Package ‘odpc’

August 18, 2017

Type Package

Title One-Sided Dynamic Principal Components

Version 1.0.0

Date 2017-08-17

Author Daniel Peña <daniel.pena@uc3m.es>,
Ezequiel Smucler <ezequiels.90@gmail.com>,
Victor Yohai <vyohai@dm.uba.ar>

Maintainer Ezequiel Smucler <ezequiels.90@gmail.com>

Description Functions to compute the one-sided dynamic principal components ('odpc') introduced in Smucler, Peña and Yohai (2017) <arXiv:1708.04705>. 'odpc' is a novel dimension reduction technique for multivariate time series, that is useful for forecasting. These dynamic principal components are defined as the linear combinations of the present and past values of the series that minimize the reconstruction mean squared error.

License GPL (>= 2)

Imports methods, Rcpp (>= 0.12.7), gdpc, forecast

LinkingTo Rcpp, RcppArmadillo (>= 0.7.500.0.0)

Depends R (>= 3.3.1)

NeedsCompilation yes

Encoding UTF-8

Repository CRAN

Date/Publication 2017-08-18 08:21:42 UTC

R topics documented:

`odpc-package` *One-Sided Dynamic Principal Components*

Description

Computes One-Sided Dynamic Principal Components.

Details

Package:	odpc
Type:	Package
Version:	1.0.0
Date:	2017-08-17
Depends:	R (>= 3.3.1)
License:	GPL (>= 2)
Imports:	methods, Rcpp (>= 0.12.7), gdpc, forecast
LinkingTo:	Rcpp, RcppArmadillo (>= 0.7.500.0.0)
NeedsCompilation:	yes

Index:

```
components.odpcs   Get One-Sided Dynamic Principal Components from an odpcs object.

odpc    Computes One-Sided Dynamic Principal Components.

fitted.odpcs    Get reconstructed time series from an odpcs object.

forecast.odpcs    Get forecasts from an odpcs object.

plot.odpc   Plots an odpc object.
```

Author(s)

Daniel Peña, Ezequiel Smucler, Victor Yohai

Maintainer: Ezequiel Smucler <ezequiels.90@gmail.com>

References

Peña D., Smucler E. and Yohai V.J. (2017). "Forecasting Multiple Time Series with One-Sided Dynamic Principal Components." Available at https://arxiv.org/abs/1708.04705.

components.odpcs *Get One-Sided Dynamic Principal Components From an odpcs Object*

Description

Get One-Sided Dynamic Principal Components from an odpcs object.

Usage

```
## S3 method for class 'odpcs'
components(object, which_comp = 1)
```

Arguments

object	An object of class odpcs, usually the result of odpc.
which_comp	Numeric vector indicating which components to get. Default is 1.

Value

A list whose entries are the desired dynamic principal components.

Author(s)

Daniel Peña, Ezequiel Smucler, Victor Yohai

See Also

odpc, forecast.odpcs

Examples

```
T <- 200 #length of series
m <- 10 #number of series
set.seed(1234)
f <- rnorm(T + 1)
x <- matrix(0, T, m)
u <- matrix(rnorm(T * m), T, m)
for (i in 1:m) {
  x[, i] <- 10 * sin(2 * pi * (i/m)) * f[1:T] + 10 * cos(2 * pi * (i/m)) * f[2:(T + 1)] + u[, i]
}
fit <- odpc(x, ks = matrix(c(1, 1, 1, 0), 2, 2))
comps <- components.odpcs(fit, which_comp = c(1, 2))
```

fitted.odpcs *Get Reconstructed Time Series From an odpcs Object*

Description

Get reconstructed time series from an odpcs object.

Usage

```
## S3 method for class 'odpcs'
fitted(object, num_comp = 1, ...)
```

Arguments

object	An object of class odpcs, usually the result of odpc.
num_comp	Integer indicating how many components to use for the reconstrucion. Default is 1.
...	Additional arguments for compatibility.

Value

A matrix that is the reconstruction of the original series.

Author(s)

Daniel Peña, Ezequiel Smucler, Victor Yohai

See Also

odpc

Examples

```
T <- 200 #length of series
m <- 10 #number of series
set.seed(1234)
f <- rnorm(T + 1)
x <- matrix(0, T, m)
u <- matrix(rnorm(T * m), T, m)
for (i in 1:m) {
 x[, i] <- 10 * sin(2 * pi * (i/m)) * f[1:T] + 10 * cos(2 * pi * (i/m)) * f[2:(T + 1)] + u[, i]
}
fit <- odpc(x, ks = matrix(c(1, 1, 1, 0), 2, 2))
recons <- fitted(fit, num_comp = 2)
```

`forecast.odpcs` *Get Forecast From an odpcs Object*

Description

Get forecasts from an `odpcs` object.

Usage

```
## S3 method for class 'odpcs'
forecast(object, h, Z = NULL, add_residuals = FALSE, ...)
```

Arguments

`object`	An object of class odpcs, usually the result of odpc.
`h`	Integer. Number of periods for forecasting.
`Z`	Original data. Only used if add_residuals = TRUE.
`add_residuals`	Logical. Should the forecasts of the reconstruction residuals be added to the final forecast? Default is FALSE.
`...`	Additional arguments to be passed to `auto.arima`.

Details

Suppose q dynamic principal components were fitted to the data, each with (k_1^i, k_2^i) lags, $i = 1, \ldots, q$. Let $\widehat{\mathbf{f}}_T^i$ be the vector with the estimated values for the i-th dynamic principal component and $\widehat{\mathbf{B}}^i$, $\widehat{\boldsymbol{\alpha}}^i$ be the corresponding loadings and intercepts. Forecasts of the series are built by first fitting a SARIMA model to the components using `auto.arima` and getting their forecasts using `forecast.Arima`. Let $\widehat{f}_{T+h|T}^i$ for $h > 0$ be the forecast of f_{T+h}^i with information until time T. Then the h-steps ahead forecast of $\mathbf{z}_T$ is obtained as

$$\widehat{z}_{T+h|T,j} = \sum_{i=1}^{q} \left(\widehat{\alpha}_j^i + \sum_{v=0}^{k_2^i} \widehat{b}_{v,j}^i \widehat{f}_{T+h-v|T}^i \right) \quad j = 1, \ldots, m.$$

If add_residuals = TRUE, univariate SARIMA models are fitted to the residuals of the reconstruction, and their forecasts are added to the forecasts described above.

Value

A matrix that is the h-step ahead forecast of the original series.

Author(s)

Daniel Peña, Ezequiel Smucler, Victor Yohai

See Also

odpc, components.odpcs, fitted.odpcs, auto.arima, forecast.Arima

Examples

```
T <- 201 #length of series
m <- 10 #number of series
set.seed(1234)
f <- matrix(0, 2 * T + 1, 1)
v <- rnorm(2 * T + 1)
f[1] <- rnorm(1)
theta <- 0.7
for (t in  2:(2 * T)){
  f[t] <- theta * f[t - 1] + v[t]
}
f <- f[T:(2 * T)]
x <- matrix(0, T, m)
u <- matrix(rnorm(T * m), T, m)
for (i in 1:m) {
  x[, i] <- sin(2 * pi * (i/m)) * f[1:T] + cos(2 * pi * (i/m)) * f[2:(T + 1)] + u[, i]
}
fit <- odpc(x[1:(T - 1), ], ks = c(1))
forecasts <- forecast.odpcs(fit, h = 1)
mse <- mean((x[T, ] - forecasts)**2)
mse
```

---

odpc — *Fitting of One-Sided Dynamic Principal Components*

Description

Computes One-Sided Dynamic Principal Components for a given number of lags.

Usage

```
odpc(Z, ks, method, ini = 'classic', tol = 1e-04, niter_max = 500)
```

Arguments

Z	Data matrix. Each column is a different time series.
ks	Matrix or vector of integers. If a matrix, each row is the vector with number of lags to use for each component. First column has the number of lags used to define the dynamic principal component (k_1), second column has the number of lags of the dynamic principal component used to reconstruct the series (k_2). If a vector, its entries are taken as both k_1 and k_2 for each component
method	A string specifying the algorithm used. Options are 'ALS' or 'mix'. See details below.

`ini` A string specifying the initial estimator to be used, either 'classic' or 'gdpc'. Default is 'classic'. See details below.

`tol` Relative precision. Default is 1e-4.

`niter_max` Integer. Maximum number of iterations. Default is 500.

Details

Consider the vector time series $\mathbf{z}_1, \ldots, \mathbf{z}_T$, where $\mathbf{z}_t = (z_{t,1}, \ldots, z_{t,m})'$. Let $\mathbf{a} = (\mathbf{a}_0', \ldots, \mathbf{a}_{k_1}')'$, where $\mathbf{a}_h' = (a_{h,1}, ..., a_{h,m})$, be a vector of dimension $m(k_1+1) \times 1$, let $\boldsymbol{\alpha}' = (\alpha_1, \ldots, \alpha_m)$ and $\mathbf{B}$ the matrix that has coefficients $b_{h,j}$ and dimension $(k_2+1) \times m$. Consider

$$f_t = \sum_{j=1}^{m} \sum_{h=0}^{k_1} a_{h,j} z_{t-h,j} \quad t = k_1+1, \ldots, T,$$

and suppose we use f_t and k_2 of its lags to reconstruct the series as

$$z_{t,j}^R(\mathbf{a}, \boldsymbol{\alpha}, \mathbf{B}) = \alpha_j + \sum_{h=0}^{k_2} b_{h,j} f_{t-h}.$$

Let

$$MSE(\mathbf{a}, \boldsymbol{\alpha}, \mathbf{B}) = \frac{1}{T-(k_1+k_2)} \sum_{j=1}^{m} \sum_{t=(k_1+k_2)+1}^{T} (z_{t,j} - z_{t,j}^R(\mathbf{a}, \boldsymbol{\alpha}, \mathbf{B}))^2$$

be the reconstruction MSE. The first one-sided dynamic principal component is defined as the series

$$\widehat{f}_t = \sum_{j=1}^{m} \sum_{h=0}^{k_1} \widehat{a}_{h,j} z_{t-h,j} \quad t = k_1+1, \ldots, T,$$

for optimal values $(\widehat{\mathbf{a}}, \widehat{\boldsymbol{\alpha}}, \widehat{\mathbf{B}})$ that satisfy

$$MSE(\widehat{\mathbf{a}}, \widehat{\boldsymbol{\alpha}}, \widehat{\mathbf{B}}) = \min_{\|\mathbf{a}\|=1, \boldsymbol{\alpha}, \mathbf{B}} MSE(\mathbf{a}, \boldsymbol{\alpha}, \mathbf{B}).$$

The second one-sided dynamic principal component is defined as the first one-sided dynamic principal component of the residuals. Higher order principal components are defined similarly.

If method = 'ALS', an Alternating Least Squares type algorithm is used to compute the solution. If 'mix' is chosen, in each iteration Least Squares is used to compute the matrix of loadings and intercepts, but one iteration of Coordinate Descent is performed to compute the vector a that defines the dynamic principal component. By default, 'ALS' is used when the number of series is less than 40, else 'mix' is used.

If ini = 'classic', the ordinary (non-dynamic) principal components of the data are used as a starting point of the iterations. If ini = 'gdpc', the generalized dynamic principal components are used.

Value

An object of class `odpcs`, that is, a list of length equal to the number of computed components. The i-th entry of this list is an object of class `odpc`, that is, a list with entries

f	Coordinates of the i-th dynamic principal component corresponding to the periods $k_1 + 1, \ldots, T$.
mse	Mean squared error of the reconstruction using the first i components.
k1	Number of lags used to define the i-th dynamic principal component f.
k2	Number of lags of f used to reconstruct.
alpha	Vector of intercepts corresponding to f.
a	Vector that defines the i-th dynamic principal component
B	Matrix of loadings corresponding to f. Row number k is the vector of $k - 1$ lag loadings.
call	The matched call.
conv	Logical. Did the iterations converge?

`components`, `fitted`, `plot` and `print` methods are available for this class.

Author(s)

Daniel Peña, Ezequiel Smucler, Victor Yohai

References

Peña D., Smucler E. and Yohai V.J. (2017). "Forecasting Multiple Time Series with One-Sided Dynamic Principal Components." Available at https://arxiv.org/abs/1708.04705.

See Also

`plot.odpc`, `fitted.odpcs`, `components.odpcs`, `forecast.odpcs`, `gdpc`, `gdpc`

Examples

```
T <- 200 #length of series
m <- 10 #number of series
set.seed(1234)
f <- rnorm(T + 1)
x <- matrix(0, T, m)
u <- matrix(rnorm(T * m), T, m)
for (i in 1:m) {
   x[, i] <- 10 * sin(2 * pi * (i/m)) * f[1:T] + 10 * cos(2 * pi * (i/m)) * f[2:(T + 1)] + u[, i]
}
fit <- odpc(x, ks = c(1))
fit
fit2 <- odpc(x, ks = c(1), ini = 'gdpc')
fit2
```

plot.odpc *Plot One-Sided Dynamic Principal Components*

Description

Plots an odpc object.

Usage

```
## S3 method for class 'odpc'
plot(x, which = 'Component', which_load = 0, ...)
```

Arguments

x	An object of class odpc, usually one of the entries of the result of odpc.
which	String. Indicates what to plot, either 'Component' or 'Loadings'. Default is 'Component'.
which_load	Lag number indicating which loadings should be plotted. Only used if which = 'Loadings'. Default is 0.
...	Additional arguments to be passed to the plotting functions.

Author(s)

Daniel Peña, Ezequiel Smucler, Victor Yohai

See Also

odpc

Examples

```
T <- 200 #length of series
m <- 10 #number of series
set.seed(1234)
f <- rnorm(T + 1)
x <- matrix(0, T, m)
u <- matrix(rnorm(T * m), T, m)
for (i in 1:m) {
   x[, i] <- 10 * sin(2 * pi * (i/m)) * f[1:T] + 10 * cos(2 * pi * (i/m)) * f[2:(T + 1)] + u[, i]
}
fit <- odpc(x, ks = c(1))
plot(fit[[1]], xlab = '', ylab = '')
```

Index