Package 'CepLDA'

January 18, 2016

Type Package

Title Discriminant Analysis of Time Series in the Presence of Within-Group Spectral Variability

Version 1.0.0

Date 2015-12-19

URL http://astro.temple.edu/~tuf28410/

Author Zeda Li, Robert Krafty

Maintainer Zeda Li<zeda.li@temple.edu>

Depends R(>= 3.0.1), astsa, MASS, class, multitaper

Description Performs cepstral based discriminant analysis of groups of time series when there exists Variability in power spectra from time series within the same group as described in R.T. Krafty (2016) ``Discriminant Analysis of Time Series in the Presence of Within-Group Spectral Variability" Journal of Time Series Analysis.

License GPL (>= 2)

LazyData true

RoxygenNote 5.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2016-01-18 00:35:29

R topics documented:

cep.get 2
cep.lda 3
cep.mtm 5
Lopt.get 6
lperd.mtm 7
plot.ceplda 8
predict.ceplda 8
print.ceplda 10
r.cond.ar2 10
recon 11

Index **13**

`cep.get` *Obtain Data Frame to be Used in* `Lopt.get` *and* `predict.ceplda`

Description

Returns a data frame containing raw cepstra coefficients and the group membership from multiple time seres.

Usage

```
cep.get(y,x,nw,k)
```

Arguments

`y`	n-vector indicating group membership
`x`	*N* by *n* matrix containing *n* time series, each with length *N*.
`nw`	Width of tapers used in multitaper spectral estimation. Default is set to 4
`k`	Number of tapers used in multitaper spectral estimation. Default is set to 7

Value

`D.hat`	Data frame containing group information and raw cepstral coefficients.

Author(s)

Zeda Li <<zeda.li@temple.edu>>

References

Krafty, RT(2016) Discriminant Analysis of Time Series in the Presence of Within-Group Spectral Variability. *Journal of Time series analysis*

See Also

`predict.ceplda`, `Lopt.get`

Examples

```
## Simulate dataset
nj = 50  #number of series in training data
N = 500  #length of time series
data1 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.01,.7),r.phi2=c(-.12,-.06),r.sig2=c(.3,3))
data2 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.5,1.2),r.phi2=c(-.36,-.25),r.sig2=c(.3,3))
data3 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.9,1.5),r.phi2=c(-.56,-.75),r.sig2=c(.3,3))
dat <- cbind(data1$X,data2$X,data3$X)
y <- c(rep(1,nj),rep(2,nj),rep(3,nj))
data.cep <- cep.get(y,dat,4,7)
dim(data.cep)
```

`cep.lda` *Discriminant Analysis of Time Series in the Presence of Within-Group Spectral Variability*

Description

The main program.

Usage

```
cep.lda(y,x,xNew,L,mcep,nw,k,cv,tol)
```

Arguments

`y`	n-vector indicating group membership of training time series.
`x`	*N* by *n* matrix containing *n* training time series each with length *N*.
`xNew`	*N* by *nNew* matrix, containing nNew time series whose memberships are predicted.
`L`	Number of cepstral coefficients used in the lda. If FALSE, cross-validation is used for the data driven selection of L. Default is FALSE.
`mcep`	Maximum number of cepstral coefficients considerd. Default is set to 10.
`nw`	Width of tapers used in multitaper spectral estimation. Default is set to 4.
`k`	Number of tapers used in multitaper spectral estimation. Default is set to 7.
`cv`	If TRUE, returns results (classes and posterior probabilities) for leave-one-out cross-validation. Note that, if the prior is estimated, the proportions in the whole dataset are used. As with the standard lda function, if used, prediction on a test data set cannot be done and weight functions are not produced (simular to the predict.lda). Default is FALSE.
`tol`	Tolerance to decide if a matrix is singular; it will reject variables and linear combinations of unit-variance variables whose variance is less than tol^2.

Value

List with 5 elements

`C.lda`	lda output on the cepstral scale. Similar to output of `lda(MASS)` function.
`cep.data`	Data frame containing cepstral coefficients and group information from training data.
`Lopt`	Number of cepstral coefficients used.
`lspec`	Estimated log-spectral weight functions.
`predict`	Results of classification. If external data xNew is supplied, these data are classified. If not, biased classification of the training data x is returned. For unbiased leave-out-one cross-validated classification of training data, use cv=TRUE.

Author(s)

Zeda Li <<zeda.li@temple.edu>>; Robert Krafty <<rkrafty@pitt.edu>>

References

Krafty, RT (2016) Discriminant Analysis of Time Series in the Presence of Within-Group Spectral Variability. *Journal of Time Series Analysis*

See Also

predict.ceplda, plot.ceplda, print.ceplda, Lopt.get

Examples

```
## Simulate training data
nj = 50  #number of series in training data
N = 500  #length of time series
traindata1 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.01,.7),r.phi2=c(-.12,-.06),r.sig2=c(.3,3))
traindata2 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.5,1.2),r.phi2=c(-.36,-.25),r.sig2=c(.3,3))
traindata3 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.9,1.5),r.phi2=c(-.56,-.75),r.sig2=c(.3,3))
train <- cbind(traindata1$X,traindata2$X,traindata3$X)
y <- c(rep(1,nj),rep(2,nj),rep(3,nj))

## Fit the discriminant analysis
fit <- cep.lda(y,train)
fit  #displays group means and cepstral weight functions

## Discriminant plot
plot(fit)

## Plot log-spectral weights
par(mfrow=c(1,2))
plot(fit$lspec$frq, fit$lspec$dsc[,1],type='l',xlab="frequency", ylab="log-spectral weights")
plot(fit$lspec$frq, fit$lspec$dsc[,2],type='l',xlab="frequency", ylab="log-spectral weights")

## Bias classification of training data
mean(fit$predict$class == y) #classifictaion rate
table(y,fit$predict$class)

## Fit the discriminant analysis while classifing training data via cross-validation
fit.cv <- cep.lda(y,train, cv=TRUE)
mean(fit.cv$predict$class == y) #classifictaion rate
table(y,fit.cv$predict$class)

## Simulate test data
testdata1 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.01,.7),r.phi2=c(-.12,-.06),r.sig2=c(.3,3))
testdata2 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.5,1.2),r.phi2=c(-.36,-.25),r.sig2=c(.3,3))
testdata3 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.9,1.5),r.phi2=c(-.56,-.75),r.sig2=c(.3,3))
test <- cbind(testdata1$X,testdata2$X,testdata3$X)
yTest <- c(rep(1,nj),rep(2,nj),rep(3,nj))

## Fit discriminant analysis and classify new data
```

```
fit.pre <- cep.lda(y,train,test)
mean(fit.pre$predict$class == y)
table(yTest,fit.pre$predict$class)
```

---

cep.mtm *Multitaper Estimation of Cepstral Coefficients and the Log-Spectrum*

Description

Returns multitaper estimated cepstra coefficients and log-spectrum for univariate time series.

Usage

```
cep.mtm(x,nw,k)
```

Arguments

x	Univariate time series of length N.
nw	Width of tapers. Default is set to 4
k	Number of tapers. Default is set to 7

Value

a list with 4 elements

quef	Quefencies.
cep	Raw cepstra coefficients from 0:(N-1)
freq	Frequencies between 0 and 1
lspec	Log-spectrum.

Author(s)

Robert Krafty <<rkrafty@pitt.edu>>

Examples

```
## simulate a time series
N = 500 #length of each series
dat <- r.cond.ar2(N=N,nj=1,r.phi1=c(.01,.7),r.phi2=c(-.12,-.06),r.sig2=c(.3,3))$X

## Fit multiaper
cep <- cep.mtm(dat)

## Plot the cepstral coefficients
plot(cep$quef, cep$cep)

## Plot the log spectrum
plot(cep$freq, cep$lspec, type="l")
```

Lopt.get *Optimal Choice of L*

Description

Data driven selection of the number of cepstral coefficients (L) via leave-one-out cross-validation.

Usage

```
Lopt.get(data,mcep)
```

Arguments

data	Data frame containing cepstral coefficients and group information. Obtained from cep.get or cep.lda.
mcep	Maximum number of cepstral coefficient considerd. Default is set to 10

Value

Lopt	Optimal number of cepstral coefficients

Author(s)

Robert Krafty <rkrafty@pitt.edu>

References

Krafty, RT (2016) Discriminant Analysis of Time Series in the Presence of Within-Group Spectral Variability. *Journal of Time series analysis*

See Also

cep.lda, cep.get

Examples

```
## Simulate data
ntrain = 50  #number of series in training data
Nlength = 500 #length of each series
set.seed(2016)
traindata1 <- r.cond.ar2(N=Nlength,nj=ntrain,r.phi1=c(.01,.7),r.phi2=c(-.12,-.06),r.sig2=c(.3,3))
traindata2 <- r.cond.ar2(N=Nlength,nj=ntrain,r.phi1=c(.5,1.2),r.phi2=c(-.36,-.25),r.sig2=c(.3,3))
traindata3 <- r.cond.ar2(N=Nlength,nj=ntrain,r.phi1=c(.9,1.5),r.phi2=c(-.56,-.75),r.sig2=c(.3,3))
train <- cbind(traindata1$X,traindata2$X,traindata3$X)
## group information
y <- c(rep(1,ntrain),rep(2,ntrain),rep(3,ntrain))
dat <- cep.get(y,train)
Lopt.get(dat,10)
```

`lperd.mtm` *Multitaper Log-Periodogram*

Description

Obtain multivariate log-periodogram. Not often used by itself, but is used as part of `cep.mtm`.

Usage

```
lperd.mtm(x,nw,k)
```

Arguments

`x`	*N* by *n* matrix containing *n* training time series each with length *N*.
`nw`	Width of tapers used in multitaper spectral estimation.
`k`	Number of tapers used in multitaper spectral estimation.

Value

`freq`	Fourier frequencies from 0 to 1.
`lspec`	Log-periodogram.
`spec`	Periodigram on the natural scale.

Author(s)

Robert Krafty <rkrafty@pitt.edu>

References

Krafty, RT (2016) Discriminant Analysis of Time Series in the Presence of Within-Group Spectral Variability. *Journal of Time series analysis*

See Also

`cep.mtm`, `cep.lda`

plot.ceplda *Plot Method for Class 'ceplda'*

Description

Plots a set of data on one, two or more linear discriminants.

Usage

```
## S3 method for class 'ceplda'
plot(x, ...)
```

Arguments

x	an object of class "ceplda".
...	additional arguments.

Details

This function is a method for the generic function plot() for class "ceplda". It can be invoked by calling plot(x) for an object x of the appropriate class, or directly by calling plot.ceplda(x) regardless of the class of the object. Details usage of this function is equivalent to plot.lda(MASS).

See Also

cep.lda, plot.lda

predict.ceplda *Classify Multivariate Time Series*

Description

Classify time series. Run as part of cep.lda, and can be run seperatly after running cep.lda.

Usage

```
## S3 method for class 'ceplda'
predict(object, newdata, ...)
```

Arguments

object	Object of class "ceplda".
newdata	Data frame of cases to be classified. Data frame can be obtained by cep.get.
...	argument based from or to other methods.

Details

This function is a method for the generic function `predict()` for class "ceplda". It can be invoked by calling `predict(x)` for an object x of the appropriate class, or directly by calling `predict.lda(x)` regardless of the class of the object. Details usage of this function is similar to `predict.lda(MASS)`.

Value

List with components

`class`	Classification result (a factor).
`posterior`	Posterior class probabilities.
`x`	Scores from test data.

See Also

`cep.lda`, `predict.lda`

Examples

```
## See cep.lda for predicting new data simultaneously while fitting a model to training data.
## Below is predicting new data after fitting a model to the training data.

## Simulate training data
nj = 50  #number of series in training data
N = 500  #length of time series
traindata1 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.01,.7),r.phi2=c(-.12,-.06),r.sig2=c(.3,3))
traindata2 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.5,1.2),r.phi2=c(-.36,-.25),r.sig2=c(.3,3))
traindata3 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.9,1.5),r.phi2=c(-.56,-.75),r.sig2=c(.3,3))
train <- cbind(traindata1$X,traindata2$X,traindata3$X)
y <- c(rep(1,nj),rep(2,nj),rep(3,nj))

## Fit the discriminant analysis
fit <- cep.lda(y,train)

## Simulate test data
testdata1 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.01,.7),r.phi2=c(-.12,-.06),r.sig2=c(.3,3))
testdata2 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.5,1.2),r.phi2=c(-.36,-.25),r.sig2=c(.3,3))
testdata3 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.9,1.5),r.phi2=c(-.56,-.75),r.sig2=c(.3,3))
test <- cbind(testdata1$X,testdata2$X,testdata3$X)
yTest <- c(rep(1,nj),rep(2,nj),rep(3,nj))


## Classifty new data
pre <- predict(fit,cep.get(yTest,test))
mean(pre$class == yTest)
table(yTest,pre$class)
```

print.ceplda *Print Method for Class 'ceplda'*

Description

Print the results from `cep.lda`

Usage

```
## S3 method for class 'ceplda'
print(x, ...)
```

Arguments

`x`	an object of class "ceplda".
`...`	additional arguments.

See Also

`cep.lda`

r.cond.ar2 *Generate Random AR(2) Time Series*

Description

Simulates multiple AR(2) time series.

Usage

```
r.cond.ar2(N,nj,r.phi1,r.phi2,r.sig2)
```

Arguments

`N`	Length of the series
`nj`	Number of series generated.
`r.phi1`	Range of first AR order coefficient. It is a vector contains minimum and maximum possible coefficients.
`r.phi2`	Range of second AR order coefficient. It is a vector contains minimum and maximum possible coefficients.
`r.sig2`	Range of conditional innovation variances. It is a vector contains minimum and maximum possible variances.

Value

a list with 2 elements

X	*N* by *nj* matrix of time series
cep	3 by *nj* matrix of parameters (phi1, phi2, sig2)

Author(s)

Robert Krafty <<rkrafty@pitt.edu>>

References

Krafty, RT (2016) Discriminant Analysis of Time Series in the Presence of Within-Group Spectral Variability. *Journal of Time series analysis*

See Also

cep.lda

Examples

```
## Simulate data
nj = 50  #number of series in training data
N = 500  #length of time series
data1 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.01,.7),r.phi2=c(-.12,-.06),r.sig2=c(.3,3))
data2 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.5,1.2),r.phi2=c(-.36,-.25),r.sig2=c(.3,3))
data3 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.9,1.5),r.phi2=c(-.56,-.75),r.sig2=c(.3,3))
data <- cbind(data1$X,data2$X,data3$X)
y <- c(rep(1,nj),rep(2,nj),rep(3,nj))
```

---

recon *Convert Cepstral Coefficients into Log-Spectra.*

Description

Returns a log-spectrum at a given set of frequencies from a cepstrum.

Usage

```
recon(wts,fqs)
```

Arguments

wts	Cepstral coefficients.
fqs	Frequencies to evaluate the log-spectrum.

Value

dsc	The log-spectrum.

Author(s)

Robert Krafty <<rkrafty@pitt.edu>>

Examples

```
## Simulate dataset
nj = 50  #number of series in training data
N = 500  #length of time series
data1 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.01,.7),r.phi2=c(-.12,-.06),r.sig2=c(.3,3))
data2 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.5,1.2),r.phi2=c(-.36,-.25),r.sig2=c(.3,3))
data3 <- r.cond.ar2(N=N,nj=nj,r.phi1=c(.9,1.5),r.phi2=c(-.56,-.75),r.sig2=c(.3,3))
dat <- cbind(data1$X,data2$X,data3$X)
y <- c(rep(1,nj),rep(2,nj),rep(3,nj))
data.cep <- cep.get(y,dat,4,7)

## Convert cepstral coefficients into log-spectra
frqs <- seq(from=0, to=.5, by=1/(dim(data.cep)[2]-1))
lspec <- matrix(0,dim(data.cep)[1], length(frqs))
## rows of lspec matrix contains log-spectra
for(i in 1:dim(data.cep)[1]){
  lspec[i,] <- recon(data.cep[,i],frqs)
}
```

Index

cep.get, 2, *6*
cep.lda, 3, *6–11*
cep.mtm, 5, *7*

Lopt.get, *2*, *4*, 6
lperd.mtm, 7

plot.ceplda, *4*, 8
plot.lda, *8*
predict.ceplda, *2*, *4*, 8
predict.lda, *9*
print.ceplda, *4*, 10

r.cond.ar2, 10
recon, 11