

The copyright of this thesis vests in the author. No quotation from it or information derived from it is to be published without full acknowledgement of the source. The thesis is to be used for private study or non-commercial research purposes only.

Published by the University of Cape Town (UCT) in terms of the non-exclusive license granted to UCT by the author.

Robust Beta Estimation and Applications

by

Mohamed Rezah Khan (KHNMOH018)

Dissertation presented in fulfilment of the requirements of the degree of Masters in
Statistical Science

at the

University of Cape Town

Supervisor: Prof CG Troskie

Submission Date: October 2003

Robust Beta Estimation and Applications

ACKNOWLEDGEMENTS

I would like to thank my supervisor, Professor Troskie for guiding my academic career and introducing me to areas of statistics I would not otherwise have explored. I would also like to thank him for his valuable insights, the use of his computer programs and models and most of all his patience in what has been a very long process.

I would like to thank the Foundation for Research and Development for its very generous financial assistance in the first two years of this degree, without which this course of study would not have been possible.

I would like to thank my parents and close friends for their support and encouragement. They gave me the strength to persevere and finish this dissertation.

University of Cape Town

Robust Beta Estimation and Applications

TABLE OF CONTENTS

Synopsis	1
1. Introduction.	7
Part I: Robust regression estimators.	9
2. The adaptive L_p estimator.	10
2.1. Definition of the model.	10
2.2. Estimating p .	10
2.3. Distributional properties of the L_p estimator.	11
2.4. Masking and swamping.	13
2.5. The WLS minimisation algorithm for the adaptive L_p estimator.	15
3. Bounded influence weighted least squares.	16
3.1. Calculation of the weighing factors.	16
3.2. Other bounded influence estimators .	17
4. A simulation study.	22
4.1. Structure.	22
4.2. Design and consequences.	24
4.3 The simulation results.	26
5. A comparison of models.	34
Part II: Modern portfolio theory.	37
6. Notation and assumptions.	37
7. Classical formulations.	40
7.1. The Markowitz formulation and solution.	40
7.2. The efficient frontier.	44
8. The Sharpe index models.	45
8.1. The Sharpe single index model.	45
8.2. The Sharpe multiple index model.	49
8.3. A comparison of the Markowitz and Sharpe single and multiple index models.	53
9. Innovations to the classical models.	
9.1. Generalisation of the Markowitz formulation.	56
9.2. Innovations to the Sharpe index models.	57
9.3. Innovations to the Sharpe multiple index model.	62
9.4. Further innovations to the single and multiple index models by Troskie.	67

Robust Beta Estimation and Applications

TABLE OF CONTENTS

Part III: Application of robust beta estimation to modern portfolio.	70
10. Data used in the analysis.	71
11. Analysis of the shares used.	72
12. The Sharpe single index model.	73
12.1. Ordinary least squares vs L_1 and L_p regression.	73
12.2. Weighted OLS, L_1 and L_p vs non-weighted OLS, L_1 and L_p .	78
12.3. The Sharpe Multiple index model.	85
13. Conclusions.	91
References.	92
Appendix A.	95
Appendix B.	96
Appendix C.	98

SYNOPSIS

Modern portfolio theory was developed by Harry Markowitz more than forty years ago and is now considered to be an indispensable tool in portfolio construction. Sharpe introduced the index models as a simplification of the original Markowitz formulation, as this required fewer parameters to be estimated. One of the premises underlying the Sharpe Index model was that the returns of shares and market proxies followed a normal distribution and under this assumption, model parameters could best be estimated using Ordinary Least Squares (OLS) regression. More recent empirical evidence has however cast doubt over the assumption of normality and has suggested that market returns tend to be non-normal. If the assumption of normality is no longer upheld then OLS or Maximum Likelihood Estimates may no longer produce the best estimates of parameters and hence may compromise optimal portfolio constructions.

In addition to the parameter estimation problems, at the time of the initial formulation of the portfolio models, managers were not allowed to participate in short sales (selling a share one does not own). This practice is now quite common in most developed markets and any portfolio formulation model needs to be generalised to allow for this.

Empirical studies have shown that share returns and the returns of market proxies do not follow a normal distribution but rather seem to follow a skew distribution and has long tails. It is the aim of this thesis to explore robust regression procedures which should be an improvement of OLS when the data does not come from a normal distribution. The robust regression procedures will then be used to estimate the parameters used in the Sharpe Index models and examine whether or not they aid in the portfolio construction process. The robust procedures will be applied to the classical portfolio formulations as well as the generalised models.

Consider the model $Y = X\beta + e$ where $E(e) = 0$, $var(e) = \sigma^2 I$ and $X_{n \times q}$ is a data matrix of explanatory variables.

To estimate β under the assumptions of normality and where $e \sim N(0, \sigma^2 I)$ the maximum likelihood estimate for β is $\hat{\beta} = (X'X)^{-1}X'Y$.

This is also the ordinary least squares estimate (OLS) of β with no explicit distributional assumption of e . When outliers and influential observations are present in the data matrix X , and this typically happens when the underlying distributions has long tails, then the above estimates can be very far removed from the "true β 's".

The adaptive L_p estimator

The so called adaptive L_p estimator was defined by Sposito, Kennedy and Gentle (1977) for the case $q = 2$ and extended by Barr(1981) for the general case. If the

model $Y = X\beta + e$ is redefined as $Y_i = x_{(i)}\beta + e_i$, $i = 1, \dots, n$ with $x_{(i)}$ the i^{th} row of X , then the adaptive estimator minimizes $\sum_{i=1}^n |Y_i - x_{(i)}\beta|^p$, and p is estimated in an adaptive way using the kurtosis of the estimated residuals. The case $p = 2$ is the OLS case. For values of $p = 1$ and $p = \infty$ exact linear programming solutions for β are available. In the research area of robust regression, values of p not equal to those above, have been shown to be important.

To find the adaptive p the algorithm proposed by Gonin and Money (1985b, 89) for non-linear estimation can be used. Let $p = 2$ and fit a model using least squares. Compute the sample kurtosis $\hat{\gamma}_2$ and estimate p by $\hat{p} = \frac{9}{\hat{\gamma}_2} + 1$.

This estimated value of p is used in an iterated procedure to find new estimates for β and new estimates for the residuals. The new sample kurtosis is then computed from the residuals, and from $\hat{p} = \frac{9}{\hat{\gamma}_2} + 1$ a new estimate for p . The process is repeated until no further change in the values of p is detected.

The algorithm of Sposito *et al* considers minimizing: $I = \sum_{i=1}^n W_i R_i^2$ where the R_i are the residuals and W_i are weighting factors. This method is known as Weighted Least Squares (WLS).

Bounded Influence adaptive (BWLS)

The bounded influence adaptive WLS estimator BWLS uses as starting point the Mallows weights for the design matrix X after which the OLS residuals are used to start the iterative weighting procedure to estimate p . Thus the BWLS will minimize $L_p^w = \sum_{i=1}^n w_i |Y_i - x_{(i)}\beta|^p$ where w_i are the Mallows weights. The BWLS will initially down weight the effect of influential observations in the X - space for a hopefully better start of the iteration process but after the initial estimate of β and the residuals the weights w_i remain fixed and the remaining iteration depends only on the *new weighted* residuals of the next fit through the iteration process described above.

The bounded influence estimators developed by de Jongh, de Wet and Welsh (1989) use the following weight function introduced by Mallows (1973).

Denote the n observations of $(j - 1)^{th}$ independent variables by x_{1j}, \dots, x_{nj} for $j = 2, \dots, q$. Order the n observations $x_{(1)j}, \dots, x_{(n)j}$, and define r_{1j}, \dots, r_{nj} as the vector of ranks of x_{1j}, \dots, x_{nj} . Let $L = [\pi n] + 1$ and $U = n + 1 - L$. The weights associated with the $(j - 1)^{th}$ independent variable are now defined as

$$\begin{aligned} W_{ij} &= 1, L \leq r_{ij} \leq U, \\ &= \frac{x_{(L)j} - x_{(U)j}}{D_{ij}}, r_{ij} < L, \\ &= \frac{x_{(U)j} - x_{(L)j}}{D_{ij}}, r_{ij} > U, \end{aligned}$$

where $D_{ij} = 2x_{ij} - x_{(U)j} - x_{(L)j}$ ($i = 1, \dots, n$).

The Mallows weights are now defined as $W_i = \sum_{j=2}^n W_{ij}$.

Chatterjee and Mächler (1997) proposed on similar lines a bounded iterative weighted least squares procedure which according to them performed very well, when used with most data sets, except those that may be severely effected by masking and swamping (i.e. the set of Hawkins, Bradu and Kass (1989)). Their algorithm is a weighted least squares algorithm (*WLS*) which gives low weights to points with: (i) high leverage and (ii) large residuals. The weights are determined by the residuals and is therefore also an iterative one.

A simulation study

To compare the least squares, robust and bounded (weighted) robust and least squares estimators a simulation study was conducted on data provided by de Jongh. This data was used by de Jongh (1984) to compare several estimators of β under normal and heavy tailed distributions for the error term e .

The simulation study showed that *OLS* estimates can be used for normal data as well as for data with slightly longer tails. Under normal theory the estimates will be maximum likelihood and the powerful distributional results on hypothesis testing, confidence intervals, tests for outliers, etc., are available. When the kurtosis is very large, then a bounded procedure that bounds both the leverage points and the residuals should be used.

Modern Portfolio Theory

Consider a portfolio consisting of p stocks. Let the vector of stock returns for the portfolio be $\mathbf{R} = (R_1 \cdots R_p)'$ with the expected returns $E(\mathbf{R}) = \boldsymbol{\mu}$. Assume that the covariance between returns of different shares are non-zero thus the covariance

matrix of the stock returns is $\boldsymbol{\Sigma} = E(\mathbf{R} - \boldsymbol{\mu})(\mathbf{R} - \boldsymbol{\mu})'$. $\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_{11} & \cdots & \sigma_{1p} \\ \vdots & & \vdots \\ \sigma_{p1} & \cdots & \sigma_{pp} \end{pmatrix}$ where

$\sigma_{ii} = \sigma_i^2$ is the variance of the i 'th and σ_{ij} the covariance between the i 'th and j 'th stock. Further assume that $\mathbf{R} \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, the multivariate normal. The fact that the weights of each of the stocks are constrained to sum to one implies that the portfolio manager cannot invest more money in stocks than he has available. Let $\mathbf{W} = (w_1 \cdots w_p)'$ and P be the return of the portfolio. Thus $P = \mathbf{W}'\mathbf{R}$ subject to $\sum_{i=1}^p w_i = 1$. The expected return of the portfolio is $E(P) = \mathbf{W}'\boldsymbol{\mu}$ and the variance is $var(P) = \mathbf{W}'\boldsymbol{\Sigma}\mathbf{W}$

Our portfolio problem is then

$$\begin{aligned}\max_{w_i} E(P) &= \mathbf{W}'\boldsymbol{\mu} \\ \min_{w_i} \sigma_p^2 &= \mathbf{W}'\boldsymbol{\Sigma}\mathbf{W} \\ \sum_{i=1}^p w_i &= 1.\end{aligned}$$

This formulation is due to Harry Markowitz. Markowitz also constrained the weights such that $0 \leq w_i \leq 1$ for $i = 1 \dots p$.

The Efficient Frontier

To these ideas, Markowitz added the concept of the Efficient Frontier. A portfolio is called efficient if: for a given amount of risk, the expected return is maximised, or for a given amount of return expected risk is minimised, and the portfolio is legitimate (no negative holdings). Let σ_p^2 represent the x-axis and E_k represent the y-axis on a Cartesian Plane. The plot of all efficient portfolios on this set of axis is called the efficient frontier.

The Sharpe Index Models

Let R_t be the return of the portfolio and I_t be the return of the market proxy at time t . The model can thus be written as $R_t = \alpha + \beta I_t + e_t$ where $e_t \sim N(0, \sigma^2)$. This is known as the Sharpe index model. Let $E(I) = \mu_I$ and $var(I) = \sigma_I^2$. For each stock let $E_i = E(R_i) = \alpha_i + \beta_i \mu_I$. The variance is $var(R_i) = \sigma_{ii} = \sigma_i^2$ and the covariance is $cov(R_i, R_j) = \beta_i \beta_j \sigma_I^2$. Thus the total number of quantities that we need to estimate for the Sharpe model is $3p + 2$. It is important to note that the Markowitz model requires the estimation of $p + \frac{1}{2}p(p + 1)$ quantities.

Similarly the multi - index model can be written as

$$\begin{aligned}R_{it} &= \alpha_i + \beta_{i1}I_1 + \beta_{i2}I_2 + \dots + \beta_{iM}I_M + e_{it}, \\ i &= 1, \dots, p, \quad t = 1, \dots, N.\end{aligned}$$

Empirical study on the JSE

It is widely believed that stock and index returns do not conform to normality and in particular have distributions with long tails and large kurtosis. In this section robust beta estimation is applied to the Sharpe Index models and an empirical analysis is performed using nine shares that are listed on the Johannesburg Stock Exchange. The aim of this analysis to determine whether or not robust beta estimation

can produce superior regression fits as well as improved efficient frontiers. The shares selected for the analysis are:

- Anglos
- De Beers
- Gencor
- Implats
- Metlife
- Remgro
- SAB
- Sappi
- Stanbic

These shares were selected because they represent a cross section of companies listed on the JSE. The chosen shares all have large market capitalisations and do not have any of the problems associated with thin trading, a problem that afflicts a large number of shares listed on the JSE. The problem of thin trading is discussed by Bowie (1994) and will not be dealt with in this thesis. For the Sharpe Single Index model the JSE All Share Index was used as the market proxy.

For the Sharpe Multiple Index model the following indices were used as market proxies:

- JSE All Share Index
- JSE Banks Index
- JSE Gold Index
- JSE Insurance Index
- JSE Platinum Index

The robust procedures used are non-weighted L_1 and L_p beta estimates and weighted OLS, L_1 and L_p beta estimates using the Mallows weights discussed in earlier.

The study showed that even though the R^2 value for the OLS estimates are generally higher than that of the L_1 and L_p estimates the differences are quite small. The residual error terms, σ_e^2 , for the L_p model was generally lower than the OLS model. The L_1 model did not generally show any reduction in σ_e^2 when compared to OLS. The values for α and β were generally quite different for the three models.

Three sets of efficient frontiers were generated for each of the four models, each with different sets of upper and lower bounds. The bounds used (0,1), i.e. the weight assigned to any share must be between 0% and 100%, (0, 0.25) and (-1, 1).

All three index models outperformed the sample mean and covariance model over the entire efficient frontier and appears to justify the use of the index models in practice. The index model models provided superior efficient frontiers and appeared to justify the use of the index models in practice. There was very little difference between the L_1 and L_p models for the (0,1) formulations

The tightening or relaxation of bounds do not appear to influence the results in a material way.

Weighted regression using Mallows type weights on the explanatory variables was performed using OLS, L_1 and L_p estimation. The weighted L_1 and L_p estimates are thus a two tier approach to robust regression estimation. A summary of the regression output is given below.

The weighted estimates generated efficient frontiers that outperformed the non-weighted equivalents. The out performance of the weighted L_1 and L_p estimates over the non-weighted equivalents did not appear to be very large. Weighted OLS did however significantly outperform non-weighted OLS and reinforces the earlier suggestion that it provided a superior fit to the model. The formulation with (0, 0.25) bounds produced very similar results.

The efficient frontiers for the Sharpe Multiple Index model were generated from the regression output in the previous section. As was done for the single index model, three sets of efficient frontiers were generated subject to different constraints. The constraints used were again (0, 1), (0, 0.25) and (-1, 1). All of the Multiple Index models outperformed the sample mean and covariance model and again provided justification for using the index models instead of the simple sample estimates. The robust procedures produced superior efficient frontiers. There appears to be little difference between the L_1 and L_p models. The tightening or relaxation of restrictions on the portfolio formulation did not materially influence the results. We can thus conclude that the robust estimation techniques considered did aid in portfolio constructions.

1. INTRODUCTION

1.1. Background to the problem.

Modern portfolio theory was developed by Harry Markowitz more than forty years ago and is now considered to be an indispensable tool in portfolio construction. Sharpe introduced the index models as a simplification of the original Markowitz formulation as this required fewer parameters to be estimated. One of the premises underlying the Sharpe Index model was that the returns of shares and market proxies followed a normal distribution and under this assumption, model parameters could best be estimated using Ordinary Least Squares (OLS) regression. More recent empirical evidence has however cast doubt over the assumption of normality and has suggested that market returns tend to be non-normal. If the assumption of normality is no longer upheld then OLS or Maximum Likelihood Estimates may no longer produce the best estimates of parameters and hence may compromise optimal portfolio constructions.

In addition to the parameter estimation problems, at the time of the initial formulation of the portfolio models, managers were not allowed to participate in short sales (selling a share one does not own). This practice is now quite common in most developed markets and any portfolio formulation model needs to be generalised to allow for this.

1.2. Statement of the problem and objectives of this thesis.

Empirical studies have shown that share returns and the returns of market proxies do not follow a normal distribution but rather seem to follow a skew distribution and has long tails. It is the aim of this thesis to explore robust regression procedures which should be an improvement of OLS when the data does not come from a normal distribution. The robust regression procedures will then be used to estimate the parameters used in the Sharpe Index models and examine whether or not they aid in the portfolio construction process. The robust procedures will be applied to the classical portfolio formulations as well as the generalised models. The robust procedures will also be tested with further innovations to the Sharpe Index Models introduced by Troskie.

1.3. Outline of this thesis.

This report is divided into three parts. The first part deals with robust regression estimation, various robust techniques are discussed after which a simulation study is conducted to test the appropriateness of these techniques. The second part deals with portfolio theory and the Markowitz and Sharpe formulations are discussed. These

models are then generalised and the innovations by Troskie introduced. In the final part robust regression estimation is applied to the Sharpe Index Model where its appropriateness will be discussed thoroughly.

To perform the calculations needed for this thesis computer programs were written in Fortran. The programs can be used for further research in modern portfolio theory, for example applications to Markov Chain Monte Carlo (Bayesian) estimates and posterior distributions of the mean and covariance matrix including the single and multiple index sharp regression models. This includes research in the area of Bayesian Networks in decision theory of modern portfolio analysis. The source code for the programs is in Appendix A.

University of Cape Town

Part I: Robust Regression Estimators

Consider the model

$$Y = X\beta + e \quad (1)$$

where

$$E(e) = 0,$$

$$\text{var}(e) = \sigma^2 I$$

and

$$X_{n \times q}$$

is a data matrix of fixed explanatory variables.

The distribution of e , the error term, is not known and it could in fact have skewness and kurtosis properties that drastically departs from that of the normal distribution. The distribution may have long tails with very large kurtosis. This is typically the case when dealing with stock price returns, where very large values for the kurtosis are often observed.

To estimate β under the assumptions of normality and where $e \sim N(0, \sigma^2 I)$ the maximum likelihood estimate for β is

$$\hat{\beta} = (X'X)^{-1}X'Y. \quad (2)$$

This is also the ordinary least squares estimate (OLS) of β with no explicit distributional assumption of e . When outliers and influential observations are present in the data matrix X , and this typically happens when the underlying distributions have long tails, then the above estimates can be very far removed from the "true β 's". This topic has had considerable discussion in statistical literature and a number of estimators have been proposed for β that are more robust to deviations from normality than the OLS estimator. Of these, perhaps the best known are the M -estimators which are direct extensions of the M -estimators for location. Another well-known class of estimators for the location case is the L -estimators and in particular the trimmed means. This class, however, does not generalize as easily to the regression case owing to the ordering of the observations that is needed. In Koenker and Bassett (1978), this problem was overcome by first defining regression quantiles as the solution to a minimization problem and using these in a natural way to define L -estimators and in particular regression trimmed means.

2. THE ADAPTIVE L_p ESTIMATOR

2.1. Definition of the model.

The so called adaptive L_p estimator was defined by Sposito, Kennedy and Gentle (1977) for the case $q = 2$ and extended by Barr(1981) for the general case. If the model (1) is redefined as

$$Y_i = x_{(i)}\beta + e_i, \quad i = 1, \dots, n \quad (3)$$

with $x_{(i)}$ the i^{th} row of X , then the adaptive estimator minimizes

$$\sum_{i=1}^n |Y_i - x_{(i)}\beta|^p \quad (4)$$

and p is estimated in a adaptive way using the kurtosis of the estimated residuals. The case $p = 2$ is the OLS case. For values of $p = 1$ and $p = \infty$ exact linear programming solutions for β are available. In the research area of robust regression, values of p not equal to those above, have been shown to be important; in particular $1 < p < 3$.

2.2. Estimating p .

To find the adaptive p the same algorithm as proposed by Gonin and Money (1985b, 89) for non-linear estimation can be used. The model starts by setting $p = 2$ and fitting a model using least squares, then compute the sample kurtosis $\hat{\gamma}_2$ and estimate p by

$$\hat{p} = \frac{9}{\hat{\gamma}_2} + 1. \quad (5)$$

This estimated value of p is used in an iterated procedure to find new estimates for β and new estimates for the residuals. The new sample kurtosis is then computed from the residuals, and from (5) a new estimate for p . The process is repeated until no further change in the values of p is detected. Theoretical convergence can however not be proved.

To compute the estimated kurtosis $\hat{\gamma}_2$, The unbiased estimates of the second and fourth moments were used. These are

$$\hat{\mu}_2 = \sum_{i=1}^n (\hat{e}_i - \bar{e})^2 / (n - 1)$$

and

$$\hat{\mu}_4 = \frac{(n^2 - 2n + 3)}{(n-1)(n-2)(n-3)} \sum_{i=1}^n (\hat{e}_i - \bar{e})^4 - \frac{3(n-1)(2n-3)}{n(n-2)(n-3)} \hat{\mu}_2^2.$$

2.3. Distributional properties of the L_p estimator.

Since the above procedure minimizes $\sum_{i=1}^n |Y_i - x_{(i)}\beta|^p$ it is an L_p - norm estimator and has the following statistical properties which were developed by Nynquist (1980, 1983) (see Gonin and Money (1989)).

Theorem 2.1. Consider the model $Y = X\beta + e$ and let $\hat{\beta}$ be the estimator chosen so that

$$\sum_{i=1}^n |Y_i - x_{(i)}\beta|^p$$

is a minimum where $1 \leq p \leq \infty$.

Assume that:

A_1 : The errors e_i are independent, identically distributed with common distribution F .

A_2 : The L_1 and L_∞ - norm estimators are unique (in general the L_p - norm estimators are unique for $1 < p < \infty$).

A_3 : The matrix $Q = \lim_{n \rightarrow \infty} (X'X)/n$ is positive definite with rank q .

A_{4a} : F is continuous with $F'(0) > 0$ when $p = 1$.

A_{4b} : When $1 < p < \infty$ the following expectations exist:

$$E\{|e_i|^{p-1}\}, E\{|e_i|^{2p-2}\} \\ \text{and } E\{|e_i|^{p-1} \text{sign}(e_i)\} = 0.$$

Under the above four assumptions

$$\sqrt{n}(\hat{\beta} - \beta) \rightarrow N(0, w_p^2 Q^{-1}) \text{ as } n \rightarrow \infty \quad (6)$$

where

$$w_p^2 = 1/|2F'(0)|^2 \text{ if } p = 1 \\ = \frac{E\{|e_i|^{2p-2}\}}{|(p-1)E\{|e_i|^{p-2}\}|^2} \text{ if } 1 < p < \infty. \quad (7)$$

Koenker and Bassett (1978) considered the case where $p = 1$ and they showed that under the same conditions as in Theorem 1 above that

$$\sqrt{n}(\hat{\beta} - \beta) \rightarrow N(0, \lambda^2 Q^{-1}) \text{ as } n \rightarrow \infty \quad (8)$$

where λ^2/n is the variance of the sample median of residuals.

Several estimators for λ has been proposed (see Gonin and Money (1989) for details). The estimate proposed by McKean and Schrader (1987) is based on the α - percent non-parametric confidence interval for λ . The estimate for λ is given by

$$\hat{\lambda}_{1-\alpha} = \frac{\sqrt{\tilde{n}}(\hat{e}_{[n-l+1]} - \hat{e}_l)}{z_{\alpha/2}} \quad (9)$$

where $\tilde{n} = n - q$ and asymptotically

$$l = \frac{\tilde{n} - 1}{2} - z_{\alpha/2} \left[\frac{\tilde{n}}{4} \right]^{1/2}$$

and l is usually rounded to the nearest integer. Using this estimate we can set up confidence intervals for the individual β_i .

To set confidence intervals on the β_i for the case $1 < p < \infty$ we need an estimator for the quantity w_p^2 . By performing a simulation study Gonin and Money (1989) proposed that the sample moments of the residuals be used to estimate w_p^2 . Their estimate of w_p^2 was calculated as

$$\tilde{w}_p^2 = \frac{m_{2p-2}}{|(p-1)m_{p-2}|^2} \quad (10)$$

with

$$m_r = \frac{1}{n} \sum_{i=1}^n |\hat{e}_i|^r \quad (11)$$

where \hat{e}_i is the i^{th} estimated residual from the L_p - norm fit. Numerous problems arise in the estimation of w_p^2 using formulae (10) above. When an error term near zero is encountered, and when $p \approx 1$, division by zero is possible. The estimator of w_p^2 given by (10) is thus not a very good estimator if p is close to 1. One small residual, taken to a negative power, can inflate the divisor of the estimate of w_p^2 , to such an degree that w_p^2 tends to zero.

The problem of estimating w_p^2 using the above formulae was further investigated by Thiart (1994) by firstly perturbing m_r given by (11) as

$$m_r = \frac{1}{n} \sum_{i=1}^n (\hat{e}_i^2 + c^2)^{r/2} \quad (12)$$

with c a small fixed quantity, and secondly by a binomial expansion of

$$(\hat{e}_i^2 + c^2)^{r/2} \quad (13)$$

to yield an approximate estimate of w_p^2 as

$$\tilde{w}_p^2 = \frac{c^2 n}{(p-1)} \cdot \frac{[n + (p-1)c^{-2} \sum \hat{e}_i^2]}{[n + \{(p-2)/2\}c^{-2} \sum \hat{e}_i^2]} \quad (14)$$

Thiart (1994) experimented with several values for c in a simulation study and settled for a value of $c = 0.0145$. Thiart suggested that her perturbed estimator is only marginally successful for $1.2 < p < 2$ but for $1 < p < 1.2$ the perturbed estimator is not suitable. It is recommended that in this case the estimator of McKean and Schrader (1987) be used.

2.4. Masking and swamping.

A difficulty with the Adaptive L_p Estimator is that the procedure starts with $p = 2$, that is, the least squares residuals. It will work well, even for some long tailed distributions, but it cannot adapt easily when outliers and influential observations are present in the Y vector, and the X data matrix. Outliers in the Y - space may not be that serious but outliers in the X - space, called influential observations, can cause havoc with the estimates of β and hence also with the estimated residuals. The problem of *masking* and *swamping* has been identified as one that could lead to very misleading results when dealing with OLS estimates and their residuals (see Hawkins, Bradu and Kass (1984)).

If this is suspected there is a danger that the process may start in the wrong place. Even if it is not suspected, simply because it is not easy to identify, it is better to be cautious, and start the process with acceptable estimates for β and hence also for the residuals. A procedure is thus needed that will give reasonable initial estimates for β that should not seriously be effected by outliers or influential observations.

Mallows type bounded influence regression trimmed means developed by de Jongh, de Wet and Welsh (1988) are extensions of the regression trimmed means of Koenker and Bassett (1978) and Welsh (1987) (can) provide a solution to this. Consider the regression model (1) in the form

$$Y_i = x_{(i)}\beta + e \quad (15)$$

where $x_{(i)} = (1, x_{i2}, \dots, x_{iq})$ is the i^{th} row of the regression matrix X (consisting of $(q - 1)$ explanatory variables). Assume that the errors are independently and

identically distributed according to some unknown distribution F and is symmetric about zero.

The bounded influence regression trimmed mean estimators of β use as key concept the influence function of Hampel (1968,1974). If $0 < \alpha < 0.5$ then the influence function of the $100\alpha\%$ - regression trimmed mean estimators of Koenker and Bassett (1978) and Welsh (1987) has the following form:

$$\Lambda(Y, x) = Q^{-1} x'_{(i)} g_{\alpha}(Y - x_{(i)}\beta) \quad (16)$$

where Λ is the influence function, $Q = \lim_{n \rightarrow \infty} (X'X)/n$ and $g_{\alpha}(\cdot)$ is the influence function of a $100\alpha\%$ trimmed mean in the location case; that is

$$\begin{aligned} g_{\alpha}(u) &= \frac{F^{-1}(\alpha)}{1-2\alpha}, \quad u < F^{-1}(\alpha) \\ &= \frac{u}{1-2\alpha}, \quad F^{-1}(\alpha) \leq u \leq F^{-1}(1-\alpha) \\ &= \frac{F^{-1}(1-\alpha)}{1-2\alpha}, \quad u > F^{-1}(1-\alpha). \end{aligned}$$

Note that $g_{\alpha}(\cdot)$ limits the effect of the residuals and $\Lambda(Y, x)$ is thus bounded in the dependent variable space. Therefore one can conjecture that in small samples these ordinary robust estimators will be able to handle outliers in the Y -space but not in the X -space (see de Jongh, de Wet and Welsh (1987)).

Outliers in the X -space are influential points and they lie on the border of the convex hull. They can dramatically influence the regression estimates (see Chatterjee and Hadi (1986), Denby and Larson (1977)). These points usually have large leverage h_{ii} or Cook's distance D_i or both where

$$D_i = \frac{t_i^2}{q} \frac{h_{ii}}{1-h_{ii}} \quad (17)$$

and

$$t_i = \frac{\hat{e}_i}{\sqrt{s_{(i)}^2(1-h_{ii})}} \quad (18)$$

is the studentised residual having a t -distribution with $n - q - 1$ degrees of freedom. The quantity h_{ii} , called the leverage, is the i^{th} diagonal element of the HAT matrix $H = X(X'X)^{-1}X'$ and $s_{(i)}^2 = \{\hat{e}'\hat{e} - \hat{e}_i^2/(1-h_{ii})\}/(n-q-1)$ and is the estimate of σ^2 when deleting the i^{th} observation especially when the i^{th} observation

is spurious. If the i^{th} observation is spurious then it will always be a better estimate for σ^2 than the normal estimate for σ^2 , namely $s^2 = \sum_{i=1}^n \hat{e}_i^2 / (n - q)$.

It has been demonstrated (see Chatterjee and Hadi (1986)) that large values of h_{ii} or D_i , or both, have serious effects on the OLS estimates of β .

2.5. The WLS minimisation algorithm for the Adaptive L_p Estimator.

The algorithm of Sposito *et al* considers minimizing :

$$I = \sum_{i=1}^n W_i R_i^2 \quad \hat{\beta}_{WLS} = (X' W X)^{-1} X' W Y$$

(19)

W is diag weight matrix

where the R_i are the residuals and W_i are weighting factors. This method is known as Weighted Least Squares (WLS) (see Barr (1981)).

Slossmacher (1973) proposed the following algorithm:

$$I(k+1) = \sum_{i=1}^n (R(k+1)_i)^2 / |R(k)|^{2-p}, \quad (20)$$

where $1/|R(k)|^{2-p}$ is the weighting factor.

If $|R(k)_i - R(k+1)_i| \sim 0$ for $i = 1, \dots, n$, then

$$I(k+1) \sim \sum_{i=1}^n |R(k+1)_i|^p, \quad (21)$$

and the scheme has converged yielding $\hat{\beta}'$ s that minimizes $\sum_{i=1}^n |Y_i - x_i \beta|^p$ to some predetermined level of accuracy. Slossmacher has suggested that observations close to the fitted line (i.e. less than some predetermined error parameter) may be deleted but may be reintroduced if the residual increases at a later stage.

The algorithm is completed if

- (i) the scheme converges (the Sposito *et al* convergence criterion was used)
- (ii) the norm increases from one iteration to the next. Porter and Winstanley (1979) has suggested that control should only be returned once two successive increases in the norm has occurred, after observations have been deleted.
- (iii) the number of iterations exceeds 50.

3. BOUNDED INFLUENCE WEIGHTED LEAST SQUARES

The bounded influence adaptive *WLS* estimator *BWLS* uses as starting point the Mallows weights for the design matrix X after which the *OLS* residuals are used to start the iterative weighting procedure to estimate p . Thus the *BWLS* will minimize

$$L_p^w = \sum_{i=1}^n w_i |Y_i - x_{(i)}\beta|^p \quad (22)$$

where w_i are the Mallows weights.

The *BWLS* will initially down weight the effect of influential observations in the X - space for a hopefully better start of the iteration process but after the initial estimate of β and the residuals the weights w_i remain fixed and the remaining iteration depends only on the *new weighted* residuals of the next fit through the iteration process described above. Thus, this is therefore exactly the same as adaptive *WLS* applied to the transformed data $Y_i^* = \sqrt{w_i}Y_i$ and $X_{ij}^* = \sqrt{w_i}X_{ij}$ and the adaptive *WLS* procedure will minimize

$$L_p^w = \sum_{i=1}^n |Y_i^* - x_{(i)}^*\beta|^p. \quad (23)$$

The asymptotic distributions will therefore be

$$\sqrt{n}(\hat{\beta} - \beta) \rightarrow N(0, \lambda^2 Q^{*-1}) \text{ as } n \rightarrow \infty \text{ for } p = 1 \quad (24)$$

and

$$\sqrt{n}(\hat{\beta} - \beta) \rightarrow N(0, w_p^2 Q^{*-1}) \text{ as } n \rightarrow \infty \text{ for } 1 < p < \infty \quad (25)$$

with $Q^* = \lim_{n \rightarrow \infty} (X'WX)/n$.

The problem of finding a suitable estimate for w_p^2 however still remains.

3.1. Calculation of the weighting factors.

The bounded influence estimators developed by de Jongh, de Wet and Welsh (1989) use the following weight function introduced by Mallows (1973).

Denote the n observations of $(j - 1)^{th}$ independent variables by x_{1j}, \dots, x_{nj} for $j = 2, \dots, q$. Order the n observations $x_{(1)j}, \dots, x_{(n)j}$, and define r_{1j}, \dots, r_{nj} as the

vector of ranks of x_{1j}, \dots, x_{nj} . Let $L = [\pi n] + 1$ and $U = n + 1 - L$. The weights associated with the $(j - 1)$ th independent variable are now defined as

$$\begin{aligned} W_{ij} &= 1, L \leq r_{ij} \leq U, \\ &= \frac{x_{(L)j} - x_{(U)j}}{D_{ij}}, r_{ij} < L, \\ &= \frac{x_{(U)j} - x_{(L)j}}{D_{ij}}, r_{ij} > U, \end{aligned}$$

where $D_{ij} = 2x_{ij} - x_{(U)j} - x_{(L)j}$ ($i = 1, \dots, n$).

The Mallows weights are now defined as

$$W_i = \sum_{j=2}^n W_{ij}. \tag{26}$$

Note that the weights are chosen so that the outliers in the independent variable space - the influential observations - are given less weight according to their distance from the centre of the independent variable space. In this case the basic idea is to give weight to those Y_i having their corresponding $x_{(i)}$ in the centre and to decrease weight if the $x_{(i)}$ is in the tails of the independent variable space (borders of the convex hull). De Jongh *et al* choose $\pi = 0.15$ which they showed worked well in practice.

3.2. Other bounded influence estimators.

Koenker Bassett trimmed mean estimators.

De Jongh *et al* defined $100\alpha\%$ bounded influence Koenker - Bassett (KB) trimmed mean estimators as the solution to the minimization problem

$$\text{Min} \sum_{i=1}^n d_i^w(\alpha) (Y_i - x_{(i)}\beta)^2, \tag{27}$$

where

$$\begin{aligned} d_i^w(\alpha) &= \begin{cases} w_i, & x_{(i)}\hat{\beta}_{RQ}^w \leq Y_i \leq x_{(i)}\hat{\beta}_{RQ}^w(1 - \alpha), \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \tag{28}$$

Here $\hat{\beta}_{RQ}^w$ is the θ -th (bounded influence) sample regression quantile (RQ) ($0 < \theta < 1$) which is obtained as any solution to the minimization problem

$$\text{Min} \left\{ \theta \sum_1 w_i |Y_i - x_{(i)}\beta|^0 + (1 - \theta) \sum_2 w_i |Y_i - x_{(i)}\beta|^1 \right\}$$

This is the normal regression quantiles!

Check this?

Here is θ and $1 - \theta$.

where \sum_1 is taken over $\{i : Y_i - x_{(i)}\beta \geq 0\}$ and \sum_2 is taken over $\{i : Y_i - x_{(i)}\beta < 0\}$. This estimator is a weighted least squares estimator with weights $w_i = d_i^w(\alpha)$.

Let $D_\alpha^w = \text{diag}(d_i^w(\alpha))$, then

$$\hat{\beta}_{KB}^w(\alpha) = (X' D_\alpha^w X)^{-1} X' D_\alpha^w Y \quad (29)$$

and is a weighted least squares estimator.

Essentially one can think of $\hat{\beta}_{KB}^w$ as the weighted least squares estimator (with weights w_i) of the observations remaining after discarding observations lying outside the $\alpha - th$ and $(1 - \alpha) - th$ bounded influence regression quantiles. The purpose of the weights is to down weight observations having high leverage.

With $w_i = 1$, ($1 \leq i \leq n$), $\hat{\beta}_{KB}^w(\alpha) = \hat{\beta}_{KB}(\alpha)$ which is the original Koenker - Bassett estimator. De Jongh *et al* found that a trimming proportion of $\alpha = 15\%$ works well in practice.

Bounded influence Welsh Estimators.

The bounded influence Welsh estimator is defined by de Jongh *et al* as follows. Let $\hat{\beta}_{1N}$ be any initial estimator for β . Denote the residuals from this fit by

$$r_i = Y_i - x_{(i)}\hat{\beta}_{1N}, \quad i = 1, \dots, n.$$

Let $r_{(i)}$ be the i^{th} ordered value of the set of residuals $\{r_i\}$. For $0 < \theta < 1$ and $S = [n\theta]$ define

$$\begin{aligned} \xi_\theta &= r_{(s)}, \quad s = n\theta, \\ &= r_{(s+1)} \text{ otherwise.} \end{aligned}$$

with $[u]$ denoting the largest integer contained in u . The $100\alpha\%$ Welsh (WE) trimmed mean estimator is defined as

$$\hat{\beta}_{WE}^{(w)}(\alpha) = (X' W B_\alpha X)^{-1} X' W (B_\alpha Y + h_\alpha(r)) \quad (30)$$

where $W = \text{diag}(w_i)$ and $B_\alpha = \text{diag}(b_i(\alpha))$ with

$$\begin{aligned} b_i(\alpha) &= 1, \text{ if } \xi_\alpha \leq r_i \leq \xi_{1-\alpha} \\ &= 0 \text{ otherwise} \end{aligned}$$

Mallows? ↘

where $h_\alpha(r)$ is a vector with i^{th} component $h_\alpha(r_i)$, where

$$\begin{aligned} h_\alpha(u) &= \xi_\alpha - \alpha(\xi_\alpha + \xi_{1-\alpha}), u < \xi_\alpha \\ &= -\alpha(\xi_\alpha + \xi_{1-\alpha}), \xi_\alpha \leq u \leq \xi_{1-\alpha} \\ &= \xi_{1-\alpha} - \alpha(\xi_\alpha + \xi_{1-\alpha}), u > \xi_{1-\alpha}. \end{aligned}$$

Again with $w_i = 1$ ($1 \leq i \leq n$) $\hat{\beta}_{WE}^{(w)}(\alpha) = \hat{\beta}_{WE}(\alpha)$ which is the original Welsh estimator. The trimming proportion of $\alpha = 10\%$ was proposed by De Jongh *et al.*

De Jongh *et al.* showed that

$$n^{1/2}(\hat{\beta}_{KB}^{(w)}(\alpha) - \beta) \rightarrow^D N(0, \sigma_\alpha^2 Q_w^{-1} B_w Q_w^{-1}). \quad (31)$$

and that

$$n^{1/2}(\hat{\beta}_{WE}^{(w)}(\alpha) - \beta) \rightarrow^D N(0, \sigma_\alpha^2 Q_w^{-1} B_w Q_w^{-1}). \quad (32)$$

where

$$\lim_{n \rightarrow \infty} (X'WX)/n = Q_w, \quad \lim_{n \rightarrow \infty} (X'W^2X)/n = B_w \quad (33)$$

and

$$W = \text{diag}(w_i) \quad \text{and} \quad W^2 = WW.$$

Confidence intervals for the β_i can therefore be set up in the usual way but we need an estimator for σ_α^2 . Several estimators have been proposed for σ_α^2 . De Jongh (1985) has shown that the following two estimators have performed well.

Let

$$r_i(\alpha) = Y_i - x_{(i)} \hat{\beta}_{KB}, \quad i = 1, \dots, n$$

for the i^{th} residual and $r_{(1)} < r_{(2)} < \dots < r_{(n)}$ the ordered values. The first estimator for σ_α^2 is given by

$$s_1^2(\alpha) = (1 - 2\alpha)^{-2} \{1/(n - q)\} \cdot \left[\sum_{i=[n\alpha+1]}^{n-[n\alpha]} r_{(i)}^2 + \alpha(r_{([n\alpha]+1)}^2 + r_{(n-[n\alpha])}^2) \right] \quad (34)$$

where $[x]$ denotes the largest integer in x . Consistency follows using results of Bickel (1973).

The second estimator was suggested by Ruppert and Carroll (1980) and modified by Koenker (1983) to make it invariant with respect to a shift in the location. This is given by

$$s_2^2(\alpha) = (1 - 2\alpha)^{-2} \left[\{1/(n - q)\} \cdot \sum_{i=1}^n b_i r_i^2 + \alpha(c_1^2(\alpha) + c_2^2(\alpha)) \right]$$

where

$$c_1(\alpha) = \bar{x}' (\hat{\beta}_{RQ}(\alpha) - \hat{\beta}_{KB}(\alpha))$$

and

$$c_2(\alpha) = \bar{x}' (\hat{\beta}_{RQ}(1 - \alpha) - \hat{\beta}_{KB}(\alpha))$$

where \bar{x}' denotes a row vector consisting of the means of the columns of X . Consistency of $s_2^2(\alpha)$ is shown in Ruppert and Carroll (1980).

Chatterjee - Mächler weighted least squares.

Chatterjee and Mächler (1997) proposed on similar lines a bounded iterative weighted least squares procedure which according to them performed very well, when used with most data sets, except those that may be severely effected by masking and swamping (i.e. the set of Hawkins, Bradu and Kass (1989)). Their algorithm is a weighted least squares algorithm (*WLS*) which gives low weights to points with: (i) high leverage and (ii) large residuals. The weights are determined by the residuals and is therefore also an iterative one.

Their algorithm which will be denoted by *CWLS* is as follows:

Step 0 : Given weights

$$w_i^0 = 1 / \max(h_{ii}, \bar{h}), \quad \text{where } \bar{h} = q/n \quad (35)$$

and h_{ii} is the leverage of the i^{th} point. Compute the weighted least squares regression

$$\hat{\beta}^0 = \arg \min_{\beta} \sum_{i=1}^n w_i^0 (Y_i - \beta_1 X_{1i} - \dots - \beta_q X_{qi})^2 \quad (36)$$

Step j : ($j = 1, 2, \dots$)

Compute new weights from the residuals of the fit,

$$\begin{aligned} r^{j-1} &= Y - \hat{Y} = Y - X \hat{\beta}^{j-1} \\ w_i^j &= (1 - h_{ii})^2 / \max(|r_i^{j-1}|, \text{med}_i(|r_i^{j-1}|)) \quad i = 1, \dots, n \end{aligned}$$

where $\text{med}_i(x_i) = \text{median}(x_1, \dots, x_n)$.

Compute the weighted least squares regression

$$\hat{\beta}^j = \arg \min_{\beta} \sum_{i=1}^n w_i^j (Y_i - \beta_1 X_{1i} - \dots - \beta_q X_{qi})^2. \quad (37)$$

This is carried on until convergence.

There are similarities and differences between the *WLS* and *CWLS* procedures. These similarities and differences are quite important and will be highlighted in the computations in the simulation study in the next section.

University of Cape Town

4. A SIMULATION STUDY

To compare the least squares, robust and bounded (weighted) robust and least squares estimators a simulation study was conducted on data used by de Jongh. This data was used by de Jongh (1984) to compare several estimators of β under normal and heavy tailed distributions for the error term e . The data for the design matrix X is given in Appendix A.

The regression model used is of the form

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i3} + e_i$$

where $\beta_0 = 0$, and $\beta_i = 1.0$, $i = 1, 2, 3$.

Error terms were generated using the distributions discussed below. These were then combined with the above information to produce simulated data for the dependent variable Y .

The contaminated normal and slash distributions are often used in studying the behaviour of statistical procedures under heavy tailed distributions. The contaminated distributions are of the form

$$F(x) = (1 - \epsilon)\Phi(x) + \epsilon\Phi(x/\eta) \quad (38)$$

where $0 < \epsilon < 1$ and $\Phi(\cdot)$ is the standard normal distribution function. Typically $\eta > 1$ and $\Phi(x/\eta)$ is the distribution of the contaminated data, while ϵ is the probability of obtaining a contaminated observation. These distributions will be denoted $CN(\epsilon, \eta^2)$. The distribution function of the slash distribution is

$$F(x) = \Phi(x) + (\Phi(x) - \Phi(0))/x. \quad (39)$$

This distribution has the very long tails of the Cauchy distribution but with a normal like centre. The Cauchy distribution is relatively more peaked than the normal and slash distributions. Note that the probability density function of the slash distribution is given by

$$f(x) = (\Phi(0) - \Phi(x))/x^2.$$

The decision to concentrate on these types of distributions was primarily taken because random variables from these distributions may be generated as ratios of normal over independent variables.

Random numbers from $CN(\epsilon, \eta^2)$ may be generated as the ratio Z/V , where Z is a standard normal variate and

$$V = \begin{cases} 1 & \text{with probability } 1 - \epsilon \\ 1/\eta & \text{with probability } \epsilon \end{cases}$$

independent of Z . Random numbers from the slash distribution may be generated similarly with the only exception being that V is an independent uniform $(0, 1)$ variable. The five distributions that are used are given below and are denoted by F_1, F_2, \dots, F_5 :

$$\begin{aligned} F_1 &: N(0, 1) \\ F_2 &: CN(0.1, 9) \\ F_3 &: CN(0.2, 9) \\ F_4 &: CN(0.1, 100) \\ F_5 &: \text{Slash} \end{aligned}$$

Using the five distributions above 100 observations from each distribution were generated.

The different estimators were computed for each distribution. The following statistic

$$\sum_{i=1}^{100} (\hat{\beta}_i - \beta_i)^2$$

was computed for each estimator and averaged over the 100 observations.

The estimators used were

- ↓ LS : Least Squares.
- ↓ W : 10% Welsh (trimmed mean using LS residuals. to start iteration) $w_i = 1$
- ↓ WL_1 : 10% Welsh trimmed mean using L_1 errors. to start iteration $w_i = 1$
- ↓ KB : 15% Koenker-Bassett trimmed mean. $w_i = 1$
- ↓ BKB : 15% bounded influence Koenker-Bassett trimmed mean. $w_i = \text{Mallows}$
- ↓ BW : 10% (Bounded) Welsh trimmed mean with least squares error, to start $w_i = \text{Mallows}$ (40)

- * TB : 50% Tukey Biweight.
- √ WLS : Adapted WLS with p estimated from the kurtosis. see p 15
- √ $BWLS$: Bounded Adapted WLS . p 16
- $CWLS$: Chatterjee and Mächler WLS . adaptive L_p from p_{10}
- $CWLS_1$: Add constant to small errors. p_{10} est is same as p_{15} est.
- $CWLS_2$: Bounded with initial weights.

The WL_1 estimator is the 10% Welsh trimmed mean estimator using as initial estimates of the regression coefficients and residuals the results from a L_1 fit.

TB is the 50% Tukey Biweight: It does not correspond to any maximum likelihood or least squares estimator, but is an example of a function chosen to minimize a weighted sum of the residuals, where large, outlying residuals have small weights.

Design and Consequences.

The aim of the design was to determine the combined effect of influential observations in the explanatory data set - the X -space, also called high leverage points and distributions for which the error term e in the regression model has normal distributions and also contaminated normal distributions with long tails (large kurtosis). The Slash distribution has the property that it has very long tails (large kurtosis).

Two \mathbf{X} -design matrices were considered. In Data Set One observations (14) and (39) were constructed to be high leverage (influential points). High leverage points are measured by the diagonal elements h_{ii} of the "HAT" matrix where

$$\mathbf{H} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$$

with \mathbf{X} the $(n \times k)$ design matrix of explanatory variables. In the simulation data $n = 41$, $k = 4$ where the first term is a constant term. This data set - called - 41mat1.dat - in the program is given in Appendix A. self 4.1, 12.1, ..., 15 ?

High leverage points are points with values larger than $2k/n$ (see Belsley *et al*). For the simulated data sets $2k/n = 0.1951$. For data set one the 14th observation has a leverage value of $h_{14,14}^{(1)} = 0.2366$ while the 39th observation has a leverage point of $h_{39,39}^{(1)} = 0.2560$. Both these points are therefore influential. In Data Set Two - called - 41mat2.dat - in the program these two points (14 and 39) were made even more extreme by adding 100 to observation 14 of explanatory variable x_2 and 100 to observation 39 of explanatory variable x_3 . The leverage values are $h_{14,14}^{(2)} = 0.9619$ year!

and $h_{39,39}^{(2)} = 0.9788$, respectively. Thus they can be considered extreme influential points.

The consequences can clearly be demonstrated as follows. We use the first Y -observation simulated from each of the five distributions, given above, using data set one. These five simulated Y -vectors are also given in Appendix A.

A summary of the results follow. A detailed statistical output, as is given by EVIEWS4, is listed in Appendix B.

Summary of Least Squares Regression Results on Simulation Data.

Table 1.

Data Set One		Data Set Two					
Dist.	β_0	β_1	β_2	β_3	R^2	F -stat	AIC
1 Y_1	.1195	1.045(25.8)	1.021(25.9)	0.931(19.1)	.9778	542.9	2.6453
2 Y_2	-.1772	1.066(25.7)	1.093(27.2)	1.030(20.6)	.9793	583.9	2.6860
3 Y_3	-.0713	.9520(17.7)	.9669(18.5)	.9998(15.4)	.9589	287.8	3.2085
4 Y_4	.5031	.9277(4.45)	1.074(5.31)	.8800(3.51)	.6066	19.02	5.9164
5 Y_5	1.234	.7822(2.87)	1.075(4.05)	.8148(2.48)	.4385	9.63	6.4546
1 Y_1	-.2928	1.162(6.13)	.0286(0.75)	.1284(3.27)	.5426	14.6	5.6702
2 Y_2	-.5866	1.203(5.82)	.0191(0.46)	.1366(3.19)	.5135	13.0	5.8437
3 Y_3	-.5100	1.108(6.04)	.0259(0.70)	.1458(3.85)	.5493	15.0	5.6033
4 Y_4	.1640	1.054(3.66)	-.0042(-.07)	.1253(2.10)	.2906	5.05	6.5060
5 Y_5	.8901	.8594(2.52)	.0175(.25)	..1056(1.50)	.1700	2.53	6.8455

The quantities in brackets are the t -statistics.

The results as shown in Table 1. are quite remarkable. For distributions 1, 2 and 3 the least squares results are quite robust. This is also born out by the full simulation study given below. The harmful effect of the presence of both influential observations (high leverage points) and error distributions with large tails (large kurtosis), introducing outliers, can clearly be seen. Of particular interest is the behaviour of the Akaike Information Criterion (AIC) where the change from $AIC = 2.6483$ for Distribution 1 of the first data set and $AIC = 6.8455$ for Distribution 5 of the second data set can be considered to be very large indeed.

The above results are those obtained from least squares without using any robust procedures- downweighting large residuals, downweighting large leverage points or both. This will be done in the next section. Nevertheless it is interesting to note the effect of downweighting the high leverage by using weighted regression. We first

use the Mallows weights and secondly straightforward deletion of the high leverage points (observations (14) and (19)). We use the data set two and the Y -vector (Y_5) simulation from Distribution -5- the Slash distribution. The results are reported in Table 2. The EVIEWS4 output is again listed in Appendix B. For a graphical illustration of this see the scatter plot of unweighted Anglo returns vs the JSE returns and Mallows downweighted returns of these two variables in Appendix B.

Table 2.

D-5*	Data			Set	Two		
	Y_5	β_0	β_1	β_2	β_3	R^2	F -stat.
M	1.271	1.073(2.03)	1.861(4.26)	1.697(2.73)	.4432	9.82	6.4463
D	1.336	.7789(2.51)	1.116(3.85)	.8059(2.10)	.4377	9.08	6.5096

* D-5 - Distribution 5, M - Mallows weights, D - Delete observations (14) and (39).

The Mallows weights perform marginally better than straightforward deletion of observations (14) and (39). Looking at the Mallows weights for data set two given in Appendix B, it is clear that the weights for observation (14) - 0.008, and observation (39) - 0.006, are quite small, but it is also clear that other observations are also downweighted by the Mallows weights. [Thus it seems that downweighting is preferred to deletion.] This will be further discussed after the full simulation.

The simulation results are reported below.

The Simulation Results. Data Set One.

Distribution 1. $N(0,1)$

	β_0	β_1	β_2	β_3
LS	.02437	.00272	.00197	.00328
W	.02640	.00292	.00221	.00354
WL ₁	.02637	.00291	.00226	.00350
KB	.02713	.00351	.00248	.00457
BKB	.02980	.00400	.00274	.00549
BW	.02887	.00370	.00242	.00455
TB	.02676	.00295	.00229	.00373
WLS	.02460	.00273	.00204	.00342
BWLS	.01682	.00335	.00227	.00436
CWLS	.02403	.00306	.00211	.00358
CWLS ₁	.02403	.00306	.00211	.00358
CWLS ₂	.02403	.00306	.00211	.00358

Is strong statement!

The average skewness was 0.0233 and average kurtosis 2.8676.

Distribution 2. CN(0.1,9)

	β_0	β_1	β_2	β_3
<i>LS</i>	.04097	.00278	.00314	.00677
<i>W</i>	.03527	.00237	.00261	.00447
<i>WL₁</i>	.03470	.00242	.00261	.00469
<i>KB</i>	.03343	.00272	.00286	.00447
<i>BKB</i>	.04202	.00389	.00412	.00631
<i>BW</i>	.04138	.00348	.00365	.00540
<i>TB</i>	.03590	.00241	.00298	.00465
<i>WLS</i>	.03953	.00247	.00311	.00484
<i>BWLS</i>	.02655	.00312	.00379	.00557
<i>CWLS</i>	.04330	.00305	.00384	.00741
<i>CWLS₁</i>	.04330	.00305	.00384	.00741
<i>CWLS₂</i>	.04330	.00305	.00384	.00741

The average skewness was 0.0105 and average kurtosis 5.7354.

Distribution 3. CN(.2,9)

	β_0	β_1	β_2	β_3
<i>LS</i>	.05897	.00628	.00546	.00850
<i>W</i>	.03164	.00494	.00396	.00597
<i>WL₁</i>	.03387	.00501	.00401	.00656
<i>KB</i>	.03689	.00536	.00456	.00636
<i>BKB</i>	.04690	.00701	.00523	.00914
<i>BW</i>	.04063	.00693	.00495	.00843
<i>TB</i>	.02785	.00453	.00335	.00592
<i>WLS</i>	.03253	.00495	.00389	.00651
<i>BWLS</i>	.02183	.00639	.00459	.00734
<i>CWLS</i>	.06303	.00744	.00621	.00973
<i>CWLS₁</i>	.06303	.00744	.00621	.00973
<i>CWLS₂</i>	.06303	.00744	.00621	.00973

The average skewness was -.0018 while the average kurtosis was 5.9263.

Distribution 4. CN(.1,100)

	β_0	β_1	β_2	β_3
<i>LS</i>	.21073	.02002	.01784	.03148
<i>W</i>	.03306	.00468	.00312	.00688
<i>WL₁</i>	.03287	.00315	.00277	.00548
<i>KB</i>	.03845	.00325	.00272	.00526
<i>BKB</i>	.05196	.00438	.00422	.00690
<i>BW</i>	.04034	.00409	.00373	.00681
<i>TB</i>	.03104	.00215	.00195	.00425
<i>WLS</i>	.04482	.00409	.00289	.00608
<i>BWLS</i>	.02580	.00403	.00314	.00719
<i>CWLS</i>	.23852	.02412	.02118	.03319
<i>CWLS₁</i>	.23852	.02412	.02118	.03319
<i>CWLS₂</i>	.23852	.02412	.02118	.03319

The average skewness was -.0048 while the average kurtosis was 15.2330.

Distribution 5. Slash

	β_0	β_1	β_2	β_3
<i>LS</i>	121.11499	13.11039	6.04391	4.01387
<i>W</i>	2.13230	20.71956	3.98874	6.64015
<i>WL₁</i>	.23654	.02205	.03696	.03749
<i>KB</i>	.24401	.01786	.02822	.03391
<i>BKB</i>	.30722	.02205	.03217	.04096
<i>BW</i>	.24869	.02651	.03478	.04564
<i>TB</i>	.14984	.01175	.02245	.02322
<i>WLS</i>	.16619	.01230	.02388	.02441
<i>BWLS</i>	.10186	.01458	.02466	.02539
<i>CWLS</i>	147.24425	25.03178	9.16650	7.48097
<i>CWLS₁</i>	147.24425	25.03178	9.16650	7.48097
<i>CWLS₂</i>	147.24425	25.03178	9.16650	7.48097

The average skewness was .0367 while the average kurtosis was 20.6926.

Data Set Two.**Distribution 1. N(0,1)**

	β_0	β_1	β_2	β_3
<i>LS</i>	.02460	.00315	.00009	.00011
<i>W</i>	.02717	.00332	.00009	.00011
<i>WL₁</i>	.02728	.00337	.00010	.00011
<i>KB</i>	.02758	.00359	.00009	.00011
<i>BKB</i>	.02876	.00470	.00235	.00531
<i>BW</i>	.02987	.00414	.00217	.00466
<i>TB</i>	.02746	.00394	.00010	.00011
<i>WLS</i>	.23241	.00325	.00023	.00019
<i>BWLS</i>	.01641	.00377	.00131	.00224
<i>CWLS</i>	.02456	.00325	.00023	.00020
<i>CWLS₁</i>	.02456	.00325	.00023	.00020
<i>CWLS₂</i>	.02456	.00325	.00023	.00020

The average skewness was 0.0233 and average kurtosis 2.8676.

Distribution 2. CN(0.1,9)

	β_0	β_1	β_2	β_3
<i>LS</i>	.04290	.00293	.00017	.00007
<i>W</i>	.03536	.00228	.00017	.00007
<i>WL₁</i>	.03515	.00224	.00017	.00007
<i>KB</i>	.03487	.00267	.00018	.00007
<i>BKB</i>	.04313	.00432	.00346	.00590
<i>BW</i>	.04321	.00391	.00335	.00547
<i>TB</i>	.03837	.00229	.00017	.00007
<i>WLS</i>	.39033	.00289	.00038	.00019
<i>BWLS</i>	.02625	.00340	.00195	.00288
<i>CWLS</i>	.04462	.00331	.00038	.00035
<i>CWLS₁</i>	.04462	.00331	.00038	.00035
<i>CWLS₂</i>	.04462	.00331	.00038	.00035

The average skewness was 0.0105 and average kurtosis 5.7354.

Distribution 3. CN(.2,9)

	β_0	β_1	β_2	β_3
<i>LS</i>	.06530	.00696	.00027	.00022
<i>W</i>	.03701	.00528	.00027	.00021
<i>WL₁</i>	.03724	.00528	.00027	.00020
<i>KB</i>	.03880	.00559	.00028	.00021
<i>BKB</i>	.05112	.00857	.00434	.00988
<i>BW</i>	.04477	.00795	.00480	.00910
<i>TB</i>	.03273	.00533	.00027	.00021
<i>WLS</i>	.34754	.00652	.00041	.00028
<i>BWLS</i>	.02351	.00759	.00233	.00339
<i>CWLS</i>	.06703	.00789	.00066	.00056
<i>CWLS₁</i>	.06703	.00789	.00066	.00056
<i>CWLS₂</i>	.06703	.00789	.00066	.00056

The average skewness was -.0018 while the average kurtosis was 5.9263.

Distribution 4. CN(.1,100)

	β_0	β_1	β_2	β_3
<i>LS</i>	.22542	.02325	.00102	.00070
<i>W</i>	.03281	.00457	.00096	.00258
<i>WL₁</i>	.03365	.00340	.00095	.00059
<i>KB</i>	.03455	.00402	.00097	.00062
<i>BKB</i>	.05220	.00566	.00457	.00062
<i>BW</i>	.04610	.00568	.00421	.00781
<i>TB</i>	.03139	.00254	.00094	.00059
<i>WLS</i>	.45532	.00424	.00044	.00072
<i>BWLS</i>	.02617	.00464	.00136	.00319
<i>CWLS</i>	.23300	.02715	.00227	.00178
<i>CWLS₁</i>	.23300	.02715	.00227	.00178
<i>CWLS₂</i>	.23300	.02715	.00227	.00178

The average skewness was -.0048 while the average kurtosis was 15.2330.

Distribution 5. Slash

	β_0	β_1	β_2	β_3
<i>LS</i>	139.30646	14.23385	.04590	.10661
<i>W</i>	1.92363	26.44555	2.51754	.24945
<i>WL₁</i>	.27175	.02736	.01471	.03771
<i>KB</i>	.23297	.02369	.01463	.03822
<i>BKB</i>	.31313	.02494	.03329	.03743
<i>BW</i>	.25440	.03325	.03183	.04593
<i>TB</i>	.13318	.01667	.01443	.03784
<i>WLS</i>	1.65899	.01807	.01333	.02264
<i>BWLS</i>	.09568	.01749	.01430	.01139
<i>CWLS</i>	140.96313	21.50420	.50584	.20280
<i>CWLS₁</i>	140.96313	21.50420	.50584	.20280
<i>CWLS₂</i>	140.96313	21.50420	.50584	.20280

From the simulation study it can be seen that in general the average skewness was small and close enough to zero to indicate a symmetric distribution. In comparing the actual estimates for the regression coefficients $(\beta_0, \beta_1, \beta_2, \beta_3)$ for different types of estimators care should be taken when comparing the results of β_0 for *WLS* and *BWLS* with the results of β_0 for the other estimators. This is because in the formulation of *WLS* by Barr (1980) the X-data matrix was centered and the regression coefficients computed from the centered matrix. In the centering only β_1, β_2 and β_3 are computed which will of course be the same as those computed for the uncentered data. In the computation though of $\hat{\beta}_0$ the formula

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}_1 - \hat{\beta}_2 \bar{x}_2 - \hat{\beta}_3 \bar{x}_3 \quad (41)$$

is used with the result that influential observations in the X -space could seriously affect the computations of the means \bar{x}_1, \bar{x}_2 and \bar{x}_3 and hence also the computation of $\hat{\beta}_0$.

For Distribution 1 - $N(0,1)$, the average kurtosis is 2.8676, not surprising for data generated from the normal distribution. The skewness and kurtosis will be the same for both data sets since the same seed was used for the simulation of the error distributions. For the estimates of β_1, β_2 and β_3 the *MSE's* for all estimation procedures, for both data sets, are small and the results are remarkably similar. Only *WLS* shows a slight departure for the constant term β_0 . for the reasons stated above.

It is clear that the *OLS* gives satisfactory results even though large influential observations are present (observations (14) and (39)) for the second data set. The *OLS* estimates can be used without hesitation and the normal diagnostics and statistics like the leverage, studentised residuals and Cook's distance can be used to refine the data if necessary.

For Distribution 2 - $CN(.1,9)$, the average kurtosis is 5.7354, showing that the contamination had the desired effect. Again the results are similar, for both data sets, except again for the constant term β_0 for WLS . The OLS still perform well.

For Distribution 3. - $CN(.2,9)$, the average kurtosis is 5.9263. One immediately sees the effect of the longer tails. The MSE 's for all the estimators are larger than for the other two distributions. Nevertheless, there is no obvious winner. The performance of OLS remains remarkable.

For Distribution 4. - $CN(.1,100)$, the average kurtosis is 15.2330, indicating clearly the effect of the contamination. The kurtosis is very large - five times as large as for the normal distribution. It is immediately clear from the MSE 's that the robust estimators are beginning to perform better than the OLS estimator for both data sets.. The performance of WLS (apart from the constant term) is very good indeed.

For Distribution 5, - Slash, the average kurtosis is 20.6926, nearly seven times that of a normal distribution. The OLS , W and the $CWLS$ perform extremely poorly for both data sets.. The robust estimators do very well. The best performer is $BWLS$, and WLS , apart from the constant term, also does very well. The performance of $BWLS$ and WLS for data set two is especially impressive.

Thus for non-normal data, where large kurtosis is present, the bounded influence and WLS estimators are superior. Because of the ease of computation the WLS estimator requires serious consideration.

The performance of $CWLS_1$ with small constant added to small residuals (see discussion below), is exactly the same as $CWLS$. For $CWLS_2$ where the new weights are bounded to be not larger than the Chatterjee initial weights (see also discussion below), the results are also identical. This seemed surprising at first, but these are the results at convergence and they must be the same. It is the number of iterations that differ.

For Distribution 5. - Slash the number of iterations for $CWLS$ before convergence was 87 , while for $CWLS_1$ the number of iterations before convergence was 67 while for $CWLS_2$ the number was 55.

Thus the last two procedures do have an effect and should be considered for further investigation.

In conclusion the OLS estimates can be used for normal data as well as for data with slightly longer tails. Under normal theory the estimates will be maximum likelihood and the powerful distributional results on hypothesis testing, confidence

intervals, tests for outliers, etc., are available. When the kurtosis is very large, then a bounded procedure that bounds both the leverage points and the residuals should be used. Such procedures are the bounded Koenker-Bassett (*BKB*) and the bounded Welsh (*BW*). These procedures are difficult to implement because of computational sophistication. The adapted *WLS* is easy to implement, since it uses an easy programmable algorithm. Its performance is as good, if not better, than that of *BKB*, and *BW*. The bounded *WLS* called *BWLS*, also performs very well. It uses the Mallows weights as bounds and is therefore more difficult to implement. The performance of *BWLS* is so good that an investigation to the implementation of easier weights seems justified.

University of Cape Town

5. A COMPARISON OF MODELS

The simulation study in the previous section *CWLS* did not perform as well as expected. To try and understand the reasons for this the following comparative analysis between *WLS* and *CWLS* was performed.

Step 0 : In *WLS* the high leverage points (large h_{ii}) are down weighted using the Mallows weights. In *CWLS* the high leverage points are down weighted using (36). In the simulation study conducted in the previous section the large values for the leverage h_{ii} are for observations (14) and (39) with values of .9619 and .9788, respectively. Since in the simulation study $q = 4$, $\sum_{i=1}^n h_{ii} = 4$. These two points therefore contribute 24.05% and 24.50% to the value of the total leverage. The normal cut-off point is considered to be $2q/n$ and since $n = 41$ this cut-off point is 0.1951. Thus both these points can be considered extremely influential. None of the other design points exceed this cut-off value.

The Mallows weights for these two points are 0.0080 and 0.0059. The total of the Mallows weights is 28.7365 while the average weight is 0.7009. Thus the contribution of these two observations are 0.0278% and 0.0205% of the total weight and 1.1414% and 0.8418% of the average weight.

The Chatterjee weights for these two points are 1.0396 and 1.0216, respectively. The total Chatterjee weight is 386.2240 while the average weight is 9.4201. Thus the contribution of these two observations are 0.2692% and 0.2645% of the total weight and 11.0360% and 10.8449% of the average weight. The Mallows weights, therefore, down weights these observations considerably more than the Chatterjee weights. Nevertheless, down weighting these observations is the correct strategy considering the sizes of the leverages.

Several other authors have proposed other types of bounded influence weights as initial weights. This will not be pursued further but is a topic for further research.

Step j : For both *WLS* and *CWLS* the weights are functions of both the residuals and the leverage. For *WLS* the Mallows weights remain the same but for *CWLS* the leverage weights are changed to new weights $v_i^2 = (1 - h_{ii})^2$ and remains the same for all further iterations. To see and compare the effect of the change for observations (14) and (39) the new weights become 0.0015 and .0004944. The sum of the weights is 36.0566 while the average weight is 0.8550. Thus the contribution of these two observations to the total weight is 0.0043% and 0.0014% while their contribution to the average weight is 0.1754% and 0.0578%. These weights are now considerably less than the Mallows weights. It can be argued that these weights

are so small that these two observations are effectively eliminated when using $v_i^2 = (1 - h_{ii})^2$ unless the new weights obtained from the residuals increase dramatically at the same time. The combined weights of the leverage and the residuals are $w_i^j = (1 - h_{ii})^2 / \max(|r_i^{j-1}|, \text{med}_i(|r_i^{j-1}|))$ so that v_i^2 is small if h_{ii} is large which means that the i^{th} observation is on the border of the convex hull and is unlikely that the residual will be small. More serious, however, is when the h_{ii} is small so that v_i^2 is close to one. The weight will then mainly depend on the residual or the median of the residuals of the $(j - 1)^{\text{th}}$ step. The median of the residuals, in general will be small but could also be zero. The weight will then depend entirely on the residual $|r_i^{j-1}|$ and if this is small - near zero - then the weight w_i^j will be very large. We consider this to be a serious drawback of the Chatterjee weights and of *CWLS*. To give an example, for observation (14) identified as influential the Mallows' weight is 0.0080, the initial Chatterjee weight is 1.0396 and the Chatterjee weight after the first iteration is .00823, Convergence takes place after 8 iterations and the weight at convergence is .00063. For observation (9) the Mallows weight is .6008, the initial Chatterjee weight is 10.5200 and after the first iteration is 5.1612. The weight at convergence is 1161.8914. In this case the error distribution was normal. The situation gets worse for long-tailed distributions.

Thus small errors - near zero - are a problem with *CWLS*. The *WLS* delete small errors when the errors become smaller than a certain error rate. It is therefore recommend the following modification of *CWLS*.

Step 0 : Use as starting point the initial Chatterjee weights or the weights $v_i^2 = (1 - h_{ii})^2$, since they are easy to compute the Mallows' weights work well, but they are not easy to compute. The Mallows' weights is some where in between the above two.

Step j : Use the Chatterjee weights but check for small residuals and medians. If the residuals are near zero make a modification by :

(i) Delete them or use the algorithm of Porter and Winstanley (1979) that is used in *WLS*.

(ii) Add a small quantity d to the residuals as suggested by Thiart (1994). The quantity $d = 0.0145$ is suggested.

(iii) Bound the weights, for example, to be never larger than the Chatterjee initial weights, if these are the starting points.

These recommendations were implemented in the above simulation study but, except for quicker convergence, the results were not good. Considerable care must therefore be taken with the choice of weights when using robust regression

Application to Portfolio Theory.

Robust regression estimation will be applied to portfolio theory in section Part III. The robust estimators that will be used are L_1 , L_p , OLS, *weighted* L_1 , *weighted* L_p and *weighted* OLS. These estimators were chosen for their ease of computation.

University of Cape Town

Part II: Modern Portfolio Theory

6. NOTATION AND ASSUMPTIONS

Let P_T be the price of a stock (or share) at some time T with P_0 the price at time $T = 0$, the start of a time period under consideration. Of interest to the portfolio manager is the price change $P_T - P_0$, or more particularly the relative price change, called the simple return,

$$\frac{P_T - P_0}{P_0}. \quad (42)$$

The simple return (42) is often expressed as a percentage

$$\frac{P_T - P_0}{P_0} 100\% \quad (43)$$

giving the percentage change in the stock price over the time period T . The price P_T is considered to be a random variable having some distribution. Suppose there are n time periods in the time interval $T - T_0$. Suppose further that these time periods are equally spaced in time (for example days, weeks, months, quarters, years etc.). Let us denote these prices by P_t , $t = 1, \dots, n$.

Definition 1.

$$R_t = \frac{P_t - P_{t-1}}{P_{t-1}}, \quad t = 1, \dots, n \quad (44)$$

is the return of the stock. (45)

Note that $P_T = P_n$. Clearly any P_t or R_t is also a random variable with some mean (expected value) and variance.

From the compound interest formula in Finance

$$(1 + R_T) = (1 + R_1)(1 + R_2) \dots (1 + R_n)$$

so that

$$\log(1 + R_T) = \sum_{t=1}^n \log(1 + R_t).$$

The R_t , (for $t = 1, \dots, n$), are random variables (rvs) and assume that they have the same mean and variance. Past experience has shown that this is probably not true over long periods of time. It is, however a reasonable assumption over short periods of time. We also make the assumption that the R_t are independently distributed. Our

assumption is thus that stock price returns, R_t , are independently and identically distributed (i.i.d.).

The Central Limit Theorem then says that

$$\log(1 + R_T) = \sum_{t=1}^n \log(1 + R_t),$$

which is the sum of n i.i.d. random variables, follows a Normal distribution with some mean and variance.

Now

$$\log(1 + R_T) = \log\left(1 + \frac{P_n - P_0}{P_0}\right) \quad (46)$$

$$\begin{aligned} \log(1 + R_T) &= \log\left(1 + \frac{P_T - P_0}{P_0}\right) \\ &= \log P_T - \log P_0 \\ &= \log(P_T/P_0) \end{aligned} \quad (47)$$

$$= \log P_T - \log P_0 \quad (48)$$

Definition 2.

$$R = \log P_T - \log P_0 \quad (49)$$

The random variable R is thus distributed (for large n) as a Normal distribution with mean $E(R) = \mu$ and variance $\text{var}(R) = \sigma^2$.

That is

$$R \simeq N(\mu, \sigma^2).$$

Thus

$$\log P_T = \log P_0 + R \quad \text{or} \quad P_T = P_0 e^R. \quad (50)$$

If $R \sim N(\mu, \sigma^2)$ then the moment generating function of R is

$$\begin{aligned} M_R(t) &= E(e^{tR}) \\ &= \exp\left(t\mu + \frac{1}{2}t^2\sigma^2\right). \end{aligned} \quad (51)$$

From

$$P_T = P_0 e^R \quad (52)$$

so that

$$E(P_T) = P_0 e^{\mu + \sigma^2/2}. \quad (53)$$

The random variable

$$R = \log P_T - \log P_0$$

is known as the log return and the assumption is made that log returns are Normally distributed. Alternatively, the statement is made that stock prices P_T are log-normally distributed. For the rest of this thesis returns will be taken to mean log returns of shares.

For the return $R = \log P_T - \log P_0$ to have a Normal distribution $N(\mu, \sigma^2)$ the time span $T - T_0$ must be large enough for the Central Limit Theorem to hold. Fama stated in Foundations of Finance:

"The frequency distributions, the comparisons of average relative frequencies with normal probabilities and the studentised ranges, all lead to the conclusion that distributions of monthly returns are closer to normal than distributions of daily returns. This is inconsistent with the hypothesis that return distributions are non-normal symmetric stable, which implies that distributions of daily and monthly returns should have about the same degree of leptokurtosis. Moreover, although the evidence also suggests that distributions of monthly returns are slightly leptokurtic relative to normal distributions, let us tentatively accept the normal model as a working approximation for monthly returns.

The assumptions that distributions of returns on stocks and portfolios are normally distributed is used to develop a model for portfolio decisions by investors. The usefulness of the portfolio model depends not on whether the normality assumption which underlies is an exact description of the world (we know it is not), but on whether the model yields useful insights into the essential ingredients of rational portfolio decision. If the model does well on this score, we can live with the small observed departures from normality in monthly returns, at least until better models come along".

We will thus also make the assumption of normal distribution of stock returns in this section but will relax this assumption in Part III.

7. CLASSICAL FORMULATIONS

7.1. The Markowitz formulation and solution.

Consider a portfolio consisting of p stocks. Let the vector of stock returns for the portfolio be

$$\mathbf{R} = \begin{pmatrix} R_1 \\ \vdots \\ R_p \end{pmatrix} \quad (54)$$

with the expected returns

$$E(\mathbf{R}) = \boldsymbol{\mu}. \quad (55)$$

Assume that the covariance between returns of different shares are non-zero thus the covariance matrix if the stock returns is

$$\boldsymbol{\Sigma} = E(\mathbf{R} - \boldsymbol{\mu})(\mathbf{R} - \boldsymbol{\mu})'$$

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_{11} & \cdots & \sigma_{1p} \\ \vdots & & \vdots \\ \sigma_{p1} & \cdots & \sigma_{pp} \end{pmatrix}$$

where $\sigma_{ii} = \sigma_i^2$ is the variance of the i 'th and σ_{ij} the covariance between the i 'th and j 'th stock.

Further assume that

$$\mathbf{R} \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \quad (56)$$

the multivariate normal.

A portfolio of stocks can be thought of as a cash investment, of say w_i in each stock. We can assume that w_i is the proportion of available cash that is invested in stock i (w_i is sometimes called the weight of stock i). If w_i is thought of as a proportion (or weight) then

$$\sum_{i=1}^p w_i = 1. \quad (57)$$

The fact that the weights of each of the stocks are constrained to sum to one implies that the portfolio manager cannot invest more money in stocks than he has available.

Let

$$\mathbf{W} = \begin{pmatrix} w_1 \\ \vdots \\ w_p \end{pmatrix} \quad (58)$$

and P be the return of the portfolio.

Thus

$$\begin{aligned} P &= \mathbf{W}'\mathbf{R} \\ &= \sum_{i=1}^p w_i R_i, \end{aligned} \quad (59)$$

subject to

$$\sum_{i=1}^p w_i = 1.$$

The expected return of the portfolio is

$$\begin{aligned} E(P) &= \mathbf{W}'E(\mathbf{R}) \\ &= \mathbf{W}'\boldsymbol{\mu} \end{aligned} \quad (60)$$

$$= \sum_{i=1}^p w_i \mu_i \quad (61)$$

$$= \mu_p \quad (62)$$

and the variance is

$$\begin{aligned} \text{var}(P) &= \mathbf{W}'\boldsymbol{\Sigma}\mathbf{W} \\ &= \sum_{i=1}^p w_i w_j \sigma_{ij} = \sigma_p^2. \end{aligned}$$

If in addition $R \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ then $P \sim N(\mu_p, \sigma_p^2)$.

By changing the weights, w_i , the value, as well as the expected value of the portfolio is changed.

The objective of portfolio managers is to maximize the return of the portfolio. This is however only half of the portfolio problem. For any given return the portfolio manager would also try to minimize the variance of the portfolio. In Finance the variance of a portfolio is often called its risk. The portfolio problem thus involves choosing the weights, w_i , such that the expected return $E(P) = \mu_p$ is maximized and the risk (or variance) σ_p^2 is a minimized.

Our portfolio problem is then

$$\max_{w_i} E(P) = \mathbf{W}'\boldsymbol{\mu} \quad (63)$$

$$= \sum_{i=1}^p w_i \mu_i$$

$$\min_{w_i} \sigma_p^2 = \mathbf{W}'\boldsymbol{\Sigma}\mathbf{W} \quad (64)$$

$$= \sum_{i=1}^p w_i w_j \sigma_{ij} \quad \text{subject to}$$

$$\sum_{i=1}^p w_i = 1. \quad (65)$$

This formulation is due to Harry Markowitz.

Markowitz also constrained the weights such that

$$0 \leq w_i \leq 1 \quad \text{for } i = 1 \dots p.$$

Portfolios that satisfied this condition are said to be legitimate.

The Markowitz portfolio problem is then formally,

$$\max_{w_i} E(P) = \mathbf{W}'\boldsymbol{\mu} \quad (66)$$

$$= \sum_{i=1}^p w_i \mu_i$$

$$\min_{w_i} \sigma_p^2 = \mathbf{W}'\boldsymbol{\Sigma}\mathbf{W} \quad (67)$$

$$= \sum_{i,j=1}^p w_i w_j \sigma_{ij} \text{ subject to}$$

$$\sum_{i=1}^p w_i = 1, \quad (68)$$

$$0 \leq w_i \leq 1, \quad i = 1, \dots, p. \quad (69)$$

This is a non-linear (quadratic) programming problem called a *QP – problem*. To solve the problem one needs either to fix the return μ_p and then minimize the variance σ_p^2 , or fix the variance and then maximize the return. They will both lead to the same solution. For the following calculations the return will be fixed and the used to solve for the minimum variance. Thus we need to solve the following constrained *QP* problem.

$$\begin{aligned} \text{Min } \sigma^2 &= \mathbf{W}'\Sigma\mathbf{W} \\ &= \sum_{i=1}^p w_i w_j \sigma_{ij} \end{aligned}$$

subject to

$$\begin{aligned} \mu_p &= \mathbf{W}'\boldsymbol{\mu}_i \\ &= \sum_{i=1}^p w_i \mu_i \end{aligned} \quad (70)$$

$$= E_k \quad (71)$$

and

$$\sum_{i=1}^p w_i = 1, \quad 0 \leq w_i \leq 1,$$

where E_k is a fixed expected return.

Lagrange multipliers can be used to solve for the equality constraints. Thus the constrained *QP* problem becomes

$$\text{Min } Z = \mathbf{W}'\Sigma\mathbf{W} - \phi \left(\sum_{i=1}^p w_i \mu_i - E_k \right) \quad (72)$$

$$-\lambda \left(\sum_{i=1}^p w_i - 1 \right) \text{ subject to}$$

$$0 \leq w_i \leq 1, \quad i = 1, \dots, p.$$

This is a standard QP problem with bounds $0 \leq w_i \leq 1$, $i = 1, \dots, p$. This formulation of the QP problem can be generalized to solve problems with other bounds for w_i . An example of different bounds would be

$$L_i \leq w_i \leq U_i, \quad i = 1, \dots, N, \quad (73)$$

where L_i is a lower bound imposed on a stock (buy at least a proportion L_i of a stock), and U_i is an upper bound imposed on a stock (do not buy more than a proportion U_i of the stock). Note we still have the constraint that $L_i \geq 0$.

To these ideas, Markowitz added the concept of an Efficient Frontier, which is discussed in the below.

7.2. The Efficient Frontier.

Definition 3. A portfolio is called efficient if:

- (1) For a given amount of risk, the expected return is maximized, or
For a given amount of return, the expected risk is minimized, and
- (2) The portfolio is legitimate (no negative sales).

Definition 4. Let σ_p^2 represent the x-axis and E_k represent the y-axis on a Cartesian Plane. The plot of all efficient portfolios on this set of axis is called the efficient frontier.

To compute the efficient frontier a large number of portfolio returns are generated between the largest and smallest possible returns for the portfolio. Each portfolio return is then used to find a set of weights (for the shares in the portfolio) which minimises the variance for a portfolio with that return. The returns and minimum variances are then used as the co-ordinates for the above plot. The aim of a portfolio manager is thus to have his portfolio on the efficient frontier.

8. THE SHARPE INDEX MODELS

8.1. The Sharpe Single Index Model.

Let the log return of a particular stock be

$$R_t = \log P_t - \log P_{t-1}, \quad t = 1, \dots, N \quad (74)$$

where the time t is sufficiently large enough for R_t to follow a Normal $N(\mu_r, \sigma_r^2)$ distribution.

Definition 5. Let R_t be the return at time t .

Definition 6. Let I_t be the return of the market proxy, then

$$I_t = \log I_t - \log I_{t-1}, \quad t = 1, \dots, N$$

Assume that the market proxy also follows a Normal distribution, $N(\mu_I, \sigma_I^2)$. We then have that

$$\begin{pmatrix} R_t \\ I_t \end{pmatrix} \sim N \left[\begin{pmatrix} \mu_r \\ \mu_I \end{pmatrix}, \begin{pmatrix} \sigma_r^2 & \sigma_{rI} \\ \sigma_{Ir} & \sigma_I^2 \end{pmatrix} \right] \quad (75)$$

has a bivariate normal distribution. Where $\sigma_{Ir} = \sigma_{rI}$ is the covariance between the return of stock r and the market proxy I .

From the properties of the bivariate normal distribution

$$E(R_t/I_t) = \alpha + \beta I_t \quad (76)$$

and

$$\begin{aligned} \text{var}(R_t/I_t) &= \sigma_{r,I}^2 = \sigma_I^2(1 - \rho^2) \\ &= \sigma^2 \end{aligned} \quad (77)$$

where ρ is the correlation between the stock return and the market proxy.

The model can thus be written as

$$R_t = \alpha + \beta I_t + e_t \quad (78)$$

where

$$e_t \sim N(0, \sigma^2). \quad (79)$$

It is usually assumed that the disturbances e_t are independently distributed over time, that is

$$E(e_t e_s) = 0, \text{ for } t \neq s. \quad (80)$$

This is known as the Sharpe index model.

The parameter β plays an important role in the area of finance. It can be used as a measure of the volatility of the security, relative to the market proxy. If β is greater than *one*, then, when the market rises, it is obvious that the return of a security will rise more rapidly than the return on the market. Similarly, if the market falls, the return on the security will fall more rapidly than the return on the market, and thus the security can be regarded as more volatile and hence more risky than the market. The converse is true if β is less than *one*.

Some important aspects of the assumptions of the model given in (76) which implies (78).

- (i) The bivariate normal assumption (or multivariate normal for several securities and/ or indices) appears to be well accepted in the literature.
- (ii) If the bivariate normality is not feasible then the linearity assumption (76) appears to be very well satisfied (Fama, Fisher, Jensen and Roll (1969)).
- (iii) The assumption is made that the beta coefficient is fairly stable over time, especially as the length of the period under consideration increases. This implies that historical data can be used to estimate the parameters.
- (iv) Even if the beta coefficients do change over time, the ranking of the securities in order of risk does not. This is very important if the index model is used for portfolio optimization.
- (v) The beta coefficients do give a fairly good measure of the risk inherent in a security.
- (vi) The value of beta in any period can be related to some fundamental characteristics of the firm in that period.
- (vii) The error terms e_t , if not normally distributed, then we still have $E(e_t) = 0$, $E(e_t)^2 = \sigma^2$ and $E(e_t e_s) = 0$, $t \neq s$

This implies that least squares may provide can be used to provide beta estimates.

We formally write the single model as

$$R_t = \alpha + \beta I_t + e_t, \quad t = 1, \dots, N \quad (81)$$

where the time periods t is the choice of the portfolio manager.

The assumptions are

$$\begin{aligned} E(e_t^2) &= \sigma_e^2 \\ E(e_t e_s) &= 0, \quad t \neq s = 1, \dots, N. \end{aligned} \quad (82)$$

This is our assumption for a single stock R_t .

Let the i 'th stock be

$$R_{it} = \alpha_i + \beta_i I_t + e_{it}, \quad i = 1, \dots, p; \quad t = 1, \dots, N, \quad (83)$$

where all stocks are regressed on the same single index I .

Our assumptions now become

$$E(e_{it}^2) = \sigma_{ei}^2 \quad (84)$$

$$E(e_{it} e_{is}) = 0, \quad t \neq s = 1, \dots, N, \quad (85)$$

$$E(e_{it} I_t) = 0, \quad t = 1, \dots, N, \quad (86)$$

$$E(e_{it} e_{jt}) = 0, \quad t = 1, \dots, N. \quad (87)$$

- Equation (84) assumes that each stock has a unique variance for the error term.
- Equation (85) assumes that the error terms for each of the stocks are independent over time
- Equation (86) assumes that the errors of each stock and the explanatory variable I are un-correlated - this is the usual assumption in regression.
- Equation (87) assumes that the error terms of the stocks are un-correlated, that is the stocks are only related through their mutual relationship with the index I .

This implies that least squares may provide can be used to provide beta estimates.

We formally write the single model as

$$R_t = \alpha + \beta I_t + e_t, \quad t = 1, \dots, N \quad (81)$$

where the time periods t is the choice of the portfolio manager.
The assumptions are

$$\begin{aligned} E(e_t^2) &= \sigma_e^2 \\ E(e_t e_s) &= 0, \quad t \neq s = 1, \dots, N. \end{aligned} \quad (82)$$

This is our assumption for a single stock R_t .

Let the i 'th stock be

$$R_{it} = \alpha_i + \beta_i I_t + e_{it}, \quad i = 1, \dots, p; \quad t = 1, \dots, N, \quad (83)$$

where all stocks are regressed on the same single index I .

Our assumptions now become

$$E(e_{it}^2) = \sigma_{e_i}^2 \quad (84)$$

$$E(e_{it} e_{is}) = 0, \quad t \neq s = 1, \dots, N, \quad (85)$$

$$E(e_{it} I_t) = 0, \quad t = 1, \dots, N, \quad (86)$$

$$E(e_{it} e_{jt}) = 0, \quad t = 1, \dots, N. \quad (87)$$

- Equation (84) assumes that each stock has a unique variance for the error term.
- Equation (85) assumes that the error terms for each of the stocks are independent over time
- Equation (86) assumes that the errors of each stock and the explanatory variable I are un-correlated - this is the usual assumption in regression.
- Equation (87) assumes that the error terms of the stocks are un-correlated, that is the stocks are only related through their mutual relationship with the index I .

Let

$$E(I) = \mu_I \text{ and } \text{var}(I) = \sigma_I^2 \quad (88)$$

be the mean and variance of the index.

For each stock let

$$E_i = E(R_i) = \alpha_i + \beta_i \mu_I. \quad (89)$$

The variance is

$$\begin{aligned} \text{var}(R_i) &= \text{var}(\alpha_i + \beta_i I + e_i) \\ &= \beta_i^2 \sigma_I^2 + \sigma_{e_i}^2 \\ &= \sigma_{ii} = \sigma_i^2 \end{aligned} \quad (90)$$

and the covariance is

$$\begin{aligned} \text{cov}(R_i R_j) &= E[(R_i - E(R_i))(R_j - E(R_j))] \\ &= E[(\beta_i(I - \mu_I) + e_i)(\beta_j(I - \mu_I) + e_j)] \\ &= \beta_i \beta_j \sigma_I^2. \end{aligned} \quad (91)$$

Our standard portfolio problem then becomes

$$\begin{aligned} \text{Max } Z &= \phi \mu_p - \sigma_p^2 \\ &= \phi \mathbf{W}' \boldsymbol{\mu} - \mathbf{W}' \boldsymbol{\Sigma} \mathbf{W} \text{ subject to} \\ \sum_{i=1}^p w_i &= 1, \end{aligned} \quad (92)$$

where

$$\begin{aligned} \mu_p &= \sum_{i=1}^p w_i E_i \\ &= \sum_{i=1}^p w_i (\alpha_i + \beta_i \mu_I) \end{aligned} \quad (93)$$

and

$$\begin{aligned} \sigma_p^2 &= \mathbf{W}' \boldsymbol{\Sigma} \mathbf{W} \\ &= \sum_{i=1}^p \sum_{j=1}^p w_i w_j \sigma_{ij} \end{aligned}$$

These assumptions are identical to the Single Index Model where in equation (79) we now also assume that the disturbance term e_{it} is also independent of the Indices I_j , $j = 1, \dots, M$. This again is a normal assumption in regression. We further assume that the Indices are dependent with covariances given by c_{jk} (see equation (81)).

Let

$$\begin{aligned} E_i &= E(R_i) = \alpha_i + \beta_{i1}I_1 + \dots + \beta_{iM}I_M, \\ i &= 1, \dots, p. \end{aligned} \quad (102)$$

For a portfolio $P = W'R = \sum_{i=1}^p w_i R_i$
we have that

$$\mu_p = \sum_{i=1}^p w_i E_i$$

and

$$\sigma_p^2 = \sum_{i=1}^p w_i w_j \sigma_{ij} \quad (103)$$

$$= \sum_{i=1}^p w_i^2 \sigma_{ei}^2 \quad (104)$$

$$+ \sum_{i=1}^p w_i^2 \sum_{k,l} \beta_{ik} \beta_{il} c_{kl} \quad (105)$$

$$+ \sum_{i \neq j} \sum_j w_i w_j \sum_{k,l} \beta_{ik} \beta_{jl} c_{kl} \quad (106)$$

$$= \sum_{i=1}^p w_i^2 \sigma_{ei}^2 \quad (107)$$

$$+ \sum_i \sum_j w_i w_j \sum_k \sum_l \beta_{ik} \beta_{jl} c_{kl} \quad (108)$$

$$= \sum_{i=1}^p w_i^2 \sigma_{ei}^2 \quad (109)$$

$$+ \sum_k \sum_l \sum_i \sum_j w_i \beta_{ik} w_j \beta_{jl} c_{kl} \quad (110)$$

$$+ \sum_i \sum_j w_i w_j \sum_k \sum_l \beta_{ik} \beta_{jl} c_{kl} \quad (108)$$

$$= \sum_{i=1}^p w_i^2 \sigma_{ei}^2 \quad (109)$$

$$+ \sum_k \sum_l \sum_i \sum_j w_i \beta_{ik} w_j \beta_{jl} c_{kl} \quad (110)$$

$$= \sum_{i=1}^p w_i^2 \sigma_{ei}^2 \quad (111)$$

$$+ \sum_k \sum_l \left(\sum_i w_i \beta_{ik} \right) \left(\sum_j w_j \beta_{jl} \right) c_{kl} \quad (112)$$

$$= \sum_{i=1}^p w_i^2 \sigma_{ei}^2 + \sum_k \sum_l \beta_{pk} \beta_{pl} c_{kl} \quad (113)$$

where

$$\beta_{pk} = \sum_{i=1}^p w_i \beta_{ik}, k = 1, \dots, M$$

$$\beta_{pl} = \sum_{j=1}^p w_j \beta_{jl}, l = 1, \dots, M.$$

Note that these two are the same and need only to be computed once.

The Sharpe formulation is then (minimize instead of maximize)

$$\text{Min } Z = -\Lambda \mu_p + \sigma_p^2$$

subject to

$$\beta_{p1} = \sum_{i=1}^p w_i \beta_{i1}$$

$$\beta_{p2} = \sum_{i=1}^p w_i \beta_{i2}$$

\vdots

$$\beta_{pM} = \sum_{i=1}^p w_i \beta_{iM}$$

$$\begin{aligned}
\text{Min } Z' &= -\Lambda\mu_p + \sigma_p^2 \\
&+ \lambda_1(\beta_{p1} - \sum_{i=1}^p w_i\beta_{i1}) \\
&+ \lambda_2(\beta_{p2} - \sum_{i=1}^p w_i\beta_{i2}) \\
&\vdots \\
&+ \lambda_M(\beta_{pM} - \sum_{i=1}^p w_i\beta_{iM}) \\
&+ \lambda_f(1 - \sum_{i=1}^p w_i).
\end{aligned}$$

The solution requires that the partial derivative of Z' with respect to each variable be set to zero.

For each i from 1 to p ,

$$\frac{\partial Z'}{\partial w_i} = 2\sigma_{ei}^2 - E_i\Lambda - \lambda_1\beta_{i1} - \dots - \lambda_M\beta_{iM} = 0.$$

For each j from 1 to M ,

$$\frac{\partial Z'}{\partial \beta_{pj}} = 2\beta_{p1}c_{j1} + 2\beta_{p2}c_{j2} + \dots + 2\beta_{pM}c_{jM} + \lambda_j = 0.$$

For each j from 1 to M ,

$$\frac{\partial Z'}{\partial \lambda_j} = \beta_{pj} - \beta_{1j}w_1 - \dots - \beta_{pj}w_p = 0.$$

For λ_f ,

$$\frac{\partial Z'}{\partial \lambda_f} = 1 - w_1 - \dots - w_p = 0.$$

Below is the system of linear equations $AX = B$ that need to be solved.

$$\begin{pmatrix}
 w_1 & \cdots & w_p & \beta_{p1} & \cdots & \beta_{pM} & \lambda_1 & \cdots & \lambda_M & \lambda_f \\
 2\sigma_{e1}^2 & \cdots & 0 & 0 & \cdots & 0 & -\beta_{11} & \cdots & -\beta_{1M} & -1 \\
 0 & \ddots & \vdots & \vdots & \cdots & 0 & \vdots & & \vdots & \ddots \\
 0 & \cdots & 2\sigma_{ep}^2 & 0 & \cdots & 0 & -\beta_{p1} & \cdots & -\beta_{pM} & -1 \\
 0 & \cdots & 0 & 2c_{11} & \cdots & 2c_{1M} & 1 & 0 & \cdots & 0 \\
 \vdots & & & \vdots & \ddots & \vdots & & \ddots & & \vdots \\
 0 & \cdots & 0 & 2c_{M1} & \cdots & 2c_{MM} & 0 & \cdots & 1 & 0 \\
 -\beta_{11} & & -\beta_{p1} & 1 & 0 & & & & & 0 \\
 \vdots & & \vdots & 0 & \ddots & 0 & \cdots & & & 0 \\
 -\beta_{1M} & & -\beta_{pM} & \vdots & & 1 & 0 & \cdots & & 0 \\
 -1 & \cdots & -1 & 0 & \cdots & & & & & 0
 \end{pmatrix}
 =
 \begin{pmatrix}
 0 \\
 \Lambda E_1 \\
 \vdots \\
 \Lambda E_p \\
 0 \\
 \vdots \\
 0 \\
 -1
 \end{pmatrix}$$

$$AX = B,$$

with solution

$$X = A^{-1}B.$$

provided that no other equality, in-equality or bounds are imposed. If there are such restrictions they must be added to the systems of equations. The problem may then no longer be a calculus problem but rather a QP problem.

8.3. A comparison of the Markowitz and Sharpe single and multiple index models.

Efficient frontiers for each of the above formulations were generated for nine JSE listed shares. The shares used in the analysis are,

- Anglos
- De Beers
- Gencor
- Implats
- Metlife
- Remgro
- SAB

- Sappi
- Stanbic

A sample of 149 returns were used to estimate the required parameters and a detailed analysis of this sample is given in Part III.

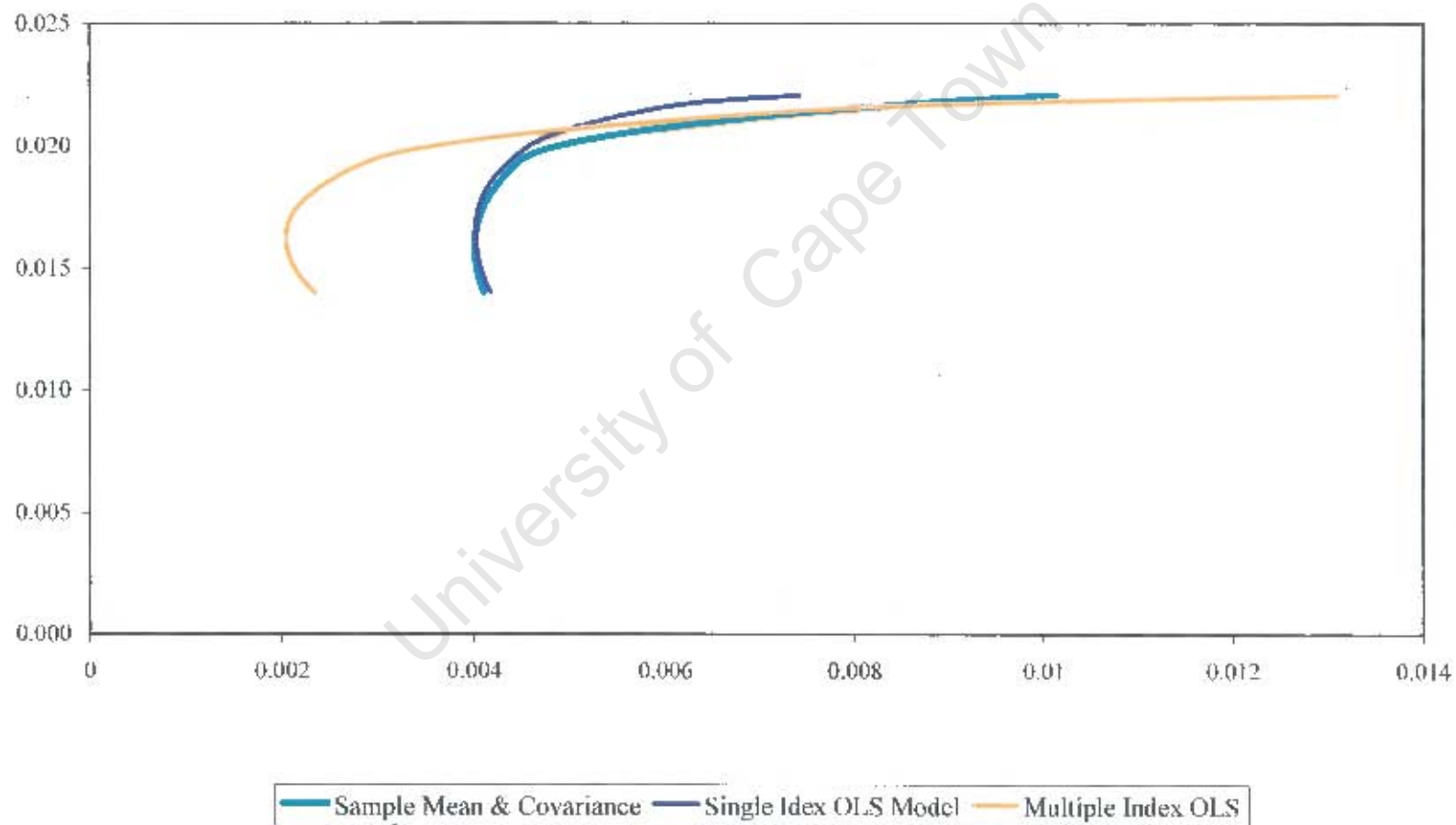
The sample mean and covariance matrix was used as inputs to the Markowitz formulation. For the Sharpe single index model parameters were estimated by regressing returns against the JSE All Share index return. The parameters for the Sharpe multiple index were estimated by regressing returns against the following indices:

- JSE All Share Index
- JSE Banks Index
- JSE Gold Index
- JSE Insurance Index
- JSE Platinum Index

The efficient frontiers for the three formulations are given in Figure 1 below. The differences between the Markowitz formulation and Sharpe single index model is slight and simplifications in the single index model appears not to affect the efficient frontiers adversely. The multiple index model outperforms the other two formulations by a long way and clearly benefits from the additional information that it incorporates into its formulation.

Figure 1

Classical Formulations



9. INNOVATIONS TO THE CLASSICAL MODELS

9.1. Generalisation of the Markowitz Formulation.

The classical Markowitz formulation of the portfolio problem is

$$\text{Min } \sigma_p^2 = \mathbf{W}'\Sigma\mathbf{W} \quad (114)$$

$$= \sum_{i=1}^p w_i w_j \sigma_{ij} \quad (115)$$

subject to

$$\begin{aligned} \mu_p &= \mathbf{W}'\boldsymbol{\mu}_p \\ &= \sum_{i=1}^p w_i \mu_i \\ &= E_k \end{aligned}$$

and subject to

$$\sum_{i=1}^p w_i = 1$$

and

$$0 \leq w_i \leq 1,$$

where

$$\Sigma = (\sigma_{ij}), i, j = 1, \dots, p.$$

Short sales (negative w_i) are common in markets around the world and it is no longer necessary to restrict portfolios to be legitimate.

The generalised formulation of the Markowitz Model is

$$\text{Min } \sigma_p^2 = \mathbf{W}'\Sigma\mathbf{W} \quad (116)$$

$$= \sum_{i=1}^p \sum_{j=1}^p w_i w_j \sigma_{ij} \quad (117)$$

subject to

$$\begin{aligned}\mu_p &= \mathbf{W}'\boldsymbol{\mu}_p \\ &= \sum_{i=1}^p w_i \mu_i \\ &= E_k\end{aligned}$$

and subject to

$$\sum_{i=1}^p w_i = 1$$

where

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1p} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \cdots & \sigma_{pp} \end{pmatrix} \quad (118)$$

where $\sigma_{ii} = \sigma_i^2$ is the variance of the i 'th stock and σ_{ij} is the covariance between the i 'th and j 'th stock. In this formulation no restrictions are put on the weights w_i . Because the weights can take negative values they include the possibility of short selling.

9.2. Innovations to the Sharpe single index model.

Use of Bayesian estimates in the Sharpe single index model.

The posterior mean of the predictive distribution of a future observation of the return of the JSE would be the ideal choice. If we call this posterior predictive mean I_f and if

$$\mathbf{R}_f = \begin{pmatrix} R_{1f} \\ R_{2f} \\ \vdots \\ R_{pf} \end{pmatrix}$$

are the future predictive returns of the p -stocks, then setting a prior distribution for

$$\begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{pmatrix}$$

we can derive the posterior distribution of

$$\begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

from which we can derive the posterior distribution of the predictive distribution of R_f which will depend on the posterior predictive mean I_f . The posterior mean and covariance matrix of R_f can be computed and used as the input for portfolio analysis. The rest of the calculations will be exactly the same as given above.

An alternative semi Bayesian approach would be to use the confidence and predictive intervals of future values of each of the elements of

$$R_f = \begin{pmatrix} R_{1f} \\ R_{2f} \\ \vdots \\ R_{pf} \end{pmatrix}$$

with I_f as the value for the market proxy.

Non-zero covariances between residuals of shares.

The classical Sharpe single index model is formulated as

$$R_{it} = \alpha_i + \beta_i I_t + e_{it}, \quad i = 1, \dots, p; \quad t = 1, \dots, N,$$

$$E(e_{it}^2) = \sigma_{ei}^2 \tag{119}$$

$$E(e_{it}e_{is}) = 0, \quad t \neq s = 1, \dots, N, \tag{120}$$

$$E(e_{it}I_t) = 0, \quad t = 1, \dots, N, \tag{121}$$

$$E(e_{it}e_{jt}) = 0, \quad t = 1, \dots, N. \tag{122}$$

In vector notation

$$\mathbf{R}_t = \boldsymbol{\alpha} + \beta I_t + \mathbf{e}_t \quad t = 1, \dots, N \quad (123)$$

where

$$\mathbf{R}_t = \begin{pmatrix} R_{1t} \\ \vdots \\ R_{pt} \end{pmatrix}, \quad \boldsymbol{\alpha} = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_p \end{pmatrix}, \quad \beta = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_p \end{pmatrix} \quad \text{and} \quad \mathbf{e}_t = \begin{pmatrix} e_{1t} \\ \vdots \\ e_{pt} \end{pmatrix} \quad (124)$$

so that (dropping the index t)

$$E(\mathbf{R}) = \boldsymbol{\alpha} + \beta \mu_I \quad (125)$$

and

$$\text{cov}(\mathbf{e}) = \begin{pmatrix} \sigma_{e1}^2 & 0 & \dots & 0 \\ 0 & \sigma_{e2}^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & \sigma_{ep}^2 \end{pmatrix}. \quad (126)$$

This implies that

$$\text{cov}(\mathbf{R}) = \sigma_I^2 \beta \beta' + \begin{pmatrix} \sigma_{e1}^2 & 0 & \dots & 0 \\ 0 & \sigma_{e2}^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & \sigma_{ep}^2 \end{pmatrix}. \quad (127)$$

Here we have assumed that all e_{it} , $i = 1, \dots, p$, $t = 1, \dots, N$ are independent. We now relax this assumption by assuming that the disturbances e_{it} of the different stocks are dependent (correlated).

Thus

$$\begin{aligned} E(e_{it}e_{jt}) &= \sigma_{ij}, \quad i \neq j \\ &= \sigma_{ei}^2 \quad i = j \\ &= \sigma_{ii}. \end{aligned}$$

or

$$E(\mathbf{e}\mathbf{e}') = \boldsymbol{\Omega} = \begin{pmatrix} \sigma_{e1}^2 & \sigma_{12} & \dots & \sigma_{1p} \\ \sigma_{21} & \sigma_{e2}^2 & \dots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \dots & \dots & \sigma_{ep}^2 \end{pmatrix} \quad (128)$$

with

$$\text{cov}(\mathbf{R}) = \sigma_I^2 \boldsymbol{\beta} \boldsymbol{\beta}' + \boldsymbol{\Omega} = \boldsymbol{\Phi} \quad (129)$$

For portfolio $P = \mathbf{W}'\mathbf{R}$ we have

$$E(P) = \mathbf{W}'(\boldsymbol{\alpha} + \boldsymbol{\beta}\mu_I) = \mu_p \quad (130)$$

and

$$\text{var}(P) = \mathbf{W}'(\sigma_I^2 \boldsymbol{\beta} \boldsymbol{\beta}' + \boldsymbol{\Omega})\mathbf{W} \quad (131)$$

$$= \mathbf{W}'\boldsymbol{\Phi}\mathbf{W} = \sigma_p^2 \quad (132)$$

and we maximise

$$Z = \phi\mu_p - \sigma_p^2 \quad (133)$$

subject to

$$\sum_{i=1}^p w_i = 1$$

and any other equality or inequality constraints and bounds which the portfolio manager would like to impose.

This is again a QP problem and can be solved by using the simultaneous set of equations given by (116) with $\boldsymbol{\Phi}$ replacing $\boldsymbol{\Sigma}$.

The quantities to be estimated are:

$$\mu_I, \sigma_I^2, \boldsymbol{\alpha}, \boldsymbol{\beta} \text{ and } \boldsymbol{\Omega}. \quad (134)$$

To estimate $\boldsymbol{\Omega}$ let

$$\hat{\mathbf{E}} = \begin{pmatrix} \hat{e}_{11} & \hat{e}_{12} & \dots & \hat{e}_{1N} \\ \hat{e}_{21} & \hat{e}_{22} & \dots & \hat{e}_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ \hat{e}_{p1} & \dots & \dots & \hat{e}_{pN} \end{pmatrix} \quad (135)$$

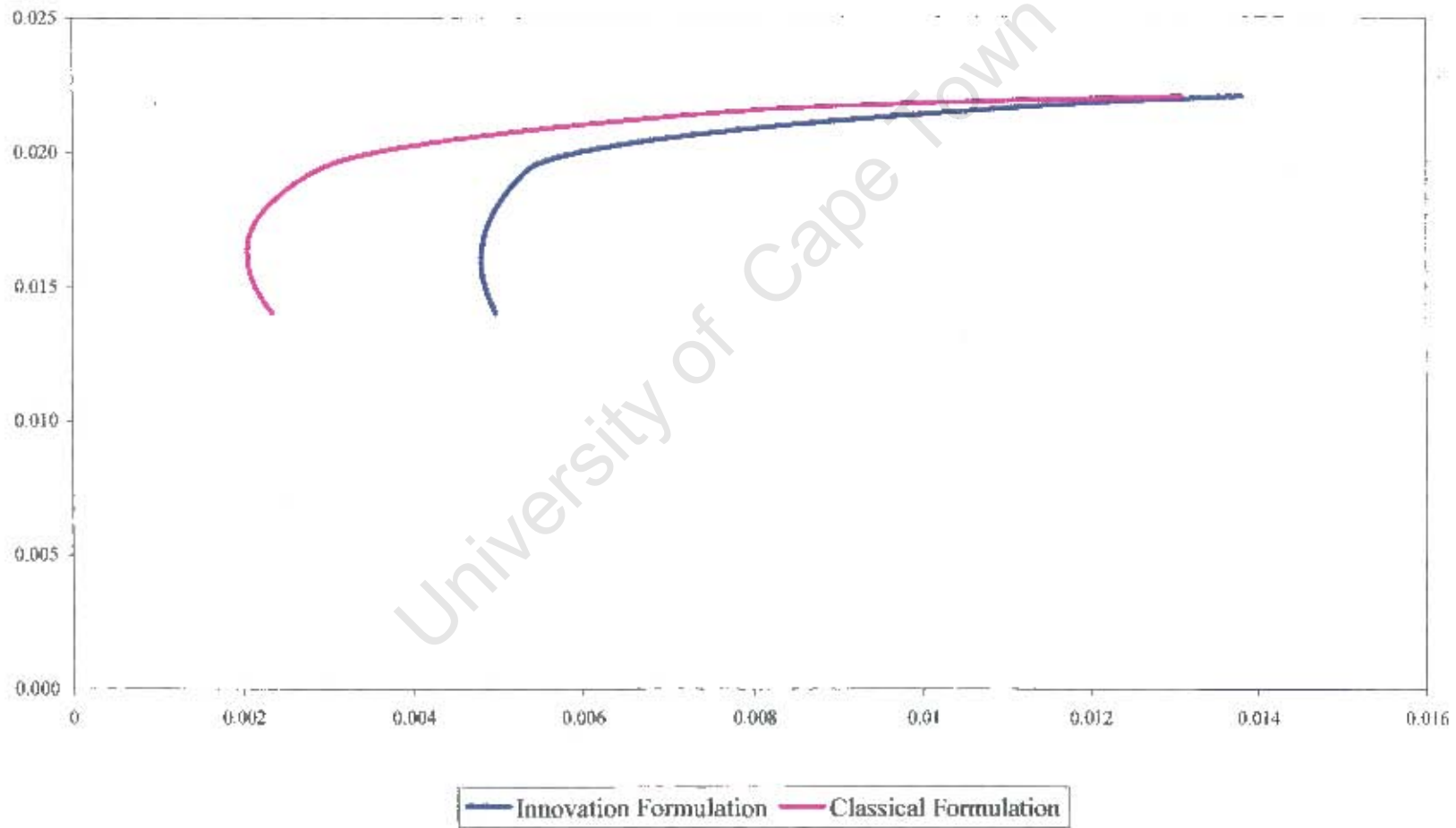
then

$$\hat{\boldsymbol{\Omega}} = \frac{1}{N-2} \hat{\mathbf{E}}\hat{\mathbf{E}}' \quad (136)$$

and

$$\hat{\boldsymbol{\Phi}} = \hat{\sigma}_I^2 \hat{\boldsymbol{\beta}}\hat{\boldsymbol{\beta}}' + \hat{\boldsymbol{\Omega}}. \quad (137)$$

Figure 2 below gives a plot of the efficient frontiers of the Sharpe single index model with and without non-zero covariances for the residuals. The efficient frontiers are generated using the same data discussed earlier. As can be seen from the plot there appears to be no advantage in including the error covariance terms in the formulation because it is dominated across all possible returns by the classical formulation.

Figure 2**Sharpe Single Index Classical vs Innovation Formulation**

9.3. Innovations to the Sharpe multiple index model.

The revised multi - index model can be written as

$$\begin{aligned} R_{it} &= \alpha_i + \beta_{i1}I_{1t} + \beta_{i2}I_{2t} + \dots + \beta_{iM}I_{Mt} + e_{it}, \\ i &= 1, \dots, p, \quad t = 1, \dots, N, \end{aligned} \quad (138)$$

with the following assumptions

$$E(e_{it}^2) = \sigma_{ei}^2 \quad (139)$$

$$E(e_{it}e_{jt}) = \sigma_{ij}, \quad i \neq j, \quad t = 1, \dots, N. \quad (140)$$

$$E(e_{it}e_{is}) = 0, \quad t \neq s = 1, \dots, N, \quad (141)$$

$$E(e_{it}I_{jt}) = 0, \quad j = 1, \dots, M, \quad t = 1, \dots, N, \quad (142)$$

$$E(I_{jt}I_{kt}) = c_{jk}, \quad j, k = 1, \dots, M \quad (143)$$

It is now also assumed that the disturbance term e_{it} has a covariance term (correlated) but from assumption (142) it remains independent of the Indices I_j , $j = 1, \dots, M$. This last assumption is again the normal assumption in regression. We further assume that the Indices are dependent with covariances given by c_{jk} .

Let

$$\begin{aligned} E_i &= E(R_i) = \alpha_i + \beta_{i1}\mu_1 + \dots + \beta_{iM}\mu_M, \\ i &= 1, \dots, p. \end{aligned} \quad (144)$$

Let

$$\boldsymbol{\alpha} = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_p \end{pmatrix}, \quad \boldsymbol{\beta} = \begin{pmatrix} \beta_{11} & \dots & \beta_{1M} \\ \vdots & \vdots & \vdots \\ \beta_{p1} & \dots & \beta_{pM} \end{pmatrix}, \quad \mathbf{e}_t = \begin{pmatrix} e_{1t} \\ \vdots \\ e_{pt} \end{pmatrix}$$

$$\mathbf{R}_t = \begin{pmatrix} R_{1t} \\ \vdots \\ R_{pt} \end{pmatrix}, \quad \mathbf{I}_t = \begin{pmatrix} I_{1t} \\ \vdots \\ I_{Mt} \end{pmatrix}, \quad t = 1, \dots, N,$$

then

$$\mathbf{R}_t = \boldsymbol{\alpha} + \boldsymbol{\beta}\mathbf{I}_t + \mathbf{e}_t \quad (145)$$

with

$$E(\mathbf{R}_t) = \alpha + \beta\mu_{\mathbf{I}} \quad (146)$$

and

$$\mu_{\mathbf{I}} = \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_M \end{pmatrix}. \quad (147)$$

The covariance matrix of R_t is then

$$\begin{aligned} cov(\mathbf{R}_t) &= E[\mathbf{R}_t - E(\mathbf{R}_t)][\mathbf{R}_t - E(\mathbf{R}_t)]' \\ &= E[\alpha + \beta\mathbf{I}_t + \mathbf{e}_t - \alpha + \beta\mu_{\mathbf{I}}][\alpha + \beta\mathbf{I}_t + \mathbf{e}_t - \alpha + \beta\mu_{\mathbf{I}}]' \\ &= E[\beta(\mathbf{I}_t - \mu_{\mathbf{I}}) + \mathbf{e}_t][\beta(\mathbf{I}_t - \mu_{\mathbf{I}}) + \mathbf{e}_t]' \\ &= \beta E(\mathbf{I}_t - \mu_{\mathbf{I}})(\mathbf{I}_t - \mu_{\mathbf{I}})'\beta' + E(\mathbf{e}_t\mathbf{e}_t') \quad \text{since } E(\mathbf{I}_t\mathbf{e}_t') = 0 \\ &= \beta\mathbf{C}\beta' + \Omega \\ &= \Phi. \end{aligned} \quad (148)$$

For portfolio

$$\begin{aligned} P &= \mathbf{W}'\mathbf{R} \\ &= \sum_{i=1}^p w_i R_i \end{aligned}$$

we have that

$$E_p = E(P) = \mathbf{W}'(\alpha + \beta\mu_{\mathbf{I}}) \quad (149)$$

and

$$\sigma_p^2 = var(P) = \mathbf{W}'\Phi\mathbf{W}. \quad (150)$$

The portfolio problem is then

$$\min Z = -\Lambda E_p + \sigma_p^2 \quad (151)$$

$$= -\Lambda\mathbf{W}'(\alpha + \beta\mu_{\mathbf{I}}) + \mathbf{W}'\Phi\mathbf{W}. \quad (152)$$

subject to

$$\sum_{i=1}^p w_i = 1$$

and any other equalities, in-equalities or bounds that the portfolio manager would like to impose on the portfolio.

This is again a QP -problem. If there are no further equalities, in-equalities or bounds, then the problem can be solved using calculus by using equations (92) with Φ substituted for Σ .

Our estimates would be

$$\hat{E}_p = \mathbf{W}'\hat{\mathbf{E}} \quad (153)$$

where

$$\hat{\mathbf{E}} = \begin{pmatrix} \hat{E}_1 \\ \vdots \\ \hat{E}_p \end{pmatrix}$$

can again be estimated in two ways that is

$$\hat{E}_i = \frac{1}{N} \sum_{t=1}^N R_{it} \quad (154)$$

or

$$\hat{\mathbf{E}}_i = \hat{\boldsymbol{\alpha}} + \hat{\boldsymbol{\beta}}\hat{\boldsymbol{\mu}}_{\mathbf{I}} \quad (155)$$

where

$$\hat{\boldsymbol{\alpha}} = \begin{pmatrix} \hat{\alpha}_1 \\ \vdots \\ \hat{\alpha}_p \end{pmatrix}, \quad \hat{\boldsymbol{\beta}} = \begin{pmatrix} \hat{\beta}_{11} & \cdots & \hat{\beta}_{1M} \\ \vdots & \vdots & \vdots \\ \hat{\beta}_{p1} & \cdots & \hat{\beta}_{pM} \end{pmatrix} \quad (156)$$

and

$$\hat{\boldsymbol{\mu}}_{\mathbf{I}} = \begin{pmatrix} \hat{\mu}_{I_1} \\ \vdots \\ \hat{\mu}_{I_M} \end{pmatrix} \quad (157)$$

with $\hat{\mu}_{\mathbf{I}}$ some estimate of $\mu_{\mathbf{I}}$ (forecast or prediction or Bayesian estimate). If

$$\hat{\mu}_{I_j} = \frac{1}{N} \sum_{t=1}^N I_{jt}$$

(which is the sample mean of the j 'th Index) then \hat{E}_i and $\hat{\mu}_{\mathbf{I}}$ are identical.

For the estimate of σ_p^2 we have

$$\hat{\sigma}_p^2 = \mathbf{W}'\hat{\Phi}\mathbf{W}$$

with

$$\hat{\Phi} = \hat{\boldsymbol{\beta}}\hat{\mathbf{C}}\hat{\boldsymbol{\beta}}' + \hat{\Omega}.$$

The matrix \hat{C} is the estimated covariance matrix of the M Indices and $\hat{\Omega}$ is estimated by

$$\hat{\Omega} = \frac{1}{N - M - 1} \hat{E} \hat{E}' \quad (158)$$

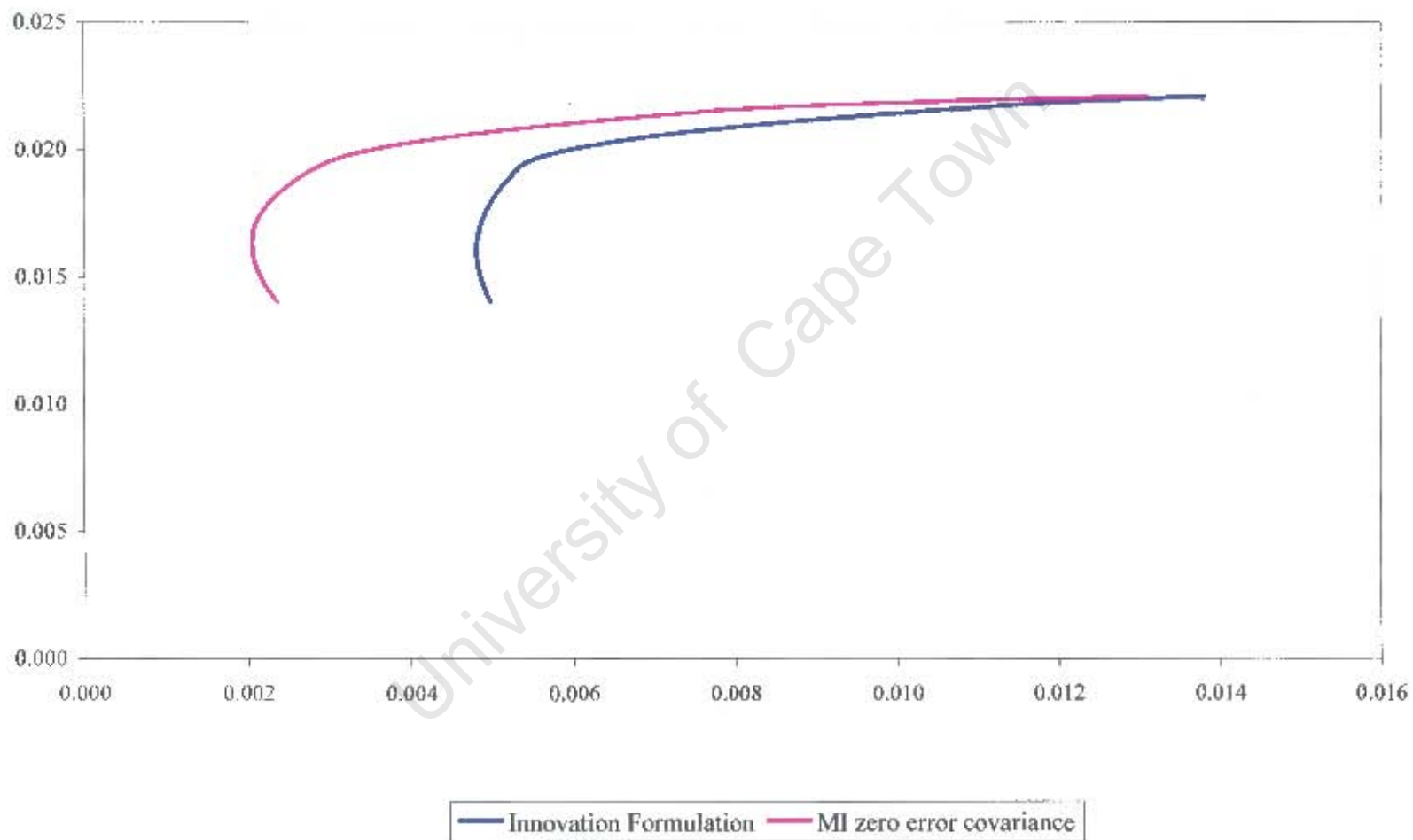
and

$$\hat{E} = \begin{pmatrix} \hat{e}_{11} & \hat{e}_{12} & \dots & \hat{e}_{1N} \\ \hat{e}_{21} & \hat{e}_{22} & \dots & \hat{e}_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ \hat{e}_{p1} & \dots & \dots & \hat{e}_{pN} \end{pmatrix}. \quad (159)$$

Figure 3 below gives a plot of the efficient frontiers of the of the Sharpe multiple index model with and without non-zero covariances for the residuals. The efficient frontiers are generated using the same data discussed earlier. As can be seen from the plot there is no advantage in including the error covariance terms in the formulation as the classical formulation dominates the other across all possible returns.

Figure 3

Sharpe Single Index Classical vs Innovation Formulation



9.4. Further innovations to the single and multiple index models by Troskie.

For the single index model

$$\mathbf{R} = \boldsymbol{\alpha} + \beta I + e$$

let $P = W'R$ be our portfolio with mean $E(P) = \mu = W'E = W'\mu$ and variance σ_p^2 . The portfolio manager may decide that he wants a given μ for his portfolio say,

$$\begin{aligned} \mu_0 &= W'E_0 \\ &\geq E_I \end{aligned}$$

where E_I is say the mean of the market proxy, say the overall index. The strategy is to construct portfolios that will beat the market.

Suppose we further want to construct a given β_0 for our portfolio that is

$$\begin{aligned} W'\beta &= \beta_0 \\ &> 1 \end{aligned}$$

- high risk portfolio when the market is in an upswing or

$$\begin{aligned} W'\beta &= \beta_0 \\ &< 1 \end{aligned}$$

- low risk portfolio when the market is in a downswing.

Thus our most general portfolio problem is

$$\text{Minimize } \sigma_p^2 = W'\Phi W \tag{160}$$

subject to

$$W'\mu = W'E \tag{161}$$

$$= \mu$$

$$W'\hat{\beta} = \hat{\beta}$$

$$W'\mathbf{1} = 1, \mathbf{1}' = (1, \dots, 1),$$

and adding any other bounds, equalities and in-equalities.

This is a general QP -problem and the solution is straightforward - if it exists. If a solution can not be obtained -the iterative process will not converge - then some of the assumptions must be relaxed or changed.

If there are no bounds and in-equality constraints then the problem can be solved using calculus. The calculus procedure with no bounds which allows for both positive (buying) and negative (short - selling) trading is the most common practice.

The Lagrangian equation becomes

$$Z' = \mathbf{W}'\Phi\mathbf{W} + 2\lambda_\beta(\mathbf{W}'\hat{\beta} - \hat{\beta}) + 2\lambda_\mu(\mathbf{W}'\boldsymbol{\mu} - \mu) + 2\lambda(\mathbf{W}'\mathbf{1} - 1) \quad (162)$$

with minimum solution

$$\begin{pmatrix} \Phi & \hat{\beta} & \mu & \mathbf{1} \\ \hat{\beta}' & 0 & 0 & 0 \\ \mu' & 0 & 0 & 0 \\ \mathbf{1}' & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{W} \\ \lambda_\beta \\ \lambda_\mu \\ \lambda \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \hat{\beta} \\ \mu \\ 1 \end{pmatrix} \quad (163)$$

or

$$\mathbf{M}\mathbf{u} = \mathbf{v} \quad (164)$$

where

$$\mathbf{M} = \begin{pmatrix} \Phi & \Gamma \\ \Gamma' & \mathbf{0} \end{pmatrix}, \quad \text{for } \Gamma = (\hat{\beta}, \mu, \mathbf{1}),$$

with solution

$$\mathbf{u} = \mathbf{M}^{-1}\mathbf{v},$$

provided M is non-singular.

The matrix M can be evaluated using the rule for partitioned inverses

$$\mathbf{M}^{-1} = \begin{bmatrix} \Phi^{-1}(\mathbf{I} - \Gamma(\Gamma'\Phi^{-1}\Gamma)^{-1}\Gamma'\Phi^{-1}) & \Phi^{-1}\Gamma(\Gamma'\Phi^{-1}\Gamma)^{-1} \\ (\Gamma'\Phi^{-1}\Gamma)^{-1}\Gamma'\Phi^{-1} & -(\Gamma'\Phi^{-1}\Gamma)^{-1} \end{bmatrix}. \quad (165)$$

If the solution of W is written as \hat{W} then the portfolio variance is given by

$$\sigma_p^2 = \hat{\mathbf{W}}'\Phi\hat{\mathbf{W}} \quad (166)$$

with

$$\hat{\sigma}_p^2 = \hat{\mathbf{W}}'\hat{\Phi}\hat{\mathbf{W}}. \quad (167)$$

For the multi-index model given by

$$\mathbf{R}_t = \boldsymbol{\alpha} + \boldsymbol{\beta}\mathbf{I}_t + \mathbf{e}_t, \quad \boldsymbol{\beta}_{(p \times M)}$$

with

$$E_p = E(P) = \mathbf{W}'(\boldsymbol{\alpha} + \boldsymbol{\beta}\boldsymbol{\mu}_T) \quad (168)$$

and

$$\sigma_p^2 = \text{var}(P) = \mathbf{W}'\boldsymbol{\Phi}\mathbf{W}. \quad (169)$$

the constraints will be

$$\begin{aligned} \mathbf{W}'\boldsymbol{\mu} &= \mathbf{W}'\mathbf{E} \\ &= \boldsymbol{\mu} \end{aligned} \quad (170)$$

and

$$\mathbf{W}'\boldsymbol{\beta} = \hat{\boldsymbol{\beta}}_0' = (\hat{\beta}_{01}, \dots, \hat{\beta}_{0M})$$

or

$$\boldsymbol{\beta}'\mathbf{W} = \hat{\boldsymbol{\beta}}_0.$$

The systems of equations will be the same but with the $(p \times 1)$ vector $\hat{\boldsymbol{\beta}}$ replaced by the $(p \times M)$ matrix $\hat{\boldsymbol{\beta}}$ and the scalar $\hat{\beta}$ replaced by the $(p \times 1)$ vector $\hat{\boldsymbol{\beta}}_0$.

Part III: Application of Robust Beta Estimation to Modern Portfolio Theory

From the simulation study conducted in Part I it is clear that OLS is not a good estimation procedure when the data does not conform to the normal distribution and highlights the need for robust beta estimation procedures. It is widely believed that stock price and index movements do not conform to normality and in particular have distributions with long tails and large kurtosis. In this section robust beta estimation is applied to the Sharpe Index models and an empirical analysis is performed using nine shares that are listed on the Johannesburg Stock Exchange. The aim of this analysis is to determine whether or not robust beta estimation can produce superior regression fits as well as improved efficient frontiers.

University of Cape Town

10. DATA USED IN THE ANALYSIS

The nine shares were used in the analysis these are:

- Anglos
- De Beers
- Gencor
- Implats
- Metlife
- Remgro
- SAB
- Sappi
- Stanbic

These shares were selected because they represent a cross section of companies listed on the JSE. The chosen shares all have large market capitalisations and do not have any of the problems associated with thin trading, a problem that afflicts a large number of shares listed on the JSE. The problem of thin trading is discussed by Bowie (1994) and will not be dealt with in this thesis. For the Sharpe Single Index model the JSE All Share Index was used as the market proxy.

For the Sharpe Multiple Index model the following indices were used as market proxies:

- JSE All Share Index
- JSE Banks Index
- JSE Gold Index
- JSE Insurance Index
- JSE Platinum Index

Month end prices and index values were collected for each of the shares and indices for the period July 1988 to December 2000. This yielded 150 data points. From these 149 monthly returns were calculated, using equation (49), for each of the shares and indices. A complete list of share prices and index values is given in the appendix.

11. ANALYSIS OF THE SHARES USED

The table below contains the summary statistics of shares used in the analysis.

Share	μ	σ^2	Skewness	Kurtosis	Z_1	Z_2	χ^2
Anglos	0.013	0.010	0.231	4.677	1.151	4.178	18.779
De Beers	0.011	0.009	-0.246	3.805	-1.226	2.005	5.523
Gencor	0.017	0.009	-0.624	4.523	-3.112	3.795	24.085
Implats	0.017	0.015	0.145	2.722	0.722	-0.693	1.001
Metlife	0.022	0.011	-0.442	5.493	-2.204	6.212	43.448
Remgro	0.018	0.008	0.162	3.516	0.810	1.287	2.312
SAB	0.012	0.006	-0.228	4.615	-1.137	4.025	17.494
Sappi	0.004	0.014	0.734	5.407	3.659	5.998	49.371
Stanbic	0.021	0.009	-1.132	10.512	-5.642	18.717	382.148

From the above table it is clear that there are significant departures from normality for some of the shares, particularly Stanbic and Metlife that have large negative skewness and very large kurtosis. From the simulation study in Part I we can deduce that Ordinary Least Squares may not yield an acceptable regression fit and alternative robust procedures may need to be used.

Z_1 and Z_2 and χ^2 are defined as follows

$$Z_1 = \frac{\sum_{i=1}^n (x_i - \bar{x})^3 / n}{s^3 \sqrt{6/n}} \sim N(0, 1)$$

$$Z_2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^4 / n - 3}{s^4 \sqrt{24/n}} \sim N(0, 1)$$

$$Z_1^2 + Z_2^2 \sim \chi_2^2$$

and represent a skewness and kurtosis test for normality. The critical values for at a 5% significance level for Z_1 and Z_2 are ± 1.96 and for $\chi^2 \approx 5$. It is clear that there are some significant departures from normality.

The summary statistics of the JSE All Share Index are given below.

Proxy	μ	σ^2	Skewness	Kurtosis	Z_1	Z_2	χ^2
JSE All Share Index	0.01023	0.004	-1.53687	11.19216	-7.65869	20.41202	475.306

As can be seen from the table above the Index clearly does not conform to the normal distribution.

12. THE SHARPE SINGLE INDEX MODEL

The Sharpe Single Index Model is widely used in practice with parameters estimated using Ordinary Least Squares (OLS). The efficacy of robust procedures has already been discussed above, in this section we compare the OLS approach various robust procedures to test their effectiveness. The robust procedures used are non-weighted L_1 and L_p beta estimates and weighted OLS, L_1 and L_p beta estimates using the Mallows weights discussed in Part I.

12.1. Ordinary Least Squares vs L_1 and L_p regression.

Regression Summary.

The nine shares were regressed against the JSE All Share Index. A summary of the OLS, L_1 and L_p regressions is given in the tables below and a brief discussion follows.

OLS	$\hat{\alpha}$	$\hat{\beta}$	$\hat{\sigma}_e^2$	R^2	β_{lower} 95%	β_{upper} 95%	t - stat
Anglos	-0.0006	1.3649	0.0033	0.6681	1.20316	1.52174	17.2032
De Beers	0.0001	1.0924	0.0046	0.4831	0.90824	1.2766	11.7215
Gencor	0.0083	0.8931	0.0065	0.3048	0.67325	1.11295	8.0281
Implats	0.0070	1.0103	0.0118	0.2366	0.71449	1.30605	6.7500
Metlife	0.0121	0.9800	0.0046	0.3169	0.45183	0.99418	5.2690
Remgro	0.0082	0.9703	0.0042	0.4453	0.79377	1.14864	10.8621
SAB	0.0027	0.9054	0.0034	0.4652	0.74716	1.06363	11.3078
Sappi	-0.0077	1.1461	0.0094	0.3344	0.88257	1.40963	8.5947
Stanbic	0.0105	0.9882	0.0060	0.3701	0.77805	1.19835	9.2931

L_1	$\hat{\alpha}$	$\hat{\beta}$	$\hat{\sigma}_e^2$	R^2	β_{lower} 95%	β_{upper} 95%	t - stat
Anglos	-0.0052	1.4066	0.0047	0.6658	1.22017	1.59311	14.9078
De Beers	-0.0002	1.2032	0.0070	0.4781	0.97465	1.43151	10.4164
Gencor	0.0096	0.9846	0.0035	0.3011	0.82375	1.14546	12.0967
Implats	0.0071	1.0991	0.0146	0.2347	0.77008	1.42827	6.6013
Metlife	0.0152	0.7230	0.0070	0.2951	0.45183	0.99418	5.2690
Remgro	0.0037	0.9410	0.0046	0.4418	0.75560	1.12646	10.0289
SAB	0.0083	0.8464	0.0057	0.4592	0.64068	1.0521	8.1312
Sappi	-0.0047	1.0302	0.0054	0.3308	0.82966	1.2307	10.1530
Stanbic	0.0105	0.7999	0.0065	0.3562	0.58008	1.01975	7.1910

L_p	\hat{p}	$\hat{\alpha}$	$\hat{\beta}$	$\hat{\sigma}_e^2$	R^2	β_{lower} 95%	β_{upper} 95%	t - stat
Anglos	1.282	-0.0046	1.4076	0.0009	0.666	1.20814	1.52174	17.20316
De Beers	1.673	0.0004	1.0973	0.0050	0.483	0.90547	1.43151	10.41638
Gencor	1.138	0.0093	0.9897	0.0016	0.301	0.87992	1.09954	17.81252
Implats	1.910	0.0065	1.0199	0.0114	0.237	0.77008	1.42817	6.60128
Metlife	1.866	0.0124	0.9643	0.0050	0.317	0.73071	1.19792	5.26895
Remgro	1.689	0.0070	0.9628	0.0036	0.445	0.79851	1.12708	11.58172
SAB	1.661	0.0039	0.8765	0.0034	0.465	0.71782	1.0352	10.91546
Sappi	1.314	-0.0056	1.0647	0.0034	0.332	0.88771	1.20564	13.01205
Stanbic	1.544	0.0114	0.9332	0.0054	0.369	0.73221	1.13426	9.17446

The tables above produce a number of interesting results, even though the R^2 value for the OLS estimates are generally higher than that of the L_1 and L_p estimates the differences are quite small. The residual error terms, σ_e^2 , for the L_p model was generally lower than the OLS model. The L_1 model did not generally show any reduction in σ_e^2 when compared to OLS. The values for α and β were generally quite different for the three models.

Efficient Frontiers.

Though the regression output showed that the robust procedures produced improved results with regard to some aspects, we needed to test whether these procedures could help to produce superior portfolio construction tools. In order to test this, the results of the above regression were applied to the Sharpe single index portfolio model (equations 81 to 94). For the purposes of comparison an efficient frontier was also generated using the sample means and covariance matrix. The mean vectors and covariance matrices for each of the efficient frontiers are included the appendix.

Three sets of efficient frontiers were generated for each of the four models, each with different sets of upper and lower bounds. The bounds used (0,1), i.e. the weight assigned to any share must be between 0% and 100%, (0, 0.25) and (-1, 1). The three sets of efficient frontiers are plotted below in Figures 4 to 6.

Figure 4

Sharpe Single Index Model (0 - 1 Bounds)

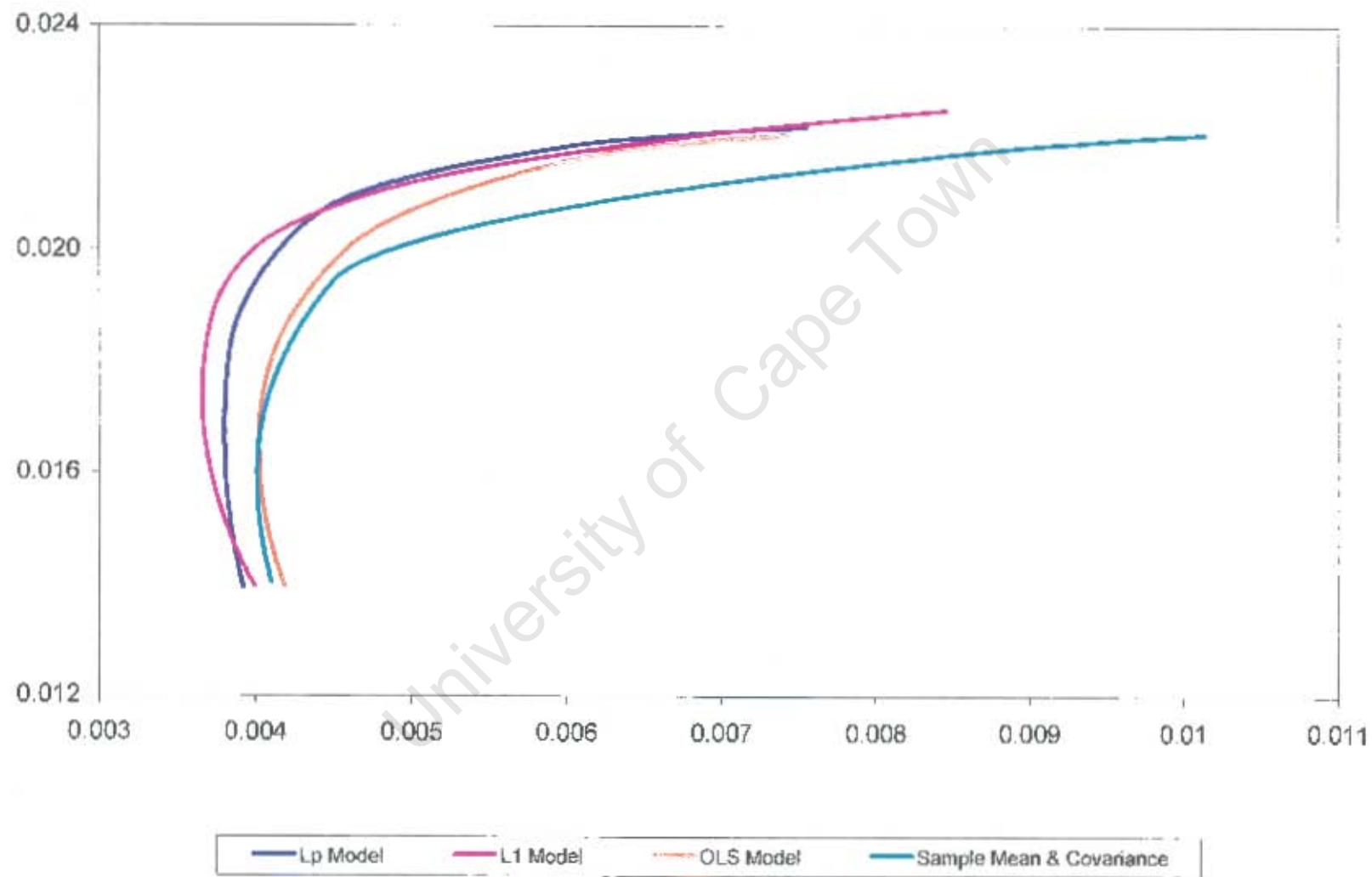


Figure 5 Sharpe Single Index Model (0 - 0.25 Bounds)

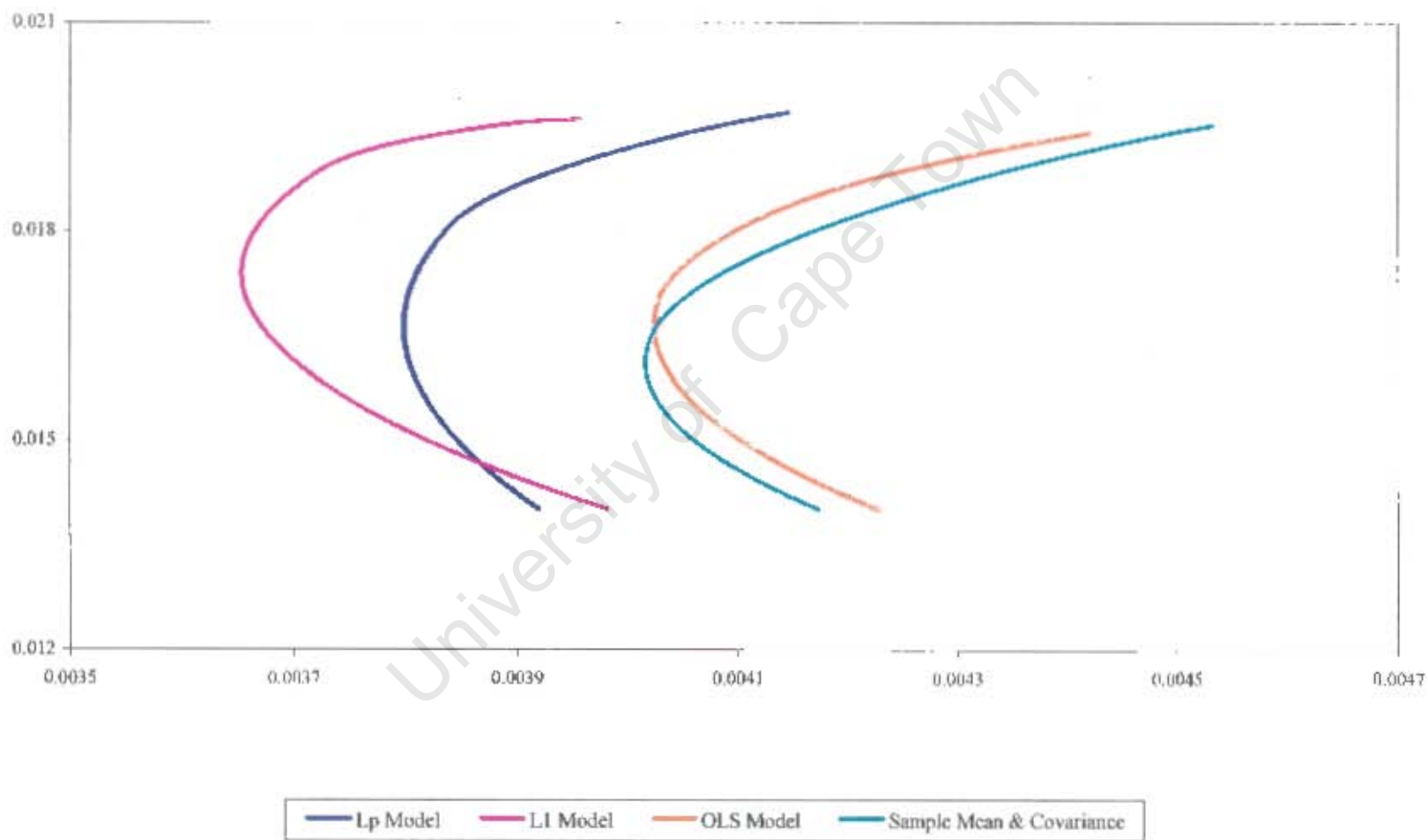
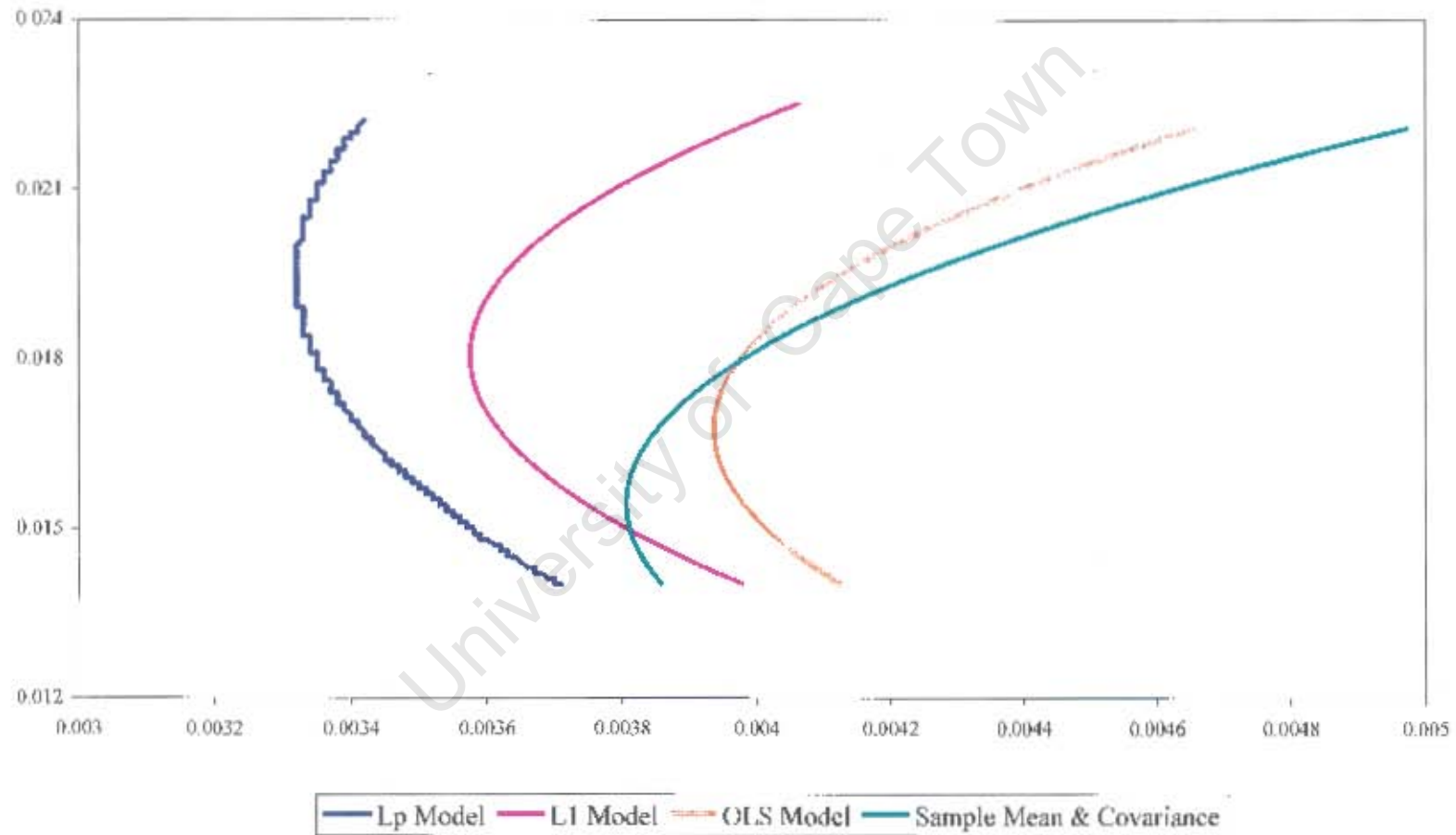


Figure 6

Sharpe Single Index Model (-1 - 1 Bounds)



All three index models outperformed the sample mean and covariance model over the entire efficient frontier and appears to justify the use of the index models in practice. The formulation with $(0, 1)$ bounds is given in Figure 1. From this plot it is clear that the index model models provide superior efficient frontiers and appears to justify the use of the index models in practice. There seems to be very little to separate the L_1 and L_p models for the $(0, 1)$ formulations.

It is important to consider a formulation with a reduced upper bound as many fund managers have restrictions on the maximum proportion of a portfolio they can invest in a single stock. Looking at the efficient frontiers generated with $(0, 0.25)$ bounds it is clear that the robust procedures again have the upper hand, with OLS performing only marginally better than sample mean and covariance model. The out performance of L_p estimation over L_1 estimation appears to have increased in this formulation but this primarily the result of the change in scale and there is still very little to choose between the two formulations. What is significant though, is that the tightening of restrictions in the model does not significantly alter the previous results.

The formulation with $(-1, 1)$ bounds, a relaxation of the initial assumptions once again produces very similar results. The two robust estimators again outperform the other two formulations.

It is thus not unreasonable to conclude that even though the robust procedures have not necessarily produced regression fits with superior R^2 figures, the reduced σ_e^2 appear to have played a significant part in generating superior efficient frontiers. The tightening or relaxation of bounds do not appear to influence the results in a material way.

12.2. Weighted OLS, L_1 and L_p vs non-weighted OLS, L_1 and L_p .

Regression Output.

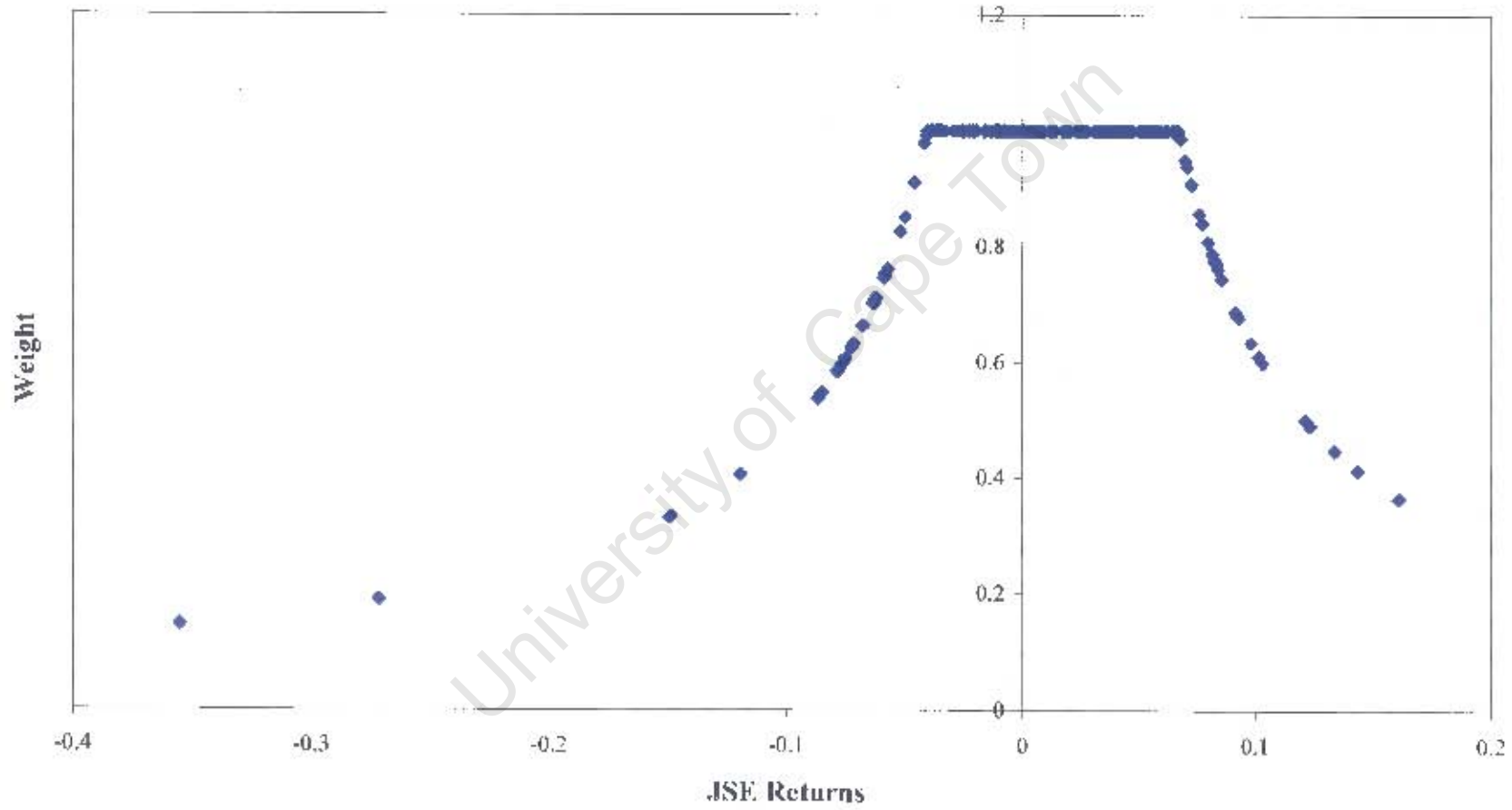
Weighted regression using Mallows type weights on the explanatory variables was performed using OLS, L_1 and L_p estimation. This was done using the techniques discussed in Part I. The weighted L_1 and L_p estimates are thus a two tier approach to robust regression estimation. A summary of the regression output is given below.

OLS	$\hat{\alpha}$	$\hat{\beta}$	$\hat{\sigma}_e^2$	R^2	$\beta_{\text{lower}} 95\%$	$\beta_{\text{upper}} 95\%$	t - stat	
Anglos	-0.0031	1.4681	0.0025	0.7466	1.28282	1.64241	13.9726	
De Beers	0.0001	1.1920	0.0038	0.5673	0.96411	1.41993	10.3363	
Gencor	0.0070	0.9837	0.0058	0.3836	0.70369	1.26362	6.9435	
Implats	0.0060	1.2056	0.0102	0.3388	0.83336	1.57792	6.4001	
Metlife	0.0127	0.8963	0.0038	0.3796	0.57964	1.17659	5.8141	
Remgro	0.0068	1.0471	0.0039	0.4843	0.81693	1.27728	8.9901	
SAB	0.0012	0.8464	0.0030	0.5275	0.71698	1.11926	9.0208	
Sappi	-0.0099	1.0302	0.0080	0.4309	0.76017	1.41926	6.5349	
Stanbic	0.0133	0.7215	0.0050	0.4675	0.5448	1.06738	6.0968	
L_1	$\hat{\alpha}$	$\hat{\beta}$	$\hat{\sigma}_e^2$	R^2	$\beta_{\text{lower}} 95\%$	$\beta_{\text{upper}} 95\%$	t - stat	
Anglos	-0.0049	1.3951	0.0034	0.6663	1.18007	1.6102	12.8197	
De Beers	-0.0011	1.1920	0.0074	0.4768	0.90018	1.53435	10.4164	
Gencor	0.0073	0.9837	0.0037	0.2977	0.80514	1.25386	12.0967	
Implats	0.0074	1.2056	0.0125	0.2325	0.72814	1.55354	5.4630	
Metlife	0.0152	0.8963	0.0074	0.2951	0.35586	1.08943	3.8936	
Remgro	0.0025	1.0471	0.0032	0.4405	0.81168	1.22601	9.7194	
SAB	0.0082	0.9181	0.0057	0.4592	0.5682	1.12459	6.0126	
Sappi	-0.0047	1.0897	0.0040	0.3308	0.79701	1.26339	8.7307	
Stanbic	0.0142	0.8061	0.0053	0.3430	0.45435	0.98856	5.3378	
L_p	\hat{p}	$\hat{\alpha}$	$\hat{\beta}$	$\hat{\sigma}_e^2$	R^2	$\beta_{\text{lower}} 95\%$	$\beta_{\text{upper}} 95\%$	t - stat
Anglos	1.2820	-0.0053	1.4389	0.0031	0.6647	1.23539	1.64241	13.9726
De Beers	1.6731	-0.0008	1.1887	0.0039	0.4794	0.95728	1.4202	10.1496
Gencor	1.1376	0.0078	1.0435	0.0071	0.2960	0.73308	1.35395	6.6430
Implats	1.9097	0.0055	1.2079	0.0100	0.2275	0.84018	1.57558	6.4619
Metlife	1.8658	0.0129	0.8781	0.0039	0.3135	0.57964	1.17659	5.8141
Remgro	1.6890	0.0057	1.0326	0.0035	0.4430	0.81572	1.24948	9.4093
SAB	1.6609	0.0025	0.8913	0.0032	0.4651	0.68384	1.09871	8.4910
Sappi	1.3138	-0.0066	1.0387	0.0052	0.3315	0.77201	1.30535	7.6975
Stanbic	1.5438	0.0134	0.8053	0.0045	0.3573	0.55874	1.05181	6.4552

The plot below gives the Mallows weights applied to the JSE All Share Index for the weighted procedures. Of particular interest are the large negative weights given to two months that had particularly low returns. In these months the markets fell by more than twenty percent.

Figure 7

Mallows weights assigned to JSE Returns



The $\hat{\sigma}_e^2$ values for the weighted regression are generally lower than that of the non-weighted equivalent. The R^2 values for the L_1 and L_p estimates are similar to those of the non-weighted estimates. The weighted OLS estimates however produced significantly higher R^2 values as well as lower $\hat{\sigma}_e^2$ and clearly provided an improved fit when compared to the non-weighted OLS.

Efficient Frontiers.

Efficient frontiers generated from the weighted regression estimates for each of the different bounded formulations discussed in the previous section are plot below in Figures 8 to 10. The efficient frontiers generated from the non-weighted regression are plotted on the same set of axis as the output from the weighted regression.

University of Cape Town

Figure 8

Sharpe Single Index Model (0 - 1 Bounds)

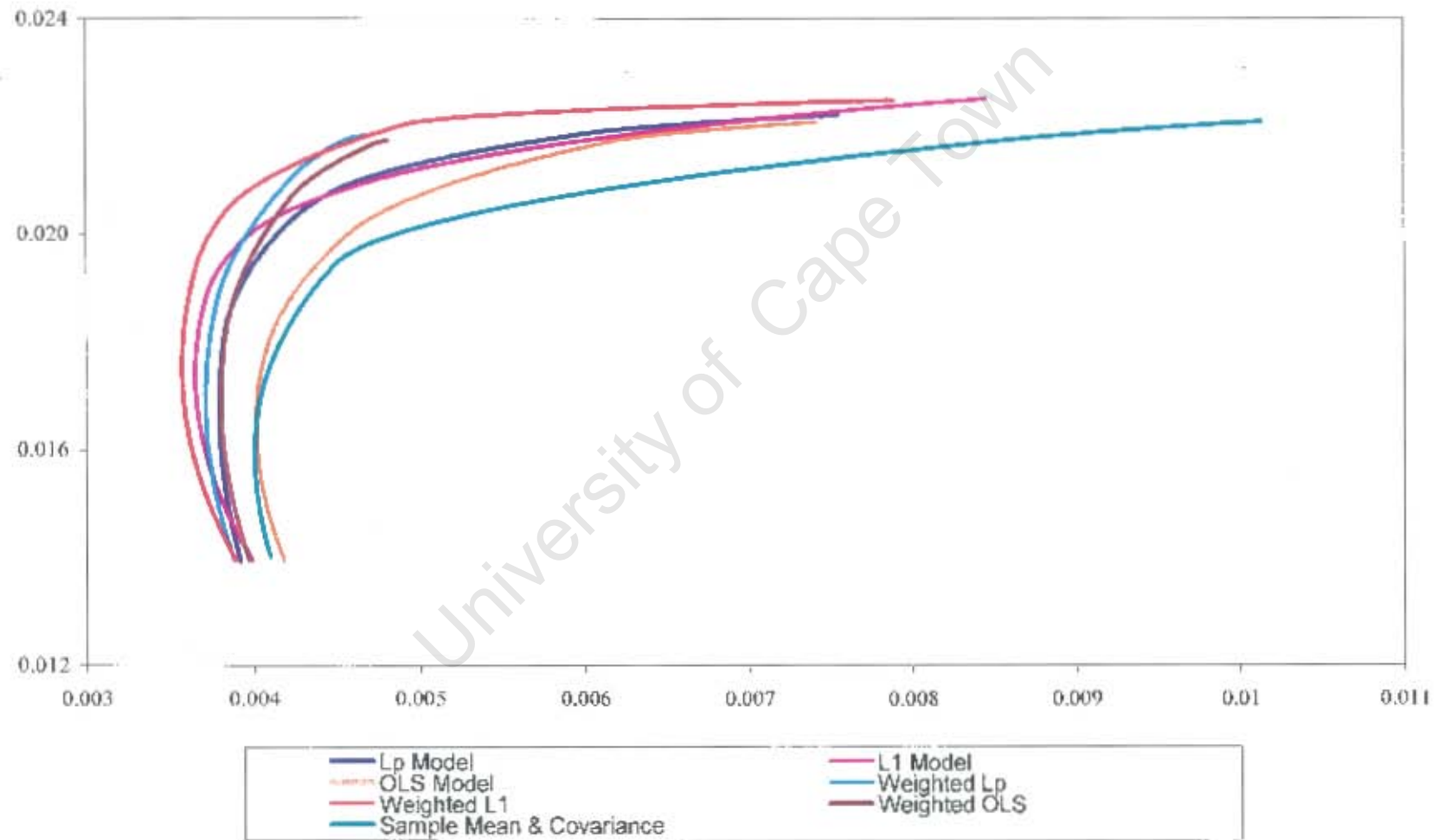


Figure 9

Sharpe Single Index Model (0 - 0.25 Bounds)

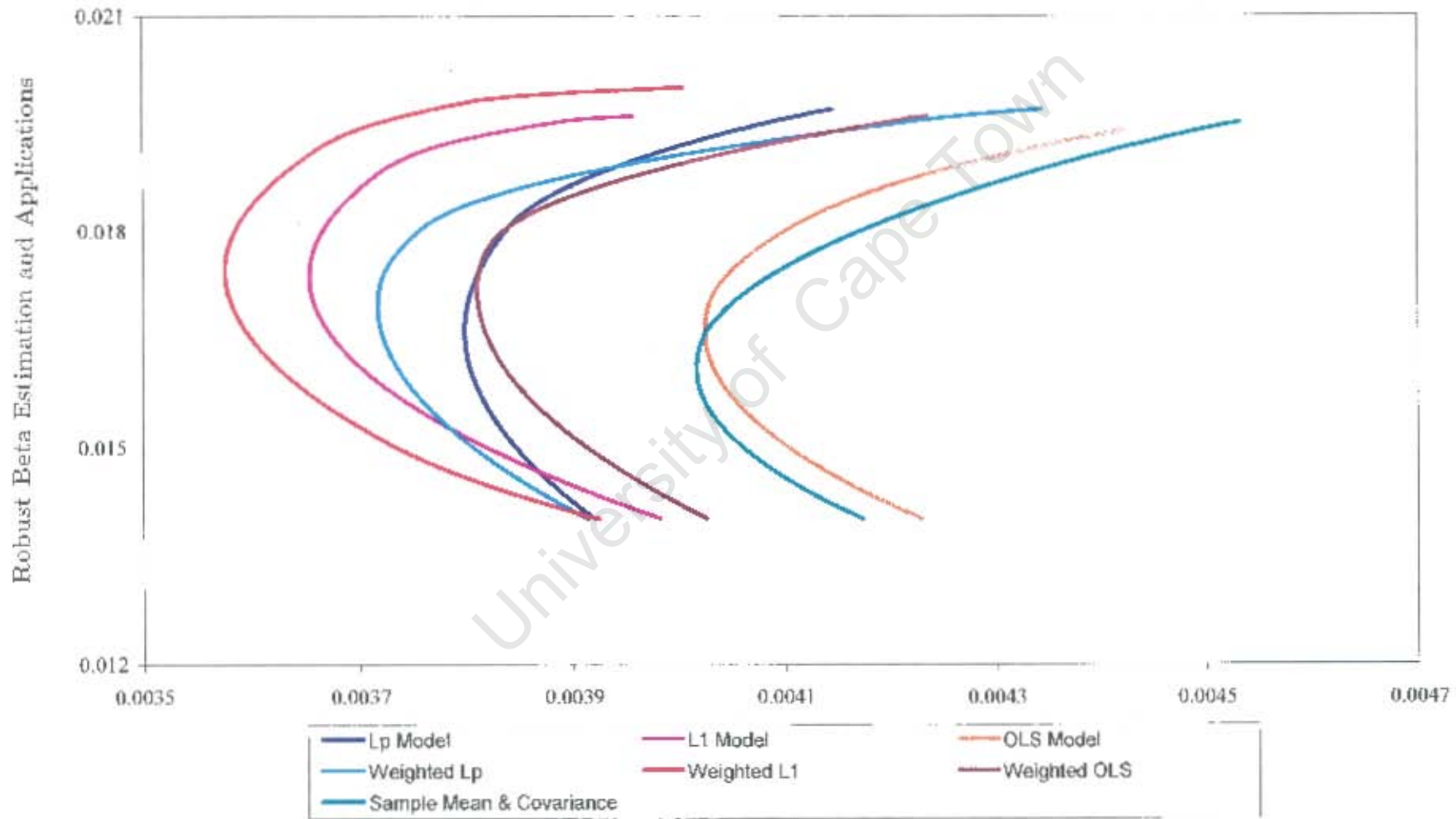
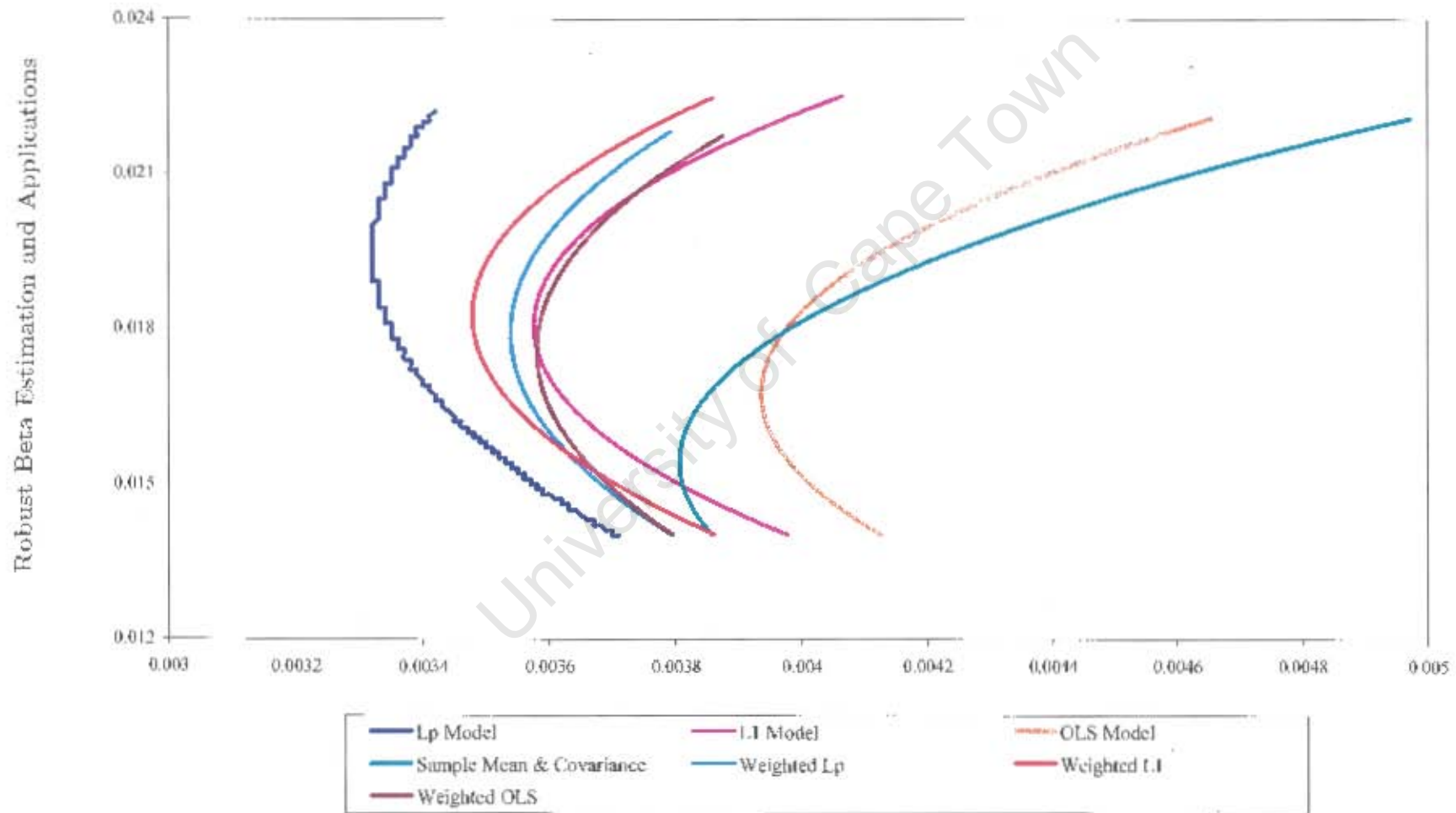


Figure 10

Sharpe Single Index Model (-1 - 1 Bounds)



From Figure 8 it is clear that the weighted estimates generated efficient frontiers that outperformed the non-weighted equivalents. The out performance of the weighted L_1 and L_p estimates over the non-weighted equivalents did not appear to be very large. Weighted OLS did however significantly outperform non-weighted OLS and reinforces the earlier suggestion that it provided a superior fit to the model. The plot with $(0, 0.25)$ bounds produced very similar results though the smaller scale does highlight the improvements that the weighted estimates make in the generation of efficient frontiers. The $(-1, 1)$ reinforces the results from the two previous plots.

12.3. The Sharpe Multiple Index Model.

Robust regression estimation was also applied to the Sharpe Multiple Index Model. The robust techniques used were L_1 and L_p regression. These were compared to the OLS multiple regression. The market proxies that were regressed the nine shares used in the analysis are listed below.

- JSE All Share Index
- JSE Banks Index
- JSE Gold Index
- JSE Insurance Index
- JSE Platinum Index

The regression output and the resulting efficient frontiers are given below.

Regression Output.

OLS	$\hat{\alpha}$	$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$	$\hat{\beta}_4$	$\hat{\beta}_5$	$\hat{\sigma}_e^2$	R^2
Anglos	0.0035	1.7027	0.1548	0.0739	-0.4854	0.0703	0.0063	0.7428
De Beers	0.0019	1.3290	0.4179	0.0029	-0.3551	0.0957	0.0086	0.5408
Gencor	0.0091	0.8593	-0.1362	0.0460	-0.3233	0.3497	0.0051	0.4721
Implats	0.0019	0.0894	0.0762	0.0739	-0.0593	0.9984	0.0091	0.7518
Metlife	0.0053	0.6345	-0.0749	-0.2124	0.6402	-0.0905	0.0009	0.4915
Remgro	0.0078	1.0577	-0.2160	-0.0551	-0.0212	-0.0043	0.0099	0.4532
SAB	0.0010	0.9709	-0.1434	-0.1070	0.1698	-0.1395	0.0028	0.5408
Sappi	-0.0066	1.9621	-0.1794	-0.2686	-0.5650	-0.0041	0.0038	0.3895
Stanbic	0.0063	1.1177	-0.2996	-0.2684	0.2666	-0.1462	0.0018	0.5163

L_1	$\hat{\alpha}$	$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$	$\hat{\beta}_4$	$\hat{\beta}_5$	$\hat{\sigma}_e^2$	R^2
Anglos	0.0024	1.6151	0.2032	0.0854	-0.4079	0.0424	0.0012	0.7299
De Beers	0.0094	1.2849	0.3338	0.0056	-0.4169	0.0827	0.0036	0.4856
Gencor	0.0111	0.6271	-0.1564	0.0307	-0.1633	0.3681	0.0032	0.4262
Implats	0.0024	0.0756	0.2229	0.0696	-0.0842	0.9752	0.0017	0.7386
Metlife	0.0026	0.4971	0.0135	-0.1881	0.7069	-0.0969	0.0014	0.4807
Remgro	0.0025	1.0706	-0.1714	-0.0807	0.0766	-0.0415	0.0060	0.4796
SAB	0.0043	1.0789	-0.0162	-0.1116	0.1067	-0.1662	0.0020	0.5507
Sappi	-0.0021	1.3140	-0.0098	-0.0589	-0.2766	-0.0398	0.0013	0.2699
Stanbic	0.0045	0.6814	-0.3706	-0.1805	0.4866	-0.0332	0.0036	0.4749

L_p	$\hat{\alpha}$	$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$	$\hat{\beta}_4$	$\hat{\beta}_5$	$\hat{\sigma}_e^2$	R^2
Anglos	0.0013	0.0021	1.6379	0.1656	0.0752	-0.4324	0.0013	0.7428
De Beers	0.0032	0.0028	1.3329	0.3691	0.0031	-0.3706	0.0032	0.5408
Gencor	0.0029	0.0118	0.7294	-0.1369	0.0443	-0.2043	0.0029	0.4721
Implats	0.0018	0.0019	0.0871	0.0871	0.0745	-0.0577	0.0018	0.7518
Metlife	0.0014	0.0052	0.6154	-0.0755	-0.2054	0.6482	0.0014	0.4915
Remgro	0.0061	0.0063	1.0508	-0.2087	-0.0723	0.0046	0.0061	0.4532
SAB	0.0021	0.0013	0.9739	-0.1052	-0.1060	0.1658	0.0021	0.5408
Sappi	0.0014	-0.0061	1.5298	-0.0402	-0.1444	-0.3611	0.0014	0.3895
Stanbic	0.0037	0.0049	0.9681	-0.3230	-0.2333	0.3405	0.0037	0.5163

In what is a similar pattern to the results of the single index model, the OLS estimates generally produce higher R^2 statistics but the L_1 and L_p estimates have generally produced lower $\hat{\sigma}_e^2$ values.

Efficient Frontiers.

The efficient frontiers for the Sharpe Multiple Index model were generated from the regression output in the previous section. As was done for the single index model, three sets of efficient frontiers were generated subject to different constraints. The constraints used were again (0, 1), (0, 0.25) and (-1, 1). The efficient frontiers generated from the three regression outputs as well as the efficient frontier generated by sample mean and variance are plotted on the same system of axis. The plots are given below.

Figure 11

Sharpe Multiple Index Model (0 - 1 Bounds)

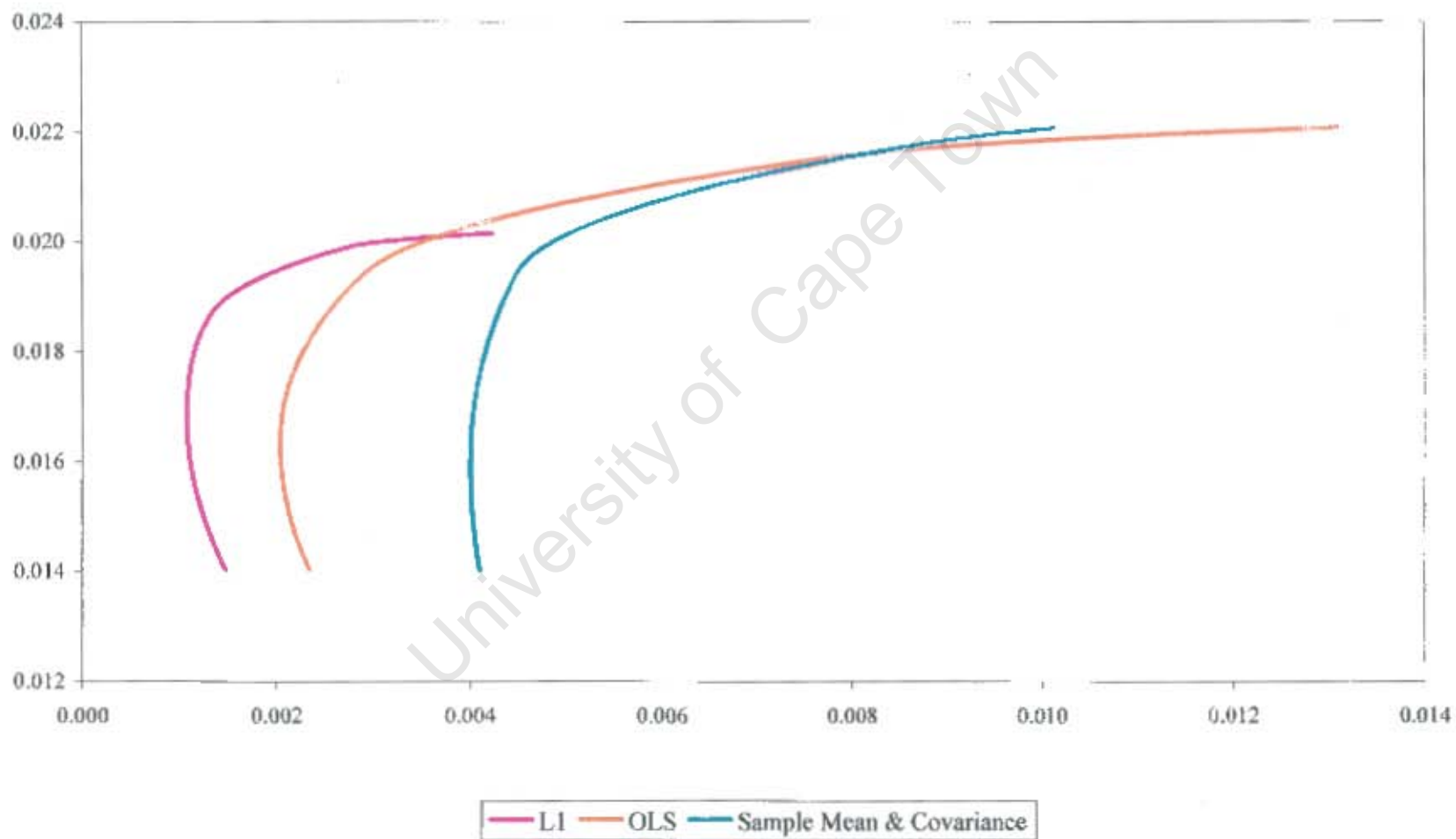


Figure 12

Sharpe Multiple Index (0 - 0.25 Bounds)

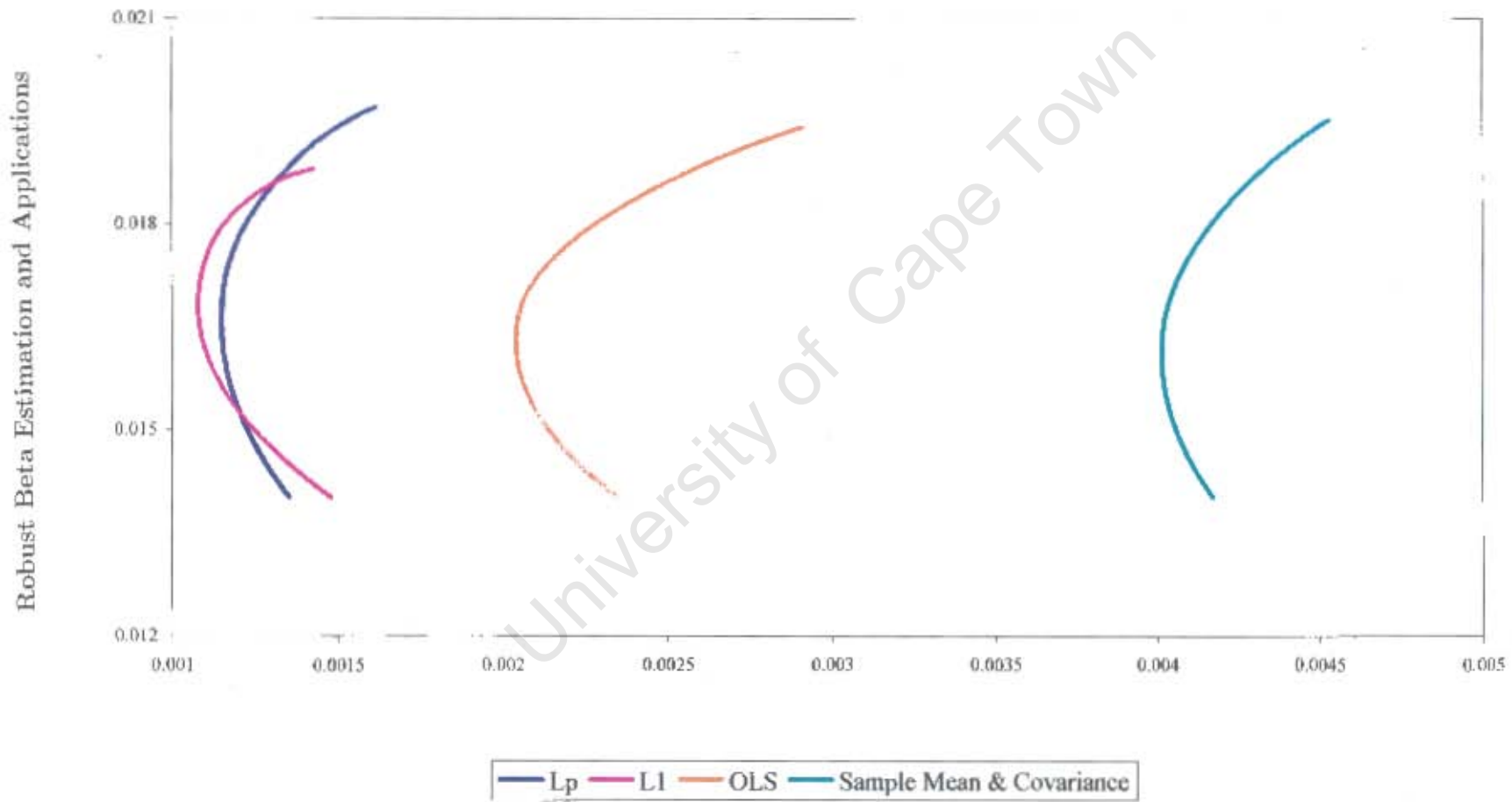
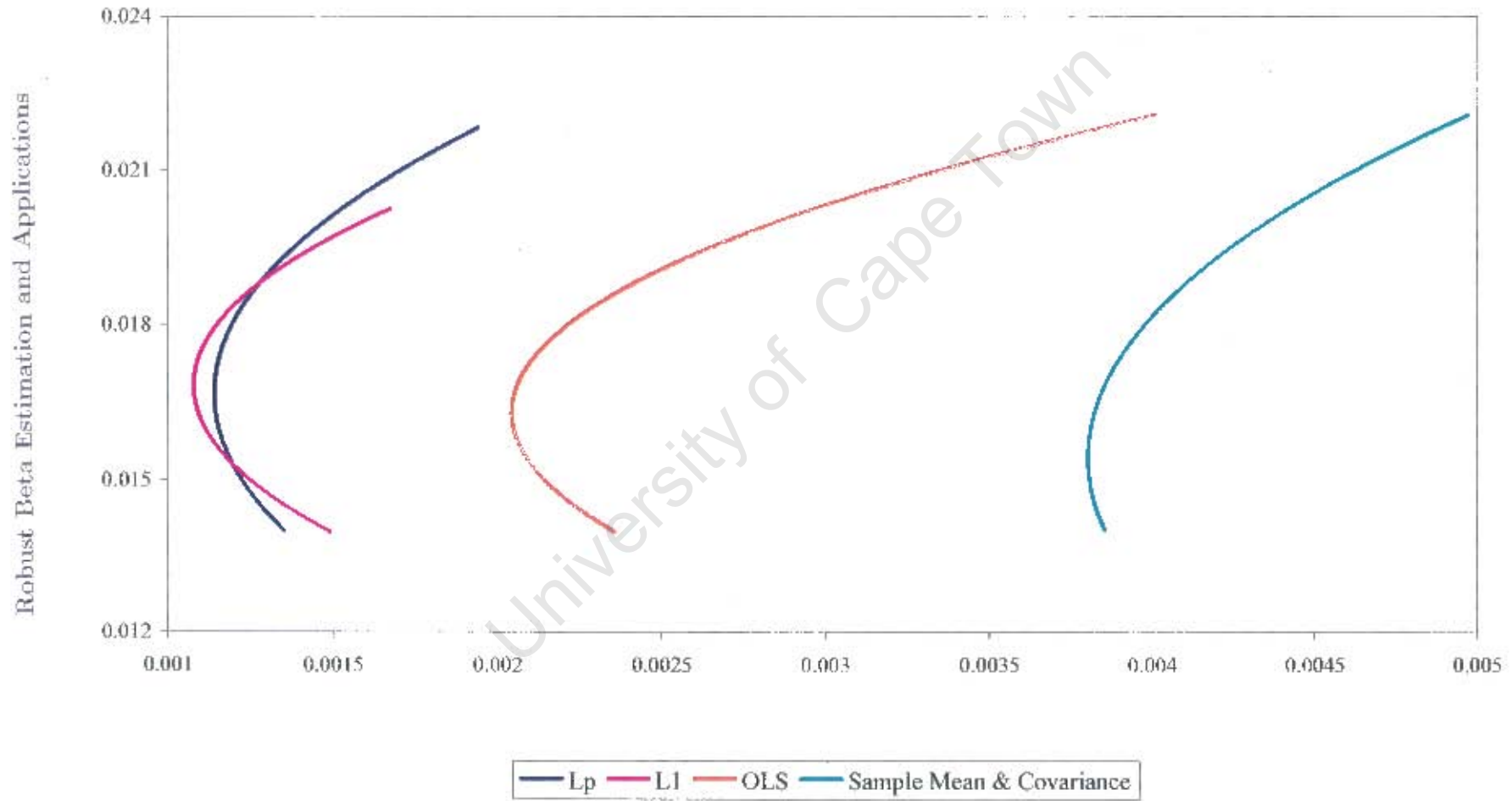


Figure 13

Sharpe Multiple Index (-1 - 1 Bounds)



All of the Multiple Index models outperform the sample mean and covariance model and again provided justification for using the index models instead of the simple sample estimates. From the three plots above it is clear that the robust procedures produce superior efficient frontiers. There appears to be little difference between the L_1 and L_p models. The tightening or relaxation of restrictions on the portfolio formulation does not materially influence the results. We can thus conclude that the robust estimation techniques considered here do aid in portfolio constructions.

University of Cape Town

13. CONCLUSIONS

The belief that stock returns do not follow the normal distribution, and tend to have longer tails, together with the results of the simulation study, in Part I, provides evidence of the inappropriateness of Ordinary Least Squares when estimating the beta coefficients of shares. The results of the simulation study shows that robust procedures are better suited to estimating parameters when the kurtosis is particularly large.

The analysis of the stocks used reinforced the idea that their returns do not follow a normal distribution. The robust beta estimates did generally not produce regressions fits with higher R^2 values but did reduce the residual variance. The robust beta estimates generated superior efficient frontiers. The reason for this is twofold, the first is the lower residual variance which reduces the variance of overall variance. The second reason can be found by comparing the values of the robust betas to the OLS betas, the robust betas are generally higher than that OLS betas because the robust procedures have effectively down weighted the influence of two large negative returns.

Whether down weighting of influential points is appropriate is for stock market returns is a cause of considerable debate amongst statisticians. The robust estimates may have produced superior efficient frontiers but the danger is that the overall risk of the portfolio may be under estimated.

References

- Baesel, J.B. (1974). On the assessment of risk: Some further considerations. *Journal of Science*, 29, 1491-1494.
- Barr, G.D.I. (1981). A contribution to adaptive robust estimation. Ph.D. thesis, University of Cape Town. 30
- Bickel, P.J. (1973). On some analogues to linear combinations of order statistics in the linear model. *Annals of Statistics*, 1, 595-616.
- Blume, M.E. (1971). On the assessment of risk, *Journal of Finance*, 26, 1-10.
- Chatterjee, S. and Hadi, A.S. (1986) Insufficient observations, high leverage points and outliers in linear regression. *Statistical Science*, 1, 379-416.
- Chatterjee, S. and Mächler, M. 1997. Robust Regression: A Weighted Least Squares Approach. Personal Communication.
- de Jongh, P.J. (1985). The regression trimmed mean: An empirical study. Unpublished Ph.D. thesis, University of Cape Town.
- de Jongh, P.J. and de Wet, T. (1986). Confidence intervals for regression parameters based on trimmed means. *SA Statistical Journal*, 20, 137-164.
- de Jongh, P.J. and de Wet, T. (1987). Comment on "The trimmed mean in the linear model" by A.H. Welsh. *Annals of Statistics*, 15, 36-38.
- de Jongh, P.J., de Wet, T. and Welsh, A.H. (1988). Mallows-type bounded influence regression trimmed means. *JASA*, Vol. 83, No. 403, 805-810.
- Denby, L. and Larsen, W.A. (1977). Robust regression estimators compared via Monte Carlo. *Communications Statistics, Theory and Methods*, 6, 335-363.
- Fama, E.F., Fisher, L, Jensen, M.C. and Roll, R. (1969). The adjustment of stock prices to new information. *International Economic Review*, 10, 1-21.
- Gonin, R. and Money, A.H. (1985b). Nonlinear Lp-norm estimation: Part II -

The asymptotic distribution of exponent, p , as a function of the sample kurtosis. *Communications Statistics, Theory and Methods*, 14, 841-849.

Gonin, R. and Money, A.H. (1989). *Nonlinear L_p -norm Estimation*. Marcel Dekker Inc., New York and Basel.

Gross, R. (1974). Risk analysis of ordinary shares quoted on the Johannesburg Stock Exchange. *The Investment Analysts Journal*, 4, 30-34. 31

Hampel, F.R. (1968). Contribution to the theory of robust estimation. Unpublished Ph.D. thesis, University of California Berkeley.

Hampel, F.R. (1974). The influence curve and its role in robust regression. *JASA*, 69, 383-394.

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

Hill, R.W. (1982). Robust regression when there are outliers in the carriers: The univariate case. *Communications Statistics, Theory and Methods*, 11, 849-869.

Huber, P.J. (1983). Minimax aspects of bounded influence regression. *JASA*, 78, 66-80.

Koenker, R. (1983). Personal communication.

Koenker, R.W. and Bassett, G.W. (1978). Regression quantiles. *Econometrica*, 46, 33-50.

Levy, R.A. (1971). On the short term stationarity of beta coefficients. *Financial Analyst Journal*, 27, 55-62.

Mallows, C.L. (1973). Influence functions. Unpublished paper presented at a conference on Robust Regression held at Cambridge, Mass. and sponsored by the National Bureau for Economic Research.

McDonald, J.B. and Newey, W.K. (1988). Partially adaptive estimation of regression models via the generalized t-distribution. *Economic Theory*, 4, 428-457.

McKean, J.W. and Schrader, R.M. (1987). Least absolute errors analysis of

- variance. In Y. Dodge. *Statistical Data Analyses*, 5, 381-389.
- Meinhold, R.J. and Singpurwalla, N.D. (1983). Understanding the Kalman filter. *The American Statistician*, 37, 123-127.
- Modigliani, F. and Poque, G.A. (1974). An introduction to risk and return: Concepts and evidence. *Financial Analyst Journal*, 30, 69-86.
- Nynquist, H. (1980). Recent studies in Lp-norm estimation. Ph.D. thesis, University of Unica, Sweden.
- Nynquist, H. (1983). The optimal Lp-norm estimator in linear regression models. *Communications Statistics, Theory and Methods*, 12, 2511-2524.
- Porter, M.A. and Winstanley, D.J. (1979). Remark AS R29. Remarks on AS110: Lp-norm fit of a straight line. *Applied Statistics*, 112-113.
- Rosenberg, B. and McKibben, W. (1973). The prediction of systematic and specific risk in common stocks. *Journal of Financial and Quantitative Analysis*, 8, 317-333.
- Ruppert, D. and Carroll, R.J. (1980). Trimmed least squares estimates in the linear model. *SASA*, 75, 828-838.
- Sharpe, W.F. (1964). Capital asset prices: A theory of market equilibrium under conditions of risk. *Journal of Finance*, 19, 425-442.
- Slossmacher, E.J. (1973). An iterative technique for absolute deviations curve fitting. *JASA*, 68, 857-859.
- Sposito, V.A., Kennedy, W.J. and Gentle, J.E. (1977). Lp-norm fit of a straight line. *Applied Statistics*, 26, 114-118.
- Thiart, C, 1994. Aspects of Estimation in the Linear Model with Special Reference to Collinearity. Unpublished Ph.D. Thesis. University of Cape Town.
- Vasicek, O. A. 1973. A Note on Using Cross-Sectional Information in Bayesian Estimation of Security Betas. *Journal of Finance*, Vol. 28, 1233-1239.
- Welsh, A. H. (1987). The trimmed mean in the linear model. *Annals of Statistics*, Vol. 15, No. 1, 20-36, 33

Appendix A

Data Set One					Data Set Two				
Y1	Y2	Y3	Y4	Y5	Y1	Y2	Y3	Y4	Y5
-10.4425	-10.2606	-9.4413	-8.7177	-11.8848	-10.4425	-10.2606	-9.4413	-8.7177	-11.8848
2.8663	1.9475	1.5545	27.4358	18.4401	2.8663	1.9475	1.5545	27.4358	18.4401
4.052	3.6897	2.3131	2.2425	3.2465	4.052	3.6897	2.3131	2.2425	3.2465
-1.726	-2.2668	-0.4489	-1.2483	9.0057	-1.726	-2.2668	-0.4489	-1.2483	9.0057
-4.2427	-4.3072	-3.9135	-5.4177	-3.155	-4.2427	-4.3072	-3.9135	-5.4177	-3.155
-9.423	-9.5797	-7.4684	-7.8966	10.3902	-9.423	-9.5797	-7.4684	-7.8966	10.3902
-1.7206	-0.7194	0.5432	-0.3935	-1.8607	-1.7206	-0.7194	0.5432	-0.3935	-1.8607
-0.8163	-0.3952	1.4005	-0.3293	-0.707	-0.8163	-0.3952	1.4005	-0.3293	-0.707
3.8661	4.5141	4.1087	3.9899	5.5603	3.8661	4.5141	4.1087	3.9899	5.5603
-6.1713	-6.2241	-7.515	-6.429	-0.6636	-6.1713	-6.2241	-7.515	-6.429	-0.6636
-0.3807	-0.9435	-3.7842	-1.8052	-1.3191	-0.3807	-0.9435	-3.7842	-1.8052	-1.3191
0.7443	-1.936	0.0816	1.4952	2.9943	0.7443	-1.936	0.0816	1.4952	2.9943
-4.0861	-5.242	-5.8114	-2.9406	-4.5659	-4.0861	-5.242	-5.8114	-2.9406	-4.5659
3.3979	1.9037	2.9097	-0.5953	1.6849	103.3979	101.9037	102.9097	99.4047	101.6849
-3.6102	-5.2148	-4.0165	-4.0812	-4.488	-3.6102	-5.2148	-4.0165	-4.0812	-4.488
10.103	10.7425	7.5069	10.2113	8.5787	10.103	10.7425	7.5069	10.2113	8.5787
4.4403	3.4213	5.1473	3.7795	0.774	4.4403	3.4213	5.1473	3.7795	0.774
7.5791	7.385	5.8284	7.0922	5.6438	7.5791	7.385	5.8284	7.0922	5.6438
-0.4604	-1.2084	-1.9737	-0.9594	-0.2911	-0.4604	-1.2084	-1.9737	-0.9594	-0.2911
-1.825	-3.6679	-2.3222	-3.315	-3.1383	-1.825	-3.6679	-2.3222	-3.315	-3.1383
-6.1727	-7.0724	-6.2462	-6.9626	-5.1687	-6.1727	-7.0724	-6.2462	-6.9626	-5.1687
9.3474	9.6297	8.7136	8.9716	7.3168	9.3474	9.6297	8.7136	8.9716	7.3168
4.195	3.9458	4.8553	2.3266	7.9989	4.195	3.9458	4.8553	2.3266	7.9989
3.3442	3.0341	1.5703	1.3966	4.0194	3.3442	3.0341	1.5703	1.3966	4.0194
-3.398	-4.9731	-6.107	-4.7929	-5.1065	-3.398	-4.9731	-6.107	-4.7929	-5.1065
0.4324	-0.1072	-0.1048	-0.1071	-2.5419	0.4324	-0.1072	-0.1048	-0.1071	-2.5419
-0.1835	1.8718	0.7496	1.7519	-15.5986	-0.1835	1.8718	0.7496	1.7519	-15.5986
-3.7106	-3.8522	-5.1275	-2.8114	-3.6216	-3.7106	-3.8522	-5.1275	-2.8114	-3.6216
-8.0643	-6.7974	-6.9546	-6.8774	-2.7761	-8.0643	-6.7974	-6.9546	-6.8774	-2.7761
-2.6031	-3.9239	-2.7986	-3.6383	-2.7708	-2.6031	-3.9239	-2.7986	-3.6383	-2.7708
-10.8643	-12.1145	-10.8148	-12.5744	-11.6357	-10.8643	-12.1145	-10.8148	-12.5744	-11.6357
6.834	9.5401	8.973	0.117	14.8919	6.834	9.5401	8.973	0.117	14.8919
4.7047	5.5636	4.3284	5.4676	17.7841	4.7047	5.5636	4.3284	5.4676	17.7841
13.468	13.1653	11.9149	10.9108	10.5634	13.468	13.1653	11.9149	10.9108	10.5634
3.0158	1.6574	0.4001	1.6406	5.7376	3.0158	1.6574	0.4001	1.6406	5.7376
-3.8504	-1.2658	-0.9296	5.4306	-4.494	-3.8504	-1.2658	-0.9296	5.4306	-4.494
-0.4354	-2.6167	-1.3798	-1.655	-6.147	-0.4354	-2.6167	-1.3798	-1.655	-6.147
-3.4819	-5.2204	-2.8303	-1.5287	-2.9041	-3.4819	-5.2204	-2.8303	-1.5287	-2.9041
4.0209	4.0901	5.8435	4.9984	4.9652	104.0209	104.0901	105.8435	104.9984	104.9652
7.4935	8.5976	9.0406	8.5729	5.2274	7.4935	8.5976	9.0406	8.5729	5.2274
-0.4951	-1.0891	0.5158	-1.9032	0.3891	-0.4951	-1.0891	0.5158	-1.9032	0.3891

Appendix B

University of Cape Town

Dependent Variable: Y1
 Method: Least Squares
 Date: 08/15/03 Time: 17:45
 Sample: 1 41
 Included observations: 41

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	0.119547	0.135570	0.881812	0.3836
X1	1.045258	0.040566	25.76685	0.0000
X2	1.021064	0.039431	25.89527	0.0000
X3	0.931443	0.048886	19.05341	0.0000
R-squared	0.977790	Mean dependent var		0.140022
Adjusted R-squared	0.975989	S.D. dependent var		5.596232
S.E. of regression	0.867158	Akaike info criterion		2.645277
Sum squared resid	27.82264	Schwarz criterion		2.812455
Log likelihood	-50.22818	F-statistic		542.9741
Durbin-Watson stat	1.883830	Prob(F-statistic)		0.000000

University of Cape Town

Dependent Variable: Y2
 Method: Least Squares
 Date: 08/15/03 Time: 17:45
 Sample: 1 41
 Included observations: 41

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	-0.177176	0.138359	-1.280549	0.2083
X1	1.065660	0.041401	25.74017	0.0000
X2	1.092615	0.040242	27.15125	0.0000
X3	1.030190	0.049892	20.64852	0.0000
R-squared	0.979315	Mean dependent var	-0.153637	
Adjusted R-squared	0.977638	S.D. dependent var	5.918225	
S.E. of regression	0.885000	Akaike info criterion	2.686010	
Sum squared resid	28.97933	Schwarz criterion	2.853188	
Log likelihood	-51.06320	F-statistic	583.9258	
Durbin-Watson stat	2.272161	Prob(F-statistic)	0.000000	

University of Cape Town

Dependent Variable: Y3
 Method: Least Squares
 Date: 08/15/03 Time: 17:45
 Sample: 1 41
 Included observations: 41

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	-0.071383	0.179667	-0.397305	0.6934
X1	0.952011	0.053761	17.70820	0.0000
X2	0.966895	0.052256	18.50298	0.0000
X3	0.999818	0.064787	15.43235	0.0000
R-squared	0.958906	Mean dependent var	-0.041206	
Adjusted R-squared	0.955574	S.D. dependent var	5.452391	
S.E. of regression	1.149221	Akaike info criterion	3.208514	
Sum squared resid	48.86625	Schwarz criterion	3.375692	
Log likelihood	-61.77454	F-statistic	287.7939	
Durbin-Watson stat	1.424469	Prob(F-statistic)	0.000000	

University of Cape Town

Dependent Variable: Y4
 Method: Least Squares
 Date: 08/15/03 Time: 17:45
 Sample: 1 41
 Included observations: 41

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	0.503133	0.695794	0.723106	0.4742
X1	0.927659	0.208200	4.455616	0.0001
X2	1.074080	0.202372	5.307452	0.0000
X3	0.880004	0.250901	3.507380	0.0012
R-squared	0.606631	Mean dependent var		0.508570
Adjusted R-squared	0.574736	S.D. dependent var		6.824758
S.E. of regression	4.450578	Akaike info criterion		5.916413
Sum squared resid	732.8830	Schwarz criterion		6.083591
Log likelihood	-117.2865	F-statistic		19.01975
Durbin-Watson stat	2.058265	Prob(F-statistic)		0.000000

University of Cape Town

Dependent Variable: Y5
 Method: Least Squares
 Date: 08/15/03 Time: 17:45
 Sample: 1 41
 Included observations: 41

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	1.234896	0.910662	1.356042	0.1833
X1	0.782203	0.272494	2.870533	0.0067
X2	1.074720	0.264866	4.057591	0.0002
X3	0.814962	0.328381	2.481758	0.0177
R-squared	0.438546	Mean dependent var		1.228627
Adjusted R-squared	0.393023	S.D. dependent var		7.476646
S.E. of regression	5.824959	Akaike info criterion		6.454649
Sum squared resid	1255.416	Schwarz criterion		6.621827
Log likelihood	-128.3203	F-statistic		9.633454
Durbin-Watson stat	1.939887	Prob(F-statistic)		0.000078

University of Cape Town

Dependent Variable: Y1
 Method: Least Squares
 Date: 08/15/03 Time: 17:45
 Sample: 1 41
 Included observations: 41

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	-0.292837	0.628729	-0.465761	0.6441
XO1	1.162537	0.189630	6.130557	0.0000
XO2	0.028633	0.038175	0.750046	0.4580
XO3	0.128438	0.039212	3.275435	0.0023
R-squared	0.542607	Mean dependent var		0.140022
Adjusted R-squared	0.505521	S.D. dependent var		5.596232
S.E. of regression	3.935225	Akaike info criterion		5.670281
Sum squared resid	572.9817	Schwarz criterion		5.837459
Log likelihood	-112.2408	F-statistic		14.63109
Durbin-Watson stat	1.914309	Prob(F-statistic)		0.000002

University of Cape Town

Dependent Variable: Y2
 Method: Least Squares
 Date: 08/15/03 Time: 17:45
 Sample: 1 41
 Included observations: 41

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	-0.586613	0.685691	-0.855507	0.3978
XO1	1.203637	0.206810	5.820005	0.0000
XO2	0.019148	0.041633	0.459915	0.6483
XO3	0.136629	0.042765	3.194873	0.0029
R-squared	0.513561	Mean dependent var	-0.153637	
Adjusted R-squared	0.474120	S.D. dependent var	5.918225	
S.E. of regression	4.291754	Akaike info criterion	5.843736	
Sum squared resid	681.5085	Schwarz criterion	6.010913	
Log likelihood	-115.7966	F-statistic	13.02100	
Durbin-Watson stat	1.924460	Prob(F-statistic)	0.000006	

University of Cape Town

Dependent Variable: Y3
 Method: Least Squares
 Date: 08/15/03 Time: 17:45
 Sample: 1 41
 Included observations: 41

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	-0.510033	0.608031	-0.838827	0.4070
XO1	1.108043	0.183387	6.042091	0.0000
XO2	0.025889	0.036918	0.701250	0.4875
XO3	0.145838	0.037922	3.845771	0.0005
R-squared	0.549357	Mean dependent var	-0.041206	
Adjusted R-squared	0.512819	S.D. dependent var	5.452391	
S.E. of regression	3.805680	Akaike info criterion	5.603334	
Sum squared resid	535.8784	Schwarz criterion	5.770512	
Log likelihood	-110.8684	F-statistic	15.03499	
Durbin-Watson stat	1.916534	Prob(F-statistic)	0.000001	

University of Cape Town

Dependent Variable: Y4
 Method: Least Squares
 Date: 08/15/03 Time: 17:45
 Sample: 1 41
 Included observations: 41

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	0.164042	0.954867	0.171795	0.8645
XO1	1.054478	0.287996	3.661433	0.0008
XO2	-0.004282	0.057977	-0.073860	0.9415
XO3	0.125317	0.059553	2.104300	0.0422

R-squared	0.290642	Mean dependent var	0.508570
Adjusted R-squared	0.233127	S.D. dependent var	6.824758
S.E. of regression	5.976530	Akaike info criterion	6.506025
Sum squared resid	1321.599	Schwarz criterion	6.673203
Log likelihood	-129.3735	F-statistic	5.053283
Durbin-Watson stat	1.972279	Prob(F-statistic)	0.004925

University of Cape Town

Dependent Variable: Y5
 Method: Least Squares
 Date: 08/15/03 Time: 17:45
 Sample: 1 41
 Included observations: 41

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	0.890087	1.131531	0.786622	0.4365
XO1	0.859444	0.341279	2.518302	0.0162
XO2	0.017484	0.068704	0.254488	0.8005
XO3	0.105622	0.070571	1.496676	0.1430
R-squared	0.170009	Mean dependent var		1.228627
Adjusted R-squared	0.102712	S.D. dependent var		7.476646
S.E. of regression	7.082275	Akaike info criterion		6.845535
Sum squared resid	1855.869	Schwarz criterion		7.012713
Log likelihood	-136.3335	F-statistic		2.526257
Durbin-Watson stat	1.674645	Prob(F-statistic)		0.072369

University of Cape Town

Dependent Variable: Y5
 Method: Least Squares
 Date: 08/15/03 Time: 18:17
 Sample: 1 41
 Included observations: 41

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	1.270901	0.914357	1.389940	0.1729
X1W	1.072633	0.528656	2.028979	0.0497
X2W	1.861097	0.436580	4.262898	0.0001
X3W	1.696898	0.620999	2.732529	0.0096
R-squared	0.443187	Mean dependent var		1.228627
Adjusted R-squared	0.398040	S.D. dependent var		7.476646
S.E. of regression	5.800836	Akaike info criterion		6.446349
Sum squared resid	1245.039	Schwarz criterion		6.613527
Log likelihood	-128.1502	F-statistic		9.816538
Durbin-Watson stat	2.165300	Prob(F-statistic)		0.000067

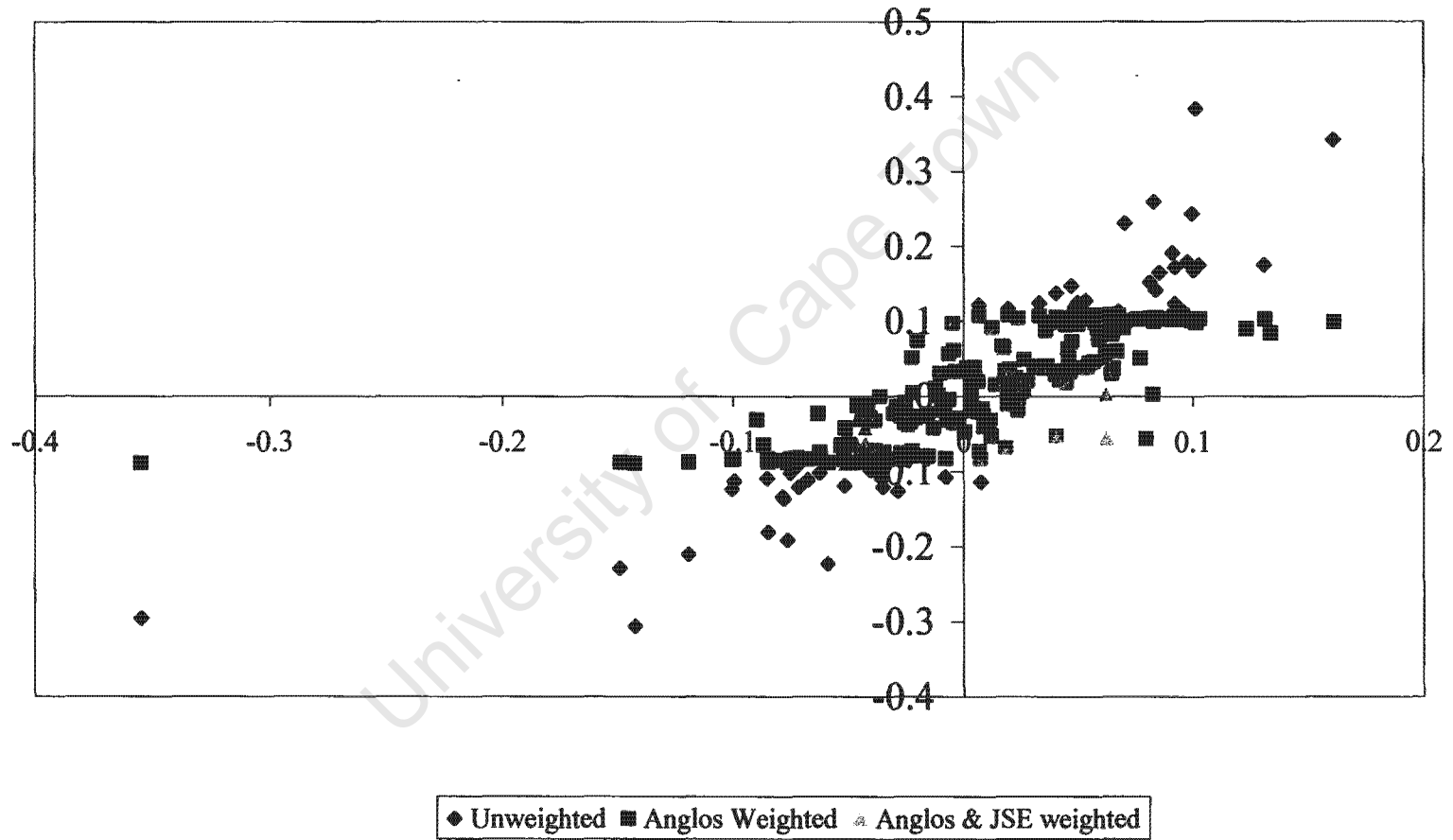
University of Cape Town

Dependent Variable: Y5D
 Method: Least Squares
 Date: 08/15/03 Time: 18:28
 Sample: 1 39
 Included observations: 39

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	1.335902	0.964043	1.385728	0.1746
X1D	0.778916	0.310493	2.508645	0.0169
X2D	1.115627	0.290136	3.845179	0.0005
X3D	0.805860	0.384419	2.096307	0.0434
R-squared	0.437732	Mean dependent var	1.121117	
Adjusted R-squared	0.389538	S.D. dependent var	7.645756	
S.E. of regression	5.973789	Akaike info criterion	6.509554	
Sum squared resid	1249.015	Schwarz criterion	6.680176	
Log likelihood	-122.9363	F-statistic	9.082633	
Durbin-Watson stat	1.960760	Prob(F-statistic)	0.000139	

University of Cape Town

Figure 14 Scatter Plot of Weighted and Unweighted Eeturns



Appendix C

The programs used to perform the regression analysis and to generate the efficient frontiers used in this thesis are listed below. All programs were written in Microsoft FORTRAN 90. Use of the Microsoft International Mathematics and Statistics Library was used wherever possible to aid the computation. The code is listed in the following sequence:

- Markowitz Sample and Mean Covariance model
- Sharpe Single Index Models using OLS, L_1 and L_p regression
- Sharpe Multiple Index Model using OLS regression
- Sharpe Multiple Index Model using L_1 and L_p regression

Markowitz Sample Mean and Covariance model.

```
!DECLARE THE USE OF THE INTERNATIONAL MATHEMATICS AND STATISTICS LIBRARY
USE MSIMSL
```

```
!DECLARE ALL THE VARIABLES THAT ARE TO BE USED IN THE PROGRAM
!ALL AARRAYS AND MATRICES ARE DYNAMICALLY ASSIGNED WITH
!SIZES TO BE ALLOCATED LATER
REAL, DIMENSION (:,:), ALLOCATABLE :: RAWDATA, LOG_RETURNS, &
    MNMX, CONSTR, HESSIAN, PHI, PROD, COV
REAL, DIMENSION (:), ALLOCATABLE :: EXPRET, RET_VEC, RHS, W, ALAMDA
INTEGER, DIMENSION (:), ALLOCATABLE :: IACT
DIMENSION INCD(1), SIGMA(501), EFFRET(501)
REAL MX, MN, JUMP, ADJUS, DIAG, SUMWT
INTEGER NACT, NR, NVAR, NRMIS
```

```
! NUMBER OF ROWS IN THE DATA SET
NR=150
```

```
! NUMBER OF EXPLANATORY VARIABLES IN THE DATA SET
NVAR=9
```

```
! ASSIGN DIMENSIONS TO DYNAMIC ARRAYS AND MATRICES
ALLOCATE(RAWDATA(NR, NVAR), LOG_RETURNS(NR-1, NVAR), COV(NVAR, NVAR), &
```

```

          PROD(NR-NVAR, NVAR))
    ALLOCATE(PHI(NVAR, NVAR), CONSTR((2+(NVAR*2)), NVAR))
    ALLOCATE(HESSIAN(NVAR, NVAR), MNMX(15, 1))
    ALLOCATE(RHS(2+(NVAR*2)), W(NVAR), EXPRET(NVAR))
    ALLOCATE(ALAMDA(NVAR), RET_VEC(NVAR))
    ALLOCATE(IACT(NVAR))

! OPEN THE DATA FILE
OPEN(4, FILE='C:\Data.txt')

! READ IN THE VALUES OF THE DATA FILE
DO I=1, NR
  READ(4, *) (RAWDATA(I, J), J=1, NVAR)
ENDDO

! COMPUTE THE LOG RETURNS OF THE DATA
DO I=1, NR-1
  DO J=1, NVAR
    LOG_RETURNS(I, J) = (LOG(RAWDATA(I+1, J)) - LOG(RAWDATA(I, J)))
  ENDDO
ENDDO

! CALCULATE THE THE COVARINCE MATRIX BETWEEN SHARES
CALL CORVC(0, NR-1, NVAR, LOG_RETURNS, NR-1, 0, 0, 0, 0, &
  EXPRET, PHI, NVAR, INCD, 1, NOBS, NMISS, SUMWT)

! CONVERT THE COVARIANCE MATRIX TO THE FORMAT REQUIRED FOR
! THE QUADTRATIC OPTIMISER
DO I=1, NVAR
  DO J=1, NVAR
    HESSIAN(I, J) = 2*PHI(I, J)
  ENDDO
ENDDO

! READ IN VALUES FOR THE CONSTRIANTS
DO I=1, ((NVAR*2)+2)
  DO J=1, NVAR
    CONSTR(I, J) = 0
  ENDDO
ENDDO

```

```

ENDDO

DO I=1,NVAR
  CONSTR(1,I)=EXPRET(I)
  CONSTR(2,I)=1
  CONSTR(I+2,I)=1
  CONSTR(I+2+NVAR,I)=-1
  RHS(I+2)=-1
  RHS(I+2+NVAR)=-1
  RET_VEC(I)=0
ENDDO

RHS(2)=1

ADJUS=12
!CALCULATE MIN AND MAX RETURNS
CALL UVSTA(0,NVAR,1,EXPRET,NVAR,0,0,0,95.0,95.0,0, &
           MNMX,15,NRMISS)

MN=MNMX(6,1)
MX=MNMX(7,1)

!CALCULATE THE RETURN ITERATION SIZE FOR THE GENERATION
!OF THE EFFICIENT FRONTIER
JUMP=(MX-MN-0.0002)/500
!LOOP OF THE QUADRATIC OPTIMISATION ROUTINE FOR THE
!GENERATION OF THE EFFICIENT FRONTIER
DO K=0,500
  RHS(1)=(JUMP*K)+MN+0.0001
  CALL QPROG(NVAR,(NVAR*2)+2,2,CONSTR,(NVAR*2)+2,RHS, &
            RET_VEC,HESSIAN,NVAR,DIAG,W,NACT,IACT,ALAMDA)
  SIGMA(K+1)=BLINF(NVAR,NVAR,PHI,NVAR,W,W)
  EFFRET(K+1)=(JUMP*K)+MN+0.0001
ENDDO

!WRITE EFFICIENT FRONTIER DATA TO AN OUTPUT FILE
OPEN(7,FILE='C:\MSc\Output Files\VarOutput.xls')

DO I=1,501
  WRITE(7,*)SIGMA(I),',',',',EFFRET(I)

```

```

ENDDO

!WRITE COVARIANCE MATRIX TO AN OUTPUT FILE
OPEN(8,FILE='C:\MSc\Output Files\CovOutput.xls')

DO I=1,NVAR
  WRITE(8,*)(PHI(I,J),J=1,NVAR)
ENDDO

!WRITE EXPECTED RETURNS TO AN OUTPUT FILE
OPEN(9,FILE='C:\MSc\Output Files\ExpRetOutput.xls')

DO I=1,NVAR
  WRITE(9,*)EXPRET(I)
ENDDO
END

```

Sharpe Single Index Models using OLS, L_1 and L_p regression. USE

```

MSIMSLSD
IMPLICIT REAL*8(A-H,O-Z)
PARAMETER(NR=2000,NC=10)
parameter (max=5000)
PARAMETER (NIND=1,NCOEF=2)
REAL*8 Y(NR),X(NR,NC),R(NR),E1(NR)
REAL*8 X1(NR,NC),COVRES(NC,NC)
REAL*8 RESMAT(NR,NC),BETMAT(NC,NC)
REAL*8 RESL1(NR,NC),RESLP(NR,NC)
REAL*8 RMATL1(NC,NC),RMATLP(NC,NC)
REAL*8 PSIMAT(NC,NC),DIAMAT(NC,NC)
REAL*8 XR(NR),YR(NR),DF,PR
REAL*8 BP(NC,1),WV(NR),A(NC,NC),CKB(NC),RES(NR)
REAL*8 AR,BR,ABDEV,SSDEV
REAL*8 WWV(NR),WYV(NR)
REAL*8 COVB(NC,NC),UPPER(NC),LOWER(NC)
REAL*8 SE(NC),TI(NC),SKEW,CURT,Z1,Z2,CHI
REAL*8 XBAR(NC),SIGMA(NC,NC),SIGRES(NC,NC)
REAL*8 XPORT(NR,NC),BETLS(NC),BETL1(NC),BETLP(NC)
REAL*8 ROBL(NC),ROBL1(NC),ROBLP(NC)
REAL*8 RVARLP(NC),RVARL1(NC),RVARLS(NC)
REAL*8 RSCALP(NC),RSCAL1(NC),RSCALS(NC)

```

```

REAL*8 VINDEX
character*9 SHARE(MAX),INDEX,SHA
CHARACTER*10 FSHA,FINDEX,FSHARE(MAX)
CHARACTER*9 FNAME
CHARACTER*9 INFILE
CHARACTER*8 NDATE
character*13 outfile,NEWFILE,PRIFILE,RISKFILE
integer*4 knum,NSHA
integer*4 date(max),MDATE(MAX),DINDX(MAX),nda(max)
integer*4 last(max),MLAST(MAX)
integer*4 LINDX(MAX),MINDX(MAX),nla(max)
integer*4 hi(max),MHI(MAX),HINDX(MAX),nhi(max)
integer*4 lo(max),MLO(MAX),LONDX(MAX),nlo(max)
integer*4 vo(max),MVO(MAX),VINDX(MAX),nvo(max)
integer*4 LTR(max)
integer*4 MRAW(MAX,11)
REAL DLAST(MAX),DHI(MAX),DLO(MAX),DVO(MAX),DLTR(MAX)
INTEGER IDATE
INTEGER JDATE
INTEGER*4 LCLO,IDUM
! WE NOW DECLARE DIMENSIONS OF DRLLP
INTEGER INTCEP,LDR,LDX,NCOEF,NIND
INTEGER IFRQ,IIND,INDIND(NIND),IRANK,IRSP,ITER,IWT,
& MAXIT,NOU,NRMISS
REAL*8 B(NCOEF),DFE,E(NR),ELP,EPS,P,
& RCOV(NC,NC),SCALE2,TOL
character*9 NSHARE,NINDEX
! EXTERNAL DMACH,DRLLP
! DATA TOLER/1.E-10/
! I20=19000000
10 write(*,*)
WRITE(*,1240)
1240 FORMAT( /,' ENTER NAME OF INDEX FOR BETAS',/)
READ(*,160) INDEX
160 FORMAT(A9)
C NEWFILE='STOCK.UIT'
C OPEN(9,FILE=NEWFILE,STATUS='UNKNOWN',CARRIAGECONTROL='LIST')
CALL FSTOCK(INDEX,FINDEX)
OPEN(12,FILE=FINDEX,STATUS='OLD',CARRIAGECONTROL='LIST')
I=0

```

```

DO WHILE(.NOT.EOF(12))
  I=I+1
  READ(12,*) NINDEX,DINDEX(I),LINDEX(I),HINDEX(I),LONINDEX(I),
  &VINDEX(I),LCLO, IDUM
ENDDO
NINDEX=I
! DO II=1,NINDEX-1
! E1(II)=LOG(DFLOAT(LINDEX(II+1)))-LOG(DFLOAT(LINDEX(II)))
! ENDDO
! NI=NINDEX-1
! CALL MOMENT(E1,NI,AVE,ADEV,SDEV,VAR,AM4,SKEW,CURT)
! WRITE(9,109) INDEX
! 109 FORMAT(16H STATISTICS FOR ,A11)
! CALL SKECUR(NI,SKEW,CURT,Z1,Z2,CHI)
! WRITE(9,110)NI,AVE,VAR,SDEV,SKEW,CURT
! 110 FORMAT(5H N = ,I5,7H AVE = ,F8.4,7H VAR = ,F8.4,
! &8H SDEV = ,F8.4,8H SKEW = ,F8.4,8H CURT = ,F8.4,/)
! WRITE(9,1091) Z1,Z2,CHI
! 1091 FORMAT(5H Z1= ,F8.4,5H Z2= ,F8.4,7H CHI = ,F8.4,/)
WRITE(*,1140)
1140 FORMAT(' ENTER FILE OF SHARES FOR PORTFOLIO=SHARE.DAT',/)
READ(*,5) FNAME
5 FORMAT(A9)
INFILE=FNAME
WRITE(*,*) INFILE
PAUSE
CLOSE(12)
OPEN (10,FILE=INFILE,STATUS='OLD',CARRIAGECONTROL='LIST')
I=0
DO WHILE(.NOT.EDF(10))
  I=I+1
  READ(10,*) SHARE(I)
  CALL FSTOCK(SHARE(I),FSHARE(I))
  WRITE(*,2213) SHARE(I),FSHARE(I)
  2213 FORMAT(1X,A9,A10)
ENDDO
PAUSE
NSHA=I
OPEN(24,FILE='STOCK.INP',STATUS='UNKNOWN',
+ CARRIAGECONTROL='LIST')

```

```

OPEN(20,FILE='BRAD.INP',STATUS='UNKNOWN',CARRIAGECONTROL='LIST')
OPEN(21,FILE='BRADLP.INP',STATUS='UNKNOWN',
+ CARRIAGECONTROL='LIST')
OPEN(22,FILE='BRADL1.INP',STATUS='UNKNOWN',
+ CARRIAGECONTROL='LIST')
OPEN(23,FILE='BRADLS.INP',STATUS='UNKNOWN',
+ CARRIAGECONTROL='LIST')
OPEN(31,FILE='BSCADLP.INP',STATUS='UNKNOWN',
+ CARRIAGECONTROL='LIST')
OPEN(32,FILE='BSCAL1.INP',STATUS='UNKNOWN',
+ CARRIAGECONTROL='LIST')
OPEN(33,FILE='BSCALS.INP',STATUS='UNKNOWN',
+ CARRIAGECONTROL='LIST')
PAUSE
WRITE(*,*) 'EARLIEST DATE FOR JSE-OVER 19850327'
WRITE(*,*) 'ENTER EARLIEST DATE-YYYYMMDD-FIRST=0'
READ(*,161) IDATE
161 FORMAT(I8)
! IDATE=I20+IDATE
WRITE(*,*) 'ENTER LATEST DATE-YYYYMMDD '
READ(*,161) JDATE
! JDATE=I20+JDATE
! WRITE(*,*) 'ENTER NAME OF SHARE'
! READ(*,1660) SHA
!1660 FORMAT(A9)
! WRITE(*,175) SHA
! PAUSE
! 175 FORMAT(1X,A9)
WRITE(*,*) 'WE NOW DO A LOOP OF NSHA'
NAVE=0
AVEBET=0.0
AVESTD=0.0
BAYES=.52167
SBAYES=.23017
DO 9898 KK=1,NSHA
! CALL FSTOCK(SHA,FSHA)
OPEN(11,FILE=FSHA(KK),STATUS='OLD',CARRIAGECONTROL='LIST')
J=0
DO WHILE(.NOT.EOF(11))
J=J+1

```

```

read(11,*) NSHARE,
2 date(j),dlast(j),dhi(j),dlo(j),dvo(j),dltr(j),IDUM
last(j)=ifix(dlast(j))
! WRITE(*,*) DATE(J),LAST(J),DLAST(J)
hi(j)=ifix(dhi(j))
lo(j)=ifix(dlo(j))
vo(j)=ifix(dvo(j))
ltr(j)=ifix(dltr(j))
ENDDO
KNUM=J
J=0
DO 150 I=1,KNUM
IF((DATE(I).GE.IDATE).AND.(DATE(I).LE.JDATE)) THEN
J=J+1
nda(j)=date(i)
nla(j)=last(i)
C WRITE(*,1551) NDA(J),NLA(J)
C PAUSE
nhi(j)=hi(i)
nlo(j)=lo(i)
nvo(j)=vo(i)
ELSE
ENDIF
150 continue
1551 FORMAT(2I9)
nin=j
C 150 CONTINUE
101 format(1x,a11,5i9)
CALL MONTH(MAX,NIN,nda,nla,nhi,nlo,nvo,
+ NOUT,MDATE,MLAST,MHI,MLO,MVO)
NIN=NOUT
KUMVO=0
DO 1605 I=1,NOUT
DATE(I)=MDATE(I)
LAST(I)= MLAST(I)
HI(I)=MHI(I)
LO(I)= MLO(I)
VO(I)=MVO(I)
KUMVO=KUMVO+VO(I)
1605 CONTINUE

```

```
IF(KUMVO.EQ.0) THEN
DO 16621 I=1,NOUT
nda(I)=DATE(I)
nla(I)= LAST(I)
nhi(I)=HI(I)
nlo(I)= LO(I)
nvo(I)=VO(I)
16621 CONTINUE
nin=nout
call pritra(max,nin,nda,nla,nhi,nlo,nvo,
&nout,mdate,mlast,mhi,mlo,mvo)
GOTO 16601
ELSE
ENDIF
CALL TRATRA(MAX,NIN,DATE, LAST, HI, LO, VO,
+ NOUT, MDATE, MLAST, MHI, MLO, MVO)
16601 CONTINUE
C DO 1601 I=1,NOUT
C WRITE(*,*) NSHARE,MDATE(I),MLAST(I),MHI(I),MLO(I),MVO(I)
C 1601 CONTINUE
C pause
J=0
DO 1800 I=1,NOUT
DO 1805 J=1,NINDX
IF(DINDX(J).EQ.MDATE(I)) THEN
MINDX(I)=LINDX(J)
GOTO 1800
ELSE
ENDIF
1805 CONTINUE
1800 CONTINUE
1810 CONTINUE
N=NOUT
PRIFILE=NSHARE//'.PRI'
RISKFILE=NSHARE//'.RIS'
OPEN(7,FILE=PRIFILE,STATUS='UNKNOWN',CARRIAGECONTROL='LIST')
OPEN(9,FILE=RISKFILE,STATUS='UNKNOWN',CARRIAGECONTROL='LIST')
C IF(KK.EQ.1) THEN
C DO 3219 I=1,N
C MRAW(I,1) = MDATE(I)
```

```

C MRAW(I,2) = MINDX(I)
C write(*,3218) mraw(i,1),mraw(i,2)
C 3219 CONTINUE
C 3218 format(2i7)
C else
C endif
DO I=1,N
WRITE(7,111) NSHARE,MDATE(I),MLAST(I),MINDX(I)
MRAW(I,1)=MDATE(I)
MRAW(I,2)= MINDX(I)
MRAW(I,KK+2)=MLAST(I)
ENDDO
111 format(A11,3i9)
IF(N.LE.24) THEN
GOTO 9898
ELSE
ENDIF
! WRITE(*,*) N
! PAUSE
DO JT=1,N
X(JT,1)=1.
X1(JT,1)=1.
ENDDO
DO 2 I=1,N
Y(I)=DFLOAT(MLAST(I))
X(I,2)=DFLOAT(MINDX(I))
2 CONTINUE
! PAUSE
DO 1 I=1,N-1
Y(I)=LOG(Y(I+1)/Y(I))
YR(I)=Y(I)
X1(I,2)=Y(I)
X(I,2)=LOG(X(I+1,2)/(X(I,2)))
E1(I)=X(I,2)
XR(I)=X(I,2)
1 CONTINUE
N=N-1
CALL MOMENT(E1,N,AVE,ADEV,SDEV,VAR,AM4,SKEW,CURT)
VINDEXT=VAR
CALL SKECUR(N,SKEW,CURT,Z1,Z2,CHI)

```

```

WRITE(9,2233) INDEX,N,AVE,VAR,SDEV,SKEW,CURT,Z1,Z2,CHI
2233 FORMAT(A9,I5,3F8.5,/,5F10.5)
NP=2
C NOW CALL MALLOWS FOR WEIGHT FUNCTION WW
Q=0.15D0
CALL SWALLOW(X,N,NP,NR,NC,WWV,Q)
CALL SWALLOW(X1,N,NP,NR,NC,WV,Q)
! WRITE(9,15)(WWV(I),I=1,N)
! 15 FORMAT(' MALLOWS WEIGHTS ',//,3(7F10.3,/) )
DO 877 II=1,N
C WV(II)=WWV(II)
WV(II)=1.0D0
877 CONTINUE
IWT=0
C
C N=N-1
! do 3 i=1,n
! write(*,4213) I,y(i),x(i,1),x(i,2)
! 3 continue
! 4213 FORMAT(I6,3F8.4)
C
CALL MOMENT(Y,N,AVE,ADEV,SDEV,VAR,AM4,SKEW,CURT)
VARY=VAR
if(var.eq.0) then
goto 9898
else
endif
FMOVE=FLOAT(MLAST(N+1))-FLOAT(MLAST(1))
FMOVE=FMOVE/FLOAT(MLAST(1))*100
CALL SKECUR(N,SKEW,CURT,Z1,Z2,CHI)
WRITE(9,2222) NSHARE,MDATE(1),MDATE(N+1),
& MLAST(1),MLAST(N+1),FMOVE
2222 FORMAT(A11,2I9,2I7,F10.2)
WRITE(9,2223) N,AVE,VAR,SDEV,SKEW,CURT,Z1,Z2,CHI
2223 FORMAT(I5,3F8.5,/,5F10.5)
C NOTE NP=NUMBER OF INDICES + CONSTANT TERM(ALFA)
C CALL LEAST SQUARES ESTIMATE
C SUBROUTINE REGRES(Y,X,C,A,WV,N,NR,NP,RES)
! 8860 CONTINUE
CALL REGRES(Y,X,CKB,A,WV,N,NR,NP,RES)

```

```

CALL MOMENT(RES,N,AVE,ADEV,SDEV,VAR,AM4,SKEW,CURT)
P=2.00
! WRITE(9,3288) P,SKEW,CURT
CALL SVAR(N,RES,NP,RVAR)
! WRITE(*,1223) RVAR
! DO I=1,NP
! DO J=1,NP
! WRITE(*,1223) A(I,J)
! COVLS(I,J)=RVAR*A(I,J)
! WRITE(*,1223) COVLS(I,J)
! ENDDO
! ENDDO
! 1223 FORMAT(F8.5)
! DO I=1,NP
! WRITE(*,4448) (COVLS(I,J),J=1,NP)
! ENDDO
! PAUSE
WRITE(*,*) ' LEAST SQUARES'
WRITE(*,555) (CKB(II),II=1,NP)
555 FORMAT(2F10.4)
! PAUSE
C write(9,1112)
C 1112 format(//,' LEAST SQUARES estimate',//)
C write(9,556) ckb(1),ckb(2)
C 556 format(' ALFA = ',f10.6, ' Beta = ',f10.6,//)
C CALCULATE WEIGHTED LEAST SQUARESTED
C WITH ADAPTIVE P
! 8860 CONTINUE
CALL WLSP(Y,X,BP,N,NR,NP,NC,P,R,IO,I1,WV,WV)
WRITE(*,1001) P
1001 FORMAT(' P= ',F10.4)
CALL SVAR(N,R,NP,RVAR)
P1=P
CALL MOMENT(R,N,AVE,ADEV,SDEV,VAR,AM4,SKEW,CURT)
WRITE(*,*) VAR,RVAR
SSY=DFLOAT(N-1)*VARY
SSE=DFLOAT(N-NP)*RVAR
RSQ2=1.DO-SSE/SSY
WRITE(9,3289) P,BP(2,1),BP(1,1),SKEW,CURT
3289 FORMAT(8H WLS P =,F7.5,7H ALFA =,F7.5,

```

```

& 7H BETA =,F7.5,7H SKEW =,F8.4,7H CURT =,F8.4)
WRITE(9,3887) RVAR,RSQ2,VARY
3887 FORMAT(8H RVAR = ,F7.5,8H RSQ2 = ,F7.5,8H VARY = ,F7.5)
! CALCULATE ROBUST M-ESTIMATORS
CALL MEDFIT(XR,YR,N,AR,BR,ABDEV)
SSDEV=0.
DO JJ=1,N
D=YR(JJ)-(BR*XR(JJ)+AR)
SSDEV=SSDEV+D**2
ENDDO
VARDEV=SSDEV/DFLOAT(N-NP)
RSQ2=1.00-SSDEV/SSY
WRITE(*,4447) AR,BR,VARDEV,RSQ2
WRITE(9,4447) AR,BR,VARDEV,RSQ2
4447 FORMAT(10H MEDFIT = ,4F8.5)
DO I=1,N
X(I,1)=XR(I)
X(I,2)=YR(I)
X(I,3)=WWV(I)
X(I,4)=WYV(I)
ENDDO
OUTFILE=NSHARE//'.RET'
OPEN(8,FILE=OUTFILE,STATUS='UNKNOWN',CARRIAGECONTROL='LIST')
DO I=1,N
WRITE(8,112) NSHARE,MDATE(I+1),X(I,1),X(I,2),X(I,3),X(I,4)
XPORT(I, KK)=X(I,2)
ENDDO
112 format(A11,I8,4F12.8)
! PAUSE
LDX=NR
LDCOV=NC
NOBS=N
NCOL=NC
LDR=NC
INTCEP=1
IIND=1
INDIND(1)=1
IRSP=2
IFRQ=0
PR=0.975

```

```
DF=FLOAT(NOBS-NCOEF)
TOL=100*DMACH(4)
MAXIT=100
EPS=0.0001
write(*,4442) p
4442 format(f10.6)
8860 CONTINUE
CALL DRLLP(NOBS,NCOL,X,LDX,INTCEP,IIND,INDIND,IRSP,
& IFRQ,IWT,P,TOL,MAXIT,EPS,B,RCOV,LDR,IRANK,
& DFE,E,SCALE2,ELP,ITER,NRMISS)
CALL SVAR(N,E,NP,RVAR)
RSS=FLOAT(N-NP)*RVAR
WRITE(*,5128) RVAR,SCALE2
write(9,4128) rvar,scale2
4128 format(8H RVAR = ,f10.5,10H SCALE2 = ,f10.5)
5128 FORMAT(2F10.5)
SSY=FLOAT(N-1)*VARY
RSQ2=1.-RSS/SSY
! WRITE(*,4448) B
IF(IWT.EQ.0) THEN
BETLP(KK)=B(2)
RVARLP(KK)=RVAR
RSCALP(KK)=SCALE2
WRITE(*,7884) BETLP(KK),RVARLP(KK)
C PAUSE
ELSE
ENDIF
CALL DRCOVB(NCOEF,RCOV,LDR,SCALE2,COVB,LDCOV)
! DO 4444 I=1,NCOEF
! 4444 WRITE(*,4448) (COVB(I,J),J=1,NCOEF)
CALL DCONFIL(PR,DF,NC,NCOEF,B,COVB,UPPER,LOWER,SE,TI)
WRITE(9,4442) P
DO 3329 I=1,NCOEF
3329 WRITE(9,3441) B(I),SE(I),TI(I),LOWER(I),UPPER(I)
3441 FORMAT(5F10.5)
CALL MOMENT(E,N,AVE,ADEV,SDEV,VAR,AM4,SKEW,CURT)
CALL SKECUR(N,SKEW,CURT,Z1,Z2,CHI)
WRITE(9,6287) SKEW,CURT,Z1,Z2,CHI
WRITE(9,6288) SCALE2,RSQ2,VARY
6288 FORMAT(3F10.5)
```

```

6287 FORMAT(5F10.5)
P=1.00
write(*,4441) P
4441 format(f8.4)
CALL DRLLP(NOBS,NCOL,X,LDX,INTCEP,IIND,INDIND,IRSP,
& IFRQ,IWT,P,TOL,MAXIT,EPS,B,RCOV,LDR,IRANK,
& DFE,E,SCALE2,ELP,ITER,NRMISS)
! WRITE(*,4448) B
CALL SVAR(N,E,NP,RVAR)
WRITE(*,5123) RVAR,SCALE2
5123 FORMAT(2F10.5)
RSS=FLOAT(N-NP)*RVAR
SSY=FLOAT(N-1)*VARY
RSQ2=1.-RSS/SSY
IF(IWT.EQ.0) THEN
BETL1(KK)=B(2)
RVARL1(KK)=RVAR
RSCAL1(KK)=SCALE2
WRITE(*,7884) BETL1(KK),RVARL1(KK)
C PAUSE
ELSE
ENDIF
CALL DRCOVB(NCOEF,RCOV,LDR,SCALE2,COVB,LDCOV)
! DO 3444 I=1,NCOEF
! 3444 WRITE(*,4448) (COVB(I,J),J=1,NCOEF)
CALL DCONFIL(PR,DF,NC,NCOEF,B,COVB,UPPER,LOWER,SE,TI)
WRITE(9,4441) P
DO 3328 I=1,NCOEF
3328 WRITE(9,3441)B(I),SE(I),TI(I),LOWER(I),UPPER(I)
CALL MOMENT(E,N,AVE,ADEV,SDEV,VAR,AM4,SKEW,CURT)
CALL SKECUR(N,SKEW,CURT,Z1,Z2,CHI)
WRITE(9,3287) SKEW,CURT,Z1,Z2,CHI
3287 FORMAT(5F10.5)
WRITE(9,6288)SCALE2,RSQ2,VARY
P=2.00
write(*,4443) P
4443 format(f8.4)
CALL DRLLP(NOBS,NCOL,X,LDX,INTCEP,IIND,INDIND,IRSP,
& IFRQ,IWT,P,TOL,MAXIT,EPS,B,RCOV,LDR,IRANK,
& DFE,E,SCALE2,ELP,ITER,NRMISS)

```

```

CALL SVAR(N,E,NP,RVAR)
WRITE(*,4348) RVAR,SCALE2
4348 FORMAT(2F10.5)
! PAUSE
RSS=FLOAT(N-NP)*SCALE2
SSY=FLOAT(N-1)*VARY
RSQ2=1.- RSS/SSY
! WRITE(*,4448) B
IF(IWT.EQ.0) THEN
BETLS(KK)=B(2)
DO 4433 II=1,N
RESMAT(II,KK)=E(II)
4433 CONTINUE
RVARLS(KK)=RVAR
RSCALS(KK)=SCALE2
WRITE(*,7884) BETLS(KK),RVARLS(KK)
C PAUSE
7884 FORMAT(2F12.9)
ELSE
ENDIF
CALL DRCOV(B,NCOEF,RCOV,LDR,SCALE2,COVB,LDCOV)
! DO 2444 I=1,NCOEF
! 2444 WRITE(*,4448) (COVB(I,J),J=1,NCOEF)
CALL DCONFIL(PR,DF,NC,NCOEF,B,COVB,UPPER,LOWER,SE,TI)
CALL MOMENT(E,N,AVE,ADEV,SDEV,VAR,AM4,SKEW,CURT)
CALL SKECUR(N,SKEW,CURT,Z1,Z2,CHI)
WRITE(9,4441) P
DO 3327 I=1,NCOEF
3327 WRITE(9,3441)B(I),SE(I),TI(I),LOWER(I),UPPER(I)
WRITE(9,3287) SKEW,CURT,Z1,Z2,CHI
WRITE(9,6288) SCALE2,RSQ2,VARY
if(iwt.eq.0) then
AVEBET=AVEBET+B(2)
AVESTD=AVESTD+SE(2)
NAVE=NAVE+1
WEIGHT=SBAYES**2/(SBAYES**2+SE(2)**2)
BAYES=WEIGHT*B(2)+(1-WEIGHT)*BAYES
VARB=WEIGHT**2*SE(2)**2 + (1-WEIGHT)**2*SBAYES**2
SVARB=DSQRT(VARB)
TBAYES=BAYES/SVARB

```

```
WRITE(9,6233) BAYES,SVARB,TBAYES
6233 FORMAT(14H BAYES BETA = ,F10.5,8H S.E. = ,F10.5,
&10H t-stat = ,F10.5)
else
endif
IF(IWT.EQ.3) THEN
GOTO 180
ELSE
DO 3339 I=1,N
WV(I)=WV(I)
3339 CONTINUE
IWT=3
P=P1
GOTO 8860
ENDIF
180 CONTINUE
9898 CONTINUE
DO 6344 II=1,N+1
WRITE(24,6345) (MRAW(II,JJ),JJ=1,11)
6344 CONTINUE
6345 FORMAT(I8,10I7)
C PAUSE
CALL COVRET(NR,NC,N,NSHA,XPORT,XBAR,SIGMA)
DO 9881 I=1,NSHA
WRITE(*,*) XBAR(I)
9881 CONTINUE
PAUSE
DO 9885 I=1,NSHA
WRITE(20,9884) XBAR(I)
WRITE(21,9884) XBAR(I)
WRITE(22,9884) XBAR(I)
WRITE(23,9884) XBAR(I)
WRITE(31,9884) XBAR(I)
WRITE(32,9884) XBAR(I)
WRITE(33,9884) XBAR(I)
9885 CONTINUE
9884 FORMAT(F12.9)
DO 9883 J=1,NSHA
DO 9883 I=1,NSHA
WRITE(*,*) SIGMA(I,J)
```

```
WRITE(20,9884) SIGMA(I,J)
9883 CONTINUE
DO 3320 I=1,NSHA
WRITE(*,3319) BETLP(I),BETL1(I),BETLS(I)
3320 CONTINUE
3319 FORMAT(3F12.9)
C PAUSE
DO 3221 I=1,NSHA
WRITE(21,9884) RVARLP(I)
WRITE(31,9884) RSCALP(I)
3221 CONTINUE
WRITE(21,9884) VINDEX
WRITE(31,9884) VINDEX
DO 3321 I=1,NSHA
WRITE(21,9884) BETLP(I)
WRITE(31,9884) BETLP(I)
3321 CONTINUE
DO 3222 I=1,NSHA
WRITE(22,9884) RVARL1(I)
WRITE(32,9884) RSCAL1(I)
3222 CONTINUE
WRITE(22,9884) VINDEX
WRITE(32,9884) VINDEX
DO 3322 I=1,NSHA
WRITE(22,9884) BETL1(I)
WRITE(32,9884) BETL1(I)
3322 CONTINUE
DO 3223 I=1,NSHA
WRITE(23,9884) RVARLS(I)
WRITE(33,9884) RSCALS(I)
3223 CONTINUE
WRITE(23,9884) VINDEX
WRITE(33,9884) VINDEX
DO 3323 I=1,NSHA
WRITE(23,9884) BETLS(I)
WRITE(33,9884) BETLS(I)
3323 CONTINUE
DO 2220 I=1,NSHA
DO 2220 J=1,NSHA
SOM=0.0
```

```
DO 2210 K=1,N
SOM =SOM+RESMAT(K,I)*RESMAT(K,J)
2210 CONTINUE
SIGRES(I,J)=SOM/DFLOAT(N-2)
2220 CONTINUE
DO 5223 II=1,NSHA
5223 WRITE(*,2224)(SIGRES(II,JJ),JJ=1,NSHA)
2224 FORMAT(6F12.9)
PAUSE
DO 2225 II=1,NSHA
DO 2226 JJ=1,NSHA
BETMAT(II,JJ)=BETLS(II)*BETLS(JJ)
2226 CONTINUE
2225 CONTINUE
DO 2227 II=1,NSHA
2227 WRITE(*,2224) (BETMAT(II,JJ),JJ=1,NSHA)
PAUSE
CALL INNOV(NR,NC,N,NSHA,VINDEX,RESMAT,BETLS,PSIMAT,DIAMAT)
DO 4221 II=1,NSHA
4221 WRITE(*,4431) PSIMAT(II,II),DIAMAT(II,II)
4431 FORMAT(2F12.9)
OPEN(28,FILE='BPSI.INP',STATUS='UNKNOWN',CARRIAGECONTROL='LIST')
OPEN(29,FILE='BDIA.INP',STATUS='UNKNOWN',CARRIAGECONTROL='LIST')
OPEN(34,FILE='BBAYES.INP',STATUS='UNKNOWN',CARRIAGECONTROL='LIST')
OPEN(35,FILE='BL1.INP',STATUS='UNKNOWN',CARRIAGECONTROL='LIST')
OPEN(36,FILE='BLP.INP',STATUS='UNKNOWN',CARRIAGECONTROL='LIST')
DO 4434 II=1,NSHA
WRITE(28,4533) XBAR(II)
WRITE(34,4533) XBAR(II)
WRITE(35,4533) XBAR(II)
WRITE(36,4533) XBAR(II)
4434 WRITE(29,4533) XBAR(II)
4533 FORMAT(F12.9)
DO 4435 II=1,NSHA
DO 4436 JJ=1,NSHA
WRITE(28,4533) PSIMAT(II,JJ)
4436 CONTINUE
4435 CONTINUE
DO 4437 II=1,NSHA
DO 4438 JJ=1,NSHA
```



```

WRITE(29,4533) DIAMAT(II,JJ)
4438 CONTINUE
4437 CONTINUE
AVEBET=AVEBET/DFLOAT(NAVE)
AVESTD=AVESTD/DFLOAT(NAVE)
WRITE(*,9891) NAVE,AVEBET,AVESTD
9891 FORMAT(8H NAVE = ,I5,16H AVERAGE BETA = ,F9.5,
&15H AVERAGE STD = ,F9.5)
close(11)
close(8)
close(9)
close(12)
end
SUBROUTINE MONTH(MAX,NIN,DATE,LAST,HI,LO,VO,
+ NOUT,MDATE,MLAST,MHI,MLO,MVO)
C CALCULATES MONTHLY DATA
integer*4 DATE(MAX)
integer*4 LAST(MAX)
integer*4 HI(MAX)
integer*4 LO(MAX)
integer*4 VO(MAX)
integer*4 MDATE(MAX)
integer*4 MLAST(MAX)
integer*4 MHI(MAX)
integer*4 MLO(MAX)
integer*4 MVO(MAX)
I=0
CUMVO=0
CUMLO=99999999
CUMHI=0
DO 9010 J=1,NIN-1
IF(LAST(J).GT.CUMHI) CUMHI=LAST(J)
IF(LAST(J).LT.CUMLO) CUMLO=LAST(J)
CUMVO=CUMVO+VO(J)
IF(INT(DATE(J)/100).NE.INT(DATE(J+1)/100)) THEN
C STRUCK END OF MONTH
I=I+1
MDATE(I)=DATE(J)
MLAST(I)=LAST(J)
MHI(I)=HI(J)

```

```

MLO(I)=LO(J)
MVO(I)=CUMVO
CUMVO=0
CUMLO=9999999
CUMHI=0
ENDIF
9010 CONTINUE
NOUT=I
WRITE(*,*) NOUT
! PAUSE
DO 9020 J=I+1,MAX
MDATE(J)=0
MLAST(J)=0
MHI(J)=0
MLO(J)=0
MVO(J)=0
9020 CONTINUE
NOUT=I
RETURN
END
SUBROUTINE TRATRA(MAX,NIN,DATE,LAST,HI,LO,VO,
+ NOUT,MDATE,MLAST,MHI,MLO,MVO)
C CALCULATES MONTHLY DATA
integer*4 DATE(MAX)
integer*4 LAST(MAX)
integer*4 HI(MAX)
integer*4 LO(MAX)
integer*4 VO(MAX)
integer*4 MDATE(MAX)
integer*4 MLAST(MAX)
integer*4 MHI(MAX)
integer*4 MLO(MAX)
integer*4 MVO(MAX)
I=0
DO 4100 J=1,NIN
IF(VO(J).GT.0) THEN
I=I+1
MDATE(I)=DATE(J)
MLAST(I)=LAST(J)
MHI(I)=HI(J)

```

```
MLO(I)=LO(J)
MVO(I)=VO(J)
GOTO 4100
ELSE
ENDIF
4100 CONTINUE
NOUT=I
RETURN
END
SUBROUTINE PRITRA(MAX,NIN,DATE,LAST,HI,LO,VO,
+ NOUT,MDATE,MLAST,MHI,MLO,MVO)
C CALCULATES MONTHLY DATA ONLY IF SHARE MOVES
integer*4 DATE(MAX)
integer*4 LAST(MAX)
integer*4 HI(MAX)
integer*4 LO(MAX)
integer*4 VO(MAX)
integer*4 MDATE(MAX)
integer*4 MLAST(MAX)
integer*4 MHI(MAX)
integer*4 MLO(MAX)
integer*4 MVO(MAX)
MDATE(1)=DATE(1)
MLAST(1)=LAST(1)
MHI(1)=HI(1)
MLO(1)=LO(1)
MVO(1)=VO(1)
I=1
DO 4100 J=1,NIN-1
IF(LAST(J+1)-LAST(J).NE.0) THEN
I=I+1
MDATE(I)=DATE(J+1)
MLAST(I)=LAST(J+1)
MHI(I)=HI(J+1)
MLO(I)=LO(J+1)
MVO(I)=VO(J+1)
GOTO 4100
ELSE
ENDIF
4100 CONTINUE
```

```

NOUT=I
RETURN
END
SUBROUTINE WLSP(Y,X,B,N,NR,NP,NC,P,R,IO,I1,WWV,WYV)
C WEIGHTED LEAST SQUARES ESTIMATION WITH ADAPTIVE P
C LP-NORM ESTIMATION SEE BARR,GDI(1981)
IMPLICIT REAL*8(A-H,O-Z)
C
C CHANGE DIMENSION STATEMENTS TO FIT PROBLEM ON HAND
PARAMETER(NROW=2000,NCOL=10)
DIMENSION X(NR,NC),Y(NR),B(NC,1),R(NR)
DIMENSION WWV(NR),WYV(NR)
DIMENSION Z(NROW,NCOL)
C NOTE MATRIX Z=MATRIX(X(2,NC),Y) I.E. NO CONSTANT
DATA EPS/1.0D-4/
NVE=NP
IO=0
I1=0
DO 1 I=1,N
DO 2 J=1,NP
Z(I,J)=X(I,J+1)*DSQRT(WWV(I))
2 CONTINUE
1 CONTINUE
DO 3 I=1,N
3 Z(I,NP)=Y(I)*DSQRT(WYV(I))
P=2.0
176 PCON=P
CALL WLS(Z,B,P,NVE,NC,N,NR,IFAU, R,SD,RATE,NPO,IT)
WRITE(*,150) IFAU
IF(IFAU.EQ.0) THEN
IO=IO+1
ELSE
ENDIF
IF(IFAU.EQ.1) THEN
I1=I1+1
ELSE
ENDIF
150 FORMAT( ' IFAU= ',I4)
WRITE(*,100) (B(I,1),I=1,NVE)
100 FORMAT(1X,6F10.4)

```

```

CALL MOMENT(R,N,AVE,ADEV,SDEV,VAR,AM4,SKEW,CURT)
WRITE(*,160) VAR,SKEW,CURT
160 FORMAT(' VAR= ',F10.4,' SKEW= ',F10.4,' CURT= ',F10.4)
P=9.0/(CURT**2.0)+1.0
CURT2=AM4/(VAR**2.0)
P1=9.0/(CURT2**2.0)+1.0
WRITE(*,170) P,P1
170 FORMAT(' P= ',F10.4,' P1= ',F10.4)
P=P1
IF((PCON-P).LT.EPS) THEN
GOTO 180
ELSE
ENDIF
GOTO 176
180 WRITE(*,190) P
190 FORMAT(' CONVERGED, P= ',F10.4)
RETURN
END
C
SUBROUTINE WLS(Z,B,P,NVE,NC,N,NR,IFAUULT,R,SD,RATE,NPO,IT)
C WEIGHTED LEAST SQUARES PROGRAM
C
C DOUBLE ASCII FORTRAN
C EXTENSION OF SPOSITO,VA GENTLE,J.E. & KENNEDY,W.J.(1977)
C ALGORITHM AS110
C LP-NORM FIT WITH ADAPTIVE P
C DIMENSIONED TO HANDLE 9 X'S
IMPLICIT REAL*8(A-H,O-Z)
PARAMETER(NROW=2000,NCOL=10)
REAL*8 Z(NR,NC),ZC(NCOL,NCOL),XC(NCOL,NCOL),YC(NCOL,1),R(NR)
REAL*8 V(3),B(NC,1),VMEAN(NCOL),DMROW(NCOL),A(NCOL)
REAL*8 W(NCOL),V1(NCOL,NCOL),B1(NCOL),Y1(NCOL),E(NROW)
C
C
C DEFINITIONS OF VARIABLES
C N = SAMPLE SIZE
C P=P IN LP-NORM
C Z=AUGMENTED (X,Y) MATRIX
C ZC=AUGMENTED MOMENT MATRIX
C XC=X MOMENT MATRIC

```

```
C YC = Y MOMENT MATRIX
C SD= LP-NORM
C R=VECTOR OF RESIDUALS
C RATE = RATE OF CHANGE OF NORMAT TIME OF CONTROL RETURN
C IT=NUMBER OF ITERATIONS
C NPO = NUMBER OF POINTS OMITTED
C IFAULT=0 ] CONVERGENCE
C =1 ] NORM INCREASED
C =2 ] MAXIMUM ITERATIONS EXCEEDED
C =3 ] XC-NONSINGULAR
C
C INIALLIZATION
C
DATA EPS/1.0D-6/,MAXIT/200/
V(1)=3
IFault=0
EPS2=2.*EPS
SD=0.
SD4=-10.
WP=P-2.
C
C
DO 1 I=1,N
1 R(I)=1.
DO 11 IT=1,MAXIT
NPO=0
C
C INITIALIZATION OF ARRAYS
SUMWT=0.
DO 13 I=1,NVE
VMEAN(I)=0.
DO 13 J=1,NVE
ZC(J,I)=0.
13 CONTINUE
C
C COMPUTE MATRIX OF SUMS OF SQUATES AND CROSS-
C PRODUCTS USING HERRAMAN ALGORITHM
C SEE HERRAMAN.C(168) ALGORITHM AS12
C APL. STATIST. 17,289-292
C
```

```

DO 14 NN=1,N
DO 15 I=1,NVE
15 DMROW(I)=Z(NN,I)
C WRITE(*,*) (DMROW(I),I=1,NVE)
ABSRI=ABS(R(NN))
IF(ABSRI.LE.EPS) GOTO 18
WEIGHT=ABSRI**WP
SUMWT=SUMWT + WEIGHT
DIV=WEIGHT/SUMWT
DO 16 I=1,NVE
DMROW(I)=DMROW(I)-VMEAN(I)
DI=DMROW(I)
DO 17 J=1,I
DIJ=DI*DMROW(J)*WEIGHT
ZC(I,J)=ZC(I,J)+DIJ-DIJ*DIV
ZC(J,I)=ZC(I,J)
17 CONTINUE
VMEAN(I)=VMEAN(I)+DI*DIV
16 CONTINUE
GOTO 14
18 NPO=NPO+1
14 CONTINUE
130 FORMAT(1X,6F10.4)
C WRITE(*,130) (VMEAN(I),I=1,NVE)
C DO 120 I=1,NVE
C 120 WRITE(*,130) (ZC(I,J),J=1,NVE)
C
C
NVE1=NVE-1
DO 2 I=1,NVE1
YC(I,1)=ZC(I,NVE)
Y1(I)= YC(I,1)
DO 2 J=1,NVE1
2 XC(I,J)=ZC(I,J)
C
C
C COMPUTE REGRESSION COEFFICINTS
CALL SVDCMP(XC,NVE1,NVE1,10,10,W,V1)
C XC DESTROYED REPLCED BY U WHERE XC=UWV1'
CALL SVBKS(B,XC,W,V1,NVE1,NVE1,10,10,Y1,B1)

```

```
WRITE(*,100) (B1(JJ),JJ=1,NVE1)
DO 110 JJ=1,NVE1
100 FORMAT(1X,6F10.4)
110 B(JJ,1)=B1(JJ)
SXM=0.
DO 21 J=1,NVE1
21 SXM=SXM+B(J,1)*VMEAN(J)
DO 22 I=1,N
E(I)=0.
DO 22 J=1,NVE1
22 E(I)=E(I)+Z(I,J)*B(J,1)
B(NVE,1)=VMEAN(NVE)-SXM
C CALCULATE RESIDUALS AND TEST FOR CONVERGENCE
C
SD2=0.
ISW=0
DO 4 I=1,N
RES=Z(I,NVE)-B(NVE,1)-E(I)
ABSRI=ABS(RES)
IF(ABS(ABSRI-ABS(R(I)))) .GT. EPS2) ISW=1
SD2=SD2+ABSRI**P
R(I)=RES
4 CONTINUE
C WRITE(*,100) (R(I),I=1,N)
RATE=ABS(SD2-SD)/SD2
IF(ISW.EQ.0) GOTO 99
IF(IT.EQ.1) GOTO 5
C
C
C TEST FOR INCREASE IN NORM
C SEE PORTER, M.A. & WINSTANLEY, D.J. (1979) AS R290N AS110
C APL. STATIST., 28, 112-113
SD3=SD2-SD
IF(NPO.EQ.0) GOTO 41
IF(SD3.GT.EPS.AND.SD4.GT.EPS) GOTO 7
SD4=SD3
GOTO 5
41 IF(SD3.GT.EPS) GOTO 7
SD4=-10
C
```

```

5 SD=SD2
DO 51 J=1,NVE
51 A(J)=B(J,1)
11 CONTINUE
C
C FAILED TO CONVERGE IN MAXIMUM ITERATIONS
IFAUULT=2
GOTO 99
C
C NORM INCREASED,RESTORE BETAS AND R,THEN STOP
C
7 IFAULT=1
DO 71 J=1,NVE
71 B(J,1)=A(J)
DO 81 I=1,N
R(I)=0.
DC 81 J=1,NVE1
81 R(I)=R(I)+Z(I,J)*B(J,1)
DO 8 I=1,N
8 R(I)=Z(I,NVE)-B(NVE,1) -R(I)
GOTO 99
C
C XC NON-SINGULAR
98 IFAULT=3
C
99 RETURN
END
SUBROUTINE DMULT(A,B,C,N,K,M,IMAX,JMAX,KMAX)
REAL*8 A(IMAX,JMAX),B(JMAX,KMAX),C(IMAX,KMAX)
C MUTIPLIES MATRIX WITH MATRIX B GIVING C
DO 1 J=1,M
DO 1 I=1,N
C(I,J)=0.
DO 1 KK=1,K
1 C(I,J)=C(I,J)+A(I,KK)*B(KK,J)
RETURN
END
C
C
SUBROUTINE MOMENT(DATA,N,AVE,ADEV,SDEV,VAR,AM4,SKEW,CURT)

```

```

IMPLICIT REAL*8(A-H,O-Z)
DIMENSION DATA(N)
IF(N.LE.1)PAUSE 'N must be at least 2'
S=0.
AN=DFLOAT(N)
DO 11 J=1,N
S=S+DATA(J)
11 CONTINUE
AVE=S/N
ADEV=0.
VAR=0.
SKEW=0.
CURT=0.
DO 12 J=1,N
S=DATA(J)-AVE
ADEV=ADEV+ABS(S)
P=S*S
VAR=VAR+P
P=P*S
SKEW=SKEW+P
P=P*S
CURT=CURT+P
12 CONTINUE
ADEV=ADEV/N
VAR=VAR/DFLOAT(AN-1.0)
SDEV=SQRT(VAR)
IF(VAR.NE.0.)THEN
SKEW=SKEW/(N*SDEV**3)
AM4=CURT
ACON=(AN**2.0-2.0*AN+3.0)/((AN-1.0)*(AN-2.0)*(AN-3.0))
A41=ACON*AM4
A42=(3.0*(AN-1.0)*(2.0*AN-3.0))/(AN*(AN-2.0)*(AN-3.0))
A42=A42*VAR**2.0
AM4=A41-A42
CURT=CURT/(N*VAR**2)
ELSE
! PAUSE 'no skew or kurtosis when zero variance'
ENDIF
RETURN
END

```

```
C
C
SUBROUTINE SVDCMP(A,M,N,MP,NP,W,V)
PARAMETER (NMAX=100)
double precision A(MP,NP),W(NP),V(NP,NP),RV1(NMAX)
double precision g,scale,anorm,s,F,H
DOUBLE PRECISION C,X,Y,Z
G=dbl(0.0)
SCALE=dbl(0.0)
ANORM=dbl(0.0)
DO 25 I=1,N
L=I+1
RV1(I)=SCALE*G
G=dbl(0.0)
S=dbl(0.0)
SCALE=dbl(0.0)
IF (I.LE.M) THEN
DO 11 K=I,M
SCALE=SCALE+DABS(A(K,I))
11 CONTINUE
IF (SCALE.NE.dbl(0.0)) THEN
DO 12 K=I,M
A(K,I)=A(K,I)/SCALE
S=S+A(K,I)*A(K,I)
12 CONTINUE
F=A(I,I)
G=-DSIGN(DSQRT(S),F)
H=F*G-S
A(I,I)=F-G
IF (I.NE.N) THEN
DO 15 J=L,N
S=DBLE(0.0)
DO 13 K=I,M
S=S+A(K,I)*A(K,J)
13 CONTINUE
F=S/H
DO 14 K=I,M
A(K,J)=A(K,J)+F*A(K,I)
14 CONTINUE
15 CONTINUE
```

```
ENDIF
DO 16 K= I,M
A(K,I)=SCALE*A(K,I)
16 CONTINUE
ENDIF
ENDIF
W(I)=SCALE *G
G=DBLE(0.0)
S=DBLE(0.0)
SCALE=DBLE(0.0)
IF ((I.LE.M).AND.(I.NE.N)) THEN
DO 17 K=L,N
SCALE=SCALE+DABS(A(I,K))
17 CONTINUE
IF (SCALE.NE.DBLE(0.0)) THEN
DO 18 K=L,N
A(I,K)=A(I,K)/SCALE
S=S+A(I,K)*A(I,K)
18 CONTINUE
F=A(I,L)
G=-DSIGN(DSQRT(S),F)
H=F*G-S
A(I,L)=F-G
DO 19 K=L,N
RV1(K)=A(I,K)/H
19 CONTINUE
IF (I.NE.M) THEN
DO 23 J=L,M
S=DBLE(0.0)
DO 21 K=L,N
S=S+A(J,K)*A(I,K)
21 CONTINUE
DO 22 K=L,N
A(J,K)=A(J,K)+S*RV1(K)
22 CONTINUE
23 CONTINUE
ENDIF
DO 24 K=L,N
A(I,K)=SCALE*A(I,K)
24 CONTINUE
```

```

ENDIF
ENDIF
ANORM=DMAX1(ANORM,(DABS(W(I))+DABS(RV1(I))))
25 CONTINUE
DO 32 I=N,1,-1
IF (I.LT.N) THEN
IF (G.NE.DBLE(0.0)) THEN
DO 26 J=L,N
V(J,I)=(A(I,J)/A(I,L))/G
26 CONTINUE
DO 29 J=L,N
S=DBLE(0.0)
DO 27 K=L,N
S=S+A(I,K)*V(K,J)
27 CONTINUE
DO 28 K=L,N
V(K,J)=V(K,J)+S*V(K,I)
28 CONTINUE
29 CONTINUE
ENDIF
DO 31 J=L,N
V(I,J)=DBLE(0.0)
V(J,I)=DBLE(0.0)
31 CONTINUE
ENDIF
V(I,I)=DBLE(1.0)
G=RV1(I)
L=I
32 CONTINUE
DO 39 I=N,1,-1
L=I+1
G=W(I)
IF (I.LT.N) THEN
DO 33 J=L,N
A(I,J)=DBLE(0.0)
33 CONTINUE
ENDIF
IF (G.NE.DBLE(0.0)) THEN
G=DBLE(1.0)/G
IF (I.NE.N) THEN

```

```
DO 36 J=L,N
S=DBLE(0.0)
DO 34 K=L,M
S=S+A(K,I)*A(K,J)
34 CONTINUE
F=(S/A(I,I))*G
DO 35 K=I,M
A(K,J)=A(K,J)+F*A(K,I)
35 CONTINUE
36 CONTINUE
ENDIF
DO 37 J=I,M
A(J,I)=A(J,I)*G
37 CONTINUE
ELSE
DO 38 J= I,M
A(J,I)=DBLE(0.0)
38 CONTINUE
ENDIF
A(I,I)=A(I,I)+DBLE(1.0)
39 CONTINUE
DO 49 K=N,1,-1
DO 48 ITS=1,30
DO 41 L=K,1,-1
NM=L-1
IF ((DABS(RV1(L))+ANORM).EQ.ANORM) GO TO 2
IF ((DABS(W(NM))+ANORM).EQ.ANORM) GO TO 1
41 CONTINUE
1 C=DBLE(0.0)
S=DBLE(1.0)
DO 43 I=L,K
F=S*RV1(I)
IF ((DABS(F)+ANORM).NE.ANORM) THEN
G=W(I)
H=DSQRT(F*F+G*G)
W(I)=H
H=DBLE(1.0)/H
C= (G*H)
S=-(F*H)
DO 42 J=1,M
```

```

Y=A(J,NM)
Z=A(J,I)
A(J,NM)=(Y*C)+(Z*S)
A(J,I)=- (Y*S)+(Z*C)
42 CONTINUE
ENDIF
43 CONTINUE
2 Z=W(K)
IF (L.EQ.K) THEN
IF (Z.LT.DBLE(0.0)) THEN
W(K)=-Z
DO 44 J=1,N
V(J,K)=-V(J,K)
44 CONTINUE
ENDIF
GO TO 3
ENDIF
IF (ITS.EQ.30) PAUSE 'No convergence in 30 iterations'
X=W(L)
NM=K-1
Y=W(NM)
G=RV1(NM)
H=RV1(K)
F=((Y-Z)*(Y+Z)+(G-H)*(G+H))/(DBLE(2.0)*H*Y)
G=DSQRT(F*F+DBLE(1.0))
F=((X-Z)*(X+Z)+H*((Y/(F+DSIGN(G,F)))-H))/X
C=DBLE(1.0)
S=DBLE(1.0)
DO 47 J=L,NM
I=J+1
G=RV1(I)
Y=W(I)
H=S*G
G=C*G
Z=DSQRT(F*F+H*H)
RV1(J)=Z
C=F/Z
S=H/Z
F= (X*C)+(G*S)
G=- (X*S)+(G*C)

```

```

H=Y*S
Y=Y*C
DO 45 NM=1,N
X=V(NM, J)
Z=V(NM, I)
V(NM, J)= (X*C)+(Z*S)
V(NM, I)=- (X*S)+(Z*C)
45 CONTINUE
Z=DSQRT(F*F+H*H)
W(J)=Z
IF (Z.NE.DBLE(0.0)) THEN
Z=DBLE(1.0)/Z
C=F*Z
S=H*Z
ENDIF
F= (C*G)+(S*Y)
X=- (S*G)+(C*Y)
DO 46 NM=1,M
Y=A(NM, J)
Z=A(NM, I)
A(NM, J)= (Y*C)+(Z*S)
A(NM, I)=- (Y*S)+(Z*C)
46 CONTINUE
47 CONTINUE
RV1(L)=DBLE(0.0)
RV1(K)=F
W(K)=X
48 CONTINUE
3 CONTINUE
49 CONTINUE
RETURN
END
SUBROUTINE SVBKS(U,W,V,M,N,MP,NP,B,X)
PARAMETER (NMAX=100)
Double precision U(MP,NP),W(NP),V(NP,NP)
double precision B(MP),X(NP),TMP(NMAX),S
DO 12 J=1,N
S=dble(0.)
IF(W(J).NE.0.)THEN
DO 11 I=1,M

```

```

S=S+U(I,J)*B(I)
11 CONTINUE
S=S/W(J)
ENDIF
TMP(J)=S
12 CONTINUE
DO 14 J=1,N
S=dbl(0.)
DO 13 JJ=1,N
S=S+V(J,JJ)*TMP(JJ)
13 CONTINUE
X(J)=S
14 CONTINUE
RETURN
END
C SUBROUTINE FOR MATRIX INVERSION
C *****
C
C
C MAXIMUM PIVOT STRATEGY
C APPLIED NUMERICAL METHODS , EXAMPLE 5.2
C
SUBROUTINE INVERT(N,A,X,EPS,INDIC,NRC,DINVER)
C CALL INVERT(NP,A,XS,EPS,-1,NCOL,DETER)
IMPLICIT REAL*8(A-H,O-Z)
REAL*8 A,X,EPS,DINVER
C
C CHANGE DIMENSION STATEMENT TO FIT PROBLEM ON HAND
C
DIMENSION IROW(50),JCOL(50),JORD(50),Y(50),A(NRC,NRC),X(N)
DO 3 I=1,50
IROW(I)=0
JCOL(I)=0
JORD(I)=0
Y(I)=0.
3 CONTINUE
MAX=N
IF(INDIC.GE.0)MAX=N+1
IF(N.LE.50) GO TO 5
WRITE(6,200)

```

```

NM1=N-1
DO 22 I=1,NM1
  IP1=I+1
  DO 22 J=IP1,N
    IF(JORD(J).GE.JORD(I)) GO TO 22
    JTEMP=JORD(J)
    JORD(J)=JORD(I)
    JORD(I)=JTEMP
    INTCH=INTCH+1
  22 CONTINUE
  IF(INTCH/2*2.NE.INTCH) DETER=-DETER
  IF(INDIC.LE.0) GO TO 26
  DINVER=DETER
  RETURN
26 DO 28 J=1,N
  DO 27 I=1,N
    IROWI=IROW(I)
    JCOLI=JCOL(I)
    27 Y(JCOLI)=A(IROWI,J)
  DO 28 I=1,N
    28 A(I,J)=Y(I)
  DO 30 I=1,N
  DO 29 J=1,N
    IROWJ=IROW(J)
    JCOLJ=JCOL(J)
    29 Y(IROWJ)=A(I,JCOLJ)
  DO 30 J=1,N
    30 A(I,J)=Y(J)
  DINVER=DETER
  RETURN
200 FORMAT( 10HON TOO BIG )
C
END
C
C SUBROUTINE THAT PERFORMS REGRESSION ANALYSIS ON DATA
C *****
C
SUBROUTINE REGRES(Y,X,C,A,B,N,NR,NP,RES)
IMPLICIT REAL*8(A-H,O-Z)
C

```

```

DINVER=0.
RETURN
5 DETER=1.
DO 18 K=1,N
  KM1=K-1
  PIVOT=0.
  DO 11 I=1,N
    DO 11 J=1,N
      IF(K.EQ.1) GO TO 9
      DO 8 ISCAN=1,KM1
      DO 8 JSCAN=1,KM1
      IF(I.EQ.IROW(ISCAN)) GO TO 11
      8 IF(J.EQ.JCOL(JSCAN)) GO TO 11
      9 IF(DABS(A(I,J)).LE.DABS(PIVOT)) GO TO 11
      PIVOT=A(I,J)
      IROW(K)=I
      JCOL(K)=J
      11 CONTINUE
      IF(DABS(PIVOT).GT.EPS)GO TO 13
      DINVER=0.
      RETURN
      13 IROWK=IROW(K)
      JCOLK=JCOL(K)
      DETER=PIVOT*DETER
      DO 14 J=1,MAX
      14 A(IROWK,J)=A(IROWK,J)/PIVOT
      A(IROWK,JCOLK)=1./PIVOT
      DO 18 I=1,N
      AIJCK=A(I,JCOLK)
      IF(I.EQ.IROWK) GO TO 18
      A(I,JCOLK)=-AIJCK/PIVOT
      DO 17 J=1,MAX
      17 IF(J.NE.JCOLK) A(I,J)=A(I,J)-AIJCK*A(IROWK,J)
      18 CONTINUE
      DO 20 I=1,N
      IROWI=IROW(I)
      JCOLI=JCOL(I)
      JORD(IROWI)=JCOLI
      20 IF(INDIC.GE.0) X(JCOLI)=A(IROWI,MAX)
      INTCH=0

```

```
C CHANGE DIMENSION STATEMENTS TO FIT PROBLEM ON HAND
C
PARAMETER(NCOL=2)
DIMENSION X(NR,NP),Y(NR),A(NP,NP),B(NR),C(NP),XS(NCOL),RES(NR)
DATA EPS/1.E-10/
DO 20 I=1,NP
DO 20 J=1,NP
SOM=0.
DO 10 K=1,N
SOM=SOM+B(K)*X(K,I)*X(K,J)
10 CONTINUE
A(I,J)=SOM
20 CONTINUE
WRITE(*,*) 'XTX COMPUTED USING INVERT'
DO 1111 I=1,NP
1111 WRITE(*,1112) (A(I,J),J=1,NP)
1112 FORMAT(6F8.4)
CALL INVERT(NP,A,XS,EPS,-1,NP,DETER)
DO 50 I=1,NP
SOM=0.
DO 40 J=1,NP
SUM=0.
DO 30 K=1,N
SUM=B(K)*X(K,J)*Y(K)+SUM
30 CONTINUE
SOM=(A(I,J))*SUM+SOM
40 CONTINUE
C(I)=SOM
50 CONTINUE
DO 70 I=1,N
SUM=0.0
DO 60 J=1,NP
SUM=C(J)*X(I,J)+SUM
60 CONTINUE
RES(I)=Y(I)-SUM
70 CONTINUE
RETURN
END
C
C
```

```
SUBROUTINE QSORT(N,X,NR)
C
C.....QUICKSORT - A PARTITION SORTING ALGORITHM
C.....SORTS AN ARRAY X, SIZE N, IN ASCENDING ORDER
C.....THE EXPECTED TIME COMPLEXITY OF THIS ALGORITHM IS O(N*LOGN)
C.....NOTE: FOR N > 1000 ENLARGE THE INDEXES OF IONDER AND IBO
C..... DOUBLE PRECISION IS USED
C
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION IONDER(15),IBO(15)
DIMENSION X(NR)
ITOP = 1
IONDER(1) = 1
IBO(1) = N
C
500 IF (ITOP.EQ.0) GO TO 800
ILG = IONDER(ITOP)
IBG = IBO(ITOP)
ITOP = ITOP - 1
I = ILG
J = IBG
C
510 IF (I.EQ.J) GO TO 600
IF (X(I).GT.X(J)) GO TO 515
J = J-1
GO TO 510
515 HOU = X(I)
X(I) = X(J)
X(J) = HOU
I = I+1
C
520 IF (I.EQ.J) GO TO 600
IF (X(I).GT.X(J)) GO TO 525
I = I+1
GO TO 520
525 HOU = X(I)
X(I) = X(J)
X(J) = HOU
J = J-1
GO TO 510
```

```

C
600 IF (I.EQ.ILG) GO TO 540
IF (ILG.EQ.I-1) GO TO 530
ITOP = ITOP+1
IONDER(ITOP) = ILG
IBO(ITOP)= I-1
C
530 IF (I.EQ.IBG) GO TO 550
540 IF (I+1.EQ.IBG) GO TO 550
ITOP = ITOP+1
IONDER(ITOP) = I+1
IBO(ITOP) = IBG
550 GO TO 500
C
800 RETURN
END
C
SUBROUTINE SVAR(N,R,NP,VAR)
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION R(N)
VAR=0.0
DO 10 I=1,N
10 VAR=VAR+R(I)**2.0
VAR=VAR/DFLOAT(N-NP)
RETURN
END
SUBROUTINE SVARQ(Q,N,R,NP,VARQ)
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION R(N)
SUMQ=0.0
AN=DFLOAT(N)*Q
NQ=DINT(AN)
DO 20 I=NQ+1,N-NQ
20 SUMQ=SUMQ+R(I)**2.0
BOP1=(DFLOAT(N-NP))**(-1)
A=(1.0-Q*2.0)**(-2)
VARQ=A*((BOP1*SUMQ) +Q*(R(NQ+1)**2.0+R(N-NQ)**2.0))
RETURN
END
SUBROUTINE FSTOCK(SHARE,FSHARE)

```

```

CHARACTER*9 SHARE
CHARACTER*10 FSHARE
  CHARACTER*1 EXT
  IF(SHARE(9:9).EQ.' ') THEN
    FSHARE=SHARE
    GOTO 100
  ELSE
    EXT=SHARE(9:9)
    FSHARE=SHARE(1:8)//'.'//EXT
  ENDIF
100 CONTINUE
RETURN
END
SUBROUTINE MEDFIT(X,Y,NDATA,A,B,ABDEV)
  PARAMETER (NMAX=2000)
  EXTERNAL ROFUNC
  COMMON /ARRAYS/ NDATA,XT(NMAX),YT(NMAX),ARR(NMAX),AA,ABDEVT
  REAL*8 X(NDATA),Y(NDATA)
  REAL*8 A,B,ABDEV
  SX=0.
  SY=0.
  SXY=0.
  SXX=0.
  DO 11 J=1,NDATA
    XT(J)=X(J)
    YT(J)=Y(J)
    SX=SX+X(J)
    SY=SY+Y(J)
    SXY=SXY+X(J)*Y(J)
    SXX=SXX+X(J)**2
11 CONTINUE
  NDATA=NDATA
  DEL=NDATA*SXX-SX**2
  AA=(SXX*SY-SX*SXY)/DEL
  BB=(NDATA*SXY-SX*SY)/DEL
  CHISQ=0.
  DO 12 J=1,NDATA
    CHISQ=CHISQ+(Y(J)-(AA+BB*X(J)))**2
12 CONTINUE
  SIGB=SQRT(CHISQ/DEL)

```

```

B1=BB
F1=ROFUNC(B1)
B2=BB+SIGN(3.*SIGB,F1)
F2=ROFUNC(B2)
1 IF(F1*F2.GT.0.)THEN
  BB=2.*B2-B1
  B1=B2
  F1=F2
  B2=BB
  F2=ROFUNC(B2)
  GOTO 1
ENDIF
SIGB=0.01*SIGB
2 IF(ABS(B2-B1).GT.SIGB)THEN
  BB=0.5*(B1+B2)
  IF(BB.EQ.B1.OR.BB.EQ.B2)GOTO 3
  F=ROFUNC(BB)
  IF(F*F1.GE.0.)THEN
    F1=F
    B1=BB
  ELSE
    F2=F
    B2=BB
  ENDIF
  GOTO 2
ENDIF
3 A=AA
  B=BB
  ABDEV=ABDEVT/NDATA
  RETURN
END
FUNCTION ROFUNC(B)
PARAMETER (NMAX=2000)
COMMON /ARRAYS/ NDATA,X(NMAX),Y(NMAX),ARR(NMAX),AA,ABDEV
N1=NDATA+1
NML=N1/2
NMH=N1-NML
DO 11 J=1,NDATA
  ARR(J)=Y(J)-B*X(J)
11 CONTINUE

```

```
CALL SORT(NDATA,ARR)
AA=0.5*(ARR(NML)+ARR(NMH))
SUM=0.
ABDEV=0.
DO 12 J=1,NDATA
D=Y(J)-(B*X(J)+AA)
ABDEV=ABDEV+ABS(D)
SUM=SUM+X(J)*SIGN(1.0,D)
12 CONTINUE
ROFUNC=SUM
RETURN
END
SUBROUTINE SORT(N,RA)
REAL RA(N)
L=N/2+1
IR=N
10 CONTINUE
IF(L.GT.1)THEN
L=L-1
RRA=RA(L)
ELSE
RRA=RA(IR)
RA(IR)=RA(1)
IR=IR-1
IF(IR.EQ.1)THEN
RA(1)=RRA
RETURN
ENDIF
ENDIF
I=L
J=L+L
20 IF(J.LE.IR)THEN
IF(J.LT.IR)THEN
IF(RA(J).LT.RA(J+1))J=J+1
ENDIF
IF(RRA.LT.RA(J))THEN
RA(I)=RA(J)
I=J
J=J+J
ELSE
```

```

J=IR+1
ENDIF
GO TO 20
ENDIF
RA(I)=RRA
GO TO 10
END
C
C
C
C *****
C BARRODALE & ROBERTS ALGORITHM FOR L1 PROBLEM
C *****
C
C
C
SUBROUTINE L1(Y1,X1,M,N,TOLER,X,E,NUNIQ,GEWIG,O)
IMPLICIT REAL*8(A-H,O-Z)
! COMMON X1,Y1
DIMENSION X1(M,N),Y1(M),GEWIG(M)
REAL*8 MIN,MAX,X(N),E(M)
DIMENSION A(M+2,6),B(M)
INTEGER OUT,S(M)
LOGICAL STAGE,TEST
DATA BIG/1.E37/
M2=M+2
N2=N+2
DO 9 I=1,M
DO 8 J=1,N
A(I,J)=GEWIG(I)*X1(I,J)
8 CONTINUE
B(I)=GEWIG(I)*Y1(I)
9 CONTINUE
M1=M+1
N1=N+1
DO 10 J=1,N
A(M2,J)=J
X(J)=0.
10 CONTINUE
DO 40 I=1,M

```

```
A(I,N2)=N+I
A(I,N1)=B(I)
IF(B(I).GE.0.) GO TO 30
DO 20 J=1,N2
A(I,J)=-A(I,J)
20 CONTINUE
30 E(I)=0.
40 CONTINUE
DO 60 J=1,N1
SUM=0.DO
DO 50 I=1,M
F=1.
IF(A(I,N2).LT.0.)F=-1.
SUM=SUM+A(I,J)*(2.*0*F-F+1.)
50 CONTINUE
A(M1,J)=SUM
60 CONTINUE
STAGE=.TRUE.
KOUNT=0
KR=1
KL=1
70 MAX=-1.
DO 80 J=KR,N
IF(DABS(A(M2,J)).GT.N) GO TO 80
D=DABS(A(M1,J))
IF(D.LE.MAX) GO TO 80
MAX=D
IN=J
80 CONTINUE
IF(A(M1,IN).GE.0.) GO TO 100
DO 90 I=1,M2
A(I,IN)=-A(I,IN)
90 CONTINUE
100 K=0
DO 110 I=KL,M
D=A(I,IN)
IF(D.LE.TOLER) GO TO 110
K=K+1
B(K)=A(I,N1)/D
S(K)=I
```

```
TEST=.TRUE.
110 CONTINUE
120 IF(K.GT.0) GO TO 130
TEST=.FALSE.
GO TO 150
130 MIN=BIG
DO 140 I=1,K
IF(B(I).GE.MIN) GO TO 140
J=I
MIN=B(I)
OUT=S(I)
140 CONTINUE
B(J)=B(K)
S(J)=S(K)
K=K-1
150 IF(TEST.OR. .NOT.STAGE) GO TO 170
DO 160 I=1,M2
D=A(I,KR)
A(I,KR)=A(I,IN)
A(I,IN)=D
160 CONTINUE
KR=KR+1
GO TO 260
170 IF(TEST) GO TO 180
A(M2,N1)=2.
GO TO 350
180 PIVOT=A(OUT,IN)
IF(A(M1,IN)-PIVOT-PIVOT.LE.TOLER) GO TO 200
DO 190 J=KR,N1
D=A(OUT,J)
A(M1,J)=A(M1,J)-D-D
A(OUT,J)=-D
190 CONTINUE
A(OUT,N2)=-A(OUT,N2)
GO TO 120
200 DO 210 J=KR,N1
IF(J.EQ.IN) GO TO 210
A(OUT,J)=A(OUT,J)/PIVOT
210 CONTINUE
DO 230 I=1,M1
```

```
IF(I.EQ.OUT) GO TO 230
D=A(I,IN)
DO 220 J=KR,N1
IF(J.EQ.IN) GO TO 220
A(I,J)=A(I,J)-D*A(OUT,J)
220 CONTINUE
230 CONTINUE
DO 240 I=1,M1
IF(I.EQ.OUT) GO TO 240
A(I,IN)=-A(I,IN)/PIVOT
240 CONTINUE
A(OUT,IN)=1./PIVOT
D=A(OUT,N2)
A(OUT,N2)=A(M2,IN)
A(M2,IN)=D
KOUNT=KOUNT+1

IF(.NOT.STAGE) GO TO 270
KL=KL+1
DO 250 J=KR,N2
D=A(OUT,J)
A(OUT,J)=A(KOUNT,J)
A(KOUNT,J)=D
250 CONTINUE
260 IF(KOUNT+KR.NE.N1) GO TO 70
STAGE=.FALSE.
270 MAX=-BIG
DO 290 J=KR,N
D=A(M1,J)
IF(D.GE.0.) GO TO 280
IF(D.GT.(-2.)) GO TO 290
D=-D-2.
280 IF(D.LE.MAX) GO TO 290
MAX=D
IN=J
290 CONTINUE
IF(MAX.LE.TOLER) GO TO 310
IF(A(M1,IN).GT.0.) GO TO 100
DO 300 I=1,M2
A(I,IN)=-A(I,IN)
```

```

300 CONTINUE
A(M1,IN)=A(M1,IN) - 2.
GO TO 100
310 L=KL-1
DO 330 I=1,L
C
C
C
C RANK
C
C
C
SUBROUTINE RANK(N,RANG,ZAT,NR, SORT)
C
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION RANG(NR),ZAT(NR),SORT(NR),BANG(NR)
NTERT=0
DO 40 I=1,N
IF(I.LE.NTERT)GO TO 40
NTEL=1
SOM=0.0
DO 10 J=I+1,N
IF(SORT(I).NE.SORT(J))GO TO 20
NTEL=NTEL+1
SOM=DFLOAT(J)+SOM
10 CONTINUE
20 SOM=SOM+DFLOAT(I)
SSS=SOM/NTEL
DO 30 K=I,NTEL+I-1
BANG(K)=SSS
30 CONTINUE
NTERT=I+NTEL-1
40 CONTINUE
DO 50 I=1,N
DO 50 J=1,N
IF(ZAT(I).EQ.SORT(J))RANG(I)=BANG(J)
50 CONTINUE
RETURN
END
SUBROUTINE SKECUR(N,SKEW,CURT,Z1,Z2,CHI)

```

```

REAL*8 SKEW, CURT, Z1, Z2, CHI
Z1=SKEW/DSQRT(6.0/DFLOAT(N))
Z2=(CURT-3.0)/DSQRT(24.0/DFLOAT(N))
CHI=Z1**2+Z2**2
RETURN
END
SUBROUTINE COVRET(NR,NC,N,NP,X,XBAR,SIGMA)
REAL*8 X(NR,NC),XBAR(NC),SIGMA(NC,NC)
REAL*8 MAT(NR,NC)
DO 6000 J=1,NP
XBAR(J)=0.0
DO 605 I=1,N
XBAR(J)=XBAR(J)+X(I,J)
605 CONTINUE
XBAR(J)=XBAR(J)/DFLOAT(N)
6000 CONTINUE
DO 6010 I=1,N
DO 6020 J=1,NP
MAT(I,J)=X(I,J)-XBAR(J)
6020 CONTINUE
6010 CONTINUE
DO 20 I=1,NP
DO 20 J=1,NP
SOM=0.0
DO 10 K=1,N
SOM =SOM+MAT(K,I)*MAT(K,J)
10 CONTINUE
SIGMA(I,J)=SOM/DFLOAT(N-1)
20 CONTINUE
RETURN
END
SUBROUTINE INNOV(NR,NC,N,NSHA,VINDEX,RESMAT,BETA,PSIMAT,DIAMAT)
REAL*8 RESMAT(NR,NC),BETA(NC),PSIMAT(NC,NC),DIAMAT(NC,NC)
REAL*8 SIGRES(NC,NC),BETMAT(NC,NC),VINDEX
DO 2220 I=1,NSHA
DO 2220 J=1,NSHA
SOM=0.0
DO 2210 K=1,N
SOM =SOM+RESMAT(K,I)*RESMAT(K,J)
2210 CONTINUE

```

```

    SIGRES(I,J)=SOM/DFLOAT(N-2)
2220 CONTINUE
    DO 5223 II=1,NSHA
5223 WRITE(*,2224)(SIGRES(II,JJ),JJ=1,NSHA)
2224 FORMAT(6F12.9)
    PAUSE
    DO 2225 II=1,NSHA
    DO 2226 JJ=1,NSHA
        BETMAT(II,JJ)=BETA(II)*BETA(JJ)
2226 CONTINUE
2225 CONTINUE
    DO 2227 II=1,NSHA
2227 WRITE(*,2224)(BETMAT(II,JJ),JJ=1,NSHA)
    DO 4441 II=1,NSHA
    DO 4442 JJ=1,NSHA
        PSIMAT(II,JJ)=VINDEXT*BETMAT(II,JJ)
        DIAMAT(II,JJ)=PSIMAT(II,JJ)
4442 CONTINUE
4441 CONTINUE
    DO 4445 II=1,NSHA
    DO 4446 JJ=1,NSHA
        PSIMAT(II,JJ)=PSIMAT(II,JJ)+SIGRES(II,JJ)
4446 CONTINUE
4445 CONTINUE
    DO 4447 II=1,NSHA
        DIAMAT(II,II)=DIAMAT(II,II)+SIGRES(II,II)
4447 CONTINUE
    RETURN
    END

```

Sharpe Multiple Index OLS model.

```

!DECLARE THE USE OF THE INTERNATIONAL MATHEMATICS AND STATISTICS LIBRARY
USE MSIMSL

```

```

!DECLARE ALL THE VARIABLES THAT ARE TO BE USED IN THE PROGRAM
!ALL ARRAYS AND MATRICES ARE DYNAMICALLY ASSIGNED WITH SIZES
!TO BE ALLOCATED LATER
REAL, DIMENSION (:,:),ALLOCATABLE ::RAWDATA, &

```

```

LOG_RETURNS,ZEROCOV,STATS,MNMX
REAL, DIMENSION (:,:),ALLOCATABLE :: RET_ERROR,CONSTR, &
CONSTR,HESSIAN,PHI,BETAMAT,PROD,PRXCOV
REAL, DIMENSION (:,:),ALLOCATABLE :: IDP,COV,RSQ,ESTATS
REAL, DIMENSION (:),ALLOCATABLE :: EXPRET,IND_MEAN,SHARE_ERROR
REAL, DIMENSION (:),ALLOCATABLE :: PRXMN,RET_VEC,RHS,W,ALAMDA,ERROR_MEAN

REAL, DIMENSION (:),ALLOCATABLE :: SHARE,BETAVEC
INTEGER,DIMENSION (:), ALLOCATABLE :: IACT
DIMENSION INCD(1),SIGMA(501),EFFRET(501)
REAL MX,MN,JUMP,ADJUS
REAL DIAG,SUMWT,SST,SSE
INTEGER NACT,NR,NC,NVAR,DATE,NRMISS

!NUMBER OF ROWS IN THE DATA SET
NR=150

!NUMBER OF COLUMNS IN THE DATA SET
NC=14

!NUMBER OF EXPLANATORY VARIABLES IN THE DATA SET
NVAR=9

!ASSIGN DIMENSIONS TO DYNAMIC ARRAYS AND MARICES
ALLOCATE(RAWDATA(NR,NC),LOG_RETURNS(NR-1,NC),RSQ(NVAR,2), &
COV(NVAR,NVAR),ZEROCOV(NVAR,NVAR),PROD(NR-NVAR,NVAR))
ALLOCATE(IDP((NR-1),(NC-NVAR)),PRXCOV((NC-NVAR),(NC-NVAR)), &
PHI(NVAR,NVAR),CONSTR((2+(NVAR*2)),NVAR))
ALLOCATE(RET_ERROR(NR-1,NVAR),HESSIAN(NVAR,NVAR), &
BETAMAT(NVAR,(NC-NVAR+1)),STATS(15,NC),ESTATS(15,1),MNMX(15,1))
ALLOCATE(RHS(2+(NVAR*2)),W(NVAR),EXPRET(NVAR), &
PRXMN(NC-NVAR-DATE),ERROR_MEAN(NVAR),SHARE(NR-1))
ALLOCATE(ALAMDA(NVAR),BETAVEC(NC-NVAR+1),RET_VEC(NVAR), &
IND_MEAN(NC-NVAR),SHARE_ERROR(NR-1))
ALLOCATE(IACT(NVAR))

!OPEN THE DATA FILE
OPEN(4,FILE='C:\Data.txt')

!READ IN THE VALUES OF THE DATA FILE

```

```

DO I=1,NR
  READ(4,*)(RAWDATA(I,J),J=1,NC)
ENDDO

!COMPUTE THE LOG RETURNS OF THE DATA
DO I=1,NR-1
  DO J=1,NC
    LOG_RETURNS(I,J)=(LOG(RAWDATA(I+1,J)) -LOG(RAWDATA(I,J)))
  ENDDO
ENDDO

!READ IN THE DATA RELATING TO THE INDICES
DO I=1,NR-1
  DO J=1,NC-NVAR
    IDP(I,J)=LOG_RETURNS(I,J+NVAR)
  ENDDO
ENDDO

!CALCULATE AND STORE REGRESSION COEFFICIENTS
DO I=1,NVAR

  !READ THE DATA RELATING TO AN INDIVIDUAL SHARE INTO AN ARRAY
  DO J=1,(NR-1)
    SHARE(J)=LOG_RETURNS(J,I)
  ENDDO

  !COMPUTE THE COEFFICIENTS OF THE MULTIPLE REGRESSION
  !FOR THE SHARE IN THE ABOVE MATRIX
  CALL RLSE((NR-1),SHARE,NC-NVAR,IDP,(NR-1),1,BETAVEC,SST,SSE)

  !READ IN THE REGRESSION COEFFICIENTS
  !FOR THE ABOVE SHARE INTO A MATRIX
  DO J=1,(NC-NVAR+1)
    BETAMAT(I,J)=BETAVEC(J)
  ENDDO

  !CALCULATE THE R SQUARED FIGURE FOR THE REGRESSION
  RSQ(I,2)=1-(SSE/SST)

ENDDO

```

```

!CALCULATE SUMMARY STATS
CALL UVSTA(0,NR-1,NC,LOG_RETURNS,NR-1,0,0,0, &
          95.0,95.0,0,STATS,15,NRMISS)

!CALCULATE THE MEAN OF EACH OF THE INDICES
DO I=1,NC-NVAR
  IND_MEAN(I)=STATS(1,NVAR+I)
ENDDO

!CALCULATE THE PREDICTED RETURN FOR EACH OF THE
!SHARES USING THE REGRESSION ABOVE
DO I=1,NVAR
  EXPRET(I)=0
  DO J=1,NC-NVAR
    EXPRET(I)=EXPRET(I)+(BETAMAT(I,(J+1))*IND_MEAN(J))
  ENDDO
  EXPRET(I)=EXPRET(I)+BETAMAT(I,1)
ENDDO

!CALCULATE THE RESIDUALS FOR EACH OBSERVATION OF EACH SHARE
DO I=1,NR-1
  DO J=1,NVAR
    RET_ERROR(I,J)=LOG_RETURNS(I,J)-EXPRET(J)
  ENDDO
ENDDO

DO I=1,NVAR
  !READ IN THE RESIDUAL ERROR VALUES INTO A MATRIX
  DO J=1,NR-1
    SHARE_ERROR(J)=RET_ERROR(J,I)
  ENDDO

!CALCULATE THE RESIDUAL VARIANCE FOR EACH SHARE
CALL UVSTA(0,NVAR,1,SHARE_ERROR,NVAR,0,0,0,95.0, &
          95.0,0,ESTATS,15,NRMISS)
RSQ(I,1)=ESTATS(2,1)

```

```

ENDDO

!CALCULATE THE COVARIANCE MATRIX OF THE RESIDUAL OF THE SHARES
CALL CORVC(0,NR-1,NVAR,RET_ERROR,NR-1,0,0,0,0,ERROR_MEAN, &
          COV,NVAR,INCD,1,NOBS,NMISS,SUMWT)
CALL UVSTA(0,NVAR,1,SHARE_ERROR,NVAR,0,0,0,95.0, &
          95.0,0,ESTATS,15,NRMISS)
CALL CORVC(0,NR-1,(NC-NVAR-DATE),IDP,NR-1,0,0,0,0,PRXMN, &
          PRXCOV,(NC-NVAR-DATE),INCD,1,NOBS,NMISS,SUMWT)
!CALCULATE THE THE COVARINCE MATRIX BETWEEN SHARES
CALL MXYTF((NC-NVAR-DATE),(NC-NVAR-DATE),PRXCOV,(NC-NVAR-DATE),NVAR,
&
(NC-NVAR-DATE),BETAMAT,NVAR,(NC-NVAR-DATE),NVAR,PROD,(NC-NVAR-DATE))
CALL MRRRR(NVAR,(NC-NVAR-DATE),BETAMAT,NVAR,(NC-NVAR-DATE), &
NVAR,PROD,(NC-NVAR-DATE),NVAR,NVAR,PHI,NVAR)

DO I=1,NVAR
DO J=1,NVAR
  ZEROCOV(I,J)=0
ENDDO
ZEROCOV(I,I)=COV(I,I)
ENDDO

COVTYP=1

IF (COVTYP == 0) THEN
DO I=1,NVAR
DO J=1,NVAR
  PHI(I,J)=PHI(I,J)+ZEROCOV(I,J)
ENDDO
ENDDO
ELSE
DO I=1,NVAR
DO J=1,NVAR
  PHI(I,J)=PHI(I,J)+COV(I,J)
ENDDO
ENDDO
END IF

!CONVERT THE COVARIANCE MATRIX TO THE FORMAT REQUIRED FOR

```

```

!THE QUADTRATIC OPTIMISER
DO I=1,NVAR
  DO J=1,NVAR
    HESSIAN(I,J)=2*PHI(I,J)
  ENDDO
ENDDO

!READ IN VALUES FOR THE CONSTRAINTS
DO I=1,((NVAR*2)+2)
  DO J=1,NVAR
    CONSTR(I,J)=0
  ENDDO
ENDDO

DO I=1,NVAR
  CONSTR(1,I)=EXPRET(I)
  CONSTR(2,I)=1
  CONSTR(I+2,I)=1
  CONSTR(I+2+NVAR,I)=-1
  RHS(I+2)=0
  RHS(I+2+NVAR)=-1
  RET_VEC(I)=0
ENDDO

RHS(2)=1

!CALCULATE MIN AND MAX RETURNS
CALL UVSTA(0,NVAR,1,EXPRET,NVAR,0,0,0,95.0, &
  95.0,0,MNMX,15,NRMISS)
MN=MNMX(6,1)
MX=MNMX(7,1)

!CALCUALTE THE RETURN ITERATION SIZE FOR THE
!GENERATION OF THE EFFICIENT FRONTIER
JUMP=(MX-MN-0.0002)/500

!LOOP OF THE QUADRATIC OPTIMISATION ROUTINE FOR THE
!GENERATION OF THE EFFICIENT FRONTIER
DO K=0,500

```

```

RHS(1)=(JUMP*K)+MN +0.0001
CALL QPROG(NVAR, (NVAR*2)+2, 2, CONSTR, (NVAR*2)+2, RHS, &
          RET_VEC, HESSIAN, NVAR, DIAG, W, NACT, IACT, ALAMDA)
SIGMA(K+1)=BLINF(NVAR, NVAR, PHI, NVAR, W, W)
EFFRET(K+1)=(JUMP*K)+MN +0.0001
ENDDO

```

```

!WRITE EFFICIENT FRONTIER DATA TO AN OUTPUT FILE
OPEN(7, FILE='C:\MSC\MSCDATA1.txt')

```

```

DO I=1, 501
  WRITE(7, *) SIGMA(I), ' ', ' ', EFFRET(I)
ENDDO

```

```

END

```

Sharpe Multiple Index L_1 and L_p model.

```

!DECLARE THE USE OF THE INTERNATIONAL MATHEMATICS AND STATISTICS LIBRARY
USE MSIMSL

```

```

!DECLARE ALL THE VARIABLES THAT ARE TO BE USED IN THE PROGRAM
!ALL ARRAYS AND MATRICES ARE DYNAMICALLY ASSIGNED WITH SIZES
!TO BE ALLOCATED LATER

```

```

REAL, DIMENSION (:, :), ALLOCATABLE :: HESSIAN, RAWDATA, &
  LOG_RETURNS, COV, QR_DECOMP, LP_COEFF_MAT, STATS
REAL, DIMENSION (:, :), ALLOCATABLE :: IDP, PRXCOV, PROD, &
  PHI, CONSTR, ZERO_COV, LP_INP, RESID_MAT, MNMX
REAL, DIMENSION (:), ALLOCATABLE :: PRXMN, W, EXPRET, &
  SHARE, ERROR_MEAN, SHARE_RET, RESID
REAL, DIMENSION (:), ALLOCATABLE :: RET_VEC, IND_MEAN, &
  RHS, ALAMDA, P_VEC, LP_COEFF
INTEGER, DIMENSION (:), ALLOCATABLE :: IACT
DIMENSION INCD(1), SIGMA(501), EFFRET(501), INDIND(1)
REAL DIAG, MX, MN, JUMP, P_VAL, DFE, SCALE2, ELP
INTEGER NACT, NR, NC, NVAR, ITER, NRMIS, IRANK, SNRMIS

```

```

! NUMBER OF ROWS IN THE DATA SET
NR=150

```

```
! NUMBER OF COLUMNS IN THE DATA SET
NC=14

! NUMBER OF EXPLANATORY VARIABLES IN THE DATA SET
NVAR=9

! ASSIGN DIMENSIONS TO DYNAMIC ARRAYS AND MATRICES
ALLOCATE(RAWDATA(NR,NC), LOG_RETURNS(NR-1,NC), COV(NVAR,NVAR), &
        ZERO_COV(NVAR,NVAR), PROD(NR-NVAR,NVAR))
ALLOCATE(IDP(NR-1,(NC-NVAR)), PRXCOV((NC-NVAR),(NC-NVAR)), &
        PHI(NVAR,NVAR), STATS(15,NC))
ALLOCATE(SHARE_RET(NR-1), RESID(NR-1), P_VEC(NC-NVAR), &
        LP_COEFF(NC-NVAR+1), MNMX(15,1))
ALLOCATE(LP_INP(NR-1,NC-NVAR+1), RESID_MAT(NR-1,NVAR), &
        QR_DECOMP(NC-NVAR+1,NC-NVAR+1))
ALLOCATE(HESSIAN(NVAR,NVAR), CONSTR((2+(NVAR*2)),NVAR), &
        LP_COEFF_MAT(NVAR,NC-NVAR+1))
ALLOCATE(RHS(2+(NVAR*2)), W(NVAR), EXPRET(NVAR), PRXMN(NC-NVAR), &
        ERROR_MEAN(NVAR), SHARE(NR-1))
ALLOCATE(ALAMDA(NVAR), RET_VEC(NVAR), IND_MEAN(NC-NVAR))
ALLOCATE(IACT(NVAR))

! OPEN THE DATA FILE
OPEN(4,FILE='C:\MSC\Stock Prices\MMIdata20010908.txt')

! READ IN THE VALUES OF THE DATA FILE
DO I=1,NR

    READ(4,*)(RAWDATA(I,J),J=1,NC)

ENDDO

! COMPUTE THE LOG RETURNS OF THE DATA
DO I=1,NR-1

    DO J=1,NC
```

```
LOG_RETURNS(I,J)=(LOG(RAWDATA(I+1,J)) -LOG(RAWDATA(I,J)))

ENDDO

ENDDO

!COMPUTE SUMMARYY STATS OF THE DATA
CALL UVSTA(0,NR-1,NC,LOG_RETURNS,NR-1,0,0,0,95.0,95.0,0,STATS,15,SNRMISS)

! READ IN THE DATA RELATING TO THE INDICES INTO A SEPERATE MATRIX
DO I=1,NR-1

    DO J=1,NC-NVAR

        IDP(I,J)=LOG_RETURNS(I,J+NVAR)

    ENDDO

ENDDO

!ASSIGN P VALUES TO EACH SHARE FOR LP REGRESSION
!FOR THE L1 REGRESSION P_VEC(I)= 1 FOR I= 1 TO 9
P_VEC(1) = 1.281961
P_VEC(2) = 1.67309
P_VEC(3) = 1.1376
P_VEC(4) = 1.909679
P_VEC(5) = 1.865809
P_VEC(6) = 1.688961
P_VEC(7) = 1.660927
P_VEC(8) = 1.313803
P_VEC(9) = 1.543754

!READ IN THE EXPLANATORY VARIABLES FOR THE LP REGRESSION ROUTINE
DO I= 1,NR-1

    DO J=1,NC-NVAR

        LP_INP(I,J)=IDP(I,J)
```

```

ENDDO

ENDDO

!LOOP TO CALCULATE THE LP REGRESSION COEFFICIENTS FOR EACH OF THE SHARES

DO I = 1,NVAR

  !READ IN P VALUE FOR SHARE I
  P_VAL=P_VEC(I)

  !READ IN THE DEPENDENT VARIABLE FOR THE LP REGRESSION ROUTINE
  DO J=1,NR-1

    LP_INP(J,NC-NVAR+1)=LOG_RETURNS(J,I)

  ENDDO

  !RLLP ROUTINE TO CALCULATE THE LP REGRESSION COEFFICIENTS FOR SHARE
I
  CALL RLLP(NR-1,NC-NVAR+1,LP_INP,NR-1,1,-NC+NVAR,INDIND,NC-NVAR+1, &
    0,0,P_VAL,100*AMACH(4),100,100*AMACH(4),LP_COEFF,QR_DECOMP, &
    NC-NVAR+1,IRANK,DFE,RESID,SCALE2,ELP,ITER,NRMISS)

  !READ IN THE RESIDUALS FOR SHARE I INTO THE MATRIX OF RESIDUALS
  DO J=1,NR-1

    RESID_MAT(J,I)= RESID(J)

  ENDDO

  !READ IN THE COEFFICIENTS FOR SHARE I IN A MATRIX OF REGRESSION COEFFICIENTS
  DO J=1,NC-NVAR+1

    LP_COEFF_MAT(I,J)=LP_COEFF(J)

  ENDDO

  !END OF LP REGRESSION CALCULATION LOOP
ENDDO

```

```

!CALCULATE THE MEAN OF EACH OF THE INDICES
DO I=1,NC-NVAR

    IND_MEAN(I)=STATS(1,NVAR+I)

ENDDO

!CALCULATE THE PREDICTED RETURN FOR EACH OF THE SHARES
DO I=1,NVAR

    EXPRET(I)=0

    DO J=1,NC-NVAR
        EXPRET(I)=EXPRET(I)+(LP_COEFF_MAT(I,(J+1))*IND_MEAN(J))
    ENDDO

    EXPRET(I)=EXPRET(I)+LP_COEFF_MAT(I,1)

ENDDO

!CALCULATE THE COVARIANCE MATRIX OF THE RESIDUAL OF THE SHARES
CALL CORVC(0,NR-1,NVAR,RESID_MAT,NR-1,0,0,0,0,ERROR_MEAN,COV, &
           NVAR,INCD,1,NOBS,NMISS,SUMWT)
!CALCULATE THE THE COVARINCE MATRIX BETWEEN SHARES
CALL MXYTF((NC-NVAR),(NC-NVAR),PRXCOV,(NC-NVAR),NVAR,(NC-NVAR), &
           LP_COEFF_MAT,NVAR,(NC-NVAR),NVAR,PROD,(NC-NVAR))
CALL MRRRR(NVAR,(NC-NVAR),LP_COEFF_MAT,NVAR,(NC-NVAR),NVAR, &
           PROD,(NC-NVAR),NVAR,NVAR,PHI,NVAR)

DO I=1,NVAR
    DO J=1,NVAR
        ZEROCOV(I,J)=0
    ENDDO
    ZEROCOV(I,I)=COV(I,I)
ENDDO

```

```
!IF COVTYPE = 0 THEN COVARIANCE BETWEEN RESIDUALS ARE ASSUMED
!TO BE ZERO, OTHERWISE THEY ARE NONZERO
COVTYP=0

IF (COVTYP == 0) THEN
  DO I=1,NVAR
    DO J=1,NVAR
      PHI(I,J)=PHI(I,J)+ZEROCOV(I,J)
    ENDDO
  ENDDO
ELSE
  DO I=1,NVAR
    DO J=1,NVAR
      PHI(I,J)=PHI(I,J)+COV(I,J)
    ENDDO
  ENDDO
END IF

!CONVERT THE COVARIANCE MATRIX TO THE FORMAT REQUIRED FOR
!THE QUADTRATIC OPTIMISER
DO I=1,NVAR

  DO J=1,NVAR

    HESSIAN(I,J)=2*PHI(I,J)

  ENDDO

ENDDO

!READ IN VALUES FOR THE CONSTRIANTS
DO I=1,((NVAR*2)+2)
  DO J=1,NVAR

    CONSTR(I,J)=0

  ENDDO

ENDDO
```

```

DO I=1,NVAR

  CONSTR(1,I)=EXPRET(I)
  CONSTR(2,I)=1
  CONSTR(I+2,I)=1
  CONSTR(I+2+NVAR,I)=-1
  RHS(I+2)=-1
  RHS(I+2+NVAR)=-1
  RET_VEC(I)=0

ENDDO

RHS(2)=1

!CALCULATE MIN AND MAX RETURNS
CALL UVSTA(0,NVAR,1,EXPRET,NVAR,0,0,0,95.0,95.0,0,MNMX,15,NRMISS)

MX=MNMX(7,1)
MN=MNMX(6,1)

!CALCULATE THE RETURN ITERATION SIZE FOR THE GENERATION
!OF THE EFFICIENT FRONTIER
JUMP=(MX-MN-0.0002)/500
!LOOP OF THE QUADRATIC OPTIMISATION ROUTINE FOR THE GENERATION
!OF THE EFFICIENT FRONTIER
DO K=0,500

  RHS(1)=(JUMP*K)+MN+0.0001
  CALL QPROG(NVAR,(NVAR*2)+2,2,CONSTR,(NVAR*2)+2,RHS,RET_VEC, &
            HESSIAN,NVAR,DIAG,W,NACT,IACT,ALAMDA)
  SIGMA(K+1)=BLINF(NVAR,NVAR,PHI,NVAR,W,W)

  EFFRET(K+1)=(JUMP*K)+MN +0.0001

ENDDO

!WRITE EFFICIENT FRONTIER DATA TO AN OUTPUT FILE
OPEN(7,FILE='C:\MSC\MSCDATA1.txt')
```

```
DO I=1,501

  WRITE(7,*)SIGMA(I),'' ',EFFRET(I)

ENDDO
!WRITE COVARIANCE MATRIX TO AN OUTPUT FILE
OPEN(8,FILE='C:\MSCDATA2.txt')

DO I=1,NVAR

  WRITE(8,*)(PHI(I,J),J=1,NVAR)

ENDDO
!WRITE EXPECTED RETURNS TO AN OUTPUT FILE
OPEN(9,FILE='C:\MSCDATA3.txt')

DO I=1,NVAR

  WRITE(9,*)EXPRET(I)

ENDDO

END
```

University of Cape Town