Package ‘preproviz’

July 9, 2016

Title Tools for Visualization of Interdependent Data Quality Issues
Version 0.2.0
Date 2016-7-9

Description Data quality issues such as missing values and outliers are often interdependent, which makes preprocessing both time-consuming and leads to suboptimal performance in knowledge discovery tasks. This package supports preprocessing decision making by visualizing interdependent data quality issues through means of feature construction. The user can define his own application domain specific constructed features that express the quality of a data point such as number of missing values in the point or use nine default features. The outcome can be explored with plot methods and the feature constructed data acquired with get methods.

Depends R (>= 3.2.2)
License GPL-2
LazyData true
Imports caret, DMwR, randomForest, ClustOfVar, reshape2, ggplot2, ggdendro, gridExtra, methods, utils, stats
Suggests testthat, rmarkdown, knitr, preprocomb
Collate '00Utils.R' '01BaseClass.R' '02DefaultFeatures.R'
   '03AnalysisClass.R' '04ControlClass.R' '05ReportingClass.R'
   '06RunClass.R' 'DefaultControl.R'

URL https://github.com/mvattulainen/preproviz
BugReports https://github.com/mvattulainen/preproviz/issues
VignetteBuilder knitr
RoxygenNote 5.0.1
NeedsCompilation no
Author Markus Vattulainen [aut, cre]
Maintainer Markus Vattulainen <markus.vattulainen@gmail.com>
Repository CRAN
Date/Publication 2016-07-09 10:10:07
**AnalysisClass-class**

*An S4 class representing analysis data*

**Description**

An S4 class representing analysis data

**Slots**

- `objectname` (character) Name of the object
- `basetable` (data frame) A data frame containing the original data
- `numericbasetable` (data frame) A data frame containing the original data without class labels.
- `classlabel` (factor) A vector of class labels of the original data
- `constructeddata` (data frame) Constructed data. Feature vectors from computevalue combined as a data frame
- `minmaxconstructeddata` (data frame) Min-max normalized constructed data
- `combineddata` (data frame) Basedata and constructed data combined (note: may include missing values)
combinednumericdata (data frame) Bsedata and constructed data combined without class labels
longformatmixmaxconstructeddata (data frame) Minmaxconstructeddata in long format
distancematrix (matrix) Distance matrix of minmaxconstructeddata
dendogram (dendrogram) Variable clusters (note: not in use)
lofscores (numeric) A vector of LOF scores
cmds (data frame) Classical multidimensional scaling two-dimensional data point computed from
minmaxconstructeddata
variableimportancedata (data frame) Constructed features and their random forest variable im-
portance scores for predicting classlabel
lofsumdata (data frame) mixmax normalized LOF scores of minmaxconstructed data summed
with minmax normalized LOF scores of numerichasedata

Description
An abstract S4 class representing contructed features

Slots
  objectname (character) name of the object
  valuevector (numeric) constructed feature vector
  isvalid (logical) result of object validation
  preimpute (logical) whether valuevector iss computed before missing value imputation

computeValue
  generic function for computing constructed feature vectors

Description
generic function for computing constructed feature vectors

Usage
computeValue(object, dataobject)

Arguments
  object (sub class object inherited from BaseClass)
  dataobject (DataClass)

Value
  (numeric) feature vector
constructfeature \hspace{0.2cm} \textit{constructor function for adding constructed features to the system}

\begin{description} \item[Description] \textit{constructor function for adding constructed features to the system} \item[Usage] \texttt{constructfeature(classname, operation, mode = "all", impute = FALSE)} \item[Arguments] \begin{itemize} \item \texttt{classname} \hspace{0.2cm} \texttt{(character)} name of the inherited class \item \texttt{operation} \hspace{0.2cm} \texttt{(expression)} feature construction operation. The expression is evaluated by \texttt{computevalue} method. \item \texttt{mode} \hspace{0.2cm} \texttt{(character)} Mode of data to be used in construction. Defaults to "all", option "numeric" for numeric data without class labels. \item \texttt{impute} \hspace{0.2cm} \texttt{(logical)} Impute whether construction is done before missing value imputation. Defaults to "FALSE" \end{itemize} \item[Value] \hspace{0.2cm} \texttt{(BaseClass)} A sub class inherited from \texttt{BaseClass} and \texttt{computevalue} method for the class \end{description}

ControlClass-class \hspace{0.2cm} \textit{An S4 class representing setups to be executed}

\begin{description} \item[Description] \textit{An S4 class representing setups to be executed} \item[Slots] \begin{itemize} \item \texttt{setups} A list of \texttt{SetUpClass} objects \end{itemize} \end{description}
Description

DataClass object can be initialized only for a data frame that has a) one class label columns of class 'factor' and b) other columns are of type 'numeric'.

Slots

- name (character) name of the setup object
- basedata (data frame) original data to be visualized
- imputedbase (data frame) missing value in original data imputed with Knn imputation
- numericdata (data frame) numeric columns of original data
- imputednumeric (data frame) imputed numeric columns

defaultParameters

defaultParameters

Description

defaultParameters include nine experimental constructed features (technically, subclasses)

Usage

defaultParameters

Format

An object of class ParameterClass of length 1.
**initializecontrolclassobject**

*constructor function for initializing a ControlClass object*

**Description**

constructor function for initializing a ControlClass object

**Usage**

`initializecontrolclassobject(setups)`

**Arguments**

- `setups` (list) Name of SetUpClass objects

**Value**

(ControlClass) object

---

**initializedataobject**

*constructor function for initializing a DataClass object*

**Description**

constructor function for initializing a DataClass object

**Usage**

`initializedataobject(data)`

**Arguments**

- `data` (data frame)

**Value**

(DataClass)
initializeparameterclassobject

constructor function for initialzing a ParameterClass objects

Description
constructor function for initialzing a ParameterClass objects

Usage
initializeparameterclassobject(parameters)

Arguments
parameters (list) Name of sub classes

Value
( ParameterClass ) object

initializesetupclassobject

constructor function for initializing a SetUpClass object

Description
constructor function for initializing a SetUpClass object

Usage
initializesetupclassobject(objectname, parameterobject, dataobject)

Arguments
objectname (character) Name of the setup
parameterobject (ParameterClass)
dataobject (DataClass)
ParameterClass-class

An S4 class representing selected constructed features

Description

ParameterClass is a class containing a list of sub class objects (i.e. constructed features, inherited from BaseClass). A ParameterClass object in a SetUpClass object defines which constructed features are computed from a DataClass object.

Slots

parameters A list of sub class objects

plotCMDS

generic function for plotting classical multidimensional scaling

Description

generic function for plotting classical multidimensional scaling

Usage

plotCMDS(object)

## S4 method for signature 'ReportClass'
plotCMDS(object)

## S4 method for signature 'RunClass'
plotCMDS(object)

Arguments

object (ReportClass or RunClass)
plotDENSITY

Description

generic function for plotting density estimates of constructed features

Usage

plotDENSITY(object)

# S4 method for signature 'ReportClass'
plotDENSITY(object)

# S4 method for signature 'RunClass'
plotDENSITY(object)

Arguments

object (ReportClass or RunClass)

plotHEATMAP
generic function for plotting heatmap

Description

generic function for plotting heatmap

Usage

plotHEATMAP(object)

# S4 method for signature 'ReportClass'
plotHEATMAP(object)

# S4 method for signature 'RunClass'
plotHEATMAP(object)

Arguments

object (ReportClass or RunClass)
plotLOFSUM

**generic function for plotting lof sum of constructed features**

**Description**

generic function for plotting lof sum of constructed features

**Usage**

plotLOFSUM(object)

## S4 method for signature 'ReportClass'
plotLOFSUM(object)

## S4 method for signature 'RunClass'
plotLOFSUM(object)

**Arguments**

object (ReportClass or RunClass)

---

plotOUTLIERS

**generic function for plotting density of LOF scores**

**Description**

generic function for plotting density of LOF scores

**Usage**

plotOUTLIERS(object)

## S4 method for signature 'ReportClass'
plotOUTLIERS(object)

## S4 method for signature 'RunClass'
plotOUTLIERS(object)

**Arguments**

object (ReportClass or RunClass)
plotVARCLUST  

**generic function for plotting variable clusters**

**Description**

generic function for plotting variable clusters

**Usage**

plotVARCLUST(object)

```r
## S4 method for signature 'ReportClass'
plotVARCLUST(object)

## S4 method for signature 'RunClass'
plotVARCLUST(object)
```

**Arguments**

- object (ReportClass or RunClass)

---

plotVARIMP  

**generic function for plotting variable importance**

**Description**

generic function for plotting variable importance

**Usage**

plotVARIMP(object)

```r
## S4 method for signature 'ReportClass'
plotVARIMP(object)

## S4 method for signature 'RunClass'
plotVARIMP(object)
```

**Arguments**

- object (ReportClass or RunClass)
ReportClass-class

**preproviz**

*the MAIN execution function*

**Description**

for simple exploration preproviz() takes data frame (one factor variable, other variables numeric) as an argument. Two data sets can be compared by providing them as a list. For complex setups a ControlClass object can be passed as an argument. See Vignette for examples. The output can be visualized with PLOT functions.

**Usage**

preproviz(controlobject)

**Arguments**

controlobject  (data frame/list/ControlClass object)

**Value**

(RunClass) object

**Examples**

```r
## result1 <- preproviz(iris)
## plotDENSITY(result1)
##
## iris2 <- iris
## iris2[sample(1:150,30),1] <- NA
## result2 <- preproviz(list(iris, iris2))
## plotVARCLUST(result2)
```

---

**ReportClass-class**  *An S4 class representing visualizations*

**Description**

An S4 class representing visualizations

**Slots**

density  density plot of all constructed features
heatmap  heatmap
cmds  classical multidimensional scaling
variableclusters  hierarchical variable clustering
RunClass-class

- outliers LOF outlier scores
- varimp variable importance
- lofsum sum of LOF scores

---

RunClass-class  
An S4 class representing preproviz output (data and visualizations)

Description

RunClass is an class contain ReportClass and AnalysisClass objects as separate lists. A RunClass object is the output of running the main function preproviz() and can be studied with either get or plot methods.

Value

A RunClass object

Slots

- reports A list of ReportClass objects
- analysis A list of AnalysisClass object

---

SetUpClass-class  
An S4 class representing setups

Description

SetUpClass is an class containing a DataClass object, a ParameterClass object and the name of the SetUpClass object

Slots

- data (DataClass)
- parameters (ParameterClass)
- objectname (character)
Index

*Topic datasets
    defaultParameters, 5

AnalysisClass-class, 2

BaseClass-class, 3

computeValue, 3
constructFeature, 4
ControlClass-class, 4

DataClass-class, 5
defaultParameters, 5

initializecontrolclassobject, 6
initializedataobject, 6
initializeparameterclassobject, 7
initializesetupclassobject, 7

ParameterClass-class, 8
plotCMDs, 8
plotCMDs, ReportClass-method (plotCMDs), 8
plotCMDs, RunClass-method (plotCMDs), 8
plotDENSITY, 9
plotDENSITY, ReportClass-method (plotDENSITY), 9
plotDENSITY, RunClass-method (plotDENSITY), 9
plotHEATMAP, 9
plotHEATMAP, ReportClass-method (plotHEATMAP), 9
plotHEATMAP, RunClass-method (plotHEATMAP), 9
plotLOFSUM, 10
plotLOFSUM, ReportClass-method (plotLOFSUM), 10
plotLOFSUM, RunClass-method (plotLOFSUM), 10
plotOUTLIERS, 10
plotOUTLIERS, ReportClass-method (plotOUTLIERS), 10
plotOUTLIERS, RunClass-method (plotOUTLIERS), 10
plotVARCLUST, 11
plotVARCLUST, ReportClass-method (plotVARCLUST), 11
plotVARCLUST, RunClass-method (plotVARCLUST), 11
plotVARIMP, 11
plotVARIMP, ReportClass-method (plotVARIMP), 11
plotVARIMP, RunClass-method (plotVARIMP), 11
preproviz, 12
ReportClass-class, 12
RunClass-class, 13

SetUpClass-class, 13