Package ‘oosse’

February 7, 2024

Type Package

Title Out-of-Sample R² with Standard Error Estimation

Version 1.0.11

Maintainer Stijn Hawinkel <`stijn.hawinkel@psb.ugent.be`>

Description Estimates out-of-sample R² through bootstrap or cross-validation as a measure of predictive performance. In addition, a standard error for this point estimate is provided, and confidence intervals are constructed.

License GPL-2

Encoding UTF-8

RoxygenNote 7.3.0

Imports stats, BiocParallel, Matrix, methods, doParallel, Rdpack

RdMacros Rdpack

Suggests knitr,rmarkdown,testthat,randomForest,glmnet

LazyData true

VignetteBuilder knitr

Depends R (>= 4.2.0)

BugReports `https://github.com/sthawinke/oosse`

NeedsCompilation no

Author Stijn Hawinkel [cre, aut] (<`https://orcid.org/0000-0002-4501-5180`>)

Repository CRAN

Date/Publication 2024-02-07 11:30:05 UTC

R topics documented:

`boot632` *The .632 bootstrap estimation of the MSE*

Description

The .632 bootstrap estimation of the MSE

Usage

```
boot632(y, x, id, fitFun, predFun)
```

Arguments

`y`	The vector of outcome values
`x`	The matrix of predictors
`id`	the sample indices resampled with replacement
`fitFun`	The function for fitting the prediction model
`predFun`	The function for evaluating the prediction model

Details

The implementation follows (Efron and Tibshirani 1997)

Value

The MSE estimate

References

Efron B, Tibshirani R (1997). “Improvements on cross-validation: The 632+ bootstrap method.” *J. Am. Stat. Assoc.*, **92**(438), 548 - 560.

See Also

estMSE bootOob

`boot632multiple` *Repeated .632 bootstrapa*

Description

Repeated .632 bootstrapa

Usage

```
boot632multiple(nBootstraps, y, ...)
```

Arguments

`nBootstraps`	The number of .632 bootstraps
`y`	The vector of outcome values
`...`	passed onto boot632

Value

The estimated MSE

`bootOob` *The oob bootstrap (smooths leave-one-out CV)*

Description

The oob bootstrap (smooths leave-one-out CV)

Usage

```
bootOob(y, x, id, fitFun, predFun)
```

Arguments

`y`	The vector of outcome values
`x`	The matrix of predictors
`id`	sample indices sampled with replacement
`fitFun`	The function for fitting the prediction model
`predFun`	The function for evaluating the prediction model

Details

The implementation follows (Efron and Tibshirani 1997)

Value

matrix of errors and inclusion times

References

Efron B, Tibshirani R (1997). “Improvements on cross-validation: The 632+ bootstrap method.” *J. Am. Stat. Assoc.*, **92**(438), 548 - 560.

See Also

estMSE boot632

`Brassica` *Gene expression and phenotypes of Brassica napus (rapeseed) plants*

Description

RNA-sequencing data of genetically identical Brassica napus plants in autumn, with 5 phenotypes next spring, as published by De Meyer S, Cruz DF, De Swaef T, Lootens P, Block JD, Bird K, Sprenger H, Van de Voorde M, Hawinkel S, Van Hautegem T, Inzé D, Nelissen H, Roldán-Ruiz I, Maere S (2022). “Predicting yield traits of individual field-grown Brassica napus plants from rosette-stage leaf gene expression.” *bioRxiv*. doi:10.1101/2022.10.21.513275, https://www.biorxiv.org/content/early/2022/10

Usage

```
Brassica
```

Format

A list with two components Expr and Pheno

Expr Matrix with Rlog values of 1000 most expressed genes

Pheno Data frame with 5 phenotypes and x and y coordinates of the plants in the field

Source

doi:10.1101/2022.10.21.513275

References

(De Meyer et al. 2022)

`buildConfInt` *Calculate a confidence interval for R², MSE and MST*

Description

Calculate a confidence interval for R², MSE and MST

Usage

```
buildConfInt(oosseObj, what = c("R2", "MSE", "MST"), conf = 0.95)
```

Arguments

`oosseObj`	The result of the R2oosse call
`what`	For which property should the ci be found: R² (default), MSE or MST
`conf`	the confidence level required

Details

The upper bound of the interval is truncated at 1 for the R² and the lower bound at 0 for the MSE

The confidence intervals for R² and the MSE are based on standard errors and normal approximations. The confidence interval for the MST is based on the chi-squared distribution as in equation (16) of (Harding et al. 2014), but with inflation by a factor (n+1)/n. All quantities are out-of-sample.

Value

A vector of length 2 with lower and upper bound of the confidence interval

References

Harding B, Tremblay C, Cousineau D (2014). "Standard errors: A review and evaluation of standard error estimators using Monte Carlo simulations." *The Quantitative Methods for Psychology*, **10**(2), 107 - 123.

See Also

R2oosse

Examples

```
data(Brassica)
fitFunLM = function(y, x){lm.fit(y = y, x = cbind(1, x))}
predFunLM = function(mod, x) {cbind(1,x) %*% mod$coef}
R2lm = R2oosse(y = Brassica$Pheno$Leaf_8_width, x = Brassica$Expr[, 1:10],
fitFun = fitFunLM, predFun = predFunLM, nFolds = 10)
buildConfInt(R2lm)
buildConfInt(R2lm, what = "MSE")
buildConfInt(R2lm, what = "MST")
```

`checkFitFun` *Check whether supplied prediction function meets the requirements*

Description

Check whether supplied prediction function meets the requirements

Usage

```
checkFitFun(fitFun, reqArgs = c("y", "x"))
```

Arguments

`fitFun`	The prediction function, or its name as character string
`reqArgs`	The vector of required arguments

Value

Throws an error when requirements not met, otherwise returns the function

---

`estCorMSEMST` *Estimate correlation between MSE and MST estimators*

Description

Estimate correlation between MSE and MST estimators

Usage

```
estCorMSEMST(
  y,
  x,
  fitFun,
  predFun,
  methodMSE,
  methodCor,
  nBootstrapsCor,
  nFolds,
  nBootstraps
)
```

Arguments

`y`	The vector of outcome values
`x`	The matrix of predictors
`fitFun`	The function for fitting the prediction model
`predFun`	The function for evaluating the prediction model
`methodMSE`	The method to estimate the MSE, either "CV" for cross-validation or "bootstrap" for .632 bootstrap
`methodCor`	The method to estimate the correlation between MSE and MST estimators, either "nonparametric" or "jackknife"
`nBootstrapsCor`	The number of bootstraps to estimate the correlation
`nFolds`	The number of outer folds for cross-validation
`nBootstraps`	The number of .632 bootstraps

Value

the estimated correlation

`estMSE` *Estimate MSE and its standard error*

Description

Estimate MSE and its standard error

Usage

```
estMSE(
  y,
  x,
  fitFun,
  predFun,
  methodMSE,
  nFolds,
  nInnerFolds,
  cvReps,
  nBootstraps
)
```

Arguments

`y`	The vector of outcome values
`x`	The matrix of predictors
`fitFun`	The function for fitting the prediction model
`predFun`	The function for evaluating the prediction model

`methodMSE`	The method to estimate the MSE, either "CV" for cross-validation or "bootstrap" for .632 bootstrap
`nFolds`	The number of outer folds for cross-validation
`nInnerFolds`	The number of inner cross-validation folds
`cvReps`	The number of repeats for the cross-validation
`nBootstraps`	The number of .632 bootstraps

Details

The nested cross-validation scheme follows (Bates et al. 2023), the .632 bootstrap is implemented as in (Efron and Tibshirani 1997)

Value

A vector with MSE estimate and its standard error

References

Bates S, Hastie T, Tibshirani R (2023). “Cross-validation: What does it estimate and how well does it do it?” *J. Am. Stat. Assoc.*, **118**(ja), 1 - 22. doi:10.1080/01621459.2023.2197686, https://doi.org/10.1080/01621459.2023.2

Efron B, Tibshirani R (1997). “Improvements on cross-validation: The 632+ bootstrap method.” *J. Am. Stat. Assoc.*, **92**(438), 548 - 560.

`formatSeconds` *Format seconds into human readable format*

Description

Format seconds into human readable format

Usage

```
formatSeconds(seconds, digits = 2)
```

Arguments

`seconds`	The number of seconds to be formatted
`digits`	the number of digits for rounding

Value

A character vector expressing time in human readable format

`getSEsNested` *Calculate standard error on MSE from nested CV results*

Description

Calculate standard error on MSE from nested CV results

Usage

```
getSEsNested(cvSplitReps, nOuterFolds, n)
```

Arguments

`cvSplitReps`	The list of outer and inner CV results
`nOuterFolds`	The number of outer folds
`n`	The sample size

Details

The calculation of the standard error of the MSE as proposed by (Bates et al. 2023)

Value

The estimate of the MSE and its standard error

References

Bates S, Hastie T, Tibshirani R (2023). "Cross-validation: What does it estimate and how well does it do it?" *J. Am. Stat. Assoc.*, **118**(ja), 1 - 22. doi:10.1080/01621459.2023.2197686, https://doi.org/10.1080/01621459.2023.2

See Also

estMSE

`isPD` *Helper function to check if matrix is positive definite*

Description

Helper function to check if matrix is positive definite

Usage

```
isPD(mat, tol = 1e-06)
```

Arguments

`mat`	The matrix
`tol`	The tolerance

Value

A boolean indicating positive definiteness

`processOob` *Process the out-of-bag bootstraps to get to standard errors following Efron 1997*

Description

Process the out-of-bag bootstraps to get to standard errors following Efron 1997

Usage

```
processOob(x)
```

Arguments

`x`	the list with out=of=bag bootstrap results

Value

out-of-bag MSE estimate and standard error

`R2oosse` *Estimate out-of-sample R² and its standard error*

Description

Estimate out-of-sample R² and its standard error

Usage

```
R2oosse(
  y,
  x,
  fitFun,
  predFun,
  methodMSE = c("CV", "bootstrap"),
  methodCor = c("nonparametric", "jackknife"),
  printTimeEstimate = TRUE,
```

```
    nFolds = 10L,
    nInnerFolds = nFolds - 1L,
    cvReps = 200L,
    nBootstraps = 200L,
    nBootstrapsCor = 50L,
    ...
)
```

Arguments

`y`	The vector of outcome values
`x`	The matrix of predictors
`fitFun`	The function for fitting the prediction model
`predFun`	The function for evaluating the prediction model
`methodMSE`	The method to estimate the MSE, either "CV" for cross-validation or "bootstrap" for .632 bootstrap
`methodCor`	The method to estimate the correlation between MSE and MST estimators, either "nonparametric" or "jackknife"
`printTimeEstimate`	A boolean, should an estimate of the running time be printed?
`nFolds`	The number of outer folds for cross-validation
`nInnerFolds`	The number of inner cross-validation folds
`cvReps`	The number of repeats for the cross-validation
`nBootstraps`	The number of .632 bootstraps
`nBootstrapsCor`	The number of bootstraps to estimate the correlation
`...`	passed onto fitFun and predFun

Details

Implements the calculation of the R² and its standard error by (Hawinkel et al. 2023). Multithreading is used as provided by the BiocParallel or doParallel packages, A rough estimate of expected computation time is printed when printTimeEstimate is true, but this is purely indicative. The options to estimate the mean squared error (MSE) are cross-validation (Bates et al. 2023) or the .632 bootstrap (Efron and Tibshirani 1997).

Value

A list with components

`R2`	Estimate of the R² with standard error
`MSE`	Estimate of the MSE with standard error
`MST`	Estimate of the MST with standard error
`corMSEMST`	Estimated correlation between MSE and MST estimators
`params`	List of parameters used
`fullModel`	The model trained on the entire dataset using fitFun
`n`	The sample size of the training data

References

Bates S, Hastie T, Tibshirani R (2023). “Cross-validation: What does it estimate and how well does it do it?” *J. Am. Stat. Assoc.*, **118**(ja), 1 - 22. doi:10.1080/01621459.2023.2197686, https://doi.org/10.1080/01621459.2023.2

Efron B, Tibshirani R (1997). “Improvements on cross-validation: The 632+ bootstrap method.” *J. Am. Stat. Assoc.*, **92**(438), 548 - 560.

Hawinkel S, Waegeman W, Maere S (2023). “Out-of-sample R2: Estimation and inference.” *Am. Stat.*, 1 - 16. doi:10.1080/00031305.2023.2216252, https://doi.org/10.1080/00031305.2023.2216252.

See Also

buildConfInt

Examples

```
data(Brassica)
#Linear model
fitFunLM = function(y, x){lm.fit(y = y, x = cbind(1, x))}
predFunLM = function(mod, x) {cbind(1,x) %*% mod$coef}
y = Brassica$Pheno$Leaf_8_width
R2lm = R2oosse(y = Brassica$Pheno$Leaf_8_width, x = Brassica$Expr[, 1:10],
fitFun = fitFunLM, predFun = predFunLM, nFolds = 10)
```

`RsquaredSE` *Calculate out-of-sample R^2 and its standard error based on MSE estimates*

Description

Calculate out-of-sample R^2 and its standard error based on MSE estimates

Usage

```
RsquaredSE(MSE, margVar, SEMSE, n, corMSEMST)
```

Arguments

`MSE`	An estimate of the mean squared error (MSE)
`margVar`	The marginal variance of the outcome, not scaled by (n+1)/n
`SEMSE`	The standard error on the MSE estimate
`n`	the sample size of the training data
`corMSEMST`	The correlation between MSE and marginal variance estimates

Details

This function is exported to allow the user to estimate the MSE and its standard error and the correlation between MSE and MST estimators himself. The marginal variance is scaled by (n+1)/n to the out-of-sample MST, so the user does not need to do this.

Value

A vector with the R² and standard error estimates

References

Hawinkel S, Waegeman W, Maere S (2023). “Out-of-sample R2: Estimation and inference.” *Am. Stat.*, 1 - 16. doi:10.1080/00031305.2023.2216252, https://doi.org/10.1080/00031305.2023.2216252.

See Also

R2oosse

Examples

```
#The out-of-sample R² calculated using externally provided estimates
RsquaredSE(MSE = 3, margVar = 4, SEMSE = 0.4, n = 50, corMSEMST = 0.75)
```

`simpleCV` *Perform simple CV, and return the MSE estimate*

Description

Perform simple CV, and return the MSE estimate

Usage

```
simpleCV(y, x, fitFun, predFun, nFolds)
```

Arguments

`y`	The vector of outcome values
`x`	The matrix of predictors
`fitFun`	The function for fitting the prediction model
`predFun`	The function for evaluating the prediction model
`nFolds`	The number of outer folds for cross-validation

Value

The MSE estimate

Index