0.1 sur: Seemingly Unrelated Regression

sur extends ordinary least squares analysis to estimate system of linear equations with correlated error terms. The seemingly unrelated regression model can be viewed as a special case of generalized least squares.

Syntax

```
> fml <- list ("mu1" = Y1 ~ X1,
               "mu2" = Y2 ~ X2,
               "mu3" = Y3 ~ X3)
> z.out<-zelig(formula = fml, model = "2sls", data = mydata)
> x.out <- setx(z.out)
> s.out <- sim(z.out, x = x.out)
```

Inputs

sur regression specification has at least M equations ($M \geq 2$) corresponding to the dependent variables $(Y_1, Y_2, \ldots, Y_M)$.

- formula:a list whose elements are formulae corresponding to the M equations and their respective dependent and explanatory variables. For example, when there are no constraints on the coefficients:

  ```
  > fml <- list(mu1 = Y1 ~ X1, mu2 = Y2 ~ X2, mu3 = Y3 ~ X3)
  ```

  "mu1" is the label for the first equation with Y1 as the dependent variable and X1 as the explanatory variable. Similarly "mu2" and "mu3" are the labels for the Y2 and Y3 equations.

- tag: Users can also put constraints on the coefficients by using the special function tag. tag takes two parameters. The first parameter is the variable whose coefficient needs to be constrained and the second parameter is label for the constrained coefficient. Each label uniquely identifies the constrained coefficient. For example:

  ```
  > fml <- list(mu1 = Y1 ~ tag(Xc, "constrain1") + X1, mu2 = Y2 ~
  +     tag(Xc, "constrain1") + X2, mu3 = Y3 ~ X3)
  ```

Additional Inputs

sur takes the following additional inputs for model specifications:

- TX: an optional matrix to transform the regressor matrix and, hence, also the coefficient vector (see details). Default is NULL.

- maxiter: maximum number of iterations.

- `tol`: tolerance level indicating when to stop the iteration.
- `rcovformula`: formula to calculate the estimated residual covariance matrix (see details). Default is equal to 1.
- `probdfsys`: use the degrees of freedom of the whole system (in place of the degrees of freedom of the single equation to calculate probability values for the t-test of individual parameters.
- `solvetol`: tolerance level for detecting linear dependencies when inverting a matrix or calculating a determinant. Default is `solvetol`=

  `.Machine\$double.eps.`

- `saveMemory`: logical. Save memory by omitting some calculation that are not crucial for the basic estimate (e.g McElroy's R^2).

Details

The matrix `TX` transforms the regressor matrix (X) by $X* = X \times TX$. Thus, the vector of coefficients is now $b = TX \times b*$ where b is the original(stacked) vector of all coefficients and $b*$ is the new coefficient vector that is estimated instead. Thus, the elements of vector b and $b_i = \sum_j TX_{ij} \times b_j*$. The TX matrix can be used to change the order of the coefficients and also to restrict coefficients (if TX has less columns than it has rows). If iterated (with `maxit`>1), the covergence criterion is

$$\sqrt{\frac{\sum_i (b_{i,g} - b_{i,g-1})^2}{\sum_i b_{i,g-1}^2}} < tol$$

where $b_{i,g}$ is the ith coefficient of the gth iteration step. The formula (`rcovformula` to calculate the estimated covariance matrix of the residuals($\hat{\Sigma}$)can be one of the following (see Judge et al., 1955, p.469): if `rcovformula`= 0:

$$\hat{\sigma_{ij}} = \frac{\hat{e_i}\prime\hat{e_j}}{T}$$

if `rcovformula`= 1 or `rcovformula`='geomean':

$$\hat{\sigma_{ij}} = \frac{\hat{e_i}\prime\hat{e_j}}{\sqrt{(T - k_i) \times (T - k_j)}}$$

if `rcovformula`= 2 or `rcovformula`='Theil':

$$\hat{\sigma_{ij}} = \frac{\hat{e_i}\prime\hat{e_j}}{T - k_i - k_j + tr[X_i(X_i\prime X_i)^{-1}X_i\prime X_j(X_j\prime X_j)^{-1}X_j\prime]}$$

if `rcovformula`= 3 or `rcovformula`='max':

$$\hat{\sigma_{ij}} = \frac{\hat{e_i}'\hat{e_j}}{T - max(k_i, k_j)}$$

If $i = j$, formula 1, 2, and 3 are equal. All these three formulas yield unbiased estimators for the diagonal elements of the residual covariance matrix. If $ineqj$, only formula 2 yields an unbiased estimator for the residual covariance matrix, but it is not necessarily positive semidefinit. Thus, it is doubtful whether formula 2 is really superior to formula 1 (Theil, 1971, p.322).

Examples

Attaching the example dataset:

```
> data(grunfeld)
```

Formula:

```
> formula <- list(mu1 = Ige ~ Fge + Cge, mu2 = Iw ~ Fw + Cw)
```

Estimating the model using `sur`:

```
> z.out <- zelig(formula = formula, model = "sur", data = grunfeld)
```

```
> summary(z.out)
```

Set explanatory variables to their default (mean/mode) values

```
> x.out <- setx(z.out)
```

Simulate draws from the posterior distribution:

```
> s.out <- sim(z.out, x = x.out)
```

```
> summary(s.out)
```

Model

The basic seemingly unrelated regression model assumes that for each individual observation i there are M dependent variables $(Y_{ij}, j = 1, \ldots, M)$ each with its own regression equation:

$$Y_{ij} = X'_{ij}\beta_j + \epsilon_{ij}, \quad \text{for} \quad i = 1, \ldots, N \quad \text{and} \quad j = 1, \ldots, M$$

when X_{ij} is a k-vector of explanatory variables, β_j is the coefficients of the explanatory variables,

- The *stochastic component* is:

  $$\epsilon_{ij} \;\sim\; \mathcal{N}(0, \sigma_{ij})$$

  where within each j equation, $epsilon_{ij}$ is identically and independently distributed for $i = 1, \ldots, M$,

  $$\text{Var}(\epsilon_{ij}) = \sigma_j \quad \text{and} \quad \text{Cov}(\epsilon_{ij}, \epsilon_{i\prime j}) = 0, \quad \text{for} \quad i \neq i\prime, \quad \text{and} \quad j = 1, \ldots, M$$

  However, the error terms for the *ith* observation can be correlated across equations

  $$\text{Cov}(\epsilon_{ij}, \epsilon_{ij\prime}) \neq 0, \quad \text{for} \quad j \neg j\prime, \quad \text{and} \quad i = 1, \ldots, N$$

- The *systematic component* is:

  $$\mu_{ij} = E(Y_i j) = X_{ij}\beta_j, \quad \text{for} \quad i = 1, \ldots, N, \quad \text{and} \quad j = 1, \ldots, M$$

See Also

For information about two stage least squares regression, see Section **??** and `help(2sls)`. For information about three stage least squares regression, see Section **??** and `help(3sls)`.

Quantities of Interest

Output Values

The output of each Zelig command contains useful information which you may view. For example, if you run:

```
z.out <- zelig(formula=fml, model = "sur", data)
```

then you may examine the available information in `z.out` by using `names(z.out)`, see the draws from the posterior distribution of the `coefficients` by using `z.out$coefficients`, and view a default summary of information through `summary(z.out)`. Other elements available through the `$` operator are listed below:

- `rcovest`: residual covariance matrix used for estimation.

- mcelr2: McElroys R-squared value for the system.
- method: Estimation method.
- g: number of equations.
- n: total number of observations.
- k: total number of coefficients.
- ki: total number of linear independent coefficients.
- df: degrees of freedom of the whole system.
- iter: number of iteration steps.
- b: vector of all estimated coefficients.
- t: t values for b.
- se: estimated standard errors of b.
- bt: coefficient vector transformed by TX.
- p: p values for b.
- bcov: estimated covariance matrix of b.
- btcov: covariance matrix of bt.
- rcov: estimated residual covariance matrix.
- drcov: determinant of rcov.
- rcor: estimated residual correlation matrix.
- olsr2: system OLS R-squared value.
- y: vector of all (stacked) endogenous variables.
- x: matrix of all (diagonally stacked) regressors.
- data: data frame of the whole system (including instruments).
- TX: matrix used to transform the regressor matrix.
- rcovformula: formula to calculate the estimated residual covariance matrix.
- probdfsys: system degrees of freedom to calculate probability values?.
- solvetol: tolerance level when inverting a matrix or calculating a determinant.

- `eq`: a list that contains the results that belong to the individual equations.
- `eqnlabel*`: the equation label of the ith equation (from the labels list).
- `formula*`: model formula of the ith equation.
- `n*`: number of observations of the ith equation.
- `k*`: number of coefficients/regressors in the ith equation (including the constant).
- `ki*`: number of linear independent coefficients in the ith equation (including the constant differs from k only if there are restrictions that are not cross equation).
- `df*`: degrees of freedom of the ith equation.
- `b*`: estimated coefficients of the ith equation.
- `se*`: estimated standard errors of b of the ith equation.
- `t*`: t values for b of the ith equation.
- `p*`: p values for b of the ith equation.
- `covb*`: estimated covariance matrix of b of the ith equation.
- `y*`: vector of endogenous variable (response values) of the ith equation.
- `x*`: matrix of regressors (model matrix) of the ith equation.
- `data*`: data frame (including instruments) of the ith equation.
- `fitted*`: vector of fitted values of the ith equation.
- `residuals*`: vector of residuals of the ith equaiton.
- `ssr*`: sum of squared residuals of the ith equation.
- `mse*`: estimated variance of the residuals (mean of squared errors) of the ith equation.
- `s2*`: estimated variance of the residents($\hat{sigma}^2$) of the ith equation.
- `rmse*`: estimated standard error of the reiduals (square root of mse) of the ith equation.
- `s*`: estimated standard error of the residuals ($\hat{\sigma}$) of the ith equation.
- `r2*`: R-squared (coefficient of determination).
- `adjr2*`: adjusted R-squared value.
- `maxiter`: maximum number of iterations.
- `tol`: tolerance level indicating when to stop the iteration.

How to Cite

To cite the *sur* Zelig model:

> Ferdinand Alimadhi, Ying Lu, and Elena Villalon. 2007. ”sur: Seemingly Unrelated Regression” in Kosuke Imai, Gary King, and Olivia Lau, ”Zelig: Everyone’s Statistical Software,”`http://gking.harvard.edu/zelig`

To cite Zelig as a whole, please reference these two sources:

> Kosuke Imai, Gary King, and Olivia Lau. 2007. “Zelig: Everyone’s Statistical Software,” `http://GKing.harvard.edu/zelig`.

> Imai, Kosuke, Gary King, and Olivia Lau. (2008). “Toward A Common Framework for Statistical Analysis and Development.” Journal of Computational and Graphical Statistics, Vol. 17, No. 4 (December), pp. 892-913.

See also

The sur function is adapted from the `systemfit` library (Hamann and Henningsen 2005).

Bibliography

Hamann, J. and Henningsen, A. (2005), *systemfit: Simultaneous Equation Systems in R Package*.