Package ‘gstar’

June 28, 2019

Type Package
Title Generalized Space-Time Autoregressive Model
Version 0.1.0
Depends R (>= 2.10), ggplot2
Imports dplyr, xts, zoo, reshape2
License GPL-2 | GPL-3
Encoding UTF-8
LazyData true
RoxygenNote 6.1.1
Suggests testthat
NeedsCompilation no
Author Ahmad Zaenal [aut, cre],
Fiqry Revadiansyah [aut]
Maintainer Ahmad Zaenal <ahmadzaenal125@gmail.com>
Repository CRAN
Date/Publication 2019-06-28 15:10:06 UTC

R topics documented:

  gstar                              ........................................ 2
  Loc                                ........................................ 4
  LocationCPI                        ........................................ 5
  performance                        ........................................ 5
  plot.gstar                         ........................................ 6
  predict.gstar                      ........................................ 6
  summary.gstar                      ........................................ 7

Index 8
Description

gstar function return the parameter estimation of Generalized Space-Time Autoregressive Model.

Usage

gstar(x, weight, p = 1, d = 0, est = "OLS")

Arguments

x          a dataframe, matrix or xts or ts object that contain time series data.
weight     a spatial weight ncol(x) * ncol(x) with diagonal = 0.
p          an autoregressive order, value must be greater than 0.
d          a lag differencing order, value must be greater than 0.
est        estimation method, currently only OLS available, another estimation will be added later.

Value

gstar returns output similar to lm, the detail are shown in the following list :

  • coefficients - a named vector of coefficients.
  • AIC - A version of Akaike’s An Information Criterion (the calculation is similar to aic in lm method )

References


See Also

summary for summarize the model that has been built. Also use predict to predict model to testing or new data.
Examples

```r
library(gstar)
library(xts)
data("LocationCPI")

#-----Use data with xts object-----#
x = xts(LocationCPI[, -1], order.by = as.Date(LocationCPI[, 1]))

s <- round(nrow(x) * 0.8) ## split into training and testing (80:20)
x_train <- x[1:s, ]
x_test <- x[-c(1:s), ]

weight = matrix(c(0, 1, 1, 1,
                  1, 0, 1, 1,
                  1, 1, 0, 1,
                  1, 1, 1, 0), ncol = 4, nrow = 4)  # create the uniform weight.

weight = weight/(ncol(x) - 1)  # the sum of weight is equal to 1 every row.

fit <- gstar(x_train, weight = weight,
             p = 1, d = 0, est = "OLS")
summary(fit)

performance(fit)
performance(fit, x_test)  ## to check the performance with testing data

predict(fit, n = 10)  # forecast 10 data ahead

plot(fit)
plot(fit, n_predict = 10)  # plot with 10 forecasting data
plot(fit, testing = x_test)

#-----Use dataframe or matrix-----#
x2 <- LocationCPI
x2$Date <- NULL  # remove the date column

data(Loc)
dst <- as.matrix(dist(Loc[, -1], diag = TRUE, upper = TRUE))
dst1 <- matrix(0, nrow = nrow(dst), ncol = ncol(dst))

for(i in 1:nrow(dst)) {
  for(j in 1:ncol(dst)){
    if(j == i) next
    dst1[i, j] <- sum(dst[i, -j])/sum(dst[i, ])
  }
}

weight_inverse_distance <- matrix(0, nrow =
nrow(dst), ncol = ncol(dst))

for(i in 1:nrow(dst)) {
  for(j in 1:ncol(dst)){
    if(j == i) next
    weight_inverse_distance[i, j] <- sum(dst[i, j])/sum(dst[i,])
  }
}

fit_inverse_distance <- gstar(x, weight =
  weight_inverse_distance, p = 2, d = 1, est = "OLS")

summary(fit_inverse_distance)
performance(fit_inverse_distance)
predict(fit_inverse_distance)
plot(fit_inverse_distance)

----

| Loc                  | Coordinate of several region In Indonesia |

Description

A dataset containing the coordinate several region In Indonesia i.e Semarang, Surakarta, Tegal and Purwokerto.

Usage

data(Loc)

Format

A data frame with 4 rows and 3 variables:

City  Name of region/city
latitude The latitude coordinate of each location
longitude The longitude coordinate of each location
LocationCPI

Consumer Price Index (CPI) in several region In Indonesia

Description
A dataset containing the Consumer Price Index (CPI) in several region In Indonesia i.e Semarang, Surakarta, Tegal and Purwokerto, it is time series data with monthly periodicity from Jan 2006 to Sep 2014

Usage
data(LocationCPI)

Format
A time series data frame with 105 rows and 5 variables:
- Date  date of CPI, monthly
- Purwokerto  The CPI of Purwokerto region
- Surakarta  The CPI of Purwokerto region
- Semarang  The CPI of Purwokerto region
- Tegal  The CPI of Purwokerto region

Source
https://www.bps.go.id/

performance

Calculate performance of prediction or forecasting

Description
Calculate performance of prediction or forecasting

Usage
performance(object, testing = NULL, ...)

Arguments
- object  an object of class "gstar".
- testing  a dataframe or matrix or xts object that contain testing data. Please be noted, if you fill the differencing order in the model estimation, you do not need difference your data anymore because we already cover that in this function
- ...  further arguments passed to or from other methods.
Value

- MSE for all data - Mean Square Error for all the data combined
- MSE for each location - Mean Square Error for each spatial location
- MAPE for all data - Mean Absolute Percentage Error for all the data combined
- MAPE for each location - Mean Absolute Percentage Error for each spatial location

plot.gstar  Plotting the gstar object

Description

plotting the gstar object

Usage

```r
## S3 method for class 'gstar'
plot(x, testing = NULL, n_predict = NULL, ...)
```

Arguments

- **x**: an object of class "gstar".
- **testing**: The testing data to be plotted.
- **n_predict**: The number of steps ahead for which prediction is required.
- **...**: further arguments passed to or from other methods.

predict.gstar  Predicting the gstar object

Description

Predicted values based on gstar object object

Usage

```r
## S3 method for class 'gstar'
predict(object, n = NULL, ...)
```

Arguments

- **object**: an object of class "gstar".
- **n**: The number of steps ahead for which prediction is required.
- **...**: further arguments passed to or from other methods.
Summary of Generalized Space-Time Autoregressive Fits

This function is similar to summary of "lm" or "glm" object.

Usage

```r
## S3 method for class 'gstar'
summary(object, ...)
```

Arguments

- `object`: an object of class "gstar".
- `...`: further arguments passed to or from other methods.
  - `coefficients`: a named vector of coefficients.
  - `AIC`: A version of Akaike's An Information Criterion.
Index

*Topic datasets
  Loc, 4
  LocationCPI, 5

gstar, 2

Loc, 4
LocationCPI, 5

performance, 5
plot.gstar, 6
predict, 2
predict.gstar, 6

summary, 2
summary.gstar, 7