Package 'CCM'

October 12, 2022

Type Package

Title Correlation Classification Method

Version 1.2

Date 2018-04-05

Author Garrett M. Dancik and Yuanbin Ru

Maintainer Garrett M. Dancik <dancikg@easternct.edu>

Description

Classification method described in Dancik et al (2011) <doi:10.1158/0008-5472.CAN-11-2427> that classifies a sample according to the class with the maximum mean (or any other function of) correlation between the test and training samples with known classes.

License GPL (>= 2)

LazyLoad yes

Repository CRAN

Date/Publication 2018-04-12 12:58:17 UTC

NeedsCompilation no

R topics documented:

CCM-package 2
cor.by.class 3
create.CCM 4
data.expr 5
data.gender 6
plot.CCM 6
predict.CCM 8

Index **10**

CCM-package *Correlation classification method (CCM)*

Description

Classification method that classifies an observation based on its correlation with observations having known class labels. There are two main functions. The function `create.CCM` creates a correlation matrix of correlations between training and test samples. Both Pearson's and Spearman's rank-based correlations are supported. The function `predict.CCM` assigns class labels to test observations according to the class that has the highest mean correlation by default. However, any (user-defined) function in addition to the mean (e.g., median, max) can be specified.

For a complete list of functions, use 'library(help="CCM")'

Details

Package:	CCM
Type:	Package
Version:	1.2
Date:	2018-04-05
License:	GPL(>=2)
LazyLoad:	yes

Author(s)

Garrett M. Dancik and Yuanbin Ru
Maintainer: Garrett M. Dancik <dancikg@easternct.edu>

See Also

`create.CCM`; `predict.CCM`; `plot.CCM`

Examples

```
## load data ##
data(data.expr)
data(data.gender)

## check within class correlations ##
## outliers may be caused by poor quality ##
## observations or may indicate CCM is not appropriate ##
K = cor.by.class(data.expr, data.gender)
## visualize the results ##
boxplot(K, xlab = "gender")
```

```
## split dataset into training / testing ##
train.expr = data.expr[,1:20]
test.expr = data.expr[,21:40]
train.gender = data.gender[1:20]
test.gender = data.gender[21:40]

## CCM using spearman correlation ##
K = create.CCM(test.expr, train.expr, method = "spearman")

## predict based on the class with the highest mean correlation (the default) ##
p = predict(K, train.gender)
table(pred = p, true = test.gender) # check accuracy

## plot correlations for the 3rd observation ##
plot(K, train.gender, index = 3, main = "correlations for obs #3",
  xlab = "gender", ylab = "correlation")
```

`cor.by.class` *Finds within class correlations*

Description

Finds within class correlations between samples of each class type, which is useful for identifying extreme observations and assessing whether CCM is appropriate for classification.

Usage

```
cor.by.class(x, y, method = "pearson", use = "complete")
```

Arguments

`x`	data matrix with variables in rows and samples in columns
`y`	classes corresponding to the columns of `x`
`method`	the type of correlation to use, either 'pearson' (the default) or 'spearman'
`use`	instructions for handling missing values. See details and `cor`. All values are used by default.

Details

Calculates correlations between each pair of observations within each class. The correlation between an observation and itself is ignored.

The default correlation is the Pearson product moment correlation. If `method` is 'spearman', then the Spearman's rank correlation is used, which is the Pearson correlation calculated using the ranks of the data.

Correlations are calculated class-wise on the matrix of observations of each class separately. Therefore, missing values may be handled differently for different classes.

Value

A list with each element a vector of correlations between samples of a different class.

Author(s)

Garrett M. Dancik and Yuanbin Ru

Examples

```
data(data.expr)
data(data.gender)
K = cor.by.class(data.expr, data.gender)
## visualize the results ##
boxplot(K, xlab = "gender")
```

---

create.CCM *Creates a CCM correlation matrix*

Description

Creates a CCM correlation matrix that calculates correlations between test and training samples or between each test sample

Usage

```
create.CCM(X.test, X.train = NULL, method = "pearson", use = "everything", verbose = 1)
```

Arguments

X.test	matrix of test samples with rows containing variables and columns containing observations
X.train	optional matrix of training samples with rows containing variables and columns containing observations
method	the type of correlation to use, either 'pearson' (the default) or 'spearman'
use	instructions for handling missing values. See details and cor. All values are used by default.
verbose	A value of '0' will suppress output

Details

The default correlation is the Pearson product moment correlation. If method is 'spearman', then the Spearman's rank correlation is used, which is the Pearson correlation calculated using the ranks of the data.

If X.train is NULL then correlations are calculated between each column of X.test, but the correlation between a sample and itself will be assigned the value *NA*.

When X.train is specified, correlations are calculated sequentially between each test observation and all training observations using apply. Note that if missing values are present, they may be handled differently for different test samples.

Value

A CCM correlation matrix with element (i,j) as follows: if `X.train` is not NULL, then the correlation between the i(th) test sample and the j(th) training sample; otherwise the correlation between the i(th) and j(th) test samples, with NA along the diagonal.

Author(s)

Garrett M. Dancik and Yuanbin Ru

See Also

`cor`, the function used to calculate correlations; `predict.CCM`, for classification based on the CCM matrix; `plot.CCM` for plotting correlations of test samples

Examples

```
data(data.expr)
data(data.gender)

## split dataset into training / testing ##
train.expr = data.expr[,1:20]
test.expr = data.expr[,21:40]
train.gender = data.gender[1:20]
test.gender = data.gender[21:40]

## CCM using spearman correlation ##
K = create.CCM(test.expr, train.expr, method = "spearman")
## predict based on the class with the highest mean correlation (the default) ##
p = predict(K, train.gender)
table(pred = p, true = test.gender) # check accuracy

## CCM using pearson correlation ##
K = create.CCM(test.expr, train.expr, method = "pearson")
## predict based on the class with the maximum correlation
p = predict(K, train.gender, func = max)
table(pred = p, true = test.gender) # check accuracy

### leave-one-out cross validation on entire dataset ###
K = create.CCM(data.expr, method = "spearman")
p = predict(K, data.gender)
table(pred = p, true = data.gender) # check accuracy
```

---

data.expr *Sample gene expression and gender data*

Description

Simulated gene expression data

Usage

```
data(data.expr)
```

Format

A 100x40 (probe x sample) data matrix of gene expression data

Examples

```
data(data.expr)
data(data.gender)
```

---

data.gender *Sample gene expression and gender data*

Description

Simulated gender data that corresponds with `data.expr`

Usage

```
data(data.gender)
```

Format

A 40 element vector containing simulated gender information as a factor (M/F), corresponding to columns of data.expr

Examples

```
data(data.expr)
data(data.gender)
```

---

plot.CCM *Plot CCM correlations*

Description

Constructs a boxplot of correlations by class for a test sample

Usage

```
## S3 method for class 'CCM'
plot(x, y, index,  no.plot, ...)
```

Arguments

x	a CCM correlation matrix as obtained from create.CCM
y	classes corresponding to the training samples (columns) of 'K'
index	the test sample to include in the plot, corresponding to the row of 'K'.
no.plot	if TRUE then plotting is turned off and a list of correlations is returned
...	additional arguments for boxplot

Details

This function generates a boxplot of correlations between the training samples and a specific test sample by class

Value

if no.plot is TRUE, then a list of correlations by class

Author(s)

Garrett M. Dancik and Yuanbin Ru

See Also

boxplot; create.CCM for creating the CCM correlation matrix

Examples

```
## load data ##
data(data.expr)
data(data.gender)

## split dataset into training / testing ##
train.expr = data.expr[,1:20]
test.expr = data.expr[,21:40]
train.gender = data.gender[1:20]
test.gender = data.gender[21:40]

## CCM using spearman correlation ##
K = create.CCM(test.expr, train.expr, method = "spearman")

## plot correlations for the 3rd observation ##
plot(K, train.gender, index = 3, main = "correlations for obs #3",
  xlab = "gender", ylab = "correlation")
```

predict.CCM *Classification from a CCM correlation matrix*

Description

Classification as a function of a CCM correlation matrix that contains the correlations between test and training samples

Usage

```
## S3 method for class 'CCM'
predict(object, y, func = mean, ret.scores = FALSE, ...)
```

Arguments

object	a CCM correlation matrix object obtained from create.CCM
y	classes corresponding to the training samples (columns) of 'object'
func	the function that determines how a test sample is classified, defaulting to *mean*. See details.
ret.scores	If set to TRUE then a matrix of results by class are returned (see details); otherwise a vector of classifications/predictions is returned (the default)
...	Additional arguments to func

Details

The function func can be any R function whose first argument is a vector of correlations (x). The CCM assigns each test sample the class that maximizes func(x). If func is *mean* (the default), the classification is the class with the highest mean correlation. Other useful values for func include *median* and *max*.

If ret.scores is TRUE, then a matrix of results by class is returned, where the i(th) column corresponds to the i(th) test sample and each row corresponds to a possible class. Entry (i,j) contains func(x), where x is a vector of correlations between the i(th) test sample and all training samples with the class in row j.

Value

The test sample classifications as a vector or a matrix of results by class.

Note

If the *max* function is used for func, then the CCM is identical to a 1-nearest neighbor classifier with distance = 1 - r, where 'r' is the correlation (pearson or spearman) specified in the call to create.CCM

Author(s)

Garrett M. Dancik and Yuanbin Ru

See Also

create.CCM

Examples

```
data(data.expr)
data(data.gender)

## split dataset into training / testing ##
train.expr = data.expr[,1:20]
test.expr = data.expr[,21:40]
train.gender = data.gender[1:20]
test.gender = data.gender[21:40]

## CCM using spearman correlation ##
K = create.CCM(test.expr, train.expr, method = "spearman")
## predict based on the class with the highest mean correlation (the default) ##
p = predict(K, train.gender)
table(pred = p, true = test.gender) # check accuracy

## CCM using pearson correlation ##
K = create.CCM(test.expr, train.expr, method = "pearson")
## predict based on the class with the maximum correlation
p = predict(K, train.gender, func = max)
table(pred = p, true = test.gender) # check accuracy
```

Index

∗ **datasets**
    data.expr, 5
    data.gender, 6
∗ **hplot**
    CCM-package, 2
    plot.CCM, 6
∗ **methods**
    CCM-package, 2
    create.CCM, 4
    plot.CCM, 6
    predict.CCM, 8
∗ **package**
    CCM-package, 2

boxplot, *7*

CCM *(*CCM-package*)*, 2
CCM-package, 2
cor, *5*
cor.by.class, 3
create.CCM, *2*, 4, *7–9*

data.expr, 5
data.gender, 6

plot.CCM, *2*, *5*, 6
predict.CCM, *2*, *5*, 8