Package ‘auRoc’

April 4, 2020

**Title** Various Methods to Estimate the AUC

**Version** 0.2-1

**Author** Dai Feng [aut, cre], Damjan Manevski [auc], Maja Pohar Perme [auc]

**Description** Estimate the AUC using a variety of methods as follows:
1. frequentist nonparametric methods based on the Mann-Whitney statistic or kernel methods.
2. frequentist parametric methods using the likelihood ratio test based on higher-order asymptotic results, the signed log-likelihood ratio test, the Wald test, or the approximate "t" solution to the Behrens-Fisher problem.
3. Bayesian parametric MCMC methods.

**Maintainer** Dai Feng <daifeng.stat@gmail.com>

**Depends** R (>= 3.0.2), rjags (>= 3-11), ProbYX(>= 1.1)

**Imports** coda(>= 0.16-1), MBESS(>= 3.3.3)

**License** GPL

**NeedsCompilation** no

**LazyData** true

**Repository** CRAN

**Date/Publication** 2020-04-04 00:30:08 UTC

**R topics documented:**

- auc.nonpara.kernel ................................................. 2
- auc.nonpara.mw .................................................. 3
- auc para.bayes ..................................................... 5
- auc para.frequentist ............................................. 6
- petBrainGlioma ................................................... 8

Index 9
**Description**

Obtain the point estimate and the confidence interval of the AUC using kernel methods.

**Usage**

```r
auc.nonpara.kernel(x, y, conf.level=0.95,
                   integration=c("FALSE", "TRUE"),
                   bw=c("nrd0", "sj"), nint=512,
                   method=c("mw", "jackknife", "bootstrapP", "bootstrapBCa"),
                   nboot)
```

**Arguments**

- `x`: a vector of observations from class P.
- `y`: a vector of observations from class N.
- `conf.level`: confidence level of the interval. The default is 0.95.
- `integration`: a logical value. If its value is `FALSE` then the results are based on the kernel estimates of the CDF; otherwise the PDF. The default value is `FALSE`.
- `nint`: the number of equally spaced points at which the density is to be estimated. The default is 512.
- `method`: a method used to construct the CI. `mw` uses the sd based on the Mann-Whitney statistic; `jackknife` uses the jackknife method; `bootstrapP` uses the bootstrap with percentile CI; `bootstrapBCa` uses bootstrap with bias-corrected and accelerated CI. The default is `mw`. It can be abbreviated.
- `nboot`: number of bootstrap iterations.

**Details**

The AUC essentially depends on the CDFs of two classes N and P. We could use kernel smoothing methods to obtain the CDFs. The methods implemented in this function construct the CI based on two different strategies: the first uses kernel smoothing to estimate the PDFs and then the CDFs; and the second starts from the estimate of the CDFs directly. Gaussian kernel is used.

**Value**

Point estimate and lower and upper bounds of the CI of the AUC.
Note

The observations from class P tend to have larger values than that from class N.

Author(s)

Dai Feng

References


Examples

```r
#Example 1
data(petBrainGlioma)
y <- subset(petBrainGlioma, grade==1, select="FDG", drop=TRUE)
x <- subset(petBrainGlioma, grade==2, select="FDG", drop=TRUE)
AUC <- auc.nonpara.kernel(x, y)

## Not run:
#Example 2
data(petBrainGlioma)
y <- subset(petBrainGlioma, grade==1, select="ACE", drop=TRUE)
x <- subset(petBrainGlioma, grade==2, select="ACE", drop=TRUE)
AUC <- auc.nonpara.kernel(x, y, integration="TRUE",
                        bw="sj", method="bootstrapBCa", nboot=999)

## End(Not run)
```

Description

Obtain the point estimate and the confidence interval of the AUC by various methods based on the Mann-Whitney statistic.
Usage

```r
auc.nonpara.mw(x, y, conf.level=0.95, 
    method=c("newcombe", "pepe", "delong", "DL.corr", 
               "jackknife", "bootstrapP", "bootstrapBCa"), 
    nboot)
```

Arguments

- **x**: a vector of observations from class P.
- **y**: a vector of observations from class N.
- **conf.level**: confidence level of the interval. The default is 0.95.
- **method**: a method used to construct the CI. *newcombe* is the method recommended in Newcombe (2006); *pepe* is the method proposed in Pepe (2003); *delong* is the method proposed in Delong et al. (1988); *DL.corr* is a method proposed in Perme and Manevski (2018); *jackknife* uses the jackknife method; *bootstrapP* uses the bootstrap with percentile CI; *bootstrapBCa* uses bootstrap with bias-corrected and accelerated CI. The default is *newcombe*. It can be abbreviated.
- **nboot**: number of bootstrap iterations.

Details

The function implements various methods based on the Mann-Whitney statistic.

Value

Point estimate and lower and upper bounds of the CI of the AUC.

Note

The observations from class P tend to have larger values then that from class N.

Author(s)

Dai Feng, Damjan Manevski, Maja Pohar Perme

References


Margaret Sullivan Pepe (2003) The statistical evaluation of medical tests for classification and prediction. *Oxford University Press*
Examples

data(petBrainGlioma)
y <- subset(petBrainGlioma, grade==1, select="FDG", drop=TRUE)
x <- subset(petBrainGlioma, grade==2, select="FDG", drop=TRUE)
auc.nonpara.mw(x, y)
auc.nonpara.mw(x, y, method="delong")

 auc.para.bayes

AUC by the Bayesian MCMC

Description
Obtain the point estimate and the credible interval of the AUC using the Bayesian MCMC.

Usage

auc.para.bayes(x, y, conf.level=0.95,
dist=c("normalDV", "normalEV", "exponential"),
nburn=1000, niter=10000, nthin=1, seed=100)

Arguments
x a vector of observations from class P.
y a vector of observations from class N.
conf.level confidence level of the interval. The default is 0.95.
dist the name of a parametric distribution. normalEV stands for normal distributions with equal variance; normalDV stands for normal distributions with unequal variances; exponential stands for exponential distributions. The default is normalDV. It can be abbreviated.
nburn number of burn-in. The default is 1000.
niter number of iterations. The default is 10000.
nthin number of thinning interval. The default is 1.
seed the seed. The default is 100.

Details
Use the Bayesian MCMC to estimate the parameters of the distributions and hence the AUC values.

Value
Point estimate and lower and upper bounds of the CI of the AUC.
Note

The observations from class P tend to have larger values then that from class N.

Author(s)

Dai Feng

References


Examples

```r
#Example 1
data(petBrainGlioma)
y <- subset(petBrainGlioma, grade==1, select="FDG", drop=TRUE)
x <- subset(petBrainGlioma, grade==2, select="FDG", drop=TRUE)
auc.para.bayes(x, y, dist="exp")

#Example 2
data(petBrainGlioma)
y <- subset(petBrainGlioma, grade==1, select="ACE", drop=TRUE)
x <- subset(petBrainGlioma, grade==2, select="ACE", drop=TRUE)
auc.para.bayes(x, y, dist="normalDV")
```

 auc.para.frequentist  

 **Description**

Obtain the point estimate and the confidence interval of the AUC using some frequentist parametric methods.

 **Usage**

```r
auc.para.frequentist(x, y, conf.level=0.95, 
dist=c("normalDV", "normalEV", "exponential"), 
method=c("lrstar", "lr", "wald", "RG1", "RG2"))
```
arguments

- x: a vector of observations from class P.
- y: a vector of observations from class N.
- conf.level: confidence level of the interval. The default is 0.95.
- dist: the name of a parametric distribution. normalEV stands for normal distributions with equal variance; normalDV stands for normal distributions with unequal variances; exponential stands for exponential distributions. The default is normalDV. It can be abbreviated.
- method: a method used to construct the CI. lrstar uses the likelihood ratio test based on higher-order asymptotic results; lr uses the signed log-likelihood ratio test; wald uses the Wald test; RG1 is the approximate "t" solution to the Behrens-Fisher problem; RG2 is the normal approximation to RG1. RG1 and RG2 are for normal distributions. The default is lrstar. It can be abbreviated.

details

Use a variety of frequentist methods for different parametric models to estimate the AUC.

value

Point estimate and lower and upper bounds of the CI of the AUC.

note

The observations from class P tend to have larger values than those from class N.

author(s)

Dai Feng

references


examples

```r
# Example 1
data(petBrainGlioma)
y <- subset(petBrainGlioma, grade==1, select="FDG", drop=TRUE)
x <- subset(petBrainGlioma, grade==2, select="FDG", drop=TRUE)
auc.para.frequentist(x, y, dist="exp")
```
# Example 2

data(petBrainGlioma)
y <- subset(petBrainGlioma, grade==1, select="ACE", drop=TRUE)
x <- subset(petBrainGlioma, grade==2, select="ACE", drop=TRUE)
auc.para.frequentist(x, y, method="RG1")

---

petBrainGlioma  
*Standard Uptake Value (SUV) for Brain Glioma Grading*

**Description**

SUVs from FDG PET and ACE PET used in differentiating brain tumors.

**Usage**

petBrainGlioma

**Format**

A data-frame presenting the SUVs.

**Source**

Tatsuro Tsuchida, Hiroaki Takeuchi, Hidehiko Okazawa, Tetsuya Tsujikawa, and Yasuhisa Fujibayashi (2008) Grading of brain glioma with $^{11}$C-acetate PET: comparison with $^{18}$F-FDG PET. *Nuclear medicine and biology* 35(2) 171-176
Index

*Topic datasets
  petBrainGlioma, 8
*Topic htest
  auc.nonpara.kernel, 2
  auc.nonpara.mw, 3
  auc.para.bayes, 5
  auc.para.frequentist, 6

auc.nonpara.kernel, 2
auc.nonpara.mw, 3
auc.para.bayes, 5
auc.para.frequentist, 6

petBrainGlioma, 8