)}}{s^{(0)}}$ to obtain preliminary weights $w_i^{(0)}$. These give the value of $\boldsymbol{W}^{(0)}$ for $\hat{\boldsymbol{\beta}}^{(0)}$.

4. Set $I = 1$. Using $\boldsymbol{W}^{(0)}$ in (2.6), one obtains the estimate $\hat{\boldsymbol{\beta}}^{(1)} = \left(\boldsymbol{X}^T\boldsymbol{W}^{(0)}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{W}^{(0)}\boldsymbol{Y}$.

5. Using $\hat{\boldsymbol{\beta}}^{(1)}$ new residuals, $r_i^{(1)}$, can be found, which, via calculation of $s^{(1)}$ and application of the weight function yield $\boldsymbol{W}^{(1)}$.

6. Set $I = 2$. A new estimate for $\boldsymbol{\beta}$ is found using $\boldsymbol{W}^{(1)}$. This is $\hat{\boldsymbol{\beta}}^{(2)}$. $r_i^{(2)}$ and $s^{(2)}$, and in turn the next weight matrix, $\boldsymbol{W}^{(2)}$, are then found.

7. This process is now iterated such that at $I = q$

$$\hat{\boldsymbol{\beta}}^{(q)} = \left(\boldsymbol{X}^T\boldsymbol{W}^{(q-1)}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{W}^{(q-1)}\boldsymbol{Y},$$

   until the estimates of $\boldsymbol{\beta}$ converge, at which point the final M-estimate has been found.

Convergence tends to be reached quickly, and the procedure is usually stopped once the estimate changes by less than a selected percentage between iterations, or after a fixed number of iterations have been carried out. When iterating until a convergence criterion has been met, the criterion should be of the form

$$\frac{\parallel \hat{\boldsymbol{\beta}}^{(q+1)} - \hat{\boldsymbol{\beta}}^{(q)} \parallel}{\parallel \hat{\boldsymbol{\beta}}^{(q+1)} \parallel} < \varepsilon$$

where ε is a small positive number, often fixed at 0.0001. This is slightly different to the convergence criterion used by the *rlm* function in R, which iterates until the percentage change in the size of the residuals between iterations is smaller than ε:

$$\frac{\parallel \boldsymbol{r}^{(q+1)} - \boldsymbol{r}^{(q)} \parallel}{\parallel \boldsymbol{r}^{(q+1)} \parallel} < \varepsilon \quad \text{where} \quad \boldsymbol{r} = \begin{pmatrix} r_1 \\ \vdots \\ r_n \end{pmatrix}.$$

2.4 Weight functions and their implications

Choosing a weight function to apply to the scaled residuals corresponds directly to choosing a probability distribution function for the errors. However, by choosing a function that results in a *robust* estimator, it is not necessary to assume that the errors have exactly this distribution. In practice, one can simply choose a weight function to apply without considering the associated $f(\epsilon)$. Several suggestions for weight functions, with their corresponding score function and objective function, have been made in the literature. Each set of functions given includes tuning constants, which allow for the shape of the function to be slightly altered (Draper and Smith 1998: 571). Table 2.1, adapted from Draper and Smith (1998: 570, Table 25.1) and Andersen (2008: 22, Table 2.1), with reference to Rousseeuw and Leroy (1987: 148, 129), shows the three most commonly used M-estimators.

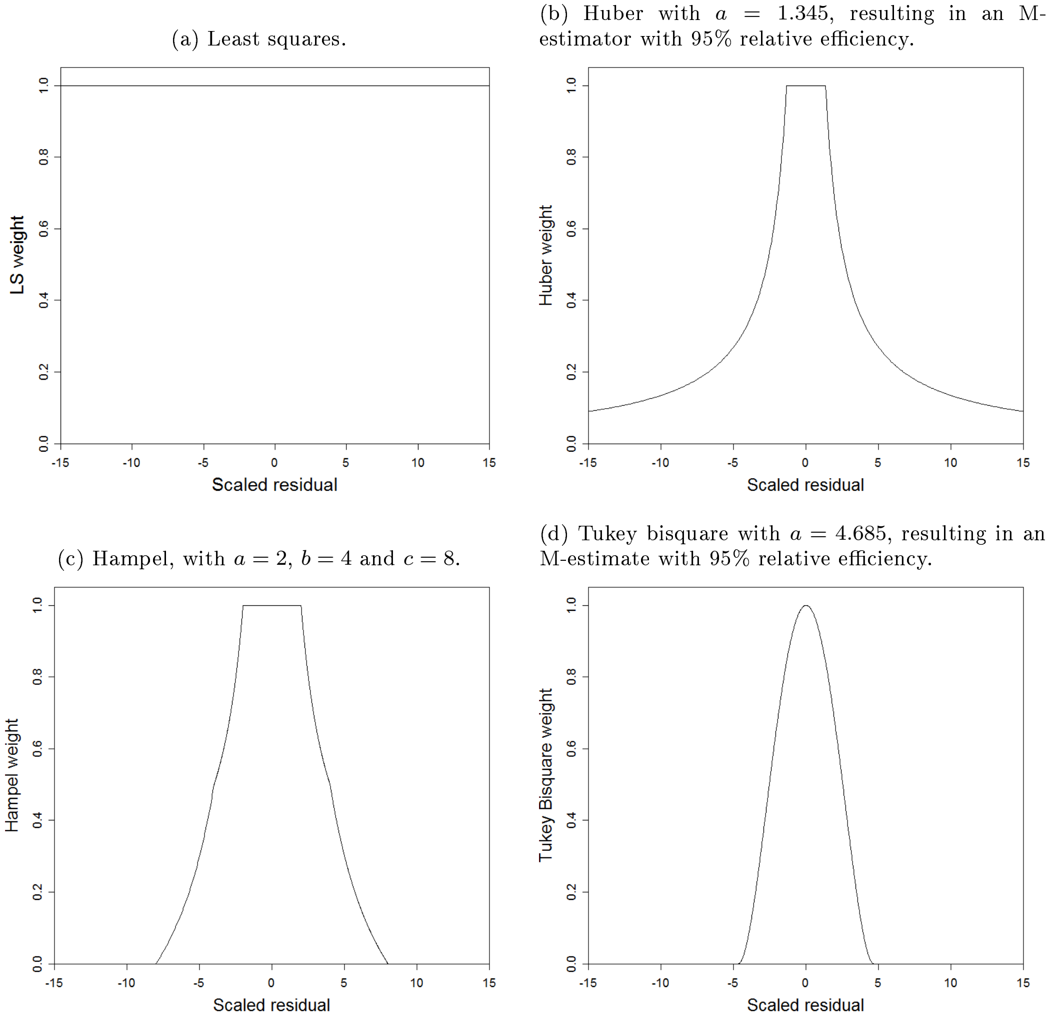
2.5 Properties of M-estimators

Figure 2.3 illustrates the different ways in which the $w(u)$ weight the scaled residuals. As Figure 2.3 shows, robust M-estimators' weight functions give reduced weights at the tails compared to the least squares estimator, which gives weight one to all observations. This means that unusually large residuals have a much smaller effect on the estimate than they would have had if using the least squares method. As a result M-estimators are more robust to heavy-tailed error distributions and non-constant error variance (Andersen 2008: 53). The choice of weight function determines the nature of the robustness to outliers and research has focused on finding functions that make the associated M-estimator as robust as possible, but still fairly efficient (Rousseeuw and Leroy 1987: 148). Changing the tuning constants also alters the relative efficiency, and the resistance of the resulting M-estimator. There are standard values (or ranges) for tuning constants, resulting in estimators with 95% asymptotic relative efficiency. The standard values have been used in Figure 2.3. The differences between M-estimators can be understood through consideration of Figure 2.3.

Table 2.1: Popular functions for M-estimators

	Objective Function $\rho\left(u\right)$	Score Function $\psi\left(u\right)$	Weight Function $w\left(u\right)=\frac{\psi(u)}{u}$
(a) Least Squares	$\frac{1}{2}u^2 \quad -\infty \leq u \leq \infty$	u	1
(b) Huber (1973), $a>0$	$\begin{cases} \frac{1}{2}u^2 & \text{if } \|u\|<a \\ a\|u\|-\frac{1}{2}a^2 & \text{if } \|u\| \geq a \end{cases}$	$\begin{cases} u & \text{if } \|u\|<a \\ a\,\text{sign}\,u & \text{if } \|u\| \geq a \end{cases}$	$\begin{cases} 1 & \text{if } \|u\|<a \\ \frac{a}{\|u\|} & \text{if } \|u\| \geq a \end{cases}$
(c) Hampel, $a,b,c>0$	$\begin{cases} \frac{1}{2}u^2 & \text{if } \|u\|<a \\ a\|u\|-\frac{1}{2}a^2 & \text{if } a \leq \|u\| < b \\ a\frac{c\|u\|-\frac{1}{2}u^2}{c-b}-\frac{7a^2}{6} & \text{if } b \leq \|u\| \leq c \\ a\left(b+c-a\right) & \text{otherwise} \end{cases}$	$\begin{cases} u & \text{if } \|u\|<a \\ a\,\text{sign}\,u & \text{if } a \leq \|u\| < b \\ a\frac{c\,\text{sign}\,u-u}{c-b} & \text{if } b \leq \|u\| \leq c \\ 0 & \text{otherwise} \end{cases}$	$\begin{cases} 1 & \text{if } \|u\|<a \\ \frac{a}{\|u\|} & \text{if } a \leq \|u\| < b \\ a\frac{c/\|u\|-1}{c-b} & \text{if } b \leq \|u\| \leq c \\ 0 & \text{otherwise} \end{cases}$
(d) Tukey bisquare, $a>0$	$\begin{cases} \frac{a^2}{6}\left(1-\left(1-\left(\frac{u}{a}\right)^2\right)^3\right) & \text{if } \|u\| \leq a \\ \frac{1}{6}a^2 & \text{if } \|u\| > a \end{cases}$	$\begin{cases} u\left(1-\left(\frac{u}{a}\right)^2\right)^2 & \text{if } \|u\| \leq a \\ 0 & \text{if } \|u\| > a \end{cases}$	$\begin{cases} \left(1-\left(\frac{u}{a}\right)^2\right)^2 & \text{if } \|u\| \leq a \\ 0 & \text{if } \|u\| > a \end{cases}$

Figure 2.3: Plots of the weight functions of the Table 2.1 using typical tuning constants.



(a) Least squares.

(b) Huber with $a = 1.345$, resulting in an M-estimator with 95% relative efficiency.

(c) Hampel, with $a = 2$, $b = 4$ and $c = 8$.

(d) Tukey bisquare with $a = 4.685$, resulting in an M-estimate with 95% relative efficiency.

The Huber M-estimator corresponds to a probability distribution for the errors which is normal in the centre but like a double exponential distribution in the tails (Hogg 1979: 109). The resulting weight function gives those observations whose scaled residual is within the central bound weight one, whilst scaled residuals outside the bound receive smaller weights - increasingly so. The curve of the Huber weight function never gives any scaled residuals zero, so all data is considered.

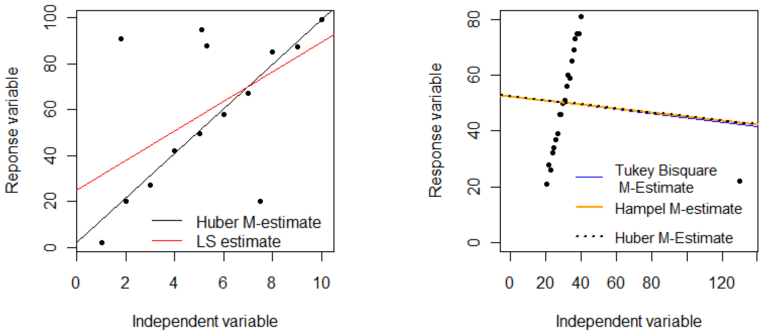
Some M-estimators do assign weight zero to scaled residuals. These are called ‘redescending’ M-estimators, defined such that $\psi(u) = 0$ if $|u| > a$. The Hampel ‘three-part redescending’ M-estimator is an example (see Figure 2.3c). Within a central bound, determined by a tuning constant, the Hampel weight function has a similar shape to Huber’s. However, outside this region the weights given to scaled residuals decrease more rapidly, until eventually absolute scaled residuals larger than another tuning constant are given weight zero. This makes the Hampel M-estimator more resistant to regression outliers (Rousseeuw and Leroy 1987: 148).

The Tukey bisquare weight function, also referred to as the biweight function, produces an M-estimator that is also more resistant to regression outliers than the Huber M-estimator (Andersen 2008: 19). It is another redescending M-estimator, which can be seen in Figure 2.3d. The figure also illustrates how the estimate gives a far smaller proportion of the scaled residuals weight one than either of the Huber or the Hampel M-estimators.

Whilst the different M-estimators have varying levels of resistance to outliers, they have been defined to be much more resistant to vertical outliers than the least squares estimator. Figure 2.5a demonstrates this increased robustness. Moreover, they are highly efficient; when the errors have expectation 0, constant variance and are uncorrelated, M-estimators are 95% as efficient as the least squares estimator, and when there is a heavy-tailed error distribution and non-constant error variance, M-estimators are *more* efficient than the least squares estimator (Andersen 2008: 53).

However, like the least squares estimator, M-estimators are still defined assuming that there are no errors in the independent variables, $\boldsymbol{x}_i$; M-estimators do not consider leverage. One bad leverage point can cause the estimator to entirely break down, and thus M-estimators have an overall breakdown point of 0%. In comparison to this, other early robust estimators of regression, such as the 'LTS' and 'LMS' estimators, were very inefficient but reached breakdown points of 50% (Andersen 2008: 58).

Figure 2.4: The influence of outliers on the M-estimator.



(a) Plot demonstrating robustness of the Huber M-estimator to multiple vertical outliers.

(b) Plot showing vulnerability of M-estimators to a single leverage point. The three lines are almost identical. The LS line for these data (not shown here) is extremely similar to these estimates.

In conclusion, M-estimators were the first estimators that were as robust as possible with regards to vertical outliers whilst retaining a high relative efficiency, but their low breakdown point means that in some situations they perform no better than the least squares estimator. As a result they are not used as robust estimators of regression today. However, M-estimation remains a important technique, as it is used within modern robust regression techniques which do produce highly robust and asymptotically efficient estimates (Andersen 2008: 53).

2.6 An example

The M-estimation procedure will now be demonstrated using real data: the US Judge Ratings data set. The data set contains lawyers' ratings of state judges in the US Superior Court, and contains forty three observations of twelve independent variables. In this example a single independent variable, the diligence rating, is considered, to enable visual comparison between the non-robust initial estimate and the final M-estimate. The response variable is physical ability rating. The sample is shown in Appendix A. The IRLS procedure is followed, using the Huber weight function and the re-scaled MAD scale estimator. The convergence criterion being used is

$$\frac{\parallel \hat{\boldsymbol{\beta}}^{(q+1)} - \hat{\boldsymbol{\beta}}^{(q)} \parallel}{\parallel \hat{\boldsymbol{\beta}}^{(q+1)} \parallel} < 0.0001.$$

Let the i^{th} diligence rating be denoted diligence$_i$, with $i = 1, \dots, n$, and the i^{th} physical ability rating be denoted physical$_i$. Then $\boldsymbol{X}$ is a 43×2 matrix such that

$$\boldsymbol{X} = \begin{pmatrix} 1 & \text{diligence}_{\mathit{1}} \\ \vdots & \vdots \\ 1 & \text{diligence}_{\mathit{43}} \end{pmatrix} \text{ and } \boldsymbol{Y} = \begin{pmatrix} \text{physical}_{\mathit{1}} \\ \vdots \\ \text{physical}_{\mathit{43}} \end{pmatrix}.$$

IRLS Iteration I=0:

- The initial LS estimate is found: $\hat{\boldsymbol{\beta}}^{(0)} = \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{Y} = \begin{pmatrix} 1.412 \\ 0.847 \end{pmatrix}$.

- The initial residuals, $r_i^{(0)} = y_i - \boldsymbol{x}_i^T\hat{\boldsymbol{\beta}}^{(0)}$ are calculated, which allow the initial scale estimate to be found: $s^{(0)} = 1.4826 \times \text{median} \mid r_i^{(0)} \mid = 1.4286 \times 0.179 \ldots = 0.266 \ldots$

- The initial standardised residuals, $u_i^{(0)} = \frac{r_i^{(0)}}{s^{(0)}}$, are calculated.

- The Huber weight function with $a = 1.345$ is applied to $u_i^{(0)}$.
  $w_i^{(0)} = 1$ if $\mid u_i^{(0)} \mid < 1.345$, else $w_i^{(0)} = \frac{1.345}{|u_i^{(0)}|}$.

- Finally, $\boldsymbol{W}^{(0)} = diag\left(\left\{w_i^{(0)} : i = 1, \dots, 43\right\}\right)$.

Table 2.2: Residuals, scaled residuals and weights from Iteration I=0

i	1	2	3	4	5	6	7	8	9	10	11
$r_i^{(0)}$	0.698	-0.119	-0.126	-0.073	-1.423	-0.019	0.311	1.064	0.011	0.220	0.714
$u_i^{(0)}$	2.618	-0.446	-0.471	-0.275	-5.336	-0.071	1.167	3.987	0.043	0.825	2.675
$w_i^{(0)}$	0.514	1	1	1	0.252	1	1	0.337	1	1	0.503

i	12	13	14	15	16	17	18	19	20	21	22
$r_i^{(0)}$	-0.617	0.922	-0.978	0.205	0.059	-0.617	0.329	-0.026	0.192	0.068	0.222
$u_i^{(0)}$	-2.313	3.457	-3.665	0.768	0.222	-2.312	1.232	-0.096	0.719	0.255	0.833
$w_i^{(0)}$	0.582	0.389	0.367	1	1	0.582	1	1	1	1	1

i	23	24	25	26	27	28	29	30	31	32	33
$r_i^{(0)}$	-1.969	-0.141	-0.004	0.042	0.105	0.011	0.035	-0.143	-0.134	-0.010	0.081
$u_i^{(0)}$	-7.381	-0.528	-0.014	0.157	0.393	0.043	0.132	-0.536	-0.503	-0.039	0.303
$w_i^{(0)}$	0.182	1	1	1	1	1	1	1	1	1	1

i	34	35	36	37	38	39	40	41	42	43
$r_i^{(0)}$	0.011	0.140	-0.449	-0.180	0.474	0.822	-0.349	0.253	0.359	0.029
$u_i^{(0)}$	0.043	0.5234	-1.685	-0.674	1.778	3.082	-1.310	0.947	1.347	0.108
$w_i^{(0)}$	1	1	0.798	1	0.756	0.436	1	1	0.999	1

IRLS Iteration I=1:

- The new estimate for $\boldsymbol{\beta}$ is calculated: $\hat{\boldsymbol{\beta}}^{(1)} = \left(\boldsymbol{X}^T\boldsymbol{W}^{(0)}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{W}^{(0)}\boldsymbol{Y} = \begin{pmatrix} 2.002\ldots \\ 0.777\ldots \end{pmatrix}$.
- $\frac{\|\hat{\boldsymbol{\beta}}^{(1)}-\hat{\boldsymbol{\beta}}^{(0)}\|}{\|\hat{\boldsymbol{\beta}}^{(1)}\|} = 0.276405\ldots > 0.0001$, so the procedure continues.
- $r_i^{(1)} = y_i - \boldsymbol{x}_i^T\hat{\boldsymbol{\beta}}^{(1)}$, are calculated, which allow the new scale estimate to be found: $s^{(1)} = 1.4826 \times \text{median} \mid r_i^{(1)} \mid = 0.230\ldots$ so the standardised residuals, $u_i^{(1)} = \frac{r_i^{(1)}}{s^{(1)}}$, can be calculated.
- The Huber weight function with $a = 1.345$ is applied to $u_i^{(1)}$
  $w_i^{(1)} = 1$ if $\mid u_i^{(1)} \mid < 1.345$, else $w_i^{(1)} = \frac{1.345}{|u_i^{(1)}|}$.
- Finally, $\boldsymbol{W}^{(1)} = diag\left(\left\{w_i^{(1)} : i = 1, \ldots, 43\right\}\right)$.

$\vdots$

IRLS Iteration I=9:

- $\hat{\boldsymbol{\beta}}^{(9)} = \left(\boldsymbol{X}^T\boldsymbol{W}^{(8)}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{W}^{(8)}\boldsymbol{Y} = \begin{pmatrix} 2.2680\ldots \\ 0.7462\ldots \end{pmatrix}$.
- $\frac{\|\hat{\boldsymbol{\beta}}^{(9)}-\hat{\boldsymbol{\beta}}^{(8)}\|}{\|\hat{\boldsymbol{\beta}}^{(9)}\|} = 0.000166\ldots > 0.0001$, so the procedure continues.
- Residuals, scale estimate, and weights are calculated as demonstrated previously.

IRLS Iteration I=10:

- $\hat{\boldsymbol\beta}^{(10)} = \left(\boldsymbol{X}^T\boldsymbol{W}^{(9)}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{W}^{(9)}\boldsymbol{Y} = \begin{pmatrix}2.268\ldots\\0.746\ldots\end{pmatrix}.$
- $\frac{\|\hat{\boldsymbol\beta}^{(10)}-\hat{\boldsymbol\beta}^{(9)}\|}{\|\hat{\boldsymbol\beta}^{(10)}\|} = 0.0000719\ldots < 0.0001$, so this is the final M-estimate.
- $r_i^{(1)} = y_i - \boldsymbol{x}_i^T\hat{\boldsymbol\beta}^{(1)} \rightarrow s^{(1)} = 1.4826 \times \text{median} \mid r_i^{(1)} \mid = 0.217\ldots$

Table 2.3: Final weightings given to observations in IRLS procedure

i	1	2	3	4	5	6	7	8	9	10	11
$w_i^{(0)}$	0.501	1	1	1	0.181	1	1	0.403	1	1	0.480
i	12	13	14	15	16	17	18	19	20	21	22
$w_i^{(0)}$	0.395	0.386	0.256	1	1	0.395	1	1	1	1	1
i	23	24	25	26	27	28	29	30	31	32	33
$w_i^{(0)}$	0.133	1	1	1	1	1	1	1	1	1	1
i	34	35	36	37	38	39	40	41	42	43	
$w_i^{(0)}$	1	1	0.634	1	0.711	0.445	0.810	1	1	1	

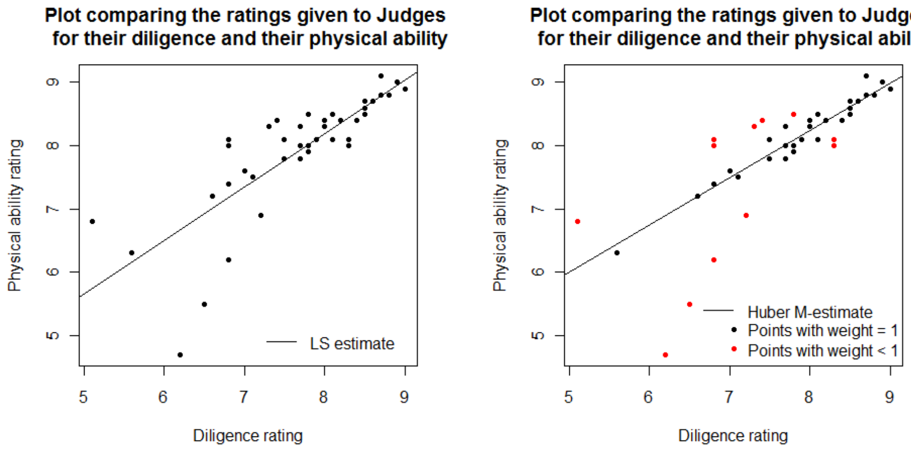


Figure 2.5: The LS line has clearly been tilted downwards, preventing it from representing the general trend in the majority of the data. Although the final M-estimate is an improvement on the least squares estimate in this respect, the line still appears slightly tilted downwards. It seems that the vertical outliers are influencing the M-estimator.

Chapter 3

The MM-estimator: A High Breakdown, Efficient Estimator

MM-estimators combine the high asymptotic relative efficiency of M-estimators with the high breakdown of a class of estimators called S-estimators. Introduced by Yohai (1987), they were among the first robust estimators to have these two properties simultaneously. The 'MM' refers to the fact that multiple M-estimation procedures are carried out in the computation of the estimator. It is highly worthwhile to consider the MM-estimator as 'it is perhaps now the most commonly employed robust regression technique' Andersen (2008: 56). This chapter defines MM-estimators, explaining the origins of their impressive robustness properties, and demonstrating these properties through examples using both real and simulated data. However, before considering the MM-estimator, the S-estimator needs introducing.

3.1 S-estimators

S-estimators form a class of high breakdown estimators of regression, which are also affine, scale and regression equivariant (Rousseeuw and Leroy 1987: 135-136). Rousseeuw and Yohai (1984) first proposed these estimators, choosing to call them S-estimators because they were based on estimates of scale. In the same way that the least squares estimator minimises the variance of the residuals, S-estimators minimise the dispersion of the residuals, $s\left(r_1\left(\boldsymbol{\beta}\right),\ldots,r_n\left(\boldsymbol{\beta}\right)\right)$. Rousseeuw and Leroy (1987: 135-136) define the dispersion, s, of the residuals as the solution of

$$\frac{1}{n}\sum_{i=1}^{n}\rho\left(\frac{r_i}{s}\right)=K \tag{3.1}$$

where K is a constant and the objective function ρ satisfies the following conditions:

1. ρ is symmetric and continuously differentiable, and $\rho\left(0\right)=0$.
2. There exists $a>0$ such that ρ is strictly increasing on $[0,a]$ and constant on $[a,\infty)$.
3. $\frac{K}{\rho(a)}=\frac{1}{2}$.

It follows that an S-estimator is the estimator $\hat{\boldsymbol{\beta}}$ that results in the s defined by (3.1) being minimal. The second condition on the objective function means that the associated score function will be redescending. A possible objective function to use is the one associated with the Tukey bisquare weight function, given in Table 2.1. The third condition is not strictly necessary, but is required to obtain a breakdown point of 50%. Often, K is chosen so that the resulting s is an estimator for σ when the errors really are normally distributed. In order to do this, K is set to $E_\Phi\left(\rho\left(u\right)\right)$, which is the expected value of the objective function if it is assumed that u has a standard normal distribution (Rousseeuw and Leroy 1987: 135, 139).

When using the Tukey bisquare objective function, Rousseeuw and Yohai (1984: 261) state that setting $a = 1.547$ satisfies the third condition, and so results in an S-estimator with 50% BDP. This will now be demonstrated. The Tukey bisquare objective function can be written as

$$\rho\left(u\right) \;=\; \begin{cases} \frac{u^2}{2} - \frac{u^4}{2a^2} + \frac{u^6}{6a^4} & \text{if } |u| \leq a \\ \frac{a^2}{6} & \text{if } |u| > a \end{cases}.$$

For $u \sim N\left(0,1\right)$, the probability density function is $f\left(u\right) = \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{u^2}{2}\right)$. Hence with $a = 1.547$,

$$\begin{aligned}
K = E_\Phi\left(\rho\left(u\right)\right) &= \int\limits_{-\infty}^{+\infty} \rho\left(u\right) f\left(u\right)\, dx \\
&= 2\int\limits_{0}^{+\infty} \rho\left(u\right) f\left(u\right)\, dx \quad \text{(since both } \rho\left(u\right) \text{ and } f\left(u\right) \text{ are symmetric functions)} \\
&= 2\int\limits_{0}^{+\infty} \rho\left(u\right) \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{u^2}{2}\right) dx \\
&= 2\int\limits_{0}^{1.547} \left(\frac{u^2}{2} - \frac{u^4}{2\times 1.547^2} + \frac{u^6}{6\times 1.547^4}\right)\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{u^2}{2}\right) dx \\
&\quad + 2\int\limits_{1.547}^{+\infty} \frac{1.547^2}{6}\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{u^2}{2}\right) dx \\
&= \sqrt{\frac{2}{\pi}}\int\limits_{0}^{1.547} \left(\frac{u^2}{2} - \frac{u^4}{4.786} + \frac{u^6}{34.365}\right)\exp\left(-\frac{u^2}{2}\right) dx \\
&\quad + \frac{1.547^2}{3}\int\limits_{1.547}^{+\infty} \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{u^2}{2}\right) dx \\
&= \sqrt{\frac{2}{\pi}}\int\limits_{0}^{1.547} \left(\frac{u^2}{2} - \frac{u^4}{4.786} + \frac{u^6}{34.365}\right)\exp\left(-\frac{u^2}{2}\right) dx + 0.798\times\left(1-\Phi\left(1.547\right)\right) \\
&= \sqrt{\frac{2}{\pi}}\times 0.189 + 0.798\times\left(1-0.939\right) \quad \text{(using R for windows and Maple 12)} \\
&= 0.199.
\end{aligned}$$

Now $\rho\left(1.547\right) = \frac{1.547^2}{6} = 0.399$ and so, ignoring the rounding errors, it is indeed the case that

$$\frac{K}{\rho\left(a\right)} = \frac{E_\Phi\left(\rho\left(u\right)\right)}{\rho\left(1.547\right)} = \frac{1}{2}.$$

Rousseeuw and Leroy (1987: 139) go into more detail about the breakdown point of S-estimators . Really, an S-estimator whose objective function satisfies the three conditions has BDP

$$\frac{\frac{n}{2} - p + 2}{n}$$

which only tends to 0.5 as $n \to \infty$. However, since there is only slight dependence on n it is said that the estimator has BDP 50%. They also point out that if the third condition were rewritten

$$\frac{K}{\rho\left(a\right)} = \alpha$$

where $0 < \alpha \leq \frac{1}{2}$, the resulting S-estimator would have a breakdown point tending to α as $n \to \infty$.

The asymptotic efficiency of this class of estimators depends on the objective function by which they are defined. Unfortunately, the tuning constants of this function cannot be chosen to give the estimator simultaneously high breakdown point and high asymptotic efficiency. Table 3.1, after Rousseeuw and Yohai (1984: 268, Table 3), provides a summary of the effect of the choice of a on the BDP and efficiency of an S-estimator defined using the Tukey bisquare weight function. Just as the low breakdown point of M-estimators hindered their usefulness, the low efficiency when high breakdown point is achieved makes S-estimators a bad choice of robust regression estimator. However, as will be discussed in the following sections, such a high breakdown estimator yields a very useful initial estimate within more complex robust regression estimation processes, as the resulting estimators inherit their high breakdown point (Rousseeuw and Leroy 1987: 143).

Table 3.1: Asymptotic relative efficiency of S-estimators for different values of BDP, making use of Tukey bisquare function

BDP	Efficiency	a	K
50%	28.7%	1.547	0.1995
45%	37.0%	1.756	0.2312
40%	46.2%	1.988	0.2634
35%	56.0%	2.251	0.2957
30%	66.1%	2.560	0.3278
25%	75.9%	2.973	0.3593
20%	84.7%	3.420	0.3899
15%	91.7%	4.096	0.4194
10%	96.6%	5.182	0.4475

3.2 MM-estimators

Yohai (1987: 644) describes the three stages that define an MM-estimator:

Stage 1 A high breakdown estimator is used to find an initial estimate, which we denote $\tilde{\boldsymbol{\beta}}$. The estimator need not be efficient. Using this estimate the residuals, $r_i\left(\tilde{\boldsymbol{\beta}}\right) = y_i - \boldsymbol{x}_i^T\tilde{\boldsymbol{\beta}}$, are computed.

Stage 2 Using these residuals from the robust fit and (3.1), an M-estimate of scale with 50% BDP is computed. This $s\left(r_1\left(\tilde{\boldsymbol{\beta}}\right),\ldots,r_n\left(\tilde{\boldsymbol{\beta}}\right)\right)$ is denoted s_n. The objective function used in this stage is labelled ρ_0.

Stage 3 The MM-estimator is now defined as an M-estimator of $\boldsymbol{\beta}$ using a redescending score function, $\psi_1(u) = \frac{\partial\rho_1(u)}{\partial u}$, and the scale estimate s_n obtained from Stage 2. So an MM-estimator $\hat{\boldsymbol{\beta}}$ is defined as a solution to

$$\sum_{i=1}^{n} \boldsymbol{x}_{ij}\psi_1\left(\frac{y_i - \boldsymbol{x}_i^T\boldsymbol{\beta}}{s_n}\right) = 0 \qquad j = 1,\ldots,p. \tag{3.2}$$

The objective function ρ_1 associated with this score function does not have to be the same as ρ_0 but it must satisfy:
i) ρ is symmetric and continuously differentiable, and $\rho(0) = 0$.
ii) There exists $a > 0$ such that ρ is strictly increasing on $[0, a]$ and constant on $[a, \infty)$.
iii) $\rho_1(u) \leq \rho_0(u)$.
A final condition that must be satisfied by the solution to (3.2) is that

$$\sum_{i=1}^{n} \rho_1\left(\frac{y_i - \boldsymbol{x}_i^T\hat{\boldsymbol{\beta}}}{s_n}\right) \leq \sum_{i=1}^{n} \rho_1\left(\frac{y_i - \boldsymbol{x}_i^T\tilde{\boldsymbol{\beta}}}{s_n}\right).$$

Often S-estimators are used in stage one (Andersen 2008: 57). Since the third stage is just M-estimation with an extra condition on the solution, the IRLS procedure described in Section 2.3 can be used to find a potential solution to (3.2), simply by keeping the scale estimate fixed at s_n in each iteration. The MM-estimation option in the R MASS package, again using the function *rlm,* uses an S-estimator with a Tukey bisquare objective function, having set $a = 1.548$ in the first two stages. Such an S-estimator has a BDP very close to 50%. In the final stage, the *rlm* function also uses the Tukey bisquare objective function.

3.3 Properties of MM-estimators

The first two stages of the MM-estimation process are responsible for the estimator having high breakdown point, whilst the third stage aims for high asymptotic relative efficiency. This is why ρ_0 and ρ_1 need not be the same, and why the estimator chosen in stage can be inefficient. Yohai (1987: 646, 654) proved when he first introduced MM-estimators that if in the first stage an estimator with 50% BDP is used, the final MM-estimator will also have 50% BDP. The high breakdown point of one such MM-estimator is demonstrated in Figure (3.1), in which the MM-estimator is the default MM-estimator provided by the R function *rlm.* The ability of the estimator to withstand multiple leverage points without breaking down is particularly impressive.

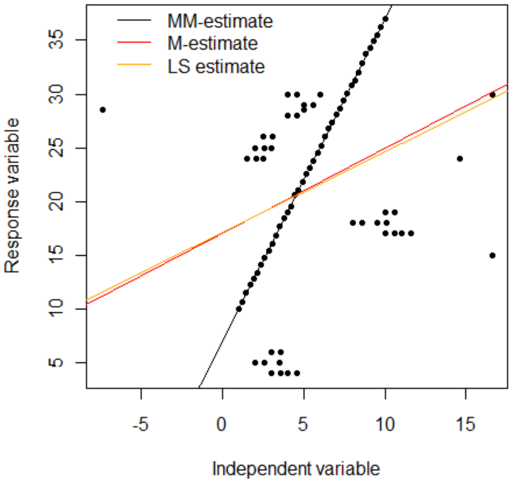


Figure 3.1: The plot contains 79 data. 40 have been generated using the model $y = 3x + 7 + \epsilon_i$, where $\epsilon_i \sim N\,(0,\,0.1)$. Even with the extra 39 data present, which are not from this model and include multiple bad leverage points, the MM-estimator has almost perfectly found the actual $\boldsymbol{\beta}$, returning the estimate $\hat{\boldsymbol{\beta}} = \begin{pmatrix} 7.151 \\ 2.998 \end{pmatrix}$.

Whilst the breakdown point of an MM-estimator depends on the choice of tuning constants in the first two stages, the asympotic relative efficiency of the MM-estimator is determined by the choice of the tuning constants in the third stage (Hadi and Simonoff 1993: 1265; Yohai 1987: 648). Therefore, unlike M-estimators, the breakdown point and relative efficiency of MM-estimators are independent of each other, so whilst the breakdown point remains fixed at 50%, the asymptotic relative efficiency can be set as close to 1 as desired (Bianco et al 2005: 519; Yohai 1987: 648). The default setting of the *rlm* function uses $a = 4.685$ in the last stage to obtain 95% asymptotic relative efficiency.

Another property that the MM-estimator can inherit from the estimator used in the first stage is the 'exact fit property' (Yohai 1987: 646). This property is another way of looking at the resistance of an estimator to outlying data. An estimator has the exact fit property if given any sample of n observations $\left(\boldsymbol{x}_i^T,\, y_i\right)$, where at least $n - \frac{n}{2} + 1$ of the observations exactly satisfy $y = \boldsymbol{x}_i^T\boldsymbol{\beta}$, the estimate obtained is exactly equal to $\boldsymbol{\beta}$ irrespective of the other observations (Rousseeuw and Leroy 1987: 60). The least squares estimator and M-estimators do not have this property, but the more resistant S-estimators do, and so in turn do MM-estimators using S-estimators in the first stage (see Figure 3.2 and Example 3.1).

Finally, if the estimator used in the first stage is regression and/or affine equivariant, the resulting MM-estimator will be too (Yohai 1987: 645). MM-estimators are also scale equivariant, which can be shown in the same manner as it was for M-estimators in Example 2.2.

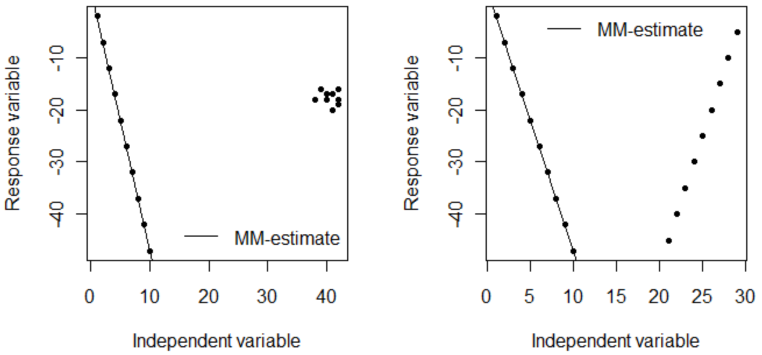


Figure 3.2: With one independent variable, if the majority of observations lie exactly on a line, an estimator with the exact fit property will always find this line. These two plots demonstrate that the MM-estimator will indeed find the this exact line even when $n - \frac{n}{2} - 1$ form a cluster of bag leverage points or lie exactly on another line.

Example 3.1 The Exact Fit Property of MM-estimators

i	1	2	3	4	5	6	7	8	9	10
x_{i2}	1	2	3	4	5	6	7	8	9	10
x_{i3}	2	1	6	10	3	7	5	8	9	4
x_{i4}	6	10	1	7	5	3	8	9	4	2
y_i	33	48	29	67	40	46	470	522	334	287

The first $i = 1, \ldots, 6$ of the ten observations in the table satisfy

$$y_i = (x_{i1},\, x_{i2},\, x_{i3},\, x_{i4}) \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix},$$

whereas the last $i = 7, \ldots, 10$ satisfy

$$y_i = (x_{i1},\, x_{i2},\, x_{i3},\, x_{i4}) \begin{pmatrix} 17 \\ 20 \\ -3 \\ 41 \end{pmatrix}.$$

Using the *rlm* function in R, the MM-estimate obtained for such a data set is exactly $\hat{\boldsymbol{\beta}} = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix}$ - it has not been influenced by the remaining four observations *at all.* On the other hand, the Huber M-estimate obtained for the same data is $\hat{\boldsymbol{\beta}} = \begin{pmatrix} -296.186 \\ 54.448 \\ 2.038 \\ 31.476 \end{pmatrix}$, which is not representative of either of the two relationships from which the observations came. This is clearly another worthwhile property for an estimator to have.

3.4 Examples of MM-estimation

Example 3.2

This example demonstrates the usefulness of a high breakdown estimator in identifying potentially troublesome observations in a multidimensional data set. The R data set *airquality* will be used. This set contains, for each of the 153 days between 1st May and 30th September 1973, observations for 6 air quality measures in New York. The response variable will be mean ozone in parts per billion from 1300 to 1500 hours at Roosevelt Island, denoted 'O', and the three independent variables will be:

1. Maximum daily temperature in degrees Fahrenheit at La Guardia Airport, denoted 'T'.
2. Average wind speed in miles per hour at 0700 and 1000 hours at LaGuardia Airport, denoted 'Wi'.
3. Solar radiation in Langleys in the frequency band 4000–7700 Angstroms from 0800 to 1200 hours at Central Park, denoted 'S'.

Of the 153 observations, 42 did not contain all the information required and so were omitted, meaning $n = 111$ and $p = 4$. Hence, $\boldsymbol{X}$ is a 111×4 matrix and $\boldsymbol{Y}$ and $\boldsymbol{\epsilon}$ are both 111×1 matrices:

$$\boldsymbol{X} = \begin{pmatrix} 1 & \mathrm{T}_1 & \mathrm{Wi}_1 & \mathrm{S}_1 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \mathrm{T}_{111} & \mathrm{Wi}_{111} & \mathrm{S}_{111} \end{pmatrix} \quad \boldsymbol{Y} = \begin{pmatrix} \mathrm{O}_1 \\ \vdots \\ \mathrm{O}_{111} \end{pmatrix} \text{ and } \boldsymbol{\epsilon} = \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_{111} \end{pmatrix}$$

The model being fitted is $\boldsymbol{Y} = \boldsymbol{X\beta} + \boldsymbol{\epsilon}$, where each ϵ_i is a random variable with mean 0 and variance σ^2. The least squares estimate for $\boldsymbol{\beta}$ is $\hat{\boldsymbol{\beta}}_{\mathrm{LS}} = \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{Y} = \begin{pmatrix} -64.342 \\ 1.652 \\ -3.334 \\ 0.060 \end{pmatrix}$.

Having loaded the data set and named the air quality measures for the remaining 111 observations O, T, Wi and S, the *rlm* function is applied to the data in R to find an MM-estimate as follows:

```
library(MASS)
rlm(O~T+Wi+S,method="MM")
```

To reiterate, this produces an MM-estimate using an S-estimator with the Tukey bisquare objective function in the first stage, with the default tuning constant $a = 1.548$ to gain 50% BDP, which provides an appropriate scale estimate s_n for the second stage, and in the third stage the Tukey bisquare objective function with $a = 4.685$ is used to obtain 95% asymptotic relative efficiency. The command produces the output:

```
Call: rlm(formula = O ~ T + Wi + S, method = "MM")
Converged in 7 iterations
Coefficients:
(Intercept)            T           Wi            S
-85.17264411   1.78202541  -2.25803571   0.04480061
Degrees of freedom: 111 total; 107 residual
Scale estimate: 17.2
```

So the MM-estimate is

$$\hat{\boldsymbol{\beta}}_{\mathrm{MM}} = \begin{pmatrix} -85.173 \\ 1.782 \\ -2.258 \\ 0.045 \end{pmatrix}$$

and s_n is 17.2. The output also shows that from the initial S-estimate seven iterations of the IRLS procedure were carried out before convergence was reached. One can obtain this initial S-estimate by fixing the maximum number of iterations of the IRLS procedure to 0, using the argument `maxit=0` in the *rlm* function. In this case the resulting S-estimate is

$$\hat{\boldsymbol{\beta}}_{\mathrm{S}} = \begin{pmatrix} -84.810 \\ 1.779 \\ -2.300 \\ 0.045 \end{pmatrix}.$$

It will now be verified that this initial estimate and $s_n = 17.2$ satisfy (3.1): $\frac{1}{n}\sum_{i=1}^{n} \rho\left(r_i/s\right) = E_{\Phi}\left(\rho\left(u\right)\right)$. Since $\rho\left(u\right)$ is the Tukey bisquare objective function with $a = 1.548$, it can be shown that $E_{\Phi}\left(\rho\left(u\right)\right) = 2.00$ (to 3 significant figures), using the same method of calculation as in Section 3.1. Using R as shown in Appendix B, it can then be confirmed that

$$\frac{1}{n}\sum_{i=1}^{n} \rho\left(\frac{r_i}{s}\right) = \frac{1}{111}\sum_{i=1}^{111} \rho\left(\frac{\mathrm{O}_i - (1,\, \mathrm{T}_i,\, \mathrm{Wi}_i,\, \mathrm{S}_i)\,\hat{\boldsymbol{\beta}}_{\mathrm{S}}}{17.2}\right) = 2.00 \quad \text{(to 3s.f.)}\,.$$

The *rlm* function is used to find the Huber M-estimate for $\boldsymbol{\beta}$, as follows:

```
rlm(O~T+Wi+S,method="M",psi=psi.huber)
```

giving

$$\hat{\boldsymbol{\beta}}_{M} = \begin{pmatrix} -78.434 \\ 1.745 \\ -2.644 \\ 0.049 \end{pmatrix}.$$

Both $\hat{\boldsymbol{\beta}}_{\mathrm{MM}}$ and $\hat{\boldsymbol{\beta}}_{\mathrm{M}}$ are noticeably different to $\hat{\boldsymbol{\beta}}_{\mathrm{LS}}$, indicating that the least squares estimator is probably being influenced by outlying data. They are also quite different to each other, suggesting that there could perhaps be leverage points affecting the M-estimator. By studying the final weights given to each observation in the 7th iteration of the IRLS process, such influential cases can be identified. The final weights are extracted using the command `rlm(O~T+Wi+S,method="MM")$w`.

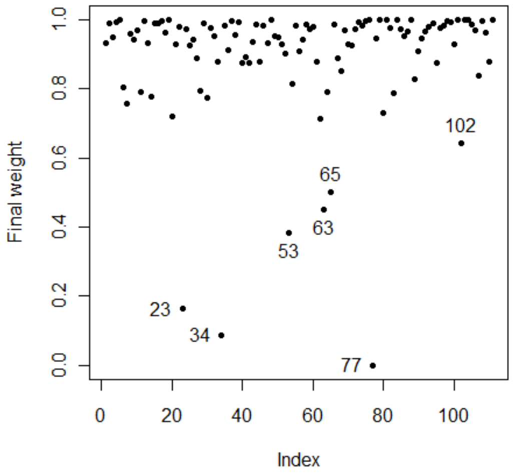


Figure 3.3: Index plot of weights associated with the MM-estimate for air quality data.

Figure 3.3 allows seven points to be identified as having low weights compared to the majority of observations. In fact, the 77th observation has been assigned weight 0. Removing these cases and reapplying the least squares estimator to the remaining 104 observations produces a new estimate for $\boldsymbol{\beta}$,

$$\hat{\boldsymbol{\beta}}_{\widetilde{LS}} = \begin{pmatrix} -83.824 \\ 1.732 \\ -1.993 \\ 0.042 \end{pmatrix},$$

which is much closer to the MM-estimate. This suggests that these 7 points, which represent around 6% of the sample, were together influencing the least squares estimator, and ought to be studied further and explained before a final model can fitted.

Example 3.3

This example relates to the example in Section 2.6. Applying the *rlm* function to the US Judge Ratings data, the MM-estimate $\hat{\boldsymbol{\beta}} = \begin{pmatrix} 3.225 \\ 0.632 \end{pmatrix}$ is obtained. The IRLS procedure converged on this estimate after 11 iterations, using the M-estimate of scale $s_n = 0.254$ in each one. Figure 3.4 compares this MM-estimate to the M-estimate of Section 2.6.

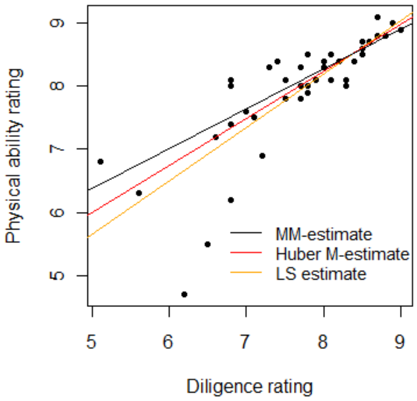


Figure 3.4: Plot comparing the ratings given to judges for their diligence and their physical ability, demonstrating the better fit of the MM-estimate to the majority of the data.

Since the MM-estimator has a 50% breakdown point, and hence allows for leverage, one would expect that if the response and dependent variable were swapped, the estimator would perform equally well. However, Figure 3.5 shows that this is not the case; the MM-estimate is not a good fit at all.

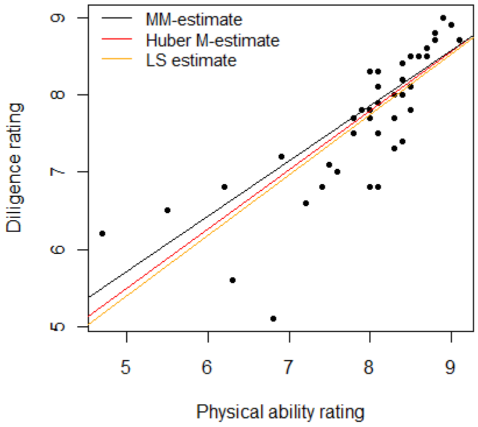


Figure 3.5: Plot demonstrating the difference in the performance of all three estimates with the swapped variables.

Furthermore, Figure 3.6 demonstrates that despite the estimator's impressive properties, it can break down at levels of contamination far lower than 50%. In this instance the data set was deliberately constructed to cause the MM-estimate to break down, yet considering the apparent popularity of the estimator it was surprising that this was so simple to achieve. So why is it possible to break down the estimator here if the breakdown point is 50% and there is less contamination than this? Clearly, the MM-estimator has limitations which have not been addressed so far in this report; Chapter 4 attempts to answer this question.

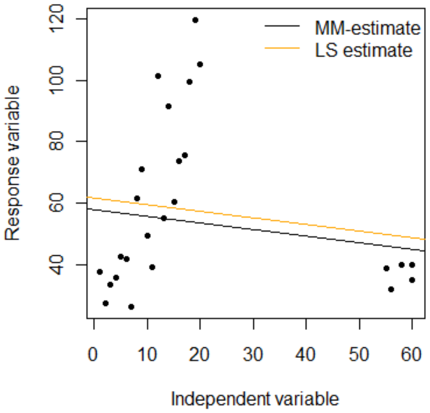


Figure 3.6: Twenty of the twenty five data fit a linear relationship well, with the remaining five data forming an cluster of outliers with high leverage. Despite there being only 20% contamination, the MM-estimator has broken down and gives an estimate that is no better than the LS estimate.

Chapter 4

Sources of problems for the MM-estimator

In order to better understand the MM-estimator, this chapter scrutinizes the breakdown and efficiency properties first defined in Chapter 1. In doing so, it is realised why high breakdown and high relative efficiency are not sufficient conditions for a reliably robust estimator.

4.1 Breakdown point

Hadi and Simonoff (1993: 1265) point out that a lot of literature on robust statistics from the 1980s focussed on high breakdown estimators. Likewise, this report has so far concentrated on finding a high breakdown estimator. However, Rousseeuw and Leroy (1987: 154) emphasise that a high breakdown point is a necessary but not *sufficient* condition for a good robust estimator. As Figure 3.6 demonstrated so clearly, purely having 50% BDP does not mean you have a reliable estimator. Olive and Hawkins (2008: 5) actually describe it as 'folklore' that high breakdown estimators are good.

The start of the problem lies in the fact that breakdown point considers only whether the influence of outlying data can become *infinite*. As Bianco *et al* (2005: 511-512) explain, if an estimator has a breakdown point ε then for any fraction of outliers smaller than ε the estimate remains in a closed bounded set, but as the fraction of outliers approaches ε, this set becomes very large. This means that although the estimate cannot break down, it can get quite far from the true $\boldsymbol{\beta}$. How robust an estimate really is to outliers somewhat depends on the rate at which this set grows as the fraction of outliers increases.

This motivates the consideration of a new robustness property - how much impact can one observation really have on an estimate? The 'influence function' goes some way towards answering this. Rousseeuw and Leroy (1987: 187-188) describe the relationship between the concept of influence function and breakdown point using the metaphor of a stone hanging on the end of a beam; whilst breakdown point corresponds to the weight of the smallest stone which will cause the beam to snap, the influence function considers how much smaller stones will cause the beam to bend.

Bich (2001: 673) describes the influence function as depicting the effect on an estimator of an observation that does not fit the assumed distribution of the data, which occurs with a probability ε arbitrarily close to zero. This sort of contamination is referred to as an 'infinitesimal perturbation'. Thus, another way to consider the influence function is that it reflects the bias caused by such an infinitesimal perturbation (Rousseeuw and Leroy 1987: 186). As such, the influence function can be thought of as an indicator of the 'bias robustness' of an estimator. The infinitesimal perturbation can also be interpreted as contamination in a sample of *size* ε, with $\varepsilon \to 0$.

For the estimator $\boldsymbol{T}$, applied to data with the distribution F, subject to contamination of size ε, with the distribution $\delta_{y,\boldsymbol{x}}$, concentrated at the point $(y, \boldsymbol{x})$, the estimator's asymptotic bias is defined as

$$\mathbf{b}\left(\boldsymbol{T}, F, \varepsilon, y, \boldsymbol{x}\right) = \boldsymbol{T}\left(\left(1-\varepsilon\right)F + \varepsilon\delta_{y,\boldsymbol{x}}\right) - \boldsymbol{T}\left(F\right),$$

and the influence function for the estimator at $\boldsymbol{x}$ can be defined as the limit as $\varepsilon \to 0$ of the asymptotic bias, standardised by the size of contamination:

$$\mathrm{IF}\left(\boldsymbol{T}, F, \varepsilon, y, \boldsymbol{x}\right) = \lim_{\varepsilon \to 0} \frac{\mathbf{b}\left(\boldsymbol{T}, F, \varepsilon, y, \boldsymbol{x}\right)}{\varepsilon}$$

(Yohai 1987: 650). This function is considered to be a function of $\boldsymbol{x}$ (Bich 2001: 673).

Surprisingly, considering the popularity of the MM-estimator, Yohai (1987: 650) shows that for MM-estimators the influence function is not bounded, meaning that even with a large sample of data, a small amount of contamination could have an arbitrarily large effect on the estimator. However, the contamination proportion ε tending to zero is unrealistic for the sort of data to which the estimator will be applied; rather, there will generally be a 'small' postitive proportion of contamination. Therefore, Yohai (1987: 650) deemed it more meaningful to consider the nature of the ε-influence curve for the estimator. This is simply the standardised bias,

$$\frac{\mathbf{b}\left(\boldsymbol{T}, F, \varepsilon, y, \boldsymbol{x}\right)}{\varepsilon},$$

with $\varepsilon > 0$ again representing a proportion of contamination.

Despite the rather worrying property of an unbounded influence curve, Yohai (1987: 650) states that this ε-influence curve *is* bounded for any $\varepsilon < 0.5$, which is perhaps why authors such as Andersen (2005) mistakenly claim that the estimator has a bounded *influence* curve. This ε-influence curve boundedness corresponds directly to the MM-estimator's 50% finite sample BDP. It is this curve which can be used to explain the impact of erroneous observations on the estimator for contamination levels lower than 50%; although the curve is bounded here, this will not prevent it from producing rather inconveniently large values for lower values of ε, thus demonstrating that perhaps it is not, after all, as robust an estimator as hoped. Example 4.1 demonstrates a consequence of this.

Example 4.1

Fifteen data were first generated using the model $y_i = (1,\, x_i)\begin{pmatrix} 7 \\ 5 \end{pmatrix} + \epsilon_i$ where $\epsilon_i \sim N\left(0, 6\right)$ with x ranging from 1 to 15. Then a random sample of five of these x-values were replaced so that they came from a Cauchy distribution with probability distribution function $f(x) = \frac{1}{50\pi\left(1+\left(\frac{x}{50}\right)^2\right)}$. These five data represent a contamination proportion of $\varepsilon = \frac{1}{3}$, and indeed have produced 5 visibly outlying data. The x-values were contaminated rather than the y, to enable the presence of leverage points in the contaminated sample.

The dashed line in the plot indicates the line $y = 7 + 5x$, which is the slope corresponding to the real $\boldsymbol{\beta}$ from which most of the data originates. The solid line represents the MM-estimate for this $\boldsymbol{\beta}$, which is $\hat{\boldsymbol{\beta}} = \begin{pmatrix} 24.8 \\ 3.24 \end{pmatrix}$. Although this is admittedly fairly different to the real $\boldsymbol{\beta}$, a slope estimate 3.24 is not too far from the actual slope parameter 5. Indeed, Figure 4.1 demonstrates that this estimate is *almost* a good fit to the majority of the data. As such it would be unfair to say that the estimator had completely broken down, especially when it is compared to the least squares estimator's attempt (the dotted grey line). The MM-estimator has definitely been somewhat biased by the outlying data though, as it no longer represents the uncontaminated data well.

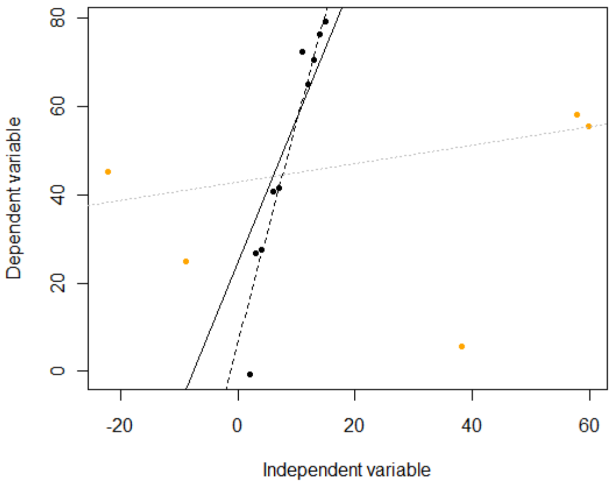


Figure 4.1: Plot demonstrating the MM-estimator (solid line) having been subject to bias and the LS estimator (dotted grey line) breaking down when applied to a contaminated data set for which the majority of the data fit relationship represented by the dashed line. The contaminated set of data was constructed using the code given in Appendix B.

Whilst the weakness of the concept of breakdown point goes some way to explaining why the MM-estimator performs a little less well than expected, it has not been able to explain why in Figure 3.6 the estimator was not just subject to large bias, but actually completely broke down. The estimator still needs more careful consideration; the following sections continue to try to explain this behaviour.

4.2 The consequences of high relative efficiency

Estimators with high relative efficiency have been sought after because, for data from populations for which the error distribution really is $\epsilon_i \sim N\left(0, \sigma^2\right)$, such estimators will perform very similarly to the least squares estimator. Since the least squares estimator is the minimum variance estimator for data of this type, it makes sense to hope for a robust estimator to mimic its behaviour. However, this creates a problem; designing the estimator to encourage it to behave like the least squares estimator in the presence of normal errors also encourages it to in situations where there are outliers present and the errors are far from normal (Hadi and Simonoff 1993: 1268). For example, in Figure 4.2 the high efficiency MM-estimator is performing far more similarly to the LS estimator than the low efficiency S-estimator of the same 50% breakdown point. This lower efficiency estimator has definitely performed much better.

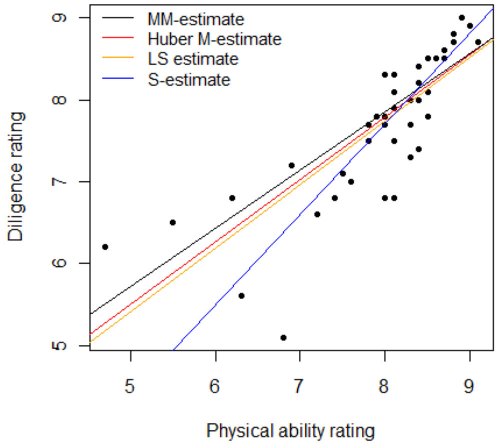


Figure 4.2: The US Judge Ratings sample with the variables swapped as discussed in Example 3.3.

Indeed, when he introduced the estimator, Yohai (1987: 648) discussed the dangers of choosing a high tuning constant in the third stage to gain 95% asymptotic relative efficiency. Whilst the MM-estimator's breakdown point would remain fixed, as it is determined in the first two stages and independent of the third, the estimator would actually end up being more sensitive to outliers and so less robust than it would have been had a lower efficiency been chosen. Despite this, as mentioned in Section (3.3), the default setting of the *rlm* function in R is to use the tuning constant for which the estimator gains 95% asymptotic relative efficiency. One can specify other asymptotic relative efficiency levels for an MM-estimator using the Tukey bisquare objective function in the third stage, from the information in Table 3.1 of Section 3.1, which indicates the level of asymptotic relative efficiency that various values of tuning constant will result in. For example, if the tuning constant is set to $a = 3.42$ the resulting MM-estimator will have 84.7% asymptotic relative efficiency.

Figure 4.3 demonstrates an example of the superior performance of this lower efficiency MM-estimator to the usual 95% efficient MM-estimator. The sample shown was constructed in a similar manner to that of Example 4.1, containing 20 data of which 12 have normally distributed errors, and the remaining 8 (shown in blue in the plot) have been contaminated using a Cauchy distribution (see Appendix B).

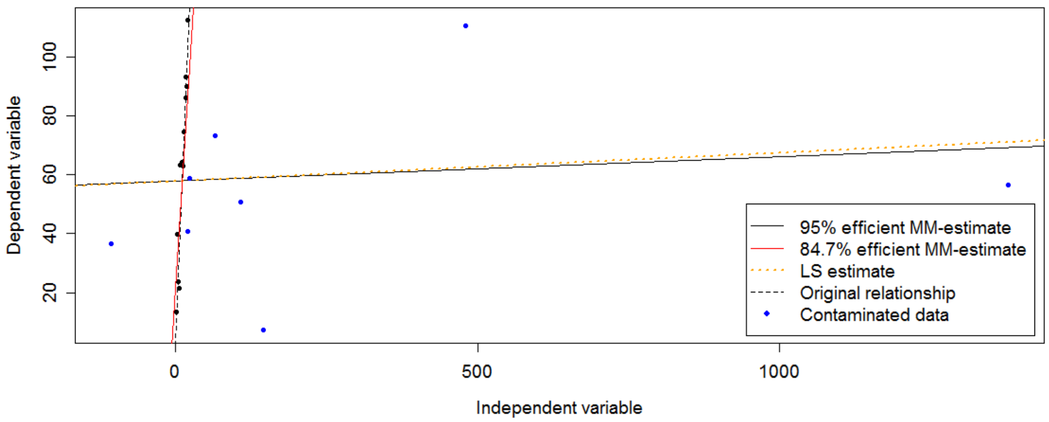


Figure 4.3: With only 84.7% asymptotic relative efficiency the MM-estimator performs very well. However, the MM-estimator with 95% asymptotic relative efficiency has broken down and its performance is very similar to the least squares estimator.

Example 4.2 Aircraft

The data used in this example correspond to three of the variables of the aircraft data set, given in Appendix A, which can be found in both Rousseeuw and Leroy (1987: 154) and the 'robustbase' package in R. The data represent the aspect ratio, lift-to-drag ratio and weight in pounds of 23 aircraft built between 1947 and 1949. In an attempt to model the aspect ratio on the lift-to-drag ratio and the weight of the planes, the least squares estimator and MM-estimates of decreasing asymptotic relative efficiency, constructed using the *rlm* function in R, were applied to the data. The differences in the size of the corresponding MM-estimates and the least squares estimate,

$$\| \hat{\boldsymbol{\beta}}_{\text{MM}} - \hat{\boldsymbol{\beta}}_{\text{LS}} \|$$

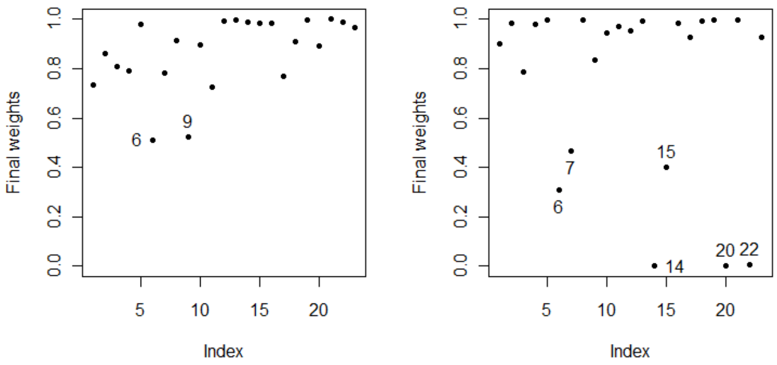
were calculated in order to compare their performance to that of the least squares estimator. The code used in R is given in Appendix B. Table 4.1 summarises the findings.

Table 4.1

Estimator	$\hat{\beta}_{\text{intercept}}$	$\hat{\beta}_{\text{lift-to-drag}}$	$\hat{\beta}_{\text{weight}}$	Difference between MM and LS estimates $\| \hat{\boldsymbol{\beta}}_{\text{MM}} - \hat{\boldsymbol{\beta}}_{\text{LS}} \|$
LS estimator	5.512	0.054	-0.0000970	-
95% efficient MM-estimator	5.495	0.041	-0.0000948	0.021
91.7% efficient MM-estimator	5.486	0.038	-0.0000941	0.031
84.7% efficient MM-estimator	5.131	1.653	-0.0002932	1.644
75.9% efficient MM-estimator	5.036	1.833	-0.0003131	1.842
66.1% efficient MM-estimator	4.935	1.981	-0.0003285	2.012

It is clear that the MM-estimators with 95% and 91.7% asymptotic relative efficiencies are performing very similarly to the least squares estimator, with only a slight trend of decreasing size in the three components of $\hat{\boldsymbol{\beta}}$. This is reflected in the small and similar values in the final column of Table 4.1. However, when the asymptotic relative efficiency was reduced to 84.7% there was a sudden large increase in the size of both $\hat{\beta}_{\text{lift}-\text{to}-\text{drag}}$ and $\hat{\beta}_{\text{weight}}$, with the former becoming over 40 times larger. In fact, in reducing the relative efficiency from 91.7% to 84.7%, the difference between the MM-estimate and the least squares estimate increases by a factor of over 50. Despite this sudden change, the three lower-efficiency MM-estimates seem to perform similarly to each other. This is a strong indication that there are influential observations in the set biasing the LS estimator, and that the first two MM-estimators are performing like the LS estimator due to their very high relative efficiency.

Figure 4.4: Index plots of final weights in the MM-estimation procedure.



(a) 91.7% efficient MM-estimator.

(b) 84.7% efficient MM-estimator.

As the index plots of the weights show, the final weights for the less efficient estimator were generally closer to one, and certainly the majority lie between 0.9 and 1. There are six unusually low weights immediately visible, three of which have weight zero, certainly indicating possible influential data. The weights for the more efficient estimator were fairly evenly spread between 0.7 and 1, with only two points slightly lower. Only one of these is one amongst the 6 cases identified in Figure 4.4b, and there is no clear indication that the three points given weight 0 by the less efficient estimator could be causing problems at all. To see how strongly these three cases were affecting the least squares and MM-estimators with high asymptotic relative efficiencies, the data were removed and the estimators re-applied. Table 4.2 shows the results. The six estimators are now far more consistent with each other.

Table 4.2

Estimator	$\hat{\beta}_{\text{intercept}}$	$\hat{\beta}_{\text{lift-to-drag}}$	$\hat{\beta}_{\text{weight}}$
LS estimator	5.402	1.331	-0.000261
95% efficient MM-estimator	5.154	1.636	-0.000292
91.7% efficient MM-estimator	5.070	1.773	-0.000307
84.7% efficient MM-estimator	4.940	1.977	-0.000328
75.9% efficient MM-estimator	4.844	2.010	-0.000329
66.1% efficient MM-estimator	4.783	1.992	-0.000324

Although it would seem that the problems with the MM-estimator could be solved by simply setting the asymptotic relative efficiency to be around 70-85%, this sadly still does not solve the problem in all situations. For example, in Figure 4.5, for the sample previously observed in Figure 3.6, the MM-estimator with a relative efficiency of just 91.7% has also broken down in the presence of just 20% contamination, despite its lower efficiency. Clearly, there are other factors at play.

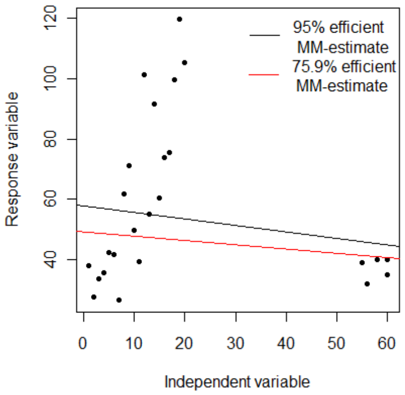


Figure 4.5

4.3 A theoretical estimator

Perhaps the most severe cause of these unexpected ‘lapses’ in the MM-estimator’s performance is that the theory behind its impressive breakdown point and relative efficiency is based on *asymptotic* distributions and large sample theory (Maronna and Yohai 2010: 3168). As Olive and Hawkins (2008: 2, 9) point out, whilst the theory behind high breakdown estimators may be elegant, the theoretical performance of an estimator and its actual performance can be entirely different. They warn that large sample theory is particularly irrelevant if the estimator is not even realistically implementable on large samples.

Unfortunately, a less desirable property which MM-estimators inherit from S-estimators used in stage one is that as n and p become large, the intensive computation necessary to calculate estimates becomes extremely time-consuming (Frosch Møller *et al* 2005: 553; Hadi and Simonoff 1993: 1265). For MM-estimation to be a practical option, fast, approximate algorithms based on subsampling the data have had to be found to compute the S-estimates (Bianco *et al* 2005: 521). Whilst enabling their application, Olive (2008: 228) warns of the dangers of using such approximate algorithms to calculate high breakdown estimates; the algorithms act as estimators for the estimator and as such are subject to more uncertainty and potentially have lower breakdown points than the original estimator. Furthermore, he points out that often no distinction is even drawn between the theoretical estimator, and the algorithm used to estimate it.

Only the *application* of an estimator to data sets can really provide the information needed to better understand the practical properties of the estimator and its estimation process. Whilst the examples based on real data sets in this chapter have given indications of the quality of the estimators' finite sample performances, it has not really been possible to say for certain how 'well' the estimators were performing because the actual parameter values were not known. Crucially, for most real data sets neither the true distribution nor the contamination level are known; it is therefore very difficult to understand exactly what conditions the estimators might be working under. Consequently, it is certainly unfair to draw general conclusions on the differences between estimators' practical and theoretical performances using such examples.

Statisticians such as Hadi and Simonoff (1993) and Maronna and Yohai (2010) have therefore performed simulation studies using sample sizes beneath 100. Simulated data has the advantage that the true structure of the data sets is known, and the data can be specifically constructed to focus on individual aspects of performance. The next chapter exploits this technique in an effort to gain greater insight into the changing performance of MM-estimators with small n, where asymptotic results do not hold. Firstly Maronna and Yohai's (2010) simulation study will be discussed, and then two further simulation studies performed.

Chapter 5

Simulation studies of the MM-estimator

5.1 Maronna and Yohai's simulation study

The results of Maronna and Yohai's study indicate that often, in practical applications, the estimator has not only a considerably lower breakdown point, but also a lower relative efficiency than the theory would suggest. In addition the scale estimator of stage two is subject to bias. Together, these points explain why it is possible to cause the MM-estimator to break down for contamination levels far lower than its theoretical breakdown point.

Within the MM-estimation procedure, the methodology for the scale estimation and selection of a tuning constant to fix relative efficiency levels is based on asymptotic results. Maronna and Yohai (2010: 3168) explain that these results are based on the assumption that p is fixed, and $n \to \infty$. They warn however that for many real data sets $\frac{p}{n}$ is 'large'. For the sample shown in Figure 4.5, where the MM-estimate has not performed as expected $\frac{p}{n} = \frac{2}{25} = 8\%$.

In finite samples where $\frac{p}{n}$ is small enough the theoretical results *will* hold, but for data sets where $\frac{p}{n}$ is too big two problems arise. Firstly, the scale estimator is affected by downwards bias and so will underestimate the true error scale. Secondly, the actual relative efficiency is much lower than the desired one because the tuning constant has been chosen according to asymptotic theory and is simply unsuitable (Maronna and Yohai 2010: 3168, 3169, 3171-3172). As a result of their simulations, alterations to the MM-estimator are suggested to correct the scale estimate in order to counteract this behaviour, with the specific advice depending on quite how large $\frac{p}{n}$ is. Furthermore, reasons are suggested for this loss of efficiency.

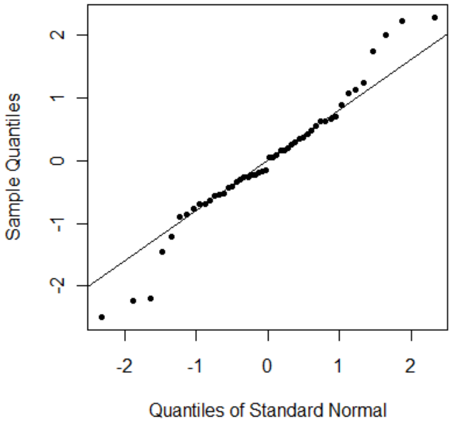


Figure 5.1: Gaussian quantile plot of the residuals from an MM-estimate for $\boldsymbol{\beta}$ obtained from normally distributed data with $n = 50$ and $p = 10$

One such explanation is that even for uncontaminated normally distributed data, when $\frac{p}{n}$ is large the distribution of a sample can be quite far from normal. Figure 5.1 demonstrates an example of this. It shows the Gaussian quantile plot of the residuals of a sample for which $\boldsymbol{Y} = \boldsymbol{X\beta} + \boldsymbol{\epsilon}$ with $n = 50$, $p = 10$, $\boldsymbol{\beta} = \mathbf{0}$ and the errors have a standard normal distribution. The $\boldsymbol{X}$ matrix was generated such that the first column consisted of n ones, and the nine remaining columns were each the numbers 1,. . ., 50 in differing orders. The standard *rlm* MM-estimator was used to calculate the residuals. The quantile plot shows that a greater proportion of the residuals are 'large' than expected from a normal distribution; the curve is far steeper at each extreme, indicating that the residuals' distribution is heavy-tailed compared to a normal distribution. As a result, a larger proportion of the observations will be down-weighted than the standard asymptotic theory suggests, and it is to this that Maronna and Yohai (2010: 3172) attribute the blame for the decreased efficiency of the estimator.

To show that the effect is not simply caused by n being too small, and that decreasing the size of $\frac{p}{n}$ improves this heavy-tailedness, the response variable was recalculated as $\boldsymbol{Y} = \boldsymbol{X\beta} + \boldsymbol{\epsilon}$ with again $n = 50$, $\boldsymbol{\beta} = \mathbf{0}$, and the errors sampled from a standard normal distribution, but now the $\boldsymbol{X}$ matrix consisted of a column of n ones and just one of the nine other columns of the original $\boldsymbol{X}$. In this way p was reduced to 2, thereby reducing $\frac{p}{n}$ to $\frac{2}{50} = 0.04$. The Gaussian quantile plots generated from the residuals of the new MM-estimates obtained using each of the nine possible choices for the second column are shown in Figure 5.2. As expected, the residuals are much more normally distributed, so as Maronna and Yohai (2010: 3168) assert, the efficiency of the estimator will have been affected less.

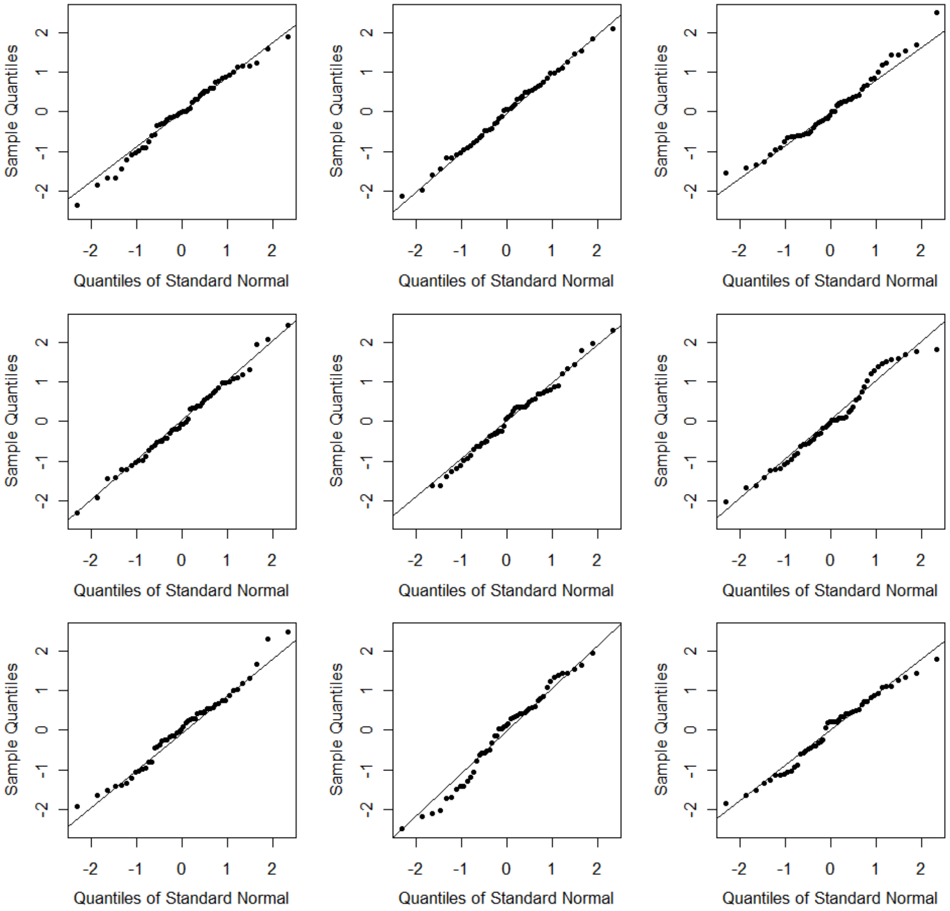


Figure 5.2: Gaussian quantile plots for each of the nine explanatory variables in the sample used to construct Figure 5.1.

5.2 The effect of varying $\frac{p}{n}$ on MM-estimators' relative efficiencies and scale estimates

In order to study the effect of varying $\frac{p}{n}$ on the efficiency and scale estimate of MM-estimators, the data in this simulation were constructed in such a way that the assumptions for the least squares estimator were met. Therefore, the data was obtained using the model $\boldsymbol{Y} = \boldsymbol{X\beta} + \boldsymbol{\epsilon}$ where $\epsilon_i \sim N(0, 1)$, $\boldsymbol{X}$ is a $n \times p$ matrix, $\boldsymbol{\beta}$ is a $p \times 1$ vector, and $x_{i1} = 1$ for $i = 1, \ldots n$, to give an intercept term. It was set that $\boldsymbol{\beta} = \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 1 \end{pmatrix}$ and each $n \times 1$ column of $\boldsymbol{X}$ after the first column of n ones was generated as a sample of length n from a normal distribution with mean zero, and standard deviation 20. The least squares estimator and two MM-estimators, of nominal relative efficiencies 95% and 75.9% were then applied to the data. The least squares estimates were obtained via the *lm* function in R. The *rlm* function in R was used to find the two MM-estimates, first setting the tuning constant used in the third stage to the default $a = 4.685$ to make the function act as a 95% relative efficiency MM-estimator, and then setting $a = 2.973$ to make the function act as a 75.9% relative efficiency MM-estimator. $M = 1000$ replications were performed for each combination of n and p with p ranging from 1 to 10, and n taking the values 10, 20, 30, 40, 50 and 200.

Following Bianco *et al* (2005: 521-522), for each combination of n and p the TMSE (total mean square error) for each of the three estimators over the 1000 replications was calculated, by

$$\text{TMSE} = \frac{1}{M} \sum_{i=1}^{M} \sum_{j=1}^{p} \left(\hat{\beta}_{ij} - \beta_{ij} \right)^2 .$$

Then, as in the simulation of Bianco *et al*, the relative efficiency of the MM-estimators was calculated as the ratio of the TMSE of the least squares estimator and the TMSE of the MM-estimator. Also, for each set of 1000 replications of a particular n and p combination, the average scale estimate of the two MM-estimators was found, using

$$\bar{s} = \frac{1}{M} \sum_{i=1}^{M} s_i$$

where each s_i represents the scale estimate produced by the MM-estimation procedure applied to the ith sample with that combination of p and n. The scale estimates produced by the two MM-estimators were the same as each other since only the tuning constant of the third stage had been altered and the scale estimates are obtained before this stage. Since the samples were drawn from uncontaminated normal distributions, the scale estimator within the MM-estimation procedures was an estimator for σ. The average estimate for σ from the models fitted by the least squares estimator over the 1000 samples was also recorded for comparison. In this simulation $\sigma = 1$, so one would have hoped that all three estimators would produce values fairly close to one for their scale estimates. Finally, the combination $n = 10$, $p = 10$ was not used, as the least squares scale estimator involves $n - p$ on the denominator and hence is not defined, and when $n = p$ the MM-estimation procedure will not work as the initial S-estimate and scale estimate cannot be found.

An example of the code used is given in Appendix B. Table 5.1 shows the results of the simulation and these calculations. Table 5.2 shows the results of a repetition of this simulation and these calculations, which will be considered shortly.

Results

Immediately noticable in Table 5.1 is that for both MM-estimators, the actual relative efficiencies are generally much lower than the nominal ones. This supports the findings of Maronna and Yohai that the relative efficiencies of the MM-estimator decrease significantly for larger $\frac{p}{n}$. Furthermore, although p and n change quite significantly, it can be seen from the table that for similar values of $\frac{p}{n}$ the actual relative efficiency of the estimator are fairly constant. For example, looking at $\frac{p}{n} = 0.2$ for the (n, p) combinations $(6, 30)$, $(8, 40)$ and $(10, 50)$ the relative efficiencies of the two MM-estimators were 88% and 58%, 89% and 56% and 88% and 57% respectively. Generally though, the case for $n = 10$ doesn't quite fit this pattern for $p = 1, \ldots 5$ and certainly, something peculiar has happened to the relative efficiencies when $\frac{p}{n} > 0.5$. The average scale estimates also dropped dramatically, to the order of 0.0001 when p became larger than n. The relative efficiencies of between 0 and 1% indicate that the TMSE for each of the

MM-estimators for $n = 10$ and $p = 6$, 7, 8, and 9 were over 100 times as large as those of the least squares estimator. Looking in more detail at the simulation results showed that whilst the average parameter estimates were fairly reasonable, the huge TMSEs were the result of enormous amounts of variation in the MM-estimators' intercept estimates, $\hat{\beta}_0$; for both the 95% and 75.9% MM-estimators the value of

$$\frac{1}{M}\sum_{i=1}^{M}\left(\hat{\beta}_{i0}-\beta_{i0}\right)^2=\frac{1}{1000}\sum_{i=1}^{1000}\left(\hat{\beta}_{i0}\right)^2$$

reached as high as the hundreds.

The simulation was repeated to gain an idea of the accuracy of these estimated relative efficiencies and level of scale underestimation. Comparing the new results (Table 5.2) with Table 5.1 reveals considerable variation in some of the estimated relative efficiencies, despite the large number of replications. Whilst most of the differences are in the region of 3%, when $n = 10$ the differences reach as large as 17% for the MM-estimator with nominal 95% relative efficiency.

Despite this, the general trend in decreasing efficiency with increasing $\frac{p}{n}$, seen in the first simulation, is still very prominent in the second run of the simulation with 1000 samples. In both of these simulations, for the smallest values of $\frac{p}{n}$ investigated, the estimated relative efficiencies for the two estimators became reasonably close to the asymptotic relative efficiency. Furthermore the relative efficiencies for combinations of n and p that have similar values of $\frac{p}{n}$ remain fairly consistent. The scale estimates are very consistent however, with both runs of the simulation demonstrating that for large $\frac{p}{n}$ there is significant underestimation of the scale on average, with the effect worsening as $\frac{p}{n}$ grows larger. Whilst even the least squares estimator appears to underestimate the scale as $\frac{p}{n}$ increases beyond 0.5, for all the values studied smaller than this the least squares scale estimate is consistently very close to the correct value of 1.

Table 5.1: Approximate relative efficiencies and average scale estimates of the two MM-estimators from the first simulation with 1000 samples of data for each p and n combination

(a) Relative efficiency of the MM-estimator with nominal 95% relative efficiency

95% efficient MM-estimator		p								
		2	3	4	5	6	7	8	9	10
n	10	85	80	69	63	0	0	0	1	-
	20	92	90	89	87	84	82	69	63	59
	30	94	92	92	91	88	85	90	85	80
	40	94	94	94	93	97	91	89	88	86
	50	94	96	93	94	93	93	91	92	88
	200	94	96	94	94	93	97	94	96	96

75.9% efficient MM-estimator		p								
		2	3	4	5	6	7	8	9	10
n	10	65	58	52	47	0	0	0	1	-
	20	68	68	63	59	52	50	42	43	38
	30	74	69	65	65	58	54	55	52	47
	40	74	73	71	70	72	59	56	55	54
	50	76	76	70	71	68	67	63	67	57
	200	76	76	76	74	73	79	74	80	78

(b) Relative efficiency of the MM-estimator with nominal 75.9% relative efficiency

		p								
		2	3	4	5	6	7	8	9	10
n	10	0.97	0.96	0.96	0.95	0.94	0.94	0.89	0.78	-
		0.97	0.93	0.92	0.88	0.00	0.00	0.00	0.00	-
	20	0.99	0.98	0.99	0.98	0.98	0.99	0.98	0.98	0.98
		0.97	0.95	0.93	0.90	0.87	0.85	0.83	0.80	0.77
	30	0.99	0.98	0.99	0.99	0.99	0.98	0.98	0.99	0.98
		0.98	0.95	0.93	0.92	0.91	0.88	0.87	0.85	0.82
	40	0.99	0.99	0.99	0.99	0.99	1.00	0.99	0.99	1.00
		0.98	0.97	0.95	0.94	0.93	0.91	0.90	0.89	0.87
	50	1.00	0.99	1.00	1.00	0.99	0.99	0.99	0.99	1.00
		0.98	0.97	0.97	0.96	0.94	0.93	0.91	0.91	0.90
	200	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1.00	0.99	0.99	0.99	0.99	0.98	0.98	0.98	0.97

(c) Average scale estimates over the 1000 replications with $\sigma = 1$, with the least squares estimate given first and the MM-estimate second

Table 5.2: Approximate relative efficiencies and average scale estimates of the two MM-estimators from the second run of the simulation with 1000 samples of data for each p and n combination

(a) Relative efficiency of the MM-estimator with nominal 95% relative efficiency in the second run of the simulation

95% efficient MM-estimator		p								
		2	3	4	5	6	7	8	9	10
n	10	89	63	79	53	0	0	0	1	-
	20	90	91	92	87	85	80	71	69	56
	30	94	95	93	91	88	91	86	82	80
	40	95	91	93	91	90	92	88	87	88
	50	95	95	94	95	94	90	95	91	91
	200	96	94	93	96	94	94	95	96	93

75.9% efficient MM-estimator		p								
		2	3	4	5	6	7	8	9	10
n	10	68	53	57	44	0	0	0	1	-
	20	69	68	67	58	57	48	45	45	34
	30	74	75	70	63	62	59	57	48	44
	40	77	71	70	68	63	65	57	58	53
	50	76	75	74	72	70	63	68	62	61
	200	78	75	73	78	74	74	74	77	72

(b) Relative efficiency of the MM-estimator with nominal 75.9% relative efficiency in the second run of the simulation

		p								
		2	3	4	5	6	7	8	9	10
n	10	0.97	0.96	0.97	0.95	0.96	0.91	0.86	0.80	-
		0.98	0.94	0.94	0.86	0.00	0.00	0.00	0.00	-
	20	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98
		0.96	0.94	0.92	0.89	0.87	0.85	0.81	0.79	0.77
	30	0.99	0.99	0.98	0.99	0.99	0.99	0.99	0.99	0.99
		0.98	0.97	0.94	0.92	0.91	0.89	0.87	0.86	0.83
	40	1.00	0.99	1.00	0.99	0.99	0.99	1.00	0.99	0.99
		0.98	0.96	0.96	0.94	0.93	0.91	0.90	0.89	0.87
	50	1.00	1.00	1.00	0.99	1.00	0.99	0.99	0.99	1.00
		0.99	0.98	0.96	0.95	0.95	0.93	0.91	0.91	0.90
	200	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1.00	0.99	0.99	0.98	0.99	0.98	0.98	0.98	0.97

(c) Average scale estimates over the second simulation with 1000 replications and $\sigma = 1$, with the least squares estimate given first and the MM-estimate second.

In part, the inconsistencies could be a result of the subsampling of the data used in the approximate S-estimation within the MM-estimation procedure. Since the subsamples are selected randomly, if the estimator were applied to a particular data set, and then reapplied to exactly the same data set, the two MM-estimates returned would be slightly different. To illustrate this, samples of four independent variables were generated, using different orderings of one to twenty, and the response variable y generated using $\boldsymbol{\beta} = 0$ and $\epsilon_i \sim N(0,1)$. Whilst applying the least squares estimator to the data twice returned identical estimates, when the *rlm* function was applied to the data to find the MM-estimate twice, two (different) values for $\hat{\boldsymbol{\beta}}$ were returned

$$\hat{\boldsymbol{\beta}} = \begin{pmatrix} 0.361736 \\ 0.002754 \\ 0.063939 \\ -0.011168 \\ -0.071962 \end{pmatrix} \text{ and } \hat{\boldsymbol{\beta}} = \begin{pmatrix} 0.361731 \\ 0.002757 \\ 0.063938 \\ -0.011168 \\ -0.071965 \end{pmatrix}.$$

Clearly, the difference between estimates is not large, and whilst the effect would be compounded during the calculation of the relative efficiencies in the simulation, it does not seem a large enough effect to cause the degree of inconsistency seen in the two tables.

It seems more likely then that the number of replications is too low. This idea would be supported if the estimated relative efficiencies became more consistent across two runs of a simulation in which M was much larger. Unfortunately, due to the extensive computation time, it was not possible to repeat the complete simulation using a larger value of M. Using a larger $M = 4000$, however, it *was* possible to carry out a reduced simulation, and repeat it to give an indication of the variability of the results. Table C.1 and Table C.2 of Appendix C show these results.

The average scale estimates are consistent with those previously obtained. Furthermore, as suspected, the simulations demonstrate that with a larger M the estimated relative efficiencies are slightly more consistent across the two repetitions, particularly for the smaller n values such as 10. It certainly seems that in order to give accurate values for the relative efficiency of the MM-estimators at particular values of large $\frac{p}{n}$, a simulation with at least tens of thousands of replications would be needed.

To summarise, whilst a simulation with more replications would have allowed stronger conclusions to be drawn, the simulations that *have* been carried out demonstrate that, as the value of $\frac{p}{n}$ increases, there is a significant loss of MM-estimator relative efficiency and significant underestimation of the scale parameter, with the effects worsening with increasing $\frac{p}{n}$. The simulations also suggest that for $\frac{p}{n}$ smaller than around 0.05 the underestimation of the scale parameter is only slight and the relative efficiencies close to the nominal ones.

5.3 The effect of large $\frac{p}{n}$ on the breakdown point of the MM-estimator

Following from the discussion of the MM-estimator breaking down unexpectedly, this simulation aims to investigate:

- How often does the MM-estimator break in the presence of less than 50% contamination?
- How does the size of $\frac{p}{n}$ affect this?

To study the second question, p was fixed at 3, and n varied between 10 and 50. For each value of n, 4000 samples of data were generated using the model

$$\boldsymbol{Y} = \begin{pmatrix} 1 & x_{11} & x_{21} \\ \vdots & \vdots & \vdots \\ 1 & x_{1n} & x_{2n} \end{pmatrix} \boldsymbol{\beta} + \boldsymbol{\epsilon}$$

where $\epsilon_i \sim N(0,6)$, and $\boldsymbol{\beta}$ is a 3×1 vector $\begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}$. The second and third columns of $\boldsymbol{X}$ correspond to two independent variables $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, generated as samples of length n from a normal distribution with mean zero and standard deviation 10.

For each value of n, the default *rlm* estimator in R was applied to each of the 4000 samples, producing 4000 estimates for $\boldsymbol{\beta}$. Then, in each of the 4000 samples, a single contaminated observation was introduced in the nth position; both x_{1n} and x_{2n} were replaced with values randomly sampled from a Cauchy distribution with location parameter 100, and scale parameter 15. This corresponded to introducing leverage points into the data as the response variables were not changed. The MM-estimator was then reapplied and the number of samples for which the MM-estimator had then broken down recorded.

Next, $x_{1(n-1)}$ and $x_{2(n-1)}$ were replaced likewise and the MM-estimator reapplied, with the number of samples for which the MM-estimator had broken down recorded again. The process continued in this manner until each $x_{1(n/2)}$ and $x_{2(n/2)}$ had been contaminated, so there was 50% contamination in each sample. Since the elements of $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ were generated as random variables, there was no loss of generality in replacing the data in this ordered fashion.

The decision of what constitutes a 'broken down' estimate is slightly arbitrary. In this simulation, the criterion used was that breakdown had occurred if

$$\| \begin{pmatrix} \hat{\beta}_2 \\ \hat{\beta}_3 \end{pmatrix} - \begin{pmatrix} \beta_2 \\ \beta_3 \end{pmatrix} \| > 0.9.$$

To give an idea of the strictness of the criterion, the value 0.9 might correspond to $\hat{\beta}_2$ remaining close to the true value and $\hat{\beta}_3$ growing either larger than about 1.9, or smaller than about 0.1. The criterion would also be violated if both $\hat{\beta}_2$ and $\hat{\beta}_3$ became larger than about 1.65, or smaller than about 0.35. As β_2 and β_3 are actually 1, this criterion seemed reasonable.

The criterion specifically does not involve the intercept term; when it was included, it was not possible to distinguish between the following two situations:

1. The estimates of the two 'slope' parameters change slightly, causing larger but not unreasonable changes in the intercept estimate.

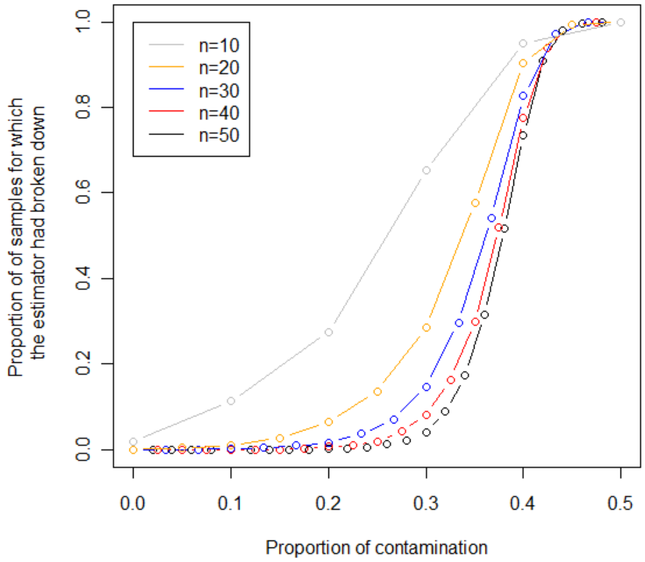
2. The intercept term remains fairly close to 0 but the slope estimates change significantly.

This was not acceptable, as the first situation does not really imply that the estimator has broken down, whilst the second does. Instead, this criterion does not alert one if the slope parameter estimates remain close to the actual values and the intercept term increases dramatically. For this to have happened, the estimated regression plane would have to have been significantly lifted above the data without being tilted at all. Since this seemed a fairly unlikely reaction to the contamination, the criterion was deemed acceptable.

An example of the code used is in Appendix B, and the results of the simulation are given in Table C.3 of Appendix C. Figure 5.3 illustrates the findings.

Figure 5.3 shows that as the contamination level approached 50% there was a dramatic increase in the proportion of samples for which the MM-estimator broke down. With less than 10% contamination, for most values of n, the proportion of samples for which the estimator broke down was negligible. The plot strongly indicates that introducing the same proportion of contamination into a sample had a much worse effect for smaller values of n. This suggests that with larger $\frac{p}{n}$, contamination affects the estimator more, making it less robust. For $n = 10$, corresponding to $\frac{p}{n} = 0.3$, the proportion of samples for which the estimator broke down warranted concern even for modest contamination. For values of n giving $0.1 \leq \frac{p}{n} \leq 0.05$, it

Figure 5.3: Proportion of the 4000 simulated samples for which the MM-estimator broke down, for varying n and contamination levels of up to $\varepsilon = 0.5$.



was only above approximately 30% contamination that there was a significant increase in the proportion of samples for which the estimator broke down.

To summarise, this simulation has suggested that with smaller $\frac{p}{n}$ values, the MM-estimator is more robust to contamination. Of course, without conducting simulations in which p is also varied it is difficult to understand exactly the nature of the relationship between $\frac{p}{n}$, contamination level, and the likelihood of the MM-estimator breaking down. Nevertheless this simulation *has* indicated that with $p = 3$, if it is suspected that data might contain as much as 30% contamination, one ought to have at least 40 data in a sample if one is to rely on the MM-estimator's high breakdown property.

Chapter 6

Conclusion

Robust regression techniques aiming to represent the majority of a sample can be extremely valuable in detecting data that would undermine the least squares estimator's performance. Increasingly sophisticated estimators have been proposed with ever more desirable properties. High breakdown point, high efficiency and bounded influence functions have been the main concerns. Through studying the progression from the M to the MM-estimator in Chapters 2 and 3, this report illustrates how less successful estimators have been built upon and combined to obtain more robust ones.

Many of the proposed estimators that have obtained these three properties are only theoretical, in the sense that they are based on asymptotic or large-sample results, with the practical application of the estimator to data not actually being feasible. This introduces the need for faster algorithms that approximate the estimators. Even these often require lengthy calculations involving many iterations and repeated subsampling, made possible by advances in computer technology.

Chapter 4 highlighted that whilst the theoretical properties of an estimator are important, it is more important to consider the behaviour of the approximated estimator applied to 'real-world' finite data sets. Careful simulations can provide insights into this. Using such simulations and specific examples, this report has demonstrated the real-world behaviour of the MM-estimator. Suspicions were confirmed that its real performance was quite different to that suggested by the associated theory, and far from ideal.

A key idea in the use of robust regression techniques is being able to rely on them *not* to be influenced by individual observations or subtrends in the data, so that if the least squares estimator and the robust estimator coincide, the least squares estimator can be considered reliable. This is desirable as this is the minimum variance estimator, and there are well-researched and thorough diagnostic tools associated with it. It is therefore very important to know if you can indeed rely on the robust estimator in the first place. If not, even if the two estimates are the same one cannot conclude that they both represent the data well. Furthermore, if the two estimates are different one needs to know if the robust estimate is actually representing the majority of the data or if it too may have been negatively influenced. This report draws the conclusion that in order to understand how and why aspects of a sample might be influencing an estimator, it is crucial to look critically at how the estimator performs in reality, as well as in theory. Without understanding the real-world, finite-sample properties of the estimator one cannot justifiably draw conclusions from the results of the robust parameter estimation.

Bibliography

Andersen, R. (2008). *Modern Methods For Robust Regression.* Thousand Oaks: SAGE Publications.

Bianco, A. M., M. Garcia Ben and V. J. Yohai (2005). 'Robust estimation for linear regression with asymmetric errors.' *The Canadian Journal of Statistics,* Vol. 33, No. 4, pp. 511-528.

Bich, W. (2001). 'Estimation and uncertainty in metrology.' *Recent Advances in Metrology and Fundamental Constants,* eds. T. J. Quinn, S. Leschiutta and P. Tavella. Amsterdam: IOS Press, pp. 653-746.

Coakley, C. W. and T. P. Hettmansperger (1983). 'A Bounded Influence, High Breakdown, Efficient Regression Estimator.' *Journal of the American Statistical Association,* Vol. 88, No. 423, pp. 872-880.

Draper, N. R. and K. Smith (1998). *Applied Regression Analysis.* Third edition. New York: Wiley.

Frosch Møller, S., J. von Frese and R. Bro (2005). 'Robust methods for multivariate data analysis.' *Journal of Chemometrics,* Vol. 19, No. 10, pp. 549-563.

Geary, R. C. (1947). 'Testing for Normality.' *Biometrika,* Vol. 34, No. 3/4, pp. 209-242.

Hadi, A. S. and J. S. Simonoff (1993). 'Procedures for the Identification of Multiple Outliers in Linear Models.' *Journal of the American Statistical Association,* Vol. 88, No. 424, pp. 1264-1272.

Hampel, F. R. (1973). 'Robust estimation: A condensed partial survey.' *Probability Theory and Related Fields,* Vol. 27, No. 2, pp. 87-104.

Hogg, R. V. (1979). 'Statistical Robustness: One View of Its Use in Applications Today.' *The American Statistician,* Vol. 33, No. 3, pp. 108-115.

Huber, P. J. (1973). 'Robust Regression: Asymptotics, Conjectures and Monte Carlo.' *The Annals of Statistics,* Vol. 1, No. 5, pp. 799-821.

Huber, P. J. (1996). *Robust Statistical Procedures.* Second edition. Philadelphia: SIAM.

Huber, P. J. and E. M. Ronchetti (2009). *Robust Statistics.* Second edition. Hoboken: Wiley.

Maronna, R. A. and V. J Yohai (2010). 'Correcting MM estimates for "fat" data sets.' *Computational Statistics and Data Analysis,* Vol. 54, No. 12, pp. 3168-3173.

Olive, D. J. (2008). ‘Applied Robust Statistics’ [Online]. URL http://www.math.siu.edu/olive/ol-bookp.htm [last accessed: 3 April 2011].

Olive, D. J. and D. M. Hawkins (2008). ‘The Breakdown of Breakdown’ [Online]. URL http://www.math.siu.edu/olive/ppbdbd.pdf [last accessed: 3 Apil 2011].

R Development Core Team (2011). *R: A language and environment for statistical computing.* R Foundation for Statistical Computing. Vienna, Austria. URL http://www.R-project.org.

Rice, J. A. (1995). *Mathematical Statistics and Data Analysis.* Second edition. Belmont: Duxbury Press.

Rousseeuw, P. J., Christophe Croux, Valentin Todorov, Andreas Ruckstuhl, Matias Salibian-Barrera, Tobias Verbeke, Manuel Koller and Martin Maechler (2011). ’robustbase: Basic Robust Statistics’. *R package version 0.7-3.* URL http://CRAN.R-project.org/package=robustbase

Rousseeuw, P. J. and A. M. Leroy (1987). *Robust Regression and Outlier Detection.* Hoboken: Wiley.

Rousseeuw, P. J. and V. J. Yohai (1984). ‘Robust regression by means of S-estimators.’ *Robust and Nonlinear Time Series Analysis,* eds. J. Franke, W. Härdel, and D. Martin. New York: Springer-Verlag, pp. 256-272.

Schrader, R. M. and T. P. Hettmansperger (1980). ‘Robust Analysis of Variance Based Upon a Likelihood Ratio Criterion.’ *Biometrika*, Vol. 67, No. 1, pp. 93-101.

Stigler, S. M. (1986). *The History of Statistics: The Measurement of Uncertainty before 1900.* Cambridge (Mass.): The Belknap Press of Harvard University Press.

Street, J. O., R. J. Carroll and D. Ruppert (1988). ‘A note on Computing Robust Regression Estimates Via Iteratively Reweight Least Squares.’ *The American Statistician,* Vol. 42, No. 2, pp. 152-154.

Takeaki, K. and K. Hiroshi (2004). *Generalized Least Squares.* Chichester: Wiley.

Yohai, V. J. (1987). ‘High Breakdown-Point and High Efficiency Robust Estimates for Regression.’ *The Annals of Statistics,* Vol. 15, No. 2, pp. 642-656.

Appendix A

Data sets

US Judge Ratings data:

Package: datasets, version 2.9.0
Description: Lawyers' Ratings of State Judges in the US Superior Court
Usage: USJudgeRatings
Source: New Haven Register, 14 January, 1977
Sample used in Section 2.6, Section 3.4 and Section 4.2:

i	1	2	3	4	5	6	7	8	9	10	11
DILG	7.3	8.5	7.8	8.8	6.5	8.5	8.7	5.1	8.7	8.1	7.4
PHYS	8.3	8.5	7.9	8.8	5.5	8.6	9.1	6.8	8.8	8.5	8.4

i	12	13	14	15	16	17	18	19	20	21	22
DILG	7.2	6.8	6.8	8.0	7.7	7.2	7.5	7.8	6.6	7.1	6.8
PHYS	6.9	8.1	6.2	8.4	8.0	6.9	8.1	8.0	7.2	7.5	7.4

i	23	24	25	26	27	28	29	30	31	32	33
DILG	6.2	7.7	8.6	8.9	8.0	8.7	8.2	9.0	8.4	7.9	8.5
PHYS	4.7	7.8	8.7	9.0	8.3	8.8	8.4	8.9	8.4	8.1	8.7

i	34	35	36	37	38	39	40	41	42	43
DILG	8.7	5.6	8.3	8.1	7.8	6.8	8.3	7.0	7.7	7.5
PHYS	8.8	6.3	8.0	8.1	8.5	8.0	8.1	7.6	8.3	7.8

Air quality data:

Package: datasets, version 2.9.0
Description: Daily air quality measurements in New York, May to September 1973
Usage: airquality
Source: The data were obtained from the New York State Department of Conservation (ozone data) and the National Weather Service (meteorological data).
Chambers, J. M., W. S. Cleveland, B. Kleiner, and P. A. Tukey 1983. *Graphical Methods for Data Analysis*. Belmont (CA): Wadsworth.
Sample used in Example 3.2:

i	1	2	3	4	5	6	7	8	9	10
O	41	36	12	18	23	19	8	16	11	14
T	190	118	149	313	299	99	19	256	290	274
Wi	67	72	74	62	65	59	61	69	66	68
S	7.4	8	12.6	11.5	8.6	13.8	20.1	9.7	9.2	10.9

i	11	12	13	14	15	16	17	18	19	20
O	18	14	34	6	30	11	1	11	4	32
T	65	334	307	78	322	44	8	320	25	92
Wi	58	64	66	57	68	62	59	73	61	61
S	13.2	11.5	12	18.4	11.5	9.7	9.7	16.6	9.7	12

i	21	22	23	24	25	26	27	28	29	30
O	23	45	115	37	29	71	39	23	21	37
T	13	252	223	279	127	291	323	148	191	284
Wi	67	81	79	76	82	90	87	82	77	72
S	12	14.9	5.7	7.4	9.7	13.8	11.5	8	14.9	20.7

i	31	32	33	34	35	36	37	38	39	40
O	20	12	13	135	49	32	64	40	77	97
T	37	120	137	269	248	236	175	314	276	267
Wi	65	73	76	84	85	81	83	83	88	92
S	9.2	11.5	10.3	4.1	9.2	9.2	4.6	10.9	5.1	6.3

i	41	42	43	44	45	46	47	48	49	50
O	97	85	10	27	7	48	35	61	79	63
T	272	175	264	175	48	260	274	285	187	220
Wi	92	89	73	81	80	81	82	84	87	85
S	5.7	7.4	14.3	14.9	14.3	6.9	10.3	6.3	5.1	11.5

i	51	52	53	54	55	56	57	58	59	60
O	16	80	108	20	52	82	50	64	59	39
T	7	294	223	81	82	213	275	253	254	83
Wi	74	86	85	82	86	88	86	83	81	81
S	6.9	8.6	8	8.6	12	7.4	7.4	7.4	9.2	6.9

i	61	62	63	64	65	66	67	68	69	70
O	9	16	122	89	110	44	28	65	22	59
T	24	77	255	229	207	192	273	157	71	51
Wi	81	82	89	90	90	86	82	80	77	79
S	13.8	7.4	4	10.3	8	11.5	11.5	9.7	10.3	6.3

i	71	72	73	74	75	76	77	78	79	80
O	23	31	44	21	9	45	168	73	76	118
T	115	244	190	259	36	212	238	215	203	225
Wi	76	78	78	77	72	79	81	86	97	94
S	7.4	10.9	10.3	15.5	14.3	9.7	3.4	8	9.7	2.3

i	81	82	83	84	85	86	87	88	89	90
O	84	85	96	78	73	91	47	32	20	23
T	237	188	167	197	183	189	95	92	252	220
Wi	96	94	91	92	93	93	87	84	80	78
S	6.3	6.3	6.9	5.1	2.8	4.6	7.4	15.5	10.9	10.3

i	91	92	93	94	95	96	97	98	99	100
O	21	24	44	21	28	9	13	46	18	13
T	230	259	236	259	238	24	112	237	224	27
Wi	75	73	81	76	77	71	71	78	67	76
S	10.9	9.7	14.9	15.5	6.3	10.9	11.5	6.9	13.8	10.3

i	101	102	103	104	105	106	107	108	109	110
O	24	16	13	23	36	7	14	30	14	18
T	238	201	238	14	139	49	20	193	191	131
Wi	68	82	64	71	81	69	63	70	75	76
S	10.3	8	12.6	9.2	10.3	10.3	16.6	6.9	14.3	8

i	111
O	20
T	223
Wi	68
S	11.5

Aircraft Data:

Package: robustbase
Description: Aircraft Data, deals with 23 single-engine aircraft built over the years 1947-1979, from Office of Naval Research.
Usage: aircraft
Source: Rousseeuw, P. J. and Leroy, A. M. 1987. *Robust Regression and Outlier Detection.* Hoboken: Wiley. pp. 154, Table 22.
Sample used in Section 4.2:

i	1	2	3	4	5	6	7	8
Aspect Ratio	6.3	6	5.9	3	5	6.3	5.6	3.6
Lift-to-drag Ratio	1.7	1.9	1.5	1.2	1.8	2	1.6	1.2
Weight (lb)	8176	6699	9663	12837	10205	14890	13836	11628

i	9	10	11	12	13	14	15	16
Aspect Ratio	2	2.9	2.2	3.9	4.5	4.3	4	3.2
Lift-to-drag Ratio	1.4	2.3	1.9	2.6	2	9.7	2.9	4.3
Weight (lb)	15225	18691	19350	20638	12843	13384	13307	29855

i	17	18	19	20	21	22	23
Aspect Ratio	4.3	2.4	2.8	3.9	2.8	1.6	3.4
Lift-to-drag Ratio	4.3	2.6	3.7	3.3	3.9	4.1	2.5
Weight (lb)	29277	24651	28539	8085	30328	46172	17836

Appendix B

R Transcripts

Weight functions

```
#Huber weight function with a=1.345
    hub1=function(x) {1.345/abs(x) }
    hub2=function(x) { x^0 }
    x<-seq(-15,-1.345,length=200)
    y<-hub1(x)
    plot(x, y, type="l", xlab="Scaled residual", ylab="Huber weight",xlim=c(-15,15),
         ylim=c(0,1.05),yaxs="i",xaxs="i",cex.lab=1.7,cex.axis=1.2)
    curve(hub2(x), from = -1.345, to = 1.345, add = TRUE,type = "l")
    curve(hub1(x), from = 1.345, to = 15, n = 200, add = TRUE,type = "l")
#Hampel weight function with a=2,b=4,c=8
    hamp1=function(x) {x-x}
    hamp2=function(x) {((8/abs(x))-1)*2/(8-4)}
    hamp3=function(x) {2/abs(x)}
    hamp4=function(x) {x^0}
    x<-seq(-15,-8,length=200)
    y<-hamp1(x)
    plot(x, y, type="l", xlab="Scaled residual", ylab="Hampel weight",
         xlim=c(-15,15),ylim=c(0,1.05),yaxs="i",xaxs="i",cex.lab=1.7,cex.axis=1.2)
    curve(hamp2(x), from = -8, to = -4, add = TRUE, type = "l")
    curve(hamp3(x), from = -4, to = -2, n = 300, add = TRUE,type = "l")
    curve(hamp4(x), from = -2, to = 2, n = 400, add = TRUE,type = "l")
    curve(hamp3(x), from = 2, to = 4, n = 300, add = TRUE,type = "l")
    curve(hamp2(x), from = 4, to = 8, n = 400, add = TRUE,type = "l")
    curve(hamp1(x), from = 8, to = 15, n = 200, add = TRUE,type = "l")
#Tukey bisquare with a=4.685
    Tukey1=function(x) {x-x}
    Tukey2=function(x) {(1-(x/4.685)^2)^2}
    x<-seq(-15,-4.685,length=50)
    y<-Tukey1(x)
    plot(x, y, type="l", xlab="Scaled residual", ylab="Tukey Bisquare weight",
         xlim=c(-15,15),ylim=c(0,1.05),yaxs="i",xaxs="i",cex.lab=1.7,cex.axis=1.2)
    curve(Tukey2(x), from = -4.685, to = 4.685, n=500, add = TRUE,type = "l")
    curve(Tukey1(x), from = 4.685, to = 15, n = 200, add = TRUE,type = "l")
#Least squares weight function
    LS=function(x) {x^0}
    x<-seq(-15,15,length=4)
    y=LS(x)
    plot(x,y,type="l",xlab="Scaled residual",ylab="LS weight",xlim=c(-15,15),
         ylim=c(0,1.05),xaxs="i",yaxs="i",cex.lab=1.8,cex.axis=1.2)
```

M-estimation example of Section 2.6

```
#Preliminaries
data(USJudgeRatings)
attach(USJudgeRatings)
X<-matrix(c(rep(1,43),DILG),byrow=FALSE,ncol=2)
Y<-matrix(c(PHYS))
#Iteration 0
   B0<-solve(t(X)%*%X)%*%t(X)%*%Y  #LS estimate
     r0<-Y-X%*%B0 #initial residuals
     s0<-1.4826*median(abs(r0)) #scale estimate
     u0<-r0/s0 #scaled residuals
     w0<-c((ifelse(abs(u0)>=1.345,1.345/abs(u0),1))) #weights
   W0<-diag(w0)
#Iteration 1
   B1<-solve(t(X)%*%W0%*%X)%*%t(X)%*%W0%*%Y
     r1<-Y-X%*%B1 #residuals
   #Convergence?
     norm<-function(a){sqrt(sum(a^2))}
     c<-norm(B1-B0)/norm(B1)
     ifelse(c<0.0001,"My convergence criterion met: YES",
            "My convergence criterion met: NO")
     c<-norm(r1-r0)/norm(r1)
     ifelse(c<0.0001,"R’s convergence criterion met: YES",
            "R’s convergence criterion met: NO")
     s1<-1.4826*median(abs(r1)) #scale estimate
     u1<-r1/s1 #scaled residuals
     w1<-c((ifelse(abs(u1)>=1.345,1.345/abs(u1),1))) #weights
   W1<-diag(w1)
#Iteration 2
   B2<-solve(t(X)%*%W1%*%X)%*%t(X)%*%W1%*%Y
     r2<-Y-X%*%B2 #residuals
   #Convergence?
     c<-norm(B2-B1)/norm(B2)
     ifelse(c<0.0001,"My convergence criterion met: YES",
            "My convergence criterion met: NO")
     c<-norm(r2-r1)/norm(r2)
     ifelse(c<0.0001,"R’s convergence criterion met: YES",
            "R’s convergence criterion met: NO")
     s<-1.4826*median(abs(r2)) #scale estimate
     u<-r2/s #scaled residuals
     w<-c((ifelse(abs(u)>=1.345,1.345/abs(u),1))) #weights
   W2<-diag(w2)
⋮
```

S-estimation within Example 3.2 of Section 3.4

```
#Verifying scale estimate and S-estimate fit definition
#using a Tukey bisquare objective function and
   a=1.548
   rS<-rlm(O~T+Wi+S,method="MM",maxit=0)$res
   sn<-rlm(O~T+Wi+S,method="MM",maxit=0)$s
   u<-rS/sn
   rho<-c(ifelse(abs(u)<=a,(a^2/6*(1-(1-(u/a)^2)^3)),a^2/6))
   1/111*sum(rho)
```

Code used to construct contaminated data set as used in Example 4.1 of Section 4.1

```
#15 points with 30% contamination
#Sample generated with
x<-seq(1,15,length=15)
y<-c(5*x+7+rnorm(15,0,6))
s1<-sample(15,5)
for(j in s1){
   x[j]<-x[j]+rcauchy(1, location = 0, scale = 50)
   }
```

Code used to construct contaminated data set as used in Figure 4.3 of Section B

```
#MM-estimate with 95% efficiency and MM-estimate with 84.7%
 efficiency norm<-function(a){sqrt(sum(a^2))}
#Sample generated with
x<-seq(1,20,length=20)
y<-c(5*x+7+rnorm(20,0,8))
s1<-sample(20,8)
for(j in s1){
   x[j]<-x[j]+rcauchy(1, location = 0, scale = 70)
   }
```

Code used in Example 4.2 of Section 4.2

```
library(MASS)
library(robustbase)
attach(aircraft)
#Constructing Table 4.1 (Full sample)
```

```
  lm(X1~X2+X3)$coefficients
  rlm(X1~X2+X3,method="MM")$coefficients
  rlm(X1~X2+X3,method="MM",c=4.096,maxit=200)$coefficients
  rlm(X1~X2+X3,method="MM",c=3.420)$coefficients
  rlm(X1~X2+X3,method="MM",c=2.937)$coefficients
  rlm(X1~X2+X3,method="MM",c=2.56)$coefficients
  AMM<-rlm(X1~X2+X3,method="MM")$coefficients
  BMM<-rlm(X1~X2+X3,method="MM",c=4.096,maxit=200)$coefficients
  CMM<-rlm(X1~X2+X3,method="MM",c=3.420)$coefficients
  DMM<-rlm(X1~X2+X3,method="MM",c=2.937)$coefficients
  EMM<-rlm(X1~X2+X3,method="MM",c=2.56)$coefficients
  norm<-function(a){sqrt(sum(a^2))}
    norm(AMM-LS)
    norm(BMM-LS)
    norm(CMM-LS)
    norm(DMM-LS)
    norm(EMM-LS)
```

Example of the type of code used in the simulations of Section 5.2

```
library(MASS)
n=10
M=1000
#Generating the M samples of each of the 9 x-variables n=10
B<-as.data.frame(matrix(rep(0,n*M),ncol=M))
for (m in 1:M) {B[,m]<-rnorm(n,mean=0,sd=100)}; X2<-B
B<-as.data.frame(matrix(rep(0,n*M),ncol=M))
for (m in 1:M) {B[,m]<-rnorm(n,mean=0,sd=100)}; X3<-B
...
B<-as.data.frame(matrix(rep(0,n*M),ncol=M))
for (m in 1:M) {B[,m]<-rnorm(n,mean=0,sd=100)}; X10<-B
###########################FOR ONE X VARIABLE P IS TWO############################
...
##########################FOR FOUR X VARIABLES P IS FIVE##########################
p=5
q=3*(p+1)
Beta<-c(0,rep(1,p-1)) #This allows for easy changing of parameters
Be<-rep(c(Beta,1),3)
Y<-as.data.frame(matrix(rep(0,n*M),ncol=M))
for (m in 1:M)
     {Y[,m]<-X2[,m]*Beta[2]+X3[,m]*Beta[3]+X4[,m]*Beta[4]+X5[,m]*Beta[5]+
      rnorm(n,mean=0,sd=1)}          #Matrix of the M response vectors
#Calculating the estimates for Beta and scale
LS<-matrix(rep(0,p*M),ncol=M)
HMM<-matrix(rep(0,p*M),ncol=M)
LMM<-matrix(rep(0,p*M),ncol=M)
for (i in 1:p)
```

```
    {for (m in 1:M)
         {LS[i,m]<-lm(Y[,m]~X2[,m]+X3[,m]+X4[,m]+X5[,m])$coefficients[i]}}
for (i in 1:p)
    {for (m in 1:M)
    {HMM[i,m]<-rlm(Y[,m]~X2[,m]+X3[,m]+X4[,m]+X5[,m],
                   method="MM",maxit=2000)$coefficients[i]}}
for (i in 1:p)
    {for (m in 1:M)
    {LMM[i,m]<-rlm(Y[,m]~X2[,m]+X3[,m]+X4[,m]+X5[,m],
                   method="MM",maxit=2000,c=2.973)$coefficients[i]}}
LSscale<-matrix(rep(0,M),ncol=M)
HMMscale<-matrix(rep(0,M),ncol=M)
LMMscale<-matrix(rep(0,M),ncol=M)
for (m in 1:M)
    {LSscale[m]<-summary(lm(Y[,m]~X2[,m]+X3[,m]+X4[,m]+X5[,m]))$s}
for (m in 1:M)
    {HMMscale[m]<-rlm(Y[,m]~X2[,m]+X3[,m]+X4[,m]+X5[,m],
                      method="MM",maxit=2000)$s}
for (m in 1:M)
    {LMMscale[m]<-rlm(Y[,m]~X2[,m]+X3[,m]+X4[,m]+X5[,m],
                      method="MM",maxit=2000,c=2.973)$s}
#Combining the results into one matrix of all
#1000 scale and parameter estimates
Sim<-(matrix(rep(0,M*q),ncol=M))
for (m in 1:M)
{Sim[,m]<-c(LS[1,m],LS[2,m],LS[3,m],LS[4,m],LS[5,m],LSscale[m],
            HMM[1,m],HMM[2,m],HMM[3,m],HMM[4,m],HMM[5,m],HMMscale[m],
            LMM[1,m],LMM[2,m],LMM[3,m],LMM[4,m],LMM[5,m],LMMscale[m])}
#Including the average estimates over the M samples, SEs
Mean<-apply(Sim,1,mean)
SE<-apply(Sim,1,sd)
Sim<-cbind(Sim,Mean,SE)
row.names(Sim)=c("LS1","LS2","LS3","LS4","LS5","LSscale",
                 "95MM1","95MM2","95MM3","95MM4","95MM5","95MMscale",
                 "76MM1","76MM2","76MM3","76MM4","76MM5","76MMscale")
#For TMSE, find MSEs to sum:
  #MSE:
  Squares<-matrix(rep(0,M*q),ncol=M)
  MSE<-c(0,q)
  for (i in 1:q)
      {for(m in 1:M)
          {Squares[i,m]<-((Sim[i,m]-Be[i])^2)}}
  for (j in 1:q) {MSE[j]<-1/M*sum(Squares[j,])}
  Sim5<-cbind(Sim,MSE)
#For the TMSE sum the MSEs over the p parameters for each estimator:
TMSE<-c(sum(MSE[1:p]),sum(MSE[(p+2):(2*p+1)]),sum(MSE[(2*p+3):(3*p+2)]))
#Finding the relative efficiencies as ratios of TMSEs:
REFF5<-c(TMSE[1]/TMSE[2],TMSE[1]/TMSE[3])
##########################FOR FIVE X VARIABLES P IS SIX##########################
```

```
p=6
q=3*(p+1)
Beta<-c(0,rep(1,p-1))
Be<-rep(c(Beta,1),3)
Y<-as.data.frame(matrix(rep(0,n*M),ncol=M))
for (m in 1:M) {C[,m]<-X2[,m]*Beta[2]+X3[,m]*Beta[3]+X4[,m]*Beta[4]+
                       X5[,m]*Beta[5]+X6[,m]*Beta[6]+rnorm(n,mean=0,sd=1)}
...
##########################FOR NINE X VARIABLES P IS TEN##########################
p=10
q=3*(p+1)
...
###############TABLES OF RELATIVE EFFICIENCIES AND SCALE ESTIMATES###############
#Table of relative efficiencies:
REFFTABLE10<-rbind( c(c(2:10)/n),
                    c(REFF2[1],REFF3[1],REFF4[1],REFF5[1],REFF6[1],
                      REFF7[1],REFF8[1],REFF9[1],REFF10[1]),
                    c(REFF2[2],REFF3[2],REFF4[2],REFF5[2],REFF6[2],
                      REFF7[2],REFF8[2],REFF9[2],REFF10[2]) )
row.names(REFFTABLE10)=c("p/n","REFF95MM","REFF76MM")
REFFTABLE10<-as.data.frame(REFFTABLE10)
names(REFFTABLE10)<-c(2:10)
REFFTABLE10
#Scale estimates for each p value for this n choice:
p=2; Sim2[c(p+1,2*(p+1),3*(p+1)),1001]
p=3; Sim3[c(p+1,2*(p+1),3*(p+1)),1001]
...
p=10; Sim10[c(p+1,2*(p+1),3*(p+1)),1001]
```

Example of the code used in Section 5.3

```
#The following code produces the proportion of the M samples with n data,
#3 regression coefficients and contamination of one extra point at a time:
library(MASS)
M=4000
n=10
p=3
#Generating the M uncontaminated samples
X2<-as.data.frame(matrix(rep(0,n*M),ncol=M))
for (m in 1:M) {X2[,m]<-rnorm(n,mean=0,sd=10)}
X3<-as.data.frame(matrix(rep(0,n*M),ncol=M))
for (m in 1:M) {X3[,m]<-rnorm(n,mean=0,sd=10)}
Beta<-c(0,rep(1,(p-1)))
Y3<-as.data.frame(matrix(rep(0,n*M),ncol=M))
for (m in 1:M) {Y3[,m]<-X2[,m]*Beta[2]+X3[,m]*Beta[3]+rnorm(n,mean=0,sd=6)}
#Increasing the contaminated points one at a time and each time
#finding proportion of the 4000 samples for which the estimator has ‘broken down’:
#I consider it to have broken down if the ‘contamvector’ has an element >0.9
```

```
    #Uncontaminated samples
    norm<-function(a){sqrt(sum(a^2))}
    contamvector<-rep(0,M)
    for (m in 1:M) {
     contamvector[m]<- norm(
      rlm(Y3[,m]~X2[,m]+X3[,m],method="MM",maxit=2000)$coefficients[2:p]-Beta[2:p]
     )}
    number.bd<-0
    for (m in 1:M) {
     ifelse(contamvector[m]>0.9, number.bd<-number.bd+1,number.bd<-number.bd)}
    prop0<-number.bd/M #proportion of samples that made estimator break down

    #For one contaminted data
    c=1
    for (j in (n-(c-1)):(n-(c-1))) {X2[j,]<-rcauchy(M,100,15)}
    for (j in (n-(c-1)):(n-(c-1))) {X3[j,]<-rcauchy(M,100,15)}
    contamvector<-rep(0,M)
    for (m in 1:M) {
     contamvector[m]<-norm(
      rlm(Y3[,m]~X2[,m]+X3[,m],method="MM",maxit=2000)$coefficients[2:p]-Beta[2:p]
     )}
    number.bd<-0
    for (m in 1:M) {
     ifelse(contamvector[m]>0.9, number.bd<-number.bd+1,number.bd<-number.bd)}
    prop1<-number.bd/M
...
    #For 5 contaminted data
    c=5
    for (j in (n-(c-1)):(n-(c-1))) {X2[j,]<-rcauchy(M,100,15)}
    for (j in (n-(c-1)):(n-(c-1))) {X3[j,]<-rcauchy(M,100,15)}
    contamvector<-rep(0,M)
    for (m in 1:M) {
     contamvector[m]<-norm(
      rlm(Y3[,m]~X2[,m]+X3[,m],method="MM",maxit=2000)$coefficients[2:p]-Beta[2:p]
     )}
    number.bd<-0
    for (m in 1:M) {
     ifelse(contamvector[m]>0.9, number.bd<-number.bd+1,number.bd<-number.bd)}
    prop5<-number.bd/M
#Results
breakdownproportionstable10<-rbind(c(0:5),c(prop0,prop1,prop2,prop3,prop4,prop5))
row.names(breakdownproportionstable10)=c("Prop data contaminated",
 "Prop samples broken down")
breakdownproportionstable10
```

Appendix C

Simulation Results

Table C.1: Approximate relative efficiencies of the two MM-estimators obtained via two runs of 4000 samples of data for each p and n combination. (Referred to in Section 5.2)

(a) First run with 4000 samples: MM-estimator with nominal 95% relative efficiency

		p			
		2	3	4	5
n	10	91	81	73	62
	20	93	91	89	88
	30	93	92	92	91
	40	95	92	94	94
	50	94	94	95	93
	200	95	95	95	95

(b) Second run with 4000 samples: MM-estimator with nominal 95% relative efficiency

		p			
		2	3	4	5
n	10	89	82	76	56
	20	93	90	90	88
	30	94	93	93	90
	40	94	94	93	92
	50	94	95	94	93
	200	96	95	94	95

		p			
		2	3	4	5
n	10	70	58	55	48
	20	73	67	63	60
	30	74	70	69	65
	40	74	70	71	70
	50	74	74	73	71
	200	76	77	77	76

(c) First run with 4000 samples: MM-estimator with nominal 75.9% relative efficiency

		p			
		2	3	4	5
n	10	69	62	57	42
	20	74	69	64	59
	30	75	74	71	61
	40	75	74	72	69
	50	75	76	72	71
	200	77	76	74	76

(d) Second run with 4000 samples: MM-estimator with nominal 75.9% relative efficiency

Table C.2: Comparing the average estimate of $\sigma = 1$ given by the least squares estimator and the MM-estimators between the two simulations with 4000 replications for each p, n combination. The average scale estimates are very similar to those obtained from the simulation with only 1000 replications. (Referred to in Section 5.2)

		p			
		2	3	4	5
n	10	0.96	0.96	0.96	0.95
		0.96	0.92	0.92	0.86
	20	0.99	0.99	0.98	0.99
		0.97	0.95	0.92	0.91
	30	0.99	0.99	0.99	0.99
		0.98	0.96	0.94	0.92
	40	0.99	0.99	1.00	0.99
		0.98	0.97	0.96	0.94
	50	1.00	1.00	0.99	0.99
		0.98	0.98	0.97	0.95
	200	1.00	1.00	1.00	1.00
		1.00	0.99	0.99	0.99

		p			
		2	3	4	5
n	10	0.98	0.96	0.96	0.95
		0.97	0.93	0.92	0.86
	20	0.99	0.99	0.98	0.98
		0.97	0.96	0.92	0.90
	30	0.99	0.99	0.99	0.99
		0.97	0.97	0.95	0.93
	40	0.99	0.99	0.99	0.99
		0.98	0.97	0.96	0.94
	50	0.99	0.99	0.99	0.99
		0.98	0.98	0.96	0.95
	200	1.00	1.00	1.00	1.00
		1.00	0.99	0.99	0.99

Table C.3: Proportion of the 4000 samples for which the MM-estimator of nominal 95% efficiency broke down, with varying n and the contamination increasing up to 50%. (Referred to in Section 5.3)

		Number of data in sample contaminated												
		0	1	2	3	4	5	6	7	8	9	10	11	12
n	10	0.01875	0.11325	0.27475	0.65250	0.95100	0.99800	-	-	-	-	-	-	-
	20	0.00000	0.00475	0.01025	0.02675	0.06375	0.13625	0.28500	0.57550	0.90475	0.99225	0.99975	-	-
	30	0.00000	0.00050	0.00075	0.00200	0.00400	0.00925	0.01625	0.03675	0.07000	0.14800	0.29675	0.54225	0.82600
	40	0.00000	0.00000	0.00000	0.00025	0.00000	0.00025	0.00075	0.00300	0.00650	0.00975	0.01950	0.04250	0.08200
	50	0.00000	0.00000	0.00000	0.00000	0.00000	0.00025	0.00025	0.00000	0.00000	0.00050	0.00150	0.00275	0.00600

		Number of data in sample contaminated												
		13	14	15	16	17	18	19	20	21	22	23	24	25
n	10	-	-	-	-	-	-	-	-	-	-	-	-	-
	20	-	-	-	-	-	-	-	-	-	-	-	-	-
	30	0.97175	0.99800	0.99975	-	-	-	-	-	-	-	-	-	-
	40	0.16375	0.29825	0.52075	0.77425	0.93950	0.99375	0.99950	1.00000	-	-	-	-	-
	50	0.01250	0.02250	0.04175	0.08850	0.17325	0.31525	0.51700	0.73375	0.90850	0.98050	0.99725	1.00000	1.00000

Acknowledgements

I would like to thank Professor Coolen and Dr Einbeck, who supervised this project.