

**LEAST ABSOLUTE DEVIATION ESTIMATION OF
MULTI-EQUATION LINEAR ECONOMETRIC MODELS**
A STUDY BASED ON MONTE CARLO EXPERIMENTS

MADHUCHHANDA DASGUPTA

DEPARTMENT OF ECONOMICS

**SUBMITTED
IN PARTIAL FULFILMENT OF THE REQUIREMENT OF
THE DEGREE OF DOCTOR OF PHILOSOPHY IN ECONOMICS**

OF

NORTH EASTERN HILL UNIVERSITY, SHILLONG

**NORTH-EASTERN HILL UNIVERSITY
DEPARTMENT OF ECONOMICS**

May 2004

I, Ms Madhuchhanda DasGupta, hereby declare that the subject matter of the thesis is the record of the work done by me, that the contents of this thesis did not form basis of the award of any previous degree to me or to the best of my knowledge to anybody else, and that the thesis has not been submitted by me for any research degree in any other University/Institute.

This is being submitted to the North-Eastern Hill University for the degree of Doctor of Philosophy in Economics.

(Madhuchhanda DasGupta)
Candidate

(N. Srivastav)
Professor & Head

(SK. Mishra)
Professor of Economics
Supervisor

Acknowledgements

It is my pleasure to acknowledge my heartfelt gratitude and indebtedness to my esteemed teacher and supervisor Prof. S. K. Mishra, Department of Economics, NEHU for his invaluable guidance, wise counsel and ungrudging help in every stage of my research work. I have no hesitation to record here that Prof. Mishra has not only aroused my interest in Econometrics but also motivated me to work on the present topic. My research work, indeed, would not have seen the light of the day, had I not received from Prof. Mishra constant advice, encouragement and also logistical support.

I take this opportunity to express my gratefulness to my revered teachers Prof. N. Srivastav, Head of the Dept. of Economics, NEHU, Prof. E.D. Thomas, Dean, School of Economics, Management & Information Sciences, NEHU, Dr. A.C. Dubey and Dr. B. Mishra for their encouragement and advice.

Thanks are also due to my friends Ms. Chandana Bhattacharjee, Ms. Sharmila Das Talukdar and my colleague Mrs. Mayashree B. Das for their help and moral support.

I owe a great debt of gratitude to my father who has been my pillar of strength and constant source of inspiration and my mother for her silent sacrifice to give me comfort during the course of my work.

Finally, my longstanding thanks are also due to my brother, Joy, for sustained encouragement in moments of despair. I am also grateful to my cousin Duke who has helped me immensely in various ways for the successful completion of the present work.

Shillong,

(Madhuchhanda DasGupta)

The May, 2004.

CONTENTS

Chapter	Page
Chapter 1 Introduction	1-15
I The Background	1
II Objectives of the Study	11
III The Methodology	12
IV Organization of the Work	14
Chapter 2 The Framework of Linear Econometric Models and their Estimation	16-36
I Introduction	16
II The General Simultaneous Equation Model	17
III Estimation of Single Equation Econometric Models	19
IV Estimation of Multi-equation or Simultaneous Equation Econometric Models	23
Chapter 3 LAD Estimation: A Literature Survey	37-89
I Introduction	37
II Survey of Literature on Properties and Algorithms of LAD Estimator	38
III Details of different Computational Algorithms for LAD Estimation	71
Chapter 4 The Methodology of Monte Carlo Experiments	90-108
I Introduction	90
II The Monte Carlo Method	91
III Random Numbers, their Distributions and Generation	96
Chapter 5 Performance of LAD in Estimation of Single-Equation Linear Models	110-137
I Introduction	110
II Specification of Single Equation Model	111
III Numerical Details	113
IV Details of Specification wise No. of Experiments	115
V Findings of Monte Carlo Experiments	116
Chapter 6 Performance of LAD in Estimation of Multi-Equation Linear Models	138-177
I Introduction	138
II The Steps in Monte Carlo Experiments	141
III Candidate Methods of Estimation (Candidate Estimators) for Monte Carlo Experiments	144

IV	Specification of Simultaneous – Equation Model	145
V	Numerical Details	147
VI	Details of Specification wise No. of Experiments	149
VII	Findings of Monte Carlo Experiments	151
Chapter 7	Summary and Concluding Remarks	178-198
I	Introduction	178
II	The Motivation to Investigation	178
III	The Methodology of Investigation and Findings	180
IV	Lines of Investigation in Future	196
Appendix 1	A Computer Program to Generate Random Numbers and Show their Frequency Distribution	199-201
Appendix 2	Source Codes of Computer Programs for Single Equation Models	202-213
Appendix 3	Source Codes of Computer Programs for Multi Equation Models	214-252
Appendix 4	Sporadic Errors in Explanatory Variables	253-267
Bibliography		268-276

LIST OF TABLES

Table	Tables in Chapter 5 - Performance of LAD in Estimation of Single- Equation Linear Models	Page
1.	Effect of Distribution	119
2.	Effect of Sample Size	119
3.	Effect of Model Size	119
4.	Effect of No. of Outliers	119
5.	Effect of Size of Outliers	120
6.	Effect of Distribution and Sample Size	121
7.	Effect of Sample Size and Model Size	121
8.	Effect of Sample Size and No. of Outliers	121
9.	Effect of Distribution and Model Size	121
10.	Effect of Distribution and No. of Outliers	122
11.	Effect of Distribution and Size of Outliers	122
12.	Effect of Sample Size and No. of Outliers	122
13.	Effect of Model Size and No. of Outliers	123
14.	Effect of Model Size and No. of Outliers	123
15.	Effect of Model Size and No. of Outliers	123
16.	A Comparative view of Relative RMS of LAD and LADRW Algorithms	125
17.	Summary Statistics Regarding LAD and LADRW Relative RMS	126
18.	Percentiles of LAD and LAD-RW Relative RMS	126
19.	Difference in LAD and LADRW Explained by Decision Variables	126
20.	Difference in LAD and LADRW Explained by Decision Variables	126
21.	Effect of Decision Variables on Relative (to OLS) Norm of LAD Estimator	128
22.	Effect of Decision Variables on Relative (to OLS) Norm of LAD-RW Estimator	128
23.	Effect of Decision Variables on Logarithm of Relative (to OLS) Norm of LAD Estimator	128
24.	Effect of Decision Variables on Logarithm of Relative (to OLS) Norm of LADRW Estimator	129
25.	BLAD Probit Regression	135
26.	BLADRW Probit Regression	135
27.	BLAD Probit Regression	136
28.	BLAD Probit Regression	136
29.	BLADRW Probit Regression	136
30.	Composit Probit Regression	136
31.	BLADRW Probit Regression	136
32.	Composit Probit Regression	136

Table	Tables in Chapter 6 - Performance of LAD in Estimation of Multi- Equation Linear Models	Page
Table 1.	Effect of Model Size	152
Table 2.	Effect of Sample Size	152
Table 3.	Effect of Distribution	152
Table 4.	Effect of Standard Deviation	153
Table 5.	Effect of No. of Outliers	153
Table 6.	Effect of Size of Outliers	153
Table 7.	Effect of Model Size and Sample Size	155
Table 8a.	Effect of Model Size and Error Distribution	156
Table 8b.	Effect of Model Size and Error Distribution	156
Table 9.	Effect of Model Size and SD Size	156
Table 10a.	Effect of Model Size and No. of Outliers	156
Table 10b.	Effect of Model Size and No. of Outliers	157
Table 11a.	Effect of Model Size and Outlier Size	157
Table 11b.	Effect of Model Size and Outlier Size	157
Table 11c.	Effect of Model Size and Outlier Size	157
Table 12a.	Effect of Sample Size and Distribution	158
Table 12b.	Effect of Sample Size and Distribution	158
Table 13.	Effect of Sample Size and SD Size	158
Table 14a.	Effect of Sample Size and No. of Outliers	158
Table 14b.	Effect of Sample Size and No. of Outliers	159
Table 15a.	Effect of Sample Size and Outlier Size	159
Table 15b.	Effect of Sample Size and Outlier Size	159
Table 15c.	Effect of Sample Size and Outlier Size	159
Table 16.	Effect of Distribution and SD Size	160
Table 17a.	Effect of Distribution and No. of Outliers	160
Table 17b.	Effect of Distribution and No. of Outliers	160
Table 18a.	Effect of Distribution and Outlier Size	161
Table 18b.	Effect of Distribution and Outlier Size	161
Table 18c.	Effect of Distribution and Outlier Size	161
Table 19a.	Effect of No. of Outliers and Outlier Size	162
Table 19b.	Effect of No. of Outliers and Outlier Size	162
Table 19c.	Effect of No. of Outliers and Outlier Size: LAD-GIL	162
Table 20.	Effect of Outlier Size and SD Size	163
Table 21.	Effect of No. of Outliers and SD Size	163
Table 22.	Results of Probit Analysis on LS-LAD Model – 1	167
Table 23.	Results of Probit Analysis on LS-LAD Model-2	167
Table 24.	Results of Probit Analysis on LS-LAD Model-3	168
Table 25.	Results of Probit Analysis on LAD- LS Model-1	168
Table 26.	Results of Probit Analysis on LAD- LS Model-2	168
Table 27.	Results of Probit Analysis on LAD- LS Model-3	168
Table 28.	Results of Probit Analysis on LS-GILN Model-1	168
Table 29.	Results of Probit Analysis on LS-GILN Model-2	169
Table 30.	Results of Probit Analysis on LS-GILN Model-3	169
Table 31.	Results of Probit Analysis on LS-GILN Model-4	169
Table 32.	Results of Probit Analysis on LAD-GILN Model-1	170

Table 33.	Results of Probit Analysis on LAD-GILN Model-2	170
Table 34.	Results of Probit Analysis on LAD-GILN Model-3	170
Table 35.	Results of Probit Analysis on LAD-GILN Model-4	171
Table 36.	Results of Probit Analysis on LAD-GILN Model-5	171
Table 37.	Results of Probit Analysis on LAD-GILN Model-6	171
Table 38.	Results of Probit Analysis on LAD-GILN Model-7	171
Table 39.	A GIST of Results of Probit Analysis on LAD-GILN Models	172
Table 40.	Probit Regression Models Estimates* (Coefficients) of Regression Parameters	172
Table 41.	Results of Probit Analysis on LAD-LAD Model-1	173
Table 42.	Results of Probit Analysis on LAD-LAD Model-2	173
Table 43.	Results of Probit Analysis on LAD-LAD Model-3	173
Table 44.	Results of Probit Analysis on LAD-LAD Model-4	174
Table 45.	Results of Probit Analysis on LAD-LAD Model-5	174
Table 46.	Results of Probit Analysis on LAD-LAD Model-6	174
Table 47.	Results of Probit Analysis on LAD-LAD Model-7	175
Table 48.	Results of Probit Analysis on LAD-LAD Model-8	175
Table 49.	Results of Probit Analysis on LAD-LAD Model-9	175
Table 50.	Results of Probit Analysis on LAD-LAD Model-10	176
Table 51.	Results of Probit Analysis on LAD-LAD Model-11	176
Table 52.	A Gist of Results of Probit Analysis on LAD-LAD Models	176
Table 53.	A Summary of Performance of Estimators and Explanatory Variables Evaluated by Probit Regression Model	177

DasGupta, Madhuchhanda (2004) LEAST ABSOLUTE DEVIATION
ESTIMATION OF MULT-EQUATION LINEAR ECONOMETRIC MODELS : A
STUDY BASED ON MONTE CARLO EXPERIMENTS, Doctoral Dissertation,
Dept. of Economics, North Eastern Hill University, Shillong (India);
Supervisor – Prof. SK Mishra

CHAPTER 1

INTRODUCTION

I. The Background: Every *real empirical or experiential science* (against the ideal science such as mathematics or logic which derives implications from a closed system of axioms) is an organized description of empirical inter-relations among the variables of the system of its concern. For example, physics investigates into the relationship between electrical conductivity of a metallic wire and its length and thickness, and other such relations; chemistry purports to establish the relationship between the rate of various types of chemical reactions and the temperature and pressure that the reactants are subject to, and so on; botany studies the relationship between the growth of the biomass of a species of plants and the nutritional contents of the soil and the physico-chemical properties of the environment that support plant life, and such issues. Similarly, economics is concerned with the relationship between consumption expenditure of households on a specific commodity, income of the household and its size and the price of the commodity, and the likes. On constructive thinking and empirical evidence each branch of *real science* describes such relationships of its concern and puts them in an organic order to make an ever-growing edifice of its literature.

Real sciences originate from and feed on the empirical experiences accumulated either by observation or by experimentation. Growth of a *real science* is squarely based on constructive thinking that suggests the scientists to gather information on relevant variables and the suitable methods to process the information so accumulated to establish the relationship among them. Therefore, a *real science* progresses with a hypothesis in

which the values of a dependent variable is associated with the values of a single or several explanatory variables and it is held that variations in (the value of) the dependent variable is associated with the variations in (the values of) the explanatory variables in a particular manner. Here ‘variation’ is very crucial. It is the cornerstone of all empirical investigations. All empirical knowledge emanate from the analysis of variation. Therefore, the scientist collects information on the dependent variable (say, y) and the explanatory variables (say, x_1, x_2, x_3) taking on several values. Our set of observations may thus be described as:

$$\begin{bmatrix} y_1, x_{11}, x_{12}, x_{13} \\ y_2, x_{21}, x_{22}, x_{23} \\ y_3, x_{31}, x_{32}, x_{33} \\ \dots, \dots, \dots, \dots \\ y_n, x_{n1}, x_{n2}, x_{n3} \end{bmatrix} \quad \dots (1.1).$$

The scientist holds or conjectures that $y = a_0 + a_1x_1 + a_2x_2 + a_3x_3$ (where a_j for $j=0, 1, 2$ and 3 are non-zero numerical constants) is the most possible relationship between y and x 's. However, the correct relationship could be $v = f(\chi_1, \chi_2, \chi_4, \chi_5, \chi_6, \omega_1, \omega_2, \zeta_1, \zeta_2)$, where $f(\cdot)$ is a nonlinear function. In the scientist's scheme, y, x_1 and x_2 are the most appropriate representatives that measure v, χ_1 and χ_2 respectively. One must note that all hypotheses in a *real science* relate concepts and only few concepts can be perfectly measured by a number (denial of strict *arithmomorphism* as argued by **N. Georgescu-Roegen**, 1971). Therefore, empirically obtained numerical measurements (data) on any conceptual entity is only a representative of that entity. Now, the scientist has mis-specified the correct relationship by deviating from the fact in many possible ways; first, he has not been able to measure the concepts perfectly accurately – he is working only

with the best possible numerical measures of the concepts to be correlated; secondly he has not been able to incorporate all relevant explanatory variables that explain y – he has failed to include representative variables that measure $\chi_4, \chi_5, \chi_6, \omega_1, \omega_2, \zeta_1$ and ζ_2 . Thirdly, he has included x_3 among the explanatory variables while it is none of them in fact; and fourthly (may not be lastly), he has thought of $y = f(\cdot)$ a linear function of x while in fact it is not. It is also not unlikely that he has exchanged a dependent variable for an explanatory variable or that his dependent variable and one of the explanatory variables are co-effects of the rest of the explanatory variables. It is also plausible that explaining $y = f(x)$ is incomplete: $y = f(x)$ and $z = \psi(x, q)$ jointly would be more appropriate. And all these do not exhaust the list of mis-specifications.

The scientist should therefore compensate for all such possible errors by adding an error (disturbance or residual) term to his hypothesis and express his model as:

$$y = a_0 + a_1x_1 + a_2x_2 + a_3x_3 + e \quad \dots \quad (1.2)$$

where the last term captures the effects of all mis-specifications committed by him.

Since the error is due to the deviation of the explicitly expressed model and the true relationship and there is no one-to-one correspondence between the scientist's committed specification and that omitted by him, error vector varies from sample to sample even if the values of the variables specified in the model are fixed. This also amounts to say that y varies from sample to sample for fixed values of x 's. More explicitly, in the subscripted matrices $[]_1, []_2, []_3$ and $[]_4$ detailed out below,

$$\begin{bmatrix} y_{11}, x_{11}, x_{12}, x_{13} \\ y_{12}, x_{21}, x_{22}, x_{23} \\ y_{13}, x_{31}, x_{32}, x_{33} \\ \dots, \dots, \dots, \dots \\ y_{1n}, x_{n1}, x_{n2}, x_{n3} \end{bmatrix}_1 \quad
\begin{bmatrix} y_{21}, x_{11}, x_{12}, x_{13} \\ y_{22}, x_{21}, x_{22}, x_{23} \\ y_{23}, x_{31}, x_{32}, x_{33} \\ \dots, \dots, \dots, \dots \\ y_{2n}, x_{n1}, x_{n2}, x_{n3} \end{bmatrix}_2 \quad
\begin{bmatrix} y_{31}, x_{11}, x_{12}, x_{13} \\ y_{32}, x_{21}, x_{22}, x_{23} \\ y_{33}, x_{31}, x_{32}, x_{33} \\ \dots, \dots, \dots, \dots \\ y_{3n}, x_{n1}, x_{n2}, x_{n3} \end{bmatrix}_3 \quad
\begin{bmatrix} y_{41}, x_{11}, x_{12}, x_{13} \\ y_{42}, x_{21}, x_{22}, x_{23} \\ y_{43}, x_{31}, x_{32}, x_{33} \\ \dots, \dots, \dots, \dots \\ y_{4n}, x_{n1}, x_{n2}, x_{n3} \end{bmatrix}_4 \quad \dots \quad (1.3)$$

x's are fixed and yet the y's are different across the matrices. That is to say that the matrices $[]_1$, $[]_2$, $[]_3$ and $[]_4$ differ from one another in that their y columns are not identical (though their x columns are identical). This is so because for the four samples even though x's are identical, hidden ($\chi_4, \chi_5, \chi_6, \omega_1, \omega_2, \zeta_1, \zeta_2$) are different giving rise to different error vectors (e's). This gives rise to stochasticity in the error, e. The error is a random variable following a particular probability distribution. The scientist may have some speculation, conjecture or hypothesis regarding the probability distribution of the error term, e.

The method of Least Squares: The scientist aims at obtaining the best values of $\mathbf{a} = [a_0, a_1, a_2, a_3]'$. Since no value of \mathbf{a} can satisfy all equations, he has to define the meaning of the qualification '*best*'. It is natural to define '*best*' \mathbf{a} – call it $\hat{\mathbf{a}}$ – such that it yields an error vector, e with the minimum norm. If the scientist's choice is to minimize the Euclidean norm of the error vector, it is tantamount to minimize the squared Euclidean norm of the error vector. Therefore, he chooses to minimize

$$S = \sum_{i=1}^n e_i^2 = \sum_{i=1}^n \{y_i - (a_0 + a_1 x_{i1} + a_2 x_{i2} + a_3 x_{i3})\}^2.$$

To minimize S he differentiates S partially with respect to a_i ($i=0, 1, 2, 3$) and sets those partial derivatives equal to zero. This gives the so-called 'normal equations'.

This system of normal equations in this example will have four unknowns in four equations. Assuming linear independence of these equations, the system may be solved for

a. This method of obtaining 'best' **a** ($= \hat{a}$) is called the *Principle of Least Squares*.

Under the Gauss-Markov assumptions $\hat{a} = (X'X)^{-1} X'y$ is the best linear unbiased estimator of **a** in $y = X\mathbf{a} + e$. These assumptions are: (i) $E(e_i) = 0 \Rightarrow E(y_i) = \sum_{j=0}^m a_j x_{ij}; \forall i, i=1,2,\dots,n$. In general, $E(e) = 0 \Rightarrow E(y_i) = X\mathbf{a}$. Here $E(.)$ means the expectation of the random variable (.). (ii) $E(ee') = \sigma^2 I$ or geometrically, the spherical disturbances. (iii) The matrix of sample values of the explanatory variables, X , is a fixed non-stochastic matrix of full rank.

However, reality often breaks away with the Gauss-Markov assumptions. Very often disturbances are non-spherical. Occasionally, in single equation models, the assumption regarding the non-stochasticity of the sample values of the explanatory variables may not be satisfied. That makes the (ordinary) Least Squares estimator, \hat{a} , an inconsistent estimator. The violation of the first Gauss-Markov assumption introduces bias into the Least Squares estimator. The estimator continues to be biased even if the sample size is increased indefinitely.

The real world data often consist of disturbances that are non-normally distributed, some of which permit the variate to take on non-negative values only. It has been known since Pareto, that income distribution bears testimony to distribution of the error term with infinite variance. Works of **Meyer & Glauber** (1964), **Fama** (1965) and **Mandlebroth** (1967), among others, confirm that economic data series like prices in commodity and financial markets present a class of distribution with infinite variance. An infinite variance

means “thick tail” which implies that large values or ‘*outliers*’ are present. Sporadic errors in X also are frequently present.

We recall that in defining the ‘best’ solution the scientist’s choice was to minimize the Euclidean norm of the error vector, tantamount to minimization of the squared Euclidean norm of the error vector. That choice yielded the Least Squares estimator. However, this method places a relatively heavy weight on outliers and, therefore, in the presence of outliers, the method can lead to extremely sensitive estimates. This implies that in repeated sampling from a distribution where outliers are prevalent, LS estimates will vary considerably and will fail to establish reliability, becoming extremely sample dependent. Moreover, it is also not possible to obtain a meaningful variance estimate in the cases where the variance does not exist and the LS method in such cases will cease to possess its minimum variance properties. Thus, it becomes necessary to look for an alternative, a more ‘robust’ estimator, which gives relatively less weight to outliers and hence is not affected by their presence or the distribution that the error term characterizes.

Therefore, one parts with the convention of minimizing the Euclidean norm of the disturbance vector and, instead, defines the ‘best’ solution, \tilde{a} , of \mathbf{a} in $y = X\mathbf{a} + e$ that

minimizes the *absolute norm* of e (or yields minimal $\sum_{i=1}^n |e_i| = \sum_{i=1}^n \left| y_i - \sum_{j=0}^m \tilde{a}_j x_{ij} \right|$). It may be

noted that the objective of the scientist is to obtain the best solutions – the solution that best represents the factual relationship among the variables - the solution that is least affected by the sample aberrations – and not to stick to any convention or prefer one norm to the others. Evidently, since the ‘least absolute norm’ estimator minimizes the sum of absolute residuals, it is less influenced by outliers and, in the presence of outliers, its sampling variability is less than that of the LS estimator. This estimator is called by

various names such as LAD (Least Absolute Deviation), LAR (Least Absolute Residual) and MAD (minimum Absolute Deviation) estimator.

The idea is not new. Among the multitude of measures of central tendency, the arithmetic mean is obtained by minimizing the Euclidean norm of errors and therefore, it is an estimator of the population mean that minimizes the error variance (or the squared expected Euclidean norm of deviations about the sample mean). However, the median is obtained by minimizing the absolute norm of deviations from the measure of the central tendency. Thus we define arithmetic mean, \hat{a} , and median, \tilde{a} in

$$y_i = x_i \mathbf{a} \quad \dots (1.4)$$

(where x is a unitary vector or $x_i = 1$ for $i = 1, 2, \dots, n$) such that \hat{a} minimizes $S_2 =$

$$\sum_{i=1}^n |y_i - x_i \hat{a}|^2 \equiv \sum_{i=1}^n (y_i - x_i \hat{a})^2 \text{ for the choice of } \hat{a} \text{ as an estimator of } \mathbf{a} \text{ and } \tilde{a} \text{ minimizes } S_1 = \sum_{i=1}^n |y_i - x_i \tilde{a}| \text{ as an estimator of } \mathbf{a} \text{ in (1.4) above. It is well known that } \tilde{a} \text{ (the median of } y) \text{ is unaffected by extremal values of the variate } y \text{ and therefore, the outliers in } y, \text{ while } \hat{a} \text{ (the arithmetic mean of } y) \text{ is oversensitive to the extremal values and outliers.}$$

Considering in a more general framework, \tilde{a} and \hat{a} are obtained by minimization of the Minkowski norm L_p for $p=1$ and $p=2$ respectively.

The recommendations for using LAD estimator may be traced back to **Gauss** and **Laplace** (1818) as mentioned by **Taylor** (1974). **Edgeworth** (1887, 1888, 1923), **Rhodes** (1930) and **Singleton** (1940) investigated into this method of estimation. But at those times, computational difficulties involved with the method went in its disfavour. However, development of linear programming (LP) and fast computing machines intensified the

interest of researchers in estimation of regression models by minimization of L_1 norm. **Charnes, Cooper & Ferguson** (1955) were the first to transform the problem of estimation of parameters of a linear (regression) model into an LP problem. Later, iterative and search methods to obtain the solution were also discovered. **Fisher** (1961), **Ashar & Wallace** (1963), **Meyer & Glauber** (1964), **Rice & White** (1964), **Usoy** (1967), **Oveson** (1968), **Roberts & Ben-Israel** (1969), **Abdelmalek** (1971, 1974), **Blattberg & Sargent** (1971), **Smith & Hall** (1972), **Barrodale & Roberts** (1973), **Schlossmacher** (1973), **Fair** (1974), **Taylor** (1974), **Nyquist & Westlund** (1977), **Bassett & Koenker** (1978), **Powell** (1984), etc. intensively worked on the L_1 estimator of single equation (regression) models. **Powell** (1984) deserves a special mention since he extended the application of LAD estimation to censored regression.

Investigations into the performance of L_1 estimator of single equation models clearly establish its superiority to the conventional L_2 (that is, Least Squares) estimator when errors contain outliers and hence are thick-tailed. In this regard, **Bassett** and **Koenker** (1978) deserve a special mention. They analytically established that L_1 estimator has strictly smaller asymptotic ellipsoids than L_2 estimator for linear models from any F for which the sample median is a more efficient estimator of location than the sample mean. Side by side Monte Carlo experiments also were conducted to compare L_1 and L_2 estimators. These studies also strongly indicated the superiority of L_1 to L_2 estimator for single equation linear models with disturbances infested with outliers. **Pollard** (1991), **Phillips** (1991), **Chen** (1996), **Hitomi & Kagitara** (2001), etc. are some recent works on LAD estimation and its extension.

Multi-equation linear econometric models, described as $YA + XB + E = 0$, were first estimated by minimization of L_2 norm. These estimators are collectively called the k-class estimators (Indirect Least Squares – ILS, Two-Stage Least Squares – 2-SLS and Limited Information Max Likelihood – LIML). The ILS has very limited application due to its applicability in case of exactly identified equations only. In 2-SLS (suitable to estimating exactly as well as over-identified equations), OLS is used to estimate the matrix of Reduced Form Coefficients (P) at the first stage, which also gives estimated Y ($\hat{Y} = XP$). In the second stage, Y is replaced by \hat{Y} if and only if it appears as an explanatory variable in any structural equation. After this replacement, OLS is applied to estimate the structural equations, one at a time. Thus, OLS is applied twice, once at each stage. The 2-SLS is also an Instrumental Variable method of estimation. In spite of OLS – the basic building block of 2-SLS – being an ideal estimator if the required conditions for its application are met, 2-SLS is ordinarily a biased but consistent estimator. It was found that L_2 -based k-class estimator performs extremely poorly when non-normally distributed, hyper-kurtic or outlier-infested errors are met with.

Glahe & Hunt (1970) were the first to apply LAD estimation method to multi-equation linear models. **Amemiya** (1980, 1982) extended LAD (L_1 estimator) to multi-equation models. **Newey** (1985), **Pagan** (1986), **Fair** (1994) and **Kim & Muller** (2000) are some important studies on 2-SLAD estimation.

Amemiya (1982) is certainly a landmark in the history of investigation into the possibility of applying L_1 estimation method to multi-equation linear models. He conjectured that if the disturbances in the structural as well as reduced form equations of a multi-equation model follow a non-normal distribution, it would be better to apply LAD to

the reduced form equation as well as to the structural equation to be estimated. Based on this conjecture he developed D2SLAD (Double Two-Stage Least Absolute) estimator. Thus he generalized LAD to include 2-SLS as its special case. Amemiya's work is theoretical and his conclusions relate to asymptotic properties (consistency) of LAD in estimating the multi-equation model. He suggested estimation by D2SLAD (Double 2-Stage LAD) in case of full non-normal and outlier infested errors. Amemiya suggested Monte Carlo experiments to be carried out in order to study the properties of 2SLAD estimator vis-à-vis those of D2SLAD and 2-SLS.

What has been described above is in essence an investigation into an estimator parallel to the Basman-Theil estimator (2-SLS) of a multi-equation system, where instead of the Euclidean norm the absolute norm is minimized. The idea of estimation in two stages is maintained. However, there is another parallel to 2-SLS suggested by **Khazzoom** (1976) that is more akin to the ILS.

Khazzoom investigated into generalization of ILS for an over-identified equation. He estimated reduced form equations of a multi-equation linear econometric model by OLS but (in the second stage) instead of estimating the (modified) structural equations by OLS (or the Instrumental Variable method) as done by the 2-SLS, he applied generalized inverse of the relevant sub-matrix of reduced form coefficients to obtain the structural coefficients. This may be called the GILS (Generalized Indirect Least Squares) estimator.

It is natural to conjecture that since Khazzoom minimizes the Euclidean norm of errors, his method of estimation may share the shortcomings of ILS and 2-SLS, especially when the error vectors contain outliers. It is plausible that in presence of outliers an

estimator parallel to that of Khazzoom (but minimizes the absolute norm of errors) is more effective.

II. Objectives of the Study: The description in the preceding section suggests investigation in several lines. In the two-stage estimation procedure one may replace OLS by LAD

- (i) at the first stage only (to obtain reduced form coefficients) and apply OLS at the second stage, or
- (ii) at the second stage only (to obtain the structural coefficients while the reduced form coefficients at the first stage are obtained by OLS), or
- (iii) at the first stage to obtain reduced form coefficients and consequently at the second stage also to obtain the structural coefficients. Suggestions given by Amemiya need implementation and empirical/experimental corroboration.

Similarly, in case of the Khazzoom estimator that applies OLS to obtain the reduced form coefficients, one may apply LAD instead for the same purpose. The present study basically aims at investigating into these possibilities. In particular, the generic objectives of the present study are given as hereunder:

- (i) To study the relative performance of the LAD estimator (vis-à-vis OLS) in case of single equation models containing error vectors of different statistical distributions infested with outliers. It is required since single equation estimation is the basic building block in all methods that concern us.

(ii) To study the relative performance of the LAD estimator applied at different stages (vis-à-vis the standard 2-SLS estimator) in estimating the multi-equation models containing error vectors of different statistical distributions infested with outliers.

(iii) To study the relative performance of the LAD-based Khazzoom type estimator vis-à-vis the original Khazzoom estimator as well as various types of two-stage estimators applying OLS or LAD estimator in estimating the multi-equation models containing error vectors of different statistical distributions infested with outliers.

III. The Methodology: Owing to very poor mathematical tractability of LAD estimator we have relied on the *Monte Carlo method* to study the properties of various methods of estimation that we purport to investigate into. The Monte Carlo method of investigating into the properties of a mathematical system is basically experimental and numerical in nature. This method may be loosely described as a statistical simulation method, where statistical simulation is defined in quite general terms to be any method that utilizes sequences of random numbers to perform the simulation. The main idea behind the method is either to construct a stochastic model that is in agreement with the whole problem analytically or to simulate the whole problem directly. In both cases, an element of randomness has to be introduced according to some well-defined rules. Then a large number of trials are performed, the results are observed, and finally a statistical analysis is undertaken in the usual way. The advantages of the method are that even very difficult or analytically intractable problems can often be treated very easily, and desired modifications can be applied without much trouble. Nevertheless, the conclusions emanating from Monte Carlo studies are relatively less (vis-à-vis analytical methods)

accurate and the large number of trials that are necessary. Now, when very fast computers are readily available, the number of calculation in large number of trials is no longer a serious matter. This facility may be utilized to increase the number of trials sufficiently large so as to attain an acceptable degree of precision. Thus the trade-off between number and precision goes in favour of the Monte Carlo method.

In this investigation we have limited ourselves to deal with linear econometric models. Our single equation models are described as $y = X\mathbf{a} + e$. The multi-equation models are described as $YA + XB + E = 0$. The disturbance term (e or E as the case may be) is stochastic and follows either of the five distributions : (i) Normal, (ii) Cauchy, (iii) Beta₁, (iv) Beta₂ and (v) Gamma. We have experimented with the cases when the disturbance term is infested with outliers, different in numbers and sizes. In the simulation we have experimented with the models of various sizes (different number of variables or equations) as well as with various sample sizes implying Y and X of varying dimensions.

In case of single equation models described as $y = X\mathbf{a} + e$, (i) first X of the desired dimension is generated. The coefficients, \mathbf{a} , are assumed. These X and \mathbf{a} remain fixed in a single bout of experiment. (ii) Then e of desired distribution is generated and $y = X\mathbf{a} + e$ is obtained. The desired number of outliers (of sizes within some desired range) are generated and added to y . Next, the desired method(s) of estimation is (are) applied to obtain \hat{a} which is the estimator of \mathbf{a} . The step # ii is repeated a large, say m , number of times. This gives m number of \hat{a} 's which are subjected to statistical analysis to assess the properties of the method(s) of estimation.

In case of multi-equation models described as $YA + XB + E = 0$: (i) first X of the desired dimension is generated and the coefficients A and B are assumed. From A and B

we obtain $\Pi = -BA^{-1}$. (ii) Then E of desired distribution is generated and $Y = X\Pi - EA^{-1}$ is obtained. The desired number of outliers (of sizes within some desired range) are generated and added to Y . Next, the desired method(s) of estimation is (are) applied to obtain \hat{A} and \hat{B} which are the estimators of A and B . The step # ii is repeated a large, say m , number of times. This gives m number of \hat{A} 's and \hat{B} 's which are subjected to statistical analysis to assess the properties of the method(s) of estimation.

IV. Organization of the Work: The subsequent part of this work is organized as follows: First, in chapter-2, the framework of linear econometric models is presented. Presently, estimation of econometric models by minimization of the Euclidean norm of errors is the most popular practice and hence, most of the estimation techniques are based on the Least Squares principle directly or indirectly. Details on these techniques are given therein.

In chapter-3 we present a literature survey on LAD estimation and the relevant numerical algorithms for that purpose. Estimation by minimization of absolute errors is mostly intractable by analytical methods and its performance (vis-à-vis LS-based methods) can in general be assessed only by simulation-based empirical methods such as the Monte Carlo method. In chapter-4, we present the salient features of the Monte Carlo method and give an outline that paves a way to using this simulation method to assess the relative performance of LAD-based estimators in single as well as multi equation econometric models. Since the Monte Carlo method is based on generating random numbers, we also present a description of some relevant statistical distributions that random numbers can follow and the computer-based techniques that may be used to generate random variates following such statistical distributions.

Next, in chapter-5, we conduct Monte Carlo experiments to compare the performance of LAD-based and LS-based methods of estimation of single equation econometric models. The residual terms in these models follow five different statistical distributions viz. normal, Cauchy, Gamma, Beta_1 and Beta_2 . The residual terms may contain outliers different in number and magnitude.

Chapter-6 is an extension of the earlier chapter to the case of multi-equation econometric models. However, unlike in case of single-equation econometric models, we have several LS-based methods of estimation in case of the multi-equation econometric models. That is so because these models are estimated in two stages. Consequently, it is possible to replace LS by LAD at any one or both of these stages, giving rise to many possible combinations and possibilities of a comparative study in their performance. This has been done in Chapter-6.

We present a summary of our results in chapter-7. We have also visualized the lines of future research in application of LAD estimators of econometric models.

DasGupta, Madhuchhanda (2004) LEAST ABSOLUTE DEVIATION ESTIMATION OF MULT-EQUATION LINEAR ECONOMETRIC MODELS : A STUDY BASED ON MONTE CARLO EXPERIMENTS, Doctoral Dissertation, Dept. of Economics, North Eastern Hill University, Shillong (India); Supervisor – Prof. SK Mishra

CHAPTER 2

THE FRAMEWORK OF LINEAR ECONOMETRIC MODELS AND THEIR ESTIMATION

I. Introduction: Construction of an econometric model is a pre-requisite for any econometric research. A model abstractly represents reality by bringing out what is relevant to a particular question neglecting all other unnecessary details. These models may be of different sizes and complexities. The most simple of them may involve single relationship among economic variables and, therefore, can be summed up in a single equation such as:

$$y = a_0x_0 + a_1x_1 + \dots + a_kx_k + e$$

where x_0 is a unit (column) vector. In the matrix form this model is simply expressed as $y = Xa + e$. Such a model is popularly known as the multi-variate regression model. However, this simple relationship is the most fundamental building block of more complex models known as the multi-equation models that may consist of an impressive array of relations.

Multi-equation models almost invariably exhibit simultaneity relations in different variables of the model. Simultaneity occurs in an econometric model within which more than one causal relationship among the variables is specified. When interrelations exist among the variables, the single equation specification (as illustrated above) with its one way implied causality from predetermined to one endogenous variable is neither an accurate nor sufficient representation. In such cases, the economic variables are determined by a complete system of equations. A complete system is the representation

(model) in which there are as many equations as endogenous variables (whose formation is to be 'explained' by the equations). The equations are usually of four types: equations of economic behaviour, institutional rules, technological laws of transformation and identities. The term 'structural equation' is used to comprise all four types of equations.

Systems of structural equation may be composed entirely on the basis of economic theory or on the dual basis of economic theory combined with systematically collected statistical data for the relevant variable for a given period. These models serve as an indispensable aid to forecasting and an invaluable guide to policy-making whether for a firm or a governmental agency.

Since a characteristic feature of most economic models is interdependence and joint determination of several variables, the use of simultaneous equation model is more meaningful and logical. The seminal papers from which the simultaneous equation model developed establish the importance of joint endogeneity for statistical analysis of economic relationships. **Haavelmo** (1943, 1944) realised that in the presence of jointly endogenous variables, a joint probability distribution is necessary to analyse the data.

II. The General Simultaneous Equation Model: Before proceeding further let us develop the general simultaneous equation model. For this let us take a linear simultaneous equation model containing m structural relations. The model may be best described in a matrix form as $YA + XB + E = 0$, where, Y represents m number of endogenous variables, the X represents k number of pre-determined variables, E symbolizes the stochastic disturbance terms m in number and A and B are the matrices of structural coefficients of compatible order. For the purpose of estimation, n is the sample size of data on Y and X . The underlying theory in general will satisfy that only a small

number of all the variables of the model will occur in any one equation, and the matrices A and B will have a considerable number of zeros. Further in each structural equation, one endogenous variable can by convention be regarded as the dependent variable, the coefficient of which is equal to unity. This is known as the normalization rule. By a suitable arrangement of the equations and the endogenous variables in the model, this rule helps to make the main diagonal elements such that $a_{ii} = -1$ for all i in the matrix A. To take care of the constant terms in the equations, the first column of X is a unit vector such that $x_{i1} = 1$ for all $i = 1, 2, \dots, n$.

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mm} \end{bmatrix}; B = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1m} \\ b_{21} & b_{22} & \cdots & b_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ b_{k1} & b_{k2} & \cdots & b_{km} \end{bmatrix}; 0 = \begin{bmatrix} 0_{11} & 0_{12} & \cdots & 0_{1m} \\ 0_{21} & 0_{22} & \cdots & 0_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ 0_{n1} & 0_{n2} & \cdots & 0_{nm} \end{bmatrix}$$

$$Y = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1m} \\ y_{21} & y_{22} & \cdots & y_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ y_{n1} & y_{n2} & \cdots & y_{nm} \end{bmatrix}; X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1k} \\ x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nk} \end{bmatrix}; E = \begin{bmatrix} e_{11} & e_{12} & \cdots & e_{1m} \\ e_{21} & e_{22} & \cdots & e_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ e_{n1} & e_{n2} & \cdots & e_{nm} \end{bmatrix}$$

In the model described above, A is a square matrix and is assumed to be non-singular, since if it were not, one or more of the structural relations would merely be a linear combination of other structural relations, thus being redundant, or if the rows of B did not obey the same linear restrictions as the rows of A, the m structural equations would be inconsistent. The B matrix is generally not a square matrix. The m equations in the structural form jointly determine for each observation, the value of m endogenous variables, given k pre-determined variables, m stochastic disturbance term and m(m+k-1) coefficients of the system. The stochastic disturbance terms are assumed to be identically

and independently distributed over the samples with zero mean and constant covariance matrix.

III. Estimation of Single Equation Econometric Models: Single equation econometric models are often estimated by the method of Least Squares that minimizes the squared Euclidean norm of the error vector, S . It is tantamount to minimization of the Euclidean norm of the error vector.

$$S = \sum_{i=1}^n e_i^2 = \sum_{i=1}^n \{y_i - (a_0 + a_1 x_{i1} + a_2 x_{i2} + \dots + a_k x_{ik})\}^2.$$

To minimize S we differentiate S partially with respect to a_i ($i=0, 1, 2, \dots, k$) and set those partial derivatives equal to zero.

$$\begin{aligned} \frac{\partial S}{\partial a_0} &= -2 \sum_{i=1}^n \{y_i - (a_0 + a_1 x_{i1} + a_2 x_{i2} + a_3 x_{i3})\} = 0 \\ \frac{\partial S}{\partial a_1} &= -2 \sum_{i=1}^n \{y_i - (a_0 + a_1 x_{i1} + a_2 x_{i2} + a_3 x_{i3})\} x_{i1} = 0 \\ &\dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \\ \frac{\partial S}{\partial a_k} &= -2 \sum_{i=1}^n \{y_i - (a_0 + a_1 x_{i1} + a_2 x_{i2} + \dots + a_k x_{ik})\} x_{ik} = 0 \end{aligned} \quad \dots (2.1)$$

Simplifying the above equations we obtain the so-called normal equations as follows:

$$\begin{aligned} \sum_{i=1}^n y_i &= n a_0 + a_1 \sum_{i=1}^n x_{i1} + a_2 \sum_{i=1}^n x_{i2} + \dots + a_k \sum_{i=1}^n x_{ik} \\ \sum_{i=1}^n y_i x_{i1} &= a_0 \sum_{i=1}^n x_{i1} + a_1 \sum_{i=1}^n x_{i1}^2 + a_2 \sum_{i=1}^n x_{i1} x_{i2} + \dots + a_k \sum_{i=1}^n x_{i1} x_{ik} \\ &\dots \quad \dots \quad \dots \quad \dots \quad \dots \\ \sum_{i=1}^n y_i x_{ik} &= a_0 \sum_{i=1}^n x_{ik} + a_1 \sum_{i=1}^n x_{i1} x_{ik} + a_2 \sum_{i=1}^n x_{i2} x_{ik} + \dots + a_k \sum_{i=1}^n x_{ik}^2 \end{aligned} \quad \dots (2.2)$$

$= (Y - X\hat{a})' (Y - X\hat{a})$ of the disturbance vector e . It also implies that the Euclidean norm of e is minimized by \hat{a} . If we considered $y^* = (y - e)$ instead of y , the system $y^* = X\mathbf{a}$ is over-determined, though consistent. The over-determined system of equations will admit generalized solution

$$\hat{a}^* = (X'X)^{-1} X'y^* = (X'X)^{-1} X'y - (X'X)^{-1} X'e.$$

Now if $(X'X)^{-1} X'e = 0$, we have $\hat{a} = \hat{a}^*$.

Strictly speaking, the method of Least Squares to obtain \hat{a} is a mathematical rather than statistical method because it does not incorporate randomness or probability considerations in obtaining \hat{a} nor does it give us any hint on the statistical properties of \hat{a} , although \hat{a} , the best estimator of \mathbf{a} , is obtained by minimizing the sample $S = e'e$ which is stochastic in nature. Since $\hat{a} = (X'X)^{-1} X'y$ where y is stochastic, \hat{a} assumes stochasticity because of the simple fact that the function of a random variable is in turn a random variable.

It may be shown (**Johnston**, pp. 171-174) that under certain assumptions $\hat{a} = (X'X)^{-1} X'y$ is the best linear unbiased estimator (**blue**) of \mathbf{a} in $y = X\mathbf{a} + e$. These assumptions are known as the Gauss-Markov assumptions of (linear) regression model $y = Xa + e$ (**Intriligator**, pp. 106-109). These assumptions are: (i) $E(e_i) = 0 \Rightarrow E(y_i) = \sum_{j=0}^m a_j x_{ij}; \forall i, i=1,2,\dots,n$. In general, $E(e) = 0 \Rightarrow E(y_i) = X\mathbf{a}$. Here $E(.)$ means the expectation of the random variable $(.)$. (ii) $E(ee') = \sigma^2 I$. This assumption asserts that each disturbance e_i is distributed with the same positive and finite variance σ^2 and all

disturbances are pair-wise uncorrelated. This is also called the assumption of homoskedasticity and non-autocorrelation (or geometrically, the spherical disturbances).

(iii) The matrix of sample values of the explanatory variables, X , is a fixed non-stochastic matrix of full rank. Although the assumption of normality of e is not required to prove **blue** properties of \hat{a} , such an assumption makes \hat{a} the most efficient estimator of \mathbf{a} in $y = X\mathbf{a} + e$ since in that case \hat{a} is also a maximum likelihood estimator of \mathbf{a} , attaining the Cramer-Rao bound (**Intriligator**, p. 108).

However, reality often breaks away with the Gauss-Markov assumptions that generously bestow upon the Least Squares estimator, \hat{a} , the properties of an ideal estimator. Very often disturbances are non-spherical. It has been shown that if e has a non-spherical distribution, the Aitken estimator (or Generalized Least Squares - **GLS** - estimator) \hat{a} of \mathbf{a} is **blue**. The GLS estimator \hat{a} is given by

$$\hat{a} = (X' \Omega^{-1} X)^{-1} X' \Omega^{-1} y \quad \dots (2.4)$$

The estimator is 'general' in the sense that for spherical disturbances $\Omega = I$ \hat{a} reduces to the (ordinary) Least Squares estimator \hat{a} , that is, $\hat{a} = \hat{a}$. In practice, however, Ω is not known and therefore it has to be estimated from the sample observations (**Intriligator**, pp. 168-173) and consequently the GLS estimator is only consistent (**Theil**, pp. 398-400).

Occasionally, in single equation models, the assumption regarding the non-stochasticity of the sample values of the explanatory variables may not be satisfied. That makes the (ordinary) Least Squares estimator, \hat{a} , an inconsistent estimator (**Theil**, p. 608). There are two suitable estimators that deserve a mention in this regard. First, the

Wald-Hooper-Theil estimator (**Hooper, JW & H Theil** (1958)) and the second, the Instrumental Variable method (**Reiersol, O.** (1950)). In multi-equation (or simultaneous equation) models stochasticity of explanatory variables is invariably met with. Therefore, (ordinary) Least Squares estimator is an inconsistent estimator of the structural parameters of multi-equation models. Other methods such as Two-Stage Least Squares, Limited Information Maximum Likelihood method, etc. are applied to estimation of parameters of these models.

The violation of the first Gauss-Markov assumption is no less serious. It introduces bias into the Least Squares estimator. If the estimator continues to be biased even if the sample size is increased indefinitely (although in practice it is often impossible to increase the sample size indefinitely), it is futile to investigate into other desirable properties of the estimator. The presence of outliers in the errors often leads to the violation of the first Gauss-Markov assumption and the method of Least Squares yields unreliable estimates of the parameters. This fact prompts us to investigate into the alternative methods of estimation such as the Least Absolute Deviation Estimation.

IV. Estimation of Multi-equation or Simultaneous Equation Econometric Models: It is well-known now that in a system of simultaneous equations the method of (ordinary) least squares yields biased and inconsistent estimators (Intriligator, p. 376) of the structural coefficients. This was first established by Haavelmo (1943, 1947). He showed that simultaneity induces correlation between the regressors and the residuals giving inconsistent and biased estimates. In his path-breaking monograph, “*The probability Approach in Econometrics*” (1944) he developed a method for solving the problem of single equation bias.

Marschak (1953) has explained why we must try to establish the structural coefficients. The basic argument is that by its very definition each structural equation describes a specific, distinct and thereby autonomous link in the economic process. This autonomous character of structural relations also means that the structural coefficients are more stable, more like physical constants, than are the reduced form composites. Structural coefficients unlike the reduced form coefficients are more easily judged by intuition and their changes are better capable of reasonable discussion and interpretation. Structural coefficients are more easily comparable with the accumulated empirical evidence, which refers to the economic structure and not to the reduced form.

In the way to establishing the structural coefficients of a simultaneous equation model one has to estimate the reduced form equations. Haavelmo proposed to first estimate the reduced form of a system of simultaneous equations by ordinary least squares and then to derive estimates for the parameters of the structural form by *indirect least squares*. In the case of exactly identified systems, **Haavelmo** (1943, 1944, 1947) showed that this method is equivalent to the method of maximum likelihood. Considerable theoretical investigations were also carried out by the researchers in **Cowles Foundation**, e.g., **Koopmans** (1950) and **Hood & Koopmans** (1953), etc. on methods of estimating structural parameters of a system of simultaneous linear stochastic equations. The estimates have been derived from maximum likelihood considerations and proved to be asymptotically unbiased and efficient.

Haavelmo's investigations made the 'identification problem' central to the estimation of simultaneous equation models. The identification problem may be cogently described as follows.

From the structural equations $YA + XB + E = 0$, we may obtain the reduced form equations through pre-multiplying the system of structural equations by A^{-1} . That is explicitly, $YAA^{-1} + XBA^{-1} + EA^{-1} = 0$ or $Y = X\Pi + U$, where $\Pi = -BA^{-1}$ and $U = -EA^{-1}$. Now since the reduced form equations are amenable to estimation by Ordinary Least Squares, we obtain the least squares estimates of Π as $P = (X'X)^{-1}X'Y$. But $\Pi A = -B$ and hence $P\hat{A} = -\hat{B}$. This gives us a system of k equations in $m(m+k-1)$ unknowns even after applying the normalization rule on A such that $a_{ii} = -1$ for all $i = 1, 2, \dots, m$. Obviously, such a system of equations admits no determinate solution. For any particular structural equation, say j^{th} equation, we have $P\hat{a}_j = -\hat{b}_j$ which is a system of k equations in $m+k-1$ unknowns. Unless we assume some (at least as many as $m-1$) elements of \hat{a}_j or \hat{b}_j equal to zero, we cannot estimate other remaining elements of \hat{a}_j and \hat{b}_j . Thus, the identifiability or the zero restriction on at least $m-1$ elements of \hat{a}_j and \hat{b}_j is the basic requirement for obtaining the estimates of the coefficients of the j^{th} structural equation. Moreover, the system of equations $P\hat{a}_j = -\hat{b}_j$ must satisfy the rank condition which is the basic algebraic requirement for solving any system of linear equations.

In what follows we describe various methods to estimate the structural equations of a multi-equation (simultaneous equation) model. These methods minimize the Euclidean norm of errors in the dependent variable directly or indirectly. Identifiability of each equation in the model is presumed.

Indirect Least Squares and Generalized Indirect Least Squares: The Indirect Least Squares (ILS) method was first proposed by **Girshick** (ref. by Hood & Koopmans, 1953, p. 140). It was generalized by **Khazzoom** (1976), though he did not name it the

Generalized Indirect Least Squares (GILS). For a comprehensive treatment we will describe here the method of GILS only.

From the structural equations $YA + XB + E = 0$ we derive the reduced form equations $Y = X\Pi + U$ and estimate its coefficients by $P = (X'X)^{-1}X'Y$ by OLS. Since $(X'X)^{-1}X' = X^{-g}$ (the generalized inverse of X), we may also write $P = X^{-g}Y$.

The question that arises now is whether we can obtain \hat{A} and \hat{B} from the knowledge of P . This is the problem of identification. If all the columns of A and B can be obtained then A and B can also be completely obtained. If a particular column of A and B (say, a_j and b_j) can be obtained from the relation

$$Pa_j = -b_j \quad \dots (2.5)$$

then the j^{th} equation is identifiable. Since this is true for any equation and (2.5) holds for any particular column of A and B , dropping the subscript j the equation (2.5) can be rewritten as

$$\underset{k \times m}{P} \underset{m \times 1}{a} = \underset{k \times 1}{-b} \quad \dots (2.6)$$

The j^{th} element of a is -1 if (2.6) relates to the equation j . The system (2.6) is in k equations and $k+m-1$ unknowns. So equation (2.6) cannot be solved for a and b .

Now, suppose, from a-priori information the values of some (say k_2) elements of b and some other elements of a can be obtained such that k_2 elements of b and m_2 elements of a are known. In that case, out of k elements of b only $k-k_2=k_1$ elements are unknown. Similarly out of m elements of a , $m-m_2=m_1$ elements are unknown. Thus, we proceed to identification by restriction on the structural coefficients matrix a and b .

Pre-multiplying (2.6) by a suitable permutation matrix R we may obtain

$$RP a = - R b \quad \dots (2.7)$$

such that b can be partitioned into two sub matrices, first of which, b_1 has k_1 unknown elements and the second, b_2 has k_2 known elements.

Correspondingly, RP can also be partitioned into P_1 and P_2 . It may noted that such a permutation effects only a re-shuffling of equations in (2.6) and does not have any bearing on its solution. Consequently, equation (2.7) can thus be re-written in a partitioned form as

$$\begin{bmatrix} P_1 \\ P_2 \end{bmatrix} a = - \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}; \quad R^{-1} \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = b \quad \dots (2.7a)$$

Here, P_1 , P_2 , b_1 and b_2 are $k_1 \times m$, $k_2 \times m$, $k_1 \times 1$ and $k_2 \times 1$ matrices respectively.

Again, pre-multiplying a by a suitable permutation matrix, S , and post-multiplying RP by $S^{-1} = S'$ in equation (2.7a) we obtain

$$\begin{bmatrix} P_1 \\ P_2 \end{bmatrix} S^{-1} S a = \begin{bmatrix} P_1 \\ P_2 \end{bmatrix} S^{-1} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}; \quad S^{-1} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = a; \quad S^{-1} S = I$$

where a_1 contains m_1 unknown elements and a_2 contains m_2 known elements. Equation (2.7a) may now be rewritten as

$$RPS^{-1}Sa = -Rb \quad \dots (2.7b)$$

This operation helps to shuffle and partition RP into two column submatrices, without any effect to the solution of the system of equations in (2.6). In the partitioned form (2.7b) can be written as

$$\begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = - \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \quad \dots (2.8)$$

From (2.8) we obtain two equations,

$$P_{11}a_1 + P_{12}a_2 = -b_1 \quad \dots (2.9)$$

$$P_{21}a_1 + P_{22}a_2 = -b_2 \quad \dots (2.10)$$

Now, since a_2 and b_2 are known, we get

$$P_{21}a_1 = -[b_2 + P_{22}a_2] \quad \dots (2.11)$$

Here P_{21} is a $k_2 \times m_1$ matrix giving rise to k_2 equations in m_1 unknowns as in (2.10).

If P_{21} is a square (rectangular) matrix in k_2 rows and m_1 column and $\text{rank}(P_{21}) = r \leq m_1$, then the Moore-Penrose inverse of P_{21} that is P_{21}^+ exists and is unique. Using this generalized inverse of P_{21} we obtain from (2.9) and (2.11)

$$a_1 = -P_{21}^+[b_2 + P_{22}a_2] \quad \dots (2.12)$$

$$b_1 = -[P_{11}a_1 + P_{12}a_2] \quad \dots (2.13)$$

This formulation and consequent technique in (2.12) and (2.13) is GILS. **The ILS is only a special case of GILS when $k_2 = m_1 = r$ and therefore, $P_{21}^+ = P_{21}^{-1}$.** However, in case $k_2 \geq m_1$ and $r = m_1$, then P_{21}^+ yields least squares g-inverse, P_{21}^{-1} or P_{21}^{-g} .

GILS in Another Form: Rewriting equation (2.8) as

$$\begin{bmatrix} P_{11} & I \\ P_{21} & 0 \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = - \begin{bmatrix} P_{12}a_2 \\ b_2 + P_{22}a_2 \end{bmatrix} \quad \dots (2.14)$$

which may be again rewritten as

$$\begin{bmatrix} I & P_{11} \\ 0 & P_{21} \end{bmatrix} \begin{bmatrix} \frac{b_1}{a_1} \\ a_1 \end{bmatrix} = - \begin{bmatrix} \frac{P_{12}a_2}{b_2 + P_{22}a_2} \end{bmatrix} \quad \dots (2.15)$$

From (2.15) we obtain

$$\begin{bmatrix} I & P_{11} \\ 0 & P_{21} \end{bmatrix}^+ \begin{bmatrix} \frac{P_{12}a_2}{b_2 + P_{22}a_2} \\ a_1 \end{bmatrix} = - \begin{bmatrix} \frac{b_1}{a_1} \end{bmatrix} \quad \dots (2.16)$$

Now,

$$\begin{bmatrix} I & P_{11} \\ 0 & P_{21} \end{bmatrix} \begin{bmatrix} I & -P_{11}P_{21}^+ \\ 0 & P_{21}^+ \end{bmatrix} \begin{bmatrix} I & P_{11} \\ 0 & P_{21} \end{bmatrix} = \begin{bmatrix} I & P_{11} \\ 0 & P_{21} \end{bmatrix} \quad \dots (2.17)$$

or,

$$\begin{bmatrix} I & 0 \\ 0 & P_{21}P_{21}^+ \end{bmatrix} \begin{bmatrix} I & P_{11} \\ 0 & P_{21} \end{bmatrix} = \begin{bmatrix} I & P_{11} \\ 0 & P_{21} \end{bmatrix} \quad \dots (2.18)$$

or,

$$\begin{bmatrix} I & P_{11} \\ 0 & P_{21}P_{21}^+P_{21} \end{bmatrix} = \begin{bmatrix} I & P_{11} \\ 0 & P_{21} \end{bmatrix} \quad \dots (2.19)$$

Since $P_{21} P_{21}^+ P_{21} = P_{21}$ (Theil, pp. 269-70), (2.19) is equal to (2.17).

Thus,

$$\begin{bmatrix} I & P_{11} \\ 0 & P_{21} \end{bmatrix}^+ = \begin{bmatrix} I & -P_{11}P_{21}^+ \\ 0 & P_{21}^+ \end{bmatrix} \quad \dots (2.20)$$

It goes without saying that since the least squares g-inverse is only a special case of

Moore-Penrose inverse, the above proof is true for g-inverse $[\bullet]^{-g}$ as well.

Equation (2.16) may now be written as

$$\begin{bmatrix} I & -P_{11}P_{21}^+ \\ 0 & P_{21}^+ \end{bmatrix} \begin{bmatrix} \frac{P_{12}a_2}{b_2 + P_{22}a_2} \\ b_2 + P_{22}a_2 \end{bmatrix} = - \begin{bmatrix} \frac{b_1}{a_1} \end{bmatrix} \quad \dots (2.21)$$

Hence, $a_1 = -P_{21}^+[b_2 + P_{22}a_2]$ and $b_1 = -[P_{11}a_1 + P_{12}a_2]$ which is the same as in (2.12) and (2.13) before.

Two Stage Least Squares: The most widely used method for estimating the parameters of an equation of a simultaneous equation model is the 2-SLS developed by H. Theil and independently by Basman who called it Generalized Classical Linear Estimation method. It is applicable to over-identified/exactly identified equations. In the case of exactly identified equation, the 2-SLS estimates are identical with the ILS estimates given by equations (2.12) and (2.13). The basic idea behind the 2-SLS is to substitute each endogenous explanatory variable (that is correlated with the residuals) by a corresponding composite variable (uncorrelated with the residuals) which is a linear function of all exogenous variables in the model. Since the surrogate (instrumental) variables are uncorrelated with the residuals, the estimates of the parameters will be consistent.

Let us suppose that we are interested in estimating the first equation of the general interdependent system of equations.

$$y_1 = Y_1a_1 + X_1b_1 + e_1 \quad \dots (2.22)$$

In the notation above, y_1 is the dependent endogenous variable in the first structural equation and on account of normalization rule the coefficient associated with it is unity. Y_1 and X_1 are explanatory endogenous and exogenous variable (respectively) that appear in the first structural equation while a_1 and b_1 are the coefficients associated with them. In the first stage, reduced form regression of Y_1 on all the pre-determined variables in the

system (X) is performed. The estimates \hat{Y}_1 thus obtained are then used to replace the actual observations on the Y_1 variables. Hence,

$$\hat{Y}_1 = X (X'X)^{-1} X'Y_1 \quad \dots (2.23)$$

Application of OLS in the second stage to the equation thus transformed yields the estimating equations.

$$\begin{pmatrix} \hat{Y}_1' \hat{Y}_1 & \hat{Y}_1' X_1 \\ X_1' \hat{Y}_1 & X_1' X_1 \end{pmatrix} \begin{pmatrix} c \\ d \end{pmatrix} = \begin{pmatrix} \hat{Y}_1' y_1 \\ X_1' y_1 \end{pmatrix} \quad \dots (2.24)$$

The vector $\begin{bmatrix} c \\ d \end{bmatrix}$ now denotes the 2SLS estimator of $\begin{bmatrix} a_1 \\ b_1 \end{bmatrix}$. Since the above system contains the same number of equations and unknowns, i.e. $m+k-1$ equations in $m+k-1$ unknowns, the system in general will have a unique solution.

The form in which the 2SLS equations are usually presented can be derived in the following way:

The matrix Y_1 can be written as

$$Y_1 = \hat{Y}_1 + E_1 \quad \dots (2.25)$$

Now, since the OLS residuals are orthogonal to the estimated value of the dependent variable and to each of the explanatory variables, therefore,

$$\hat{Y}_1' E_1 = 0 \quad \text{and} \quad X_1' E_1 = 0$$

Using equation (2.25) we get,

$$\begin{aligned}
\hat{Y}_1' \hat{Y}_1 &= (Y_1 - E_1)' (Y_1 - E_1) \\
&= Y_1' Y_1 - Y_1' E_1 - E_1' Y_1 + E_1' E_1 \\
&= Y_1' Y_1 - (\hat{Y}_1 + E_1)' E_1 - E_1' (\hat{Y}_1 + E_1) + E_1' E_1 \\
&= Y_1' Y_1 - \hat{Y}_1' E_1 - E_1' E_1 - E_1' \hat{Y}_1 - E_1' E_1 + E_1' E_1 \\
&= Y_1' Y_1 - E_1' E_1
\end{aligned}$$

$$\begin{aligned}
\hat{Y}_1' X_1 &= (Y_1 - E_1)' X_1 \\
&= Y_1' X_1 \\
X_1' \hat{Y}_1 &= X_1' (Y_1 - E_1) \\
&= X_1' Y_1 \\
\text{and } \hat{Y}_1' y_1 &= (Y_1 - E_1)' y_1
\end{aligned}$$

Thus for $\kappa = 1$ the 2SLS estimator can now be written as

$$\begin{pmatrix} Y_1' Y_1 - \kappa(E_1' E_1) & Y_1' X_1 \\ X_1' Y_1 & X_1' X_1 \end{pmatrix} \begin{pmatrix} c \\ d \end{pmatrix} = \begin{pmatrix} (Y_1 - \kappa E_1)' y_1 \\ X_1' y_1 \end{pmatrix} \quad \dots (2.26)$$

The form shows clearly how the 2SLS estimator differs from the inconsistent OLS estimator, which is given by

$$\begin{pmatrix} Y_1' Y_1 & Y_1' X_1 \\ X_1' Y_1 & X_1' X_1 \end{pmatrix} \begin{pmatrix} c \\ d \end{pmatrix} = \begin{pmatrix} Y_1' y_1 \\ X_1' y_1 \end{pmatrix} \quad \dots (2.27)$$

or

$$\begin{pmatrix} Y_1' Y_1 - \kappa(E_1' E_1) & Y_1' X_1 \\ X_1' Y_1 & X_1' X_1 \end{pmatrix} \begin{pmatrix} c \\ d \end{pmatrix} = \begin{pmatrix} (Y_1 - \kappa E_1)' y_1 \\ X_1' y_1 \end{pmatrix} \text{ for } \kappa = 0 \quad \dots (2.28)$$

Limited-Information Maximum Likelihood Estimator: The limited information maximum likelihood (LIML) method is another alternative approach that gives consistent

estimates and was developed by **Anderson & Rubin**. They estimated the parameters of an equation by maximizing the likelihood function subject to appropriate constraints using Lagrange multipliers. Hood & Koopmans have used a different method to develop the LIML estimator. They arrived at the estimates of the parameters of an equation by maximizing the likelihood function for the observations on the endogenous variables included in that equation disregarding the over identifying restrictions on other structural equations. This method assumes that the structural disturbances are normally distributed.

The structural equation to be estimated is again

$$y_1 = Y_1 a_1 + X_1 b_1 + e_1 \quad \dots (2.29)$$

Equation (2.29) can be written as

$$y_1^* = X_1 b_1 + e_1 \quad \dots (2.30)$$

where, $y_1^* = y_1 - Y_1 a_1$. Now y_1^* is a linear combination of the endogenous variables included in the first structural equation. The coefficient b_1 in equation (2.30) can be estimated by ordinary least squares method. Thus, the estimator is

$$\hat{b}_1 = (X_1' X_1)^{-1} X_1' y_1^* \quad \dots (2.31)$$

and the sum of squared errors is

$$\begin{aligned} S_1 &= (y_1^* - X_1 b_1)' (y_1^* - X_1 b_1) \\ &= y_1^{*'} y_1^* - y_1^{*'} X_1 (X_1' X_1)^{-1} X_1' y_1^* \quad \dots (2.32) \end{aligned}$$

Now, let the relation which includes all the predetermined variables be given by

$$y_1^* = X b + \varepsilon_1 \quad \dots (2.33)$$

where, X is an $n \times k$ matrix of all pre-determined variables and

$$b = \begin{bmatrix} b_1 \\ O^{**} \end{bmatrix}$$

Application of OLS to equation (2.33) gives

$$\hat{b} = (X'X)^{-1} X' y_1^* \quad \dots (2.34)$$

and the sum of squared errors is

$$\begin{aligned} S_2 &= (y_1^* - Xb)'(y_1^* - Xb) \\ &= y_1^{*'} y_1^* - y_1^{*'} X (X'X)^{-1} X' y_1^* \end{aligned} \quad \dots (2.35)$$

Using equations (2.32) and (2.35) we can obtain the ratio between the two sums as

$$l = \frac{S_1}{S_2} = \frac{y_1^{*'} y_1^* - y_1^{*'} X_1 (X_1' X_1)^{-1} X_1' y_1^*}{y_1^{*'} y_1^* - y_1^{*'} X (X'X)^{-1} X' y_1^*} \quad \dots (2.36)$$

The ratio l cannot be less than one simply because that the explanatory power of X cannot be less than X_1 and therefore, the variance S_1 cannot be less than the variance S_2 .

To estimate the elements of a_1 , let us suppose that

$$y_1^* = Y_{1\Delta} a_{1\Delta} \quad \dots (2.37)$$

$$\text{where } Y_{1\Delta} = \begin{bmatrix} Y_1 & Y_2 & \dots & Y_m \end{bmatrix} \quad \text{and} \quad a_{1\Delta} = \begin{bmatrix} 1 \\ a_{12} \\ \vdots \\ a_{1m} \end{bmatrix}$$

The ratio l can be written as

$$l = \frac{a'_{1\Delta} W_1^* a_{1\Delta}}{a'_{1\Delta} W_1 a_{1\Delta}} \quad \dots (2.38)$$

where,

$$W_1^* = Y'_{1\Delta} Y_{1\Delta} - (Y'_{1\Delta} X_1)(X'_1 X_1)^{-1} X'_1 Y_{1\Delta}$$

$$W_1 = Y'_{1\Delta} Y_{1\Delta} - (Y'_{1\Delta} X)(X'X)^{-1} X'Y_{1\Delta}$$

The limited information maximum likelihood method is to choose such values of the coefficient that would minimize 'l'. Thus, on differentiating 'l' with respect to $a_{1\Delta}$ we have

$$\frac{\partial l}{\partial a_{1\Delta}} = \frac{2(W_1^* a_{1\Delta})(a'_{1\Delta} W_1 a_{1\Delta}) - 2(W_1 a_{1\Delta})(a'_{1\Delta} W_1^* a_{1\Delta})}{(a'_{1\Delta} W_1 a_{1\Delta})^2} \quad \dots (2.39)$$

The necessary condition for minimization requires that the first order derivative is equated to zero. Therefore,

$$\frac{2}{(a'_{1\Delta} W_1 a_{1\Delta})} \left[W_1^* a_{1\Delta} - \left(\frac{a'_{1\Delta} W_1^* a_{1\Delta}}{a'_{1\Delta} W_1 a_{1\Delta}} \right) W_1 a_{1\Delta} \right] = 0$$

Using equation (2.38) we get,

$$W_1^* a_{1\Delta} - l W_1 a_{1\Delta} = 0 \quad \dots (2.40)$$

Now, for $a_{1\Delta} \neq 0$, the determinant of the matrix formed must be equal to zero, i.e.,

$$|W_1^* - l W_1| = 0 \quad \dots (2.41)$$

The elements of W_1^* and W_1 can be obtained from the sample observations (as defined for (2.38)). The equation (2.41), therefore, becomes a polynomial of m degree in l

which must be solved for the smallest root \hat{l} . This may also be considered as an eigen value problem (Krishnamurthy & Sen, pp. 248-249). Substituting the value of \hat{l} thus obtained in equation (2.40) and solving the simplified equation

$$(W_1^* - lW_1)\hat{a}_{1\Delta} = 0 \quad \dots (2.42)$$

the estimator $\hat{a}_{1\Delta}$ can be obtained. Next, the elements of b_1 can be obtained by

$$\hat{b}_1 = (X_1'X_1)^{-1}(X_1'y_1^*) = (X_1'X_1)^{-1}X_1'Y_{1\Delta}\hat{a}_{1\Delta} \quad \dots (2.43)$$

Equations (2.42) and (2.43) define the LIML estimates of the structural equation.

The equation-by-equation estimation methods, namely GILS, ILS, 2SLS and LIML discussed so far, are all essentially limited information estimators. In the estimation of any structural equation, these estimators do not take into account complete information on all structural equations in the model. Although the estimators give consistent estimates, they are not asymptotically efficient, as they do not consider the correlation of disturbances across the equation. This deficiency can be overcome by estimating all the equations of the system simultaneously. For this purpose the full information methods such as 3SLS or FIML can be used (Intriligator, pp. 402-416; Johnston, pp. 486-492; Theil, pp. 508-528). Since we have not used these estimators in our investigation, we do not consider it necessary to describe them here.

DasGupta, Madhuchhanda (2004) LEAST ABSOLUTE DEVIATION ESTIMATION OF MULT-EQUATION LINEAR ECONOMETRIC MODELS : A STUDY BASED ON MONTE CARLO EXPERIMENTS, Doctoral Dissertation, Dept. of Economics, North Eastern Hill University, Shillong (India); Supervisor – Prof. SK Mishra

CHAPTER 3

LAD ESTIMATION: A LITERATURE SURVEY

I. Introduction: As it has been mentioned before, the Least Squares method of estimation of parameters of linear (regression) models performs well provided that the residuals (disturbances or error) are well behaved (preferably normally or near-normally distributed and not infested with large size outliers) and follow Gauss-Markov assumptions. However, models with the disturbances that are prominently non-normally distributed and contain sizeable outliers fail estimation by the Least Squares method. An intensive research has established that in such cases estimation by the Least Absolute Deviation method performs well.

This chapter has two-fold objectives, the first relates to the investigation in the properties of the Least Absolute Deviation (LAD) estimator in estimating regression models and the second relates to the details of the numerical algorithms to compute the regression coefficients by minimizing the sum of absolute disturbances. First, we address to the survey of literature on the study of the properties of the LAD estimator (in single as well as multi-equation models) together with the numerical algorithms to estimate the regression coefficients by this method. Subsequently we deal with the details of different numerical algorithms (linear programming based methods, iterative methods and non-linear search methods) developed for computing the coefficients of a (linear) regression model.

Estimation of the parameters of a (linear) regression equation is fundamentally a problem of finding solution to an over-determined and inconsistent system of (linear) equations. The over-determined system of equations is inherently an inconsistent system

for it cannot have any solution that exactly satisfies all the equations. Therefore, the ‘*solution*’ leaves most of the equations (if not all) unsatisfied by a quantity (of either sign) called the residual or the error terms. It is held that these residuals should be as small as possible and this fact determines the quality of the ‘*solution*’. It is accomplished by minimization of a particular norm of the residual vector, $\|e\|$, in the sample.

II. Survey of Literature on Properties and Algorithms of LAD Estimator: The method to solve an over-determined system of (linear) algebraic equations dates back to **KF Gauss** and **PS Laplace** as mentioned by **Taylor** (1974). These mathematicians suggested (and used) the method of Least Squares, which minimizes the sum of square of residuals in the equation (which amounts to minimization of Euclidean Vector norm of the residuals). They also suggested (and used) the method of Least Absolutes, which minimizes the sum of absolute residuals in the equations (which amounts to minimization of absolute norm of the residuals). In the sense of *Minkowski norm*, the methods of Least Squares (L_2) and Least absolute (L_1) are expressed as

$$\text{Min (S)} = \underset{\tilde{a}}{\text{Min}} \left[\left(\sum_{i=1}^n \left| y_i - \sum_{j=1}^k \tilde{a}_j X_{ij} \right|^p \right)^{\frac{1}{p}} \right] \text{ for } p=2 \text{ and } p=1, \text{ respectively.}$$

The Least Squares method is computationally convenient because minimization of *Euclidean norm* is amenable to calculus methods. This convenience led to its popularity. On the other hand, there is a lot of mathematical difficulty in working with absolute value functions on account of their lack of amenability to calculus methods.

Econometricians generally take for granted that the error terms implicit in the models are generated by distributions having a finite variance. However, since the time of

Pareto it has been known that the distribution of income bears evidence that the error distribution could have infinite variance. Works of many econometricians, namely, **Meyer & Glauber** (1964), **Fama** (1965) and **Mandlebroth** (1967) on economic data series like prices in financial and commodity markets indicate that infinite variance distributions exist abundantly. The distribution of firms by size, behaviour of speculative prices and various other recent economic phenomena also display similar trends.

An infinite variance means fat tails and fat tails mean a lot of outliers. Since the method of least squares places heavy weights on the error terms, we look to an alternative, more robust estimator, which minimizes the absolute values and not the squared values of the error term. The Least Absolute Deviation (LAD) estimator, suggested by **Gauss** and **Laplace**, is such an estimator that minimizes the absolute value of the disturbance term. This estimator measures the error term as the absolute distance of the estimated values from the true values and belongs to the median family of estimators.

The recommendations for using L_1 norm may be traced back to **Edgeworth** (1887, 1888, 1923), **Rhodes** (1930) and **Singleton** (1940). Edgeworth and Rhodes pointed out that random sampling and normal distribution, which are needed to justify the method of Least Squares, as an optimal method, often does not exist. In other circumstances least squares may give undue weights to extreme observations. The method proposed by Edgeworth applies to only two variables. The methods of Rhodes and Singleton, while extending the proposals of Edgeworth to more than two dimensions, become extremely unwieldy as the dimension of the model increases.

The Linear Programming based algorithm to LAD Estimation: With the advent of linear programming as a mathematical method to solve ‘corner-optimum’ problem that defies

solution by calculus method, feasibility of parameter estimation by minimization of the sum of absolute residuals received a major breakthrough. **Charnes, Cooper & Ferguson** (1955) showed that a certain management problem involving a minimization of absolute values could be transformed to standard linear programming form by employing the device of representing a deviation as the difference between two non-negative variables. The paper by **Charnes *et al.*** (1955) is considered to be a seminal paper for giving a new lease of life to L_1 regression. **Fisher** (1961) showed how a curve can be fitted with minimum absolute deviation (rather than squared deviations) using linear programming. His article reviewed the formulation of this application of linear programming. Fisher pointed out that to fit a linear function

$$\hat{X}_1 = a_0 + a_{12}X_2 + \dots + a_{1K}X_K \quad \dots (3.1)$$

to the observations

$$\begin{bmatrix} X_{11} & X_{12} & \dots & X_{1K} \\ X_{21} & X_{22} & \dots & X_{2K} \\ \dots & \dots & \dots & \dots \\ X_{n1} & X_{n2} & \dots & X_{nK} \end{bmatrix}$$

(X_{ij} representing observation i on variable j , and $n > k$) by minimizing the sum of absolute deviations

$$S = \sum_{i=1}^n |X_{i1} - \hat{X}_{i1}|$$

The parameters in equation (3.1) can be expressed as

$$\begin{aligned} a_0 &= y_1 - Z_1 \\ a_{12} &= y_2 - Z_2 \\ &\dots \dots \dots \\ a_{1K} &= y_K - Z_K \end{aligned} \quad \dots (3.2)$$

$$X_1 = f_2(X_2) + \dots + f_K(X_K).$$

It can be transformed into a linear function by making transformations of the independent variable as is done in least squares regression. If unequal weighting of the observed data is desired in the fitting, the desired weights, rather than unit weighting, should be inserted as coefficients in the objective function (3.5).

Ashar & Wallace (1963) studied the statistical properties of regression parameters estimated by minimization of L_1 norm. **Huber** (1964) explored the properties of L_1 regression in its robustness to wild fluctuations in the magnitude of residual elements.

Meyer & Glauber (1964) for the first time directly compared L_1 and L_2 regression estimators. They estimated their investment model by minimization of L_1 as well as L_2 norm and tested the regression equations obtained on post-sample data by using those equations to forecast the nine (in some cases eleven) observations subsequent to the period of fit. They found that with very few exceptions, the equations estimated by L_1 minimization outperformed the ones estimated by L_2 minimization even on criteria (such as sum of squared forecast errors) with respect to which, L_2 regression is ordinarily thought to be remarkably suitable or optimal.

Rice & White (1964) compared L_1 , L_2 and L_∞ (minimization of maximum deviation) norms for a single equation model. In their paper they observed that for the important problem of smoothing and estimation in the presence of wild points (outliers), the L_1 norm appears to be markedly superior among the L_p ($1 \leq p \leq \infty$) norms.

Usow (1967) studied the L_1 approximation for discrete functions and the discretization effects while the functions in original are continuous. This paper is more concerned with approximation of functions using L_1 norm rather than estimation of

regression parameters and their statistical properties, though its mathematical approach has a significant bearing on the development of statistical theory relating to properties of L_1 regression.

In 1973 **Barrodale** and **Roberts** presented an algorithm for L_1 -approximation by modifying the simplex method of linear programming, which is computationally superior to the algorithms given by Usow, Robers and Robers and Dantzig. The algorithm is an improved version of the primal algorithm described by Barrodale and Young in 1966. In the improved version, Barrodale and Roberts were able to significantly reduce the total number of iterations required by discovering how to pass through several neighbouring simplex vertices in a single iteration.

In the paper, the general l_1 -linear approximation problem was stated as follows: Let $f(x)$ be given real-valued function defined on a discrete subset $X=\{x_1, x_2, \dots, x_m\}$ of Euclidean space E^n . Given $n(\leq m)$ real-valued function $\phi_j(x)$ defined on X , a linear approximating function $L(A, x) = \sum_{j=1}^n a_j \phi_j(x)$ was formed for any set $A=\{a_1, a_2, \dots, a_n\}$ of real numbers. The l_1 -problem is to determine a best $L(A^*, x)$ which minimizes

$$\sum_{i=1}^m |f(x_i) - L(A, x_i)| \quad \dots (3.6)$$

The l_1 -problem (3.6) was restated as a linear programming problem. For the l_1 -problem it was assumed $\phi_{ji} \equiv \phi_j(x_i)$, $f_i \equiv f(x_i)$, and positive variables e_i , v_i , b_j , c_j were defined by putting $f_i - \sum_{j=1}^n a_j \phi_{ji} = e_i - v_i$, $i=1, 2, \dots, m$ and $a_j = b_j - c_j$ for $j=1, 2, \dots, n$. Then a best l_1 -approximation corresponds to an optimal solution to the primal of the linear programming problem:

$$\text{Minimize } \sum_{i=1}^m (e_i + v_i)$$

$$\text{subject to } f_i = \sum_{j=1}^n (b_j - c_j) \phi_{j,i} + e_i - v_i, i = 1, 2, \dots, m \text{ and } b_j, c_j, e_i, v_i \geq 0 \quad \dots (3.7)$$

It was proved that if the column rank of the $m \times n$ matrix $\Phi = \{\phi_{ji}\}^T$ is $K (\leq n)$ then there exists a best l_1 - approximation which interpolates $f(x)$ in at least K points on X .

Authors like **Wagner** and **Robinowitz** (for example) have suggested that the dual of the problem should be solved when m is large. However, it is found that an application of bounded-variable simplex method of Dantzig to the dual problem leads to a less efficient algorithm in general than solving the primal problem by their version of the standard form of the simplex method.

Although several alternatives to the standard form of the simplex method can be used to solve a linear programming problem (two forms of the revised simplex method, the primal-dual algorithm, the dual simplex algorithm, etc.), the denseness of the condensed tableau corresponding to (3.7), the availability of an initial basic feasible solution, and the simplicity with which the idea of passing through several vertices in a single iteration can be implemented, combine together to make the standard form of the simplex method the most economical algorithm for l_1 -problem.

An inspection of the linear programming problem reveals that (i) an initial basic feasible solution is immediately available and (ii) only n columns are needed to store the information contained in the right-hand side of the equality constraints. Thus denoting the columns of the simplex tableau corresponding to (3.7) by R, b_j, c_j, e_i and v_i , an initial basis is provided by e_1, e_2, \dots, e_m whenever each f_i is positive. If an f_i is negative, the sign of the corresponding row is changed and e_i is replaced by v_i in the basis. It is also assumed

that $b_j = -c_j$ and $e_i = -v_i$, and that the sum of the marginal (or reduced) costs of b_j and c_j is zero and of e_i and v_i is -2 . Thus the condensed form of the simplex method (in which the basis is suppressed) can be applied to just n columns, which initially contain b_1, b_2, \dots, b_n . Within an array of dimensions $(m+2) \times (n+2)$ which initially contains the data in Table-1 (including labels for the basic and nonbasic vectors), the simplex iterations can be performed. The program requires additional arrays totaling $3m+n$ words: these are used as workspace and for storing output information.

**Table-1. Initial condensed simplex tableau for the algorithm
(assuming each f_i is nonnegative)**

Basis	R	b_1	b_2	...	b_n
e_1	f_1	$\phi_{1,1}$	$\phi_{2,1}$...	$\phi_{n,1}$
e_2	f_2	$\phi_{1,2}$	$\phi_{2,2}$...	$\phi_{n,2}$
.
e_m	f_m	$\phi_{1,m}$	$\phi_{2,m}$...	$\phi_{n,m}$
Marginal Cost	$\sum_{i=1}^m f_i$	$\sum_{i=1}^m \phi_{1,i}$	$\sum_{i=1}^m \phi_{2,i}$...	$\sum_{i=1}^m \phi_{n,i}$

The algorithm is implemented in two stages. In Stage-1 the choice of pivotal column is restricted to the vectors b_j and c_j during the first n iterations. The vector with the largest marginal cost is chosen to enter the basis. From among the basic vectors e_i and v_i the vector which causes the maximum reduction in the objective function is chosen to leave the basis. At the end of Stage-1, the rank $K(\leq n)$ of the matrix Φ is determined by the total number of vectors b_j, c_j in the basis. Since K of the vectors e_i (or v_i) have been removed from the basis, the current simplex tableau represents an approximation, which interpolates at least K data points. If the approximation interpolates more than K data

points, then the simplex tableau is degenerate: this does not cause any problems in practice.

In Stage-2 non-basic e_i or v_i are interchanged with basic e_i or v_i : the basic b_j and c_j vectors are not allowed to leave the basis during this stage. The vectors entering and leaving the basis are chosen in a similar manner as in Stage-1. The algorithm terminates when all the marginal costs are negative. In Stage-2 each simplex tableau corresponds to an approximation, which interpolates K data points (assuming nondegeneracy). $K-1$ of these points remains fixed in each iteration. The vector entering the basis determines which point is to be dropped from the interpolating set while the vector leaving the basis determines the new point of interpolation.

Since some of the basic vectors b_j and c_j can have negative values associated with them the final tableau at the end of Stage-2 may be infeasible. The solution can be made feasible and optimal by interchanging such basic vectors b_j (or c_j) with the corresponding nonbasic vectors c_j (or b_j).

The main modification of the simplex method is in choosing the vector e_i or v_i to leave the basis. The vector that causes the maximum reduction in the objective function is chosen in both the stages.

The study conducted by **Oveson** (1968) on the LAD estimator gave a new thrust to the investigation into the properties and applicability of the estimator. It had also been almost well established that in the presence of errors generated by thick-tailed distribution, L_1 regression performed better than L_2 regression.

In 1969 **Roberts and Ben-Israel** applied a new method for linear programming to the dual formulation of the l_1 -problem. Their new method (which they call interval linear

programming) is capable of solving any bounded-variable linear programming problem, and so it is natural to apply it to the l_1 -problem in particular.

In 1971 **Abdelmalek** described an algorithm, which determines best l_1 -approximations as the limit of best lp -approximations as $P \rightarrow 1^+$. His technique thus obtains a solution to a linear problem by solving a sequence of non-linear problems.

Indirect or Iterative Algorithms: **Spyropoulos, Kiountouzis & Young** (1973) and **Abdelmalek** (1974) made a progress in this direction. **Schlossmacher** (1973) and **Fair** (1974) also proposed an improved algorithm for L_1 estimation that is very similar to iterative weighted least squares. A common problem with the iterative least squares procedure is that, in any given iteration, some of the residuals may be zero or very close to zero, thereby, making construction of weights difficult. Fair and Schlossmacher dealt with this problem in different ways. When a residual was less than 0.00001, Fair set it equal to 0.00001 while Schlossmacher ignored the observation, at least for the given iteration, by setting the weight equal to zero. Although the two solutions are to some extent contradictory, both authors reported satisfactory results in their empirical work.

Along these efforts, a number of works using Monte Carlo method of simulation to compare the sampling properties of L_1 regression with new alternatives to L_2 norm estimators were done. **Blattberg & Sargent** (1971) pointed out that if the disturbances follow a two-tailed exponential distribution with density function

$$f(e_i) = (2\lambda)^{-1} \exp\left\{-\frac{|e_i|}{\lambda}\right\} \quad \dots (3.8)$$

then maximization of the likelihood function is equivalent to minimization of $\sum_{i=1}^n |e_i|$ and

so the least absolute deviation estimator becomes the maximum likelihood estimator. The

superiority of L_1 norm estimator over L_2 norm estimator in finite samples, when errors follow the density in (3.8) was confirmed in a Monte Carlo study by **Smith and Hall** (1972).

Taylor (1974) gave the condition under which the L_1 norm estimator is unbiased and consistent and discussed some of the problems encountered when trying to establish a distribution theory, under the assumptions that (i) e_i are independent, identically distributed random variables with a continuous distribution function F and median zero, and (ii) $\lim_{n \rightarrow \infty} n^{-1} X'X = Q$ is a positive definite matrix.

Nyquist & Westlund (1977) compared the two estimators (L_1 and L_2 norm estimators) with regard to their statistical properties. In 1978, **Bassett & Koenker** developed the asymptotic theory of Least absolute error regression. Their article resolved a long-standing open question concerning the LAE (alias LAD) estimator by establishing its asymptotic normality under general conditions, thereby extending a result of **PS Laplace** to the general linear model. *The result confirmed that for the general linear model the LAE estimator is a natural analog of the sample median.* The authors proved that in the general linear model with independent and identically distributed errors and distribution function F , the estimator which minimizes the sum of absolute residuals is demonstrated to be consistent and asymptotically Gaussian with covariance matrix

$W^2 Q^{-1}$, where $\lim_{n \rightarrow \infty} n^{-1} X'X = Q$ and W^2 is the asymptotic variance of the ordinary

sample median from samples with distribution F . This indicated that for any error distribution for which median is more efficient than the mean as an estimator of location, the least absolute error estimator has smaller asymptotic ellipsoids than the least squares

estimator and therefore is more efficient than LS estimator. In the paper, a number of equivariance properties of LAE estimator was stated and proved. It was proved that the LAE estimator is affine equivariant, scale and shift equivariant and equivariant to reparameterization of design. Though Least Squares estimator shares the same properties, typically robust alternatives to least squares are not equivariant in one or more of the above senses.

It was also observed that in a scatter of sample observations in \mathbb{R}^2 with the LAE solution line slicing through the scatter, as long as the moving observations lie on the same side of the original line, the solution is unaffected. This property is not shared by least squares, and although obvious in the case of median, it seems to capture part of the intuitive flavour of LAE's median-type robustness and insensitivity to outlying observations.

Powell (1984) proposed an alternative to maximum likelihood estimation of the parameters of the Censored Regression Model. He generalized the Least Absolute Deviations estimation for the standard linear regression model. The estimator was found by minimizing $\sum |y_i - \max(0, x_i s \beta)|$.

In the paper, he showed that the Censored Least Absolute Deviation (CLAD) estimator is robust to heteroskedasticity and is consistent and asymptotically normal for a wide class of error distribution. Consistency of the asymptotic covariance matrix was also proved. As a consequence, tests of hypothesis concerning the unknown regression coefficient can be constructed which are valid in large samples. He also opined that the Censored LAD estimator can be computed using "direct search" methods developed for nonlinear programming.

Pollard (1991) presented an alternative approach for studying the asymptotic theory of Least Absolute Deviation (LAD) estimator in a simple regression context. The approach was built on the convexity of the LAD criterion function to construct a quadratic approximation whose minimand is close enough to the LAD estimator for the latter to share the same asymptotic normal distribution.

Phillips (1991) presented the asymptotic theory for the LAD estimator in a regression model setting using generalized functions of random variables and generalized Taylor series expansions. The approach was justified by the smoothing that was delivered in the limit by the asymptotics, whereby the generalized functions were forced to appear as linear functionals wherein they became real valued. He studied models with fixed random regressors, and autoregressions with infinite variance errors and a unit root. His approach enabled the development of higher order asymptotic expansion of the distribution of the LAD estimator. The results obtained also showed that the LAD estimator converges at a faster rate in the unit root model for $0 < \alpha < 2$ than the OLS estimator.

Weiss (1991) established that it was possible to use the Least Absolute Deviation (LAD) estimator to estimate the parameters of a nonlinear dynamic model. He considered a model given by

$$y_t = g(x_t, \beta_0) + e_t$$

where g = a known function

$$x_t = (y_{t-1}, \dots, y_{t-p}, z_t)$$

z_t = vector of exogenous variables

$\beta_0 = (k \times 1)$ vector of unknown parameter

e_t = unobserved error term which satisfies median $(e_t/I_t) = 0$

$I_t = \sigma -$ algebra (information set at period t) generated by $\{x_{t-i}\} (i \geq 0)$ and

$$\{e_{t-i}\} (i \geq 1).$$

The Nonlinear Least Absolute Deviations (NLAD) estimator was defined as the solution of the problem:

$$\min_{\beta} Q_T(\beta) \equiv \min_{\beta} \frac{1}{T} \sum_{t=1}^T |y_t - g(x_t, \beta)|$$

The author investigated the model and proved theoretically that the NLAD estimator $\hat{\beta}$ was consistent and asymptotically normal under certain assumptions.

Chen (1996) investigated the linear regression model

$$Y_i = x_i' \beta_0 + e_i ; \quad 1 \leq i \leq n, \quad n \geq 1$$

under the assumptions that the random error e_i belongs to a certain class F of distributions in \mathbb{R}^∞ , that each e_i has a unique median zero and for each e_i there must be at least linear accumulation of probability in the vicinity of zero. He showed that the sufficient

condition $d_n \equiv \max_{1 \leq i \leq n} x_i' \left(\sum_{j=1}^n x_j x_j' \right)^{-1} x_i = O\left(1/\log n\right)$ for strong consistency of the

LAD estimate $\hat{\beta}_n$ of β_0 given by Chen et al. (1992) fails. The author proved that for any constant sequence $D_n \uparrow \infty$, the condition $d_n = O(D_n/\log n)$ is no longer sufficient.

Combining recent advances in interior point methods for solving linear programs with a new statistical preprocessing approach for l_1 -type problems, **Portnoy and Koenker** (1997) obtained a 10 to 100 fold improvement in computational speeds over current (simplex-based) l_1 algorithms in large problems, demonstrating that l_1 methods

can be made competitive with l_2 methods in terms of computational speed throughout the entire range of problem sizes.

Breidt, Davis and Trindade (2000) studied the Least Absolute Deviation estimation for All-Pass time series models. An All-Pass time series model is an autoregressive moving average model in which all the roots of the autoregressive polynomial are reciprocals of roots of the moving average polynomial and vice versa. The uncorrelated (white noise) time series generated by the All-Pass models are not independent in the non-Gaussian case. The authors opined that an approximation to the likelihood of the model in the case of Laplace (two-sided exponential) noise yields a modified absolute deviation criterion, which can be used even if the underlying noise is not Laplacian. They established the asymptotic normality for LAD estimators of the model parameters under general conditions. Behaviour of the estimators in finite samples was also studied via simulation.

Furno (2000) compared the performance of LAD and OLS in the linear regression model with random coefficient autocorrelated (RCA) errors. The presence of thick tailed error distribution led to the estimation of the RCA model by Least Absolute Deviation (LAD) estimator. It is known that when error follows a double exponential distribution, LAD coincides with maximum likelihood. In all other cases, the estimator is less affected by observations coming from tails, since it minimizes the absolute value and not the squared value of the residuals. In case of leptokurtic error distribution, the LAD estimator is particularly useful. Furno proved that the LAD estimator for randomly autocorrelated errors is asymptotically normal. The more general random coefficient ARMA models for the error term was also considered in the study and the resulting

heteroskedasticity was analysed. Monte Carlo experiments revealed that LAD improved upon OLS in case of RCA errors, both in terms of bias reduction and efficiency gains. However, in the case of constant autocorrelation model, the results confirmed that LAD is not advantageous, especially in small samples, since its sampling distribution differs from the asymptotic one.

Hitomi and Kagihara (2001) proposed a NSLAD (nonlinear Smoothed LAD) estimator that is practically computable and has the same asymptotic properties as the NLAD estimator in Weiss' (1991) nonlinear dynamic model. Monte Carlo experiments were conducted to compare the performance of the NSLAD and the nonlinear least-squares (NLS) estimators. In the study two types of error distributions were considered – standard normal distribution where the NLS estimator becomes MLE and the Laplace distribution where the NLAD estimator is MLE. The results reported indicate that as the sample size increases the bias becomes negligible and the difference between NSLAD and NLS estimators ceases. While the NLS estimator was found to have a smaller standard deviation when the error term's distribution was standard normal, the NSLAD estimator had a smaller standard deviation when the error term followed Laplace distribution. No difference was found in the performance of the two estimators with respect to median and quartiles. Although NLS had a marginal edge over NSLAD as far as computation time was concerned, NSLAD was found to take relatively lesser time when the error term followed Laplace distribution.

Sakata (2001) proposed a general estimation principle based on the assumption that instrumental variables (IV) do not explain the error term in a structural equation. He opined that unlike the IV estimators such as two-stage least squares estimator, the

estimators based on the proposed principle are independent of the normalization constraint. Based on this new principle, he proposed the L_1IV estimator, which is an IV estimation counterpart of the LAD estimator. The author investigated the asymptotic properties of the L_1IV estimator. A consistent estimator of its asymptotic covariance matrix and a consistent specification test based on the L_1IV estimator were proposed. The problem of identification in L_1IV estimation was also discussed.

Estimation of Multi-equations Models: By the middle of 1960's, multi equation econometric models and techniques used for estimating their parameters had already gained a solid ground. The method of limited information maximum likelihood (LIML) was developed in the late 1940's (**Haavelmo**, 1947). But the use of least squares method for estimation of parameters of a multi-equation econometric model had to wait until **H Theil** (1953) used repeated least squares to estimation of parameters of a regression equation in the multi-equation model. **RL Basmann** (1957) used least squares repeatedly for estimating parameters of a multi-equation linear econometric model. **H Theil** (1961) developed the method of Two-Stage Least Squares (2-SLS). Very soon **A Zellner** and **H Theil** (1962) developed Three-Stage Least Squares (3-SLS) also.

It would be befitting to describe the nature and the issues related with the estimation of multi-equation (linear) models once again (we have already described them in chapter-2). We are of the opinion that recapitulation will help us in pin-pointing the nodes at which the least squares technique may be replaced by LAD.

A multi-equation (linear) system may be described as $YA + XB + E = 0$, where, $Y(n,m)$ is a matrix representing m number of *endogenous variables* each in n number of observations, $X(n,k)$ is a matrix representing k number of *predetermined* variables each in

n observations, $E(n,m)$ is a matrix representing m number of *stochastic vectors* (error terms in the model) each in n elements and $0(n,m)$ is a null matrix in n rows and m columns. Associated with Y and X there are the coefficient matrices, $A(m,m)$ and $B(k,m)$ respectively, called the *structural coefficients matrices*. It is assumed that the model $YA + XB + E = 0$ is complete, which implies that the model has as many (linearly independent) equations as the number of endogenous variables and the matrix A is a regular (not singular) matrix. While Y is a matrix of stochastic vectors ($Y = \Upsilon + \varepsilon$, where Υ is the matrix of true endogenous variables and ε , different from E , is the matrix of disturbances), some, but not all, of the vectors in X may be stochastic ($X_j = \mathbb{X}_j + v_j$). In case X is a non-stochastic vector, it is called an exogenous variable. It is also pertinent to note that the structural coefficient matrix A has a special structure such that the elements in its principal diagonal are all minus unity (-1) or $a_{ij} = -1 \ \forall \ i = j$. Further, depending on the nature of the model, A may be diagonal (that is $A = -I$, a negatively signed identity matrix), lower triangular (where $a_{ij} = 0 \ \forall \ i < j$) or upper triangular (where $a_{ij} = 0 \ \forall \ i > j$) characterizing a recursive model, block-diagonal, or finally a regular one (which characterizes a true simultaneous model).

Empirically, we collect data on Y and the exogenous variables (that may make a full or partial X). Thus, empirical Y has two strains of error or $Y = \Upsilon + \varepsilon + E$. It is assumed that v_j (in the pre-determined variables comprising X) and E_j are orthogonal (linearly independent).

Our objective is to estimate A and B . First, we simplify our model by a transformation of its equations (called structural equations) into another type of equations (called reduced form equations) in which $Y - XP - \eta = 0$. This transformation is effected

by post-multiplying our structural model $YA + XB + E = 0$ by the inverse of the coefficients matrix A . That is:

$$YAA^{-1} + XBA^{-1} + EA^{-1} = 0A^{-1} \quad \text{or} \quad Y - X\Pi - \eta = 0, \text{ where, } \Pi = -BA^{-1} \text{ and } \eta = -EA^{-1}.$$

The reduced form model $Y - X\Pi - \eta = 0$ may be rewritten as $Y = X\Pi + \eta$. Having assumed that the stochastic terms in X , if any, and E are orthogonal, it is obvious that vectors in η and v are orthogonal across η and v . This result prompts us to estimate Π by a suitable method such as the method of ordinary least squares (OLS). The OLS estimator of Π is given by:

$$P = (X'X)^{-1} X'Y \quad \text{or} \quad P = \{(X'X)^{-1} X'\}Y.$$

In $P = \{(X'X)^{-1} X'\}Y$, factor $\{(X'X)^{-1} X'\}$ has a special interpretation. It is the generalized inverse (more exactly, the least squares g-inverse) of X . That is, $X^{-g} = [X'X]^{-1} X'$, such that $X^{-g} X = \{(X'X)^{-1} X'\}X = (X'X)^{-1} X'X = I$ and $XX^{-g} = X\{(X'X)^{-1} X'\} = I_d$, an idempotent matrix. Having obtained P , one may obtain the expected Y by the relationship $\hat{Y} = XP$.

However, our objective was to estimate A and B , and instead, we have estimated $\Pi = -BA^{-1}$. The question is: can we obtain, through some algebraic manipulation, the estimated A and B (that is, \hat{A} and \hat{B}), and if the answer is in an affirmative, then under what conditions can we obtain \hat{A} and \hat{B} ? This is the problem of identification.

It is obvious that if each column of A as well as B could be known, A and B in full can be known. Hence, we will try to answer the question posed above for a particular equation (say, r^{th} one) in the model $YA + XB + E = 0$. Since $\Pi = -BA^{-1}$, it implies $P = -\hat{B}\hat{A}^{-1}$ or $\hat{P}\hat{A} = -\hat{B}$. For the r^{th} equation, only the respective r^{th} columns of \hat{A} and \hat{B}

would be used. Thus, for the r^{th} equation we solve the system of equations given by $P\hat{a}_r = -\hat{b}_r$, where \hat{a}_r and \hat{b}_r are referring to the r^{th} columns of the expected A and B matrices respectively.

Since A is an $m \times m$ matrix and B is a $k \times m$ matrix, $\Pi = -BA^{-1}$ is a $k \times m$ matrix. Therefore, the expression $P\hat{a}_r = -\hat{b}_r$ is a system of k (linear) equations involving $m+k$ unknowns. It is obvious that we cannot (uniquely) determine $m+k$ unknowns and thus the system of equations $P\hat{a}_r = -\hat{b}_r$ is indeterminate.

We may proceed further by augmenting the system with m number of additional (independent) equations in $\hat{a}_{sr} \in \hat{a}_r$ or $\hat{b}_{tr} \in \hat{b}_r$ (or both). The most straightforward way to do that is to set some μ_r unknowns ($\hat{a}_{sr} \in \hat{a}_r$ or $\hat{b}_{tr} \in \hat{b}_r$ or both) equal to zero. It amounts to zero restriction on some μ_r structural coefficients in the r^{th} structural equation. It is obvious that $\mu_r \geq m$, else the problem is indeterminate. In case $\mu_r = m$, we have as many equations as the unknowns, and further assuming that no equation is linearly dependent on the others, the unknowns (remaining after the zero restriction) can uniquely be determined. In this case we say that the equation $P\hat{a}_r = -\hat{b}_r$ is *exactly identified*. However, if $\mu_r > m$, we have the equations larger in number than the unknowns, and the system of equations is over-determined. Generally, such an over-determined system is also inconsistent. That is to say that the solutions (values of the unknowns obtained from such an over-determined system of equations) do not satisfy all the equations. In this case we say that the equation $P\hat{a}_r = -\hat{b}_r$ is *over-identified*.

To formalize what we have mentioned above, let us categorize the elements of \hat{a}_r and \hat{b}_r into two (disjoint) categories, namely, the unknown ones and the known ones. We will use the subscripts 1 and 2 (respectively) to identify them. Thus, $[\hat{a}_r]_1$ and $[\hat{b}_r]_1$ are the partitioned vectors (columns) of $[\hat{a}_r]$ and $[\hat{b}_r]$, whose elements are some (say, m_1 and k_1 respectively) unknown quantities. Similarly, $[\hat{a}_r]_2$ and $[\hat{b}_r]_2$ are the partitioned vectors (columns) of $[\hat{a}_r]$ and $[\hat{b}_r]$, whose elements are ($m_2 = m - m_1$ and $k_2 = k - k_1$ respectively) known quantities. Note that in order to avoid the under-identifiability of the r^{th} structural equation it is necessary that $k_1 + m_1 = \mu_r \geq m$.

We have mentioned earlier that the elements in the principal diagonal of matrix A are all minus unity ($a_{ii} = -1 \ \forall \ i$). Presently, we are concerned with the r^{th} column of the matrix A. Thus, the r^{th} element of $[\hat{a}_r] = a_{rr} = -1$. In our scheme of categorized partition, therefore, the element a_{rr} would belong to $[\hat{a}_r]_2$. Further, due to zero restriction on the coefficients all the rest elements of $[\hat{a}_r]_2$ are zero and all the elements of $[\hat{b}_r]_2$ are zero.

In the said scheme of categorized partition it would be helpful to use the permutation matrix operation on $[\hat{a}_r]$ and $[\hat{b}_r]$. Let $G(m,m)$ be the permutation matrices obtained by permutating the columns of the identity matrix $I(m,m)$ and let $H(k,k)$ be the permutation matrix obtained by permutating the rows of the identity matrix $I(k,k)$ such that:

$$G[\hat{a}_r] = \frac{[\hat{a}_r]_1}{[\hat{a}_r]_2} \quad \text{and} \quad H[\hat{b}_r] = \frac{[\hat{b}_r]_1}{[\hat{b}_r]_2}$$

Therefore, the system of equations $P\hat{a}_r = -\hat{b}_r$ is transformed (rearranged) as follows:

$$H P G^{-1} G[\hat{a}_r] = H[\hat{b}_r].$$

Due to pre-multiplication of P by H , the rows of P are permuted in correspondence with $H[\hat{b}_r]$ and due to post-multiplication of P by G^{-1} , the columns of P are permuted in accordance with $G[\hat{a}_r]$. Let us rename $H P G^{-1}$ as Q . It is to be noted that Q is numerically known since P , G and H are all known. Then,

$$Q \frac{[\hat{a}_r]_1}{[\hat{a}_r]_2} = \frac{[\hat{b}_r]_1}{[\hat{b}_r]_2}, \text{ or}$$

$$\begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} \begin{bmatrix} [\hat{a}_r]_1 \\ [\hat{a}_r]_2 \end{bmatrix} = \begin{bmatrix} [\hat{b}_r]_1 \\ [\hat{b}_r]_2 \end{bmatrix}.$$

In this scheme, Q_{11} is a $k_1 \times m_1$ matrix, Q_{12} is a $k_1 \times m_2$ matrix, Q_{21} is a $k_2 \times m_1$ matrix and Q_{22} is a $k_2 \times m_2$ matrix. This gives us two equations:

$$[Q_{11}] [\hat{a}_r]_1 + [Q_{12}] [\hat{a}_r]_2 = [\hat{b}_r]_1$$

$$[Q_{21}] [\hat{a}_r]_1 + [Q_{22}] [\hat{a}_r]_2 = [\hat{b}_r]_2$$

Since $[\hat{a}_r]_2$ and $[\hat{b}_r]_2$ are known (more specifically, only one element of $[\hat{a}_r]_2$ is -1 and other elements are zero, and all the elements of $[\hat{b}_r]_2$ are zero), $[Q_{21}]^g [[\hat{b}_r]_2 - [Q_{22}] [\hat{a}_r]_2] = [\hat{a}_r]_1$. In particular, since $[\hat{b}_r]_2 = [0]$, we have $[\hat{a}_r]_1 = -[Q_{21}]^g [Q_{22}] [\hat{a}_r]_2$. Once $[\hat{a}_r]_1$ is obtained, one may subsequently obtain $[\hat{b}_r]_1$ in $[Q_{11}] [\hat{a}_r]_1 + [Q_{12}] [\hat{a}_r]_2 = [\hat{b}_r]_1$ by substitution.

In case $[Q_{21}]$ is a square matrix of full rank (the r^{th} equation is exactly identifiable), $[Q_{21}]^g = [Q_{21}]^{-1}$. Obtaining $[\hat{a}_r]_1 = -[Q_{21}]^{-1} [Q_{22}] [\hat{a}_r]_2$ and subsequently $[\hat{b}_r]_1$ by substitution (applicable only if the r^{th} structural equation is exactly identifiable) is called the method of *Indirect Least Squares*.

However, if $[Q_{21}]$ is not a square matrix or it is deficient in rank $[Q_{21}]^{-1}$ would not exist. The restriction of $\mu_r \geq m_1$ together with the assumption that $[Q_{21}]$ is of a rank m_1 guarantees that $[Q_{21}]^{-g}$ (or the least squares generalized inverse of $[Q_{21}]$) exists (**Krishnamurthy & Sen**, pp. 153-185). That is to say that the least squares solution of $[\hat{a}_r]_1$ exists. In the worst case, when the rank of $[Q_{21}]$ is $< m_1$, only the proper Moore-Penrose inverse of $[Q_{21}]$ or $[Q_{21}]^+$ exists (**Krishnamurthy & Sen**, pp. 191-193). In that case $[\hat{a}_r]_1$ cannot be known or estimated uniquely.

We have seen that in case $[Q_{21}]$ is a square matrix, $[Q_{21}]^{-1}$ exists (provided that $[Q_{21}]$ has a full rank of m_1). Since $[Q_{21}]$ is a $k_2 \times m_1$ matrix, its being a square matrix implies that $k_2 = m_1$. Now k_2 means the number of elements in $[\hat{b}_r]_2$ all set to zero, which in turn implies the number of pre-determined variables appearing in the model $YA + XB + E = 0$, but absent from the r^{th} equation. Similarly, m_1 means the number of endogenous variables with unknown structural coefficients that appear in the r^{th} equation. We have seen that exactly one more endogenous variable (y_r) appears in the r^{th} equation, but its coefficient is minus unity (-1) due to which fact $a_{rr} = -1$. Therefore, it is said that the *necessary condition for exact identification* of the r^{th} equation is that the number of endogenous variables appearing in it is equal to the number of pre-determined variables absent from it plus one. The sufficient condition for exact identification is, of course, that $[Q_{21}]$ has a full rank of m_1 .

In case of an over-identification where $k_2 \geq m_1$ the number of endogenous variables appearing in the particular equation (the r^{th} one) must be less than the number of pre-determined variables absent from the model plus one. This is the necessary condition for over-identification. The sufficient condition is that $[Q_{21}]$ has a full rank of m_1 .

Therefore, the r^{th} equation would be under-identified if and only if either (or both) of the two conditions is (are) satisfied: (i) $k_2 < m_1$ (ii) rank of $[Q_{21}]$ is deficient or $\text{rank}([Q_{21}]) < m_1$. It is obvious that in the case where $k_2 < m_1$, $\text{rank}([Q_{21}]) \leq k_2 < m_1$. Therefore, $k_2 < m_1$ guarantees under-identification. However, the rank of $[Q_{21}]$ might be deficient ($\text{rank}([Q_{21}]) < m_1$) even if $k_2 \geq m_1$ provided that there are enough number of linear dependencies in the equation system $[Q_{21}] [\hat{a}_r]_1 + Q_{22}[\hat{a}_r]_2 = [\hat{b}_r]_2$.

We have seen that in case of over-identification $k_2 > m_1$, due to which the system of equations described by $[Q_{21}] [\hat{a}_r]_1 + Q_{22}[\hat{a}_r]_2 = [\hat{b}_r]_2$ has the number of equations larger than the number of unknowns to be determined. Consequently, $[Q_{21}]^{-1}$ is not defined. It is natural to think of obtaining $[Q_{21}]^g$ and $[\hat{a}_r]_1 = -[Q_{21}]^g[Q_{22}] [\hat{a}_r]_2$. However, Henri Theil and R L Basman appear not to have been attracted by this route to estimation of $[\hat{a}_r]_1$ and subsequently obtaining $[\hat{b}_r]_1$ in $[Q_{11}] [\hat{a}_r]_1 + [Q_{12}] [\hat{a}_r]_2 = [\hat{b}_r]_1$. Instead, they obtain expected Y (say, \hat{Y}) by the reduced form equations (that is $\hat{Y} = X P$). Then, in any particular (over-identified) equation, say the r^{th} equation, each y_s (with undetermined coefficients, $s \neq r$) is replaced by the corresponding \hat{y}_s such that $\hat{Y}a_r + Xb_r = 0$. Since y_r in the r^{th} equation appears with a known coefficient ($a_{rr} = -1$), y_r is not replaced by \hat{y}_r . Then estimation of a_r and b_r by OLS is permissible as the error in y_r (dependent endogenous variable) is no longer correlated with the errors in the explanatory variables (Y_s or X_r) and the Gauss-Markov conditions are satisfied. By OLS, therefore, the unknown coefficients in a_r and b_r appearing in $\hat{Y}a_r + Xb_r = 0$ are estimated. Thus, first OLS is used to obtain P and subsequently, OLS is used once again on $\hat{Y}a_r + Xb_r = 0$ to obtain the unknown coefficients in a_r and b_r . On account of applying OLS at two stages,

the method is called the *Two-Stage Least Squares* (2-SLS). From the procedure and the conditions governing its application it is clear that 2-SLS is an Instrumental variable approach to estimation of $Y_r a_r + X_r b_r = 0$, where each y_s (with un-determined coefficients, $s \neq r$) is replaced by the instrumental variable \hat{y}_s . It follows from this the 2-SLS estimator is (usually) biased but consistent.

It is natural to explore the possibility of obtaining $[\hat{a}_r]_1$ and $[\hat{b}_r]_1$ in an over-identified structural equation by using the least squares inverse of $[Q_{21}]$, that is $[Q_{21}]^{-g}$ as mentioned earlier. However, it took a long time to attract one's attention since **Basmann** (1957) and **Theil** (1961) developed 2-SLS. **Khazzoom** (1976) investigated into generalization of ILS (evidently ignored by Theil and Basmann) for an over-identified equation. Khazzoom estimates reduced form equations of a multi-equation linear econometric model by OLS but (in the second stage) instead of estimating the (modified) structural equations by OLS (or the Instrumental variable method) as done in the 2-SLS, he applies generalized inverse of the relevant submatrix of reduced form coefficients to obtain the structural coefficients. More explicitly, for the model $YA + XB + E = 0$ (the reduced form equations being $Y = X\Pi + U$, $\Pi = -BA^{-1}$ and $P = \hat{\Pi}$), in the relationship $Pa_j = -b_j$ for any (j^{th}) structural equation, we have

$$\begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = - \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \text{ where } a_1 \text{ and } b_1 \text{ are unknown structural coefficients,}$$

$a_2 = (0 \ 0 \ \dots \ 0 \ -1)'$ and $b_2 = (0 \ 0 \ \dots \ 0 \ 0)'$. From this we obtain $\hat{a}_1 = -P_{21}^{-g}(b_2 + P_{22}a_2)$ and $\hat{b}_1 = -(P_{11}\hat{a}_1 + P_{12}a_2)$. This is GILN_2 (alias GILS or Generalized Indirect Least Euclidean-Norm) estimator because ILS (Indirect Least Squares) estimator, which is applicable only

in case of an exactly identified structural equation, is defined as c and $\hat{b}_1 = -(P_{11}\hat{a}_1 + P_{12}a_2)$.

If P_{21} is a square matrix of full rank, $P_{21}^{-g} = P_{21}^{-1}$. Therefore, GILN₂ is applicable in case of any structural equation exactly or over identified. In Khazzoom estimator $P = (X'X)^{-1}X'Y$ is the matrix of Reduced Form coefficients estimated by OLS.

Estimation of Multi-equation Models by LAD: So far we have seen how the method of Least squares is applied to estimation of the structural coefficients in a multi-equation (linear) model. Now we turn to the application of LAD to estimation of the same. Our survey suggests that the L_1 norm (LAD) estimation entered the domain of multi-equation model with the paper published by **Glahe & Hunt** (1970). Since then works on searching a suitable, fast and convenient numerical method (algorithm) for L_1 norm estimator continued. Glahe & Hunt were perhaps the first to extend L_1 estimation to multi-equation models, compare the estimated parameters with those estimated through L_2 estimation and use of Monte Carlo Methods for their performance appraisal.

In their paper, a distribution sampling study comprising of four major experiments has been described. All the experiments have been based upon the exactly specified, over-identified simultaneous equation model

$$Y_1 + A_{12}Y_2 + B_{11}Z_1 + B_{12}Z_2 + B_{10} = E_1 \quad \dots (3.9)$$

$$Y_1 + A_{22}Y_2 + B_{23}Z_3 + B_{24}Z_4 + B_{20} = E_2 \quad \dots (3.10)$$

where Y_1 and Y_2 are jointly determined endogenous variables; Z_1, Z_2, Z_3 and Z_4 are exogenous variables; E_1 and E_2 are the random error terms which are assumed to be normally and independently distributed with a zero mean and standard deviation of ten (except in the experiment involving heteroskedasticity).

A single structure for the basic model presented above was used throughout. For the exogenous variables, economic time series data covering the period 1960-1964 were chosen. The values chosen were quarterly values for farm income (Z_1), farm equipment price index (Z_2), personal income (Z_3) and adjusted money supply (Z_4). Except for the experiment involving multi-collinearity, the data were randomly shuffled to purge the inherent multi-collinearity present in most economic time series data.

The structural equations were transformed to the reduced form equation to generate data. A random normal deviate generator was used to generate errors for sample sizes ten and twenty. With these data the values for the endogenous variables were calculated. Keeping the vectors of exogenous variables constant for each set of data, fifty sets of data were generated for each sample size.

In each experiment six estimators were tested. These estimators were direct least squares (DLS), direct least absolute (DLA), two-stage least squares (TSLS), two-stage least absolute (TSLA), least squares no restrictions (LSNR), and least absolute no restrictions (LANR). (Direct application of least squares or the method of least absolute to the reduced form yielded LSNR and LANR estimators). The first two pairs were used to compute moments of the distribution of each parameter estimate, for sample sizes ten and twenty, based upon the fifty replications. All three pairs were used to compute conditional predictions of each of the jointly determined variables.

Each of the four major experiments conducted was divided into sub-categories where small sample sizes of ten and twenty were tested. The first experiment was conducted using the classical simultaneous equation model. Normally and independently distributed error terms with mean = zero and standard deviation = 10, uncorrelated

exogenous variables and correct specification of the model were used. The second experiment considered a level of multi-collinearity among the explanatory variables. Heteroskedasticity was considered in the third experiment. The variance considered was a monotonic function increasing over time given by $\sigma_u^2 = (\sigma_o + i)^2$ where $\sigma_o = 5$ and $i = 0, 1, \dots, N-1$. In the fourth experiment misspecified model was investigated. The model was misspecified by including an additional exogenous variable and a parameter with a true value of zero in the estimation sequence. The endogenous variables were generated in the same manner. The computational method used in L_1 estimation was based on Usow L_1 Fit Algorithm, developed by **Usow**.

The study was concerned with two major objectives - the estimation of structural parameters and conditional prediction. Examining the means and standard deviations of the estimates of structural parameters some summary statistics were prepared. Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) were used for an evaluation of the performance of the estimators on the basis of smallest bias and smallest standard deviation. Rankings of the actual results by smallest RMSE and MAE were prepared and from those rankings summaries and summary statistics were calculated.

To test the consistency of the total rankings of the estimators, Kendall's coefficients of concordance, W , was used. The hypothesis that there was no difference between estimators (when paired) in the number of times one estimator produced smaller MAE's than another one in each experiment was tested using the Cochran Q test. The hypothesis was accepted at 0.05 level. The Wilcoxin matched-pairs signed-ranks test was used to compare the L_1 and L_2 estimators to determine whether or not one was significantly different from another one. To check for the normality of sample

distributions of the *studentized* ratios of structural-coefficient estimates, Kolmogoroff-Smirnov test as explained by Birnbaum was used. The ratio used has been given by $T_{\hat{\theta}^K} = (\hat{\theta}^K - \theta^*) / \hat{\sigma}_{\hat{\theta}^K}$, where θ^* is the hypothesized value of θ .

The result of the four experiments showed that the two direct methods were best overall estimators for making conditional predictions, whether MAE or RMSE criterion were used and was true for both sample sizes. When errors were normally distributed and no substantial multicollinearity was present, none of the reduced form estimators was “poor”. But in the presence of multicollinearity DLS fell off sharply in predictive ability. When other problems existed, LSNR or LANR proved to be more reliable since they were the methods with least variability of the six studied.

It was also observed that LANR performed as well as LSNR and both outperformed the solved reduced-form methods. The structural estimators, DLA and TSLA, did not outperform DLS and TSLS. They did succeed in doing as well as the least squares estimators in many respects. The authors, therefore, concluded that L_1 norm estimator should prove equal or superior to L_2 norm estimators for model using a structure similar to the one used in the study. They, however, held that with an increase in sample size the superiority of the L_1 norm estimator loses its edge over L_2 norm estimator.

Amemiya (1980) developed the two-stage least absolute deviation estimator, which is rather analogous to two-stage least squares by **Theil** (1961). **Amemiya** (1982) further extended the method to provide it a mathematical and statistical basis in the direction of consistency and related statistical properties. In this paper (of 1982) he defined a class of estimators called the two-stage least absolute deviation estimators

(2SLAD) and derived their asymptotic properties. The problem of finding the optimal member of the class was also considered.

Amemiya (1982) also pointed out that in structural equations and reduced form equations as given below:

$$YA + XB + E = ZC + E \text{ and}$$

$$Y = X\pi + V; \text{ where } Z = (Y, X) \text{ and } C = \begin{pmatrix} A \\ B \end{pmatrix}$$

how one defines the LAD (least absolute deviation) estimator analogue of 2SLS (two-stage least squares estimator) in the estimation of C ? Amemiya points out that the authors of all previous studies (before Amemiya wrote that article) on the subject defined LAD as the value of C that minimized

$$S_a = \sum |Y_1 - P_1' ZC|, \text{ where } P = X(X'X)^{-1}X'$$

It was rather natural to define LAD that way since then. They interpreted 2SLS so as to minimize $S_L = \sum (Y_1 - P_1' ZC)^2$. However, if one wanted to use an interpretation of 2SLS as the instrumental variable estimator minimizing $S_{1L} = \sum (P'Y - P'ZC)^2$, one would define 2SLAD analogously to minimize $S_{1A} = \sum |P'Y - P'ZC|$. Combining the above two ideas, 2SLAD can be defined as a class of estimators obtained by minimizing

$$S_{qA} = \sum |qY + (1-q)P'Y - P'ZC|$$

where q is the parameter to be determined by the researcher. The minimization of

$$S_{qL} = \sum \{qY + (1-q)P'Y - P'ZC\}^2$$

yields 2SLS for any value of q whereas minimization of its absolute analogue (S_{qA}) depends crucially on the value of q . If $q=0$, it yields the estimator which is asymptotically equivalent to 2SLS. Thus, in the asymptotic sense the class of 2SLAD estimator contains 2SLS as a special case. This finding by Amemiya has a very powerful generalizing effect on the estimators.

In the article, Amemiya proved the strong consistency and the asymptotic normality of the LAD estimator in the standard regression model. Though the asymptotic normality was proved by Bassett and Koenker prior to Amemiya, the method used by Amemiya is simple to understand and more easily generalizable to other models such as simultaneous equation models or non-linear regression models.

Given the standard regression model $Y = Xa + E$, where X is a nxk matrix of bounded constants such that $\lim_{n \rightarrow \infty} n^{-1}X'X$ is a finite positive-definite matrix and E is a n -vector of i.i.d random variables, the LAD estimator has been defined to be a value of \tilde{a} that minimizes $S = \sum_{i=1}^n |Y_i - X'_i \tilde{a}| - \sum_{i=1}^n |E_i|$, where X'_i is the i^{th} row of X . The second term of the right-hand side of the equation does not affect the minimization since it is independent of \tilde{a} . It was added to facilitate proof of consistency without assuming the existence of a finite first moment. The strong consistency of LAD was proved by showing that $n^{-1}S$ converges almost surely uniformly in \tilde{a} to a function which attains the minimum at a , the true value. Strong consistency of 2SLAD for any value of $q > 0$ followed from the strong consistency of LAD. Asymptotic normality of 2SLAD was proved only for the case where E and V are normally distributed.

In the 2SLAD estimation studied, it was assumed that the minimization of the sum of absolute deviation is applied only to a specific equation to be estimated and not to all the reduced form equations. In other words, LAD was applied only in the second stage of regression and not in the first. The author, however, opined that if V as well as E follows a non-normal distribution, it would be better to apply LAD to the reduced form equation as well as to the structural equation to be estimated.

Applying LAD to each of the reduced form equations, $Y = X\pi + V$, $\hat{\pi}$ was obtained and then minimizing $\sum_{i=1}^n |Y_i - X_i'\hat{\pi} - X_{ii}'\tilde{a}|$, the double two-stage least absolute deviation estimator (D2SLAD) was developed. It was shown that even under the fully non-normal case D2SLAD is far inferior to 2SLAD for q between 0.2 and 0.5 for realistic values of the parameters. This result showed that in applying the LAD estimation to 2SLS, it is much more important to use LAD in the second stage than in the first stage.

Since $\hat{\pi}$ is a strongly consistent estimator of π , the strong consistency of D2SLAD followed easily from the strong consistency of LAD. Asymptotic normality of D2SLAD, however, has not been proved.

Asymptotic variance of 2SLAD and D2SLAD in a partially non-normal case (where E follows a mixture of normal distributions and V is normal) and fully non-normal case (where both E and V follow a mixture of non-normal distributions) were obtained. It was observed that when all the error terms follow a mixture of normal distributions, 2SLAD with a small value of q somewhere between 0 and 0.5 is recommended and it does not pay to use the more complicated D2SLAD.

Amemiya suggested Monte Carlo experiments to be carried out in order to study the properties of 2SLAD estimator (by minimization of S_{qA}) which may be compared with the properties of the estimator obtained by minimization of S_{qL} and S_{qA} for $q=0$. It appears that no study has yet been carried out to implement the suggestions made by Amemiya.

We have already described the work of **Khazzoom** (1976) who generalized Indirect Least Squares estimator for (exactly or over-) identified equations. From this, it may follow that if LAD performs better than OLS in estimating the matrix of reduced form coefficients, application of generalized inverse on such matrix (of reduced form coefficients) would be better than the GILS suggested by Khazzoom. A more generalized name – ***Generalized Indirect Least Norm (GILN)*** - may be given to the family of such methods for the minimand norm may be Euclidean (as suggested by Khazzoom) or absolute. It appears that this possibility is hitherto unexplored.

Fair (1994) estimated the US model by 2SLS, 2SLAD, 3SLS and FIML. Median unbiased (MU) estimates were also obtained for eighteen lagged dependent variable coefficients. The 2SLS asymptotic distribution was compared to the exact distribution and was found to be close. A comparative study of four sets of estimates, that is, 2SLS, 2SLAD, 3SLS and FIML was made. The results obtained showed that the estimates are fairly close to each other with the FIML being the farthest apart. The 3SLS estimator was found to be more efficient than the 2SLS estimator. The 2SLS standard errors were on an average 28 percent larger than the 3SLS standard errors. And the 3SLS standard errors were on average smaller (19 percent) than the FIML standard errors. To compare the different sets of coefficient estimates, the sensitivity of the predictive accuracy of the model to the different sets was also examined. The RMSEs were found to be very similar

across all the five sets of estimates. No one set of estimates dominated the other and in general the differences were found to be quite small. The author also compared the US model to the VAR5/2, VAR4 and AC models. The US model was found to do well in the tests relative to the VAR and AC models.

Kim and Muller (2000) presented the asymptotic properties of two-stage quantile regression estimators. In their paper, they derived the asymptotic representation of the estimators and proved the asymptotic normality with quantile regression predictions. The asymptotic variance matrix and asymptotic bias were discussed. They also analysed the asymptotic normality and the asymptotic covariance matrix with LS predictions. The results obtained permitted valid inferences in structural models estimated by using quantile regressions, in which the possible endogeneity of some explanatory variables was treated via ancillary predictive equations. Simulation results illustrated the usefulness of this approach.

III. Details of different Computational Algorithms for LAD Estimation: Now we provide the details of the numerical algorithms mentioned in the preceding section of this chapter. The Numerical algorithms for the purpose may broadly be classified into three heads: (i) those based on LP formulation, (ii) those based on iterative schemes - using the fixed point theorem which says that a continuous and bounded function onto itself has at least one point on which $x = f(x)$ and (iii) those based on the multi-variate search methods of non-linear programming.

To begin, let us have a closer look into the problem of minimum absolute deviation estimation of regression models: Let the linear regression equation to be estimated be

$$y = Xa + e \quad \dots (3.11)$$

The Least Absolute Deviation (LAD) estimator is given by that value of \tilde{a} , which minimizes

$$S = \sum_{i=1}^n |e_i| = \sum_{i=1}^n \left| y_i - \sum_{j=0}^m \tilde{a}_j x_{ij} \right| \quad \dots (3.12)$$

This function is defined in a similar way to the least squares estimator. But, unlike the least squares estimator that minimizes the sum of the squared residuals, the LAD estimator minimizes the sum of their absolute values of the residuals. Consequently, relative to the least squares estimator, it is less influenced by outliers and in repeated sampling from a distribution where outliers are prevalent, its sampling variability is likely to be less than that of the LS estimator.

Since equation (3.12) represents an absolute value function, it is not differentiable at every point. Hence, differentiation, which is conventionally used for minimization in the least squares method, cannot be applied here. In fact, the minimum point of S is a vertex and hence non-differentiable. The problem, therefore, has to be solved by devices other than calculus.

To probe into this aspect further, let us take the simple linear regression model with one independent variable and an intercept, namely,

$$Y_i = a_0 + a_1 X_i + e_i, \quad i=1, 2, \dots, n \quad \dots (3.13)$$

Now, we need to find \hat{a}_0 and \hat{a}_1 so that

$$S = \sum |Y_i - \hat{Y}_i| \quad \dots (3.14)$$

is minimized. Here, $\hat{Y}_i = \hat{a}_0 + \hat{a}_1 X_i$.

Suppose we choose \hat{a}_0 such that the regression line includes the point of means.

This implies that the origin is translated to the point (\bar{Y}, \bar{X}) . Then by changing the co-ordinates

$$y_i = Y_i - \bar{Y}$$

and $x_i = X_i - \bar{X}$

equations (3.13) and (3.14) can be written as

$$y_i = a_1 x_i + e_i \quad \dots (3.15)$$

(since $\hat{a} = 0$ in the new co-ordinate system)

and
$$S = \sum |y_i - \hat{y}_i| = \sum |y_i - \hat{a}_1 x_i| \quad \dots (3.16)$$

An element of S is represented as

$$S_i = |y_i - \hat{a}_1 x_i| \quad \dots (3.17)$$

This is a broken line ($Y = |X|$) in the (s, a_1) plane and is composed of two half- lines with a minimum equal to zero at

$$\hat{a}_i = \frac{y_i}{x_i} \quad (\text{obtained by equating } S_i \text{ to zero}). \quad \dots (3.18)$$

It has been found that S_i is always convex upwards since the slope of the half-line to the left of \hat{a}_i is equal to $-|x_i|$ and to the right of \hat{a}_i is equal to $|x_i|$. As a result, $S = \sum S_i$ considered in the (s, a_1) plane, being the sum of connected half- lines will consist of connected line segments. The slope of S at any point a_1 will be the aggregate of the slopes of S_i at that value of a_1 . Now, since the slopes change only at the minimum

points of S_i , the slope of S also can change only at these points. S is also convex since each S_i is convex.

It can be inferred from the above that only points corresponding to the minimum points of the individual S_i are required to be considered in minimizing S which are finite in number and have value equal to y_i/x_i . Hence, the regression line will pass through the point of means and also the observation corresponding to the minimizing i . Therefore, determination of the regression line is by the point of means and the observation associated with the minimizing \hat{a}_i .

Let us now consider the case where the regression line does not pass through the point of means. Here the estimation problem is to find \hat{a}_0 and \hat{a}_1 such that

$$S = \sum |Y_i - \hat{a}_0 - \hat{a}_1 X_i| \quad \dots (3.19)$$

is minimized. A typical element of S may be represented as

$$S_i = \sum |Y_i - \hat{a}_0 - \hat{a}_1 X_i| \quad \dots (3.20)$$

It is composed of two half-planes in (S, a_0, a_1) space that intersect in the (a_0, a_1) plane (in the three dimension case).

Thus, a family of (a_0, a_1) corresponding to the minimum of S_i is found to exist and is given by the equation

$$Y_i - a_0 - a_1 X_i = 0 \quad \dots (3.21)$$

S being a sum of half-planes, its surface is composed of connected plane segments. It is convex upwards because the S_i are convex. S , therefore, will have a minimum unique region, which will be a point if the S_i which intersects at the (a_0, a_1) plane are not parallel to one another, or a line if the family (a_0, a_1) obtained on minimizing the S_i are the same

for each S_i or a closed polygon if different families of (a_0, a_1) are obtained for different S_i . A point represents a unique solution to the estimation problem, whereas a line or closed polygon represents multiple solutions.

Since the direction cosines of the constituent half-planes of the S_i change only at their lines of intersection, the boundaries of the plane segments that compose S will be directly above the set of lines

$$Y_i - a_0 - a_1 X_i = 0, \quad i=1,2,\dots,n \quad \dots (3.22)$$

in the (a_0, a_1) plane. A unique solution to the problem must be above one of the points of intersection of the lines in equation (3.22). On the other hand, a non-unique solution will lie either on a closed interval of a line in equation (3.22) constituted by the intersection of that line with two other parallel lines, or on a closed polygon formed by three or more of the lines.

The three dimensional case discussed so far can be extended to n independent variables. Here each S_i is composed of two n -dimensional half-hyperplanes in the $(S, \tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_n)$ space intersecting in the $(\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_n)$ hyperplane and is convex upward. Since $S = \sum S_i$, therefore, S will form a polygonal surface whose edges will lie above the $(n-1)$ dimensional hyperplanes which are the result of the intersections of the half-hyperplanes of S with the $(\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_n)$ hyperplane. S is convex since the S_i are convex. When the solution to the estimation problem is unique, the minimum of S_i will lie above the intersection of n of these $(n-1)$ dimensional hyperplanes. These n observations determine the regression hyperplane.

In the cases with larger dimensions it may be impossible to find a solution to the estimating problem due to the manifold increase in the complexity of the calculations.

Therefore, in order to find an optimal solution, the problem can be converted to a linear programming problem and a solution sought.

LAD Estimation as a Linear Programming Problem: Linear Programming (developed in the late 1940's) immediately attracted the attention of econometricians seeking optimization of a linear function with corner solution at a vertex. Therefore, LAD estimation was originally converted to an equivalent linear programming problem.

In matrix form, the linear programming problem is to minimize

$$A = P'Z \quad \dots (3.23)$$

subject to the constraints

$$BZ \geq a_1 \quad \dots (3.24)$$

$$Z \geq 0 \quad \dots (3.25)$$

The function that is being minimized in equation (3.12) is first converted into an equivalent linear form by writing e_i as the difference between two non-negative variables.

$$e_i = v_i - w_i, \quad v_i, w_i \geq 0 \quad \dots (3.26)$$

and then minimizing the quantity

$$S^* = \sum v_i + \sum w_i \quad \dots (3.27)$$

Although S^* of equation (3.27) and S of equation (3.12) are different functionals, the values of v and w that minimize S^* give the value of e that minimizes S . This implies that equation (3.27) is equivalent to equation (3.12). In the optimal solution also S^* and S are equal.

The constraints in equation (3.24), which are given by n observations, can be re-written in the form of equation (3.28)

$$\begin{bmatrix}
x_{11} & -x_{11} & x_{12} & -x_{12} & \cdots & x_{1K} & -x_{1K} & 1 & -1 & 0 & 0 & \cdots & 0 & 0 \\
x_{21} & -x_{21} & x_{22} & -x_{22} & \cdots & x_{2K} & -x_{2K} & 0 & 0 & 1 & -1 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
x_{n1} & -x_{n1} & x_{n2} & -x_{n2} & \cdots & x_{nK} & -x_{nK} & 0 & 0 & 0 & 0 & \cdots & 1 & -1
\end{bmatrix} \times \begin{bmatrix}
\tilde{a}_1 \\
\tilde{a}_1^* \\
\tilde{a}_2 \\
\tilde{a}_2^* \\
\vdots \\
\tilde{a}_K \\
\tilde{a}_K^* \\
v_1 \\
w_1 \\
v_2 \\
w_2 \\
\vdots \\
v_n \\
w_n
\end{bmatrix}$$

$$= \begin{bmatrix}
y_1 \\
y_2 \\
\vdots \\
y_n
\end{bmatrix} \quad \dots(3.28)$$

Here e has been replaced by $v - w$ and the variable x appears twice, once positively and once negatively. $\tilde{a}_1^*, \tilde{a}_2^*, \dots, \tilde{a}_K^*$ are the coefficients of the K new x variables. The independent variables x and $-x$ correspond to the matrix B in equation (3.24), the regression coefficients $\tilde{a}_1, \tilde{a}_1^*, \tilde{a}_2, \tilde{a}_2^*, \dots, \tilde{a}_K, \tilde{a}_K^*$ (which are non-negative) to the vector Z and y to the vector of constraints a_1 . The residual term $v_1, w_1, v_2, w_2, \dots, v_n, w_n$ represent the introduction of $2n$ “slack” activities. The restriction in equation (3.25) corresponds to the vector of non-slack activities level.

In the problem there are altogether $2(K+n)$ activities. Succeeding pairs of activities, namely, $(\tilde{a}_j, \tilde{a}_j^*)$ and (v_i, w_i) of the matrix Z in equation (3.28) are dependent

and therefore in any solution, only one member of each pair for each i and j can be non-zero. This is necessary that the activities in any basis must be independent. Because only one of the regression coefficient can be non-zero, the negative of each independent variable has been included to allow the coefficients of the independent variable to be of either sign. The values taken by S^* in equation (3.27) and S in equation (3.12) will be equal since both v_i and w_i cannot be non-zero. Since the activities corresponding to the independent variables have zero “cost” in the minimand, the independent variables will always be present in the optimal solution. This implies that the optimal solution will contain $n-K$ residuals (i.e., slack activities) and K residuals will therefore be zero. This corresponds to what was earlier observed that the regression hyperplane is determined by a subset of n observations.

The above formulation of the problem will always yield a solution if degeneracy is not considered. However, if the number of observations is large, the computational procedure will be cumbersome and lengthy. Wagner has shown that by considering the dual of the above formulation and solving it as a problem in bounded variables using special simplex procedures developed by Charnes and Lemke (1954), Dantzig (1955) and Wagner (1958), the number of activities and constraints can be substantially reduced.

Although linear programming is a computational method that can be used in L_1 estimation, the method is known to be computationally unwieldy. There are iterative methods as well as multivariate search methods that can be used to minimize S and hence determine that value of \tilde{a} when S is minimum.

The Iterative Algorithm: Schlossmacher (1973) and Fair (1974) suggested an iterative method in the case of a linear model, $Y = Xa + e$. In this method, the steps for computing are as follows:

Step I: Obtain the LS estimates of \hat{a} , \hat{Y} and \hat{e} using the formulae

$$\hat{a} = (X'X)^{-1} X'Y; \quad \hat{Y} = X\hat{a}; \quad \hat{e} = Y - \hat{Y}$$

Step II: Compute $\hat{W}_{ij} = \frac{1}{|\hat{e}_i|}$ for $i=j$, else $\hat{W}_{ij} = 0$ $i, j = 1, 2, \dots, n$.

Step III: Compute $\hat{a} = (X'\hat{W}X)^{-1} X'\hat{W}Y$; $\hat{Y} = X\hat{a}$; $\hat{e} = Y - \hat{Y}$

Step IV: If the values of \hat{a} are stable (convergence has been reached to the pre-assigned accuracy) then stop, otherwise go to step II.

While using this algorithm, two points should be taken care of. First, that X includes the unitary vector often designed for taking care of the constant term in the model and second, that in case $|e_i|$ (for any i) is very small (smaller than say, 0.00001) than that e_i is set equal to a small value (say, 0.00001 as done by Fair or completely ignored i.e. set equal to zero as done by Schlossmacher). The asymptotic variance-covariance matrix of \hat{a} is given (Taylor, 1974) by

$$\hat{v} = 0.25 \left\{ \hat{f}(0) \right\}^{-2} (X'X)^{-1}, \quad \text{where,} \quad \hat{f}(0) = (p - q - 1) / \{n(\hat{e}_p - \hat{e}_q)\}$$

here p and q are integers such that $p \geq \frac{n}{2} + 1 = q$ for even n or $p = \frac{n}{2} + 0.5 = q$ for odd n .

Further that $e_1 \leq e_2 \leq e_s \leq \dots \leq e_n$ are ordered L_1 residuals. Though the best values of p and q cannot be ascertained, it has been suggested that $3n/4$ and $n/4$ are the most appropriate values of p and q respectively as these values are not much affected by extreme values.

Huber (1973) put forward an estimator that minimizes appropriately weighted squared deviations for small residuals, and absolute deviations for large residuals.

The estimator minimizes

$$\sum_{i=1}^n f(y_i - x_i' a)$$

where, $f(e_i) = \frac{1}{2} e_i^2$ for $|e_i| < \delta$

$$= \delta |e_i| - \frac{1}{2} \delta^2 \quad \text{for } |e_i| \geq \delta, \text{ and } \delta \text{ is a pre-assigned constant.}$$

A set of normal equations that can be solved iteratively starting with either LS or LAD estimator for a has been suggested by Huber. However, for any given observation the appropriate functions can change from iteration to iteration.

There is yet another method suggested by **Anscombe** (1967). The method minimizes squared deviations for small errors, absolute deviations for moderate errors and rejects observations with large errors. The estimator is obtained by minimizing the weighted least squares function.

$$\sum_{i=1}^n w_i (y_i - x_i' a)^2 \quad \dots (3.29)$$

where, $w_i = 1$ if $|\hat{e}_i| \leq m_1$; or $w_i = m_1/|e_i|$ if $m_1 < |\hat{e}_i| \leq m_2$; or $w_i = 0$ if $|\hat{e}_i| > m_2$

m_1 and m_2 are either pre-assigned multiples of the standard deviation of e_i or pre-assigned constants. The steps to be followed for minimizing Equation (3.29) iteratively are:

Step I: Calculate either the LS or LAD estimates for a say \tilde{a}

Step II: First calculate the errors $\hat{e}_i = y_i - x_i' \tilde{a}$ and then the corresponding weights.

Step III: Using weighted least squares $\hat{a} = (X'WX)^{-1} X'Wy$

where $W = \text{diagonal } (W_1, W_2, \dots, W_n)$, estimate a .

Step IV: Repeat the process until \hat{a} converges.

LAD Estimation as Multivariate Search Problem: Multivariable search problems start from some base point and move towards the optimum based on sequential improvement in the value of the objective function. The step size may be fixed or accelerated and decelerated subject to a set of rules. There are several procedures based on this technique.

(i) Fletcher-Powell Algorithm: The method requires several alternatives starting points as multi-modal functions are suspected. To minimize: $F(a_1, a_2, \dots, a_m)$ the steps are as follows:

Step I: Select a starting point.

Step II: Determine the search direction. In normalized form this can be written as

$$M_i^{(k)} = \left\{ \frac{-\sum_{j=1}^N H_{ij} \left(\frac{\delta F}{\delta a_j} \right)}{\left[\sum_{l=1}^N \left(\sum_{j=1}^N H_{lj} \left(\frac{\delta F}{\delta a_j} \right) \right)^2 \right]^{\frac{1}{2}}} \right\}^k, \text{ where, } i = 1, 2, \dots, m$$

here, k = iteration index (starting point $k=0$),

M_i = direction vector components,.

$\delta F / \delta a_j$ = slope vector components,

H_{ij} = elements of a symmetric positive definite matrix (mxm), (initially an identity matrix).

The initial direction therefore, is the path of steepest descent.

Step III: Conduct a one-dimensional search in the direction chosen in step II in the following manner until a minimum is located utilizing the relation $a_{i(new)} = a_{i(old)} + SM_i; i = 1, 2, \dots, m$ where S is the step size in the direction of search.

(1) evaluate the objective function first.

(2) then, incrementing a distance Δa to the independent variable, evaluate the objective function again. If a function improvement is obtained then double the step size for the next function evaluation. But, if a function improvement is not obtained in the first step, reverse the direction by locating the next point at a distance $-\Delta a$ from the starting point.

(3) after the first step, double the step size if a function improvement is obtained and halve it if a worse function evaluation is obtained.

(4) when a local optimum is encountered, the procedure will yield three points (a_k, a_{k-1}, a_{k-2}) straddling the optimum. Locate an additional point a_{k+1} such that $a_{k+1} = a_{k-1} + \Delta a/2$ which is the minimum in the direction of search, where Δa is the current step size. Then retain the best three points (call them a_1, a_2, a_3).

(5) then a quadratic equation, ϕ , is curve-fitted to the three retained points.

The optimum location, a^* , located by setting $\delta\phi/\delta a = 0$, is

$$a^* = \frac{1}{2} \left[\frac{(a_2^2 - a_3^2)F(a_1) + (a_3^2 - a_1^2)F(a_2) + (a_1^2 - a_2^2)F(a_3)}{(a_2 - a_3)F(a_1) + (a_3 - a_1)F(a_2) + (a_1 - a_2)F(a_3)} \right]$$

(6) then compare the objective function at a^* with the best previous point subject to a convergence limit,

$$\left| a^* - a_i (best) \right| \leq \lim$$

If the above criterion is satisfied, the procedure stops. If not, worst point is replaced by a^* and a new quadratic surface is fitted and local optimum is obtained.

Step IV: Make a convergence check. If convergence is achieved, terminate the procedure. If convergence is not achieved, choose a new search direction as per step II. Calculate $H^{(k+1)}$ as follows:

$$H^{(k+1)} = H^{(k)} + A^{(k)} - B^{(k)}$$

where,

$$A^{(k)} = \frac{\Delta a^{(k)} \left(\Delta a^{(k)} \right)^t}{\left(\Delta a^{(k)} \right)^t \left(\Delta G^{(k)} \right)}$$

$$B^{(k)} = \frac{H^{(k)} \Delta G^{(k)} \left(\Delta G^{(k)} \right)^t H^{(k)}}{\left(\Delta G^{(k)} \right)^t H^{(k)} \Delta G^{(k)}}$$

$$\Delta a^{(k)} = a^{(k+1)} - a^{(k)} \quad (\text{difference in location between iterations})$$

$$\Delta G^{(k)} = \frac{\partial F^{(k+1)}}{\partial a} - \frac{\partial F^{(k)}}{\partial a} \quad (\text{difference in slopes between iterations})$$

Perform a new one dimensional search in the new direction. Repeat the process until convergence is obtained.

Although the above algorithm can effectively locate the minimum, the fact that this method uses derivatives, would possibly be its drawback. Since LAD is essentially an absolute value function, it is not differentiable exactly at its minimum point. The minimum of an absolute value function is located in a vertex of the simplex.

Therefore, derivative-free search methods of finding the minimal value of S and subsequently the vector \tilde{a} are recommended. Such derivative-free search methods have been developed by R. Hooke, TA Jeeves, HH Rosenbrock and others. Two such algorithms which can effectively be used to minimize the absolute value function and are also relatively easy to program on a computer are discussed below.

(ii). Nelder-Meade Algorithm: JA Nelder and R Meade developed an algorithm using a regular geometric figure called a simplex, consisting of $m+1$ vertices, to find the minimum of a multivariable unconstrained non-linear function (**Kuester & Mize**, 1973). This simplex method utilizes reflected, expanded and contracted points to locate the minimum.

To minimize the function: $F(a_1, a_2, \dots, a_m)$, the steps are as follows:

Step I: First, pick a starting point, say \hat{a}_1

Step II: Next, construct a simplex that consists of the starting point and the following additional points as:

$$\hat{a}_j = \hat{a}_1 + \xi_j, \quad j = 2, 3, 4, \dots, m+1$$

where, ξ_j is determined from the following table:

j	$\xi_{1,j}$	$\xi_{2,j}$	\dots	$\xi_{m-1,j}$	$\xi_{m,j}$
2	p	q	\dots	q	q
3	q	p	\dots	q	q
\vdots	\vdots	\vdots		\vdots	\vdots
\vdots	\vdots	\vdots		\vdots	\vdots
m	q	q	\dots	p	q
$m+1$	q	q	\dots	q	p

where, $p = \frac{\sigma}{m\sqrt{2}} \left((m+1)^{\frac{1}{2}} + m - 1 \right)$; $q = \frac{\sigma}{N\sqrt{2}} \left((m+1)^{\frac{1}{2}} - 1 \right)$; m=total number of variables and σ = side length of simplex.

Step III: Calculation of the Centroid.

Once the simplex is formed, the objective function is evaluated at each point. The worst point (highest value of objective function) is replaced by a new point. Three operations are used – reflection, contraction and expansion.

A reflection point is first located thus:

$$a_{i,j} \text{ (reflected)} = \bar{a}_{i,c} + \alpha (\bar{a}_{i,c} - a_{i,j}(\text{worst})) ; i = 1, 2, \dots, m$$

where, α is a positive constant, $\bar{a}_{i,c}$ are the centroid co-ordinates of all points excluding the worst point. It is calculated as

$$\bar{a}_{i,c} = \frac{1}{K-1} \left[\sum_{j=1}^K a_{i,j} - a_{i,j}(\text{worst}) \right], \quad i = 1, 2, \dots, m$$

Set $K = m+1$.

Step IV: Check Reflected Point:

If the reflected point has the worst objective function value of the current points, a contracted point is located as follows:

$$a_{i,j} \text{ (contracted)} = \bar{a}_{i,c} - \beta (\bar{a}_{i,c} - a_{i,j}(\text{worst})), \quad i = 1, 2, \dots, m \text{ and } \beta \text{ lies between 0 and 1.}$$

If the reflected point, though better than the worst point, is still not the best point, a contracted point is calculated from the reflected point as follows.

$$a_{i,j} \text{ (contracted)} = \bar{a}_{i,c} - \beta (\bar{a}_{i,c} - a_{i,j}(\text{reflected})), \quad i = 1, 2, \dots, m.$$

The objective function is now evaluated at the contracted point. If an improvement over the current points is achieved, the process is restarted. If an improvement is not achieved, the points are moved one-half the distance towards the best points:

$$a_{i,j} \text{ (new)} = (a_{i,j} \text{ (best)} + a_{i,j} \text{ (old)}) / 2, \quad i = 1, 2, \dots, m.$$

The process is then restarted.

Step V: Calculate Expansion Point: If the reflected point calculated in step III is the best point, an expansion point is calculated as follows:

$$a_{i,j} \text{ (expansion)} = \bar{a}_{i,c} + \gamma(a_{i,j} \text{ (reflected)} - \bar{a}_{i,c}), \quad i = 1, 2, \dots, m$$

where, γ is a positive constant. If the expansion point is an improvement over the reflected point, the expansion point replaces the reflected point and the process restarted. If the expansion point is not an improvement over the reflected point, the reflected point is retained and the process restarted.

Step VI: The procedure is terminated when the convergence criterion is satisfied or a specified number of iterations has been exceeded.

(iii) Hooke-Jeeves Algorithm: The algorithm developed by R Hooke and TA Jeeves is based on the direct search method (**Kuester & Mize**, 1973). The method assumes a unimodal function and is derivative free. If more than one minimum exists or the shape of the surface of the simplex is unknown, several sets of starting points are required. The steps involved are:

Step I: Select a base point and evaluate the objective function.

Step II: Make local searches in each direction by stepping a_i a distance S_i to each side and evaluate the objective function to ascertain whether a lower function value can be obtained.

Step III: If there is no improvement i.e., the function does not decrease, reduce the step size and make searches from the previous best point.

Step IV: If the value of the objective function has decreased, locate a “ temporary head”

$a_{i,0}^{(k+1)}$ using the two previous base points $a_i^{(k+1)}$ and $a_i^{(k)}$ such that

$$a_{i,0}^{(k+1)} = a_i^{(k+1)} + \beta \left(a_i^{(k+1)} - a_i^{(k)} \right)$$

where, the variable index i varies from 1 to m; 0 denotes the temporary head; k is stage index (a stage is the end of m searches) and $\beta \geq 1$.

Step V: If the temporary head gives a lower function value, make a new local search for the temporary head and locate a new head. Check the value of the function F. Continue this method of expansion as long as the function value keeps decreasing.

Step VI: If the temporary head does not result in a lower function value, make a search from the previous best point.

Step VII: Continue the procedure until the convergence criterion is satisfied. Once convergence is achieved, the procedure is terminated.

Random Walk Method: This method is based on generating a sequence of improved approximations to the minimum, each derived from the preceding approximation (**Rao**, 1978, pp. 252-254). Thus if a_i is the approximation to the minimum obtained in the (i-1)th stage (or step or iteration), the new or improved approximation in the ith stage is found from the relation

$$a_{i+1} = a_i + \lambda u_i$$

where λ is a prescribed scalar step length, and u_i is a unit random vector generated in the ith stage. The detailed procedure of this method is given by the following steps.

1. Start with an initial point a_i and a scalar step length that is sufficiently large in relation to the final accuracy desired. Find the functional value $F_1 = F(a_1)$.
2. Set the iteration number $i = 1$.
3. Generate a set of n random numbers and formulate the unit random vector u_i .
4. Find the new value of the objective function as $F = F(a_1 + \lambda u)$.
5. Compare the values of F and F_1 . If $F < F_1$, set $a_i = a_1 + \lambda u$, and $F_1 = F$, and repeat step 3 through 5. If $F \geq F_1$, just repeat step 3 through 5.
6. If a sufficiently large number of iterations (N) cannot produce a better point, a_{i+1} , reduce the scalar length λ and go to step 3.
7. If an improved point could not be generated even after reducing the value of λ below a small number ε , take the current point a_i as the desired optimum point, and stop the procedure.

Random Walk Method with Direction Exploitation: In the random walk method described above, we proceed to generate a new unit random vector u_{i+1} as soon as we find that u_i is successful in reducing the function value for a fixed step length λ . However, we may expect to achieve a further decrease in the function value by taking a longer step length along the direction u_i . Thus the random walk method can be improved if each successful direction is exploited until it fails to be useful (Rao, 1978, pp. 255-256). This can be achieved by using any of the one-dimensional minimization method. According to this procedure, the new point a_{i+1} is found as

$$a_{i+1} = a_i + \lambda_i^* u_i$$

where λ_i^* is the optimal step length found along the direction u_i so that

$$F_{i+1} = F(a_i + \lambda_i^* u_i) = \min_{\lambda_i} F(a_i + \lambda_i u_i).$$

Random walk methods can work even if the objective function F is discontinuous or non-differentiable at some of the points. This property makes this method quite suitable for minimizing the absolute norm of residuals.

DasGupta, Madhuchhanda (2004) LEAST ABSOLUTE DEVIATION ESTIMATION OF MULT-EQUATION LINEAR ECONOMETRIC MODELS : A STUDY BASED ON MONTE CARLO EXPERIMENTS, Doctoral Dissertation, Dept. of Economics, North Eastern Hill University, Shillong (India); Supervisor – Prof. SK Mishra

CHAPTER 4

THE METHODOLOGY OF MONTE CARLO EXPERIMENTS

I. Introduction: There are two approaches to studying the properties of a mathematical system (representing a real-world system): the first, *analytical* and the second, *experimental*. In the analytical approach, the mathematical system is described in great details (so as to represent the real-world system as accurately as possible). All the axioms and assumptions underlying the system are exhaustively and explicitly described with utmost accuracy. All definitions relevant to the system are accurately and explicitly stated. All the operations that the mathematical system would be subjected to are accurately and exhaustively described in their details. Then, appropriate mathematical methods are applied to draw conclusions from the axioms, assumptions and statements that describe the said mathematical system. All the conclusions drawn in this process are the implications of the mathematical system so described. Nevertheless, mathematical methods applied to draw conclusions have some impact on the conclusions. A careful attempt is made, therefore, to apply mathematical methods such that they modify the conclusions as little as possible. These conclusions are deductive in nature and have general applicability.

However, complicated systems cannot be described in very accurate and exhaustive details. Even if the system is described in details, appropriate methods to draw conclusions may be either unavailable or intractable due mainly to the complexity of the system. To investigate into the properties of such a mathematical system by analytical methods is very difficult, if not impossible. Experimental methods are applied to investigate into the properties of such systems. The conclusions drawn from experimental

methods provide a great insight into the nature of the system. However, these conclusions have lesser degree of certainty as well as applicability. These conclusions are inductive inferences – generalizations based on the study of samples – and, therefore, probabilistic in nature.

II. The Monte Carlo Method: The Monte Carlo method of investigating into the properties of a mathematical system is basically experimental and numerical in nature. Numerical methods that are known as ‘Monte Carlo methods’ can be loosely described as statistical simulation methods, where statistical simulation is defined in quite general terms to be any method that utilizes sequences of random numbers to perform the simulation. Monte Carlo methods have been used for centuries, but only in the past several decades has the technique gained the status of a full-fledged numerical method capable of addressing the most complex applications. The credit for inventing the Monte Carlo method often goes to Stanislaw Ulam, a Polish born mathematician who worked for John von Neumann on the United States’ Manhattan Project during World War II. Ulam is primarily known for designing the hydrogen bomb with Edward Teller in 1951. Quoted in **Eckhardt** (1987), Ulam describes the incident as:

The first thoughts and attempts I made to practice [the Monte Carlo Method] were suggested by a question which occurred to me in 1946 as I was convalescing from an illness and playing solitaires. The question was what are the chances that a Canfield solitaire laid out with 52 cards will come out successfully? After spending a lot of time trying to estimate them by pure combinatorial calculations, I wondered whether a more practical method than “abstract thinking” might not be to lay it out say one hundred times and simply observe and count the number of successful plays. This was already possible to envisage with the beginning of the new era of fast computers, and I immediately thought of problems of neutron diffusion and other questions of mathematical physics, and more generally how to change processes described by certain differential equations into an equivalent form interpretable as a

succession of random operations. Later ... [in 1946, I] described the idea to John von Neumann, and we began to plan actual calculations.

Working with John von Neumann and Nicholas Metropolis, he developed algorithms for computer implementations, as well as exploring means of transforming non-random problems into random forms that would facilitate their solution via statistical sampling. This work transformed statistical sampling from a mathematical curiosity to a formal methodology applicable to a wide variety of problems. It was Metropolis who named the new methodology after the casinos of Monte Carlo. Ulam and Metropolis published their paper on Monte Carlo method in 1949.

Monte Carlo method is now used routinely in many diverse fields, from the simulation of complex physical phenomena such as radiation transport in the earth's atmosphere and the simulation of the esoteric sub-nuclear processes in high energy physics experiments, to the mundane, such as the simulation of simple games. The analogy of Monte Carlo methods to games of chance is a good one, but the *game* is a physical system, and the outcome of the game is not a pot of money or stack of chips (unless simulated) but rather a solution to some problem. The *winner* is the scientist, who judges the value of his results on their intrinsic worth, rather than the extrinsic worth of his holdings.

Statistical simulation methods may be contrasted to conventional numerical discretization methods, which typically are applied to ordinary or partial differential equations that describe some underlying physical or mathematical system. In many applications of Monte Carlo, the physical process is simulated directly, and there is no need to even write down the differential equations that describe the behavior of the system. The only requirement is that the physical (or mathematical) system be described

by probability density functions (pdf's). Assuming that the evolution of the physical system can be described by probability density functions, then the Monte Carlo simulation can proceed by sampling from these pdf's, which necessitates a fast and effective way to generate random numbers uniformly distributed on the interval $[0,1]$. The outcomes of these random samplings, or trials, must be accumulated or tallied in an appropriate manner to produce the desired result, but the essential characteristic of Monte Carlo is the use of random sampling techniques (and perhaps other algebra to manipulate the outcomes) to arrive at a solution of the physical problem. In contrast, a conventional numerical solution approach would start with the mathematical model of the physical system, discretizing the differential equations and then solving a set of algebraic equations for the unknown state of the system.

Nevertheless, this general description of Monte Carlo methods may not directly apply to some applications. It is natural to think that Monte Carlo methods are used to simulate random, or stochastic, processes, since these can be described by pdf's. However, this coupling is actually too restrictive because many Monte Carlo applications have no apparent stochastic content, such as the evaluation of a definite integral or the solution of a system of linear equations. However, in these cases and others, one can pose the desired solution in terms of pdf's, and while this transformation may seem artificial, this step allows the system to be *treated* as a stochastic process for the purpose of simulation and hence Monte Carlo methods can be applied to simulate the system. Therefore, one takes a broad view of the definition of Monte Carlo methods and includes in the Monte Carlo rubric all methods that involve statistical simulation of some underlying system, whether or not the system represents a real physical process. The range of applications that have been addressed with statistical simulation techniques is enormous, from the simulation of

galactic formation to quantum chromodynamics to the solution of systems of linear equations.

Major Components of a Monte Carlo Algorithm: Given the working definition of Monte Carlo method, let us now describe briefly the major components of a Monte Carlo method. These components comprise the foundation of most Monte Carlo applications. The primary components of a Monte Carlo simulation method include the following:

Probability distribution functions (pdf's) - the physical (or mathematical) system must be described by a set of pdf's.

Random number generator - a source of random numbers uniformly distributed on the unit interval must be available.

Sampling rule - a prescription for sampling from the specified pdf's, assuming the availability of random numbers on the unit interval, must be given.

Scoring (or tallying) - the outcomes must be accumulated into overall tallies or scores for the quantities of interest.

Additionally, an estimate of the statistical error (variance) as a function of the number of trials and other quantities have to be determined. To enhance the speed at which the experiments are carried out, methods for reducing the variance in the estimated solution (resulting into reduction of the computational time for Monte Carlo simulation) are applied. For complicated and large systems, the *Parallelization and vectorization*- based algorithms are used such as to allow the Monte Carlo method to be implemented efficiently on advanced computer architectures.

Thus, the Monte Carlo method is an artificial sampling method, which may be used for solving complicated problems in analytic formulation and for simulating purely statistical problems. The method is being used more and more in recent years, especially

in those cases in recent years, where the number of factors included in the problem is so large that an analytical solution is either extremely involved or rather impossible. The main idea behind the Monte Carlo method is either to construct a stochastic model that is in agreement with the whole problem analytically or to simulate the whole problem directly. In both cases, an element of randomness has to be introduced according to some well-defined rules. Then a large number of trials are performed, the results are observed, and finally a statistical analysis is undertaken in the usual way. The advantages of the method are, above everything that even very difficult problems can often be treated very easily, and desired modifications can be applied without much trouble. However, the disadvantages of Monte Carlo method are the relatively poor (vis-à-vis analytical methods) precision and the large number of trials that are necessary. Now, when very fast computers are readily available, the cumber of calculation in large number of trials is no longer a serious matter. This facility may be utilized to increase the number of trials sufficiently large so as to attain an acceptable degree of precision.

Monte Carlo Method and Study of Properties of Estimators of a Linear Model: A number of attempts have been made to investigate into the small sample properties of various estimators of linear model. We find a detailed discussion on the application of Monte Carlo method to study the properties of the estimation methods of multi-equation models in **Intriligator**, 1978, pp. 416-420.

In our endeavour to gauge the performance of LAD estimator vis-a vis LS estimator, we propose to use of Monte Carlo experiments. The essence of the Monte Carlo study is that instead of estimating unknown parameters using a specific technique, known parameters, which are chosen before hand, are estimated using different techniques. Comparisons between estimated and true parameters are subsequently used to

make inference about the different techniques. It simulates the process of estimating parameters using a controlled setting in which the true parameter values are known, like a ‘laboratory’ in which controlled experiments on econometric estimators can be studied.

III. Random Numbers, their Distributions and Generation: As mentioned before, random numbers play a very important role in the Monte Carlo method. Random numbers are almost always generated on computers by using some mathematical method. Such random numbers (generated by using some mathematical formula) are in fact **pseudo-random numbers**, so called because there exists a rule (the mathematical formula) for generating them and they have a well-defined periodicity. However, if the period is large enough and the numbers so generated satisfy certain statistical tests, then these numbers are random for all practical purposes. There are three leading methods for generating rectangular or uniformly distributed (pseudo) random numbers, namely the Mid-square method, the Fibonacci method and the Power method. These methods can be combined or hybridized also. Random numbers with various statistical distributions are obtained by suitable transformation of rectangular distributed random numbers. In most cases, first the rectangular distributed random numbers are transformed into normally distributed (standard) random numbers by an application of the Central Limit Theorem (i.e., the average of sufficiently large number of rectangular distributed random numbers has normal distribution) and suitable scaling.

Alternatively, if r_1 and r_2 are the two uniformly distributed random numbers lying between zero and unity, then $u = \sqrt{-2\ln(r_1)} \cos(2\pi r_2)$ follows standard normal distribution. Here $\pi = 4 \tan^{-1}(1) \cong 3.1415926535897932$ (Texas Instruments, p. 54). Then these normally distributed random numbers are transformed into other random numbers with desired distribution.

Pseudo-random Number Generators: Some of the pseudo-random number generators are discussed below.

(i) **Generator 1. (Mid-square method):** A Sequence of the pseudo-random numbers $\{r_i\}$, $i = 1, 2, \dots$, where each r_i is a 38 bit number is generated by the recursive algorithm defining

$$r_{i+1} = \text{mid (part of) } 38 \text{ bits of } r_i^2$$

$$r_1 = 01\ 0000\ 1011\ 1011\ 1011\ 1111\ 1010\ 0100\ 1101\ 1110$$

$i = 1, 2, \dots$ to any large integer.

This sequence terminates in zero for $i = 750000$ and has been very thoroughly tested for randomness and found useful in many practical problems.

The sequence of random numbers generated by this method may exhaust in the case of large problems. In such circumstances, the same sequence can be used repeatedly without repetition of results, provided the initial r_1 is not used at the identical place it was first called.

Alternatively, the random numbers generated can be increased two or four times by shifting the available random numbers. This procedure involves using in sequence not only r_i but also the fractional part of $2^{\alpha_1} r_i$, $2^{\alpha_2} (2^{\alpha_1} r_i)$, etc as random numbers before squaring r_i to obtain r_{i+1} . (It should however, be noted that through such shifts the left most digit is lost).

(ii) **Generator 2 (Mid-square-bit method):** Algorithm for generating a sequence of pseudo- random numbers $\{r_i\}$, $i=1, 2, \dots$ is given as follows (**Krishnamurthy & Sen**, p. 303).

Let r_1 , be an 8 bit random number.

- (a). Compute r_1^2 which is a 16 bit number.
- (b). Pick up its 8th (or 9th) bit, place it in the 8th- bit position of r_2
- (c). Take r_{11} = mid (part of) 8 bits of r_1^2 . Square r_{11} . Pick up the 8th (or 9th) bit of r_{11}^2 , place it in the 7th- bit position of r_2 .
- (d). Take r_{12} = mid (part of) 8 bits of r_{11}^2 . Square r_{12} . Pick up the 8th (or 9th) bit of r_{12}^2 , place it in the 6th bit position of r_2 .
- (e) Continue the process five more times to obtain an 8-bit r_2 .

Using this algorithm, r_3 can now be generated from r_2 . The process can be continued to generate r_i . Although, such generation is computationally more time consuming, the method has been tested and found to be very useful.

Generator 3 (Power residue method): The power residue method is the most commonly used method to generate pseudo random sequences. A sequence of integers r_1, r_2, \dots is defined recursively by

$$r_{i+1} \equiv \alpha r_i \pmod{\mu}$$

where r_1 = a starting value, α and μ are certain integers and the congruential notation means that r_{i+1} is the (positive) remainder when αr_i is divided by μ . Now, the sequence r_1, r_2, \dots will be a periodic sequence whose period cannot exceed μ , since division by μ can produce at most μ different remainders (viz, 0, 1, 2, ..., $\mu-1$). The integers α and μ should be selected in such a manner that they are capable of producing a very long period relative to the number of random numbers required in computation.

The Power residue algorithm for the generation of pseudo- random numbers is:

- (1) Let the given computer have a word length of K bits. Also let the arithmetic be carried out with the binary point to the extreme right or, equivalently, let the arithmetic be the integer.

- (2) Choose α = an integer of the form $8x \pm 3$ and close to $2^{K/2}$
- (3) Chose $\mu = 2^K$
- (4) Choose r_1 = any odd integer. Now αr_1 , =2K- bit number.
- (5) Discard the K high-order bit. The residue μ_2 is then the K low-order bit number.

The process is iterated. To obtain a pseudo random number r_i in $[0,1]$, the binary point is considered to be at the extreme left. The period of the sequence r_1, r_2, \dots is 2^{K-2} for a 32-bit computer, this sequence has a period $2^{30} \doteq 10^9$.

On most of the Personal Computers, we may generate uniformly distributed random numbers lying between 0 and 1 by the following algorithm:

- (1). Declare IU and IV as the two 2-byte integers.
- (2). Feed a seed, IU (preferably odd and of five digits, but less than 32766)
- (3). Define $IV = 259 \times IU$
- (4). If $IV \geq 0$ then $R = IV$; Re-initiate $IU = IV$; Standardize $R = R \times 0.3051851E-04$.

Store/print the random number R

Go to step (3) to generate the next random number with the re-initiated IU.

Else

- (5). Replace IV by $IV + 32767 + 1$; $R = IV$; Re-initiate $IU = IV$;

Standardize $R = R \times 0.3051851E-04$.

Store/print the random number R

Go to step (3) to generate the next random number with the re-initiated IU.

Distributions of Random Numbers: There are different types of random numbers, some discrete and others continuous. They follow different types of distribution. Data collected from nature following random processes throw up different types of distribution. Karl Pearson identified eight types of empirical distributions (Kapur & Saxena, pp. 296-300).

Additionally, there are umpteen number of theoretical distributions. Here we give the salient features of some distributions relevant for our purpose.

The Normal Distribution: A continuous random variable, e , has the normal distribution if it has the probability density function (pdf)

$$f(e) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{1}{2}\left(\frac{e-\mu}{\sigma}\right)^2\right\}; \text{where } -\infty < e < \infty$$

In the standard normal distribution the mean is taken as zero and the standard deviation as unity. The pdf now reduces to

$$f(e) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}e^2\right)$$

The important features of this function are :

1. The normal probability curve is bell shaped and symmetrical about the ordinate at $e = \mu$.
2. The ordinate decreases rapidly as e increases numerically. Curve extends to infinity on either side of the mean. The maximum ordinate is at $e = \mu$ and is given by

$$Y_{\max} = \frac{1}{\sigma\sqrt{2\pi}}.$$

3. The ordinate at $e = \mu$ divides the area under the normal curve into two equal parts.

Thus the median of the distribution coincides with the mean and the mode.

4. The two points of inflexion of the normal curve are equidistant from the mean.
5. For large values of σ , the normal curve tends to be flatter, while for small values of σ , it has a small peak.

6. No portion of the curve lies below the X-axes since normal probability function can not be negative.

Non-normal Distributions: The random nature of the error term has prompted econometricians to assume that the disturbance term, which is an influence of innumerable many factors not accounted for in the model, approaches normality according to the Central Limit Theorem. But **Bartels** (1977) is of the opinion that there are limit theorems, which are just likely to be relevant when considering the sum of number of components in a regression disturbance that leads to non-normal stable distribution characterised by infinite variance. Thus, the possibility of the error term following a non-normal distribution exists. In such cases, the least squares estimator will give less efficient estimates. Non-normal distributions are those, which acquire skewed, platykurtic or leptokurtic shapes when plotted i.e., they are asymmetrical about the mean or deviate significantly from the bell-shaped normal distribution. Three important and well-known non-normal distributions that have been considered in the study are discussed below.

Cauchy Distribution: This distribution is named after its discoverer KL Cauchy (1789-1857). The distribution is useful in statistical theory as it is an example of a distribution for which a number of theorems fail and hence emphasizes the need for scrutiny of the conditions under which statistical theorem holds.

The pdf of the Cauchy-distributed variate, e , is given by:

$$f(e) = \frac{1}{\pi} \cdot \frac{1}{1 + (e - \mu)^2}, \quad -\infty < e < \infty$$

The pdf curve is symmetrical about the point $e = \mu$ and thus $e = \mu$ gives us the median of the distribution. Also the density function is obviously maximum at $e = \mu$ so that this point is also the mode. The mean is given by:

$$\mu'_1 = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{ede}{1+(e-\mu)^2} = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{(e-\mu)de}{1+(e-\mu)^2} + \frac{\mu}{\pi} \int_{-\infty}^{\infty} \frac{de}{1+(e-\mu)^2}$$

Now since $\frac{1}{\pi} \int_{-\infty}^{\infty} \frac{de}{1+(e-\mu)^2} = 1$, we have

$$\mu'_1 = \mu + \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{ydy}{1+y^2} = \mu + \frac{1}{\pi} \left[\lim_{e, e' \rightarrow \infty} \int_{-e}^{e'} \frac{ydy}{1+y^2} \right]$$

If e, e' approach infinity independently, the second integral does not converge to a limit and thus, in the most general sense, the mean of the Cauchy distribution does not exist. However, if $e' = e$, the integral vanishes and thus if we take the principal value of the integral, the mean of the Cauchy distribution is at μ . Thus, in this sense the median and mode of the distribution coincide and the distribution is symmetrical.

The variance of the distribution is $\frac{1}{\pi} \int_{-\infty}^{\infty} \frac{(e-\mu)^2}{1+(e-\mu)^2} de$ and the integral does not converge. Thus for Cauchy distribution, the second and higher moments about the mean do not exist. The cumulative distribution function (cdf) of the distribution of e is :

$$F(e) = \frac{1}{\pi} \int_{-\infty}^e \frac{de}{1+(e-\mu)^2} = \frac{1}{\pi} \left[\tan^{-1}(e-\mu) + \frac{\pi}{2} \right]$$

The characteristic function of the distribution about the mean is

$$\phi(t) = E \left[\exp \{ it(e-\mu) \} \right] = \frac{1}{\pi} \int_{-\infty}^{\infty} \exp \{ it(e-\mu) \} \left\{ \frac{1}{1+(e-\mu)^2} \right\} de$$

$$= \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{\exp(iy)}{1+y^2} dy = \exp\{-|t|\}.$$

The Cauchy-distributed variate, e , relates to normally distributed variate in an interesting manner which is relevant to generate e from the latter. If u_1 and u_2 are independent normal variates with means m_1 and m_2 and variances σ_1^2 and σ_2^2 then the variate $e = \frac{u_1 - m_1}{u_2 - m_2}$ is Cauchy distributed (**Kapur & Saxena**, 1982; p. 427). In particular, the quotient of two independent standard normal variates is Cauchy distributed.

Gamma Distribution: The continuous random variate, e , which is distributed with pdf

$$f(e) = \frac{m^l}{\Gamma(l)} e^{l-1} \exp(-me) \quad , e \geq 0, l \geq 0, m > 0$$

is called a Gamma Variate with parameters l and m , and the distribution is called the Gamma distribution. The function $f(e)$ represents a probability density, since its integral over the range $(0, \infty)$ is unity.

The moment generating function (mgf) with respect to origin for gamma distribution with $m=l$ is

$$M_{e=0}(t) = \int_0^{\infty} \exp(te) \frac{1}{\Gamma(l)} e^{l-1} \exp(-e) de = \frac{1}{\Gamma(l)} \int_0^{\infty} e^{l-1} \exp\{-e(t-1)\} de = (1-t)^{-l},$$

provided that $|t| < 1$

Differentiating $M_{e=0}(t)$, r times with respect to t and putting $t=0$, we find

$$\mu'_r = l(l+1) \dots (l+r-1)$$

In particular, $\mu'_1 = l$, $\mu'_2 = l(l+1)$ and $\mu_2 = l$, $\mu_3 = 2l$ and $\mu_4 = 6l + 3l^2$. Hence,

$$\beta_1 = \frac{\mu_3^2}{\mu_2^3} = \frac{4}{l} \text{ and } \beta_2 = \frac{\mu_4}{\mu_2^2} = 3 + \frac{6}{l}$$

Hence as $l \rightarrow \infty, \beta_1 \rightarrow 0$ and $\beta_2 \rightarrow 3$ so that Gamma distribution tends to normal distribution as the parameter tends to infinity.

Two Important Properties of Gamma Variates: Gamma variates have two important properties relevant to our investigation; (i) the sum of two independent Gamma Variates with parameters l and m is a Gamma Variate with parameter $l+m$ and (ii) if u is a standard normal variate, then $e = \frac{1}{2}u^2$ is a Gamma variate with parameter $\frac{1}{2}$ (**Kapur & Saxena**, pp. 288,289). Using these properties, it is possible to generate Gamma distributed random numbers from standard normal random numbers.

Beta Distributions: There are two types of Beta-distributed random variables, the first is called β_1 -distributed, and the second is called β_2 -distributed. The continuous random variate, e , which is distributed with pdf

$$f(e) = \frac{1}{B(l,m)} e^{l-1} (1-e)^{m-1}, 0 \leq e \leq 1; l, m > 0$$

$$= 0, \text{ elsewhere}$$

is a Beta variate of the first kind with parameters l and m and is referred to as a $\beta_1(l, m)$ variate. The mgf with respect to the origin for this distribution does not have a simple form but the moments are easily found directly.

$$\mu'_r \text{ about origin} = \frac{1}{B(l,m)} \int_0^1 e^{r+l-1} (1-e)^{m-1} de$$

$$= \frac{B(r+l,m)}{B(l,m)}$$

$$= \frac{l(l+1)\dots(l+r-1)}{(l+m)(l+m+1)\dots(l+m+r-1)}, r = 1, 2, \dots$$

$$\text{In particular, } \mu'_1 = \frac{l}{l+m}, \mu'_2 = \frac{l(l+1)}{(l+m)(l+m+1)}, \mu'_3 = \frac{lm}{(l+m)^2(l+m+1)}$$

Putting $r=3$ and $r=4$ we successively obtain μ_3 and μ_4 and hence we can show that

$$\mu_3 = \frac{2lm(m-l)}{(l+m)^3(l+m+1)(l+m+2)}$$

$$\mu_4 = \frac{3lm\{lm(l+m-6)+2(l+m)^2\}}{(l+m)^4(l+m+1)(l+m+2)(l+m+3)}$$

$$\text{In case } l=m, f(e) = \frac{1}{B(1,1)} \cdot 1 = 1, 0 < e < 1$$

which is the probability function of uniform distribution in the range $(0,1)$

Beta Distribution of the Second Kind: The continuous random variate, e , which is distributed with pdf

$$f(e) = \frac{1}{B(l,m)} \cdot \frac{e^{l-1}}{(1+e)^{l+m}}, e \geq 0; l, m > 0$$

$$= 0 \text{ for } e < 0.$$

is a Beta variate of second kind with parameters l and m and may be referred to as

$\beta_2(l, m)$ variate. The function given above represents a probability density since

$$\int_0^\infty f(e) de = \int_0^\infty \frac{1}{B(l,m)} \cdot \frac{e^{l-1}}{(1+e)^{l+m}} de$$

$$= \frac{1}{B(l,m)} \int_0^1 y^{m-1} (1-y)^{l-1} dy$$

$$\text{on putting } 1+e = y^{-1}, de = \frac{dy}{y^2}$$

$$\therefore \int_0^\infty f(e) de = 1 \quad \text{since } B(l,m) = B(m,l)$$

$$\begin{aligned}
\text{Also } \mu'_r \text{ about origin} &= \int_0^{\infty} \frac{1}{B(l, m)} \cdot \frac{e^{l+r-1}}{(1+e)^{l+m}} de \\
&= \frac{1}{B(l, m)} \int_0^1 y^{m-r-1} (1-y)^{l+r-1} dy, \quad 1+e = y^{-1} \\
&= \frac{1}{B(l, m)} \int_0^1 Z^{l+r-1} (1-Z)^{m-r-1} dz \quad \text{where } Z=1-y \\
\therefore \mu'_r &= \frac{B(l+r, m-r)}{B(l, m)} = \frac{l(l+1)\dots(l+r-1)}{(m-1)(m-2)\dots(m-r)}; \quad r < m.
\end{aligned}$$

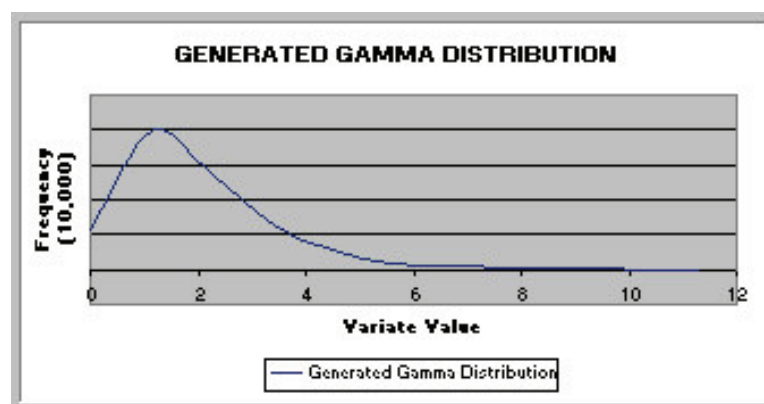
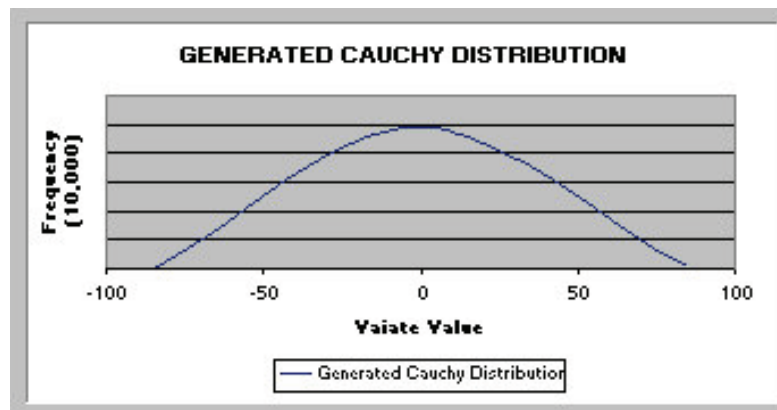
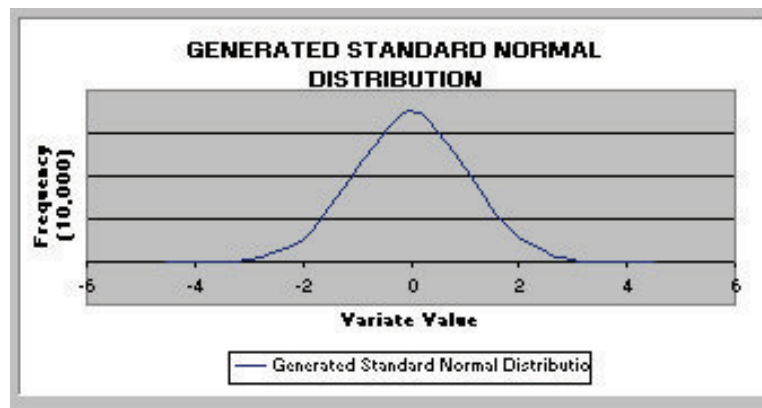
Beta-distributed random numbers are related to Gamma-distributed random numbers in a very interesting manner that may be exploited to generate Beta-distributed variates from the Gamma-distributed variates. From two independent Gamma variates, g_1 and g_2 with parameters l and m respectively, we may obtain $e_1 = g_1/(g_1 + g_2)$, which is a $\beta_1(l, m)$ distributed variate, and $e_2 = g_1/g_2$, which is a $\beta_2(l, m)$ distributed variate (Kapur & Saxena, 1982; p. 292).

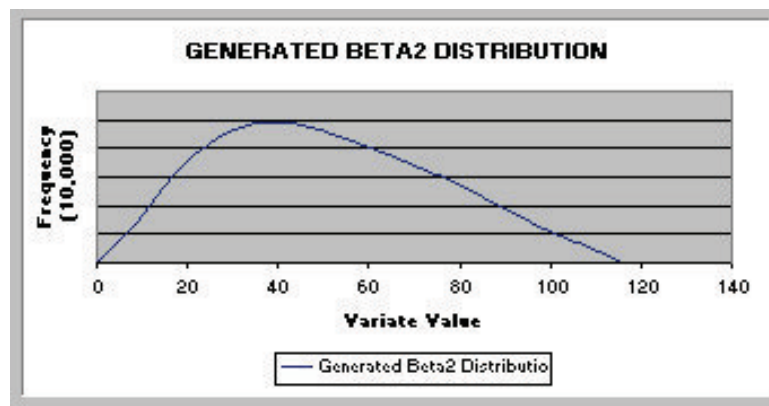
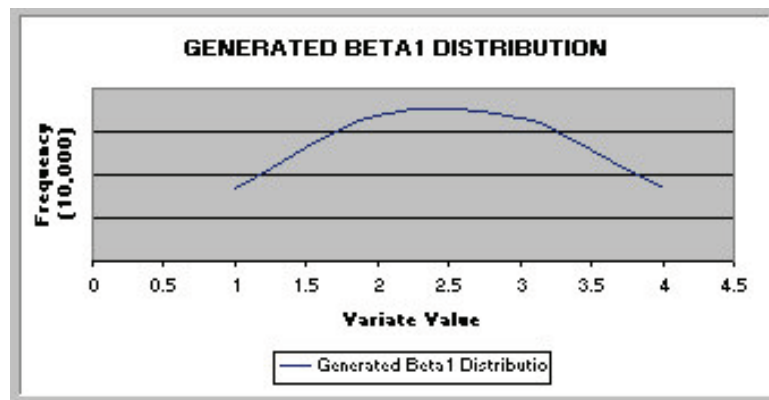
Generation of Random numbers following different Distributions: Simulation requires generation of a large number of random numbers that follow the desired distribution. The enterprise starts with the generation of uniformly distributed random numbers lying between 0 and 1. Uniformly distributed random numbers may be transformed into normally distributed random numbers, $x \sim N(0,1)$, by the transformation $x = \sqrt{-2 \ln(u_1)} \{\cos(2\pi u_2)\}$ where u_1 and u_2 are uniformly distributed independent random numbers lying between (0,1) and x is the standard normal variate (Knuth (1969), Texas Instruments Inc (1979), p 54). Alternatively, one may generate $x \sim N(0,1)$ from uniformly distributed $u(0,1)$ numbers, by using the Central Limit Theorem (Gillett 1979, p. 519; Kapur & Saxena, 1982, p. 386). However, this method is less accurate and time consuming than Knuth's method. Normally distributed variate, x , may be used to generate Gamma distributed variate, g , since, if x is a standard normal variate, then g

$= \frac{x^2}{2}$ is a Gamma variate with parameter $\frac{1}{2}$ (**Kapur & Saxena**, 1982; p 288). Due to the additive property of Gamma variates, if x_i ($i=1,2,\dots,n$) are n independent normal variates with means m_i and standard deviations σ_i then $g = \frac{1}{2} \sum_{i=1}^n \frac{(x_i - m_i)^2}{\sigma_i^2}$ is a Gamma variate with parameter $\frac{1}{2}n$ (**Kapur & Saxena**, 1982; p. 289). From two independent normally distributed variates x_1 and x_2 we may obtain a Cauchy distributed variate, since, if x_1 and x_2 are independent normal variates with means m_1 and m_2 and variances σ_1^2 and σ_2^2 then the variate $c = \frac{x_1 - m_1}{x_2 - m_2}$ is Cauchy distributed (**Kapur & Saxena**, 1982; p. 427).

In particular, the quotient of two independent standard normal variates is Cauchy distributed. From two independent Gamma variates, g_1 and g_2 with parameters l and m respectively, we may obtain $v_1 = \frac{g_1}{g_1 + g_2}$, which is a $\beta_1(l, m)$ distributed variate, and $v_2 = \frac{g_1}{g_2}$, which is a $\beta_2(l, m)$ distributed variate (**Kapur & Saxena**, 1982; p. 292). In general, starting from uniformly distributed variates, we may obtain a variate with almost any kind of distribution by a sequence of suitable transformations.

Graphic presentations of different distributions generated on computer (using the procedure mentioned above) give a fairly representative view of their nature. Among these, Gamma is a very skewed distribution for small shape and scale parameters. Gamma variates are non-negative. Beta variates are non-negative and largely platy-kurtic.





DasGupta, Madhuchhanda (2004) LEAST ABSOLUTE DEVIATION ESTIMATION OF MULT-EQUATION LINEAR ECONOMETRIC MODELS : A STUDY BASED ON MONTE CARLO EXPERIMENTS, Doctoral Dissertation, Dept. of Economics, North Eastern Hill University, Shillong (India); Supervisor – Prof. SK Mishra

CHAPTER 5

PERFORMANCE OF LAD IN ESTIMATION OF SINGLE-EQUATION LINEAR MODELS

I. Introduction: In this chapter we study Least Absolute Deviation (LAD) estimator as an alternative to OLS estimator of parameters of a single linear regression model. To estimate the regression parameters by LAD we use Fair-Schlossmacher algorithm. It is an iterative method. It works like a weighted Least Squares method of estimation, where weights are inverse of the absolute value of the residuals obtained by OLS. These weights change from iteration to iteration until convergence (up to a pre-determined accuracy level) is reached. This algorithm works better than most of the search methods of non-linear optimization.

After we obtain the estimates of regression coefficients by the Fair-Schlossmacher algorithm, we try to investigate if these estimates may further be improved (yield smaller value of the loss function). We have tried with several methods (Hooke-Jeeves, Nelder-Mead, Rosenbrock and Powel) but with no success. However, the Random Walk method has succeeded in improving the parameter estimates most of the time. It may be mentioned that the Random Walk method by itself does not minimize the loss function effectively. However, it improves the Fair-Schlossmacher estimates quite effectively. This procedure – first obtaining LAD estimates by Fair-Schlossmacher method and thereafter improving it by the Random Walk method – has been named as the LADRW estimation. Therefore, this method also is a candidate method of estimation.

II. Specification of Single Equation Model: A single equation model is specific in the sense of (i) the number of explanatory variables (except the constant) in the model, (ii) distribution (or probability density function) of disturbance vector in the model including the magnitude of variance of the disturbance vector, (iii) presence (number and size) of outliers in the disturbance vector and lastly (iv) the sample size or the number of cases (or no. of observations). Specification about linearity or otherwise also is important. However, in our experiments we have considered linear models only. Other particulars of the specification are given as follows.

a). Size of the Single Equation Model: In our endeavor to ascertain the efficiency of the LAD estimator, Monte Carlo experiments have been conducted using three models of different sizes.

(i) **Model 1:** Model#1 comprises of 1 explanatory variable and additionally a constant.

The Model #1 is:

$$y_i = \sum_{j=1}^{1+1} a_j x_{ij} + e_i; i = 1, n; x_{i,m+1} = 1 \quad \forall i(1, n)$$

(ii) **Model 2:** Model#2 has 3 explanatory variables and an additional constant. The model #2 is:

$$y_i = \sum_{j=1}^{3+1} a_j x_{ij} + e_i; i = 1, n; x_{i,m+1} = 1 \quad \forall i(1, n)$$

(iii) **Model 3:** Model#3 has 5 explanatory variables and an additional constant. The model#3 is:

$$y_i = \sum_{j=1}^{5+1} a_j x_{ij} + e_i; i=1, n; x_{i,m+1} = 1 \quad \forall i(1, n)$$

In the above models, the y 's represent the dependent variables and the x 's represent the explanatory variables. The error (residual or disturbance) term is e . The sample size is n .

b). Sample Size: Samples of two different sizes have been taken in the study

(i) Sample of size (no. of cases or observations) = 20.

(ii) Sample of size (no. of cases or observations) = 50.

c). Distribution of Disturbance term: To study the effect of the probability distribution of the error term on the efficiency of the candidate estimator vis-à-vis OLS, we have considered five probability distributions. They are (i) Normal, (ii) Cauchy, (iii) Gamma, (iv) Beta₁, and (v) Beta₂. Gamma, Beta₁ and Beta₂ distributed errors have been generated with parameter of size =2. To help the visualization of the nature of these distributions, the figures in the preceding chapter give their pictorial view. As already mentioned earlier, these graphs were obtained by frequency classification and plotting of 10,000 random numbers (of specified distribution) generated by a computer program.

d). Standard Deviation of the Disturbance term: The standard deviation of error used in the study is 0.1.

e). Number and size of Outliers: Outliers of different number and magnitude have also been introduced to the error vector to examine whether their presence, number and size affect the efficiency and bias of the estimator under study.

- (i) The numbers of outliers used in the study are 0 (i.e. absence of outliers), 1, 3 and 5.
- (ii) The magnitudes of outliers (when present) are 0.5, 1 and 2 times the mean value of the dependent variable (\bar{Y}_i) in a particular model.

III. Numerical Details: Monte Carlo experiments are basically numerical methods. Additionally, they are computer-based since it is practically impossible to conduct such experiments manually. Once the models are specified, random numbers are generated in accordance with the specifications and the parameters are estimated by the candidate methods of estimation. Necessary information regarding the experiments conducted in this study is given below.

1. We have generated uniformly distributed random numbers lying between zero and unity (0,1). With the random numbers, the matrix of explanatory variables, $x(n,m-1)$, has been formed. The number of rows (cases or observations, n) as well as the number of columns (explanatory variables less one = $m-1$) are as specified in a particular model. The last (that is m^{th}) column of x is a unit vector ($=1$ for all i ; $i=1,n$). To scale up the magnitude of the numerical values constituting the matrix X (except for the last column), we have multiplied the elements, x_{ij} ($i=1,n$; $j=1,m-1$) by the scalars 10. Appropriate scaling is needed to keep up accuracy in the numerical analysis.

2. Then we have chosen (subjectively) appropriate number of regression coefficients, including the constant and used them as parameters to obtain $y_i = \sum_{j=1}^m a_j x_{ij}; i=1,n$. This gives us the non-stochastic y vector.

3. Next, the disturbance vector, e_i ; $i=1,n$ of appropriate distribution has been generated and added to y to obtain the stochastic dependent variable y .
4. The accuracy level of the LAD estimates has been fixed at 0.0001.
5. The tolerance level for accuracy which is required for convergence of parameter estimates has been fixed at 0.0001.
6. We have estimated the regression parameters, \mathbf{a} , by OLS.
7. We have estimated the regression parameters, \mathbf{a} , by LAD estimator. To obtain the LAD estimates, 100 iterations have been performed. It is assumed that such a number of iteration is sufficient for an accurate estimation.
8. The experiment conducted has been replicated 100 times – each time drawing fresh samples of random disturbance vectors (while keeping the structure fixed), adding them to the dependent variable (non-stochastic y as in 3 above) and estimating the parameters by the candidate methods of estimation OLS and LAD).

The step-wise procedure may be described as follows:

- (i) Generate uniformly distributed random numbers to obtain the $x(n,m)$ scale it up by a predetermined positive scalar (say 10) and augment $x(n,m)$ by a unit vector to obtain $x(n,m+1)$. Here n and m are pre-specified.
- (ii) Generate $y = \mathbf{x}\mathbf{a}$.
- (iii) Using another seed, generate a set of random variables of desired distribution (agreeing with the order of y vector) and add them to y . The dependent variable thus obtained is now stochastic.
- (iv) Decide on the number and the sizes of outliers and add them to y_i ($i=1,n$) at randomly chosen locations.

- (v) Estimate the parameter vector \mathbf{a} (call it \hat{a}) by different estimators (OLS and LAD).
- (vi) Repeat steps (iii) to (v) for the desired number of times (100 here) for replication.
- (vii) Use the results in (vi) to obtain RMS of deviations of the estimated \hat{a} from the true \mathbf{a} while \hat{a} is estimated by the different estimators (OLS and LAD). Obtain relative RMS (relative to OLS =1) for different estimators.

Step (5) is repeated for different distributions (of error vectors) with 0.1 magnitude of standard deviations, different numbers and magnitudes of outliers and different model sizes of different sample sizes (no. of cases or observations).

IV. Details of Specification-wise no. of experiments: It is facilitating to describe the details of specification-wise number of experiments conducted in our study. In all, we have conducted 300 experiments (each one with 100 replicates yielding one RMS for each candidate estimator, OLS and LAD). These details are as follows:

Size of Model: Model#1 (100), Model#2 (100), Model#3 (100)

Sample Size: Small i.e. 20 observation (150), Larger, i.e. 50 observations (150)

Distributions: Normal (60), Cauchy (60), Gamma (60), B_1 (60), B_2 (60)

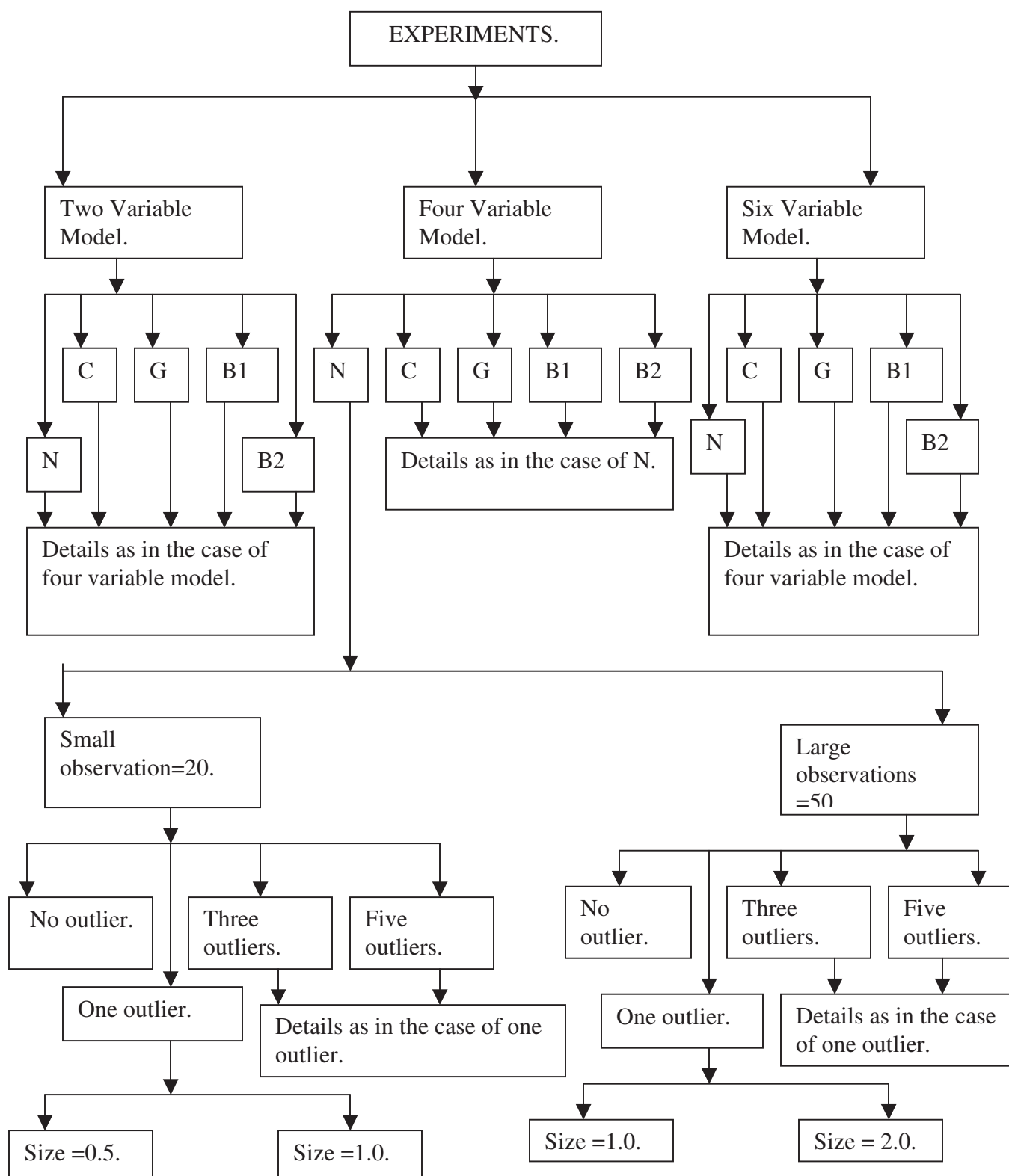
Number of Outliers: None (120), One (60), Three (60), Five (60)

Size of Outliers: None/zero (120), Small (90), Large (90). In the experiments we have used four different factors (call k) for generating outliers of different sizes. For small samples we use two values of k (0.5 and 1.0). For large samples we use $k=1.0$ and $k=2.0$. The size of outliers is $k*\text{mean}(y)$ or k times the mean of the dependent variable. The specified number of outliers are added to different observations of the dependent variable (y) at randomly selected locations (cases or observation). The value of the decision

variable OSIZE takes on these values of k . Thus OSIZE takes on three different values – 0.5, 1.0 and 2.0 depending on the specified conditions.

V. Findings of Monte Carlo Experiments: In accordance with the design of Monte Carlo Experiments we have run the simulation program for 300 times with the specifications as given in **Details of Specification-wise no. of experiments** in the earlier section of this chapter. The computer program (source codes of computing programs in FORTRAN 77) is appended. The program estimates 100 replicates (with different disturbance vectors of a specified distribution and different outliers specific in number and size) of regression parameters of a specified model by OLS, LAD (obtained by the Fair-Schlossmacher algorithm) and LADRW (Random Walk method of optimization applied on the iterative estimates obtained by the Fair-Schlossmacher algorithm of LAD estimation) and the RMS of the regression coefficients (with the true parameters as a reference) of the latter two estimators with respect to those of the OLS estimator. Thus for each run we obtain two values of relative norms, one for LAD and the other for LADRW estimator. We have also converted the relative norms of each candidate estimator (LAD and LADRW) into two binary variables (one each for LAD and LADRW such that if the relative norm of a candidate estimator is smaller than the OLS norm then the binary variable takes on a value of unity or zero otherwise. In this scheme, the *unity* value of the related binary value signifies *success* (that is, the candidate estimator has succeeded in estimating the parameters of the regression equation more efficiently than does the OLS estimator) while the value of *zero* signifies that OLS estimator dominates (or is as good as) the candidate estimator and it is considered as a *failure*.

DESIGN OF EXPERIMENTS: SINGLE EQUATION MODEL



a). Single Criterion Frequency Distribution of Success of Different Estimators: First we present the frequency distribution of success of different candidate estimators in dominating OLS as tabulated in the tables 1 through 5. The entries in the cells of these tables are the number of success (out of the total number of experimental trials or cases - given in the rightmost columns of the tables) and where the ratio of the number of success to the number of trials (cases) is larger than 0.5, the entry has been highlighted. We observe the following:

- (1). Among the different disturbance distributions only Cauchy and B_2 favour LAD (or LADRW) estimators against the OLS estimator. Among the distributions, Cauchy is absolutely dominating.
- (2). Large size samples favour LAD (as well as LADRW) estimator vis-à-vis the OLS estimator.
- (3). Larger size models (with 4 or 6 parameters) favour LAD estimator against the OLS estimator, but LADRW dominates OLS for all model sizes.
- (4). No outlier or single outlier cases give no edge to the LAD estimator against the OLS estimator, but for larger number (3 or 5) of outliers LAD outperforms OLS. However, LADRW performs better than OLS in presence of outliers (1, 3 or 5) while it has no edge over OLS if no outlier is present. LAD and LADRW outperform OLS for whatever size of outliers.

We conclude therefore that models with larger number of parameters, sufficiently large sample size (observations) and Cauchy or B_2 distributed errors favour LAD estimator against the OLS estimator. Performance of LAD (or LADRW) in case of Cauchy distributed disturbances is spectacular. We observe that out of 60 experiments

with Cauchy distributed disturbance term LAD outperforms OLS in 59 cases. LADRW outperforms OLS in 60 out of 60 experiments.

TABLE 1. EFFECT OF DISTRIBUTION

Sl. No.	DISTBUTION	LAD	LADRW	CASES
1	n	18	23	60
2	c	<u>59</u>	<u>60</u>	60
3	g	21	24	60
4	v	10	14	60
5	b	<u>53</u>	<u>52</u>	60

Note: v stands for Beta₁ and b or B stands for B₂.

TABLE 2. EFFECT OF SAMPLE SIZE

Sl. No.	SAMPLE SIZE	LAD	LADRW	CASES
1	s	73	81	150
2	l	<u>88</u>	<u>92</u>	150

TABLE 3. EFFECT OF MODEL SIZE

Sl. No.	MODEL SIZE	LAD	LADRW	CASES
1	2	49	<u>58</u>	100
2	4	<u>56</u>	<u>58</u>	100
3	6	<u>56</u>	<u>57</u>	100

TABLE 4. EFFECT OF NO. OF OUTLIERS

Sl. No.	NO. OF OUTLIERS	LAD	LADRW	CASES
1	0	41	41	120
2	1	30	<u>32</u>	60
3	3	<u>42</u>	<u>49</u>	60
4	5	<u>48</u>	<u>51</u>	60
Total (AT LEAST ONE OUTLIER)		<u>120</u>	<u>132</u>	180

TABLE 5. EFFECT OF SIZE OF OUTLIERS

Sl. No.	OUTLIER SIZE	LAD	LADRW	CASES
1	0	10	10	30
2	a	11	10	30
3	b	10	10	30
4	c	10	11	30
Total (ZERO OUTLIERS)		41	41	120
5	1	<u>43</u>	<u>51</u>	90
6	2	<u>77</u>	<u>81</u>	90

NOTE: In 0, a, b, and c categories SIZE of OUTLIERS = 0

b). Double Criteria Frequency Distribution of Success (and failures) of Different

Estimators: Next we present the frequency distribution of success (as well as failure) of different candidate estimators (LAD and LADRW) in dominating the OLS estimator as tabulated in the tables 6 through 15. The leading entries (odd number rows) in the cells of these tables are the number of success (out of the total number of experimental trials or cases) while the trailing entries (even number rows) are the number of failure. Where the number of success is larger than the number of failure (i.e. the probability of success is larger than 0.5), the success entry has been highlighted. In summarizing our findings estimator-wise, we observe that (i) for Cauchy and B_2 distributed disturbance terms LAD (as well as LADRW) estimator outperforms the OLS estimator irrespective of presence or absence of outliers or size of the model or the sample size. Presence of outliers further strengthens the LAD (as well as LADRW) estimator, (ii) Cauchy distributed disturbance term is especially favourable to the LAD/LADRW estimator and (iii) presence of larger number and/or large size outliers favour LAD/LADRW for Normal, Gamma and B_1 distributed disturbance term also.

Table 6: EFFECT OF DISTRIBUTION AND SAMPLE SIZE

		LAD					LADRW				
		N	C	G	V	B	N	C	G	V	B
S	6	<u>29</u>	7	5	<u>26</u>		9	<u>30</u>	9	6	<u>27</u>
	24	1	23	25	4		21	0	21	24	3
L	12	<u>30</u>	14	5	<u>27</u>		14	<u>30</u>	15	8	<u>25</u>
	18	0	16	25	3		16	0	15	22	5

Table 7: EFFECT OF SAMPLE SIZE AND MODEL SIZE

		LAD			LADRW		
		2	4	6	2	4	6
s	22	<u>26</u>		25	<u>29</u>	<u>26</u>	<u>26</u>
	28	24		25	21	24	24
l	<u>27</u>	<u>30</u>		<u>31</u>	<u>29</u>	<u>32</u>	<u>31</u>
	23	20		19	21	18	19

Table 8: EFFECT OF SAMPLE SIZE AND No. OF OUTLIERS

		LAD				LADRW			
		0	1	3	5	0	1	3	5
s	20	12	<u>20</u>	<u>21</u>		21	14	<u>23</u>	<u>23</u>
	40	18	10	9		39	16	7	7
l	21	<u>18</u>	<u>22</u>	<u>27</u>		20	<u>18</u>	<u>26</u>	<u>28</u>
	39	12	8	3		40	12	4	2

Table 9: EFFECT OF DISTRIBUTION AND MODEL SIZE

		LAD					LADRW				
		N	C	G	V	B	N	C	G	V	B
2	6	<u>20</u>	8	2	<u>13</u>		10	<u>20</u>	10	6	<u>12</u>
	14	0	12	18	7		10	0	10	14	8
4	6	<u>20</u>	6	4	<u>20</u>		7	<u>20</u>	7	4	<u>20</u>
	14	0	14	16	0		13	0	13	16	0
6	6	<u>19</u>	7	4	<u>20</u>		6	<u>20</u>	7	4	<u>20</u>
	14	1	13	16	0		14	0	13	16	0

Table 10: EFFECT OF DISTRIBUTION AND No. OF OUTLIERS

LAD						LADRW					
	N	C	G	V	B		N	C	G	V	B
0	0	<u>23</u>	0	0	<u>18</u>	0	<u>24</u>	0	0	<u>17</u>	
	24	1	24	24	6	24	0	24	24	7	
1	3	<u>12</u>	4	0	<u>11</u>	4	<u>12</u>	4	1	<u>11</u>	
	9	0	8	12	1	8	0	8	11	1	
3	6	<u>12</u>	<u>8</u>	4	12	<u>9</u>	<u>12</u>	<u>10</u>	6	<u>12</u>	
	6	0	4	8	0	3	0	2	6	0	
5	<u>9</u>	<u>12</u>	<u>9</u>	6	<u>12</u>	<u>10</u>	<u>12</u>	<u>10</u>	<u>7</u>	<u>12</u>	
	3	0	3	6	0	2	0	2	5	0	

Table 11: EFFECT OF DISTRIBUTION AND SIZE OF OUTLIERS

LAD								LADRW							
	0	a	b	c	S ₀	1	2		0	a	b	c	S ₀	1	2
n	0	0	0	0	0	3	15	0	0	0	0	0	0	7	<u>16</u>
	6	6	6	6	24	15	3	6	6	6	6	6	24	11	2
c	<u>6</u>	<u>6</u>	<u>6</u>	<u>5</u>	<u>23</u>	<u>18</u>	<u>18</u>	<u>6</u>	<u>6</u>	<u>6</u>	<u>6</u>	<u>6</u>	<u>24</u>	<u>18</u>	<u>18</u>
	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
g	0	0	0	0	0	5	<u>16</u>	0	0	0	0	0	0	8	<u>16</u>
	6	6	6	6	24	13	2	6	6	6	6	6	24	10	2
v	0	0	0	0	0	0	<u>10</u>	0	0	0	0	0	0	1	<u>13</u>
	6	6	6	6	24	18	8	6	6	6	6	6	24	17	5
b	<u>4</u>	<u>5</u>	<u>4</u>	<u>5</u>	<u>18</u>	<u>17</u>	<u>18</u>	<u>4</u>	<u>4</u>	<u>4</u>	<u>4</u>	<u>5</u>	<u>17</u>	<u>17</u>	<u>18</u>
	2	1	2	1	6	1	0	2	2	2	2	1	7	1	0

Note: Column S₀ is the sum of (0,a,b and c) for zero no. of outliers.

Table 12: EFFECT OF SAMPLE SIZE AND No. OF OUTLIERS*

LAD								LADRW							
	0	a	b	c	S ₀	1	2		0	a	b	c	S ₀	1	2
s	5	5	5	5	20	17	<u>36</u>	5	5	5	6	21	22	<u>38</u>	
	10	10	10	10	40	28	9	10	10	10	9	39	23	7	
l	5	6	5	5	21	<u>26</u>	<u>41</u>	5	5	5	5	20	<u>29</u>	<u>43</u>	
	10	9	10	10	29	19	4	10	10	10	10	40	16	2	

Table 13: EFFECT OF MODEL SIZE AND No. OF OUTLIERS

LAD					LADRW			
	0	1	3	5	0	1	3	5
2	10	10	<u>13</u>	<u>16</u>	9	<u>12</u>	<u>18</u>	<u>19</u>
	30	10	7	4	31	8	2	1
4	16	10	<u>14</u>	<u>16</u>	16	10	<u>16</u>	<u>16</u>
	24	10	6	4	24	10	4	4
6	15	10	<u>15</u>	<u>16</u>	16	10	<u>15</u>	<u>16</u>
	25	10	5	4	24	10	5	4

Table 14: EFFECT OF MODEL SIZE AND No. OF OUTLIERS*

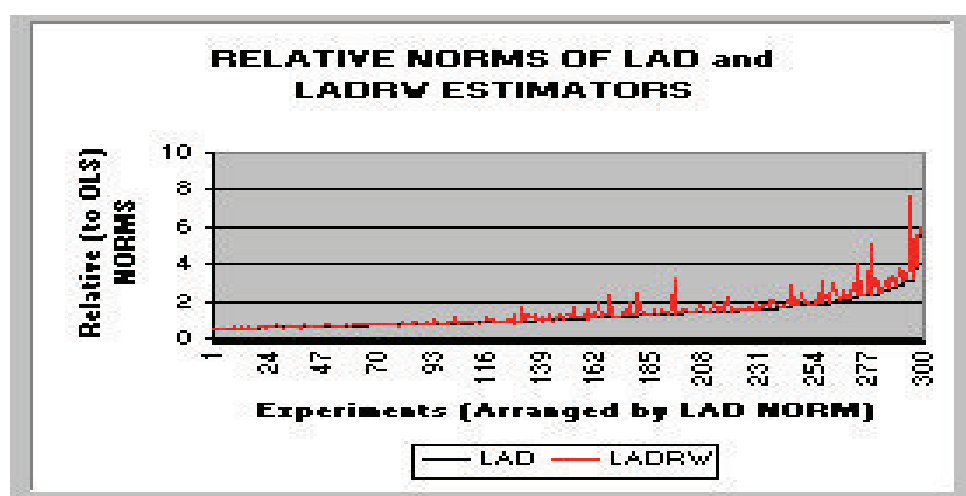
LAD								LADRW							
	0	a	b	c	S ₀	1	2		0	a	b	c	S ₀	1	2
2	2	3	2	3	10	14	<u>25</u>	2	2	2	3	9	<u>20</u>	<u>29</u>	
	8	7	8	7	30	16	5	8	8	8	7	31	10	1	
4	4	4	4	4	16	14	<u>26</u>	4	4	4	4	16	<u>16</u>	<u>26</u>	
	6	6	6	6	24	16	4	6	6	6	6	24	14	4	
6	4	4	4	3	15	15	<u>26</u>	4	4	4	4	16	15	<u>26</u>	
	6	6	6	7	25	15	4	6	6	6	6	24	15	4	

Table 15: EFFECT OF MODEL SIZE AND No. OF OUTLIERS*

LAD								LADRW							
	0	a	b	c	S ₀	1	2		0	a	b	c	S ₀	1	2
0	10	11	10	10	41	0	0	10	10	10	11	41	0	0	
	20	19	20	20	79	0	0	20	20	20	19	79	0	0	
1	0	0	0	0	0	11	<u>19</u>	0	0	0	0	0	11	<u>21</u>	
	0	0	0	0	0	19	11	0	0	0	0	0	19	9	
3	0	0	0	0	0	14	<u>28</u>	0	0	0	0	0	<u>19</u>	<u>30</u>	
	0	0	0	0	0	16	2	0	0	0	0	0	11	0	
5	0	0	0	0	0	<u>18</u>	<u>30</u>	0	0	0	0	0	<u>21</u>	<u>30</u>	
	0	0	0	0	0	12	0	0	0	0	0	0	9	0	

* Column S₀ is the sum of (0,a,b and c) for zero no. of outliers.

c). A Comparative Study of LAD and LADRW estimation Methods: LAD as well as LADRW minimize the absolute norm of the disturbance vector. LAD uses Fair-Schlossmacher algorithm only, but LADRW applies random walk minimization procedure after the Fair-Schlossmacher algorithm fails to improve the estimates. It is likely, therefore, that LADRW would have an edge over the simple Fair-Schlossmacher LAD estimation procedure. Indeed we have found that out of 300 experiments LADRW outperforms LAD in 282 (94%) cases. The summary statistics of relative (to OLS) norms (tables 16 through 18) clearly indicate the superiority of LADRW over LAD. A comparison of relative norms (see figure below) of LAD and LADRW suggests that LADRW enhances larger LAD norms more effectively. A regression analysis (with a constant term included) to explain the difference between the relative RMS of the two algorithms suggest that Cauchy distribution of disturbance term and the number as well as size of outliers cause LADRW to perform better than LAD (table 19). Exclusion of the constant term from the regression model makes B_1 a significant determinant (table 20).



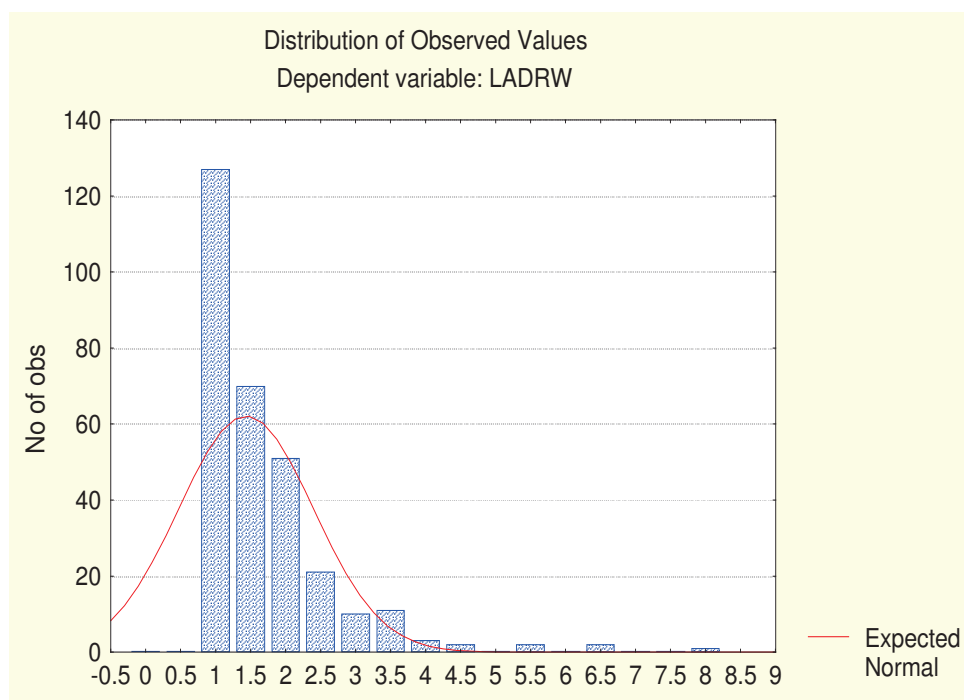
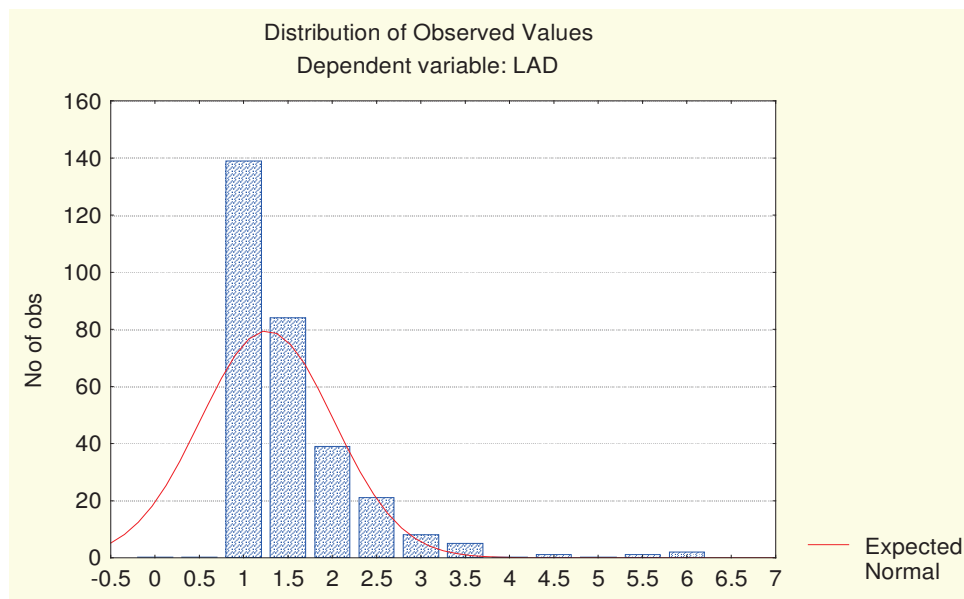


Table 16. A Comparative view of Relative RMS of LAD and LADRW Algorithms

Relation	LAD > LADRW	LAD = LADRW	LAD < LADRW	Total
No. of cases	17	1	282	300

Table 17. Summary Statistics regarding LAD and LAD-RW Relative RMS

Statistics	Cases	Mean	Median	Mode	Minimum	Maximum
LAD	300	1.262053	1.045	0.708	0.502	5.78
LADRW	300	1.432847	1.1985	0.731	0.518	7.673

Table 18. PERCENTILES OF LAD and LAD-RW Relative RMS

Percentiles	10	20	30	40	50	60	70	80	90
LAD	0.637	0.708	0.792	0.885	1.045	1.235	1.404	1.657	2.180
LADRW	0.646	0.736	0.814	0.934	1.199	1.371	1.591	1.906	2.511

Table 19. Difference in LAD and LADRW explained by Decision Variables ($R^2 = 0.252$)

Variables	Regress. Coeff	Std. Err	Beta	t value	Sig.
(Constant)	-0.05816	.052		-1.116	.265
C	0.265	.063	.270	4.217	.000
G	-0.02525	.063	-.026	-.402	.688
B1	-0.05980	.063	-.061	-.952	.342
B2	0.01683	.063	.017	.268	.789
OBSERV	0.06756	.043	.086	1.554	.121
NOUT	0.02621	.013	.129	1.973	.049
OSIZE	0.161	.039	.285	4.104	.000
Linear Regression with the constant term					

Table 20. Difference in LAD and LADRW explained by Decision Variables ($R^2 = 0.371$)

Variables	Regress. Coeff	Std. Err	Beta	t value	Sig.
N	-0.05816	.052	-.061	-1.116	.265
C	0.207	.052	.216	3.970	.000
G	-0.08341	.052	-.087	-1.601	.111
B1	-0.118	.052	-.123	-2.264	.024
B2	-0.4133	.052	-.043	-.793	.428
OBSERV	0.6756	.043	.112	1.554	.121
NOUT	0.2621	.013	.162	1.973	.049
OSIZE	0.161	.039	.364	4.104	.000
Linear Regression without the constant term					

d). Effect of Decision Variables on RMS (norm relative to OLS) of LAD/LADRW

Estimators: As mentioned earlier, we have measured the effectiveness of LAD and LADRW estimators by the RMS of the estimated regression coefficients vis-à-vis the true coefficients. We obtain three RMS values in every experiment (with 100 replicates), one each for OLS, LAD and LADRW. Then we have obtained two relative RMS, say $R_{LAD} = (RMS_{OLS} / RMS_{LAD})$ and $R_{LADRW} = (RMS_{OLS} / RMS_{LADRW})$. If $R_{LAD} > 1$ then LAD outperforms OLS. Similarly, if $R_{LADRW} > 1$ then LADRW outperforms OLS.

Now we investigate into the power of different decision variables (specification of the model such as distribution of the disturbance term, number and size of outliers, sample size and the model size, i.e. number of regression coefficients to be estimated in a model) in explaining the magnitude of relative norms, R_{LAD} and R_{LADRW} . We have accomplished this by regression analysis. First, for the LAD estimator, we find (table 21) that except for B_1 all regression coefficients are significant. At that, the beta coefficients (table 21, col. 4) are much larger for Cauchy and B_2 distributed disturbance term. The third most influential decision variable is the outlier size. For the LADRW estimator (table 22), the regression coefficients for all decision variables but COEFF (no. of coefficients in the model referring to the model size) are statistically significant. The regression coefficient associated with B_1 distributed disturbance term, which was insignificant in case of LAD, is now significant at 5% level of significance. Cauchy and B_2 distributed disturbance term and outlier size keep up their predominance as in case of LAD (compare tables 21 and 22, col. 4). The figures regarding residuals and estimated dependent variables ($RLAD$ and $RLADRW$) indicate nonlinearity and, perhaps,

heteroskedasticity. The coefficient of determination (R^2) is very large for LAD as well as LADRW estimator. Logarithmic transformation of the dependent variables (Tables 23 and 24) do not improve our conclusions.

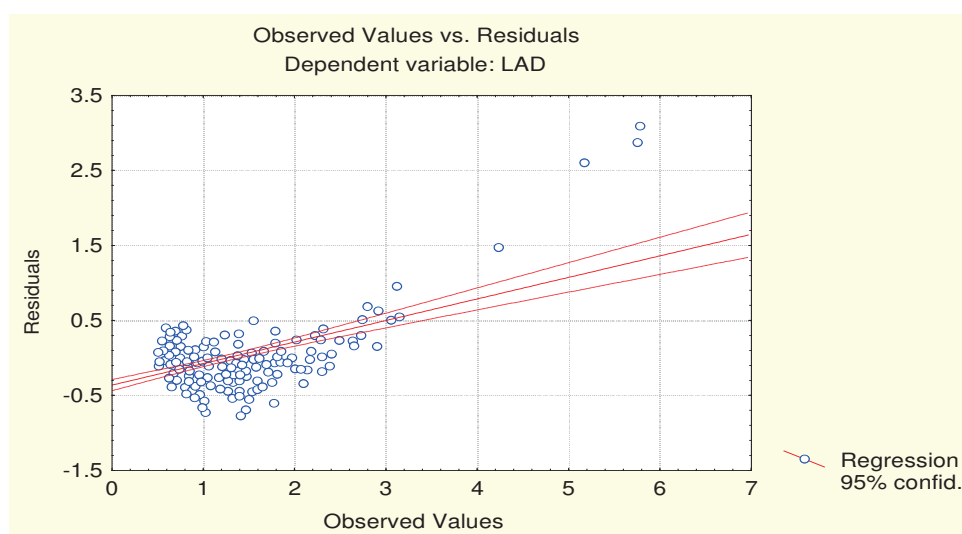
Table 21. Effect of Decision Variables on Relative (to OLS) Norm of LAD Estimator						
Statistic	Regression Coefficient	Std. Error	Beta	Std. Error of Beta	(291) value	Sig (probability)
Variable						
N	0.05823	0.02581	0.19130	0.08480	2.26	0.02482
C	0.37519	0.02581	1.23255	0.08480	14.54	0.00000
G	0.06163	0.02581	0.20246	0.08480	2.39	0.01760
B1	0.01232	0.02581	0.04048	0.08480	0.48	0.63344
B2	0.31641	0.02581	1.03946	0.08480	12.26	0.00000
NOBSERV	0.05148	0.02489	0.10695	0.05171	2.07	0.03950
COEFF	0.20680	0.04256	0.07032	0.01447	4.86	0.00000
NOUT	0.11503	0.02844	0.06387	0.01579	4.04	0.00007
OSIZE	0.26467	0.03074	0.40159	0.04664	8.61	0.00000
Linear Regression without the constant term;				($R^2 = 0.9247$, $F(9,291) = 397.14$)		

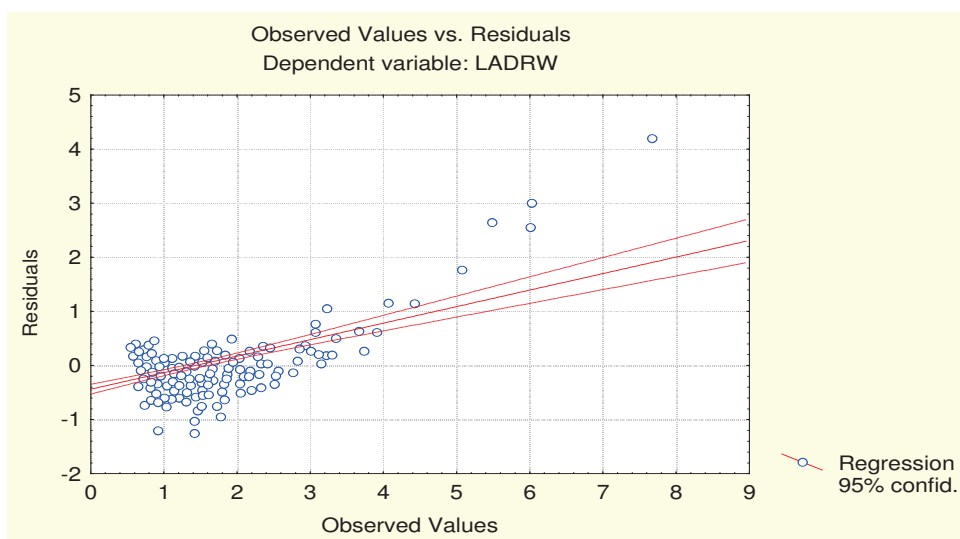
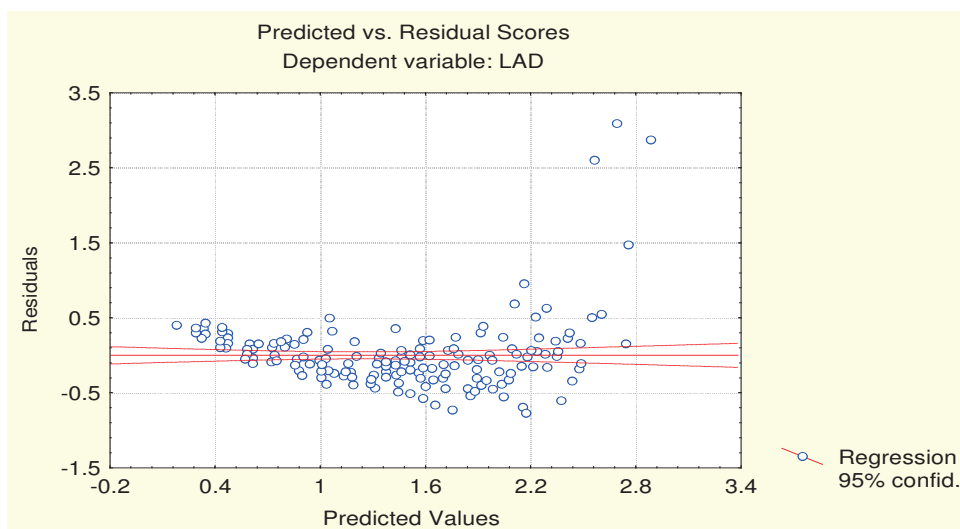
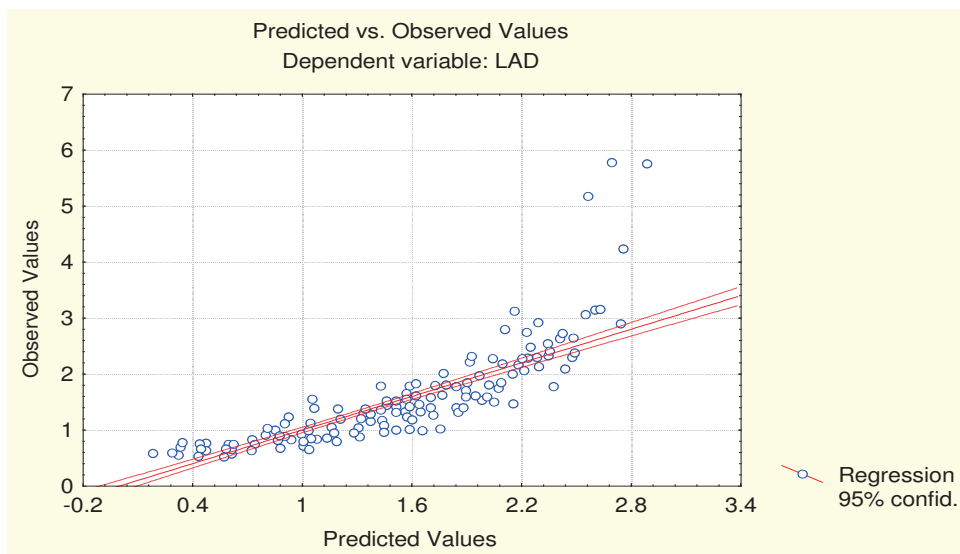
Table 22. Effect of Decision Variables on Relative (to OLS) Norm of LAD-RW Estimator						
Statistic	Regression Coefficient	Std. Error	Beta	Std. Error of Beta	(291) value	Sig (probability)
Variable						
N	0.11236	0.02897	0.43363	0.11180	3.88	0.00013
C	0.45085	0.02897	1.73990	0.11180	15.56	0.00000
G	0.10872	0.02897	0.41955	0.11180	3.75	0.00021
B1	0.05779	0.02897	0.22302	0.11180	1.99	0.04700
B2	0.33651	0.02897	1.29863	0.11180	11.62	0.00000
NOBSERV	0.07150	0.02793	0.17451	0.06818	2.56	0.01098
COEFF	-0.01203	0.04776	-0.00481	0.01908	-0.25	0.80134
NOUT	0.13810	0.03192	0.09008	0.02082	4.33	0.00002
OSIZE	0.31560	0.03449	0.56254	0.06149	9.15	0.00000
Linear Regression without the constant term;				($R^2 = 0.9052$, $F(9,291) = 308.62$)		

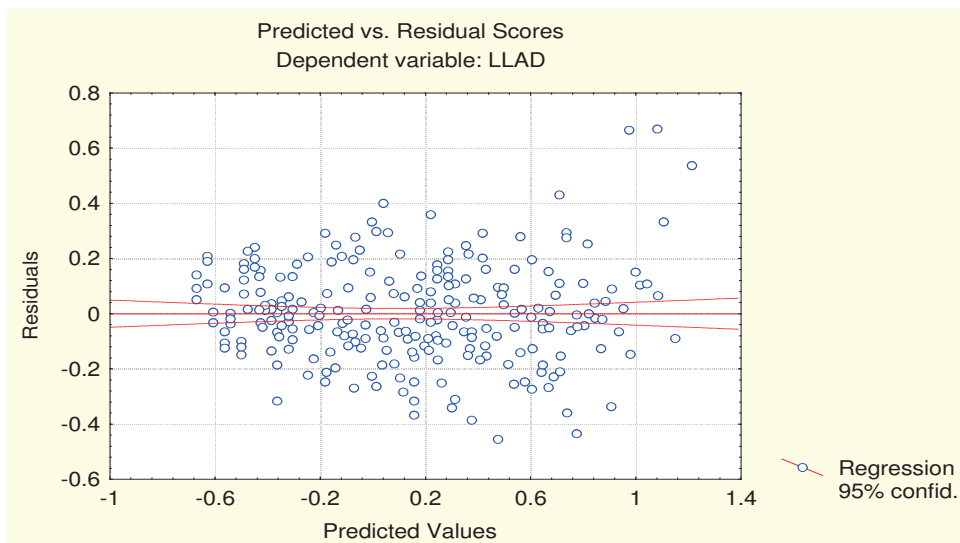
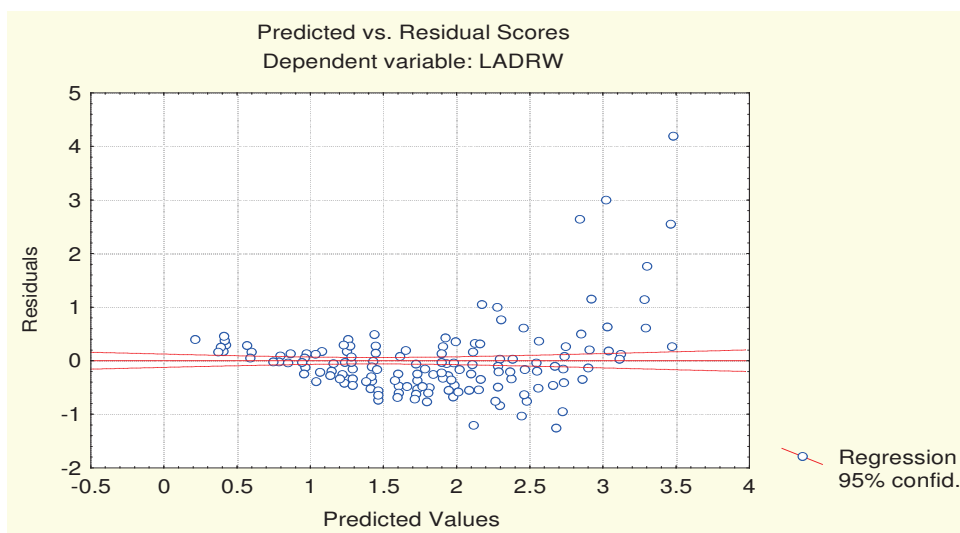
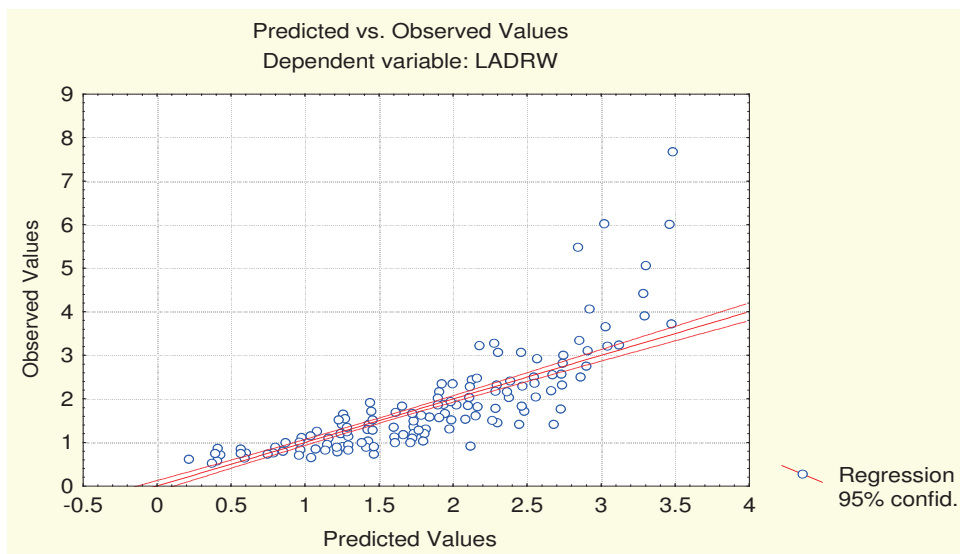
Table 23. Effect of Decision Variables on Logarithm of Relative (to OLS) Norm of LAD Estimator						
Statistic	Regression Coefficient	Std. Error	Beta	Std. Error of Beta	(291) value	Sig (prob)
Variable						
N	-0.5038	0.03136	-0.5567	0.03465	-16.067	0.00000
C	0.16496	0.03136	0.18229	0.03465	5.261	0.00000
G	-0.489	0.03136	-0.5404	0.03465	-15.595	0.00000
B1	-0.6658	0.03136	-0.7357	0.03465	-21.233	0.00000
B2	0.04506	0.03136	0.04979	0.03465	1.437	0.15178
NOBSERV	0.06016	0.03023	0.04205	0.02113	1.99	0.04754
COEFF	0.28107	0.05169	0.03215	0.00591	5.437	0.00000
NOUT	0.28568	0.03455	0.05336	0.00645	8.269	0.00000
OSIZE	-0.5038	0.03136	-0.5567	0.03465	-16.067	0.00000
Linear Regression without the constant term;				($R^2 = 0.88891$, $F(9,291) = 258.72$)		

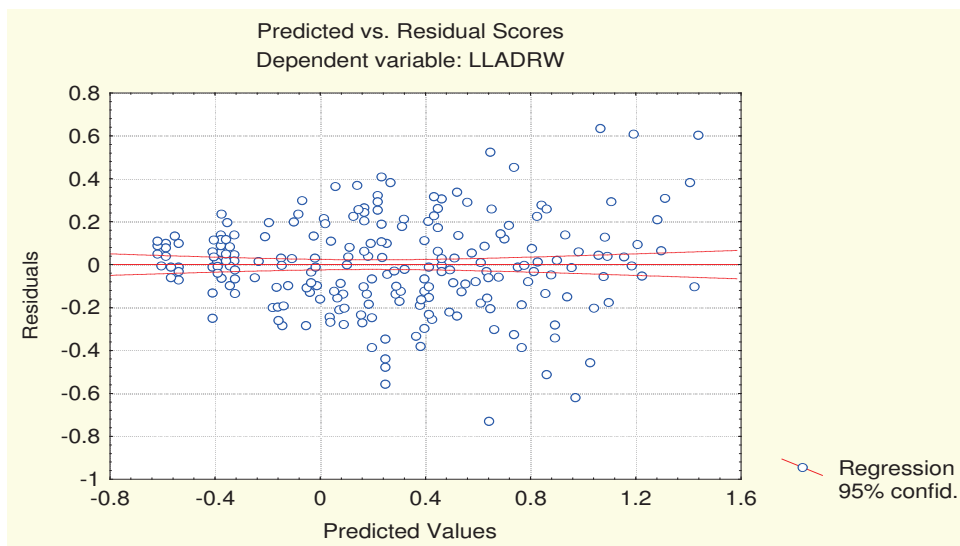
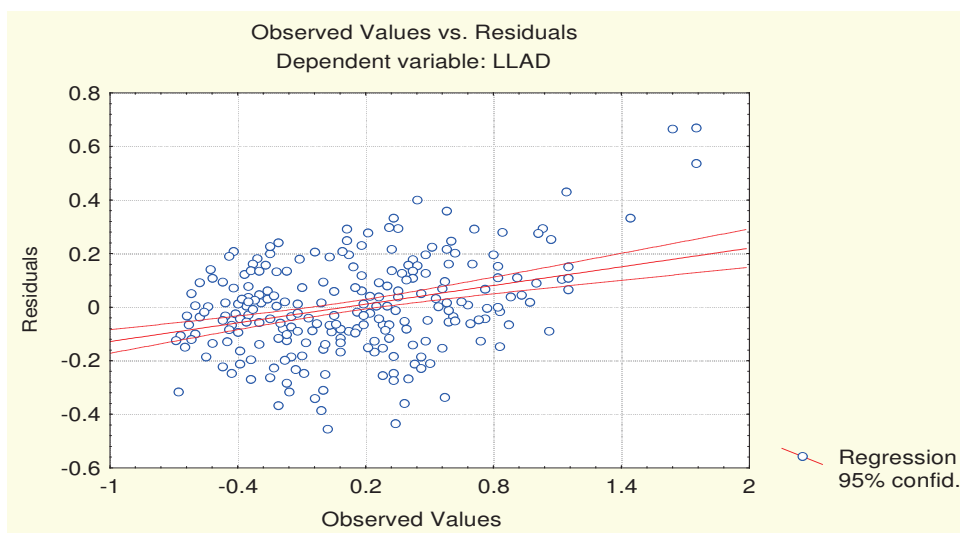
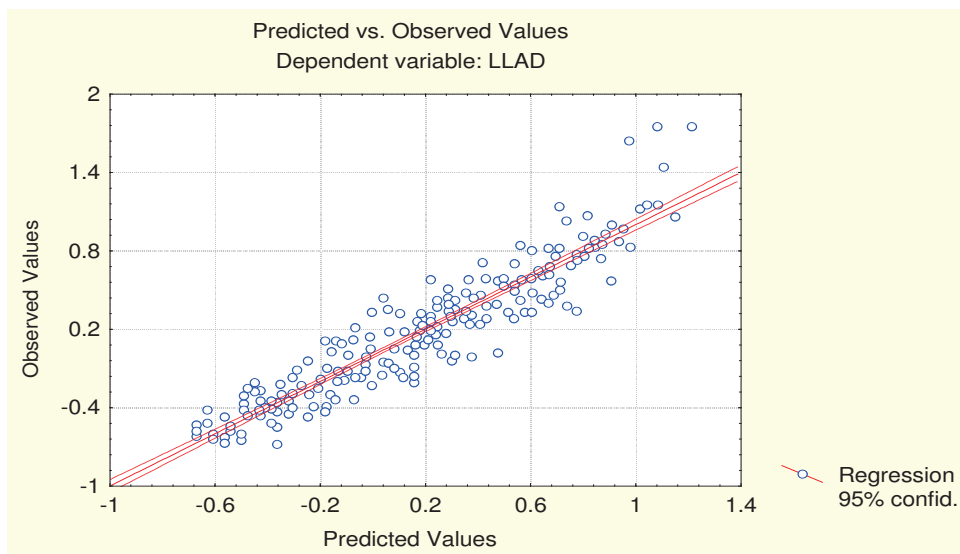
**Table 24. Effect of Decision Variables
on Logarithm of Relative (to OLS) Norm of LADRW Estimator**

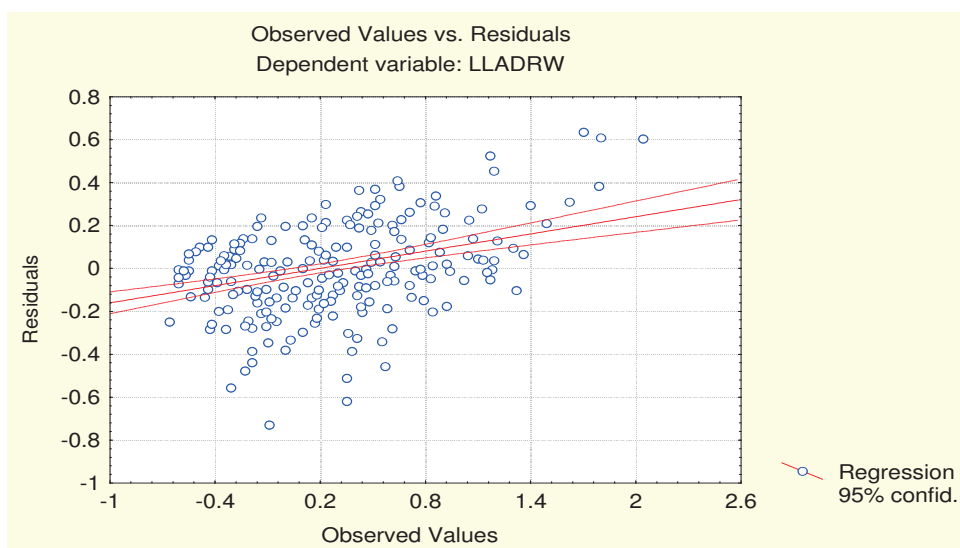
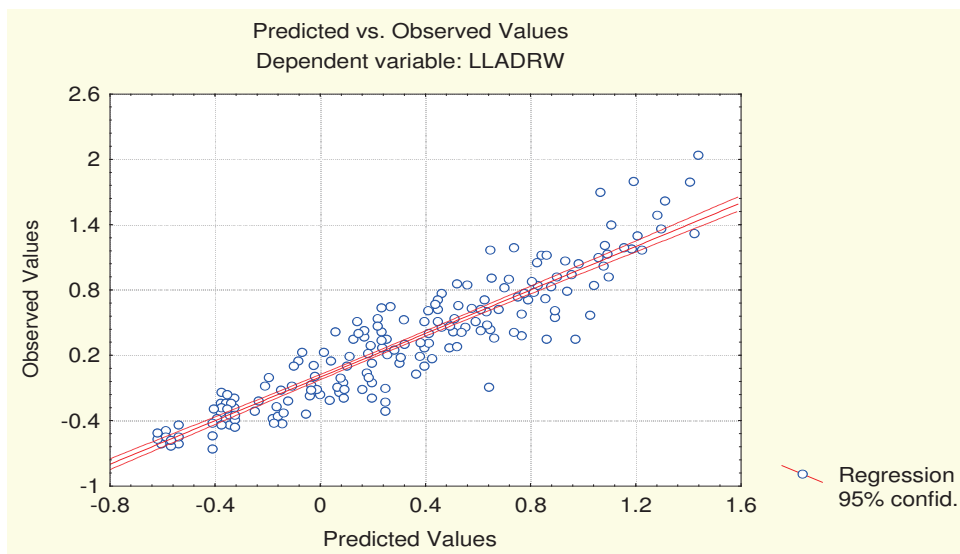
Statistic	Regression Coefficient	Std. Error	Beta	Std. Error of Beta	(291) value	Sig (prob)
Variable						
N	-0.28281	0.03228	-0.36344	0.04149	-8.760	0.00000
C	0.33220	0.03228	0.42690	0.04149	10.290	0.00000
G	-0.28022	0.03228	-0.36010	0.04149	-8.680	0.00000
B1	-0.44623	0.03228	-0.57344	0.04149	-13.822	0.00000
B2	0.16489	0.03228	0.21190	0.04149	5.108	0.00000
NOBSERV	0.06202	0.03113	0.05041	0.02530	1.992	0.04726
COEFF	-0.05770	0.05323	-0.00768	0.00708	-1.084	0.27923
NOUT	0.29006	0.03557	0.06301	0.00773	8.154	0.00000
OSIZE	-0.28281	0.03228	-0.36344	0.04149	-8.760	0.00000
Linear Regression without the constant term; $(R^2 = 0.88223, F(9,291) = 242.22)$						











e). Probit Analysis: In the preceding sections we have made an attempt to look into the performance of the candidate estimators in accordance with single criterion and double criteria. We also attempted to explain the relative norms of the candidate estimators by the decision variables (specification of model). Now we make an attempt to apply Probit Analysis to assess the effect of different criteria on the performance of estimators.

We have two dependent variables, each binary in nature, representing the success or failure (1 or 0) of the candidate estimator (LAD as well as LADRW) in outperforming OLS. On the other hand, specifications may be used as the independent variables that explain the variations in the dependent variables. Accordingly, we have used the following independent (explanatory) variables that represent the model specifications:

N (Normal Distribution): A binary variable, taking on a value of unity if the disturbance term follows a normal distribution, zero otherwise.

G (Gamma Distribution): A binary variable, taking on a value of unity if the disturbance term follows a Gamma distribution, zero otherwise.

B₁ (Beta₁ Distribution): A binary variable, taking on a value of unity if the disturbance term follows a Beta₁ distribution, zero otherwise.

B₂ (Beta₂ Distribution): A binary variable, taking on a value of unity if the disturbance term follows a B₂ distribution, zero otherwise.

NOBSERV (Sample size): A binary variable, taking on a value of unity if the number of observations has a larger size = 50, zero otherwise.

COEFF (Size of the model): This variable takes on the value of 2 for a model with two explanatory variables, 4 for a model of four explanatory variables and 6 for a model with 6 explanatory variables.

NOOUT (No. of Outliers): This variable takes on the value of 1 if the disturbance contains one outlier, 3 if the disturbance contains three outliers, 5 if the disturbance contains five outliers, and 0 if the disturbance contains no outlier.

OSIZE (Outliers of a large size): To recapitulate (what has been mentioned earlier), in the experiments we have used four different factors (call k) for generating outliers of

different sizes. For small samples we use two values of k (0.5 and 1.0). For large samples we use $k=1.0$ and $k= 2.0$. The size of outliers is $k*\text{mean}(y)$ or k times the mean of the dependent variable. The specified number of outliers are added to different observations of the dependent variable (y) at randomly selected locations (cases or observation). The value of the decision variable $OSIZE$ takes on these values of k . Thus $OSIZE$ takes on three different values – 0.5, 1.0 and 2.0 depending on the specified conditions.

Binary coding of distribution variables necessitate that any four (out of 5 namely, N , C , G , B_1 and B_2) of them at most can at a time be included in the list of explanatory variables, since all five, if included, make up for a perfect multi-collinearity. In view of the spectacular performance of LAD and LADRW for Cauchy distributed disturbance term (exhibiting 59 and 60 successes respectively out of 60 experiments) we have chosen to drop out Cauchy from the list of explanatory variables. Accordingly, our general Probit Regression model may be described as follows:

$$R_i = k_0 + a_1N + a_2G + a_3B_1 + a_4B_2 + b_1NOBSRV + b_2COEFF + c_1NOUT + c_2OSIZE + error$$

where R_i is a binary measure (taking on a numerical value of unity or zero) of the success/failure of one of the estimators, namely LAD or LADRW.

Table 25. BLAD Probit regression; **(No of 0's:139 No. of 1's:161: R=0.62236)**

	Const	N	G	B2	NOBSRV	COEFF	NOUT	OSIZE
Estimate	-0.7442	-0.9511	-0.7797	1.0905	0.0684	0.0575	0.1326	0.8837
Std.Err.	0.2699	0.2415	0.2295	0.2651	0.1750	0.0504	0.0557	0.1981
t(292)	-2.76	-3.94	-3.40	4.11	0.39	1.14	2.38	4.46
p-level	0.006	0.000	0.001	0.000	0.696	0.255	0.018	0.000

Table 26. BLADRW Probit regression; **(No of 0's:139 No. of 1's:161: R=0.62692)**

	Const	N	G	B2	NOBSRV	COEFF	NOUT	OSIZE
Estimate	-0.3368	-0.8693	-0.8114	0.9751	0.0266	-0.0176	0.1980	0.8430
Std.Err.	0.2551	0.2448	0.2403	0.2790	0.1710	0.0513	0.0637	0.1949
t(292)	-1.32	-3.55	-3.38	3.49	0.16	-0.34	3.11	4.32
p-level	0.188	0.000	0.001	0.001	0.876	0.732	0.002	0.000

Table 27. BLAD Probit regression; (No of 0's:139 No. of 1's:161: R=0.68728)								
	Const	G	B1	B2	NOBSRV	COEFF	NOUT	OSIZE
Estimate	-0.6871	-0.9962	-1.7798	0.9574	0.0430	0.0747	0.1781	0.8717
Std.Err.	0.2574	0.2217	0.2885	0.2437	0.1694	0.0497	0.0568	0.1707
t(292)	-2.67	-4.49	-6.17	3.93	0.25	1.50	3.13	5.11
p-level	0.008	0.000	0.000	0.000	0.800	0.134	0.002	0.000

Table 28. BLAD Probit regression; (No of 0's:139 No. of 1's:161: R=0.68417)						
	Const	G	B1	B2	NOUT	OSIZE
Estimate	-0.36403	-0.97404	-1.74530	1.01170	0.16984	0.87214
Std.Err.	0.13320	0.21575	0.27821	0.25773	0.05404	0.16102
t(294)	-2.73	-4.51	-6.27	3.93	3.14	5.42
p-level	0.007	0.000	0.000	0.000	0.002	0.000

Table 29. BLADRW Probit regression (No of 0's:127 No. of 1's:173: R=0.70076)								
	Const	G	B1	B2	NOBSRV	COEFF	NOUT	OSIZE
Estimate	-0.17200	-1.27654	-2.00777	0.77401	-0.06756	-0.00513	0.26018	1.02743
Std.Err.	0.24452	0.31956	0.39401	0.25780	0.16956	0.05177	0.07238	0.21530
t(292)	-0.70	-3.99	-5.10	3.00	-0.40	-0.10	3.59	4.77
p-level	0.482	0.000	0.000	0.003	0.691	0.921	0.000	0.000

Table 30. COMPOSIT Probit regression (No of 0's:125 No. of 1's:175: R=0.71308)								
	Const	G	B1	B2	NOBSRV	COEFF	NOUT	OSIZE
Estimate	-0.09252	-1.26298	-1.99899	0.96154	-0.02105	-0.03185	0.25933	1.02199
Std.Err.	0.24551	0.30450	0.37634	0.28313	0.16922	0.05215	0.07017	0.20659
t(292)	-0.38	-4.15	-5.31	3.40	-0.12	-0.61	3.70	4.95
p-level	0.707	0.000	0.000	0.001	0.901	0.542	0.000	0.000

Table 31. BLADRW Probit regression (No of 0's:127 No. of 1's:173: R=0.70055)					
	Const	G	B1	B2	NOUT
Estimate	-0.22111	-1.27045	-2.00753	0.76828	0.26540
Std.Err.	0.13276	0.30780	0.38318	0.24641	0.07077
t(294)	-1.67	-4.13	-5.24	3.12	3.75
p-level	0.097	0.000	0.000	0.002	0.000

Table 32. COMPOSIT Probit regression (No of 0's:125 No. of 1's:175: R=0.71261)					
	Const	G	B1	B2	NOUT
Estimate	-0.22146	-1.28923	-2.03286	0.91328	0.26672
Std.Err.	0.13102	0.31051	0.38648	0.25741	0.07084
t(294)	-1.69	-4.15	-5.26	3.55	3.77
p-level	0.092	0.000	0.000	0.000	0.000

The results of Probit Analysis are tabulated in the tables, 25 through 32. It is especially to mention that inclusion of N (normal disturbance) and B₁ (Beta₁ disturbance) together as the explanatory variables in the Probit regression makes the estimation procedure unstable due to ill conditioned variance-covariance matrix of explanatory

variables. Hence, only one of them (at a time) may be included in the list of explanatory variables. Accordingly, we find that N , B_1 and G have depressing effect on LAD and LADRW estimators. B_2 gives an edge to LAD (as well as LADRW) estimator over OLS estimator. Presence and size of outliers favour LAD (as well as LADRW) estimator.

DasGupta, Madhuchhanda (2004) LEAST ABSOLUTE
DEVIATION ESTIMATION OF MULTI-EQUATION LINEAR
ECONOMETRIC MODELS : *A STUDY BASED ON MONTE
CARLO EXPERIMENTS*, Doctoral Dissertation, Dept. of
Economics, North Eastern Hill University, Shillong (India)
Supervisor : Prof. SK Mishra

CHAPTER 6

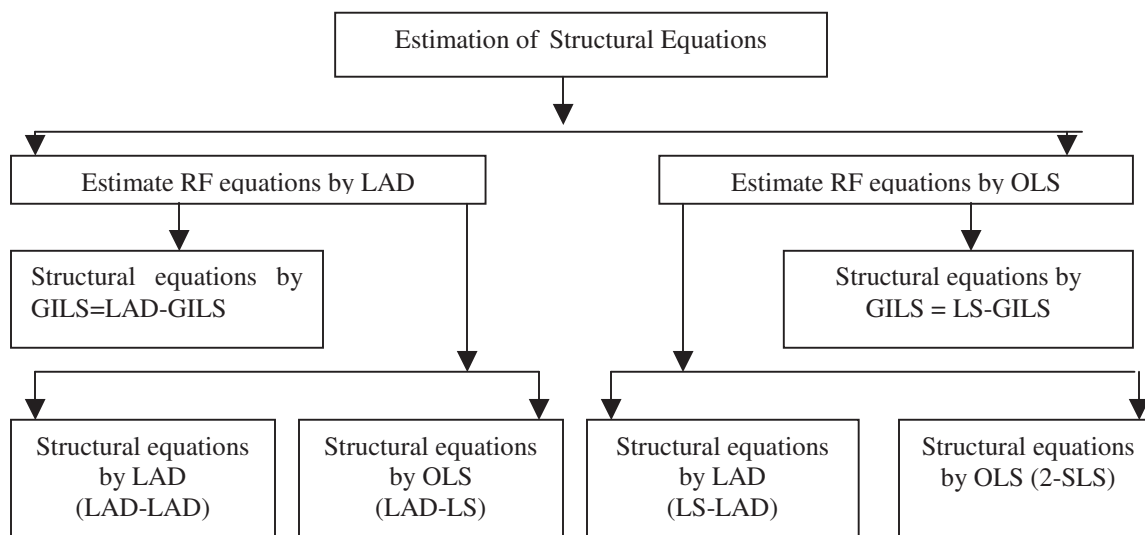
PERFORMANCE OF LAD IN ESTIMATION OF MULTI-EQUATION LINEAR MODELS

I. Introduction: We have noted that the LAD estimator performs better than OLS if the errors in a regression equation are either quite skewed or they are infested with outliers. The 2-SLS and the GILS both obtain the (estimated) reduced form coefficient matrix, P , by OLS. Now, if Y contains errors that are quite skewed or if the errors contain outliers, OLS cannot be expected to perform well while the LAD estimator may work better. Then, is it advisable to use the LAD estimator for obtaining P ? We know that in the second stage, 2-SLS uses the Least Squares method once again. Is it advisable to use the LAD estimator once again at this stage? In short, is 2-Stage LAD estimation advisable? Similarly, at the second stage of GILS (solving $[Q_{21}] [\hat{a}_r]_1 + Q_{22}[\hat{a}_r]_2 = [\hat{b}_r]_2$), is it advisable to use the LAD estimator, in case the structural equation concerned is over-identified? Should we benefit by using *Indirect LAD estimator* (in case of exact identification) and *2-stage LAD estimator* (in case of over-identification)? These questions motivate us to seek their answers.

It is pertinent to look into the possibility of estimating reduced form equations by LAD vis-à-vis OLS more closely. In the reduced form equations ($Y = X\Pi + U$), the errors are $U = -EA^{-1}$. Now, suppose that vectors of E (error vectors in different structural equations, say E_r for the r^{th} equation; $r = 1, 2, \dots, m$) are either skewed or contain outliers of unspecified numbers and magnitudes, is it not likely that U_r , being the weighted sum, or the linear combination, of E_r (that is, $\Pi = -EA^{-1}$), will tend to be normally distributed

due to a consequence of the *Central Limit theorem*? This would be more likely if A is larger (that is, the number of structural equations in the model is larger). This possibility weakens the case of LAD against OLS in estimating the reduced form equations. Nevertheless, it depends on the effectiveness of the Central limit theorem in making U near-normal. In the case of small models, this possibility is only remote.

We know that all k -class estimators are instrumental variable estimators with different values of k . Especially, the 2-SLS estimator is a k -class estimator with $k = 1$, in which explanatory endogenous variables (in any particular equation of the multi-equation model) are replaced by their expected counterparts. Symbolically, if (say) in the r^{th} equation y_r is the dependent variable and y_s ($y_s \in Y$; $r \neq s$) is one of the (endogenous) explanatory variables, then y_s is replaced by $\hat{y}_s = y_s - k\hat{u}_s$, where $k = 1$ and \hat{u}_s is the estimated reduced form residual obtained by considering y_s as the dependent variable, all exogenous variables in the model (that is X) as explanatory variables and running LS estimation procedure on the equation $y_s = Xp + u_s$. Other estimators such as OLS and Limited Information Max. Likelihood (LIML) use different values of k ; for OLS $k = 0$ while $k = \lambda$ for LIML. The basic requirement for an instrumental variable estimation are two-fold: first that the instrumental variable (\hat{y}_s in our illustration) and the original variable (y_s) should be highly correlated (the more the better, ideally = 1) and the error in the instrumental variable (true $y_s + e_s - \hat{y}_s = \epsilon_s$, say, the error in \hat{y}_s) and the error in the original variable (e_s in $y = \text{true } y_s + e_s$) should be poorly correlated (the near-zero the better, ideally = 0). If errors across the equations are orthogonal, the scheme of choosing instrumental variable adopted by 2-SLS is ideal.



In view of the requirements to choose appropriate instrumental variables, it is natural to think whether one may consider \hat{y}_s estimated by means of the LAD estimator instead of \hat{y}_s estimated by the OLS estimator. In case the former (LAD-estimated \hat{y}_s) is more strongly correlated with y_s than the latter (OLS-estimated \hat{y}_s) and the permeant error ε_s in LAD-estimated \hat{y}_s is more weakly correlated with e_s (vis-à-vis ε_s in OLS-estimated \hat{y}_s) it would perhaps be more appropriate to use LAD-estimated \hat{y}_s as the instrumental variable replacing y_s . In real life situation, one does not know e_s as well as ε_s . However, in a Monte Carlo experiment one knows them. Therefore, it is pertinent to investigate and compare LAD in obtaining e_s , ε_s and \hat{y}_s with OLS in obtaining their counterparts while errors are non-normal or contain outliers. The tree of investigation may proceed along the scheme depicted in the figure above.

II. The Steps in Monte Carlo Experiments: In an attempt to answer the questions raised above, we describe the step-wise procedure of Monte Carlo experiments. The steps in the experiment are as follows:

1. Choose the size of a model, that is, choose m and k or the number of structural equations (= number of endogenous variables = m) and the number of pre-determined variables (k) in the model.

2. Choose the structure of the model, that is, decide as to the appearance (or absence) of different endogenous and pre-determined variables in each structural equation of the model. Naturally, in the r^{th} structural equation, the r^{th} endogenous variable would appear with a known coefficient = -1 and s^{th} endogenous variable would appear with an unknown coefficient ($s \neq r$; $s = 1, 2, \dots, \mu_r$), while absent endogenous variables ($m-1-\mu_r$ in number) would have zero coefficients. Similarly, the present pre-determined variables (k_r in number) would appear with unknown coefficients and the absent ones ($k-k_r$ in number) would have zero coefficients. This amounts to first deciding on two m -element vectors, say, $\mu(m)$ and $K(m)$ such that any element of $\mu(m) = 0 \leq \mu_r \leq m-1$ and that of $K(m) = 0 \leq k_r \leq k$. Then, decide the structures of $\alpha(m, m)$ and $\beta(k, m)$ such that (i) $\alpha_{ii} = -1 \ \forall \ i = j$,

$$\alpha_{ij} = \text{either } 0 \text{ or } 1 \ \forall \ i \neq j; \ \beta_{ij} = \text{either } 0 \text{ or } 1 \ \forall \ i, j \text{ and (ii) } \sum_{i=1, i \neq j}^m \alpha_{ij} = \mu_r.$$

3. Choose the values of unknown structural coefficients: In each structural equation, some endogenous and pre-determined variables appear with unknown coefficients. We assign some numerical value to these coefficients. Having done so, our parametric A and B are now fixed. Call them A and B respectively. Obey the restriction such that A is not singular. If A is singular, change some coefficients to make it non-singular.

4. Find $P = -A^{-1}B$, or the reduced form coefficient matrix.
5. Decide the number of samples (n) to be drawn.
6. Make the matrix $X(n,k)$ by assigning some numerical values to x_{ij} , $i = 1, n$ and $j = 1, k-1$, and $x_{ik} = 1 \forall n$. That is, the k^{th} column of X is a unit vector.
7. Find $Y = XP$.

By now, we have known the population Y (that is Y) for a fixed X . What we have fixed are: A and B (structural parameters), P (reduced form parameters derived from the structural parameters), n (the sample size), X (fixed matrix of exogenous variables) and Y (fixed matrix of non-stochastic part of endogenous variables derived from X and P). Our next step would be to generate a matrix $R(n,m)$ of random numbers with a specific probability distribution and the specific parameters of that particular probability distribution.

8. Specify a probability distribution and its parameters
9. Generate a matrix $R(n,m)$ of the specified probability distribution and its parameters.
10. Find $Y(n,m) = Y(n,m) + R(n,m)$. Now Y is stochastic.
11. Proceed.
12. Apply 2-SLS to obtain the estimated A , that is \hat{A} and estimated B , that is \hat{B} . Store them in a file, say AB2SLS, and keep it open.
13. Apply GILS to obtain the estimated A , that is \hat{A} and estimated B , that is \hat{B} . Store them in a file, say ABGILS, and keep it open.
14. Apply LAD to obtain the estimated A , that is \hat{A} and estimated B , that is \hat{B} . Store them in a file, say ABLAD, and keep it open.

We have conducted the first experiment on estimating the matrices of structural coefficients by three methods and stored the results in three different files. Now, we have to conduct several such experiments. Let us specify the number of experiments to conduct, say, 100. So we repeat steps 9 through 14. We have already conducted the first experiment, Additionally, we conduct 99 more experiments so that the total number of experiments = 100. All these results are sequentially stored in the respective files. In each file, AB2SLS, ABGILS and ABLAD we have the couplets of estimated A and estimated B hundreds in number. We proceed.

15. Close the files AB2SLS, ABGILS and ABLAD. Then open the files for reading.

16. Find Root Mean Square (RMS) for each equation in each file, defined as:

$$\text{RMS}_{\text{rf}} = \left[\frac{1}{100} \sum_{i=1}^{100} \left[\sum_{j=1}^{m_r} (\hat{A}_{ij} - A_j)^2 + \sum_{j=1}^{k_r} (\hat{B}_{ij} - B_j)^2 \right] / (m_r + k_r) \right]^{0.5}$$

for $f = 1, 2, 3$ and $r = 1, 2, \dots, m$. In the expression above, 100 appears as the number of experiments conducted. The RMS vector is computed for each of the three files, $f = 1, 2, 3$. So, we have RMS_1 , RMS_2 and RMS_3 . The estimator (2-SLS, GILS or LAD) with the smallest RMS_r (for a particular equation, r) is more efficient.

17. Find Root Mean Square of Bias as follows:

$$\text{RMSB}_{\text{rf}} = \left[\left\{ \sum_{j=1}^{m_r} \left(\frac{\sum_{i=1}^{100} \hat{A}_{ij}}{100} - A_j \right)^2 + \sum_{j=1}^{k_r} \left(\frac{\sum_{i=1}^{100} \hat{B}_{ij}}{100} - B_j \right)^2 \right\} / (m_r + k_r) \right]^{0.5}$$

for $f = 1, 2, 3$ and $r = 1, 2, \dots, m$. In the expression above, 100 appears as the number of experiments conducted. The RMSB vector is computed for each of the three files, $f = 1, 2, 3$. So, we have RMSB_1 , RMSB_2 and RMSB_3 . The estimator (2-SLS, GILS or LAD) with the smallest RMSB_r (for a particular equation, r) is the least biased.

18. Run steps 8 through 17 for different probability distributions and for different parameter specifications to observe if distribution of errors and specification of parameters of a distribution matters.

19. Look at the step 11. It is a blank proceed. It may be replaced by a process that introduces outliers (very large errors in comparison with other elements in the error vectors) to the error vector. One may decide on the number of outliers and their size. These outliers may be introduced into the observations selected randomly.

20. With the introduction of outliers run steps 8 through 17 to examine whether presence, number, and sizes of outliers affect efficiency and bias of the estimators.

The above experiment may be conducted several times to ascertain the power of the estimators.

III. Candidate Methods of Estimation (Candidate Estimators) for Monte Carlo

Experiments: For this study we have considered six different methods of estimation: (i) Two-Stage Least Squares (**2-SLS** or LS-LS estimator), which estimates the reduced form coefficients (in each equation of the simultaneous-equations models) by OLS (ordinary least squares) in the first stage, obtains the estimated vectors of endogenous variables (\hat{Y}), replaces the stochastic endogenous variables (Y) appearing as explanatory variables in different equations by the estimated vectors of endogenous variables (\hat{Y}) and in the second stage once again applies OLS to estimate the structural coefficients in which explanatory Y have been replaced by the expected \hat{Y} ; (ii) **LS-LAD**, which estimates reduced form coefficients by OLS but in the second stage estimates the structural coefficients (having replaced explanatory Y by \hat{Y}) by the Least Absolute Deviation (LAD) estimator; (iii) **LAD-LS**, which applies LAD estimator at the first stage to

estimate the reduced form coefficients, but at the second stage (having replaced explanatory Y by \hat{Y}) applies OLS to estimate the structural coefficients; (iv) **LAD-LAD**, which applies LAD estimator at the first stage to estimate reduced form coefficients and (having replaced explanatory Y by \hat{Y}) once again applies LAD estimator to estimate the structural coefficients; (v) **LS-GILN**, which estimates reduced form coefficients by OLS and applies Generalized Indirect Least Norm estimation (GILN) on the matrix of reduced form coefficients (Π) to obtain the structural coefficients; and lastly, (vi) **LAD-GILN**, which applies LAD estimator to obtain the matrix of reduced form coefficients (Π) and thereupon GILN is applied to obtain the structural coefficients. All of these methods of estimation are equation-by-equation estimation (or limited information estimation) methods.

IV. Specification of Simultaneous-Equation Model: A simultaneous-equation model is specific in the sense of (i) size or the number of equations in the model, (ii) structure (appearance/absence of various endogenous and pre-determined variables in different equations) of the model, (iii) distribution (or probability density function) of disturbance vector in different equations of the model including the magnitude of variance of the disturbance vector, (iv) presence (number and size) of outliers in the disturbance vector and lastly (v) the sample size or the number of cases (or no. of observations). Specification about linearity or otherwise also is important. However, in our experiments we have considered linear models only. Other particulars of the specification are:

a). Size of Simultaneous-Equations Model: In our endeavor to ascertain the efficiency of the LAD estimator, Monte Carlo experiments have been conducted using three models of different sizes.

(i) **Model 1:** Model#1 comprises of 3 equations and 5 pre-determined variables, the last of which is a constant. The model consists of 10 parameters, which are to be estimated. The structure of Model #1 is:

Structure of Simultaneous-Equations Model # 1								
VARIABLES	ENDOGENOUS			PRE-DETERMINED				
EQUATIONS	Y1	Y2	Y3	X1	X2	X3	X4	X5
Eqn1	-1	0	1	1	0	0	0	1
Eqn2	1	-1	0	0	1	0	0	1
Eqn3	0	1	-1	0	0	1	1	1

(ii) **Model 2:** Model#2 is a 5-equation model with 7 pre-determined variables of which the last variable is a constant. The model requires an estimation of 24 parameters in all. The model #2 is:

Structure of Simultaneous-Equations Model # 2												
VARIABLES	ENDOGENOUS					PRE-DETERMINED						
EQUATIONS	Y1	Y2	Y3	Y4	Y5	X1	X2	X3	X4	X5	X6	X7
Eqn1	-1	0	1	1	0	1	0	0	0	1	0	1
Eqn2	1	-1	0	0	1	0	1	0	0	0	0	1
Eqn3	1	1	-1	0	0	1	0	1	0	0	0	1
Eqn4	1	0	0	-1	0	0	0	1	1	1	0	1
Eqn5	1	1	0	0	-1	0	0	1	0	0	1	1

(iii) **Model 3:** Model#3 is composed of 7 equations in 10 pre-determined variables. The last variable is a constant. There are 54 parameters in the model, which are to be estimated. The structure of the model#3 is:

Structure of Simultaneous-Equations Model # 3																	
Vari-ables	ENDOGENOUS							PRE-DETERMINED									
Equa-tions	Y1	Y2	Y3	Y4	Y5	Y6	Y7	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10
Eqn 1	-1	0	0	1	0	1	1	1	0	0	1	0	0	0	1	1	1
Eqn 2	1	-1	1	0	1	0	0	0	1	0	0	0	1	1	0	0	1
Eqn 3	1	0	-1	1	0	1	1	0	0	1	0	0	1	0	0	0	1
Eqn 4	1	0	0	-1	1	1	1	1	0	0	1	1	0	0	1	0	1
Eqn 5	1	1	0	0	-1	0	0	1	1	0	0	0	0	1	0	1	1
Eqn 6	1	0	0	1	1	-1	1	0	1	1	0	1	1	0	0	0	1
Eqn 7	1	1	1	0	0	0	-1	0	0	0	0	1	0	1	1	0	1

In the above models, the Y's represent the endogenous variables and the X's represent the pre-determined variables.

b). Sample Size: Samples of two different sizes have been taken in the study

(i) Sample of size (no. of cases or observations) = 20.

(ii) Sample of size (no. of cases or observations) = 50.

c). Distribution of Disturbance term: To study the effect of the probability distribution of the error term on the efficiency of the six estimators considered in the study, namely, 2SLS, LS-LAD, LAD-LS, LAD-LAD, LS-GILN and LAD-GILN, we have considered five probability distributions. They are (i) Normal, (ii) Cauchy, (iii) Gamma, (iv) Beta₁, and (v) Beta₂. Gamma, Beta₁ and Beta₂ distributed errors have been generated with parameter of size =2.

d). Standard Deviation of the Disturbance term: The standard deviation of error used in the study is of two sizes namely: (i) 0.001 and (ii) 0.3.

e). Number and size of Outliers: Outliers of different number and magnitude have also been introduced to the error vector to examine whether their presence, number and size affect the efficiency and bias of the estimator under study.

(i) The numbers of outliers used in the study are 1, 2 and 3.

(ii) The magnitudes of outliers used are 0.5, 1 and 1.5 times the mean value of the dependent variable (\bar{Y}_i) in a particular model.

V. Numerical Details: Monte Carlo experiments are numerical in nature. Moreover, it is practically impossible to conduct such experiments manually and hence computer must be used. Once models are specified, random numbers are generated in accordance with the specifications and the parameters are estimated by the candidate methods of

estimation. Necessary information regarding the experiments conducted in this study are given below.

1. We have generated two uniformly distributed random numbers lying between zero and unity (0,1). With the first random number, the structural coefficient matrices A and B have been formed. From the second random number, the matrix of exogenous variables, X, has been formed. The entries in A, B and X are to begin with lie between 0 and 1, in accordance with the structure of the model. To scale up the magnitude of the numerical values constituting the matrices X, A and B, we have multiplied the matrices X, A and B by the scalars 10 , 1 and 30 respectively. Appropriate scaling is needed to keep up accuracy in the numerical analysis.
2. The accuracy level of the LAD estimates has been fixed at 0.0001.
3. The tolerance level for accuracy which is required for convergence of parameter estimates has been fixed at 0.0001.
4. To obtain the LAD estimates, 100 iterations have been performed. It is assumed that such a number of iteration is sufficient of an accurate estimation.
5. The experiment conducted has been replicated 50 times – each time drawing fresh samples of random disturbance vectors (while keeping the structure, the exogenous and the endogenous variables fixed), adding them to the endogenous variables and estimating the parameters by the different methods of estimation (2-SLS, LS-LAD, LAD-LS, LAD-LAD, LS-GILN and LAD-GILN).

The step-wise procedure may be described as follows:

- (i) Generate two uniformly distributed random numbers (r_1 and r_2). With r_1 generate the matrices of structural coefficients, $A[m,m]$ and $B[k,m]$, and with r_2 generate the matrix

of pre-determined variables, $X[n,k]$. Here m is the number of equations (= no. of endogenous variables) in the model, k is the number of pre-determined variables in the model and n is the number of cases (observations or sample size).

(ii) Generate $Y = X\Pi = -XBA^{-1}$. The endogenous variables ($Y[n,m]$) thus generated are non-stochastic. Here $\Pi[k,m]$ ($= -BA^{-1}$) is the matrix of Reduced form parameters.

(iii) Using another seed, generate a set of random variables of desired distribution (agreeing with the order of Y matrix) and add them to Y . The endogenous variables thus obtained ($Y[n,m]$) are now stochastic.

(iv) Decide on the number and the sizes of outliers and add them to each J of $Y[i,j]_i$ ($i=1,n$) at randomly chosen locations.

(v) Estimate the parameters in A and B (call them \hat{A} and \hat{B}) by different estimators.

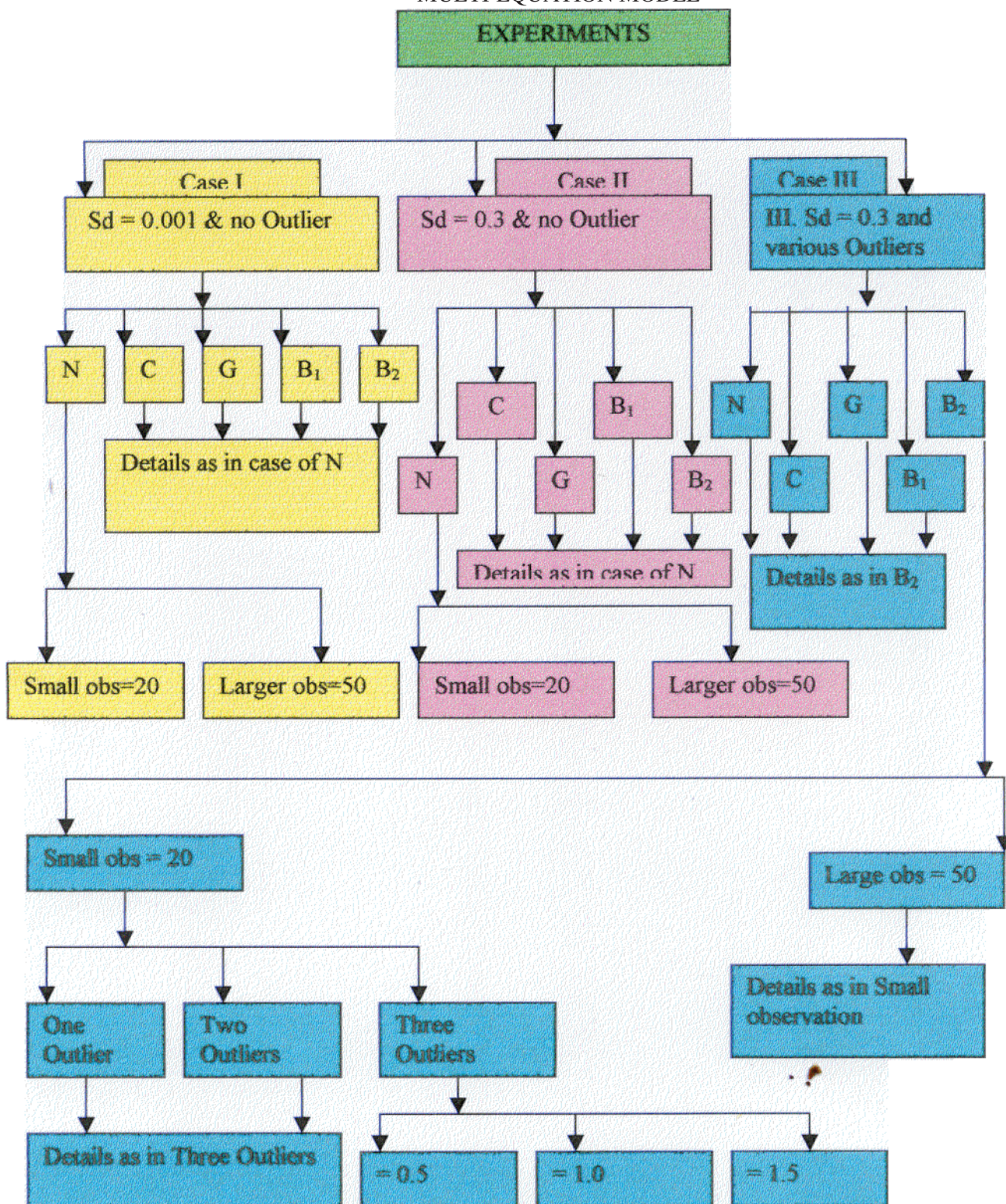
(vi) Repeat steps (iii) to (v) for the desired number of times (50 here) for replication.

(vii) Use the results in (vi) to obtain RMS of deviations of the estimated \hat{A} and \hat{B} from the true A and B , while \hat{A} and \hat{B} are estimated by the different estimators including 2-SLS. Obtain relative RMS (relative to 2-SLS RMS =1) for the different estimators.

Step (5) is repeated for different distributions (of error vectors) with different magnitudes of standard deviations, different numbers and magnitudes of outliers and different model sizes of different sample sizes (no. of cases or observations).

VI. Details of Specification-wise no. of experiments: It is facilitating to describe the details of specification-wise number of experiments conducted in our study. In all, we have conducted 630 experiments (each one with 50 replicates yielding one RMS for each candidate estimator). These details are as follows:

DESIGN OF EXPERIMENTS:
MULTI-EQUATION MODEL



Size of Model: Model#1 (210), Model#2 (210), Model#3 (210)

Sample Size: Small i.e. 20 observation (315), Larger, i.e. 50 observations (315)

Distributions: Normal (126), Cauchy (126), Gamma (126), B1 (126), B2 (126)

Standard Deviation of Disturbance Vector: Small (180), Large (450)

Number of Outliers: None (360), One (90), Two (90), Three (90)

Size of Outliers: None/zero (360), Small (90), Medium (90), Large (90)

VII. Findings of Monte Carlo Experiments: In accordance with the design of Monte Carlo Experiments we have run the simulation program for 630 times with the specifications as given in ‘details of specification-wise no. of experiments’ earlier. The set of computer programs (source codes of computing programs in FORTRAN 77 and a unifying program in the Batch-processing language) is appended. We have converted the relative norms of each candidate estimator into five binary variables (one each for LS-LAD, LAD-LS, LAD-LAD, LS-GILN and LAD-GILN) such that if the relative norm of a candidate estimator is smaller than the 2-SLS norm then the binary variable takes on a value of unity or zero otherwise. In this scheme, the *unity* value of the related binary value signifies *success* (that is, the candidate estimator has succeeded in estimating the parameters of the structural equations more efficiently than does the 2-SLS estimator) while the value of *zero* signifies that 2-SLS estimator dominates (or is as good as) the candidate estimator and it is considered as a *failure*.

a). Single Criterion Frequency Distribution of Success of Different Estimators: First we present the frequency distribution of success of different candidate estimators in dominating 2-SLS as tabulated in the tables 1 through 6. The entries in the cells of these

tables are the number of success (out of the total number of experimental trials or cases - given in the rightmost columns of the tables) and where the ratio of the number of success to the number of trials (cases) is larger than 0.5, the entry has been highlighted.

We observe the following:

Effect of Model Size: For M_3 (7-equations model) LS-LAD and LAD-LAD estimators outperform 2-SLS.

Effect of Sample Size: No estimator outperforms 2-SLS.

Effect of Distribution of Disturbance term: For Cauchy and B_2 (named as B or b) distributed disturbance terms LAD-LAD and LAD-GILN outperform 2-SLS.

Effect of Size of Std. Deviation of Disturbance term: No estimator outperforms 2-SLS.

Effect of No. of Outliers: For 2 and 3 outliers LAD-LAD outperforms 2-SLS.

Effect of Size of Outliers: For medium and large size outliers LAD-LAD estimator outperforms 2-SLS.

Table 1: Effect of Model Size

	Ls_Lad	Lad_Ls	Lad_Lad	LS_Giln	Lad_Giln	Cases
M_1	00	04	47	23	61	210
M_2	12	04	62	40	94	210
M_3	123	100	153	42	78	210

Table 2: Effect of Sample Size

	Ls_Lad	Lad_Ls	Lad_Lad	LS_Giln	Lad_Giln	Cases
S	91	82	145	59	90	315
L	44	26	117	46	143	315

Table 3: Effect of Distribution

	Ls_Lad	Lad_Ls	Lad_Lad	LS_Giln	Lad_Giln	Cases
N	20	25	32	18	13	126
C	41	21	81	22	85	126
G	16	22	29	29	14	126
V	23	19	34	18	29	126
B	35	21	86	18	92	126

Note: V stands for β_1 and b or B stands for B_2 .

Table 4: Effect of Standard Deviation

	Ls_Lad	Lad_Ls	Lad_Lad	LS_Giln	Lad_Giln	Cases
S	35	14	56	31	70	180
L	100	94	206	80	163	450

Table 5: Effect of No. of Outliers

	Ls_Lad	Lad_Ls	Lad_Lad	LS_Giln	Lad_Giln	Cases
E	35	14	56	29	70	180
0	49	41	57	28	52	180
1	20	19	38	05	32	90
2	14	16	50	17	41	90
3	17	18	51	26	38	90

Table 6: Effect of Size of Outliers

	Ls_Lad	Lad_Ls	Lad_Lad	LS_Giln	Lad_Giln	Cases
0	21	12	24	07	26	60
1	12	07	19	06	21	60
2	12	08	21	10	19	60
3	17	09	21	14	22	60
4	11	11	20	11	16	60
5	11	08	18	09	18	60
S	19	19	26	06	29	90
M	16	18	47	14	40	90
B	16	16	66	28	42	90

Entries are the frequencies while relative norms are greater than unity in magnitude.

b). Double Criteria Frequency Distribution of Success (and failures) of Different

Estimators: Next we present the frequency distribution of success (as well as failure) of different candidate estimators in dominating 2-SLS as tabulated in the tables 7 through 21. The leading entries (odd number rows) in the cells of these tables are the number of success (out of the total number of experimental trials or cases) while the trailing entries (even number rows) are the number of failure. Where the number of success is larger than the number of failure (i.e. the probability of success is larger than 0.5), the success entry has been highlighted. In summarizing our findings estimator-wise, we observe the following:

LS-LAD Estimator: It outperforms 2-SLS for - M3 (Small sample), M3 (Cauchy and B_2), M3 (Large SD), M3 (No Outlier), M3 (No Outlier, Small, medium and large size Outliers). Overall, it appears to perform well for M3 in general.

LAD-LS Estimator: It outperforms 2-SLS for – M3 (Small sample), M3 (Normal and Cauchy), M3 (Large SD), M3 (No Outlier), M3 (Small, medium and large size Outliers). Overall, it appears to perform well for M3 in general.

LAD-LAD Estimator: It outperforms 2-SLS for – M3 (Small and Large sample sizes), M2 (B_2), M3 (N,C,G, B_1 , B_2), M3 (Large SD), M3 (No Outlier), M3 (No outlier as well as Small, medium and large size Outliers), Small sample(Cauchy and B_2), Large sample (Cauchy and B_2), Small sample (Large SD), Small sample (with 1/2/3 number of Outliers), Large sample (3 outliers), Small sample (Medium and large outliers), Large sample (large Outliers), Cauchy (Small as well as Large SD), B_2 (Small as well as large SD), Cauchy (No Outlier, One/two/three outliers), B_2 (One/two/three Outliers), Normal (Large size Outlier), Cauchy (No Outlier as well as Small, medium and large size Outliers), Gamma (Large size Outliers), B_1 (Large size Outliers), B_2 (No Outlier, Medium and Large size Outliers), Two/three outliers (Small and Large sample size), One outlier (Large size outlier), Two/three outliers (Medium and as well as Large size outliers), Medium/Large size Outliers (Small as well as large SD). Evidently, large-size model, Cauchy and B_2 distribution of disturbance term favour LAD-LAD estimator to outperform 2-SLS. Further, the number & size of outliers present in the observations add to its efficiency.

LS-GILN Estimator: It outperforms 2-SLS only for three outliers of large size. The performance of this estimator is the worst of all candidate estimators.

LAD-GILN Estimator: It outperforms 2-SLS for – M2(Large sample), M3(large sample), M1/M2/M3(Cauchy and B_2), M2(one or 2 no. of outliers), M2(Small, medium and large size outliers), Small sample(B_2), Large sample(Cauchy and B_2), Large sample(3 no. of outliers), Large sample(Medium and large size outliers), Cauchy(Small as well large SD), B_2 (Small as well as large SD), Cauchy(Two/three outliers), B_2 (No outlier, One/two/three outliers), Cauchy(No outliers, Medium/large outliers), B_2 (No outlier, Small/medium/large outliers), two outliers(Large size outlier). It appears that LAD-GILN is the second best estimator (the best one being the LAD-LAD estimator) and performs well for larger model size, larger sample size, Cauchy and B_2 distributed disturbance term with or without outliers.

It appears that the presence of sporadic outlier disturbances (and their size) in the dependent variable immensely affect the performance of OLS estimator. As a result, application of OLS at either the first or the second stage of estimation mars the performance of the two-stage estimators. LAD-LAD and LAD-GILN estimators apply OLS at no stage. This may be the reason for their better performance. On the other hand, LS-LAD, LAD-LS and LS-GILN estimators apply OLS at the first or the second stage and the outliers present in the dependent variable take their toll.

Table 7: EFFECT OF MODEL SIZE AND SAMPLE SIZE

	LS_LAD		LAD_LS		LAD_LAD		LS-GIL		LAD_GIL	
	S	L	S	L	S	L	S	L	S	L
M1	0	0	2	2	25	22	18	5	25	36
	105	105	103	103	80	83	87	100	80	69
M2	3	9	3	1	28	34	26	14	41	53
	102	96	102	104	77	71	79	91	64	52
M3	88	35	77	23	92	61	15	27	24	54
	17	70	28	82	13	44	90	78	81	51

Table 8a: EFFECT OF MODEL SIZE AND ERROR DISTRIBUTION

	LS_LAD					LAD_LS					LAD_LAD				
	N	C	G	V	B	N	C	G	V	B	N	C	G	V	B
M1	0	0	0	0	0	0	1	0	0	3	2	21	3	2	19
	42	42	42	42	42	42	41	42	42	39	40	21	39	40	23
M2	0	8	0	2	2	0	2	0	1	1	4	19	3	10	26
	42	34	42	40	40	42	40	42	41	41	38	23	39	32	16
M3	20	33	16	21	33	25	18	22	18	17	26	41	23	22	41
	22	9	26	21	9	17	24	20	24	25	16	1	19	20	1

Table 8b: EFFECT OF MODEL SIZE AND ERROR DISTRIBUTION

	LS_GIL					LAD_GIL				
	N	C	G	V	B	N	C	G	V	B
M1	3	5	6	4	5	0	29	2	0	30
	39	37	36	38	37	42	13	40	42	12
M2	7	10	11	6	6	4	29	5	21	35
	35	32	31	36	36	38	13	37	21	7
M3	8	7	12	8	7	9	27	7	8	27
	34	35	30	34	35	33	15	35	34	15

Table 9: EFFECT OF MODEL SIZE AND SD SIZE

	LS_LAD		LAD-LS		LAD-LAD		LS-GIL		LAD-GIL	
	d	D	d	D	d	D	d	D	d	D
M1	0	0	3	1	8	39	10	13	21	40
	60	150	57	149	52	111	50	137	39	110
M2	5	7	2	2	18	44	4	36	19	75
	55	143	58	148	42	106	56	114	41	75
M3	30	93	9	91	30	123	15	27	30	48
	30	57	51	59	30	27	45	123	30	102

Table 10a: EFFECT OF MODEL SIZE AND NO. OF OUTLIERS

	LS_LAD					LAD_LS					LAD_LAD				
	E	0	1	2	3	E	0	1	2	3	E	0	1	2	3
M1	0	0	0	0	0	3	1	0	0	0	8	5	8	11	15
	60	60	30	30	30	57	59	30	30	30	52	55	22	19	15
M2	5	7	0	0	0	2	2	0	0	0	18	16	4	14	10
	55	53	30	30	30	58	58	30	30	30	42	44	26	16	20
M3	30	42	20	14	17	9	38	19	16	18	30	46	26	25	26
	30	18	10	16	13	51	22	11	14	12	30	14	4	5	4

Table 10b: EFFECT OF MODEL SIZE AND NO. OF OUTLIERS

LS_GIL						LAD_GIL					
	E	0	1	2	3		E	0	1	2	3
M1	10	3	0	3	7		21	14	5	9	12
	50	57	30	27	23		39	46	25	21	18
M2	4	10	4	13	9		19	20	18	22	15
	56	50	26	17	21		41	40	12	8	15
M3	15	15	1	1	10		30	18	9	10	11
	45	45	29	29	20		30	42	21	20	19

Table 11a: EFFECT OF MODEL SIZE AND OUTLIER SIZE

LS_LAD										LAD_LS									
	0	1	2	3	4	5	S	M	B		0	1	2	3	4	5	S	M	B
M1	0	0	0	0	0	0	0	0	0		2	0	1	0	0	1	0	0	0
	20	20	20	20	20	20	30	30	30		18	20	19	20	20	19	30	30	30
M2	6	1	0	3	1	1	0	0	0		2	0	0	0	2	0	0	0	0
	14	19	20	17	19	19	30	30	30		18	20	20	20	18	20	30	30	30
M3	15	11	12	14	10	10	19	16	16		8	7	7	9	9	7	19	18	16
	5	9	8	6	10	10	11	14	14		12	13	13	11	11	13	11	12	14

Table 11b: EFFECT OF MODEL SIZE AND OUTLIER SIZE

LAD_LAD										LAD_GIL									
	0	1	2	3	4	5	S	M	B		0	1	2	3	4	5	S	M	B
M1	4	1	2	0	3	3	4	12	18		1	2	3	2	2	3	0	2	8
	16	19	18	20	17	17	26	18	12		19	18	17	18	18	17	30	28	22
M2	5	5	7	6	6	5	0	9	19		2	2	1	3	4	2	5	8	12
	15	15	13	14	14	15	30	21	11		18	18	19	17	16	18	25	22	17
M3	15	13	12	15	11	10	22	26	29		4	2	6	9	5	4	1	4	7
	5	7	8	5	9	10	8	4	1		16	18	14	11	15	16	29	26	23

Table 11c: EFFECT OF MODEL SIZE AND OUTLIER SIZE

LAD_GIL									
	0	1	2	3	4	5	S	M	B
M1	6	5	7	6	5	6	3	11	12
	14	15	13	14	15	14	27	19	18
M2	11	7	5	6	5	5	16	19	20
	9	13	15	14	15	15	14	11	10
M3	9	9	7	10	6	7	10	10	10
	11	11	13	10	14	13	20	20	20

Table 12a: EFFECT OF SAMPLE SIZE AND DISTRIBUTION

LS_LAD						LAD_LS					LAD_LAD				
	N	C	G	V	B	N	C	G	V	B	N	C	G	V	B
S	17	20	16	18	20	17	16	18	15	16	23	<u>32</u>	24	24	<u>42</u>
	46	43	47	45	43	46	47	45	48	47	40	<u>31</u>	39	39	<u>21</u>
L	3	21	0	5	15	8	5	4	4	5	9	<u>49</u>	5	10	<u>44</u>
	60	42	63	58	48	55	58	59	59	58	54	<u>14</u>	58	53	<u>19</u>

Table 12b: EFFECT OF SAMPLE SIZE AND DISTRIBUTION

LS_GIL						LAD_GIL				
	N	C	G	V	B	N	C	G	V	B
S	11	12	18	8	10	8	26	10	14	<u>32</u>
	52	51	45	55	53	55	37	53	49	31
L	7	10	11	10	8	5	<u>59</u>	4	15	<u>60</u>
	56	53	52	53	55	58	4	59	48	3

Table 13: EFFECT OF SAMPLE SIZE AND SD SIZE

LS_LAD		LAD-LS		LAD-LAD		LS-GIL		LAD-GIL	
d	D	d	D	d	D	d	D	d	D
S	20 71	13 69	26 <u>119</u>	22 37	33 57				
	70 154	77 156	64 106	68 188	57 168				
L	15 29	1 25	30 87	7 39	37 106				
	75 196	89 200	60 138	83 186	53 119				

Table 14a: EFFECT OF SAMPLE SIZE AND NO. OF OUTLIERS

LS_LAD						LAD_LS					LAD_LAD				
	E	0	1	2	3	E	0	1	2	3	E	0	1	2	3
S	20	28	14	14	15	13	26	14	15	14	26	37	<u>23</u>	<u>31</u>	<u>28</u>
	70	62	31	31	30	77	64	31	30	31	64	53	22	14	17
L	15	21	6	0	2	1	15	5	1	4	30	30	15	19	<u>23</u>
	75	69	39	45	43	89	75	40	44	41	60	60	30	26	22

Table 14b: EFFECT OF SAMPLE SIZE AND NO. OF OUTLIERS										
LS_GIL						LAD_GIL				
	E	0	1	2	3	E	0	1	2	3
S	22	4	2	16	15	33	11	12	20	14
	68	86	43	29	30	57	79	33	25	31
L	7	24	3	1	11	37	41	20	21	<u>24</u>
	83	66	42	44	34	53	49	25	24	21

Table 15a: EFFECT OF SAMPLE SIZE AND OUTLIER SIZE

LS_LAD										LAD_LS									
	0	1	2	3	4	5	S	M	B		0	1	2	3	4	5	S	M	B
S	11	7	8	11	5	6	15	15	13	7	5	7	7	8	5	15	14	14	
	19	23	22	19	25	24	30	30	32	23	25	23	23	22	25	30	31	31	
L	10	5	4	6	6	5	4	1	3	5	2	1	2	3	3	4	4	2	
	20	25	26	24	24	25	41	44	42	25	28	29	28	27	27	41	41	43	

Table 15b: EFFECT OF SAMPLE SIZE AND OUTLIER SIZE

	LAD_LAD										LS_GIL									
	0	1	2	3	4	5	S	M	B		0	1	2	3	4	5	S	M	B	
S	12	10	12	12	9	8	16	<u>26</u>	<u>40</u>		2	2	6	6	5	5	3	10	20	
	18	20	18	18	21	22	29	19	5		28	28	24	24	25	25	42	35	25	
L	12	9	9	9	11	10	10	21	<u>26</u>		5	4	4	8	6	4	3	4	8	
	18	21	21	21	19	20	35	24	19		25	26	26	22	24	26	42	41	37	

Table 15c: EFFECT OF SAMPLE SIZE AND OUTLIER SIZE

LAD-GIL									
	0	1	2	3	4	5	S	M	B
S	11	7	7	9	3	7	12	16	18
	19	23	23	21	27	23	33	29	27
L	15	14	12	13	13	11	17	<u>24</u>	<u>24</u>
	15	16	18	17	17	19	28	21	21

Table 16: EFFECT OF DISTRIBUTION AND SD SIZE

	LS_LAD		LAD-LS		LAD-LAD		LS-GIL		LAD-GIL	
	d	D	d	D	d	D	d	D	d	D
N	3	17	2	23	3	29	4	14	3	10
	33	73	34	67	33	61	32	76	33	80
C	14	27	3	19	<u>20</u>	<u>61</u>	6	16	<u>28</u>	<u>57</u>
	22	63	33	72	16	29	30	74	8	33
G	3	13	3	19	3	26	7	22	3	11
	33	77	33	71	33	64	29	68	33	79
V	3	20	1	18	4	30	5	13	4	25
	33	70	35	72	32	60	31	77	32	65
B	12	23	5	16	<u>26</u>	<u>60</u>	7	11	<u>32</u>	<u>60</u>
	24	67	31	74	10	30	29	79	4	30

Table 17a: EFFECT OF DISTRIBUTION AND NO. OF OUTLIERS

	LS_LAD					LAD_LS					LAD_LAD				
	E	0	1	2	3	E	0	1	2	3	E	0	1	2	3
N	3	6	5	3	3	2	10	5	4	4	3	7	7	8	7
	33	30	13	15	15	34	26	13	14	14	33	29	11	10	11
C	14	17	5	2	3	3	9	2	3	4	<u>20</u>	<u>24</u>	<u>11</u>	<u>13</u>	<u>13</u>
	22	19	13	16	15	33	27	16	15	14	16	12	7	5	5
G	3	4	3	3	3	3	8	4	3	4	3	8	5	5	8
	33	32	15	15	15	33	28	14	15	14	33	28	13	13	10
V	3	9	4	3	4	1	9	4	3	2	4	7	5	<u>10</u>	8
	33	27	14	15	14	35	27	14	15	16	32	29	13	8	10
B	12	13	3	3	4	5	5	4	3	4	<u>26</u>	<u>21</u>	<u>10</u>	<u>14</u>	<u>15</u>
	24	23	15	15	14	31	31	14	15	14	10	15	8	4	3

Table 17b: EFFECT OF DISTRIBUTION AND NO. OF OUTLIERS

	LS_GIL					LAD_GIL				
	E	0	1	2	3	E	0	1	2	3
N	4	4	1	5	4	3	3	3	4	0
	32	32	17	13	14	33	33	15	14	18
C	6	7	0	4	5	<u>28</u>	<u>22</u>	9	<u>13</u>	<u>13</u>
	30	29	18	14	13	8	14	9	5	5
G	7	10	1	4	7	3	2	2	3	4
	29	26	17	14	11	33	34	16	15	14
V	5	4	2	2	5	4	4	7	7	7
	31	32	16	16	13	32	32	11	11	11
B	7	3	1	2	5	<u>32</u>	<u>21</u>	<u>11</u>	<u>14</u>	<u>14</u>
	29	33	17	16	13	4	15	7	4	4

Table 18a: EFFECT OF DISTRIBUTION AND OUTLIER SIZE

LS_LAD										LAD_LS									
	0	1	2	3	4	5	S	M	B		0	1	2	3	4	5	S	M	B
N	3	1	2	2	1	0	4	4	3		2	3	1	3	1	2	5	5	3
	9	11	10	10	11	12	14	14	15		10	9	11	9	11	10	13	13	15
C	6	5	4	<u>7</u>	4	5	4	3	3		1	0	2	2	5	2	3	3	3
	6	7	8	5	8	7	14	15	15		11	12	10	10	7	10	15	15	15
G	1	1	1	2	1	1	3	3	3		2	3	1	2	2	1	3	4	4
	11	11	11	10	11	11	15	15	15		10	9	11	10	10	11	15	14	14
V	5	1	1	2	2	1	3	3	5		3	1	1	2	2	1	3	3	3
	7	11	11	10	10	11	15	15	13		9	11	11	10	10	11	15	15	15
B	6	4	4	4	3	4	5	3	2		4	0	3	0	1	2	5	3	3
	6	8	8	8	9	8	13	15	16		8	12	9	12	11	10	13	15	15

Table 18b: EFFECT OF DISTRIBUTION AND OUTLIER SIZE

LAD_LAD										LS_GIL									
	0	1	2	3	4	5	S	M	B		0	1	2	3	4	5	S	M	B
N	3	1	2	3	1	0	4	7	<u>11</u>		0	1	3	2	2	0	1	2	7
	9	11	10	9	11	12	14	11	7		12	11	9	10	10	12	17	16	11
C	<u>7</u>	6	<u>8</u>	<u>7</u>	<u>7</u>	<u>9</u>	<u>8</u>	<u>14</u>	<u>15</u>		2	1	0	3	5	2	1	3	5
	5	6	4	5	5	3	10	4	3		10	11	12	9	7	10	17	15	13
G	2	3	1	2	2	1	3	5	<u>10</u>		2	2	3	3	4	5	1	4	7
	10	9	11	10	10	11	15	13	8		10	10	9	9	8	9	17	14	11
V	4	1	1	2	2	1	3	7	<u>13</u>		0	0	2	4	0	3	2	2	5
	8	11	11	10	10	11	15	11	5		12	12	10	8	12	9	16	16	13
B	<u>8</u>	<u>8</u>	<u>9</u>	<u>7</u>	<u>8</u>	<u>7</u>	8	<u>14</u>	<u>17</u>		3	2	2	2	0	1	1	3	4
	4	4	3	5	4	5	10	4	1		9	10	10	10	12	11	17	15	14

Table 18c: EFFECT OF DISTRIBUTION AND OUTLIER SIZE

LAD_GIL									
	0	1	2	3	4	5	S	M	B
N	2	2	1	1	0	0	2	2	3
	10	10	11	11	12	12	16	16	15
C	<u>9</u>	<u>8</u>	<u>8</u>	<u>9</u>	<u>7</u>	<u>9</u>	7	<u>14</u>	<u>14</u>
	3	4	4	3	5	3	11	4	4
G	0	2	0	2	1	0	1	3	5
	12	10	12	10	11	12	17	15	13
V	5	0	0	1	1	1	7	7	7
	7	12	12	11	11	11	11	11	11
B	<u>10</u>	<u>9</u>	<u>10</u>	<u>9</u>	<u>7</u>	<u>8</u>	<u>12</u>	<u>14</u>	<u>13</u>
	2	3	2	3	5	4	6	4	5

Table 20: EFFECT OF OUTLIER SIZE AND SD SIZE

	LS_LAD		LAD-LS		LAD-LAD		LS-GIL		LAD-GIL	
	d	D	d	D	d	D	d	D	d	D
0	9	12	5	7	14	10	3	4	15	11
	21	18	25	23	16	20	27	26	15	19
1	6	6	2	5	8	11	3	3	11	10
	24	24	28	25	22	19	27	27	19	20
2	5	7	2	6	10	11	7	3	12	7
	25	23	28	24	20	19	23	27	18	23
3	8	9	3	6	9	12	9	5	13	9
	22	21	27	24	21	18	21	25	17	21
4	3	8	2	9	7	13	4	7	8	8
	27	22	28	21	23	17	26	23	22	22
5	4	7	0	8	8	10	3	6	11	7
	26	23	30	22	22	20	27	24	19	23
S	0	19	0	19	0	26	0	6	0	29
	0	71	0	71	0	64	0	84	0	61
M	0	16	0	18	0	<u>47</u>	0	14	0	40
	0	74	0	72	0	43	0	76	0	50
B	0	16	0	16	0	<u>66</u>	0	28	0	42
	0	74	0	74	0	24	0	62	0	48

Table 21: EFFECT OF NO. OF OUTLIERS AND SD SIZE

	LS_LAD		LAD-LS		LAD-LAD		LS-GIL		LAD-GIL	
	d	D	d	D	d	D	d	D	d	D
E	35	0	14	0	56	0	29	0	70	0
	145	0	166	0	124	0	151	0	110	0
0	0	49	0	41	0	67	0	28	0	52
	0	131	0	139	0	113	0	152	0	128
1	0	20	0	19	0	38	0	5	0	32
	0	70	0	71	0	52	0	85	0	58
2	0	14	0	16	0	<u>50</u>	0	17	0	41
	0	76	0	74	0	40	0	73	0	49
3	0	17	0	18	0	<u>51</u>	0	26	0	38
	0	73	0	72	0	39	0	64	0	52

c). Probit Analysis: In the preceding sections we have made an attempt to look into the performance of the candidate estimators in accordance with single criterion and double criteria. Application of more than two criteria for studying the effects of different specifications on the performance of the candidate estimators is cumbersome and

confusing. In view of this, we have made an attempt to apply Probit Analysis to assess the effect of different criteria on the performance of estimators. It may be noted that out of 630 experiments (of which 126 have models with normal residuals, and 360 have outlier-free residuals), the clear dominance of alternative estimators over 2-SLS was : LAD-LS (108), LS-GILN (111), LS-LAD (135), LAD-GILN (233) and LAD-LAD (262). Seen as such, LAD-GILN and LAD-LAD have commendable performance.

We have five dependent variables, each binary in nature, representing the success or failure (1, 0) of the estimator in outperforming 2-SLS. On the other hand, specifications may be used as the independent variables that explain the variations in the dependent variables. Accordingly, we have used the following independent (explanatory) variables that represent the model specifications:

N (Normal Distribution): A binary variable, taking on a value of unity if the disturbance term follows a normal distribution, zero otherwise.

C (Cauchy Distribution): A binary variable, taking on a value of unity if the disturbance term follows a Cauchy distribution, zero otherwise.

G (Gamma Distribution): A binary variable, taking on a value of unity if the disturbance term follows a Gamma distribution, zero otherwise.

B₁ (Beta₁ Distribution): A binary variable, taking on a value of unity if the disturbance term follows a Beta₁ distribution, zero otherwise.

B₂ (Beta₂ Distribution): A binary variable, taking on a value of unity if the disturbance term follows a B₂ distribution, zero otherwise.

SSD (Disturbance term with a small Standard Deviation): A binary variable, taking on a value of unity if the disturbance term has a small standard deviation = 0.001, zero otherwise.

BSD (Disturbance term with a larger Standard Deviation): A binary variable, taking on a value of unity if the disturbance term has a larger standard deviation = 0.3, zero otherwise.

OSS (Outliers of a small size): A binary variable, taking on a value of unity if the disturbance term has a small size outlier = 0.50 times the mean of the dependent variable, zero otherwise.

OSM (Outliers of a medium size): A binary variable, taking on a value of unity if the disturbance term has a medium size outlier = 1.00 times the mean of the dependent variable, zero otherwise.

OSB (Outliers of a large size): A binary variable, taking on a value of unity if the disturbance term has a medium size outlier = 1.50 times the mean of the dependent variable, zero otherwise.

MODEL (Size of the model): This variable takes on the value of 3 for a model with three equations, 5 for a model of 5 equations and 7 for a model with 7 equations.

OBSERV (Sample size): A binary variable, taking on a value of unity if the number of observations has a larger size = 50, zero otherwise.

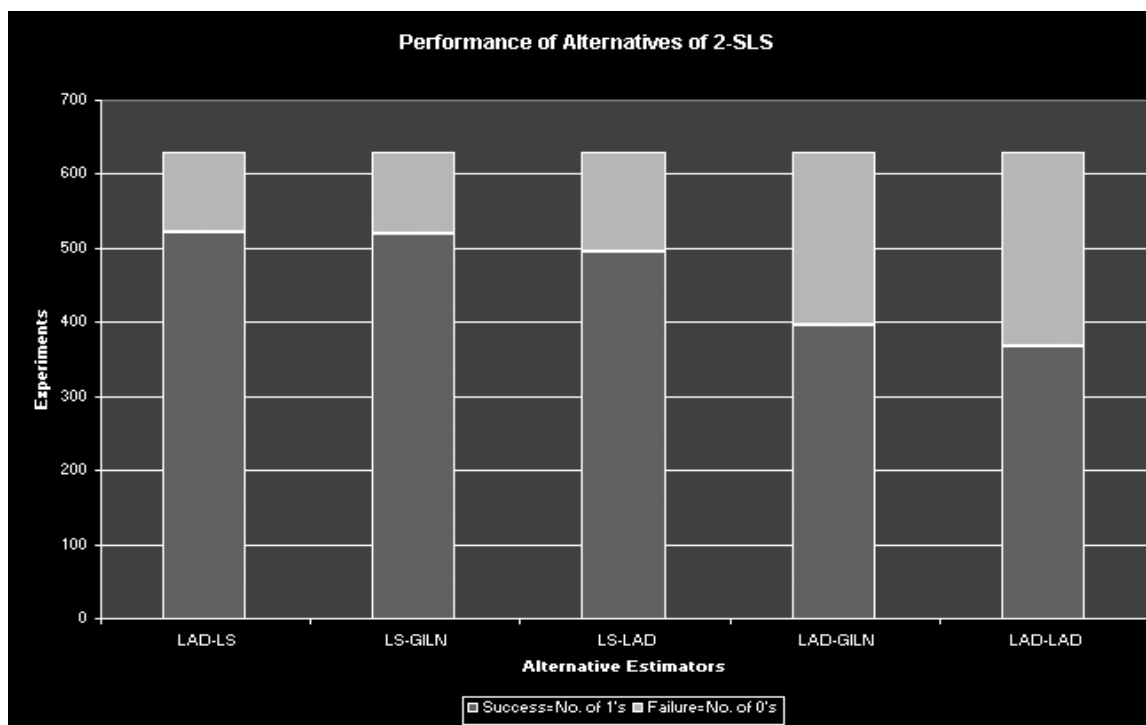
NOOUT (No. of Outliers): This variable takes on the value of 1 if the disturbance contains one outlier, 2 if the disturbance contains two outliers, 3 if the disturbance contains three outliers, and 0 if the disturbance contains no outlier.

Of distribution variables any four (out of 5 namely, N, C, G, B₁ and B₂) at most can at a time be included in the list of explanatory variables, since all five, if included, make up for a perfect multi-collinearity. Similarly, either SSD or BSD can be included among the explanatory variables since they together make up for a perfect multi-collinearity. Accordingly, our general Probit Regression model may be described as follows:

$$R_i = k_0 + a_1N + a_2C + a_3G + a_4B_1 + a_5B_2 + b_1SSD + b_2BSD + c_1OSS + c_2OSM + c_3OSB + r_1MODEL + r_2OBSERV + r_3NOUT + error$$

where R_i (binary measure of the success/failure of one of the estimators, namely LS-LAD, LAD-LS, LAD-LAD, LS-GILN or LAD-GILN). There would be some zero restrictions on the coefficients (implying absence of some particular explanatory variables), These restrictions are: (i) at least one of the a_j will be zero and (ii) either b₁ or b₂ (or both) will be zero. Other coefficients may be zero or non-zero.

The results of Probit Analysis are presented in the following tables (22 through 52). In most of these tables (such as table 22, etc.), the constant and coefficients estimated by Probit analysis together with their standard error of estimates, 't' values and probability level of significance have been reported. However, in a few tables (such as table 24), the estimates of constant and coefficients alone- without their standard error, 't' value and prob. level of significance – have been presented, since we could not estimate them on account of instability in computation. Estimated regression parameters without their variance and probability level of significance may carry some numerical sense, but they cannot be used to draw any reliable conclusions.



1. LS-LAD Estimator [0's = 495 (78.57%) and 1's = 135 (21.43%)] Tables 22 – 24.

Table 22. Results of Probit Analysis on LS-LAD MODEL-1

(Final loss: 57.588486091 R=.67608 Variance explained: 45.708%)

	Const.B0	C	G	B2	B1	BSD
Estimate	12.972	1.016929	-0.2145	1.016363	-0.11387	1.111137
Std.Err.	2672.913	0.536967	0.400879	0.536726	0.369484	0.260645
t (619)	0.005	1.893836	-0.535073	1.893633	-0.308185	4.263026
p-level	0.996	0.058713	0.592792	0.05874	0.758045	0.000023

OSS	OSM	OSB	MODEL	OBSERV
3.7355	3.0757	3.5786	4.5515	-2.24346
578.6135	568.0178	628.142	287.8463	32.85842
0.0065	0.0054	0.0057	0.0158	-0.06828
0.9949	0.9957	0.9955	0.9874	0.94559

Table 23. Results of Probit Analysis on LS-LAD MODEL - 2

(Final loss: 57.758058080 R=.67489 Variance explained: 45.548%)

	Const.B0	C	G	B2	B1	BSD	OSM
Estimate	6.6846	1.00372	-0.165058	1.003753	-0.160269	1.375412	2.6298
Std.Err.	536.4836	0.544654	0.415157	0.54469	0.378493	0.251181	118.9776
t(620)	0.0125	1.842858	-0.397581	1.842798	-0.42344	5.475782	0.0221
p-level	0.9901	0.065827	0.691076	0.065836	0.672121	0	0.9824

OSB	MODEL	OBSERV
1.195103	4.17228	-1.79615
6.183752	49.90763	9.12829
0.193265	0.0836	-0.19677
0.846815	0.9334	0.84407

Table 24. Results of Probit Analysis on LS-LAD MODEL - 3

(Final loss: 63.844444564 R=.63095 Variance explained: 39.810%)

	Const.B0	C	B2	OSM	OSB	MODEL	OBSERV
Estimate	.014820	.508784	.509028	-2.50163	-2.15845	8.539513	-2.23612

2. LAD-LS Estimator [0's = 522 (82.86%) and 1's = 108 (17.14%)] Tables 25 – 27.

Table 25. Results of Probit Analysis on LAD-LS MODEL-1

(Final loss: 41.833333512 R=.72974 Variance explained: 53.251%)

	Const.B0	C	G	B2	B1	BSD
Estimate	117.771400	0.430350	0.430195	-9.203730	-3.388040	9.633954

OSS	OSM	OSB	MODEL	OBSERV
9.052706	8.381168	6.825761	15.896940	-11.4740

Table 26. Results of Probit Analysis on LAD-LS MODEL - 2

(Final loss: 42.333333358 R=.72590 Variance explained: 52.693%)

	Const.B0	C	G	B2	B1	BSD	OSM
Estimate	100.7849	0.430843	0.430935	-13.2181	-9.08827	14.0798	8.623681

OSB	MODEL	OBSERV
6.298482	16.86109	-10.9622

Table 27. Results of Probit Analysis on LAD-LS MODEL - 3

(Final loss: 58.844444475 R=.58516 Variance explained: 34.242%)

	Const.B0	C	B2	OSM	OSB	MODEL	OBSERV
Estimate	-.017995	.508456	.508266	-2.57452	-2.61524	8.980038	-2.34459

3. LS-GILN Estimator [0's = 519 (82.38%) and 1's = 111 (17.62%)] Tables 28-31.

Table 28. Results of Probit Analysis on LS-GILN MODEL-1

(Final loss: 86.659218670 R=.22872 Variance explained: 5.2313%)

	Const.B0	C	G	B2	B1	BSD
Estimate	-1.16897	0.050829	0.324826	-0.082948	-0.12644	0.00436
Std.Err.	0.32894	0.186914	0.170757	0.198302	0.202258	0.198711
t (619)	-3.55372	0.27194	1.902273	-0.418289	-0.625143	0.021939
p-level	0.00041	0.785759	0.057599	0.675881	0.532108	0.982504

OSS	OSM	OSB	MODEL	OBSERV
-0.64836	0.003488	0.473462	0.073898	-0.00592
0.37051	0.215857	0.180431	0.036929	0.00429
-1.7499	0.016159	2.624063	2.001109	-1.37985
0.08063	0.987112	0.008903	0.045817	0.16813

Table 29. Results of Probit Analysis on LS-GILN MODEL - 2

(Final loss: 86.728418538 R= 0.22706; Variance explained: 5.1556%)

	Estimate	Std.Err.	t (619)	p-level
Const.B0	-.40145	.33955	-1.18227	.23755
C	.038687	.242300	.159666	.873197
G	.320953	.226485	1.417106	.156955
B2	-.016729	.247495	-.067594	.946130
B1	-.225418	.270901	-.832107	.405669
BSD	-2.85763	1.17481	-2.43241	.01528
OSM	1.368767	.939062	1.457589	.145461
OSB	2.003701	.966681	2.072764	.038607
MODEL	.041962	.048475	.865634	.387026
OBSERV	-.02587	.00652	-3.96813	.00008
NOUT	.679522	.169111	4.018197	.000066

Table 30. Results of Probit Analysis on LS-GILN MODEL – 3

(Final loss: 87.448130354 R=.20901 Variance explained: 4.3685%)

	Const.B0	C	G	B2	B1	BSD	OSM
Estimate	-1.05447	0.040429	0.302702	-0.08073	-0.112993	-0.179200	0.183059
Std.Err.	0.314360	0.187858	0.171551	0.199075	0.201850	0.161300	0.193197
t (620)	-3.35439	0.215209	1.764503	-0.40554	-0.559790	-1.110930	0.947527
p-level	0.000840	0.829675	0.078140	0.685222	0.575825	0.267030	0.343739

OSB	MODEL	OBSERV
0.6548940	0.0643260	-0.00752
0.1659460	0.0366430	0.004370
3.9464320	1.7554530	-1.71841
0.0000880	0.0796760	0.086220

Table 31. Results of Probit Analysis on LS-GILN MODEL - 4

(Final loss: 88.481971203 R=.17994 Variance explained: 3.2380%)

	Const.B0	C	B2	OSM	OSB	MODEL	OBSERV
Estimate	-1.13949	-.036616	-.165448	.084131	.562434	.065837	-.00566
Std.Err.	.25480	.151351	.166013	.175270	.138591	.036215	.00408
t(623)	-4.47209	-.241930	-.996596	.480007	4.058236	1.817940	-1.38713
p-level	.00001	.808914	.319348	.631391	.000056	.069553	.16590

4. LAD-GILN Estimator [0's =397 (63.02%) and 1's = 233 (36.98%)] Tables 32-38.

Table 32. Results of Probit Analysis on LAD-GILN MODEL-1

(Final loss: 85.287907296 R=.64740 Variance explained: 41.913%)

	Const.B0	C	G	B2	B1	BSD
Estimate	-4.1195	3.155233	-0.030486	3.37653	1.006337	-0.65431
Std.Err.	0.7469	0.621093	0.615405	0.644996	0.463965	0.21417
t (619)	-5.5155	5.080128	-0.049538	5.234962	2.168994	-3.05501
p-level	0	0	0.960507	0	0.030463	0.00235

OSS	OSM	OSB	MODEL	OBSERV
0.067873	0.848432	0.928644	-0.015825	0.053537
0.29256	0.251695	0.268254	0.048038	0.01204
0.231996	3.370879	3.461806	-0.329432	4.446586
0.816618	0.000796	0.000574	0.741941	0.00001

Table 33. Results of Probit Analysis on LAD-GILN MODEL – 2

(Final loss: 85.295478413 R=.64736 Variance explained: 41.907%)

	Const.B0	C	G	B2	B1	BSD	OSM
Estimate	-4.08645	3.130142	-0.023476	3.348295	1.000908	-0.63437	0.830469
Std.Err.	0.71183	0.598506	0.620358	0.618255	0.464572	0.1925	0.237619
t(620)	-5.74079	5.229924	-0.037843	5.415718	2.154475	-3.29533	3.494965
p-level	0	0	0.969825	0	0.031587	0.00104	0.000508

OSB	MODEL	OBSERV
0.909613	-0.013612	0.05276
0.253103	0.046219	0.010898
3.59384	-0.294518	4.841257
0.000352	0.768461	0.000002

Table 34. Results of Probit Analysis on LAD-GILN MODEL - 3

(Final loss: 87.205790860 R=.63723 Variance explained: 40.606%)

	Const.B0	C	G	B2	B1	OSM	OSB
Estimate	-3.89864	2.689656	-0.043287	2.900686	0.864517	0.504024	0.562003
Std.Err.	0.61912	0.460604	0.473746	0.479674	0.363262	0.196777	0.204884
t(621)	-6.29711	5.839416	-0.091372	6.047204	2.379875	2.561399	2.74303
p-level	0	0	0.927226	0	0.017619	0.01066	0.006263

MODEL	OBSERV
-0.003717	0.046367
0.043826	0.009712
-0.084818	4.774026
0.932434	0.000002

Table 35. Results of Probit Analysis on LAD-GILN MODEL - 4

(Final loss: 88.972275670 R=.62772 Variance explained: 39.403%)

	Const.B0	C	B2	OSM	OSB	MODEL	OBSERV
Estimate	-3.65058	2.379498	2.594783	.438951	.502348	-.003021	.049106
Std.Err.	.57509	.359687	.385618	.191254	.200937	.046389	.010783
t(623)	-6.34783	6.615466	6.728892	2.295124	2.500020	-.065132	4.553896
p-level	.00000	.000000	.000000	.022057	.012675	.948090	.000006

Table 36. Results of Probit Analysis on LAD-GILN MODEL - 5

(Final loss: 88.972972053 R=.62772 Variance explained: 39.403%)

	Const.B0	C	B2	OSM	OSB	OBSERV
Estimate	-3.65119	2.370894	2.585693	.439568	.503179	.048871
Std.Err.	.57311	.331435	.356232	.190725	.200297	.010075
t (624)	-6.37082	7.153420	7.258445	2.304728	2.512164	4.850489
p-level	.00000	.000000	.000000	.021509	.012251	.000002

Table 37. Results of Probit Analysis on LAD-GILN MODEL - 6

(Final loss: 86.988081904 R=.63839 Variance explained: 40.755%)

	Const.B0	C	B2	SSD	OSS	OSM	OSB
Estimate	-4.427640	2.773550	2.993900	0.671478	-0.028879	0.797253	0.86487
Std.Err.	0.747110	0.458020	0.486787	0.233665	0.314139	0.263632	0.27309
t (621)	-5.926380	6.055528	6.150323	2.873677	-0.091932	3.024116	3.16693
p-level	0.000000	0.000000	0.000000	0.004196	0.926782	0.002597	0.00162

MODEL	OBSERV
-0.01536	0.05566
0.049236	0.01216
-0.31202	4.57968
0.755134	0.00000

Table 38. Results of Probit Analysis on LAD-GILN MODEL - 7

Final loss: 86.988081904 R=.63839 Variance explained: 40.755%

	Const.B0	C	B2	BSD	OSS	OSM	OSB
Estimate	-3.75616	2.77355	2.9939	-0.67148	-0.028879	0.797254	0.86487
Std.Err.	0.61657	0.457439	0.486179	0.23362	0.315913	0.263677	0.27313
t (621)	-6.09203	6.06322	6.158022	-2.87421	-0.091415	3.023602	3.16653
p-level	0	0	0	0.00419	0.927192	0.002601	0.00162

MODEL	OBSERV
-0.01536	0.05566
0.049216	0.01214
-0.31214	4.58366
0.755037	0.00000

Table 39. A GIST OF Results of Probit Analysis on LAD-GILN MODELS
[s (x) = Significant (insignificant) at 5% - or less level of Significance]

MODEL	1	2	3	4	5*	6	7
<i>N</i>	-	-	-	-	-	-	-
<i>C</i>	S	S	S	S	S	S	S
<i>G</i>	X	X	X	-	-	-	-
<i>B1</i>	S	S	S	-	-	-	-
<i>B2</i>	S	S	S	S	S	S	S
<i>SSD</i>	-	-	-	-	-	S	-
<i>BSD</i>	S	S	-	-	-	-	S
<i>OSS</i>	X	-	-	-	-	X	X
<i>OSM</i>	S	S	S	S	S	S	S
<i>OSB</i>	S	S	S	S	S	S	S
<i>MODEL</i>	X	X	X	X	-	X	X
<i>OBSERV</i>	S	S	S	S	S	S	S
<i>NOUT</i>	-	-	-	-	-		

▪ means a variable not appearing in the model.

Some More Results of Probit Analysis on LS-LAD, LAD-LS and LAD-GILN

Table 40. Probit Regression models

Estimates* (Coefficients) of Regression Parameters
(No Standard Errors could be estimated due to ill
conditioned Matrices)

ESTIMATOR	LSLAD	LADLS	LADGILN
Const.B0	492.2791	108.6147	-11.6156
C	1.016650	.430469	10.39778
G	-.214281	.430340	1.208702
B2	1.016652	-21.4800	10.81038
B1	-.113823	-12.2892	7.482589
BSD	1.111051	21.91036	-1.00659
OSM	-20.9887	7.167700	.634941
OSB	-31.1156	13.57137	26.19755
MODEL	128.5822	33.55067	-.037582
OBSERV	-69.6195	-17.1950	.072052
NOUT	61.89762	4.115859	.245192
No. of 0's	495	522	397
No. of 1's	135	108	233
Final Loss	57.5884838	41.8333333	114.82024
R	0.67608	0.72974	0.46689
Var Expl %	45.708	53.251	21.799

* Estimates are not reliable

5. LAD-LAD Estimator [0's = 368 (58.41%) and 1's = 262 (41.59%)] Tables 41-51.

Table 41. Results of Probit Analysis on LAD-LAD MODEL-1

(Final loss: 79.525547544 R=.69308 Variance explained: 48.037%)

	Const.B0	C	G	B2	B1	SSD
Estimate	-3.48671	1.689329	-0.118133	1.880458	0.028356	-0.33797
Std.Err.	0.38771	0.229555	0.174506	0.241081	0.17161	0.14962
t (619)	-8.99306	7.359147	-0.676956	7.800105	0.165233	-2.25884
p-level	0	0	0.498687	0	0.868814	0.02424

OSS	OSM	OSB	MODEL	OBSERV
-0.52324	0.749134	1.754328	0.586701	-0.01813
0.20556	0.190566	0.238384	0.063169	0.00432
-2.54543	3.931096	7.359248	9.28783	-4.19703
0.01116	0.000094	0	0	0.00003

Table 42. Results of Probit Analysis on LAD-LAD MODEL-2

(Final loss: 79.529092191 R=.69307 Variance explained: 48.034%)

	Const.B0	C	G	B2	SSD	OSS
Estimate	-3.47349	1.675774	-0.13198	1.86718	-0.33877	-0.52283
Std.Err.	0.37849	0.213707	0.152787	0.226642	0.14945	0.20531
t (620)	-9.17715	7.841457	-0.863815	8.238456	-2.26675	-2.54649
p-level	0	0	0.388023	0	0.02375	0.01112

OSM	OSB	MODEL	OBSERV
0.747227	1.755375	0.58696	-0.01814
0.189897	0.238421	0.063148	0.00432
3.9349	7.362496	9.294999	-4.20288
0.000093	0	0	0.00003

Table 43. Results of Probit Analysis on LAD-LAD MODEL-3

(Final loss: 152.57038078 R=.05547 Variance explained: .30769%)

	Const.B0	C	G	B2	BSD	OSS
Estimate	2.1517	0.79124	-0.005745	1.667483	0.742743	-0.294682
Std.Err.	237.8461	0.413434	0.348338	0.791596	0.435464	0.38418
t (620)	0.009	1.913822	-0.016492	2.106482	1.705638	-0.767041
p-level	0.9928	0.056103	0.986847	0.035564	0.088576	0.443349

OSM	OSB	MODEL	OBSERV
0.799979	3.179967	0.864001	-0.40944
0.549723	1.245559	0.349777	11.98544
1.455241	2.553043	2.470151	-0.03416
0.146109	0.010917	0.013774	0.97276

Table 44. Results of Probit Analysis on LAD-LAD MODEL-4

(Final loss: 79.525547544 R=.69308 Variance explained: 48.037%)

	Const.B0	C	G	B2	B1	BSD
Estimate	-3.82468	1.68933	-0.118133	1.880459	0.028356	0.337967
Std.Err.	0.41944	0.229428	0.174344	0.240973	0.17076	0.149645
t (619)	-9.1186	7.36322	-0.677587	7.803599	0.166055	2.258456
p-level	0	0	0.498287	0	0.868168	0.024265

OSS	OSM	OSB	MODEL	OBSERV
-0.52324	0.749135	1.754329	0.586702	-0.01813
0.20558	0.190558	0.238387	0.063185	0.00432
-2.5452	3.93128	7.359158	9.285492	-4.19676
0.01116	0.000094	0	0	0.00003

Table 45. Results of Probit Analysis on LAD-LAD MODEL-5

(Final loss: 79.614951277 R=.69266 Variance explained: 47.978%)

	Const.B0	N	C	B2	SSD	OSS
Estimate	-3.53411	0.043938	1.734826	1.926834	-0.34117	-0.52316
Std.Err.	0.38247	0.148474	0.216851	0.230351	0.14933	0.2053
t (620)	-9.24019	0.295931	8.000094	8.364765	-2.28469	-2.54827
p-level	0	0.767382	0	0	0.02267	0.01107

OSM	OSB	MODEL	OBSERV
0.74385	1.760601	0.587716	-0.01818
0.19051	0.239733	0.063366	0.00432
3.904509	7.34401	9.275007	-4.20484
0.000105	0	0	0.00003

Table 46. Results of Probit Analysis on LAD-LAD MODEL-6

(Final loss: 79.626177524 R=.69261 Variance explained: 47.971%)

	Const.B0	C	B2	SSD	OSS	OSM
Estimate	-3.51815	1.719273	1.910865	-0.34022	-0.52395	0.746808
Std.Err.	0.37768	0.20953	0.222927	0.14911	0.20522	0.19051
t (621)	-9.31523	8.205394	8.571704	-2.28173	-2.55314	3.920048
p-level	0	0	0	0.02284	0.01091	0.000098

OSB	MODEL	OBSERV
1.759795	0.587351	-0.01815
0.239213	0.063254	0.00431
7.356608	9.285591	-4.20674
0	0	0.00003

Table 47. Results of Probit Analysis on LAD-LAD MODEL-7

(Final loss: 79.626177524 R=.69261 Variance explained: 47.971%)

	Const.B0	C	B2	BSD	OSS	OSM
Estimate	-3.85837	1.719272	1.910865	0.34022	-0.52395	0.746808
Std.Err.	0.41113	0.20955	0.222953	0.149115	0.20522	0.190503
t (621)	-9.3849	8.204589	8.570716	2.281593	-2.55303	3.920198
p-level	0	0	0	0.022851	0.01092	0.000098

OSB	MODEL	OBSERV
1.759795	0.587351	-0.01815
0.239212	0.063265	0.00431
7.356643	9.284008	-4.20676
0	0	0.00003

Table 48. Results of Probit Analysis on LAD-LAD MODEL - 8

(Final loss: 80.465897407; R= 0.68864; Variance explained: 47.422%)

	Estimate	Std.Err.	t(619)	p-level
Const.B0	-3.70678	.40246	-9.21031	.00000
C	1.637305	.223215	7.335106	.000000
G	-.120887	.170792	-.707800	.479336
B2	1.844453	.238770	7.724805	.000000
B1	.023231	.167254	.138899	.889575
BSD	.194524	.145066	1.340942	.180431
OSM	.875303	.214182	4.086722	.000050
OSB	1.858589	.258262	7.196515	.000000
MODEL	.568052	.060475	9.393214	.000000
OBSERV	-.01776	.00428	-4.15357	.00004
NOU	-.005236	.071263	-.073476	.941451

Table 49. Results of Probit Analysis on LAD-LAD MODEL - 9

(Final loss: 80.466586121; R= 0.68863; Variance explained: 47.422%)

	Estimate	Std.Err.	t(619)	p-level
Const.B0	-3.70460	.40067	-9.24613	.00000
C	1.636808	.222770	7.347533	.000000
G	-.120836	.170407	-.709104	.478526
B2	1.844610	.238500	7.734217	.000000
B1	.023297	.166082	.140277	.888487
BSD	.191383	.138505	1.381778	.167538
OSM	.867434	.184959	4.689873	.000003
OSB	1.850632	.233764	7.916657	.000000
MODEL	.567729	.060213	9.428700	.000000
OBSERV	-.01777	.00427	9.428700	.00004

Table 50. Results of Probit Analysis on LAD-LAD MODEL - 10

(Final loss: 80.720763957 R=.68743 Variance explained: 47.256%)

	Const.B0	C	G	B2	B1	OSM	OSB
Estimate	-3.54747	1.628814	-0.120596	1.829346	0.031776	0.933656	1.907921
Std.Err.	0.37088	0.220673	0.17137	0.233624	0.167385	0.179377	0.229595
t(621)	-9.56501	7.381129	-0.703717	7.830313	0.189837	5.204982	8.309931
p-level	0	0	0.481873	0	0.849499	0	0

MODEL	OBSERV
0.561024	-0.01771
0.05875	0.00425
9.549385	-4.16676
0	0.00004

Table 51. Results of Probit Analysis on LAD-LAD MODEL - 11

Final loss: 80.833323316 R=.68689 Variance explained: 47.182%

	Const.B0	C	B2	OSM	OSB	MODEL	OBSERV
Estimate	-3.57772	1.657526	1.858374	.931565	1.913109	.561400	-.01772
Std.Err.	.36108	.200966	.215743	.179432	.230431	.058798	.00424
t(623)	-9.90832	8.247798	8.613813	5.191757	8.302320	9.547899	-4.17497
p-level	.00000	.000000	.000000	.000000	.000000	.000000	.00003

Table 52. A GIST OF Results of Probit Analysis on LAD-LAD MODELS

[s (x) = Significant (insignificant) at 5% - or less level of Significance]

MODEL	1	2	3	4	5	6*	7*	8	9	10	11
<i>N</i>	-	-	-	-	X	-	-	-	-	-	-
<i>C</i>	S	S	S	S	S	S	S	S	S	S	S
<i>G</i>	X	X	X	X	-	-	-	X	X	X	-
<i>B1</i>	X	-	-	X	-	-	-	X	X	X	-
<i>B2</i>	S	S	S	S	S	S	S	S	S	S	S
<i>SSD</i>	S	S	-	-	S	S	-	-	-	-	-
<i>BSD</i>	-	-	S	S	-	-	S	X	X	-	-
<i>OSS</i>	S	S	X	S	S	S	S	-	-	-	-
<i>OSM</i>	S	S	X	S	S	S	S	S	S	S	S
<i>OSB</i>	S	S	S	S	S	S	S	S	S	S	S
<i>MODEL</i>	S	S	S	S	S	S	S	S	S	S	S
<i>OBSERV</i>	S	S	X	S	S	S	S	S	S	S	S
<i>NOUT</i>	-	-	-	-	-	-	-	X	-	-	-

- means a variable not appearing in the model.

A summary table below indicates that LAD-LAD model 6 and 7 are the best (followed by LAD-GILN model 5). LAD-LAD outperforms 2-SLS for Cauchy and B_2 distributed disturbance term with small/large SD containing small/medium/large outliers for smaller as well as larger models of whatever sample size. LAD-GILN is the second best estimator. These findings agree with those obtained in case of double criteria analysis presented earlier.

Table 53. A Summary of Performance of Estimators and Explanatory Variables Evaluated by Probit Regression Model			
Estimator	LAD-LAD		LAD-GILN
Criteria/Variables	Model 6*	Model 7*	Model 5 *
<i>N</i>	-	-	-
<i>C</i>	s	s	S
<i>G</i>	-	-	–
<i>B1</i>	-	-	–
<i>B2</i>	s	s	S
<i>SSD</i>	s	-	–
<i>BSD</i>	-	s	–
<i>OSS</i>	s	s	–
<i>OSM</i>	s	s	S
<i>OSB</i>	s	s	S
<i>MODEL</i>	s	s	–
<i>OBSERV</i>	s	s	S
<i>NOUT</i>	-	-	-
Here s stands for Probit Regression coefficient statistically significant			

DasGupta, Madhuchhanda (2004) LEAST ABSOLUTE DEVIATION ESTIMATION OF MULT-EQUATION LINEAR ECONOMETRIC MODELS : A STUDY BASED ON MONTE CARLO EXPERIMENTS, Doctoral Dissertation, Dept. of Economics, North Eastern Hill University, Shillong (India); Supervisor – Prof. SK Mishra

CHAPTER 7

SUMMARY AND CONCLUDING REMARKS

I. Introduction: This chapter has two objectives; first to summarize the investigation that is spread through out the foregoing chapters and secondly to conclude our investigation and give an outline of the possible investigation that may follow ours.

II. The Motivation to Investigation: Ideal properties of the Ordinary Least Squares (OLS or L_2 norm) estimator of single-equation linear econometric model $y = X\mathbf{a} + e$, while e obeys Gauss-Markov conditions, are well known. Additionally, if e is normally distributed, OLS estimator of \mathbf{a} is also the Maximum Likelihood (ML) estimator. However, when e is non-normally distributed, hyper-kurtic or infested with sizeable outliers, OLS estimator fails to perform.

The real world data often consist of disturbances that are non-normally distributed, some of which permit the variate to take on non-negative values only and some others are distributed with infinite variance. An infinite variance means ‘thick tails’, which implies that large values or ‘outliers’ are present. It has been observed that in such cases ***Least Absolute Deviation*** (LAD or L_1 norm) estimator performs very well. Sporadic errors in X (where sample X is true $X + \Xi$, and Ξ is a sparse matrix with nonzero elements substantial in size) also vitiate OLS estimation. There too, LAD performs well.

The recommendations for using LAD estimator may be traced back to Gauss and Laplace. But poor amenability to analytical techniques and cumber in computation went

in its disfavour. After computers came in to reduce the burden of numerical computation of statistical/mathematical methods in scientific research, it could become possible to lay hands on the studies of properties as well as applications of the LAD estimator. Charnes, Cooper & Ferguson (1955) were the first to transform the problem of estimation of parameters of a linear (regression) model into an LP problem and solve it by minimization of absolute deviations. Later, iterative and search methods to obtain the solution were also discovered. At present LAD is an oft-used alternative estimator to OLS.

Multi-equation linear econometric models, described as $YA + XB + E = 0$, are often estimated by minimization of L_2 norm. Two-Stage Least Squares (2-SLS) is the most-used method of estimation of multi-equation econometric models. In 2-SLS estimation, OLS is used to estimate the matrix of Reduced Form Coefficients (P) at the first stage, which also gives estimated Y ($\hat{Y} = XP$). In the second stage, Y is replaced by \hat{Y} iff it appears as an explanatory variable in any structural equation. After this replacement, OLS is applied to estimate the structural equations, one at a time. Thus, OLS is applied twice, once at each stage. In spite of OLS – the basic building block of 2-SLS – being an ideal estimator if the required conditions for its application are met, 2-SLS is ordinarily a biased but consistent estimator. It has been found that 2-SLS performs poorly when non-normally distributed, hyper-kurtic or outlier-infested errors are met with.

Glahe & Hunt (1970) were the first to apply LAD estimation method to multi-equation linear models. Thereafter, a number of works appeared which contributed significantly to the understanding of properties and applicability of LAD estimation applied to multi-equation econometric models. In real life, small or medium size samples

are important. Small or medium sample properties of an estimator are difficult to obtain by analytical methods. Therefore, Amemiya (1982) suggested Monte Carlo experiments to assess the performance of LAD-based estimators of multi-equation models with small or medium size samples. Our literature survey suggests that perhaps no study was conducted in this line. Possibly, frequent application of single equation estimation by LAD attracted intensive research as it is in the domain of statistics in which users from many disciplines are interested while multi-equation estimation is limited to econometric models only. These observations motivated us to take up the investigation at hand.

III. The Methodology of Investigation and Findings: In view of the poor amenability of LAD estimators to analytical methods, especially in estimating the multi-equation econometric models, we have chosen the simulation method, namely, the Monte Carlo method, to address to the problem. As stated earlier, Amemiya also suggested the same. The Monte Carlo method starts by postulating a specific model and assign numerical values to all parameters in the model. In a single equation model such as $y = X\mathbf{a} + e$, X and \mathbf{a} are assigned specific numerical values. The dimension of $X(n,m)$ is fixed. Properties of e (such as its pdf or distribution and the parameters of the distribution) are specified. For fixed $X\mathbf{a}$, disturbances (e) are generated with the given specification and, using the candidate estimator, \mathbf{a} is estimated to give \hat{a} repeatedly for a large (say, r) number of times. This gives an array of estimated parameters, say $\hat{a}(r)$. Then $\text{mean}(\hat{a}(r))$, $\text{variance}(\hat{a}(r))$ and $\text{RMS}(\hat{a}(r))$ are obtained. If needed, higher order moments of $\hat{a}(r)$ may be obtained to determine its distribution. Performance of the candidate estimator is judged on these criteria. Smaller are the bias and the variance,

better is the performance of the candidate estimator. Smaller RMS may be used as a comprehensive measure of better performance. This exercise makes a unit experiment. Such experiments are carried out repeatedly. Conclusions are derived from the results of these experiments.

In case of a multi-equation models, the experiment starts with given (pre-assigned) $X(n,k)$, $A(m,m)$ and $B(k,m)$ matrices as well as the specifications regarding the structural disturbances, E . Then the matrix of sample disturbances, E , is generated with the pre-specified statistical properties. With the given X , $\Pi = -BA^{-1}$ and $U = -EA^{-1}$, $Y = X\Pi + U$ is obtained. The candidate estimator is now applied to obtain \hat{A} and \hat{B} . The process of generation of E , $Y = X\Pi + U$ and estimation of \hat{A} and \hat{B} is repeated for a large number of times and finally bias, variance and RMS are obtained as in case of single equation models. This exercise makes a unit experiment. Such experiments are carried out repeatedly. Conclusions are derived from the results of these experiments.

In Monte Carlo experiments parameters such as sample size (n), model size (m in single equation models, and m and k in multi-equation models) and distribution of E and the parameters of chosen distribution are variables that may influence the performance of a candidate estimator. Therefore, one may vary some or all of these parameters to study the performance of the candidate estimator.

It is also possible to study the performance of a candidate estimator in presence of outliers. Then experiments are carried out with $y = Xa + e + O$ or $YA + XB + E + O = 0$, where O is the matrix of outliers. Usually O has a sparse structure, a few of its elements being non-zero and the rest zero. Non-zero elements are usually randomly spaced and random in magnitude, though within a given range. Large and numerous outliers have a

great impact on the performance of LS-based estimators while LAD-based estimators are usually immune to their number and size.

The Candidate Estimators: For single equation models LAD estimator is the direct contestant against the OLS estimator. But in case of multi-equation models, the 2-SLS (or LS-LS), which is the standard LS-based estimator, may have numerous contestants. The performance of 2-SLS is the reference for the contestant estimators. Among the latter, we have included (i) LS-LAD, (ii) LAD-LS, (iii) LAD-LAD, (iv) LS-GILN and (v) LAD-GILN estimators.

LS-LAD estimator obtains P (the matrix of Reduced form Coefficients) by OLS, but the structural coefficients (a_j and b_j) by LAD estimator. LAD-LS estimator obtains P by LAD, but the structural coefficients (a_j and b_j) by OLS. It may be considered as a variant of IV (Instrumental Variable) estimators where LAD-estimated \hat{Y} are used as the instrumental variables for Y . LAD-LAD estimator is D2SLAD of Amemiya (1982), which obtains P as well as the structural coefficients (a_j and b_j) by LAD estimator. The LS-GILN is the Khazzoom estimator (Khazzoom, 1976) that estimates reduced form equations of a multi-equation linear econometric model by OLS but (in the second stage) instead of estimating the (modified) structural equations by OLS (or the Instrumental variable method) as done by the 2-SLS, it applies generalized inverse of the relevant submatrix of reduced form coefficients to obtain the structural coefficients. In LS-based GILN (Khazzoom estimator or GILS), $P = (X'X)^{-1}X'Y$ is the matrix of Reduced Form coefficients estimated by OLS. However, if P is estimated by LAD, the GILN is LAD-GILN, which may perform better than LS-GILN (= GILS) if the structural disturbances are non-normal or infested with outliers.

Distribution of Disturbance: LS-based estimators of linear econometric model with normal disturbance have several merits, but for non-normal disturbances they may lose them. Therefore, in this study we compare the performance of candidate (LAD-based) estimators with LS-based estimators for the linear models with normally as well as non-normally distributed disturbances. Among the non-normal distributions we have chosen (i) Beta₁, (ii) Beta₂ (iii) Cauchy, and (iv) Gamma distributions. Among these, Gamma is a very skewed distribution for small shape and scale parameters. Gamma variates are non-negative. Beta variates are non-negative and largely platy-kurtic. Notoriety of Cauchy variates is well known. The 2nd and higher moments of Cauchy distributed variates do not exist (in the population), though in samples they may be computed, but are unpredictable as they vary wildly from sample to sample. Yet, Cauchy distributed variates have zero mean and it is a symmetric distribution.

The Process of Generating Random Variates with different Distributions: Variates following these distributions are computer generated. The program runs the following process (and exploits the relevant theorems regarding their nature). First, uniformly distributed random numbers are generated by some suitable method. Uniformly distributed random numbers may be transformed into normally distributed random numbers, $N(0,1)$. Standard normal variates may in turn be used to obtain Gamma, Beta or Cauchy distributed variates by suitable transformation. In general, starting from uniformly distributed variates, we may obtain a variate with almost any kind of distribution by a sequence of relevant transformations.

The Design of Monte Carlo Experiments with Single Equation Linear Models: A

single equation linear econometric model is $y = X\mathbf{a} + \mathbf{e} + \mathbf{O}$, where \mathbf{e} is the disturbance term with a specified distribution and \mathbf{O} is the sparse vector of outliers with a specified number of nonzero elements lying within a specified range and spaced randomly over the sample observations. $X(n,m)$ is the (fixed) non-stochastic matrix, n referring to the sample size and m referring to the number of explanatory variables (model size). We have experimented with two sample sizes; $n = 20$ and $n = 50$. Models are of three different sizes; $m = 2$ (Model₁), $m = 4$ (Model₂) and $m = 6$ (Model₃). Disturbances (\mathbf{e}) are of five different distribution types; Normal, Cauchy, Gamma(2), Beta₁(2,2) and Beta₂(2,2). Disturbances have standard deviations = 0.1 relative to the mean magnitude of the dependent variable, y . The numbers of outliers (non-zero elements in \mathbf{O}) are: 0 (i.e. absence of outliers), 1, 3 and 5. The ranges of magnitudes of outliers (when present) are (0, 0.5), (0, 1) and (0, 2) times the mean value of the dependent variable, y , in a particular model.

To obtain \tilde{a} by minimizing $S_1 = \sum_{i=1}^n \left| y_i - \sum_{j=1}^m a_j x_{ij} \right|$, where \tilde{a} is the LAD estimator of

\mathbf{a} in $y = X\mathbf{a} + \mathbf{e} + \mathbf{O}$, we have used Fair (1974) algorithm. The algorithm is iterative in nature and exploits the Brouwer-Kakutani fixed point theorem. We were able to improve upon the accuracy of the LAD estimator so obtained by application of the *random walk method*, giving us the LADRW estimates.

For every experiment, hundred (100) replicates were used – that is, with fixed $X\mathbf{a}$, disturbances, \mathbf{e} and outliers, \mathbf{O} were generated 100 times. Based on these 100 replicates,

$\text{RMS}_1 = \left[\frac{1}{100} \sum_{k=1}^{100} \left\{ \sum_{j=1}^m (\tilde{a}_{kj} - a_j)^2 \right\} \right]^{0.5}$, \tilde{a} being the LAD estimator of \mathbf{a} , was obtained. The

OLS estimators of \mathbf{a} (that is, \hat{a}) was obtained as $\hat{a}=(X'X)^{-1}X'y$ for each of the 100 replicates and from this $RMS_2 = [\frac{1}{100} \sum_{k=1}^{100} \{\sum_{j=1}^m (\hat{a}_{kj} - a_j)^2\}]^{0.5}$ was obtained. The quotient of the two RMS or $\rho = RMS_2/RMS_1$ was obtained. If, $\rho > 1$, it implies that LAD has outperformed OLS and vice versa. Such 300 experiments were carried out with different parameters. The distributions of the number of experiments according to various parameters of experiments are given below. The figures within the parentheses are the number of experiments carried out.

Size of Model: Model#1 (100), Model#2 (100), Model#3 (100)

Sample Size: Small i.e. 20 observation (150), Larger, i.e. 50 observations (150)

Distributions: Normal (60), Cauchy (60), Gamma (60), B_1 (60), B_2 (60)

Number of Outliers: None (120), One (60), Three (60), Five (60)

Size of Outliers: None/zero (120), Small (90), Large (90).

Findings of Monte Carlo Experiments with Single Equation Linear Models: First we present the frequency distribution of success of different candidate estimators in dominating OLS as tabulated in the tables 1 through 5. The entries in the cells of these tables are the number of success (out of the total number of experimental trials or cases - given in the rightmost columns of the tables) and where the ratio of the number of success to the number of trials (cases) is larger than 0.5, the entry has been highlighted. We observe the following:

(1). Among the different disturbance distributions only Cauchy and B_2 favour LAD (or LADRW) estimators against the OLS estimator. Among the distributions, Cauchy is absolutely dominating.

(2). No outlier or single outlier cases give no edge to the LAD estimator against the OLS estimator, but for larger number (3 or 5) of outliers LAD outperforms OLS. However, LADRW performs better than OLS in presence of outliers (1, 3 or 5) while it has no edge over OLS if no outlier is present. LAD and LADRW outperform OLS for whatever size of outliers.

We conclude therefore that Cauchy or B_2 distributed errors favour LAD estimator against the OLS estimator. Performance of LAD (or LADRW) in case of Cauchy distributed disturbances is spectacular. We observe that out of 60 experiments with Cauchy distributed disturbance term LAD outperforms OLS in 59 cases. LADRW outperforms OLS in 60 out of 60 experiments.

TABLE 1. EFFECT OF DISTRIBUTION

Sl. No.	DISTBUTION	LAD	LADRW	CASES
1	n	18	23	60
2	c	<u>59</u>	<u>60</u>	60
3	g	21	24	60
4	v	10	14	60
5	b	<u>53</u>	<u>52</u>	60

TABLE 2. EFFECT OF SAMPLE SIZE

Sl. No.	SAMPLE SIZE	LAD	LADRW	CASES
1	s	73	81	150
2	l	<u>88</u>	<u>92</u>	150

TABLE 3. EFFECT OF MODEL SIZE

Sl. No.	MODEL SIZE	LAD	LADRW	CASES
1	2	49	<u>58</u>	100
2	4	<u>56</u>	<u>58</u>	100
3	6	<u>56</u>	<u>57</u>	100

TABLE 4. EFFECT OF NO. OF OUTLIERS

Sl. No.	NO. OF OUTLIERS	LAD	LADRW	CASES
1	0	41	41	120
2	1	30	<u>32</u>	60
3	3	<u>42</u>	<u>49</u>	60
4	5	<u>48</u>	<u>51</u>	60
Total (AT LEAST ONE OUTLIER)		<u>120</u>	<u>132</u>	180

TABLE 5. EFFECT OF SIZE OF OUTLIERS

Sl. No.	OUTLIER SIZE	LAD	LADRW	CASES
1	0	10	10	30
2	a	11	10	30
3	b	10	10	30
4	c	10	11	30
Total (ZERO OUTLIERS)		41	41	120
5	s	<u>43</u>	<u>51</u>	90
6	b	<u>77</u>	<u>81</u>	90

NOTE: In 0, a, b, and c categories SIZE of OUTLIERS = 0

Probit Analysis (Single-Equation Models): For a comprehensive analysis of the results of our experiments we construct a variable, R, which takes on only two values, zero or unity. $R_i = 1$ if $\rho (= RMS_2/RMS_1) > 1$, zero otherwise. In other words, R takes on a value of unity if LAD (or LADRW) outperforms OLS, else $R = 0$. We go in for the *Probit analysis* of R. Since we have carried out 300 experiments, R is a vector of 300 elements. This vector may be considered as a sample R from the population R since one may carry out many more experiments to obtain a still larger number of ρ to construct R with many more elements. Then we postulate that the value of R_i is conditioned by the model specification in terms of sample size, model size, distributional specifications of the disturbances and the number and size of outliers that make up the dependent variable y in

$y = X\mathbf{a} + e + O$. Therefore, in the Probit analysis, we have the following explanatory variables:

N (Normal Distribution): A binary variable, taking on a value of unity if the disturbance term follows a normal distribution, zero otherwise.

G (Gamma Distribution): A binary variable, taking on a value of unity if the disturbance term follows a Gamma distribution, zero otherwise.

B₁ (Beta₁ Distribution): A binary variable, taking on a value of unity if the disturbance term follows a Beta₁ distribution, zero otherwise.

B₂ (Beta₂ Distribution): A binary variable, taking on a value of unity if the disturbance term follows a B₂ distribution, zero otherwise.

NOBSRV (Sample size): A binary variable, taking on a value of unity if the number of observations has a larger size = 50, zero otherwise.

COEFF (Size of the model): This variable takes on the value of 2 for a model with two explanatory variables, 4 for a model of four explanatory variables and 6 for a model with 6 explanatory variables.

NOOUT (No. of Outliers): This variable takes on the value of 1 if the disturbance contains one outlier, 3 if the disturbance contains three outliers, 5 if the disturbance contains five outliers, and 0 if the disturbance contains no outlier.

OSIZE (Outliers of a large size): To recapitulate (what has been mentioned earlier), in the experiments we have used four different factors (call k) for generating outliers of different sizes. For small samples we use two values of k (0.5 and 1.0). For large samples we use $k=1.0$ and $k= 2.0$. The size of outliers is $k*\text{mean}(y)$ or k times the mean of the dependent variable. The specified number of outliers are added to

different observations of the dependent variable (y) at randomly selected locations (cases or observation). The value of the decision variable $OSIZE$ takes on these values of k . Thus $OSIZE$ takes on three different values – 0.5, 1.0 and 2.0 depending on the specified conditions.

Binary coding of distribution variables necessitate that any four (out of 5 namely, N , C , G , B_1 and B_2) of them at most can at a time be included in the list of explanatory variables, since all five, if included, make up for a perfect multi-collinearity. In view of the spectacular performance of LAD and LADRW for Cauchy distributed disturbance term (exhibiting 59 and 60 successes respectively out of 60 experiments) we have chosen to drop out Cuachy from the list of explanatory variables. Accordingly, our general Probit Regression model may be described as follows:

$$R_i = k_0 + a_1N + a_2G + a_3B_1 + a_4B_2 + b_1NOBSRV + b_2COEFF + c_1NOUT + c_2OSIZE + error$$

where R_i (binary measure of the success/failure of one of the estimators, namely LAD or LADRW).

The results of Probit Analysis suggest that inclusion of N (normal disturbance) and B_1 (Beta₁ disturbance) together as the explanatory variables in the Probit regression makes the estimation procedure unstable due to ill conditioned variance-covariance matrix of explanatory variables. Hence, only one of them (at a time) may be included in the list of explanatory variables. Accordingly, we find that N , B_1 and G have depressing effect on LAD and LADRW estimators. B_2 gives an edge to LAD (as well as LADRW) estimator over OLS estimator. Presence and size of outliers favour LAD (as well as LADRW) estimator. Sample size ($n = NOBSRV$) and model size ($m = COEFF$) do not matter.

The Design of Monte Carlo Experiments with Multi-Equation Linear Models:

A multi-equation linear econometric model is $YA + XB + E + O = 0$ where E is the disturbance matrix with a specified distribution and O is the sparse matrix of outliers with a specified number of nonzero elements lying within a specified range and spaced randomly over the sample observations of different structural equations. $Y(n,m)$ is the matrix of n sample observations on m endogenous variables and $X(n,k)$ is the (fixed) non-stochastic matrix of n sample observations on k pre-determined (exogenous) variables. The number of equations in the model, m , determines the model size. We have experimented with two sample sizes; $n = 20$ and $n = 50$. Models are of three different sizes; $m = 3$ (Model₁ with 10 parameters), $m = 5$ (Model₂ with 24 parameters) and $m = 7$ (Model₃ with 54 parameters). Disturbances (E) are of five different distribution types; Normal, Cauchy, Gamma(2), Beta₁(2,2) and Beta₂(2,2). Disturbances have two alternative standard deviations, very small (0.001) and sufficiently large (0.3) relative to the mean magnitude of the dependent variable in a particular (say j^{th}) structural equation, y_j . The numbers of outliers (non-zero elements in O_j) are: 0 (i.e. absence of outliers), 1, 2 and 3. The ranges of magnitudes of outliers (when present) are (0, 0.5), (0, 1) and (0, 1.5) times the mean value of the dependent variable, y_j , in the j^{th} equation of the model.

For every experiment fifty (50) replicates were used. Such 630 experiments were carried out with different parameters. **Fair-Schlossmacher** algorithm was used to find the solutions. The distributions of the number of experiments according to various parameters of experiments are given below. The figures within the parentheses are the number of experiments carried out.

Size of Model: Model#1 (210), Model#2 (210), Model#3 (210)

Sample Size: Small i.e. 20 observation (315), Larger, i.e. 50 observations (315)

Distributions: Normal (126), Cauchy (126), Gamma (126), B_1 (126), B_2 (126)

Standard Deviation of Disturbance Vector: Small (180), Large (450)

Number of Outliers: None (360), One (90), Two (90), Three (90)

Size of Outliers: None/zero (360), Small (90), Medium (90), Large (90).

Candidate Estimators of Multi-equation Models: We have six candidate estimators, LS-LS (= 2-SLS of Theil-Basmann), LS-LAD, LAD-LS, LAD-LAD (D2SLAD of Amemiya), LS-GILN (GILS or Khazzoom estimator) and LAD-GILN. The 2-SLS is the referent to compare the performance of the other five estimators.

Findings of Monte Carlo Experiments with Multi-Equation Linear Models: First we present the frequency distribution of success of different candidate estimators in dominating 2-SLS as tabulated in the tables 6 through 11. The entries in the cells of these tables are the number of success (out of the total number of experimental trials or cases - given in the rightmost columns of the tables) and where the ratio of the number of success to the number of trials (cases) is larger than 0.5, the entry has been highlighted.

We observe the following:

Effect of Model Size: For M_3 (7-equations model) LS-LAD and LAD-LAD estimators outperform 2-SLS.

Effect of Sample Size: No estimator outperforms 2-SLS.

Effect of Distribution of Disturbance term: For Cauchy and B_2 (named as B) distributed disturbance terms LAD-LAD and LAD-GILN outperform 2-SLS.

Effect of Size of Std. Deviation of Disturbance term: No estimator outperforms 2-SLS.

Effect of No. of Outliers: For 2 and 3 outliers LAD-LAD outperforms 2-SLS.

Effect of Size of Outliers: For medium and large size outliers LAD-LAD estimator outperforms 2-SLS.

Table 6. Effect of Model Size

	Ls_Lad	Lad_Ls	Lad_Lad	LS_Giln	Lad_Giln	Cases
M ₁	00	04	47	23	61	210
M ₂	12	04	62	40	94	210
M ₃	123	100	153	42	78	210

Table 7. Effect of Sample Size

	Ls_Lad	Lad_Ls	Lad_Lad	LS_Giln	Lad_Giln	Cases
S	91	82	145	59	90	315
L	44	26	117	46	143	315

Table 8. Effect of Distribution

	Ls_Lad	Lad_Ls	Lad_Lad	LS_Giln	Lad_Giln	Cases
N	20	25	32	18	13	126
C	41	21	81	22	85	126
G	16	22	29	29	14	126
V	23	19	34	18	29	126
B	35	21	86	18	92	126

Table 9. Effect of Standard Deviation

	Ls_Lad	Lad_Ls	Lad_Lad	LS_Giln	Lad_Giln	Cases
S	35	14	56	31	70	180
L	100	94	206	80	163	450

Table 10. Effect of No. of Outliers

	Ls_Lad	Lad_Ls	Lad_Lad	LS_Giln	Lad_Giln	Cases
E	35	14	56	29	70	180
0	49	41	57	28	52	180
1	20	19	38	05	32	90
2	14	16	50	17	41	90
3	17	18	51	26	38	90

Table 11. Effect of Size of Outliers

	Ls_Lad	Lad_Ls	Lad_Lad	LS_Giln	Lad_Giln	Cases
0	21	12	24	07	26	60
1	12	07	19	06	21	60
2	12	08	21	10	19	60
3	17	09	21	14	22	60
4	11	11	20	11	16	60
5	11	08	18	09	18	60
S	19	19	26	06	29	90
M	16	18	47	14	40	90
B	16	16	66	28	42	90

Entries are the frequencies of relative norms > 1.

Probit Analysis (Multi-equation Models): For a comprehensive analysis of the results of our experiments, we construct a variable, R, which takes on only two values, zero or

unity. $R_C = 1$ if $\rho (= (RMS_{2-SLS})/(RMS_e)) > 1$, zero otherwise. In other words, R takes on a value of unity if a particular candidate estimator (LS-LAD, LAD-LS, LAD-LAD, LS-GILN and LAD-GILN) outperforms 2-SLS, else $R = 0$. We go in for the *Probit analysis* of R_C . Since we have carried out 630 experiments, R is a vector of 630 elements for each candidate estimator. This vector may be considered as a sample R from the population R since one may carry out many more experiments to obtain a still larger number of ρ to construct R with many more elements. Then we postulate that the value of R_i is conditioned by the model specification in terms of sample size, model size, distributional specifications of the disturbances and the number and size of outliers that make up the dependent variables Y in $YA + XB + E + O$.

Thus, we have five dependent variables, each binary in nature, representing the success (1) or failure (0) of the estimator in outperforming 2-SLS. On the other hand, specifications may be used as the independent variables that explain the variations in the dependent variables. Accordingly, we have used the following independent (explanatory) variables that represent the model specifications:

N (Normal Distribution): A binary variable, taking on a value of unity if the disturbance term follows a normal distribution, zero otherwise.

C (Cauchy Distribution): A binary variable, taking on a value of unity if the disturbance term follows a Cauchy distribution, zero otherwise.

G (Gamma Distribution): A binary variable, taking on a value of unity if the disturbance term follows a Gamma distribution, zero otherwise.

B₁ (Beta₁ Distribution): A binary variable, taking on a value of unity if the disturbance term follows a Beta₁ distribution, zero otherwise.

B₂ (Beta₂ Distribution): A binary variable, taking on a value of unity if the disturbance term follows a B₂ distribution, zero otherwise.

SSD (Disturbance term with a small Std Deviation): A binary variable, taking on a value of 1 if the disturbance term has a small standard deviation = 0.001, zero otherwise.

BSD (Disturbance term with a larger Standard Deviation): A binary variable, taking on a value of unity if the disturbance term has a larger standard deviation = 0.3, zero otherwise.

OSS (Outliers of a small size): A binary variable, taking on a value of unity if the disturbance term has a small size outlier = 0.50 times the mean of the dependent variable, zero otherwise.

OSM (Outliers of a medium size): A binary variable, taking on a value of unity if the disturbance term has a medium size outlier = 1.00 times the mean of the dependent variable, zero otherwise.

OSB (Outliers of a large size): A binary variable, taking on a value of unity if the disturbance term has a medium size outlier = 1.50 times the mean of the dependent variable, zero otherwise.

MODEL (Size of the model): This variable takes on the value of 3 for a model with three equations, 5 for a model of 5 equations and 7 for a model with 7 equations.

OBSERV (Sample size): A binary variable, taking on a value of unity if the number of observations has a larger size = 50, zero otherwise.

NOUT (No. of Outliers): This variable takes on the value of 1 if the disturbance contains one outlier, 2 if the disturbance contains two outliers, 3 if the disturbance contains three outliers, and 0 if the disturbance contains no outlier.

Of distribution variables any four (out of 5 namely, N, C, G, B₁ and B₂) at most can at a time be included in the list of explanatory variables, since all five, if included, make up for a perfect multi-collinearity. Similarly, either SSD or BSD can be included among the explanatory variables since they together make up for a perfect multi-collinearity. Accordingly, our general Probit Regression model may be described as follows:

$$R_i = k_0 + a_1N + a_2C + a_3G + a_4B_1 + a_5B_2 + b_1SSD + b_2BSD + c_1OSS + c_2OSM + c_3OSB + r_1MODEL + r_2OBSERV + r_3NOUT + error$$

where R_i (binary measure of the success/failure of one of the estimators, namely LS-LAD, LAD-LS, LAD-LAD, LS-GILN or LAD-GILN). There would be some zero restrictions on the coefficients (implying absence of some particular explanatory variables), These restrictions are: (i) at least one of the a_j will be zero and (ii) either b_1 or b_2 (or both) will be zero. Other coefficients may be zero or non-zero.

The results of Probit Analysis indicate that except LAD-GILN and LAD-LAD, other estimators do not outperform 2-SLS in any significant manner. The probability of success for LS-LAD, LAD-LS and LS-GILN are 135/630 (21.43%), 108/630 (17.14%) and 111/630 (17.62%) respectively. Even in the experiments with outliers (270 in number or 42.86% of the total number of experiments), the probability of their success is less than 0.5. However, LAD-GILN and LAD-LAD have probability of success 233/630 (36.98%) and 262/630 (41.59%) respectively. This indicated that in presence of outliers

or non-normal disturbance, they outperform 2-SLS in the majority number of cases as we have seen in the tables above.

As table 12 indicates, out of 11 LAD-LAD Probit models, #6 and #7 are the best (followed by LAD-GILN Probit model #5, among the 7 alternative models). LAD-LAD outperforms 2-SLS for Cauchy and B_2 distributed disturbance term with small/large SD containing small/medium/large outliers for smaller as well as larger models of whatever sample size. This result is consistent with the performance of LAD in estimation of the parameters of single equation models. LAD-GILN is the second best estimator.

Table 12. A Summary Table on Performance of Estimators of Multi-Equation Models with their Explanatory Variables [s (x) = Significant (insignificant) at 5% - or less level of Significance]			
Estimator	LAD-LAD		LAD-GILN
Criteria/Variables	Model 6*	Model 7*	Model 5 *
<i>N (Normal)</i>	-	-	-
<i>C (Cauchy)</i>	s	s	S
<i>G (Gamma)</i>	-	-	—
<i>B1 (Beta₁)</i>	-	-	—
<i>B2 (Beta₂)</i>	s	s	S
<i>SSD (Small sd)</i>	s	-	—
<i>BSD (Large sd)</i>	-	s	—
<i>OSS (Small Outlier)</i>	s	s	—
<i>OSM (Med. Outlier)</i>	s	s	S
<i>OSB (Large Outlier)</i>	s	s	S
<i>MODEL (Model size)</i>	s	s	—
<i>OBSERV (Observations)</i>	s	s	S
<i>NOUT (No. of Outliers)</i>	-	-	-
Note: ‘-’ means a variable not appearing in the model.			

IV. Lines of Investigation in Future: Our investigation and its findings are in no way the final. We envisage the following lines of improvement that may decide among the 2-SLS estimator and its LAD-based alternative, especially the LAD-LAD estimator:

Model-based Improvements: Testing the relative performance of LAD estimators for larger models comprising more structural equations and higher degree of over-

identification. That would be more realistic since the real world multi-equation econometric models are usually sizeable.

Mixed Residuals: In our investigation we have assumed that in a given model all endogenous variables have the disturbance terms with identical distribution. But that may not be very realistic. It is likely that in the same model some endogenous variables have disturbances of one kind (say normal) while the others may have disturbances of another kind (say, Cauchy or Gamma). As a result, depending on the mix, the reduced form residuals will have different types of distribution. It is difficult to postulate a-priori as to the performance of the estimators in case of such mixed distribution of disturbances. An exhaustive investigation would be needed to conclude anything substantial regarding the performance of the alternative estimators in the situations of this sort.

Generalization of LAD estimators to the System Methods of Estimation: Our investigation has been limited to a comparative study of the estimators that use limited information and estimate the structural coefficients equation by equation. In doing so, it is tacitly assumed that the disturbances across the structural equations are orthogonal among themselves. This assumption is far from reality. By dropping these assumptions, Theil and Zellner extended 2-SLS to 3-SLS. It is worth investigating if there can be a LAD-based estimator parallel to the 3-SLS, and if yes, how will it perform vis-à-vis the 3-SLS estimator.

Algorithmic Improvements: A linear programming formulation of LAD estimator applicable to multi-equation models is unwieldy. Hence, one has to resort to either of the two types of algorithms; (i) repeated weighted least squares algorithm such as the Fair-Schlossmacher algorithm, or (ii) search algorithms of non-linear programming. Our

experience suggests that none of them is infallible or commendably accurate. These algorithms also partake of the cumulative inaccuracy caused by the sequence of innumerably large number of approximations that must be resorted to when we implement these algorithms on a computer. Additionally, these algorithms are time-consuming due to slow convergence, especially in the vicinity of the optimal point.

It may be fruitful to apply the Genetic algorithms (GA) to search the optimal LAD estimators faster and more successfully. Genetic algorithms, a class of *heuristic* techniques of optimization, were introduced by **Holland** (1975). To develop the method, several important contributions were made by **Goldberg** (1989) and **Wright** (1991). Initially GAs found their applications in engineering problems and multi-criteria decision-making. Of late, some significant works have been done to apply GAs to economics (e.g. **Arifovic** (1994), **Price** (1997), **Cooper** (2000), etc.). However, as **Dirk & Doherr** (2002) note, *the applications of GAs in econometrics are rare*. The only study they mention to be aware of is **Dorsey and Mayer** (1995) that applies GAs to several optimization problems in econometric estimation and carries out Monte Carlo simulations. Dirk & Doherr maintain that especially for semi-parametric estimations GAs may be a useful alternative to the existing optimization algorithms. They demonstrate the applicability of GAs to estimate a *censored least absolute deviation* (CLAD) model. This brings our point home that GAs may be a fruitful alternative to traditional algorithms applied to LAD estimation.

DasGupta, Madhuchhanda (2004) LEAST ABSOLUTE DEVIATION ESTIMATION OF MULT-EQUATION LINEAR ECONOMETRIC MODELS : A STUDY BASED ON MONTE CARLO EXPERIMENTS, Doctoral Dissertation, Dept. of Economics, North Eastern Hill University, Shillong (India); Supervisor – Prof. SK Mishra

APPENDIX-1

A Computer Program to Generate Random Numbers and Show their Frequency Distribution

```
C      MAIN PROGRAM
      DIMENSION X(10000),C(50,2),F(50),Z(50)
      CHARACTER FIL *18, COMMA *2
      COMMA=', '
      WRITE(*,*) 'MAIN PROGRAM FOR SIMULATING RANDOM NUMBERS'
      N=10000
      WRITE(*,*) 'FILE TO STORE FREQUENCY DISTRIBUTION ?'
      READ(*,*) FIL
      WRITE(*,*) 'NO. OF CLASSES [4 < M <=50] ?'
      READ(*,*) M
      IF(M.LT.5) THEN
      WRITE(*,*) 'SINCE M GIVEN BY YOU IS < 5, M=10 ASSUMED'
      M=10
      ENDIF
      WRITE(*,*) 'FEED SEED & SCALE OF RANDOM NUMBER TO GENERATE'
      WRITE(*,*) '(SEED <= 32000 IN MAGNITUDE)'
      READ(*,*) IU,SCALE
      WRITE(*,*) 'FEED ID(0=N;1=C;2=G;3=B2;4=B1) and IG for ID >1'
      READ(*,*) ID
      IF(ID.GT.1) THEN
      WRITE(*,*) 'FEED IG (Shape Parameter) => 1'
      READ(*,*) IG
      ENDIF
      DO 3 I=1,N
      X(I)=ERS(IU,IV,ID,IG)*SCALE
3 CONTINUE
C      FIND MAX AND MIN OF X
      AMX=X(1)
      AMN=X(1)
      DO 8 I=1,N
      IF(X(I).GT.AMX) AMX=X(I)
      IF(X(I).LT.AMN) AMN=X(I)
8 CONTINUE
C      DEFINING CLASSES
      RMN=ABS(AMN)
      IF(RMN.LT.AMX) RMN=AMX
      R=2*RMN/(M-1)
      C(1,1)= -(RMN+R/2.0)
      DO 1 I=2,M
      C(I,1)=C(I-1,1)+R
1 CONTINUE
      DO 2 I=1,M
      C(I,2)=C(I,1)+R
      Z(I)=(C(I,1)+C(I,2))/2.0
      F(I)=0.0
2 CONTINUE
      DO 4 I=1,N
      DO 4 J=1,M
```

```

      IF ((X(I) .LT. C(J,2)) .AND. (X(I) .GE. C(J,1))) F(J)=F(J)+1
4  CONTINUE
      ZF=0.0
      ZZF=0.0
      DO 5 I=1,M
      ZF=ZF+Z(I)*F(I)
      ZZF=ZZF+F(I)*Z(I)*Z(I)
5  CONTINUE
      AM=ZF/N
      SD=(ZZF/N-AM*AM)
      SD=SQRT(SD)
      DO 6 I=1,M
      WRITE(*,9) C(I,1),C(I,2),F(I)
6  CONTINUE
      WRITE(*,*) '-----'
9  FORMAT(1X,F15.2,8H          to,F15.2,3X,F12.0)
      WRITE(*,*) 'MIN and MAX values are:',AM,AMX
      WRITE(*,*) '          MEAN & SD are:',AM,SD
      AM=0.0
      SD=0.0
      DO 7 I=1,N
      AM=AM+X(I)
      SD=SD+X(I)*X(I)
7  CONTINUE
      AM=AM/N
      SD=SD/N-AM*AM
      SD=SQRT(SD)
      WRITE(*,*) 'UNGROUPED MEAN & SD are:',AM,SD
      OPEN(7,FILE=FILE,STATUS='NEW')
c    DO 10 I=1,M
c    WRITE(7,*) C(I,1),COMMA,C(I,2),COMMA,Z(I),COMMA,INT(F(I))
c 10  CONTINUE
c    write(7,*) '-----'
c    DO 11 I=1,M
c    WRITE(7,12) C(I,1),C(I,2),Z(I),INT(F(I))
11  CONTINUE
12  format(3f15.4,i7)
      CLOSE(7)
      END

C  -----
C  This Program generates error term for regression equation
      FUNCTION ERS(IU,IV,ID,IG)
      INTEGER *2 IU,IV
      ER1=ER(IU,IV)
      ER2=ER(IU,IV)
C  -----
C  Normally distributed as average of two normals
C  IF(ID.EQ.0) ERS=(ER1+ER2)/2.0
C  IF(ID.EQ.0) ERS=ER1
C  Cauchy-distributed error as ratio of two std normal variates
C  IF(ID.EQ.1) ERS=(ER1/ER2)
C  Gamma-distributed error as average of IG squared normals
C  ----- IG IS THE SHAPE PARAMETER -----
      ER1=0.0
      ER2=0.0
      DO 2 JG=1,IG
      ER1=ER1+(ER(IU,IV))**2
      ER2=ER2+(ER(IU,IV))**2

```



```

2 CONTINUE
  IF (ID.EQ.2) ERS=(ER1+ER2)/2.0
C   Beta2-distributed as a ratio of square of two normals
  IF (ID.EQ.3) ERS=ER1/ER2
C   Beta1-distributed as  $x/(x+y)$ ,  $x$  and  $y$  being normal
  IF (ID.EQ.4) ERS=ER1/(ER1+ER2)
C   -----
  RETURN
  END
C   -----

C   -----
C   Program to generate  $N(0,s)$  from Rectangular Random Numbers
C   It uses Variate Transformation for this purpose.
  FUNCTION ER(IU,IV)
  INTEGER *2 IU,IV
C   DOUBLE PRECISION ER,RAND,U
  DIMENSION U(2)
  DO 1 I=1,2
    U(I)=RAND(IU,IV)
1 CONTINUE
C   -----
C   IF U1 AND U2 ARE UNIFORMLY DISTRIBUTED RANDOM NUMBERS (0,1),
C   THEN  $X=[(-2*\ln(U1))^{.5}]*(\cos(2*PI*U2))$  IS  $N(0,1)$ 
C    $PI = 4*ARCTAN(1.0) = 2*1.57079632679489$ 
C    $PI*2 = 6.2831853071797$ 
C   -----
C    $ER=DSQRT(-2.0*DLOG(U(1)))*DCOS(U(2)*6.2831853071797)$ 
C    $ER=SQRT(-2.0*ALOG(U(1)))*COS(U(2)*6.2831853071797)$ 
  RETURN
  END
  FUNCTION RAND(IU,IV)
C   DOUBLE PRECISION RAND
  INTEGER *2 IU,IV
  IV=IU*259
  IF (IV.GE.0) GOTO 2
  IV=IV+32767+1
2 RAND=IV
  IU=IV
  RAND=RAND*0.30517578125E-04
  RETURN
  END

```

APPENDIX-2

SOURCE CODES OF COMPUTER PROGRAMS FOR SINGLE EQUATION MODELS

```
C      MAIN PROGRAM =====
COMMON ML, NL, JRRL
DOUBLE PRECISION YO(100), XO(100,10), Y(100), X(100,10), OLSS(10)
DOUBLE PRECISION YC(100), XC(100,10), ER(100), YH(100), A(10), EPS
DOUBLE PRECISION XX(10,10), XY(10), AA(10), B(10), MADS(10), AVB(10)
DOUBLE PRECISION AVA(10), RSQ, QD, AH(10), AVH(10), HOOK(10), HOOKN
DOUBLE PRECISION TOL, TTOL, SCALE, YM, SDEV, SIZE, OLSN, AMADN, RNORM
INTEGER *2 IUX, IU, IV, IUU, SGNS(30)
CHARACTER *11 FIL, RFIL
CHARACTER *6 NAM(10)
C      =====
C      PARAMETERS FOR DIMENSION: CHANGE THEM IF THE PROBLEM IS LARGER
C      EPS=0.00000001
      NL=100
      ML=10
      FIL='MYFIL'
      RFIL='RWMYFIL'
C      CHANGE THE FORMAT IF REQUIRED
10  FORMAT(8F10.2)
C      -----
      WRITE(*,*) 'FEED NO. OF OBSERVATIONS AND NO. OF VARIABLES'
      READ(*,*) N,M
      WRITE(*,*) 'FEED THE TOLERANCE LEVEL FOR ACCURACY (.0001)'
      WRITE(*,*) 'This is required for iterative solution of parameters'
      READ(*,*) TOL
      TTOL=TOL
      WRITE(*,*) 'FEED THE SEED TO GENERATE RANDOM NUMBERS'
      READ(*,*) IU
      WRITE(*,*) 'FEED DIFFERENT SEED RANDOM NUMBERS-RANDOM WALK'
      READ(*,*) IUX
      IUU=IU
      CALL YXGEN(M,N,YC,XC,AA,SCALE,IU)
C      DO 2 I=1,N
C      WRITE(*,10) YC(I), (XC(I,J), J=1,M)
C      2 CONTINUE
      WRITE(*,*) 'FEED STANDARD DEVIATION (SD) OF ERROR TERM AND JRRL.'
      WRITE(*,*) 'SD may not exceed 0.3 else R Squares come poor.'
      WRITE(*,*) 'FOR JRRL=0 SD IS RELATIVE TO 100 ELSE IT IS RELATIVE'
      WRITE(*,*) 'TO AVERAGE VALUE OF Y. Your Options ?'
      READ(*,*) SDEV, JRRL
      YM=0.0
      DO 8 I=1,N
      YM=YM+YC(I)
8  CONTINUE
      YM=YM/FLOAT(N)
      WRITE(*,*) 'AVERAGE Y IS = ',YM
      WRITE(*,*) 'NUMBER OF OUTLIERS TO GENERATE AND THEIR SIZE with'
      WRITE(*,*) 'respect to average Y. Size may not exceed 1.0'
      WRITE(*,*) 'But it is your choice. Size = 2 is quite large'
```

```

READ(*,*) NOL,SIZE
WRITE(*,*) 'FEED THE SIGNS -1 OR 1 FOR EACH OUTLIER'
READ(*,*) (SGNS(J),J=1,NOL)
C -----
WRITE(*,*) 'NO. OF EXPERIMENTS (REPLICATES > = 50, say) ? '
READ(*,*) NTRY
C   WRITE(*,*) 'NAME TWO FILES TO STORE THE RESULTS ?'
C   READ(*,*) FIL,RFIL
OPEN(7,FILE=FIL,STATUS='NEW')
WRITE(7,*) (AA(J),J=1,M)
C   ?????????????????????????????????????????????????????????????
WRITE(*,*) 'CHOOSE ERROR DISTRIBUTION BY ONE OF THE OPTIONS'
WRITE(*,*) 'Normal=0; Cauchy=1; Gamma=2; Beta2=3; Beta1=4'
READ(*,*) ID
IF(ID.GE.2) THEN
WRITE(*,*) 'To generate Gamma or Beta distributed Random Numbers'
WRITE(*,*) 'it requires to generate Gamma from Normal distn.'
WRITE(*,*) 'PARAMETER FOR GAMMA DISTRIBUTION (ig)'
WRITE(*,*) 'It generates G(ig) and Beta1 or Beta2 of (.5ig,.5ig)'
READ(*,*) IG
ENDIF
C   ?????????????????????????????????????????????????????????????
WRITE(*,*) 'SET MAX ITERATION FOR MAD ESTIMATION'
READ(*,*) ITMX
DO 100 IT=1,NTRY
C   *****
DO 1 I=1,N
YO(I)=YC(I)
Y(I)=YC(I)
DO 1 J=1,M
XO(I,J)=XC(I,J)
X(I,J)=XC(I,J)
1 CONTINUE
C   -----
C   -----GENERATION OF ERRORS AND ADDING THEM TO YC(I)-----
CALL ERGEN(N,YO,Y,ER,SDEV,IU,IV,ID,IG)
C   -----
C   WRITE(*,*) ' PRINT OF XC, YC AND YO'
C   DO 6 I=1,N
C   WRITE(*,10) (XC(I,J),J=1,M),YC(I),YO(I)
C   6 CONTINUE
C   PAUSE
C   -----GENERATION OF OUTLIER ERROR(S)-----
IF(NOL.GT.0) CALL OUTLI(YO,Y,N,IU,IV,NOL,SIZE,SGNS)
C   *****
C   -----GOING TO MIN ABS DEVIATION ESTIMATION-----
CALL MAD(YO,XO,Y,X,M,N,TOL,ITMX,AA,XX,XY,A,YH,ER,RSQ,B)
WRITE(7,*) (A(J),J=1,M)
C   NOW RANDOM WALK METHOD WILL TRY TO IMPOVE UPON MAD ESTIMATION
CALL RWALK(A,M,N,YO,XO,IUX,EPS,QD)
QD=QD/FLOAT(N)
C   *****
C   PAUSE
WRITE(7,*) (A(J),J=1,M),RSQ,QD
WRITE(*,99) IT,NTRY
100 CONTINUE
99 FORMAT(1X,/, '----- EXPERIMENT NO.',I5, ' OVER ',I5, ' -----',/)

```

```

CLOSE (7)
OPEN (7, FILE=FILE)
READ (7, *) (AA (J) , J=1, M)
OLSN=0.0
AMADN=0.0
HOOKN=0.0
DO 3 J=1, M
  OLSS (J)=0.0
  MADN (J)=0.0
  HOOK (J)=0.0
  AVA (J)=0.0
  AVB (J)=0.0
  AVH (J)=0.0
3 CONTINUE
DO 4 I=1, NTRY
  READ (7, *) (B (J) , J=1, M)
  READ (7, *) (A (J) , J=1, M)
  READ (7, *) (AH (J) , J=1, M) , RSQ, QD
  DO 4 J=1, M
    OLSS (J)=OLSS (J) + (B (J) -AA (J) ) **2
    MADN (J)=MADN (J) + (A (J) -AA (J) ) **2
    HOOK (J)=HOOK (J) + (AH (J) -AA (J) ) **2
    AVA (J)=AVA (J) +A (J)
    AVB (J)=AVB (J) +B (J)
    AVH (J)=AVH (J) +AH (J)
4 CONTINUE
CLOSE (7)
DO 5 J=1, M
  OLSN=OLSN+OLSS (J)
  AMADN=AMADN+MADN (J)
  HOOKN=HOOKN+HOOK (J)
  AVA (J)=AVA (J) /NTRY
  AVB (J)=AVB (J) /NTRY
  AVH (J)=AVH (J) /NTRY
5 CONTINUE
RNORM=DSQRT (OLSN/AMADN)
HNORM=DSQRT (OLSN/HOOKN)
OPEN (9, FILE=RFIL, STATUS='NEW')
C   WRITE (9, *) 'TRUE COEFFICIENTS'
C   WRITE (9, *) (AA (J) , J=1, M)
C   WRITE (9, *) 'MEAN OLS COEFFICIENTS'
C   WRITE (9, *) (AVB (J) , J=1, M)
C   WRITE (9, *) 'MEAN MAD COEFFICIENTS'
C   WRITE (9, *) (AVA (J) , J=1, M)
C   WRITE (9, *) 'OLS ROOT MEAN SQUARE'
C   WRITE (9, *) (DSQRT (OLSS (J) /FLOAT (NTRY) ) , J=1, M)
C   WRITE (9, *) 'MAD ROOT MEAN SQUARE'
C   WRITE (9, *) (DSQRT (MADN (J) /FLOAT (NTRY) ) , J=1, M)
C   WRITE (9, *) 'RATIO OF RMS OLS/MAD'
C   WRITE (9, *) (DSQRT (OLSS (J) /MADN (J) ) , J=1, M)
C   WRITE (9, *) 'OVERALL RATIO OF NORMS (EFFICIENCY OF MAD) = ', RNORM
  NAM (1)='TRUEC'
  NAM (2)='MOLSC'
  NAM (3)='MMADC'
  NAM (4)='LSRMS'
  NAM (5)='MDRMS'
  NAM (6)='LMRMS'

```

```

      NAM(7)='GLRMS'
      NAM(8)='RWMDC'
      NAM(9)='RWRMS'
      NAM(10)='RWLRMS'
      WRITE(9,*)'FILE NAME = ',RFIL
      WRITE(9,*)'N - M - TOL - IUU - SDEV - NOL - SIZE - NTRY-
1 ID - IG - SCALE'
      WRITE(9,12) N,M,TTOL,IUU,SDEV,NOL,SIZE,NTRY,ID,IG,SCALE
12  FORMAT(1X,2I4,F7.4,I5,F7.2,I5,F6.2,I6,2I3,F5.0)
      WRITE(9,*)'SIGNS OF OUTLIERS',(SGNS(J),J=1,NOL)
      WRITE(9,11) NAM(1),(AA(J),J=1,M)
      WRITE(9,11) NAM(2),(AVB(J),J=1,M)
      WRITE(9,11) NAM(3),(AVA(J),J=1,M)
      WRITE(9,11) NAM(8),(AVH(J),J=1,M)
      WRITE(9,11) NAM(4),(DSQRT(OLSS(J)/FLOAT(NTRY)),J=1,M)
      WRITE(9,11) NAM(5),(DSQRT(MADS(J)/FLOAT(NTRY)),J=1,M)
      WRITE(9,11) NAM(9),(DSQRT(HOOK(J)/FLOAT(NTRY)),J=1,M)
      WRITE(9,11) NAM(6),(DSQRT(OLSS(J)/MADS(J)),J=1,M)
      WRITE(9,11) NAM(10),(DSQRT(OLSS(J)/HOOK(J)),J=1,M)
      WRITE(9,11) NAM(7),RNORM,HNORM
11  FORMAT(1X,A5,2X,15F9.3)
      CLOSE(9)
      WRITE(*,*)'RESULTS STORED IN FILES:',FIL,RFIL
      END

C  =====
C  This Program generates X and Y for Simulation
      SUBROUTINE YXGEN(M,N,YO,XO,AA,SCALE,IU)
      COMMON ML,NL,JRRL
      INTEGER *2 IU,IV
      DOUBLE PRECISION YO(NL),XO(NL,ML),AA(ML),SCALE,R
      WRITE(*,*)'FEED THE COEFFICIENTS'
      READ(*,*)(AA(J),J=1,M)
      WRITE(*,*)'FEED SCALING FACTOR FOR X'
      READ(*,*) SCALE
      DO 1 I=1,N
      DO 2 J=1,M-1
      R=RAND(IU,IV)*SCALE
2  XO(I,J)=R
      XO(I,M)=1.0
1  CONTINUE
      DO 3 I=1,N
      YO(I)=0.0
      DO 3 J=1,M
      YO(I)=YO(I)+XO(I,J)*AA(J)
3  CONTINUE
      RETURN
      END

C  -----
C  This Program generates error term for regression equation
      SUBROUTINE ERGEN(N,YO,Y,ER,SDEV,IU,IV,ID,IG)
      COMMON ML,NL,JRRL
      INTEGER *2 IU,IV
      DOUBLE PRECISION YO(NL),Y(NL),ER(NL),ER1,ER2,ERNORM,S,SS,YM
      DOUBLE PRECISION SDEV
      DO 1 I=1,N
      ER1=ERNORM(IU,IV)

```

```

      ER2=ERNORM(IU, IV)
C -----
C Normally distributed as average of two normals
  IF(ID.EQ.0) ER(I)=(ER1+ER2)/2.0
C Cauchy-distributed error as ratio of two std normal variates
  IF(ID.EQ.1) ER(I)=(ER1/ER2)
C Gamma-distributed error as average of IG squared normals
  ER1=0.0
  ER2=0.0
  DO 2 JG=1, IG
    ER1=ER1+(ERNORM(IU, IV))**2
    ER2=ER2+(ERNORM(IU, IV))**2
2 CONTINUE
  IF(ID.EQ.2) ER(I)=(ER1+ER2)/2.0
C Beta2-distributed as a ratio of square of two normals
  IF(ID.EQ.3) ER(I)=ER1/ER2
C Beta1-distributed as  $x/(x+y)$ ,  $x$  and  $y$  being normal
  IF(ID.EQ.4) ER(I)=ER1/(ER1+ER2)
C -----
1 CONTINUE
  S=0
  SS=0
  YM=0.0
  DO 3 I=1, N
    S=S+ER(I)
    SS=SS+ER(I)**2
    YM=YM+YO(I)
3 CONTINUE
  YM=YM/FLOAT(N)
  IF(JRRL.EQ.0) YM=100.0
  S=S/FLOAT(N)
  SS=DSQRT(SS/N-S**2)
  DO 4 I=1, N
    ER(I)=(ER(I)-S)/SS*SDEV*YM
    YO(I)=YO(I)+ER(I)
    Y(I)=YO(I)
4 CONTINUE
  RETURN
  END
C -----
C This Program generates Outliers for MAD estimation
  SUBROUTINE OUTLI(YO, Y, N, IU, IV, NOL, SIZE, SGNS)
  COMMON ML, NL, JRRL
  INTEGER *2 IU, IV, SGNS(30)
  DOUBLE PRECISION YO(NL), Y(NL), YM, EJ, SIZE
C -----
  YM=0.0
  DO 1 I=1, N
    YM=YM+YO(I)
1 CONTINUE
  YM=YM/FLOAT(N)
C WRITE(*,*) 'AVERAGE Y IS = ', YM
  DO 3 NT=1, NOL
2 J=RAND(IU, IV)*(N+1)
  IF(J.EQ.0) GO TO 2
  EJ=RAND(IU, IV)*YM*SIZE*SGNS(NT)
  YO(J)=YO(J)+EJ

```

```

C      WRITE(*,*) 'J,EJ,Y(J),YO(J)',J,EJ,Y(J),YO(J)
      Y(J)=YO(J)
3 CONTINUE
C      -----
      RETURN
      END
C      -----
C      Central Program for MAD estimation using Iterated Weighted LS
SUBROUTINE MAD(YO,XO,Y,X,M,N,TOL,ITMX,AA,XX,XY,A,YH,ER,RSQ,B)
COMMON ML,NL,JRRL
DOUBLE PRECISION YO(NL),XO(NL,ML),Y(NL),X(NL,ML),ER(NL)
DOUBLE PRECISION XX(ML,ML),XY(ML),YH(NL),AA(ML),B(ML),C(10)
DOUBLE PRECISION A(ML),RSQ,ABSUM,ASUM,TOL,EPSY
10 FORMAT(1X,6F13.2)
   EPSY=0.0001
   DF=FLOAT(N-M)
   ITER=0
   CALL OLS(YO,XO,Y,X,N,M,AA,XX,XY,A,YH,ER,RSQ)
   WRITE(7,*) (A(J),J=1,M)
   WRITE(*,*) 'ORDINARY LEAST SQUARES ESTIMATES'
   WRITE(*,*) (A(J),J=1,M)
   WRITE(*,*) 'LEAST SQUARES BASED RSQUARE = ',RSQ
   DO 1 J=1,M
     C(J)=A(J)
1   B(J)=A(J)
100 ITER=ITER+1
    IF(ITER.GT.ITMX) THEN
      WRITE(*,*) 'NO CONVERGENCE IN ITERATIONS = ',ITMX
      GOTO 7
    ENDIF
    ABSUM=0.0
    DO 2 I=1,N
      ER(I)=DABS(ER(I))
      ABSUM=ABSUM+ER(I)
    W=0.0
    IF(ER(I).GT.EPSY) THEN
      W=1.0/ER(I)
    ENDIF
    Y(I)=YO(I)*W
    DO 3 J=1,M
      X(I,J)=XO(I,J)*W
3   C(J)=A(J)
2 CONTINUE
    IF(ITER.EQ.1) ERSUM=ABSUM
    CALL OLS(YO,XO,Y,X,N,M,AA,XX,XY,A,YH,ER,RSQ)
    ASUM=0
    DO 4 I=1,N
      ER(I)=DABS(ER(I))
4   ASUM=ASUM+ER(I)
    DO 6 J=1,M
      IF(DABS(C(J)-A(J)).GT.TOL) GOTO 100
6 CONTINUE
7 WRITE(*,*) 'OLS COEFFICIENTS'
  WRITE(*,*) (B(J),J=1,M)
  WRITE(*,*) 'MAD COEFFICIENTS'
  WRITE(*,*) (A(J),J=1,M)
  ERSUM=ERSUM/FLOAT(N)

```

```

ASUM=ASUM/FLOAT(N)
RSQ=ASUM
WRITE(*,*) 'TOTAL NUMBER OF ITERATIONS PERFORMED ',ITER-1
WRITE(*,*) 'MEAN OLS AND MAD DEVIATIONS ',ERSUM,ASUM
RETURN
END
C -----
C Central Program for Least Squares Estimation of Regression Eqn.
SUBROUTINE OLS(YO,XO,Y,X,N,M,AA,XX,XY,A,YH,ER,RSQ)
COMMON ML,NL,JRRL
DOUBLE PRECISION Y(NL),X(NL,ML),A(ML),ER(NL),YO(NL)
DOUBLE PRECISION XX(ML,ML),XY(ML),YH(NL),AA(ML),XO(NL,ML)
DOUBLE PRECISION RSQ,S,SS,SE,ES,VYH
DF=FLOAT(N-M)
DO 1 J=1,M
XY(J)=0.0
DO 2 K=J,M
XX(J,K)=0.0
DO 10 I=1,N
10 XX(J,K)=XX(J,K)+X(I,J)*X(I,K)
XX(K,J)=XX(J,K)
2 CONTINUE
DO 1 I=1,N
XY(J)=XY(J)+X(I,J)*Y(I)
1 CONTINUE
CALL MINV(XX,M)
DO 3 J=1,M
A(J)=0.0
DO 3 K=1,M
3 A(J)=A(J)+XX(J,K)*XY(K)
SE=0.0
ES=0.0
S=0.0
SS=0.0
DO 4 I=1,N
YH(I)=0.0
DO 5 J=1,M
5 YH(I)=YH(I)+XO(I,J)*A(J)
S=S+YH(I)
SS=SS+YH(I)*YH(I)
ER(I)=YO(I)-YH(I)
SE=SE+ER(I)
ES=ES+ER(I)*ER(I)
4 CONTINUE
S=S/FLOAT(N)
SE=SE/FLOAT(N)
VYH=SS/FLOAT(N)-S*S
RSQ=VYH/(VYH+(ES/FLOAT(N)-SE*SE))
RETURN
END
C -----
C Program to generate N(0,s) from Rectangular Random Numbers
C It uses Variate Transformation for this purpose.
FUNCTION ERNORM(IU,IV)
DOUBLE PRECISION U(2),ERNORM,R
INTEGER *2 IU,IV
R=0.0

```



```

      DO 1 I=1,2
      U(I)=RAND(IU,IV)
1 CONTINUE
C -----
C   IF U1 AND U2 ARE UNIFORMLY DISTRIBUTED RANDOM NUMBERS (0,1),
C   THEN X=[(-2*ln(U1))**.5]*(COS(2*PI*U2) IS N(0,1)
C   PI = 4*ARCTAN(1.0)= 2*1.57079632679489
C   PI*2 = 6.2831853071797
C -----
      ERNORM=DSQRT(-2.0*DLOG(U(1)))*DCOS(U(2)*6.2831853071797)
      RETURN
      END
C -----
C   Generates Rectangular (0,1) Random Numbers
      FUNCTION RAND(IU,IV)
      INTEGER *2 IU,IV
      IV=IU*259
      IF(IV.GE.0) GOTO 2
      IV=IV+32767+1
2 RAND=IV
      IU=IV
      RAND=RAND*0.3051851E-04
      RETURN
      END
C ----- MATRIX INVERSION -----
      SUBROUTINE MINV(AAA,N)
      INTEGER P
      DOUBLE PRECISION A,B,S,G,Y,Z,W,PP,PP1,PP2,PP3,PP4,PP5,PP6
      DOUBLE PRECISION SKID,SKNORM
      DOUBLE PRECISION AAA(10,10),CII(10,10),X(10,10)
      DIMENSION A(10,10),B(10),S(10),Y(10),Z(10),W(10)
      DO 1234 I=1,N
      DO 1234 J=1,N
1234 A(I,J)=AAA(I,J)
      DO 58 I=1,N
      W(I)=B(I)
      DO 58 J=1,N
58 X(I,J)=A(I,J)
      PP1=0
      NN=N*N
      DO 2 I=1,N
      DO 2 J=1,N
2 PP1=PP1+DABS(A(I,J))
      PP2=PP1/NN
      INDEX=0
      DO 33 I=1,N
33 S(I)=0
      DO 506 J=1,N
      DO 506 I=1,N
      IF (I.GT.J) GO TO 503
      PP=0
      I1=I-1
      DO 501 P=1,I1
      IF (I1.LT.1) GO TO 502
501 PP=PP+A(I,P)*A(P,J)
      A(I,J)=A(I,J)-PP
502 IF (I.NE.J) GO TO 506

```

```

        IF (DABS(A(J,J)).LT.(PP2*.1D-08)) GO TO 507
        GO TO 506
503 PP=0
        J1=J-1
        DO 504 P=1,J1
        IF (J1.LT.1) GO TO 505
504 PP=PP+A(I,P)*A(P,J)
505 Y(I)=(A(I,J)-PP)/A(J,J)
        IF (I.EQ.N) GO TO 508
        GO TO 506
508 JJ1=J+1
        DO 1 P=JJ1,N
        IF (DABS(Y(P)).GT.(10.0*PP1)) GO TO 507
        1 CONTINUE
        DO 4 L=JJ1,N
        4 A(L,J)=Y(L)
        GO TO 506
C 507 I=J
507 S(J)=PP2
        A(J,J)=A(J,J)+PP2
        INDEX=INDEX+1
506 CONTINUE
        N1=N-1
        DO 17 J=1,N1
        J1=J+1
        DO 17 I=J1,N
        PP=0
        I1=I-1
        IF (J1.GT.I1) GO TO 17
        DO 18 P=J1,I1
18 PP=PP+A(I,P)*A(P,J)
17 A(I,J)=-A(I,J)-PP
21 DO 84 L=1,N
        J=N-L+1
        DO 84 M=1,J
        I=J-M+1
        IF ((I+1).GT.J) GO TO 71
        I1=I+1
        PP=0
        DO 70 P=I1,J
70 PP=PP+A(I,P)*A(P,J)
        A(I,J)=-PP/A(I,I)
        GO TO 84
71 A(I,J)=1.0/A(I,J)
84 CONTINUE
        N1=N-1
        DO 24 J=1,N1
        DO 24 I=1,N
        IF (I.GT.J) GO TO 27
        PP=0
        J1=J+1
        DO 26 P=J1,N
26 PP=PP+A(I,P)*A(P,J)
        A(I,J)=A(I,J)+PP
        GO TO 24
27 PP=0
        DO 28 P=I,N

```

```

28 PP=PP+A(I,P)*A(P,J)
   A(I,J)=PP
24 CONTINUE
   IF (INDEX.EQ.0) GO TO 59
   DO 121 L=1,N
   K=N-L+1
   IF (DABS(S(K)).LT. 0.1D-08) GO TO 121
   DO 90 J=1,N
   Z(J)=A(K,J)
90 Y(J)=A(J,K)
   G=1.0-S(K)*A(K,K)
   IF (DABS(G).LT. 0.1D-13) GO TO 7178
   DO 91 I=1,N
   DO 91 J=1,N
91 A(I,J)=A(I,J)+S(K)*Y(I)*Z(J)/G
121 CONTINUE
59 DO 45 I=1,N
   PP=0
   DO 46 P=1,N
46 PP=PP+A(I,P)*B(P)
45 Y(I)=PP
   DO 47 I=1,N
47 B(I)=Y(I)
   GO TO 1601
7178 CONTINUE
1601 PP=0
   PP1=0
   PP2=0
   DO 60 I=1,N
   PP2=PP2+W(I)*W(I)
   DO 60 J=1,N
   PP=PP+A(I,J)*A(I,J)
60 PP1=PP1+X(I,J)*X(I,J)
   PP=DSQRT(PP)
   PP1=DSQRT(PP1)
   PP2=DSQRT(PP2)
   DO 57 I=1,N
   PP4=0
   DO 73 P=1,N
73 PP4=PP4+X(I,P)*B(P)
57 W(I)=W(I)-PP4
   PP3=0
   DO 56 I=1,N
56 PP3=PP3+W(I)*W(I)
   PP3=DSQRT(PP3)
   DO 55 I=1,N
   DO 50 J=1,N
   PP4=0
   DO 54 P=1,N
54 PP4=PP4+X(I,P)*A(P,J)
50 W(J)=PP4
   DO 55 P=1,N
55 X(I,P)=W(P)
   DO 53 I=1,N
53 X(I,I)=1.0-X(I,I)
   PP4=0
   DO 52 I=1,N

```

```

      DO 52 J=1,N
52  PP4=PP4+X(I,J)*X(I,J)
      PP4=DSQRT(PP4)
      PP5=PP3/PP1
      PP6=PP*PP4*PP2/(1.0-PP4)
      DO 1236 I=1,N
      DO 1236 J=1,N
      CII(I,J)=0.0
      DO 1236 K=1,N
      CII(I,J)=CII(I,J)+AAA(I,K)*A(K,J)
1236 CONTINUE
      SKNORM=0.0
      DO 1237 I=1,N
      DO 1237 J=1,N
      SKID=0.0
      IF(I.EQ.J) SKID=1.0
      SKNORM=SKNORM+(CII(I,J)-SKID)**2
1237 CONTINUE
      SKNORM=DSQRT(SKNORM/(N*N))
C -----
C   IF(SKNORM.GT.0.0000001) THEN
C   WRITE(*,*) 'INV MATRIX IS INACCURATE. NORM=', SKNORM
C   PAUSE 'PROCEED OR ABORT'
C   ENDIF
C -----
1238 FORMAT(10F8.5)
      DO 1235 I=1,N
      DO 1235 J=1,N
1235 AAA(I,J)=A(I,J)
      RETURN
      END
C -----
C   SUBROUTINE RWALK(A,M,NV,YO,XO,IU,EPS,VO)
C   RANDOM WALK METHOD TO FIND MIN OF A MULTIVARIABLE FUNCTION
C   INTEGER *2 IU,IV
C   DOUBLE PRECISION A(10),R(10),AR(10),YO(100),XO(100,10)
C   DOUBLE PRECISION LAMBDA,EPS,F,VO,VR,RNORM,YH
C   LAMBDA=2.0
C   ITMAX=200
C   Initialisation of decision variables
C   DO 7 I=1,M
C   7 A(I)=0.0
C ----- FUNCTION EVALUATION -----
      F=0.0
      DO 11 I=1,NV
      YH=0.0
      DO 12 J=1,M
12  YH=YH+XO(I,J)*A(J)
      F=F+DABS(YO(I)-YH)
11 CONTINUE
      VO=F
C -----
      IT=0
200 IT=IT+1
      IF(IT.GT.100) THEN
      WRITE(*,*) 'NO CONVERGENCE IN 1000 ITERATIONS'
      GOTO 1000

```

```

      ENDIF
      LAMBDA=LAMBDA/2.0
      IMP=0
      DO 100 II=1,ITMAX
C      GENERATE M UNIFORMLY DISTRIBUTED RANDOM NUMBERS (-1,1)
150 DO 2 I=1,M
      R(I)=2.0*(RAND(IU,IV)-0.5)
      2 CONTINUE
C      NORMALISE THE RANDOM NUMBERS
      RNORM=0.0
      DO 3 I=1,M
      RNORM=RNORM+R(I)**2
      3 CONTINUE
      RNORM=DSQRT(RNORM)
      IF(RNORM.GT.1.0) GOTO 150
      DO 4 I=1,M
      R(I)=R(I)/RNORM
      4 CONTINUE
C      ADD RANDOM NUMBERS TIMES LAMBDA TO A VECTOR
      DO 5 I=1,M
      AR(I)=A(I)+LAMBDA*R(I)
      5 CONTINUE
C      ----- FUNCTION EVALUATION -----
      F=0.0
      DO 13 I=1,NV
      YH=0.0
      DO 14 J=1,M
14      YH=YH+XO(I,J)*AR(J)
      F=F+DABS(YO(I)-YH)
13 CONTINUE
      VR=F
C      -----
      IF(VR.LT.VO) THEN
      VO=VR
      DO 6 I=1,M
      A(I)=AR(I)
      6 CONTINUE
      IMP=1
      ENDIF
100 CONTINUE
      IF((IMP.EQ.0).OR.(LAMBDA.GT.EPS)) GOTO 200
1000 CONTINUE
C      WRITE(*,*) 'OPTIMAL VALUE =',VO,' LAMBDA =',LAMBDA
C      WRITE(*,*) 'OPTIMUM A =', (A(I),I=1,M)
      RETURN
      END

```

DasGupta, Madhuchhanda (2004) LEAST ABSOLUTE DEVIATION ESTIMATION OF MULT-EQUATION LINEAR ECONOMETRIC MODELS : A STUDY BASED ON MONTE CARLO EXPERIMENTS, Doctoral Dissertation, Dept. of Economics, North Eastern Hill University, Shillong (India); Supervisor – Prof. SK Mishra

APPENDIX-3

SOURCE CODES OF COMPUTER PROGRAMS FOR MULTI-EQUATION MODELS

DDGEN.BAT

```
PROG0.EXE
CALL PGRUN.BAT
```

PGRUN.BAT

```
DEL RESULTX
PROG1.EXE
PROG2.EXE
PROG3.EXE
COPY NORMS+GILN RESULTX
ECHO 'RESULTS ARE STORED IN FILE RESULTX'
```

C PROG0.FOR

```
C        MAIN DATA GENERATOR FOR 2-SLS, GILS AND MAD SIMULATION
C        PARAMETER (NV=100, NKV=15, NHK=15, NHV=10, IRL=4)
C        COMMON NKV, NHV, NV, NHK, NO, IRL
C        INTEGER *2 IX, IY, YORD(10,10), ZORD(10,10), IFA, JFA, MZO
C        DOUBLE PRECISION A(10,10), B(10,10), DIRCL
C        DOUBLE PRECISION PIE(10,10), Y(50,10), Z(50,10), AS(10,10)
C        CHARACTER *6 FILT(6)
C        CHARACTER *6 FZ, FY, FO, TC, PRM
C        REAL NC, NCA, NCB
C        PARAMETERS -----
C        NV=50
C        NHV=10
C        NKV=10
C        IRL=8
C        FILT(1)='FILZ'
C        FILT(2)='FILY'
C        FILT(3)='YORD'
C        FILT(4)='TRUC'
C        FILT(5)='PARA'
C        -----
C        WRITE(*,*) 'FEED NO. OF EQNS. IN SYSTEM, NO. OF EXOG. VARIABLES'
C        READ(*,*) NH, NK
C        WRITE(*,*) 'NOW FEED YORDij, ZORDik FOR ROW i ALL ELEMENTS'
C        DO 55 I=1, NH
C        READ(*,*) (YORD(I, J), J=1, NH), (ZORD(I, J), J=1, NK)
C        55 CONTINUE
C        ---TEST FOR ABSENCE OF AN EXOGENOUS VARIABLE FROM THE MODEL---
C        IFA=0
C        DO 155 J=1, NK
C        MZO=0
C        DO 156 I=1, NH
C        156 MZO=MZO+ZORD(I, J)
C        IF(MZO.EQ.0) THEN
C        WRITE(*,*) 'ABSENT FROM THE MODEL IS EXOGEN VARIABLE #', J
```

```

        IFA=1
        ENDIF
155 CONTINUE
        IF (IFA.NE.0) STOP 'MODEL IS FAULTY. CHANGE IT'
C ----- TEST FOR IDENTIFIABILITY OF EQUATIONS -----
        JFA=0
        DO 157 I=1,NH
            IFA=0
            DO 158 J=1,NH
                IF (YORD(I,J).EQ.1) IFA=IFA+1
158 CONTINUE
            MZO=0
            DO 159 J=1,NK
                IF (ZORD(I,J).EQ.0) MZO=MZO+1
159 CONTINUE
            IF (IFA.EQ.MZO) WRITE(*,*) 'EXACTLY IDENTIFIED Eqn #',i
            IF (IFA.LT.MZO) WRITE(*,*) 'OVER-IDENTIFIED Eqn #',i
            IF (IFA.GT.MZO) THEN
                WRITE(*,*) 'UNDER-IDENTIFIED Eqn #',i
                WRITE(*,*) 'CHANGE THE MODEL SUITABLY. THIS IS IMPORTANT'
                JFA=1
            ENDIF
157 CONTINUE
            IF (JFA.EQ.1) STOP 'RETHINK OVER YOUR MODEL'
C -----
        OPEN(11,FILE=FILT(3),STATUS='NEW')
        DO 56 I=1,NH
            WRITE(11,*) (YORD(I,J),J=1,NH), (ZORD(I,J),J=1,NK)
56 CONTINUE
        CLOSE(11)
C -----
        WRITE(*,*) 'SEED TO GENERATE RANDOM STRUCTURAL COEFFICIENTS ?'
        READ(*,*) IX
        WRITE(*,*) 'FEED SCALING FACTORS FOR X, A AND B'
        READ(*,*) NC,NCA,NCB
        DO 44 I=1,NH
            WRITE(*,45) (YORD(I,J),J=1,NH), (ZORD(I,J),J=1,NK)
44 CONTINUE
45 FORMAT(1X,25I3)
        DO 33 I=1,NH
            DO 33 J=1,NH
                A(J,I) = (YORD(I,J) * (RAND(IX,IY)+RAND(IX,IY)) * NCA)
                IF (YORD(I,J).EQ.-1.0) A(J,I) = (-1.0)
                AS(J,I) = A(J,I)
33 CONTINUE
        WRITE(*,*) ' A COEFFICIENT MATRIX'
        DO 50 I=1,NH
            WRITE(*,*) (A(J,I),J=1,NH)
50 CONTINUE
        DO 34 I=1,NH
            DO 34 J=1,NK
                B(J,I) = (ZORD(I,J) * (RAND(IX,IY)+RAND(IX,IY)) * NCB)
34 CONTINUE
        CALL MINV(A,NH)
        DO 35 I=1,NH
            DO 35 J=1,NK
                PIE(J,I) = 0.0

```

```

DO 35 K=1,NH
PIE (J, I)=PIE (J, I) -B (J, K) *A (K, I)
35 CONTINUE
40 FORMAT (10D16.7)
WRITE (*, *) ' PIE - Reduced Form - MATRIX'
OPEN (11, FILE=FILT (4), STATUS='NEW', ACCESS='DIRECT', RECL=IRL)
IRCL=1
DO 42 I=1, NH
WRITE (*, 40) (PIE (J, I), J=1, NK)
DO 42 J=1, NK
IRCL=IRCL+1
WRITE (11, REC=IRCL) PIE (J, I)
42 CONTINUE
420 FORMAT (D15.8)
DO 22 I=1, NH
DO 22 J=1, NH
IRCL=IRCL+1
WRITE (11, REC=IRCL) AS (J, I)
22 CONTINUE
WRITE (*, *) ' B COEFFICIENT MATRIX'
DO 41 I=1, NH
WRITE (*, 40) (B (J, I), J=1, NK)
DO 41 J=1, NK
IRCL=IRCL+1
WRITE (11, REC=IRCL) B (J, I)
41 CONTINUE
DIRCL=(FLOAT (IRCL))
WRITE (11, REC=1) DIRCL
CLOSE (11)
WRITE (*, *) 'FEED SEED TO GENERATE EXOGENOUS VARIABLES x'
READ (*, *) IX
WRITE (*, *) 'HOW MANY OBSERVATIONS TO GENERATE ?'
READ (*, *) NN
DO 61 J=1, NK-1
DO 61 I=1, NN
Z (I, J)=(INT (RAND (IX, IY) *NC*100.0)/100.0)
Z (I, NK)=(1.0)
61 CONTINUE
DO 62 I=1, NN
DO 62 J=1, NH
Y (I, J)=0
DO 62 K=1, NK
Y (I, J)=Y (I, J) -Z (I, K) *PIE (K, J)
62 CONTINUE
IRN=IRL*NN
OPEN (7, FILE=FILT (2), STATUS='NEW', ACCESS='DIRECT', RECL=IRN)
DO 71 J=1, NH
WRITE (7, REC=J) (Y (I, J), I=1, NN)
71 CONTINUE
421 FORMAT (10D16.6)
CLOSE (7)
OPEN (8, FILE=FILT (1), STATUS='NEW', ACCESS='DIRECT', RECL=IRN)
DO 72 J=1, NK
WRITE (8, REC=J) (Z (I, J), I=1, NN)
72 CONTINUE
CLOSE (8)
WRITE (*, *) ' DATA GENERATION OVER. RUNS FOR EXPERIMENTS'

```



```

OPEN(7, FILE=FILT(5), STATUS='NEW')
WRITE(7, 73) NH, NK, NN, IRL
WRITE(7, *) ' NH NK NN IRL'
CLOSE(7)
73 FORMAT(1X, 4I4)
CALL PYZ(NN, NH, NK, Y, Z, FILT)
END
C ----- UNIFORM RANDOM NUMBER GENERATION BLOCK -----
C Generates Rectangular (0,1) Random Numbers
FUNCTION RAND(IU, IV)
INTEGER *2 IU, IV
IV=IU*259
IF(IV.GE.0) GOTO 2
IV=IV+32767+1
2 RAND=IV
IU=IV
RAND=RAND*0.3051851E-04
RETURN
END
C ----- MATRIX INVERSION BLOCK -----
SUBROUTINE MINV(A, N)
INTEGER P
DOUBLE PRECISION A, B, S, X, Y, Z, W, PP1, PP2, PP3, PP4, PP5, PP6
DOUBLE PRECISION G, PP
C DOUBLE PRECISION C, P1, R
DIMENSION A(10, 10), B(10), S(10), Y(10), Z(10), W(10), X(10, 10)
DO 100 I=1, N
B(I)=1.0
100 CONTINUE
DO 58 I=1, N
W(I)=B(I)
DO 58 J=1, N
58 X(I, J)=A(I, J)
PP1=0
NN=N*N
DO 2 I=1, N
DO 2 J=1, N
2 PP1=PP1+DABS(A(I, J))
PP2=PP1/NN
INDEX=0
DO 33 I=1, N
33 S(I)=0
DO 506 J=1, N
DO 506 I=1, N
IF (I.GT.J) GO TO 503
PP=0
I1=I-1
DO 501 P=1, I1
IF (I1.LT.1) GO TO 502
501 PP=PP+A(I, P)*A(P, J)
A(I, J)=A(I, J)-PP
502 IF (I.NE.J) GO TO 506
IF (DABS(A(J, J)).LT.(PP2*.1D-08)) GO TO 507
GO TO 506
503 PP=0
J1=J-1
DO 504 P=1, J1

```

```

      IF (J1.LT.1) GO TO 505
504 PP=PP+A(I,P)*A(P,J)
505 Y(I)=(A(I,J)-PP)/A(J,J)
      IF (I.EQ.N) GO TO 508
      GO TO 506
508 JJ1=J+1
      DO 1 P=JJ1,N
      IF (DABS(Y(P)).GT.(10.0*PP1)) GO TO 507
1 CONTINUE
      DO 4 L=JJ1,N
4 A(L,J)=Y(L)
      GO TO 506
C 507 I=J
507 S(J)=PP2
      A(J,J)=A(J,J)+PP2
      INDEX=INDEX+1
506 CONTINUE
      N1=N-1
      DO 17 J=1,N1
      J1=J+1
      DO 17 I=J1,N
      PP=0
      I1=I-1
      IF (J1.GT.I1) GO TO 17
      DO 18 P=J1,I1
18 PP=PP+A(I,P)*A(P,J)
17 A(I,J)=-A(I,J)-PP
21 DO 84 L=1,N
      J=N-L+1
      DO 84 M=1,J
      I=J-M+1
      IF ((I+1).GT.J) GO TO 71
      I1=I+1
      PP=0
      DO 70 P=I1,J
70 PP=PP+A(I,P)*A(P,J)
      A(I,J)=-PP/A(I,I)
      GO TO 84
71 A(I,J)=1.0/A(I,J)
84 CONTINUE
      N1=N-1
      DO 24 J=1,N1
      DO 24 I=1,N
      IF (I.GT.J) GO TO 27
      PP=0
      J1=J+1
      DO 26 P=J1,N
26 PP=PP+A(I,P)*A(P,J)
      A(I,J)=A(I,J)+PP
      GO TO 24
27 PP=0
      DO 28 P=I,N
28 PP=PP+A(I,P)*A(P,J)
      A(I,J)=PP
24 CONTINUE
      IF (INDEX.EQ.0) GO TO 59
      DO 121 L=1,N

```

```

      K=N-L+1
      IF (DABS(S(K)).LT. 0.1D-08) GO TO 121
      DO 90 J=1,N
      Z(J)=A(K,J)
90    Y(J)=A(J,K)
      G=1.0-S(K)*A(K,K)
      IF (DABS(G).LT. 0.1D-05) GO TO 7178
      DO 91 I=1,N
      DO 91 J=1,N
91    A(I,J)=A(I,J)+S(K)*Y(I)*Z(J)/G
121  CONTINUE
59    DO 45 I=1,N
      PP=0
      DO 46 P=1,N
46    PP=PP+A(I,P)*B(P)
45    Y(I)=PP
      DO 47 I=1,N
47    B(I)=Y(I)
      GO TO 1601
7178 CONTINUE
1601 PP=0
      PP1=0
      PP2=0
      DO 60 I=1,N
      PP2=PP2+W(I)*W(I)
      DO 60 J=1,N
      PP=PP+A(I,J)*A(I,J)
60    PP1=PP1+X(I,J)*X(I,J)
      PP=DSQRT(PP)
      PP1=DSQRT(PP1)
      PP2=DSQRT(PP2)
      DO 57 I=1,N
      PP4=0
      DO 73 P=1,N
73    PP4=PP4+X(I,P)*B(P)
57    W(I)=W(I)-PP4
      PP3=0
      DO 56 I=1,N
56    PP3=PP3+W(I)*W(I)
      PP3=DSQRT(PP3)
      DO 55 I=1,N
      DO 50 J=1,N
      PP4=0
      DO 54 P=1,N
54    PP4=PP4+X(I,P)*A(P,J)
50    W(J)=PP4
      DO 55 P=1,N
55    X(I,P)=W(P)
      DO 53 I=1,N
53    X(I,I)=1.0-X(I,I)
      PP4=0
      DO 52 I=1,N
      DO 52 J=1,N
52    PP4=PP4+X(I,J)*X(I,J)
      PP4=DSQRT(PP4)
      PP5=PP3/PP1
      PP6=PP*PP4*PP2/(1.0-PP4)

```

```

RETURN
END
C  ----- DATA PRINTING BLOCK -----
SUBROUTINE PYZ (NN,NH,NK,Y,Z,FILT)
COMMON NKV,NHV,NV,NHK,NO,IRL
DOUBLE PRECISION Y(NV,NHV),Z(NV,NKV)
CHARACTER *6 FILT(6)
421 FORMAT(10D16.6)
WRITE(*,*) 'DEVICE TO PRINT Y AND Z ?'
READ(*,*) NR
IRN=IRL*NN
OPEN(7,FILE=FILT(2),STATUS='OLD',ACCESS='DIRECT',RECL=IRN)
DO 71 J=1,NH
READ(7,REC=J)(Y(I,J),I=1,NN)
71 CONTINUE
CLOSE(7)
OPEN(8,FILE=FILT(1),STATUS='OLD',ACCESS='DIRECT',RECL=IRN)
DO 72 J=1,NK
READ(8,REC=J)(Z(I,J),I=1,NN)
72 CONTINUE
CLOSE(8)
DO 1 I=1,NN
WRITE(NR,2) (Y(I,J),J=1,NH),(Z(I,J),J=1,NK)
1 CONTINUE
2 FORMAT(10D16.5)
PAUSE 'TO PROCEED PRESS ENTER KEY'
RETURN
END

```

C PROG1.FOR

```

C  MAIN PROGRAM FOR REDUCED FORM ANS STOCHASTIC Y GENERATION
C  PARAMETER (NV=50, NKV=10, NHK=10, NHV=10, IRL=8)
COMMON /CB1/ML,NL,ACCRC
INTEGER *2 YORD(10,10),ZORD(10,10)
CHARACTER FILS(6) *11, FILT(6) *6, NAM(7) *6
DOUBLE PRECISION A(10,10),B(10,10),OLSS(10),MADS(10)
DOUBLE PRECISION Y(50,10),Z(50,15),ZDZ(10,10),ZDY(10,10)
DOUBLE PRECISION PIE(10,10),YO(50),AA(10),AVA(10),AVB(10)
DOUBLE PRECISION Y1(50),Z1(50,10),AZ(10),ER(50),TCA(10,10)
DOUBLE PRECISION XY(10),YH(50),BZ(10),C(30),TCB(10,10)
DOUBLE PRECISION RCL,ACCRC
C  CHARACTER *11 FIL,RFIL,PFIL,FILY,COEF,FNRM
C  CHARACTER *6 NAM(7),FZ,FY,FO,TC,PRM,PRM2
C  PARAMETERS -----
NV=50
NHV=10
NKV=10
ML=NKV
NL=NV
IRL=8
C  ----- FILES -----
FILT(1)='FILZ'
FILT(2)='FILY'
FILT(3)='YORD'
FILT(4)='TRUC'
FILT(5)='PARA'

```

```

FILT(6)='PARA2'
FILS(1)='MMMM'
FILS(2)='RMMMM'
FILS(3)='MADPAR'
FILS(4)='YHAT'
FILS(5)='COEFF'
FILS(6)='NORMS'
C ----- STRING VARIABLES -----
NAM(1)='TRUEC'
NAM(2)='MOLSC'
NAM(3)='MMADC'
NAM(4)='LSRMS'
NAM(5)='MDRMS'
NAM(6)='LMRMS'
NAM(7)='GLRMS'
WRITE(*,*) 'CHOOSE FLS & FMD AS 2 k-VALUES IN K-CLASS ESTIMATION'
WRITE(*,*) 'k = FLS or FMD 0<= K <=1 SEE K-CLASS ESTIMATION'
READ(*,*) FLS,FMD
WRITE(*,*) 'ACCURACY IN MAD ESTIMATION IDEALLY 0.005'
READ(*,*) ACCRC
OPEN(7,FILE=FILT(5),STATUS='OLD')
READ(7,*) NH,NK,NN,IRL
WRITE(*,*) 'NOW READS YORDij, ZORDik FROM FILE - YORD'
OPEN(11,FILE=FILT(3),STATUS='OLD')
DO 12 I=1,NH
  READ(11,*) (YORD(I,J),J=1,NH),(ZORD(I,J),J=1,NK)
12 CONTINUE
  CLOSE(11)
C ----- NOW READS TRUE Y AND TRUE Z FROM FILES -----
IRN=IRL*NN
OPEN(7,FILE=FILT(2),STATUS='OLD',ACCESS='DIRECT',RECL=IRN)
DO 13 J=1,NH
  READ(7,REC=J)(Y(I,J),I=1,NN)
13 CONTINUE
  CLOSE(7)
  OPEN(8,FILE=FILT(1),STATUS='OLD',ACCESS='DIRECT',RECL=IRN)
  DO 14 J=1,NK
    READ(8,REC=J)(Z(I,J),I=1,NN)
14 CONTINUE
    CLOSE(8)
C ----- COMPUTES (estimated) REDUCED FORM COEFFICIENTS-----
C
WRITE(*,*) 'COMPUTES estimated REDUCED FORM COEFFICIENTS'
DO 1 J=1,NK
  DO 2 JJ=J,NK
    ZDZ(J,JJ)=0.0
    DO 3 I=1,NN
      3 ZDZ(J,JJ)=ZDZ(J,JJ)+Z(I,J)*Z(I,JJ)
    2 ZDZ(JJ,J)=ZDZ(J,JJ)
    DO 1 JJ=1,NH
      ZDY(J,JJ)=0.0
      DO 1 I=1,NN
        ZDY(J,JJ)=ZDY(J,JJ)+Z(I,J)*Y(I,JJ)
1 CONTINUE
  CALL MINV(ZDZ,NK)
  DO 4 J=1,NK
    DO 4 JJ=1,NH

```

```

    PIE(J,JJ)=0.0
    DO 4 I=1,NK
    PIE(J,JJ)=PIE(J,JJ)+ZDZ(J,I)*ZDY(I,JJ)
4 CONTINUE
C ----- SHOW UP PIE (Reduced Form) COEFFICIENT MATRIX -----
  WRITE(*,*) 'ESTIMATED REDUCED FORM COEFFICIENT MATRIX'
  DO 5 J=1,NH
  WRITE(*,6) (PIE(JJ,J),JJ=1,NK)
5 CONTINUE
6 FORMAT(10E16.6)
C ----- COMPARE WITH TRUE PIE MATRIX-----
  WRITE(*,*) ' TRUE PIE - Reduced Form - MATRIX'
  OPEN(11,FILE=FILT(4),STATUS='OLD',ACCESS='DIRECT',RECL=IRL)
  READ(11,REC=1) RCL
  WRITE(*,*) 'RCL =',RCL
  IRCL=1
  DO 7 I=1,NH
  DO 7 J=1,NK
  IRCL=IRCL+1
  READ(11,REC=IRCL) PIE(J,I)
7 CONTINUE
  DO 77 I=1,NH
  DO 77 J=1,NH
  IRCL=IRCL+1
  READ(11,REC=IRCL) TCA(J,I)
77 CONTINUE
  DO 78 I=1,NH
  DO 78 J=1,NK
  IRCL=IRCL+1
  READ(11,REC=IRCL) TCB(J,I)
78 CONTINUE
  CLOSE(11)
C ---PROCEEDS TO OLS & MAD ESTIMATIONS OF REDUCED FORM EQNS---
  IRLA=IRL*NK
  IRLY=IRL*NN
  OPEN(13,FILE=FILS(1),STATUS='NEW',ACCESS='DIRECT',RECL=IRLA)
  OPEN(12,FILE=FILS(4),STATUS='NEW',ACCESS='DIRECT',RECL=IRLY)
  OPEN(9,FILE=FILS(2),STATUS='NEW')
  IRA=0
  IRY=0
  DO 11 MEQN=1,NH
  DO 9 I=1,NN
  YO(I)=Y(I,MEQN)
  9 CONTINUE
  DO 10 JJ=1,NK
  10 AA(JJ)=-PIE(JJ,MEQN)
  NKOLD=NK
  CALL MADM(YO,Z,NN,NK,AA,FILS,IRA,IRY,MEQN,NTRY,
1FMDF,FLS,TOL,ITMX,NAM)
C  WRITE(*,*) 'RETURNED FROM MADM'
  NK=NKOLD
11 CONTINUE
  CLOSE(9)
  CLOSE(12)
  CLOSE(13)
C  WRITE(*,*) 'IRA[=nh(1+ntry*2)] & IRY[=ntry*nh*3]',IRA,IRY
  WRITE(*,*) 'TRUE REDUCED FORM MATRIX'

```

```

      DO 8 J=1,NH
      WRITE(*,888) (PIE(JJ,J),JJ=1,NK)
      8 CONTINUE
888 FORMAT(10D16.6)
C    --- ESTIMATION OF REDUCED FORM EQUATIONS IS OVER -----
C    *****
      OPEN(10,FILE=FILT(6),STATUS='NEW')
      WRITE(10,*) NH,NK,NTRY,NN
      CLOSE(10)
      END
C    ===== MAIN SUBROUTINE FOR MAD SIMULATION =====
      SUBROUTINE MADM(YC,XC,N,M,AA,FILS,IRA,IRY,MEQN,NTRY,FMD,FLS,TOL,
1ITMX,NAM)
      COMMON /CB1/ML,NL,ACCRC
      DOUBLE PRECISION YO(50),XO(50,10),Y(50),X(50,10),OLSS(10)
      DOUBLE PRECISION YC(50),XC(50,10),ER(50),YH(50),A(10)
      DOUBLE PRECISION AA(10),B(10),AVA(10),AVB(10),MADS(10)
      DOUBLE PRECISION YM,RSQ,OLSN,AMADN,RNORM,ACCRC
      INTEGER *2 IU,IV,IUU,SGNS(30)
      CHARACTER *11 FILS(6)
      CHARACTER *6 NAM(7)
C    =====
C    PARAMETERS FOR DIMENSION: CHANGE THEM IF THE PROBLEM IS LARGER
C    CHANGE THE FORMAT IF REQUIRED
10 FORMAT(8F10.2)
C    -----
C    WRITE(*,*) 'FEED NO. OF OBSERVATIONS AND NO. OF VARIABLES'
C    READ(*,*) N,M
C    IF(MEQN.LT.2) THEN
      OPEN(14,FILE=FILS(3),STATUS='NEW')
      WRITE(*,*) 'FEED THE TOLERANCE LEVEL FOR ACCURACY (.0001)'
      WRITE(*,*) 'This is required for iterative solution of
parameters'
      READ(*,*) TOL
      WRITE(*,*) 'FEED STANDARD DEVIATION OF ERROR TERM. This value
may'
      WRITE(*,*) 'not exceed 0.3 else R Squares come poor. Your Option'
      READ(*,*) SDEV
      WRITE(*,*) 'NUMBER OF OUTLIERS TO GENERATE AND THEIR SIZE with'
      WRITE(*,*) 'respect to average Y. Size may not exceed 1.0'
      WRITE(*,*) 'But it is your choice.Size = 2 is quite large'
      READ(*,*) NOL,SIZE
      WRITE(*,*) 'FEED THE SIGNS -1 OR 1 FOR EACH OUTLIER'
      READ(*,*) (SGNS(J),J=1,NOL)
      WRITE(*,*) 'CHOOSE ERROR DISTRIBUTION BY ONE OF THE OPTIONS'
      WRITE(*,*) 'Normal=0; Cauchy=1; Gamma=2; Beta2=3; Beta1=4'
      READ(*,*) ID
      IF(ID.GE.2) THEN
      WRITE(*,*) 'To generate Gamma or Beta distributed Random Numbers'
      WRITE(*,*) 'it requires to generate Gamma from Normal distn.'
      WRITE(*,*) 'PARAMETER FOR GAMMA DISTRIBUTION (ig)'
      WRITE(*,*) 'It generates G(ig) and Beta1 or Beta2 of (.5ig,.5ig)'
      READ(*,*) IG
      ENDIF
      WRITE(*,*) 'NO. OF EXPERIMENTS (REPLICATES > = 50, say) ? '
      READ(*,*) NTRY
C    WRITE(*,*) 'NAME the FILE TO STORE THE RESULTS ?'

```

```

C      READ(*,*) FILS(1)
      WRITE(*,*) 'SET MAX ITERATION FOR MAD ESTIMATION'
      READ(*,*) ITMX
      WRITE(*,*) 'FEED THE SEED TO GENERATE RANDOM NUMBERS'
      READ(*,*) IU
      WRITE(14,*) TOL,SDEV,NOL,SIZE,(SGNS(J),J=1,NOL)
      WRITE(14,*) ID,IG,NTRY,ITMX,IU
      CLOSE(14)
      ENDIF
      OPEN(14,FILE=FILS(3),STATUS='OLD')
      READ(14,*) TOL,SDEV,NOL,SIZE,(SGNS(J),J=1,NOL)
      READ(14,*) ID,IG,NTRY,ITMX,IU
      CLOSE(14)
      TTOL=TOL
      YM=0.0
      DO 8 I=1,N
      YM=YM+YC(I)
8 CONTINUE
      YM=YM/N
      WRITE(*,*) 'AVERAGE Y IS = ',YM
C      -----
C      OPEN(13,FILE=FILS(1),STATUS='NEW')
      IRA=IRA+1
      IRAA=IRA
      WRITE(13,REC=IRA)(AA(J),J=1,M)
C      ?????????????????????????????????????????????????????????????
C      ?????????????????????????????????????????????????????????????
      WRITE(*,*) 'NTRY IS = ',NTRY
C      PAUSE
      DO 100 IT=1,NTRY
C      *****
      DO 1 I=1,N
      YO(I)=YC(I)
      Y(I)=YC(I)
      DO 1 J=1,M
      XO(I,J)=XC(I,J)
      X(I,J)=XC(I,J)
1 CONTINUE
C      -----
C      -----GENERATION OF ERRORS AND ADDING THEM TO YC(I)-----
      CALL ERGEN(N,YO,Y,ER,SDEV,IU,IV,ID,IG)
C      PAUSE 'RETURN FROM ERGEN'
C      -----
C      WRITE(*,*) ' PRINT OF XC, YC AND YO'
      DO 6 I=1,N
      WRITE(*,10)(XC(I,J),J=1,M),YC(I),YO(I)
C      6 CONTINUE
C      PAUSE
C      -----
      CALL OUTLI(YO,Y,N,IU,IV,NOL,SIZE,SGNS)
C      PAUSE 'RETURN FROM OUTLI'
C      -----
      CALL MAD(YO,XO,Y,X,M,N,TOL,ITMX,AA,A,YH,ER,RSQ,B)
C      PAUSE 'RETURN FROM MAD'
C      *****
C      PAUSE
C      - - - - -

```



```

C      STORE stochastic Y IN FILE FILS(4)
C      Initial IRY from calling Pgm. FILS(4) #12 opened in calling Pgm.
      IRY=IRY+1
      WRITE(12,REC=IRY) (YO(I), I=1,N)
C      STORE LS EXPECTED Y IN #12 FILS(4): FLS IS k OF K-CLASS
C      WRITE(*,*) FLS,FMD , ' FLS FMD'
C      PAUSE
      DO 103 I=1,N
      Y(I)=YO(I)
      DO 133 J=1,M
133 Y(I)=Y(I)-XO(I,J)*B(J)
103 Y(I)=YO(I)-FLS*Y(I)
      IRY=IRY+1
      WRITE(12,REC=IRY) (Y(I), I=1,N)
C      STORE MAD EXPECTED Y IN #12 FILS(4): FMD IS k OD K-CLASS
      DO 104 I=1,N
      Y(I)=YO(I)
      DO 134 J=1,M
134 Y(I)=Y(I)-XO(I,J)*A(J)
104 Y(I)=YO(I)-FMD*Y(I)
      IRY=IRY+1
      WRITE(12,REC=IRY) (Y(I), I=1,N)
C      - - - - -
      IRA=IRA+1
      WRITE(13,REC=IRA) (B(J), J=1,M)
      IRA=IRA+1
      WRITE(13,REC=IRA) (A(J), J=1,M)
      WRITE(*,99) IT,NTRY
100 CONTINUE
99 FORMAT(1X,/, '----- EXPERIMENT NO.',I5, ' OVER ',I5, ' -----',/)
C      CLOSE(13)
C      OPEN(13,FILE=FILS(1))
      READ(13,REC=IRAA) (AA(J), J=1,M)
      OLSN=0.0
      AMADN=0.0
      DO 3 J=1,M
      OLSS(J)=0.0
      MADS(J)=0.0
      AVA(J)=0.0
      AVB(J)=0.0
3 CONTINUE
      IRA1=IRAA
      DO 4 I=1,NTRY
      IRA1=IRA1+1
      READ(13,REC=IRA1) (B(J), J=1,M)
      IRA1=IRA1+1
      READ(13,REC=IRA1) (A(J), J=1,M)
      DO 4 J=1,M
      OLSS(J)=OLSS(J)+(B(J)-AA(J))**2
      MADS(J)=MADS(J)+(A(J)-AA(J))**2
      AVA(J)=AVA(J)+A(J)
      AVB(J)=AVB(J)+B(J)
4 CONTINUE
C      CLOSE(13, STATUS='DELETE')
      DO 5 J=1,M
      OLSN=OLSN+OLSS(J)
      AMADN=AMADN+MADS(J)

```

```

      AVA(J)=AVA(J)/NTRY
      AVB(J)=AVB(J)/NTRY
5  CONTINUE
      RNORM=DSQRT(OLSN/AMADN)
      WRITE(9,*) 'FILE NAME = ',FILS(2)
      WRITE(9,*) '      N -M -  TOL -IUU - SDEV- NOL -SIZE -NTRY-ID-IG'
      WRITE(9,12) N,M,TTOL,IUU,SDEV,NOL,SIZE,NTRY,ID,IG
12  FORMAT(1X,I4,I3,F7.4,I5,F7.2,I5,F6.2,I6,2I3)
      WRITE(9,*) 'SIGNS OF OUTLIERS', (SGNS(J),J=1,NOL)
      WRITE(9,11) NAM(1), (AA(J),J=1,M)
      WRITE(9,11) NAM(2), (AVB(J),J=1,M)
      WRITE(9,11) NAM(3), (AVA(J),J=1,M)
      WRITE(9,11) NAM(4), (SQRT(OLSS(J)/FLOAT(NTRY)),J=1,M)
      WRITE(9,11) NAM(5), (SQRT(MADS(J)/FLOAT(NTRY)),J=1,M)
      WRITE(9,11) NAM(6), (SQRT(OLSS(J)/MADS(J)),J=1,M)
      WRITE(9,11) NAM(7),RNORM
11  FORMAT(1X,A5,2X,15F9.4)
      RETURN
      END
C  -----
C  Central Program for MAD estimation using Iterated Weighted LS
      SUBROUTINE MAD(YO,XO,Y,X,M,N,TOL,ITMX,AA,A,YH,ER,RSQ,B)
      COMMON /CB1/ML,NL,ACCRC
      DOUBLE PRECISION YO(NL),XO(NL,ML),Y(NL),X(NL,ML),A(ML),ER(NL)
      DOUBLE PRECISION YH(NL),AA(ML),B(ML),C(30)
      DOUBLE PRECISION RSQ,ASUM,ABSUM,W,ACCRCC,ACCRCC
      ACCRCC=ACCRCC
C      DO 200 I=1,N
C      WRITE(*,10) YO(I), (XO(I,J),J=1,M)
C 200  CONTINUE
      10  FORMAT(1X,10F13.2)
1234  DF=FLOAT(N-M)
      NACC=0
      ITER=0
      CALL OLS(YO,XO,Y,X,N,M,AA,A,YH,ER,RSQ)
C  -----
      DO 1 J=1,M
      C(J)=A(J)
      1  B(J)=A(J)
100  ITER=ITER+1
      IF(ITER.GT.ITMX) THEN
C      WRITE(*,*) 'NO CONVERGENCE IN ITERATIONS =',ITMX
      GOTO 7
      ENDIF
      ABSUM=0
      NCOUNT=0
      DO 2 I=1,N
      ER(I)=DABS(ER(I))
      ABSUM=ABSUM+ER(I)
      W=0.0
      IF(ER(I).GT.ACRC) THEN
      W=1.0/ER(I)
      NCOUNT=NCOUNT+1
      ENDIF
      Y(I)=YO(I)*W
      DO 3 J=1,M
      X(I,J)=XO(I,J)*W

```

```

3 C(J)=A(J)
2 CONTINUE
  IF((NCOUNT-(M+3)).LT.0) THEN
    ACCRC=ACCRC/10.0
    GOTO 1234
  ENDIF
  IF(ITER.EQ.1) ERSUM=ABSUM
  CALL OLS(YO,XO,Y,X,N,M,AA,A,YH,ER,RSQ)
  ASUM=0
  DO 4 I=1,N
    ER(I)=DABS(ER(I))
4 ASUM=ASUM+ER(I)
  DO 6 J=1,M
    IF(DABS(C(J)-A(J)).GT.TOL) GOTO 100
6 CONTINUE
C -----GOES OUT FROM ITERATIVE LOOP-----
7 CONTINUE
C -----
  WRITE(*,*) 'TOTAL NUMBER OF ITERATIONS PERFORMED ',ITER-1
  WRITE(*,*) 'OLS AND MAD DEVIATION SUMS ARE ',ERSUM,ASUM
  ACCRC=ACCRC
  RETURN
  END
C -----
C Central Program for Least Squares Estimation of Regression Eqn.
  SUBROUTINE OLS(YO,XO,Y,X,N,M,AA,A,YH,ER,RSQ)
  COMMON /CB1/ML,NL,ACCRC
  DOUBLE PRECISION Y(NL),X(NL,ML),A(ML),ER(NL),YO(NL),XO(NL,ML)
  DOUBLE PRECISION YH(NL),AA(ML),SE,ES,S,SS,RSQ,ACCRC
  DOUBLE PRECISION XX(10,10),XY(10)
  DF=FLOAT(N-M)
  DO 1 J=1,M
    XY(J)=0.0
    DO 2 K=J,M
      XX(J,K)=0.0
    DO 10 I=1,N
10 XX(J,K)=XX(J,K)+X(I,J)*X(I,K)
    XX(K,J)=XX(J,K)
  2 CONTINUE
    DO 1 I=1,N
      XY(J)=XY(J)+X(I,J)*Y(I)
  1 CONTINUE
    CALL MINV(XX,M)
    DO 3 J=1,M
      A(J)=0.0
      DO 3 K=1,M
3 A(J)=A(J)+XX(J,K)*XY(K)
      SE=0.0
      ES=0.0
      S=0.0
      SS=0.0
      DO 4 I=1,N
        YH(I)=0.0
      DO 5 J=1,M
5 YH(I)=YH(I)+XO(I,J)*A(J)
      S=S+YH(I)
      SS=SS+YH(I)*YH(I)

```

```

    ER(I)=YO(I)-YH(I)
    SE=SE+ER(I)
    ES=ES+ER(I)*ER(I)
4  CONTINUE
    S=S/FLOAT(N)
    SE=SE/FLOAT(N)
    RSQ=(SS/FLOAT(N)-S*S)/( (SS/FLOAT(N)-S*S)+(ES/FLOAT(N)-SE*SE) )
    RETURN
    END
C  ----- MATRIX INVERSION -----
    SUBROUTINE MINV(AAA,N)
    INTEGER P
    DOUBLE PRECISION A,B,S,G,Y,Z,W,PP,PP1,PP2,PP3,PP4,PP5,PP6
    DOUBLE PRECISION SKID,SKNORM
    DOUBLE PRECISION AAA(10,10),CII(10,10),X(10,10)
    DIMENSION A(10,10),B(10),S(10),Y(10),Z(10),W(10)
    DO 1234 I=1,N
    DO 1234 J=1,N
1234 A(I,J)=AAA(I,J)
    DO 58 I=1,N
    W(I)=B(I)
    DO 58 J=1,N
    58 X(I,J)=A(I,J)
    PP1=0
    NN=N*N
    DO 2 I=1,N
    DO 2 J=1,N
    2 PP1=PP1+DABS(A(I,J))
    PP2=PP1/NN
    INDEX=0
    DO 33 I=1,N
    33 S(I)=0
    DO 506 J=1,N
    DO 506 I=1,N
    IF (I.GT.J) GO TO 503
    PP=0
    I1=I-1
    DO 501 P=1,I1
    IF (I1.LT.1) GO TO 502
    501 PP=PP+A(I,P)*A(P,J)
    A(I,J)=A(I,J)-PP
    502 IF (I.NE.J) GO TO 506
    IF (DABS(A(J,J)).LT.(PP2*.1D-08)) GO TO 507
    GO TO 506
    503 PP=0
    J1=J-1
    DO 504 P=1,J1
    IF (J1.LT.1) GO TO 505
    504 PP=PP+A(I,P)*A(P,J)
    505 Y(I)=(A(I,J)-PP)/A(J,J)
    IF (I.EQ.N) GO TO 508
    GO TO 506
    508 JJ1=J+1
    DO 1 P=JJ1,N
    IF (DABS(Y(P)).GT.(10.0*PP1)) GO TO 507
    1 CONTINUE
    DO 4 L= JJ1,N

```

```

      4  A(L,J)=Y(L)
        GO TO 506
C 507 I=J
507 S(J)=PP2
      A(J,J)=A(J,J)+PP2
      INDEX=INDEX+1
506 CONTINUE
      N1=N-1
      DO 17 J=1,N1
        J1=J+1
        DO 17 I=J1,N
          PP=0
          I1=I-1
          IF (J1.GT.I1) GO TO 17
          DO 18 P=J1,I1
18      PP=PP+A(I,P)*A(P,J)
17      A(I,J)=-A(I,J)-PP
21      DO 84 L=1,N
        J=N-L+1
        DO 84 M=1,J
          I=J-M+1
          IF ((I+1).GT.J) GO TO 71
          I1=I+1
          PP=0
          DO 70 P=I1,J
70      PP=PP+A(I,P)*A(P,J)
          A(I,J)=-PP/A(I,I)
          GO TO 84
71      A(I,J)=1.0/A(I,J)
84      CONTINUE
      N1=N-1
      DO 24 J=1,N1
        DO 24 I=1,N
          IF (I.GT.J) GO TO 27
          PP=0
          J1=J+1
          DO 26 P=J1,N
26      PP=PP+A(I,P)*A(P,J)
          A(I,J)=A(I,J)+PP
          GO TO 24
27      PP=0
          DO 28 P=I,N
28      PP=PP+A(I,P)*A(P,J)
          A(I,J)=PP
24      CONTINUE
      IF (INDEX.EQ.0) GO TO 59
      DO 121 L=1,N
        K=N-L+1
        IF (DABS(S(K)).LT. 0.1D-08) GO TO 121
        DO 90 J=1,N
          Z(J)=A(K,J)
90      Y(J)=A(J,K)
          G=1.0-S(K)*A(K,K)
          IF (DABS(G).LT. 0.1D-13) GO TO 7178
          DO 91 I=1,N
            DO 91 J=1,N
91      A(I,J)=A(I,J)+S(K)*Y(I)*Z(J)/G

```

```

121 CONTINUE
59 DO 45 I=1,N
    PP=0
    DO 46 P=1,N
46 PP=PP+A(I,P)*B(P)
45 Y(I)=PP
    DO 47 I=1,N
47 B(I)=Y(I)
    GO TO 1601
7178 CONTINUE
1601 PP=0
    PP1=0
    PP2=0
    DO 60 I=1,N
    PP2=PP2+W(I)*W(I)
    DO 60 J=1,N
    PP=PP+A(I,J)*A(I,J)
60 PP1=PP1+X(I,J)*X(I,J)
    PP=DSQRT(PP)
    PP1=DSQRT(PP1)
    PP2=DSQRT(PP2)
    DO 57 I=1,N
    PP4=0
    DO 73 P=1,N
73 PP4=PP4+X(I,P)*B(P)
57 W(I)=W(I)-PP4
    PP3=0
    DO 56 I=1,N
56 PP3=PP3+W(I)*W(I)
    PP3=DSQRT(PP3)
    DO 55 I=1,N
    DO 50 J=1,N
    PP4=0
    DO 54 P=1,N
54 PP4=PP4+X(I,P)*A(P,J)
50 W(J)=PP4
    DO 55 P=1,N
55 X(I,P)=W(P)
    DO 53 I=1,N
53 X(I,I)=1.0-X(I,I)
    PP4=0
    DO 52 I=1,N
    DO 52 J=1,N
52 PP4=PP4+X(I,J)*X(I,J)
    PP4=DSQRT(PP4)
    PP5=PP3/PP1
    PP6=PP*PP4*PP2/(1.0-PP4)
    DO 1236 I=1,N
    DO 1236 J=1,N
    CII(I,J)=0.0
    DO 1236 K=1,N
    CII(I,J)=CII(I,J)+AAA(I,K)*A(K,J)
1236 CONTINUE
    SKNORM=0.0
    DO 1237 I=1,N
    DO 1237 J=1,N
    SKID=0.0

```

```

      IF (I.EQ.J) SKID=1.0
      SKNORM=SKNORM+ (CII (I,J) -SKID) **2
1237 CONTINUE
      SKNORM=DSQRT (SKNORM/ (N*N) )
C -----
C      IF (SKNORM.GT.0.0000001) THEN
C      WRITE (*,*) 'INV MATRIX IS INACCURATE. NORM=', SKNORM
C      PAUSE 'PROCEED OR ABORT'
C      ENDIF
C -----
1238 FORMAT (10F8.5)
      DO 1235 I=1,N
      DO 1235 J=1,N
1235 AAA (I,J)=A (I,J)
      RETURN
      END
C -----
C      This Program generates error term for regression equation
      SUBROUTINE ERGEN (N, YO, Y, ER, SDEV, IU, IV, ID, IG)
      COMMON /CB1/ML, NL, ACCRC
      DOUBLE PRECISION YO (NL), Y (NL), ER (NL)
      DOUBLE PRECISION ER2, ERNORM, S, SS, ACCRC
      INTEGER *2 IU, IV
      DO 1 I=1,N
      ER1=ERNORM (IU, IV)
      ER2=ERNORM (IU, IV)
C -----
C      Normally distributed as average of two normals
      IF (ID.EQ.0) ER (I)=(ER1+ER2)/2.0
C      Cauchy-distributed error as ratio of two std normal variates
      IF (ID.EQ.1) ER (I)=(ER1/ER2)
C      Gamma-distributed error as average of IG squared normals
      ER1=0.0
      ER2=0.0
      DO 2 JG=1, IG
      ER1=ER1+ (ERNORM (IU, IV) ) **2
      ER2=ER2+ (ERNORM (IU, IV) ) **2
      2 CONTINUE
      IF (ID.EQ.2) ER (I)=(ER1+ER2)/2.0
C      Beta2-distributed as a ratio of square of two normals
      IF (ID.EQ.3) ER (I)=ER1/ER2
C      Beta1-distributed as x/(x+y), x and y being normal
      IF (ID.EQ.4) ER (I)=ER1/ (ER1+ER2)
C -----
      1 CONTINUE
      S=0
      SS=0
      DO 3 I=1,N
      S=S+ER (I)
      SS=SS+ER (I) **2
      3 CONTINUE
      S=S/N
      SS=DSQRT (SS/N-S**2)
      DO 4 I=1,N
C      IF (ID.LT.2) ER (I)= ( (ER (I) -S) /SS) *SDEV*YM
      ER (I)= ( (ER (I) -S) /SS) *SDEV*100.0
      YO (I)=YO (I)+ER (I)

```

```

      Y(I)=YO(I)
4  CONTINUE
      RETURN
      END
C  -----
C  This Program generates Outliers for MAD estimation
      SUBROUTINE OUTLI(YO,Y,N,IU,IV,NOL,SIZE,SGNS)
      INTEGER *2 IU,IV,SGNS(30)
      COMMON /CB1/ML,NL,ACCRC
      DOUBLE PRECISION YO(NL),Y(NL)
      DOUBLE PRECISION YM,ACCRC
C  -----
      YM=0.0
      DO 1 I=1,N
      YM=YM+YO(I)
1  CONTINUE
      YM=YM/N
C  WRITE(*,*) 'AVERAGE Y IS = ',YM
      DO 3 NT=1,NOL
2  J=RAND(IU,IV)*(N+1)
      IF(J.EQ.0) GO TO 2
      EJ=RAND(IU,IV)*YM*SIZE*SGNS(NT)
      YO(J)=YO(J)+EJ
      Y(J)=YO(J)
3  CONTINUE
      RETURN
      END
C  -----
C  Program to generate N(0,s) from Rectangular Random Numbers
C  It uses Variate Transformation for this purpose.
      FUNCTION ERNORM(IU,IV)
      DOUBLE PRECISION ERNORM,U(2)
      INTEGER *2 IU,IV
      DO 1 I=1,2
      U(I)=RAND(IU,IV)
1  CONTINUE
C  -----
C  IF U1 AND U2 ARE UNIFORMLY DISTRIBUTED RANDOM NUMBERS (0,1),
C  THEN X=[(-2*ln(U1))**.5]*(COS(2*PI*U2) IS N(0,1)
C  PI = 4*ARCTAN(1.0)= 2*1.57079632679489
C  PI*2 = 6.2831853071797
C  -----
      ERNORM=DSQRT(-2.0*DLOG(U(1)))*DCOS(U(2)*6.2831853071797)
      RETURN
      END
C  -----
C  Generates Rectangular (0,1) Random Numbers
      FUNCTION RAND(IU,IV)
      INTEGER *2 IU,IV
      IV=IU*259
      IF(IV.GE.0) GOTO 2
      IV=IV+32767+1
2  RAND=IV
      IU=IV
      RAND=RAND*0.3051851E-04
      RETURN
      END

```


C **PROG2.FOR**

```

C     MAIN PROGRAM FOR REDUCED FORM ANS STOCHASTIC Y GENERATION
C     PARAMETER (NV=50, NKV=10, NHK=10, NHV=10, IRL=8)
C     COMMON /CB1/ML,NL,ACCRC
C     CHARACTER FILS(6) *11, FILT(6) *6, NAM(7) *6, SL2N *6
C     INTEGER *2 YORD(10,10), ZORD(10,10)
C     DOUBLE PRECISION A(10,10), B(10,10), OLSS(10), MADS(10)
C     DOUBLE PRECISION Y(50,10), Z(50,10), ZF(50,10), YH(50), SDST(4)
C     DOUBLE PRECISION PIE(10,10), YO(50), AA(10), AVA(10), AVB(10)
C     DOUBLE PRECISION Y1(50), Z1(50,10), AZ(10), ER(50), TCA(10,10)
C     DOUBLE PRECISION BZ(10), C(30), TCB(10,10), DSTL(10,2), DSTM(10,2)
C     DOUBLE PRECISION RCL, ACCRC, TOL, TTOL, OLSN, RNORM, AMADN, RSQ, DSTT
C     CHARACTER *11 FIL, RFIL, PFIL, FIFY, COEF, FNRM
C     CHARACTER *6 NAM(7), FZ, FY, FO, TC, PRM, PRM2
C     MX(M, NHMX) = M - NHMX
C     PARAMETERS -----
C     NV=50
C     NHV=10
C     NKV=10
C     ML=NKV
C     NL=NV
C     IRL=8
C     ----- FILES -----
C     FILT(1)='FILZ'
C     FILT(2)='FIFY'
C     FILT(3)='YORD'
C     FILT(4)='TRUC'
C     FILT(5)='PARA'
C     FILT(6)='PARA2'
C     FILS(1)='MMMM'
C     FILS(2)='RMMMM'
C     FILS(3)='MADPAR'
C     FILS(4)='YHAT'
C     FILS(5)='COEFF'
C     FILS(6)='NORMS'
C     SL2N='SL2N'
C     ----- STRING VARIABLES -----
C     NAM(1)='TRUEC'
C     NAM(2)='MOLSC'
C     NAM(3)='MMADC'
C     NAM(4)='LSRMS'
C     NAM(5)='MDRMS'
C     NAM(6)='LMRMS'
C     NAM(7)='GLRMS'
C     WRITE(*,*) 'NORMS ETC TO COMPUTE WITH CONSTANT OR SNS CONSTANT ?'
C     WRITE(*,*) 'IF WITH CONSTNAT FEED ZERO ELSE FEED 1'
C     READ(*,*) NHMX
C     WRITE(*,*) 'CHOOSE FLS & FMD AS 2 k-VALUES IN K-CLASS'
C     ESTIMATION'
C     WRITE(*,*) 'k = FLS or FMD 0<= K <=1 SEE K-CLASS ESTIMATION'
C     READ(*,*) FLS, FMD
C     WRITE(*,*) 'ITERATIONS & ACCURACY IN MAD; IDEALLY 100 AND 0.005'
C     READ(*,*) ITMX, ACCRC
C     OPEN(7, FILE=FILT(5), STATUS='OLD')
C     READ(7,*) NH, NK, NN, IRL

```

```

WRITE(*,*) 'READ FROM FILT5 NH NK NN IRL',NH,NK,NN,IRL
WRITE(*,*) 'NOW READS YORDij, ZORDik FROM FILE - YORD'
OPEN(11,FILE=FILT(3),STATUS='OLD')
DO 12 I=1,NH
  READ(11,*) (YORD(I,J),J=1,NH), (ZORD(I,J),J=1,NK)
  WRITE(*,41) (YORD(I,J),J=1,NH), (ZORD(I,J),J=1,NK)
12 CONTINUE
41 FORMAT(1X,20I3)
  CLOSE(11)
  NSUM=0
  DO 124 I=1,NH
    DO 124 J=1,NH
      IF(YORD(I,J).EQ.1) NSUM=NSUM+1
124 CONTINUE
    DO 125 I=1,NH
      DO 125 J=1,NK
        IF(ZORD(I,J).EQ.1) NSUM=NSUM+1
125 CONTINUE
C ----- NOW READS TRUE Y AND TRUE Z FROM FILES -----
  IRN=IRL*NN
  OPEN(7,FILE=FILT(2),STATUS='OLD',ACCESS='DIRECT',RECL=IRN)
  DO 13 J=1,NH
    READ(7,REC=J) (Y(I,J),I=1,NN)
13 CONTINUE
    CLOSE(7)
    OPEN(8,FILE=FILT(1),STATUS='OLD',ACCESS='DIRECT',RECL=IRN)
    DO 14 J=1,NK
      READ(8,REC=J) (Z(I,J),I=1,NN)
      DO 14 I=1,NN
        ZF(I,J)=Z(I,J)
14 CONTINUE
    CLOSE(8)
    WRITE(*,*) 'READING OF TRUE Y AND Z OVER. STATEMENT 69'
C -----
  6 FORMAT(10E15.5)
  OPEN(11,FILE=FILT(4),STATUS='OLD',ACCESS='DIRECT',RECL=IRL)
  READ(11,REC=1) RCL
  WRITE(*,*) 'RCL =',RCL
  IRCL=1
  DO 7 I=1,NH
    DO 7 J=1,NK
      IRCL=IRCL+1
      READ(11,REC=IRCL) PIE(J,I)
  7 CONTINUE
    DO 77 I=1,NH
      DO 77 J=1,NH
        IRCL=IRCL+1
        READ(11,REC=IRCL) TCA(J,I)
  77 CONTINUE
      DO 78 I=1,NH
        DO 78 J=1,NK
          IRCL=IRCL+1
          READ(11,REC=IRCL) TCB(J,I)
  78 CONTINUE
    CLOSE(11)
    IRLA=IRL*NK
    IRLY=IRL*NN

```

```

11 CONTINUE
   CLOSE (9)
   CLOSE (12)
   CLOSE (13)
   WRITE (*,*) 'TRUE PI A AND B READ. STATEMENT 98'
   DO 40 I=1,NH
     WRITE (*,*) (TCA (J, I) , J=1, NH)
     WRITE (*,*) (TCB (J, I) , J=1, NK)
40 CONTINUE
C *****
  OPEN (10, FILE=FILT (6) , STATUS='OLD' )
  READ (10, *)  NH, NK, NTRY, NN
  CLOSE (10)
  WRITE (*,*) 'READ FROM FILT (6) ', NH, NK, NTRY, NN
C *****
C 2-STAGE LS ANS 2-STAGE MAD ESTIMATION -----
  WRITE (*,*) 'ENTERS TWO-STAGE'
  TTOL=TOL
  INON=1
  OPEN (9, FILE=FILS (6) , STATUS='NEW' )
  DO 36 ILSM=1, 2
    IF (ILSM.EQ.1) WRITE (9,*) 'SECOND STAGE BASED ON RF BY OLS'
    IF (ILSM.EQ.2) WRITE (9,*) 'SECOND STAGE BASED ON RF BY MAD'
    IOLS=1+ILSM
    OPEN (12, FILE=FILS (4) , STATUS='OLD' , ACCESS='DIRECT' , RECL=IRLY)
    OPEN (13, FILE=FILS (5) , STATUS='NEW' )
    DO 21 MEQN=1, NH
      WRITE (*,*) 'ILSM IOLS MEQN' , ILSM, IOLS, MEQN
      ICN=0
      DO 30 J=1, NH
        IF (YORD (MEQN, J) .GT.0) THEN
          ICN=ICN+1
          AZ (ICN)=TCA (J, MEQN)
        ENDIF
30 CONTINUE
      DO 31 J=1, NK
        IF (ZORD (MEQN, J) .GT.0) THEN
          ICN=ICN+1
          AZ (ICN)=TCB (J, MEQN)
        ENDIF
31 CONTINUE
      WRITE (13, *) ICN, (AZ (J) , J=1, ICN)
      WRITE (*,*) ICN, (AZ (J) , J=1, ICN)
C -----
      WRITE (*,*) 'TRUE STR COEFF'
      DO 22 NT=1, NTRY
        IZ=0
        DO 23 JYRD=1, NH
          IF (YORD (MEQN, JYRD) .EQ.-1) THEN
            IRY=3* ((JYRD-1) *NTRY+ (NT-1) )+INON
            WRITE (*,*) 'MEQN JYRD IRY = ' , MEQN, JYRD, IRY
            READ (12, REC=IRY) (YO (I) , I=1, NN)
            ENDIF
            IF (YORD (MEQN, JYRD) .EQ.1) THEN
              IZ=IZ+1
              J=3* ((JYRD-1) *NTRY+ (NT-1) )+IOLS
C      WRITE (*,*) 'MEQN JYRD J = ' , MEQN, JYRD, J

```

```

      READ (12,REC=J) (Z(I,IZ),I=1,NN)
      ENDIF
23  CONTINUE
      NENDO=IZ
      DO 24 J=1,NK
      IF (ZORD(MEQN,J).EQ.1) THEN
      IZ=IZ+1
      DO 25 I=1,NN
C      NOTE THIS CHANGE IN SIGN -----
      Z(I,IZ)=-ZF(I,J)
25  CONTINUE
      ENDIF
24  CONTINUE
      MPD=IZ
      NEXO=IZ-NENDO
C      DO 26 I=1,NN
C      WRITE(*,27) YO(I), (Z(I,J),J=1,NENDO), (Z(I,J),J=NENDO+1,MPD)
C 26  CONTINUE
C      WRITE(*,*) 'NENDO NEXO NENDO+NEXO ',NENDO,NEXO,NENDO+NEXO
27  FORMAT(1X,10D13.5)
      WRITE(*,*) MEQN,'th Equation &',NT,'th Trial'
C      ##### GOES TO MAD ESTIMATION #####
      DO 28 I=1,NN
      Y1(I)=YO(I)
      DO 28 JZ=1,MPD
      Z1(I,JZ)=Z(I,JZ)
28  CONTINUE
      TOL=TTOL
C      WRITE(*,*) 'CALLS MAD MPD=',MPD
      CALL MAD(YO,Z,Y1,Z1,MPD,NN,TOL,ITMX,AZ,YH,ER,RSQ,BZ)
      WRITE(13,*) MPD,(BZ(J),J=1,MPD)
      WRITE(13,*) MPD,(AZ(J),J=1,MPD)
C      WRITE(*,*) MPD,(BZ(J),J=1,MPD)
C      WRITE(*,*) MPD,(AZ(J),J=1,MPD)
22  CONTINUE
21  CONTINUE
      CLOSE(13)
      CLOSE(12)
C      *****
      OPEN(13,FILE=FILS(5),STATUS='OLD')
      DO 32 MEQN=1,NH
      WRITE(9,*) 'STRUCTURAL EQUATION #',MEQN,' ===== '
      READ(13,*) M,(AA(J),J=1,M)
      OLSN=0.0
      AMADN=0.0
      DO 33 J=1,MX(M,NHMX)
      OLSS(J)=0.0
      MADN(J)=0.0
      AVA(J)=0.0
      AVB(J)=0.0
33  CONTINUE
      DO 34 I=1,NTRY
      READ(13,*) M,(BZ(J),J=1,M)
      READ(13,*) M,(AZ(J),J=1,M)
      DO 34 J=1,MX(M,NHMX)
      OLSS(J)=OLSS(J)+(BZ(J)-AA(J))**2
      MADN(J)=MADN(J)+(AZ(J)-AA(J))**2

```

```

      AVA(J)=AVA(J)+AZ(J)
      AVB(J)=AVB(J)+BZ(J)
34  CONTINUE
      DO 35 J=1,MX(M,NHMX)
      OLSN=OLSN+OLSS(J)
      AMADN=AMADN+MADS(J)
      AVA(J)=AVA(J)/NTRY
      AVB(J)=AVB(J)/NTRY
35  CONTINUE
      RNORM=DSQRT(OLSN/AMADN)
      DSTL(MEQN,ILSM)=OLSN
      DSTM(MEQN,ILSM)=AMADN
      WRITE(9,*) 'FILE NAME = ',FILS(5),'--',FILS(6)
      WRITE(9,*) 'Values: NN - M - TOL - NTRY - FLS - FMD'
      WRITE(9,112) NN,M,TTOL,NTRY,FLS,FMD
112  FORMAT(6X,I5,I5,F7.4,2X,I5,2F7.3)
      WRITE(9,111) NAM(1),(AA(J),J=1,MX(M,NHMX))
      WRITE(9,111) NAM(2),(AVB(J),J=1,MX(M,NHMX))
      WRITE(9,111) NAM(3),(AVA(J),J=1,MX(M,NHMX))
      WRITE(9,111) NAM(4),(DSQRT(OLSS(J)/FLOAT(NTRY)),J=1,MX(M,NHMX))
      WRITE(9,111) NAM(5),(DSQRT(MADS(J)/FLOAT(NTRY)),J=1,MX(M,NHMX))
      WRITE(9,111) NAM(6),(DSQRT(OLSS(J)/MADS(J)),J=1,MX(M,NHMX))
      WRITE(9,111) NAM(7),RNORM
111  FORMAT(1X,A5,2X,15F10.3)
32  CONTINUE
C *****
  CLOSE(13)
  WRITE(9,*) ' '
  WRITE(9,*) '===== '
36  CONTINUE
      OPEN(10,FILE=SL2N,STATUS='NEW')
      WRITE(10,*) (DSTL(JJ,1),JJ=1,NH),NHMX
      CLOSE(10)
      DO 122 J=1,4
122  SDST(J)=0.0
      DO 121 I=1,NH
      SDST(1)=SDST(1)+DSTL(I,1)
      SDST(2)=SDST(2)+DSTL(I,2)
      SDST(3)=SDST(3)+DSTM(I,1)
      SDST(4)=SDST(4)+DSTM(I,2)
121  CONTINUE
      DO 117 INPRS=1,2
118  WRITE(9,*) 'OLS-RF BASED OLS AND MAD NORMS OF STRUCT COEFF'
      DO 114 J=1,2
      WRITE(9,113) (DSTL(I,J),I=1,NH)
114  CONTINUE
      WRITE(9,*) 'MAD-RF BASED OLS AND MAD NORMS OF STRUCT COEFF'
      DO 115 J=1,2
      WRITE(9,113) (DSTM(I,J),I=1,NH)
115  CONTINUE
      IF(INPRS.LT.2) THEN
      DO 116 I=1,NH
      DSTT=DSTL(I,1)
      DO 116 J=1,2
      DSTL(I,J)=DSQRT(DSTL(I,J)/DSTT)
      DSTM(I,J)=DSQRT(DSTM(I,J)/DSTT)
116  CONTINUE

```

```

ENDIF
IF (INPRS.EQ.1) THEN
WRITE(9,*) '-----'
WRITE(9,*) 'RELATIVE (to 2SLS) NORMS: LSMAD,MADLS,MADMAD'
WRITE(9,*) '-----'
ENDIF
117 CONTINUE
113 FORMAT(10D12.4)
DSTT=SDST(1)
WRITE(9,*) 'RELATIVE (to 2-sls) NORMS ARE AS FOLLOWS'
WRITE(9,*) 'LS-RF MAD STR to 2-SLS =',DSQRT(DSTT/SDST(2))
WRITE(9,*) 'MAD-RF LS STR to 2-SLS =',DSQRT(DSTT/SDST(3))
WRITE(9,*) 'MAD-RF MAD STR to 2-SLS=',DSQRT(DSTT/SDST(4))
WRITE(9,*) 'STANDARD 2-SLS =', DSQRT(DSTT/SDST(1))
WRITE(9,*) '2-SLS NORM=',DSQRT(DSTT/(NTRY*(NSUM-NH*NHMX)))
CLOSE(9)
WRITE(*,*) ' 2S-LS/MAD HAVE RUN. LOOK INTO file:',FILS(6)
WRITE(*,*) 'YOU MAY RUN GILSX PROGRAM TO OBTAIN GILS ESTIMATES'
WRITE(*,*) ' THANK YOU !'
END

C -----
C Central Program for MAD estimation using Iterated Weighted LS
SUBROUTINE MAD(YO,XO,Y,X,M,N,TOL,ITMX,A,YH,ER,RSQ,B)
COMMON /CB1/ML,NL,ACCRC
DOUBLE PRECISION YO(NL),XO(NL,ML),Y(NL),X(NL,ML),A(ML),ER(NL)
DOUBLE PRECISION YH(NL),B(ML),C(10)
DOUBLE PRECISION TOL,RSQ,ASUM,ABSUM,W,ACCRC,ACCRCC
ACCRCC=ACCRC
10 FORMAT(1X,12D12.5)
1234 DF=FLOAT(N-M)
NACC=0
ITER=0
CALL OLS(YO,XO,Y,X,N,M,A,YH,ER,RSQ)
C -----
DO 1 J=1,M
C(J)=A(J)
1 B(J)=A(J)
100 ITER=ITER+1
IF(ITER.GT.ITMX) THEN
C WRITE(*,*) 'NO CONVERGENCE IN ITERATIONS =',ITMX
GOTO 7
ENDIF
ABSUM=0
NCOUNT=0
DO 2 I=1,N
ER(I)=DABS(ER(I))
ABSUM=ABSUM+ER(I)
W=0.0
IF(ER(I).GT.ACCRCC) THEN
W=1.0/ER(I)
NCOUNT=NCOUNT+1
ENDIF
Y(I)=YO(I)*W
DO 3 J=1,M
X(I,J)=XO(I,J)*W
3 C(J)=A(J)
2 CONTINUE

```

```

      IF ( (NCOUNT-(M+3)) .LT. 0 ) THEN
      ACCRC=ACCRC/10.0
      GOTO 1234
      ENDIF
      IF (ITER.EQ.1) ERSUM=ABSUM
      CALL OLS (YO,XO,Y,X,N,M,A,YH,ER,RSQ)
      ASUM=0
      DO 4 I=1,N
      ER(I)=DABS(ER(I))
4      ASUM=ASUM+ER(I)
      DO 6 J=1,M
      IF (DABS(C(J)-A(J)) .GT.TOL) GOTO 100
6      CONTINUE
C      -----GOES OUT FROM ITERATIVE LOOP-----
7      CONTINUE
C      -----
      WRITE(*,*) 'TOTAL NUMBER OF ITERATIONS PERFORMED ',ITER-1
      WRITE(*,*) 'OLS AND MAD DEVIATION SUMS ARE ',ERSUM,ASUM
      ACCRC=ACCRC
      RETURN
      END
C      -----
C      Central Program for Least Squares Estimation of Regression Eqn.
      SUBROUTINE OLS (YO,XO,Y,X,N,M,A,YH,ER,RSQ)
      COMMON /CB1/ML,NL,ACCRC
      DOUBLE PRECISION Y(NL),X(NL,ML),A(ML),ER(NL),YO(NL),XO(NL,ML)
      DOUBLE PRECISION YH(NL),SE,ES,S,SS,RSQ,ACCRC
      DOUBLE PRECISION XX(10,10),XY(10)
12      FORMAT(12D12.5)
      DF=FLOAT(N-M)
      DO 1 J=1,M
      XY(J)=0.0
      DO 2 K=J,M
      XX(J,K)=0.0
      DO 10 I=1,N
10      XX(J,K)=XX(J,K)+X(I,J)*X(I,K)
      XX(K,J)=XX(J,K)
      2      CONTINUE
      DO 1 I=1,N
      XY(J)=XY(J)+X(I,J)*Y(I)
1      CONTINUE
      CALL MINV(XX,M)
      DO 3 J=1,M
      A(J)=0.0
      DO 3 K=1,M
3      A(J)=A(J)+XX(J,K)*XY(K)
      SE=0.0
      ES=0.0
      S=0.0
      SS=0.0
      DO 4 I=1,N
      YH(I)=0.0
      DO 5 J=1,M
5      YH(I)=YH(I)+XO(I,J)*A(J)
      S=S+YH(I)
      SS=SS+YH(I)*YH(I)
      ER(I)=YO(I)-YH(I)

```

```

      SE=SE+ER(I)
      ES=ES+ER(I)*ER(I)
4  CONTINUE
      S=S/FLOAT(N)
      SE=SE/FLOAT(N)
      RSQ=(SS/FLOAT(N)-S*S)/( (SS/FLOAT(N)-S*S)+(ES/FLOAT(N)-SE*SE) )
      RETURN
      END
C  ----- MATRIX INVERSION -----
      SUBROUTINE MINV(AAA,N)
      INTEGER P
      DOUBLE PRECISION A,B,S,G,Y,Z,W,PP,PP1,PP2,PP3,PP4,PP5,PP6
      DOUBLE PRECISION SKID,SKNORM
      DOUBLE PRECISION AAA(10,10),CII(10,10),X(10,10)
      DIMENSION A(10,10),B(10),S(10),Y(10),Z(10),W(10)
      DO 1234 I=1,N
      DO 1234 J=1,N
1234  A(I,J)=AAA(I,J)
      DO 58 I=1,N
      W(I)=B(I)
      DO 58 J=1,N
58  X(I,J)=A(I,J)
      PP1=0
      NN=N*N
      DO 2 I=1,N
      DO 2 J=1,N
2  PP1=PP1+DABS(A(I,J))
      PP2=PP1/NN
      INDEX=0
      DO 33 I=1,N
33  S(I)=0
      DO 506 J=1,N
      DO 506 I=1,N
      IF (I.GT.J) GO TO 503
      PP=0
      I1=I-1
      DO 501 P=1,I1
      IF (I1.LT.1) GO TO 502
501  PP=PP+A(I,P)*A(P,J)
      A(I,J)=A(I,J)-PP
502  IF (I.NE.J) GO TO 506
      IF (DABS(A(J,J)).LT.(PP2*.1D-08)) GO TO 507
      GO TO 506
503  PP=0
      J1=J-1
      DO 504 P=1,J1
      IF (J1.LT.1) GO TO 505
504  PP=PP+A(I,P)*A(P,J)
505  Y(I)=(A(I,J)-PP)/A(J,J)
      IF (I.EQ.N) GO TO 508
      GO TO 506
508  JJ1=J+1
      DO 1 P=JJ1,N
      IF (DABS(Y(P)).GT.(10.0*PP1)) GO TO 507
1  CONTINUE
      DO 4 L=JJ1,N
4  A(L,J)=Y(L)

```



```

        GO TO 506
C 507 I=J
507 S(J)=PP2
    A(J,J)=A(J,J)+PP2
    INDEX=INDEX+1
506 CONTINUE
    N1=N-1
    DO 17 J=1,N1
        J1=J+1
        DO 17 I=J1,N
            PP=0
            I1=I-1
            IF (J1.GT.I1) GO TO 17
            DO 18 P=J1,I1
18 PP=PP+A(I,P)*A(P,J)
17 A(I,J)=-A(I,J)-PP
21 DO 84 L=1,N
    J=N-L+1
    DO 84 M=1,J
        I=J-M+1
        IF ((I+1).GT.J) GO TO 71
        I1=I+1
        PP=0
        DO 70 P=I1,J
70 PP=PP+A(I,P)*A(P,J)
    A(I,J)=-PP/A(I,I)
    GO TO 84
71 A(I,J)=1.0/A(I,J)
84 CONTINUE
    N1=N-1
    DO 24 J=1,N1
        DO 24 I=1,N
            IF (I.GT.J) GO TO 27
            PP=0
            J1=J+1
            DO 26 P=J1,N
26 PP=PP+A(I,P)*A(P,J)
            A(I,J)=A(I,J)+PP
            GO TO 24
27 PP=0
            DO 28 P=I,N
28 PP=PP+A(I,P)*A(P,J)
            A(I,J)=PP
24 CONTINUE
            IF (INDEX.EQ.0) GO TO 59
            DO 121 L=1,N
                K=N-L+1
                IF (DABS(S(K)).LT. 0.1D-08) GO TO 121
                DO 90 J=1,N
                    Z(J)=A(K,J)
90 Y(J)=A(J,K)
                G=1.0-S(K)*A(K,K)
                IF (DABS(G).LT. 0.1D-13) GO TO 7178
                DO 91 I=1,N
                    DO 91 J=1,N
91 A(I,J)=A(I,J)+S(K)*Y(I)*Z(J)/G
121 CONTINUE

```

```

59 DO 45 I=1,N
    PP=0
    DO 46 P=1,N
46 PP=PP+A(I,P)*B(P)
45 Y(I)=PP
    DO 47 I=1,N
47 B(I)=Y(I)
    GO TO 1601
7178 CONTINUE
1601 PP=0
    PP1=0
    PP2=0
    DO 60 I=1,N
    PP2=PP2+W(I)*W(I)
    DO 60 J=1,N
    PP=PP+A(I,J)*A(I,J)
60 PP1=PP1+X(I,J)*X(I,J)
    PP=DSQRT(PP)
    PP1=DSQRT(PP1)
    PP2=DSQRT(PP2)
    DO 57 I=1,N
    PP4=0
    DO 73 P=1,N
73 PP4=PP4+X(I,P)*B(P)
57 W(I)=W(I)-PP4
    PP3=0
    DO 56 I=1,N
56 PP3=PP3+W(I)*W(I)
    PP3=DSQRT(PP3)
    DO 55 I=1,N
    DO 50 J=1,N
    PP4=0
    DO 54 P=1,N
54 PP4=PP4+X(I,P)*A(P,J)
50 W(J)=PP4
    DO 55 P=1,N
55 X(I,P)=W(P)
    DO 53 I=1,N
53 X(I,I)=1.0-X(I,I)
    PP4=0
    DO 52 I=1,N
    DO 52 J=1,N
52 PP4=PP4+X(I,J)*X(I,J)
    PP4=DSQRT(PP4)
    PP5=PP3/PP1
    PP6=PP*PP4*PP2/(1.0-PP4)
    DO 1236 I=1,N
    DO 1236 J=1,N
    CII(I,J)=0.0
    DO 1236 K=1,N
    CII(I,J)=CII(I,J)+AAA(I,K)*A(K,J)
1236 CONTINUE
    SKNORM=0.0
    DO 1237 I=1,N
    DO 1237 J=1,N
    SKID=0.0
    IF(I.EQ.J) SKID=1.0

```

```

      SKNORM=SKNORM+ (CII (I, J) -SKID) **2
1237 CONTINUE
      SKNORM=DSQRT (SKNORM/ (N*N) )
C -----
C      IF (SKNORM.GT.0.0000001) THEN
C      WRITE (*, *) 'INV MATRIX IS INACCURATE. NORM=', SKNORM
C      PAUSE 'PROCEED OR ABORT'
C      ENDIF
C -----
1238 FORMAT (10F8.5)
      DO 1235 I=1,N
      DO 1235 J=1,N
1235 AAA (I, J) =A (I, J)
      RETURN
      END

C      PROG3.FOR
C ----- MAIN GILS1 -----
COMMON NKV, NHV, NV, NHK, NO, IRL
INTEGER *2 SLX (15), SLY (10), YORD (10, 10), ZORD (10, 10), ITEMP
INTEGER *2 IDX (11, 10), IDY (11, 10)
DOUBLE PRECISION PIE (10, 10), A (10, 10), B (10, 10), SA (10), SB (10)
DOUBLE PRECISION AT (10, 10), BT (10, 10), SLS2 (10), P2 (10)
DOUBLE PRECISION YJ (10), XJ (10), PIET (10, 10), PT (10)
DOUBLE PRECISION P21 (10, 10), P (10, 10), PIEX (10, 10), PIEM (10, 10)
DOUBLE PRECISION SGB (10, 10), SOB (10, 10), SGA (10, 10), SOA (10, 10)
DOUBLE PRECISION SSGB (10, 10), SSOB (10, 10), SSGA (10, 10), SSOA (10, 10)
DOUBLE PRECISION ONR (10, 2), TEMP, FN, ZAA, ZLS2
CHARACTER FILT (6) *6, SL2 *6, GIL *6, FILS (6) *11, NAM (7) *5
C -----
      FN (ZAA, NAA) =ZAA/NAA
      FILT (3) = 'YORD'
      FILT (4) = 'TRUC'
      FILT (6) = 'PARA2'
      FILS (1) = 'MMMM'
      GIL = 'GILN'
      SL2 = 'SL2N'
C -----
      NV=50
      NHV=10
      NKV=10
      IRL=8
      MC=7
      NU=11
C ----- STRING VARIABLES -----
      NAM (1) = 'TRUEC'
      NAM (2) = 'MOLSC'
      NAM (3) = 'MMADC'
      NAM (4) = 'LSRMS'
      NAM (5) = 'MDRMS'
      NAM (6) = 'LMRMS'
      NAM (7) = 'GLRMS'
C -----
      OPEN (10, FILE=FILT (6), STATUS='OLD')

```

```

C      READS --- NH  NK  NTRY and NN from file #10 FILT(6) '
      READ (10,*) NH,NK,NTRY,NN
      CLOSE (10)
      OPEN (10,FILE=SL2)
      READ (10,*) (SLS2 (JJ) , JJ=1,NH) , NHMX
      CLOSE (10)
C      ----- READS YORD AND ZORD MATRICES -----
      OPEN (11,FILE=FILT(3) , STATUS='OLD' )
      DO 1 I=1,NH
      READ (11,*) (YORD (I, J) , J=1, NH) , (ZORD (I, J) , J=1, NK)
1      CONTINUE
      CLOSE (11)
      DO 71 I=1,NH
      IDN=0
      DO 72 J=1, NH
      IF (YORD (I, J) .EQ.1) THEN
      IDN=IDN+1
      IDY (IDN, I) =J
      ENDIF
72      CONTINUE
      IDY (NU, I) =IDN
      IDN=0
      DO 73 J=1, NK
      IF (ZORD (I, J) .EQ.1) THEN
      IDN=IDN+1
      IDX (IDN, I) =J
      ENDIF
73      CONTINUE
      IDX (NU, I) =IDN
71      CONTINUE
C      ----- STARTS GILN -----
      DO 213 I=1, NK
      DO 213 J=1, NH
      SGB (I, J) =0.0
      SOB (I, J) =0.0
      SSGB (I, J) =0.0
      SSOB (I, J) =0.0
213      CONTINUE
      DO 214 I=1, NH
      DO 214 J=1, NH
      SGA (I, J) =0.0
      SOA (I, J) =0.0
      SSGA (I, J) =0.0
      SSOA (I, J) =0.0
214      CONTINUE
C      ----- READS TRUE PIE, ATRUE AND BTRUE-----
      OPEN (11,FILE=FILT(4) , STATUS='OLD' , ACCESS='DIRECT' , RECL=IRL)
      IRCL=1
      DO 2 I=1, NH
      DO 2 J=1, NK
      IRCL=IRCL+1
      READ (11,REC=IRCL) PIE (J, I)
      PIET (J, I) =PIE (J, I)
2      CONTINUE
      DO 3 I=1, NH
      DO 3 J=1, NH
      IRCL=IRCL+1

```

```

      READ (11,REC=IRCL) AT (J, I)
3  CONTINUE
      DO 4 I=1, NH
      DO 4 J=1, NK
      IRCL=IRCL+1
      READ (11,REC=IRCL) BT (J, I)
4  CONTINUE
      CLOSE (11)
701 FORMAT (1X, 10D16.6)
C  ----- READING ERRONEOUS PIES TO USE IN GILN -----
      IRLA=IRL*NK
C  WRITE (*,*) 'NO. OF TRIALS ? AND NLS (1 OLS 2 MAD) '
C  READ (*,*) NLS
      OPEN (13, FILE=FILS (1), ACCESS='DIRECT', STATUS='OLD', RECL=IRLA)
      DO 205 NLS=1, 2
      IRAL=0
      DO 200 NT=1, NTRY
      DO 201 JEQ=1, NH
      IRAL=(JEQ-1)*NTRY*2+JEQ
      READ (13, REC=IRAL) (PIEM (J, JEQ), J=1, NK)
      IRAP=IRAL+(NT-1)*2+NLS
      READ (13, REC=IRAP) (PIE (J, JEQ), J=1, NK)
      DO 201 J=1, NK
      PIE (J, JEQ) = -PIE (J, JEQ)
      PIEM (J, JEQ) = -PIEM (J, JEQ)
      PIEX (J, JEQ) = PIE (J, JEQ)
201 CONTINUE
      DO 202 J=1, NK
      WRITE (*, 203) (PIEM (J, JEQ), JEQ=1, NH)
      WRITE (*, 203) (PIET (J, JEQ), JEQ=1, NH)
202 CONTINUE
203 FORMAT (1X, 10D16.5)
C  PAUSE 'TWO TRUE PIE MATRICES '
      DO 100 IEQ=1, NH
      DO 5 J=1, NH
      YJ (J) = YORD (IEQ, J)
      SLY (J) = J
5  CONTINUE
      DO 6 J=1, NK
      XJ (J) = ZORD (IEQ, J)
      SLX (J) = J
6  CONTINUE
C  ----ARRANGE YJ IN A DESCENDING ORDER ACCORDINGLY SLY AND PIE
      DO 7 J=1, NH-1
      DO 7 JJ=J+1, NH
      IF (YJ (J) .LT. YJ (JJ)) THEN
      TEMP=YJ (J)
      YJ (J)=YJ (JJ)
      YJ (JJ)=TEMP
      ITEMP=SLY (J)
      SLY (J)=SLY (JJ)
      SLY (JJ)=ITEMP
      DO 8 I=1, NK
      TEMP=PIE (I, J)
      PIE (I, J)=PIE (I, JJ)
      PIE (I, JJ)=TEMP
8  CONTINUE

```

```

      ENDIF
7  CONTINUE
C  ----ARRANGE XJ IN A DESCENDING ORDER ACCORDINGLY SLX AND PIE
      DO 9 J=1,NK-1
      DO 9 JJ=J+1,NK
      IF (XJ(J) .LT. XJ(JJ)) THEN
      TEMP=XJ(J)
      XJ(J)=XJ(JJ)
      XJ(JJ)=TEMP
      ITEMP=SLX(J)
      SLX(J)=SLX(JJ)
      SLX(JJ)=ITEMP
      DO 10 I=1,NH
      TEMP=PIE(J,I)
      PIE(J,I)=PIE(JJ,I)
      PIE(JJ,I)=TEMP
10  CONTINUE
      ENDIF
      9  CONTINUE
102  FORMAT(2X,10(F2.0,8X))
C  DO 11 I=1,NK
C  ISI=I
C  11  WRITE(*,101) ISI, (PIET(I,J), J=1,NH)
C  WRITE(*,*) '-----'
C  WRITE(*,*) ' ', (SLY(J), J=1,NH)
C  DO 12 I=1,NK
C  12  WRITE(*,101) SLX(I), (PIE(I,J), J=1,NH)
C  PAUSE
C  -----
      NH1=0
      NK1=0
      DO 14 I=1,NH
      IF (YJ(I) .EQ. 1) THEN
      NH1=NH1+YJ(I)
      ENDIF
14  CONTINUE
      DO 15 I=1,NK
      IF (XJ(I) .EQ. 1.0) THEN
      NK1=NK1+XJ(I)
      ENDIF
15  CONTINUE
      NH2=NH-NH1
      NK2=NK-NK1
C  WRITE(*,*) 'NH1 NH2 NK1 NK2 ', NH1, NH2, NK1, NK2
C  WRITE(*,*) 'YJ ', (YJ(J), J=1,NH)
C  WRITE(*,*) 'XJ ', (XJ(J), J=1,NK)
C  PAUSE
      DO 16 I=1,NK2
      DO 16 J=1,NH1
16  P21(I,J)=PIE(I+NK1,J)
C  WRITE(*,*) ( (P21(I,J), J=1,NH1), I=1,NK2)
C  PAUSE 'P21 MATRIX'
      DO 17 J=1,NH1
      DO 17 JJ=1,NH1
      P(J,JJ)=0.0
      DO 17 I=1,NK2
17  P(J,JJ)=P(J,JJ)+P21(I,J)*P21(I,JJ)

```

```

      CALL MINV(P,NH1)
      DO 18 I=1,NK2
      P2(I) = -XJ(I+NK1)
      DO 18 J=1,NH2
18 P2(I)=P2(I)-PIE(I+NK1,J+NH1)*YJ(J+NH1)
C   WRITE(*,*) 'P2', (P2(I), I=1,NK2)
C   PAUSE
      DO 19 I=1,NH1
      PT(I)=0.0
      DO 19 J=1,NK2
19 PT(I)=PT(I)+P21(J,I)*P2(J)
      DO 20 I=1,NH1
      YJ(I)=0.0
      DO 20 J=1,NH1
20 YJ(I)=YJ(I)+P(I,J)*PT(J)
      DO 21 I=1,NK1
      XJ(I)=0.0
      DO 21 J=1,NH
21 XJ(I)=XJ(I)-PIE(I,J)*YJ(J)
C   ----- INVERSE PROCESS OF ARRANGEMENT -----
C   ----ARRANGE SLY IN AN ASCENDING ORDER ACCORDINGLY YJ
      DO 77 J=1,NH-1
      DO 77 JJ=J+1,NH
      IF(SLY(J).GT.SLY(JJ)) THEN
      TEMP=YJ(J)
      YJ(J)=YJ(JJ)
      YJ(JJ)=TEMP
      ITEMP=SLY(J)
      SLY(J)=SLY(JJ)
      SLY(JJ)=ITEMP
      ENDIF
77 CONTINUE
C   ----ARRANGE XJ IN A DESCENDING ORDER ACCORDINGLY SLX AND PIE
      DO 99 J=1,NK-1
      DO 99 JJ=J+1,NK
      IF(SLX(J).GT.SLX(JJ)) THEN
      TEMP=XJ(J)
      XJ(J)=XJ(JJ)
      XJ(JJ)=TEMP
      ITEMP=SLX(J)
      SLX(J)=SLX(JJ)
      SLX(JJ)=ITEMP
      ENDIF
99 CONTINUE
C   -----
C   WRITE(*,103) (YJ(J), J=1,NH)
C   WRITE(*,103) (XJ(J), J=1,NK)
C   WRITE(*,*) 'EQUATION #', IEQ
      WRITE(*,103) (YJ(SLY(J)), J=1,NH)
      WRITE(*,103) (AT(J,IEQ), J=1,NH)
      WRITE(*,103) (XJ(SLX(J)), J=1,NK)
      WRITE(*,103) (BT(J,IEQ), J=1,NK)
      IF(NLS.EQ.2) THEN
      DO 206 J=1,NH
      SSGA(J,IEQ)=SSGA(J,IEQ)+YJ(SLY(J))
206 SGA(J,IEQ)=SGA(J,IEQ)+(YJ(SLY(J))-AT(J,IEQ))**2
      DO 207 J=1,NK

```

```

      SSGB(J, IEQ)=SSGB(J, IEQ)+XJ(SLX(J))
207 SGB(J, IEQ)=SGB(J, IEQ)+(XJ(SLX(J))-BT(J, IEQ))**2
      ENDIF
      IF(NLS.EQ.1) THEN
      DO 208 J=1, NH
      SSOA(J, IEQ)=SSOA(J, IEQ)+YJ(SLY(J))
208 SOA(J, IEQ)=SOA(J, IEQ)+(YJ(SLY(J))-AT(J, IEQ))**2
      DO 209 J=1, NK
      SSOB(J, IEQ)=SSOB(J, IEQ)+XJ(SLX(J))
209 SOB(J, IEQ)=SOB(J, IEQ)+(XJ(SLX(J))-BT(J, IEQ))**2
      ENDIF
C      PAUSE 'THE EQUATION IS OVER NEXT STARTS'
C      -----RESTORE ORIGINAL PIE MATRIX FOR THE NEXT EQUATION-----
      DO 13 I=1, NK
      DO 13 J=1, NH
      PIE(I, J)=PIEX(I, J)
      13 CONTINUE
100 CONTINUE
101 FORMAT(1X, I4, 10E10.3)
      WRITE(*,*) 'EXPERIMENT COMPLETED = ', NT
C      PAUSE 'NEXT EXPERIMENT ====='
200 CONTINUE
      WRITE(*,*) NLS
C      PAUSE
205 CONTINUE
C      -----
      TEMP=0.0
      ZAA=0.0
      OPEN(7, FILE=GIL, STATUS='NEW')
      WRITE(7,*) '---- GENERALIZED INDIRECT LEAST NORM RESULTS ----'
      WRITE(7,*) '|||||'
      WRITE(7,*) ' MEAN - OLS/GILN COEFFICIENTS (ENDO)+(EXO) '
      DO 211 I=1, NH
      WRITE(7,*) 'STRUCTURAL COEFFICIENTS FOR EQN=', I
      NH1=IDY(NU, I)
      NK1=IDX(NU, I)
      WRITE(7, 313) NAM(1), (AT(IDY(J, I), I), J=1, NH1),
1 (BT(IDX(J, I), I), J=1, NK1)
      WRITE(7, 313) NAM(2), (FN(SSOA(IDY(J, I), I), NTRY), J=1, NH1),
1 (FN(SSOB(IDX(J, I), I), NTRY), J=1, NK1)
      WRITE(7, 313) NAM(3), (FN(SSGA(IDY(J, I), I), NTRY), J=1, NH1),
1 (FN(SSGB(IDX(J, I), I), NTRY), J=1, NK1)
      WRITE(7, 313) NAM(4), (DSQRT(FN(SOA(IDY(J, I), I), NTRY)), J=1, NH1),
1 (SQRT(FN(SOB(IDX(J, I), I), NTRY)), J=1, NK1)
      WRITE(7, 313) NAM(5), (DSQRT(FN(SGA(IDY(J, I), I), NTRY)), J=1, NH1),
1 (DSQRT(FN(SGB(IDX(J, I), I), NTRY)), J=1, NK1)
      WRITE(7, 313) NAM(6), (DSQRT(SOA(IDY(J, I), I)/SGA(IDY(J, I), I)),
1 J=1, NH1), (DSQRT(SOB(IDX(J, I), I)/SGB(IDX(J, I), I)), J=1, NK1)
211 CONTINUE
313 FORMAT(1X, A5, 1X, 10F12.3)
103 FORMAT(1X, 10F11.6)
      DO 401 I=1, NH
      NH1=IDY(NU, I)
      NK1=IDX(NU, I)
      ONR(I, 1)=0.0
      ONR(I, 2)=0.0
      DO 402 J=1, NH1

```



```

      ONR(I,1)=ONR(I,1)+SOA(IDY(J,I),I)
      ONR(I,2)=ONR(I,2)+SGA(IDY(J,I),I)
402  CONTINUE
      DO 403 J=1,NK1-NHMX
      ONR(I,1)=ONR(I,1)+SOB(IDX(J,I),I)
      ONR(I,2)=ONR(I,2)+SGB(IDX(J,I),I)
403  CONTINUE
      TEMP=TEMP+ONR(I,1)
      ZAA=ZAA+ONR(I,2)
401  CONTINUE
      WRITE(7,*)'-----'
      WRITE(7,*)'RELATIVE (ols/mad coeff) NORMS:Eqn 1,2,...,NH'
      IZERO=0
      WRITE(7,313)NAM(7),(DSQRT(ONR(I,1)/ONR(I,2)),I=1,NH)
      WRITE(7,*)'Relative (OLS/MAD) model norm = ',DSQRT(TEMP/ZAA)
      WRITE(7,*)'2-SLS NORMS RELATIVE TO OLS & MAD NORMS'
      WRITE(7,313)NAM(7),(DSQRT(SLS2(I)/ONR(I,1)),I=1,NH)
      WRITE(7,313)NAM(7),(DSQRT(SLS2(I)/ONR(I,2)),I=1,NH)
      ZLS2=0.0
      DO 404 J=1,NH
      ZLS2=ZLS2+SLS2(J)
404  CONTINUE
      TEMP=DSQRT(ZLS2/TEMP)
      ZAA=DSQRT(ZLS2/ZAA)
      WRITE(7,*)'OVERALL EFFICIENCY OF OLS-BASED GILN =',TEMP
      WRITE(7,*)'OVERALL EFFICIENCY OF MAD-BASED GILN =',ZAA
      WRITE(7,*)'|||||'
      CLOSE(7)
      WRITE(*,*)'RESULTS STORED IN FILE:',GIL
      END
C  ----- MATRIX INVERSION -----
      SUBROUTINE MINV(AAA,N)
      INTEGER P
      DOUBLE PRECISION A,B,S,G,Y,Z,W,PP,PP1,PP2,PP3,PP4,PP5,PP6
      DOUBLE PRECISION SKID,SKNORM
      DOUBLE PRECISION AAA(10,10),CII(10,10),X(10,10)
      DIMENSION A(10,10),B(10),S(10),Y(10),Z(10),W(10)
      DO 1234 I=1,N
      DO 1234 J=1,N
1234  A(I,J)=AAA(I,J)
      DO 58 I=1,N
      W(I)=B(I)
      DO 58 J=1,N
58  X(I,J)=A(I,J)
      PP1=0
      NN=N*N
      DO 2 I=1,N
      DO 2 J=1,N
2  PP1=PP1+DABS(A(I,J))
      PP2=PP1/NN
      INDEX=0
      DO 33 I=1,N
33  S(I)=0
      DO 506 J=1,N
      DO 506 I=1,N
      IF (I.GT.J) GO TO 503
      PP=0

```

```

      I1=I-1
      DO 501 P=1, I1
      IF (I1.LT.1) GO TO 502
501 PP=PP+A(I,P)*A(P,J)
      A(I,J)=A(I,J)-PP
502 IF (I.NE.J) GO TO 506
      IF (DABS(A(J,J)).LT.(PP2*.1D-08)) GO TO 507
      GO TO 506
503 PP=0
      J1=J-1
      DO 504 P=1, J1
      IF (J1.LT.1) GO TO 505
504 PP=PP+A(I,P)*A(P,J)
505 Y(I)=(A(I,J)-PP)/A(J,J)
      IF (I.EQ.N) GO TO 508
      GO TO 506
508 JJ1=J+1
      DO 1 P=JJ1, N
      IF (DABS(Y(P)).GT.(10.0*PP1)) GO TO 507
      1 CONTINUE
      DO 4 L= JJ1, N
      4 A(L,J)=Y(L)
      GO TO 506
C 507 I=J
507 S(J)=PP2
      A(J,J)=A(J,J)+PP2
      INDEX=INDEX+1
506 CONTINUE
      N1=N-1
      DO 17 J=1, N1
      J1=J+1
      DO 17 I=J1, N
      PP=0
      I1=I-1
      IF (J1.GT.I1) GO TO 17
      DO 18 P=J1, I1
      18 PP=PP+A(I,P)*A(P,J)
      17 A(I,J)=-A(I,J)-PP
      21 DO 84 L=1, N
      J=N-L+1
      DO 84 M=1, J
      I=J-M+1
      IF ((I+1).GT.J) GO TO 71
      I1=I+1
      PP=0
      DO 70 P=I1, J
      70 PP=PP+A(I,P)*A(P,J)
      A(I,J)=-PP/A(I,I)
      GO TO 84
      71 A(I,J)=1.0/A(I,J)
      84 CONTINUE
      N1=N-1
      DO 24 J=1, N1
      DO 24 I=1, N
      IF (I.GT.J) GO TO 27
      PP=0
      J1=J+1

```

```

      DO 26 P=J1,N
26  PP=PP+A(I,P)*A(P,J)
      A(I,J)= A(I,J)+PP
      GO TO 24
27  PP=0
      DO 28 P=I,N
28  PP=PP+A(I,P)*A(P,J)
      A(I,J)=PP
24  CONTINUE
      IF (INDEX.EQ.0) GO TO 59
      DO 121 L=1,N
      K=N-L+1
      IF (DABS(S(K)).LT. 0.1D-08) GO TO 121
      DO 90 J=1,N
      Z(J)=A(K,J)
90  Y(J)=A(J,K)
      G=1.0-S(K)*A(K,K)
      IF (DABS(G).LT. 0.1D-13) GO TO 7178
      DO 91 I=1,N
      DO 91 J=1,N
91  A(I,J)=A(I,J)+S(K)*Y(I)*Z(J)/G
121 CONTINUE
59  DO 45 I=1,N
      PP=0
      DO 46 P=1,N
46  PP=PP+A(I,P)*B(P)
45  Y(I)=PP
      DO 47 I=1,N
47  B(I)=Y(I)
      GO TO 1601
7178 CONTINUE
1601 PP=0
      PP1=0
      PP2=0
      DO 60 I=1,N
      PP2=PP2+W(I)*W(I)
      DO 60 J=1,N
      PP=PP+A(I,J)*A(I,J)
60  PP1=PP1+X(I,J)*X(I,J)
      PP=DSQRT(PP)
      PP1=DSQRT(PP1)
      PP2=DSQRT(PP2)
      DO 57 I=1,N
      PP4=0
      DO 73 P=1,N
73  PP4=PP4+X(I,P)*B(P)
57  W(I)=W(I)-PP4
      PP3=0
      DO 56 I=1,N
56  PP3=PP3+W(I)*W(I)
      PP3=DSQRT(PP3)
      DO 55 I=1,N
      DO 50 J=1,N
      PP4=0
      DO 54 P=1,N
54  PP4=PP4+X(I,P)*A(P,J)
50  W(J)=PP4

```

```

      DO 55 P=1,N
55  X(I,P)=W(P)
      DO 53 I=1,N
53  X(I,I)=1.0-X(I,I)
      PP4=0
      DO 52 I=1,N
      DO 52 J=1,N
52  PP4=PP4+X(I,J)*X(I,J)
      PP4=DSQRT(PP4)
      PP5=PP3/PP1
      PP6=PP*PP4*PP2/(1.0-PP4)
      DO 1236 I=1,N
      DO 1236 J=1,N
      CII(I,J)=0.0
      DO 1236 K=1,N
      CII(I,J)=CII(I,J)+AAA(I,K)*A(K,J)
1236 CONTINUE
      SKNORM=0.0
      DO 1237 I=1,N
      DO 1237 J=1,N
      SKID=0.0
      IF(I.EQ.J) SKID=1.0
      SKNORM=SKNORM+(CII(I,J)-SKID)**2
1237 CONTINUE
      SKNORM=DSQRT(SKNORM/(N*N))
C      -----
C      IF(SKNORM.GT.0.0000001) THEN
C      WRITE(*,*)'INV MATRIX IS INACCURATE. NORM=',SKNORM
C      PAUSE 'PROCEED OR ABORT'
C      ENDIF
C      -----
1238 FORMAT(10F8.5)
      DO 1235 I=1,N
      DO 1235 J=1,N
1235 AAA(I,J)=A(I,J)
      RETURN
      END

```

DasGupta, Madhuchhanda (2004) LEAST ABSOLUTE DEVIATION ESTIMATION OF MULT-EQUATION LINEAR ECONOMETRIC MODELS : A STUDY BASED ON MONTE CARLO EXPERIMENTS, Doctoral Dissertation, Dept. of Economics, North Eastern Hill University, Shillong (India); Supervisor – Prof. SK Mishra

APPENDIX-4

SPORADIC ERRORS IN EXPLANATORY VARIABLES

Sporadic errors in X (where sample X is true $X + \Xi$, and Ξ is a sparse matrix with nonzero elements substantial in size) also vitiate OLS estimation. There too, LAD performs well. We have carried out some experiments on this problem.

Table 1: Basic Variables in the Regression Model						
Sl. No.	True X^* Matrix			True Y	Error	Stochastic Y
	X^*_1	X^*_2	X^*_3	Y^*	e	$Y=Y^*+e$
1	60	21	75	1293.8	-15.58	1278.22
2	47	45	80	1466.4	-3.534	1462.866
3	54	35	90	1484.6	-5.238	1479.362
4	22	0	27	398.3	13.635	411.935
5	72	48	85	1770.7	-14.191	1756.509
6	9	44	93	1171.1	-15.17	1155.93
7	9	24	42	632.9	1.419	634.319
8	67	0	15	755.9	2.603	758.503
9	34	69	7	1155.4	17.965	1173.365
10	51	38	97	1532.5	16.363	1548.863
11	17	96	49	1552.3	4.337	1556.637
12	64	43	71	1551.6	-10.142	1541.458
13	3	86	39	1244.9	-5.09	1239.81
14	90	77	25	1893.4	-8.986	1884.414
15	86	36	10	1306.9	9.858	1316.758
16	22	81	72	1576.4	3.446	1579.846
17	56	19	7	811.6	11.419	823.019
18	76	58	36	1616.3	-11.32	1604.98
19	21	80	61	1487.5	-5.25	1482.25
20	59	14	96	1336.7	13.457	1350.157

To describe the problem we explain it with an example. Let true Y ($=Y^*$) be $Y^* = 19.7 + 9.6 X^*_1 + 11.1 X^*_2 + 6.2 X^*_3$; $Y = Y^* + e$, where e is drawn from a normal population with mean = 0 and $\sigma = 10$; see Table 1). The regression coefficients (OLS and LAD estimated) are given in table 2. Four experiments are done by adding Ξ_1 through Ξ_4 to

Table 4: Regression Coefficients of $Y = (X^* + \Xi)B + e$					
Sporadic Disturbance in X	LAD and OLS Estimators	Regression Coefficients Estimated by LAD and OLS with $X = X^* + \Xi$			
	Coefficients (True)	B0 (19.70)	B1 (9.60)	B2 (11.10)	B3 (6.20)
Ξ_1	LAD	24.367	8.791	10.762	6.759
	OLS	75.506	7.461	10.695	6.001
Ξ_2	LAD	44.068	9.445	10.975	5.928
	OLS	278.410	9.649	7.708	3.054
Ξ_3	LAD	55.953	9.221	11.105	5.671
	OLS	344.804	5.628	10.517	3.034
Ξ_4	LAD	44.440	9.422	10.948	5.962
	OLS	670.396	5.190	2.943	3.942

```

C                                     XILADR.FOR
C   Regression in presence of sporadic errors in X, independent variable
C   MAIN PROGRAM =====
COMMON ML,NL,JRRL
DOUBLE PRECISION XI(100,10)
DOUBLE PRECISION YO(100),XO(100,10),Y(100),X(100,10),OLSS(10)
DOUBLE PRECISION YC(100),XC(100,10),ER(100),YH(100),A(10),EPS
DOUBLE PRECISION XX(10,10),XY(10),AA(10),B(10),MADS(10),AVB(10)
DOUBLE PRECISION AVA(10),RSQ,QD,AH(10),AVH(10),HOOK(10),HOOKN
DOUBLE PRECISION TOL,TTOL,SCALE,YM,SDEV,SIZE,OLSN,AMADN,RNORM
INTEGER *2 IUX,IUXI,IU,IV,IUU,SGNS(30)
CHARACTER *11 FIL,RFIL
CHARACTER *5 NAM(10)
C   =====
C   PARAMETERS FOR DIMENSION: CHANGE THEM IF THE PROBLEM IS LARGER
C   EPS=0.00000001
C   NL=100
C   ML=10
C   IXI=0
C   FIL='MYFIL'
C   RFIL='RMYFIL'
C   CHANGE THE FORMAT IF REQUIRED
10 FORMAT(8F10.2)
C   -----
C   WRITE(*,*) 'FEED NO. OF OBSERVATIONS AND NO. OF VARIABLES'

C   READ(*,*) N,M
C   WRITE(*,*) 'FEED THE TOLERANCE LEVEL FOR ACCURACY (.0001)'
C   WRITE(*,*) 'This is required for iterative solution of parameters'
C   READ(*,*) TOL
C   TTOL=TOL
C   WRITE(*,*) 'FEED THE SEED TO GENERATE RANDOM NUMBERS'

C   READ(*,*) IU
C   WRITE(*,*) 'FEED SECOND SEED TO GENERATE RANDOM NUMBERS FOR xi'

```

```

READ(*,*) IUXI
JUXI=IUXI
WRITE(*,*) 'NO OF XI TO GENERATE FOR EACH X, SCALE & SGN CONTROL?'
READ(*,*) NXI, SXIK, POS
WRITE(*,*) 'FEED DIFFERENT SEED RANDOM NUMBERS-RANDOM WALK'
READ(*,*) IUX
IUU=IU
CALL YXGEN(M,N, YC, XC, AA, SCALE, IU)
C DO 2 I=1, N
C WRITE(*,10) YC(I), (XC(I,J), J=1, M)
C 2 CONTINUE
WRITE(*,*) 'FEED STANDARD DEVIATION (SD) OF ERROR TERM AND JRRL.'
WRITE(*,*) 'SD may not exceed 0.3 else R Squares come poor.'
WRITE(*,*) 'FOR JRRL=0 SD IS RELATIVE TO 100 ELSE IT IS RELATIVE'
WRITE(*,*) 'TO AVERAGE VALUE OF Y. Your Options ?'
READ(*,*) SDEV, JRRL
YM=0.0
DO 8 I=1, N
YM=YM+YC(I)
8 CONTINUE
YM=YM/FLOAT(N)
WRITE(*,*) 'AVERAGE Y IS = ', YM
WRITE(*,*) 'NUMBER OF OUTLIERS TO GENERATE AND THEIR SIZE with'
WRITE(*,*) 'respect to average Y. Size may not exceed 1.0'
WRITE(*,*) 'But it is your choice. Size = 2 is quite large'
READ(*,*) NOL, SIZE
WRITE(*,*) 'FEED THE SIGNS -1 OR 1 FOR EACH OUTLIER'
READ(*,*) (SGNS(J), J=1, NOL)
C -----
WRITE(*,*) 'NO. OF EXPERIMENTS (REPLICATES > = 50, say) ? '
READ(*,*) NTRY
C WRITE(*,*) 'NAME TWO FILES TO STORE THE RESULTS ?'
C READ(*,*) FIL, RFIL
OPEN(7, FILE=FIL, STATUS='NEW')
WRITE(7,*) (AA(J), J=1, M)
C ?????????????????????????????????????????????????????????????????
WRITE(*,*) 'CHOOSE ERROR DISTRIBUTION BY ONE OF THE OPTIONS'
WRITE(*,*) 'Normal=0; Cauchy=1; Gamma=2; Beta2=3; Beta1=4'
READ(*,*) ID
IF(ID.GE.2) THEN
WRITE(*,*) 'To generate Gamma or Beta distributed Random Numbers'
WRITE(*,*) 'it requires to generate Gamma from Normal distn.'
WRITE(*,*) 'PARAMETER FOR GAMMA DISTRIBUTION (ig)'
WRITE(*,*) 'It generates G(ig) and Beta1 or Beta2 of (.5ig, .5ig)'
READ(*,*) IG
ENDIF
C ?????????????????????????????????????????????????????????????????
WRITE(*,*) 'SET MAX ITERATION FOR MAD ESTIMATION'
READ(*,*) ITMX
DO 100 IT=1, NTRY
C ***** GENERATE XI MATRIX *****
CALL SXI(XI, N, M, NXI, SXIK, IUXI, POS)
DO 1 I=1, N
YO(I)=YC(I)
Y(I)=YC(I)
DO 1 J=1, M
XO(I, J)=XC(I, J)+XI(I, J)

```



```

      X(I,J)=XC(I,J)+XI(I,J)
1  CONTINUE
C  -----
C  -----GENERATION OF ERRORS AND ADDING THEM TO YC(I)-----
      IF (IXI.EQ.0) THEN
      CALL ERGEN(N,YO,Y,ER,SDEV,IU,IV,ID,IG)
C  -----
      DO 6 I=1,N
      YC(I)=YO(I)
6  CONTINUE
      IXI=1
      ENDIF
C  WRITE(*,*) (Y(I), I=1,N)
C  PAUSE
C  -----GENERATION OF OUTLIER ERROR(S)-----
      IF (NOL.GT.0) CALL OUTLI(YO,Y,N,IU,IV,NOL,SIZE,SGNS)
C  *****
C  -----GOING TO MIN ABS DEVIATION ESTIMATION-----
      CALL MAD(YO,XO,Y,X,M,N,TOL,ITMX,AA,XX,XY,A,YH,ER,RSQ,B)
      WRITE(7,*) (A(J), J=1,M)
C  NOW RANDOM WALK METHOD WILL TRY TO IMPROVE UPON MAD ESTIMATION
      CALL RWALK(A,M,N,YO,XO,IUX,EPS,QD)
      QD=QD/FLOAT(N)
C  *****
C  PAUSE
      WRITE(7,*) (A(J), J=1,M), RSQ, QD
      WRITE(*,99) IT,NTRY
100 CONTINUE
99  FORMAT(1X,/, '----- EXPERIMENT NO.', I5, ' OVER ', I5, ' -----', /)
      CLOSE(7)
      OPEN(7, FILE=FILE)
      READ(7,*) (AA(J), J=1,M)
      OLSN=0.0
      AMADN=0.0
      HOOKN=0.0
      DO 3 J=1,M
      OLSS(J)=0.0
      MADS(J)=0.0
      HOOK(J)=0.0
      AVA(J)=0.0
      AVB(J)=0.0
      AVH(J)=0.0
3  CONTINUE
      DO 4 I=1,NTRY
      READ(7,*) (B(J), J=1,M)
      READ(7,*) (A(J), J=1,M)
      READ(7,*) (AH(J), J=1,M), RSQ, QD
      DO 4 J=1,M
      OLSS(J)=OLSS(J)+(B(J)-AA(J))**2
      MADS(J)=MADS(J)+(A(J)-AA(J))**2
      HOOK(J)=HOOK(J)+(AH(J)-AA(J))**2
      AVA(J)=AVA(J)+A(J)
      AVB(J)=AVB(J)+B(J)
      AVH(J)=AVH(J)+AH(J)
4  CONTINUE
      CLOSE(7)
      DO 5 J=1,M

```

```

      OLSN=OLSN+OLSS (J)
      AMADN=AMADN+MADS (J)
      HOOKN=HOOKN+HOOK (J)
      AVA (J) =AVA (J) /NTRY
      AVB (J) =AVB (J) /NTRY
      AVH (J) =AVH (J) /NTRY
5  CONTINUE
      RNORM=DSQRT (OLSN/AMADN)
      HNORM=DSQRT (OLSN/HOOKN)
      OPEN (9, FILE=RFIL, STATUS='NEW')
C      WRITE (9, *) 'TRUE COEFFICIENTS'
C      WRITE (9, *) (AA (J) , J=1, M)
C      WRITE (9, *) 'MEAN OLS COEFFICIENTS'
C      WRITE (9, *) (AVB (J) , J=1, M)
C      WRITE (9, *) 'MEAN MAD COEFFICIENTS'
C      WRITE (9, *) (AVA (J) , J=1, M)
C      WRITE (9, *) 'OLS ROOT MEAN SQUARE'
C      WRITE (9, *) (DSQRT (OLSS (J) /FLOAT (NTRY) ) , J=1, M)
C      WRITE (9, *) 'MAD ROOT MEAN SQUARE'
C      WRITE (9, *) (DSQRT (MADS (J) /FLOAT (NTRY) ) , J=1, M)
C      WRITE (9, *) 'RATIO OF RMS OLS/MAD'
C      WRITE (9, *) (DSQRT (OLSS (J) /MADS (J) ) , J=1, M)
C      WRITE (9, *) 'OVERALL RATIO OF NORMS (EFFICIENCY OF MAD) = ', RNORM
      NAM(1)='TRUEC'
      NAM(2)='MOLSC'
      NAM(3)='MMADC'
      NAM(4)='LSRMS'
      NAM(5)='MDRMS'
      NAM(6)='LMRMS'
      NAM(7)='GLRMS'
      NAM(8)='HMADC'
      NAM(9)='HMRMS'
      NAM(10)='HLRMS'
      WRITE (9, *) 'FILE NAME = ', RFIL
      WRITE (9, *) ' N - M - TOL - IUU - SDEV - NOL - SIZE - NTRY-
* ID - IG - SCALE'
      WRITE (9, 12) N, M, TTOL, IUU, SDEV, NOL, SIZE, NTRY, ID, IG, SCALE
      WRITE (9, *) 'Sporadic XI parameters = iuxi - nxi - sxik - pos'
      WRITE (9, 77) juxi, nxi, sxik, pos
77  FORMAT (25X, I5, 3X, I4, 1X, F6.2, F6.0)
12  FORMAT (1X, 2I4, F7.4, I6, F7.2, I5, 2X, F6.2, I6, 5X, 2I5, 2X, F5.0)
      IF (NOL.NE.0) WRITE (9, *) 'SIGNS OF OUTLIERS', (SGNS (J) , J=1, NOL)
      IF (NOL.EQ.0) WRITE (9, *) 'NO OUTLIERS HAVE BEEN INTRODUCED'
      WRITE (9, *) '-----'
      WRITE (9, 11) NAM (1) , (AA (J) , J=1, M)
      WRITE (9, 11) NAM (2) , (AVB (J) , J=1, M)
      WRITE (9, 11) NAM (3) , (AVA (J) , J=1, M)
      WRITE (9, 11) NAM (8) , (AVH (J) , J=1, M)
      WRITE (9, 11) NAM (4) , (DSQRT (OLSS (J) /FLOAT (NTRY) ) , J=1, M)
      WRITE (9, 11) NAM (5) , (DSQRT (MADS (J) /FLOAT (NTRY) ) , J=1, M)
      WRITE (9, 11) NAM (9) , (DSQRT (HOOK (J) /FLOAT (NTRY) ) , J=1, M)
      WRITE (9, 11) NAM (6) , (DSQRT (OLSS (J) /MADS (J) ) , J=1, M)
      WRITE (9, 11) NAM (10) , (DSQRT (OLSS (J) /HOOK (J) ) , J=1, M)
      WRITE (9, 11) NAM (7) , RNORM, HNORM
11  FORMAT (1X, A5, 2X, 15F9.3)
      CLOSE (9)
      WRITE (*, *) 'RESULTS STORED IN FILES: ', FIL, RFIL

```

```

      END
C =====
C This Program generates X and Y for Simulation
      SUBROUTINE YXGEN(M,N,YO,XO,AA,SCALE,IU)
      COMMON ML,NL,JRRL
      INTEGER *2 IU,IV
      DOUBLE PRECISION YO(NL),XO(NL,ML),AA(ML),SCALE,R
      WRITE(*,*) 'FEED THE COEFFICIENTS'
      READ(*,*) (AA(J),J=1,M)
      WRITE(*,*) 'FEED SCALING FACTOR FOR X'
      READ(*,*) SCALE
      DO 1 I=1,N
      DO 2 J=1,M-1
      R=RAND(IU,IV)*SCALE
2  XO(I,J)=R
      XO(I,M)=1.0
1  CONTINUE
      DO 3 I=1,N
      YO(I)=0.0
      DO 3 J=1,M
      YO(I)=YO(I)+XO(I,J)*AA(J)
3  CONTINUE
      RETURN
      END
      SUBROUTINE SXI(XI,N,M,NXI,SXIK,IUXI,POS)
      DOUBLE PRECISION XI(100,10)
      INTEGER *2 IUXI,IV
      DO 1 J=1,M-1
      DO 1 I=1,N
      XI(I,J)=0.0
1  CONTINUE
      DO 2 J=1,M-1
      DO 2 I=1,NXI
22  LOC=RAND(IUXI,IV)*N
      IF((LOC.LE.0).OR.(LOC.GT.N)) GOTO 22
      XI(LOC,J)=RAND(IUXI,IV)*SXIK
      IF(POS.NE.0.0) THEN
      SGNX=RAND(IUXI,IV)-0.5
      SGNX=SGNX/ABS(SGNX)
      XI(LOC,J)=XI(LOC,J)*SGNX
      ENDIF
2  CONTINUE
      DO 3 I=1,N
C      WRITE(*,4) (XI(I,J),J=1,M)
C      3 CONTINUE
C      4 FORMAT(8F10.2)
C      PAUSE
      RETURN
      END
C -----
C This Program generates error term for regression equation
      SUBROUTINE ERGEN(N,YO,Y,ER,SDEV,IU,IV,ID,IG)
      COMMON ML,NL,JRRL
      INTEGER *2 IU,IV
      DOUBLE PRECISION YO(NL),Y(NL),ER(NL),ER1,ER2,ERNORM,S,SS,YM
      DOUBLE PRECISION SDEV
      DO 1 I=1,N

```

```

      ER1=ERNORM(IU, IV)
      ER2=ERNORM(IU, IV)
C -----
C   Normally distributed as average of two normals
      IF(ID.EQ.0) ER(I)=(ER1+ER2)/2.0
C   Cauchy-distributed error as ratio of two std normal variates
      IF(ID.EQ.1) ER(I)=(ER1/ER2)
C   Gamma-distributed error as average of IG squared normals
      ER1=0.0
      ER2=0.0
      DO 2 JG=1, IG
      ER1=ER1+(ERNORM(IU, IV))**2
      ER2=ER2+(ERNORM(IU, IV))**2
2 CONTINUE
      IF(ID.EQ.2) ER(I)=(ER1+ER2)/2.0
C   Beta2-distributed as a ratio of square of two normals
      IF(ID.EQ.3) ER(I)=ER1/ER2
C   Beta1-distributed as  $x/(x+y)$ ,  $x$  and  $y$  being normal
      IF(ID.EQ.4) ER(I)=ER1/(ER1+ER2)
C -----
1 CONTINUE
      S=0
      SS=0
      YM=0.0
      DO 3 I=1, N
      S=S+ER(I)
      SS=SS+ER(I)**2
      YM=YM+YO(I)
3 CONTINUE
      YM=YM/FLOAT(N)
      IF(JRRL.EQ.0) YM=100.0
      S=S/FLOAT(N)
      SS=DSQRT(SS/N-S**2)
      DO 4 I=1, N
      ER(I)=(ER(I)-S)/SS*SDEV*YM
      YO(I)=YO(I)+ER(I)
      Y(I)=YO(I)
4 CONTINUE
      RETURN
      END
C -----
C   This Program generates Outliers for MAD estimation
      SUBROUTINE OUTLI(YO, Y, N, IU, IV, NOL, SIZE, SGNS)
      COMMON ML, NL, JRRL
      INTEGER *2 IU, IV, SGNS(30)
      DOUBLE PRECISION YO(NL), Y(NL), YM, EJ, SIZE
C -----
      YM=0.0
      DO 1 I=1, N
      YM=YM+YO(I)
1 CONTINUE
      YM=YM/FLOAT(N)
C
      WRITE(*,*) 'AVERAGE Y IS = ', YM
      DO 3 NT=1, NOL
      2 J=RAND(IU, IV)*(N+1)
      IF(J.EQ.0) GO TO 2

```

```

      EJ=RAND (IU, IV) *YM*SIZE*SGNS (NT)
      YO (J)=YO (J)+EJ
C      WRITE (*, *) 'J, EJ, Y (J), YO (J) ', J, EJ, Y (J), YO (J)
      Y (J)=YO (J)
3    CONTINUE
C      -----
      RETURN
      END
C      -----
C      Central Program for MAD estimation using Iterated Weighted LS
      SUBROUTINE MAD (YO, XO, Y, X, M, N, TOL, ITMX, AA, XX, XY, A, YH, ER, RSQ, B)
      COMMON ML, NL, JRRL
      DOUBLE PRECISION YO (NL), XO (NL, ML), Y (NL), X (NL, ML), ER (NL)
      DOUBLE PRECISION XX (ML, ML), XY (ML), YH (NL), AA (ML), B (ML), C (10)
      DOUBLE PRECISION A (ML), RSQ, ABSUM, ASUM, TOL, EPSY
10   FORMAT (1X, 6F13.2)
      EPSY=0.0001
      DF=FLOAT (N-M)
      ITER=0
      CALL OLS (YO, XO, Y, X, N, M, AA, XX, XY, A, YH, ER, RSQ)
      WRITE (7, *) (A (J), J=1, M)
      WRITE (*, *) 'ORDINARY LEAST SQUARES ESTIMATES '
      WRITE (*, *) (A (J), J=1, M)
      WRITE (*, *) ' LEAST SQUARES BASED RSQUARE = ', RSQ
      DO 1 J=1, M
        C (J)=A (J)
1    B (J)=A (J)
100  ITER=ITER+1
      IF (ITER.GT.ITMX) THEN
        WRITE (*, *) 'NO CONVERGENCE IN ITERATIONS =', ITMX
        GOTO 7
      ENDIF
      ABSUM=0.0
      DO 2 I=1, N
        ER (I)=DABS (ER (I))
        ABSUM=ABSUM+ER (I)
      W=0.0
      IF (ER (I).GT.EPSY) THEN
        W=1.0/ER (I)
      ENDIF
      Y (I)=YO (I)*W
      DO 3 J=1, M
        X (I, J)=XO (I, J)*W
3    C (J)=A (J)
2    CONTINUE
      IF (ITER.EQ.1) ERSUM=ABSUM
      CALL OLS (YO, XO, Y, X, N, M, AA, XX, XY, A, YH, ER, RSQ)
      ASUM=0
      DO 4 I=1, N
        ER (I)=DABS (ER (I))
4    ASUM=ASUM+ER (I)
      DO 6 J=1, M
        IF (DABS (C (J)-A (J)).GT.TOL) GOTO 100
6    CONTINUE
7    WRITE (*, *) 'OLS COEFFICIENTS '
      WRITE (*, *) (B (J), J=1, M)
      WRITE (*, *) 'MAD COEFFICIENTS '

```

```

WRITE (*, *) (A(J), J=1, M)
ERSUM=ERSUM/FLOAT(N)
ASUM=ASUM/FLOAT(N)
RSQ=ASUM
WRITE (*, *) 'TOTAL NUMBER OF ITERATIONS PERFORMED ', ITER-1
WRITE (*, *) 'MEAN OLS AND MAD DEVIATIONS ', ERSUM, ASUM
RETURN
END
C -----
C Central Program for Least Squares Estimation of Regression Eqn.
SUBROUTINE OLS(YO, XO, Y, X, N, M, AA, XX, XY, A, YH, ER, RSQ)
COMMON ML, NL, JRRL
DOUBLE PRECISION Y(NL), X(NL, ML), A(ML), ER(NL), YO(NL)
DOUBLE PRECISION XX(ML, ML), XY(ML), YH(NL), AA(ML), XO(NL, ML)
DOUBLE PRECISION RSQ, S, SS, SE, ES, VYH
DF=FLOAT(N-M)
DO 1 J=1, M
XY(J)=0.0
DO 2 K=J, M
XX(J, K)=0.0
DO 10 I=1, N
10 XX(J, K)=XX(J, K)+X(I, J)*X(I, K)
XX(K, J)=XX(J, K)
2 CONTINUE
DO 1 I=1, N
XY(J)=XY(J)+X(I, J)*Y(I)
1 CONTINUE
CALL MINV(XX, M)
DO 3 J=1, M
A(J)=0.0
DO 3 K=1, M
3 A(J)=A(J)+XX(J, K)*XY(K)
SE=0.0
ES=0.0
S=0.0
SS=0.0
DO 4 I=1, N
YH(I)=0.0
DO 5 J=1, M
5 YH(I)=YH(I)+XO(I, J)*A(J)
S=S+YH(I)
SS=SS+YH(I)*YH(I)
ER(I)=YO(I)-YH(I)
SE=SE+ER(I)
ES=ES+ER(I)*ER(I)
4 CONTINUE
S=S/FLOAT(N)
SE=SE/FLOAT(N)
VYH=SS/FLOAT(N)-S*S
RSQ=VYH/(VYH+(ES/FLOAT(N)-SE*SE))
RETURN
END
C -----
C Program to generate N(0, s) from Rectangular Random Numbers
C It uses Variate Transformation for this purpose.
FUNCTION ERNORM(IU, IV)
DOUBLE PRECISION U(2), ERNORM, R

```

```

      INTEGER *2 IU, IV
      R=0.0
      DO 1 I=1,2
      U(I)=RAND(IU, IV)
1 CONTINUE
C -----
C IF U1 AND U2 ARE UNIFORMLY DISTRIBUTED RANDOM NUMBERS (0,1),
C THEN X=[ (-2*ln(U1))**.5]*(COS(2*PI*U2) IS N(0,1)
C PI = 4*ARCTAN(1.0)= 2*1.57079632679489
C PI*2 = 6.2831853071797
C -----
      ERNORM=DSQRT(-2.0*DLOG(U(1)))*DCOS(U(2)*6.2831853071797)
      RETURN
      END
C -----
C Generates Rectangular (0,1) Random Numbers
      FUNCTION RAND(IU, IV)
      INTEGER *2 IU, IV
      IV=IU*259
      IF(IV.GE.0) GOTO 2
      IV=IV+32767+1
2 RAND=IV
      IU=IV
      RAND=RAND*0.3051851E-04
      RETURN
      END
C ----- MATRIX INVERSION -----
      SUBROUTINE MINV(AAA, N)
      INTEGER P
      DOUBLE PRECISION A, B, S, G, Y, Z, W, PP, PP1, PP2, PP3, PP4, PP5, PP6
      DOUBLE PRECISION SKID, SKNORM
      DOUBLE PRECISION AAA(10,10), CII(10,10), X(10,10)
      DIMENSION A(10,10), B(10), S(10), Y(10), Z(10), W(10)
      DO 1234 I=1, N
      DO 1234 J=1, N
1234 A(I, J)=AAA(I, J)
      DO 58 I=1, N
      W(I)=B(I)
      DO 58 J=1, N
58 X(I, J)=A(I, J)
      PP1=0
      NN=N*N
      DO 2 I=1, N
      DO 2 J=1, N
2 PP1=PP1+DABS(A(I, J))
      PP2=PP1/NN
      INDEX=0
      DO 33 I=1, N
33 S(I)=0
      DO 506 J=1, N
      DO 506 I=1, N
      IF (I.GT.J) GO TO 503
      PP=0
      I1=I-1
      DO 501 P=1, I1
      IF (I1.LT.1) GO TO 502
501 PP=PP+A(I, P)*A(P, J)

```

```

      A(I,J)=A(I,J)-PP
502 IF (I.NE.J) GO TO 506
      IF (DABS(A(J,J)).LT.(PP2*.1D-08)) GO TO 507
      GO TO 506
503 PP=0
      J1=J-1
      DO 504 P=1,J1
      IF (J1.LT.1) GO TO 505
504 PP=PP+A(I,P)*A(P,J)
505 Y(I)=(A(I,J)-PP)/A(J,J)
      IF (I.EQ.N) GO TO 508
      GO TO 506
508 JJ1=J+1
      DO 1 P=JJ1,N
      IF (DABS(Y(P)).GT.(10.0*PP1)) GO TO 507
1 CONTINUE
      DO 4 L=JJ1,N
4 A(L,J)=Y(L)
      GO TO 506
C 507 I=J
507 S(J)=PP2
      A(J,J)=A(J,J)+PP2
      INDEX=INDEX+1
506 CONTINUE
      N1=N-1
      DO 17 J=1,N1
      J1=J+1
      DO 17 I=J1,N
      PP=0
      I1=I-1
      IF (J1.GT.I1) GO TO 17
      DO 18 P=J1,I1
18 PP=PP+A(I,P)*A(P,J)
17 A(I,J)=-A(I,J)-PP
21 DO 84 L=1,N
      J=N-L+1
      DO 84 M=1,J
      I=J-M+1
      IF ((I+1).GT.J) GO TO 71
      I1=I+1
      PP=0
      DO 70 P=I1,J
70 PP=PP+A(I,P)*A(P,J)
      A(I,J)=-PP/A(I,I)
      GO TO 84
71 A(I,J)=1.0/A(I,J)
84 CONTINUE
      N1=N-1
      DO 24 J=1,N1
      DO 24 I=1,N
      IF (I.GT.J) GO TO 27
      PP=0
      J1=J+1
      DO 26 P=J1,N
26 PP=PP+A(I,P)*A(P,J)
      A(I,J)=A(I,J)+PP
      GO TO 24

```



```

27 PP=0
   DO 28 P=1,N
28 PP=PP+A(I,P)*A(P,J)
   A(I,J)=PP
24 CONTINUE
   IF (INDEX.EQ.0) GO TO 59
   DO 121 L=1,N
   K=N-L+1
   IF (DABS(S(K)).LT. 0.1D-08) GO TO 121
   DO 90 J=1,N
   Z(J)=A(K,J)
90 Y(J)=A(J,K)
   G=1.0-S(K)*A(K,K)
   IF (DABS(G).LT. 0.1D-13) GO TO 7178
   DO 91 I=1,N
   DO 91 J=1,N
91 A(I,J)=A(I,J)+S(K)*Y(I)*Z(J)/G
121 CONTINUE
59 DO 45 I=1,N
   PP=0
   DO 46 P=1,N
46 PP=PP+A(I,P)*B(P)
45 Y(I)=PP
   DO 47 I=1,N
47 B(I)=Y(I)
   GO TO 1601
7178 CONTINUE
1601 PP=0
   PP1=0
   PP2=0
   DO 60 I=1,N
   PP2=PP2+W(I)*W(I)
   DO 60 J=1,N
   PP=PP+A(I,J)*A(I,J)
60 PP1=PP1+X(I,J)*X(I,J)
   PP=DSQRT(PP)
   PP1=DSQRT(PP1)
   PP2=DSQRT(PP2)
   DO 57 I=1,N
   PP4=0
   DO 73 P=1,N
73 PP4=PP4+X(I,P)*B(P)
57 W(I)=W(I)-PP4
   PP3=0
   DO 56 I=1,N
56 PP3=PP3+W(I)*W(I)
   PP3=DSQRT(PP3)
   DO 55 I=1,N
   DO 50 J=1,N
   PP4=0
   DO 54 P=1,N
54 PP4=PP4+X(I,P)*A(P,J)
50 W(J)=PP4
   DO 55 P=1,N
55 X(I,P)=W(P)
   DO 53 I=1,N
53 X(I,I)=1.0-X(I,I)

```

```

      PP4=0
      DO 52 I=1,N
      DO 52 J=1,N
52  PP4=PP4+X(I,J)*X(I,J)
      PP4=DSQRT(PP4)
      PP5=PP3/PP1
      PP6=PP*PP4*PP2/(1.0-PP4)
      DO 1236 I=1,N
      DO 1236 J=1,N
      CII(I,J)=0.0
      DO 1236 K=1,N
      CII(I,J)=CII(I,J)+AAA(I,K)*A(K,J)
1236 CONTINUE
      SKNORM=0.0
      DO 1237 I=1,N
      DO 1237 J=1,N
      SKID=0.0
      IF(I.EQ.J) SKID=1.0
      SKNORM=SKNORM+(CII(I,J)-SKID)**2
1237 CONTINUE
      SKNORM=DSQRT(SKNORM/(N*N))
C      -----
C      IF(SKNORM.GT.0.0000001) THEN
C      WRITE(*,*) 'INV MATRIX IS INACCURATE. NORM=', SKNORM
C      PAUSE 'PROCEED OR ABORT'
C      ENDIF
C      -----
1238 FORMAT(10F8.5)
      DO 1235 I=1,N
      DO 1235 J=1,N
1235 AAA(I,J)=A(I,J)
      RETURN
      END
C      -----
C      SUBROUTINE RWALK(A,M,NV,YO,XO,IU,EPS,VO)
C      RANDOM WALK METHOD TO FIND MIN OF A MULTIVARIABLE FUNCTION
C      INTEGER *2 IU,IV
C      DOUBLE PRECISION A(10),R(10),AR(10),YO(100),XO(100,10)
C      DOUBLE PRECISION LAMBDA,EPS,F,VO,VR,RNORM,YH
C      LAMBDA=2.0
C      ITMAX=200
C      Initialisation of decision variables
C      DO 7 I=1,M
C      7 A(I)=0.0
C      ----- FUNCTION EVALUATION -----
C      F=0.0
C      DO 11 I=1,NV
C      YH=0.0
C      DO 12 J=1,M
12  YH=YH+XO(I,J)*A(J)
      F=F+DABS(YO(I)-YH)
11  CONTINUE
      VO=F
C      -----
C      IT=0
200 IT=IT+1
      IF(IT.GT.100) THEN

```

```

WRITE(*,*) 'NO CONVERGENCE IN 100 ITERATIONS'
GOTO 1000
ENDIF
LAMBDA=LAMBDA/2.0
IMP=0
DO 100 II=1,ITMAX
C   GENERATE M UNIFORMLY DISTRIBUTED RANDOM NUMBERS (-1,1)
150 DO 2 I=1,M
    R(I)=2.0*(RAND(IU,IV)-0.5)
    2 CONTINUE
C   NORMALISE THE RANDOM NUMBERS
    RNORM=0.0
    DO 3 I=1,M
        RNORM=RNORM+R(I)**2
    3 CONTINUE
    RNORM=DSQRT(RNORM)
    IF(RNORM.GT.1.0) GOTO 150
    DO 4 I=1,M
        R(I)=R(I)/RNORM
    4 CONTINUE
C   ADD RANDOM NUMBERS TIMES LAMBDA TO A VECTOR
    DO 5 I=1,M
        AR(I)=A(I)+LAMBDA*R(I)
    5 CONTINUE
C   ----- FUNCTION EVALUATION -----
    F=0.0
    DO 13 I=1,NV
        YH=0.0
        DO 14 J=1,M
14      YH=YH+XO(I,J)*AR(J)
        F=F+DABS(YO(I)-YH)
    13 CONTINUE
    VR=F
C   -----
    IF(VR.LT.VO) THEN
        VO=VR
        DO 6 I=1,M
            A(I)=AR(I)
        6 CONTINUE
        IMP=1
    ENDIF
100 CONTINUE
    IF((IMP.EQ.0).OR.(LAMBDA.GT.EPS)) GOTO 200
1000 CONTINUE
C   WRITE(*,*) 'OPTIMAL VALUE =', VO, ' LAMBDA =', LAMBDA
C   WRITE(*,*) 'OPTIMUM A =', (A(I), I=1,M)
RETURN
END

```

DasGupta, Madhuchhanda (2004) LEAST ABSOLUTE DEVIATION ESTIMATION OF MULT-EQUATION LINEAR ECONOMETRIC MODELS : A STUDY BASED ON MONTE CARLO EXPERIMENTS, Doctoral Dissertation, Dept. of Economics, North Eastern Hill University, Shillong (India); Supervisor – Prof. SK Mishra

BIBLIOGRAPHY

Abdelmalek, NN (1971). "Linear L_1 Approximation for a Discrete Point Set and L_1 Solutions of Overdetermined Linear Equations", *Journal of Assoc. Comput. Mach.*, 18 (41-47).

Abdelmalek, NN (1974). "On the Discrete Linear L_1 Approximation and L_1 Solution of Over- determined Linear Equations", *Journal of Approximation Theory*, 11(35-53).

Amemiya, T (1980). "The Two Stage Least Absolute Deviations Estimators", Technical Report No. 297, Institute for Mathematical Studies in the Social Sciences, Stanford University, December.

Amemiya, T (1982). "Two-Stage Least Absolute Deviations Estimators", *Econometrica*, 50: 1-3 (689-711)

Anderson, TW & H Rubin (1949). "Estimation of the Parameters of a Single Equation in a Complete System of Stochastic Equations", *Annals of Mathematical Statistics*, 20:1(March) (46-63).

Anscombe, FJ (1967). "Topics in the Investigation of Linear Relations Fitted by the Method of Least Squares", *Journal of the Royal Statistical Society, Series B*, 29 (1-52).

Arifovic, J (1994). "Genetic Algorithm Learning and the Cobweb Model", *Journal of Economic Dynamics and Control*, 18 (3-28).

Ashar, VC & TD Wallace (1963). "A Sampling Study of Minimum Absolute Deviation Estimators", *Operations Research*, 11 (747-758).

Barrodale, J. (1968). " L_1 Approximation and the Analysis of Data", *Appl. Statist.* 17 (51-57).

Barrodale, I & FDK Roberts (1973). "An Improved Algorithm for Discrete L_1 Approximation", *SIAM. Journal of Numerical Analysis*, 10 (839-848).

Barrodale, I & A Young (1966). "Algorithms for Best L_1 and L_∞ Linear Approximations on a Discrete Set", *Numer. Math.*, 8 (295-306).

Bartels, R (1977). "On the Use of Limit Theorem Arguments in Economic Statistics", *American Statistician*, 31 (85-87).

Basman, RL (1957). "A Generalized Classical Method of Linear Estimation of Coefficients in a Structural Equation", *Econometrica*, 25 (77-84).

Bassett, G Jr & R. Koenker (1978). "Asymptotic Theory of Least Absolute Error Regression", *Journal of American Statistical Association*, 73 (618-622).

Birnbaum, ZW (1952) "Numerical Tabulation of the distribution of Kolmogorov's Statistic for Finite Sample Size", *Journal of the American Stat. Assoc.* 47 (425-437)

Blattberg, R & T Sargent (1971). "Regression with Non-Gaussian Stable Disturbances: Some Sampling Results", *Econometrica*, 39 (501-510).

Breidt, FJ, RA Davis & A Trindade (2000). "Least Absolute Deviation Estimation for All-Pass Time Series Models", *Annals of Statistics.*, (in press). www.stat.colostate.edu/~jbreidt/technical_reports.htm

Brouwer, LEJ (1912). "Über Abbildung von Mannigfaltigkeiten". *Mathematischen Annalen*, 71. Ref. by Takayama, A (1974) p. 260.

Charnes, A & CE Lemke (1954). "Computational Theory of Linear Programming: The Bounded Variables Problem", Graduate School of Ind. Administration, Carnegie Inst. of Technol., Pittsburg, Pennsylvania.

Charnes, A, WW Cooper & RO Ferguson (1955). "Optimal Estimation of Executive Compensation by Linear Programming" *Management Science*, 1 (138-151).

Chen, XR, ZD Bai, LC Zhao & YH Wu (1992). "Consistency of Minimum L_1 - norm Estimates in Linear Models" in *Development of Statistics: Recent Contributions from China* (249-260), Pitman Research Notes in Mathematical Series 258, Longman Scientific and Technical, Longdon.

Chen, X (1996). "On a Problem of Strong Consistency of Least Absolute Deviation Estimates". *Statistica Sinica*, 6 (481-489).

Christ, CF (1966). *Econometric Models and Methods*, John Wiley and Sons, New York.

Cooper, B (2000). "Modelling Research and Development: How Do Firms Solve Design Problems?", *Journal of Evolutionary Economics*, 10 (395-413).

Cragg, J (1967). "On The Relative Small-Sample Properties of Several Structural Equation Estimators", *Econometrica*, 35 (89-110).

Cramer, JS (1971). *Empirical Econometrics*, North Holland Publishing Company, Amsterdam.

Dantzig, GB (1955). "Upper Bounds, Secondary Constraints, and Block Triangularity in Linear Programming". *Econometrica*, 23 (166-173).

Dantzig, GB (1963). *Linear Programming and Extensions*. Princeton University Press, Princeton, N. J.

Dasgupta, M (1992). *Degree of Over-Identification and Its Effects on the Performance of Estimators*. M. Phil. Dissertation, Department of Economics, NEHU, Shillong (unpublished).

Dhrymes, PJ (1970). *Econometrics : Statistical Foundations and Applications*, Harper and Row, New York.

Dirk, C & T Doherr (2002). “Genetic Algorithms: A Tool for Optimization in Econometrics – Basic Concept and an Example for Empirical Applications”. *ZEW Discussion Paper No. 02-41*. <http://ssrn.com/abstract=320987>.

Dorsey, RE and WJ Mayer (1995), “Genetic Algorithms for Estimations Problems with Multiple Optima, Nondifferentiability, and other Irregular Features”, *Journal of Business & Economic Statistics* 13-1 (53–66).

Eckhardt, R (1987). “Stan Ulam, John von Neumann, and the Monte Carlo method”, *Los Alamos Science*, Vol. 15 (Special Issue), (131-137).

Edgeworth, FY (1887). “On Discordant Observations” , *Phil. Magazine*, 24 (364-375).

Edgeworth, FY (1888). “On a New Method of Reducing Observations Relating to Several Quantities”, *Phil. Magazine*, 5, 25 (185-191).

Edgeworth, FY (1923). “On the Use of Medians for Reducing Observations Relating to Several Quantities” , *Phil. Magazine* 6th Ser. (1074-1088).

Fair, RC (1974). “On the Robust Estimation of Econometric Models”, *Annals of Economic and Social Measurement*, 3 (667-677).

Fair, RC (1994). *Estimating and Testing the US Model*, Yale Univ. USA.

Fama, EF (1965). “The Behaviour of Stock Market Prices ”, *Journal of Business*, 38 (34-105).

Fisher, WD (1961). “A Note on Curve Fitting with Minimum Deviation by Linear Programming”, *Journal of American Statistical Association*, 56 (359-361).

Fraser, DAS (1958). *Statistics: An Introduction*. John Wiley and Sons Inc., New York.

Froberg, CE (1964). *Introduction to Numerical Analysis* Addison-Wesley Publishing Co. Inc., London.

Furno, M (2000). "LAD Estimation with Random Coefficient Autocorrelated Errors". www.eco.unicas.it/docente/furno/Text-rca.pdf

Georgescu-Roegen, N (1971). *The Entropy Law and the Economic Process*, Harvard Univ. Press, Cambridge, Mass, USA.

Gillett, BE (1979). *Introduction to Operations Research*. Tata McGraw Hill, New Delhi.

Girshick, MA & T, Haavelmo (1947). "Statistical Analysis of the Demand for Food: Examples of Simultaneous Estimation of Structural Equations", *Econometrica*, 15: 2 (79-110)

Glahe, FR & JG Hunt (1970). "The Small Sample Properties of Simultaneous Equation Least Absolute Estimators vis-a-vis Least Squares Estimators", *Econometrica*, 5: 38 (742-753).

Goldberg, DE (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison Wesley, Reading, Mass.

Goldberger, AS (1964). *Econometric Theory*. John Wiley and Sons, New York.

Gujarati, DN (1995). *Basic Econometrics*. McGraw Hill, International Editions.

Haavelmo, T (1943). "The Statistical Implications of a System of Simultaneous Equations", *Econometrica*, 11 (1-12).

Haavelmo, T (1944). "The Probability Approach in Econometrics", *Econometrica*, 12, Supplement.

Haavelmo, T (1947). "Methods of Measuring the Marginal Propensity to Consume". *Journal of American Statistical Association*, 42 (105-22).

Hammersley, JM & DC Handcomb (1964). *Monte Carlo Methods*, John Wiley and Sons, New York.

Hill, RW & PW Holland (1977). "Two Robust Alternatives to Least Squares Regression", *Journal of American Statistical Association*, 72 (828-833).

Hitomi, K & M Kagihara (2001). "Calculation Method for Non-Linear Dynamic Least Absolute Deviation Estimator". *J. of Japan Statist. Society*, 31-1 (39-51).

Holland, J (1975). *Adaptation in Natural and Artificial Systems*, Univ. of Michigan Press, Ann Arbor, USA.

Hood, WC & TC, Koopmans (Eds) (1953). *Studies in Econometric Method*, Cowles Foundation Monograph 14, John Wiley, New York.

Hooper JW & H. Theil (1958). “The Extension of Wald’s Method of Fitting Straight Lines to Multiple Regression”, *Review of International Statistical Institute*, 26 (37-47).

Huber, PJ (1964). “Robust Estimation of a Location Parameter”, *Annals of Mathematical Statistics*, 35 (73-101).

Huber, PJ (1972). “Robust Statistics: A Review”, *Annals of Mathematical Statistics*, 43 (1041-1067).

Huber, PJ (1973). “Robust Regression: Asymptotics, Conjectures, and Monte Carlo”, *The Annals of Statistics*, 1 (799-821).

Intriligator, MD (1978). *Econometric Models, Techniques and Applications*, Prentice Hall, Inc. Englewood Cliffs, New Jersey.

Johnston, J (1963). *Econometric Methods*. McGraw – Hill, New York.

Judge, GG , WE Griffiths, RC Hill & T Lee (1980). *The Theory and Practice of Econometrics*. John Wiley and Sons Inc., New York.

Kapur, JN & HC Saxena (1982). *Mathematical Statistics*. S. Chand and Co. Ltd., New Delhi.

Kendall, MG (1958). *Advanced Theory of Statistics*. Charles Griffith, London.

Kendall, M G & A Stuart (1968). *The Advanced Theory of Statistics*. Charles Griffith, London.

Khazzoom, JD (1976). “An Indirect Least Squares Estimator for Over-identified Equations”, *Econometrica*, 44, 4 (741-750).

Kim, T H & C Muller (2000). “Two-Stage Quantile Regression”_, *Discussion Paper in Economics* #00/1, Nottingham University.

Klein, LR (1953). *A Textbook of Econometrics*. Row, Peterson and Co., Evanston, Illinois.

Kmenta, J (1971). *Elements of Econometrics*. Macmillan, New York.

Knuth, DE (1969). *The Art of Computer Programming*, Addison-Wesley. London.

Koopmans, TC (1950) ed.: *Statistical Inference in Dynamic Economic Models*. John Wiley & Sons. New York.

Koopmans, TC, H Rubin and RB Leipnik (1950) : “Measuring the Equation Systems of Dynamic Economics” in TC Koopmans (ed.) , *Statistical Inference in Dynamic Economic Models*, John Wiley, New York. (53 – 237).

Koutsoyiannis, A (1977). *Theory of Econometrics: An Introductory Exposition of Econometric Methods*, 2nd Ed. Macmillan, London.

Krishnamurthy EV & SK Sen (1976). *Computer-based Numerical Algorithms*. Affiliated East-West Press, New Delhi.

Kuester, JL & JH Mize (1973). *Optimization Techniques with FORTRAN IV*, McGraw Hills, New York.

Maddala, GS (1977). *Econometrics*, McGraw Hill, New York.

Malinvaud, E (1970). *Statistical Methods of Econometrics*, 2nd rev. ed., North-Holland Publishing Co., Amsterdam.

Mandlebroth, BB (1963). “The Variation of Certain Speculative Prices”, *Journal of Business*, 36 (394-419).

Mandlebroth, BB (1967). “The Variation of Some Other Speculative Prices”, *Journal of Business*, 40 (393-413).

Marschak, J (1953). “Economic Measurements for Policy and Predictions”, in WC Hood and TC Koopmans, eds., *Studies in Econometric Method*, Cowles Foundation Monograph 14, John Wiley, New York .

Medhi, N (1992). *A Study on the Behaviour of the Minimum Absolute Deviation (MAD) Regression Estimator*, M. Phil. Dissertation. Dept. of Economics, NEHU, Shillong (unpublished).

Metropolis, N & S Ulam (1949). “The Monte Carlo method”, *Journal of the American Statistical Association*, 44 (247), (335-341).

Meyer, JR & RR Glauber (1964). *Investment Decisions: Economic Forecasting and Public Policy*, Harvard Business School Press, Cambridge, Massachusetts.

Mishra, SK (1997). “Generalisation of Indirect Least Squares to Estimation of Over-identified Equations of a Multi-Equation Linear Econometric Model”, *Assam Statistical Review*, 9:1, (57-66).

Mitchell, M (1996), *An Introduction to Genetic Algorithms*, Cambridge Univ. Press, London.

- Murphy, JL** (1973). *Introductory Econometrics*, Richard D. Irwin, Inc. New York.
- Nagar, AL** (1960). "A Monte Carlo Study of Alternative Simultaneous-Equation Estimators", *Econometrica*, 28 (573-590).
- National Bureau of Standard** (1951). "Monte Carlo Method". *Applied Math. Serial 12*, Washington.
- Naylor, TH** (Ed.) (1969). *The Design of Computer Simulation Experiments*. Duke Univ. Press, Durham, N. C.
- Newey, WK** (1985). "Semi-Parametric Estimation of Limited Dependent Variables Models with Endogenous Explanatory Variables". *Annales de l'INSEE*, 59 & 60 (219-235).
- Nyquist, H & A Westlund** (1977). *L_1 Versus L_2 Norm Estimation in Interdependent Systems when Residual Distributions are Stable*, Mimeo, Dept. of Statistics, Univ. of Umea.
- Oveson, RM** (1968). *Regression Parameter Estimation by Minimizing the Sum of Absolute Errors*, Doctoral Dissertation, Harvard University, Cambridge, Mass (unpub).
- Pagan, AR** (1986). "Two-Stage and Related Estimators and their Applications". *Review of Economic Studies*, 57 (517-538).
- Phillips PCB** (1991). "A Shortcut to LAD Estimator Asymptotics". *Econometric Theory*, Vol. 7-4 (450-463).
- Pollard, D** (1991). "Asymptotics for Least Absolute Deviation Regression Estimators". *Econometric Theory*, . 7-2 (186-199).
- Portnoy S & R Koenker** (1997). "The Gaussian Hare and the Laplacian Tortoise: Computability of Squared-error versus Absolute-error Estimators". *Statistical Science*, 12 (279-300).
- Powell, JL** (1984). "Least Absolute Deviations Estimation for the Censored Regression Model". *J. of Econometrics*, 25 (303-325).
- Price, TC** (1997). "Using Co-evolutionary Programming to Simulate Strategic Behaviour in Markets", *Journal of Evolutionary Economics*, 7 (219-254).
- Rabinowitz, P** (1968). "Applications of Linear Programming to Numerical Analysis", *SIAM Review*, 10 (121-159).
- Rao, SS** (1978). *Optimization: Theory and Applications*. Wiley Eastern, New Delhi.

Rao, CR & SK Mitra (1971). Generalised Inverse of Matrices and its Applications, Wiley, New York.

Reiersol, O (1950). "Identifiability of a Linear Relation between Variables which are Subject to Error", *Econometrica*, 18, (375-389).

Rhodes, EC (1930). "Reducing Observations by the Method of Minimum Deviations", *Phil. Magazine* 7th Ser. (974-992).

Rice, JR & JS White (1964). "Norms for Smoothing and Estimation", *SIAM Review*, 6 (243-256).

Robers, PD & A Ben-Israel (1969). "An Interval Programming Algorithm for Discrete Linear L_1 Approximation Problems", *Journal of Approximation Theory*, 2 (323-336).

Robers, PD & SS Robers (NA), "Discrete Linear L_1 Approximation by Interval Linear Programming", *Manuscript*. Ref. by Barrodale, I & FDK Roberts (1973)

Sakata, S (2001). "Instrumental Variable Estimation Based on Mean Absolute Deviation". Papers of Department of Economics, University of Michigan, USA citeseer.ist.psu.edu/sakata01instrumental.html (Submitted to Journal of Econometrics, www.econ.ubc.ca/ssakata/public_html/cv/cv.html), www.econ.lsa.umich.edu/wpweb/1liv.pdf, 67 pages.

Schlossmacher, EJ (1973). "An Iterative Technique for Absolute Deviations Curve Fitting", *Journal of the American Statistical Association*, 68 (857-859).

Simmonard, M (1966). *Linear Programming*, Prentice Hall, Englewood Cliffs, New Jersey.

Singh, LK (1993). Performance of Generalised Indirect Least Squares in Over-Identified Equation Estimation: A Monte Carlo Experiment, MPhil Dissertation, Department of Economics, NEHU, Shillong (unpublished).

Singleton, RR (1940). "A Method for Minimizing the Sum of Absolute Errors", *Am. Math. Statist.* 11 (301-310).

Smith, VK (1973). Monte Carlo Methods, Lexington Books, Lexington..

Smith, VK & TW Hall (1972). "A Comparison of Maximum Likelihood Versus BLUE Estimators", *The Review of Economics and Statistics*, 54 (186-190).

Spyropoulous, K, E Kiountouzis & A Young (1973). "Discrete Approximation in the L_1 Norm", *Computer Journal*, 16 (180-186).

Takayama, A (1974). *Mathematical Economics*, Dryden Press, Hinsdale, Illinois.

Texas Instruments Inc. (1979). *TI Programmable 58C/59 Master Library*, Texas Instruments, Inc. Texas.

Theil, H (1953). Repeated Least Squares Applied to Complete Equation Systems, Central Planning Bureau (Mimeo), The Hague.

Theil, H (1957). "Specification Errors and the Estimation of Economic Relationships", *Review of International Statistical Institute*, 25 (41-51).

Theil, H (1961). *Economic Forecasts and Policy*, 2nd ed., North Holland Publishing Co., Amsterdam.

Theil, H (1971). *Principles of Econometrics*, John Wiley and Sons, Inc. New York.

Thomas, RL (1985). *Introductory Econometrics Theory and Application*, Longman Group Limited, New York.

Taylor, LD (1974). "Estimation by Minimizing the Sum of Absolute Errors" in P. Zerebka (ed.), *Frontiers of Econometrics*, Academic Press, New York.

Usow, KH (1967). "On L_1 Approximation for Discrete Functions and Discretization Effects", *SIAM Journal of Numerical Analysis*, 4 (233-244).

Wagner, HM (1958). "A Monte Carlo Study of Estimates of Simultaneous Linear Structural Equations", *Econometrica*, 26 (117-133).

Wagner, HM (1959). "Linear Programming Techniques for Regression Analysis", *Journal of American Statistical Association*, 56 (206-212).

Weiss, AA (1991). "Estimating Non-linear Dynamic Models using Least Absolute Error Estimation". *Econometric Theory*, 7 (46-68).

Wright, AH (1991). "Genetic Algorithms for Real Parameter Optimization", in GJE Rawlings (ed) *Foundations of Genetic Algorithms*, Morgan Kauffmann Publishers, San Mateo, CA (205-218).

Zellner, A & Theil, H (1962). "Three Stage Least Squares: Simultaneous Estimation of Simultaneous Equations", *Econometrica*, 30 (54-78).

DasGupta, Madhuchhanda (2004) LEAST ABSOLUTE DEVIATION ESTIMATION OF MULT-EQUATION LINEAR ECONOMETRIC MODELS : A STUDY BASED ON MONTE CARLO EXPERIMENTS, Doctoral Dissertation, Dept. of Economics, North Eastern Hill University, Shillong (India); Supervisor – Prof. SK Mishra

BIO-DATA

NAME :- MADHUCHHANDA DASGUPTA.
FATHER'S NAME :- Shri S.C.DASGUPTA.
ADDRESS :- C/o. Shri. S.C.DASGUPTA
Nongrim Hills, Banasree,
Shillong – 793003

EDUCATIONAL OUALIFICATION: -

Sl. No	Exam Passed	Board/University	Year	Division/Grade	Percentage of marks
1.	M.Phil	N.E.H.U.	1994	A	
2.	M.A.(Eco)	N.E.H.U.	1990	I	69.88%
3.	B.A.(Eco.Hons.)	N.E.H.U.	1988	II	52.88%
4.	B.A.(Pass)	N.E.H.U.	1987	II	56.92%
5.	P.U.(Arts)	N.E.H.U.	1985	I	68.66%
6.	I.C.S.E.	Council for the Indian School Certificate Examination, New Delhi	1983	I	65%

EXPERIENCE

:- I have been working as a Lecturer in the Dept. of Economics of Women's College, Shillong since August 1993.

Place: -

Date: -

(MADHUCHHANDA DASGUPTA)