Systemfit package

Olga Mierzwa

16.05.2015

Introduction to systemfit

Many statistical analyses requires bulding models containing systems of structurally related equations. The systemfit package offers the possibilty to estimate systems of linear and non-linear equations within the R language. Package offers commonly use econemtrics techniques for estimation: "ordinary least squares" (OLS), "seemingly unrelated regression" (SUR), and the instrumental variable (IV) methods "two-stage least squares" (2SLS) and "three-stage least squares" (3SLS), where SUR and 3SLS estimations can optionally be iterated. Moreover, the systemfit package enable statistical testing.

Systemfit functions

Systemfit package was created by Arne Henningsen and Jeff D. Hamann and is publically availabe on CRAN. The package contains 32 functions. The most basic function is **systemfit** function and this document will focus on introducing it's main functionality.

Basic systemfit function

The **systemfit** function is used to estimate systems of linear equations using Ordinary Least Squares (OLS), Weighted Least Squares (WLS), Seemingly Unrelated Regression (SUR), Two-Stage Least Squares (2SLS), Weighted TwoStage Least Squares (W2SLS) or Three-Stage Least Squares (3SLS). The systemfit follows lm function functionality in terms of user interface and results. The econometrics formulas used in the systemfit package estimation are described in systemfit: A Package for Estimating Systems of Simultaneous Equations in R

Using systemfit

Systemfit function contains many arguments however there is only one which is always required - **formula**. It can be either a single formula to estimate a single model or a list of formulas to estimate the whole system. Each formula is a standard regression model in R language.

Data for the systemfit are sourced from environment unless speciied by an optional parameter **data** which should be a data frame containing the variables in the model.

```
library(systemfit)
data( "Kmenta" )
eq1 <- consump ~ price + income
eq2 <- consump ~ price + farmPrice + trend
system <- list( demand = eq1, supply = eq2 )
## estimation (be default OLS)
eqRes <- systemfit( system, data=Kmenta )
```

Commonly used argument is **method**. It is a string object that represents the estimation method: "OLS", "WLS", "SUR", "2SLS", "W2SLS", or "3SLS".

```
## iterated SUR estimation
fitsur <- systemfit( system, "SUR", data = Kmenta )
```

In case a modeller would like to estimate "IV" model, she should include **ins** argument. It should be provided in the following form. One-sided model formula specifying instrumental variables or a list of one-sided model formulas if instruments varies between different equations.

In case modeller wishes to impose restrictions on coefficients in the quations she has can express them by:

- **restrict.matrix** an optional m x n matrix where m equals to the number of restrictions, n is the number of all coefficients from all equations. There is a possibility to provide the restrictions in symbolic form.

```
restrict <- "demand_price + supply_price = 0"
fitsurRmat <- systemfit(system, method = "SUR", data = Kmenta, restrict.matrix = restrict)
```

- **restrict.rhs** an optional vector with j elements to impose linear restrictions with default which cosists of j zeros.

```
Rmat <- matrix(0, nrow = 1, ncol = 7)
Rmat[1, 2] <- 1
Rmat[1, 6] <- 1
qvec <- c(0)
fitsurRmatNum <- systemfit(system, method = "SUR", data = Kmenta, restrict.matrix = Rmat, restrict.rhs =
```

- **restrict.regMat** an optional matrix to impose restrictions on the coefficients by post-multiplying the regressor matrix with this matrix.

```
modRegMat <- matrix(0, nrow = 7, ncol = 6)
modRegMat[1:5, 1:5] <- diag(5)
modRegMat[6, 2] <- -1
modRegMat[7, 6] <- 1
fitsurRegMat <- systemfit(system, method = "SUR", data = Kmenta, restrict.regMat = modRegMat)
```

The rest of the possible systemfit parameters are described in package documentation.

Returned objects

Retured object of systemfit returns is a list of the class systemfit that consists of the all results that are producted by the entire system. One object of this list is "eq" that is also a list and contains one object for each estimated equation. Each "eq" objects are of the special class systemfit.equation and corresponds to the results for specific equation.

```
fitols <- systemfit( system, data=Kmenta )
print( fitols )
```

```
##
## systemfit results
## method: OLS
##
```

```
## Coefficients:
## demand_(Intercept)       demand_price      demand_income
##          99.895423          -0.316299           0.334636
## supply_(Intercept)       supply_price   supply_farmPrice
##          58.275431           0.160367           0.248133
##       supply_trend
##           0.248302
```

The **summary** method can be used to compute and print summary results of systemfit.

```
summary(fitols)
```

```
##
## systemfit results
## method: OLS
##
##         N DF     SSR detRCov   OLS-R2 McElroy-R2
## system 40 33 155.883 4.43485 0.709298   0.557559
##
##         N DF     SSR     MSE    RMSE       R2   Adj R2
## demand 20 17 63.3317 3.72539 1.93013 0.763789 0.735999
## supply 20 16 92.5511 5.78444 2.40509 0.654807 0.590084
##
## The covariance matrix of the residuals
##         demand  supply
## demand 3.72539 4.13696
## supply 4.13696 5.78444
##
## The correlations of the residuals
##          demand   supply
## demand 1.000000 0.891179
## supply 0.891179 1.000000
##
##
## OLS estimates for 'demand' (equation 1)
## Model Formula: consump ~ price + income
##
##               Estimate Std. Error  t value   Pr(>|t|)
## (Intercept) 99.8954229  7.5193621 13.28509 2.0906e-10 ***
## price       -0.3162988  0.0906774 -3.48818  0.0028153 **
## income       0.3346356  0.0454218  7.36729 1.0999e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.930127 on 17 degrees of freedom
## Number of observations: 20 Degrees of Freedom: 17
## SSR: 63.33165 MSE: 3.725391 Root MSE: 1.930127
## Multiple R-Squared: 0.763789 Adjusted R-Squared: 0.735999
##
##
## OLS estimates for 'supply' (equation 2)
## Model Formula: consump ~ price + farmPrice + trend
##
```

```
##               Estimate Std. Error t value   Pr(>|t|)
## (Intercept) 58.2754312 11.4629099 5.08383 0.00011056 ***
## price        0.1603666  0.0948839 1.69013 0.11038810
## farmPrice    0.2481333  0.0461879 5.37226 6.2274e-05 ***
## trend        0.2483023  0.0975178 2.54623 0.02156713 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.405087 on 16 degrees of freedom
## Number of observations: 20 Degrees of Freedom: 16
## SSR: 92.551058 MSE: 5.784441 Root MSE: 2.405087
## Multiple R-Squared: 0.654807 Adjusted R-Squared: 0.590084
```

In order to reduce the amount of the printed outcome use the optional arguments residCov and equations. Argument **residCov** controls printing the covariance matrix and the correlation matrix of the residuals. Argument **equations** controls the equations results.

```
summary(fitols, residCov = FALSE, equations = FALSE)
```

```
##
## systemfit results
## method: OLS
##
##         N DF     SSR detRCov   OLS-R2 McElroy-R2
## system 40 33 155.883 4.43485 0.709298   0.557559
##
##         N DF     SSR     MSE    RMSE       R2   Adj R2
## demand 20 17 63.3317 3.72539 1.93013 0.763789 0.735999
## supply 20 16 92.5511 5.78444 2.40509 0.654807 0.590084
##
##
## Coefficients:
##                       Estimate Std. Error  t value   Pr(>|t|)
## demand_(Intercept) 99.8954229  7.5193621 13.28509 2.0906e-10 ***
## demand_price       -0.3162988  0.0906774 -3.48818 0.00281529 **
## demand_income       0.3346356  0.0454218  7.36729 1.0999e-06 ***
## supply_(Intercept) 58.2754312 11.4629099  5.08383 0.00011056 ***
## supply_price        0.1603666  0.0948839  1.69013 0.11038810
## supply_farmPrice    0.2481333  0.0461879  5.37226 6.2274e-05 ***
## supply_trend        0.2483023  0.0975178  2.54623 0.02156713 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Testing linear restrictions

Systemfit allows to test linear restrictions by an F test, two Wald tests and an LR test. Testing can be done using method **linearHypothesis** for the F test and Wald tests, **lrtest** method for LR tests.

Method linearHypothesis

Linear.hypothesis needs unrestricted regression from systemfit as it's first argument. The second and third arguments are the restriction matrix and the optional vector. The optional argument test is a character string, "FT", "F", or "Chisq", specifying the test to conduct.

```
linearHypothesis(fitsur, Rmat, qvec, test = "FT")
```

```
## Linear hypothesis test (Theil's F test)
##
## Hypothesis:
## demand_price  + supply_farmPrice = 0
##
## Model 1: restricted model
## Model 2: fitsur
##
##   Res.Df Df      F Pr(>F)
## 1     34
## 2     33  1 0.9322 0.3413
```

```
linearHypothesis(fitsur, Rmat, qvec, test = "F")
```

```
## Linear hypothesis test (F statistic of a Wald test)
##
## Hypothesis:
## demand_price  + supply_farmPrice = 0
##
## Model 1: restricted model
## Model 2: fitsur
##
##   Res.Df Df      F Pr(>F)
## 1     34
## 2     33  1 0.6092 0.4407
```

```
linearHypothesis(fitsur, Rmat, qvec, test = "Chisq")
```

```
## Linear hypothesis test (Chi^2 statistic of a Wald test)
##
## Hypothesis:
## demand_price  + supply_farmPrice = 0
##
## Model 1: restricted model
## Model 2: fitsur
##
##   Res.Df Df  Chisq Pr(>Chisq)
## 1     34
## 2     33  1 0.6092     0.4351
```

Method lrtest

All arguments of the lrtest method for systemfit should be fitted model objects from systemfit.

```
lrtest(fitsurRmat, fitsur)
```

```
## Warning in lrtest.systemfit(fitsurRmat, fitsur): model '2' has a smaller
## log-likelihood value than the more restricted model '1'
```

```
## Likelihood ratio test
##
## Model 1: fitsurRmat
## Model 2: fitsur
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   9 -51.245
## 2  10 -51.614  1 0.7393     0.3899
```

Tests comparison

All tests check the same hypothesis and suggest the same decision: The null hypothesis cannot be rejected at any reasonable level of significance.

Tests	Results (Pr)
FT	0.3413
F	0.4407
Chisq	0.4351
LR	0.3899