EVALUATION OF ${}_{n}C_{k}$ ESTIMATORS

by

REBHI S. BSHARAT

B.A., Birzeit University, Palestine, 1999
M. S., Wichita State University, 2002

AN ABSTRACT OF A DISSERTATION

submitted in partial fulfillment of the requirements for the degree

DOCTOR OF PHILOSOPHY

Department of Statistics
College of Arts and Sciences

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2007

Abstract

Outliers in the data impair traditional estimators of location, variance, and regression parameters so researchers tend to look for robust estimators, i.e., estimators that aren't sensitive to outliers. These robust estimators can tolerate a certain proportion of outliers. Besides robustness, efficiency is another desirable property. Researchers try to find estimators that are efficient under standard conditions and use them when outliers exist in the data. In this study the robustness and efficiency of a class of estimators that we call ${}_nC_k$ estimators are investigated. Special cases of this method exist in the literature including U and generalized L-statistics. This estimation technique is based on taking all subsamples of size k from a sample of size n, finding the estimator of interest for each subsample, and specifying one of them, typically the median, or a linear combination of them as the estimator of the parameter of interest.

A simulation study is conducted to evaluate these estimators under different distributions with small sample sizes. Estimators of location, scale, linear regression and multiple regression parameters are studied and compared to other estimators existing in the literature. The concept of data depth is used to propose a new type of estimator for the regression parameters in multiple regression.

EVALUATION OF $_nC_k$ ESTIMATORS

by

REBHI S. BSHARAT

B.A., Birzeit University, Palestine, 1999
M. S., Wichita State University, 2002

A DISSERTATION

Submitted in partial fulfillment of the requirements for the degree

DOCTOR OF PHILOSOPHY

Department of Statistics
College of Arts and Sciences

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2007

Approved by:

Major Professor
James J. Higgins

Abstract

Outliers in the data impair traditional estimators of location, variance, and regression parameters so researchers tend to look for robust estimators, i.e., estimators that aren't sensitive to outliers. These robust estimators can tolerate a certain proportion of outliers. Besides robustness, efficiency is another desirable property. Researchers try to find estimators that are efficient under standard conditions and use them when outliers exist in the data. In this study the robustness and efficiency of a class of estimators that we call ${}_nC_k$ estimators are investigated. Special cases of this method exist in the literature including U and generalized L-statistics. This estimation technique is based on taking all subsamples of size k from a sample of size n, finding the estimator of interest for each subsample, and specifying one of them, typically the median, or a linear combination of them as the estimator of the parameter of interest.

A simulation study is conducted to evaluate these estimators under different distributions with small sample sizes. Estimators of location, scale, linear regression and multiple regression parameters are studied and compared to other estimators existing in the literature. The concept of data depth is used to propose a new type of estimator for the regression parameters in multiple regression.

Table of Contents

List of Figures

List of Tables

Acknowledgements

My thanks and appreciation to Dr. James Higgins for persevering with me as my advisor throughout the time it took me to complete this research and write the dissertation.

The members of my dissertation committee, Dr. Suzanne Dubnicka, Dr. Shie-Shien Yang, Dr. Paul Bimal, and Dr. Paul Smith have given their time and expertise to better my work. I thank them for their contribution and their support.

I must acknowledge Dr. John Boyer for the financial support and the teaching experience he provided during my graduate studies.

I would like to thank Dr. Rand Wilcox from the Department of Psychology at University of Southern California for his help in providing some R programs in data depth. I would like also to thank Dr. Robert Serfling from the Department of Mathematical Sciences at University of Texas at Dallas for his help in explaining some of the ideas in his previous work. I also would like to thank Dr. Kemp and John Miles for helping to solve Unix problems during my studies.

My appreciation and sincere thanks must go to my parents, brothers, sisters especially my brother, Mohammad, and my sister in-law, Hanan, for their patience and support during my graduate studies. I would also like to express my sincere thanks to my family-in-law especially my uncle Abu Yousef and my wife, Iman, for their support and encouragement during the last few months of my studies.

Dedication

This work is dedicated to my brother Khair Al-Deen Sari Bsharat for his help and for staying with my parents.

CHAPTER 1 - $_nC_k$ ESTIMATORS

1.1 U-statistics

Estimators that aren't sensitive to outliers are called robust. The sample median is a robust estimator. The sample mean and variance are examples of nonrobust estimators. Robust estimators aren't sensitive to outliers if the proportion of outliers in the data is below some specified value. This value is called the breakdown value or the breakdown point of the estimator. High-breakdown value estimators are usually sought because they resist outliers.

Hoeffding (1948) proposed a general method of estimation by deriving a general class of estimators called U-statistics. The idea is to define a kernel, h, which is a symmetric real-valued function from R^k to R such that $E(h(x_1,...,x_k)) = \theta$ where θ is a parameter of interest. It could be a parameter describing the location or scale of the model, or it could be a regression parameter in the linear model. After collecting the data, $X_1, X_2, ..., X_n$, we take all possible samples of size k from the data, find the value of h for each subsample and construct the statistic

$$U_n = \frac{\sum_{(i_1,...,i_k)} h(X_{i_1},...,X_{i_k})}{\binom{n}{k}}$$

which is an unbiased estimator for θ.

1.2 Generalized L-statistics

Serfling (1984) derived a class of statistics related to the U-statistics called generalized L-statistics or GL-statistics. After taking all subsamples of size k from the original sample, $X_1, X_2, ..., X_n$ and finding the value of h for each subsample, we sort the $\mathrm{N} = \binom{\mathrm{n}}{\mathrm{k}}$ values of h, call those sorted values $W_{1:n}, W_{2:n} ..., W_{N:n}$, take a linear combination of those ordered values using certain weights, $c_{1:n}, c_{2:n} ..., c_{N:n}$, and form the statistic $T_n = \sum_{i=1}^{N} c_{n,i} W_{i:n}$. This class of estimators is called generalized L-statistic. This statistic estimates the quantile, $G_F^{-1}(p),$ of the random variable $h(X_1, ..., X_k)$ where $0 < p < 1$ and G_F is the distribution function of this random variable. If $h(x) = x$, T_n is called an L-statistic which is a linear combination of the order statistics of the sample. Under the following regularity conditions T_n has an asymptotic normal distribution:

1. The density of $h(X_1, X_2, ..., X_k)$, g_F, exists and is positive at ξ_p i.e. $g_F(\xi_p) > 0$
2. $0 < \zeta_p = Var_{X_1}(P\{h(X_1, X_2, ..., X_k) \le \xi_p \mid X_1\}) < \infty$
3. $\int [G_F(y)(1 - G_F(y))]^{1/2} dy < \infty$

The first two conditions are necessary so that the asymptotic variance of the estimator is defined. The next Theorem from Choudhury and Serfling (1988) states the asymptotic distribution of T_n.

Theorem 1.2.1 (Choudhury and Serfling 1988) Under the regularity conditions mentioned above T_n has an asymptotic normal distribution: $\sqrt{n}(T_n - \xi_p) \xrightarrow{d} N(0, \frac{k^2 \zeta_p}{g^2(\xi_p)})$

where $\zeta_p = Var_{X_1}(P\{h(X_1, X_2, ..., X_k) \le \xi_p \mid X_1\})$ and $g(\xi_p)$ is the pdf of $h(X_1,...,X_k)$ evaluated at the population quantile, $\xi_p = G_F^{-1}(p)$. The parameter ξ_p is what is estimated by T_n and ζ_p is the variance of the conditional probability which is the probability that the function $h(X_1,...,X_k)$ is less than ξ_p given X_1.

Example 1.2.1 If $k=1$ and $h(x)=x$, then $U_n = \frac{\sum_{i=1}^{n} h(X_i)}{\binom{n}{1}} = \frac{\sum_{i=1}^{n} X_i}{n} = \overline{X}$. This is a simple case of a U-statistic. U_n estimates the population mean.

Example 1.2.2 If k=2 and $h(x_{i_1}, x_{i_2}) = x_{i_1} x_{i_2}$, then $U_n = \frac{\sum_{(i_1,i_2)}^{n} X_{i_1} X_{i_2}}{\binom{n}{2}}$. In this case, U_n estimates $E(U_n) = E(X_{i_1} X_{i_2}) = E(X_{i_1})E(X_{i_2}) = \mu\mu = \mu^2$ which is the second moment of the distribution.

Example 1.2.3 Bickel and Lehmann (1979) introduced as estimator of $med(| X_{i_1} - X_{i_2} |)$ which is a GL-statistics that estimates a measure of spread of the distribution. The estimator comes has the kernel $h(x_{i_1}, x_{i_2}) =| x_{i_1} - x_{i_2} |$ and estimates $\xi_{0.5} = G_F^{-1}(0.5)$, the median of the distribution of the random variable, $| X_{i_1} - X_{i_2} |$.

Example 1.2.4 In the simple linear regression model, $Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$, $i = 1,...,n$, where ε_i's are i.i.d. with distribution F and the predictor, X, is random variable, we may estimate the slope by $med\left(\dfrac{Y_{i_1}-Y_{i_2}}{X_{i_1}-X_{i_2}}\right)$, $i_1<i_2$. We take all subsamples of size 2 from the paired observations, find the slope for each subsample, and take the median of those slopes. This is the well-known Thiel estimator of the slope (Thiel 1950).

1.3 $_nC_k$ Estimators

Let $\boldsymbol{X}_1,\boldsymbol{X}_2,...,\boldsymbol{X}_n$ be a random sample from a univariate or multivariate distribution $f(\boldsymbol{X};\boldsymbol{\theta})$ where $\boldsymbol{\theta}$ is a constant or vector. The idea of forming an $_n\mathrm{C}_k$ estimator is similar to constructing a U or a GL-statistics. We take all subsamples of size k from the original random sample, find an estimate of $\boldsymbol{\theta}$ for each subsample, order the $N=\begin{pmatrix}n\\k\end{pmatrix}$ estimates of $\boldsymbol{\theta}$(obtained from the $N=\begin{pmatrix}n\\k\end{pmatrix}$ subsamples) according to magnitude in the univariate case or some technique for ordering vectors in the multivariate case, take a linear combination of those ordered values using certain weights, $c_{1:n},c_{2:n}...,c_{N:n}$ and form the statistic $T_{_nc_k}=\sum_{i=1}^{N}c_{n,i}W_{i:n}$. If $\boldsymbol{\theta}$ is a vector, we propose using data depth to do the ordering as discussed later.

Example 1.3.1 The slope estimator, $med\left(\dfrac{Y_{i_1}-Y_{i_2}}{X_{i_1}-X_{i_2}}\right)$, $i_1<i_2$, is an $_n\mathrm{C}_k$ estimator whether or not the X_i's are random whereas the GL-statistic assumes random X_i's. .

In Chapter 2 a small-sample simulation study is conducted to study the efficiency of the Generalized Hodges-Lehmann estimator under different distributions. Results are compared to results existing in the literature. Asymptotic and finite breakdown values of the estimator are given.

In Chapters 3 and 4, a robust ${}_nC_k$ estimator of the variance is introduced. In Chapter 3, we study this estimator under normal distribution. We adjust the estimator for bias, give its asymptotic distribution, asymptotic and finite breakdown values, and look at its simulated efficiency with respect to the sample variance. In Chapter 4 we study this variance estimator under exponential and double exponential distributions. We adjust the estimator for bias and give its efficiency with respect to the sample variance.

In the simple linear regression model, $Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$, $i = 1,\ldots,n$, where ε_i's are identically independently distributed with a continuous distribution F with median 0, interest lies in estimating the parameters β_0 and β_1, predicting the dependent variable Y ,and making confidence intervals on the mean of Y given X = x_i. In Chapter 5, an ${}_nC_k$ robust estimator of the regression parameters in simple linear regression are introduced and compared to other existing estimators in the literature by simulation. When we take all subsamples of size k from the data in the simple linear regression model, we may estimate the intercept either at the end after estimating the slope or we may estimate them simultaneously using data depth to order the vector of coefficients that we get when taking all subsamples of size k.

In Chapter 6, robust ${}_nC_k$ estimators of the regression parameters are introduced and compared to other existing estimators in the literature by simulation. Data depth is used to choose the estimate of the regression parameters.

CHAPTER 2 - ESTIMATING THE MEDIAN OF A SYMMETRIC DISTRIBUTION

2.1 Location Model

Let $X_1, X_2, ..., X_n$ be identically independently distributed continuous random variables with symmetric probability density function f(x) and median θ. Estimators of θ with high breakdown values are always sought because they resist outliers in the data. In this chapter, I will consider a class of robust ${}_{n}C_{k}$ estimators for θ.

2.2 Generalized Hodges-Lehmann Estimator

Example 2.2.1 (Hodges and Lehmann 1963) The Hodges-Lehmann estimator of θ is $med\left(\frac{X_{i_1} + X_{i_2}}{2}\right)$, $i_1 \le i_2$. This is the median of the pair-wise averages and the individual observations. We will define the median of all subsamples of size 2 from the original sample only for $i_1 < i_2$ (sampling without replacement), find the average for each sample, and take the median of those averages. This is asymptotically equivalent to the HL estimator proposed by Hodges and Lehmann 1963.

Example 2.2.2 The Generalized Hodges-Lehmann (GHL) estimator is obtained by taking all subsamples of size k without replacement, finding the mean for each subsample, and taking the median of those averages. The estimator is $\hat{\theta}_k = med(\frac{X_{i_1} + X_{i_2} + ... + X_{i_k}}{k}), k = 3,..., n-1$, Serfling (1984). In general, we take all subsamples of size k from the data, find $\mathrm{h}(x_{k_1},..., x_{i_k}) = \frac{x_{i_1} + x_{i_2} + ... + x_{i_k}}{k}$ for each subsample, let $W_{1:n}, W_{2:n}..., W_{N:n}$ be the ordered $N = \binom{n}{k}$ values of $\mathrm{h}(x_{k_1},..., x_{i_k})$, and find the linear combination $T_{{}_n c_k} = \sum_{i=1}^{N} c_{n,i} W_{i:n}$ where $c_{1:n}, c_{2:n}..., c_{N:n}$ are chosen weights. In the case of the median the weights are

$$c_{n,i} = \begin{cases} 1, & \text{for } \quad i = \frac{N+1}{2} \\ 0, & \text{otherwise} \end{cases} \qquad \text{if N is odd}$$

and

$$c_{n,i} = \begin{cases} 0.5, & \text{for } \quad i = \frac{N}{2}, \frac{N}{2}+1 \\ 0, & \text{otherwise} \end{cases} \qquad \text{if N is even.}$$

Serfling stated in Saleh (1992) that “the use of the median operation, after smoothing the data by taking a function of several observations at a time, over all subsets of the data, leads to a statistic which has a favorable combination of efficiency and robustness, i.e., smoothing followed by taking the median yields both efficiency and robustness”.

2.3 Asymptotic Properties

This class of robust estimators is a special case of the general case introduced in Section 1.2. It is a generalized L-statistic with kernel $\mathrm{h}(x_{k_1},...,x_{i_k}) = \frac{x_{i_1} + x_{i_2} + ... + x_{i_k}}{k}$. Under certain regularity conditions the random variable $\sqrt{n}(\hat{\theta}_k - \theta)$, where $\hat{\theta}_k = med(\frac{X_{i_1} + X_{i_2} + ... + X_{i_k}}{k})$, has an asymptotic normal distribution with mean zero and variance $\frac{1}{c^2}$ where the constant c is called the efficacy of the estimator and has the form

$$\frac{g(\theta)/k}{\sqrt{Var_{X_1}(P(\frac{X_1 + X_2 + X_3 + ... + X_k}{k} \leq \theta \,|\, X_1))}}.$$

where g is the density of $\frac{X_1 + X_2 + X_3 + ... + X_k}{k}$ evaluated at θ. This follows from Theorem 1.2.1 (Choudhury and Serfling 1988). For the GHL estimator sampling is done without replacement, but asymptotically sampling with and without replacement are equivalent. The following is a table of the estimators and their corresponding efficacy for different values of k. For k=2 the efficacy simplifies to $\sqrt{12}\int f^2(x)dx$ (Hettmansperger and Mckean 1998).

Table 2-1 Efficacy of GHL estimator

Estimator	Efficacy
$\overline{X}$	$\frac{1}{\sigma}$
$med(X)$	$2f(0)$
$\hat{\theta}_2$	$\sqrt{12}\int f^2(x)dx$
$\hat{\theta}_k$	$\frac{h(\theta)/k}{\sqrt{Var_{X_1}(P(\frac{X_1+X_2+X_3+\ldots+X_k}{k} \leq \theta \mid X_1))}}$

Choudhury and Serfling (1988) conducted a comparative study in which they showed that taking the median after averaging is highly efficient for k=2, 3, 4, 5 especially for heavy-tail distributions. Table 2-2 contains the asymptotic efficiencies of the GHL estimator with respect to the sample mean under different distributions (Choudhury and Serfling 1988). For the uniform and logistic distributions there is a nonmonotonic pattern in the efficiency as k increases.

Table 2-2 Asymptotic efficiencies of GHL estimator with respect to the sample mean (Choudhury and Serfling 1988)

Distribution	k=1	k=2	k=3	k=4	k=5
Normal	0.637	0.955	0.981	0.989	0.993
Uniform	0.333	1.000	0.849	0.906	0.919
Logistic	0.822	1.097	1.103	1.083	1.077
D. Exp	2.000	1.500	1.321	1.238	1.190

2.4 Breakdown Point

Definition 2.4.1 (Hettmansperger & Mckean 1998) Estimation Breakdown. Let **x** = $(x_1, x_2, ..., x_n)$ represent a realization of a sample and let $\mathbf{x}^{(m)} = (x_1^*, x_2^*, ... x_m^*, x_{m+1}, ..., x_n)$ represent the corruption of any m of the n observations. We define the bias of an estimator $\hat{\theta}$ to be

$$bias(m; \hat{\theta}, \mathbf{x}) = sup \,|\, \hat{\theta}(\mathbf{x}^{(m)}) - \hat{\theta}(\mathbf{x}) \,|,$$

where the supremum is taken over all possible corrupted samples $x^{(m)}$. Note that we corrupt the first m observations only and keep the others fixed. If the bias is infinite, we say the estimate has broken down and the finite-sample breakdown is given by

$$\varepsilon_n^* = \min\left\{\frac{m}{n} : bias(m; \hat{\theta}, \mathbf{x}) = \infty\right\}.$$

The asymptotic breakdown value of $\hat{\theta}$ is the limit of ε_n^* as n goes to infinity and it denoted by ε^*.

The asymptotic breakdown value ε_k^* of $\hat{\theta}_k$ is the proportion of arbitrarily large observations (corrupted or outlying) that the estimator can handle before giving an infinite bias (breaking down) when n is large. For example, one observation guarantees the breakdown of the sample mean. Thus the asymptotic breakdown value of the mean is the limit of $\frac{1}{n}$ as n goes to infinity which is zero. This is why we say the mean has zero asymptotic breakdown value. Another example is the median where half of the observations need to be corrupted to break down the estimator. Thus the breakdown value of the median is the limit of the ratio $\frac{n/2}{n}$ as n goes to infinity. This limit is 0.5 which is the asymptotic breakdown value of the median.

The estimator $\hat{\theta}_k$ is a median of $N = \binom{n}{k}$ averages. If the proportion of outliers in the data is ε then the proportion of clean observations is $1-\varepsilon$. The ith average based on the ith subsample is called nonconatminated if it doesn't contain any outliers. The probability that the ith average is noncontaminated is the probability that all observations in the ith subsample are not contaminated. For large samples this is approximately $(1-\varepsilon)^k$ since for large n and k << n, the subsamples behave essentially as if they were independent observations. The probability $(1-\varepsilon)^k$ can be considered as the proportion of noncontaminated averages among the N averages. The estimator will breakdown if the proportion of noncontaminated averages among the N averages satisfies $(1-\varepsilon)^k \leq 0.5$. Let ε_k^* be the smallest ε such that $(1-\varepsilon)^k \leq 0.5$. This smallest proportion of the data will break down the estimator. Thus $\varepsilon_k^* = 1 - (0.5)^{\frac{1}{k}}$. This probability argument is based on large sample (Rousseeuw and Leroy 1987). Thus ε_k^* is the asymptotic breakdown value of $\hat{\theta}_k$. Values of ε_k^* for k=2, 3, 4 are in Table 2-3. The asymptotic breakdown value of $\hat{\theta}_k$ decreases substantially as k increases.

Sometimes researchers like to use the estimator for small samples, and they need to know the number of corrupted observations that breaks down the estimator. Let m^* be of the number of corrupted observations that the estimator can handle before giving an infinite bias. For small sample the ratio $\frac{m^*}{n}$ is the finite sample breakdown value of the estimator. The estimator $\hat{\theta}_k$ is a median of $N = \binom{n}{k}$ averages and will break down if half or more of those averages are contaminated. Given n, interest lies in finding m^*, the number of outliers that will make half of those averages really large.

To find the finite breakdown value of $\hat{\theta}_k$ for any k, one has to find the minimum number of observations needed to be corrupted so that the bias goes to ∞. Because the estimator $\hat{\theta}_k$ is a median of $N = \binom{n}{k}$ averages, to break down $\hat{\theta}_k$, at least half of those averages must be corrupted. For any k, when m observations are corrupted, the number of averages not containing those corrupted m observations is $\binom{n-m}{k}$. The total number of contaminated averages is $\binom{n}{k} - \binom{n-m}{k}$. The estimator $\hat{\theta}_k$ breaks down if the total number of contaminated averages is at least equal to half of the total averages, i.e., $\binom{n}{k} - \binom{n-m}{k} \geq \binom{n}{k} / 2$. This is equivalent to $\binom{n}{k} / 2 - \binom{n-m}{k} \geq 0$. For a given k and n, finding the finite breakdown value of $\hat{\theta}_k$ is equivalent to finding the smallest m such that $\binom{n}{k} / 2 - \binom{n-m}{k} \geq 0$. Let m^* be the number of observations that breaks down the estimator. The ratio $\frac{\mathrm{m}^*}{\mathrm{n}}$ is the finite breakdown value of $\hat{\theta}_k$. There is no closed form for m^* but it can be found if n and k are given. Given n and k, we find the function $\binom{n}{k} / 2 - \binom{n-m}{k}$ for m=1, 2, 3,…., n-2, and observe when it changes its sign from negative to positive or find the smallest m such that $\binom{n}{k} /2 - \binom{n-m}{k} \geq 0$. This value of m will break down the estimator. This was done for several sample sizes. The values of n, m^*, and $\frac{\mathrm{m}^*}{\mathrm{n}}$ are given in Table 2-4 for k=2, 3, 4.

If we take large n and find $\varepsilon_{\mathrm{n}}^*$, results should be consistent with the asymptotic breakdown derivation. For n=2000 and k=2, m^*=586 and thus $\varepsilon_{\mathrm{n}}^* = \frac{\mathrm{m}^*}{\mathrm{n}} = \frac{586}{2000} = 0.293$. For n=2000 and k=3, $\varepsilon_{\mathrm{n}}^* = \frac{\mathrm{m}^*}{\mathrm{n}} = \frac{413}{2000} = 0.206$. For n=2000 and k=4, $\varepsilon_{\mathrm{n}}^* = \frac{\mathrm{m}^*}{\mathrm{n}} = \frac{318}{2000} = 0.159$. This matches the result of the previous derivation of the asymptotic breakdown value.

Finite sample breakdown values of $\hat{\theta}_k$, denoted by $\varepsilon_{\mathrm{n}}^*$, are given in Table 2-4 for different values of n and k. When the sample size is small and outliers exist in the data researchers can use Table 2-4 to decide which value of k would give a more robust estimator. For example if n=25 and there are 4 outliers in the data, one would avoid using $\hat{\theta}_4$ and could use $\hat{\theta}_2$ or $\hat{\theta}_3$ because 4 outliers guarantee to break down $\hat{\theta}_4$ whereas $\hat{\theta}_2$ or $\hat{\theta}_3$ breaks down if there are at least 8 or 5 outliers in the data respectively.

Table 2-3 Asymptotic breakdown of GHL estimator

k	ε_k^*
1	0.50
2	0.29
3	0.21
4	0.16
5	0.13
6	0.11
7	0.09
$k > 0.5n$	0.00

Table 2-4 Finite breakdown of GHL estimator

	k=2		k=3		k=4	
n	m^*	ε_n^*	m^*	ε_n^*	m^*	ε_n^*
7	2	0.29	2	0.29	1	0.14
8	3	0.38	2	0.25	1	0.12
10	3	0.30	2	0.20	2	0.20
15	5	0.33	3	0.20	3	0.20
25	8	0.32	5	0.20	4	0.16
30	9	0.30	6	0.20	5	0.17
35	11	0.31	7	0.20	6	0.17
40	12	0.30	8	0.20	7	0.18
50	15	0.30	10	0.20	8	0.16
60	18	0.30	12	0.20	10	0.17
70	21	0.30	14	0.20	11	0.16
80	24	0.30	16	0.20	13	0.16
85	25	0.29	18	0.21	14	0.16
90	27	0.30	19	0.21	15	0.17
95	28	0.29	19	0.20	15	0.16
100	30	0.30	21	0.21	16	0.16
200	59	0.30	42	0.21	32	0.16

2.5 Simulation Study

A small sample simulation study was conducted to evaluate the Generalized Hodges-Lehmann estimators under standard normal, uniform on $[0,1]$ and double exponential distribution with mean zero and variance 2, for k=1, 2, 3, 4, 5. Two sample sizes, 15, and 25 with outliers in the data were considered. The number of outliers, k_1, considered is 1, 2, or 3. For the normal distribution, outliers were randomly selected from normal with mean=6 and variance=1. For the uniform distribution, outliers were selected from uniform$[5,6]$, and for double exponential outliers were selected from shifted double exponential with mean 6. Efficiencies are ratios of empirical mean square error of the GHL estimator to the mean square error of sample mean. Results are reported in Tables 2-5 to 2-8. Table 2-9 contains efficiencies for the sample sizes 25, and 100 when there the percentage of outliers is 20%. The first five columns in each table are for n=15 and the second five columns are for n=25.

First consider Table 2-5 when there are no outliers in the data. Efficiency increases as k increases for normal and uniform distribution. For the normal distribution when there are no outliers the increase in efficiency isn't substantial as k goes from 3 to 5 compared to k=2. For the uniform distribution, the increase in efficiency is very little as k goes from 4 to 5. Therefore for the uniform distribution k=4 is the best choice. The GHL estimator is less efficient for the uniform distribution. For the double exponential distribution taking a larger k will decrease the efficiency and k=1, the median, is the most efficient.

Next consider Table 2-6 through 2-9 in which outliers are present. Generally when outliers exist in the data the GHL is more efficient. This is clear by comparing Table 2-5 to other

Tables (Table 2-6 to Table 2-9). However, the efficiency is smaller if the proportion of outliers is larger than the finite breakdown value of the estimator, and in this case it is not recommended.

Comparing k_1=1 to k_1=2, there is an increase in the efficiency but it depends on the distribution. The improvement in the efficiency is larger for the uniform distribution than the normal especially for k=1, 2, 3. For the double exponential distribution the improvement isn't very large especially as k increases and attains its maximum at k=1. Generally the estimator is better when there are two outliers than when there is one outlier in the data.

When k_1=3, n=15 and k is larger than 2, the estimator loses its efficiency because the proportion of outliers exceeds the finite breakdown value of the estimator. On the other hand when k_1=3, n=25 and k is larger than 2, the estimator has efficiency bigger than 1 because the finite breakdown value isn't reached. Comparing k_1=2 to k_1=3 when n=15, improvement in efficiency occurs only at k=1. Comparing k_1=2 to k_1=3 when n=25, improvement in efficiency occurs only at k=1, 2, 3. When k_1=3, n=25, k=2 is very efficient for heavy-tail distributions, and for double exponential k=1 is the most efficient.

Generally for the normal distribution k=2 is the best choice unless the number of outliers is large and in this case we might want use the median. For the uniform distribution k=1 or k=2 are the best choices, and we decide based on the number of outliers and the sample size. For double exponential distribution, we recommend k=1 as it shows the highest efficiency whether outliers exist in the data or not.

Table 2-5 Simulation results for n=15, 25. No outliers are in the data. Efficiencies of the GHL estimator with respect to the sample mean

$k_1 = 0$	n=15					n=25				
F	k=1	k=2	k=3	k=4	k=5	k=1	k=2	k=3	k=4	k=5
N(0,1)	0.68	0.99	1.00	1.00	1.01	0.60	0.99	1.00	1.00	0.99
U(0,1)	0.36	0.86	0.90	0.95	0.97	0.36	0.87	0.88	0.93	0.95
D.exp	1.56	1.30	1.19	1.11	1.07	1.48	1.31	1.21	1.15	1.10

Table 2-6 Simulation results for n=15, 25 with k1=1. Efficiencies of the GHL estimator with respect to the sample mean

$k_1 = 1$	n=15					n=25				
F	k=1	k=2	k=3	k=4	k=5	k=1	k=2	k=3	k=4	k=5
N(0,1)	0.97	2.56	2.27	1.99	1.73	1.37	2.04	1.95	1.86	1.74
U(0,1)	6.90	14.06	13.46	12.33	10.76	4.54	9.81	9.52	9.46	9.00
D.exp	2.43	1.99	1.66	1.42	1.27	2.38	2.01	1.70	1.53	1.40

Table 2-7 Simulation results for n=15, 25 with k1=2. Efficiencies of the GHL estimator with respect to the sample mean

$k_1 = 2$	n=15					n=25				
F	k=1	k=2	k=3	k=4	k=5	k=1	k=2	k=3	k=4	k=5
N(0,1)	4.79	4.10	2.66	1.54	0.90	3.30	3.78	3.07	2.52	2.11
U(0,1)	20.96	29.81	20.83	9.68	0.56	14.45	25.62	21.34	17.86	14.57
D.exp	4.75	2.77	1.73	1.28	1.01	4.69	3.04	2.18	1.74	1.51

Table 2-8 Simulation results for n=15, 25 with k1=3. Efficiencies of the GHL estimator with respect to the sample mean

$k_1 = 3$	n=15					n=25				
F	k=1	k=2	k=3	k=4	k=5	k=1	k=2	k=3	k=4	k=5
N(0,1)	6.60	3.70	0.99	0.92	1.03	5.35	4.62	3.10	2.10	1.60
U(0,1)	35.88	29.71	0.50	0.72	1.01	25.95	33.77	23.74	15.31	8.15
D.exp	6.54	2.29	1.09	1.01	1.05	7.08	3.52	2.14	1.54	1.39

Table 2-9 Simulation results for n=25, 100 with 20% proportion of outliers. Efficiency of the GHL estimator with respect to the sample mean

	n=25 $(k_1 = 5)$					n=100$(k_1 = 20)$				
F	k=1	k=2	k=3	k=4	k=5	k=1	k=2	k=3	k=4	k=5
N(0,1)	8.07	4.07	1.27	1.00	1.11	12.29	5.05	1.58	1.10	1.15
U(0,1)	41.66	30.92	0.57	0.74	1.04	54.32	35.94	9.23	0.77	1.05
D.exp	8.97	2.66	1.22	1.08	1.13	14.11	3.44	1.39	1.17	1.17

CHAPTER 3 - VARIANCE ESTIMATION UNDER NORMAL DISTRIBUTION

3.1 Introduction and Model

When outliers exist in the data, they have serious effects on the sample variance. To solve this problem there are two traditional solutions. The first one is to remove the outliers and use the usual estimators, and the second one is to look for robust estimators. Using robust estimators is potentially a better solution since outliers don't have to be identified. In this study a robust estimate of the variance will be considered and compared to some robust estimators existing in the literature. The number of outliers need not be specified for our proposed ${}_nC_k$ estimators. However we should make sure that they are below the breakdown value.

There are a number of situations in which variance might be important. A chemist might be interested in estimating the variance of the copper concentration in plants. Calcium concentration variability in mammalian blood needs to be below certain level to avoid severe disturbances in blood coagulation (Milton 1999).. In quality control, producers are usually concerned about controlling the variability of the production process, and thus estimating the variance is a vital problem. A researcher might be interested in estimating the mean or the variance in weight of ringed seal in different study zones (Lohr 1999). When conducting tests about location parameters, sometimes we have to find a good estimate of the variance to be able to conduct the test. These are some examples on the importance of the variance estimation problem.

Assume we have a random sample, $X_1, X_2, ..., X_n$, from a population with variance σ^2 . Interest lies in estimating the population variance, σ^2 . In this and the following chapter, we will consider a robust estimator of the variance based on the generalized L-statistic. In this chapter we restrict attention to the normal distribution.

3.2 Robust Estimator of Variance

Let $\hat{\sigma}_k^2 = \text{med}(S_{1k}^2, S_{2k}^2, ..., S_{Nk}^2)$ where we take all subsamples of size k from the random sample $X_1, X_2, ..., X_n$, find the variance for each subsample S_{ik}^2 , and take the median of those variances. If the data comes from normal distribution, it is known that $\frac{k-1}{\sigma^2} S_k^2 \sim \chi_{(k-1)}^2$. This implies that for large n relative to k, $\hat{\sigma}_k^2$ is estimating $\frac{\sigma^2}{k-1}\text{med}(\chi_{k-1}^2)$. Therefore, $\frac{(k-1)\hat{\sigma}_k^2}{\text{med}(\chi_{(k-1)}^2)}$ is a consistent and approximately unbiased estimator of σ^2 when the data have normal distribution and n is large.

For general samples sizes, we need to adjust $\hat{\sigma}_k^2$ to get an unbiased estimate of the variance. Let $d_{n,k}$ be the factor such that $E(d_{n,k}\hat{\sigma}_k^2) = \sigma^2$. For large n and k small, the proposed adjustment factor is approximately the asymptotic value $d_{\infty,k} = \frac{k-1}{\text{med}(\chi_{(k-1)}^2)}$. To determine $d_{n,k}$ in general, we simulated the value of $E(\hat{\sigma}_k^2)$ by taking 500 random samples of size n from standard normal distribution, finding $\hat{\sigma}_k^2$ for each sample, taking the mean of the $\hat{\sigma}_k^2$'s, and computing $d_{n,k}$ as $\frac{\sigma^2}{E(\hat{\sigma}_k^2)}$. This adjustment factor is the same regardless of the population variance. For the standard normal distribution simulated values of $\mathrm{d}_{n,k}$ for n=15, 25, 75, 100, and 125 are presented in Table 3-1 along with $\mathrm{d}_{\infty,k}$ in the last column. Table 3.1 indicates that as $n \rightarrow \infty$, $\mathrm{d}_{n,k} \rightarrow \mathrm{d}_{\infty,k}$. In this chapter we will give the asymptotic distribution of this estimator $d_{\infty,k}\hat{\sigma}_k^2$ and study it under standard normal distribution.

Table 3-1 Simulated values of $d_{n,k}$ and values of $d_{\infty,k}$ (in the right column)for normal distributions. Number of simulations=500. n=15, 25, 75, 100, 125

k	n=1	n=2	n=7	n=10	n=12	n=
2	1.99	2.09	2.16	2.18	2.18	2.20
3	1.32	1.38	1.42	1.43	1.43	1.44
4	1.17	1.22	1.25	1.26	1.26	1.27
5	1.10	1.15	1.17	1.18	1.18	1.19
6	1.06	1.11	1.13	1.14	1.14	1.15

3.3 Asymptotic Approximation

The estimator $\hat{\sigma}_k^2$ is a generalized L-statistic based on $h(x_1,...,x_k) = s_k^2$, and under certain regularity conditions it has an asymptotic normal distribution. The asymptotic variance of $\hat{\sigma}_k^2$ is $k^2 \dfrac{\zeta_{0.5}}{ng_F^2(\xi_{0.5})}$ where $\zeta_{0.5} = Var_{X_1}\{P(S_k^2 \leq \xi_{0.5} \mid X_1)\}$, S_k^2 is the sample variance based on k observations, $\xi_{0.5}$ is the median of the distribution of S_k^2, and g_F is the density of S_k^2 which is assumed to be positive at $\xi_{0.5}$. Thus, $\sqrt{n}(d_{\infty,k}\hat{\sigma}_k^2 - d_{\infty,k}\xi_{0.5})$ has an asymptotic normal distribution with mean 0 and variance $\dfrac{d_{\infty,k}^2 k^2 \zeta_{0.5}}{g_F^2(\xi_{0.5})}$. Note that $d_{\infty,k}\xi_{0.5} = \sigma^2$.

3.4 Asymptotic and Finite Breakdown Points

Both the sample mean and variance have a breakdown value of zero because one outlier in the data can take the bias to infinity. The GHL estimator is a median of N averages and the estimator $\hat{\sigma}_k^2$ is a median of N variances. Therefore, the finite and asymptotic breakdown value of the GHL estimator is as the same as those of $\hat{\sigma}_k^2$. From Table 2-3 we can see that the asymptotic breakdown value decreases substantially as k increases.

3.5 Efficiencies

In this section we simulated the efficiency of the proposed variance estimator with respect to the sample variance under normal distribution with and without outliers. Efficiency was determined as the ratio MSE of the sample variance divided by the MSE of the estimator, $d_{n,k}\hat{\sigma}_k^2$.

From the standard normal distribution 500 random samples of size 15, 25, and 100 were generated to evaluate the estimator $d_{n,k}\hat{\sigma}_k^2$ and the efficiency of the estimator was recorded for k=2, 3, 4, 5, 6 and k_1=0, 2, 4 where k_1 is the number of outliers in the data. The outliers were chosen from a normal distribution with mean μ_0=3 or μ_0=6 and variance 1. The efficiencies are presented in Table 3-2. The first two columns of Table 3-2 are the bias adjustment factors, $d_{n,k}$ and $d_{\infty,k}$. We simulated efficiencies of the estimators with both adjustment factors when there are no outliers in the data to see which gives higher efficiency.

Columns 3 and 4 of Table 3-2 shows that the estimator, $d_{n,k}\hat{\sigma}_k^2$, outperforms the estimator $d_{\infty,k}\hat{\sigma}_k^2$ especially for small samples, and as k increases the efficiency of $d_{n,k}\hat{\sigma}_k^2$ with respect to the sample variance gets closer to 1 faster than that of $d_{\infty,k}\hat{\sigma}_k^2$. Therefore we used $d_{n,k}\hat{\sigma}_k^2$ in all other simulations.

Table 3-2 shows that if there are outliers in the data, the estimator $d_{n,2}\hat{\sigma}_2^2$ has the highest efficiency, and as k increases, the efficiency of the estimator, $d_{n,k}\hat{\sigma}_k^2$, goes down because taking a larger subsample means outliers are more likely to appear in the subsamples and ruin the estimator. Generally when k_1 increases, the change in the efficiency depends on n and k. The smaller the value of k relative to n, the better the estimator, $d_{n,k}\hat{\sigma}_k^2$. The estimator $d_{n,2}\hat{\sigma}_2^2$ is the most efficient besides the fact it has the highest breakdown value.

Table 3-3 contains the efficiencies of the estimator $d_{n,k}\hat{\sigma}_k^2$ with respect to the sample variance when the proportion of outliers is 0.20 using the sample sizes, n=25, 100. The efficiency of the estimator appears to depend on the proportion of outliers not on the number of outliers in the sample.

Table 3-2 Columns 2 and 3 are the values of $d_{\infty,k}$ and the simulated values of $d_{n,k}$. Other columns contain the efficiency of the estimator, $\mathbf{d_{n,k}\hat{\sigma}_k^2}$ and for k_1=0, $d_{\infty,k}\hat{\sigma}_k^2$ relative to the sample variance. Data is generated from standard normal distribution. The outliers, k_1=0, 2, 4, are taken from normal distribution with mean, μ_0=3 or μ_0=6 and variance 1. Number of simulations=500. k =2, 3, 4, 5, 6. n=15, 25,100

	Bias adjustment factors		Efficiencies					
k	$d_{\infty,k}$	$d_{n,k}$	k_1 =0		k_1 =2	k_1 =4	k_1 =2	k_1 =4
			$d_{\infty,k}$	$d_{n,k}$	μ_0=3	μ_0=3	μ_0=6	μ_0=6
n=15								
2	2.20	1.99	0.51	0.78	1.77	1.14	12.70	5.51
3	1.44	1.32	0.53	0.83	1.70	0.75	9.91	0.60
4	1.27	1.17	0.58	0.89	1.42	0.76	5.76	0.79
5	1.19	1.10	0.61	0.95	0.85	0.81	0.55	0.97
6	1.15	1.06	0.64	0.95	0.88	0.88	0.68	1.08
n=25								
2	2.20	2.09	0.64	0.84	1.99	1.69	17.30	11.88
3	1.44	1.38	0.64	0.84	1.99	1.47	17.30	7.01
4	1.27	1.22	0.64	0.84	1.83	1.10	14.78	0.68
5	1.19	1.15	0.70	0.93	1.61	0.93	10.84	0.61
6	1.15	1.11	0.70	0.93	1.38	0.86	7.46	0.67
n=100								
2	2.20	2.18	0.70	0.70	1.55	2.27	14.12	25.49
3	1.44	1.43	0.70	0.70	1.55	2.65	14.12	25.49
4	1.27	1.26	0.70	0.70	1.55	2.27	14.12	25.49
5	1.19	1.18	0.70	1.06	1.55	2.27	14.12	22.66
6	1.15	1.14	1.06	1.06	1.55	1.99	14.12	18.54

Table 3-3 Entries are the efficiencies of the estimator $d_{n,k}\hat{\sigma}_k^2$ relative to the sample variance. Data is generated from standard normal distribution. The outliers, k_1 = 5 when n=25 and k_1 =20 when n=100, are taken from normal distribution with mean, μ_0 =3 or μ_0 =6 and variance 1. Number of simulations=500. k =2, 3, 4, 5, 6. n=25, 100

k	k_1 =5	k_1 =5	k_1 =20	k_1 =20
	μ_0 =3	μ_0 =6	μ_0 =3	μ_0 =6
	n=25		n=100	
2	1.46	8.55	1.57	4.30
3	1.13	1.53	1.20	0.29
4	0.91	0.53	0.98	0.32
5	0.84	0.61	0.90	0.32
6	0.82	0.70	0.88	0.31

3.6 Discussion and Comparison with Other Estimators

The estimator $d_{n,2}\hat{\sigma}_2^2$ is the best in small samples if data are normally distributed and there are some outliers. One has to make sure that the proportion of outliers is less than the breakdown point of this estimator 0.29. From Table 3-4 we saw that the values of k=5, 6 give higher efficiency under normal distribution and no outliers; however, when outliers were introduced in the data as in Table 3-5, larger value of k means faster breakdown for the estimator and since the estimator $d_{n,2}\hat{\sigma}_2^2$ has the highest breakdown, it outperforms other estimator.

Johnson, Mcguire, and Milliken (1978) introduced several estimators of variance in presence of outliers including $V_{k_1}^*$ which proved to be the best one in that paper. They assumed

k_2 of the observations come from normal distribution with mean, μ, and variance, σ^2, and k_1 of the observations, the number of outliers, come from normal with mean, $\mu+\lambda$, and variance, σ^2. Deriving $V_{k_1}^*$ is based on writing the sample variance as $\frac{\sum_{i=1}^{n}\sum_{j=1}^{n}(x_i-x_j)^2}{2n(n-1)}$, defining $u_{ij}=|x_i-x_j|$, for $i<j=2,3,...,n$, and sorting the $u_{ij}'s$ as $u_{(1)}\geq u_{(2)}...\geq u_{(N)}$, where $N=\binom{n}{2}=\frac{n(n-1)}{2}$. If there is one outlier in the data, we expect the n-1 differences between the outlier and the remaining observations to be larger than other differences. Thus they removed those differences from the sum of squares and based the estimator on the differences not including the outlier. When there are k_1 outliers in the data, the estimator

$$V_{k_1}=\frac{\sum_{i<j}u_{ij}^2-\sum_{i=1}^{k_1k_2}u_{(i)}^2}{(k_1(k_1-1)+k_2(k_2-1))}$$

and scale it by $v_{k_1}=E_{\lambda=0}(V_{k_1}/\sigma^2)$ to get the unbiased estimator $V_{k_1}^*=V_{k_1}/v_{k_1}$. Note that this estimator is based on specifying the number of outliers, k_1.

Lax (1985) presented several robust scale estimates for long-tailed symmetric distributions. These estimates included trimmed standard deviation, the median absolute deviation, M-estimates of scale, and A-estimates of scale. He compared 17 of those estimators under normal, long-tail Cauchy, and contaminated normal distributions. According to his simulation study, the following estimate of scale was selected

$$S_{ms}=\frac{ncMAD}{\sqrt{(n-1)}}\tan^{-1}\left[\frac{\sqrt{\sum_{|u_i|\leq\pi}\sin^2(u_i)}}{|\sum_{|u_i|\leq\pi}\cos(u_i)|}\right]$$

where MAD is the median absolute deviation, $u_i = \dfrac{(X_i - M)}{cMAD}$, M is the sample median and c is a specified positive constant. Lax used c=2.1 which was specified by Gross (1976). Mehrotra (1995) recommended using c=2.6 and adjusted the above robust estimate of scale to get the robust estimate of variance, $\mathrm{V_{ms}} = \mathrm{k_n}(\mathrm{S^2_{ms}})$, where

$$\mathrm{k_n} = \begin{cases} 0.973 + 3.353(10^{-6})\mathrm{n}^{3/2} - 3.686(10^{-7})\mathrm{n}^3 + 3.091\mathrm{n}^{-3/2} & \mathrm{n} \le 100 \\ 0.973 \quad \mathrm{n} > 100 \end{cases}$$

The large choice of c increases the efficiency of the scale estimate and the shrinkage factor, k_n, reduces bias for the variance estimate (Mehrotra 1995).

The estimator $\mathrm{d_{n,k}}\hat{\sigma}^2_\mathrm{k}$ was compared to $V^*_{k_1}$, V_{ms}, and S^2 for k=2, 3, 4, 5 and n=15,25,100 using different number of outliers $\mathrm{k_1}$ and different values of $\dfrac{\lambda}{\sigma}$, 0, 1.5, 3, 6. 0 Monte Carlo simulations with 500 repetitions for each sample size were used. Data were simulated from normal distribution. There were $\mathrm{n} - \mathrm{k_1}$ simulated observations from standard normal distribution and $\mathrm{k_1}$ simulated observations, the number of outliers, from normal with mean λ and variance 1. Efficiencies of the estimators with respect to the sample variance are presented in are presented in Tables 3-4 and 3-5.

After examining Table 3-4 and 3-5, it is no surprise that the sample variance is the best estimator when there are no outliers in the data. The estimator V_{ms} isn't very efficient if outliers exist in the data and competition exists between $\mathrm{d_{n,k}}\hat{\sigma}^2_\mathrm{k}$ and $V^*_{k_1}$. When n is small (n=15, 25) if we correctly specify the number of outliers and they are one to two standard deviation from the mean of the data, using $V^*_{k_1}$ is a good choice. When n is small and the outliers are more than three standard deviations from the mean of the data using $\mathrm{d_{n,2}}\hat{\sigma}^2_2$ is a good choice. If n is large (n=100) and we know the number of outlier in addition to the fact that they are less than three standard deviations from the mean of the data, then $V_{k_1}^*$ is as good as $\mathrm{d}_{\mathrm{n,k}}\hat{\sigma}_{\mathrm{k}}^2$, k=3, 4, 5. If we don't know the number of outliers and n is large, then using $\mathrm{d}_{\mathrm{n,k}}\hat{\sigma}_{\mathrm{k}}^2$, k=3, 4, 5, is really efficient. The advantage of using $\mathrm{d}_{\mathrm{n,k}}\hat{\sigma}_{\mathrm{k}}^2$, k=3, 4, 5, is that we are allowing outliers to remain in the data without affecting the estimator, and we don't have to specify their number. For n=100, k=2 gives an efficiency a little lower than the efficiency when k=3, 4, 5. Considering the computation cost for k=3, 4, 5, we would recommend using k=2.

Table 3-4 Entries are efficiencies of the estimator with respect to the sample variance. n=15, 25

k_1	$\frac{\lambda}{\sigma}$	V_1^*	V_2^*	V_3^*	$d_{n,2}\hat{\sigma}_2^2$	$d_{n,3}\hat{\sigma}_3^2$	$d_{n,4}\hat{\sigma}_4^2$	$d_{n,5}\hat{\sigma}_5^2$	V_{ms}
n=15									
0	0	0.93	0.87	0.81	0.76	0.81	0.87	0.93	0.81
1	1.5	1.00	0.95	0.90	0.86	0.95	1.00	0.82	0.86
1	3	1.69	1.79	1.79	1.74	1.91	1.79	1.30	1.24
1	6.0	8.11	12.09	13.35	12.82	15.26	13.64	8.55	5.43
2	1.5	1.00	1.00	0.97	0.88	0.97	0.93	0.88	0.82
2	3	1.34	1.69	1.82	1.77	1.69	1.42	0.85	1.14
2	6.0	1.71	7.77	11.72	12.70	9.91	5.76	0.55	3.53
3	1.5	1.00	1.00	0.97	0.90	0.93	0.90	0.90	0.83
3	3	1.07	1.30	1.49	1.52	1.04	0.84	0.81	0.97
3	6.0	1.02	1.88	6.16	7.99	0.60	0.55	0.65	1.89
n=25									
0	0	0.89	1.00	0.89	0.80	0.80	0.80	0.89	0.89
1	1.5	1.00	1.00	1.00	0.85	0.92	0.92	0.92	1.00
1	3	1.44	0.94	1.63	1.44	1.53	1.53	1.44	1.30
1	6.0	7.12	0.87	11.75	11.19	12.37	12.37	10.68	6.53
2	1.5	1.07	1.07	1.00	0.88	0.88	0.94	0.94	0.94
2	3	1.48	0.92	2.00	2.00	2.00	1.79	1.55	1.42
2	6.0	2.26	0.66	13.33	17.30	17.30	14.52	10.04	6.45
3	1.5	1.06	1.06	1.00	0.90	0.90	0.90	0.86	0.95
3	3	1.27	0.87	1.82	1.87	1.79	1.48	1.13	1.30
3	6.0	1.39	0.35	8.42	15.25	13.14	7.31	1.05	4.48

Table 3-5 Entries are efficiencies of the estimator with respect to the sample variance.

n=100

k_1	$\frac{\lambda}{\sigma}$	V_1^*	V_2^*	V_3^*	$d_{n,2}\hat{\sigma}_2^2$	$d_{n,3}\hat{\sigma}_3^2$	$d_{n,4}\hat{\sigma}_4^2$	$d_{n,5}\hat{\sigma}_5^2$	V_{ms}
0	0	1.00	1.00	1.00	0.50	0.67	0.67	1.00	0.13
1	1.5	1.00	1.00	1.00	0.50	0.67	0.67	0.67	0.13
1	3	1.33	1.33	1.33	1.00	1.33	1.33	1.33	0.31
1	6.0	4.25	5.67	5.67	4.25	5.67	5.67	5.67	1.31
2	1.5	1.50	1.50	1.50	0.75	1.00	1.00	1.00	0.21
2	3	1.20	1.50	1.50	1.20	1.50	1.50	1.50	0.55
2	6.0	2.80	7.00	9.33	11.20	14.00	14.00	14.00	5.60
3	1.5	1.00	1.00	1.00	0.75	0.75	1.00	1.00	0.23
3	3	1.25	1.67	1.67	1.43	2.00	2.00	2.00	1.25
3	6.0	1.98	4.25	8.50	17.00	19.83	19.83	19.83	14.88

CHAPTER 4 - VARIANCE ESTIMATION UNDER EXPONENTIAL AND DOUBLE EXPONENTIAL

4.1 Introduction

In this chapter we will study the estimator $\mathrm{d}_{\mathrm{n,k}}\hat{\sigma}^2_{\mathrm{k}}$ under exponential and double exponential distributions. The exponential distribution represents a skewed distribution and the double exponential represents a heavy-tail distribution.

4.2 Bias Adjustment for Exponential and Double Exponential

4.2.1 Exponential k=2

The adjustment factor $\mathrm{d}_{\infty,\mathrm{k}}$ depends on the median of the distribution of $\mathrm{S}^2_{\mathrm{k}}$. First consider the case of exponential distribution when k=2. Note that $\mathrm{S}^2_2 = 0.5(X_1 - X_2)^2$. Now if the data comes from exponential distribution, $X_1 - X_2$ has double exponential distribution, and since the absolute value of double exponential random variable is exponential random variable, $|X_1 - X_2|$ has exponential distribution. We can write the term $(X_1 - X_2)^2$ in the variance as $(|X_1 - X_2|)^{1/0.5}$ which has a Weibull distribution with $\gamma = 0.5$ and $\beta = 1$. The median of

Weibull distribution with $\gamma = 0.5$ and $\beta = 1$ is $(\ln(2))^2$. Therefore $\mathrm{d}_{\infty,2} = 1/\mathrm{med}(\mathrm{S}_2^2) = 1/\mathrm{med}(0.5(X_1 - X_2)^2) = 2(\ln(2))^{-2} = 4.16$. When data are from double exponential we couldn’t derive the distribution of the sample variance thus we left the last cell in Table 4-1 blank.

4.2.2 Other Cases

For other cases, we used simulation to approximate the adjustment factor as described in Section 3.1 of Chapter 3 except that data were simulated from exponential and double exponential instead of normal. The sample sizes used to simulate those values of $\mathrm{d}_{\mathrm{n,k}}$ are 15, 25, 75,100, and 125. For n=100 and 125, 5000 subsamples were used to approximate $\mathrm{d}_{\mathrm{n,k}}$ for all values of k. Table 4-1 has simulated values.

Table 4-1 Simulated values of $d_{n,k}$ for exponential and double exponential distributions.

Number of simulations=500. n=15, 25, 75, 100, 125

k	Exp.	D. exp.
	n=15	
2	3.31	2.52
3	1.91	1.6
4	1.62	1.34
5	1.22	1.2
6	1.23	1.14
	n=25	
2	3.64	2.74
3	2.05	1.72
4	1.63	1.44
5	1.44	1.3
6	1.33	1.22
	n=75	
2	3.99	2.9
3	2.25	1.84
4	1.79	1.56
5	1.6	1.42
6	1.48	1.32
	n=100	
2	4.02	2.92
3	2.26	1.86
4	1.83	1.56
5	1.63	1.42
6	1.5	1.32
	n=125	
2	4.05	2.96
3	2.29	1.88
4	1.85	1.58
5	1.65	1.42
6	1.52	1.34
	$n = \infty$	
2	4.16	

4.3 Efficiencies

To investigate the efficiency of the variance estimate, $d_{n,k}\hat{\sigma}_k^2$, 500 random samples of size 15,25,100 were generated from the exponential and double exponential distributions. Estimators were computed for k=2, 3, 4, 5, 6. The number of outliers were k_1=0, 2, 4 which were chosen from the original distribution with shift μ_0.

Efficiencies of the estimator $d_{n,k}\hat{\sigma}_k^2$ relative to the sample variance are presented in Table 4-2. Efficiency is the ratio of the MSE of S^2 divided by the MSE of $d_{n,k}\hat{\sigma}_k^2$. The estimator $d_{n,k}\hat{\sigma}_k^2$ does a better job when data comes from exponential than when the data comes from double exponential. The estimator, $d_{n,2}\hat{\sigma}_2^2$, does better than others. We just have to make sure that the proportion of outliers is smaller than the breakdown value of 0.29. Table 4-5 suggests that the efficiency of the estimator depends on the proportion of outliers not on the number of outliers for large samples.

Table 4-2 Entries are the efficiencies of $\mathbf{d_{n,k}\hat{\sigma}_k^2}$ with respect to the sample variance. n=15, 25,100

k	k_1=0		k_1=2, $\mu_0=6$		k_1=4, $\mu_0=6$	
	Exp.	D. Exp.	Exp.	D. Exp.	Exp.	D. Exp.
n=15						
2	1.04	0.97	17.44	1.61	3.92	0.77
3	0.95	1.03	3.13	0.96	0.40	0.32
4	0.95	1.08	2.89	0.81	0.63	0.43
5	0.75	1.05	0.54	0.47	0.77	0.53
6	0.83	1.05	0.73	0.53	0.81	0.59
n=25						
2	1.02	1.07	33.41	1.49	20.83	1.48
3	0.98	1.07	16.71	1.25	3.81	0.75
4	0.95	1.06	6.63	0.95	0.56	0.44
5	0.93	1.07	2.88	0.70	0.60	0.43
6	0.89	1.04	2.31	0.64	0.76	0.48
n=100						
2	1.43	1.06	3.56	0.48	15.02	0.99
3	1.27	1.06	5.34	0.46	23.37	0.91
4	1.26	1.05	6.41	0.45	26.29	0.82
5	1.20	1.03	8.01	0.43	21.03	0.74
6	1.20	1.06	8.01	0.43	16.18	0.70

Table 4-3 Entries are the efficiencies of $\mathbf{d_{n,k}\hat{\sigma}_k^2}$ with respect to the sample variance. Proportion of outliers=0.20. n= 25,100

k	k_1 =5 μ_0=6		k_1 =20 μ_0=6	
n=25			n=100	
	Exp.	D. Exp.	Exp.	D. Exp.
2	12.62	1.66	25.51	1.94
3	1.13	0.53	1.13	0.56
4	0.50	0.43	0.47	0.40
5	0.68	0.52	0.64	0.45
6	0.80	0.62	0.76	0.54

CHAPTER 5 - SIMPLE LINEAR REGRESSION

5.1 Introduction

In the simple linear regression model $Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$, $i = 1,...,n$, where ε_i's are identically and independently distributed with distribution F with median 0, interest lies in estimating the parameters β_0 and β_1 , predicting the dependent variable Y , and making confidence interval on the mean of Y using the predictor X . The dependent variable might be the attention span of a child in minutes and the predictor the child's IQ score. In a study of the body's ability to absorb iron and lead, data might be collected on percentage lead absorbed (Y) and percentage iron absorbed (X), and we might want to predict the percentage lead absorbed using the percentage iron absorbed (Milton 1999). Y might be the annual mean temperature and X the elevation or location expressed by latitude and longitude.

We are concerned with estimating the parameters of the model in this chapter and will compare the proposed method to some methods existing in the literature.

5.2 Existing Methods

Least square estimation, which is based on minimizing the residual sum of squares $\sum_{i=1}^{n} \varepsilon_i^2 = \sum_{i=1}^{n} (Y_i - \beta_0 - \beta_1 x_i)^2$, is one of the common ways to estimate the parameters of the model. The least square estimates aren't robust when outliers exist in the data. Least square estimation is the most efficient when data comes from normal distribution, but the method may do poorly when outliers are present or when the error distribution is heavy-tailed. A lot of methods exist in the literature to estimate the parameters of the linear model. Since we are studying robust estimators in this study, we chose some of the most efficient and robust methods existing in the literature to compare to our method.

Consider minimizing the convex function $\sum_{i<j}^{n} b_{ij} \,|\varepsilon_i - \varepsilon_j\,|$ where b_{ij} is a weight function and ε_i is the ith residual (Naranjo and Hettmansperger 1994). If $b_{ij} = 1$ the estimate that minimizes the convex function is called Wilcoxon rank-based estimate (Jaeckel 1972). The asymptotic relative efficiency of Wilcoxon rank-based estimates is 0.955 relative to least square estimates under normal distribution. For heavy-tail distributions the asymptotic efficiency is much higher (Hettmansperger and McKean 1998). Wilcoxon rank-based estimates have bounded influence function in the y-space and not in the x-space.

Naranjo and Hettmansperger (1994) considered generalized rank-based estimates (GR) by using weights that depend on the x values. These GR estimates desirable robustness properties as discussed in (Naranjo and Hettmansperger 1994). Chang et el (1999) chose weights that depend on the residuals and the x-values and showed that the estimates based on these weights have a breakdown values as high as 50% and called them high-breakdown rank estimates (HBR). Before talking about ${}_nC_k$ estimation in regression, we need to introduce concepts in data depth.

5.3 Data Depth

Data depth is a statistical analysis technique that assigns a numerical value to every point in a data set based on the centrality of this point relative to the data set (Hugg et al 2005). This idea gives a center-outward ordering or ranking of data points for multivariate data. Points that are close to the center receive a higher depth than points that are on the boundary. The center of the data set is defined by certain depth function. Examples of depth functions are halfspace depth (Tukey 1975), simplicial depth (Liu 1990), and simplicial volume depth and projection depth (Zuo and Serfling 2000a). This enables us to choose location and scale estimators based on different depth functions (Zou, Cui, and He 2004). Data depth is a nonparametric technique that doesn't need any distributional assumptions about the data. In this study we will only be concerned with location estimators derived from data depth.

The halfspace depth (HD) of a point x in R^d with respect to probability measure P on R^d is defined as the minimum probability mass carried by any closed halfspace containing x (Zuo and Sefling 2000), that is,

$$HD(x,P) = \inf\{P(H) : H \quad closed \quad halfspace, \quad x \in H\}, \quad x \in R^d .$$

Let's look at the univariate case $d = 1$ to understand the idea of halfspace depth. Given a random sample $X_1, X_2, ..., X_n$ with a distribution function, F , all values below X_i form an open halfspace, and all values less than or equal X_i form a closed halfspace. Similarly all values greater than X_i form an open halfspace and all values greater than or equal X_i form a closed halfspace. For any point, there are two associated closed halfspaces. Tukey's halfspace depth of x_i is defined as the minimum of $F(x_i)$ and $1 - F(x_i^-)$, i.e., the smallest probability associated with the two closed halfspaces formed by x_i (Wilcox 2005). Given a data set, $x_1, x_2, ..., x_n$, we find the proportion of points less than or equal to x_i and the proportion of points greater than or equal to x_i. The sample Tukey halfspace depth of $x_i, \hat{H}(x_i)$ is the smaller of those two values.

Example 5.3.1 Consider the data set, 1, 3, 5, 2, 11, 13, 20, 27, and 23. The proportion of points less than or equal to 3 is 0.33 and the proportion of points greater than or equal to 3 is 0.78. The halfspace depth of the point 3 is the minimum of 0.33 and 0.78, which is 0.33. Table 5.1 below shows the halfspace depth of each point in this data set. We can see that 11 has the maximum halfspace depth in this example. If we sort the data as in the third row of Table 5-1 and look at the depth of each point we can see as points get closer to the center their depth increases and as they get farther away from the center (on the boundary) their depth decreases.

Table 5-1 Halfspace Depth Example

x_i	1	3	5	2	11	13	20	27	23
$\hat{H}(x_i)$	0.11	0.33	0.44	0.22	0.56	0.44	0.33	0.11	0.22
$x_{(i)}$	1	2	3	5	11	13	20	23	27
$\hat{H}(x_{(i)})$	0.11	0.22	0.33	0.44	0.56	0.44	0.33	0.22	0.11

Tukey median is defined as the point with maximum halfspace depth. In the previous example, the point 11 is the Tukey median. It turns out that Tukey median is the sample median in the univariate case (Wilcox 2005). If more than one point has the maximum depth then the average of those points is the Tukey median. In bivariate or multidimensional case, the average is the center of gravity of those points with the maximum depth (Rousseeuw and Ruts 1998), i.e., the x-coordinate of the Tukey median is the average of the x-coordinates of the points with the highest Tukey depth, the y-coordinate of the Tukey median is the average of the y-coordinates of the points with the highest Tukey depth.

For the bivariate case, for any line, the points on or above the line form a closed halfspace, as do the points on or below the line. Given a data set $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$, we find the depth of each point by

1. Looking at every line passes through that point.
2. Counting the proportion of points on or above the line and the proportion of points on or below that line and recording the minimum of those two proportions.
3. Recording the minimum over all lines. This minimum is the Tukey halfspace depth of that point.

In bivariate case the data is a scatter of points in the xy-plane. Points on the boundary will have small depth and those close to the center of the scatter plot will have higher depth.

Example 5.3.2 Consider the data set (1, 3), (1, 5), (2, 1), (2, 4), (2, 6), (2.5, 4.5), (3, 2), (4, 5). Figure 5-1 is a scatter plot of the data. It is clear from the scatter plot that the point (2, 4) is the center of the data. The Tukey halfspace depth of these points is 0.125, 0.125, 0.125, 0.5, 0.125, 0.25, 0.125, and 0.125 respectively. The point (2, 4), Tukey median of the data, has the largest depth.

Example 5.3.3 Consider the data set (1, 2), (1, 5), (2, 1), (2, 3), (2, 4), (2, 6), (3, 2), (3, 5). Figure 5-2 is a scatter plot of the data. The halfspace depth of these points is 0.125, 0.125, 0.125 0.5, 0.5, 0.125, 0.125, and 0.125 respectively. It is clear from the scatter plot that the two points (2, 3) and (2,4) represent the center of the scatter plot. Those two points have the highest Tukey depth. Therefore the Tukey median of this data set is the average of those two points, 0.5(2, 3) +0.5(2, 4), which is (2, 3.5).

Data depth simply assigns high depth values for points closer to the center and low depth values for points on the boundary. Figure 5-3 gives a picture of deep points which are closer to the center and low depth points which are on the boundary in a scatter plot (Hug et el 2006). Depth contours, nested contours that enclose regions of increasing depth, provide a tool to visualize data sets (Figure 5-4). The contour of the sample αth central region is defined as the convex hull containing the most central fraction of α sample points (Hugg et el 2006).

In the three dimension case, for any plane, the points on or above that plane form a closed halfspace, as do the points on or below the plane (Wilcox 2005). Tukey median in the multidimension case is the average of all points having the maximum depth. The data can be pictured as a cloud of points and the points on the boundary have lower depth than points closer to the center of the cloud of the data. The Tukey median in multidimension has a breakdown point that can't exceed $\frac{1}{d+1}$ (Donoho and Gasko 1992). We used Tukey median to choose the ${}_nC_k$ estimator in linear regression.

Figure 5-1 Example on Halfspace Depth. Unique Tukey Median

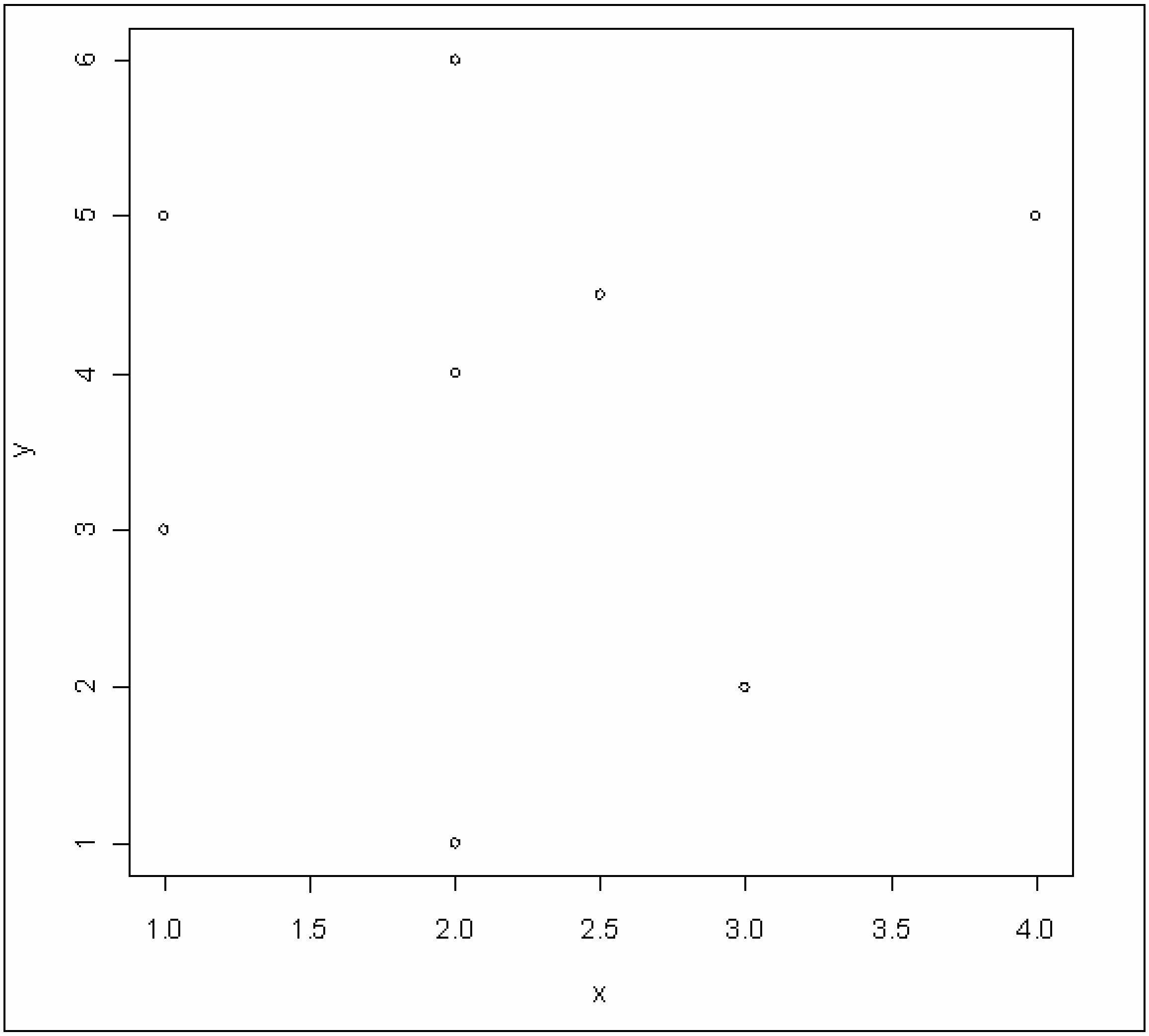


Figure 5-2 Example on Halfspace Depth. Averaging the two points with highest depth to get Tukey Median

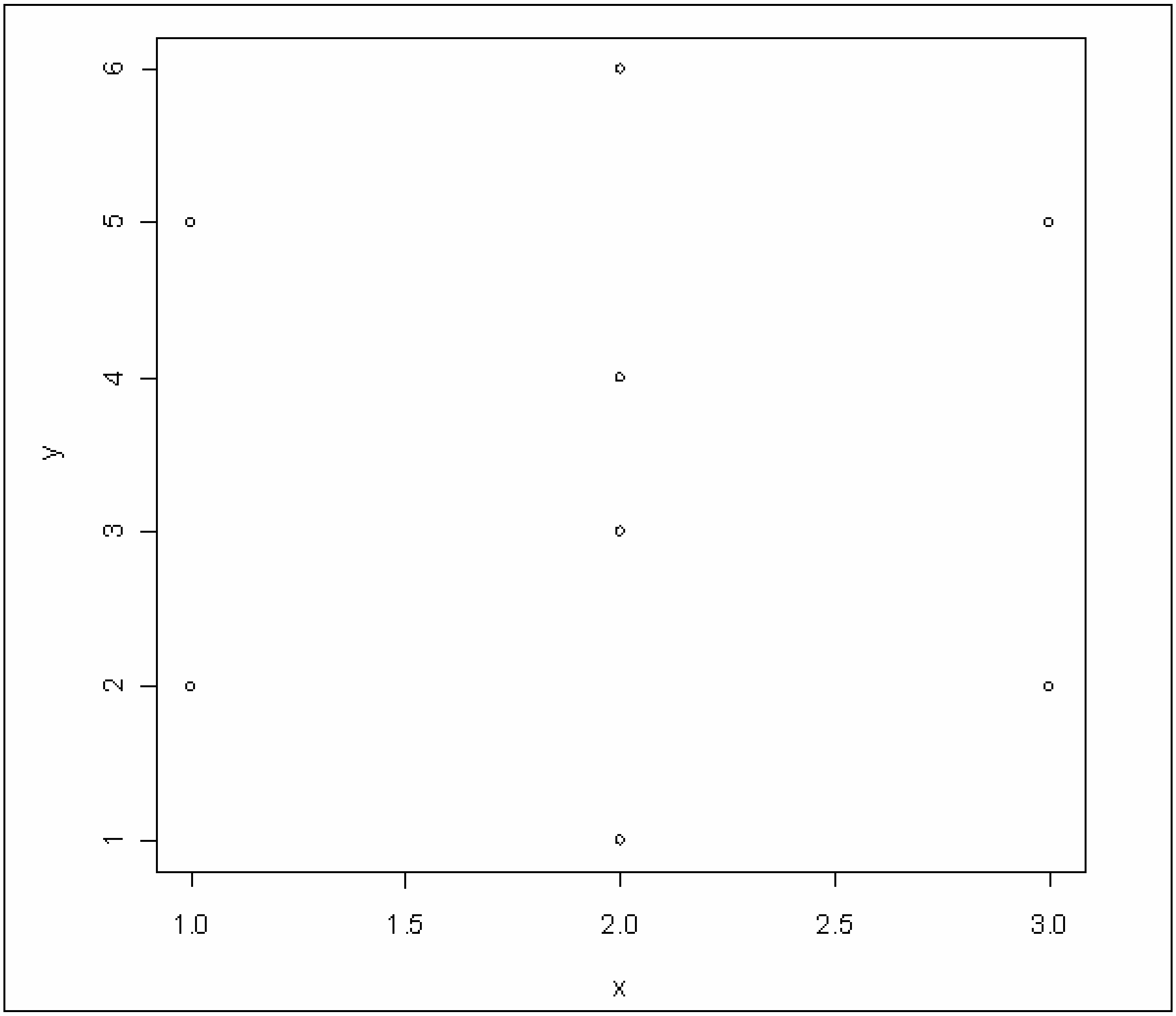


Figure 5-3 An Illustration of Data Depth (Hug et el 2006).

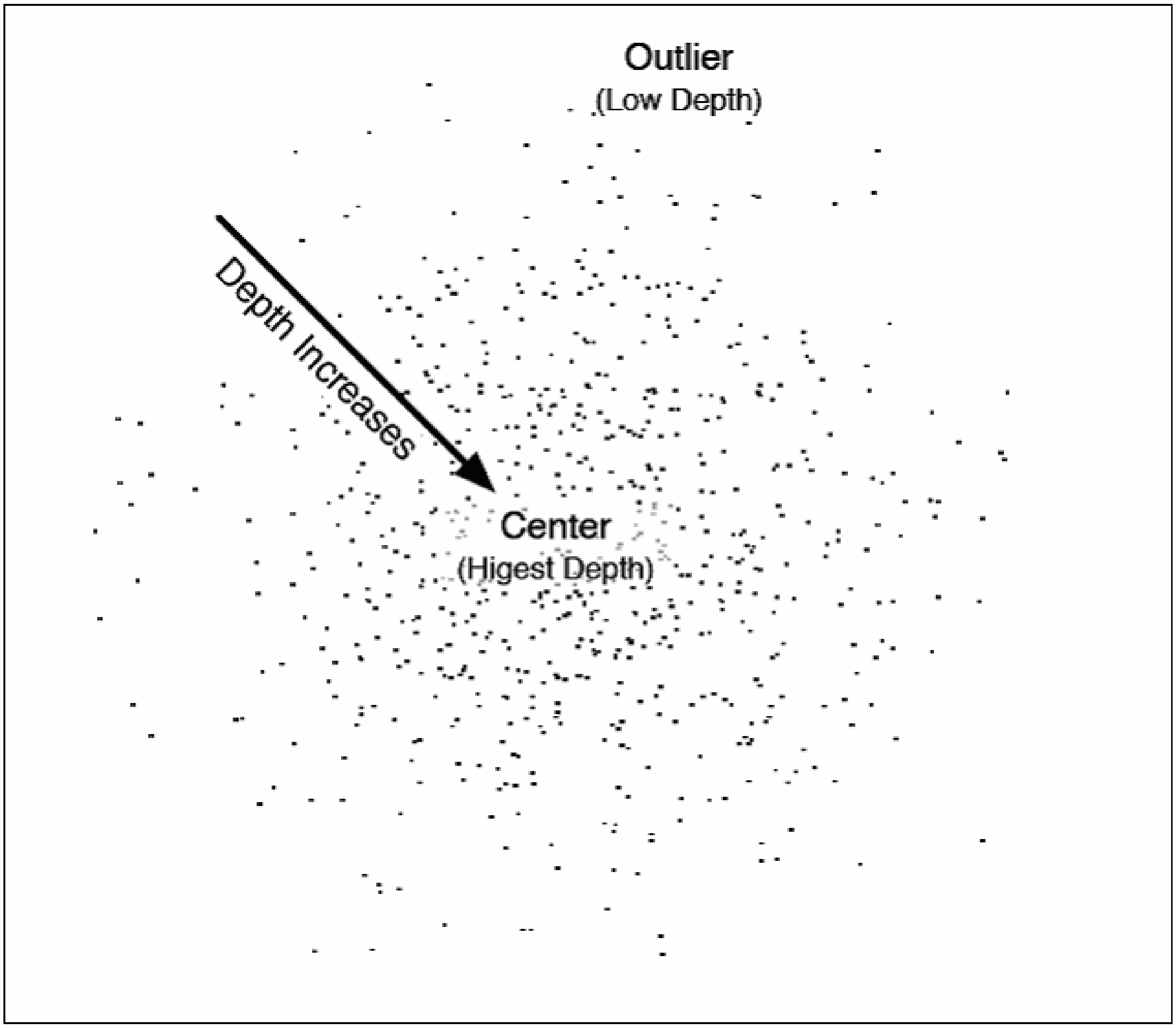
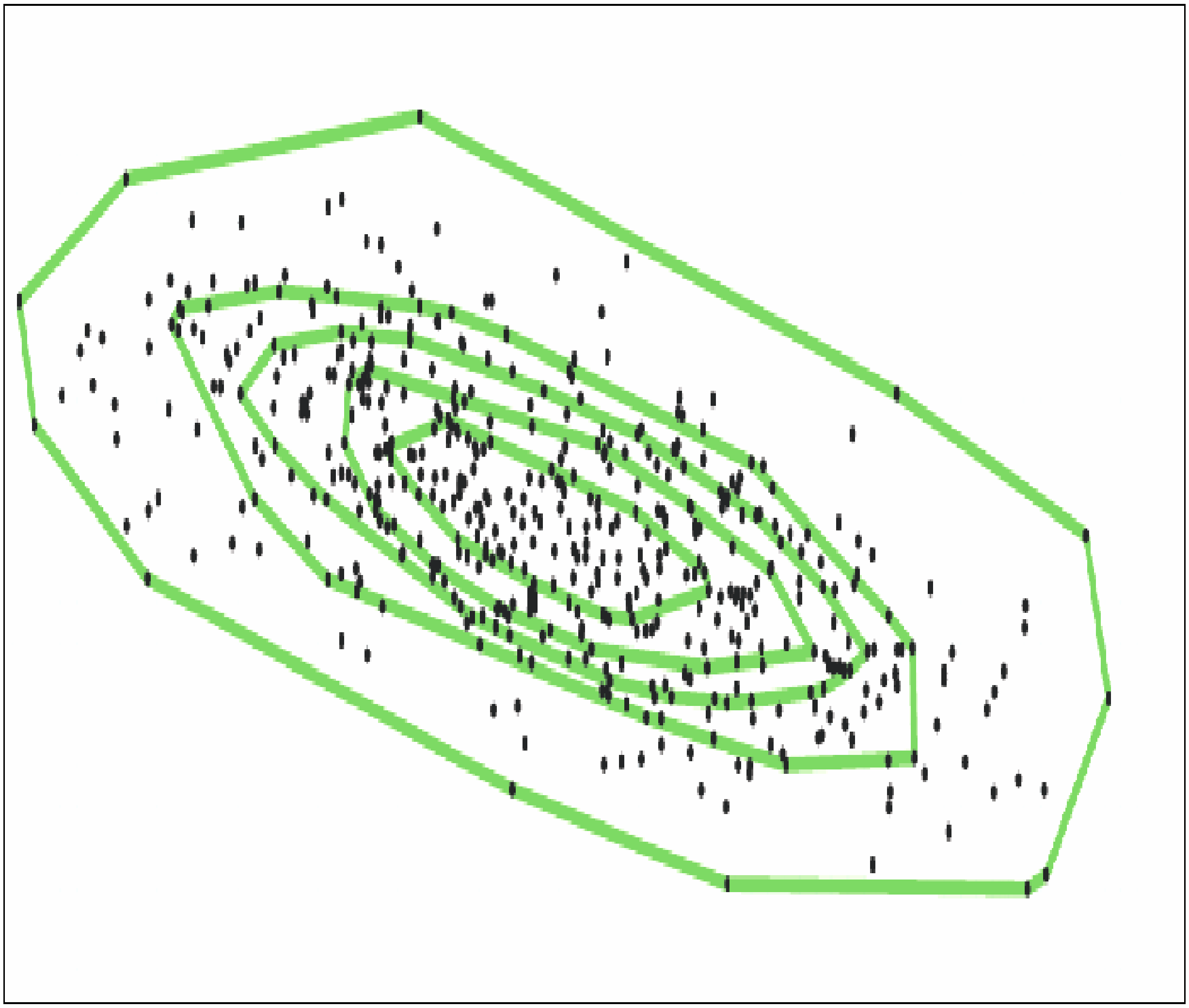


Figure 5-4 Depth Contours. The region Enclosed by the Contour of Depth α is the Set of Points such that $D(x) \geq \alpha$ (Hugg et el 2006).



5.4 ${}_{n}C_{k}$ Estimation in Simple Linear Regression

Given a paired data, $(x_1,Y_1),...(x_n,Y_n)$, we want to generalize the ${}_{n}C_{k}$ estimation idea introduced in previous Chapters to simple linear regression. Experimental units or pairs are sampled. The following are the steps to find the ${}_{n}C_{k}$ estimators of β_0 and β_1:

1. Take all possible samples of size k, k=2,3,4,…,n, without replacement from the n pairs. There are $N=\binom{n}{k}$ subsamples.
2. Find the least square estimators for each subsample. Call these estimators $(\hat{\beta}_{01},\hat{\beta}_{11}),...(\hat{\beta}_{0N},\hat{\beta}_{1N})$.
3. We considered two ways to choose the estimator from step 2:

a. Estimate the slope β_1 by the median of $\hat{\beta}_{11},...\hat{\beta}_{1N}$ and call this estimator $\hat{\beta}_{GT}$ where GT stands for generalized Thiel; estimate β_0 by $\hat{\beta}_0$ where $\hat{\beta}_0 = med(Y_i - \hat{\beta}_{GT}x_i)$, $i=1,...,N$. Thus we have the estimator $\hat{\boldsymbol{\beta}}_1 = (\hat{\beta}_0,\hat{\beta}_{GT})$ for the intercept and the slope. For k=2, the estimator $\hat{\beta}_{GT}$ is Theil estimator of the slope. For k=3, 4, 5,…,n-1, we call the estimator $\hat{\beta}_{GT}$ the generalized Thiel (GT) estimator.

b. Estimate (β_0,β_1) by the Tukey median of $(\hat{\beta}_{01},\hat{\beta}_{11}),...(\hat{\beta}_{0N},\hat{\beta}_{1N})$. Thus we have the estimator $\hat{\boldsymbol{\beta}}_2 = (\hat{\beta}_{0TM},\hat{\beta}_{1TM})$. We will refer to this estimator as Tukey median (TM).

Note that for the first method when k=2 the slope estimator is the median of

$\frac{\sum_{\mathrm{j=1}}^{2}(\mathrm{X_{ij}}-\overline{\mathrm{X}}_{\mathrm{i}})(\mathrm{Y_{ij}}-\overline{\mathrm{Y}}_{\mathrm{i}})}{\sum_{\mathrm{j=1}}^{2}(\mathrm{X_{ij}}-\overline{\mathrm{X}}_{\mathrm{i}})^2}$ where $i=1,\ldots,N$ and x_{ij} is the jth observation from the ith subsample.

Simple algebra shows that $\frac{\sum_{\mathrm{j=1}}^{2}(\mathrm{X_{ij}}-\overline{\mathrm{X}}_{\mathrm{i}})(\mathrm{Y_{ij}}-\overline{\mathrm{Y}}_{\mathrm{i}})}{\sum_{\mathrm{j=1}}^{2}(\mathrm{X_{ij}}-\overline{\mathrm{X}}_{\mathrm{i}})^2}=\frac{(\mathrm{Y_{i1}}-\mathrm{Y_{i2}})}{(\mathrm{X_{i1}}-\mathrm{X_{i2}})}$. Thus when k=2, the estimator is the median of $\frac{(\mathrm{Y_{i1}}-\mathrm{Y_{i2}})}{(\mathrm{X_{i1}}-\mathrm{X_{i2}})}$. This is the Thiel estimator of slope if the predictor is random (Thiel 1950). Other values of k, 3, 4, 5,…, n-1 are generalization of the Thiel estimator. This estimator is a median of $\begin{pmatrix}\mathrm{n}\\ \mathrm{k}\end{pmatrix}$ terms where each term has zero breakdown value. Thus its asymptotic breakdown value is 0.29, 0.21, 0.16, 0.13 and 0.11 for k=2, 3, 4, 5 respectively. The breakdown value of TM is also expected to decrease as k increases. Thus we study the estimators for small values of k.

Oja and Niinimmaa (1984) generalized the Thiel estimator to multiple linear regression. When there are p independent variables, they took all subsets of size p+1, found the least square estimator for each subset, and took the median of the N estimates for each parameter.

It is desired for regression estimators to be affine equivariant, i.e., $\mathrm{T}(\mathbf{Ax_1}+\mathbf{b},\ldots,\mathbf{Ax_n}+\mathbf{b})=\mathbf{A}\mathrm{T}(\mathbf{x_1},\ldots,\mathbf{x_n})+\mathbf{b}$ where T is the estimator, $\mathbf{A}$ is a nonsingular matrix, $\mathbf{b}$ is any vector and $(\mathbf{x_1},\ldots,\mathbf{x_n})$ is the data. This property is desired because reparametrization of the space of the $\mathbf{x}_{\mathrm{i}}$ should not change the estimate (Wilcox 2005). Thus it might not be appropriate to use the marginal median to estimate each parameter separately since this estimator isn't affine equivariant.

In multiple linear regression for k=p+1, p+2,…., n-1, we propose using the Tukey median to choose the estimator because it is a multivariate analong of the usual median. For simple linear regression the range of k is 2, 3, 4…., n. Small values of k are considered here because as k increases the breakdown point of the estimator decreases substantially. Note that each estimator based on a subsample of size k is an unbiased estimator of the true parameters, and when we take the Tukey median of $(\hat{\beta}_{01},\hat{\beta}_{11}),...(\hat{\beta}_{0N},\hat{\beta}_{1N})$, we are trying to find the closest one to the true parameter or the deepest point in the scatter plot.

5.5 Simulation Study under Different Distributions

Chang et el (1999) conducted a simulation study in the simple linear regression model to compare Wilcoxon rank-based, GR, and HBR under different distributions for the predictor and the response. The proposed ${}_nC_k$ regression estimators are compared to those estimators under the same model considered in Chang et el (1999). The model they used for simulation is $Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$, $i = 1,...,30$, where $\beta_0 = \beta_1$=0. The sample size used in this section is 30 which is different from the sample sizes in other parts of this study because this will give us the chance to compare our results to theirs. We compared the ${}_nC_k$ estimators to the most efficient estimators in their study. Partial results of their simulation study are in Table 5-2. The distributions for the X's and Y's in the regression model are of four types, Normal (N), Uniform (U), double exponential (D. exp), and contaminated normal (CN). The contaminated normal distribution CN(ε,σ^2) is defined as contaminated standard normal distribution with ε the proportion of contamination and σ the ratio of standard deviations between the contaminated and noncontaminated parts.

The first combination (normal for the error and uniform for the predictor) represents a standard situation when there are no outliers in the data. The second combination (normal for the error and CN(0.25, 100) for the predictor) represents a contamination in the x-space. The last two cases represent symmetric distribution with outliers for the error under uniform and contaminated normal for the predictor. Double exponential distribution was considered because it has a variety of application (Johnson et el 1994, p.201). Results based on 500 simulations are summarized in Table 5-3. Entries are relative efficiency which is the ratio of the MSE of the slope estimate relative to the MSE of the least square estimator based on 500 simulations.

Our efficiencies for the HBR estimates in Table 5-3 are a little different from the results in Table 5-2 from Chang et el (1999) because the way the HBR estimates are calculated is based on HBR weights which depend on initial location and scale estimates, tuning constants, and residuals from initial estimate. These weights are calculated using Fortran routines that might be different than some of the built-in R routines (Jeff Terpsta, 2007, personal communication).

The GT estimator with k=3 shows a higher efficiency than all other estimates when the response is normal and the predictor is uniform. The Wilcoxon rank-based is the most efficient in relation to the LSE whenever the x-values are contaminated. For the contaminated normal distribution, CN(0.25, 100), GT with k=3 outperforms the HBR estimate. If the response has double exponential distribution and the predictor is uniformly distributed, Theil estimator and TM with k=3 have high efficiency but not better than the rank-based and HBR estimates. We can also see that GT and TM do better when k=3 than k=2 except for when Y has double exponential distribution and X has uniform distribution. Comparing the ${}_nC_k$ estimators, GT with k=3 gives the highest efficiency with respect to the LSE except for when Y has double exponential distribution and X has uniform distribution in which Thiel estimator has the highest efficiency.

The $_nC_k$ estimation was also compared to Wilcoxon rank-based under the t-distribution with 3 degrees of freedom. Table 5-4 contains the relative efficiency of the estimate relative to the LSE for each method. It is clear that Wilcoxon rank-based is more efficient under heavy-tail distributions. We also notice that the efficiency of both GT and TM goes up at k=3 and starts to go down as k decreases.

The only case where we might recommend $_nC_k$ is when the response has normal distribution and the predictor has uniform distribution with few outliers. We recommend GT estimator with k=3 since it has a breakdown value of 0.21 and we don't have to remove the outliers from the data. Generally $_nC_k$ estimators are less efficient that the other estimators

.

Table 5-2 Efficiency of the Estimates Relative to the LSE (Chang et el 1999). n=30

Distribution of Y and X		Type of estimator	
Y	X	Rank	HBR
N	U	0.93	0.78
N	CN(0.25,100)	0.93	0.22
D. exp	U	1.34	1.42
D. exp	CN(0.15,16)	1.14	0.87

Table 5-3 Efficiency of the Estimates Relative to LSE. Wilcoxon Rank-Based and HBR vs. Others. n=30

Distribution of Y and X		Rank-based type		$_nC_k$ Type of estimator			
Y	X	Rank	HBR	Thiel(k=2)	GT(k=3)	TM(k=2)	TM (k=3)
N	U	0.95	0.94	0.92	0.97	0.67	0.91
N	CN(0.25,100)	0.98	0.57	0.50	0.58	0.24	0.40
D. exp	U	1.39	1.40	1.36	1.32	1.15	1.35
D. exp	CN(0.15,16)	1.27	1.25	1.00	1.03	0.78	0.91

Table 5-4 Efficiency to LSE under t-Distribution with df=3. n=15, 25

n=15						
		k=2	k=3	k=4	k=5	Rank
GT	Int	1.44	1.52	1.47	1.40	1.55
	Slope	1.54	1.61	1.54	1.45	1.66
TM	Int	1.27	1.59	1.54	1.46	1.55
	Slope	1.30	1.69	1.61	1.51	1.66
n=25						
		k=2	k=3	k=4	k=5	Rank
GT	Int	1.78	1.80	1.75	1.68	1.82
	Slope	1.85	1.85	1.71	1.71	1.85
TM	Int	1.51	1.80	1.72	1.65	1.82
	Slope	1.60	1.85	1.85	1.71	1.85

5.6 Simulation Study under Normal Distribution with Outliers

In this simulation study we considered the normal distribution with outliers. The response was simulated from the model $Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$, $i = 1,...,n$, where β_0=2 and β_1=1. We used $\boldsymbol{x} = (1,..., \mathrm{n})$ i.e. fixed values from 1 to n. This model was used in simulation by Morton-Jones and Henderson (2000).

For n=15 and n=25 after simulating data from the above model outliers were placed on the xy-direction by replacing the x-values of k_1 pairs by the k_1 values from (30, 40, 45, 55, 65, 75) beginning with x=4. This will create outliers with high Cook's distance. Cook's distance combines leverage and standardized residual into one overall measure of how unusual an observation is.

For n=15 and n=25 after simulating data from the above model outliers were placed on the y-direction by replacing the y-values of k_1 pairs by k_1 values from (30, 40, 45, 55, 65, 75) beginning with the pair with x=4. We only replace the y-values by these outlying points and keep the x values unchanged. This will create outliers with high standardized residuals. Large standardized residuals imply the observation has an unusual response value.

For n=15 and n=25 outliers were placed on the x-direction by replacing the x-values of k_1 pairs by k_1 values from (30, 40, 45, 55, 65, 75) beginning with the pair with x=4 and simulating y-values from the model using these x-values. This will create outliers with high leverage values. The leverage value of an observation tells whether an observation has an unusual predictor which will result in a large influence on the regression coefficient. If it is larger than $\frac{2}{n}$ the observation is considered influential.

The following factors were changed: sample size n, subsample size k, and number of outliers k_1 as shown in Table 5-5. Outliers were placed once in the x-direction, the y-direction, and xy-direction. We studied the estimators GT and TM from section 5-4 and compared them to HBR and Wilcoxon rank-based estimates. As k increases, the outliers will appear more frequently in the subsets we are taking and the estimator is expected to breakdown easily for large values of k so we focused on small values of k. Entries are relative efficiencies which is the ratio of the MSE of the LSE estimate (for intercept and slope) relative to the MSE of the estimator. For the LS estimate column, entries are the MSE's. Results based on 500 simulations are in Tables 5-6 to 5-10.

When outliers are in the y-or xy-direction the efficiency of GT and TM is the highest most of the time when k=2, 3, and as k increases the efficiency decreases substantially. Thus attention should be paid to k=2, 3 when comparing GT and TM to other estimators.

When n=15 and there are two outliers in xy-direction, HBR estimator is the most efficient in relation to the LSE. When there are 4 outliers in the xy-direction, TM with k=2 gives the highest efficiency, and when there are six outliers, GT with k=2 gives the highest efficiency. For n=25 with two or four outliers in the xy-direction, HBR is the most efficient whearas when n=25 and there are six outliers, TM with k=2 is the most efficient.

When n=15, 25 with two outliers are in y-direction, rank-based estimator is the most efficient and if there are four or six outliers, GT with k=2 is the most efficient. When outliers are in the x-direction, the LSE is the best method because outliers give perfect fit to the line in this case as seen in Table 5-10. The efficiency of the rank-based estimator is higher than that of GT, TM, or HBR in this case. When there are 4 or 6 outliers GT or TM with k=3 are better than the HBR.

Generally when outliers are in the xy-direction the HBR estimator is the most efficient, and when outliers are in the y-direction, rank-based or GT with k=2 (Thiel estimator) estimators are the main competitors. Generally GT and TM don't show much improvement over other existing methods but could be used in some special cases based on the number of outliers and the sample size.

Table 5-11 contains the user computation time in hours for the MSE of the GT and TM estimators for each k. The computation time increases substantially as k increases.

Example 5.6.1 A random sample of size 10 was generated from the model $Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$, $i = 1,...,n$, where β_0=0 and β_1=2 where $\boldsymbol{x} = (1,...,10)$. The x-values were changed for three pairs to generate outliers in the xy-direction. The data set is (1, 1.5), (2, 4.0), (20, 7.0), (23, 8.2), (27, 9.8), (6, 13.4), (7, 13.5), (8, 14.5), (9, 18.7), (10, 20.6). Figure 5-5 shows a scatter plot of the data. We want a procedure that fits most of the data without being affected too much by the outlying observations (20, 7.0), (23, 8.2), (27, 9.8). For k=2 the GT estimate is (6.95, 0.40) and the TM is (6.5, 1.0). The TM estimates are much closer to the true parameters than the GT. Figure 5-6 shows a bivariate plot of the slope and intercept estimates for k=2. In this situation taking the Tukey median of both the intercept and slope is more appropriate than taking the median of the slopes and then estimating the intercept at the end (GT). The intercept is a nuisance parameter but the way it is estimated affect the efficiency of getting a good estimate of the slope. This explains the improvement of TM over GT when outliers are in the xy-direction.

Figure 5-5 Example when outliers are in xy-direction

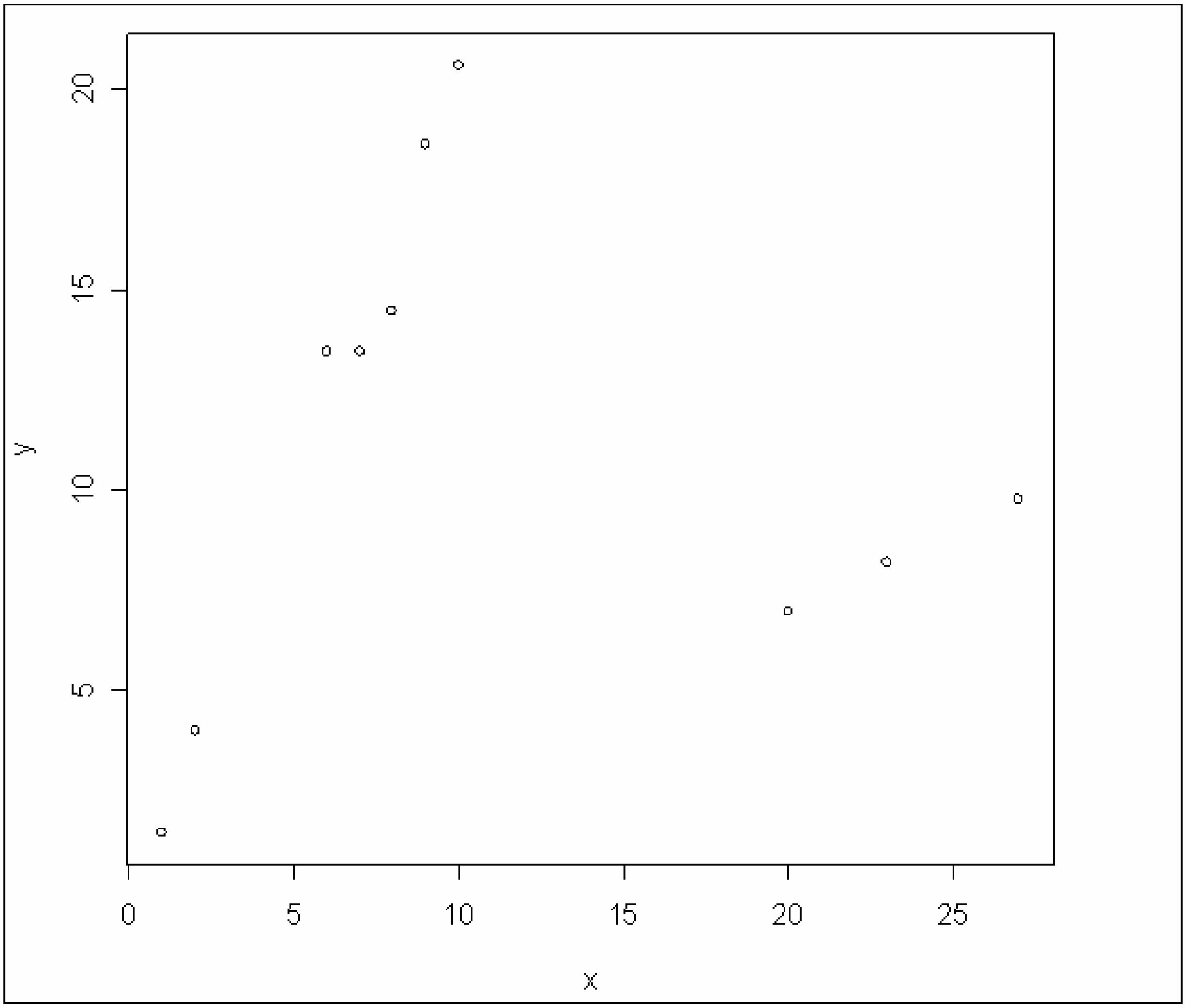


Figure 5-6 Bivariate Plot of the Slope and Intercept Estimates for k=2, n=10

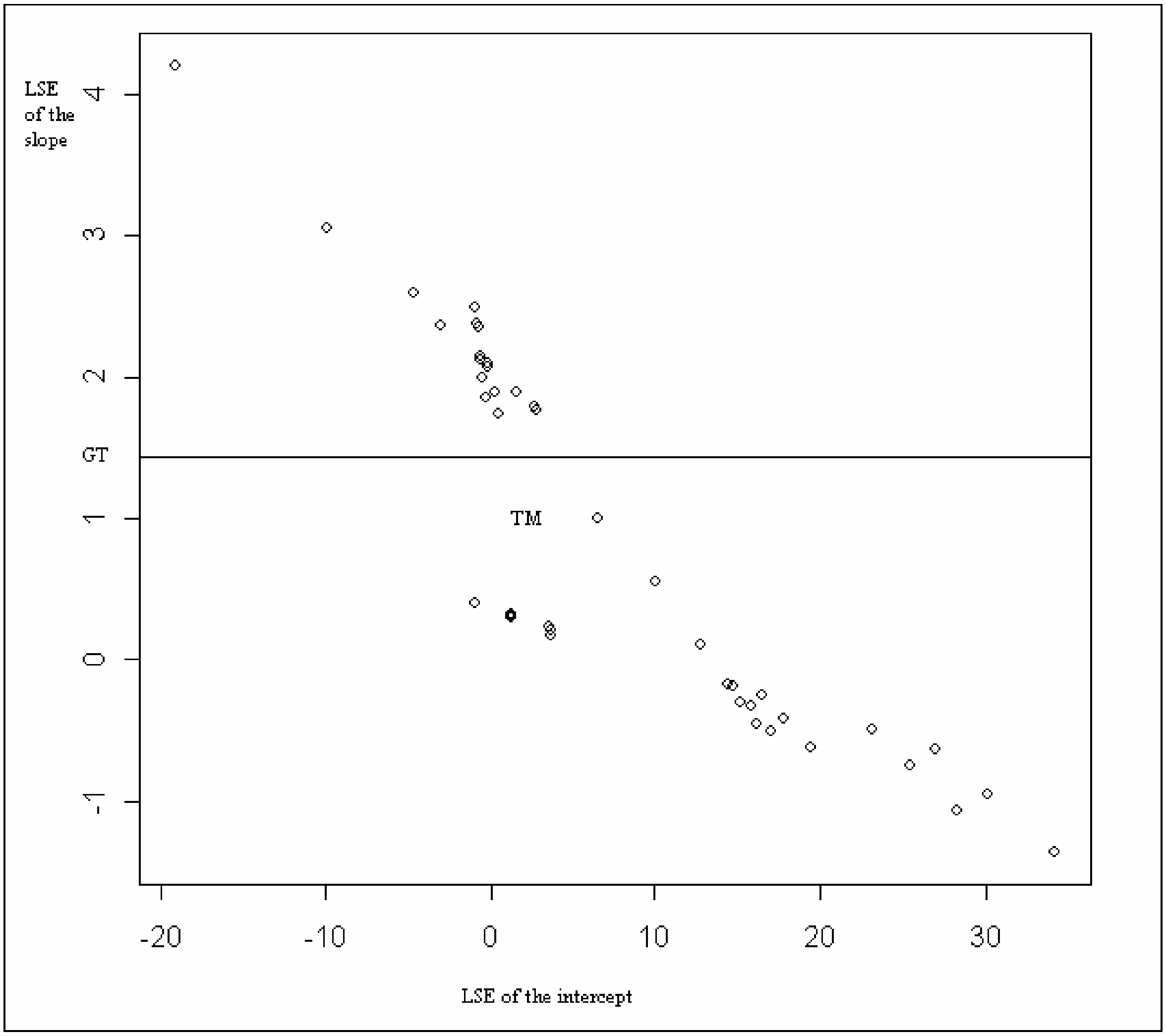


Table 5-5 Factors Changed in Normal Simulation

Factor	Levels
Sample size	15, 25
k	2, 3, 4, 5, 8
# of outliers, k_1	0, 2, 4, 6

Table 5-6 Normal distribution. Efficiency Relative to LSE. Outliers are in xy-direction. n=15

			k=2	k=3	k=4	k=5	k=8	Rank	HBR	LSE*
$k_1 = 0$	GT	Int	0.84	0.88	0.91	0.92	0.94	0.87	0.86	0.29
		Slope	0.90	0.95	0.97	0.97	1.00	0.95	0.92	0.00
	TM	Int	0.68	0.88	0.94	0.97	0.99	0.87	0.86	0.29
		Slope	0.69	0.88	0.92	0.97	1.00	0.95	0.92	0.00
$k_1 = 2$	GT	Int	43.64	35.63	13.25	0.96	0.93	0.95	87.49	55.54
		Slope	45.20	36.79	14.24	1.03	0.99	1.02	100.96	0.91
	TM	Int	40.23	37.00	22.29	13.28	1.03	0.95	87.49	55.54
		Slope	57.51	45.66	25.31	14.54	1.02	1.02	100.96	0.91
$k_1 = 4$	GT	Int	2.00	1.20	1.09	1.04	1.02	0.84	1.51	68.15
		Slope	1.82	1.12	1.05	1.02	1.02	0.96	1.43	1.03
	TM	Int	8.69	2.71	1.01	0.99	1.00	0.84	1.51	68.15
		Slope	12.40	3.40	1.01	1.00	1.00	0.96	1.43	1.03
$k_1 = 6$	GT	Int	7.10	1.49	1.14	1.06	1.05	0.91	1.11	63.26
		Slope	1.20	1.05	1.02	1.01	1.01	0.99	1.03	1.00
	TM	Int	0.65	0.96	0.98	0.99	1.01	0.91	1.11	63.26
		Slope	0.98	1.00	0.99	1.00	1.00	0.99	1.03	1.00

*Entries are mean square errors for the LSE.

Table 5-7 Normal distribution. Efficiency Relative to LSE. Outliers are on xy-direction. n=25

			k=2	k=3	k=4	k=5	k=8	Rank	HBR	LSE*
k_1=0	GT	Int	0.78	0.82	0.84	0.84	0.85	0.79	0.79	0.17
		Slope	0.88	0.88	1.00	1.00	1.00	0.88	0.88	0.00
	TM	Int	0.66	0.89	0.94	0.97	0.99	0.79	0.79	0.17
		Slope	0.70	0.88	1.00	1.00	1.00	0.88	0.88	0.00
$k_1 = 2$	GT	Int	108.63	107.11	91.93	72.41	0.87	46.76	149.24	49.10
		Slope	149.48	143.73	120.55	93.43	1.13	60.27	219.82	0.37
	TM	Int	97.53	100.12	81.94	62.02	20.39	46.76	149.24	49.10
		Slope	169.86	149.48	109.91	77.85	21.98	60.27	219.82	0.37
$k_1 = 4$	GT	Int	76.96	47.06	1.20	0.98	0.94	1.00	194.04	142.48
		Slope	73.56	46.60	1.27	1.03	0.99	1.05	178.02	0.89
	TM	Int	91.95	57.65	27.15	12.98	1.09	1.00	194.04	142.48
		Slope	127.16	70.09	30.28	14.13	1.08	1.05	178.02	0.89
$k_1 = 6$	GT	Int	10.66	1.28	1.16	1.12	1.07	0.99	28.15	193.59
		Slope	9.96	1.20	1.09	1.05	1.01	0.96	25.42	1.08
	TM	Int	33.75	10.25	0.93	0.96	0.99	0.99	28.15	193.59
		Slope	43.38	12.66	1.03	1.00	1.00	0.96	25.42	1.08

*Entries are mean square errors for the LSE.

Table 5-8 Normal distribution. Efficiency Relative to LSE. Outliers are on y-direction. n=15

			k=2	k=3	k=4	k=5	k=8	Rank	LSE*
$k_1 = 2$	GT	Int	94.48	54.80	29.00	3.81	1.90	95.34	88
		Slope	63.19	34.03	16.61	2.13	1.06	64.03	0.49
	TM	Int	57.68	37.63	15.98	4.80	1.07	95.34	88
		Slope	39.89	30.99	14.53	4.59	1.06	64.03	0.49
$k_1 = 4$	GT	Int	164.73	16.61	3.43	2.35	2.41	141.86	336
		Slope	108.13	9.04	1.86	1.27	1.30	91.01	1.27
	TM	Int	35.18	1.85	1.12	1.07	1.03	141.86	336
		Slope	23.56	1.38	0.97	0.98	1.01	91.01	1.27
$k_1 = 6$	GT	Int	342.91	11.47	5.31	4.29	4.31	170.74	597
		Slope	154.65	3.14	1.42	1.14	1.15	68.74	0.8
	TM	Int	0.58	0.40	1.06	0.98	0.99	170.74	597
		Slope	0.18	0.13	0.92	0.90	0.98	68.74	0.8

Table 5-9 Normal distribution. Efficiency Relative to LSE. Outliers are on y-direction. n=25

			k=2	k=3	k=4	k=5	k=8	Rank	LSE*
$k_1 = 2$	GT	Int	103.74	79.58	64.38	50.61	1.69	99.85	50.40
		Slope	80.82	59.74	45.80	35.23	1.04	80.82	0.14
	TM	Int	91.75	77.16	54.01	34.53	3.32	99.85	50.40
		Slope	65.43	59.74	44.32	29.87	3.24	80.82	0.14
$k_1 = 4$	GT	Int	170.34	66.41	4.58	1.71	1.44	144.56	248.34
		Slope	142.07	50.07	3.27	1.22	1.03	115.26	0.61
	TM	Int	138.15	56.84	18.40	2.54	1.22	144.56	248.34
		Slope	111.07	50.07	17.26	2.45	1.20	115.26	0.61
$k_1 = 6$	GT	Int	183.61	6.61	2.25	1.68	1.32	114.95	666.89
		Slope	160.88	5.07	1.73	1.29	1.02	94.38	1.42
	TM	Int	90.82	13.61	1.74	1.25	1.07	114.95	666.89
		Slope	75.30	12.06	1.60	1.17	1.04	94.38	1.42

Table 5-10 Normal distribution. Efficiency Relative to LSE. Outliers are on x-direction. n=15

			k=2	k=3	Rank	HBR	LSE*
$k_1 = 2$	GT	Int	0.63	0.71	0.75	0.65	0.14
		Slope	0.50	0.67	0.95	0.70	0.00
	TM	Int	0.55	0.81	0.75	0.65	0.14
		Slope	0.26	0.40	0.95	0.70	0.00
$k_1 = 4$	GT	Int	0.70	0.78	0.76	0.66	0.12
		Slope	0.67	0.67	0.89	0.62	0.00
	TM	Int	0.63	0.93	0.76	0.66	0.12
		Slope	0.40	0.67	0.89	0.62	0.00
$k_1 = 6$	GT	Int	0.69	0.79	0.78	0.70	0.13
		Slope	0.42	0.86	0.92	0.71	0.00
	TM	Int	0.59	0.93	0.78	0.70	0.13
		Slope	0.24	0.76	0.92	0.71	0.00

*Entries are mean square errors for the LSE.

Table 5-11 Computation Time in Hours for MSE of GT and TM estimators. Number of Simulations=500

	k=2	k=3	k=4	k=5	k=8
n=15	0.45	1.98	6.26	15.48	28.97
n=25	1.32	11.41	29.28	31.30	78.02
n=30	1.91	22.19			

* No runs were done for k=4, 5 when n=30.

CHAPTER 6 - MULTIPLE LINEAR REGRESSION

6.1 Introduction

We will consider the multiple regression model $Y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + ... + \beta_p x_{pi} + \varepsilon$, $i = 1,...,n$, where $\varepsilon_i 's$ are identically independently distributed with distribution function F. The model can be written in the form, $\mathbf{Y} = \mathbf{X\beta} + \boldsymbol{\varepsilon}$ where $\mathbf{Y}$ is the vector of responses, $\mathbf{X}$ is an n by p+1 matrix and $\boldsymbol{\beta}$ is a p+1 vector of parameters, and $\boldsymbol{\varepsilon}$ is the vector of errors. Multiple linear regression can be used to predict the boiling point of a hydrocarbon using the number of carbon atoms of a hydrocarbon and the molecular weight of that hydrocarbon. It can also be used to predict city's future weekly fuel consumption using the average hourly temperature and the chill index as independent variables. The chill index measures weather-related factors such as the wind velocity and the cloud cover.

The following procedure defines an ${}_{\mathrm{n}}\mathrm{C}_{\mathrm{k}}$ estimator of $\boldsymbol{\beta}$. The data can be written in a matrix form

$$\begin{bmatrix} Y_1 & X_{11} & X_{21} & . & X_{p1} \\ Y_2 & X_{12} & X_{22} & . & X_{p2} \\ . & . & . & . & . \\ . & . & . & . & . \\ Y_n & X_{1n} & X_{2n} & . & X_{pn} \end{bmatrix}$$

Each row of the above matrix corresponds to one experimental unit. We take all possible experimental units of size k from the n experimental units, find the least square estimates using each subsample, and take the Tukey median of those estimates $\hat{\boldsymbol{\beta}}_T$ of $\beta_1, \beta_2, ...\beta_p$. The intercept is estimated by $\text{med}(y_i - \mathbf{x}_i' \hat{\boldsymbol{\beta}}_T)$ where $\mathbf{x_i}$ is the ith column of the design matrix. Note that to estimate the p+1 parameters, k must be greater than or equal to p+1. The breakdown value of $\hat{\boldsymbol{\beta}}_T$ won't exceed 1/(p+1), and as k increases the breakdown value will decrease since the outliers are more likely to appear in the subsamples. Thus it is favorable to choose small values of k to estimate $\boldsymbol{\beta}$.

6.2 Designs and Simulation

We considered different designs from Hawkins and Olive (2003) to study the performance of the proposed estimation procedures with the sample sizes n=15, 25. The vector of parameters was set to 0 and errors are i.i.d. normal.

The first design is the Sphere (S). In this design the columns of the design matrix are randomly sampled from a $N(0, I)$ distribution. The second design is Vslash (V). The columns of the design matrix are randomly sampled from a $N(0, I)$ distribution and each was divided by a randomly selected uniform univariate. This design tends to produce a sprinkling of isolated very remote vectors (Hawkins and Olive 2003). The third design is Disk and Axle (DA). In this design each vector $\mathbf{x}$ of the design matrix is divided into two subvectors, x_1 ([0.8n] by 1) and x_2 ([0.2n] by 1). The first component of $\mathbf{x}_1$ is $N(0,\varepsilon^2)$ and the rest are chosen from $N(0, I)$. The first component of the second subvector $\mathbf{x}_2$ is a scaled chi-squared of p-1 degree of freedom and the rest of the subvector was chosen from $N(0, \varepsilon^2 \mathrm{I})$. This gives a 20% contamination in the x-values. The value of epsilon used here is 4.

After constructing the design matrix as described above, we placed outliers on the response vector randomly (R) or badly (B). For the randomly placed case, 0.2n outliers were placed on a randomly selected observation in the response vector. For the badly-placed option, outliers were placed corresponding to the x-outlying cases. The badly-placed option applies only for the last design (DA) only because the first two designs have no x-outlying cases.

We also considered the outlier size. For the outlying cases, we added 6 to the y value and called this plus (+) or we add +6 or -6 (the sign randomly determined) to the outlying cases and called this plus/minus (+/-). Since the vector of parameters, β, is zero, the outlier placement is applied directly to the y. We also considered DA without placing any outliers on the response vector to see the performance of the estimators when outliers are only in the x direction.

We simulated data from the previous all combinations to get nine designs. Table 6-1 contains the efficiency of each of the three estimators $\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2$ with respect to the LSE. Table 6-2 conatins results for n=25. Efficiency is the ratio of the MSE of the LSE relative to the MSE of the estimator. The methods compared are ${}_nC_k$ estimator, described in previous section, for k=3, 4, LSE, Wilcoxon rank-based, and HBR estimates. The third and fourth column represent ${}_nC_k$ for different values of k.

Table 6-1 The efficiency of each of the estimators ($\hat{\beta}_0,\hat{\beta}_1,\hat{\beta}_2$) relative to LSE for k=3, 4.

Number of simulations=500. n=15, p=2

Design	Est	k=3	k=4	Rank	HBR	LSE*
+ S R	$\hat{\beta}_0$	3.53	3.41	3.69	3.70	1.65
	$\hat{\beta}_1$	1.60	1.44	1.72	1.74	0.64
	$\hat{\beta}_2$	1.63	1.46	1.68	1.81	0.61
+ V R	$\hat{\beta}_0$	3.94	3.54	3.91	3.89	1.54
	$\hat{\beta}_1$	1.48	1.37	1.70	1.69	0.63
	$\hat{\beta}_2$	1.44	1.31	1.71	1.72	0.58
+ DA R	$\hat{\beta}_0$	4.42	4.05	4.79	5.68	1.58
	$\hat{\beta}_1$	0.87	0.92	1.53	1.18	0.16
	$\hat{\beta}_2$	0.94	1.00	1.70	1.15	0.16
+ DA B	$\hat{\beta}_0$	2.69	2.75	2.10	2.38	0.59
	$\hat{\beta}_1$	1.71	1.77	1.10	1.08	0.36
	$\hat{\beta}_2$	2.18	2.20	1.06	1.21	0.35
+- S R	$\hat{\beta}_0$	2.30	2.24	2.21	2.58	0.60
	$\hat{\beta}_1$	2.05	1.77	1.85	2.28	0.76
	$\hat{\beta}_2$	1.94	1.63	1.91	2.38	0.77
+- V R	$\hat{\beta}_0$	2.31	2.30	2.42	2.48	0.65
	$\hat{\beta}_1$	1.87	1.61	1.83	2.17	.79
	$\hat{\beta}_2$	1.84	1.62	1.64	1.93	.77
+- DA R	$\hat{\beta}_0$	2.18	2.21	2.45	2.62	0.67
	$\hat{\beta}_1$	1.02	1.04	1.55	1.26	0.21
	$\hat{\beta}_2$	1.02	1.05	1.50	1.46	0.19
+- DA B	$\hat{\beta}_0$	1.86	1.90	1.30	1.58	0.37
	$\hat{\beta}_1$	2.08	2.10	1.13	1.28	0.44
	$\hat{\beta}_2$	2.20	2.18	1.24	1.32	0.43
DA	$\hat{\beta}_0$	0.55	0.65	0.59	0.57	0.07
	$\hat{\beta}_1$	0.46	0.59	0.93	0.61	0.03
	$\hat{\beta}_2$	0.45	0.60	0.92	0.63	0.02

* For the LSE MSE's of $\hat{\beta}_0,\hat{\beta}_1,\hat{\beta}_2$ are in the last column.

Table 6-2 The efficiency of each of the estimators ($\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2$) relative to LSE for k=3, 4. Number of simulations=500. n=25, p=2

Design	Est	k=3	k=4	Rank	HBR	LSE*
+ S R	$\hat{\beta}_0$	6.33	5.71	6.43	6.58	1.52
	$\hat{\beta}_1$	1.90	1.58	2.15	2.11	0.36
	$\hat{\beta}_2$	1.71	1.49	2.10	2.07	0.33
+ V R	$\hat{\beta}_0$	6.78	6.28	6.91	6.84	1.47
	$\hat{\beta}_1$	1.86	1.63	2.12	2.16	0.32
	$\hat{\beta}_2$	1.71	1.53	2.16	2.20	0.33
+ DA R	$\hat{\beta}_0$	5.98	5.70	7.08	7.65	1.50
	$\hat{\beta}_1$	0.88	0.84	1.80	1.35	0.08
	$\hat{\beta}_2$	0.97	0.91	1.70	1.47	0.08
+ DA B	$\hat{\beta}_0$	4.39	4.47	3.90	4.19	0.85
	$\hat{\beta}_1$	1.84	1.75	0.91	0.83	0.18
	$\hat{\beta}_2$	1.86	1.74	0.95	0.94	0.20
+- S R	$\hat{\beta}_0$	2.61	2.56	2.59	2.69	0.32
	$\hat{\beta}_1$	1.89	1.66	2.18	2.40	0.42
	$\hat{\beta}_2$	1.94	1.73	2.57	2.90	0.36
+- V R	$\hat{\beta}_0$	2.64	2.64	2.68	2.79	0.38
	$\hat{\beta}_1$	2.00	1.75	2.31	2.60	0.35
	$\hat{\beta}_2$	1.95	1.74	2.26	2.45	0.37
+- DA R	$\hat{\beta}_0$	2.97	3.16	3.05	3.26	0.36
	$\hat{\beta}_1$	1.07	1.05	1.77	1.79	0.08
	$\hat{\beta}_2$	0.99	0.96	1.66	1.62	0.07
+- DA B	$\hat{\beta}_0$	2.30	2.27	1.79	2.09	0.26
	$\hat{\beta}_1$	2.32	2.14	1.02	1.19	0.25
	$\hat{\beta}_2$	2.11	1.98	1.00	1.07	0.25
DA	$\hat{\beta}_0$	0.53	0.57	0.56	0.50	0.04
	$\hat{\beta}_1$	0.47	0.53	0.98	0.59	0.01
	$\hat{\beta}_2$	0.47	0.54	0.95	0.63	0.01

* For the LSE MSE's of $\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2$ are in the last column.

A careful look at those designs showed that designs +SR,+VR, +-SR, and +-VR have outlying observations in the y-space. Designs +DAR and +-DAR have some observations with extreme response values and others with extreme predictor values. Designs +DA B and +-DA B have large Cook's distance. Design DA has outlying observations only in the x-space. For the designs +DA B and +-DA B, $_{\mathrm{n}}\mathrm{C}_2$ gives best results. The outliers in this design are badly placed, i.e., outlying y values were placed on the experimental units with outlying x- values. This is consistent with the results obtained from the simple linear regression when outliers are in the xy-direction and the percentage of contamination is between 0.20 and 0.30 because $_nC_k$ method (Tukey median, k=2) was more efficient than other methods. For design DA, the outliers are only in the x-direction and this makes LSE give perfect fit.

For designs +SR,+VR, +-SR, and +-VR, HBR is the best method. In these designs, outliers are in the y-direction In designs +DAR and +-DAR, the Wilcoxon rank-based is more efficient than $_nC_k$ and HBR because there are outliers in the x-values that have some contamination. This is consistent with the results of Chang et el(1999).

The improvement in $_nC_k$ is when data comes from the designs +DA B or +-DA B. For both designs outliers are badly placed so one might look at different contamination percentage for one of them. The design +DA B was studied under different the contamination percentage, 0.05, 0.10, 0.15 and 0.20 when the sample size n=25 and k=3. The efficiencies with respect to the least square estimator are in Table 6-3. For 0.20 contamination the efficiencies are from Table 6-2. Table 6-3 shows that $_nC_k$ outperforms other methods for those four contamination percentages. It is good to note that the efficiency of $_nC_k$ is the highest for 0.10 contamination and goes down as the percentage of contamination increases.

Table 6-3 Efficiencies for the design + DA B with different contamination percentage. n=25

Design + DA B	Est	k=3	Rank	HBR	LSE*
$\varepsilon = 0.05$	$\hat{\beta}_0$	1.43	1.17	1.27	0.12
	$\hat{\beta}_1$	2.76	1.49	1.80	0.19
	$\hat{\beta}_2$	3.09	1.54	1.95	0.21
$\varepsilon = 0.10$	$\hat{\beta}_0$	2.51	2.03	2.17	0.25
	$\hat{\beta}_1$	3.18	1.29	1.81	0.23
	$\hat{\beta}_2$	3.20	1.35	1.72	0.24
$\varepsilon = 0.15$	$\hat{\beta}_0$	3.58	2.79	2.97	0.50
	$\hat{\beta}_1$	2.68	1.05	1.20	0.22
	$\hat{\beta}_2$	2.51	1.06	1.21	0.22
$\varepsilon = 0.20$	$\hat{\beta}_0$	4.39	3.90	4.19	0.85
	$\hat{\beta}_1$	1.84	0.91	0.83	0.18
	$\hat{\beta}_2$	1.86	0.95	0.94	0.20

* For the LSE MSE's of $\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2$ are in the last column.

CHAPTER 7 - SUMMARY AND CONCLUSION

In Chapter 2 we considered estimators of the median of a symmetric distribution. We conclude that the GHL estimator doesn't show much improvement over the HL estimator. The increase in the computation cost of the GHL estimator as k increases is another drawback. The robust and efficient HL estimator can be used to estimate location parameter of the model except for the double exponential distribution where the sample median should be used to estimate the population median.

When deriving ${}_nC_k$ estimate of the variance in Chapters 3 and 4 we had to obtain bias adjustment factors that depend on n, k, and the underlying distribution. In Chapter 3 we considered the normal distribution. We concluded that if the data are normally distributed and there are some outliers in the data, we can use the estimator $d_{n,2}\hat{\sigma}_2^2$. Comparing this estimator to other estimators in the literature showed an improvement in efficiency with respect to the sample variance. For Chapter 4 when data comes form exponential or double exponential distribution the estimator $d_{n,2}\hat{\sigma}_2^2$ showed high efficiency with respect to the sample variance. The estimator $d_{n,2}\hat{\sigma}_2^2$ is robust because it has high breakdown value of 0.29 and efficient as seen in the simulation results of Chapter 3 and 4. This estimator is recommended when outliers exist in normal data or data have exponential distribution or double exponential distribution.

In simple linear regression, the efficiency and robustness of the estimator depends on the number of outliers and their direction in the data. Generally for small samples the ${}_nC_k$ estimation didn't show too much improvement over rank-based and HBR estimation in the simple linear regression model except in one special case i.e. when the sample size is 25 and there are 6 outliers in the xy-direction. Here TM with k=2 outperforms other estimators.

In multiple linear regression we used the concept of data depth to order the possible ${}_nC_k$ estimators and took the Tukey median as the actual estimator. The problem with this approach is the computation time and the complication of programming, but it appears to be efficient in one case i.e. when there are observations with outliers in both x-values and y-values i.e. outliers with high Cook's distance.

The use of ${}_nC_k$ estimators in conjunction with data depth is a new idea that may be useful if computational issues can be dealt with. Generally this technique showed some improvement over other methods especially for k=2. Larger values of k will give a higher chance for the outliers to appear in the data, and thus the estimator would break down faster. Over the class of problems considered here, the ${}_nC_k$ estimators generally seem to do better for smaller k. In particular k=2 often gave the best efficiency among the ${}_nC_k$ estimators. This is fortunate because k=2 is less computationally intensive than larger k. The ${}_nC_k$ technique is quite general and may be used both in univariate and multivariate cases.

REFERENCES

Bickel, P.J. and Lehman, E.L.(1979). Descriptive Statistics for Nonparametric Models. IV. Spread. In: J. Jureckova, Ed., Contributions to Statistics. Hajek Memorial Volume. Academia, Prague, 33-40.

Chang, W. H., Mckean, J. W., Naranjo, J. D., and Sheather, S. J. (1999). High-Breakdown Rank Regression. JASA, 94, 205-219.

Choudhury, J. and Serfling, R. J. (1988). Generalized order statistics, Bahadur representations, and sequential nonparametric fixed-width confidence intervals, Journal of Statistical Planning and Inference, 19, 269-282.

Donoho, D. L. and Gasko, M. (1992). Breakdown properties of location estimates based on halfspace depth and projected outlyingness. Ann. Stat., 20, 1803-1827.

Gross, A. M. (1976). Confidence interval robustness with long-tail symmetric distributions. JASA, 71, 409-416.

Jaeckel, L. A. (1972). Estimating regression coefficients by minimizing the dispersion of the residuals. Ann. Math. Stat., 43, 1449-1458.

Hawkins, D. M. and Olive, D. J. (2003). Inconsistency of resampling algorithms for high-breakdown regression estimators and a new algorithm (with discussion). JASA, 97, 136-159.

Hettmansperger, T. P. and Mckean, J. W. (1998). Robust nonparametric statistical methods. Edward Arnold, London.

Hoeffding, W. (1948). A class of statistics with asymptotically normal distribution. Ann. Math. Stat., 19, 293-325.

Hugg, J., Rafalin, E., Seyboth, K., and Souvaine, D. L. (2005). An experimental study of old and new depth measures. Workshop on Algorithm Engineering and Experiments. Springer-Verlag Lecture Notes in Computer Science.

Hugg, J., Rafalin, E., Seyboth, K., and Souvaine, D. L. (2006). Depth Explorer-A Software Tool for Analysis of Depth Measures. International conference on Robust Statistics.

Johnson, D. E., McGuire, S. A., and Milliken, G. A. (1978). Estimating σ^2 in the presence of outliers. Technometrics, 20, 441-455.

Johnson, N. L., Kotz, S., Balakrishnan, N. (1994). Continuous univariate distributions. 2nd Ed. New York:Wiley.

Lax, D. A. (1985). Robust estimators of scale: Finite-sample performance in long-tailed symmetric distributions. JASA, 80, 736-741.

Liu, R. Y. (1990). On a notion of data depth based on random simplices. Ann. Stat., 188, 405-414.

Lohr, S. (1999). Sampling: Design and Analysis, Pacific Grove, CA: Duxbury Press.

Mehrotra, D. V. (1995). Robust elementwise estimation of a dispersion matrix. Biometrics, 51, 1344-1351.

Milton, J. S. (1999). Statistical methods in the biological and health sciences. McGraw-Hill Series in Probability and Statistics.

Morton-Jones, T., and Henderson, R. (2000). Generalized least squares with ignored errors in variables. Technometrics, 42, 366-375.

Naranjo, J. D., and Hettmansperger, T. P. (1994). Bounded-influence rank regression. J. of the Royal Stat. Society, Ser. B, 56, 209-220.

Oja, H., and Niinimaa, A. (1984). On robust estimation of regression coefficients, Research Report, Department of Applied Mathematics and Statistics, University of Oulu, Finland. [3.6, 5.1]

Rousseeuw, P.J. and Leroy, A. M. (1987). Robust regression and outlier detection. New York:Wiley.

Rousseeuw, P.J. and Ruts, I. (1998). Constructing the bivariate Tukey median. Statistica Sinica, 8, 827-839.

Saleh, A. K. Md. E. (1992). Nonparametric statistics and related topics. Proceedings of the International Symposium, Ottawa, Canada, 5-8 May 1991. Amsterdam, The Netherlands: North-Holland.

Serfling, R.J. (1980). Approximation Theorems of Mathematical Statistics. Wiley, New York.

Serfling, R.J. (1984). Generalized L-, M- and R-statistics. Ann. Stat., 12, 76-86

Terpstra, J. (2007). Personal communication.

Thiel, H. (1950). A rank-invariant method of linear and polynomial regression analysis, III. Proc. Kon. Ned. Akad. V Wetensch. A 53, 1397-1412.

Tukey, J. W. (1975). Mathematics and the picturing of data. In Proc. International Congress of Mathematicians Vancouver 1974, 2, 523-531. Canadian Math. Congress, Montreal.

Wilcox, R. R. (2005). Introduction to robust estimation and hypothesis testing. 2nd Edition. Elsevier, Academic Press.

Zuo, Y. and Serfling, R. (2000a). General notions of statistical depth function. Ann. Stat., 28, 461-482.

Zuo, Y., Cui, H., and He, X. (2004). On the Stahel-Donoho estimator and depth-weighted means of multivariate data. Ann. Stat., 32, 167-188.

OTHER SOURCES

http://www.ndsu.nodak.edu/instruct/terpstra/Misc/

http://www.knowledgeminer.com/

http://psychology.usc.edu/faculty_homepage.php?id=43

Appendix A - R-programs

R code needed for simulation results is in the attached zip file. There are several folders named by the chapters and inside each folder, files are named either by the Table number, the job it does or both. Here are also the code for Chapter 2 simulations and some of the main functions used in the dissertation. Whenever $N = \begin{pmatrix} n \\ k \end{pmatrix}$ is larger than 5000, 5000 subsamples were taken without replacement and this is as efficient as taking all subsamples of size k. All simulation results can be reproduced using the seed 2010.

The following subset function takes the sample size, n, the subset size, k, and the vector of observations, v as input. It calculates all subsets of size k given v. For n=15, it was used for any k. For n=25, it was used only for k=2,3 only because for other values of k, $N = \begin{pmatrix} n \\ k \end{pmatrix}$ is huge (12650 for k=4) so 5000 samples of size k without replacement were taken using the function "sample(x, k ,replace=FALSE)" to find the variance estimator.

```
subsets <- function(n, r, v = 1:n) # works in S or R
  if(r <= 0) vector(mode(v), 0) else
    if(r >= n) v[1:n] else {
   rbind(cbind(v[1], Recall(n-1, r-1, v[-1])),
             Recall(n-1, r, v[-1]))
}
```

```
GHn15=function(x,k){               # Generalized Hodges-Lehmann estimator
                                   # for n=15.
 A=round(subsets(length(x),k,x),4) # all subsets of
                                   # size k are in
                                   # rows of A.
   means=apply(A,1,mean)            # mean function.
                                   #applied to rows.
   ans=median(means)                # the median of
```

```
  ans
   }


GHn25=function(x,k){                    # Generalized Hodges-Lehmann estimator
                                        # for n=25. k=2,3.
 A=round(subsets(length(x),k,x),4)      # all subsets of
                                        # size k are in
                                        # rows of A.
    means=apply(A,1,mean)               # mean function.
                                        #applied to rows.
    ans=median(means)                   # the median of

  ans
   }


GHsam=function(x, k,S=5000) {           # can be used for any sample size.
                                        # This is for n=25  k=4,5.
                                        # only change k and S here
                                        # we take random  S
                                        #samples without
                                        # replacement
                                        # each of size k from the
                                        #original data.  S here replaces  (n choose k).
  set.seed(2010)
  A=lapply(1:S, function(i) sample(x,k,replace=FALSE))


                                        # A contains several resamples.

  mymat <- do.call('rbind',A)

                                         # to convert a list to a matrix.
                                         #    resamples are in rows.

  means=apply(mymat,1,mean)
  ans=median(round(means,4))
  ans
 }

#########Comparing generalized Hodges-Lehmann estimator to sample mean
# under normal distribution ###.
# For other distributions, one has to change the pdf function.
# Sample size=15.
```

```
set.seed(2010)
B=500  # Number of simulations.
k1=0
n=15
 res=lapply(1:B, function(i) rnorm(n))


estmean=sapply(res,mean,simplify=T)# sample mean for each resample.

varm=var(estmean)                        # variance of the sample mean.

msem=(mean(estmean))^2+var(estmean)
                                         # MSE of the sample mean.

k=1
ests=sapply(res,median, simplify=T)

var1=var(ests)

mse1=(mean(ests))^2+var(ests)

bias1=mean(ests)

ef1=msem/mse1
eff1=varm/var1


####****#######************

k=2
ests=sapply(res,function(x) GHn15(x,k), simplify=T)

var2=var(ests)

mse2=(mean(ests))^2+var(ests)

bias2=mean(ests)

ef2=msem/mse2
eff2=varm/var2

#*****######************
k=3
ests=sapply(res,function(x) GHn15(x,k), simplify=T)

var3=var(ests)
```

```
mse3=(mean(ests))^2+var(ests)

bias3=mean(ests)

ef3=msem/mse3
eff3=varm/var3

#########*#######************
k=4
ests=sapply(res,function(x) GHn15(x,k), simplify=T)

var4=var(ests)

mse4=(mean(ests))^2+var(ests)

bias4=mean(ests)

ef4=msem/mse4
eff4=varm/var4

#########*#######************
k=5
ests=sapply(res,function(x) GHn15(x,k), simplify=T)

var5=var(ests)

mse5=(mean(ests))^2+var(ests)

bias5=mean(ests)

ef5=msem/mse5
eff5=varm/var5
ef1   #eff based on MSE.
ef2
ef3
ef4
ef5
#########Comparing generalized Hodges-Lehmann estimator to sample mean
# under normal distribution ###.
# For other distributions, one has to change the pdf function.
# Sample size=25.
#For small k we use the function “GHn25” and for large k we use
# GHsam.
############################
set.seed(2010)
```

```
B=500
k1=0
n=25


 res=lapply(1:B, function(i) round(rnorm(n),4))


estmean=sapply(res,mean,simplify=T)        # sample mean for each resample.

varm=var(estmean)                          # variance of the sample mean.

msem=(mean(estmean))^2+var(estmean)    # MSE of the sample mean.

##############################
k=1
ests=sapply(res,median, simplify=T)

var1=var(ests)

mse1=(mean(ests))^2+var(ests)

bias1=mean(ests)

ef1=msem/mse1
eff1=varm/var1

####*****#######************
k=2
ests=sapply(res,function(x) GHn25(x,k), simplify=T)

var2=var(ests)

mse2=(mean(ests))^2+var(ests)

bias2=mean(ests)

ef2=msem/mse2
eff2=varm/var2

#####*****#######************
k=3
ests=sapply(res,function(x) GHn25(x,k), simplify=T)

var3=var(ests)
```

```
mse3=(mean(ests))^2+var(ests)

bias3=mean(ests)

ef3=msem/mse3
eff3=varm/var3

#########*#######************
k=4
ests=sapply(res,function(x) GHsam(x,k), simplify=T)

var4=var(ests)

mse4=(mean(ests))^2+var(ests)

bias4=mean(ests)

ef4=msem/mse4
eff4=varm/var4

#########*#######************
k=5
ests=sapply(res,function(x) GHsam(x,k), simplify=T)

var5=var(ests)

mse5=(mean(ests))^2+var(ests)

bias5=mean(ests)

ef5=msem/mse5
eff5=varm/var5


##  The varnoc function finds the estimate of the variance without adjustment ##
  varnoc=function(x,k) {
     A=round(subsets(length(x),k,x),2)         # all subsets of
                                               # size k are in
                                               # rows of A.
     varian=apply(A,1,var)                     # variance function
                                               #applied to rows.
     ans=median(varian)                        # the median of
```

```
                                                  #variances.
   ans
   }


# vard function to calculate the variance estimator from Johnson, Mcguire, and
Milliken(1978).
                                  # This function calculates Vk*.
                                   # Assuming there are k1 outliers in the data and.
  vard=function(x,k1) {
                                 # this function calculates Vk*.
                                 # note that it is for n=15,25, 100
                                 # Assuming there are k1 outliers in the data and
                                 # it is not neccessary equal to the true # of outliers.
  pairs=pairup(x)
 n=length(x)
 k2=n-k1
# vk=(n-k1-1)/(k1*(k1-1)+k2*(k2-1))
 if (n==15)      vk=switch(k1,'1'=0.5913,'2'=0.4071,'3'=0.2953)
  if (n==25)       vk=switch(k1,'1'=0.6995,'2'=0.5418,'3'=0.4339)
 if (n==100)       vk=switch(k1,'1'=0.8823,'2'=0.8045,'3'=0.7421)
 diff=pairs[,1]-pairs[,2]      # find all pairs Xi-Xj
 u=rev(sort(abs(diff)))        # find the Uij
                               #  sort from largest to smallest.
 a=u^2
 term1=sum(a)
 term2=sum(a[1:(k1*k2)])    # the second expression in Vk.

 ans=(term1-term2)/(k1*(k1-1)+k2*(k2-1))
  ans=ans/vk # scaling as in the paper.
 ans
 }



# pairup function finds all pairs from the given vector#
pairup=function(x,type="less") {
 x=as.matrix(x)
 n=dim(x)[1]
i=rep(1:n,rep(n,n))
 j=rep(1:n,n)
c1=apply(x,2,function(y){rep(y,rep(length(y),length(y)))})
 c2=apply(x,2,function(y){rep(y,length(y))})
 ans=cbind(c1,c2)
ans=switch(type, less=ans[(i<j), ], leq=ans[i<=j, ], neq=ans)
ans }

## Function to calculate M-estimate of the variance##.
Vms=function(x){
 n=length(x)
k=0.973+3.353*(10^(-6))*(n^(1.5))-3.686*(10^(-7))*(n^3)+3.091*(n^(-1.5))
# if  n>100  k=.973.
 const1=n*2.6*mad(x)/sqrt(n-1)
 u=(x-median(x))/(2.6*mad(x))

 num=ifelse(abs(u)<=pi,(sin(u))^2,0)          # finding ui's such that abs(ui)<=pi
                                              # This for the expression in num.
 num=sqrt(sum(num))                           # the sum in the numerator .
 den=ifelse(abs(u)<=pi,cos(u),0)              # finding ui's such that (ui)<=pi
                                              # this is for the expression in the den.
 den=abs(sum(den))                            # the sum in the denominator.
 const2=atan(num/den)
  sms=const1*const2
 v=k*(sms)^2
 v
```

```
}



###########################################***********#######
### The function “design “ generates data from the designs in Chapter 6.###
### One has to specify the design and choose the right character value for ”method” and
“out”.  For example if we need data from design 3 with n=10 and two independent
variables, we call the function by design(10, 3, method=”DA”, eps=4, out=”PR”).

design=function(n, p, method, eps=4,out){
#out takes four character values:
# Design:  + S  R   method="S" out="PR"
# Design:  + V  R   method="V" out="PR"
# Design: + DA R   method="DA" out="PR"
# Design: + DA B   method="DA" out="PB"
# Design: +/-  S  R   method="S" out="PMR"
# Design: +/-  V  R   method="V" out="PMR"
# Design: +/-  DA R   method="DA" out="PMR"
# Design: +/-  DA B   method="DA" out="PMB"

 method=as.character(method)
 out=as.character(out)
 mu=rep(0,p)
 mu=as.vector(mu)
 sigma=diag(p)
 a=runif(p)
 k1=0.8*n
 k2=0.2*n
  if(method=="S")  X=mvrnorm(n,mu,sigma) else
 if(method=="V") { X=mvrnorm(n,mu,sigma); apply(X,2,function(x){x/runif(1)}) } else
```

```
     if(method=="DA")
     X=rbind(rnorm(p,0,eps^2),mvrnorm(k1-1,mu,sigma),rchisq(p,p-1)/sqrt(2*p),mvrnorm(k2-1,mu,eps^2*sigma))

    if(out=="PR"){ y=rnorm(n);
    i=sample(seq(1:n),k2,replace=FALSE);
    y[i]=y[i]+6 }



    if(out=="PB") {
    y=rbind(as.matrix(rnorm(1)+6),
    as.matrix(rnorm(k1-1)),
    as.matrix(rnorm(1)),
    as.matrix(rnorm(k2-1)+6)) }

    if(out=="PMR") {
    y=rnorm(n);
    i=sample(seq(1:n),k2,replace=FALSE);
    ran=sample(c(-1,1),k2,replace=TRUE);
    y[i]=y[i]+ran*6
    }
    if(out=="PMB") {
    y=rbind( as.matrix(rnorm(1)+sample(c(-1,1),1)*6),
    as.matrix(rnorm(k1-1)),
    as.matrix(rnorm(1)),
    as.matrix(rnorm(k2-1)+sample(c(-1,1),k2-1,replace=TRUE)*6)  )
     }

    Z=cbind(y,X)
    Z
```

```
}

##########################################*******###############
# The function “design9” generate data  for design “DA” the last design in Chapter 6.
design9=function(n,p,eps=4){
library(MASS)
 mu=rep(0,p)
 mu=as.vector(mu)
 sigma=diag(p)
 a=runif(p)
 k1=0.8*n
 k2=0.2*n
  X=rbind(rnorm(p,0,eps^2),mvrnorm(k1-1,mu,sigma),rchisq(p,p-1)/sqrt(2*p)
,mvrnorm(k2-1,mu,eps^2*sigma))
 Z=cbind(rnorm(n),X)
 Z
 }
### The function “betanck1” calculates the generalized form of Thiel estimator as
explained in Chapter 5. It takes a matrix with the first column as the response and the
second column as the predictor. ###
 betanck1=function(X){  # Y is a vector of responses. is the first column of X.
                        # X1 is a vector of values for the predictor. It is the
 second column of X.
   n=length(X[,1])
        N=choose(n,k)
       a=seq(1:n)
       S=5000
       numbers=subsets(n,k,a)
       # to take all N rows of size k from
       #the matrix, I do that for the indexes
       # then apply it to the rows of A
```

```
      # to get all matrices of size k.
 g=function(mat) { # X is a submatrix containing nck responses and
       # nck values of the predictor.
       ans=lm(mat[,1]~mat[,2])$coef
        ans=ans[2]
        ans }
   if (N<=5000) res=lapply(1:N, function(i) X[numbers[i,],]) else
  res=lapply(1:S, function(i) X[sample(n,k,replace=FALSE),]) # When N>5000.
     betam=sapply(res,g, simplify=TRUE) # No problem at all.
    est=median(betam,na.rm=T)
    int=median(X[,1]-est*X[,2]) # intercept estimate.
   ans=matrix(c(int,est))
   ans
      }
#########################################################
###The function "betanck2" finds the regression estimates in the simple linear regression
using Tukey median approach #######
 betanck2
function(X){  # Y is a vector of responses which is the first column of X.
 # X1 is a vector of values for the predictor which is the second column of X.
  n=length(X[,1])
   # Will give an nck estimate of beta based on halfspace depth.
    # After centering Y and X  fit regression line.
       N=choose(n,k)
      a=seq(1:n)
      numbers=subsets(n,k,a)
      # to take all N rows of size k from
      #the matrix, I do that for the indexes
      # then apply it to the rows of A
      # to get all matrices of size k.
g=function(mat) { # X is a submatrix containing nck responses and
```

```
            # nck values of the predictor.
          ans=lm(mat[,1]~mat[,2])$coef
          ans }
   S=5000
   if(N<=5000)  res=lapply(1:N, function(i) X[numbers[i,],]) else
   res=lapply(1:S, function(i) X[sample(n,k,replace=FALSE),] )
     betam=sapply(res,g, simplify=TRUE) # No problem at all.
     betams=t(betam) # transpose because dmean receives a matrix
                # of vectors each in a row.
                # vectors must be in the rows of the matrix.
   est=dmean.for(betams,tr=0.5)      # dmean(,0.5) will  give the Tukey
   est=matrix(est)
   est      }
```

####################################***#######****

####The function "betanckm" calculates the regression estimates using Tukey median in multiple linear regression ######

## Data Depth functions:

###The function dmean.for calculates Tukey median on unix machine.
## This function needs eight Fortran functions. They are "depth2.for", "depth3.for","fdepth.for","fdepthv2.for","depth2.o","depth3.o","fdepth.o", and "fdepthv2.o". The functions ending with " .o " must to be stored in the directory where R is being run. The functions ending with ".for" need to be sourced. These functions are also saved in the Folder R functions under "data depth functions" and under unix files. Note that for "dmean.for" to calculate Tukey median the trimming proportion should be 0.5, "tr=0.5", and it takes a matrix with observations in rows ####

```
 dmean.for
function(m,tr=.2,v2=T,center=NA){
```

```
# Compute multivariate measure of location
# using Donoho-Gasko method.
# v2=T, use slower but more accurate approximation
# of halfspace depth.
# v2=F, use only projection based on lines through center
# and each of n points.
if(is.list(m))m<-matl(m)
if(!is.matrix(m))stop("Data must be stored in a matrix or in list mode.")
if(ncol(m)==1){
if(tr==.5)val<-median(m)
if(tr>.5)stop("Amount of trimming must be at most .5")
if(tr<.5)val<-mean(m,tr)
}
if(ncol(m)>1){
m<-elimna(m)
if(ncol(m)==2)temp<-depth2.for(m,plotit=F)
if(ncol(m)==3)temp<-depth3.for(m)
if(!v2 && ncol(m)>3)temp<-fdepth.for(m,center=center)
if(v2 && ncol(m)>3)temp<-fdepthv2.for(m)
mdep<-max(temp)
flag<-(temp==mdep)
if(tr==.5){
if(sum(flag)==1)val<-m[flag,]
if(sum(flag)>1)val<-apply(m[flag,],2,mean)
}
if(tr<.5){
flag2<-(temp>=.2)
if(sum(flag2)==0)flag2<-flag
if(sum(flag2)==1)val<-m[flag2,]
if(sum(flag2)>1)val<-apply(m[flag2,],2,mean)
}} val }
```

Appendix B - Programs Checks

Small data sets were used to verify that the functions are right. I find the estimator by hand and use the program to find it. For example if n=5 and we have the observation, 4, 7, 2, and 8, then all subsample of size 3 are $\{4,7,2\},\{4,7,8\},\{4,2,8\},\{7,2,8\}$ and the variances of each subsample are 6.333, 4.333, 9.333, and 10.333 respectively. The median of the variances is 7.833. The program varnoc gives 7.833 for k=3.

For the efficiency calculation in section 5.5, the values of the efficiencies are consistent with the efficiencies calculated in Chang et el (1999).

In the regression model, I also tried the programs for small data sets. The Tukey median of the data set (1,1) ,(1,4),(4,1),(4,4), and (3,3) is (3,3). “dmean.for(x,0.5)=(3,3)”. The program gives this value too.