Package ‘bifurcatingr’

February 15, 2021

Title Bifurcating Autoregressive Models

Version 1.0.0

Description Estimation of bifurcating autoregressive models of any order, p, \( \text{BAR}(p) \) as well as several types of bias correction for the least squares estimators of the autoregressive parameters as described in Zhou and Basawa (2005) <doi:10.1016/j.spl.2005.04.024> and Elbayoumi and Mostafa (2020) <doi:10.1002/sta4.342>. Currently, the bias correction methods supported include bootstrap (single, double and fast-double) bias correction and linear-bias-function-based bias correction. Functions for generating and plotting bifurcating autoregressive data from any \( \text{BAR}(p) \) model are also included.

License GPL-3

Encoding UTF-8

LazyData true

Imports graphics (>= 4.0.0), igraph (>= 1.2.5), MASS (>= 7.3.0)

RoxygenNote 7.1.1

Suggests knitr, rmarkdown

Depends R (>= 2.10)

NeedsCompilation no

Author Tamer Elbayoumi [aut, cre] (<https://orcid.org/0000-0001-9071-7766>), Sayed Mostafa [aut] (<https://orcid.org/0000-0002-8113-702X>)

Maintainer Tamer Elbayoumi <tmebayoumi@ncat.edu>

Repository CRAN

Date/Publication 2021-02-15 17:10:02 UTC

R topics documented:

  bfa.boot1.ls ....................................................... 2
  bfa.boot2.ls ....................................................... 3
  bfa.boot2fast.ls .................................................. 4
  bfa.lbc.ls .......................................................... 5
Description

This function performs single bootstrapping of the least squares estimators of the autoregressive coefficients in a bifurcating autoregressive (BAR) model of any order \( p \) as described in Elbayoumi & Mostafa (2020).

Usage

\[
\text{bfa.boot1.ls}(z, p, \text{burn} = 5, \text{B}, \text{boot.est} = \text{TRUE}, \text{boot.data} = \text{FALSE})
\]

Arguments

- **z**: a numeric vector containing the tree data
- **p**: an integer determining the order of bifurcating autoregressive model to be fit to the data
- **burn**: number of tree generations to discard before starting the bootstrap sample (replicate)
- **B**: number of bootstrap samples (replicates)
- **boot.est**: a logical that determines whether the bootstrapped least squares estimates of the autoregressive coefficients should be returned. Defaults to TRUE.
- **boot.data**: a logical that determines whether the bootstrap samples should be returned. Defaults to FALSE.

Value

- **boot.est**: a matrix containing the bootstrapped least squares estimates of the autoregressive coefficients
- **boot.data**: a matrix containing the bootstrap samples used

References

Examples

```r
z <- bfa.tree.gen(31, 1, 1, 1, 0.5, 0.5, 0, 10, c(0.7))
bfa.boot1.ls(z, p=1, B=999)
```

---

### bfa.boot2.ls

**Double Bootstrap of Least Squares Estimators of BAR(p) Models**

**Description**

This function performs double bootstrapping of the least squares estimators of the autoregressive coefficients in a bifurcating autoregressive (BAR) model of any order \( p \) as described in Elbayoumi & Mostafa (2020).

**Usage**

```r
bfa.boot2.ls(z, p, burn = 5, B1, B2)
```

**Arguments**

- **z**: a numeric vector containing the tree data
- **p**: an integer determining the order of bifurcating autoregressive model to be fit to the data
- **burn**: number of tree generations to discard before starting the bootstrap sample (replicate)
- **B1**: number of bootstrap samples (replicates) used in first round of bootstrapping
- **B2**: number of bootstrap samples (replicates) used in second round of bootstrapping

**Value**

- **boot.est**: a matrix containing the first-stage bootstrapped least squares estimates of the autoregressive coefficients
- **boot2**: a matrix containing the second-stage bootstrapped least squares estimates of the autoregressive coefficients

**References**


**Examples**

```r
z <- bfa.tree.gen(31, 1, 1, 1, 0.5, 0.5, 0, 10, c(0.7))
bfa.boot2.ls(z, p=1, B1=99, B2=9)
```
bfa.boot2fast.ls  

**Fast Double Bootstrap of Least Squares Estimators of BAR(p) Models**

**Description**

This function performs fast double bootstrapping of the least squares estimators of the autoregressive coefficients in a bifurcating autoregressive (BAR) model of any order $p$ as described in Elbayoumi & Mostafa (2020).

**Usage**

```
bfa.boot2fast.ls(z, p, burn = 5, B)
```

**Arguments**

- **z**: a numeric vector containing the tree data
- **p**: an integer determining the order of bifurcating autoregressive model to be fit to the data
- **burn**: number of tree generations to discard before starting the bootstrap sample (replicate)
- **B**: number of bootstrap samples (replicates) used in first round of bootstrapping

**Value**

- **boot.est**: a matrix containing the first-stage bootstrapped least squares estimates of the autoregressive coefficients
- **boot2**: a matrix containing the second-stage bootstrapped least squares estimates of the autoregressive coefficients

**References**


**Examples**

```
z <- bfa.tree.gen(31, 1, 1, 1, 0.5, 0.5, 0, 10, c(0.7))
bfa.boot2fast.ls(z, p=1, B=499)
```
**Description**

This function performs bias correction on the least squares estimators of the autoregressive coefficients in a BAR(p) model based on the assumption that the bias of the least squares estimator is approximately linear as a function of the parameter as described in Elbayoumi & Mostafa (2020).

**Usage**

```r
bfa.lbc.ls(z, p)
```

**Arguments**

- `z`: a numeric vector containing the tree data
- `p`: an integer determining the order of bifurcating autoregressive model to be fit to the data

**Value**

- `coef.lbc`: linear-bias-function-based bias-corrected least squares estimates of the autoregressive coefficients

**References**


**Examples**

```r
z <- bfa.tree.gen(127, 1, 1, 1, 0.5, 0.5, 0, 10, c(0.7))
bfa.lbc.ls(z, p=1)
```

**Description**

This function performs Least Squares estimation of bifurcating autoregressive (BFA) models of any order as described in Zhou & Basawa (2005).
Usage

bfa.ls(
  z,
  p,
  x.data = FALSE,
  y.data = FALSE,
  resid = FALSE,
  error.cor = TRUE,
  error.var = FALSE,
  cov.matrix = FALSE,
  conf = FALSE,
  conf.level = 0.95,
  p.value = FALSE
)

Arguments

z a numeric vector containing the tree data
p an integer determining the order of bifurcating autoregressive model to be fit to the data
x.data a logical that determines whether the x data used in fitting the model should be returned. Defaults to FALSE.
y.data a logical that determines whether the y data used in fitting the model should be returned. Defaults to FALSE.
resid a logical that determines whether the model residuals should be returned. Defaults to FALSE.
error.cor a logical that determines whether the estimated correlation between pairs of model errors ($e_{2t}, e_{2t+1}$) should be returned. Defaults to TRUE.
error.var a logical that determines whether the estimated variance of the model errors should be returned. Defaults to FALSE.
cov.matrix a logical that determines whether the estimated variance-covariance matrix of the least squares estimates should be returned. Defaults to FALSE.
conf a logical that determines whether confidence intervals for model coefficients should be returned. Defaults to FALSE. If TRUE, normal confidence intervals are calculated using cov.matrix.
conf.level confidence level to be used in computing the normal confidence intervals for model coefficients when conf=TRUE. Defaults to 0.95.
p.value a logical that determines whether p-values for model coefficients should be returned. Defaults to FALSE. If TRUE, p-values are computed from normal distribution using estimated coefficients and cov.matrix.

Value

d coef a matrix containing the least squares estimates of the autoregressive coefficients
error.cor the least squares estimate of the correlation between pairs of model errors \( (e_{2t}, e_{2t+1}) \). Only returned if error.cor=TRUE

x a matrix containing the x data used in fitting the model. Only returned if x.data=TRUE

y a vector containing the y data used in fitting the model. Only returned if y.data=TRUE

resids the model residuals. Only returned if resids=TRUE

error.var the estimated variance of the model errors. Only returned if error.var=TRUE

cov.matrix the estimated variance-covariance matrix of the least squares coefficients. Only returned if cov.matrix=TRUE

conf a matrix of normal confidence intervals for model coefficients. Only returned if conf=TRUE

p.value a matrix of two-sided p-values for testing the significance of model coefficients. Computed from normal distribution and using the estimated covariance matrix cov.matrix. Only returned if p.value=TRUE

References


Examples

```r
z <- bfa.tree.gen(127, 1, 1, 1, -0.9, -0.9, 0, 10, c(0.7))
bfa.ls(z, p=1)
bfa.ls(z,p=1,conf=TRUE,cov.matrix = TRUE,conf.level = 0.9,p.value=TRUE)
```

Bias-Corrected Least Squares Estimators for Bifurcating Autoregressive Models

Description

This function performs bias correction on the least squares estimators of the autoregressive coefficients in a BAR(p) model using single, double and fast-double bootstrapping, and the linear-bias-function approach as described in Elbayoumi & Mostafa (2020).

Usage

```r
bfa.ls.bc( 
  z, 
  p, 
  method = "boot1", 
  burn = 5, 
  B1 = 999, 
  B2 = 499, 
  boot.est = TRUE, 
  boot.data = FALSE 
)
```
**Arguments**

- **z**
  a numeric vector containing the tree data

- **p**
  an integer determining the order of bifurcating autoregressive model to be fit to the data

- **method**
  method of bias correction. Currently, "boot1", "boot2", "boot2fast" and "LBC" are supported and they implement single bootstrap, double bootstrap, fast-double bootstrap, and linear-bias-function bias-correction, respectively.

- **burn**
  number of tree generations to discard before starting the bootstrap sample (replicate)

- **B1**
  number of bootstrap samples (replicates) used in first round of bootstrapping

- **B2**
  number of bootstrap samples (replicates) used in second round of bootstrapping

- **boot.est**
  a logical that determines whether the bootstrapped least squares estimates of the autoregressive coefficients should be returned. Defaults to TRUE.

- **boot.data**
  a logical that determines whether the bootstrap samples should be returned. Defaults to FALSE.

**Value**

- **coef.ls.bc**
  bias-corrected least squares estimates of the autoregressive coefficients

**References**


**Examples**

```r
z <- bfa.tree.gen(63, 1, 1, 1, 0.5, 0.5, 0, 10, c(0.7))
bfa.ls.bc(z, p=1, method="boot1")
z <- bfa.tree.gen(63, 2, 1, 1, 0.5, 0.5, 0, 10, c(0.5, 0.3))
bfa.ls.bc(z, p=2, method="LBC")
```

---

**Description**

Draw scatterplots between observations at time t and the lagged observations from the given bifurcating autoregressive tree data.

**Usage**

```
bfa.scatterplot(z, p, ...)
```
Arguments

- **z**: a numeric vector containing the tree data
- **p**: an integer determining the order of the bifurcating autoregressive model that is believed to best fit the data
- ... other graphical parameters that can be passed to `plot()` or `pairs()` (see `par` and `pairs`)

Value

A single scatterplot when \( p=1 \) or a matrix of scatterplots when \( p>1 \).

Examples

```r
z <- bfa.tree.gen(63, 1, 2, 0.5, 0.5, 0.2, 10, c(0.7))
bfa.scatterplot(z, 1)
z <- bfa.tree.gen(63, 2, 1, 2, 0.5, 0.5, 0.2, 10, c(0.7, 0.2))
bfa.scatterplot(z, 2)
bfa.scatterplot(z, 2, lower.panel=NULL)
```

---

### bfa.subtree Subtree Extractor

Description

This function extracts subtree of size \( (2^p - 1) \) from the end of a given bifurcating autoregressive tree (model) of order \( p \).

Usage

`bfa.subtree(n, p)`

Arguments

- **n**: subtree size (integer)
- **p**: an integer determining the order of bifurcating autoregressive model

Value

A numeric vector representing a subtree of size \( (2^p - 1) \) from the end of a given bifurcating autoregressive tree.

Examples

```r
bfa.subtree(31, 1)
bfa.subtree(31, 2)
```
bfa.tree.gen  Bifurcating Autoregressive Tree generator

Description

This function generate bifurcating autoregressive (BFA) trees of any size based on a BFA model of any order.

Usage

bfa.tree.gen(n, p, s1, s2, r1, r2, g, intercept, ar.coef)

Arguments

n  tree size (integer)
p  an integer determining the order of bifurcating autoregressive model
s1  standard deviation of the errors distribution
s2  standard deviation of the second component of the mixture normal distribution generating contaminated errors. s2 should be greater than s1. s2 is only effective when g>0.
r1  correlation between pairs of errors
r2  is used in combination with r1 to compute the correlation between pairs of errors in the second component of the mixture normal distribution generating the contaminated errors. r2 is only effective when g>0.
g  proportion of contamination. Defaults to zero producing non-contaminated multivariate normal errors for the tree generation.
intercept  the intercept in the BAR model generating the tree
ar.coef  a vector of length p giving the autoregressive coefficients in the BAR model generating the tree

Value

A numeric vector representing a bifurcating autoregressive (BFA) tree with n observations.

Examples

#Non-contaminted BAR(1) tree:
bfa.tree.gen(127, 1, 1, 1, 0.5, 0.5, 0, 10, c(0.7))
#Non-contaminted BAR(2) tree:
bfa.tree.gen(127, 2, 1, 1, 0.5, 0.5, 0, 10, c(0.5, 0.3))
#Contaminted BAR(1) tree:
bfa.tree.gen(127, 1, 2, 0.5, 0.5, 0.2, 10, c(0.7))
bfa.tree.plot

Plotting Bifurcating Autoregressive Trees

Description

This function graphs bifurcating autoregressive (BFA) tree data.

Usage

bfa.tree.plot(
  z,
  digits,
  shape = "none",
  vertex.size = 10,
  text.size = 1,
  text.color = "black",
  vertex.color = "gold",
  arrow.size = 0.5,
  arrow.width = 0.5,
  arrow.color = "black",
  plot.margin = -0.3
)

Arguments

z a numeric vector containing the tree data
digits an integer indicating the number of decimal places to be displayed in vertex labels
shape the shape of the vertex. Currently “circle”, “square”, “csquare”, “rectangle”, “crectangle”, “vrectangle”, and “none” are supported. Defaults to “none” which does not display the vertices at all.
vertex.size a numeric scalar or vector defining the size of the vertex or vertices. If a vector is supplied, vertex sizes may differ. Defaults to 10.
text.size the font size of vertex labels. Defaults to 1.
text.color the color of vertex labels. If it is a character vector, then it may either contain integer values, named colors or RGB specified colors with three or four bytes. Defaults to "black".
vertex.color the fill color of the vertex. If you don’t want some or all vertices to have any color, supply NA. The default is "gold". See also the options in text.color.
arrow.size the size of the arrows. The default value is 0.5.
arrow.width the width of the arrows. The default value is 0.5.
arrow.color the color of the arrows. The default is "black". See also the options in text.color.
plot.margin the amount of empty space around the plot, it is a numeric vector of length four. Usually values between 0 and 0.5 are meaningful, but negative values are also possible and in that case it will make the plot zoom in to a part of the graph. If it is shorter than four, recycling will occur. The default value is -0.3.
Details

For more details about the graph options see `igraph.plotting`.

Value

A binary tree displaying the BFA data.

Examples

```r
z <- bfa.tree.gen(31, 1, 1, 1, 0.5, 0.5, 0, 10, c(0.7))
bfa.tree.plot(z)
bfa.tree.plot(z, shape= "circle")
bfa.tree.plot(z, shape= "circle", text.color="white", vertex.color = "darkgrey")
```

---

**ecoli**  

*Lifetimes (in minutes) of lineage E. coli cells.*

Description

A real cell lineage dataset taken from Cowan and Staudte (1986). The values represent the lifetimes in minutes of lineage E. coli cells. It contains 31 observations making which can be described by a bifurcating autoregressive tree with 4 generations.

Usage

`ecoli`

Format

A data frame with 31 rows and 1 variable:

- **lifetime**  
  E. coli cells lifetime, in minutes

Source

rcontmnorm

Contaminated Normal Generator

Description
This function generates contaminated multivariate normal errors that are used in the generation of the Bifurcating autoregressive tree.

Usage
rcontmnorm(
  n,
  d = 2,
  mu1 = rep(0, d),
  sigma1 = diag(d),
  mu2 = rep(0, d),
  sigma2 = diag(d),
  g
)

Arguments
n sample size
d dimension. Defaults to 2 for bivariate normal errors.
mu1 mean vector for first multivariate normal distribution. Defaults to the zero vector.
sigma1 variance-covariance matrix for first multivariate normal distribution. Defaults to the d by d identity matrix, where d is the dimension.
mu2 mean vector for second multivariate normal distribution. Defaults to the zero vector.
sigma2 variance-covariance matrix for second multivariate normal distribution. Defaults to the d by d identity matrix, where d is the dimension.
g proportion of contamination. Defaults to zero producing non-contaminated multivariate normal data.

Value
An n by d contaminated multivariate normal matrix.

Examples
#Non-contaminated bivariate normal:
rcontmnorm(10, sigma2=2^2*matrix(c(1,0,0,1),nrow=2) , g=0)
#Contaminated bivariate normal with 20% contamination:
rcontmnorm(10, sigma2=2^2*matrix(c(1,0,0,1),nrow=2) , g=0.2)
Description

This function generates multivariate normal errors that are used in the generation of the Bifurcating autoregressive tree.

Usage

```
rmnorm(n, d = 2, mu = rep(0, d), sigma = diag(d))
```

Arguments

- `n`: sample size
- `d`: dimension. Defaults to 2 for bivariate normal errors.
- `mu`: mean vector. Defaults to the zero vector.
- `sigma`: variance-covariance matrix. Defaults to the d by d identity matrix, where d is the dimension.

Value

An n by d multivariate normal matrix.

Examples

```
rmnorm(10)
rmnorm(10, 3)
```
Index

* datasets
  ecoli, 12
  bfa.boot1.ls, 2
  bfa.boot2.ls, 3
  bfa.boot2fast.ls, 4
  bfa.lbc.ls, 5
  bfa.ls, 5
  bfa.ls.bc, 7
  bfa.scatterplot, 8
  bfa.subtree, 9
  bfa.tree.gen, 10
  bfa.tree.plot, 11

  ecoli, 12

  igraph.plotting, 12

  pairs, 9
  par, 9

  rcontmnorm, 13
  rmnorm, 14