Package ‘imbalance’

July 22, 2025

Type Package

Title Preprocessing Algorithms for Imbalanced Datasets

Version 1.0.2.1

Maintainer Ignacio Cordón <nacho.cordon.castillo@gmail.com>

Description Class imbalance usually damages the performance of classifiers. Thus, it is important to treat data before applying a classifier algorithm. This package includes recent resampling algorithms in the literature: (Barua et al. 2014) <doi:10.1109/tkde.2012.232>; (Das et al. 2015) <doi:10.1109/tkde.2014.2324567>, (Zhang et al. 2014) <doi:10.1016/j.inffus.2013.12.003>; (Gao et al. 2014) <doi:10.1016/j.neucom.2014.02.006>; (Almogahed et al. 2014) <doi:10.1007/s00500-014-1484-5>. It also includes an useful interface to perform oversampling.

License GPL (>= 2) | file LICENSE

Encoding UTF-8

LazyData true

BugReports http://github.com/ncordon/imbalance/issues

URL http://github.com/ncordon/imbalance

Depends R (>= 3.3.0)

Imports bnlearn, KernelKnn, ggplot2, utils, stats, mvtnorm, Rcpp, smotefamily, FNN, C50

Suggests testthat, knitr, rmarkdown

RoxygenNote 7.0.2

VignetteBuilder knitr

LinkingTo Rcpp, RcppArmadillo

NeedsCompilation yes

Author Ignacio Cordón [aut, cre],
Salvador García [aut],
Alberto Fernández [aut],
Francisco Herrera [aut]

Repository CRAN

Date/Publication 2020-04-07 06:51:44 UTC

Contents

`banana` *Binary banana dataset*

Description

Dataset containing two attributes as well as a class one, that, if plotted, represent a banana shape

Usage

```
banana

banana_orig
```

Format

At1 First attribute.

At2 Second attribute.

Class Two possible classes: positive (banana shape), negative (surrounding of the banana).

Shape

`banana`: A data frame with 2640 instances, 264 of which belong to positive class, and 3 variables

`banana_orig`: A data frame with 5300 instances, 2376 of which belong to positive class, and 3 variables:

Source

KEEL Repository.

`ecoli1` *Imbalanced binary ecoli protein localization sites*

Description

Imbalanced binary dataset containing protein traits for predicting their cellular localization sites.

Usage

```
ecoli1
```

Format

A data frame with 336 instances, 77 of which belong to positive class, and 8 variables:

Mcg McGeoch's method for signal sequence recognition. Continuous attribute.

Gvh Von Heijne's method for signal sequence recognition. Continuous attribute.

Lip von Heijne's Signal Peptidase II consensus sequence score. Discrete attribute.

Chg Presence of charge on N-terminus of predicted lipoproteins. Discrete attribute.

Aac Score of discriminant analysis of the amino acid content of outer membrane and periplasmic proteins. Continuous attribute.

Alm1 Score of the ALOM membrane spanning region prediction program. Continuous attribute.

Alm2 score of ALOM program after excluding putative cleavable signal regions from the sequence. Continuous attribute.

Class Two possible classes: positive (type im), negative (the rest).

Source

KEEL Repository.

See Also

Original available in UCI ML Repository.

glass0 *Imbalanced binary glass identification*

Description

Imbalanced binary classification dataset containing variables to identify types of glass.

Usage

```
glass0
```

Format

A data frame with 214 instances, 70 of which belong to positve class, and 10 variables:

RI Refractive Index. Continuous attribute.

Na Sodium, weight percent in component. Continuous attribute.

Mg Magnesium, weight percent in component. Continuous attribute.

Al Aluminum, weight percent in component. Continuous attribute.

Si Silicon, weight percent in component. Continuous attribute.

K Potasium, weight percent in component. Continuous attribute.

Ca Calcium, weight percent in component. Continuous attribute.

Ba Barium, weight percent in component. Continuous attribute.

Fe Iron, weight percent in component. Continuous attribute.

Class Two possible glass types: positive (building windows, float processed) and negative (the rest).

Source

KEEL Repository.

See Also

Original available in UCI ML Repository.

`haberman` *Haberman's survival data*

Description

The dataset contains cases from a study that was conducted between 1958 and 1970 at the University of Chicago's Billings Hospital on the survival of patients who had undergone surgery for breast cancer.

Usage

```
haberman
```

Format

A data frame with 306 instances, 81 of which belong to positive class, and 4 variables:

Age Age of patient at time of operation. Discrete attribute.

Year Patient's year of operation. Discrete attribute.

Positive Number of positive axillary nodes detected. Discrete attribute.

Class Two possible survival status: positive(survival rate of less than 5 years), negative (survival rate or more than 5 years).

Source

KEEL Repository.

See Also

Original available in UCI ML Repository.

`imbalance` *imabalance: A package to treat imbalanced datasets*

Description

Focused on binary class datasets, the `imbalance` package provides methods to generate synthetic examples and achieve balance between the minority and majority classes in dataset distributions

Oversampling

Methods to oversample the minority class: `racog`, `wracog`, `rwo`, `pdfos`, `mwmote`

Evaluation

Method to measure imbalance ratio in a given two-class dataset: `imbalanceRatio`.

Method to visually evaluate algorithms: `plotComparison`.

Filtering

Methods to filter oversampled instances `neater`.

`imbalanceRatio` *Compute imbalance ratio of a binary dataset*

Description

Given a two-class dataset, it computes its imbalance ratio as {Size of minority class}/{Size of majority class}

Usage

```
imbalanceRatio(dataset, classAttr = "Class")
```

Arguments

`dataset`	A target `data.frame` to compute its imbalance ratio
`classAttr`	A `character` containing the class name attribute.

Value

A real number in [0,1] representing the imbalance ratio of `dataset`

Examples

```
data(glass0)

imbalanceRatio(glass0, classAttr = "Class")
```

iris0 *Imbalanced binary iris dataset*

Description

Modification of iris dataset. Measurements in centimeters of the variables sepal length and width and petal length and width, respectively, for 50 flowers from each of 3 species of iris. The possible classifications are positive (setosa) and negative (versicolor + virginica).

Usage

```
iris0
```

Format

A data frame with 150 instances, 50 of which belong to positive class, and 5 variables:

SepalLength Measurement of sepal length, in cm. Continuous attribute.

SepalWidth Measurement of sepal width, in cm. Continuous attribute.

PetalLength Measurement of petal length, in cm. Continuous attribute.

PetalWidth Measurement of petal width, in cm. Continuous attribute.

Class Two possible classes: positive (setosa) and negative (versicolor + virginica).

Source

KEEL Repository.

mwmote *Majority weighted minority oversampling technique for imbalance dataset learning*

Description

Modification for SMOTE technique which overcomes some of the problems of the SMOTE technique when there are noisy instances, in which case SMOTE would generate more noisy instances out of them.

Usage

```
mwmote(
  dataset,
  numInstances,
  kNoisy = 5,
  kMajority = 3,
  kMinority,
  threshold = 5,
  cmax = 2,
  cclustering = 3,
  classAttr = "Class"
)
```

Arguments

`dataset` — `data.frame` to treat. All columns, except `classAttr` one, have to be numeric or coercible to numeric.

`numInstances` — Integer. Number of new minority examples to generate.

`kNoisy` — Integer. Parameter of euclidean KNN to detect noisy examples as those whose whole kNoisy-neighbourhood is from the opposite class.

`kMajority` — Integer. Parameter of euclidean KNN to detect majority borderline examples as those who are in any kMajority-neighbourhood of minority instances. Should be a low integer.

`kMinority` — Integer. Parameter of euclidean KNN to detect minority borderline examples as those who are in the KMinority-neighbourhood of majority borderline ones. It should be a large integer. By default if not parameter is fed to the function, $|S^+|/2$ where S^+ is the set of minority examples.

`threshold` — Numeric. A positive real indicating how much we measure tolerance of closeness to the boundary of minority boundary examples. A large integer indicates more margin of distance for a example to be considerated important boundary one.

`cmax` — Numeric. A positive real indicating how much we measure tolerance of closeness to the boundary of minority boundary examples. The larger this number, the more we are valuing boundary examples.

`cclustering` — Numeric. A positive real for tuning the output of an internal clustering. The larger this parameter, the more area focused is going to be the oversampling.

`classAttr` — `character`. Indicates the class attribute from `dataset`. Must exist in it.

Value

A `data.frame` with the same structure as `dataset`, containing the generated synthetic examples.

References

Barua, Sukarna; Islam, Md.M.; Yao, Xin; Murase, Kazuyuki. Mwmote–majority Weighted Minority Oversampling Technique for Imbalanced Data Set Learning. IEEE Transactions on Knowledge and Data Engineering 26 (2014), Nr. 2, p. 405–425

Examples

```
data(iris0)

# Generates new minority examples
newSamples <- mwmote(iris0, numInstances = 100, classAttr = "Class")
```

---

neater *Fitering of oversampled data based on non-cooperative game theory*

Description

Filters oversampled examples from a binary class `dataset` using game theory to find out if keeping an example is worthy enough.

Usage

```
neater(
  dataset,
  newSamples,
  k = 3,
  iterations = 100,
  smoothFactor = 1,
  classAttr = "Class"
)
```

Arguments

`dataset`	The original `data.frame`. All columns, except `classAttr` one, have to be numeric or coercible to numeric.
`newSamples`	A `data.frame` containing the samples to be filtered. Must have the same structure as `dataset`.
`k`	Integer. Number of nearest neighbours to use in KNN algorithm to rule out samples. By default, 3.
`iterations`	Integer. Number of iterations for the algorithm. By default, 100.
`smoothFactor`	A positive `numeric`. By default, 1.
`classAttr`	`character`. Indicates the class attribute from `dataset` and `newSamples`. Must exist in them.

Details

Uses game theory and Nash equilibriums to calculate the minority examples probability of trully belonging to the minority class. It discards examples which at the final stage of the algorithm have more probability of being a majority example than a minority one.

Value

Filtered samples as a data.frame with same structure as newSamples.

References

Almogahed, B.A.; Kakadiaris, I.A. Neater: Filtering of Over-Sampled Data Using Non-Cooperative Game Theory. Soft Computing 19 (2014), Nr. 11, p. 3301–3322.

Examples

```
data(iris0)

newSamples <- smotefamily::SMOTE(iris0[,-5], iris0[,5])$syn_data
# SMOTE overrides Class attr turning it into class
# and dataset must have same class attribute as newSamples
names(newSamples) <- c(names(newSamples)[-5], "Class")

neater(iris0, newSamples, k = 5, iterations = 100,
       smoothFactor = 1, classAttr = "Class")
```

newthyroid1 *Imbalanced binary thyroid gland data*

Description

Data to predict patient's hyperthyroidism.

Usage

```
newthyroid1
```

Format

A data frame with 215 instances, 35 of which belong to positive class, and 6 variables:

T3resin T3-resin uptake test, percentage. Discrete attribute.

Thyroxin Total Serum thyroxin as measured by the isotopic displacement method. Continuous attribute.

Triiodothyronine Total serum triiodothyronine as measured by radioimmuno assay. Continuous attribute.

Thyroidstimulating Basal thyroid-stimulating hormone (TSH) as measured by radioimmuno assay. Continuous attribute.

TSH_value Maximal absolute difference of TSH value after injection of 200 micro grams of thyrotropin-releasing hormone as compared to the basal value. Continuous attribute.

Class Two possible classes: positive as hyperthyroidism, negative as non hyperthyroidism.

Source

KEEL Repository.

See Also

Original available in UCI ML Repository.

`oversample` *Wrapper that encapsulates a collection of algorithms to perform a class balancing preprocessing task for binary class datasets*

Description

Wrapper that encapsulates a collection of algorithms to perform a class balancing preprocessing task for binary class datasets

Usage

```
oversample(
  dataset,
  ratio = NA,
 method = c("RACOG", "wRACOG", "PDFOS", "RWO", "ADASYN", "ANSMOTE", "SMOTE", "MWMOTE",
    "BLSMOTE", "DBSMOTE", "SLMOTE", "RSLSMOTE"),
  filtering = FALSE,
  classAttr = "Class",
  wrapper = c("KNN", "C5.0"),
  ...
)
```

Arguments

`dataset`	A binary class `data.frame` to balance.
`ratio`	Number between 0 and 1 indicating the desired ratio between minority examples and majority ones, that is, the quotient size of minority class/size of majority class. There are methods, such as `ADASYN` or `wRACOG` to which this parameter does not apply.
`method`	A `character` corresponding to method to apply. Possible methods are: RACOG, wRACOG, PDFOS, RWO, ADASYN, ANSMOTE, SMOTE, MWMOTE, BLSMOTE, DBSMOTE, SLMOTE, RSLSMOTE
`filtering`	Logical (TRUE or FALSE) indicating wheter to apply filtering of oversampled instances with `neater` algorithm.
`classAttr`	`character`. Indicates the class attribute from `dataset`. Must exist in it.
`wrapper`	A `character` corresponding to wrapper to apply if selected method is `wracog`. Possibilities are: `"C5.0"` and `"KNN"`.
`...`	Further arguments to apply in selected method

Value

A balanced `data.frame` with same structure as `dataset`, containing both original instances and new ones

Examples

```
data(glass0)

# Oversample glass0 to get an imbalance ratio of 0.8
imbalanceRatio(glass0)
# 0.4861111
newDataset <- oversample(glass0, ratio = 0.8, method = "MWMOTE")
imbalanceRatio(newDataset)
newDataset <- oversample(glass0, method = "ADASYN")
newDataset <- oversample(glass0, ratio = 0.8, method = "SMOTE")
```

`pdfos` *Probability density function estimation based oversampling*

Description

Generates synthetic minority examples for a numerical dataset approximating a Gaussian multivariate distribution which best fits the minority data.

Usage

```
pdfos(dataset, numInstances, classAttr = "Class")
```

Arguments

`dataset` `data.frame` to treat. All columns, except `classAttr` one, have to be numeric or coercible to numeric.

`numInstances` Integer. Number of new minority examples to generate.

`classAttr` `character`. Indicates the class attribute from `dataset`. Must exist in it.

Details

To generate the synthetic data, it approximates a normal distribution with mean a given example belonging to the minority class, and whose variance is the minority class variance multiplied by a constant; that constant is computed so that it minimizes the mean integrated squared error of a Gaussian multivariate kernel function.

Value

A `data.frame` with the same structure as `dataset`, containing the generated synthetic examples.

References

Gao, Ming; Hong, Xia; Chen, Sheng; Harris, Chris J.; Khalaf, Emad. Pdfos: Pdf Estimation Based Oversampling for Imbalanced Two-Class Problems. Neurocomputing 138 (2014), p. 248–259

Silverman, B. W. Density Estimation for Statistics and Data Analysis. Chapman & Hall, 1986. – ISBN 0412246201

Examples

```
data(iris0)

newSamples <- pdfos(iris0, numInstances = 100, classAttr = "Class")
```

---

plotComparison *Plots comparison between the original and the new balanced dataset.*

Description

It plots a grid of one to one variable comparison, placing the former dataset graphics next to the balanced one, for each pair of attributes.

Usage

```
plotComparison(dataset, anotherDataset, attrs, cols = 2, classAttr = "Class")
```

Arguments

`dataset`	A `data.frame`. The former imbalanced dataset.
`anotherDataset`	A `data.frame`. The balanced dataset. `dataset` and `anotherDataset` must have the same columns.
`attrs`	Vector of `character`. Attributes to compare. The function generates each posible combination of attributes to build the comparison.
`cols`	Integer. It indicates the number of columns of resulting grid. Must be an even number. By default, 2.
`classAttr`	`character`. Indicates the class attribute from `dataset`. Must exist in it.

Value

Plot of 2D comparison between the variables.

Examples

```
data(iris0)
set.seed(12345)

rwoSamples <- rwo(iris0, numInstances = 100)
rwoBalanced <- rbind(iris0, rwoSamples)
plotComparison(iris0, rwoBalanced, names(iris0), cols = 2, classAttr = "Class")
```

racog — *Rapidly converging Gibbs algorithm.*

Description

Allows you to treat imbalanced discrete numeric datasets by generating synthetic minority examples, approximating their probability distribution.

Usage

```
racog(dataset, numInstances, burnin = 100, lag = 20, classAttr = "Class")
```

Arguments

`dataset` `data.frame` to treat. All columns, except `classAttr` one, have to be numeric or coercible to numeric.

`numInstances` Integer. Number of new minority examples to generate.

`burnin` Integer. It determines how many examples generated for a given one are going to be discarded firstly. By default, 100.

`lag` Integer. Number of iterations between new generated example for a minority one. By default, 20.

`classAttr` `character`. Indicates the class attribute from `dataset`. Must exist in it.

Details

Approximates minority distribution using Gibbs Sampler. Dataset must be discretized and numeric. In each iteration, it builds a new sample using a Markov chain. It discards first `burnin` iterations, and from then on, each `lag` iterations, it validates the example as a new minority example. It generates $d(iterations - burnin)/lag$ where d is minority examples number.

Value

A `data.frame` with the same structure as `dataset`, containing the generated synthetic examples.

References

Das, Barnan; Krishnan, Narayanan C.; Cook, Diane J. Racog and Wracog: Two Probabilistic Oversampling Techniques. IEEE Transactions on Knowledge and Data Engineering 27(2015), Nr. 1, p. 222–234.

Examples

```
data(iris0)

# Generates new minority examples

newSamples <- racog(iris0, numInstances = 40, burnin = 20, lag = 10,
                    classAttr = "Class")

newSamples <- racog(iris0, numInstances = 100)
```

`rwo` *Random walk oversampling*

Description

Generates synthetic minority examples for a dataset trying to preserve the variance and mean of the minority class. Works on every type of dataset.

Usage

```
rwo(dataset, numInstances, classAttr = "Class")
```

Arguments

`dataset`	`data.frame` to treat. All columns, except `classAttr` one, have to be numeric or coercible to numeric.
`numInstances`	Integer. Number of new minority examples to generate.
`classAttr`	`character`. Indicates the class attribute from `dataset`. Must exist in it.

Details

Generates `numInstances` new minority examples for `dataset`, adding to the each numeric column of the j-th example its variance scalated by the inverse of the number of minority examples and a factor following a $N(0, 1)$ distribution which depends on the example. When the column is nominal, it uses a roulette scheme.

Value

A `data.frame` with the same structure as `dataset`, containing the generated synthetic examples.

References

Zhang, Huaxiang; Li, Mingfang. Rwo-Sampling: A Random Walk Over-Sampling Approach To Imbalanced Data Classification. Information Fusion 20 (2014), p. 99–116.

Examples

```
data(iris0)

newSamples <- rwo(iris0, numInstances = 100, classAttr = "Class")
```

`trainWrapper` *Generic methods to train classifiers*

Description

Generic methods to train classifiers

Usage

```
trainWrapper(wrapper, train, trainClass, ...)
```

Arguments

`wrapper`	the wrapper instance
`train`	`data.frame` of the train dataset without the class column
`trainClass`	a vector containing the class column for `train`
`...`	further arguments for `wrapper`

Value

A model which is `predict` callable.

See Also

`predict`

Examples

```
myWrapper <- structure(list(), class="C50Wrapper")
trainWrapper.C50Wrapper <- function(wrapper, train, trainClass){
  C50::C5.0(train, trainClass)
}
```

`wisconsin` *Imbalanced binary breast cancer Wisconsin dataset*

Description

Binary class dataset containing traits about patients with cancer. Original dataset was obtained from the University of Wisconsin Hospitals, Madison from Dr. William H. Wolberg.

Usage

```
wisconsin
```

Format

A data frame with 683 instances, 239 of which belong to positive class, and 10 variables:

ClumpThickness Discrete attribute.

CellSize Discrete attribute.

CellShape Discrete attribute.

MarginalAdhesion Discrete attribute.

EpithelialSize Discrete attribute.

BareNuclei Discrete attribute.

BlandChromatin Disrete attribute.

NormalNucleoli Discrete attribute.

Mitoses Discrete attribute.

Class Two possible classes: positive (cancer) and negative (not cancer).

Source

KEEL Repository.

See Also

Original available in UCI ML Repository.

wracog *Wrapper for rapidly converging Gibbs algorithm.*

Description

Generates synthetic minority examples by approximating their probability distribution until sensitivity of `wrapper` over `validation` cannot be further improved. Works only on discrete numeric datasets.

Usage

```
wracog(
  train,
  validation,
  wrapper,
  slideWin = 10,
  threshold = 0.02,
  classAttr = "Class",
  ...
)
```

Arguments

`train`	`data.frame`. A initial dataset to generate first model. All columns, except `classAttr` one, have to be numeric or coercible to numeric.
`validation`	`data.frame`. A dataset to compare results of consecutive classifiers. Must have the same structure of `train`.
`wrapper`	An `S3` object. There must exist a method `trainWrapper` implemented for the class of the object, and a `predict` method implemented for the class of the model returned by `trainWrapper`. Alternatively, it can the name of one of the wrappers distributed with the package, `"KNN"` or `"C5.0"`.
`slideWin`	Number of last sensitivities to take into account to meet the stopping criteria. By default, 10.
`threshold`	Threshold that the last `slideWin` sensitivities mean should reach. By default, 0.02.
`classAttr`	`character`. Indicates the class attribute from `train` and `validation`. Must exist in them.
`...`	further arguments for `wrapper`.

Details

Until the last `slideWin` executions of `wrapper` over `validation` dataset reach a mean sensitivity lower than `threshold`, the algorithm keeps generating samples using Gibbs Sampler, and adding misclassified samples with respect to a model generated by a former train, to the train dataset. Initial model is built on initial `train`.

Value

A `data.frame` with the same structure as `train`, containing the generated synthetic examples.

References

Das, Barnan; Krishnan, Narayanan C.; Cook, Diane J. Racog and Wracog: Two Probabilistic Oversampling Techniques. IEEE Transactions on Knowledge and Data Engineering 27(2015), Nr. 1, p. 222–234.

Examples

```
data(haberman)

# Create train and validation partitions of haberman
trainFold <- sample(1:nrow(haberman), nrow(haberman)/2, FALSE)
trainSet <- haberman[trainFold, ]
validationSet <- haberman[-trainFold, ]

# Defines our own wrapper with a C5.0 tree
myWrapper <- structure(list(), class="TestWrapper")
trainWrapper.TestWrapper <- function(wrapper, train, trainClass){
  C50::C5.0(train, trainClass)
}

# Execute wRACOG with our own wrapper
newSamples <- wracog(trainSet, validationSet, myWrapper,
                     classAttr = "Class")


# Execute wRACOG with predifined wrappers for "KNN" or "C5.0"
KNNSamples <- wracog(trainSet, validationSet, "KNN")
C50Samples <- wracog(trainSet, validationSet, "C5.0")
```

---

yeast4 *Imbalanced binary yeast protein localization sites*

Description

Imbalanced binary dataset containing protein traits for predicting their cellular localization sites.

Usage

```
yeast4
```

Format

A data frame with 1484 instances, 51 of which belong to positive class, and 9 variables:

Mcg McGeoch's method for signal sequence recognition. Continuous attribute.

Gvh Von Heijne's method for signal sequence recognition. Continuous attribute.

Alm Score of the ALOM membrane spanning region prediction program. Continuous attribute.

Mit Score of discriminant analysis of the amino acid content of the N-terminal region (20 residues long) of mitochondrial and non-mitochondrial proteins. Continuous attribute.

Erl Presence of "HDEL" substring (thought to act as a signal for retention in the endoplasmic reticulum lumen). Binary attribute. Discrete attribute.

Pox Peroxisomal targeting signal in the C-terminus. Continuous attribute.

Vac Score of discriminant analysis of the amino acid content of vacuolar and extracellular proteins. Continuous attribute.

Nuc Score of discriminant analysis of nuclear localization signals of nuclear and non-nuclear proteins. Continuous attribute.

Class Two possible classes: positive (membrane protein, uncleaved signal), negative (rest of localizations).

Source

KEEL Repository.

See Also

Original available in UCI ML Repository.

Index