Package 'TUBE'

January 8, 2021

Title Bridging Cost-sensitive and Neyman-Pearson Paradigms for Asymmetric Binary Classification

Version 0.1.0

Description To bridge cost-sensitive and Neyman-Pearson paradigms for asymmetric binary classification, we have developed two algorithms, TUBEc and TUBE (estimating the Type I error Upper Bound of a cost-sensitive classifiEr). This package contains functions of TUBE-assisted and TUBEc-assisted cost-sensitive classification methods for selecting the type I error cost so that the resulting cost-sensitive classifier has its population type I error under a target upper bound with high probability.

License GPL-3

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1

URL https://arxiv.org/abs/2012.14951

Imports MASS,
mvtnorm,
stats,
glmnet,
randomForest,
xgboost,
naivebayes

R topics documented:

classify_base *Base classification algorithm*

Description

Base classification algorithm

Usage

```
classify_base(xtrain, ytrain, xnew, method, ...)
```

Arguments

xtrain	A matrix with columns representing features and rows representing observations.
ytrain	A binary vector specifying the corresponding labels of the observations.
xnew	A matrix specifying new observations whose class labels need to be predicted.
method	A character specifying the base classification method to be used. Should be one of "LR" (logistic regression), "penLR" (penalized logistic regression) "RF" (random forest), "GB" (gradient boosting), and "NB" (naive Bayes).
...	Additional arguments parsed to the algorithms.

Value

A vector of classification scores for xnew.

gen_data *Generate test and example data*

Description

Generate test and example data

Usage

```
gen_data(model, seed = NULL, n, d, pi = 0.5, nleave = 0)
```

Arguments

model	A character specifying the model used to generate data. Should be one of "gaussian", "t", or "mixture".
seed	An integer specifying the seed (optional).
n	An integer specifying the sample size.
d	An intger specifying the feature number/dimension.
pi	A numeric specifying the proportion of class 0 sample. Defaults to 0.5.
nleave	An integer specifying the size of left-out sample. For debug only.

Value

A named list of generated data. It has two elements: x and y.

stratification *Stratify training data*

Description

Stratify training data

Usage

```
stratification(xtrain, ytrain, wcost)
```

Arguments

xtrain	A matrix with columns representing features and rows representing observations.
ytrain	A binary vector specifying the corresponding labels of the observations.
wcost	A numeric specifying the type I error cost used for stratification.

Value

A named list specifying the stratified data. It has two elements: x and y.

TUBE *TUBE-assisted cost-sensitive classification*

Description

TUBE-assisted cost-sensitive classification

Usage

```
TUBE(
  data,
  classify_fun,
  alpha = 0.05,
  delta = 0.1,
  t_cs = 0.5,
  nleave = NULL,
  cost = seq(0.1, 0.99, 0.01),
  B1 = 50,
  B2 = 200
)
```

Arguments

data	A named list specifying the training data. It should have two elements: data$x should be a matrix with columns representing features and rows representing observations. data$y should be a binary vector specifying the corresponding labels of the observations.
classify_fun	A function specifying the cost-sensitive classification algorithm.
alpha	A numeric specifying the target type I error (between 0 and 1). Defaults to 0.05.
delta	A numeric specifying the target violation rate (between 0 and 1). Defaults to 0.1.
t_cs	A numeric specifying the threshold applied to classification scores. Defaults to 0.5.
nleave	A integer specifying the sample size of left-out class 0 data. Should be smaller than the number of class 0 cases in data.
cost	A numeric vector specifying the candidate type I error costs. Should be ordered increasingly. Defaults to seq(0.1,0.99,0.01).
B1	An integer specifying the number of random data splitting. Defaults to 50.
B2	An integer specifying the number of bootstrap samples. Defaults to 200.

Value

A list with two elements. c0 gives the selected type I error cost; yhat gives the predicted class labels for data$x given c0.

Author(s)

Wei Vivian Li, <vivian.li@rutgers.edu>

Xin Tong, <xtong001@gmail.com>>

Jingyi Jessica Li, <jli@stat.ucla.edu>

References

Li WV, Tong X, Li JJ. Bridging Cost-sensitive and Neyman-Pearson Paradigms for Asymmetric Binary Classification. arXiv preprint arXiv:2012.14951. 2020 Dec 29. https://arxiv.org/abs/2012.14951

TUBEc *TUBEc-assisted cost-sensitive classification*

Description

TUBEc-assisted cost-sensitive classification

Usage

```
TUBEc(
  data,
  classify_fun,
  alpha = 0.05,
  delta = 0.1,
  t_cs = 0.5,
  nleave = NULL,
  cost = seq(0.1, 0.99, 0.01),
  B = 200
)
```

Arguments

`data`	A named list specifying the training data. It should have two elements: `data$x` should be a matrix with columns representing features and rows representing observations. `data$y` should be a binary vector specifying the corresponding labels of the observations.
`classify_fun`	A function specifying the cost-sensitive classification algorithm.
`alpha`	A numeric specifying the target type I error (between 0 and 1). Defaults to 0.05.
`delta`	A numeric specifying the target violation rate (between 0 and 1). Defaults to 0.1.
`t_cs`	A numeric specifying the threshold applied to classification scores. Defaults to 0.5.
`nleave`	A integer specifying the sample size of left-out class 0 data. Should be smaller than the number of class 0 cases in `data`.
`cost`	A numeric vector specifying the candidate type I error costs. Should be ordered increasingly. Defaults to `seq(0.1,0.99,0.01)`.
`B`	An integer specifying the number of bootstrap samples. Defaults to 200.

Value

A list with two elements. `c0` gives the selected type I error cost; `yhat` gives the predicted class labels for `data$x` given `c0`.

Author(s)

Wei Vivian Li, <vivian.li@rutgers.edu>

Xin Tong, <xtong001@gmail.com>>

Jingyi Jessica Li, <jli@stat.ucla.edu>

References

Li WV, Tong X, Li JJ. Bridging Cost-sensitive and Neyman-Pearson Paradigms for Asymmetric Binary Classification. arXiv preprint arXiv:2012.14951. 2020 Dec 29. https://arxiv.org/abs/2012.14951

Index