

Package ‘ssfa’

June 10, 2015

Type Package

Title Spatial Stochastic Frontier Analysis

Version 1.1

Date 2015-06-09

Author Elisa Fusco, Francesco Vidoli

Maintainer Elisa Fusco <fusco_elisa@libero.it>

Description Spatial Stochastic Frontier Analysis (SSFA) is an original method for controlling the spatial heterogeneity in Stochastic Frontier Analysis (SFA) models, for cross-sectional data, by splitting the inefficiency term into three terms: the first one related to spatial peculiarities of the territory in which each single unit operates, the second one related to the specific production features and the third one representing the error term.

Depends Matrix, maxLik, spdep, sp

License GPL-3

Suggests R.rsp

VignetteBuilder R.rsp

NeedsCompilation no

Repository CRAN

Date/Publication 2015-06-10 01:02:51

R topics documented:

ssfa-package	2
eff.ssfa	2
fitted.ssfa	3
Italian_W	4
L_hNV	5
L_hNV_rho	5
plot_fitted	6
plot_moran	7
residuals.ssfa	8
ssfa	9

SSFA_example_data	11
summary	12
u.ssfa	13

Index	15
--------------	-----------

ssfa-package	<i>Spatial Stochastic Frontier models</i>
--------------	---

Description

The package implements the Spatial Stochastic Frontier model for cross-sectional data introduced by Fusco and Vidoli (2013). The method controls spatial heterogeneity in SFA models by splitting the inefficiency term into three parts: the first one related to spatial peculiarities of the territory in which each single unit operates, the second one related to the specific production features and the third one representing the error term.

Details

Package: ssfa
 Type: Package
 Version: 1.1
 Date: 2015-06-09
 License: GPL-3

Author(s)

Elisa Fusco, Francesco Vidoli
 Maintainer: Elisa Fusco <fusco_elisa@libero.it>

References

Fusco, E. and Vidoli, F. (2013). *Spatial stochastic frontier models: controlling spatial global and local heterogeneity*, International Review of Applied Economics, 27(5) 679-694.

eff.ssfa	<i>SSFA efficiency</i>
----------	------------------------

Description

This function returns the technical efficiency of each producer (without local spatial effects) calculated by the Battese and Coelli (1988) formulation modified by using an autoregressive specification in the inefficiency term u .

Usage

```
eff.ssfa(object, ...)
```

Arguments

```
object      an object of class ssfa.  
...        further arguments for methods.
```

Value

Technical efficiency of each producer (without local spatial effects).

References

Battese, G. E., and T. J. Coelli (1988). *Prediction of Firm-level Technical Efficiencies with a Generalized Frontier Production Function and Panel Data*. *Journal of Econometrics* 38(3): 387-399.

Fusco, E. and Vidoli, F. (2013). *Spatial stochastic frontier models: controlling spatial global and local heterogeneity*, *International Review of Applied Economics*, 27(5) 679-694.

Kumbhakar, S. C., and C. A. K. Lovell (2000). *Stochastic Frontier Analysis*, Cambridge University Press.

See Also

[u.ssfa](#)

Examples

```
library(ssfa)  
data(SSFA_example_data)  
data(Italian_W)  
ssfa <- ssfa(log_y ~ log_x, data = SSFA_example_data, data_w=Italian_W,  
            form = "production", par_rho=TRUE)  
eff <- eff.ssfa(ssfa)
```

fitted.ssfa

SSFA fitted values

Description

This function returns the fitted values of the original data used to estimate the SSFA model.

Usage

```
## S3 method for class 'ssfa'  
fitted(object, ...)
```

Arguments

object an object of class ssfa.
... further arguments for methods.

Examples

```
library(ssfa)
data(SSFA_example_data)
data(Italian_W)
ssfa <- ssfa(log_y ~ log_x, data = SSFA_example_data, data_w=Italian_W,
             form = "production", par_rho=TRUE)
fitted.ssfa(ssfa)
```

Italian_W

Italian provinces spatial weights matrix example

Description

This is an example dataset that contains the 107 Italian provinces contiguity matrix (year 2008).

Usage

```
data(Italian_W)
```

Format

A data frame with 107 x 107 row-standardized distances between observations (Italian provinces).

References

<http://www.istat.it/it/archivio/104317#confini>.

Examples

```
data(Italian_W)
```

L_hNV	<i>SFA half-normal log likelihood function</i>
-------	--

Description

This function is used to estimate the parameters of the classical SFA model where half-normal distribution of inefficiency term is assumed.

Usage

```
L_hNV(p, y = y, X = X, sc = sc)
```

Arguments

p	a vector with the parameters to be estimated.
y	the dependent variable.
X	the model matrix.
sc	specifies the form of the frontier model (-1 = cost, 1 = production).

Value

Value of the SFA log likelihood function.

L_hNV_rho	<i>SSFA half-normal log likelihood function</i>
-----------	---

Description

This function is used to estimate the parameters of the SSFA model where half-normal distribution of inefficiency term is assumed.

Usage

```
L_hNV_rho(p, y = y, X = X, sc = sc, w = w, sigma2_sar = sigma2_sar)
```

Arguments

p	a vector with the parameters to be estimated.
y	the dependent variable.
X	the model matrix.
sc	specifies the form of the frontier model (-1 = cost, 1 = production).
w	the spatial weight matrix.
sigma2_sar	is the variance of the spatial correlated part of the inefficiency term estimated into <code>ssfa.fit</code> function.

Value

Value of the SSFA log likelihood function.

Note

Please note that `sigmau2_sar` is not a free parameter because it is estimated into the `ssfa.fit` function.

See Also

[ssfa](#)

plot_fitted

SSFA plot

Description

This function allows to plot the data and the fitted values obtained by SSFA model.

Usage

```
plot_fitted(x, y, object, xlab, ylab, main, ...)
```

Arguments

<code>x</code>	the x coordinates of points in the plot.
<code>y</code>	the y coordinates of points in the plot.
<code>object</code>	an object of class <code>ssfa</code> .
<code>xlab</code>	a title for the x axis.
<code>ylab</code>	a title for the y axis.
<code>main</code>	an overall title for the plot.
<code>...</code>	arguments to be passed to methods, such as graphical parameters (see par).

See Also

[plot](#)

Examples

```

library(ssfa)
data(SSFA_example_data)
data(Italian_W)

#### SFA and SSFA comparison
sfa <- ssfa(log_y ~ log_x, data = SSFA_example_data, data_w=Italian_W,
            form = "production", par_rho=FALSE)
ssfa <- ssfa(log_y ~ log_x, data = SSFA_example_data, data_w=Italian_W,
            form = "production", par_rho=TRUE)

sfa_fitted <- fitted.ssfa(sfa)
plot_fitted(SSFA_example_data$log_x, SSFA_example_data$log_y, ssfa)
lines(sort(SSFA_example_data$log_x), sfa_fitted[order(SSFA_example_data$log_x)],col="red")

```

plot_moran

SSFA residuals Moran plot

Description

This function allows to plot the residuals of the object against their spatially lagged values, augmented by reporting the summary of influence measures for the linear relationship between the data and the lag.

Usage

```
plot_moran(x, main, xlab, ylab, labels, listw, ...)
```

Arguments

<code>x</code>	an object of class <code>ssfa</code> .
<code>main</code>	an overall title for the plot.
<code>xlab</code>	a label for the x axis.
<code>ylab</code>	a label for the y axis.
<code>labels</code>	character labels for points with high influence measures, if set to <code>FALSE</code> , no labels are plotted for points with large influence.
<code>listw</code>	a <code>listw</code> object from <code>nb2listw</code> (see nb2listw).
<code>...</code>	arguments to be passed to methods, such as graphical parameters (see par).

References

Anselin, L. (1995). *Local indicators of spatial association*, *Geographical Analysis*, 27, 93-115.

Anselin, L. (1996). *The Moran scatterplot as an ESDA tool to assess local instability in spatial association*. pp. 111-125 in M. M. Fischer, H. J. Scholten and D. Unwin (eds) *Spatial analytical perspectives on GIS*, London, Taylor and Francis.

See Also[moran.plot](#)**Examples**

```

library(ssfa)
data(SSFA_example_data)
data(Italian_W)

#### SFA and SSFA comparison ####
sfa <- ssfa(log_y ~ log_x, data = SSFA_example_data, data_w=Italian_W,
            form = "production", par_rho=FALSE)
ssfa <- ssfa(log_y ~ log_x, data = SSFA_example_data, data_w=Italian_W,
             form = "production", par_rho=TRUE)

moran.test(residuals.ssfa(sfa), sfa$list_w)
moran.test(residuals.ssfa(ssfa), ssfa$list_w)

plot_moran(sfa, listw=sfa$list_w)
plot_moran(ssfa, listw=ssfa$list_w)

```

`residuals.ssfa`*SSFA residuals*

Description

This function returns the residuals of the fitted SSFA model.

Usage

```

## S3 method for class 'ssfa'
residuals(object, ...)

```

Arguments

```

object      an object of class ssfa.
...         further arguments for methods.

```

Examples

```

library(ssfa)
data(SSFA_example_data)
data(Italian_W)
ssfa <- ssfa(log_y ~ log_x, data = SSFA_example_data,
             data_w=Italian_W, form = "production", par_rho=TRUE)
residuals.ssfa(ssfa)

```

ssfa

Spatial stochastic frontier estimation

Description

This function estimates the Spatial Stochastic Frontier model introduced by Fusco and Vidoli (2013) in the following form:

$$\log(y_i) = \log(f(x_i; \beta_i)) + v_i - u_i$$

$$u_i = \rho \sum_i w_i u_i + \tilde{u}_i$$

where y_i are the outputs, x_i the inputs, v_i the stochastic noise, u_i the inefficiency term, ρ the spatial lag, w_i a standardized row of the spatial weights matrix and \tilde{u}_i the stochastic noise of the inefficiency term.

Usage

```
ssfa(formula, data = NULL, data_w = NULL, intercept = TRUE, pars = NULL, par_rho = TRUE,
      form = "cost")
```

Arguments

formula	an object of class <code>formula</code> (or one that can be coerced to that class): a symbolic description of the model to be fitted.
data	an optional data frame containing the variables in the model.
data_w	a data frame containing the spatial weight matrix.
intercept	logical. If true the model includes intercept.
pars	initial values for the parameters to be estimated.
par_rho	logical. If true the function estimates the Spatial Stochastic Frontier (SSFA) otherwise the classical Stochastic Frontier (SFA).
form	specifies the form of the frontier model as "cost" or "production".

Value

ssfa returns the following objects of class `ssfa`:

y	the dependent variable.
x	the covariates.
X	the model matrix.
coef	the estimated coefficients.
sc	the form of the frontier model estimated (-1 = cost, 1 = production).
hess	a symmetric matrix giving an estimate of the Hessian at the solution found.
logLik	the value of the log likelihood function.

ols	the linear model for the LR-test.
sigmau2	the estimation of sigmau2 (only if par_rho=FALSE): value of inefficiency variance.
sigmau2_dmu	the estimation of sigmau2_dmu (only if par_rho=TRUE): value of the part of the inefficiency variance due to DMU's specificities.
sigmau2_sar	the estimation of sigmau2_sar: value of the part of the inefficiency variance due to the spatial correlation.
sigmav2	the estimation of sigmav2: value of the stochastic error variance.
sigma2	the estimation of sigma2: value of the total variance.
rho	the estimation of the spatial lag parameter rho.
fun	the distribution of the inefficiency term u.
list_w	a listw object from nb2listw (See nb2listw).

Note

NOTE 1: In this version the distribution of the inefficiency term u is only "half-normal".

NOTE 2: The method used to maximize the log likelihood function is the Newton-Raphson. Please see the R function `maxNR` of the `maxLik` package for details (Henningsen and Toomet (2011)).

NOTE 3: Please note that the classical SFA inefficiency variance `sigmau2`, in the SSFA, is decomposed into `sigmau2_dmu` and `sigmau2_sar`, respectively the part of inefficiency variance due to DMU's specificities and to the spatial dependence, *i.e.* $\text{sigmau2} = \text{sigmau2_dmu} + \text{sigmau2_sar}$ and consequently the total variance is given by $\text{sigma2} = \text{sigmau2_dmu} + \text{sigmau2_sar} + \text{sigmav2}$.

Author(s)

Fusco E. and Vidoli F.

References

- Battese, G. E., and T. J. Coelli (1995). *A Model for Technical Inefficiency Effects in a Stochastic Frontier Production Function for Panel Data*. *Empirical Economics* 20(2): 325-332.
- Fusco, E. and Vidoli, F. (2013). *Spatial stochastic frontier models: controlling spatial global and local heterogeneity*, *International Review of Applied Economics*, 27(5) 679-694.
- Kumbhakar, S. C., and C. A. K. Lovell (2000). *Stochastic Frontier Analysis*, Cambridge University Press.
- Henningsen, A. and Toomet, O. (2011). *maxLik: A package for maximum likelihood estimation in R*. *Computational Statistics* 26(3), 443-458.

Examples

```
library(ssfa)
data(SSFA_example_data)
data(Italian_W)
ssfa <- ssfa(log_y ~ log_x, data = SSFA_example_data,
```

```
data_w=Italian_W, form = "production", par_rho=TRUE)

### SSFA total variance decomposition
sigma2 = ssfa$sigma2_dmu + ssfa$sigma2_sar + ssfa$sigma2_v
sigma2
ssfa$sigma2
```

SSFA_example_data	<i>Example dataset</i>
-------------------	------------------------

Description

The dataset contains the simulated data used by Fusco and Vidoli (2013) to test the model. Data Generating Process (DGP) follows the construction criteria proposed by Banker and Natarajan (2008), also used by Johnson and Kuosmanen (2011), with the addition of a strong spatial correlation in the inefficiency term through a spatial lag parameter and a contiguity matrix (107 Italian provinces contiguity matrix, year 2008).

Usage

```
data(SSFA_example_data)
```

Format

A data frame with 107 observations (Italian provinces) and 2 variables:

DMU the Decision Making Unit name.

log_x the input vector (already in logarithmic form).

log_y the output vector (already in logarithmic form).

References

Banker, R., and R. Natarajan (2008). *Evaluating Contextual Variables Affecting Productivity using Data Envelopment Analysis*. *Operations Research* 56 (1): 48-58.

Johnson, A., and T. Kuosmanen (2011). *One-stage Estimation of the Effects of Operational Conditions and Practices on Productive Performance: Asymptotically Normal and Efficient, Root-n Consistent StoNEZD Method*. *Journal of Productivity Analysis* 36:219-230.

Examples

```
data(SSFA_example_data)
```

summary

*SSFA summaries***Description**

The function `print.ssfa` is used to display the values of SFA and SSFA estimated coefficients. In particular:

- for SFA the function displays the Intercept, the regressors beta coefficients, the inefficiency variance `sigmau2`, the stochastic error variance `sigmav2` and the total variance `sigma2`;
- for SSFA the function displays, in addition, the decomposition of the inefficiency variance into `sigmau2_dmu` and `sigmau2_sar`, respectively the part of inefficiency variance due to DMU's specificities and to the spatial dependence, and finally, the spatial lag parameter `rho`.

The function `summary.ssfa` is used to display the summary results of SFA and SSFA. In particular:

- for SFA the summary shows the estimation of SFA coefficients (Intercept, beta coefficients, `sigmau2` and `sigmav2`) and others useful information as the total variance `sigma2`, the inefficiency parameter `Lambda` (`sigmau/sigmav`), the Moran I statistic, the mean of efficiency, the LR-test and the AIC values;
- for SSFA the summary shows, in addition, the decomposition of the inefficiency variance into `sigmau2_dmu` and `sigmau2_sar` and the spatial lag parameter `rho`.

Usage

```
## S3 method for class 'ssfa'
print(x, ...)
## S3 method for class 'ssfa'
summary(object, ...)
```

Arguments

<code>x</code>	an object of class <code>ssfa</code> .
<code>object</code>	an object of class <code>ssfa</code> .
<code>...</code>	further arguments for methods.

Note

Please note that the classical SFA inefficiency variance `sigmau2`, in the SSFA, is decomposed into `sigmau2_dmu` and `sigmau2_sar`, respectively the part of inefficiency variance due to DMU's specificities and to the spatial dependence, *i.e.* $\text{sigmau2} = \text{sigmau2_dmu} + \text{sigmau2_sar}$ and consequently the total variance is given by $\text{sigma2} = \text{sigmau2_dmu} + \text{sigmau2_sar} + \text{sigmav2}$.

References

- Anselin, L. (1995). *Local indicators of spatial association*, *Geographical Analysis*, 27, 93-115.
- Fusco, E. and Vidoli, F. (2013). *Spatial stochastic frontier models: controlling spatial global and local heterogeneity*, *International Review of Applied Economics*, 27(5) 679-694.
- Kumbhakar, S. C., and C. A. K. Lovell (2000). *Stochastic Frontier Analysis*, Cambridge University Press.

Examples

```
library(ssfa)
data(SSFA_example_data)
data(Italian_W)
ssfa <- ssfa(log_y ~ log_x, data = SSFA_example_data,
             data_w=Italian_W, form = "production", par_rho=TRUE)

print(ssfa)
summary(ssfa)
```

u.ssfa	<i>SSFA inefficiency</i>
--------	--------------------------

Description

This function returns the specific inefficiency of each producer (without local spatial effects) calculated by the Jondrow et al. (JLMS) (1982) formulation modified by using an autoregressive specification in the inefficiency term.

Usage

```
u.ssfa(object, ...)
```

Arguments

object	an object of class <code>ssfa</code> .
...	further arguments for methods.

Value

Inefficiency of each producer (without local spatial effects).

References

Fusco, E. and Vidoli, F. (2013) *Spatial stochastic frontier models: controlling spatial global and local heterogeneity*, International Review of Applied Economics, 27(5) 679-694. Kumbhakar, S. C., and C. A. K. Lovell. (2000) *Stochastic Frontier Analysis*, Cambridge University Press.

Jondrow, J., C. A. Knox Lovell, I. S. Materov, and P. Schmidt. (1982). *On the Estimation of Technical Inefficiency in the Stochastic Frontier Production Function Model*. Journal of Econometrics 19 (2-3): 233-238.

See Also

[eff.ssfa](#)

Examples

```
library(ssfa)
data(SSFA_example_data)
data(Italian_W)
ssfa <- ssfa(log_y ~ log_x, data = SSFA_example_data,
             data_w=Italian_W, form = "production", par_rho=TRUE)
ineff <- u.ssfa(ssfa)
```

Index

*Topic **datasets**

Italian_W, [4](#)

SSFA_example_data, [11](#)

*Topic **package**

ssfa-package, [2](#)

eff.ssfa, [2](#), [13](#)

fitted.ssfa, [3](#)

Italian_W, [4](#)

L_hNV, [5](#)

L_hNV_rho, [5](#)

maxNR, [10](#)

moran.plot, [8](#)

nb2listw, [7](#), [10](#)

par, [6](#), [7](#)

plot, [6](#)

plot_fitted, [6](#)

plot_moran, [7](#)

print.ssfa(summary), [12](#)

print.summary.ssfa(summary), [12](#)

residuals.ssfa, [8](#)

ssfa, [6](#), [9](#)

ssfa-package, [2](#)

SSFA_example_data, [11](#)

summary, [12](#)

summary.ssfa(summary), [12](#)

u.ssfa, [3](#), [13](#)