Package ‘a4Classif’

March 28, 2017

Description
Classify using the Lasso algorithm as implemented in the glmnet package

Usage
lassoClass(object, groups)
Arguments

object object containing the expression measurements; currently the only method supported is one for ExpressionSet objects

groups character string indicating the column containing the class membership

Value

object of class glmnet

Author(s)

Willem Talloen

References


See Also

glmnet

Examples

if (require(ALL)){
  data(ALL, package = "ALL")
  ALL <- addGeneInfo(ALL)
  ALL$BTtype <- as.factor(substr(ALL$BT, 0, 1))

  resultLasso <- lassoClass(object = ALL, groups = "BTtype")
  plot(resultLasso, label = TRUE,
       main = "Lasso coefficients in relation to degree of penalization.")
  featResultLasso <- topTable(resultLasso, n = 15)
}

---

pamClass

Classify using Prediction Analysis for MicroArrays

Description

Classify using the Prediction Analysis for MicroArrays (PAM) algorithm as implemented in the pamr package

Usage

pamClass(object, groups, probe2gene = TRUE)
rfClass

Arguments

object object containing the expression measurements; currently the only method supported is one for ExpressionSet objects

groups character string indicating the column containing the class membership

probe2gene logical; if TRUE Affymetrix probeset IDs are translated into gene symbols; if FALSE no such translation is conducted

Value

object of class pamClass

Author(s)

Willem Talloen

References


Available at www.pnas.org


See Also

pamr.train

rfClass Classify using Random Forests

Description

Classify using the Random Forest algorithm of Breiman (2001)

Usage

rfClass(object, groups, probe2gene = TRUE)

Arguments

object object containing the expression measurements; currently the only method supported is one for ExpressionSet objects

groups character string indicating the column containing the class membership

probe2gene logical; if TRUE Affymetrix probeset IDs are translated into gene symbols in the output object; if FALSE no such translation is conducted

Value

Object of class 'rfClass'
Note

topTable and plot methods are available for `rfClass` objects.

Author(s)

Tobias Verbeke and Willem Talloen

References


See Also

`randomForest`

---

**ROCcurve**

*Receiver operating curve*

**Description**

A ROC curve plots the fraction of true positives (TPR = true positive rate) versus the fraction of false positives (FPR = false positive rate) for a binary classifier when the discrimination threshold is varied. Equivalently, one can also plot sensitivity versus (1 - specificity).

**Usage**

```r
ROCcurve(object, groups, probesetId = NULL, geneSymbol = NULL, main = NULL, probe2gene = TRUE, ...)
```

**Arguments**

- **object**: ExpressionSet object for the experiment
- **groups**: String containing the name of the grouping variable. This should be a the name of a column in the pData of the expressionSet object.
- **probesetId**: The probeset ID. These should be stored in the featureNames of the expressionSet object.
- **geneSymbol**: The gene symbol. These should be stored in the column `Gene Symbol` in the featureData of the expressionSet object.
- **main**: Main title on top of the graph
- **probe2gene**: Boolean indicating whether the probeset should be translated to a gene symbol (used for the default title of the plot)
- **...**: Possibility to add extra plot options. See `par`

**Author(s)**

Willem Talloen
References

Some explanation about ROC can be found on http://en.wikipedia.org/wiki/ROC_curve and http://www.anaesthetist.com/mnm/stats/roc/Findex.htm. The latter has at the bottom a nice interactive tool to scroll the cut-off and to see how it affects the FP/TP table and the ROC curve.

Examples

# simulated data set
esSim <- simulateData()
ROCcurve(probesetId = 'Gene.1', object = esSim, groups = 'type', addLegend = FALSE)

# ALL data set
if (require(ALL)){
  data(ALL, package = "ALL")
  ALL <- addGeneInfo(ALL)
  ALL$BTtype <- as.factor(substr(ALL$BT,0,1))
  ROCres <- ROCcurve(gene = "ABL1", object = ALL, groups = "BTtype")
}

ROCcurve
Index

∗Topic models
  lassoClass, 1
  pamClass, 2
  rfClass, 3

glmnet, 2

lassoClass, 1

pamClass, 2
pamr.train, 3
par, 4
plot.rfClass(rfClass), 3

randomForest, 4
rfClass, 3
ROCcurve, 4