Package ‘RSDA’

October 11, 2018

Type Package

Title R to Symbolic Data Analysis

Version 2.0.8

Date 2018-10-09

Author Oldemar Rodriguez R. with contributions from Carlos Aguero, Olger Calderon, Roberto Zuniga and Jorge Arce

Maintainer Oldemar Rodriguez <oldemar.rodriguez@ucr.ac.cr>

Description Symbolic Data Analysis (SDA) was proposed by professor Edwin Diday in 1987, the main purpose of SDA is to substitute the set of rows (cases) in the data table for a concept (second order statistical unit). This package implements, to the symbolic case, certain techniques of automatic classification, as well as some linear models.

License GPL (>= 2)

Encoding UTF-8

Depends R (>= 3.4)

Imports RJSONIO, glmnet, abind, scatterplot3d, graphics, stats, utils, FactoMineR (>= 1.36), XML, scales, ggplot2, princurve, sqldf, dplyr (>= 0.7.1), tidyr, stringr, lazyeval, nloptr (>= 1.0.4) ,xtable,pander, rlang(>= 0.1.1), purrr, tidyselect, tibble, ggpolypath, reshape, randomcoloR

URL http://www.oldemarrodriguez.com

Suggests testthat

RoxygenNote 6.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2018-10-10 23:00:10 UTC

R topics documented:

abalone *SODAS XML data file.*

Description

Example of SODAS XML data file converted in a CSV file in RSDA format.

Usage

```
data(abalone)
```

Format

abalone<-SODAS.to.RSDA('C:/Program Files (x86)/DECISIA/SODAS version 2.0/bases/abalone.xml)

Source

http://www.info.fundp.ac.be/asso/sodaslink.htm

References

Bock H-H. and Diday E. (eds.) (2000).Analysis of Symbolic Data. Exploratory methods for extracting statistical information fromcomplex data. Springer, Germany.

Examples

```
data(abalone)
res<-sym.interval.pca(abalone,'centers')
class(res$Sym.Components) <- c('sym.data.table')
sym.scatterplot(res$Sym.Components[,1],res$Sym.Components[,2],
labels=TRUE,col='red',main='PCA Oils Data')
sym.scatterplot3d(res$Sym.Components[,1],res$Sym.Components[,2],
res$Sym.Components[,3],color='blue',main='PCA Oils Data')
sym.scatterplot.ggplot(res$Sym.Components[,1],
res$Sym.Components[,2],labels=TRUE)
sym.circle.plot(res$Sym.Prin.Correlations)
```

calc.burt.sym *Burt Matrix*

Description

Burt Matrix

Usage

```
calc.burt.sym(sym.data, pos.var)
```

Arguments

`sym.data`	ddd
`pos.var`	ddd

`Cardiological` *Cardiological data example*

Description

Cardiological interval data example.

Usage

```
data(Cardiological)
```

Format

An object of class `list` of length 9.

References

Billard L. and Diday E. (2006).Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Examples

```
data(Cardiological)
car.data <- Cardiological
res.cm <- sym.lm(Pulse~Syst+Diast,sym.data=car.data,method='cm')
pred.cm <- predictsym.lm(res.cm,car.data,method='cm')
RMSE.L(sym.var(car.data,1),pred.cm$Fitted)
RMSE.U(sym.var(car.data,1),pred.cm$Fitted)
R2.L(sym.var(car.data,1),pred.cm$Fitted)
R2.U(sym.var(car.data,1),pred.cm$Fitted)
deter.coefficient(sym.var(car.data,1),pred.cm$Fitted)
```

centers.interval *Compute centers of the interval*

Description

Compute centers of the interval

Usage

```
centers.interval(sym.data)
```

Arguments

sym.data	Symbolic interval data table.

Value

Centers of teh intervals.

Author(s)

Jorge Arce.

References

Arce J. and Rodriguez O. (2015) 'Principal Curves and Surfaces to Interval Valued Variables'. The 5th Workshop on Symbolic Data Analysis, SDA2015, Orleans, France, November.

Hastie,T. (1984).Principal Curves and Surface. Ph.D Thesis Stanford University.

Hastie,T. \& Weingessel,A. (2014). princurve - Fits a Principal Curve in Arbitrary Dimension.R package version 1.1–12 http://cran.r-project.org/web/packages/princurve/index.html.

Hastie,T. \& Stuetzle, W. (1989). Principal Curves. Journal of the American Statistical Association, Vol. 84-406, 502–516.

Hastie, T., Tibshirani, R. \& Friedman, J. (2008). The Elements of Statistical Learning; Data Mining, Inference and Prediction. Springer, New York.

See Also

sym.interval.pc

`cfa.scatterplot` *CFA Symbolic Scatter Plot*

Description

This function could be use to plot two symbolic variables in a X-Y plane to special case of Symbolic Correspondance Analysis.

Usage

```
cfa.scatterplot(sym.var.x, sym.var.y, num.gr1=0, labels = TRUE, ...)
```

Arguments

`sym.var.x`	First symbolic variable
`sym.var.y`	First symbolic variable
`num.gr1`	Number of modes of the first variable
`labels`	As in R plot function.
`...`	As in R plot function.

Value

Return a graphics.

Author(s)

Oldemar Rodriguez Rojas

References

Rodriguez, O. (2011).Correspondence Analysis for Symbolic MultiValued Variables. Workshop in Symbolic Data AnalysisNamur, Belgium.

See Also

sym.cfa

classic.to.sym *Generate a symbolic data table*

Description

Generate a symbolic data table from a classic data table.

Usage

```
classic.to.sym(data, concept, variables, variables.types)
```

Arguments

`data`	A data.frame.
`concept`	These are the variable that we are going to use a concepts.
`variables`	These are the variables that we want to include in the symbolic data table.
`variables.types`	A vector with names and the type of symbolic data to use, the available types are type_histogram (), type_continuous (), type.set (), type.modal (), by default type_histogram () is used for numeric variables and type_modal () for the categorical variables.

Value

The symbolic data table.

Author(s)

Carlos Aguero.

References

Bock H-H. and Diday E. (eds.) (2000). Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data. Springer, Germany.

See Also

read.sym.table

Examples

```
result <- classic.to.sym(data = iris,
                        concept = "Species",
                        variables = c(Sepal.Length,Sepal.Width,Petal.Length,Petal.Width))
result

result <- classic.to.sym(data = iris,
concept = "Species", # concepto
```

```
variables = c(Sepal.Length,Sepal.Width,Petal.Length,Petal.Width), # variable a utilizar
variables.types = c(Sepal.Length = type.interval(), # tipo para cada una de las variable
            Sepal.Width = type.interval(),
            Petal.Length = type.interval(),
            Petal.Width = type.interval()))
result
```

---

cor *Generic function for the correlation*

Description

This function compute the symbolic correlation

Usage

```
cor(x, ...)

## Default S3 method:
cor(x, y = NULL, use = "everything",
  method = c("pearson", "kendall", "spearman"), ...)

## S3 method for class 'sym.data.table'
cor(x, y, method = c("centers", "interval",
  "billard", "modal"), ...)
```

Arguments

`x`	A symbolic variable.
`...`	As in R cor function.
`y`	A symbolic variable.
`use`	An optional character string giving a method for computing covariances in the presence of missing values. This must be (an abbreviation of) one of the strings 'everything', 'all.obs', 'complete.obs', 'na.or.complete', or 'pairwise.complete.obs'.
`method`	The method to be use.

Value

Return a real number in [-1,1].

Author(s)

Oldemar Rodriguez Rojas

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

Examples

```
data(example3)
sym.data <- example3
cor(sym.data[,1], sym.data[,4], method='centers')
cor(sym.data[,2], sym.data[,6], method='centers')
cor(sym.data[,2], sym.data[,6], method='billard')
```

---

cov *Generic function for the covariance*

Description

This function compute the symbolic covariance.

Usage

```
cov(x, ...)

## Default S3 method:
cov(x, y = NULL, use = "everything",
  method = c("pearson", "kendall", "spearman"), ...)

## S3 method for class 'sym.data.table'
cov(x, y, method = c("centers", "interval",
  "billard", "modal"), na.rm = FALSE, ...)
```

Arguments

`x`	First symbolic variables.
`...`	As in R cov function.
`y`	Second symbolic variables.
`use`	an optional character string giving a method for computing covariances in the presence of missing values. This must be (an abbreviation of) one of the strings 'everything', 'all.obs', 'complete.obs', 'na.or.complete', or 'pairwise.complete.obs'.
`method`	The method to be use.
`na.rm`	As in R cov function.

Value

Return a real number.

Author(s)

Oldemar Rodriguez Rojas

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

Examples

```
data(example3)
sym.data <- example3
cov(sym.data[,1], sym.data[,4], method='centers')
cov(sym.data[,2],sym.data[,6], method='centers')
cov(sym.data[,2],sym.data[,6], method='billard')
```

---

deter.coefficient *Compute the determination cosfficient*

Description

The determination coefficient represents a goodness-of-fit measure commonly used in regression analysis to capture the adjustment quality of a model.

Usage

```
deter.coefficient(sym.var, prediction)
```

Arguments

`sym.var`	Variable that was predicted.
`prediction`	The prediction given by the model.

Value

Return the determination cosfficient.

Author(s)

Oldemar Rodriguez Rojas

References

LIMA-NETO, E.A., DE CARVALHO, F.A.T., (2008). Centre and range method to fitting a linear regression model on symbolic interval data. Computational Statistics and Data Analysis 52, 1500-1515.

LIMA-NETO, E.A., DE CARVALHO, F.A.T., (2010). Constrained linear regression models for symbolic interval-valued variables. Computational Statistics and Data Analysis 54, 333-347.

See Also

sym.glm

Examples

```
data(int_prost_test)
data(int_prost_train)
res.cm <- sym.lm(lpsa~., sym.data=int_prost_train, method='cm')
pred.cm <- predictsym.lm(res.cm,int_prost_test,method='cm')
deter.coefficient(sym.var(int_prost_test,9),pred.cm$Fitted)

res.cm.lasso <- sym.glm(sym.data=int_prost_train,response=9,method='cm',
                      alpha=1,nfolds=10,grouped=TRUE)
pred.cm.lasso <- predictsym.glm(res.cm.lasso,response=9,int_prost_test,method='cm')
deter.coefficient(sym.var(int_prost_test,9),pred.cm.lasso)
```

`display.sym.table` *display.sym.table*

Description

This function display a symbolic data table tha have been read by read.sym.table(...)

Usage

```
display.sym.table(sym.data)
```

Arguments

`sym.data`	Shoud be a Symbolic Data table that have been read with read.sym.table(...)

Details

The output will be the symbolic data table in the screen.

Value

Not value.

Author(s)

Oldemar Rodriguez Rojas

References

Billard, L and Diday, E. (2007). Symbolic Data Analysis: Conceptual Statistics and Data Mining (Wiley Series in Computational Statistics). Billard, L and Diday, E. (2003). From the Statistics of Data to the Statistics of Knowledge: Symbolic Data Analysis. Journal of the American of the Statistical Association, USA.

See Also

read.sym.table

Examples

```
data(example3)
display.sym.table(example3)
```

---

dist *Generic function for distance matrix computation*

Description

Generic function for distance matrix computation

Usage

```
dist(x, ...)

## Default S3 method:
dist(x, method = "euclidean", diag = FALSE,
  upper = FALSE, p = 2, ...)

## S3 method for class 'sym.data.table'
dist(x, q = 2, ...)
```

Arguments

`x`	An R object. Currently the are methods for numeric matrix, data.frame, dist object or symbolic data table
`...`	Further arguments passed to or from other methods
`method`	the distance measure to be used. This must be one of "euclidean", "maximum", "manhattan", "canberra", "binary" or "minkowski". Any unambiguous substring can be given.
`diag`	logical value indicating whether the diagonal of the distance matrix should be printed by print.dist.

`upper`	logical value indicating whether the upper triangle of the distance matrix should be printed by print.dist.
`p`	The power of the Minkowski distance.
`q`	q value for the Minkowski method

Value

dist returns an object of class 'dist'

Author(s)

Oldemar Rodriguez Rojas

`dist.modal` *Distance matrix for symbolic data*

Description

Distance matrix for symbolic data

Usage

```
## S3 method for class 'modal'
dist(sym.data)
```

Arguments

`sym.data`	Symbolic data table

Value

return an object of class 'dist

`dist.vect` *Compute a distance vector*

Description

Compute a distance vector

Usage

```
dist.vect(vector1, vector2)
```

Arguments

`vector1`	First vector.
`vector2`	Second vector.

Value

Eclidean distance between the two vectors.

Author(s)

Jorge Arce

References

Arce J. and Rodriguez O. (2015) 'Principal Curves and Surfaces to Interval Valued Variables'. The 5th Workshop on Symbolic Data Analysis, SDA2015, Orleans, France, November.

Hastie,T. (1984). Principal Curves and Surface. Ph.D. Thesis Stanford University.

Hastie,T. \& Weingessel,A. (2014). princurve - Fits a Principal Curve in Arbitrary Dimension.R package version 1.1–12 http://cran.r-project.org/web/packages/princurve/index.html.

Hastie,T. \& Stuetzle, W. (1989). Principal Curves. Journal of the American Statistical Association, Vol. 84-406, 502–516.

Hastie, T., Tibshirani, R. \& Friedman, J. (2008). The Elements of Statistical Learning; Data Mining, Inference and Prediction. Springer, New York.

See Also

sym.interval.pc

`dist.vect.matrix` *Compute the distance vector matrix*

Description

Compute the distance vector matrix.

Usage

```
dist.vect.matrix(vector, Matrix)
```

Arguments

`vector`	An n dimensional vector.
`Matrix`	An n x n matrix.

Value

The distance.

Author(s)

Jorge Arce.

References

Arce J. and Rodriguez O. (2015) 'Principal Curves and Surfaces to Interval Valued Variables'. The 5th Workshop on Symbolic Data Analysis, SDA2015, Orleans, France, November.

Hastie,T. (1984). Principal Curves and Surface. Ph.D Thesis Stanford University.

Hastie,T. \& Weingessel,A. (2014). princurve - Fits a Principal Curve in Arbitrary Dimension.R package version 1.1–12 http://cran.r-project.org/web/packages/princurve/index.html.

Hastie,T. \& Stuetzle, W. (1989). Principal Curves. Journal of the American Statistical Association, Vol. 84-406, 502–516.

Hastie, T., Tibshirani, R. \& Friedman, J. (2008). The Elements of Statistical Learning; Data Mining, Inference and Prediction. Springer, New York.

See Also

sym.interval.pc

`ex1_db2so` *Data example to generate symbolic objets*

Description

This is a small data example to generate symbolic objets.

Usage

```
data(ex1_db2so)
```

Format

An object of class `data.frame` with 19 rows and 5 columns.

References

Bock H-H. and Diday E. (eds.) (2000). Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data. Springer, Germany.

Examples

```
data(ex1_db2so)
ex1 <- ex1_db2so
result <- classic.to.sym(ex1, concept=c('state', 'sex'),
                         variables = c(county, group, age, age),
                         variables.types = c(county = type.continuous(),
                         group = type.histogram(),
                         age = type.histogram(),
                          age = type.set()))
result
```

---

example1 *Data Example 1*

Description

This a symbolic data table with variables of continuos, interval, histogram and set types.

Usage

```
data(example1)
```

Format

The labels $C means that follows a continuous variable, $I means an interval variable, $H means a histogram variables and $S means set variable. In the first row each labels should be follow of a name to variable and to the case of histogram a set variables types the names of the modalities (categories). In data rows for continuous variables we have just one value, for interval variables we have the minimum and the maximum of the interval, for histogram variables we have the number of modalities and then the probability of each modality and for set variables we have the cardinality of the set and next the elements of the set.
The format is the *.csv file is:
$C F1 $I F2 F2 $M F3 M1 M2 M3 $S F4 e a 2 3 g b 1 4 i k c d
Case1 $C 2.8 $I 1 2 $M 3 0.1 0.7 0.2 $S 12 1 0 0 0 1 0 0 0 1 1 0 0
Case2 $C 1.4 $I 3 9 $M 3 0.6 0.3 0.1 $S 12 0 1 0 0 0 1 0 0 0 0 1 1
Case3 $C 3.2 $I -1 4 $M 3 0.2 0.2 0.6 $S 12 0 0 1 0 0 1 1 0 0 0 1 0
Case4 $C -2.1 $I 0 2 $M 3 0.9 0.0 0.1 $S 12 0 1 0 1 0 0 0 1 0 0 1 0
Case5 $C -3.0 $I -4 -2 $M 3 0.6 0.0 0.4 $S 12 1 0 0 0 1 0 0 0 1 1 0 0
The internal format is:
$N
[1] 5
$M
[1] 4
$sym.obj.names
[1] 'Case1' 'Case2' 'Case3' 'Case4' 'Case5'
$sym.var.names
[1] 'F1' 'F2' 'F3' 'F4'

$sym.var.types [1] '$C' '$I' '$H' '$S'
$sym.var.length
[1] 1 2 3 4
$sym.var.starts
[1] 2 4 8 13
$meta

$C F1 $I F2 F2 $M F3 M1 M2 M3 $S F4 e a 2 3 g b 1 4 i k c d Case1 $C 2.8 $I 1 2 $M 3 0.1 0.7 0.2 $S 12 1 0 0 0 1 0 0 0 1 1 0 0 Case2 $C 1.4 $I 3 9 $M 3 0.6 0.3 0.1 $S 12 0 1 0 0 0 1 0 0 0 0 1 1 Case3 $C 3.2 $I -1 4 $M 3 0.2 0.2 0.6 $S 12 0 0 1 0 0 1 1 0 0 0 1 0 Case4 $C -2.1 $I 0 2 $M 3 0.9 0.0 0.1 $S 12 0 1 0 1 0 0 0 1 0 0 1 0 Case5 $C -3.0 $I -4 -2 $M 3 0.6 0.0 0.4 $S 12 1 0 0 0 1 0 0 0 1 1 0 0 $data
F1 F2 F2.1 M1 M2 M3 e a 2 3 g b 1 4 i k c d Case1 2.8 1 2 0.1 0.7 0.2 1 0 0 0 1 0 0 0 1 1 0 0 Case2 1.4 3 9 0.6 0.3 0.1 0 1 0 0 0 1 0 0 0 0 1 1 Case3 3.2 -1 4 0.2 0.2 0.6 0 0 1 0 0 1 1 0 0 0 1 0 Case4 -2.1 0 2 0.9 0.0 0.1 0 1 0 1 0 0 0 1 0 0 1 0 Case5 -3.0 -4 -2 0.6 0.0 0.4 1 0 0 0 1 0 0 0 1 1 0 0

References

Bock H-H. and Diday E. (eds.) (2000). Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data. Springer, Germany.

Examples

```
data(example1)
```

`example2` *Data Example 2*

Description

This a symbolic data table with variables of continuos, interval, histogram and set types.

Usage

```
data(example2)
```

Format

$C F1 $I F2 F2 $M F3 M1 M2 M3 $C F4 $S F5 e a 2 3 g b 1 4 i k c d
Case1 $C 2.8 $I 1 2 $M 3 0.1 0.7 0.2 $C 6.0 $S 12 1 0 0 0 1 0 0 0 1 1 0 0
Case2 $C 1.4 $I 3 9 $M 3 0.6 0.3 0.1 $C 8.0 $S 12 0 1 0 0 0 1 0 0 0 0 1 1
Case3 $C 3.2 $I -1 4 $M 3 0.2 0.2 0.6 $C -7.0 $S 12 0 0 1 0 0 1 1 0 0 0 1 0
Case4 $C -2.1 $I 0 2 $M 3 0.9 0.0 0.1 $C 0.0 $S 12 0 1 0 1 0 0 0 1 0 0 1 0
Case5 $C -3.0 $I -4 -2 $M 3 0.6 0.0 0.4 $C -9.5 $S 12 1 0 0 0 1 0 0 0 1 1 0 0

Examples

```
data(example2)
display.sym.table(example2)
```

`example3` *Data Example 3*

Description

This a symbolic data table with variables of continuos, interval, histogram and set types.

Usage

```
data(example3)
```

Format

$C F1 $I F2 F2 $M F3 M1 M2 M3 $C F4 $S F5 e a 2 3 g b 1 4 i k c d $I F6 F6 $I F7 F7 Case1 $C 2.8 $I 1 2 $M 3 0.1 0.7 0.2 $C 6.0 $S 12 1 0 0 0 1 0 0 0 1 1 0 0 $I 0.00 90.00 $I 9 24 Case2 $C 1.4 $I 3 9 $M 3 0.6 0.3 0.1 $C 8.0 $S 12 0 1 0 0 0 1 0 0 0 0 1 1 $I -90.00 98.00 $I -9 9 Case3 $C 3.2 $I -1 4 $M 3 0.2 0.2 0.6 $C -7.0 $S 12 0 0 1 0 0 1 1 0 0 0 1 0 $I 65.00 90.00 $I 65 70 Case4 $C -2.1 $I 0 2 $M 3 0.9 0.0 0.1 $C 0.0 $S 12 0 1 0 1 0 0 0 1 0 0 1 0 $I 45.00 89.00 $I 25 67 Case5 $C -3.0 $I -4 -2 $M 3 0.6 0.0 0.4 $C -9.5 $S 12 1 0 0 0 1 0 0 0 1 1 0 0 $I 20.00 40.00 $I 9 40 Case6 $C 0.1 $I 10 21 $M 3 0.0 0.7 0.3 $C -1.0 $S 12 1 0 0 0 0 0 1 0 1 0 0 0 $I 5.00 8.00 $I 5 8 Case7 $C 9.0 $I 4 21 $M 3 0.2 0.2 0.6 $C 0.5 $S 12 1 1 1 0 0 0 0 0 0 0 0 0 $I 3.14 6.76 $I 4 6

Examples

```
data(example3)
display.sym.table(example3)
```

`example4` *Data Example 4*

Description

data(example4) display.sym.table(example4)

Usage

```
data(example4)
```

Format

$C 2.8 $I 1 2 $M 3 0.1 0.7 0.2 $C 6 $S F4 e a 2 3 g b 1 4 i k c d $I 0 90 Case2 $C 1.4 $I 3 9 $M 3 0.6 0.3 0.1 $C 8.0 $S 12 1 0 0 0 1 0 0 0 1 1 0 0 $I -90.00 98.00 Case3 $C 3.2 $I -1 4 $M 3 0.2 0.2 0.6 $C -7.0 $S 12 0 1 0 0 0 1 0 0 0 0 1 1 $I 65.00 90.00 Case4 $C -2.1 $I 0 2 $M 3 0.9 0.0 0.1 $C 0.0 $S 12 0 0 1 0 0 1 1 0 0 0 1 0 $I 45.00 89.00 Case5 $C -3.0 $I -4 -2 $M 3 0.6 0.0 0.4 $C -9.5 $S 12 0 1 0 1 0 0 0 1 0 0 1 0 $I 90.00 990.00 Case6 $C 0.1 $I 10 21 $M 3 0.0 0.7 0.3 $C -1.0 $S 12 1 0 0 0 1 0 0 0 1 1 0 0 $I 5.00 8.00 Case7 $C 9.0 $I 4 21 $M 3 0.2 0.2 0.6 $C 0.5 $S 12 1 1 0 0 0 0 1 0 0 0 0 1 $I 3.14 6.76

Examples

```
data(example4)
display.sym.table(example4)
```

---

example5 *Data Example 5*

Description

This a symbolic data matrix wint continuos, interval, histograma a set data types.

Usage

```
data(example5)
```

Format

$H F0 M01 M02 $C F1 $I F2 F2 $H F3 M1 M2 M3 $S F4 E1 E2 E3 E4
Case1 $H 2 0.1 0.9 $C 2.8 $I 1 2 $H 3 0.1 0.7 0.2 $S 4 e g k i
Case2 $H 2 0.7 0.3 $C 1.4 $I 3 9 $H 3 0.6 0.3 0.1 $S 4 a b c d
Case3 $H 2 0.0 1.0 $C 3.2 $I -1 4 $H 3 0.2 0.2 0.6 $S 4 2 1 b c
Case4 $H 2 0.2 0.8 $C -2.1 $I 0 2 $H 3 0.9 0.0 0.1 $S 4 3 4 c a
Case5 $H 2 0.6 0.4 $C -3.0 $I -4 -2 $H 3 0.6 0.0 0.4 $S 4 e i g k

Examples

```
data(example5)
display.sym.table(example5)
```

example6 *Data Example 6*

Description

This a symbolic data matrix wint continuos, interval, histograma a set data types.

Usage

```
data(example6)
```

Format

$C F1 $M F2 M1 M2 M3 M4 M5 $I F3 F3 $M F4 M1 M2 M3 $C F5 $S F4 e a 2 3 g b 1 4 i k c d
Case1 $C 2.8 $M 5 0.1 0.1 0.1 0.1 0.6 $I 1 2 $M 3 0.1 0.7 0.2 $C 6.0 $S 12 1 0 0 0 1 0 0 0 1 1 0 0
Case2 $C 1.4 $M 5 0.1 0.1 0.1 0.1 0.6 $I 3 9 $M 3 0.6 0.3 0.1 $C 8.0 $S 12 0 1 0 0 0 1 0 0 0 0 1 1
Case3 $C 3.2 $M 5 0.1 0.1 0.1 0.1 0.6 $I -1 4 $M 3 0.2 0.2 0.6 $C -7.0 $S 12 0 0 1 0 0 1 1 0 0 0 1 0
Case4 $C -2.1 $M 5 0.1 0.1 0.1 0.1 0.6 $I 0 2 $M 3 0.9 0.0 0.1 $C 0.0 $S 12 0 1 0 1 0 0 0 1 0 0 1 0
Case5 $C -3.0 $M 5 0.1 0.1 0.1 0.1 0.6 $I -4 -2 $M 3 0.6 0.0 0.4 $C -9.5 $S 12 1 0 0 0 1 0 0 0 1 1 0
0

Examples

```
data(example6)
display.sym.table(example6)
```

example7 *Data Example 7*

Description

This a symbolic data matrix wint continuos, interval, histograma a set data types.

Usage

```
data(example6)
```

Format

$C F1 $H F2 M1 M2 M3 M4 M5 $I F3 F3 $H F4 M1 M2 M3 $C F5
Case1 $C 2.8 $H 5 0.1 0.2 0.3 0.4 0.0 $I 1 2 $H 3 0.1 0.7 0.2 $C 6.0
Case2 $C 1.4 $H 5 0.2 0.1 0.5 0.1 0.2 $I 3 9 $H 3 0.6 0.3 0.1 $C 8.0
Case3 $C 3.2 $H 5 0.1 0.1 0.2 0.1 0.5 $I -1 4 $H 3 0.2 0.2 0.6 $C -7.0
Case4 $C -2.1 $H 5 0.4 0.1 0.1 0.1 0.3 $I 0 2 $H 3 0.9 0.0 0.1 $C 0.0
Case5 $C -3.0 $H 5 0.6 0.1 0.1 0.1 0.1 $I -4 -2 $H 3 0.6 0.0 0.4 $C -9.5

Examples

```
data(example7)
display.sym.table(example7)
```

---

ex_cfa1 *Correspondence Analysis Example*

Description

Correspondence Analysis for Symbolic MultiValued Variables example.

Usage

```
data(ex_cfa1)
```

Format

An object of class `list` of length 9.

References

Rodriguez, O. (2011). Correspondence Analysis for Symbolic MultiValued Variables. Workshop in Symbolic Data Analysis Namur, Belgium

---

ex_cfa2 *Correspondence Analysis Example*

Description

Correspondence Analysis for Symbolic MultiValued Variables example.

Usage

```
data(ex_cfa2)
```

Format

An object of class `list` of length 9.

References

Rodriguez, O. (2011). Correspondence Analysis for Symbolic MultiValued Variables. Workshop in Symbolic Data Analysis Namur, Belgium

`ex_mcfa1` *ex_mcfa1*

Description

example for the sym.mcfa function.

Usage

```
ex_mcfa1
```

Format

An object of class `data.frame` with 130 rows and 5 columns.

`ex_mcfa2` *ex_mcfa2*

Description

example for the sym.mcfa function.

Usage

```
ex_mcfa2
```

Format

An object of class `data.frame` with 130 rows and 7 columns.

`facedata` *Face Data Example*

Description

Symbolic data matrix with all the variables of interval type.

Usage

```
data('facedata')
```

Format

$I;AD;AD;$I;BC;BC;.........

HUS1;$I;168.86;172.84;$I;58.55;63.39;.........
HUS2;$I;169.85;175.03;$I;60.21;64.38;.........
HUS3;$I;168.76;175.15;$I;61.4;63.51;.........
INC1;$I;155.26;160.45;$I;53.15;60.21;.........
INC2;$I;156.26;161.31;$I;51.09;60.07;.........
INC3;$I;154.47;160.31;$I;55.08;59.03;.........
ISA1;$I;164;168;$I;55.01;60.03;.........
ISA2;$I;163;170;$I;54.04;59;.........
ISA3;$I;164.01;169.01;$I;55;59.01;.........
JPL1;$I;167.11;171.19;$I;61.03;65.01;.........
JPL2;$I;169.14;173.18;$I;60.07;65.07;.........
JPL3;$I;169.03;170.11;$I;59.01;65.01;.........
KHA1;$I;149.34;155.54;$I;54.15;59.14;.........
KHA2;$I;149.34;155.32;$I;52.04;58.22;.........
KHA3;$I;150.33;157.26;$I;52.09;60.21;.........
LOT1;$I;152.64;157.62;$I;51.35;56.22;.........
LOT2;$I;154.64;157.62;$I;52.24;56.32;.........
LOT3;$I;154.83;157.81;$I;50.36;55.23;.........
PHI1;$I;163.08;167.07;$I;66.03;68.07;.........
PHI2;$I;164;168.03;$I;65.03;68.12;.........
PHI3;$I;161.01;167;$I;64.07;69.01;.........
ROM1;$I;167.15;171.24;$I;64.07;68.07;.........
ROM2;$I;168.15;172.14;$I;63.13;68.07;.........
ROM3;$I;167.11;171.19;$I;63.13;68.03;.........

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Examples

```
## Not run:
data(facedata)
res.vertex.ps<-sym.interval.pc(facedata,'vertex',150,FALSE,FALSE,TRUE)
class(res.vertex.ps$sym.prin.curve) <- c('sym.data.table')
sym.scatterplot(res.vertex.ps$sym.prin.curve[,1], res.vertex.ps$sym.prin.curve[,2],
                labels=TRUE,col='red',main='PSC Face Data')

## End(Not run)
```

generate.sym.table *Generate a Symbolic Data Table*

Description

This function generates a symbolic data table from a CSV data file.

Usage

```
generate.sym.table(sym.data, file, sep, dec, row.names = NULL, col.names = NULL)
```

Arguments

sym.data	Symbolic data table.
file	The name of the CSV file.
sep	As in R function read.table.
dec	As in R function read.table.
row.names	As in R function read.table.
col.names	As in R function read.table.

Value

Return a symbolic data table.

Author(s)

Oldemar Rodriguez Rojas

References

Bock H-H. and Diday E. (eds.) (2000). Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data. Springer, Germany.

Examples

```
data(example1)
generate.sym.table(example1, file='temp4.csv', sep='|',dec='.', row.names=TRUE,
                   col.names=TRUE)
ex1 <- read.sym.table('temp4.csv', header=TRUE, sep='|',dec='.', row.names=1)
```

interscal *Interscal Method*

Description

Execute Interscal Method.

Usage

```
interscal(sym.data)
```

Arguments

sym.data	The symbolic data matrix.

Value

The symbolic interval components.

Author(s)

Oldemar Rodriguez Rojas

References

Groenen, P.J.F., Winsberg, S., Rodriguez, O., Diday, E. (2006). I-Scal: Multidimensional scaling of interval dissimilarities. Computational Statistics and Data Analysis, 51, 360-378.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University

See Also

sym.interval.pca

Examples

```
## Not run:
data(oils)
res<-interscal(oils)
class(res$Sym.Components) <- c('sym.data.table')
sym.scatterplot(res$Sym.Components[,1], res$Sym.Components[,2],
                labels=TRUE,col='red',main='Interscal CFA Data')
sym.scatterplot3d(res$Sym.Components[,1], res$Sym.Components[,2],
                  res$Sym.Components[,3],color='blue',
                  labels=TRUE,main='Interscal CFA Data')
sym.scatterplot.ggplot(res$Sym.Components[,1],res$Sym.Components[,2],
                       labels=TRUE)

## End(Not run)
```

`interval.dist` *Interval Distance Matrix*

Description

Compute a distance matrix from a symbolic interval data matrix.

Usage

```
interval.dist(sym.data, distance = c('hausdorff', 'centers', 'interscal'), p = 2)
```

Arguments

`sym.data`	Symbolic data matrix with the variables of interval type.
`distance`	The distance to be use.
`p`	The p in the Hausdorff distance :

$$d(w_{u_1}, w_{u_2}) = \left(\sum_{j=1}^{m} \Phi_j(w_{u_1}, w_{u_2})^p \right)^{1/p}$$

Value

Return a R distance triangular matrix.

Author(s)

Oldemar Rodriguez Rojas

References

Groenen, P.J.F., Winsberg, S., Rodriguez, O., Diday, E. (2006). I-Scal: Multidimensional scaling of interval dissimilarities. Computational Statistics and Data Analysis, 51, 360-378.

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Examples

```
## Not run:
data(VeterinaryData)
VD <- VeterinaryData
interval.dist(VD)
interval.dist(VD,distance='centers')

## End(Not run)
```

interval.dist.tobj *Symbolic Objects Distance*

Description

Compute a distance between two symbolic objects.

Usage

```
interval.dist.tobj(sym.obj.x, sym.obj.y, distance = c('hausdorff',
'centers', 'interscal'), p = 2)
```

Arguments

sym.obj.x	First Symbolic Object
sym.obj.y	Second Symbolic Object
distance	Dsitance to be use
p	The p in the Hausdorff distance

$$d(w_{u_1}, w_{u_2}) = \left(\sum_{j=1}^{m} \Phi_j(w_{u_1}, w_{u_2})^p \right)^{1/p}$$

Value

Return a real number that is the distance between sym.obj.x and sym.obj.y

Author(s)

Oldemar Rodriguez Rojas

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

See Also

interval.dist

Examples

```
data(VeterinaryData)
VD <- VeterinaryData
interval.dist.tobj(sym.obj(VD,1),sym.obj(VD,2))
interval.dist.tobj(sym.obj(VD,1),sym.obj(VD,2),distance='centers')
```

`interval.histogram.plot` *Histogram plot for an interval variable*

Description

Histogram plot for an interval variable

Usage

```
interval.histogram.plot(x, n.bins, ...)
```

Arguments

`x`	An symbolic data table.
`n.bins`	Numbers of breaks of the histogram.
`...`	Arguments to be passed to the barplot method.

Value

A list with componets : frequency and histogram

Examples

```
data(oils)
res <- interval.histogram.plot(x = oils[,3], n.bins = 3)
res
```

`int_prost_test` *Linear regression model data example.*

Description

Linear regression model interval-valued data example.

Usage

```
data(int_prost_test)
```

Format

An object of class `list` of length 9.

References

HASTIE, T., TIBSHIRANI, R. and FRIEDMAN, J. (2008). The Elements of Statistical Learning: Data Mining, Inference and Prediction. New York: Springer.

Examples

```
data(int_prost_train)
data(int_prost_test)
res.cm<-sym.lm(lpsa~.,sym.data=int_prost_test,method='cm')
res.cm<-sym.lm(lpsa~.,sym.data=int_prost_train,method='cm')
pred.cm<-predictsym.lm(res.cm,int_prost_test,method='cm')
RMSE.L(sym.var(int_prost_test,9),pred.cm$Fitted)
RMSE.U(sym.var(int_prost_test,9),pred.cm$Fitted)
R2.L(sym.var(int_prost_test,9),pred.cm$Fitted)
R2.U(sym.var(int_prost_test,9),pred.cm$Fitted)
deter.coefficient(sym.var(int_prost_test,9),pred.cm$Fitted)

res.cm.lasso<-sym.glm(sym.data=int_prost_train,response=9,method='cm',
                      alpha=1,nfolds=10,grouped=TRUE)
plot(res.cm.lasso)
plot(res.cm.lasso$glmnet.fit, 'norm', label=TRUE)
plot(res.cm.lasso$glmnet.fit, 'lambda', label=TRUE)
pred.cm.lasso<-predictsym.glm(res.cm.lasso,response=9,int_prost_test,method='cm')
RMSE.L(sym.var(int_prost_test,9),pred.cm.lasso)
RMSE.U(sym.var(int_prost_test,9),pred.cm.lasso)
R2.L(sym.var(int_prost_test,9),pred.cm.lasso)
R2.U(sym.var(int_prost_test,9),pred.cm.lasso)
deter.coefficient(sym.var(int_prost_test,9),pred.cm.lasso)
```

---

int_prost_train *Linear regression model data example.*

Description

Linear regression model interval-valued data example.

Usage

```
data(int_prost_train)
```

Format

An object of class `list` of length 9.

References

HASTIE, T., TIBSHIRANI, R. and FRIEDMAN, J. (2008). The Elements of Statistical Learning: Data Mining, Inference and Prediction. New York: Springer.

Examples

```
data(int_prost_train)
data(int_prost_test)
res.cm<-sym.lm(lpsa~.,sym.data=int_prost_test,method='cm')
res.cm<-sym.lm(lpsa~.,sym.data=int_prost_train,method='cm')
pred.cm<-predictsym.lm(res.cm,int_prost_test,method='cm')
RMSE.L(sym.var(int_prost_test,9),pred.cm$Fitted)
RMSE.U(sym.var(int_prost_test,9),pred.cm$Fitted)
R2.L(sym.var(int_prost_test,9),pred.cm$Fitted)
R2.U(sym.var(int_prost_test,9),pred.cm$Fitted)
deter.coefficient(sym.var(int_prost_test,9),pred.cm$Fitted)

res.cm.lasso<-sym.glm(sym.data=int_prost_train,response=9,method='cm',
                      alpha=1,nfolds=10,grouped=TRUE)
plot(res.cm.lasso)
plot(res.cm.lasso$glmnet.fit, 'norm', label=TRUE)
plot(res.cm.lasso$glmnet.fit, 'lambda', label=TRUE)
pred.cm.lasso<-predictsym.glm(res.cm.lasso,response=9,int_prost_test,method='cm')
RMSE.L(sym.var(int_prost_test,9),pred.cm.lasso)
RMSE.U(sym.var(int_prost_test,9),pred.cm.lasso)
R2.L(sym.var(int_prost_test,9),pred.cm.lasso)
R2.U(sym.var(int_prost_test,9),pred.cm.lasso)
deter.coefficient(sym.var(int_prost_test,9),pred.cm.lasso)
```

---

lynne1 — *Symbolic interval data example.*

Description

Symbolic data matrix with all the variables of interval type.

Usage

```
data(lynne1)
```

Format

An object of class `sym.data.table` of length 9.

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Examples

```
data(lynne1)
display.sym.table(lynne1)
```

lynne2 — *Symbolic interval data example.*

Description

Symbolic data matrix with all the variables of interval type.

Usage

```
data(lynne2)
```

Format

An object of class `sym.data.table` of length 9.

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Examples

```
data(lynne2)
display.sym.table(lynne2)
```

mcfa.scatterplot — *Plot Interval Scatterplot*

Description

Plot Interval Scatterplot

Usage

```
mcfa.scatterplot(x, y, sym.data, pos.var)
```

Arguments

`x`	symbolic table with only one column.
`y`	symbolic table with only one column.
`sym.data`	original symbolic table.
`pos.var`	column number of the variables to be plotted.

Examples

```
data("ex_mcfa1")
sym.table <- classic.to.sym(ex_mcfa1, concept = "suspect",
                    variables.types = c(hair = type.set(),
                                        eyes = type.set(),
                                        region = type.set()))

res <- sym.mcfa(sym.table, c(1,2))
mcfa.scatterplot(res[,1], res[,2], sym.data = sym.table, pos.var = c(1,2))
```

---

mean.sym.data.table *Symbolic Mean*

Description

This function compute the symbolic mean

Usage

```
## S3 method for class 'sym.data.table'
mean(x, method = c("centers", "interval", "modal"),
  trim = 0, na.rm = F, ...)
```

Arguments

x	The symbolic variable.
method	The method to be use.
trim	As in R mean function.
na.rm	As in R mean function.
...	As in R mean function.

Value

Return a real number.

Author(s)

Oldemar Rodriguez Rojas

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

Examples

```
data(example3)
sym.data<-example3
mean(sym.data[,1])
mean(sym.data[,2])
mean(sym.data[,2], method='interval')
mean(sym.data[,3], method='modal')
```

---

median.sym.data.table *Symbolic Median*

Description

This function compute the symbolic median.

Usage

```
## S3 method for class 'sym.data.table'
median(x, na.rm = FALSE, method = c("centers",
  "interval", "modal"), ...)
```

Arguments

x	The symbolic variable.
na.rm	As in R median function.
method	The method to be use.
...	As in R median function.

Value

Return a real number.

Author(s)

Oldemar Rodriguez Rojas

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

Examples

```
data(example3)
sym.data<-example3
median(sym.data[,2])
median(sym.data[,6] ,method='interval')
median(sym.data[,3] ,method='modal')
```

`neighbors.vertex` *Compute neighbors vertex*

Description

Compute neighbors vertex

Usage

```
neighbors.vertex(vertex, Matrix, num.neig)
```

Arguments

`vertex`	Vertes of the hipercube
`Matrix`	Interval Data Matrix.
`num.neig`	Number of vertices.

Author(s)

Jorge Arce

References

Arce J. and Rodriguez O. (2015) 'Principal Curves and Surfaces to Interval Valued Variables'. The 5th Workshop on Symbolic Data Analysis, SDA2015, Orleans, France, November.

Hastie,T. (1984). Principal Curves and Surface. Ph.D Thesis Stanford University.

Hastie,T. \& Weingessel,A. (2014). princurve - Fits a Principal Curve in Arbitrary Dimension.R package version 1.1–12 http://cran.r-project.org/web/packages/princurve/index.html.

Hastie,T. \& Stuetzle, W. (1989). Principal Curves. Journal of the American Statistical Association, Vol. 84-406, 502–516.

Hastie, T., Tibshirani, R. \& Friedman, J. (2008). The Elements of Statistical Learning; Data Mining, Inference and Prediction. Springer, New York.

See Also

sym.interval.pc

`norm.vect` *Compute the norm of a vector.*

Description

Compute the norm of a vector.

Usage

```
norm.vect(vector1)
```

Arguments

`vector1`	An n dimensional vector.

Value

The L2 norm of the vector.

Author(s)

Jorge Arce

References

Arce J. and Rodriguez O. (2015) 'Principal Curves and Surfaces to Interval Valued Variables'. The 5th Workshop on Symbolic Data Analysis, SDA2015, Orleans, France, November.

Hastie,T. (1984). Principal Curves and Surface. Ph.D Thesis Stanford University.

Hastie,T. \& Weingessel,A. (2014). princurve - Fits a Principal Curve in Arbitrary Dimension.R package version 1.1–12 http://cran.r-project.org/web/packages/princurve/index.html.

Hastie,T. \& Stuetzle, W. (1989). Principal Curves. Journal of the American Statistical Association, Vol. 84-406, 502–516.

Hastie, T., Tibshirani, R. \& Friedman, J. (2008). The Elements of Statistical Learning; Data Mining, Inference and Prediction. Springer, New York.

See Also

sym.interval.pc

`oils` *Ichino Oils example data.*

Description

Symbolic data matrix with all the variables of interval type.

Usage

```
data(oils)
```

Format

$I GRA GRA $I FRE FRE $I IOD IOD $I SAP SAP
L $I 0.930 0.935 $I -27 -18 $I 170 204 $I 118 196
P $I 0.930 0.937 $I -5 -4 $I 192 208 $I 188 197
Co $I 0.916 0.918 $I -6 -1 $I 99 113 $I 189 198
S $I 0.920 0.926 $I -6 -4 $I 104 116 $I 187 193
Ca $I 0.916 0.917 $I -25 -15 $I 80 82 $I 189 193
O $I 0.914 0.919 $I 0 6 $I 79 90 $I 187 196
B $I 0.860 0.870 $I 30 38 $I 40 48 $I 190 199
H $I 0.858 0.864 $I 22 32 $I 53 77 $I 190 202

References

Cazes P., Chouakria A., Diday E. et Schektman Y. (1997). Extension de l'analyse en composantes principales a des donnees de type intervalle, Rev. Statistique Appliquee, Vol. XLV Num. 3 pag. 5-24, France.

Examples

```
data(oils)
display.sym.table(oils)
```

`pander.sym.data.table` *Pander method for symbolic data table*

Description

Prints a symbolic data table in Pandoc's markdown

Usage

```
## S3 method for class 'sym.data.table'
pander(x, caption = attr(x, "caption"), ...)
```

Arguments

`x`	a symbolic data table
`caption`	caption (string) to be shown under the table
`...`	optional parameters passed to raw pandoc.table function

`plot.sym.data.table` *Function for plotting a symbolic object*

Description

Function for plotting a symbolic object

Usage

```
## S3 method for class 'sym.data.table'
plot(x, col = NA, matrix.form = NA,
  border = FALSE, size = 1, title = TRUE, show.type = FALSE,
  font.size = 1, reduce = FALSE, hist.angle.x = 60, ...)
```

Arguments

`x`	The symbolic object.
`col`	A specification for the default plotting color.
`matrix.form`	A vector of the form c(num.rows,num.columns).
`border`	A logical value indicating whether border should be plotted.
`size`	The magnification to be used for each graphic.
`title`	A logical value indicating whether title should be plotted.
`show.type`	A logical value indicating whether type should be plotted.
`font.size`	The font size of graphics.
`reduce`	A logical value indicating whether values different from zero should be plotted in modal and set graphics.
`hist.angle.x`	The angle of labels in y axis. Only for histogram plot
`...`	Arguments to be passed to methods.

Value

A plot of the symbolic data table.

Author(s)

Andres Navarro

Examples

```
## Not run:
data(oils)
plot(oils)
plot(oils,border = T,size = 1.3)

## End(Not run)
```

---

predictsym.glm — *Predict method to Lasso, Ridge and and Elastic Net Linear regression model to interval variables*

Description

To execute Predict method to Lasso, Ridge and and Elastic Net Linear regression model to interval variables.

Usage

```
predictsym.glm(model, new.sym.data, response = 1, method = c('cm', 'crm'))
```

Arguments

model	The output of glm method.
new.sym.data	Should be a symbolic data table read with the function read.sym.table(...).
response	The number of the column where is the response variable in the interval data table.
method	'cm' to generalized Center Method and 'crm' to generalized Center and Range Method.

Value

The object returned depends the ... argument which is passed on to the predict method for glmnet objects.

Author(s)

Oldemar Rodriguez Rojas

References

Rodriguez O. (2013). A generalization of Centre and Range method for fitting a linear regression model to symbolic interval data using Ridge Regression, Lasso and Elastic Net methods. The IFCS2013 conference of the International Federation of Classification Societies, Tilburg University Holland.

See Also

sym.glm

Examples

```
data(int_prost_train)
data(int_prost_test)
res.cm.lasso<-sym.glm(sym.data=int_prost_train,response=9,method='cm',
                      alpha=1,nfolds=10,grouped=TRUE)
pred.cm.lasso<-predictsym.glm(res.cm.lasso,response=9,int_prost_test,method='cm')
plot(res.cm.lasso)
plot(res.cm.lasso$glmnet.fit, 'norm', label=TRUE)
plot(res.cm.lasso$glmnet.fit, 'lambda', label=TRUE)
RMSE.L(sym.var(int_prost_test,9),pred.cm.lasso)
RMSE.U(sym.var(int_prost_test,9),pred.cm.lasso)
R2.L(sym.var(int_prost_test,9),pred.cm.lasso)
R2.U(sym.var(int_prost_test,9),pred.cm.lasso)
deter.coefficient(sym.var(int_prost_test,9),pred.cm.lasso)
```

---

predictsym.lm *Predict method to CM and CRM Linear regression model*

Description

To execute predict method the Center Method (CR) and Center and Range Method (CRM) to Linear regression.

Usage

```
predictsym.lm(model, new.sym.data, method = c('cm', 'crm'))
```

Arguments

`model`	The output of lm method.
`new.sym.data`	Should be a symbolic data table read with the function read.sym.table(...).
`method`	'cm' to Center Method and 'crm' to Center and Range Method.

Value

predictsym.lm produces a vector of predictions or a matrix of predictions and bounds with column names fit, lwr, and upr if interval is set. For type = 'terms' this is a matrix with a column per term and may have an attribute 'constant'

Author(s)

Oldemar Rodriguez Rojas

References

LIMA-NETO, E.A., DE CARVALHO, F.A.T., (2008). Centre and range method to fitting a linear regression model on symbolic interval data. Computational Statistics and Data Analysis 52, 1500-1515.

LIMA-NETO, E.A., DE CARVALHO, F.A.T., (2010). Constrained linear regression models for symbolic interval-valued variables. Computational Statistics and Data Analysis 54, 333-347.

See Also

sym.glm

Examples

```
data(int_prost_train)
data(int_prost_test)
res.cm <- sym.lm(lpsa~.,sym.data=int_prost_train,method='cm')
pred.cm <- predictsym.lm(res.cm,int_prost_test,method='cm')
```

`print.sym.data.table` *Printing Symbolic Data Table*

Description

Printing Symbolic Data Table

Usage

```
## S3 method for class 'sym.data.table'
print(x, ...)
```

Arguments

`x`	Object of class sym.data.table
`...`	optional arguments to print o format method

R2.L *Lower boundary correlation coefficient.*

Description

Compute the lower boundary correlation coefficient for two interval variables.

Usage

```
R2.L(sym.var, prediction)
```

Arguments

`sym.var`	Variable that was predicted.
`prediction`	The prediction given by the model.

Value

The lower boundary correlation coefficient.

Author(s)

Oldemar Rodriguez Rojas

References

LIMA-NETO, E.A., DE CARVALHO, F.A.T., (2008). Centre and range method to fitting a linear regression model on symbolic interval data. Computational Statistics and Data Analysis 52, 1500-1515.

LIMA-NETO, E.A., DE CARVALHO, F.A.T., (2010). Constrained linear regression models for symbolic interval-valued variables. Computational Statistics and Data Analysis 54, 333-347.

See Also

sym.glm

Examples

```
data(int_prost_train)
data(int_prost_test)
res.cm<-sym.lm(lpsa~.,sym.data=int_prost_train,method='cm')
pred.cm<-predictsym.lm(res.cm,int_prost_test,method='cm')
R2.L(sym.var(int_prost_test,9),pred.cm$Fitted)

res.cm.lasso<-sym.glm(sym.data=int_prost_train,response=9,method='cm',
                      alpha=1,nfolds=10,grouped=TRUE)
pred.cm.lasso<-predictsym.glm(res.cm.lasso,response=9,int_prost_test,method='cm')
R2.L(sym.var(int_prost_test,9),pred.cm.lasso)
```

R2.U *Upper boundary correlation coefficient.*

Description

Compute the upper boundary correlation coefficient for two interval variables.

Usage

```
R2.U(sym.var, prediction)
```

Arguments

sym.var	Variable that was predicted.
prediction	The prediction given by the model.

Value

The upper boundary correlation coefficient.

Author(s)

Oldemar Rodriguez Rojas

References

LIMA-NETO, E.A., DE CARVALHO, F.A.T., (2008). Centre and range method to fitting a linear regression model on symbolic interval data. Computational Statistics and Data Analysis 52, 1500-1515.

LIMA-NETO, E.A., DE CARVALHO, F.A.T., (2010). Constrained linear regression models for symbolic interval-valued variables. Computational Statistics and Data Analysis 54, 333-347.

See Also

sym.glm

Examples

```
data(int_prost_train)
data(int_prost_test)
res.cm<-sym.lm(lpsa~.,sym.data=int_prost_train,method='cm')
pred.cm<-predictsym.lm(res.cm,int_prost_test,method='cm')
R2.U(sym.var(int_prost_test,9),pred.cm$Fitted)

res.cm.lasso<-sym.glm(sym.data=int_prost_train,response=9,method='cm',
                      alpha=1,nfolds=10,grouped=TRUE)
pred.cm.lasso<-predictsym.glm(res.cm.lasso,response=9,int_prost_test,method='cm')
R2.U(sym.var(int_prost_test,9),pred.cm.lasso)
```

read.sym.table *Read a Symbolic Table*

Description

It reads a symbolic data table from a CSV file.

Usage

```
read.sym.table(file, header = TRUE, sep, dec, row.names = NULL)
```

Arguments

file	The name of the CSV file.
header	As in R function read.table
sep	As in R function read.table
dec	As in R function read.table
row.names	As in R function read.table

Details

The labels $C means that follows a continuous variable, $I means an interval variable, $H means a histogram variables and $S means set variable. In the first row each labels should be follow of a name to variable and to the case of histogram a set variables types the names of the modalities (categories) . In data rows for continuous variables we have just one value, for interval variables we have the minimum and the maximum of the interval, for histogram variables we have the number of modalities and then the probability of each modality and for set variables we have the cardinality of the set and next the elements of the set.

The format is the CSV file should be like:

$C F1 $I F2 F2 $H F3 M1 M2 M3 $S F4 E1 E2 E3 E4

Case1 $C 2.8 $I 1 2 $H 3 0.1 0.7 0.2 $S 4 e g k i

Case2 $C 1.4 $I 3 9 $H 3 0.6 0.3 0.1 $S 4 a b c d

Case3 $C 3.2 $I -1 4 $H 3 0.2 0.2 0.6 $S 4 2 1 b c

Case4 $C -2.1 $I 0 2 $H 3 0.9 0.0 0.1 $S 4 3 4 c a

Case5 $C -3.0 $I -4 -2 $H 3 0.6 0.0 0.4 $S 4 e i g k

The internal format is:
$N

[1] 5
$M
[1] 4
$sym.obj.names
[1] 'Case1' 'Case2' 'Case3' 'Case4' 'Case5'
$sym.var.names
[1] 'F1' 'F2' 'F3' 'F4'
$sym.var.types
[1] '$C' '$I' '$H' '$S'
$sym.var.length
[1] 1 2 3 4
$sym.var.starts
[1] 2 4 8 13
$meta
$C F1 $I F2 F2 $H F3 M1 M2 M3 $S F4 E1 E2 E3 E4
Case1 $C 2.8 $I 1 2 $H 3 0.1 0.7 0.2 $S 4 e g k i
Case2 $C 1.4 $I 3 9 $H 3 0.6 0.3 0.1 $S 4 a b c d
Case3 $C 3.2 $I -1 4 $H 3 0.2 0.2 0.6 $S 4 2 1 b c
Case4 $C -2.1 $I 0 2 $H 3 0.9 0.0 0.1 $S 4 3 4 c a
Case5 $C -3.0 $I -4 -2 $H 3 0.6 0.0 0.4 $S 4 e i g k
$data
F1 F2 F2.1 M1 M2 M3 E1 E2 E3 E4
Case1 2.8 1 2 0.1 0.7 0.2 e g k i
Case2 1.4 3 9 0.6 0.3 0.1 a b c d
Case3 3.2 -1 4 0.2 0.2 0.6 2 1 b c
Case4 -2.1 0 2 0.9 0.0 0.1 3 4 c a
Case5 -3.0 -4 -2 0.6 0.0 0.4 e i g k

Value

Return a symbolic data table structure.

Author(s)

Oldemar Rodriguez Rojas

References

Bock H-H. and Diday E. (eds.) (2000). Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data. Springer, Germany.

See Also

display.sym.table

Examples

```
## Not run:
data(example1)
```

```
write.sym.table(example1, file='temp4.csv', sep='|',dec='.', row.names=TRUE,
                col.names=TRUE)
ex1<-read.sym.table('temp4.csv', header=TRUE, sep='|',dec='.', row.names=1)

## End(Not run)
```

RMSE.L *Lower boundary root-mean-square error*

Description

Compute the lower boundary root-mean-square error.

Usage

```
RMSE.L(sym.var, prediction)
```

Arguments

`sym.var`	Variable that was predicted.
`prediction`	The prediction given by the model.

Value

The lower boundary root-mean-square error.

Author(s)

Oldemar Rodriguez Rojas.

References

LIMA-NETO, E.A., DE CARVALHO, F.A.T., (2008). Centre and range method to fitting a linear regression model on symbolic interval data. Computational Statistics and Data Analysis 52, 1500-1515.

LIMA-NETO, E.A., DE CARVALHO, F.A.T., (2010). Constrained linear regression models for symbolic interval-valued variables. Computational Statistics and Data Analysis 54, 333-347.

See Also

sym.glm

Examples

```
data(int_prost_train)
data(int_prost_test)
res.cm<-sym.lm(lpsa~.,sym.data=int_prost_train,method='cm')
pred.cm<-predictsym.lm(res.cm,int_prost_test,method='cm')
RMSE.L(sym.var(int_prost_test,9),pred.cm$Fitted)

res.cm.lasso<-sym.glm(sym.data=int_prost_train,response=9,method='cm',
                      alpha=1,nfolds=10,grouped=TRUE)
pred.cm.lasso<-predictsym.glm(res.cm.lasso,response=9,int_prost_test,method='cm')
RMSE.L(sym.var(int_prost_test,9),pred.cm.lasso)
```

`RMSE.U` *Upper boundary root-mean-square error*

Description

Compute the upper boundary root-mean-square error.

Usage

```
RMSE.U(sym.var, prediction)
```

Arguments

`sym.var`	Variable that was predicted.
`prediction`	The prediction given by the model.

Value

The upper boundary root-mean-square error.

Author(s)

Oldemar Rodriguez Rojas

References

LIMA-NETO, E.A., DE CARVALHO, F.A.T., (2008). Centre and range method to fitting a linear regression model on symbolic interval data. Computational Statistics and Data Analysis 52, 1500-1515.

LIMA-NETO, E.A., DE CARVALHO, F.A.T., (2010). Constrained linear regression models for symbolic interval-valued variables. Computational Statistics and Data Analysis 54, 333-347.

See Also

sym.glm

Examples

```
data(int_prost_train)
data(int_prost_test)
res.cm<-sym.lm(lpsa~.,sym.data=int_prost_train,method='cm')
pred.cm<-predictsym.lm(res.cm,int_prost_test,method='cm')
RMSE.U(sym.var(int_prost_test,9),pred.cm$Fitted)

res.cm.lasso<-sym.glm(sym.data=int_prost_train,response=9,method='cm',
                      alpha=1,nfolds=10,grouped=TRUE)
pred.cm.lasso<-predictsym.glm(res.cm.lasso,response=9,int_prost_test,method='cm')
RMSE.U(sym.var(int_prost_test,9),pred.cm.lasso)
```

RSDA — *R to Symbolic Data Analysis*

Description

This work is framed inside the Symbolic Data Analysis (SDA). The objective of this work is to implement in R to the symbolic case certain techniques of the automatic classification, as well as some lineal models. These implementations will always be made following two fundamental principles in Symbolic Data Analysis like they are: Classic Data Analysis should always be a case particular case of the Symbolic Data Analysis and both, the exit as the input in an Symbolic Data Analysis should be symbolic. We implement for variables of type interval the mean, the median, the mean of the extreme values, the standard deviation, the deviation quartil, the dispersion boxes and the correlation also three new methods are also presented to carry out the lineal regression for variables of type interval. We also implement in this R package the method of Principal Components Analysis in two senses: First, we propose three ways to project the interval variables in the circle of correlations in such way that is reflected the variation or the inexactness of the variables. Second, we propose an algorithm to make the Principal Components Analysis for variables of type histogram. We implement a method for multidimensional scaling of interval data, denominated INTERSCAL.

Details

Package:	RSDA
Type:	Package
Version:	2.0.8
Date:	2018-10-09
License:	GPL (>=2)

Most of the function of the package stars from a symbolic data table that can be store in a CSV file withe follwing forma: In the first row the labels $C means that follows a continuous variable, $I means an interval variable, $H means a histogram variables and $S means set variable. In the first row each labels should be follow of a name to variable and to the case of histogram a set variables types the names of the modalities (categories) . In data rows for continuous variables we have just one value, for interval variables we have the minimum and the maximum of the interval, for histogram variables we have the number of modalities and then the probability of each modality and for set variables we have the cardinality of the set and next the elements of the set.

Author(s)

Oldemar Rodriguez Rojas
Maintainer: Oldemar Rodriguez Rojas <oldemar.rodriguez@ucr.ac.cr>

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Billard L., Douzal-Chouakria A. and Diday E. (2011) Symbolic Principal Components For Interval-Valued Observations, Statistical Analysis and Data Mining. 4 (2), 229-246. Wiley.

Bock H-H. and Diday E. (eds.) (2000). Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data. Springer, Germany.

Carvalho F., Souza R.,Chavent M., and Lechevallier Y. (2006) Adaptive Hausdorff distances and dynamic clustering of symbolic interval data. Pattern Recognition Letters Volume 27, Issue 3, February 2006, Pages 167-179

Cazes P., Chouakria A., Diday E. et Schektman Y. (1997). Extension de l'analyse en composantes principales a des donnees de type intervalle, Rev. Statistique Appliquee, Vol. XLV Num. 3 pag. 5-24, France.

Diday, E., Rodriguez O. and Winberg S. (2000). Generalization of the Principal Components Analysis to Histogram Data, 4th European Conference on Principles and Practice of Knowledge Discovery in Data Bases, September 12-16, 2000, Lyon, France.

Chouakria A. (1998) Extension des methodes d'analysis factorialle a des donnees de type intervalle, Ph.D. Thesis, Paris IX Dauphine University.

Makosso-Kallyth S. and Diday E. (2012). Adaptation of interval PCA to symbolic histogram variables, Advances in Data Analysis and Classification July, Volume 6, Issue 2, pp 147-159.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

Examples

```
data(example3)
sym.data<-example3
class(sym.data) <- c('sym.data.table')
display.sym.table(sym.data)
sym.scatterplot(sym.data[,1], sym.data[,4], col='blue',main='Main Title')

data(oils)
class(oils) <- c('sym.data.table')
res<-sym.interval.pca(oils,'centers')
sym.scatterplot(res$Sym.Components[,1],res$Sym.Components[,2],
                labels=TRUE,col='red',main='PCA Oils Data')
sym.scatterplot3d(res$Sym.Components[,1], res$Sym.Components[,2],
                  res$Sym.Components[,3],color='blue',main='PCA Oils Data')
sym.scatterplot.ggplot(res$Sym.Components[,1],
```

```
                     res$Sym.Components[,2], labels=TRUE)
sym.circle.plot(res$Sym.Prin.Correlations)

res<-sym.interval.pca(oils,'classic')
plot(res,choix='ind')
plot(res,choix='var')

data(lynne2)
res<-sym.interval.pca(lynne2,'centers')
class(res$Sym.Components) <- c('sym.data.table')
sym.scatterplot(res$Sym.Components[,1], res$Sym.Components[,2],
                labels=TRUE, col='red',main='PCA Lynne Data')
sym.scatterplot3d(res$Sym.Components[,1],res$Sym.Components[,2],
                  res$Sym.Components[,3],color='blue', main='PCA Lynne Data')
sym.scatterplot.ggplot(res$Sym.Components[,1],res$Sym.Components[,2],
                       labels=TRUE)
sym.circle.plot(res$Sym.Prin.Correlations)
```

---

sd *Generic function for the standard desviation*

Description

Compute the symbolic standard desviation.

Usage

```
sd(x, ...)

## Default S3 method:
sd(x, na.rm = FALSE, ...)

## S3 method for class 'sym.data.table'
sd(x, method = c("centers", "interval", "billard",
  "modal"), na.rm = FALSE, ...)
```

Arguments

`x`	A symbolic variable.
`...`	As in R sd function.
`na.rm`	As in R sd function.
`method`	The method to be use.

Value

return a real number.

Author(s)

Oldemar Rodriguez Rojas

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

Examples

```
data(example3)
sym.data<-example3
sd(sym.data[,1])
sd(sym.data[,2])
sd(sym.data[,6])
sd(sym.data[,6], method='interval')
sd(sym.data[,6], method='billard')
sd(sym.data[,3],method='modal')
```

`SDS.to.RSDA` *SDS SODAS files to RSDA files.*

Description

To convert SDS SODAS files to RSDA files.

Usage

```
SDS.to.RSDA(file.path, labels = FALSE)
```

Arguments

`file.path`	Disk path where the SODAS *.SDA file is.
`labels`	If we want to include SODAS SDA files lebels in RSDA file.

Value

A RSDA symbolic data file.

Author(s)

Olger Calderon and Roberto Zuniga.

References

Bock H-H. and Diday E. (eds.) (2000). Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data. Springer, Germany.

See Also

SODAS.to.RSDA

Examples

```
## Not run:
# We can read the file directly from the SODAS SDA file as follows:
# We can save the file in CSV to RSDA format as follows:
setwd('C:/Program Files (x86)/DECISIA/SODAS version 2.0/bases/')
result <- SDS.to.RSDA(file.path='hani3101.sds')
# We can save the file in CSV to RSDA format as follows:
write.sym.table(result, file='hani3101.csv', sep=';',dec='.', row.names=TRUE,

## End(Not run)
```

`SODAS.to.RSDA` *XML SODAS files to RSDA files.*

Description

To convert XML SODAS files to RSDA files.

Usage

```
SODAS.to.RSDA(XMLPath, labels = T)
```

Arguments

`XMLPath`	Disk path where the SODAS *.XML file is.
`labels`	If we want to include SODAS XML files lebels in RSDA file.

Value

A RSDA symbolic data file.

Author(s)

Olger Calderon and Roberto Zuniga.

References

Bock H-H. and Diday E. (eds.) (2000). Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data. Springer, Germany.

See Also

SDS.to.RSDA

Examples

```
## Not run:
# We can read the file directly from the SODAS XML file as follows:
# abalone<-SODAS.to.RSDA('C:/Program Files (x86)/DECISIA/SODAS version 2.0/bases/abalone.xml)
# We can save the file in CSV to RSDA format as follows:
# write.sym.table(sodas.ex1, file='abalone.csv', sep=';',dec='.', row.names=TRUE,
#                col.names=TRUE)
# We read the file from the CSV file,
# this is not necessary if the file is read directly from
# XML using SODAS.to.RSDA as in the first statement in this example.
data(abalone)
res<-sym.interval.pca(abalone,'centers')
sym.scatterplot(sym.var(res$Sym.Components,1),sym.var(res$Sym.Components,2),
               labels=TRUE,col='red',main='PCA Oils Data')
sym.scatterplot3d(sym.var(res$Sym.Components,1),sym.var(res$Sym.Components,2),
                 sym.var(res$Sym.Components,3),color='blue',main='PCA Oils Data')
sym.scatterplot.ggplot(sym.var(res$Sym.Components,1),sym.var(res$Sym.Components,2),
                      labels=TRUE)
sym.circle.plot(res$Sym.Prin.Correlations)


## End(Not run)
```

`StudentsGrades` *Data Example.*

Description

Symbolic data matrix with all the variables continuos type.

Usage

```
data(StudentsGrades)
```

Format

$C Math $C Science $C Spanish $C History $C Sport
Lucia $C 7.0 $C 6.5 $C 9.2 $C 8.6 $C 8.0
Pedro $C 7.5 $C 9.4 $C 7.3 $C 7.0 $C 7.0
Ines $C 7.6 $C 9.2 $C 8.0 $C 8.0 $C 7.5
Luis $C 5.0 $C 6.5 $C 6.5 $C 7.0 $C 9.0
Andres $C 6.0 $C 6.0 $C 7.8 $C 8.9 $C 7.3
Ana $C 7.8 $C 9.6 $C 7.7 $C 8.0 $C 6.5
Carlos $C 6.3 $C 6.4 $C 8.2 $C 9.0 $C 7.2
Jose $C 7.9 $C 9.7 $C 7.5 $C 8.0 $C 6.0

Sonia $C 6.0 $C 6.0 $C 6.5 $C 5.5 $C 8.7
Maria $C 6.8 $C 7.2 $C 8.7 $C 9.0 $C 7.0

Examples

```
data(StudentsGrades)
StudentsGrades
```

---

sym.circle.plot *Symbolic Circle of Correlations*

Description

Plot the symbolic circle of correlations.

Usage

```
sym.circle.plot(prin.corre)
```

Arguments

prin.corre	A symbolic interval data matrix with correlations between the variables and the principals componets, both of interval type.

Value

Plot the symbolic circle

Author(s)

Oldemar Rodriguez Rojas

References

Rodriguez O. (2012). The Duality Problem in Interval Principal Components Analysis. The 3rd Workshop in Symbolic Data Analysis, Madrid.

Examples

```
data(oils)
res<-sym.interval.pca(oils,'centers')
sym.circle.plot(res$Sym.Prin.Correlations)
```

sym.cor *Symbolic Correlation*

Description

This function compute the symbolic correlation

Usage

```
sym.cor(sym.var.x, sym.var.y, method = c('centers', 'interval', 'billard', 'modal')
, na.rm = FALSE, ...)
```

Arguments

sym.var.x	First symbolic variables.
sym.var.y	Second symbolic variables.
method	The method to be use.
na.rm	As in R cor function.
...	As in R cor function.

Value

Return a real number in [-1,1].

Author(s)

Oldemar Rodriguez Rojas

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

Examples

```
data(example3)
sym.data<-example3
sym.cor(sym.var(sym.data,1),sym.var(sym.data,4),method='centers')
sym.cor(sym.var(sym.data,2),sym.var(sym.data,6),method='centers')
sym.cor(sym.var(sym.data,2),sym.var(sym.data,6),method='billard')
```

sym.cov *Symbolic Covariance*

Description

This function compute the symbolic covariance.

Usage

```
sym.cov(sym.var.x, sym.var.y, method = c('centers', 'interval', 'billard', 'modal'),
na.rm = FALSE, ...)
```

Arguments

sym.var.x	First symbolic variables.
sym.var.y	Second symbolic variables.
method	The method to be use.
na.rm	As in R cov function.
...	As in R cov function.

Value

Return a real number.

Author(s)

Oldemar Rodriguez Rojas

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

Examples

```
data(example3)
sym.data<-example3
sym.cov(sym.var(sym.data,1),sym.var(sym.data,4),method='centers')
sym.cov(sym.var(sym.data,2),sym.var(sym.data,6),method='centers')
sym.cov(sym.var(sym.data,2),sym.var(sym.data,6),method='billard')
```

sym.dist.interval *Distance for Symbolic Interval Variables.*

Description

This function computes and returns the distance matrix by using the specified distance measure to compute distance between symbolic interval variables.

Usage

```
sym.dist.interval(sym.data, gamma = 0.5, method = "Minkowski",
  normalize = TRUE, SpanNormalize = FALSE, q = 1, euclidea = TRUE,
  pond = rep(1, length(variables)))
```

Arguments

sym.data	A symbolic object
gamma	gamma value for the methods ichino and minkowski.
method	Method to use (Gowda.Diday, Ichino, Minkowski, Hausdorff)
normalize	A logical value indicating whether normalize the data in the ichino or hausdorff method.
SpanNormalize	A logical value indicating whether
q	q value for the hausdorff method.
euclidea	A logical value indicating whether use the euclidean distance.
pond	A numeric vector
variables	Numeric vector with the number of the variables to use.

Value

An object of class 'dist'

Examples

```
data('table7')
ex3 <- classic.to.sym(table7, concept=c('Animal'),variables = c(Height, Weight)
,variables.types=c(Height = type.interval(), Weight = type.interval()))
sym.dist.interval(ex3,method='Gowda.Diday',normalize=FALSE)
sym.dist.interval(ex3,gamma=0.5,method='Ichino',normalize=FALSE)
sym.dist.interval(ex3,gamma=0.5,method='Minkowski',normalize=FALSE,q=1)
sym.dist.interval(ex3,gamma=0.5,method='Minkowski',normalize=FALSE,q=2)
sym.dist.interval(ex3,gamma=0.5,method='Hausdorff',normalize=FALSE,
SpanNormalize=FALSE,euclidea=TRUE)
sym.dist.interval(ex3,gamma=0.5,method='Hausdorff',normalize=FALSE,
SpanNormalize=TRUE,euclidea=TRUE)
```

sym.dist.set *Distance for Symbolic Set Variables.*

Description

This function computes and returns the distance matrix by using the specified distance measure to compute distance between symbolic interval variables.

Usage

```
sym.dist.set(sym.data, gamma = 0.5, method = "Minkowski",
  normalize = TRUE, q = 1, pond = rep(1, length(variables)))
```

Arguments

sym.data	A symbolic object
gamma	gamma value for the methods ichino and minkowski.
method	Method to use (Gowda.Diday, Ichino, Minkowski, Hausdorff)
normalize	A logical value indicating whether normalize the data in the ichino or hausdorff method.
q	q value for the Minkowski method.
pond	A numeric vector
variables	Numeric vector with the number of the variables to use.

Value

An object of class 'dist'

sym.glm *Lasso, Ridge and and Elastic Net Linear regression model to interval variables*

Description

Execute Lasso, Ridge and and Elastic Net Linear regression model to interval variables.

Usage

```
sym.glm(sym.data, response = 1, method = c('cm', 'crm'),
alpha = 1, nfolds = 10, grouped = TRUE)
```

Arguments

- `sym.data` — Should be a symbolic data table read with the function read.sym.table(...).
- `response` — The number of the column where is the response variable in the interval data table.
- `method` — 'cm' to generalized Center Method and 'crm' to generalized Center and Range Method.
- `alpha` — alpha=1 is the lasso penalty, and alpha=0 the ridge penalty. 0<alpha<1 is the elastic net method.
- `nfolds` — Number of folds - default is 10. Although nfolds can be as large as the sample size (leave-one-out CV), it is not recommended for large datasets. Smallest value allowable is nfolds=3
- `grouped` — This is an experimental argument, with default TRUE, and can be ignored by most users.

Value

An object of class 'cv.glmnet' is returned, which is a list with the ingredients of the cross-validation fit.

Author(s)

Oldemar Rodriguez Rojas

References

Rodriguez O. (2013). A generalization of Centre and Range method for fitting a linear regression model to symbolic interval data using Ridge Regression, Lasso and Elastic Net methods. The IFCS2013 conference of the International Federation of Classification Societies, Tilburg University Holland.

See Also

sym.lm

Examples

```
data(int_prost_train)
data(int_prost_test)
res.cm.lasso<-sym.glm(sym.data=int_prost_train,response=9,method='cm',
                      alpha=1,nfolds=10,grouped=TRUE)
pred.cm.lasso<-predictsym.glm(res.cm.lasso,response=9,int_prost_test,method='cm')
plot(res.cm.lasso)
plot(res.cm.lasso$glmnet.fit, 'norm', label=TRUE)
plot(res.cm.lasso$glmnet.fit, 'lambda', label=TRUE)
RMSE.L(sym.var(int_prost_test,9),pred.cm.lasso)
RMSE.U(sym.var(int_prost_test,9),pred.cm.lasso)
R2.L(sym.var(int_prost_test,9),pred.cm.lasso)
R2.U(sym.var(int_prost_test,9),pred.cm.lasso)
deter.coefficient(sym.var(int_prost_test,9),pred.cm.lasso)
```

sym.hclust *Symbolic Hierarchical Clustering*

Description

This function allows us to execute a symbolic hierarchical clustering to interval variables.

Usage

```
sym.hclust(sym.data, distance = c('hausdorff', 'centers'), p = 2,
method = c('ward.D2', 'single', 'complete', 'average', 'mcquitty',
           'median', 'centroid'), members = NULL)
```

Arguments

sym.data	The symbolic data table.
distance	The distance to be use.
p	The p in the Hausdorff distance : $$d(w_{u_1}, w_{u_2}) = \left(\sum_{j=1}^{m} \Phi_j(w_{u_1}, w_{u_2})^p\right)^{1/p}$$
method	The method to be use, like in hclust R function.
members	Like in hclust R function.

Value

Return a dendogram plot structure.

Author(s)

Oldemar Rodriguez Rojas

References

Carvalho F., Souza R.,Chavent M., and Lechevallier Y. (2006) Adaptive Hausdorff distances and dynamic clustering of symbolic interval data. Pattern Recognition Letters Volume 27, Issue 3, February 2006, Pages 167-179

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

Examples

```
## Not run:
data(oils)
sh<-sym.hclust(oils)
plot(sh)
sh<-sym.hclust(oils,'centers')
plot(sh)

## End(Not run)
```

---

sym.histogram.pca *Histogram Principal Components Analysis*

Description

This functions allows us to execute a histogram principal components analysis from a symbolic data table with continuos, interval or histogram variables that can be mixed.

Usage

```
sym.histogram.pca(sym.data, method = c('histogram', 'classic'))
```

Arguments

sym.data	Symbolic data table.
method	The method to be used.

Value

Return a symbolic data table.

Author(s)

Oldemar Rodriguez Rojas

References

Diday, E., Rodriguez O. and Winberg S. (2000). Generalization of the Principal Components Analysis to Histogram Data, 4th European Conference on Principles and Practice of Knowledge Discovery in Data Bases, September 12-16, 2000, Lyon, France.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

See Also

sym.interval.pca

Examples

```
data(example7)
res<-sym.histogram.pca(example7)
class(res) <- c('sym.data.table')
sym.scatterplot(res[,1],res[,2], labels=TRUE,col='red',main='Histogram PCA')
sym.scatterplot3d(res[,1],res[,2],res[,3],color='blue',
                  main='Histogram PCA')
```

---

sym.interval.pc — *Compute a symbolic interval principal components curves*

Description

Compute a symbolic interval principal components curves

Usage

```
sym.interval.pc(sym.data, method = c('vertex', 'centers'), maxit, plot, scale, center)
```

Arguments

`sym.data`	Shoud be a symbolic data table read with the function read.sym.table(...)
`method`	It should be 'vertex' or 'centers'.
`maxit`	Maximum number of iterations.
`plot`	TRUE to plot immediately, FALSE if you do not want to plot.
`scale`	TRUE to standardize the data.
`center`	TRUE to center the data.

Value

prin.curve: This a symbolic data table with the interval principal components. As this is a symbolic data table we can apply over this table any other symbolic data analysis method (symbolic propagation).

cor.ps: This is the interval correlations between the original interval variables and the interval principal components, it can be use to plot the symbolic circle of correlations.

Author(s)

Jorge Arce.

References

Arce J. and Rodriguez O. (2015) 'Principal Curves and Surfaces to Interval Valued Variables'. The 5th Workshop on Symbolic Data Analysis, SDA2015, Orleans, France, November.

Hastie,T. (1984). Principal Curves and Surface. Ph.D Thesis Stanford University.

Hastie,T. \& Weingessel,A. (2014). princurve - Fits a Principal Curve in Arbitrary Dimension.R package version 1.1–12 http://cran.r-project.org/web/packages/princurve/index.html.

Hastie,T. \& Stuetzle, W. (1989). Principal Curves. Journal of the American Statistical Association, Vol. 84-406, 502–516.

Hastie, T., Tibshirani, R. \& Friedman, J. (2008). The Elements of Statistical Learning; Data Mining, Inference and Prediction. Springer, New York.

See Also

sym.interval.pca

Examples

```
## Not run:
data(oils)
res.vertex.ps<-sym.interval.pc(oils,'vertex',150,FALSE,FALSE,TRUE)
class(res.vertex.ps$sym.prin.curve) <- c('sym.data.table')
sym.scatterplot(res.vertex.ps$sym.prin.curve[,1],res.vertex.ps$sym.prin.curve[,2],
                labels=TRUE,col='red',main='PSC Oils Data')

data(facedata)
res.vertex.ps<-sym.interval.pc(facedata,'vertex',150,FALSE,FALSE,TRUE)
class(res.vertex.ps$sym.prin.curve) <- c('sym.data.table')
sym.scatterplot(res.vertex.ps$sym.prin.curve[,1],res.vertex.ps$sym.prin.curve[,2],
                labels=TRUE,col='red',main='PSC Face Data')

## End(Not run)
```

---

sym.interval.pc.limits

Symbolic interval principal curves limits

Description

Symbolic interval principal curves limits.

Usage

```
sym.interval.pc.limits(sym.data, prin.curve, num.vertex, lambda, var.ord)
```

Arguments

`sym.data`	Symbolic interval data table.
`prin.curve`	Principal curves.
`num.vertex`	Number of vertices of the hipercube.
`lambda`	Lambda.
`var.ord`	Order of the variables.

Author(s)

Jorge Arce.

References

Arce J. and Rodriguez O. (2015) 'Principal Curves and Surfaces to Interval Valued Variables'. The 5th Workshop on Symbolic Data Analysis, SDA2015, Orleans, France, November.

Hastie,T. (1984). Principal Curves and Surface. Ph.D Thesis Stanford University.

Hastie,T. \& Weingessel,A. (2014). princurve - Fits a Principal Curve in Arbitrary Dimension.R package version 1.1–12 http://cran.r-project.org/web/packages/princurve/index.html.

Hastie,T. \& Stuetzle, W. (1989). Principal Curves. Journal of the American Statistical Association, Vol. 84-406, 502–516.

Hastie, T., Tibshirani, R. \& Friedman, J. (2008). The Elements of Statistical Learning; Data Mining, Inference and Prediction. Springer, New York.

See Also

sym.interval.pc

`sym.interval.pca` *Interval Principal Components Analysis.*

Description

Cazes, Chouakria, Diday and Schektman (1997) proposed the Centers and the Tops Methods to extend the well known principal components analysis method to a particular kind of symbolic objects characterized by multi–values variables of interval type.

Usage

```
sym.interval.pca(sym.data, method = c('classic', 'tops','centers',
'principal.curves', 'optimized.distance', 'optimized.variance'))
```

Arguments

`sym.data`	Shoud be a symbolic data table
`method`	It is use so select the method, 'classic' execute a classical principal component analysis over the centers of the intervals, 'tops' to use the vertices algorithm and 'centers' to use the centers algorithm.

Value

Sym.Components: This a symbolic data table with the interval principal components. As this is a symbolic data table we can apply over this table any other symbolic data analysis method (symbolic propagation).

Sym.Prin.Correlations: This is the interval correlations between the original interval variables and the interval principal components, it can be use to plot the symbolic circle of correlations.

Author(s)

Oldemar Rodriguez Rojas

References

Bock H-H. and Diday E. (eds.) (2000). Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data. Springer, Germany.

Cazes P., Chouakria A., Diday E. et Schektman Y. (1997). Extension de l'analyse en composantes principales a des donnees de type intervalle, Rev. Statistique Appliquee, Vol. XLV Num. 3 pag. 5-24, France.

Chouakria A. (1998) Extension des methodes d'analysis factorialle a des donnees de type intervalle, Ph.D. Thesis, Paris IX Dauphine University.

Makosso-Kallyth S. and Diday E. (2012). Adaptation of interval PCA to symbolic histogram variables, Advances in Data Analysis and Classification July, Volume 6, Issue 2, pp 147-159.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

See Also

sym.histogram.pca

Examples

```
## Not run:
data(oils)
res<-sym.interval.pca(oils,'centers')
class(res) <- c('sym.data.table')
sym.scatterplot(res$Sym.Components[,1],res$Sym.Components[,1],
                labels=TRUE,col='red',main='PCA Oils Data')
sym.scatterplot3d(res$Sym.Components[,1],res$Sym.Components[,2],
                  res$Sym.Components[,3],color='blue',main='PCA Oils Data')
sym.scatterplot.ggplot(res$Sym.Components[,1],res$Sym.Components[,2],
                       labels=TRUE)
```

```
sym.circle.plot(res$Sym.Prin.Correlations)

res<-sym.interval.pca(oils,'classic')
plot(res,choix='ind')
plot(res,choix='var')

data(lynne2)
res<-sym.interval.pca(lynne2,'centers')
class(res$Sym.Components) <- c('sym.data.table')
sym.scatterplot(res$Sym.Components[,1],res$Sym.Components[,2],
                labels=TRUE,col='red',main='PCA Lynne Data')
sym.scatterplot3d(res$Sym.Components[,1],res$Sym.Components[,2],
                  res$Sym.Components[,3],color='blue', main='PCA Lynne Data')
sym.scatterplot.ggplot(res$Sym.Components[,1],res$Sym.Components[,2],
                       labels=TRUE)
sym.circle.plot(res$Sym.Prin.Correlations)

data(StudentsGrades)
st<-StudentsGrades
s.pca<-sym.interval.pca(st)
plot(s.pca,choix='ind')
plot(s.pca,choix='var')

## End(Not run)
```

`sym.kmeans` *Symbolic k-Means*

Description

This is a function is to carry out a k-means overs a interval symbolic data matrix.

Usage

```
sym.kmeans(sym.data, k = 3, iter.max = 10, nstart = 1,
algorithm = c('Hartigan-Wong', 'Lloyd', 'Forgy', 'MacQueen'))
```

Arguments

`sym.data`	Symbolic data table.
`k`	The number of clusters.
`iter.max`	Maximun number of iterations.
`nstart`	As in R kmeans function.
`algorithm`	The method to be use, as in kmeans R function.

Value

This function return the following information:

K-means clustering with 3 clusters of sizes 2, 2, 4

Cluster means:

GRA FRE IOD SAP

1 0.93300 -13.500 193.500 174.75

2 0.86300 30.500 54.500 195.25

3 0.91825 -6.375 95.375 191.50

Clustering vector:

L P Co S Ca O B H

1 1 3 3 3 3 2 2

Within cluster sum of squares by cluster:

[1] 876.625 246.125 941.875

(between_SS / total_SS = 92.0
Available components:

[1] 'cluster' 'centers' 'totss' 'withinss' 'tot.withinss' 'betweenss'

[7] 'size'

Author(s)

Oldemar Rodriguez Rojas

References

Carvalho F., Souza R.,Chavent M., and Lechevallier Y. (2006) Adaptive Hausdorff distances and dynamic clustering of symbolic interval data. Pattern Recognition Letters Volume 27, Issue 3, February 2006, Pages 167-179

See Also

sym.hclust

Examples

```
data(oils)
sk<-sym.kmeans(oils,k=3)
sk$cluster
```

---

sym.lm *CM and CRM Linear regression model.*

Description

To execute the Center Method (CR) and Center and Range Method (CRM) to Linear regression.

Usage

```
sym.lm(formula, sym.data, method = c('cm', 'crm'))
```

Arguments

formula	An object of class 'formula' (or one that can be coerced to that class): a symbolic description of the model to be fitted.
sym.data	Should be a symbolic data table read with the function read.sym.table(...).
method	'cm' to Center Method and 'crm' to Center and Range Method.

Details

Models for lm are specified symbolically. A typical model has the form response ~ terms where response is the (numeric) response vector and terms is a series of terms which specifies a linear predictor for response. A terms specification of the form first + second indicates all the terms in first together with all the terms in second with duplicates removed. A specification of the form first:second indicates the set of terms obtained by taking the interactions of all terms in first with all terms in second. The specification first*second indicates the cross of first and second. This is the same as first + second + first:second.

Value

sym.lm returns an object of class 'lm' or for multiple responses of class c('mlm', 'lm')

Author(s)

Oldemar Rodriguez Rojas

References

LIMA-NETO, E.A., DE CARVALHO, F.A.T., (2008). Centre and range method to fitting a linear regression model on symbolic interval data. Computational Statistics and Data Analysis 52, 1500-1515.

LIMA-NETO, E.A., DE CARVALHO, F.A.T., (2010). Constrained linear regression models for symbolic interval-valued variables. Computational Statistics and Data Analysis 54, 333-347.

Examples

```
data(int_prost_train)
data(int_prost_test)
res.cm<-sym.lm(lpsa~.,sym.data=int_prost_train,method='cm')
pred.cm<-predictsym.lm(res.cm,int_prost_test,method='cm')
RMSE.L(sym.var(int_prost_test,9),pred.cm$Fitted)
RMSE.U(sym.var(int_prost_test,9),pred.cm$Fitted)
R2.L(sym.var(int_prost_test,9),pred.cm$Fitted)
R2.U(sym.var(int_prost_test,9),pred.cm$Fitted)
deter.coefficient(sym.var(int_prost_test,9),pred.cm$Fitted)
```

`sym.lm.bi` *Symbolic Linear Regression for two variables.*

Description

The function build a symbolic regression for two interval or continuos variables.

Usage

```
sym.lm.bi(sym.var.x, sym.var.y, method = c('mid-points', 'tops', 'inf-sup',
'billard'))
```

Arguments

`sym.var.x`	The firth symbolic variable.
`sym.var.y`	The second symbolic variable.
`method`	The thirth symbolic variable.

Value

This function return a regression structure as folllows:

$Intercept
[1] 38.64236

$Beta1
[1] 0.3081313

Author(s)

Oldemar Rodriguez Rojas

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

Examples

```
data(example3)
sym.data<-example3
class(sym.data) <- c('sym.data.table')
lm.mod<-sym.lm.bi(sym.var(sym.data,1),sym.var(sym.data,4))
sym.scatterplot(sym.data[,1],sym.data[,4],col='blue',
                main='Linear Regression')
abline(lm.mod,lwd=3)

lm.mod<-sym.lm.bi(sym.var(sym.data,2),sym.var(sym.data,6))
sym.scatterplot(sym.data[,2],sym.data[,6],
                col='blue',main='Linear Regression')
abline(lm.mod,lwd=3)

data(lynne1)
sym.data<-lynne1
class(sym.data) <- c('sym.data.table')
lm.mod<-sym.lm.bi(sym.var(lynne1,2),sym.var(lynne1,1))
sym.scatterplot(sym.data[,2],sym.data[,1],labels=TRUE,
                col='red',main='Linear Regression')
abline(lm.mod,lwd=3,col='blue')

lm.mod<-sym.lm.bi(sym.var(lynne1,2),sym.var(lynne1,1),method='inf-sup')
sym.scatterplot(sym.data[,2],sym.data[,1],labels=TRUE,
                col='red',main='Linear Regression')
abline(lm.mod$inf,lwd=3,col='blue')
abline(lm.mod$sup,lwd=3,col='blue')

lm.mod<-sym.lm.bi(sym.var(lynne1,2),sym.var(lynne1,1),method='tops')
sym.scatterplot(sym.data[,2],sym.data[,1],labels=TRUE,
                col='red',main='Linear Regression')
abline(lm.mod,lwd=3,col='blue')

lm.mod<-sym.lm.bi(sym.var(lynne1,2),sym.var(lynne1,1),method='billard')
sym.scatterplot(sym.data[,2],sym.data[,1],labels=TRUE,
                col='red',main='Linear Regression')
abline(lm.mod$Intercept,lm.mod$Beta1,lwd=3,col='blue')
```

sym.mcfa *sym.mcfa*

Description

This function executes a Multiple Correspondence Factor Analysis for variables of set type.

Usage

```
sym.mcfa(sym.data, pos.var)
```

Arguments

sym.data	A symbolic data table containing at least two set type variables.
pos.var	Column numbers in the symbolic data table that contain the set type variables.

Author(s)

Jorge Arce

References

Arce J. and Rodriguez, O. (2018). Multiple Correspondence Analysis for Symbolic Multi–Valued Variables. On the Symbolic Data Analysis Workshop SDA 2018.

Benzecri, J.P. (1973). L' Analyse des Données. Tomo 2: L'Analyse des Correspondances. Dunod, Paris.

Castillo, W. and Rodriguez O. (1997). Algoritmo e implementacion del analisis factorial de correspondencias. Revista de Matematicas: Teoria y Aplicaciones, 24-31.

Takagi I. and Yadosiha H. (2011). Correspondence Analysis for symbolic contingency tables base on interval algebra. Elsevier Procedia Computer Science, 6, 352-357.

Rodriguez, O. (2007). Correspondence Analysis for Symbolic Multi–Valued Variables. CARME 2007 (Rotterdam, The Netherlands), http://www.carme-n.org/carme2007.

Examples

```
data("ex_mcfa1")
sym.table <- classic.to.sym(ex_mcfa1, concept = "suspect",
                   variables.types = c(hair = type.set(),
                                       eyes = type.set(),
                                       region = type.set()))

res <- sym.mcfa(sym.table, c(1,2))
mcfa.scatterplot(res[,1], res[,2], sym.data = sym.table, pos.var = c(1,2))
```

`sym.mds` *Symbolic Multidemensional Scaling*

Description

This function execute a multidimensional scaling from a interval symbolic data matrix.

Usage

```
sym.mds(sym.data, distance = c('hausdorff', 'centers'), p = 2,
method = c('classic', 'INTERSCAL'))
```

Arguments

`sym.data`	The symbolic data matrix.
`distance`	The distance to be use.
`p`	The p in the Hausdorff distance $$d(w_{u_1}, w_{u_2}) = \left(\sum_{j=1}^{m} \Phi_j(w_{u_1}, w_{u_2})^p \right)^{1/p}$$
`method`	The method to be used.

Value

Return the coordanates to plot the graphic.

Author(s)

Oldemar Rodriguez Rojas

References

Groenen, P.J.F., Winsberg, S., Rodriguez, O., Diday, E. (2006). I-Scal: Multidimensional scaling of interval dissimilarities. Computational Statistics and Data Analysis, 51, 360-378.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

See Also

sym.interval.pca

Examples

```
## Not run:
data(oils)
res<-sym.mds(oils)
plot(res,pch = 23, bg = 'red', xlab = 'Score 1', ylab = 'Score 2')
res<-sym.mds(oils,distance='centers')
plot(res,pch = 23, bg = 'red', xlab = 'Score 1', ylab = 'Score 2')

## End(Not run)
```

`sym.mean` *Symbolic Mean*

Description

This function compute the symbolic mean

Usage

```
sym.mean(sym.var, method = c('centers', 'interval', 'modal'),
trim = 0, na.rm = FALSE, ...)
```

Arguments

`sym.var`	The symbolic variable.
`method`	The method to be use.
`trim`	As in R mean function.
`na.rm`	As in R mean function.
`...`	As in R mean function.

Value

Return a real number.

Author(s)

Oldemar Rodriguez Rojas

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

Examples

```
data(example3)
sym.data<-example3
sym.mean(sym.var(sym.data,1))
sym.mean(sym.var(sym.data,2))
sym.mean(sym.var(sym.data,2),method='interval')
sym.mean(sym.var(sym.data,3),method='modal')
```

`sym.median` *Symbolic Median*

Description

This function compute the symbolic median.

Usage

```
sym.median(sym.var, method = c('centers', 'interval', 'modal'), na.rm = FALSE, ...)
```

Arguments

`sym.var`	The symbolic variable.
`method`	The method to be use.
`na.rm`	As in R median function.
`...`	As in R median function.

Value

Return a real number.

Author(s)

Oldemar Rodriguez Rojas

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

Examples

```
data(example3)
sym.data<-example3
sym.median(sym.var(sym.data,1))
sym.median(sym.var(sym.data,2))
sym.median(sym.var(sym.data,6),method='interval')
sym.median(sym.var(sym.data,3),method='modal')
```

---

sym.obj — *Symbolic Object*

Description

This function get a symbolic object (row or a case) from a symbolic data table.

Usage

```
sym.obj(sym.data, number.sym.obj)
```

Arguments

`sym.data`	Symboli data matrix.
`number.sym.obj`	The number of the row for the symbolic object (case) that we want to get.

Value

Return a symbolic object with the following internal format:

$M

[1] 5

$var.types

[1] '$C' '$H' '$I' '$H' '$C'

$var.length

[1] 1 5 2 3 1

$var.names

[1] 'F1' 'F2' 'F3' 'F4' 'F5'

$obj.data.vector

F1 M1 M2 M3 M4 M5 F3 F3.1 M1.1 M2.1 M3.1 F5

Case4 -2.1 0.4 0.1 0.1 0.1 0.3 0 2 0.9 0 0.1 0

Author(s)

Oldemar Rodriguez Rojas

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Bock H-H. and Diday E. (eds.) (2000). Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data. Springer, Germany.

Examples

```
data(example7)
sym.obj(example7,4)
```

`sym.scatterplot.ggplot` *Symbolic Scatter GGPlot*

Description

This function could be use to plot two symbolic variables in a X-Y plane using ggplot R package.

Usage

```
sym.scatterplot.ggplot(sym.var.x, sym.var.y, labels = FALSE, ...)
```

Arguments

`sym.var.x`	First symbolic variable.
`sym.var.y`	Second symbolic variable.
`labels`	As in ggplot.
`...`	As in ggplot.

Value

return a ggplot graphic.

Author(s)

Oldemar Rodriguez Rojas

References

Bock H-H. and Diday E. (eds.) (2000). Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data. Springer, Germany.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

See Also

sym.scatterplot

Examples

```
data(lynne1)
class(lynne1) <- c('sym.data.table')
sym.scatterplot.ggplot(lynne1[,1], lynne1[,3],labels=TRUE)
data(oils)
sym.scatterplot.ggplot(oils[,2], oils[,3],labels=TRUE)
```

---

sym.scatterplot3d *Symbolic Scatter Plot 3D*

Description

This function could be use to plot two symbolic variables in 3D i.e. in a X-Y-Z plane.

Usage

```
sym.scatterplot3d(sym.var.x, sym.var.y, sym.var.z, labels = FALSE, ...)
```

Arguments

sym.var.x	First symbolic variable.
sym.var.y	Second symbolic variable.
sym.var.z	Third symbolic variable.
labels	As in R plot function.
...	As in R plot function.

Value

3D Plot graphic.

Author(s)

Oldemar Rodriguez Rojas

References

Bock H-H. and Diday E. (eds.) (2000). Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data. Springer, Germany.

Examples

```
data(lynne1)
class(lynne1) <- c('sym.data.table')
sym.scatterplot3d(lynne1[,1], lynne1[,2], lynne1[,3],
                  color='blue', main='Lynne Data')
```

---

sym.sd *Symbolic Standard Desviation*

Description

Compute the symbolic standard desviation.

Usage

```
sym.sd(sym.var, method =
c('centers', 'interval', 'billard', 'modal'), na.rm = FALSE, ...)
```

Arguments

sym.var	The symbolic variable.
method	The method to be use.
na.rm	As in R sd function.
...	As in R sd function.

Value

return a real number.

Author(s)

Oldemar Rodriguez Rojas

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

Examples

```
data(example3)
sym.data<-example3
sym.sd(sym.var(sym.data,1))
sym.sd(sym.var(sym.data,2))
sym.sd(sym.var(sym.data,6))
sym.sd(sym.var(sym.data,6),method='interval')
sym.sd(sym.var(sym.data,6),method='billard')
sym.sd(sym.var(sym.data,3),method='modal')
```

`sym.var` *Symbolic Variable (Feature)*

Description

This function get a symbolic variable from a symbolic data table.

Usage

```
sym.var(sym.data, number.sym.var)
```

Arguments

`sym.data`	The symbolic data table
`number.sym.var`	The number of the column for the variable (feature) that we want to get.

Value

Return a symbolic data variable with the following structure:

$N

[1] 7

$var.name

[1] 'F6'

$var.type

[1] '$I'

$obj.names

[1] 'Case1' 'Case2' 'Case3' 'Case4' 'Case5' 'Case6' 'Case7'

$var.data.vector

F6 F6.1

Case1 0.00 90.00

Case2 -90.00 98.00

Case3 65.00 90.00

Case4 45.00 89.00

Case5 20.00 40.00

Case6 5.00 8.00

Case7 3.14 6.76

Author(s)

Oldemar Rodriguez Rojas

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Bock H-H. and Diday E. (eds.) (2000). Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data. Springer, Germany.

See Also

sym.obj

Examples

```
data(example3)
sym.data<-example3
sym.var(sym.data,4)
sym.var(sym.data,6)
```

sym.variance *Symbolic Variance*

Description

Compute the symbolic variance.

Usage

```
sym.variance(sym.var, method = c('centers', 'interval', 'billard', 'modal'),
 na.rm = FALSE, ...)
```

Arguments

sym.var	The symbolic variable.
method	The method to be use.
na.rm	As in R var function.
...	As in R var function.

Value

Return a real number.

Author(s)

Oldemar Rodriguez Rojas

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

Examples

```
data(example3)
sym.data<-example3
sym.variance(sym.var(sym.data,1))
sym.variance(sym.var(sym.data,2))
sym.variance(sym.var(sym.data,6))
sym.variance(sym.var(sym.data,6),method='interval')
sym.variance(sym.var(sym.data,6),method='billard')
sym.variance(sym.var(sym.data,3),method='modal')
```

Table7 *Table7*

Description

example for the dist.interval function.

Usage

```
table7
```

Format

An object of class `data.frame` with 6 rows and 3 columns.

USCrime *Us crime classic data table*

Description

Us crime classic data table that can be used to generate symbolic data tables.

Usage

```
data(USCrime)
```

Format

An object of class `data.frame` with 1994 rows and 103 columns.

Source

http://archive.ics.uci.edu/ml/

References

HASTIE, T., TIBSHIRANI, R. and FRIEDMAN, J. (2008). The Elements of Statistical Learning: Data Mining, Inference and Prediction. New York: Springer.

Examples

```
data(USCrime)
us.crime<-USCrime
dim(us.crime)
head(us.crime)
summary(us.crime)
names(us.crime)
nrow(us.crime)
result  <- classic.to.sym(us.crime, concept='state',
                          variables =c(NumInShelters, NumImmig),
                          variables.types =c(NumInShelters = type.histogram(),
                           NumImmig = type.histogram()))
result
```

---

uscrime_int *Us crime interval data table.*

Description

Us crime classic data table genetated from uscrime data.

Usage

```
data(uscrime_int)
```

Format

An object of class `list` of length 9.

References

Rodriguez O. (2013). A generalization of Centre and Range method for fitting a linear regression model to symbolic interval data using Ridge Regression, Lasso and Elastic Net methods. The IFCS2013 conference of the International Federation of Classification Societies, Tilburg University Holland.

Examples

```
data(uscrime_int)
car.data<-uscrime_int
res.cm.lasso<-sym.glm(sym.data=car.data,response=102,method='cm',alpha=1,
                      nfolds=10,grouped=TRUE)
plot(res.cm.lasso)
plot(res.cm.lasso$glmnet.fit, 'norm', label=TRUE)
plot(res.cm.lasso$glmnet.fit, 'lambda', label=TRUE)

pred.cm.lasso<-predictsym.glm(res.cm.lasso,response=102,car.data,method='cm')
RMSE.L(sym.var(car.data,102),pred.cm.lasso)
RMSE.U(sym.var(car.data,102),pred.cm.lasso)
```

```
R2.L(sym.var(car.data,102),pred.cm.lasso)
R2.U(sym.var(car.data,102),pred.cm.lasso)
deter.coefficient(sym.var(car.data,102),pred.cm.lasso)
```

---

var *Generic function for the Variance*

Description

Compute the symbolic variance.

Usage

```
var(x, ...)

## Default S3 method:
var(x, y = NULL, na.rm = FALSE, use, ...)

## S3 method for class 'sym.data.table'
var(x, method = c("centers", "interval", "billard",
  "modal"), na.rm = FALSE, ...)
```

Arguments

x	The symbolic variable.
...	As in R median function.
y	NULL (default) or a vector, matrix or data frame with compatible dimensions to x. The default is equivalent to y = x (but more efficient).
na.rm	logical. Should missing values be removed?
use	an optional character string giving a method for computing covariances in the presence of missing values. This must be (an abbreviation of) one of the strings 'everything', 'all.obs', 'complete.obs', 'na.or.complete', or 'pairwise.complete.obs'.
method	The method to be use.

Value

Return a real number.

Author(s)

Oldemar Rodriguez Rojas

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Rodriguez, O. (2000). Classification et Modeles Lineaires en Analyse des Donnees Symboliques. Ph.D. Thesis, Paris IX-Dauphine University.

Examples

```
data(example3)
sym.data<-example3
var(sym.data[,1])
var(sym.data[,2])
var(sym.data[,6])
var(sym.data[,6], method='interval')
var(sym.data[,6], method='billard')
var(sym.data[,3], method='modal')
```

`variance.princ.curve` *Variance of the principal curve*

Description

Variance of the principal curve

Usage

```
variance.princ.curve(data,curve)
```

Arguments

`data`	Classic data table.
`curve`	The principal curve.

Value

The variance of the principal curve.

Author(s)

Jorge Arce.

References

Arce J. and Rodriguez O. (2015) ’Principal Curves and Surfaces to Interval Valued Variables’. The 5th Workshop on Symbolic Data Analysis, SDA2015, Orleans, France, November.

Hastie,T. (1984). Principal Curves and Surface. Ph.D Thesis Stanford University.

Hastie,T. \& Weingessel,A. (2014). princurve - Fits a Principal Curve in Arbitrary Dimension.R package version 1.1–12 http://cran.r-project.org/web/packages/princurve/index.html.

Hastie,T. \& Stuetzle, W. (1989). Principal Curves. Journal of the American Statistical Association, Vol. 84-406, 502–516.

Hastie, T., Tibshirani, R. \& Friedman, J. (2008). The Elements of Statistical Learning; Data Mining, Inference and Prediction. Springer, New York.

See Also

sym.interval.pc

`vertex.interval` *Vertex of the intervals*

Description

Vertex of the intervals

Usage

```
vertex.interval(sym.data)
```

Arguments

`sym.data`	Symbolic interval data table.

Value

Vertices of the intervals.

Author(s)

Jorge Arce.

References

Arce J. and Rodriguez O. (2015) 'Principal Curves and Surfaces to Interval Valued Variables'. The 5th Workshop on Symbolic Data Analysis, SDA2015, Orleans, France, November.

Hastie,T. (1984). Principal Curves and Surface. Ph.D Thesis Stanford University.

Hastie,T. \& Weingessel,A. (2014). princurve - Fits a Principal Curve in Arbitrary Dimension.R package version 1.1–12 http://cran.r-project.org/web/packages/princurve/index.html.

Hastie,T. \& Stuetzle, W. (1989). Principal Curves. Journal of the American Statistical Association, Vol. 84-406, 502–516.

Hastie, T., Tibshirani, R. \& Friedman, J. (2008). The Elements of Statistical Learning; Data Mining, Inference and Prediction. Springer, New York.

See Also

sym.interval.pc

`VeterinaryData` *Symbolic interval data example*

Description

Symbolic data matrix with all the variables of interval type.

Usage

```
data(VeterinaryData)
```

Format

$I Height Height $I Weight Weight

1 $I 120.0 180.0 $I 222.2 354.0

2 $I 158.0 160.0 $I 322.0 355.0

3 $I 175.0 185.0 $I 117.2 152.0

4 $I 37.9 62.9 $I 22.2 35.0

5 $I 25.8 39.6 $I 15.0 36.2

6 $I 22.8 58.6 $I 15.0 51.8

7 $I 22.0 45.0 $I 0.8 11.0

8 $I 18.0 53.0 $I 0.4 2.5

9 $I 40.3 55.8 $I 2.1 4.5

10 $I 38.4 72.4 $I 2.5 6.1

References

Billard L. and Diday E. (2006). Symbolic data analysis: Conceptual statistics and data mining. Wiley, Chichester.

Examples

```
data(VeterinaryData)
display.sym.table(VeterinaryData)
```

write.sym.table *Write Symbolic Data Table*

Description

This function write (save) a symbolic data table from a CSV data file.

Usage

```
write.sym.table(sym.data, file, sep, dec, row.names = NULL, col.names = NULL)
```

Arguments

sym.data	Symbolic data table
file	The name of the CSV file.
sep	As in R function read.table
dec	As in R function read.table
row.names	As in R function read.table
col.names	As in R function read.table

Value

Write in CSV file the symbolic data table.

Author(s)

Oldemar Rodriguez Rojas

References

Bock H-H. and Diday E. (eds.) (2000). Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data. Springer, Germany.

See Also

read.sym.table

Examples

```
## Not run:
data(example1)
write.sym.table(example1, file='temp4.csv', sep='|',dec='.', row.names=TRUE, col.names=TRUE)
ex1 <- read.sym.table('temp4.csv', header=TRUE, sep='|',dec='.', row.names=1)

## End(Not run)
```

Index