Paper 338-2008

Computational Algorithm for Estimation of Panel Data Models with Two Fixed Effects

Alan Ricardo da Silva, IPEA, Brazil
Patrick Franco Alves, IPEA, Brazil

ABSTRACT

The panel data models are becoming more common in relation to cross-section and time-series models for innumerable presented advantages, in addition to the computational advance that facilitated theirs utilization. This work aimed at presenting a computational algorithm for estimation of panel data models with two fixed effects, since they demand much more resources than panel data models with one effect (fixed or random). The majority of commercial softwares have such implemented routines, however, in order to reduce the processing time, they estimate the models including dummy variables to the "time" effect. Such action produces a bias in the intercept of the model, over or underestimating the true value, by incorporating the influence of the year taken as base. The algorithm presented in this work showed itself very efficient for being able to work with a large data base, when compared to other algorithms, where the estimation with the same data base wasn't possible.

1. INTRODUCTION

The panel data models are being widely used, amongst other reasons, due to the computational advance. These models present advantages in relation to cross-section and time series models for, in addition to increasing the degrees of freedom, they manage to remove the influence of the individual and/or the time of the independent variables, thus making the estimates of model coefficients more realistic.

It is possible to estimate the model considering the "individual" and "time" effects as being fixed or random effects. Depending on the problem, one or both of the effects will be able to compose the cited model (The One-Way / Two-Way Fixed Effects or The One-Way / Two-Way Random Effects). There are some commercial softwares like SAS[®], STATA, EVIEWS, LIMDEP, R amongst others, that already have such implemented routines, however, in the case of the two fixed or random effects, the intercept is incorrectly estimated or in some cases, it is impossible to estimate the models because of the lack of computational resources such as memory and hard disk.

This paper aimed at presenting a more efficient computational algorithm developed using the IML module of the software SAS[®] 8.2 (the IML codes are easily converted for others programming languages), to the estimation of the panel data models with two fixed effects (the same idea is true to the two random effects). Moreover, it is intended to argue the relation between the processing time and estend to which the model will be used, in order to make the algorithm more efficient. The paper is organized in the following manner: section 2 introduces the necessary concepts about the panel data models that will be used in the study. Section 3 presents the developed algorithm in SAS[®]/IML and a case study to evaluate the potential of the program is presented in section 4, in addition to some considerations. Finally, the main conclusions about the study are presented.

2. PANEL DATA MODELS

As described by Hsiao (1986), there are many benefits in using panel data, being the principal ones: control of the individual heterogeneity; panel data models have a greater variability, less colinearity between variables, more degrees of freedom and more efficiency; they are more capable to identify and measure effects that aren't detected in cross-section or time series data. However, these models also have some limitations, being the requirement of more computational resources the most apparent of them.

The general formularization of the panel data model is shown in the Equation 1.

$$y_{it} = \alpha + \beta X_{it} + \mu_{it} \quad i = 1,...,N; \quad t = 1,...,T$$
$$\mu_{it} = \mu_i + \lambda_t + v_{it}$$

(1)

where:
i denotes the effect of the individual, the firm, etc (for means of simplicity, it will be referenced here as "individual")

and t denotes the time; y_{it} is the value of the dependent variable of the ith individual in the time t; α is the constant of the model; β is a vector k x 1 of the coefficients; X_{it} is a matrix of the independent variables of the ith individual in the time t; and μ_{it} is the disturbance of the ith individual in the time t.

The disturbance term, in turn, is composed by some components called variance components: μ_i, λ_t e v_{it}. These represent, respectively, the non-observable effect specific to the individual i, the non-observable effect specific to the time t and the remainder disturbance. To verify if such components are fixed effects (they influence directly the dependent variable) or random (they have correlation with the independent variables), specific tests are necessary. These tests will not be detailed here, but they can be found in Baltagi (2002), Greene (2002) and Hsiao (1986).

For the estimation of the parameters of model with one fixed effect (usually the "individual" in economic phenomena), a dummy variable can be attributed to each individual (remember that N-1 dummies must be inserted in the model), or a matrix called **Q** can be used, which eliminates the necessity of dummies. The main difference between these two forms is that when the matrix **Q** is used, in addition to the faster process, the intercept of the model considers the influence of all the individuals. The matrix **Q** is shown in the Equation 2.

$$\mathbf{Q} = \mathbf{I}(T) - \frac{\mathbf{ee'}}{T}$$
(2)

where, $\mathbf{I}(T)$ is an identity matrix of dimension T and $\mathbf{e}$ is a vector of 1's also of dimension T. Thus, the vector of the parameters referring to the independent variables is estimated such as in the Equation 3.

$$\hat{\boldsymbol{\beta}} = \sum_{i=1}^{N} \left(\mathbf{X}_i' \mathbf{Q} \mathbf{X}_i \right)^{-1} \mathbf{X}_i' \mathbf{Q} \mathbf{y}_i$$
(3)

where, $\hat{\boldsymbol{\beta}}$ is the vector of the parameters to be estimated, $\mathbf{X}$ is the matrix of the independent variables and $\mathbf{y}$ is the vector of the dependent variable. It is valid to remember that the matrix $\mathbf{X}$ doesn't have to contain the vector of 1's. Therefore, when including the matrix **Q** in the estimation of $\boldsymbol{\beta}$, the intercept is nulled. This way, after the estimation, the intercept of the model can be calculated by $\alpha = \overline{y} - \overline{\mathbf{x}}\hat{\boldsymbol{\beta}}$, where $\overline{y}$ and $\overline{\mathbf{x}}$ are the averages of the dependent and independent variables, respectively.

In the case of the model with two fixed effects, the parameters can be estimated creating N-1 dummies referred to the "individuals" and T-1 dummies referred to the "time", or a matrix **Q*** can be used, as defined in the Equation 4.

$$\mathbf{Q}^* = \left(\mathbf{I}(T) - \frac{\mathbf{ee'}}{T} \right) \otimes \left(\mathbf{I}(N) - \frac{\mathbf{ll'}}{N} \right)$$
(4)

where $\mathbf{l}$ represents a vector of 1's of dimension N and $\otimes$ represents the Kronecker product, i.e., the matrix **Q*** will have a final dimension of NT x NT. In this case, the parameters about "time" can be calculated such as in the Equation 5:

$$\hat{\lambda}_t = (\overline{\mathbf{y}}_{.t} - \overline{y}_{..}) - \hat{\boldsymbol{\beta}}(\overline{\mathbf{x}}_{.t} - \overline{\mathbf{x}}_{..})$$
(5)

where $\overline{\mathbf{y}}_{.t}$ represents the average of y in the time t, $\overline{y}_{..}$ represents the average of y, $\overline{\mathbf{x}}_{.t}$ represents the matrix of the averages of $\mathbf{X}$ in the time t and $\overline{\mathbf{x}}_{..}$ represents the matrix of the averages of $\mathbf{X}$. The parameters about "individual" can be calculated such as in the Equation 6:

$$\hat{\mu}_i = (\overline{\mathbf{y}}_{i.} - \overline{y}_{..}) - \hat{\boldsymbol{\beta}}(\overline{\mathbf{x}}_{i.} - \overline{\mathbf{x}}_{..})$$
(6)

where $\overline{\mathbf{y}}_{i.}$ represents the average of y of the individual i, $\overline{y}_{..}$ represents the average of y, $\overline{\mathbf{x}}_{i.}$ represents the matrix

of the averages of $\mathbf{X}$ of the individual i and $\bar{\mathbf{x}}_{..}$ represents the matrix of the averages of $\mathbf{X}$.

Recent studies made by Park (2005) about panel data models using the softwares SAS[®], STATA, LIMDEP and SPSS, illustrated how to estimate each model using these softwares. The author considers that the method LSDV (Least Squares Dummy Variables) using in SAS[®] 9.1 estimates correctly the intercept of the model, and compares it with the others programs. In the case of the Two Fixed Effects, only the SAS 9.1 and LIMDEP 8.0 estimate the model and the results (in the example) presented differences between the intercepts. The software STATA requires that the user creates dummies variables for the "time" effect.

The LIMDEP program uses the matrix **Q*** described in this paper and because of that, the intercept is different from the estimated one by SAS[®]. The point is: which is the correct? Park (2005) considers the LSDV method the best, affirming that LIMDEP reports the incorrect intercept in the two-way fixed model. But, neither SAS[®] nor LIMDEP are capable to estimate the model with many observations (SAS, 2006; LIMDEP, 2007), because of the amount of variables and then matrix's dimension, respectively.

If the dimensions of N (mainly) and of T are high, there will be a large processing time to calculate the parameters of the model, or depending of the hardware used, no estimation will be possible. This way, the next section will discuss about a more efficient algorithm to the estimation of such models, using the matrix **Q***.

3. IMPLEMENTED ALGORITHM
The algorithm was developed using the contained theory in Baltagi (2002), Greene (2002) and Hsiao (1986). For simplification, the theory in Hsiao (1986) shows the fixed and random effects formulation only to one factor, emphasizing the incorporation of the matrix **Q**, described previously, to the estimation of the parameters of the model. On the other hand, the Baltagi's book (2002) contains the fixed and random effects formulations for one and two factors. However, the matricial formulation presented is all about the calculation of the matrices with dimension NT x NT, which demands more computational resources. Thus, the following algorithm describes a form to calculate the parameters of the model from de matrices with dimension T, which requires an N-times recalculation. This is more efficient from the computational point of view.

The macro was developed in the software SAS[®] 8.2, using the IML module (Interactive Matrix Language). It contains an algorithm for the estimation of the four most commonly used models, being them: the One-Way Fixed Effect (fixone); the Two-Way Fixed Effects (fixtwo); the One-Way Random Effect (ranone) and the Two-Way Random Effects (rantwo). But, due to the focus of the paper, only the programming about two fixed effects will be presented.

Before discussing the developed algorithm, a ponderation is valid here about the degree of use of the model. If the interest of the researcher will be only in the parameters estimate referring to the independent variables of the model, without considering the intercept magnitude, it is more advisable to use the model with one factor, adding to the matrix of the independent variables, T-1 dummies about "time". Now, if the intercept will also be considered important, it is essential to use the matrix **Q*** described previously, because the use of dummy variables produces a bias in the intercept, due to the influence of the dummy referent to the "time" chosen as base.

Allied to this degree of use of the model is the processing time, i.e., how long the researcher will wait for visualization of the results. Early in the beginning of the algorithm there is a condition about the amount of the "individuals" (the size of N). If N is bigger than 1,000, than the model will be estimated attributing dummy variables automatically to the "time" effect (as it is done in the great part of the commercial softwares). Otherwise, the estimation will use the matrix **Q***, not with dimension NT x NT, but with the use of the matrix with dimension T x T recalculated N^2 times. The dimension of N used can be modified in accordance with the interest of the researcher (this clause is in line 27 and 139 of the SAS[®] algorithm).

The algorithm developed for the utilization of the matrix **Q*** is presented as follows. As the Kronecker product multiplies each element of the matrix **A** to the each element of the matrix **B**, the algorithm idea was to create a double looping of size N so that the matrix used in the calculation had size T, which needs less space in the memory. Thus, the RAM memory together with the virtual memory do not deplete when storing the matrix **Q***. The complete program is presented in the Appendix I.

```
do i=1 to N;
   b11=j(nvar,T,0);
```

3

```
      En=I(N)[,i]-k[,i];
          do j=1 to N;
              Q=En[j]@Et;
              l1=(j-1)*T+1;
              l2=j*T;
              b11=b11+x[l1:l2,]`*Q;
          end;
      l1=(i-1)*T+1;
      l2=i*T;
      b1=b1+b11*x[l1:l2,];
      b2=b2+b11*y[l1:l2,];
end;
```

In the next section, a case study will be presented to evaluate the potential of the algorithm, as well as to verify the differences in the intercept.

4. CASE STUDY

A country with continental dimensions like Brazil has a lot of information for studies, therefore, it needs softwares capable of working with all these data. The case study has as a goal the modeling of the Brazilian GDP (Gross Domestic Product, in Brazilian currency (R$)) of the 5,560 Brazilian cities, with data of 4 years (1999 to 2002), having as independent variables the added values (in R$) in the agriculture (AGRO) and industry (IND). That is, the database has 22,240 observations and the computer used for processing was a notebook Sony Vaio CENTRINO™ 1.5 Mhz with 2MB cache, 512 MB of RAM memory and 80 GB of hard disk.

The processing using the matrix **Q*** delayed approximately 1h30min, while when using the matrix **Q** with dummy variables to the "time" effect, it took approximately 2 seconds (the same with STATA software). Using the matrix **Q*** without considering the algorithm discussed before, the program aborted after 1 minute accusing lack of storage memory. Using the SAS[®] software (method LSDV), the program halted accusing that the limit of variables to estimate was surpassed. With the LIMDEP software, there was only a version free available, which read 5,000 data. Thus, it was not possible to analyze the software's performance. The results of the models are in Table 1.

Table 1- Results of the Models Using the Matrix **Q** and **Q***.

Variable	Model Using the Matrix *Q*			Model Using the Matrix *Q**		
	Parameter	*t Test*	*P-Value*	*Parameter*	*t Test*	*P-Value*
Intercept	61,801.766	25,83	<0.0001	58,994.069	37,11	<0.0001
AGRO	1.357	17,24	<0.0001	1.357	17,24	<0.0001
IND	1.638	210,62	<0.0001	1.638	210,62	<0.0001
D1999	-6,300.597	-2,76	0.005	-3,492.9	-1,85	0.063
D2000	-3,544.833	-1,57	0.116	-737.136	-0,38	0.704
D2001	-1,385.358	-0,63	0.528	1,422.338	0,68	0.496
D2002	-	-	-	2,807.697	1,17	0.240

It is verified in Table 1 that the parameters of the independent variables are exactly the same (which was expected), however, the intercepts differ from each other. In the case of the model that used the matrix **Q** (with dummies) to estimate the parameters of the model, the intercept was overestimated for containing the information of the dummy referent to the year of 2002 (base year). This fact is easily verified when the value referent to D2002 is deducted from the intercept of the model, resulting in the true intercept, i.e., 58,994.069. Another point to analyze is the magnitude of the parameters referent to the dummy variables of the years. Using T-1 dummy variables, they make reference to the year taken as base, and when the T dummies are calculated as in the Equation 5, they make reference to the average of the all period (in this case, the dummy values add up to zero), changing completely their interpretation.

Using another data base with 90 observations, the time was insignificant, being that the LIMDEP software presented the same results of the described algorithm.

CONCLUSION

Therefore, a considerable difference is verified in the processing time of the model: approximately 1h30min when using the matrix **Q*** and only 2 seconds when using the matrix **Q** with dummy variables referent to the "time"

effects. The majority of the statistical packages that estimate panel data models worry with the processing time and sometimes they leave aside the applicability of the model. Thus, depending of the problem studied, it is important to think about the impact of the intercept magnitude, mainly in forecasting cases. Finally, the developed algorithm showed itself sufficiently adequate, since despite the time factor, it was able to present the results of the model. On the other hand, the algorithm that uses the matrix of dimension NT x NT is unsuccessful.

REFERENCES

BALTAGI, C. H. (2002), *Econometric Analysis of Panel Data*. Wiley, 3rd ed, p.31-37.

GREENE, W. H. (2002), *Econometric Analysis*. Pearson Education, 5th ed, p.283-303.

HSIAO, C. (1986), *Analysis of Panel Data*. Econometric Society Monographs. New York: Cambridge University Press, p.12-49.

LIMDEP (2007), *Econometric Software, Inc.* "*http://www.limdep.com/features/specifications.php*".

PARK, H. M. (2005), *Linear Regression Models for Panel Data Using SAS, STATA, LIMDEP, and SPSS.* The Trustees of Indiana University.

SAS (2006), *SAS OnlineDoc*. "http://support.sas.com/onlinedoc/913/docMainpage.jsp" 2006.

CONTACT INFORMATION (HEADER 1)

Your comments and questions are valued and encouraged. Contact the author at:

Name: Alan Ricardo da Silva
Enterprise: IPEA
Address: SBS - Quadra 1 - Bloco J - Ed. BNDES
City, State ZIP: Brasilia, Brazil, 70076-900
Work Phone: +556133155020 / +556192162407
E-mail: djalan@bol.com.br
Web: www.ipea.gov.br

Name: Patrick Franco Alves
Enterprise: IPEA
Address: SBS - Quadra 1 - Bloco J - Ed. BNDES
City, State ZIP: Brasilia, Brazil, 70076-900
Work Phone: +556133155020
E-mail: patrick.alves@ipea.gov.br
Web: www.ipea.gov.br

APPENDIX I – SAS$^{®}$ ALGORITHM

```
%macro tscsreg(tab=,y=,x=,cross=,year=,type=);
proc iml;
/*Type = fixone, fixtwo, ranone or rantwo*/
/*******************************************************/
/* Written By Alan Ricardo da Silva, MSc. (09/2006) */
/* E-mail: djalan@bol.com.br                        */
/*                                              */
/* References:                                  */
/* [1]HSIAO,C.(1986). "ANALYSIS OF PANEL DATA".    */
/*    NEW YORK: CAMBRIDGE UNIVERSITY PRESS          */
/*    ECONOMETRIC SOCIETY MONOGRAPHS, P.12-49       */
/* [2]GREENE,W.H.(2002). "ECONOMETRIC ANALYSIS".    */
/*    PEARSON EDUCATION, 5th ed., P.283-303        */
/* [3]BALTAGI,C.H.(2002). "ECONOMETRIC ANALYSIS     */
/*    OF PANEL DATA". WILEY, 3rd ed., P.31-37       */
/*******************************************************/
/*Fixed Effects Two-Way*/
```

```
%if %upcase(&type)=FIXTWO %then %do;
use &tab;
read all var {&y} into y;
read all var {&x} into x;
read all var {&cross &year};
nomes={Intercept &x}`;
N=ncol(unique(&cross));
T=ncol(unique(&year));
NT=N*T;
      if N>1000 then do;
_year_=unique(&year);
dummyT=j(nrow(y),T-1,0);
do j=1 to T-1;
do i=1 to nrow(y);
if &year[i]=_year_[j] then dummyT[i,j]=1;
end;
end;
x=x||dummyT;
_year_=_year_`;
if type(year)='N' then nomes=nomes//char(_year_[1:T-1],4);
else nomes=nomes//_year_[1:T-1];
e=j(T,1,1);
Q=I(T)-(e*e`)/T;
nvar=ncol(x);
b1=j(nvar,nvar,0);
b2=j(nvar,1,0);
do i=1 to N;
l1=(i-1)*T+1;
l2=i*T;
b1=b1+x[l1:l2,]`*Q*x[l1:l2,];
b2=b2+x[l1:l2,]`*Q*y[l1:l2,];
end;
b=inv(b1)*b2;
yb=sum(y)/(NT);
xb=j(1,nvar,0);
do i=1 to nvar;
xb[,i]=sum(x[,i])/(NT);
end;
a=yb-xb*b;
sigma2u=0;
do i=1 to N;
l1=(i-1)*T+1;
l2=i*T;
sigma2u=sigma2u+((y[l1:l2,]-e*a-x[l1:l2,]*b)`*Q*(y[l1:l2,]-e*a-x[l1:l2,]*b));
end;
sigma2u=sigma2u/(N*(T-1)-nvar);
      end;
      else do;
e=j(T,T,1/T);
ee=j(N*T,1,1);
k=j(N,N,1/N);
It=I(T);
nvar=ncol(x);
Et=(It-e);
b1=j(nvar,nvar,0);
b2=j(nvar,1,0);
do i=1 to N;
b11=j(nvar,T,0);
En=I(N)[,i]-k[,i];
do j=1 to N;
Q=En[j]@Et;
l1=(j-1)*T+1;
l2=j*T;
b11=b11+x[l1:l2,]`*Q;
end;
```

6

```
l1=(i-1)*T+1;
l2=i*T;
b1=b1+b11*x[l1:l2,];
b2=b2+b11*y[l1:l2,];
end;
b=inv(b1)*b2;
yb=sum(y)/(NT);
xb=j(1,nvar,0);
do i=1 to nvar;
xb[,i]=sum(x[,i])/(NT);
end;
a=yb-xb*b;
close &tab;
sort &tab out=_&tab by &year &cross;
use _&tab var {&y};
read all into y_;
use _&tab var {&x};
read all into x_;
nvar=ncol(x_);
ybt=j(T,1,0);
do i=1 to T;
l1=(i-1)*N+1;
l2=i*N;
ybt[i]=(sum(y_[l1:l2])/N);
end;
xbt=j(T,nvar,0);
do j=1 to nvar;
do i=1 to T;
l1=(i-1)*N+1;
l2=i*N;
xbt[i,j]=sum(x_[l1:l2,j])/(N);
end;
end;
dyear=j(T,1,0);
do i=1 to T;
dyear[i]=(ybt[i]-yb)-(xbt[i,]-xb)*b;;
end;
ss1=j(1,1,0);
do i=1 to N;
ss11=j(1,T,0);
En=I(N)[,i]-k[,i];
do j=1 to N;
Q=En[j]@Et;
l1=(j-1)*T+1;
l2=j*T;
ss11=ss11+(y[l1:l2]-ee[l1:l2]*a-x[l1:l2,]*b)`*Q;
end;
l1=(i-1)*T+1;
l2=i*T;
ss1=ss1+ss11*(y[l1:l2]-ee[l1:l2]*a-x[l1:l2,]*b);
end;
sigma2u=ss1/((N-1)*(T-1)-nvar);
      end;
varb=sigma2u*inv(b1);
stdb=sqrt(vecdiag(abs(varb)));
stda=sqrt(abs(sigma2u/(T*N)+xb*varb*xb`));
if N<=1000 then do;
stddyear=j(T,1,0);
do i=1 to T;
stddyear[i]=sqrt(abs(sigma2u/(N)+xbt[i,]*varb*xbt[i,]`));
end;
bvec=a//b//dyear;
tmp=stda//stdb//stddyear;
_year_=unique(&year);
_year_=_year_`;
```

7

```
if type(year)='N' then nomes=nomes//char(_year_,4);
else nomes=nomes//_year_;
end;
else do;
bvec=a//b;
tmp=stda//stdb;
end;
tstat = bvec/tmp;
probt=2*(1-probt(abs(tstat),NT-2));
param=bvec||tmp||tstat||probt;
* Print;
d1={"Estimation Method" "Number of Cross Sections   " "Number of Time Series"
"Number of Observations"}`;
d2={"FixTwo"}//char(N,6)//char(T,6)//char(NT,6);
print "Dependent Variable: &y";
print "Model Description",, d1[label=''] d2[label=''];
est={'Estimate' 'Std. Error' 't Value' 'P > |t|'};
print,"Parameter Estimates",, nomes[label='Variable']param[ colname=est label=''];
* F Test;
ybi=j(N,1,0);
do i=1 to N;
l1=(i-1)*T+1;
l2=i*T;
ybi[i]=(sum(y[l1:l2])/T);
end;
xbi=j(N,nvar,0);
do j=1 to nvar;
do i=1 to N;
l1=(i-1)*T+1;
l2=i*T;
xbi[i,j]=sum(x[l1:l2,j])/(T);
end;
end;
close &tab;
sort &tab out=_&tab by &year &cross;
use _&tab var {&y};
read all into y_;
use _&tab var {&x};
read all into x_;
nvar=ncol(x_);
ybt=j(T,1,0);
do i=1 to T;
l1=(i-1)*N+1;
l2=i*N;
ybt[i]=(sum(y_[l1:l2])/N);
end;
xbt=j(T,nvar,0);
do j=1 to nvar;
do i=1 to T;
l1=(i-1)*N+1;
l2=i*N;
xbt[i,j]=sum(x_[l1:l2,j])/(N);
end;
end;
ym=j(nrow(y),1,0);
xm=j(nrow(x),nvar,0);
ymi=j(nrow(y),1,0);
xmi=j(nrow(x),nvar,0);
ymt=j(nrow(y),1,0);
xmt=j(nrow(x),nvar,0);
do j=1 to nvar;
do i=1 to N;
l1=(i-1)*T+1;
l2=i*T;
ym[l1:l2]=y[l1:l2]-ybi[i]-ybt+yb;
```

```
xm[l1:l2,j]=x[l1:l2,j]-xbi[i,j]-xbt[,j]+xb[j];
ymi[l1:l2]=y[l1:l2]-ybi[i];
xmi[l1:l2,j]=x[l1:l2,j]-xbi[i,j];
ymt[l1:l2]=y[l1:l2]-ybt;
xmt[l1:l2,j]=x[l1:l2,j]-xbt[,j];
end;
end;
Wxxi=j(N*nvar,nvar,0);
Wxyi=j(N*nvar,1,0);
Wyyi=j(N,1,0);
do i=1 to N;
l1=(i-1)*T+1;
l2=i*T;
k1=(i-1)*nvar+1;
k2=i*nvar;
Wxxi[k1:k2,]=Wxxi[k1:k2,]+xm[l1:l2,]`*xm[l1:l2,];
Wxyi[k1:k2]=Wxyi[k1:k2]+xm[l1:l2,]`*ym[l1:l2];
Wyyi[i]=Wyyi[i]+ym[l1:l2]`*ym[l1:l2];
end;
RSS=j(N,1,0);
do i=1 to N;
k1=(i-1)*nvar+1;
k2=i*nvar;
if det(Wxxi[k1:k2,])=0 then RSS[i]=0;
else RSS[i]=Wyyi[i]-Wxyi[k1:k2]`*inv(Wxxi[k1:k2,])*Wxyi[k1:k2];
end;
S1=sum(RSS);
Wxx=j(nvar,nvar,0);
Wxy=j(nvar,1,0);
do j=1 to nvar;
do i=1 to N;
k=j+(nvar*(i-1));
Wxx[j,]=Wxx[j,]+Wxxi[k,];
Wxy[j]=Wxy[j]+Wxyi[k];
end;
end;
Wyy=sum(Wyyi);
S2=Wyy-Wxy`*inv(Wxx)*Wxy;
xmb=j(nrow(x),nvar,0);
do i=1 to nvar;
xmb[,i]=x[,i]-xb[i];
end;
ymb=y-yb;
Txx=xmb`*xmb;
Txy=xmb`*ymb;
Tyy=ymb`*ymb;
Txxi=xmi`*xmi;
Txyi=xmi`*ymi;
Tyyi=ymi`*ymi;
Txxt=xmt`*xmt;
Txyt=xmt`*ymt;
Tyyt=ymt`*ymt;
S3=Tyy-Txy`*inv(Txx)*Txy;
S3i=Tyyi-Txyi`*inv(Txxi)*Txyi;
S3t=Tyyt-Txyt`*inv(Txxt)*Txyt;
F1=((S2-S1)/((N+T-2)*nvar))/(S1/((N-1)*(T-1)*(nvar+1)));
gl11=(N+T-2)*nvar;
gl12=((N-1)*(T-1)*(nvar+1));
probf1=1-probf(F1,gl11,gl12);
F4=((S3-S2)/(N+T-2))/(S2/((N-1)*(T-1)-nvar));
gl41=(N+T-2);
gl42=((N-1)*(T-1)-nvar);
probf4=1-probf(F4,gl41,gl42);
F4i=((S3t-S2)/(N-1))/(S2/((N-1)*(T-1)-nvar));
gl41i=(N-1);
```

9

```
probf4i=1-probf(F4i,gl41i,gl42);
F4t=((S3i-S2)/(T-1))/(S2/((N-1)*(T-1)-nvar));
gl41t=(T-1);
probf4t=1-probf(F4t,gl41t,gl42);
*Print;
print,"F Test for Homogeneous Bi",,
gl11[label='Num DF'] gl12[label='Den DF'] F1[format=10.2 label='F Value']
probf1[format=pvalue6. label='Prob > F'];
print,"F Test for No Fixed Effect (ai=0 and lt=0)",,
gl41[label='Num DF'] gl42[label='Den DF'] F4[format=10.2 label='F Value']
probf4[format=pvalue6. label='Prob > F'];
print,"F Test for No Fixed Effect (ai=0 and lt ne 0)",,
gl41i[label='Num DF'] gl42[label='Den DF'] F4i[format=10.2 label='F Value']
probf4i[format=pvalue6. label='Prob > F'];
print,"F Test for No Fixed Effect (lt=0 and ai ne 0)",,
gl41t[label='Num DF'] gl42[label='Den DF'] F4t[format=10.2 label='F Value']
probf4t[format=pvalue6. label='Prob > F'];
%end;
quit;
%mend tscsreg;
```