Package ‘influenceAUC’
February 19, 2020

Type Package
Title Identify Influential Observations in Binary Classification
Version 0.1.1
Maintainer Bo-Shiang Ke <naivete0907@gmail.com>
Description Ke, B. S., Chi- ang, A. J., & Chang, Y. C. I. (2018) <doi:10.1080/10543406.2017.1377728> provide two theoretical methods (influence function and local influence) based on the area under the receiver operating characteristic curve (AUC) to quantify the numerical impact of each observation to the overall AUC. Alternative graphical tools, cumulative lift charts, are proposed to reveal the existences and approximate locations of those influential observations through data visualization.
License GPL-3
BugReports https://github.com/BoShiangKe/InfluenceAUC/issues
Encoding UTF-8
LazyData true
RoxygenNote 7.0.2
Imports dplyr, geigen, ggplot2, ggrepel, methods, ROCR
NeedsCompilation no
Author Bo-Shiang Ke [cre, aut, cph],
Yuan-chin Ivan Chang [aut],
Wen-Ting Wang [aut]
Repository CRAN
Date/Publication 2020-02-19 05:40:03 UTC

R topics documented:

IAUC ........................................... 2
ICLC ......................................... 4
LAUC ....................................... 5
plot.IAUC ................................... 7
plot.ICLC ................................... 7
plot.LAUC .................................. 8
Description

Provide two sample versions (DEIF and SIF) of influence function on the AUC.

Usage

IAUC(score, binary, threshold = 0.5, hypothesis = FALSE, testdiff = 0.5, alpha = 0.05, name = NULL)

Arguments

- **score**: A vector containing the predictions (continuous scores) assigned by classifiers; Must be numeric.
- **binary**: A vector containing the true class labels 1: positive and 0: negative. Must have the same dimensions as 'score.'
- **threshold**: A numeric value determining the threshold to distinguish influential observations from normal ones; Must lie between 0 and 1; Defaults to 0.5.
- **hypothesis**: Logical which controls the evaluation of SIF under asymptotic distribution.
- **testdiff**: A numeric value determining the difference in the hypothesis testing; Must lie between 0 and 1; Defaults to 0.5.
- **alpha**: A numeric value determining the significance level in the hypothesis testing; Must lie between 0 and 1; Defaults to 0.05.
- **name**: A vector comprising the appellations for observations; Must have the same dimensions as 'score'.

Details

Apply two sample versions of influence functions on AUC:

- deleted empirical influence function (DEIF)
- sample influence function (SIF)
The concept of influence function focuses on the deletion diagnostics; nevertheless, such techniques may face masking effect due to multiple influential observations. To thoroughly investigate the potential cases in binary classification, we suggest end-users to apply **ICLC** and **LAUC** as well. For a complete discussion of these functions, please see the reference.

**Value**

A list of objects including (1) ‘output’: a list of results with ‘AUC’ (numeric), ‘SIF’ (a list of dataframes) and ‘DEIF’ (a list of dataframes); (2) ‘rdata’: a dataframe of essential results for visualization; (3) ‘threshold’: a used numeric value to distinguish influential observations from normal ones; (4) ‘test_output’: a list of dataframes for hypothesis testing result; (5) ‘test_data’: a dataframe of essential results in hypothesis testing for visualization; (6) ‘testdiff’: a used numeric value to determine the difference in the hypothesis testing; (7) ‘alpha’: a used numeric value to determine the significance level.

**Author(s)**

Bo-Shiang Ke and Yuan-chin Ivan Chang

**References**


**See Also**

ICLC, LAUC

**Examples**

```r
library(ROCR)
data("ROCR.simple")
# print out IAUC results directly
IAUC(ROCR.simple$predictions,ROCR.simple$labels,hypothesis = "True")

data(mtcars)
glmfit <- glm(vs ~ wt + disp, family = binomial, data = mtcars)
prob <- as.vector(predict(glmfit, newdata = mtcars,type = "response"))
output <- IAUC(prob, mtcars$vs, threshold = 0.3, testdiff = 0.3,
               hypothesis = TRUE, name = rownames(mtcars))
# Show results
print(output)
# Visualize results
plot(output)
```
ICLC

**Cumulative Lift Charts**

**Description**

Show the existence and approximate locations of influential observations in binary classification through modified cumulative lift charts.

**Usage**

ICLC(score, binary, prop = 0.2)

**Arguments**

- **score**: A vector containing the predictions (continuous scores) assigned by classifiers; Must be numeric.
- **binary**: A vector containing the true class labels 1: positive and 0: negative. Must have the same dimensions as 'score.'
- **prop**: A numeric value determining the proportion; Must lie between 0 and 1; Defaults to 0.2.

**Details**

There are two types of influential cases in binary classification:

- positive cases with relatively lower scores - negative cumulative lift chart (NCLC)
- negative cases with relatively higher scores - positive cumulative lift chart (PCLC)

Each cumulative lift chart (PCLC or NCLC) identifies one type of influential observations and mark with red dotted lines. Based on the characteristics of two types of influential cases, identifying them require to search the highest and lowest proportions of 'score.'

Graphical approaches only reveal the existence and approximate locations of influential observations; it would be better to include some quantities to measure their impacts to the interested parameter. To fully investigate the potential observation in binary classification, we suggest end-users to apply two quantification methods IAUC and LAUC as well. For a complete discussion of these functions, please see the reference.

**Value**

A list of ggplot2 objects

**Author(s)**

Bo-Shiang Ke and Yuan-chin Ivan Chang
LAUC

References


See Also

IAUC, LAUC

Examples

```r
library(ROCR)
data("ROCR.simple")
output <- ICLC(ROCR.simple$predictions,ROCR.simple$labels)
plot(output)
# Customize a text size for NCLC
library(ggplot2)
output$NCLC + theme(text = element_text(size = 20))
data(mtcars)
glmfit <- glm(vs ~ wt + disp, family = binomial, data = mtcars)
prob <- as.vector(predict(glmfit, newdata = mtcars, type = "response"))
plot(ICLC(prob, mtcars$vs, 0.5))
```

Description

Apply local influence approaches in terms of slope and curvature on the AUC to quantify the impacts of all observations simultaneously.

Usage

```r
LAUC(score, binary, threshold = 0.2, name = NULL)
```

Arguments

- `score`: A vector containing the predictions (continuous scores) assigned by classifiers; Must be numeric.
- `binary`: A vector containing the true class labels 1: positive and 0: negative. Must have the same dimensions as `score`.
- `threshold`: A numeric value determining the threshold to distinguish influential observations from normal ones; Must lie between 0 and 1; Defaults to 0.2.
- `name`: A vector comprising the appellations for observations; Must have the same dimensions as `score`.
Details

The influence functions on the AUC focus on the deletion diagnostics; however, such approaches may encounter the masking effect. Rather than dealing with single observations once at a time, local influence methods address this issue by finding the weighted direction of all observations accompanied by the greatest (magnitude) slope and curvature. From the explicit formula based on the slope, local influence methods may face the imbalanced data effect. To thoroughly investigate the potential observation in binary classification, we suggest end-users to apply ICLC and IAUC as well. For a complete discussion of these functions, please see the reference.

Value

A list of objects including (1) 'output': a list of results with 'AUC' (numeric), 'Slope' (a list of dataframes) and 'Curvature' (a list of dataframes); (2) 'rdata': a dataframe of essential results for visualization (3) 'threshold': a used numeric value to distinguish influential observations from normal ones.

Author(s)

Bo-Shiang Ke and Yuan-chin Ivan Chang

References


See Also

ICLC, IAUC

Examples

library(ROCR)
data("ROCR.simple")
# print out LAUC results directly
LAUC(ROCR.simple$predictions,ROCR.simple$labels)
data(mtcars)
glmfit <- glm(vs ~ wt + disp, family = binomial, data = mtcars)
prob <- as.vector(predict(glmfit, newdata = mtcars, type = "response"))
output <- LAUC(prob, mtcars$vs, name = rownames(mtcars))
# Show results
print(output)
# Visualize results
plot(output)
### plot.IAUC

**Visualize IAUC result**

#### Description
Visualize IAUC output sequentially

#### Usage
```r
## S3 method for class 'IAUC'
plot(x, ...)
```

#### Arguments
- `x` An IAUC class object for `plot` method
- `...` Not used directly

#### See Also
- `IAUC`

#### Examples
```r
library(ROCR)
data("ROCR.simple")
output <- IAUC(ROCR.simple$predictions, ROCR.simple$labels)
plot(output)
```

### plot.ICLC

**Visualize ICLC results**

#### Description
Visualize ggplot2 objects in ICLC sequentially

#### Usage
```r
## S3 method for class 'ICLC'
plot(x, ...)
```

#### Arguments
- `x` An ICLC class object
- `...` Not used directly

#### Examples
```r
```
See Also

ICLC

Examples

library(ROCR)
data("ROCR.simple")
Coutput <- ICLC(ROCR.simple$predictions, ROCR.simple$labels)
plot(Coutput)

plot.LAUC

Visualize LAUC results

Description

Visualize LAUC output sequentially

Usage

## S3 method for class 'LAUC'
plot(x, ...)

Arguments

x

An LAUC class object for ‘plot’ method

...

Not used directly

See Also

LAUC

Examples

library(ROCR)
data("ROCR.simple")
Loutput <- LAUC(ROCR.simple$predictions, ROCR.simple$labels)
plot(Loutput)
print.IAUC

Show IAUC results

Description
Print IAUC output in detail

Usage

## S3 method for class 'IAUC'
print(x, ...)

Arguments

x An IAUC class object for `print` method
... Not used directly

See Also

IAUC

Examples

library(ROCR)
data("ROCR.simple")
Ioutput <- IAUC(ROCR.simple$predictions, ROCR.simple$labels)
print(Ioutput)

print.LAUC

Show LAUC results

Description
Print LAUC output in detail

Usage

## S3 method for class 'LAUC'
print(x, ...)

Arguments

x An LAUC class object for `print` method
... Not used directly
See Also

LAUC

Examples

library(ROCR)
data("ROCR.simple")
Loutput <- LAUC(ROCR.simple$predictions, ROCR.simple$labels)
print(Loutput)
Index

IAUC, 2, 4–7, 9
ICLC, 3, 4, 6, 8
LAUC, 3–5, 8, 10

plot.IAUC, 7
plot.ICLC, 7
plot.LAUC, 8
print.IAUC, 9
print.LAUC, 9