Roma Tre University

Department of Economics

Ph.D in Statistical Method for Economics
XXII Cycle

The class of Least Orthogonal Distance Estimator of structural parameters for SEM

TUTORS
Professor: Luciano Pieraccini
Researcher: Alessia Naccarato

CANDIDATE
Davide Zurlo

Academic Year 2009/10

Ai miei Nonni e alla mia famiglia,
perchè le cose più belle non sono solo quelle per cui lotti,
ma anche quelle che ti vengono donate per amore.

Contents

List of Figures

List of Tables

Abstract

The aim of this paper is to present the results obtained on the class of Least Orthogonal Distance Estimator - full and limited infomation - of structural parameters for SEM. The LODE method of estimation has been derived under the consideration that the over-identifying restriction are nothing else but linear relations between variables affected by error (Naccarato and Pieraccini, 2008; Naccarato, 2007).

The original form of LODE was based on characteristic roots and vectors, the simulation experiments conducted show that full information and limited information LODE works better than the other classics limited and full information estimators but in terms of variability the results wasn't so good. In this work are presented some solution adopted to reduce the estimation variability, the first solution was based on a computational procedure links to the minimization of the trace of structural errors' matrix Variance-Covariance and this procedure together with the result of the simulation experiment are in the chapter 2.

Then - always to improve the LODE performance in terms of mean square error - it was developed a new version of LODE based on Singular Value Decomposition (chapter 4 and 5) instead of Spectral Decomposition, this because an algorithm based on SVD is numerically more robust respect to algorithm based on SD, where robustness means the greatest algorithm's probability to converge (Markovsky and Van Huffel, 2007). The results of the new simulation adopting the LODE based on SVD better perfomances than the classical estimators both in terms of bias and variability.

Chapter 1

Introduction

In this thesis is introduced some innovation made to the class of LODE estimator of structural parameters in a SEM.

The principal characteristic of the Simultaneous Equation Model is that in this kind of models some of the explanatory variable (endogenous variables) present in the so called structural form of this model are correlated with the error component, so the OLS estimate applied on this multivariate regression model are unbiased and inconsistent.

There is two form to synthesize a Simultaneous Equation Model one is the aforementioned structural form that specifies the simultaneous quantitative relation that exist between the endogenous each other and between the endogenous variables and the pre-determinate variables presented in the system, the second one is the reduced form that define the vector of endogenous variables only respect to the pre-determinate variables (exogenous variables) so this reduced form can be see like a closed form of structural form respect to endogenous variables.

While the reduced form is an identified system we can not say the same things for structural form, so what we need is to impose some conditions on structural form parameters. This conditions have the form of exclusion constraints, the definition of this condition comes from the so called identification's system that links the parameters of structural form with the one of reduced form.

The Least Orthogonal Distance Estimator method of estimation has been derived under the consideration that this system are nothing else but linear relations between variables affected by error, where this variables is the parameters of reduced form and the unknowns is the structural form parameters (Naccarato and Pieraccini, 2008; Naccarato, 2007). The starting idea, whose development has given rise to Least Orthogonal Distance Estimator(LODE), can be tracked back to the work of (Pieraccini, 1969), in which 2SLS were obtained as generalized least square estimator applied to the system of identification; the result was afterwards extended to 3SLS (Pieraccini,1978). With this in mind and making reference to the work of K. Pearson "Lines and planes of closest fit"(Pearson 1901) was developed this estimator which find the estimates of structural parameters minimizing the orthogonal distance, under the consideration that the system of identification is nothing else but linear relations between variables affected by error, to do this it works with characteristic roots and vector of a matrix deriving from identification's system of SEM.

In the first formalization, LODE structural parameters' estimation of endogenous and exogenous variables was not treated symmetrically: parameters of endogenous variable was derived working only with the second equation of identification's system and so applying spectral decomposition on the matrix deriving from this equation, that doesn't involve the parameters fo exogenous variables, this one were obtained in the same way as LIML (Pieraccini 1983, 1988, 1992).

Then taking into account the results of many simulation experiment was introduced a modified version of limited Information LODE in which structural parameters are estimate both for endogenous and exogenous variables (Sbrana 2001).

Furthermore a recent contribution have increased the interest about this method: its extension to the case of Full Information (Naccarato, 2007; Naccarato and Pieraccini, 2008) which has shown the versatility of the method to cope with simultaneous estimation of the whole system's structural parameters.

1.1 Purpose and Objectives

The simulation experiment on LODE LI (Perna, 1989; Cau, 1990; Sbrana, 2001; Zurlo, 2006) showed that usually in terms of bias the LODE LI works better than the other classics limited information estimators but in terms of variability the results wasn't so good.

The first simulations applied on FI LODE which were been the start of the thesis' project showed that the passage from LI LODE to FI LODE seemed to have increased this problem as matter of the fact every structural parameters estimation was affected by the presence of few far outliers which weighed heavily on the general estimation results, for this reason i looking for finding some solution to improve the stability of FI LODE, the results of this studies are became the core of my Ph.D. Thesis.

Hence the principal aim of this work is to present some solution adopted to reduce the FI LODE's estimation variability, the first solution was based on a computational procedure links to the minimization of the trace of structural errors' matrix Variance-Covariance but it was just an empirical solution.

The subsequent developments point out that the modified version of LODE, which works with the whole system and estimate together the structural parameters of endogenous and exogenous variables, could bring some problem, because the matrix that derive from the system of identification isn't compose only from "variables affected by error" so the orthogonal minimization could yield biased estimate. For this reason the idea is to come back to the primal version of LODE and estimating before the parameters of endogenous variables with LODE and then the ones of exogenous variables.

Coming back to the primal version the last solution involve led to use Singular Value Decomposition instead of Spectral Decomposition, this because an algorithm based on SVD is numerically more robust respect to algorithm based on SD, where robustness means the greatest algorithm's probability to converge (Markovsky and Van Huffel, 2007).

1.2 Outline of the chapters

This thesis is organized into seven chapters. In chapter 2 there is a briefly introduction on Simultaneous equation models to establish notation then the two estimator based on orthogonal distance Limited and Full Information Least Orthogonal Distance are presented .

The chapter 3 is dedicated to the computational procedure to minimize the Variance-Covariance matrix of errors component and to the result of simulation experiment conduct by comparing the results of LODE with the other two classical estimator for SEM Three Stage Least Square (3SLS) and Full Information Maximum Likelihood (FIML).

The problem caused by the simultaneous estimate of all endogenous and exogenous parameters is explained in chapter 4.

Chapter 5 is devoted to the LODE based on Singular Value Decomposition and in the subsequent chapter the result of new simulation experiment on FI LODE with empirical correction FI LODE SVD before and FI LODE SVD 3SLS FIML later are illustrated.

Finally (Chapter 8) conclusion and and future developments, while in the appendix A is reported the program script of the simulation experiment conduct with the software E-Views 7

Chapter 2

LODE for SEM

2.1 Introduction

In this chapter we will introduced the LODE full and limited information estimator of structural parameters in simultaneous equation model. So in the first section (1.2) is presented the context where this estimator works hence there will be a short introduction to simultaneous equation model notation and to conditions of identification show in a new context (Naccarato, 2007), in the other two section (1.3) and (1.4) are presented respectively the last version of limited information and the full information LODE estimator.

2.2 The simultaneous equations model

The simultaneous equation model are multivariate regession model and are characterized by the fact that some of the explanatory variable , called endogenous, are correlated with the error component, so the OLS estimate for the coefficients of this variable are unbiased and inconsistent, for this reason are been development other kind of estimator: limited information estimator where the coefficients of system's equations are estimate equation to equation singularly (Two stage least square and limited information maximum likehood) and full information estimator where are estimate the coefficients of the whole system at the same time (Three stage least square and full

information maximum likehood).

Making use of standard notations, the *structural form* of a simultaneous equations model can be defined as follows:

$$\underset{n,m}{Y}\underset{m,m}{\Gamma} + \underset{n,k}{X}\underset{k,m}{B} + \underset{n,m}{U} = \underset{n,m}{0} \tag{2.1}$$

where Y is the $n \times m$ matrix of endogenous variables and Γ is the corresponding $m \times m$ matrix of structural parameters, X is the $n \times k$ matrix of exogenous variables and B is the $k \times m$ matrix of their structural parameters. Finally U is the $n \times m$ matrix of disturbances for which standard hypotheses are supposed to hold:

$$\begin{aligned} E(vecU) &= 0 \\ E(vecU(vecU)^T &= \Omega \otimes I \end{aligned} \tag{2.2}$$

where

$$\Omega = \begin{bmatrix} \sigma_1^2 & \cdots & \sigma_{1m} \\ \vdots & \ddots & \vdots \\ \sigma_{m1} & \cdots & \sigma_m^2 \end{bmatrix}$$

is the variance-covariance matrix of the disturbances U, constant for all the observations. Furthermore it is generally assumed that:

$$\begin{aligned} \underset{n\to\infty}{plim}\tfrac{1}{n}U^TU &= \Omega \\ \underset{n\to\infty}{plim}\tfrac{1}{n}X^TU &= \underset{k,m}{0} \\ \underset{n\to\infty}{plim}\tfrac{1}{n}X^TX &= \underset{k,m}{\Sigma_x} \end{aligned} \tag{2.3}$$

Under non singularity condition for Γ, if not one or more structural equation would be only a linear combination of the others, we can rewrite the system highlighting the endogenous variable Y, this is the *reduced form* of the equations and it is derived as:

$$\underset{n,m}{Y} = \underset{n,k}{X}\underset{k,m}{\Pi} + \underset{n,m}{V} \tag{2.4}$$

where:

$$\begin{aligned} \underset{k,m}{\Pi} &= -\underset{k,m}{B}\,\underset{m,m}{\Gamma^{-1}} \\ \underset{k,m}{V} &= -\underset{k,m}{U}\,\underset{m,m}{\Gamma^{-1}} \end{aligned} \tag{2.5}$$

considering 2.4 and 2.5 it is clear the link between endogenous variable Y and accidental component of structural equation U

The last equation in 2.5 represents the matrix of reduced form disturbances, for which it is possible to write:

$$\begin{aligned} E(V) &= 0 \\ E\left(V^T V\right) &= n\left(\Gamma^{-1}\right)^T \Omega \Gamma^{-1} \end{aligned} \tag{2.6}$$

Post-multiplying by Γ the first equation in (5) we obtain:

$$\underset{k,m}{\Pi}\,\underset{m,m}{\Gamma} = -\underset{k,m}{B} \tag{2.7}$$

which represents the relation between reduced and structural form parameters.

Even if we consider known the parameter Π of reduced form the system does not admit a unique solution because 2.7 is a system of k equations with $m \times (m+k)$ unknowns. In order to find the solution with respect to Γ and B in terms of Π, we need to impose others conditions on structural form parameters. This conditions have the form of exclusion constraints, in the sense that each equation does not include all the endogenous and exogenous variables and not all the elements of Γ and B are different from zero.

Then it is possible to consider the following partition of the overall matrix of endogenous variables with respect to $i-th$ structural form equation:

$$\underset{n,m}{Y_i} = \left[\underset{n,m_1}{Y_{i1}} \;\vdots\; \underset{n,m_2}{Y_{i2}} \right]$$

where the first columns m_{1i} refer to the endogenous variables included in $i-th$ equation and the last m_{2i} columns refer to those excluded. In the same way the vectors of Γ's in $i-th$ equation can be reordered as:

$$\underset{m,1}{\Gamma_i} = \begin{bmatrix} \underset{m_{1i,1}}{\Gamma_{1i}} \\ \vdots \\ \underset{m_{2i},1}{0} \end{bmatrix}$$

where the first m_{1i} elements of Γ_i refer to Y_{i1} endogenous variables included in the $i-th$ equation. Notice that defining the vector Γ_i no normalization rule has yet been introduced.

Similarly, let us consider the partition:

$$\underset{n,k}{X_i} = \begin{bmatrix} \underset{n,k_{1i}}{X_{1i}} & \vdots & \underset{n,k_{2i}}{X_{2i}} \end{bmatrix}$$

where X_{1i} and X_{2i} are the sub-matrices corresponding to the exogenous variables included in and excluded from the $i-th$ equation respectively.

Accordingly let us define

$$\underset{k,1}{B_i} = \begin{bmatrix} \underset{k_{1i,1}}{B_{1i}} \\ \vdots \\ \underset{k_{2i},1}{0} \end{bmatrix}$$

where the first k_{1i} parameters are related to the exogenous variables included in the $i-th$ equation.

Therefore the $i-th$ structural equation can be expressed as:

$$Y_{1i}\Gamma_{1i} + X_{1i}B_{1i} = U_i.$$

notice that different orderings of variables correspond to each equation of the system.

The $i-th$ reduced form will be

$$\begin{cases} Y_{1i} = X_{1i}\Pi_{11}^i + X_{2i}\Pi_{12}^i + V_{1i} \\ Y_{2i} = X_{1i}\Pi_{21}^i + X_{2i}\Pi_{22}^i + V_{2i} \end{cases}$$

where Π_{11}^i refers to the $i-th$ equation RF parameters of endogenous

and exogenous variables included, Π^i_{12} refers to the endogenous included and exogenous excluded ones, while Π^i_{21} refers to parameters of endogenous excluded and exogenous variables included and Π^i_{22} refers to the $i-th$ equation RF parameters of endogenous and exogenous variables excluded.

Respect to the $i-th$ structural equation of the system, relation (2.4) can be written as and it takes the of *identification system*:

$$\begin{cases} \underset{k_{1i},m_{1i}}{\Pi^i_{11}} \underset{m_{1i},1}{\Gamma_{1i}} = \underset{k_{1i},1}{B_{1i}} \\ \underset{k_{2i},m_{1i}}{\Pi^i_{12}} \underset{m_{1i},1}{\Gamma_{1i}} = \underset{k_{2i},1}{0} \end{cases} \tag{2.8}$$

or equivalently

$$\begin{bmatrix} \underset{k_{1i},m_{1i}}{\Pi^i_{11}} & I_{k_{1i}} \\ \underset{k_{2i},m_{1i}}{\Pi^i_{12}} & \underset{k_{2i},k_{1i}}{0} \end{bmatrix} \begin{bmatrix} \underset{m_{1i},1}{\Gamma_{1i}} \\ \underset{k_{1i},1}{B_{1i}} \end{bmatrix} = 0 \tag{2.9}$$

Let Π^i_* denote the matrix of the parameters of reduced form in which the elements are ordered with respect to the endogenous and exogenous variables included and excluded from $i-th$ equation:

$$\Pi^i_* = \begin{bmatrix} \underset{k_{1i},m_{1i}}{\Pi^i_{11}} & I_{k_{1i}} \\ \underset{k_{2i},m_{1i}}{\Pi^i_{12}} & \underset{k_{2i},k_{1i}}{0} \end{bmatrix},$$

now the condition for solving the system (2.8) depends on the rank of $\Pi^i_*$and this conditions determinate the number of costraints applied on endogenous and esogenous variables in the system.

Usually rank conditions for the identification of a simultaneous equation system, as well as order conditions, are obtained after applying the normalization rules: in this case this doesn't happen.

We will present the identifiably condition without normalization rules (Naccarato, 2007) because one peculiarity of LODE estimator is that it does not impose a prior choice of dependent variable, because it works directly with identification system, so the normalization rules can be apply after the parameter estimation.

Condition 1– system (2.8) admits a unique solution – up to a proportionality constant – if the rank:

$$r\left(\Pi_*^i\right) = m_{1i} + k_{1i} - 1 \tag{2.10}$$

the proof follows directly from the Rouchè-Capelli theorem.

Condition 2– $r\left(\Pi_*^i\right) = m_{1i} + k_{1i} - 1$ if and only if

$$r\left(\Pi_{12}^i\right) = m_{1i} - 1 \tag{2.11}$$

proof Naccarato(2007).

Substituting the reduced form parameters their with OLS estimates, we know that the OLS estimate of Π that are

$$\hat{\Pi} = \left(X^T X\right)^{-1} X^T Y = \Pi + \left(X^T X\right)^{-1} X^T U$$

and under the hypothesis (2.3) is BLUE estimator and

$$\underset{n\to\infty}{plim}\hat{\Pi} = \Pi.$$

But after the substitution, the system (2.8) becomes:

$$\begin{cases} \underset{k1,m1}{\hat{\Pi}_{11}^i} \underset{m1,m}{\Gamma_{1i}} + \underset{k1,m}{B_{1i}} = \varepsilon_{1i} \\ \underset{k2,m1}{\hat{\Pi}_{12}^i} \underset{m1,m}{\Gamma_{1i}} = \varepsilon_{2i} \end{cases} \tag{2.12}$$

so that in both equations an error component occurs because of the use of the estimates $\hat{\Pi}$ instead of the true values Π: then rank conditions cannot be verified. The rank of Π_{12}^i cannot therefore be used as an identification criterion and we need to define the so-called "order conditions" which are related to the number of the equations and unknowns in the system (2.12) and are a direct consequence of rank conditions.

Condition 3– If rank condition (2.10) is satisfied, the matrix $\hat{\Pi}_*^i$ is of order greater or equal to $m_{1i} + k_{1i}$. Then, it has to be:

$$k \geq k_{1i} + m_{1i}$$

i.e:

$$k_{2i} \geq m_{1i} - 1$$

and the number of excluded exogenous variables has to be greater than or equal to the number of included endogenous ones. This is the formulation generally used for order conditions.

Exact identification will occur when:

$$k_{2i} = m_{1i} - 1,$$

while under identification will occur if:

$$k_{2i} \leq m_{1i} - 1.$$

In the first case there is a unique solution while in the second one there is no solution.

2.3 Limited information LODE

The Least Orthogonal Distance Estimator (LODE) is based on characteristic roots and vector of a matrix deriving from 2.12. The starting idea of this method can be tracked back to the work of Pieraccini(1969), in which 2SLS were obtained as generalized least square estimator applied to a system 2.12. With this in mind and making reference to the work of K. Pearson "Lines and planes of closet fit" (1901), the LODE method of estimation has been derived under the consideration that the over-identifying system are nothing else but linear relations between variables affected by error (Pieraccini, 1988), the LI LODE presented here is a more recent development (Sbrana, 2001; Naccarato and Pieraccini, 2008) of its original version .

Defining :

$$\underset{k,m_{1i}+k_{1i}}{\hat{\Pi}^i_*} = \begin{bmatrix} \underset{k_{1i},m_{1i}}{\Pi^i_{11}} & I_{k_{1i}} \\ \underset{k_{2i},m_{1i}}{\Pi^i_{12}} & \underset{k_{2i},k_{1i}}{0} \end{bmatrix}, \quad \underset{m_{1i}+k_{1i},1}{\delta_i} = \begin{bmatrix} \underset{m_{1i},1}{\Gamma_{1i}} \\ \underset{k_{1i},1}{B_{1i}} \end{bmatrix}, \quad \underset{k,1}{\varepsilon_i} = \begin{bmatrix} \underset{k_{1i},1}{\varepsilon_{1i}} \\ \underset{k_{21},1}{\varepsilon_{2i}} \end{bmatrix} \tag{2.13}$$

we have

$$\hat{\Pi}^i_* \delta_i = \varepsilon_i \tag{2.14}$$

where

$$\underset{k,1}{\varepsilon_i} = \left(X_i^T X_i\right)^{-1} X_i^T V_{1i} \Gamma_{1i} = \left(X_i^T X_i\right)^{-1} X_i^T U_i$$

the variance-covariance matrix of ε_i comes out to be

$$E\left(\varepsilon_i \varepsilon_i^T\right) = \left(X_i^T X_i\right)^{-1} X_i^T E\left(U_i U_i^T\right) X_i \left(X_i^T X_i\right)^{-1} = \sigma_i^2 \left(X^T X_i\right)^{-1} \tag{2.15}$$

remembering that σ_i^2 is the variance of accidental component if $i - th$ equation.

From 2.15 is clear that the error of 2.14 are correlated each other, but it is possible to apply a transformation to eliminate this correlation.

Let us now set

$$\left(X_i^T X_i\right)^{-1} = T \Lambda T^T$$

the matrix Λ being the diagonal matrix of characterstic roots of $X_i^T X_i$ and the matrix T_i the one of characteristic vectors. Defining

$$Q = T \Lambda^{\frac{1}{2}} T^T$$

so that it is

$$QQ = T \Lambda T^T = X_i^T X_i$$

and applying to the error component can the following transformation

$$\omega_i = Q\varepsilon_i$$

it will be

$$E\left(\omega_i^T \omega_i\right) = E\left(\varepsilon_i^T X^T X \varepsilon_i\right) = k\sigma_i^2$$

so that the sample estimate of σ_i^2 will be, using 2.14,

$$\hat{\sigma}_i^2 = \frac{1}{k}\omega_i^T \omega_i = \frac{1}{k}\delta_i^T \Pi_*^{iT} X^T X \Pi_*^i \delta \tag{2.16}$$

The LODE method is based on the minimization (2.16) i.e. on finding the vector which minimizes the sample residual variance for the $i-th$ structural equation. Since it can be easily shown that

$$\begin{aligned} \underset{m_{1i}+k_{1i},k}{\hat{\Pi}_*^{iT}} \left(\underset{k,k}{X^T X}\right) \underset{m_{1i}+k_{1i},k}{\hat{\Pi}_*^i} &= \begin{bmatrix} \underset{m_{1i},m_{1i}}{\hat{\Pi}_i^T X^T X \hat{\Pi}_i} & \underset{m_{1i},k_{1i}}{\hat{\Pi}_i^T X^T X} \\ \underset{k_{1i},m_{1i}}{X^T X \hat{\Pi}_i} & \underset{k_{1i},k_{1i}}{X^T X} \end{bmatrix} = \\ \begin{bmatrix} \underset{m_{1i},m_{1i}}{Y_{1i}^T X (X^T X)^{-1} X^T Y_{1i}} & \underset{m_{1i},k_{1i}}{Y_{1i}^T X_{1i}} \\ \underset{k_{1i},m_{1i}}{X^T Y_{1i}} & \underset{k_{1i},k_{1i}}{X^T X} \end{bmatrix} &= \underset{m_{1i}+k_{1i},m_{1i}+k_{1i}}{A_{ii}} \end{aligned} \tag{2.17}$$

then, disregarding the constant $\frac{1}{k}$, the quadratic form to be minimized becomes

$$\delta_i^T A_{ii} \delta_i \tag{2.18}$$

where the reasons for using the symbol A_{ii} will become clear when treating the full information version of LODE method.

LODE estimator has then to be proportional to the vector, say P, such that

$$P^T A_{ii} P = min \tag{2.19}$$

where, to make the solution univocally determined, the condition

$$P^T P = 1 \tag{2.20}$$

(2.20) has to be added. As it is well known, to find the minimum of (2.19) under condition (2.20) one has to minimize the function

$$G = P^T A_{ii} P - \lambda \left(P^T P - 1 \right)$$

with respect to p_i $(i = 1, ..., m_{1i} + k_{1i})$ and to the Lagrange multiplier λ.

The system obtained equating to zero the partial derivatives with respect to P and λ will then be

$$\begin{cases} \frac{\partial G}{\partial P} = 2A_{ii}P - 2\lambda P = 0 \\ \frac{\partial G}{\partial \lambda} = P^T P - 1 = 0 \end{cases} \tag{2.21}$$

whose solutions will be obtained solving the system

$$(A_{ii} - \lambda I)P = 0 \tag{2.22}$$

under condition given by the second of (2.21). Let us remember that to obtain a solution for (2.22), has to be the solution of the determinantal equation

$$|A_{ii} - \lambda I| = 0 \tag{2.23}$$

which, being a polynomial of degree $s \leq m_{1i} + k_{1i} - 1$ in λ, give raise to s roots such that

$$\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_s \geq 0,$$

The vector associated to the smallest root of equation (2.23) is then the solution of the problem. As a consequence the equation

$$X \hat{\Pi}^i_* P_s = 0 \tag{2.24}$$

is the expression of the $(s-1)$ dimensional subspace spanned by the first $(s-1)$ principal axis, i. e. the one which minimizes the sum of squares of the

orthogonal distances between the observed points and the subspace itself. In other words (2.24) will be the last principal component.

Introducing at this point the normalization rule for $i-th$ structural equation, least orthogonal distance estimator of δ are defined as

$$\hat{\delta}_i = \begin{bmatrix} \hat{\Gamma}_{1i} \\ \hat{B}_{1i} \end{bmatrix} = -\frac{1}{p_{0i}} P_s$$

where p_{0i} is the element of the characteristic vector associated with the right hand side endogenous variable in the $i-th$ structural equation.

The estimate of $i-th$ structural equation variance of disturbances will be as a consequence

$$\hat{\sigma}_i^2 = \frac{1}{kp_{0i}^2} \lambda_s$$

Notice that when the $i-th$ equation is exactly identified equation (2.23) will have $(m_{1i}+k_{1i}-1)$ roots the last one being $\lambda_{m_{1i}+k_{1i}-1} = 0$ so that equation (2.18) will have a unique solution that coincides with ILS estimator.

On the contrary when $i-th$ equation is under identified $(k_{2i} < m_{1i}-1)$ the characteristic root equal to zero will have multiplicity equal to $r = m_{1i}-1-k_{2i}$ and the system (2.22) will have infinite to the r solutions.

2.4 Full information LODE

Knowing that the result on Pieraccini(1969) was afterwards extended to 3SLS (Pieraccini, 1978) and since it is well known that Full Information estimators are asymptotically more efficient than Limited Information ones, (Goldberger, 1964, pp. 346-356, Judge et al., 1985) it is worthwhile to generalize LODE method to a full information context (Naccarato, 2007; Naccarato and Pieraccini, 2008).

Defining

$$\underset{mk,s}{\hat{\Pi}_*} = \begin{bmatrix} \underset{k,m_{11}+k_{11}}{\hat{\Pi}^1_*} & \underset{k,m_{12}+k_{12}}{0} & \cdots & \underset{k,m_{1m}+k_{1m}}{0} \\ \underset{k,m_{11}+k_{11}}{0} & \underset{k,m_{11}+k_{11}}{\hat{\Pi}^2_*} & \cdots & \underset{k,m_{1m}+k_{1m}}{0} \\ \vdots & \vdots & \ddots & \vdots \\ \underset{k,m_{11}+k_{11}}{0} & \underset{k,m_{12}+k_{12}}{0} & \cdots & \underset{k,m_{11}+k_{11}}{\hat{\Pi}^m_*} \end{bmatrix}, \tag{2.25}$$

$$\underset{s,1}{\delta} = \begin{bmatrix} \underset{m_{11}+k_{11},1}{\delta_1} \\ \underset{m_{12}+k_{12},1}{\delta_1} \\ \vdots \\ \underset{m_{1m}+k_{1m},1}{\delta_1} \end{bmatrix}$$

and

$$\underset{mk,1}{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_m \end{bmatrix} \tag{2.26}$$

Relation (2.14) between reduced and structural form parameters for the whole system of equation can be written as

$$\underset{mk,s}{\hat{\Pi}_*} \underset{s,1}{\delta} = \underset{mk,1}{\varepsilon} \tag{2.27}$$

where it is .

$$s = \sum_{i=1}^{m} (m_{1i} + k_{1i})$$

and

$$\underset{km,1}{\varepsilon} = \left[I_m \otimes \underset{k,n}{(X^T X)^{-1} X^T} \right] \underset{nm,1}{vec(V\Gamma)} = \left[I_m \otimes \underset{k,n}{(X^T X)^{-1} X^T} \right] \underset{nm,1}{vec(U)}$$

Because of (2.15) applied to the vector defined in (2.26) the variance-covariance matrix of the error component can be written in the following way

$$E(\varepsilon\varepsilon^T) = \underset{mk,mk}{\Sigma} = \underset{m,m}{\Omega} \otimes \left(\underset{k,k}{X^T X} \right)^{-1} \tag{2.28}$$

And taking into account that it is

$$E\left(\varepsilon^T \varepsilon\right) = tr\{\Omega\} tr\left\{\left(X^T X\right)^{-1}\right\} = \sum_{i=1}^{m} \sigma_i^2 \sum_{j=1}^{k} d_i^{jj} \tag{2.29}$$

where d^{jj} are the diagonal elements of $\left(X^T X\right)^{-1}$, to obtain full information LODE, considering the correlation of the error (2.28), it is necessary to minimize the quadratic form

$$\varepsilon^T \left(\underset{m,m}{\Omega} \otimes \left(\underset{k,k}{X^T X} \right)^{-1} \right)^{-1} \varepsilon$$

that means

$$\underset{1,s}{\delta^T} \underset{s,mk}{\hat{\Pi}_*^T} \left(\underset{m,m}{\Omega} \otimes \left(\underset{k,k}{X^T X} \right)^{-1} \right)^{-1} \underset{s,mk}{\hat{\Pi}_*} \underset{1,s}{\delta} = \delta^T \hat{\Pi}_*^T \Omega^{-1} \otimes \left(X^T X\right) \hat{\Pi}_* \delta \tag{2.30}$$

i.e. to consider the characteristic vector associated with the smallest characteristic root of the matrix

$$\underset{s,s}{A} = \hat{\Pi}_*^T \left(\Omega^{-1} \otimes \left(X^T X\right)\right) \hat{\Pi}_* \tag{2.31}$$

where its explicit form is the following

$$A = \begin{bmatrix} \hat{\Pi}_*^{1T}\sigma^{11}\left(X^TX\right)\hat{\Pi}_*^1 & \cdots & \hat{\Pi}_*^{1T}\sigma^{1i}\left(X^TX\right)\hat{\Pi}_*^i & \cdots & \hat{\Pi}_*^{1T}\sigma^{1m}\left(X^TX\right)\hat{\Pi}_*^m \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \hat{\Pi}_*^{iT}\sigma^{i1}\left(X^TX\right)\hat{\Pi}_*^1 & \cdots & \hat{\Pi}_*^{iT}\sigma^{ii}\left(X^TX\right)\hat{\Pi}_*^i & \cdots & \hat{\Pi}_*^{mT}\sigma^{im}\left(X^TX\right)\hat{\Pi}_*^m \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \hat{\Pi}_*^{mT}\sigma^{m1}\left(X^TX\right)\hat{\Pi}_*^1 & \cdots & \hat{\Pi}_*^{mT}\sigma^{mi}\left(X^TX\right)\hat{\Pi}_*^i & \cdots & \hat{\Pi}_*^{mT}\sigma^{mm}\left(X^TX\right)\hat{\Pi}_*^m \end{bmatrix}$$

with σ^{ij} being the element of the matrix Ω^{-1}.

While the block diagonal elements of A are of the form (2.17) – now it is clear the reason for using the proposed notation – the extra diagonal block elements are

$$A_{ij} = \sigma^{ij} \begin{bmatrix} \underset{m_{1i},k}{\hat{\Pi}_i^T} \underset{k,k}{X^TX} \underset{k,m_{1j}}{\hat{\Pi}_j^T} & \underset{m_{1i},k}{\hat{\Pi}_i^T} \underset{k,n}{X^T} \underset{n,k_{1j}}{X_{1j}} \\ \underset{m_{1i},k}{X_{1i}} \underset{n,k}{X} \underset{k,m_{1j}}{\hat{\Pi}_j} & \underset{m_{1i},kn,k_{1j}}{X_{1i}^T X_{1j}} \end{bmatrix}$$

that come out to be

$$A_{ij} = \sigma^{ij} \begin{bmatrix} \underset{m_{1i},m_{1j}}{Y_{1i}^T X(X^TX)^{-1}X^TY_{1j}} & \underset{m_{1i},k_{1j}}{Y_{1i}^T X_{1j}} \\ \underset{k_{1i},m_{1i}}{X_{1i}^T Y_{1i}} & \underset{k_{1i},k_{1j}}{X_{1i}^T X_{1j}} \end{bmatrix}$$

The characteristic vector associated with the smallest characteristic root of matrix $\underset{s,s}{A}$ minimizes the quadratic form (2.30).

Let a be the smallest characteristic root of $\underset{s,s}{A}$ and P_a be the associated characteristic vector. The characteristic vector $P_a$multiplied by m suitable constants gives FI LODE.

Defining C as the block diagonal matrix

$$C = \begin{bmatrix} c_1 I_{m_{1i}+k_{1i}} & & \\ & \ddots & \\ & & c_1 I_{m_{1i}+k_{1i}} \end{bmatrix}$$

in which $c_i$are defined as follows

$$c_i = -\frac{1}{p_{0i}}$$

with p_{0i} being the characteristic vector's element corresponding to the endogenous variable y_{0i} chosen to be at left hand side in $i-th$ structural equation.

The FI estimator is then

$$\hat{\delta} = CP_a \tag{2.32}$$

the last step of this LODE FI presentation regards the estimate of disturbances variance-covariance matrix Ω, like we can see the Matrix Equation (2.30) which defines explicitly the quadratic form to be minimized is a function of Ω which is unknown. It is then necessary to estimate it. As usual it is possible to go through a two stage procedure: in the first stage estimates of the SF parameters are obtained using LI LODE which are then used to calculate the matrix of SF disturbances

$$\hat{U} = -\hat{V}\hat{\Gamma}$$

$\hat{V}$ been the matrix of RF equations' OLS residuals.

The matrix $\hat{\Omega}$ is then computed in the following way

$$\hat{\Omega} = G^{-\frac{1}{2}}\hat{U}^T\hat{U}G^{-\frac{1}{2}}$$

where

$$G^{-\frac{1}{2}} = \begin{bmatrix} \frac{1}{\sqrt{g_1}} & 0 & \cdots & \cdots & 0 \\ 0 & \ddots & \cdots & \cdots & \vdots \\ 0 & \cdots & \frac{1}{\sqrt{g_i}} & \cdots & 0 \\ \vdots & \cdots & \cdots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & \frac{1}{\sqrt{g_m}} \end{bmatrix}$$

with

$$g_i = n - m_{1i} - k_{1i}$$

It has to be notice that consistency of limited information SF parameters estimators implies the consistency of the variance covariance matrix estimators.

The second stage structural parameters estimates are then obtained introducing $\hat{\Omega}$ in equation (2.30). Full Information LODE is then proportional to the characteristic vector associated to the smallest characteristic root of

$$\hat{A} = \hat{\Pi}_*^T \left(\hat{\Omega}^{-1} \otimes \left(X^T X \right) \right) \hat{\Pi}_* \tag{2.33}$$

it is important notice that minimizes the quadratic form (2.33) means (like it's easy to see in 2.29) to minimize the trace of the sample estimate of the matrix Ω i.e. $\sum_{i=1}^{m} \hat{\sigma}_i^2$. In other words vector $\hat{\delta}$ gives rise to a matrix $\hat{\Omega}$ such that

$$tr\left(\hat{\Omega}\right) = \sum_{i=1}^{m} \hat{\sigma}_i^2 = min$$

from which follows according to (5.9), the consistent of this estimator is proved on Naccarato(2007)

Chapter 3

Simulation Experiment

3.1 Introduction

In this chapter is presented the first simulation experiment conduct to evaluate the features of the LODE FI.

Such evalutation has been conducted by comparing the results of LODE with Three Stage Least Square (3SLS) and Full Information Maximum Likelihood (FIML). In literature there are two main approaches to this kind of comparison: analytical (that focuses on searching the theoretical distribution of parameter estimators), or computational (based on Monte Carlo simulations).

As is well known, the difficulty in simultaneous equations estimation is the nonlinear relationship between Reduced Form (RF) and Structural Form (SF) coefficients. 3SLS, as well as FIML derives estimators under the hypothesis of identification restrictions. Thus the analytical approach refers to models that satisfy some sort of identification restrictions. This makes the analytical results unsuitable for more general applications.

The computational approach it is suited to handle more general models. It consists in choosing a model and assuming one or more structures by assigning specific numerical values to the parameters and to the variance-covariance matrix of the SF errors. Subsequently, samples of different sizes are extracted from the assumed error distribution and from each of the pre-

determined structures. After describing the experimental design (§ 1.2), a new computational procedure for FI LODE is briefly outlined and from all the next section and chapters we will refer to the FI LODE that use this computational procedure as FI LODE with empirical correction (§ 1.3). The results of the experiment when the distribution of the error component is Normal and when it is Uniform are presented (§ 1.4). Small sample results are then shown (§ 1.5).

3.2 The design of the experiment

Remembering the structural form

$$\underset{n,m}{Y}\underset{m,m}{\Gamma} + \underset{n,k}{X}\underset{k,m}{B} + \underset{n,m}{U} = \underset{n,m}{0}$$

The simulation has been conducted using the three equation model proposed by Cragg in 1967:

$$\begin{cases} y_1 = -0.89y_2 - 0.16y_3 + 44 + 0.74x_2 + 0.13x_5 \\ y_2 = -0.74y_1 + 62 + 0.7x_3 + 0.96x_5 + 0.06x_7 \\ y_3 = -0.29y_2 + 40 + 0.53x_3 + 0.11x_4 + 0.56x_6 \end{cases}$$

In our experiment it is then $n = 20$, 30, 100 and $m = 3$. Accordingly the structural form parameters matrices are

$$\Gamma = \begin{bmatrix} 1 & -0.89 & -0.16 \\ -0.74 & 1 & 0 \\ 0 & -0.29 & 1 \end{bmatrix} \quad B = \begin{bmatrix} 44 & 62 & 40 \\ 0.74 & 0 & 0 \\ 0 & 0.7 & 0.53 \\ 0 & 0 & 0.11 \\ 0.13 & 0.96 & 0 \\ 0 & 0 & 0.56 \\ 0 & 0.06 & 0 \end{bmatrix}$$

which have to be used to compute the reduced form of the system

$$\underset{n,m}{Y} = \underset{n,kk,m}{X\,\Pi} + \underset{n,m}{V}$$

where:

$$\underset{k,m}{\Pi} = -\underset{k,mm,m}{B\,\Gamma^{-1}} = \begin{bmatrix} 353.2 & 323.37 & 133.78 \\ 2.41 & 1.78 & 0.52 \\ 2.41 & 2.48 & 0.52 \\ 0.06 & 0.04 & 0.12 \\ 3.35 & 3.44 & 0.99 \\ 0.29 & 0.21 & 0.62 \\ 0.18 & 0.19 & 0.06 \end{bmatrix}$$

from this point the generation procedures starts going trough the following three steps.

1. *Exogenous variables generation.* For each sample size exogenous variables are generated from uniform distribution in the following intervals:

$$X_2 = [10 - 20]\,,\ X_3 = [15 - 27]\,,\ X_4 = [3 - 7]\,,$$
$$X_5 = [3 - 7]\,,\ X_6 = [20 - 50]\,,\ X_7 = [7 - 13]$$

   Exogenous values are kept constant for each sample size during the simulation experiment.

2. *Computation of endogenous variables unaffected by error.* Endogenous variables are generated through reduced form equation. Using the following notation for the endogenous variables not affected by error

$$Y^* = X\Pi \tag{3.1}$$

   where X is the matrix of generated exogenous variables.

3. *Variance covariance matrix of error component generation.* Taking in mind that

$$\underset{n,m}{V} = -\underset{n,mm,m}{U\,\Gamma^{-1}}$$

and that RF variance-covariance matrix

$$\Sigma = (\Gamma^{-1})\Omega\Gamma^{-1} \tag{3.2}$$

where Ω is the variance-covariance matrix of the SF error components. The matrix Ω has been chosen in the following way:
a) its diagonal elements (i.e the variances of the SF error component) are obtained as a proportion of the variance of i.e.

$$\omega_{ii} = \sigma_Z^2 S_i \tag{3.3}$$

where is a proportionality coefficient chosen randomly from a uniform distribution in three intervals :

$$[0.2 - 0.25]\,,\ [0.4 - 0.5]\,,\ [0.75 - 0.8]\,.$$

b) its extra diagonal elements (i.e. the covariances between error components in SF equations) are obtained generating randomly $m(m-1)/2$ correlation coefficients ρ_{ij} in the following intervals:

$$[0.1 - 0.2]\,,\ [0.4 - 0.5]\,,\ [0.8 - 0.9]$$

To each one of them is assigned a random sign. The covariance between error components in equation i and in equation j is computed as

$$\omega_{ij} = \rho_{ij}(\omega_{ii}\omega_{jj})^{\frac{1}{2}}$$

Then the matrix Σ is obtained according to (3.2).

4. *Genarating error components according to Normal and Uniform distributions.* For each sample of n observations, m series of random numbers are generated independently from a standardized Normal distribution and from a Uniform distribution in the interval $\left[-\sqrt{3}, \sqrt{3}\right]$ to have zero mean and variance one. According to the spectral decom-

position theorem the symmetric matrix Σ can be expressed as

$$\Sigma = P\Lambda P^T$$

Where P and Λ are respectively the matrix of characteristic vectors and the diagonal one of characteristic roots. Let

$$Q = P\Lambda^{\frac{1}{2}}P^T$$

$$\Sigma = Q^T Q$$

and let the matrix C $n \times m$, compose by the m series generated from Normal or Uniform distribution, then the set of contemporaneous dependent error components are obtained multiplying C with Q so

$$V = CQ \tag{3.4}$$

is a (multivariate) normally or uniformally distributed matrix with variance-covariance matrix Σ.

5. *Observed endogenous variables* adding V to the right side hand of (3.1), the matrix of observed endogenous variable is obtained.

The design of the experiment can be synthesized in the following table

Table 3.1: Simulation Scenarios

	S_i		
ρ_{ij}	0.2-0.25	0.4-0.5	0.75-0.8
0.1-0.2	N=20 N=30 N=100	N=20 N=30 N=100	N=20 N=30 N=100
0.4-0.5	N=20 N=30 N=100	N=20 N=30 N=100	N=20 N=30 N=100
0.8-0.9	N=20 N=30 N=100	N=20 N=30 N=100	N=20 N=30 N=100

The 27 scenarios listed are repeated for Normal and Uniform error components and for each scenario 500 samples have been considered.

To analyze the results of the simulation experiment we have taken into consideration two indicators which represent two relative measures of bias and variability around the parameter value:

- for bias, the following indicator has been considered

$$\varphi = \frac{\left(\hat{\theta} - \theta\right)}{\theta}$$

  (i.e. the bias divided by the fixed initial parameter value) where $\hat{\theta}$ is the average of estimated parameter over the 500 samples and θ is one of the γ or β parameters;

- for variability

$$\psi = \frac{RMSE}{\theta}$$

  where RMSE is the Root Mean Square Error of $\hat{\theta}$ which is divided by the initial parameter value.

The use of relative measures has been made to facilitate comparison among estimates of different parameters.

3.3 Computational procedure

First results of the simulations showed some problem with the estimation variability, this problem is not new for LODE estimator.

Precedent simulation experiment on LODE LI (Perna, 1989; Cau, 1990; Sbrana, 2001; Zurlo, 2006) had shown that usually in terms of bias the LODE LI works better than the other classics limited information estimators but it had some problem with the stability of the estimate, now the passage from LI LODE to FI LODE seems to have increased this problem, notice that the LODE LI enter in the estimation procedure of LODE FI.

Looking to results of this simulation can be notice that this variability comes from the fact that in the 500 estimation conducted for all 27 scenarios, every structural parameters estimation was affected by about 10 far outliers, where far outliers mean the value up to

$$\hat{\theta} + 3\sigma_{\hat{\theta}}$$

or down to

$$\hat{\theta} - 3\sigma_{\hat{\theta}}$$

where $\hat{\theta}$ is the average of estimated parameter over the 500 samples and $\sigma_{\hat{\theta}}$is the standard deviation of the same estimated parameter, this outliers doesn't give problem to the bias that was good compare with the 3SLS and FIML, but clearly the result in term of ψ was strongly influenced by this outliers.

In chapter's rest is illustrated the empirical computational procedure adopted to limited this problem.

We know from chapter 1 that the FI LODE estimator is obtained choosing the vector that minimize this quadratic form

$$\delta^T \hat{\Pi}_*^T \left(\Omega^{-1} \otimes \left(X^T X\right)\right) \hat{\Pi}_* \delta \tag{3.5}$$

this minimization is obtained trough those eigenvectors which are associated with the m smaller characteristic roots of the matrix

$$\underset{s,s}{A} = \hat{\Pi}_*^T \left(\Omega^{-1} \otimes \left(X^T X\right)\right) \hat{\Pi}_* =$$

$$= \begin{bmatrix} \hat{\Pi}_*^{1T}\sigma^{11}\left(X^T X\right)\hat{\Pi}_*^1 & \cdots & \hat{\Pi}_*^{iT}\sigma^{1i}\left(X^T X\right)\hat{\Pi}_*^i & \cdots & \hat{\Pi}_*^{mT}\sigma^{1m}\left(X^T X\right)\hat{\Pi}_*^m \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \hat{\Pi}_*^{1T}\sigma^{i1}\left(X^T X\right)\hat{\Pi}_*^1 & \cdots & \hat{\Pi}_*^{iT}\sigma^{ii}\left(X^T X\right)\hat{\Pi}_*^i & \cdots & \hat{\Pi}_*^{mT}\sigma^{im}\left(X^T X\right)\hat{\Pi}_*^m \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \hat{\Pi}_*^{1T}\sigma^{m1}\left(X^T X\right)\hat{\Pi}_*^1 & \cdots & \hat{\Pi}_*^{iT}\sigma^{mi}\left(X^T X\right)\hat{\Pi}_*^i & \cdots & \hat{\Pi}_*^{mT}\sigma^{mm}\left(X^T X\right)\hat{\Pi}_*^m \end{bmatrix}$$

Remembering that to minimize the quadratic form (3.5) means find the minimum of the trace of the sample estimate of the variance-covariance matrix of error component in structural form Ω i.e. $\sum_{i=1}^{m} \hat{\sigma}_i^2$.

It has to be stressed that the minimization of $tr\left(\hat{\Omega}\right)$ does not imply the minimization of each term of the sum, i.e. of every residual variance $\hat{\sigma}^2$ of the m equations.

With this in mind, the computational procedure for FI LODE that has been developed goes along the following lines of reasoning.

Let us assume – for the moment – that the error components are uncorrelated between equations (in this case the A matrix is block-diagonal)

$$A = \begin{bmatrix} \underset{s_1,s_1}{A_{11}} & \underset{s_1,s_2}{0} & \cdots & \underset{s_1,s_m}{0} \\ \underset{s_1,s_2}{0} & \underset{s_2,s_2}{A_{22}} & \ddots & \underset{s_2,s_m}{0} \\ \vdots & \ddots & \ddots & \vdots \\ \underset{s_m,s_1}{0} & \underset{s_m,s_2}{0} & \cdots & \underset{s_m,s_m}{A_{mm}} \end{bmatrix}$$

and let λ_{Amin} be its smallest characteristic root. Let $\lambda_{A_{11}min}...\lambda_{A_{mm}min}$ be the set of the smallest characteristic root of the m block diagonal matrices . It has to be

$$\lambda_{Amin} = min\{\lambda_{A_{11}min}, ..., \lambda_{A_{mm}min}\}$$

and it has to be noticed that it is not known "a priori" to which one of the smallest characteristic roots (i.e. to which one of the block diagonal matrix) it corresponds. Furthermore in this situation the characteristic vectors have

non zero elements only in correspondence to the block diagonal matrix to which they refer so if the characteristic vectors refer to the eigenvalue of A_{11}we will have

$$P_{\lambda_{A_{11}}} = \begin{bmatrix} \underset{s_1,1}{P} \\ \underset{s_2,1}{0} \\ \vdots \\ \underset{s_m,1}{0} \end{bmatrix}$$

where

$$s_i = m_{1i} + k_{1i}$$

is the sum of endogenous and esogenous variable presented in $i - th$ equation.

The FI LODE then reduces to the LI one if all the m smallest characteristic roots of the block diagonal matrices and their associated vectors are taken into account simultaneously.

In the usual case of correlated disturbances the matrix A is no more block diagonal. In analogy to the preceding case, the first m smaller characteristic roots and their associated vectors have been taken into account in the computational procedure. The characteristic vectors associated to the m smaller characteristic roots of matrix A are partitioned according to each equation

$$P_{\lambda_{Aii}} = \begin{bmatrix} \begin{bmatrix} \underset{s_{1i},1}{P} \end{bmatrix} \\ \begin{bmatrix} \underset{s_{2i},1}{P} \end{bmatrix} \\ \vdots \\ \begin{bmatrix} \underset{s_{mi},1}{P} \end{bmatrix} \end{bmatrix}$$

with

$$i = 1, ...m$$

The sub-vector, among the m, which minimizes the estimate residual variance of his equation gives, after normalization, the FI LODE of each equation's structural parameters. The reordered set of minimizing sub-vectors gives the vector of estimates.

In the from the next chapter until the endo of this thesis we will refere to this method as the LODE FI with empirical correction.

3.4 Results of the experiment

To synthesize results of the simulation experiment, the percentage of times in which parameters' estimators present the lowest bias or RMSE among the three estimation method has been considered both for Normal and Uniform distribution.

3.4.1 Normal error component

First let us consider the case in which the error component is distributed according to a Multivariate Normal $(0, \Sigma)$, where Σ is the variance-covariance matrix of reduced form and the error component with that variance is obtained through (3.4) starting from the generation of normally standardized independent random numbers. For small sample sizes most of time LODE estimator outperforms the other two estimator precisely 5 times on 9 in terms of bias, while FIML estimator display the best results considering the whole simulation scenarios (Tab.3.2).

Table 3.2: Tab. 1 Relative frequency distribution of FI LODE, 3SLS and FIML presenting a lower bias gruoped by S_i, ρ_{ij} and sample size - Normal error component

		ρ_{ij}								
		0.1-0.2			0.4-0.5			0.8-0.9		
S_i	Sample Size	LODE	3SLS	FIML	LODE	3SLS	FIML	LODE	3SLS	FIML
0.2-0.25		53.33	0.00	46.67	26.67	13.33	60.00	33.33	13.33	53.33
0.4-0.5	20	73.33	6.67	20.00	46.67	13.33	40.00	73.33	0.00	26.67
0.75-0.8		66.67	13.33	20.00	33.33	13.33	53.33	46.67	0.00	53.33
0.2-0.25		26.67	0.00	73.33	20.00	0.00	80.00	13.33	0.00	86.67
0.4-0.5	30	26.67	0.00	73.33	20.00	0.00	80.00	53.33	20.00	26.67
0.75-0.8		26.67	0.00	73.33	46.67	0.00	53.33	46.67	0.00	53.33
0.2-0.25		6.67	0.00	93.33	6.67	6.67	86.67	26.67	0.00	73.33
0.4-0.5	100	0.00	0.00	100.00	13.33	0.00	86.67	6.67	6.67	86.67
0.75-0.8		6.67	0.00	93.33	0.00	6.67	93.33	13.33	0.00	86.67

Note that when the correlation coefficient between the disturbances displays small values $(0,1-0,2)$ and for all the variance values (S_i) considered in the experiment, LODE estimator shows similar performances of FIML. For increasing values of the correlation coefficient this result becomes weaker, that is the frequency of better results of LODE estimator decrease when the correlation between the error components increases.

In terms of RMSE, the estimates obtained with FIML and 3SLS estimators show more frequently lower values than LODE method (Tab.3.3).

Notice that when LODE is compared only to 3SLS estimators LODE estimators display lower bias than 3SLS almost for all simulation conditions and for all sample sizes considered, confirming in this way the results already obtained.

Table 3.3: Relative frequency distribution of FI LODE, 3SLS and FIML presenting a lower RMSE gruoped by S_i, ρ_{ij} and sample size - Normal error component

		ρ_{ij}								
		0.1-0.2			0.4-0.5			0.8-0.9		
S_i	Sample Size	LODE	3SLS	FIML	LODE	3SLS	FIML	LODE	3SLS	FIML
0.2-0.25		6.67	33.33	60.00	0.00	26.67	73.33	20.00	20.00	60.00
0.4-0.5	20	20.00	40.00	40.00	13.33	26.67	60.00	40.00	20.00	40.00
0.75-0.8		26.67	46.67	26.67	13.33	20.00	66.67	6.67	26.67	66.67
0.2-0.25		6.67	13.33	80.00	0.00	6.67	93.33	13.33	6.67	80.00
0.4-0.5	30	6.67	53.33	40.00	0.00	33.33	66.67	26.67	20.00	53.33
0.75-0.8		40.00	33.33	26.67	46.67	40.00	13.33	20.00	33.33	46.67
0.2-0.25		13.33	6.67	80.00	6.67	20.00	73.33	6.67	6.67	86.67
0.4-0.5	100	6.67	13.33	80.00	0.00	13.33	86.67	0.00	13.33	86.67
0.75-0.8		20.00	6.67	73.33	13.33	26.67	60.00	6.67	20.00	73.33

Since 3SLS estimation is generally preferred to FIML, because the latter has sometimes computational problems, it is worthwhile to stress the point. Furthermore normality assumption for the error component is often not practical.

3.4.2 Uniform error component

In order to obtain results comparable with normally distributed error components, a second simulation experiment has been carried out using the Uniform distribution in $(-\sqrt{3}, \sqrt{3})$.

About the bias of the estimators it has to be notice that LODE estimator shows lower bias than 3SLS and FIML more frequently than the results obtained under Normality condition (Tab.3.4). This is particularly true for the scenarios related to small sample sizes. Similarly to what has been seen previously, when the correlation coefficient between the disturbances increases FIML estimator presents more frequently estimates affected by lower bias.

Table 3.4: Tab. 3 Relative frequency distribution of FI LODE, 3SLS and FIML presenting a lower bias gruoped by S_i, ρ_{ij} and sample size - Uniform error component

		ρ_{ij}								
		0.1-0.2			0.4-0.5			0.8-0.9		
S_i	Sample Size	LODE	3SLS	FIML	LODE	3SLS	FIML	LODE	3SLS	FIML
0.2-0.25		53.33	6.67	40.00	33.33	20.00	46.67	73.33	13.33	13.33
0.4-0.5	20	66.67	0.00	33.33	80.00	6.67	13.33	6.67	33.33	60.00
0.75-0.8		93.33	0.00	6.67	40.00	33.33	26.67	60.00	0.00	40.00
0.2-0.25		53.33	0.00	46.67	53.33	0.00	46.67	0.00	0.00	100.00
0.4-0.5	30	53.33	0.00	46.67	73.33	0.00	26.67	0.00	0.00	100.00
0.75-0.8		53.33	6.67	40.00	20.00	0.00	80.00	20.00	0.00	80.00
0.2-0.25		20.00	0.00	80.00	40.00	26.67	33.33	13.33	13.33	73.33
0.4-0.5	100	26.67	0.00	73.33	6.67	0.00	93.33	6.67	0.00	93.33
0.75-0.8		60.00	6.67	33.33	13.33	0.00	86.67	13.33	0.00	86.67

The comparison in terms of RMSE (Tab.3.5) shows that FIML estimators are still to be preferred since the number of times they produce estimates with lower RMSE is very high for all the scenarios considered.

Table 3.5: Tab. 4 Relative frequency distribution of FI LODE, 3SLS and FIML presenting a lower RMSE gruoped by S_i, ρ_{ij} and sample size - Uniform error component

		ρ_{ij}								
		0.1-0.2			0.4-0.5			0.8-0.9		
S_i	Sample Size	LODE	3SLS	FIML	LODE	3SLS	FIML	LODE	3SLS	FIML
0.2-0.25		20.00	33.33	46.67	26.67	13.33	60.00	6.67	13.33	86.67
0.4-0.5	20	6.67	80.00	13.33	6.67	20.00	73.33	0.00	13.33	86.67
0.75-0.8		6.67	60.00	33.33	13.33	46.67	40.00	26.67	26.67	46.67
0.2-0.25		40.00	13.33	46.67	13.33	26.67	60.00	0.00	13.33	86.67
0.4-0.5	30	6.67	20.00	73.33	20.00	13.33	66.67	0.00	26.67	73.33
0.75-0.8		6.67	60.00	33.33	0.00	26.67	73.33	13.33	20.00	66.67
0.2-0.25		40.00	13.33	46.67	26.67	20.00	53.33	6.67	26.67	66.67
0.4-0.5	100	0.00	13.33	86.67	6.67	20.00	73.33	6.67	13.33	80.00
0.75-0.8		0.00	6.67	93.33	0.00	26.67	73.33	6.67	6.67	86.67

Considering that standardized Uniform distribution has a very short range of variation, in order to evaluate more deeply the effect of more scattered errors components a second Uniform distribution has been considered in the interval $(-10, 10)$. In point of fact in this situation, LODE estimator performs better than FIML in terms of both bias and RMSE.

These results represent an improvements with respect to the previous uniform distribution; the bias of LODE estimators are largely better than

FIML estimators (Tab.3.6). In point of fact LODE estimators perform better than the others in terms of bias, in most of the scenarios. This happens more frequently when dealing with small sample. Moreover, it has to be noticed that – differently from the previous two cases considered – the results related to LODE estimators do not seem to be affected by the correlation between the error components as on the contrary it is for the other two methods.

Table 3.6: Tab. 5 Relative frequency distribution of FI LODE, 3SLS and FIML presenting a lower bias gruoped by S_i, ρ_{ij} and sample size - Unif(-10, 10) error component

		ρ_{ij}								
		0.1-0.2			0.4-0.5			0.8-0.9		
S_i	Sample Size	LODE	3SLS	FIML	LODE	3SLS	FIML	LODE	3SLS	FIML
0.2-0.25		60.00	13.33	26.67	40.00	13.33	46.67	40.00	6.67	53.33
0.4-0.5	20	40.00	13.33	46.67	60.00	6.67	33.33	60.00	13.33	26.67
0.75-0.8		40.00	40.00	20.00	33.33	20.00	46.67	46.67	13.33	40.00
0.2-0.25		86.67	13.33	0.00	73.33	13.33	13.33	53.33	13.33	33.33
0.4-0.5	30	46.67	33.33	20.00	60.00	13.33	26.67	66.67	13.33	20.00
0.75-0.8		66.67	33.33	0.00	46.67	20.00	33.33	60.00	26.67	13.33
0.2-0.25		80.00	13.33	6.67	73.33	20.00	6.67	46.67	6.67	46.67
0.4-0.5	100	80.00	20.00	0.00	13.33	86.67	0.00	66.67	6.67	26.67
0.75-0.8		80.00	20.00	0.00	26.67	0.00	73.33	86.67	13.33	0.00

As far as RMSE of estimators are concerned (Tab.3.7), when the disturbances are uniformly distributed in the comparison has to be made only between LODE and 3SLS, since every time FIML estimators produce higher RMSE than the other two methods. 3SLS estimation mostly presents a lower RMSE, with the exception of some cases in which LODE outperforms it.

Table 3.7: Tab. 6 Relative frequency distribution of FI LODE, 3SLS and FIML presenting a lower RMSE gruoped by S_i, ρ_{ij} and sample size - Unif(-10,10) error component

		ρ_{ij}								
		0.1-0.2			0.4-0.5			0.8-0.9		
S_i	Sample Size	LODE	3SLS	FIML	LODE	3SLS	FIML	LODE	3SLS	FIML
0.2-0.25		40.00	60.00	0.00	26.67	66.67	6.67	53.33	46.67	6.67
0.4-0.5	20	53.33	46.67	0.00	20.00	73.33	6.67	20.00	80.00	0.00
0.75-0.8		53.33	46.67	0.00	46.67	53.33	0.00	46.67	53.33	0.00
0.2-0.25		46.67	53.33	0.00	13.33	86.67	0.00	40.00	60.00	0.00
0.4-0.5	30	60.00	40.00	0.00	6.67	93.33	0.00	40.00	60.00	0.00
0.75-0.8		53.33	40.00	6.67	53.33	46.67	0.00	20.00	80.00	0.00
0.2-0.25		40.00	60.00	0.00	20.00	80.00	0.00	40.00	40.00	20.00
0.4-0.5	100	6.67	93.33	0.00	33.33	66.67	0.00	53.33	46.67	0.00
0.75-0.8		26.67	73.33	0.00	13.33	86.67	0.00	46.67	53.33	0.00

The results of this experiment seem to have solve the problem of far outliers, none of 27 scenarios there is a far outliers, and show a good performance of LODE FI and as it was pointed out in the previous paragraph, LODE method seems to work particularly well when dealing with small samples. A recently article (Zurlo, Naccarato 2009) go in greater details about this point dedicated a section to the case of sample size equal to 20, highlighting the good features of LODE estimator with empirical correction in this situation, but like it was said at the beginning section 1.4 it is just an empirical solution in the next two chapter i will present a more formal solution for LODE FI.

Chapter 4

GLODE AND LODE

4.1 Introduction

As Said in the previous chapter the first simulation experiment's result on FI LODE showed some problem about the estimation variability, in that chapter at section 3.3 shows an empirical solution to this problem, the next two chapter that are the core of this Phd's thesis try to find a more analytic solution to this problem.

One cause can come from the fact that full and limited information LODE estimates at the same time the parameters of exogenous variables and those of endogenous variables, this practice type can lead to a biased estimation of exogenous variable.

Notice that in this chapter we will see two different version of LODE LI to understand we will call the last version of limited information lode, presented in chapter 1, generalized LODE LI(GLODE LI) from the name give to its from Sbrana (2001) while the primal version (Pieraccini, 1988), briefly illustrated in section 4.3, will be simply LODE LI

4.2 Intuitive Least Orthogonal Distance Estimator interpretation

As shown in the chapter 1 the GLODE estimators estimate together both parameters of exogenous variable and endogenous using $\hat{\Pi}^i_*$(2.13) for limited information (and $\hat{\Pi}_*$(2.25) for full information), this technique can bring some problem. The problem will be explain for $i-th$ equation.

First of all, to better understand, we have to introduce the normalization rule for Γ_i parameters of endogenous variables, so if we consider the $i-th$ equation of a simultaneous equations model we will have

$$\underset{m,1}{\Gamma_i} = \begin{bmatrix} -1 \\ \cdots \\ \underset{m_{1i}-1,1}{\Gamma_{1i}} \\ \cdots \\ \underset{m_{2i},1}{0} \end{bmatrix}$$

where the first element -1 is the coefficient of dependent variable in the $i-th$ equation, now the matrix Y_i will be

$$\underset{n,m}{Y_i} = \begin{bmatrix} \underset{n,1}{Y_{0i}} & \vdots & \underset{n,m_{1i}-1}{Y_{1ie}} & \vdots & \underset{n,m_{2i}}{Y_{2i}} \end{bmatrix}$$

where Y_{0i} is the dependent variable vector of $i-th$ equation and Y_{1ie} are the other endogenous variable included in equation, if we recall the partition of variable exogenous X_i and their coefficient B_i

$$\underset{n,k}{X_i} = \begin{bmatrix} \underset{n,k_{1i}}{X_{1i}} & \vdots & \underset{n,k_{2i}}{X_{2i}} \end{bmatrix}, \ \underset{k,1}{B_{1i}} = \begin{bmatrix} \underset{k_{1i},1}{B_{1i}} \\ \cdots \\ \underset{k_{2i},1}{0} \end{bmatrix}$$

the $i-th$ equation can be written as

$$Y_{0i} = Y^*_{1i}\Gamma_{1i} + X_{1i}B_{1i} + U_i$$

and the matrix Π_i is :

$$\underset{k_i,m_i}{\Pi^i} = \begin{bmatrix} \underset{k_{1i},1}{\pi^i_{01}} & \underset{k_{1i},m_{1i}-1}{\Pi^i_{11e}} & \underset{k_{1i},m_{2i}-1}{\Pi^i_{21}} \\ \underset{k_{2i},1}{\pi^i_{02}} & \underset{k_{2i},m_{1i}-1}{\Pi^i_{12e}} & \underset{k_{2i},m_{2i}-1}{\Pi^i_{22}} \end{bmatrix}$$

where π^i_{01} and π^i_{02} are the coefficient vector of reduced form related to endogenous dependent variable and to exogenous variable respectively included and excluded from $i-th$ equation , Π^i_{11e} and Π^i_{12e} are the coefficient matrix of reduced form related to the other endogenous variable included and to exogenous variable respectively included and excluded (notice that this matrices it's equal to $\Pi^i_{11}$and Π^i_{12} less the first columns), while Π^i_{21} refers to parameters of endogenous excluded and exogenous variables included and Π^i_{22} refers to the $i-th$ equation RF parameters of endogenous and exogenous variables excluded.

Then the identification system(2.12), after the substitution of parameters Π with their OLS estimates $\hat{\Pi}$, will be

$$\begin{cases} \underset{k_{1i},1}{\hat{\pi}^i_{01}} = \underset{k_{1i},m_{1i}-1}{\hat{\Pi}^i_{11e}} \underset{m_{1i}-1,1}{\Gamma_{1i}} + \underset{k_{1i},1}{B_{1i}} + \underset{k_{1i},1}{\varepsilon_{1i}} \\ \underset{k_{2i},1}{\hat{\pi}^i_{02}} = \underset{k_{2i},m_{1i}-1}{\hat{\Pi}^i_{12e}} \underset{m_{1i}-1,1}{\Gamma_{1i}} + \underset{k_{2i},1}{\varepsilon_{2i}} \end{cases} \tag{4.1}$$

looking at the second equation of model (4.1),it is known from Pieraccini(1969) that

$$\varepsilon_{2i} = R_{2i} X_i^T U_{1i}$$

where R_{2i} comes from

$$\underset{k,k}{(X_i^T X_i)^{-1}} = \begin{bmatrix} \underset{k_{1i},k}{R_{1i}} \\ \underset{k_{2i},k}{R_{2i}} \end{bmatrix} = \begin{bmatrix} \underset{k_{1i},k_{1i}}{R_{11ii}} & \underset{k_{1i},k_{2i}}{R_{12ii}} \\ \underset{k_{2i},k_{1i}}{R_{21ii}} & \underset{k_{2i},k_{21}}{R_{22ii}} \end{bmatrix}$$

with

$$E(\varepsilon_{2i}) = 0$$

and

$$E(\varepsilon_{2i}\varepsilon_{2i}^T) = \sigma_i^2 R_{22ii} \tag{4.2}$$

the errors result correlated each other, so we need to apply a transformation to eliminate this correlation, according to the Spectral Decomposition Theorem the symmetric matrix R_{22ii} can be expressed as:

$$R_{22ii}^{-1} = C\Lambda C^T$$

where C and Λ are respectively the matrix of eigenvectors and the diagonal matrix of eigenvalues of R_{22ii}^{-1}.

Let:

$$Q_{22i} = C\Lambda^{\frac{1}{2}}C^T$$

and pre-multiplying the element of the second equation of (4.1) for Q_{22i} it will become

$$Q_{22i}\hat{\pi}_{02}^i = Q_{22i}\hat{\Pi}_{12e}^i\Gamma_{1i} + Q_{22i}\varepsilon_{2i} \tag{4.3}$$

with

$$E(Q_{22i}\varepsilon_{2i}\varepsilon_{2i}^T Q_{22i}^T) = \sigma_i^2 I \tag{4.4}$$

Anderson(1976) proved that the equation (4.3) is mathematically identical with the problem to estimate a coefficient that links two variables affected by error, and denote with $\underset{k,1}{y}^0$and $\underset{k,m}{X}^0$ the true value of y and X $k > m$, so we have

$$\underset{k,1}{y} = \underset{k,1}{y}^0 + \underset{k,1}{\varepsilon_y}$$

$$\underset{k,m}{X} = \underset{k,m}{X}^0 + \underset{k,m}{\varepsilon_X}$$

and the link between the two variables is

$$\underset{k,1}{y^0} = \underset{k,m}{X^0} \underset{m,1}{\beta}$$

but to estimate β we work with the equation

$$\underset{k,1}{y} = \underset{k,m}{X} \underset{k,1}{\beta} + \underset{k,1}{\varepsilon} \tag{4.5}$$

with

$$\varepsilon = \varepsilon_y + \varepsilon_X \beta$$

and

$$\underset{k,1}{\varepsilon} \sim i.i.d\,(0, \sigma^2 I)$$

a strongly consistent estimator of this model use the least orthogonal distance Adcock (1878) Pearson (1901) Koopmans (1937) and Madasky (1959), what is it the difference between estimator based on least square distance or least orthogonal distance?

The LS solution is obtained by projecting y orthogonally into the column space of X and solving $y' = X\beta$, this estimator works when $X = X^0$, so when it is error free. An illustration of the geometry of LS solution is depicted in Fig.4.1 for m=2, the LS estimator minimizes the square norm of the vector $y - y'$then

$$\underset{y}{min} \, \|y - y'\|_2^2$$

it means that it tries to minimize projecting vector's lenght of y in X to estimate β.

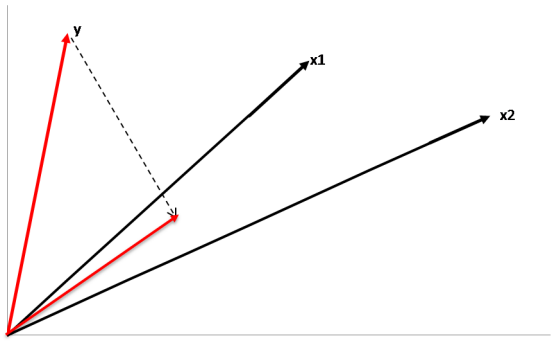

Figure 4.1: LS estimates

Indeed the LO solution is obtained by approximating the columns x_i of X and y by x_i' and y' until y' is in the column space of $X^{'}$ and $y' = X^{'}\beta$, illustrated in Fig.4.2, is tantamount to finding a closest subspace create by the column of $\left[y'; X^{'}\right]$ to the column space of $[y; X]$ it means minimize the Frobenius norm of the matrix $[y; X] - \left[y'; X^{'}\right]$ this because both y and X are subject to error then

$$\underset{[y;X]}{min} \left\| [y; X] - \left[y'; X^{'}\right] \right\|_F^2$$

LO estimates minimizes, simultaneously, vector's lenght of the projecting matrix $[y; X] - \left[y'; X^{'}\right]$to estimate β.

Notice that the Frobenius norm of a matrix M of dimension $m \times n$ is defined by

$$||M||_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} m_{ij}^2} = \sqrt{tr(M^T M)}$$

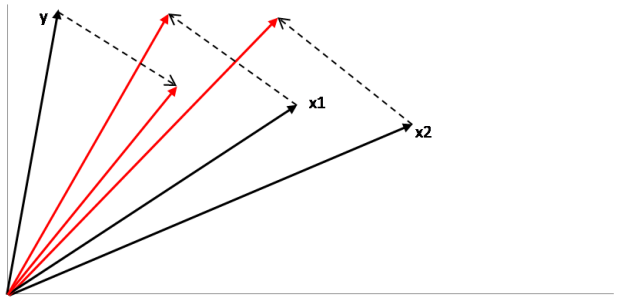

Figure 4.2: LO estimates

4.3 LODE and GLODE

In this section we will apply this interpretation of the least orthogonal solution on our case, in the first part we briefly recall the first version of LODE Pieraccini (1988), then we will compare this version with the actual GLODE with the aim of illustrating the main difference and the problem of this last version.

Replacing the elements y and X of equation (4.3) respectively with $Q_{22i}\hat{\pi}^i_{02}$ and $Q_{22i}\hat{\Pi}^i_{12e}$ is reasonable using LO solution to estimate Γ_{1i}, because $\hat{\pi}^i_{02}$ and $\hat{\Pi}_{12i}$ come from the OLS estimates of reduced form so they are both affected by error.

Calling

$$\omega_{i2} = Q_{22i}\varepsilon_{2i}$$

We know from (4.4) that

$$E\left(\omega_{i2}\omega_{i2}^T\right) = \sigma_i^2 I$$

remembering that if you have a vector V and a matrix H, this relation is true $V^T HV = tr\{V^T VH\}$and using the trace properties, obtained

$$E\left(\omega_{i2}^T\omega_{i2}\right) = E\left(\varepsilon_{2i}^T R_{22ii}^{-1}\varepsilon_{2i}\right) = k_{2i}\sigma_i^2$$

and because of

$$\varepsilon_{2i} = \hat{\Pi}^{i}_{12i}\Gamma_{1i}$$

we have

$$E\left(\omega^{T}_{i2}\omega_{i2}\right) = E\left(\Gamma^{T}_{1i}\hat{\Pi}^{iT}_{12}R^{-1}_{22ii}\hat{\Pi}^{i}_{12}\Gamma_{1i}\right) = k_{2i}\sigma^{2}_{i}$$

with

$$\underset{k_{2i},m_{1i}}{\hat{\Pi}^{i}_{12}} = \begin{bmatrix} \underset{k_{2i},1}{\pi^{i}_{02}} & \underset{k_{2i},m_{1i}-1}{\Pi^{i}_{12e}} \end{bmatrix}$$

is now reasonable, to estimate Γ_{1i}, choosing the vector P_{2i} that minimize

$$P^{T}_{2i}\hat{\Pi}^{iT}_{12}R^{-1}_{22ii}\hat{\Pi}^{i}_{12}P_{2i} = \hat{\sigma}^{2}_{i}$$

i.e. the eigenvector that correspond to the smallest eigenvalue of

$$\hat{\Pi}^{Ti}_{12}R^{-1}_{22ii}\hat{\Pi}^{i}_{12}$$

the proof is equal to the one give in section 1.2 for LODE LI generalized and comes from Pieraccini (1988), hence

$$\hat{\Gamma}_{1i} = -\frac{1}{p_{02i}}P_{2i}$$

where p_{02i} is the element of the eigenvector associated with the right hand side endogenous variable in the $i-th$ structural equation.

To estimate B_{1i} we have just to apply the OLS on this equation

$$Y_{0i} - Y_{1ie}\hat{\Gamma}_{1i} = X_{1i}B_{1i} + E_{i}$$

where $E_{i}$is the new error component of $i-th$ equation, so

$$\hat{B}_{1i} = (X^{T}_{1i}X_{1i})^{-1}X^{T}_{1i}(Y_{0i} - Y^{*}_{1i}\hat{\Gamma}_{1i}) \tag{4.6}$$

consistency of this method is proved in Perna (1988)

As mentioned early this is the first version of LODE limited information,

afterwards this practice was replaced by another one introduced by Sbrana (2001) and introduce in section 2.3, in this case it used $\hat{\Pi}^i_*$, to estimate simultaneously B_{1i} and Γ_{1i} from the system(4.1).

$\hat{\Pi}^i_*$ can be rewrite as

$$\underset{k,m_1+k_1}{\hat{\Pi}^i_*} = \begin{bmatrix} \underset{k_{1i},1}{\hat{\pi}^i_{01}} & \underset{k_{1i},m_{1i}-1}{\hat{\Pi}^i_{11e}} & \underset{k_{1i}}{I} \\ \underset{k_{2i},1}{\hat{\pi}^i_{02i}} & \underset{k_{2i},m_{1i}-1}{\hat{\Pi}^i_{12e}} & \underset{k_{1i},k_{2i}}{0} \end{bmatrix}$$

where $\hat{\pi}_{01i}$ and $\hat{\pi}_{02i}$ are the OLS estimates of coefficient vector of reduced form, related to endogenous dependent variable and to exogenous variable respectively included and excluded from $i-th$ equation , $\hat{\Pi}_{11i}$ and $\hat{\Pi}_{12i}$ are the OLS estimates of coefficient matrix of reduced form related to the other endogenous variable included and to exogenous variable respectively included and excluded, while the error component of all the system (4.1) is

$$\underset{k,1}{\varepsilon_i} = \begin{bmatrix} \underset{k_{1i},1}{\varepsilon_{1i}} \\ \underset{k_{2i},1}{\varepsilon_{2i}} \end{bmatrix}$$

and it is know from Pieraccini(1978) that

$$E(\varepsilon_i) = 0$$

and

$$E(\varepsilon_i \varepsilon_i^T) = \sigma_i^2 (X_i^T X_i)^{-1}$$

as it was shown with the second equation of system (4.1) according to the Spectral Decomposition Theorem the symmetric matrix $(X_i^T X_i)^{-1}$ can be expressed as:

$$(X_i^T X_i)^{-1} = T \Xi T^T$$

where T and Ξ are respectively the matrix of eigenvectors and the diagonal matrix of eigenvalues of $(X_i^T X_i)^{-1}$.

Let:

$$Q = T\Xi^{\frac{1}{2}}T^T$$

pre-multiplying the element of (4.1) for Q it will become

$$Q\hat{\pi}^i = Q\hat{\Pi}^i_e\delta_i + Q\varepsilon_i \tag{4.7}$$

with

$$\underset{k,1}{\hat{\pi}^i} = \begin{bmatrix} \underset{k_{1i},1}{\hat{\pi}^i_{01}} \\ \underset{k_{2i},1}{\hat{\pi}^i_{02}} \end{bmatrix} \quad \underset{k,\, m_{1i}-1+k_{1i}}{\hat{\Pi}^i_{e*}} = \begin{bmatrix} \underset{k_{1i},m_{1i}-1}{\hat{\Pi}^i_{11e}} & \underset{k_{1i}}{I} \\ \underset{k_{2i},m_{1i}-1}{\hat{\Pi}^i_{12e}} & \underset{k_{2i},k_{1i}}{0} \end{bmatrix} \quad \underset{m_{1i}+k_{1i}-1,1}{\delta_i} = \begin{bmatrix} \underset{m_{1i}-1,1}{\Gamma_{1i}} \\ \underset{k_{1i,1}}{B_{1i}} \end{bmatrix}$$

and the variance of accidental component will be

$$E(Q\varepsilon_i\varepsilon_i^T Q^T) = \sigma_i^2 I$$

in the GLODE is applied the LO solution on equation (4.7) estimating Γ_{1i} and B_{1i}, but while in equation (4.3) all columns of matrix dependent variables $\hat{\Pi}_{12i}$ are subject to error, in equation (4.7) only the first m_{1i-1} columns of the matrix dependent variables $\hat{\Pi}_i$ are subject to error, because comes from OLS estimates, the other k_{1i} columns (that were introduced to work with whole system and to be able to estimate B_{1i}) can be seen like error free variable, then in this case the best solution for B_{1i} is the LS.

4.4 biased estimates of GLODE

The aim of this section is to explain and better understand, where problem GLODE come.

If we take the matrix

$$A_{2ii} = \hat{\Pi}^{iT}_{12} R^{-1}_{22ii} \hat{\Pi}^i_{12}$$

we can say that

$$\widehat{E(A_{2ii})}(k_{2i}) = \frac{1}{k_{2i}}\{\hat{\Pi}_{12}^{iT} R_{22ii}^{-1} \hat{\Pi}_{12}^{i}(k_{2i})\}$$

and we can easily assume

$$\widehat{E(A_{2ii})}(k_{2i}) \to E(A_{2ii})$$

as

$$k_{2i} \to \infty$$

where

$$E(A_{2ii}) = E\{\hat{\Pi}_{12}^{iT} R_{22ii}^{-1} \hat{\Pi}_{12}^{i}(k_{2i})\}$$

if we assume that noise corrupt all the elements of $Q_{22i}\hat{\Pi}_{12}^{i}$ and that this error have variance equal to $\sigma_i^2 I$ (4.4) then, as we know from Dunne and Goffrey (2003), $E(A_{2ii})$ can be written as

$$E(A_{2ii}) = E\{\Pi_{12}^{iT} R_{22ii}^{-1} \Pi_{12}^{i}(k_{2i})\} + \sigma_i^2 I \tag{4.8}$$

the LODE solution is the eigenvector that correspond to the smallest eigenvalue of $E(A_{2ii})$ therefore as can be seen from (4.8), the error modifies the eigenvalues, but not the direction of eigenvector.

If we have a symmetric matrix A with eigenvector v and correspondent eigenvalues a we know that $(A - aI)v = 0$, in the same way a matrix $B = A + cI$ with c that it's a scalar will have eigenvalue equal to $c + a$ and eigenvector equal to v this because

$$(B - (a + c)I)v = (A + cI - (a + c)I)v = (A - aI)v = 0.$$

Furthermore, since the noise changes all of the eigenvalues equally, the eigenvectors for error $E(A_{2ii})$ or error free $E\{\Pi_{12}^{iT} R_{22ii}^{-1} \Pi_{12}^{i}(k_{2i})\}$ case must be the same with $k_{2i} \to \infty$ and in this circumstances the LODE estimator works and is unbiased.

Conversely in the case of GLODE where we work with a matrix $\hat{\Pi}_i^*$ where

not of the element are affected by error. Here, the noise does not modify all of the eigenvalues of $E(A_{ii})$ equally where

$$E(A_{ii}) = E\{\hat{\Pi}_i^{*T}\left(X^T X\right)\hat{\Pi}_i^*(k_i)\}$$

because the last k_{1i} columns are not affected by error and so the eigenvectors for error $E(A_{ii})$ or error free $E\{\Pi_i^{*T}\left(X^T X\right)\Pi_i^*(k_i)\}$ will not be the same. Thus, we can conclude that GLODE will yield a biased estimator for B_{1i} in this case.

For this reason the idea is to come back to the primal version of LODE and estimating before the Γ_{1i} from (4.3) and then B_{1i} using OLS (4.6).

4.5 FI LODE not generalized

Taking into account the result of section 3.4 and notice that if the error components are uncorrelated between equations (in this case the A matrix is block-diagonal) the full information estimator equal limited information, the full information LODE can be rewrite according to the primal version of LODE LI, it means that the estimation procedure is divided in two parts, like in LODE LI, before we provide the estimation of Γ structural parameters using orthogonal distance minimization and then, through OLS, we will have our $\hat{B}$.

Notice that the second equation of system of identification(2.8) for the whole system of equation can be written as

$$\underset{r,z}{\hat{\Pi}_*} = \begin{bmatrix} \underset{k_{21},m_{11}}{\hat{\Pi}_{12}^1} & \underset{k_{21},m_{12}}{0} & \cdots & \underset{k_{21},m_{1m}}{0} \\ \underset{k_{22},m_{11}}{0} & \underset{k_{22},m_{12}}{\hat{\Pi}_{12}^2} & \cdots & \underset{k_{22},m_{1m}}{0} \\ \vdots & \vdots & \ddots & \vdots \\ \underset{k_{2m},m_{11}}{0} & \underset{k_{2m},m_{12}}{0} & \cdots & \underset{k_{2m},m_{1m}}{\hat{\Pi}_*^m} \end{bmatrix}$$

with

$$z = \sum_{i=1}^{m} m_{1i}$$

$$r = \sum_{i=1}^{m} k_{2i}$$

$$\underset{z,1}{\Gamma_1} = \begin{bmatrix} \underset{m_{11},1}{\Gamma_{11}} \\ \underset{m_{12},1}{\Gamma_{12}} \\ \vdots \\ \underset{m_{1m},1}{\Gamma_{1m}} \end{bmatrix}$$

and

$$\underset{r,1}{\varepsilon_2} = \begin{bmatrix} \varepsilon_{21} \\ \varepsilon_{22} \\ \vdots \\ \varepsilon_{2m} \end{bmatrix} \tag{4.9}$$

with

$$\underset{r,1}{\varepsilon_2} = \left[I_m \otimes \underset{k_2,n}{R_2 X_2^T} \right] \underset{nm_1,1}{vec(V\Gamma)} = \left[I_m \otimes \underset{k_2,n}{R_2 X_2^T} \right] \underset{nm_1,1}{vec(U)}$$

where R_2 comes from

$$\underset{k,k}{(X^T X)^{-1}} = \begin{bmatrix} \underset{k_1,k}{R_1} \\ \underset{k_2,k}{R_2} \end{bmatrix} = \begin{bmatrix} \underset{k_1,k_1}{R_{11}} & \underset{k_1,k_2}{R_{12}} \\ \underset{k_2,k_1}{R_{21}} & \underset{k_2,k_2}{R_{22}} \end{bmatrix}$$

where the matrix X $n \times k$ represents all the k X included and excluded in each m equation of the system.

Because of (4.2) applied to the vector defined in 4.9 the variance-covariance matrix of the accidental component will be

$$E(\varepsilon_2 \varepsilon_2^T) = \underset{r,r}{\Sigma} = \underset{m,m}{\Omega} \otimes \underset{k_2,k_2}{R_{22}}$$

and matrix, from which we calculate the characteristic vector and the characteristic root, to minimize the trace of this variance-covariance matrix, is

$$\underset{z,z}{A_2} = \hat{\Pi}_{12}^T \left(\Omega^{-1} \otimes R_{22}^{-1}\right) \hat{\Pi}_{12}$$

where its explicit form is the following

$$A_2 = \begin{bmatrix} \hat{\Pi}_{12}^{1T}\sigma^{11}R_{2211}^{-1}\hat{\Pi}_{12}^{1} & \cdots & \hat{\Pi}_{12}^{1T}\sigma^{1i}R_{221i}^{-1}\hat{\Pi}_{12}^{i} & \cdots & \hat{\Pi}_{12}^{1T}\sigma^{1m}R_{221m}^{-1}\hat{\Pi}_{12}^{m} \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \hat{\Pi}_{12}^{iT}\sigma^{i1}R_{22i1}^{-1}\hat{\Pi}_{12}^{1} & \cdots & \hat{\Pi}_{12}^{iT}\sigma^{ii}R_{22ii}^{-1}\hat{\Pi}_{12}^{i} & \cdots & \hat{\Pi}_{12}^{iT}\sigma^{im}R_{22im}^{-1}\hat{\Pi}_{12}^{m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \hat{\Pi}_{12}^{mT}\sigma^{m1}R_{22m1}^{-1}\hat{\Pi}_{12}^{1} & \cdots & \hat{\Pi}_{12}^{mT}\sigma^{mi}R_{22mi}^{-1}\hat{\Pi}_{12}^{i} & \cdots & \hat{\Pi}_{12}^{mT}\sigma^{mm}R_{22mm}^{-1}\hat{\Pi}_{12}^{m} \end{bmatrix}$$

where R_{22ij}^{-1}comes from

$$\underset{k,k}{(X_i^T X_j)^{-1}} = \begin{bmatrix} \underset{k_{1i},k_{1j}}{R_{11ij}} & \underset{k_{1i},k_{2j}}{R_{12ij}} \\ \underset{k_{2i},k_{1j}}{R_{12ij}} & \underset{k_{2i},k_{2j}}{R_{22ij}} \end{bmatrix}$$

remembering that the column of X_i and X_j it's the same the only difference between this two matrix it's the ordered of this column, that depends on what it is the exogenous variables included in and excluded from the $i-th$ and $j-th$ equations.

In the next chapter we will see the estimation's procedure of structural parameters but, before to do this, it will be introduced the singular value decomposition, to have a complete vision of the new FI LODE.

Chapter 5

LODE based on SVD

5.1 Introduction

As it has been showed in the simulation experiment FI LODE, after the computational procedure of estimates' correction, gave good result respect to classical estimator in term of both bias and RMSE, but as we said this procedure is just an estimator's empirical correction.

The aim was to find a more formal solution to the problem of far outliers in FI LODE estimator.

It was found out that instability came from the algorithm used to calculate eigenvectors and eigenvalues, principally the problem came from the difficulty of traditional algorithm to calculate characteristic polynomial in high-dimension matrix.

After the results presented in the previus chapter, we are looking for finding characteristics vectors and characteristics roots of square matrix $\hat{\Pi}_{12}^{iT} R_{22ii}^{-1} \hat{\Pi}_{12}^{i}$ for limited information and $\hat{\Pi}_{12}^{T} R_{22}^{-1} \hat{\Pi}_{12}$ for full information, to do this it has been used an algorithm based on Singular Value Decomposition that is numerically more robust respect to spectral decomposition where robustness means the greatest algorithm's probability to converge (Markovsky and Van Huffel, 2004).

5.2 Singolar Value Decomposition

Theorem 5.1 *if* $C\epsilon\mathbb{R}^{k\times m}$ *then exist orthonormal matrices* $U = [U_1, ..., U_k]\epsilon\mathbb{R}^{k\times k}$ *and* $V = [V_1, ..., V_m]\epsilon\mathbb{R}^{m\times m}$ *such that*

$$U^T CV = \Lambda = diag(\lambda_1, ..., \lambda_p) \tag{5.1}$$

with

$$p = min\{k, m\}$$

Proof. for the proof see (Golub and Van Loan, 1981)

The λ_i are the *singular values* of C and their set is called the *singular value spectrum.* The vector U_i and V_i are the $i-th\ left$ *singular vector* and the $i-th$ *right singular vector*, respectively. The triplet (U_i, λ_i, V_i) is called a *singular triplet.* Futhermore comparing the columns in the equations $CV = U\Lambda$ and $C^T U = \Lambda^T V$ we have

$$CV_i = \lambda_i U_i, \quad C^T U_i = \lambda_i V_i \quad i = 1, ..., p \tag{5.2}$$

The SVD reveals a great deal about the structure of a matrix. If the SVD of C is given by Theorem 5.1 and we define r by

$$\lambda_1 \geq \cdots \geq \lambda_r > \lambda_{r+1} = \cdots = \lambda_p = 0$$

then

$$rank(C) = r$$

$$R(C) = R([U_1, ..., U_r])$$

$$N(C) = R([V_{r+1}, ..., V_n])$$

where $R(C)$represented the column space of C and $N(C)$is the null space of C

Moreover, if $U_R = [U_1, ..., U_r]$, $\Lambda_R = diag(\lambda_1, ..., \lambda_r)$ and $V_R = [V_1, ..., V_r]$ then we have the SVD exspansion

$$C = U_R \Lambda_R V_R^T = \sum_{i=1}^{r} \lambda_i U_i V_i \tag{5.3}$$

this equation called *dyadic decomposition* decomposes the matrix C of rank r in a sum of r matrix of rank 1. Also the *Frobenius norm* is neatly characterized in terms of SVD:

$$||C||^2 = \sum_{i=1}^{m} \sum_{j=1}^{k} c_{ij}^2 = \lambda_1^2 + ... + \lambda_p^2.$$

with $p = min\{m, k\}$

Finally from (5.1) it follows that

$$C^T C = V \Lambda^T \Lambda V^T \tag{5.4}$$

and

$$CC^T = V \Lambda \Lambda^T V^T$$

Thus λ_i^2, $i = 1, ..., p$ are the eigenvalues of the symmetric and nonnegative definite matrices $\underset{m,m}{C^T C}$ and $\underset{k,k}{CC^T}$ with p equal respectively to m or k and V_i and U_i are the corrisponding eigenvectors.

The SVD plays an important role in matrix approximation problems. In the theorem below we consider the approximation of one matrix by another of lower rank.

Theorem 5.2 *Let the* SVD *of* $C \epsilon \mathbb{R}^{k \times m}$ *be given by* $C = \sum_{i=1}^{r} \sigma_i U_i V_i^T$ *with* $r = rank(C)$.

If $k < r$ *and* $C_k = \sum_{i=1}^{k} \sigma_i U_i V_i^T$ *then*

$$min||C - C_k||_F = \sqrt{\sum_{i=k+1}^{p} \sigma_i^2}$$

with

$$p = min\{m, n\}$$

Proof Eckart and Young (1936).

5.3 LODE LI based on SVD

What we are interested in is the case where we have $k > m$, considering $k \times m$ the dimesion of C, now if we take the square matrix $C^T C$ we will have

$$C^T C = V \Lambda^T \Lambda V^T$$

where :

- V is the eigenvectors matrix of $C^T C$ and is equal to the matrix of right singular vector of C
- the matrix $\Lambda^T \Lambda$ is the diagonal matrix of $C^T C$'s eigenvalues and its elements λ_i^2 are equal to the square of singular values of C

Recalling the square matrix, from where we have to calculate eigenvector and eigenvalues to obtain the LI LODE estimates (section 3.3)

$$\hat{\Pi}_{12}^{Ti} R_{22ii}^{-1} \hat{\Pi}_{12}^{i} \tag{5.5}$$

now to obtain the LI LODE based on SVD let

$$C = \underset{k_{2i}, m_{1i}}{Q_{22i} \hat{\Pi}_{12}^{i}} \tag{5.6}$$

with

$$k_{2i} > m_{1i}$$

rembering that

$$Q_{22i} Q_{22i}^T = R_{22ii}^{-1}$$

like was shown in section 3.2(), hence:

- the eigenvector $P_{m_{1i}}$ that corresponds to the smallest eigenvalue $\lambda_{m_{1i}}^2$ of matrix $C^T C$ for which $P C^T C P^T = min$

- is equal to the right singular vector $V_{m_{1i}}$that corresponds to the smallest singular value $\lambda_{m_{1i}}$ of matrix C.

Since $V_{m_{1i}} = P_{m_{1i}}$ and the estimation of structural parameters of second equation of identification system for $i-th$ equation

$$Q_{22i}\hat{\Pi}^i_{12}\Gamma_{1i} = Q_{22i}\varepsilon_{2i} \tag{5.7}$$

is obtain applying the Singular value Decomposition on 5.6.

Indeed if $\lambda_{m_{1i}} \neq 0$, the rank of 5.6 is m_{1i} and this is a full rank matrix and the equation 5.7 is upper identified. To obtain a solution, the rank of 5.6 must be reduced to $m_{1i} - 1$.

We are looking for finding an approximation of $Q_{22i}\hat{\Pi}^i_{12}$ and the connected vector $V_{m_{1i}}$(the pedix of V will be clear soon) for which this equation is true

$$Q_{22i}\hat{\Pi}^i_{12}V_{m_{1i}} = 0$$

Using the theorem 4.2 we know that the best rank $m_{1i}-1$ approximation $(Q_{22i}\hat{\Pi}^i_{12i})'$of $Q_{22i}\hat{\Pi}^i_{12i}$, is given by

$$(Q_{22i}\hat{\Pi}_{12i})' = U\Lambda'V$$

with

$$\Lambda' = diag(\lambda_1, ..., \lambda_{m_{1i}-1}, 0)$$

the minimal correction is then

$$\underset{Q_{22i}\hat{\Pi}^i_{12i}}{min} \left\| Q_{22i}\hat{\Pi}^i_{12i} - (Q_{22i}\hat{\Pi}^i_{12i})' \right\|^2_F = \lambda_{m1i}$$

and the solution is given by the only vector $V_{m_{1i}}$that belongs to $N(Q_{22i}\hat{\Pi}^i_{12i})'$(the null space of the approximation matrix) so the last right singular vector of $Q_{22i}\hat{\Pi}^i_{12i}$.

Using now the normalization rule for $i-th$ structural equation, the estimate of structural parameter Γ_{1i} are defined as

$$\hat{\Gamma}_{1i} = -\frac{1}{v_{0i}} V_{m_{1i}}$$

where v_{0i} is the element of the characteristic vector associated with the right hand side endogenous variable in the $i-th$ structural equation.

So given that

$$(Q_{22i}\hat{\Pi}^{i}_{12i})' V_{m_{1i}} = 0$$

$(Q_{22i}\hat{\Pi}^{i}_{12i})'$ rappresents the $(m_{1i}-1)$ dimensional subspace spanned by the first $(m_{1i}-1)$ principal axis that minimize the sum of squares of the orthogonal distance between the obseved points and the subspace itself.

Observe that, like we have said at the end of section 1.3 for LODE LI in the same way here, if the $i-th$ equation is exaclty identified equation the last singolar value $\lambda_{m1i} = 0$ and that equation will have a unique solution that coincides with ILS estimator.

On the contrary when $i-th$ equation is under identified the singular value equal to zero will have multiplicity grater than one and the system will have infinite solutions.

It is important noice that, after some simulation experiment, the result of estimation given by LODE LI primal version (section 3.3) using spectral decomposition or singular value decomposition is exactly the same, this probably because the square matrix 5.5 it's no so big to create problem in calculating characteristic polynomial.

5.4 FI LODE based on SVD

As seen in the last section of previous chapter the LODE FI estimate of

$$\underset{z,1}{\Gamma_1} = \begin{bmatrix} \underset{m_{11},1}{\Gamma_{11}} \\ \underset{m_{12},1}{\Gamma_{12}} \\ \vdots \\ \underset{m_{1m},1}{\Gamma_{1m}} \end{bmatrix}$$

comes from the minimization of this quadratic form

$$\Gamma_1^T \hat{\Pi}_{12}^T \left(\Omega^{-1} \otimes R_{22}^{-1}\right) \hat{\Pi}_{12} \Gamma_1$$

indeed with LODE FI based on SVD we work with this matrix

$$\underset{r,z}{Q_2 \hat{\Pi}_{12}} \tag{5.8}$$

$$z = \sum_{i=1}^{m} m_{1i}$$

$$r = \sum_{i=1}^{m} k_{2i}$$

with

$$Q_2 Q_2^T = \left(\Omega^{-1} \otimes R_{22}^{-1}\right).$$

Given the second equation of identification system for the whole system

$$Q_2 \hat{\Pi}_{12} \Gamma_1 = Q_2 \varepsilon_2$$

what we have to do is to find the $z-1$ subspace of 5.8 that minimize the sum of squares of the orthogonal distance between the obseved points and the subspace itself.

As with limited infromation LODE, with the theorem 4.2 we can prove that the best rank $z-1$ matrix approximation of $Q_2 \hat{\Pi}_{12}$, is the matrix that is equal to

$$(Q_2 \hat{\Pi}_{12})' = T \Xi' V$$

where the matrix T and V are respectivelly the matrix of left and right singular vector of $Q_2 \hat{\Pi}_{12}$ and the matrix Ξ' is equal to the diagonal matrix of singular value Ξ of $Q_2 \hat{\Pi}_{12}$except for the element z that is zero

$$\Xi = diag(\xi_1, ..., \xi_{z-1}, \xi_z)$$

$$\Xi' = diag(\xi_1, ..., \xi_{z-1}, 0)$$

the minimal corrections then is

$$\underset{Q_2\hat{\Pi}_{12}}{min} \left\| Q_2\hat{\Pi}_{12} - (Q_2\hat{\Pi}_{12})' \right\|_F^2 = \xi_z$$

and this equation is true

$$(Q_2\hat{\Pi}_{12})'V_z = 0$$

and so the solution is given by the right singular vector V_z of $Q_2\hat{\Pi}_{12}$ that corresponds to the last singular value ξ_z of Ξ and that is equal to the last eigenvector $\hat{\Pi}_{12}^T Q_2^T Q_2 \hat{\Pi}_{12}$.

Now Defining W as the block diagonal matrix

$$W = \begin{bmatrix} w_1 I_{m_{1i}} & & \\ & \ddots & \\ & & w_1 I_{m_{1i}} \end{bmatrix}$$

in which $c_i$are defined as follows

$$w_i = -\frac{1}{v_{0i}}$$

with v_{0i} being the right singular vector's element corresponding to the endogenous variable y_{0i} chosen to be at left hand side in $i - th$ structural equation.

The FI estimator is then

$$\hat{\Gamma}_1 = WV_z \tag{5.9}$$

equal

$$\underset{mn,1}{Y_0} = \begin{bmatrix} \underset{n,1}{Y_{01}} \\ \vdots \\ \underset{n,1}{Y_{0i}} \\ \vdots \\ \underset{n,1}{Y_{0m}} \end{bmatrix}$$

where Y_{0i} is the dependent endogenous variable of $i-th$ equation, and let

$$\underset{n,z}{Y_{1e}} = \begin{bmatrix} \underset{n,m_1}{Y_{11e}} & & 0 \\ & \ddots & \\ 0 & & \underset{n,m_{mi}}{Y_{1me}} \end{bmatrix}$$

that is the block diagonal matrix of endogenous explanatory variable of each system equation, and finally let

$$\underset{n,h}{X_1} = \begin{bmatrix} \underset{n,k_{11}}{X_{11}} & & 0 \\ & \ddots & \\ 0 & & \underset{n,k_{1m}}{X_{1m}} \end{bmatrix}$$

with

$$h = \sum_{i=1}^{m} k_{1i}$$

that is the block diagonal matrix of exogenous variable of each system equation, then to have the estimation of parameters B_1 we have to apply the OLS on this equation

$$Y_0 - Y_{1e}\hat{\Gamma}_{1e} = X_1 B_1 + E$$

where E

$$E = \begin{bmatrix} E_1 \\ \vdots \\ E_i \\ \vdots \\ E_m \end{bmatrix}$$

is the vector of the new error component, hence the $\hat{B}_1$ will be

$$\hat{B}_1 = (X_1^T X_1)^{-1} X_1^T (Y_0 - Y_{1e}\hat{\Gamma}_{1e})$$

where the matrix $\Omega^{-1} \otimes \left(X^T X\right)$is the inverse of Variance-coraviance matrix

Chapter 6

New Simulation Experiment

6.1 Introduction

In this chapter is presented the second simulation experiment conduct to evaluate the LODE FI based on SVD respect to the FIML, 3SLS and what we have called LODE FI with empirical correction.

The design experiment and the simulated data are the same of the simulation experiment, chapter 2, to have a more easy comparison with the previous simulation experiment.

Hence the model is a three equation model and the parameters of endogenous and esogenous variables to estimate are(), the errors are been generate from a standardized Normal distribution and from a Uniform distribution in the interval $\left[-\sqrt{3}, \sqrt{3}\right]$ and so the simulatio scenarios repeted for the two different error distribution is

Table 6.1: Simulation Scenarios

	S_i		
ρ_{ij}	0.2-0.25	0.4-0.5	0.75-0.8
0.1-0.2	N=20 N=30 N=100	N=20 N=30 N=100	N=20 N=30 N=100
0.4-0.5	N=20 N=30 N=100	N=20 N=30 N=100	N=20 N=30 N=100
0.8-0.9	N=20 N=30 N=100	N=20 N=30 N=100	N=20 N=30 N=100

In section 1.2 the simulation result of LODE FI with empirical correction is compared with LODE FI based on SVD with error component normally and uniformally distributed. The performance of all the estimator divided by results when the error component is Normal and when it is Uniform are presented (§ 1.3).

6.2 LODE FI with empirical correction and LODE FI based on SVD

Like in chapter 2 to synthesize results of the simulation experiment, the table are based on the percentage of times in which parameters' estimators present the lowest bias or variability.

Remembering that with bias we mean

$$\varphi = \frac{\left(\hat{\theta} - \theta\right)}{\theta}$$

(i.e. the bias divided by the fixed initial parameter value) where $\hat{\theta}$ is the average of estimated parameter over the 500 samples and θ is one of the γ or β parameters.

For variability

$$\psi = \frac{RMSE}{\theta}$$

where RMSE is the Root Mean Square Error of $\hat{\theta}$ which is divided by the initial parameter value.

6.2.1 Normal error component

First let us consider the case in which the error component is distributed according to a Multivariate Normal.

In terms of bias the LODE based on SVD mostly show the best results, the only situation where LODE with empirical correction give a lower bias than the other LODE is when the sample size is 20 and S_i is between 0.75 and 0.8 and when sample size si 30 and S_i is 0.2-0.25 (6.2) for all the other combination LODE SVD gives the highest percentage as matter of the fact 19 times on 27 scenarios it works better than LODE with empirical correction.

Table 6.2: Relative frequency distribution of FI LODE SVD, FI LODE presenting a lower bias gruoped by S_i, ρ_{ij} and sample size - Normal error component

		ρ_{ij}					
		0.1-0.2		0.4-0.5		0.8-0.9	
S_i	n	LODE SVD	LODE	LODE SVD	LODE	LODE SVD	LODE
0.2-0.25		80	20	60	40	100	0
0.4-0.5	20	80	20	60	40	80	20
0.75-0.8		0	100	20	80	20	80
0.2-0.25		0	100	20	80	20	80
0.4-0.5	30	60	40	80	20	60	40
0.75-0.8		60	40	40	60	60	40
0.2-0.25		40	60	100	0	80	20
0.4-0.5	100	100	0	60	40	100	0
0.75-0.8		80	20	100	0	100	0

In terms of RMSE too LODE SVD generally seems to go better than the other one, here too 19 times on 27 LODE SVD have the lowest variability, but the LODE with empirical correction show more frequently lower values with low sample size 20 and 30 and high correlation coefficent ρ_{ij} 0.8-0.9(6.3).

It's not a chance say that the LODE SVD works generally better than LODE with empirical correction with normal error distribution even if we

consider only small samples where generally LODE with correction works better but not as LODE SVD.

Table 6.3: Relative frequency distribution of FI LODE SVD, FI LODE presenting a lower RMSE gruoped by S_i, ρ_{ij} and sample size - Normal error component

		ρ_{ij}					
		0.1-0.2		0.4-0.5		0.8-0.9	
S_i	n	LODE SVD	LODE	LODE SVD	LODE	LODE SVD	LODE
0.2-0.25		86.67	13.33	73.33	26.67	73.33	26.67
0.4-0.5	20	26.67	73.33	73.33	26.67	0.00	100.00
0.75-0.8		46.67	53.33	0.00	100.00	0.00	100.00
0.2-0.25		86.67	13.33	80.00	20.00	53.33	46.67
0.4-0.5	30	93.33	6.67	33.33	66.67	73.33	26.67
0.75-0.8		86.67	13.33	46.67	53.33	33.33	66.67
0.2-0.25		73.33	26.67	86.67	13.33	86.67	13.33
0.4-0.5	100	86.67	13.33	93.33	6.67	53.33	46.67
0.75-0.8		80.00	20.00	93.33	6.67	93.33	6.67

6.2.2 Uniform error component

Looking to the results of this simulation where the errors components are uniformally distributed between $(-\sqrt{3}, \sqrt{3})$, the comparison between the two LODE goes in the same way of simulation with normally distributed component, this means that LODE FI SVD outperforms LODE with empirical correction in terms of bias and RSME, with this error distribution LODE with empirical correction works better only with low value of ρ_{ij} 0.1-0.2 and 0.4-0.5 and sample size equal to 20.

In Terms of lowest bias LODE SVD presents the best result 21 times on 27 (6.6)

Table 6.4: Relative frequency distribution of FI LODE SVD, FI LODE presenting a lower bias gruoped by S_i, ρ_{ij} and sample size - Uniform error component

		ρ_{ij}					
		0.1-0.2		0.4-0.5		0.8-0.9	
S_i	n	LODE SVD	LODE	LODE SVD	LODE	LODE SVD	LODE
0.2-0.25		20	80	80	20	80	20
0.4-0.5	20	20	80	0	100	100	0
0.75-0.8		40	60	0	100	80	20
0.2-0.25		100	0	80	20	100	0
0.4-0.5	30	80	20	80	20	100	0
0.75-0.8		60	40	100	0	80	20
0.2-0.25		80	20	100	0	80	20
0.4-0.5	100	40	60	80	20	80	20
0.75-0.8		80	20	80	20	80	20

If we consider the variability of the estimate the things seems to go a little bit better for LODE with correction which however have just 11 more good results than LODE SVD so other 16 combination latter works better.

Hence we can conclude that this new formulation of LODE based on SVD improved the performance of LODE estimator

Table 6.5: Relative frequency distribution of FI LODE SVD, FI LODE presenting a lower RMSE gruoped by S_i, ρ_{ij} and sample size - Uniform error component

		ρ_{ij}					
		0.1-0.2		0.4-0.5		0.8-0.9	
S_i	n	LODE SVD	LODE	LODE SVD	LODE	LODE SVD	LODE
0.2-0.25		13.33	86.67	60.00	40.00	33.33	66.67
0.4-0.5	20	40.00	60.00	40.00	60.00	86.67	13.33
0.75-0.8		26.67	73.33	40.00	60.00	26.67	73.33
0.2-0.25		60.00	40.00	73.33	26.67	60.00	40.00
0.4-0.5	30	73.33	26.67	93.33	6.67	60.00	40.00
0.75-0.8		60.00	40.00	40.00	60.00	20.00	80.00
0.2-0.25		53.33	46.67	40.00	60.00	66.67	33.33
0.4-0.5	100	73.33	26.67	86.67	13.33	80.00	20.00
0.75-0.8		86.67	13.33	80.00	20.00	33.33	66.67

Only in the simulation experiment where the errors are generating from Uniform $(-10, 10)$ the LODE with empirical correction gives better results than LODE SVD, expecially in terms of RMSE where in all the scenarios the percentage attributed to LODE with empirical correction is higher than the one of LODE SVD

Table 6.6: Relative frequency distribution of FI LODE SVD, FI LODE presenting a lower bias gruoped by S_i, ρ_{ij} and sample size - Unif(-10,10) error component

		ρ_{ij}					
		0.1-0.2		0.4-0.5		0.8-0.9	
S_i	n	LODE SVD	LODE	LODE SVD	LODE	LODE SVD	LODE
0.2-0.25		0	100	0	100	40	60
0.4-0.5	20	20	80	20	80	0	100
0.75-0.8		60	40	20	80	0	100
0.2-0.25		80	20	20	80	40	60
0.4-0.5	30	20	80	40	60	60	40
0.75-0.8		0	100	20	80	60	40
0.2-0.25		0	100	100	0	40	60
0.4-0.5	100	20	80	40	60	40	60
0.75-0.8		20	80	60	40	0	100

Table 6.7: Relative frequency distribution of FI LODE SVD, FI LODE presenting a lower RMSE gruoped by S_i, ρ_{ij} and sample size - Unif(-10,10) error component

		ρ_{ij}					
		0.1-0.2		0.4-0.5		0.8-0.9	
S_i	n	LODE SVD	LODE	LODE SVD	LODE	LODE SVD	LODE
0.2-0.25		0	100	7	93	7	93
0.4-0.5	20	0	100	0	100	0	100
0.75-0.8		0	100	7	93	0	100
0.2-0.25		0	100	0	100	0	100
0.4-0.5	30	0	100	0	100	27	73
0.75-0.8		0	100	7	93	0	100
0.2-0.25		47	53	27	73	13	87
0.4-0.5	100	13	87	13	87	27	73
0.75-0.8		0	100	20	80	7	93

6.3 General results of the experiment

Given the best results of LODE based on SVD with the simulated model with Normal and Uniform $(-\sqrt{3}, \sqrt{3})$ distribution of the errors, in this section we will compare the features of this implemantation of LODE estimators with the two classical estimators 3SLS and FIML of simultaneous equation model.

6.3.1 Normal error component

In the case of normal error component now, unlike to the first simulation experiment, the best estimator in terms of bias it's no more FIML but LODE

based on SVD for more and less all the scenarios, it presents 20 times 27 the high percentage, like it's easy to see in (6.8), notice that LODE based on SVD is indicated with LODE.

This good performances becomes more evident if we see only on small samples, n=20, in this case 8 times on 9 the LODE SVD has a lowest bias and LODE seems to works very well in terms of bias with equation with high correlation coefficent, ρ_{ij} between $[0.8 - 0.9]$, here too 8 times on 9 LODE have highest percentage values.

Table 6.8: Relative frequency distribution of FI LODE SVD, 3SLS and FIML presenting a lower bias gruoped by S_i, ρ_{ij} and sample size - Normal error component

		ρ_{ij}								
		0.1-0.2			0.4-0.5			0.8-0.9		
S_i	n	LODE	3SLS	FIML	LODE	3SLS	FIML	LODE	3SLS	FIML
0.2-0.25		80	0	20	100	0	0	60	20	20
0.4-0.5	20	100	0	0	80	20	0	100	0	0
0.75-0.8		80	0	20	20	80	0	80	20	0
0.2-0.25		80	0	20	40	0	60	60	0	40
0.4-0.5	30	60	0	40	40	0	60	80	20	0
0.75-0.8		20	0	80	100	0	0	100	0	0
0.2-0.25		20	0	80	60	0	40	60	0	40
0.4-0.5	100	80	0	20	60	0	40	20	0	80
0.75-0.8		20	0	80	80	0	20	80	0	20

While in terms of RMSE, FIML remain the estimator that show more lower values than LODE SVD and 3SLS (6.9).

LODE SVD seems to work better with n=20 and n= 30 scenarios so remain the features of LODE to have good performance when dealing with low number of observation.

Table 6.9: Relative frequency distribution of FI LODE SVD, 3SLS and FIML presenting a lower RMSE gruoped by S_i, ρ_{ij} and sample size - Normal error component

		ρ_{ij}								
		0.1-0.2			0.4-0.5			0.8-0.9		
S_i	n	LODE	3SLS	FIML	LODE	3SLS	FIML	LODE	3SLS	FIML
0.2-0.25		40	27	33	7	27	67	40	20	40
0.4-0.5	20	13	40	47	40	20	40	0	40	60
0.75-0.8		7	60	33	0	27	73	0	33	67
0.2-0.25		27	13	60	7	7	87	13	7	80
0.4-0.5	30	33	53	13	13	27	60	27	40	33
0.75-0.8		33	40	27	27	40	33	13	33	53
0.2-0.25		0	7	93	0	20	80	0	7	93
0.4-0.5	100	13	13	73	13	13	73	0	13	87
0.75-0.8		7	13	80	7	20	73	0	20	80

6.3.2 Uniform component

Using the uniform distribution for the error component in the simulation experiment the good results just presented doesn't change, the LODE based on SVD is the best estimator in terms of bias together with FIML, it present 14 times the lower bias against the 13 times of FIML.(6.10).

Particularly the LODE SVD results go better with the increased of correlation coefficent.

Table 6.10: Relative frequency distribution of FI LODE SVD, 3SLS and FIML presenting a lower bias gruoped by S_i, ρ_{ij} and sample size - Uniform error component

		ρ_{ij}								
		0.1-0.2			0.4-0.5			0.8-0.9		
S_i	n	LODE	3SLS	FIML	LODE	3SLS	FIML	LODE	3SLS	FIML
0.2-0.25		0	0	100	40	0	60	80	0	20
0.4-0.5	20	20	20	60	0	20	80	100	0	0
0.75-0.8		20	0	80	0	20	80	40	20	40
0.2-0.25		60	0	40	40	0	60	60	0	40
0.4-0.5	30	20	0	80	80	0	20	20	20	60
0.75-0.8		80	0	20	80	20	0	60	0	40
0.2-0.25		40	0	60	40	0	60	80	0	20
0.4-0.5	100	0	0	100	60	0	40	0	0	100
0.75-0.8		80	0	20	60	0	40	60	0	40

even considering the RMSE the conclusions are the same done with normal distribution FIML have the lowest RMSE with respect to the other estimator (6.11)

Table 6.11: Relative frequency distribution of FI LODE SVD, 3SLS and FIML presenting a lower RMSE gruoped by S_i, ρ_{ij} and sample size - Uniform error component

		ρ_{ij}								
		0.1-0.2			0.4-0.5			0.8-0.9		
S_i	n	LODE	3SLS	FIML	LODE	3SLS	FIML	LODE	3SLS	FIML
0.2-0.25		0	33	67	30	20	50	7	20	73
0.4-0.5	20	13	80	7	0	27	73	0	13	87
0.75-0.8		0	67	33	0	53	47	0	27	73
0.2-0.25		26	13	60	7	27	67	20	13	67
0.4-0.5	30	20	27	53	20	20	60	0	27	73
0.75-0.8		13	60	27	0	33	67	20	13	67
0.2-0.25		40	13	47	0	27	73	7	20	73
0.4-0.5	100	7	13	80	7	20	73	0	13	87
0.75-0.8		0	13	87	0	20	80	7	7	87

6.3.3 Variability in LODE based on SVD and 3SLS

Given the fact that the best estmator in terms of variability results the FIML and given too that sometimes this estimator could have computational problem, this sub-secton is dedicated to compare 3SLS and LODE SVD in terms of RMSE.

As it is easy to see in the next table LODE estimator presents greater frequencies of estimates with lower RMSE than 3SLS in most of analyzed scenarios, in 6.12 there is the comparison in terms of RMSE for the simulation with normal error components

Table 6.12: Relative frequency distribution of FI LODE SVD, 3SLS presenting a lower RMSE gruoped by S_i, ρ_{ij} and sample size - Normal error component

		ρ_{ij}					
		0.1-0.2		0.4-0.5		0.8-0.9	
S_i	n	LODE SVD	3SLS	LODE SVD	3SLS	LODE SVD	3SLS
0.2-0.25		67	33	60	40	80	20
0.4-0.5	20	20	80	60	40	0	100
0.75-0.8		27	73	7	93	0	100
0.2-0.25		80	20	67	33	80	20
0.4-0.5	30	47	53	33	67	47	53
0.75-0.8		60	40	33	67	13	87
0.2-0.25		80	20	67	33	80	20
0.4-0.5	100	87	13	87	13	67	33
0.75-0.8		87	13	73	27	80	20

The simulation with error uniformally distributed gives the same results

15 times on 27 the percentage link to LODE SVD is higher than the percentage of 3SLS

Hence in terms of variability too the LODE SVD works better of 3SLS

Table 6.13: Relative frequency distribution of FI LODE SVD, 3SLS presenting a lower RMSE gruoped by S_i, ρ_{ij} and sample size - Normal error component

		ρ_{ij}					
		0.1-0.2		0.4-0.5		0.8-0.9	
S_i	n	LODE SVD	3SLS	LODE SVD	3SLS	LODE SVD	3SLS
0.2-0.25		13	87	60	40	73	27
0.4-0.5	20	13	87	40	60	73	27
0.75-0.8		13	87	20	80	27	73
0.2-0.25		80	20	60	40	40	60
0.4-0.5	30	67	33	60	40	53	47
0.75-0.8		27	73	27	73	27	73
0.2-0.25		87	13	47	53	67	33
0.4-0.5	100	80	20	73	27	60	40
0.75-0.8		80	20	67	33	33	67

Chapter 7

Conclusion

The aim of this work was to find a solution to reduced the high variability shows from FI LODE, previously was introduced a computational, to overcome this problem, based on the choice of the first m sub-eigenvectors that minimize the trace of the residual errors' variance-covariance matrix with this LODE FI implementation was conducted first simulation experiment considering different scenarios according to error components' variance and correlation and for each scenario three sample sizes (20, 30 and 100) have been considered, furthermore two distributional hypotheses about disturbances, Normal and Uniform, have been introduced.

The results of the experiment have not highlighted strong differences between the performances of the three methods as far as the distribution of error component is concerned. Both $N(1, 0)$ and $U(-\sqrt{3}, \sqrt{3})$ give almost the same results.

A hypothesis of a greater variance Uniform distribution has been then introduced for the generation of error components, namely a Uniform distribution in the interval $(-10, 10)$. With respect to this last situation a very strong difference among estimation methods has been observed: LODE presents always bias very much lower than FIML; also in comparison of 3SLS the LODE bias is lower.

The most interesting result of the study is the very good performance of LODE in small samples.

This outcomes plus the fact that LODE estimator does not impose a prior choice of dependent variable in system's equation, furthermore one of the principal characteristic of this method is that the estimator does not depend on the distribution of the error component, gave the impetus to continue the work with this estimator and to look for finding a more formal solution for LODE FI variability. The result of this research is illustrated in chapter 5 and, as we said, it comes from two different changes applied on FI LODE.

The first one was to come back to the primal version of LODE LI and this version was readapted on the full information case, the second one is to replace the spectral decomposition, used to calculate eigenvalues and eigenvectors of matrix $\hat{\Pi}_{12}^T Q_2^T Q_2 \hat{\Pi}_{12}$, with the Singular Value Decomposition apply on $Q_2 \hat{\Pi}_{12}$, knowing that the right singular vectors of this matrix accordig to the theory match the eigenvectors of the previous matrix, but the algorithm based on this decomposition is numerically more robust.

After this the Montecarlo experiment was replaced with the new FI LODE based on SVD, in the first part comparing this one with the FI LODE with empirical correction and LODE FI based on SVD presents for all simulation scenarios and with error distributed as $N(0,1)$and $U(-\sqrt{3},\sqrt{3})$ lower bias and RMSE respect to other LODE FI, while the LODE FI with empirical correction works better if the distribution of accidental component is a Uniform between $(-10,10)$ in term bias and RMSE, in the second part of the simulation was evaluated the LODE FI based on SVD respect to the FIML and 3SLS for $N(0,1)$ and $U(-\sqrt{3},\sqrt{3})$ here too for mostly of simulation scenarios and both the error distribution LODE SVD have the lowest bias, while in terms of RMSE the best results come from FIML estimation but the LODE perfoms better than 3SLS in terms of variability, knowing that usually for computational problem the 3SLS is favorited to FIML are evident the very good results of this simulation.

What Have to be done, the first thing is to find the distribution of LODE FI estimate to be able to create a test, some analytical solution can come from Gleser(1981), Gleser proof that a similar estimator based on orthogonal distance estimator applied on multivariate errors in variables model have a multinormal distribution of its estimates.

Need to create a package on R and so leave E-Views for an open source software, furthermore class of LODE estimator have to be tested with a real data application.

Appendix A

Appendix

```
'CREATION OF SIMULATION DATA

series X1 = 1
series X2 = @runif(10, 20)
series X3 = @runif(15, 27)
series X4 = @runif(3, 12)
series X5 = @runif(3, 7)
series X6 = @runif(20, 50)
series X7 = @runif(7, 13)
group groupx X1 X2 X3 X4 X5 X6 X7
matrix x = @convert(groupx)
matrix (7, 3) b
b.fill 44, 0.74, 0, 0, 0.13, 0, 0, 62, 0, 0.7, 0, 0.96, 0, 0.06, 40, 0, 0.53, 0.11, 0, 0.56, 0
vector (3, 1) b1
b1.fill 44, 0.74, 0.13
vector (4, 1) b2
b2.fill 62, 0.7, 0.96, 0.06
vector (4, 1) b3
b3.fill 40, 0.53, 0.11, 0.56
matrix (3, 3) g
g.fill -1, 0.89, 0.16, 0.74, -1, 0, 0, 0.29, -1
vector (3, 1) g1
g1.fill 1, 0.89, 0.16
```

```
vector (2, 1) g2
g2.fill 1, 0.74
vector (2, 1) g3
g3.fill 1, 0.29
matrix invg = @inverse(g)
matrix p = - b*invg
matrix y = x *p
vector Y1 = @columnextract (y,1)
vector Y2 = @columnextract (y,2)
vector Y3 = @columnextract (y,3)
series serY1
series serYb
series serYc
mtos(Y1, serY1)
mtos(Y2, serY2)
mtos(Y3, serY3)
matrix ygamma = Y*g
vector ygamma1 = @columnextract (ygamma,1)
vector ygamma2 = @columnextract (ygamma,2)
vector ygamma3 = @columnextract (ygamma,3)
for !i = 1 to 3
scalar S{!i} = @runif(0.4, 0.5)
scalar ro{!i} = @runif(0.4, 0.5)
scalar vary{!i} = @var(ygamma{!i})*z{!i}
next
scalar covy1y2 = (vary1*vary2)^0.5
scalar covy1y3 = (vary1*varyc)^0.5
scalar covy3y2 = (vary3*vary2)^0.5
scalar covy1y2div = ro1*covy1y2
scalar covy1y3div = ro2*covy1y3
scalar covy3y2div = ro3*covy3y2
for !m = 1 to 3
scalar cas{!m} = @runif(0, 1)
if cas{!m}>0.5 then
cas{!m} = - 1
```

```
else
cas{!m} = 1
endif
next
scalar covy1y2cas = covy1y2div*cas1
scalar covy1y3cas = covy1y3div*cas2
scalar covy3y2cas = covy3y3div*cas3
sym (3, 3) vary
vary.fill vary1, covy1y2cas, covy1y3cas, vary2, covy3y2cas, vary3
sym varv =@transpose(invg)*vary*invg
vector t = @eigenvalues(varv)
matrix p2 = @eigenvectors(varv)
scalar t1 = t(1)^0.5
scalar t2 = t(2)^0.5
scalar t3 = t(3)^0.5
sym (3, 3) tdiag
tdiag.fill t1, 0, 0, t2, 0, t3
matrix o2 = p2*tdiag*@transpose(p2)
for !z = 1 to 500
series vser{!z} = nrnd
series dser{!z} =nrnd
series gser{!z} = nrnd
group groupe{!z} vser{!z} dser{!z} gser{!z}
matrix e{!z} = @convert(groupe{!z})
matrix ea{!z} = e{!z}*o2
matrix yfin{!z} = y + ea{!z}
vector ya{!z} = @columnextract (yfin{!z},1)
vector yb{!z} = @columnextract (yfin{!z},2)
vector yc{!z} = @columnextract (yfin{!z},3)
series serya{!z}
series seryb{!z}
series seryc{!z}
mtos(ya{!z}, serya{!z})
mtos(yb{!z}, seryb{!z})
mtos(yc{!z}, seryc{!z})
```

'CREATION OF INCLUDED AND EXCLUDED VARIABLES GROUP

```
group groupinca{!z} serya{!z} seryb{!z} seryc{!z}
matrix yinca{!z} = @convert(groupinca{!z})
group groupdipa{!z} seryb{!z} seryc{!z}
matrix ydipa{!z} = @convert(groupdipa{!z})
group groupescxa X3 X4 X6 X7
matrix xesca = @convert(groupescxa)
group groupincxa X1 X2 X5
matrix xinca = @convert(groupincxa)
group groupincb{!z} seryb{!z} serya{!z}
matrix yincb{!z} = @convert(groupincb{!z})
group groupincxb X1 X3 X5 X7
matrix xincb = @convert(groupincxb)
group groupescxb X2 X4 X6
matrix xescb = @convert(groupescxb)
group groupincc{!z} seryc{!z} seryb{!z}
matrix yincc{!z} = @convert(groupincc{!z})
group groupincxc X1 X3 X4 X6
matrix xincc = @convert(groupincxc)
group groupescxc X2 X5 X7
matrix xescc = @convert(groupescxc)
sym xsec = @transpose(x)*x
matrix xsecsec =@inverse(xsec)
group groupsainca{!z} seryb{!z} seryc{!z}
matrix sayinca{!z} = @convert(groupsainca{!z})
matrix (20, 5) zuno{!z}
matplace(zuno{!z}, sayinca{!z}, 1, 1)
matplace(zuno{!z}, xinca, 1, 3)
matrix (20, 5) zdue{!z}
matplace(zdue{!z}, ya{!z}, 1, 1)
matplace(zdue{!z}, xincb, 1, 2)
matrix (20, 5) ztre{!z}
matplace(ztre{!z}, yb{!z}, 1, 1)
matplace(ztre{!z}, xincc, 1, 2)
```

```
'GLODE LI

'ESTIMATE FIRST EQUATION

matrix auuu{!z} = @transpose(yinca{!z})*x*xsecsec*@transpose(x)*yinca{!z}
matrix aduu{!z} = @transpose(yinca{!z})*xinca
matrix atuu{!z} = @transpose(xinca)*yinca{!z}
matrix aquu = @transpose(xinca)*xinca
matrix (6, 6) auu{!z}
matplace(auu{!z}, auuu{!z}, 1, 1)
matplace(auu{!z}, aduu{!z}, 1, 4)
matplace(auu{!z}, atuu{!z}, 4, 1)
matplace(auu{!z}, aquu, 4, 4)
sym (6) symauu{!z} = auu{!z}
matrix auuvec{!z} = @eigenvectors(symauu{!z})
matrix vecauu{!z} = -auuvec{!z}/auuvec{!z}(1, 1)
vector vecauufin{!z} = @columnextract(vecauu{!z}, 1)

'ESTIMATE SECOND EQUATION

matrix budd{!z} = @transpose(yincb{!z})*x*xsecsec*@transpose(x)*yincb{!z}
matrix bddd{!z} = @transpose(yincb{!z})*xincb
matrix btdd{!z} = @transpose(xincb)*yincb{!z}
matrix bqdd = @transpose(xincb)*xincb
matrix (6, 6) bdd{!z}
matplace(bdd{!z}, budd{!z}, 1, 1)
matplace(bdd{!z}, bddd{!z}, 1, 3)
matplace(bdd{!z}, btdd{!z}, 3, 1)
matplace(bdd{!z}, bqdd, 3, 3)
sym (6) symbdd{!z} = bdd{!z}
matrix bddvec{!z} = @eigenvectors(symbdd{!z})
matrix vecbdd{!z} = -bddvec{!z}/bddvec{!z}(1, 1)
vector vecbddfin{!z} = @columnextract(vecbdd{!z}, 1)
```

'ESTIMATE THIRD EQUATION

```
matrix cutt{!z} = @transpose(yincc{!z})*x*xsecsec*@transpose(x)*yincc{!z}
matrix cdtt{!z} = @transpose(yincc{!z})*xincc
matrix cttt{!z} = @transpose(xincc)*yincc{!z}
matrix cqtt = @transpose(xincc)*xincc
matrix (6, 6) ctt{!z}
matplace(ctt{!z}, cutt{!z}, 1, 1)
matplace(ctt{!z}, cdtt{!z}, 1, 3)
matplace(ctt{!z}, cttt{!z}, 3, 1)
matplace(ctt{!z}, cqtt, 3, 3)
sym (6) symctt{!z} = ctt{!z}
matrix cttvec{!z} = @eigenvectors(symctt{!z})
matrix vecctt{!z} = -cttvec{!z}/cttvec{!z}(1, 1)
vector veccttfin{!z} = @columnextract(vecctt{!z}, 1)
```

'RESULTS

```
matrix (6, 500) lodealim
colplace( lodealim, vecauufin{!z}, {!z})
matrix (6, 500) lodeblim
colplace( lodeblim, vecbddfin{!z}, {!z})
matrix (6, 500) lodeclim
colplace( lodeclim, veccttfin{!z}, {!z})
```

'ESTIMATE OF Π WITH OLS

```
equation lsa{!z}.ls serya{!z} c X2 X3 X4 X5 X6 X7 @ c X2 X3 X4 X5 X6 X7
equation lsb{!z}.ls seryb{!z} c X2 X3 X4 X5 X6 X7 @ c X2 X3 X4 X5 X6 X7
equation lsc{!z}.ls seryc{!z} c X2 X3 X4 X5 X6 X7 @ c X2 X3 X4 X5 X6 X7
vector coefa{!z} = lsa{!z}.@coefs
vector coefb{!z} = lsb{!z}.@coefs
vector coefc{!z} = lsc{!z}.@coefs
system ls{!z}
ls{!z}.append serya{!z} = c(1) + c(2)*X2 + c(3)*X3 + c(4)*X4 + c(5)*X5 + c(6)*X6
+ c(7)*X7
ls{!z}.append seryb{!z} = c(8) + c(9)*X2 + c(10)*X3 + c(11)*X4 + c(12)*X5 +
c(13)*X6 + c(14)*X7
```

```
ls{!z}.append seryc{!z} = c(15) + c(16)*X2 + c(17)*X3 + c(18)*X4 + c(19)*X5 +
c(20)*X6 + c(21)*X7
ls{!z}.ls
scalar coeflsaza{!z} = ls{!z}.@coefs(1)
scalar coeflsara{!z} = ls{!z}.@coefs(3)
scalar coeflsata{!z} = ls{!z}.@coefs(4)
scalar coeflsaea{!z} = ls{!z}.@coefs(2)
scalar coeflsaua{!z} = ls{!z}.@coefs(5)
scalar coeflsaia{!z} = ls{!z}.@coefs(6)
scalar coeflsaoa{!z} = ls{!z}.@coefs(7)
scalar coeflsbzb{!z} = ls{!z}.@coefs(8)
scalar coeflsbrb{!z} = ls{!z}.@coefs(10)
scalar coeflsbub{!z} = ls{!z}.@coefs(12)
scalar coeflsbob{!z} = ls{!z}.@coefs(14)
scalar coeflsbeb{!z} = ls{!z}.@coefs(9)
scalar coeflsbtb{!z} = ls{!z}.@coefs(11)
scalar coeflsbib{!z} = ls{!z}.@coefs(13)
scalar coeflsczc{!z} = ls{!z}.@coefs(15)
scalar coeflscrc{!z} = ls{!z}.@coefs(17)
scalar coeflscuc{!z} = ls{!z}.@coefs(19)
scalar coeflscoc{!z} = ls{!z}.@coefs(21)
scalar coeflscec{!z} = ls{!z}.@coefs(16)
scalar coeflsctc{!z} = ls{!z}.@coefs(18)
scalar coeflscic{!z} = ls{!z}.@coefs(20)
matrix (7, 3) plsa{!z}
matrix (7, 2) plsb{!z}
matrix (7, 2) plsc{!z}
plsa{!z}.fill coeflsaza{!z}, coeflsaea{!z}, coeflsaua{!z}, coeflsara{!z}, coeflsata{!z},
coeflsaia{!z}, coeflsaoa{!z}, coeflsbzb{!z}, coeflsbeb{!z}, coeflsbub{!z}, coeflsbrb{!z},
coeflsbtb{!z}, coeflsbib{!z}, coeflsbob{!z}, coeflsczc{!z}, coeflscec{!z}, coeflscuc{!z},
coeflscrc{!z}, coeflsctc{!z}, coeflscic{!z}, coeflscoc{!z}
plsb{!z}.fill coeflsbzb{!z}, coeflsbrb{!z}, coeflsbub{!z}, coeflsbob{!z}, coeflsbeb{!z},
coeflsbtb{!z}, coeflsbib{!z}, coeflsaza{!z}, coeflsara{!z}, coeflsaua{!z}, coeflsaoa{!z},
coeflsaea{!z}, coeflsata{!z}, coeflsaia{!z}
```

plsc{!z}.fill coeflsczc{!z}, coeflscrc{!z}, coeflsctc{!z}, coeflscic{!z}, coeflscec{!z}, coeflscuc{!z}, coeflscoc{!z}, coeflsbzb{!z}, coeflsbrb{!z}, coeflsbtb{!z}, coeflsbib{!z}, coeflsbeb{!z}, coeflsbub{!z}, coeflsbob{!z}
matrix (4, 3) zero1
zero1 = 0
matrix (3, 4) zero2
zero2 = 0
matrix i1 = @identity(3)
matrix i2 = @identity(4)

'CREATION of $\hat{\Pi}$

matrix (7, 6) plsatot{!z}
matplace(plsatot{!z}, plsa{!z}, 1, 1)
matplace(plsatot{!z}, i1, 1, 4)
matplace(plsatot{!z}, zero1, 4, 4)
matrix (7, 6) plsbtot{!z}
matplace(plsbtot{!z}, plsb{!z}, 1, 1)
matplace(plsbtot{!z}, i2, 1, 3)
matplace(plsbtot{!z}, zero2, 5, 3)
matrix (7, 6) plsctot{!z}
matplace(plsctot{!z}, plsc{!z}, 1, 1)
matplace(plsctot{!z}, i2, 1, 3)
matplace(plsctot{!z}, zero2, 5, 3)

'LODE LI BASED ON SVD

'CREATION OF $\hat{\Pi}^i_{12}$

matrix plsescbtot{!z} = @subextract(plsb{!z}, 5,1,7,2)
matrix plsincbtot{!z} = @subextract(plsb{!z}, 1,1,4,2)
matrix plsescatot{!z} = @subextract(plsa{!z}, 4,1,7,3)
matrix plsincatot{!z} = @subextract(plsa{!z}, 1,1,3,3)
matrix plsescctot{!z} = @subextract(plsc{!z}, 5,1,7,2)
matrix plsincctot{!z} = @subextract(plsc{!z}, 1,1,4,2)
matrix (20,7) xa
matplace(xa, xinca, 1, 1)

```
matplace(xa, xesca, 1,4)
matrix (20,7) xb
matplace(xb, xincb, 1, 1)
matplace(xb, xescb, 1,5)
matrix (20,7) xc
matplace(xc, xincc, 1, 1)
matplace(xc, xescc, 1,5)
```

'ESTIMATE OF FIRST EQUATION

```
sym xesca2 = @transpose(xesca)*xesca
vector valxesca = @eigenvalues(xesca2)
matrix vecxesca = @eigenvectors(xesca2)
scalar valxesca1 = valxesca(1)^0.5
scalar valxesca2 = valxesca(2)^0.5
scalar valxesca3 = valxesca(3)^0.5
scalar valxesca4 = valxesca(4)^0.5
sym (4, 4) mvalxesca
mvalxesca.fill valxesca1, 0, 0,0,valxesca2, 0,0, valxesca3,0,valxesca4
matrix rxesca = vecxesca*mvalxesca*@transpose(vecxesca)
matrix dajeesca{!z} = (rxesca)*plsescatot{!z}
matrix vdajeesca{!z}
vector adajeesca{!z}
matrix udajeesca{!z} = @svd(dajeesca{!z}, adajeesca{!z}, vdajeesca{!z})
matrix svdauuesc{!z} = -vdajeesca{!z}/vdajeesca{!z}(1, 3)
vector gauufinesc{!z} = @columnextract(svdauuesc{!z}, 3)
vector gregafin{!z} = @subextract(gauufinesc{!z}, 2,1,3,1)
vector residua{!z} = ya{!z} - ydipa{!z}*gregafin{!z}
series serresa{!z}
mtos(residua{!z} , serresa{!z})
equation resa{!z}.ls serresa{!z} X1 X2 X5
vector bregafin{!z} = resa{!z}.@coefs
```

'ESTIMATE OF SECOND EQUATION

```
sym xescb2 = @transpose(xescb)*xescb
vector valxescb = @eigenvalues(xescb2)
```

```
matrix vecxescb = @eigenvectors(xescb2)
scalar valxescb1 = valxescb(1)^0.5
scalar valxescb2 = valxescb(2)^0.5
scalar valxescb3 = valxescb(3)^0.5
sym (3, 3) mvalxescb
mvalxescb.fill valxescb1, 0, 0,valxescb2, 0, valxescb3
matrix rxescb = vecxescb*mvalxescb*@transpose(vecxescb)
matrix dajeescb{!z} = (rxescb)*plsescbtot{!z}
matrix vdajeescb{!z}
vector adajeescb{!z}
matrix udajeescb{!z} = @svd(dajeescb{!z}, adajeescb{!z}, vdajeescb{!z})
matrix svdbddesc{!z} = -vdajeescb{!z}/vdajeescb{!z}(1, 2)
vector gbddfinesc{!z} = @columnextract(svdbddesc{!z}, 2)
vector gregbfin{!z} = @subextract(gbddfinesc{!z}, 2,1,2,1)
vector residub{!z} = yb{!z} - ya{!z}*gregbfin{!z}
series serresb{!z}
mtos(residub{!z} , serresb{!z})
equation resb{!z}.ls serresb{!z} X1 X3 X5 X7
vector bregbfin{!z} = resb{!z}.@coefs
```

'ESTIMATE OF THIRD EQUATION

```
sym xescc2 = @transpose(xescc)*xescc
vector valxescc = @eigenvalues(xescc2)
matrix vecxescc = @eigenvectors(xescc2)
scalar valxescc1 = valxescc(1)^0.5
scalar valxescc2 = valxescc(2)^0.5
scalar valxescc3 = valxescc(3)^0.5
sym (3, 3) mvalxescc
mvalxescc.fill valxescc1, 0, 0,valxescc2, 0, valxescc3
matrix rxescc = vecxescc*mvalxescc*@transpose(vecxescc)
matrix dajeescc{!z} = (rxescc)*plsescctot{!z}
matrix vdajeescc{!z}
vector adajeescc{!z}
matrix udajeescc{!z} = @svd(dajeescc{!z}, adajeescc{!z}, vdajeescc{!z})
matrix svdcttesc{!z} = -vdajeescc{!z}/vdajeescc{!z}(1, 2)
```

```
vector gcttfinesc{!z} = @columnextract(svdcttesc{!z}, 2)
vector gregcfin{!z} = @subextract(gcttfinesc{!z}, 2,1,2,1)
vector residuc{!z} = yc{!z} - yb{!z}*gregcfin{!z}
series serresc{!z}
mtos(residuc{!z} , serresc{!z})
equation resc{!z}.ls serresc{!z} X1 X3 X4 X6
vector bregcfin{!z} = resc{!z}.@coefs
```

'RESULTS

```
matrix (4, 500) betaclimsvd
colplace( betaclimsvd, bregcfin{!z}, {!z})
matrix (2, 500) gammaclimsvd
colplace( gammaclimsvd, gcttfinesc{!z}, {!z})
matrix (6,500) lodeclimsvd
matplace(lodeclimsvd, gammaclimsvd, 1, 1)
matplace(lodeclimsvd, betaclimsvd, 3, 1)
matrix (4, 500) betablimsvd
colplace( betablimsvd, bregbfin{!z}, {!z})
matrix (2, 500) gammablimsvd
colplace( gammablimsvd, gbddfinesc{!z}, {!z})
matrix (6,500) lodeblimsvd
matplace(lodeblimsvd, gammablimsvd, 1, 1)
matplace(lodeblimsvd, betablimsvd, 3, 1)
matrix (3, 500) betaalimsvd
colplace( betaalimsvd, bregafin{!z}, {!z})
matrix (3, 500) gammaalimsvd
colplace( gammaalimsvd, gauufinesc{!z}, {!z})
matrix (6,500) lodealimsvd
matplace(lodealimsvd, gammaalimsvd, 1, 1)
matplace(lodealimsvd, betaalimsvd, 4, 1)
```

'LODE FI with empirical correction

'CREATION OF EXTRADIAGONAL ELEMENTS OF A_{22}

```
matrix auud{!z} = @transpose(yinca{!z})*x*xsecsec*@transpose(x)*yincb{!z}
matrix adud{!z} = @transpose(yinca{!z})*xincb
matrix atud{!z} = @transpose(xinca)*yincb{!z}
matrix aqud = @transpose(xinca)*xincb
matrix (6, 6) aud{!z}
matplace(aud{!z}, auud{!z}, 1, 1)
matplace(aud{!z}, adud{!z}, 1, 3)
matplace(aud{!z}, atud{!z}, 4, 1)
matplace(aud{!z}, aqud, 4, 3)
matrix auut{!z} = @transpose(yinca{!z})*x*xsecsec*@transpose(x)*yincc{!z}
matrix adut{!z} = @transpose(yinca{!z})*xincc
matrix atut{!z} = @transpose(xinca)*yincc{!z}
matrix aqut = @transpose(xinca)*xincc
matrix (6, 6) aut{!z}
matplace(aut{!z}, auut{!z}, 1, 1)
matplace(aut{!z}, adut{!z}, 1, 3)
matplace(aut{!z}, atut{!z}, 4, 1)
matplace(aut{!z}, aqut, 4, 3)
matrix budu{!z} = @transpose(yincb{!z})*x*xsecsec*@transpose(x)*yinca{!z}
matrix bddu{!z} = @transpose(yincb{!z})*xinca
matrix btdu{!z} = @transpose(xincb)*yinca{!z}
matrix bqdu = @transpose(xincb)*xinca
matrix (6, 6) bdu{!z}
matplace(bdu{!z}, budu{!z}, 1, 1)
matplace(bdu{!z}, bddu{!z}, 1, 4)
matplace(bdu{!z}, btdu{!z}, 3, 1)
matplace(bdu{!z}, bqdu, 3, 4)
matrix budt{!z} = @transpose(yincb{!z})*x*xsecsec*@transpose(x)*yincc{!z}
matrix bddt{!z} = @transpose(yincb{!z})*xincc
matrix btdt{!z} = @transpose(xincb)*yincc{!z}
matrix bqdt = @transpose(xincb)*xincc
matrix (6, 6) bdt{!z}
matplace(bdt{!z}, budt{!z}, 1, 1)
```

```
matplace(bdt{!z}, bddt{!z}, 1, 3)
matplace(bdt{!z}, btdt{!z}, 3, 1)
matplace(bdt{!z}, bqdt, 3, 3)
matrix cutu{!z} = @transpose(yincc{!z})*x*xsecsec*@transpose(x)*yinca{!z}
matrix cdtu{!z} = @transpose(yincc{!z})*xinca
matrix cttu{!z} = @transpose(xincc)*yinca{!z}
matrix cqtu = @transpose(xincc)*xinca
matrix (6, 6) ctu{!z}
matplace(ctu{!z}, cutu{!z}, 1, 1)
matplace(ctu{!z}, cdtu{!z}, 1, 4)
matplace(ctu{!z}, cttu{!z}, 3, 1)
matplace(ctu{!z}, cqtu, 3, 4)
matrix cutd{!z} = @transpose(yincc{!z})*x*xsecsec*@transpose(x)*yincb{!z}
matrix cdtd{!z} = @transpose(yincc{!z})*xincb
matrix cttd{!z} = @transpose(xincc)*yincb{!z}
matrix cqtd = @transpose(xincc)*xincb
matrix (6, 6) ctd{!z}
matplace(ctd{!z}, cutd{!z}, 1, 1)
matplace(ctd{!z}, cdtd{!z}, 1, 3)
matplace(ctd{!z}, cttd{!z}, 3, 1)
matplace(ctd{!z}, cqtd, 3, 3)
```

'CREATION OF A_{22}

```
vector residuoa{!z} = ya{!z} - x*coefa{!z}
vector residuob{!z} = yb{!z} - x*coefb{!z}
vector residuoc{!z} = yc{!z} - x*coefc{!z}
matrix (20, 3) newres{!z}
matplace(newres{!z}, residuoa{!z}, 1, 1)
matplace(newres{!z}, residuob{!z}, 1, 2)
matplace(newres{!z}, residuoc{!z}, 1, 3)
matrix (3, 3) gammatot{!z}
gammatot{!z}.fill -1, vecauufin{!z}(2), vecauufin{!z}(3), vecbddfin{!z}(2), -1, 0, 0,
vecctfin{!z}(2), -1
matrix ulode{!z} = -newres{!z}*gammatot{!z}
sym sigma{!z} = (@transpose(ulode{!z})*ulode{!z})/14
```

```
matrix ulode{!z} = -newres{!z}*gammatot{!z}
sym sigma{!z} = (@transpose(ulode{!z})*ulode{!z})/14
sym sigmainv{!z} = @inverse(sigma{!z})
matrix sauu{!z} = sigmainv{!z}(1, 1)*auu{!z}
matrix saud{!z} = sigmainv{!z}(1, 2)*aud{!z}
matrix saut{!z} = sigmainv{!z}(1, 3)*aut{!z}
matrix sbdu{!z} = sigmainv{!z}(2, 1)*bdu{!z}
matrix sbdd{!z} = sigmainv{!z}(2, 2)*bdd{!z}
matrix sbdt{!z} = sigmainv{!z}(2, 3)*bdt{!z}
matrix sctu{!z} = sigmainv{!z}(3, 1)*ctu{!z}
matrix sctd{!z} = sigmainv{!z}(3, 2)*ctd{!z}
matrix sctt{!z} = sigmainv{!z}(3, 3)*ctt{!z}
matrix (18, 18) a{!z}
matplace(a{!z}, sauu{!z}, 1, 1)
matplace(a{!z}, saud{!z}, 1, 7)
matplace(a{!z}, saut{!z}, 1, 13)
matplace(a{!z}, sbdu{!z}, 7, 1)
matplace(a{!z}, sbdd{!z}, 7, 7)
matplace(a{!z}, sbdt{!z}, 7, 13)
matplace(a{!z}, sctu{!z}, 13, 1)
matplace(a{!z}, sctd{!z}, 13, 7)
matplace(a{!z}, sctt{!z}, 13, 13)
sym sa{!z} = a{!z}
```

'CHOICE OF SUB-EIGENVECTORS WHICH MINIMIZE THE ESTIMATE RESIDUAL VARIANCE

```
matrix kautovec{!z} = @eigenvectors(sa{!z})
matrix kautoval{!z} = @eigenvalues(sa{!z})
matrix kveca{!z} = -kautovec{!z}/kautovec{!z}(1, 1)
vector finafina{!z} = @subextract(kveca{!z}, 1, 1, 6, 1)
matrix kvecab{!z} = -kautovec{!z}/kautovec{!z}(1, 2)
vector finafinb{!z} = @subextract(kvecab{!z}, 1, 2, 6, 2)
matrix kvecac{!z} = -kautovec{!z}/kautovec{!z}(1, 3)
vector finafinc{!z} = @subextract(kvecac{!z}, 1, 3, 6, 3)
matrix kvecb{!z} = -kautovec{!z}/kautovec{!z}(7, 1)
```

```
vector finbfina{!z} = @subextract(kvecb{!z}, 7, 1, 12, 1)
matrix kvecbb{!z} = -kautovec{!z}/kautovec{!z}(7, 2)
vector finbfinb{!z} = @subextract(kvecbb{!z}, 7, 2, 12, 2)
matrix kvecbc{!z} = -kautovec{!z}/kautovec{!z}(7, 3)
vector finbfinc{!z} = @subextract(kvecbc{!z}, 7, 3, 12, 3)
matrix kvecc{!z} = -kautovec{!z}/kautovec{!z}(13, 1)
vector fincfina{!z} = @subextract(kvecc{!z}, 13, 1, 18, 1)
matrix kveccb{!z} = -kautovec{!z}/kautovec{!z}(13, 2)
vector fincfinb{!z} = @subextract(kveccb{!z}, 13, 2, 18, 2)
matrix kveccc{!z} = -kautovec{!z}/kautovec{!z}(13, 3)
vector fincfinc{!z} = @subextract(kveccc{!z}, 13, 3, 18, 3)
vector provaaa{!z} = @subextract(kveca{!z}, 2, 1, 6, 1)
vector provaab{!z} = @subextract(kvecab{!z}, 2, 2, 6, 2)
vector provaac{!z} = @subextract(kvecac{!z}, 2, 3, 6, 3)
vector erroreaa{!z} = ya{!z} - zuno{!z}*provaaa{!z}
vector erroreab{!z} = ya{!z} - zuno{!z}*provaab{!z}
vector erroreac{!z} = ya{!z} - zuno{!z}*provaac{!z}
vector varianzaaa{!z} = @transpose(erroreaa{!z})*erroreaa{!z}
vector varianzaab{!z} = @transpose(erroreab{!z})*erroreab{!z}
vector varianzaac{!z} = @transpose(erroreac{!z})*erroreac{!z}
vector provaba{!z} = @subextract(kvecb{!z}, 8, 1, 12, 1)
vector provabb{!z} = @subextract(kvecbb{!z}, 8, 2, 12, 2)
vector provabc{!z} = @subextract(kvecbc{!z}, 8, 3, 12, 3)
vector erroreba{!z} = yb{!z} - zdue{!z}*provaba{!z}
vector errorebb{!z} = yb{!z} - zdue{!z}*provabb{!z}
vector errorebc{!z} = yb{!z} - zdue{!z}*provabc{!z}
vector varianzaba{!z} = @transpose(erroreba{!z})*erroreba{!z}
vector varianzabb{!z} = @transpose(errorebb{!z})*errorebb{!z}
vector varianzabc{!z} = @transpose(errorebc{!z})*errorebc{!z}
vector provaca{!z} = @subextract(kvecc{!z}, 14, 1, 18, 1)
vector provacb{!z} = @subextract(kveccb{!z}, 14, 2, 18, 2)
vector provacc{!z} = @subextract(kveccc{!z}, 14, 3, 18, 3)
vector erroreca{!z} = yc{!z} - ztre{!z}*provaca{!z}
vector errorecb{!z} = yc{!z} - ztre{!z}*provacb{!z}
vector errorecc{!z} = yc{!z} - ztre{!z}*provacc{!z}
```

```
vector varianzaca{!z} = @transpose(erroreca{!z})*erroreca{!z}
vector varianzacb{!z} = @transpose(errorecb{!z})*errorecb{!z}
vector varianzacc{!z} = @transpose(errorecc{!z})*errorecc{!z}
vector (3, 1) minvara{!z}
minvara{!z}.fill varianzaaa{!z}(1), varianzaab{!z}(1), varianzaac{!z}(1)
scalar mina{!z} = @min(minvara{!z})
vector (3, 1) minvarb{!z}
minvarb{!z}.fill varianzaba{!z}(1), varianzabb{!z}(1), varianzabc{!z}(1)
scalar minb{!z} = @min(minvarb{!z})
vector (3, 1) minvarc{!z}
minvarc{!z}.fill varianzaca{!z}(1), varianzacb{!z}(1), varianzacc{!z}(1)
scalar minc{!z} = @min(minvarc{!z})
if mina{!z} - varianzaaa{!z}(1) = 0 then matrix kvecaa{!z} = -kautovec{!z}/kautovec{!z}(1,
1)
vector finafin{!z} = @subextract(kvecaa{!z}, 1, 1, 6, 1)
scalar auto_presoa{!z} = 1
else
if mina{!z} - varianzaab{!z}(1) = 0 then
matrix kvecaa{!z} = -kautovec{!z}/kautovec{!z}(1, 2)
vector finafin{!z} = @subextract(kvecaa{!z}, 1, 2, 6, 2)
scalar auto_presoa{!z} = 2
else
if mina{!z} - varianzaac{!z}(1) = 0 then matrix kvecaa{!z} = -kautovec{!z}/kautovec{!z}(1,
3)
vector finafin{!z} = @subextract(kvecaa{!z}, 1, 3, 6, 3)
scalar auto_presoa{!z} = 3
endif
endif
endif
if minb{!z} - varianzaba{!z}(1) = 0 then matrix kvecba{!z} = -kautovec{!z}/kautovec{!z}(7,
1)
vector finbfin{!z} = @subextract(kvecba{!z}, 7, 1, 12, 1)
scalar auto_presob{!z} = 1
else
if minb{!z} - varianzabb{!z}(1) = 0 then
```

```
matrix kvecba{!z} = -kautovec{!z}/kautovec{!z}(7, 2)
vector finbfin{!z} = @subextract(kvecba{!z}, 7, 2, 12, 2)
scalar auto_presob{!z} = 2
else
if minb{!z} - varianzabc{!z}(1) = 0 then matrix kvecba{!z} = -kautovec{!z}/kautovec{!z}(7,
3)
vector finbfin{!z} = @subextract(kvecba{!z}, 7, 3, 12, 3)
scalar auto_presob{!z} = 3
endif
endif
endif
if minc{!z} - varianzaca{!z}(1) = 0 then matrix kvecca{!z} = -kautovec{!z}/kautovec{!z}(13,
1)
vector fincfin{!z} = @subextract(kvecca{!z}, 13, 1, 18, 1)
scalar auto_presoc{!z} = 1
else
if minc{!z} - varianzacb{!z}(1) = 0 then
matrix kvecca{!z} = -kautovec{!z}/kautovec{!z}(13, 2)
vector fincfin{!z} = @subextract(kvecca{!z}, 13, 2, 18, 2)
scalar auto_presoc{!z} = 2
else
if minc{!z} - varianzacc{!z}(1) = 0 then matrix kvecca{!z} = -kautovec{!z}/kautovec{!z}(13,
3)
vector fincfin{!z} = @subextract(kvecca{!z}, 13, 3, 18, 3)
scalar auto_presoc{!z} = 3
endif
endif
endif
matrix (6, 500) lodea
colplace( lodea, finafin{!z}, {!z})
matrix (6, 500) lodeb
colplace( lodeb, finbfin{!z}, {!z})
matrix (6, 500) lodec
colplace( lodec, fincfin{!z}, {!z})
```

'3SLS AND FIML

```
system tresls{!z}
tresls{!z}.append serya{!z} = c(1) + c(2)*X2 + c(3)*X5 + c(4)*seryb{!z} + c(5)*seryc{!z}
@ X2 X5 seryb{!z} seryc{!z} c(1)
tresls{!z}.append seryb{!z} = c(6) + c(7)*X3 + c(8)*X5 + c(9)*X7 + c(10)*serya{!z}
@ X3 X5 X7 serya{!z} c(6)
tresls{!z}.append seryc{!z} = c(11) + c(12)*X3 + c(13)*X4 + c(14)*X6 + c(15)*seryb{!z}
@ X3 X4 X6 seryb{!z} c(11)
tresls{!z}.3sls
vector coeftresls{!z} = tresls{!z}.@coefs
matrix (15, 500) fintresls
colplace(fintresls, coeftresls{!z}, {!z})
system fumil{!z}
fumil{!z}.append serya{!z} = c(1) + c(2)*X2 + c(3)*X5 + c(4)*seryb{!z} + c(5)*seryc{!z}
@ X2 X5 seryb{!z} seryc{!z} c(1)
fumil{!z}.append seryb{!z} = c(6) + c(7)*X3 + c(8)*X5 + c(9)*X7 + c(10)*serya{!z}
@ X3 X5 X7 serya{!z} c(6)
fumil{!z}.append seryc{!z} = c(11) + c(12)*X3 + c(13)*X4 + c(14)*X6 + c(15)*seryb{!z}
@ X3 X4 X6 seryb{!z} c(11)
fumil{!z}.fiml
vector coeffumil{!z} = fumil{!z}.@coefs
matrix (15, 500) finfumil
colplace(finfumil, coeffumil{!z}, {!z})
```

'LODE FI BASED ON SVD

```
matrix (3, 3) gammatotnew{!z}
gammatotnew{!z}.fill -1, gauufinesc{!z}(2), gauufinesc{!z}(3), gbddfinesc{!z}(2), -1,
0, 0, gcttfinesc{!z}(2), -1
matrix ulodenew{!z} = -newres{!z}*gammatotnew{!z}
sym sigmanew{!z} = (@transpose(ulodenew{!z})*ulodenew{!z})/14
sym sigmainvnew{!z} = @inverse(sigmanew{!z})
matrix (4, 3) plsaesc{!z}
plsaesc{!z}.fill coeflsara{!z}, coeflsata{!z}, coeflsaia{!z}, coeflsaoa{!z}, coeflsbrb{!z},
coeflsbtb{!z}, coeflsbib{!z}, coeflsbob{!z}, coeflscrc{!z}, coeflsctc{!z}, coeflscic{!z},
coeflscoc{!z}
```

```
matrix (3, 2) plsbesc{!z}
plsbesc{!z}.fill coeflsbeb{!z}, coeflsbtb{!z}, coeflsbib{!z}, coeflsaea{!z}, coeflsata{!z},
coeflsaia{!z}
matrix (3, 2) plscesc{!z}
plscesc{!z}.fill coeflscec{!z}, coeflscuc{!z}, coeflscoc{!z}, coeflsbeb{!z}, coeflsbub{!z},
coeflsbob{!z}
matrix xescab2 = @transpose(xesca)*xescb
matrix xescba2 = @transpose(xescb)*xesca
matrix xescac2 = @transpose(xesca)*xescc
matrix xescca2 = @transpose(xescc)*xesca
matrix xesccb2 = @transpose(xescc)*xescb
matrix xescbc2 = @transpose(xescb)*xescc
matrix suuesc{!z} = sigmainvnew{!z}(1, 1)*xesca2
matrix sudesc{!z} = sigmainvnew{!z}(1, 2)*xescab2
matrix sutesc{!z} = sigmainvnew{!z}(1, 3)*xescac2
matrix sduesc{!z} = sigmainvnew{!z}(2, 1)*xescba2
matrix sddesc{!z} = sigmainvnew{!z}(2, 2)*xescb2
matrix sdtesc{!z} = sigmainvnew{!z}(2, 3)*xescbc2
matrix stuesc{!z} = sigmainvnew{!z}(3, 1)*xescca2
matrix stdesc{!z} = sigmainvnew{!z}(3, 2)*xesccb2
matrix sttesc{!z} = sigmainvnew{!z}(3, 3)*xescc2
matrix (10, 10) preaesc{!z}
matplace(preaesc{!z}, suuesc{!z}, 1, 1)
matplace(preaesc{!z}, sudesc{!z}, 1, 5)
matplace(preaesc{!z}, sutesc{!z}, 1, 8)
matplace(preaesc{!z}, sduesc{!z}, 5, 1)
matplace(preaesc{!z}, sddesc{!z}, 5, 5)
matplace(preaesc{!z}, sdtesc{!z}, 5, 8)
matplace(preaesc{!z}, stuesc{!z}, 8, 1)
matplace(preaesc{!z}, stdesc{!z}, 8, 5)
matplace(preaesc{!z}, sttesc{!z}, 8, 8)
sym spreaesc{!z} = preaesc{!z}
matrix escvec{!z} = @eigenvectors(spreaesc{!z})
vector escval{!z} = @eigenvalues(spreaesc{!z})
for !v = 1 to @rows(escval{!z})
```

```
vector radesc{!v} = escval{!z}({!v})^0.5
matrix (@rows(escval{!z}), 1) mradesc{!z}
rowplace(mradesc{!z}, radesc{!v},{!v})
vector vradesc{!z} = mradesc{!z}
matrix dradesc{!z} = @makediagonal(vradesc{!z})
matrix stanesc{!z} = escvec{!z}*dradesc{!z}*@transpose(escvec{!z})
next
matrix (10, 7) plsesctot{!z}
matplace(plsesctot{!z},plsaesc{!z},1,1)
matplace(plsesctot{!z},plsbesc{!z}, 5, 4)
matplace(plsesctot{!z},plscesc{!z},8,6)
```

'CREATION OF $\hat{\Pi}_{12}$

```
matrix plescsvd{!z} = stanesc{!z}*plsesctot{!z}
matrix plsvdaesc{!z} = @subextract(plescsvd{!z}, 1, 1,10,1)
matrix plsvdbesc{!z} = @subextract(plescsvd{!z}, 1, 4, 10, 4)
matrix plsvdcesc{!z} = @subextract(plescsvd{!z}, 1, 6, 10, 6)
matrix plsvdaaesc{!z} = @subextract(plescsvd{!z}, 1, 2, 10, 3)
matrix plsvdbbesc{!z} = @subextract(plescsvd{!z}, 1, 5, 10, 5)
matrix plsvdccesc{!z} = @subextract(plescsvd{!z}, 1, 7, 10, 7)
matrix (10, 7)plsescnew{!z}
matplace(plsescnew{!z},plsvdaesc{!z},1,1)
matplace(plsescnew{!z},plsvdbesc{!z}, 1, 2)
matplace(plsescnew{!z},plsvdcesc{!z}, 1, 3)
matplace(plsescnew{!z},plsvdaaesc{!z}, 1, 4)
matplace(plsescnew{!z},plsvdbbesc{!z}, 1 ,6)
matplace(plsescnew{!z},plsvdccesc{!z}, 1 ,7)
matrix vplsvdesc{!z}
vector aplsvdesc{!z}
matrix uplsvdesc{!z} = @svd(plsescnew{!z}, aplsvdesc{!z}, vplsvdesc{!z})
matrix escud{!z} = @subextract(vplsvdesc{!z}, 1, 5, 3, 7)
matrix esctot{!z} = @subextract(vplsvdesc{!z}, 1, 5, 7, 7)
matrix proesc{!z} = -esctot{!z}*@inverse(escud{!z})
scalar gapri{!z} = proesc{!z}(4,1)
scalar gasec{!z} = proesc{!z}(5,1)
```

```
scalar gbpri{!z} = proesc{!z}(6,2)
scalar gcpri{!z} = proesc{!z}(7,3)
vector (3) gammaa{!z}
gammaa{!z}.fill -1, gapri{!z}, gasec{!z}
vector (2) gammab{!z}
gammab{!z}.fill -1, gbpri{!z}
vector (2) gammac{!z}
gammac{!z}.fill -1, gcpri{!z}
vector gregafinful{!z} = @subextract(gammaa{!z}, 2,1,3,1)
vector residuaful{!z} = ya{!z} - ydipa{!z}*gregafinful{!z}
series serresaful{!z}
mtos(residuaful{!z} , serresaful{!z})
equation resaful{!z}.ls serresaful{!z} X1 X2 X5
vector bregafinful{!z} = resaful{!z}.@coefs
vector gregbfinful{!z} = @subextract(gammab{!z}, 2,1,2,1)
vector residubful{!z} = yb{!z} - ya{!z}*gregbfinful{!z}
series serresbful{!z}
mtos(residubful{!z} , serresbful{!z})
equation resbful{!z}.ls serresbful{!z} X1 X3 X5 X7
vector bregbfinful{!z} = resbful{!z}.@coefs
vector gregcfinful{!z} = @subextract(gammac{!z}, 2,1,2,1)
vector residucful{!z} = yc{!z} - yb{!z}*gregcfinful{!z}
series serrescful{!z}
mtos(residucful{!z} , serrescful{!z})
equation rescful{!z}.ls serrescful{!z} X1 X3 X4 X6
vector bregcfinful{!z} = rescful{!z}.@coefs
matrix (3, 500) betaasvd2
colplace( betaasvd2, bregafinful{!z}, {!z})
matrix (4, 500) betabsvd2
colplace( betabsvd2, bregbfinful{!z}, {!z})
matrix (4, 500) betacsvd2
colplace( betacsvd2, bregcfinful{!z}, {!z})
matrix (3, 500) gammaasvd2
colplace( gammaasvd2, gammaa{!z}, {!z})
matrix (2, 500) gammabsvd2
```

```
colplace( gammabsvd2, gammab{!z}, {!z})
matrix (2, 500) gammacsvd2
colplace( gammacsvd2, gammac{!z}, {!z})
```

'RESULTS

```
matrix (6, 500) lodeasvd
matplace(lodeasvd, gammaasvd2, 1, 1)
matplace(lodeasvd, betaasvd2, 4, 1)
matrix (6, 500) lodebsvd
matplace(lodebsvd, gammabsvd2, 1, 1)
matplace(lodebsvd, betabsvd2, 3, 1)
matrix (6, 500) lodecsvd
matplace(lodecsvd, gammacsvd2, 1, 1)
matplace(lodecsvd, betacsvd2, 3, 1)
next
```

Bibliography

[1] Adcock(1878) A problem in the least square, the Analysis, 5 pp 53-54

[2] Anderson T. W. (1976) Estimation of linear functional relationship: approximate distribusion and connections with simultaneous Equation Econometrics, J Roy. Statistics Soc . Ser. B 38 pp 1-20

[3] Cau G. (1990) Stimatori alternativi nei modelli ad equazioni simultanee: un esperimento di simulazione, Unpublished thesis, Università "La Sapienza", Rome

[4] Dunne B. E. and Williamson G. A. QR-Based Tls and Mixed LS-TLS algorithm Applications to Adaptive IIR Filtering (2003), IEEE Transactions on Signal Processing, Vol 51, NO. 2

[5] Fernando K. e Nicholson H.(1985) Identification of linear systems with input output noise, IEE Proc. D. 132, pp 30 36

[6] Gleser L. (1973) Estimation in a multivariate "errors in variable model" regression model : Large sample results Ann. Statist Vol 9 pp 525-534

[7] Goldeberger A.S. (1964), Econometric Theory, John Wiley & Sons, Inc.

[8] Golub G. H. (1973) Some modified matrix eigenvalue problem, SIAM Rev., vol 15, pp 318-344

[9] Golub G. H. e Van Loan C. (1980)An analysis of the total least square problem, SIAM J. Number Anal. 17

[10] Golub G. H. e Van Loan C.(1989) Matrix computations 2nd ed. Hopkins University Press, Baltimora

[11] Judge G. G., Griffith W. E., Hill R. C. , Lutkepol H. and Lee T. C.(1985), the Theory and practice of Econometrics, John Wiley & Sons, Inc.

[12] Levin M. (1964) Estimation of a system pulse transfer function in the presence of noise, IEEE Trans. Automat. Control. AC-9 pp 229-235

[13] Markovsky I.e Van Huffel S. (2007) Overview on total least square Signal Processing Vol 87 pp 2283-2303

[14] Naccarato e Pieraccini (2008) a Full information least distance estimator Statistica Vol. spec. pp52-76

[15] Naccarato A., Zurlo D. (2007), Least Orthogonal Distance Estimation of simultaneus equations: a simulation experiment, Collana del Dipartimento di Economia, Università degli Studi RomaTre, Working Paper n 90

[16] Naccarato A., Zurlo D. (2008), A Monte Carlo Study on Full Information Methods in Simultaneous Equation Models, Quaderni di Statistica vol. 10

[17] Naccarato A. (2007) Full Information Least Orthogonal Distance, Quaderni di Statistica vol. 10

[18] Pearson K. (1901) On lines and Planes of Closest Fit System of Points in Space, University College, London

[19] Perna C. (1989) Un confronto tra due metodi di stima: il 2SLS ed il LODE, Quaderni di Statistica e Econometria, vol. XI, pp. 23-43

[20] Perna C. (1988), La consistenza dello stimatore LODE nei sistemi ad equazioni simultanee, Quaderni di Statistica e Econometria, vol. X, pp. 15-24

[21] Pieraccini L. (1992), Metodi di stima dei parametri strutturali nei modelli lineari ad equazioni simultanee: una rassegna ed una proposta Atti della XXXI Riunione Scientifica della Società Italiana di Statistica pp.435-446

[22] Pieraccini L. (1988) Il metodo L.O.D.E. per la stima dei parametri strutturali di un sistema di equazioni simultanee, Quaderni di statistica e Econometria, Vol. X, pp 5-14

[23] Pieraccini L. (1983) The Estimation of Structural Prameters inSimultaneous Equation Models, Quaderni di Statistica e Econometria Vol. V, pp.3-20

[24] Pieraccini L. (1978) Su alcuni metodi di stima dei parametri strutturali di un sistema si equazioni simultanee, Quaderni di statistica e Econometria, Vol I, pp.1-30

[25] Pieraccini L. (1969), Su interpretazione alternativa del metodo dei minimi quadrati a due stadi, Statistica, vol 4 XI, pp. 786-802

[26] Sbrana G. (2001), Una Generalizzazione del metodo LODE per la stima dei parametri strutturali di un sistema ad equazioni simultanee, Quaderni di Statistica Vol. 3

[27] T. Soderstrom (2007) Error in Variables methods in system identification, Automatica Vol. 43, pp 939-958

[28] Van Huffel S. (1989) The partial total least square algorithm J. comput. Anal Vol. 93, pp. 149-162

[29] Van Huffel S. (1988) Analysis an properties of the generalized total least square problem SIAM J. Matrix Anal. Appl. Vol 10 pp 294-315

[30] Van Huffel S. (2007) Total least square and error in variable model Computational Stat. and Data Anal. Vol 52 pp 1076-1079

[31] Van Huffel S. (2002) Total Least Squares and Errors-In-Variables Modeling: Analysis, Algorithms and Applications, Kluwer Academic Publications

[32] Zurlo (2006) Un esperimento Montecarlo sugli stimatori dei parametri strutturali dei sistemi ad equazioni simultanee, Unpublished thesis, University of Roma Tre, Rome