Package 'preprosim'

July 26, 2016

Type Package

Title Lightweight Data Quality Simulation for Classification

Version 0.2.0

Date 2016-07-26

Description Data quality simulation can be used to check the robustness of data analysis findings and learn about the impact of data quality contaminations on classification. This package helps to add contaminations (noise, missing values, outliers, low variance, irrelevant features, class swap (inconsistency), class imbalance and decrease in data volume) to data and then evaluate the simulated data sets for classification accuracy. As a lightweight solution simulation runs can be set up with no or minimal up-front effort.

License GPL-2

LazyData TRUE

Imports DMwR, reshape2, ggplot2, methods, stats, caret, doParallel, foreach, e1071

Suggests gbm, preprocomb, preproviz, knitr, rmarkdown

URL https://github.com/mvattulainen/preprosim

BugReports https://github.com/mvattulainen/preprosim/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 2016-07-26 12:14:58

R topics documented:

`changeparam` *Change simulation control parameter object*

Description

Preprosim parameter objects contain eight contaminations: noise, lowvar, misval, irfeature, classswap, classimbalance, volumedecrease and outlier. Each contamination has three sub parameters: cols as columns the contamination is applied to, param as the parameter of the contaminations itself (i.e. intensity of contamination) and order as order in which the parameter is applied to the data.

Usage

```
changeparam(object, contamination, param, value)
```

Arguments

`object`	(preprosimparameter object)
`contamination`	(character) one of the following: noise, lowvar, misval, irfeature, classswap, classimbalance, volumedecrease, outlier
`param`	(character) one of the following: cols, param, order
`value`	(numeric) scalar (for order) or vector (for cols and param) of parameter values

Details

The order of contaminations (cols parameter) must be between 1 and 8, and no two contaminations can have the same order. The contamination parameter (param parameter) must start with 0 (e.g. param="param", value=c(0,0.3))

Value

preprosimparameter class object

Examples

```
pa <- newparam(iris)
pa <- changeparam(pa, "noise", "cols", value=1)
pa <- changeparam(pa, "noise", "param", value=c(0,0.1))
pa <- changeparam(pa, "noise", "order", value=1)
```

`getpreprosimdata` *Get simulation run result data*

Description

Get simulation run result data

Usage

```
getpreprosimdata(object, type = "accuracy", x, z)
```

Arguments

`object`	(preprosimanalysis class object) object
`type`	(character) type of data: accuracy, varimportance, outliers or xz
`x`	(character) x axis contamination
`z`	(character) z axis contamination

Details

contaminations are : noise, lowvar, misval, irfeature, classswap, classimbalance, volumedecrease, outlier

Examples

```
## res <- preprosimrun(iris)
## getpreprosimdata(res, "accuracy")
## getpreprosimdata(res, type="xz", x="misval", z="noise")
```

getpreprosimdf *Get a contaminated data frame*

Description

Get a contaminated data frame

Usage

```
getpreprosimdf(object, paramvector)
```

Arguments

object	(preprosimanalysis class object) object to be plotted
paramvector	(numeric) contamination combinations to be searched for

Examples

```
## res <- preprosimrun(iris)
## df <- preprosimdf(res, c(0,0,0,0,0,0,0,0)) # returns uncontaminated original data set
```

newparam *Create new simulation control parameter object*

Description

Preprosim parameter objects contain eight contaminations: noise, lowvar, misval, irfeature, classswap, classimbalance, volumedecrease and outlier. Each contamination has three sub parameters: cols as columns the contamination is applied to, param as the parameter of the contamination itself (i.e. intensity of contamination) and order as order in which the parameter is applied to the data.

Usage

```
newparam(dataframe, type = "default", x, z)
```

Arguments

dataframe	(data frame) original data to be used in simulations
type	(character) creation type: empty, default or custom, defaults to "default"
x	(character) primary contamination of interest such as "misval"
z	(character) secondary contamination of interest such as "noise"

Details

For argument type: empty creates a preprosimparameter object with empty params (but not empty cols or order). default creates 6561 combinations with all params 0, 0.1, 0.2. custom creates params seq(0, 0.9, by 0.1) for primary (x) and 0., 0.1, 0.2 for secondary (z). The implicit y (not an argument) refers to classification accuracy.

Value

preprosimparameter class object

Examples

```
pa <- newparam(iris)
pa1 <- newparam(iris, "empty")
pa2 <- newparam(iris, "custom", "misval", "noise")
```

---

preprosimanalysis-class

An S4 class representing simulation run results

Description

An S4 class representing simulation run results

Slots

`grid` (data frame) data frame consisting of combinations of preprosimparameters

`data` (list) list of simulated data sets

`output` (numeric) vector of classification accuracies

`variableimportance` (data frame) data frame consisting of variable importance values

`outliers` (numeric) vector of outlier scores

---

preprosimparameter-class

An S4 class representing simulation control parameters

Description

An S4 class representing simulation control parameters

Slots

`noisecol` (numeric)

`noiseparam` (numeric)

`noiseorder` (numeric)

`noisefunction` (character)

`lowvarcol` (numeric)

`lowvarparam` (numeric)

`lowvarorder` (numeric)

`lowvarfunction` (character)

`misvalcol` (numeric)

`misvalparam` (numeric)

`misvalorder` (numeric)

`misvalfunction` (character)

`irfeaturecol` (numeric)

`irfeatureparam` (numeric)

`irfeatureorder` (numeric)

`irfeaturefunction` (character)

`classswapcol` (numeric)

`classswapparam` (numeric)

`classswaporder` (numeric)

`classswapfunction` (character)

`classimbalancecol` (numeric)

`classimbalanceparam` (numeric)

`classimbalanceorder` (numeric)

`classimbalancefunction` (character)

`volumedecreasecol` (numeric)

`volumedecreaseparam` (numeric)

`volumedecreaseorder` (numeric)

`volumedecreasefunction` (character)

`outliercol` (numeric)

`outlierparam` (numeric)

`outlierorder` (numeric)

`outlierfunction` (character)

preprosimplot *Plot simulation run results*

Description

Plot simulation run results

Usage

```
preprosimplot(object, type = "accuracy", x, z)
```

Arguments

object	(preprosimanalysis class object) object to be plotted
type	(character) type of plot: accuracy, varimportance, outliers or xz; defaults to accuracy
x	(character) x axis contamination
z	(character) z axis contamination plotted as panels

Details

contaminations are : noise, lowvar, misval, irfeature, classswap, classimbalance, volumedecrease, outlier

Examples

```
## res <- preprosimrun(iris)
## preprosimplot(res)
## preprosimplot(res, type="xz", x="misval", z="noise")
```

preprosimrun *Run simulation*

Description

Run simulation

Usage

```
preprosimrun(data, param = newparam(data, "default"), seed = 1,
  caretmodel = "gbm", holdoutrounds = 10, cores = 1, verbose = TRUE,
  fitmodels = TRUE)
```

Arguments

data	(data frame) one factor columns for class labels, other columns numeric, no missing values
param	(preprosimparameter object) simulation parameters, defaults to parameters set automatically for data.
seed	(integer) seed to be used for reproducible results, defaults to 1
caretmodel	(character) a model from package Caret, defaults to gbm (gbm must be installed before preprosimrun)
holdoutrounds	(integer) number of holdout rounds, defaults to 10
cores	(integer) number of cores used in parallel processing, defaults to 1
verbose	(boolean) progress information outputted, defaults to TRUE
fitmodels	(boolean) whether classification models are fitted, defaults to TRUE (FALSE: get only the contaminated datasets)

Details

caretmodel must be able to deal with missing values and have in-build variable importance such as rpart and gbm. Note: caret message will be outputted regardless of verbose.

Value

preprosimanalysis class object

Examples

```
res <- preprosimrun(iris, param=newparam(iris, "custom", x="misval", z="noise"), fitmodels=FALSE)
```

Index