the symbolicDA package

March 19, 2018

Title: Analysis of Symbolic Data

Version: 0.5-3

Date: 2018-03-19

Author: Andrzej Dudek, Marcin Pelka <marcin.pelka@ue.wroc.pl>, Justyna Wilk <justyna.wilk@ue.wroc.pl> (to 2017-09-20), Marek Walesiak <Marek.Walesiak@ue.wroc.pl> (from 2018-02-01)

Maintainer: Andrzej Dudek <andrzej.dudek@ue.wroc.pl>

Depends: clusterSim, XML

Imports: rgl, shapes, e1071, ade4, cluster

Description: Symbolic data analysis methods: importing/exporting data from ASSO XML Files, distance calculation for symbolic data (Ichino-Yaguchi, de Carvalho measure), zoom star plot, 3d interval plot, multidimensional scaling for symbolic interval data, dynamic clustering based on distance matrix, HINoV method for symbolic data, Ichino's feature selection method, principal component analysis for symbolic interval data, decision trees for symbolic data based on optimal split with bagging, boosting and random forest approach (+visualization), kernel discriminant analysis for symbolic data, Kohonen's self-organizing maps for symbolic data, replication and profiling, artificial symbolic data generation.


License: GPL (>= 2)

URL: http://kei.ue.wroc.pl/symbolicDA

Encoding: UTF-8

R topics documented:

- bagging.SDA
Bagging algorithm for optimal split based on decision tree for symbolic objects

**Usage**

```
bagging.SDA(sdt, formula, testSet, mfinal=20, rf=FALSE,...)
```
The bagging, which stands for bootstrap aggregating, was introduced by Breiman in 1996. The diversity of classifiers in bagging is obtained by using bootstrapped replicas of the training data. Different training data subsets are randomly drawn with replacement from the entire training data set. Then each training data subset is used to train a decision tree (classifier). Individual classifiers are then combined by taking a simple majority vote of their decisions. For any given instance, the class chosen by most number of classifiers is the ensemble decision.

An object of class `bagging.SDA`, which is a list with the following components:

- **predclass**: the class predicted by the ensemble classifier
- **confusion**: the confusion matrix for ensemble classifier
- **error**: the classification error
- **pred**: ?
- **classfinal**: final class memberships

**Author(s)**

Andrzej Dudek <andrzej.dudek@ue.wroc.pl> Marcin Pelka <marcin.pelka@ue.wroc.pl>
Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland
http://keii.ue.wroc.pl/symbolicDA

**References**


**See Also**

`boosting.SDA`, `random_forest.SDA`, `decisionTree.SDA`
Examples

#Example will be available in next version of package, thank You for your patience :-)

boosting.SDA

boosting.SDA

Boosting algorithm for optimal split based decision tree for symbolic objects

Description

Boosting algorithm for optimal split based decision tree for symbolic objects, "symbolic" version of adabag.M1 algorithm

Usage

boosting.SDA(sdt,formula,testSet, mfinal = 20,...)

Arguments

- sdt: Symbolic data table
- formula: formula as in lm function
- testSet: a vector of integers indicating classes to which each objects are allocated in learning set
- mfinal: number of partial models generated
- ...: arguments passed to decisionTree.SDA function

Details

Boosting, similar to bagging, also creates an ensemble of classifiers by resampling the data. The results are then combined by majority voting. Resampling in boosting provides the most informative training data for each consecutive classifier. In each iteration of boosting three weak classifiers are created: the first classifier C1 is trained with a random subset of the training data. The training data subset for the next classifier C2 is chosen as the most informative subset, given C1. C2 is trained on a training data only half of which is correctly classified by C1 and the other half is misclassified. The third classifier C3 is trained with instances on which C1 and C2 disagree. Then the three classifiers are combined through a three-way majority vote.

Value

- formula: a symbolic description of the model that was used
- trees: trees built while making the ensemble
- weights: weights for each object from test set
- votes: final consensus clustering
- class: predicted class memberships
- error: error rate of the ensemble clustering

Author(s)

Andrzej Dudek <andrzej.dudek@ue.wroc.pl> Marcin Pelka <marcin.pelka@ue.wroc.pl>
Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland
http://keii.ue.wroc.pl/symbolicDA
References


See Also

`bagging.SDA`, `random.forest.SDA`, `decisionTree.SDA`

Examples

#Example will be available in next version of package, thank you for your patience :)
***cluster.Description.SDA***

*description of clusters of symbolic objects*

**Description**

description of clusters of symbolic objects is obtained by a generalisation operation using in most cases descriptive statistics calculated separately for each cluster and each symbolic variable.

**Usage**

```r
classic.Description.SDA(table.Symbolic, clusters, precision=3)
```

**Arguments**

- `table.Symbolic`: Symbolic data table
- `clusters`: a vector of integers indicating the cluster to which each object is allocated
- `precision`: Number of digits to round the results

**Value**

A List of cluster numbers, variable number and labels.

The description of clusters of symbolic objects which differs according to the symbolic variable type:

- for interval-valued variable:
  - "min value" - minimum value of the lower-bounds of intervals observed for objects belonging to the cluster
  - "max value" - maximum value of the upper-bounds of intervals observed for objects belonging to the cluster

- for multinominal variable:
  - "categories" - list of all categories of the variable observed for symbolic belonging to the cluster

- for multinominal with weights variable:
  - "min probabilities" - minimum weight of each category of the variable observed for objects belonging to the cluster
  - "max probabilities" - maximum weight of each category of the variable observed for objects belonging to the cluster
  - "avg probabilities" - average weight of each category of the variable calculated for objects belonging to the cluster
  - "sum probabilities" - sum of weights of each category of the variable calculated for objects belonging to the cluster

**Author(s)**

Andrzej Dudek <andrzej.dudek@ue.wroc.pl>, Justyna Wilk <justyna.wilk@ue.wroc.pl> Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland [http://keii.ue.wroc.pl/symbolicDA](http://keii.ue.wroc.pl/symbolicDA)
data_symbolic

References


See Also

SClust, DClust; hclust in stats library; pam in cluster library

Examples

```r
# LONG RUNNING - UNCOMMENT TO RUN
# data("cars", package="symbolicDA")
# y<-cars
# cl<-SClust(y, 4, iter=150)
# print(cl)
# o<-cluster.Description.SDA(y, cl)
# print(o)
```

data_symbolic  Symbolic interval data

Description

Artificially generated symbolic interval data

Format

3-dimensional array: 125 objects, 6 variables, third dimension represents begining and end of interval, 5-class structure

Source

Artificially generated data
DClust

Dynamical clustering based on distance matrix

Description

Dynamical clustering of objects described by symbolic and/or classic (metric, non-metric) variables based on distance matrix

Usage

DClust(dist, cl, iter=100)

Arguments

dist distance matrix
cl number of clusters or vector with initial prototypes of clusters
iter maximum number of iterations

Details

See file ../doc/DClust_details.pdf for further details

Value

a vector of integers indicating the cluster to which each object is allocated

Author(s)

Andrzej Dudek <andrzej.dudek@ue.wroc.pl>, Justyna Wilk <justyna.wilk@ue.wroc.pl> Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland http://keii.ue.wroc.pl/symbolicDA

References


See Also

SClust.dist.SDA; dist in stats library; dist.GDM in clusterSim library; pam in cluster library
Examples

# LONG RUNNING - UNCOMMENT TO RUN
#data("cars",package="symbolicDA")
#sdt<-cars
#dist<-dist.SDA(sdt, type="U_3")
#clust<-DClust(dist, cl=5, iter=100)
#print(clust)

decisionTree.SDA (Decison tree for symbolic data)

Description

Optimal split based decision tree for symbolic objects

Usage

decisionTree.SDA(sdt,formula,testSet,treshMin=0.0001,treshW=-1e10,
tNodes=NULL,minSize=2,epsilon=1e-4,useEM=FALSE,
multiNominalType="ordinal",rf=FALSE,rf.size,objectSelection)

Arguments

sdt Symbolic data table
formula formula as in ln function
testSet a vector of integers indicating classes to which each objects are allocated in
learnig set
treshMin parameter for tree creation algorithm
treshW parameter for tree creation algorithm
tNodes parameter for tree creation algorithm
minSize parameter for tree creation algorithm
epsilon parameter for tree creation algorithm
useEM use Expectation Optimalization algorithm for estinating conditional probabili-
ties
multiNominalType "ordinal" - functione treats multi-nominal data as ordered or "nominal" func-
tione treats multi-nomianal data as unordered (longer performance times)
rf if TRUE symbolic variables for tree creation are randomly chosen like in random
forest algorithm
rf.size the number of variables chosen for tree creation if rf is true
objectSelection optional, vector with symbolic object numbers for tree creation

Details

For further details see ..doc/decisionTree_SDA.pdf
Value

- nodes: nodes in tree
- nodeObjects: contribution of each objects nodes in tree
- conditionalProbab: conditional probability of belongingness of nodes to classes
- prediction: predicted classes for objects from testSet

Author(s)

Andrzej Dudek <andrzej.dudek@ue.wroc.pl> Marcin Pelka <marcin.pelka@ue.wroc.pl>
Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland
http://keii.ue.wroc.pl/symbolicDA

References


See Also

- bagging.SDA, boosting.SDA, random.forest.SDA, draw.decisionTree.SDA

Examples

C example 1
C long running - uncomment to run
C file samochody.xml needed in this example
C can be found in /inst/xml library of package
C sda<parse.SO("samochody")
C tree<decisionTree.SDA(sda, "Typ_samochodu-.", testSet=1:33)
C summary(tree) # a very general information
C tree # summary information

dist.SDA

distance measurement for symbolic data

Description

calculates distances between symbolic objects described by interval-valued, multinominal and multinominal with weights variables

Usage

dist.SDA(table.Symbolic,type="U_2",subType=NULL, gamma=0.5, power=2, probType="J", probAggregation="P_1", s=0.5, p=2, variableSelection=NULL, weights=NULL)
Arguments

table  symbolic data table

  type  distance measure for boolean symbolic objects: \( H, U_2, U_3, U_4, C_1, SO_1, SO_2, SO_3, SO_4, SO_5 \); mixed symbolic objects: \( L_1, L_2 \)

  subType  comparison function for \( C_1 \) and \( SO_1 \): \( D_1, D_2, D_3, D_4, D_5 \)

  gamma  gamma parameter for \( U_2 \) and \( U_3 \), gamma \([0, 0.5]\)

  power  power parameter for \( U_2 \) and \( U_3 \); power \([1, 2, 3, \ldots]\)

  probType  distance measure for probabilistic symbolic objects: \( J, CHI, REN, CHER, LP \)

  probAggregation  agregation function for \( J, CHI, REN, CHER, LP \): \( P_1, P_2 \)

  s  parameter for Renyi (REN) and Chernoff (CHE) distance, \( s \) \([0, 1]\)

  p  parameter for Minkowski (LP) metric; \( p=1 \) - manhattan distance, \( p=2 \) - euclidean distance

  variableSelection  numbers of variables used for calculation or NULL for all variables

  weights  weights of variables for Minkowski (LP) metrics

Details

Distance measures for boolean symbolic objects:

\( H \) - Hausdorff’s distance for objects described by interval-valued variables, \( U_2, U_3, U_4 \) - Ichino-Yaguchi’s distance measures for objects described by interval-valued and/or multinominal variables, \( C_1, SO_1, SO_2, SO_3, SO_4, SO_5 \) - de Carvalho’s distance measures for objects described by interval-valued and/or multinominal variables.

Distance measurement for probabilistic symbolic objects consists of two steps: 1. Calculation of distance between objects for each variable using componentwise distance measures: \( J \) (Kullback-Leibler divergence), \( CHI \) (Chi-2 divergence), \( REN \) (Renyi’s divergence), \( CHER \) (Chernoff’s distance), \( LP \) (modified Minkowski metrics). 2. Calculation of aggregative distance between objects based on componentwise distance measures using objectwise distance measure: \( P_1 \) (manhattan distance), \( P_2 \) (euclidean distance).

Distance measures for mixed symbolic objects - modified Minkowski metrics: \( L_1 \) (manhattan distance), \( L_2 \) (euclidean distance).

See file \( .. /doc/dist_SDA.pdf \) for further details

Value

distance matrix of symbolic objects

Author(s)

Andrzej Dudek \(<\text{andrzej.dudek}@\text{ue.wroc.pl}\>\), Justyna Wilk \(<\text{justyna.wilk}@\text{ue.wroc.pl}\>\) Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland http://keii.ue.wroc.pl/symbolicDA
References


See Also

DClust.index.Gld; dist.Symbolic in clusterSim library

Examples

```r
# LONG RUNNING - UNCOMMENT TO RUN
#data("cars",package="symbolicDA")
#dist<-dist.SDA(cars, type="U_3", gamma=0.3, power=2)
#print(dist)
```

draw.decisionTree.SDA  **Draws optimal split based decision tree for symbolic objects**

**Description**

Draws optimal split based decision tree for symbolic objects

**Usage**

```r
draw.decisionTree.SDA(decisionTree.SDA,boxWidth=1,boxHeight=3)
```

**Arguments**

- `decisionTree.SDA`: optimal split based decision tree for symbolic objects (result of `decisionTree.SDA` function)
- `boxWidth`: width of single box in drawing
- `boxHeight`: height of single box in drawing

**Details**

Draws optimal split based decision (classification) tree for symbolic objects.
Value

A draw of optimal split based decision (classification) tree for symbolic objects.

Author(s)

Andrzej Dudek <andrzej.dudek@ue.wroc.pl> Marcin Pelka <marcin.pelka@ue.wroc.pl>
Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland
http://keii.ue.wroc.pl/symbolicDA

References


See Also
decisionTree.SDA

Examples

# LONG RUNNING - UNCOMMENT TO RUN
# Files samochody.xml and wave.xml needed in this example
# can be found in /inst/xml library of package

# Example 1
#sdac<parse.SO("samochody")
#tree<decisionTree.SDA(sda, "Typ_samochodu~.", testSet=26:33)
#draw.decisionTree.SDA(tree,boxWidth=1,boxHeight=3)

# Example 2
#sdac<parse.SO("wave")
#tree<decisionTree.SDA(sda, "WaveForm~.", testSet=1:30)
#draw.decisionTree.SDA(tree,boxWidth=2,boxHeight=3)

generate.SO generation of artificial symbolic data table with given cluster structure

description of artificial symbolic data table with given cluster structure

Usage
generate.SO(numObjects,numClusters,numIntervalVariables,numMultivaluedVariables)
Arguments

numObjects
   number of objects in each cluster
numClusters
   number of objects
numIntervalVariables
   Number of symbolic interval variables in generated data table
numMultivaluedVariables
   Number of symbolic multi-valued variables in generated data table

Value

data
   symbolic data table with given cluster structure
clusters
   vector with cluster numbers for each object

Author(s)

Andrzej Dudek <andrzej.dudek@ue.wroc.pl>
Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland
http://keii.ue.wroc.pl/symbolicDA

References


See Also

see symbolic.object for symbolic data table R structure representation

Examples

# Example will be available in next version of package, thank You for your patience :-(

HINoV.SDA  Modification of HINoV method for symbolic data

Description

Carmone, Kara \& Maxwell’s Heuristic Identification of Noisy Variables (HINoV) method for symbolic data

Usage

HINoV.SDA(table.Symbolic, u=NULL, distance="H", Index="cRAND", method="pam",...)
**Arguments**

- **table.Symbolic**: symbolic data table
- **u**: number of clusters
- **distance**: symbolic distance measure as parameter type in `dist.SDA`
- **method**: clustering method: "single", "ward", "complete", "average", "mcquitty", "median", "centroid", "pam" (default), "SClust", "DClust"
- **Index**: "cRAND" - adjusted Rand index (default); "RAND" - Rand index
  - additional argument passed to `dist.SDA` function

**Details**

For HINoV in symbolic data analysis there can be used methods based on distance matrix such as hierarchical ("single", "ward", "complete", "average", "mcquitty", "median", "centroid") and optimization methods ("pam", "DClust") and also methods based on symbolic data table ("SClust").

See file `../doc/HINoVSDA_details.pdf` for further details

**Value**

- **parim**: $m \times m$ symmetric matrix ($m$ - number of variables). Matrix contains pairwise adjusted Rand (or Rand) indices for partitions formed by the $j$-th variable with partitions formed by the $l$-th variable
- **topri**: sum of rows of `parim`
- **stopri**: ranked values of `topri` in decreasing order

**Author(s)**

Andrzej Dudek <andrzej.dudek@ue.wroc.pl>, Justyna Wilk <justyna.wilk@ue.wroc.pl> Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland [http://keii.ue.wroc.pl/symbolicDA](http://keii.ue.wroc.pl/symbolicDA)

**References**


IchinoFS.SDA

Ichino’s feature selection method for symbolic data

Description

Ichino’s method for identifying non-noisy variables in symbolic data set

Useage

IchinoFS.SDA(table.Symbolic)

Arguments

table.Symbolic  symbolic data table

Details

See file ../doc/IchinoFSSDA_details.pdf for further details

Value

plot  plot of the gradient illustrating combinations of variables, in which the axis of ordinates (Y) represents the maximum number of mutual neighbor pairs and the axis of the abscissae (X) corresponds to the number of features (m)

combination  the best combination of variables, i.e. the combination most differentiating the set of objects

maximum results  step-by-step combinations of variables up to m variables

calculation results  .............

Author(s)

Andrzej Dudek <andrzej.dudek@ue.wroc.pl>, Justyna Wilk <justyna.wilk@ue.wroc.pl> Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland http://keii.ue.wroc.pl/symbolicDA
References


See Also

HINOV.SDA; HINOV.Symbolic in clusterSim library

Examples

```R
# LONG RUNNING – UNCOMMENT TO RUN
#data("cars", package="symbolicDA")
#sdt<-cars
#ichinos<-IchinoFS.SDA(sdt)
#print(ichino)
```

index.G1d  Calinski-Harabasz pseudo F-statistic based on distance matrix

Description

Calculates Calinski-Harabasz pseudo F-statistic based on distance matrix

Usage

```R
index.G1d (d, cl)
```

Arguments

d  
distance matrix (see dist.SDA)

cl  
a vector of integers indicating the cluster to which each object is allocated

Details

See file `../doc/indexG1d_details.pdf` for further details

Value

value of Calinski-Harabasz pseudo F-statistic based on distance matrix

Author(s)

Andrzej Dudek <andrzej.dudek@ue.wroc.pl>, Justyna Wilk <justyna.wilk@ue.wroc.pl> Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland http://keii.ue.wroc.pl/symbolicDA
References


Milligan, G.W., Cooper, M.C. (1985), *An examination of procedures of determining the number of cluster in a data set*, "Psychometrika", vol. 50, no. 2, 159-179. Available at: [https://dx.doi.org/10.1007/BF02294245](https://dx.doi.org/10.1007/BF02294245).


See Also


Examples

```
# LONG RUNNING - UNCOMMENT TO RUN
# Example 1
#library(stats)
data("cars",package="symbolicDA")
x<-cars
d<-dist.SDA(x, type="U_2")
wynik<-hclust(d, method="ward", members=NULL)
clusters<-cutree(wynik, 4)
Gld<index.Gld(d, clusters)
print(Gld)

# Example 2

#data("cars",package="symbolicDA")
#md <- dist.SDA(cars, type="U_3", gamma=0.5, power=2)
# nc - number_of_clusters
#min_nc=2
#max_nc=10
#res <- array(0,c(max_nc-min_nc+1,2))
#res[,1] <- min_nc:max_nc
#clusters <- NULL
#for (nc in min_nc:max_nc)
#{
#cl2 <- pam(md, nc, diss=TRUE)
#res[nc-min_nc+1,2] <- Gld <- index.Gld(md,cl2$clustering)
#clusters <- rbind(clusters, cl2$clustering)
#}
#print(paste("max Gld for",(min_nc:max_nc)[which.max(res[,2])],"clusters="mmax(res[,2])))
#print("clustering for max Gld")
#print(clusters[which.max(res[,2]),])
```
interscal.SDA

Multidimensional scaling for symbolic interval data - InterScal algorithm

Description
Multidimensional scaling for symbolic interval data - InterScal algorithm

Usage
interscal.SDA(x,d=2,calculateDist=FALSE)

Arguments
x
symbolic interval data: a 3-dimensional table, first dimension represents object number, second dimension - variable number, and third dimension contains lower- and upper-bounds of intervals (Simple form of symbolic data table)

d
Dimensionality of reduced space

calculateDist
if TRUE x are treated as raw data and min-max dist matrix is calculated. See details

Details
Interscal is the adaptation of well-known classical multidimensional scaling for symbolic data. The input for Interscal is the interval-valued dissimilarity matrix. Such dissimilarity matrix can be obtained from symbolic data matrix (that contains only interval-valued variables), judgements obtained from experts, respondents. See Lechevallier Y. (2001) for details on calculating interval-valued distance. See file ../doc/Symbolic_MDS.pdf for further details

Value
xprim
coordinates of rectangles

stress.sym
final STRESSSym value

Author(s)
Andrzej Dudek <andrzej.dudek@ue.wroc.pl> Marcin Pelka <marcin.pelka@ue.wroc.pl>
Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland
http://keii.ue.wroc.pl/symbolicDA
References


See Also

iscal.SDA, symscal.SDA

Examples

# LONG RUNNING - UNCOMMENT TO RUN
# sda<parse.SOL("samochody")
# data<sdadindivIC
# mds<interscal.SDA(data, d=2, calculateDist=TRUE)

iscal.SDA

Multidimensional scaling for symbolic interval data - IScal algorithm

Description

Multidimensional scaling for symbolic interval data - IScal algorithm

Usage

iscal.SDA(x,d=2,calculateDist=FALSE)

Arguments

x symbolic interval data: a 3-dimensional table, first dimension represents object number, second dimension - variable number, and third dimension contains lower- and upper-bounds of intervals (Simple form of symbolic data table)

d Dimensionality of reduced space

calculateDist if TRUE x are treated as raw data and min-max dist matrix is calculated. See details

Details

IScal, which was proposed by Groenen et. al. (2006), is an adaptation of well-known nonmetric multidimensional scaling for symbolic data. It is an iterative algorithm that uses I-STRESS objective function. This function is normalized within the range [0; 1] and can be interpreted like classical STRESS values. IScal, like Interscal and SymScal, requires interval-valued dissimilarity matrix. Such dissimilarity matrix can be obtained from symbolic data matrix (that contains only interval-valued variables), judgements obtained from experts, respondents. See Lechevallier Y. (2001) for details on calculating interval-valued distance. See file ../doc/Symbolic_MDS.pdf for further details.
Kernel discriminant analysis for symbolic data

Usage

\texttt{kernel.SDA(sdt, formula, testSet, h, ...)}

Arguments

- \texttt{sdt}: symbolic data table
- \texttt{formula}: a formula, as in the \texttt{lm} function
- \texttt{testSet}: vector with numbers objects \texttt{ij} test set
- \texttt{h}: kernel bandwidth size
- \texttt{...}: arguments passed to \texttt{dist.SDA} function

Value

- \texttt{xprim}: coordinates of rectangles
- \texttt{STRESSSym}: final \texttt{STRESSSym} value

Author(s)

Andrzej Dudek \texttt{<andrzej.dudek@ue.wroc.pl>}

Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland

\texttt{http://keii.ue.wroc.pl/symbolicDA}

References


See Also

\texttt{interscal.SDA,symscal.SDA}

Examples

\begin{verbatim}
# Example will be available in next version of package, thank You for your patience :-)
\end{verbatim}
**Details**

Kernel discriminant analysis for symbolic data is based on the intensity estimator (that is based on dissimilarity measure for symbolic data) due to the fact that classical well-known density estimator can not be applied. Density estimator can not be applied due to the fact that symbolic objects are not object of euclidean space and the integral operator for symbolic data is not applicable.

For further details see `../doc/Kernel_SDA.pdf`.

**Value**

vector of class belongs of each object in test set

**Author(s)**

Andrzej Dudek <andrzej.dudek@ue.wroc.pl>
Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland
http://keii.ue.wroc.pl/symbolicDA

**References**


**See Also**

dist.SDA

**Examples**

```r
# Example 1
# LONG RUNNING - UNCOMMENT TO RUN
# sda<parse.c("samochody")
# model<-kernel.SDA(sda, "Typ_samochodu-", testSet=6:16, h=0.75)
# print(model)
```

---

Kohonen’s self-organizing maps for symbolic interval-valued data

**Description**

Kohonen’s self-organizing maps for a set of symbolic objects described by interval-valued variables

**Usage**

```r
kohonen.SDA(data, rlen=100, alpha=c(0.05,0.01))
```
Arguments

data  symbolic data table in simple form (see \texttt{SO2Simple})
rlen  number of iterations (the number of times the complete data set will be presented to the network)
alpha  learning rate, determining the size of the adjustments during training. Default is to decline linearly from 0.05 to 0.01 over rlen updates

Details

See file \texttt{../doc/kohonenSDA_details.pdf} for further details

Value

clas  vector of mini-class belonginers in a test set
prot  prototypes

Author(s)

Andrzej Dudek \texttt{<andrzej.dudek@ue.wroc.pl>}, Justyna Wilk \texttt{<justyna.wilk@ue.wroc.pl>} Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland \url{http://keii.ue.wroc.pl/symbolicDA}

References


See Also

\texttt{SO2Simple}; som in kohonen library

Examples

# Example will be available in next version of package, thank you for your patience :-}
parse.SO

Reading symbolic data table from ASSO-format XML file

Description
Kohonen self organizing maps for symbolic data with interval variables

Usage
parse.SO(file)

Arguments
file file name without xml extension

Details
see symbolic.object for symbolic data table R structure representation

Value
Symbolic data table parsed from XML file

Author(s)
Andrzej Dudek <andrzej.dudek@ue.wroc.pl>
Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland
http://keii.ue.wroc.pl/clusterSim

References
Billard L., Diday E. (eds.) (2006), Symbolic Data Analysis, Conceptual Statistics and Data Mining,
John Wiley & Sons, Chichester.

statistical information from complex data, Springer-Verlag, Berlin.

Diday E., Noirhomme-Fraiture M. (eds.) (2008), Symbolic Data Analysis with SODAS Software,
John Wiley & Sons, Chichester.

See Also
save.SO, generate.SO

Examples
#cars<-parse.SO("cars")
PCA.centers.SDA

principal component analysis for symbolic objects described by symbolic interval variables. Centers algorithm

Description

principal component analysis for symbolic objects described by symbolic interval variables. Centers algorithm

Usage

`PCA.centers.SDA(t, pc.number=2)`

Arguments

- `t`: symbolic interval data: a 3-dimensional table, first dimension represents object number, second dimension - variable number, and third dimension contains lower- and upper-bounds of intervals (Simple form of symbolic data table)
- `pc.number`: number of principal components

Details

See file `./doc/PCA_SDA.pdf` for further details

Value

Data in reduced space (symbolic interval data: a 3-dimensional table)

Author(s)

Andrzej Dudek <andrezj.dudek@ue.wroc.pl>
Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland
http://keii.ue.wroc.pl/symbolicDA

References


See Also

`PCA.mrPCA.SDA, PCA.spaghetti.SDA, PCA.sPCA.SDA, PCA.vertices.SDA`

Examples

# Example will be available in next version of package, thank you for your patience :-)
PCA.mr pca.SDA

principal component analysis for symbolic objects described by symbolic interval variables. Midpoints and radii algorithm

Description

principal component analysis for symbolic objects described by symbolic interval variables. Midpoints and radii algorithm

Usage

PCA.mr pca.SDA(t, pc.number=2)

Arguments

t symbolic interval data: a 3-dimensional table, first dimension represents object number, second dimension - variable number, and third dimension contains lower- and upper-bounds of intervals (Simple form of symbolic data table)

pc.number number of principal components

Details

See file ../doc/PCA_SDA.pdf for further details

Value

Data in reduced space (symbolic interval data: a 3-dimensional table)

Author(s)

Andrzej Dudek <andrzej.dudek@ue.wroc.pl>
Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland
http://keii.ue.wroc.pl/symbolicDA

References


See Also

PCA.cent ers.SDA, PCA.spaghetti.SDA, PCA.spca.SDA, PCA.vertices.SDA

Examples

# Example will be available in next version of package, thank you for your patience :-)
PCA.spaghetti.SDA

Principal component analysis for symbolic objects described by symbolic interval variables. Spaghetti algorithm

Description

Principal component analysis for symbolic objects described by symbolic interval variables. Spaghetti algorithm

Usage

PCA.spaghetti.SDA(t, pc.number=2)

Arguments

t symbolic interval data: a 3-dimensional table, first dimension represents object number, second dimension - variable number, and third dimension contains lower- and upper-bounds of intervals (Simple form of symbolic data table)

pc.number number of principal components

Details

See file ./doc/PCA_SDA.pdf for further details

Value

Data in reduced space (symbolic interval data: a 3-dimensional table)

Author(s)

Andrzej Dudek <andrzej.dudek@ue.wroc.pl>

Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland

http://keii.ue.wroc.pl/symbolicDA

References


See Also

PCA.centers.SDA, PCA.mrPCA.SDA, PCA.spca.SDA, PCA.vertices.SDA

Examples

# Example will be available in next version of package, thank you for your patience :-)}
PCA.spca.SDA principal component analysis for symbolic objects described by symbolic interval variables. 'Symbolic' PCA algorithm

Description

principal component analysis for symbolic objects described by symbolic interval variables. 'Symbolic' PCA algorithm

Usage

PCA.spca.SDA(t, pc.number=2)

Arguments

t symbolic interval data: a 3-dimensional table, first dimension represents object number, second dimension - variable number, and third dimension contains lower- and upper-bounds of intervals (Simple form of symbolic data table)

pc.number number of principal components

Details

See file ../doc/PCA_SDA.pdf for further details

Value

Data in reduced space (symbolic interval data: a 3-dimensional table)

Author(s)

Andrzej Dudek <andrzej.dudek@ue.wroc.pl>
Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland http://kei.ue.wroc.pl/symbolicDA

References


See Also

PCA.centers.SDA, PCA.mrpca.SDA, PCA.spaghetti.SDA, PCA.vertices.SDA

Examples

# Example will be available in next version of package, thank you for your patience :-)

# Example will be available in next version of package, thank you for your patience :-}
Description

Principal component analysis for symbolic objects described by symbolic interval variables. Vertices algorithm

Usage

PCA.vertices.SDA(t, pc.number=2)

Arguments

t: symbolic interval data: a 3-dimensional table, first dimension represents object number, second dimension - variable number, and third dimension contains lower- and upper-bounds of intervals (Simple form of symbolic data table)

pc.number: number of principal components

Details

See file ./doc/PCA_SDA.pdf for further details

Value

Data in reduced space (symbolic interval data: a 3-dimensional table)

Author(s)

Andrzej Dudek <andrzej.dudek@ue.wroc.pl>
Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland
http://keii.ue.wroc.pl/symbolicDA

References


See Also

PCA.centers.SDA, PCA.mrPCA.SDA, PCA.spaghetti.SDA, PCA.spca.SDA

Examples

# Example will be available in next version of package, thank you for your patience :-(
plot3dInterval

3D plot for symbolic interval-valued data

Description
3-dimensional plot for symbolic objects described by interval-valued variables

Usage
plot3dInterval(data, colors)

Arguments
data symbolic data table consists of a set of symbolic objects described by interval-valued variables
colors set of colors (see colors) to mark symbolic objects

Value
3-dimensional plot for symbolic interval-valued data in which each axis represents a symbolic interval-valued variable and each cuboid represents a symbolic object

Author(s)
Andrzej Dudek <andrzej.dudek@ue.wroc.pl>, Justyna Wilk <justyna.wilk@ue.wroc.pl> Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland http://keii.ue.wroc.pl/symbolicDA

References


See Also
SClust;plotInterval in clusterSim library

Examples
# LONG RUNNING - UNCOMMENT TO RUN
#means <- matrix(c(0,0,0,
#0,6,0,
#0,6,6,
#6,0,0,
#6,0,6,
#6,6,0,
#6,6,6),8,3,byrow=TRUE)
#means<-means*1.5
**random.forest.SDA**

Random forest algorithm for optimal split based decision tree for symbolic objects

**Description**

Random forest algorithm for optimal split based decision tree for symbolic objects

**Usage**

```r
random.forest.SDA(sdt, formula, testSet, mfinal = 100, ...)
```

**Arguments**

- `sdt` Symbolic data table
- `formula` formula as in `lm` function
- `testSet` a vector of integers indicating classes to which each objects are allocated in learning set
- `mfinal` number of partial models generated
- `...` arguments passed to `decisionTree.SDA` function

**Details**

`random.forest.SDA` implements Breiman’s random forest algorithm for classification of symbolic data set.

**Value**

Section details goes here

**Author(s)**

Andrzej Dudek <andrzej.dudek@ue.wroc.pl> Marcin Pelka <marcin.pelka@ue.wroc.pl>
Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland
http://keii.ue.wroc.pl/symbolicDA

**References**


See Also

`bagging.SDA`, `boosting.SDA`, `decisionTree.SDA`

Examples

```r
# Example will be available in next version of package, thank you for your patience :-)
```

---

### Description

Replication analysis for cluster validation of symbolic data

### Usage

```r
replication.SDA(table.Symbolic, u=2, method="SClust", S=10, fixedAsample=NULL, ...)
```

### Arguments

- `table.Symbolic`: symbolic data table
- `u`: number of clusters given arbitrarily
- `method`: clustering method: "SClust" (default), "DClust", "single", "complete", "average", "mcquitty", "median", "centroid", "ward", "pam", "diana"
- `S`: the number of simulations used to compute average adjusted Rand index
- `fixedAsample`: if NULL a sample is generated randomly, otherwise this parameter contains object numbers arbitrarily assigned to A sample
- `...`: additional argument passed to `dist.SDA` function

### Details

See file `./doc/replicationSDA_details.pdf` for further details

### Value

- `A`: 3-dimensional array containing data matrices for A sample of objects in each simulation (first dimension represents simulation number, second - object number, third - variable number)
- `B`: 3-dimensional array containing data matrices for B sample of objects in each simulation (first dimension represents simulation number, second - object number, third - variable number)
- `medoids`: 3-dimensional array containing matrices of observations on `u` representative objects (medoids) for A sample of objects in each simulation (first dimension represents simulation number, second - cluster number, third - variable number)
- `clusteringA`: 2-dimensional array containing cluster numbers for A sample of objects in each simulation (first dimension represents simulation number, second - object number)
**RSDA2SymbolicDA**

clusteringB  2-dimensional array containing cluster numbers for B sample of objects in each simulation (first dimension represents simulation number, second - object number)

clusteringBB 2-dimensional array containing cluster numbers for B sample of objects in each simulation according to 4 step of replication analysis procedure (first dimension represents simulation number, second - object number)

cRand  value of average adjusted Rand index for S simulations

**Author(s)**
Andrzej Dudek <andrzej.dudek@ue.wroc.pl>, Justyna Wilk <justyna.wilk@ue.wroc.pl> Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland http://keii.ue.wroc.pl/symbolicDA

**References**


**See Also**

dist.SDA, SClust, DClust; hclust in stats library; pam in cluster library; replication.Mod in clusterSim library

**Examples**

```r
#data("cars",package="symbolicDA")
#set.seed(123)
#w<-replication.SDA(cars, u=3, method="SClust", S=10)
#print(w)
```

**RSDA2SymbolicDA**  Read a Symbolic Table from

**Description**

It reads a symbolic data table from a CSV file or converts RSDA object to SymbolicDA "symbolic" class type object
Usage

RSDA2SymbolicDA(rsda.object=NULL, from.csv=F, file=NULL, header = TRUE, sep, dec, row.names = NULL)

Arguments

rsda.object object of class "symb.data.table" from (former) RSDA package
from.csv object of class "symb.data.table" from (former) RSDA package
file optional, The name of the CSV file in RSDA format (see details)
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

(as in (former) RSDA package) 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] "SC" "SI" "SH" "SS"
$sym.var.length
save.SO

[1] 1 2 3 4
$sym.var.starts
[1] 2 4 8 13
$sym.meta
SC F1 $I F2 F2 $SH F3 M1 M2 M3 $SS F4 E1 E2 E3 E4
Case1 $SC 2.8 $SI 1 2 $SH 3 0.1 0.7 0.2 $SS 4 e g k i
Case2 $SC 1.4 $SI 3 9 $SH 3 0.6 0.3 0.1 $SS 4 a b c d
Case3 $SC 3.2 $SI -1 4 $SH 3 0.2 0.2 0.6 $SS 4 2 1 b c
Case4 $SC -2.1 $SI 0 2 $SH 3 0.9 0.0 0.1 $SS 4 3 4 c a
Case5 $SC -3.0 $SI -4 -2 $SH 3 0.6 0.0 0.4 $SS 4 e i g k
$sym.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 in form of SymbolicDA "symbolic" class type object.

Author(s)

Andrzej Dudek

With included code from (former) RSDA package by Oldemar Rodriguez Rojas

References


See Also
display.sym.table

Examples

# Example will be available in next version of package, thank You for your patience :-)

save.SO  saves symbolic data table of 'symbolic' class to xml file

Description

saves symbolic data table of 'symbolic' class to xml file (ASSO format)

Usage

save.SO(sdt, file)
Arguments

sdt          Symbolic data table
file         file name with extension

Details

see symbolic.object for symbolic data table R structure representation

Value

No value returned

Author(s)

Andrzej Dudek <andrzej.dudek@ue.wroc.pl>
Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland
http://keii.ue.wroc.pl/symbolicDA

References


See Also

generate.SO, subsdt.SDA, parse.SO

Examples

#data("cars",package="symbolicDA")
#save.SO(cars,file="cars_backup.xml")

SClust

Dynamical clustering of symbolic data

Description

Dynamical clustering of symbolic data based on symbolic data table

Usage

SClust(table.Symbolic, cl, iter=100, variableSelection=NULL, objectSelection=NULL)
Arguments

tableSymbolic  symbolic data table
c1               number of clusters or vector with initial prototypes of clusters
iter             maximum number of iterations
variableSelection vector of numbers of variables to use in clustering procedure or NULL for all variables
objectSelection  vector of numbers of objects to use in clustering procedure or NULL for all objects

Details
See file ../doc/SClust_details.pdf for further details

Value
a vector of integers indicating the cluster to which each object is allocated

Author(s)
Andrzej Dudek <andrzej.dudek@ue.wroc.pl>, Justyna Wilk <justyna.wilk@ue.wroc.pl> Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland
http://kei11.ue.wroc.pl/symbolicDA

References


See Also
dclust; kmeans in stats library

Examples

# LONG RUNNING - UNCOMMENT TO RUN
#data("cars", package="symbolicDA")
#sdt<-cars
#clust<-SClust(sdt, cl=3, iter=50)
#print(clust)
simple2SO

Change of representation of symbolic data from simple form to symbolic data table

Description

Change of representation of symbolic data from simple form to symbolic data table

Usage

simple2SO(x)

Arguments

x  symbolic interval data: a 3-dimensional table, first dimension represents object number, second dimension - variable number, and third dimension contains lower- and upper-bounds of intervals

Details

see symbolic.object for symbolic data table R structure representation

Value

Symbolic data table in full form

Author(s)

Andrzej Dudek <andrzej.dudek@ue.wroc.pl>

Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland

http://keii.ue.wroc.pl/symbolicDA

References


See Also

link{SO2Simple}

Examples

# Example will be available in next version of package, thank You for your patience :-}
SO2Simple

SO2Simple  Change of representation of symbolic data from symbolic data table to simple form

Description

Change of representation of symbolic data from symbolic data table to simple form

Usage

SO2Simple(sd)

Arguments

sd  Symbolic data table in full form

Details

see symbolic.object for symbolic data table R structure representation

Value

symbolic interval data: a 3-dimensional table, first dimension represents object number, second dimension - variable number, and third dimension contains lower- and upper-bounds of intervals

Author(s)

Andrzej Dudek <andrzej.dudek@ue.wroc.pl>
Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland
http://kei1.ue.wroc.pl/symbolicDA

References


See Also

link{simple2SO}

Examples

# Example will be available in next version of package, thank You for your patience :-)

Subset of symbolic data table

**Description**

This method creates symbolic data table containing only objects, whose indices are given in second argument.

**Usage**

```r
subsdt.SDA(sdt, objectSelection)
```

**Arguments**

- `sdt`  
  Symbolic data table

- `objectSelection`  
  vector containing symbolic object numbers, default value - all objects from sdt

**Details**

see `symbolic.object` for symbolic data table R structure representation.

**Value**

Symbolic data table containing only objects, whose indices are given in second argument. The result is of 'symbolic' class

**Author(s)**

Andrzej Dudek <andrzej.dudek@ue.wroc.pl>
Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland
http://keli.ue.wroc.pl/symbolicDA

**References**


**See Also**

- `generate.S0`, `save.S0`, `parse.S0`

**Examples**

# Example will be available in next version of package, thank You for your patience :-)

symbolic.object  

Symbolic data table Object

Description
These are objects representing symbolic data table structure

Details
For all fields symbol N.A. means not available value.
For further details see ./doc/SDA.pdf

Value

individuals  data frame with one row for each row in symbolic data table with following columns:
num - symbolic object (described by symbolic data table row) ordering number, usually from 1 to number of symbolic objects;
name - short name of symbolic object with no spaces;
label - full descriptive name of symbolic object.

variables  data frame with one row for each column in symbolic data table with following columns:
num - symbolic variable (adequate to symbolic data table column) ordering number, usually from 1 to number of symbolic variables;
name - short name of symbolic variable with no spaces;
label - full descriptive name of symbolic variable;
type - type of symbolic variable: IC (InterContinous) - Symbolic interval variable type, every realization of symbolic variable of this type on symbolic object takes form of numerical interval; C (Continous) - Symbolic interval variable type, every realization of symbolic variable of this type on symbolic object takes form of numerical interval for which beginning is equal to end (equivalent to simple "numeric" value); MN (MultiNominal) - every realization of multi nominal symbolic variable on symbolic objects takes form of set of nominal values; NM ((Multi) Nominal Modif) - every realization of nominal symbolic variable on symbolic objects takes form of distribution of probabilities (set of nominal values with weights summing to one); N (Nominal) - every realization of nominal symbolic variable on symbolic objects is one value (or N.A.)
details - id of this variable in details table appropriate for this kind of variable (detailsN for nominal and multi nominal variables, detailsIC for symbolic interval variables, detailsC for continuous (metric single-valued) variables, detailsNM of multi nominal with weights variables).

detailsC  data frame describing symbolic continuous (metric, single-valued) variables details with following columns:
na - number of N.A. (not available) variables realization;
num - not used, left for compatibility with ASSO-XML specification;
min - beginning of interval representing symbolic interval variable domain (minimal value of all realizations of this variable on all symbolic objects);
max - end of interval representing symbolic interval variable domain (maximal value of all realizations of this variable on all symbolic objects).
detailsIC  data frame describing symbolic inter-continuous (symbolic interval) variables details with following columns:
na - number of N.A. (not available) variables realizations;
nu - not used, left for compatibility with ASSO-XML specification;
min - beginning of interval representing symbolic interval variable domain (minimal value of all beginnings of interval realizations of this variable on all symbolic objects);
max - end of interval representing symbolic interval variable domain (maximal value of all ends of interval realizations of this variable on all symbolic objects).

detailsN  data frame describing symbolic nominal and multi nominal variables details with following columns:
na - number of N.A. variables realizations;
nu - not used, left for compatibility with ASSO-XML specification;
modals - number of categories in symbolic variable domain. Each categorie is described in detailsListNom.

detailsListNom  data frame describing every category of symbolic nominal and multi nominal variables, with following columns:
details_no - number of variable in detailsN to which domain belongs category;
um - number of category within variable domain;
name - category short name
label - category full name

detailsNM  data frame describing symbolic multi nominal modiff (categories sets with weights) variables details with following columns:
na number of N.A. (not available) variables realizations.
nu not used, left for compatibility with ASSO-XML specification
modals number of categories in symbolic variable domain. Each categorie is described in detailsListNomModiff

detailsListNomModif  data frame describing every category of symbolic multi nominal modiff variables, with following columns
details_no - number of variable in detailsNM to which domain belongs category
num - number of category within variable domain
name - category short name
label - category full name

indivIC  array of symbolic interval variables realizations, with dimensions nr_of_objects X nr_of_variables X 2 containing beginnings and ends of intervals for given object and variable. For values different type than symbolic interval array contains zeros

indivC  array of symbolic continues variables realizations, with dimensions nr_of_objects X nr_of_variables X 1 containing single values - realizations of variable on symbolic object. For values different type than symbolic continous array contains zeros

indivN  data frame describing symbolic nominal and multi nonimal variables realizations with folowing columns:
indiv - id of symbolic object from individuals;
variable - id of symbolic object from variables;
value - id of category object from detailsListNom;
When this data frame contains line $i,j,k$ it means that category $k$ belongs to set that is realization of $j$-th symbolic variable on $i$-th symbolic object.

 indivNM data frame describing symbolic multi nonimal modiff variables realizations with following columns:
indiv - id of symbolic object from individuals;
variable - id of symbolic object from variables;
value - id of category object from detailsListNom;
frequency - weight of category;
When this data frame contains line $i,j,k,w$ it means that category $k$ belongs to set that is realization of $j$-th symbolic variable on $i$-th symbolic object with weight(probability) $w$.

Structure
The following components must be included in a legitimate symbolic object.

See Also
dist.SDA.

---

**symscal.SDA**

Multidimensional scaling for symbolic interval data - SymScal algorithm

**Description**

Multidimensional scaling for symbolic interval data - symScal algorithm

**Usage**

```r
symscal.SDA(x,d=2,calculateDist=FALSE)
```

**Arguments**

- **x** symbolic interval data: a 3-dimensional table, first dimension represents object number, second dimension - variable number, and third dimension contains lower- and upper-bounds of intervals (Simple form of symbolic data table)
- **d** Dimensionality of reduced space
- **calculateDist** if TRUE x are treated as raw data and min-max dist matrix is calculated. See details

**Details**

SymScal, which was proposed by Groenen et. al. (2005), is an adaptation of well-known nonmetric multidimensional scaling for symbolic data. It is an iterative algorithm that uses STRESS objective function. This function is unnormalized. IScal, like Interscal and SymScal, requires interval-valued dissimilarity matrix. Such dissimilarity matrix can be obtained from symbolic data matrix (that contains only interval-valued variables), judgements obtained from experts, respondents. See Lechevallier Y. (2001) for details on calculating interval-valued distance. See file `../doc/Symbolic_MDS.pdf` for further details
Value

xprim coordinates of rectangles
STRESSSym final STRESSSym value

Author(s)

Andrzej Dudek <andrzej.dudek@ue.wroc.pl>
Department of Econometrics and Computer Science, University of Economics, Wroclaw, Poland
http://keii.ue.wroc.pl/symbolicDA

References


See Also

iscal.SDA, interscal.SDA

Examples

# Example will be available in next version of package, thank You for your patience :-)
**Arguments**

- `table.Symbolic` symbolic data table
- `j` symbolic object number in symbolic data table used to create the chart
- `variableSelection` numbers of symbolic variables describing symbolic object used to create the chart, if NULL all variables are used
- `offset` relational offset of chart (margin size)
- `firstTick` place of first tick (relational to length of axis)
- `labelCex` labels cex parameter of labels
- `labelOffset` relational offset of labels
- `tickLength` relational length of single tick of axis
- `histWidth` histogram (for modal variables) relational width
- `histHeight` histogram (for modal variables) relational height
- `rotateLabels` if TRUE labels are rotated due to rotation of axes
- `variableCex` cex parameter of names of variables

**Value**

zoom star chart for selected symbolic object in which each axis represents a symbolic variable. Depending on the type of symbolic variable their implementations are presented as:

a) rectangle - interval range of interval-valued variable,

b) circles - categories of multinominal (or multinominal with weights) variable from among coloured circles means categories of the variable observed for the selected symbolic object

bar chart - additional chart for multinominal with weights variable in which each bar represents a weight (percentage share) of a category of the variable

**Author(s)**

Andrzej Dudek <andrzej.dudek@ue.wroc.pl>, Justyna Wilk <justyna.wilk@ue.wroc.pl> Department of Econometrics and Computer Science, Wroclaw University of Economics, Poland [http://keii.ue.wroc.pl/symbolicDA](http://keii.ue.wroc.pl/symbolicDA)

**References**


**See Also**

`plot3dInterval`, `plotInterval` in clusterSim
Examples

# LONG RUNNING - UNCOMMENT TO RUN
# Example 1
#data("cars", package="symbolicDA")
#sdt<-cars
#zoomStar(sdt, j=12)

# Example 2
#data("cars", package="symbolicDA")
#sdt<-cars
#variables<-as.matrix(sdt$variables)
#indivN<-as.matrix(sdt$indivN)
#dist<-as.matrix(dist.SDA(sdt))
#classes<-DClust(dist, cl=5, iter=100)
#for(i in 1:max(classes)){
#  #getOption("device")()
#  #zoomStar(sdt, .medoid2(dist, classes, i))}
# Index

*Topic **MDS**
  interscal.SDA, 19
  iscal.SDA, 20
  symscal.SDA, 43
*Topic **Multidimensional**
  interscal.SDA, 19
  iscal.SDA, 20
  symscal.SDA, 43
*Topic **PCA**
  PCA.centers.SDA, 25
  PCA.mrPCA.SDA, 26
  PCA.spaghetti.SDA, 27
  PCA.spca.SDA, 28
  PCA.vertices.SDA, 29
*Topic **Principal components analysis**
  PCA.centers.SDA, 25
  PCA.mrPCA.SDA, 26
  PCA.spaghetti.SDA, 27
  PCA.spca.SDA, 28
  PCA.vertices.SDA, 29
*Topic **RSDA**
  RSDA2SymbolicDA, 33
*Topic **SDA**
  symbolic.object, 41
*Topic **SymbolicDA**
  RSDA2SymbolicDA, 33
*Topic **Symbolic**
  RSDA2SymbolicDA, 33
*Topic **Table**
  RSDA2SymbolicDA, 33
*Topic **cluster**
  bagging.SDA, 2
  boosting.SDA, 4
  cluster.Description.SDA, 6
  DClust, 8
  HINoV.SDA, 14
  IchinoFS.SDA, 16
  index.G1d, 17
  kernel.SDA, 21
  kohonen.SDA, 22
  parse.SO, 24
  replication.SDA, 32
  save.SO, 35
  SClust, 36
  simple2SO, 38
  SO2Simple, 39
  subsdt.SDA, 40
  zoomStar, 44
*Topic **datasets**
  cars, 5
  data_symbolic, 7
*Topic **decision tree**
  draw.decisionTree.SDA, 12
  random.forest.SDA, 31
*Topic **hplot**
  plot3dInterval, 30
*Topic **multivariate**
  replication.SDA, 32
*Topic **scaling**
  interscal.SDA, 19
  iscal.SDA, 20
  symscal.SDA, 43
*Topic **symbolic,SDA**
  decisionTree.SDA, 9
  dist.SDA, 10
  generate.SO, 13
*Topic **symbolic**
  symbolic.object, 41
  bagging.SDA, 2, 5, 10, 32
  boosting.SDA, 3, 4, 10, 32
  cars, 5
  cluster.Description.SDA, 6
  colors, 30
  data_symbolic, 7
  DClust, 7, 8, 12, 18, 33, 37
  decisionTree.SDA, 3, 5, 9, 13, 32
  dist.SDA, 8, 10, 15, 17, 22, 32, 33, 43
  draw.decisionTree.SDA, 10, 12
  generate.SO, 13, 24, 36, 40
  hclust, 7
  HINoV.SDA, 14, 17
  IchinoFS.SDA, 16
INDEX

index.G1d, 12, 17
interscal.SDA, 19, 21, 44
iscal.SDA, 20, 20, 44

kernel.SDA, 21
kmeans, 37
kohonen.SDA, 22

parse.SO, 24, 36, 40
PCA.centers.SDA, 25, 26–29
PCA.mrpca.SDA, 25, 26, 27–29
PCA.spaghetti.SDA, 25, 26, 27, 28, 29
PCA.spc.a.SDA, 25–27, 28, 29
PCA.vertices.SDA, 25–28, 29
plot3dInterval, 30, 45

random.forest.SDA, 3, 5, 10, 31
replication.SDA, 32
RSDA2SymbolicDA, 33

save.SO, 24, 35, 40
SClust, 7, 8, 18, 30, 33, 36
simple2SO, 38
SO2Simple, 23, 39
subsdt.SDA, 36, 40
symbolic.object, 14, 24, 36, 38–40, 41
symscal.SDA, 20, 21, 43

zoomStar, 44