# Package 'survriskpred'

June 27, 2019

**Title** Survival Risk Group Prediction

**Version** 0.2

**Description** Implements the survival risk prediction tool in ArrayTools

**Depends** R (>= 3.1.0)

**License** Same as BRB-ArrayTools
(https://brb.nci.nih.gov/BRB-ArrayTools/)

**Imports** impute, glmnet, survivalROC

**Author** BRB-ArrayTools team `<arraytools@emmes.com>`

**Maintainer** BRB-ArrayTools team `<arraytools@emmes.com>`

**RoxygenNote** 6.1.1

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

## R topics documented:

---

| plotKMCurve | *Plot Kaplan-Meier curves for risk groups obtained from cross validation.* |
|---|---|

---

## Description

This function plots Kaplan-Meier curves for risk groups obtained from cross validation using models with gene expression only, with covariates only and with both gene expression and covariates. If no covariates are provided, only the Kaplan-Meier curve related with the model with gene expression only can be plotted. If any covariates are provided, Kaplan-Meier curves for three models can be given.

## Usage

```
plotKMCurve(list, model)
```

## Arguments

list            list returned by function `survRiskPred`.

model           numeric for a model considering

- 1: gene expression only
- 2: covariates only
- 3: both gene expression and covariates

## Examples

```
res <- test.survRiskPredict("Pomeroy")
plotKMCurve(res, 1)
plotKMCurve(res, 2)
plotKMCurve(res, 3)
```

---

plotROCCurveRisk     *Plot ROC curves.*

---

## Description

This function plots the ROC curves at a specified landmark time using the Nearest Neighbor Estimation (NNE) method of Heagerty, Lumley and Pepe, 2000. If no covariates are provided, only the ROC curve with the model considering gene expression only will be plotted. If any covariates are provided, two ROC curves with models considering gene expression only and considering both gene expression and covariates will be given.

## Usage

```
plotROCCurveRisk(list)
```

## Arguments

list            list returned by function `survRiskPred`.

## Examples

```
res <- test.survRiskPredict("Pomeroy")
plotROCCurveRisk(res)
```

---

```
survRiskPredict          Survival risk prediction
```

---

**Description**

This function implements the survial risk prediction tool in BRB-ArrayTools

**Usage**

```
survRiskPredict(exprTrain, covTrain, exprTest = NULL, covTest = NULL,
    geneid, status, tme, geneSelect = "pc", nriskgroups = 2,
    progIndexPerc, cvMethod = "10fold", nperm = 0, landmarktime = 0,
    alpha = 0.001, ncomp = 2, mixing = 1, pcrgenes = 10, projectPath,
    outputName = "SurvivalRiskPrediction", generateHTML = FALSE)
```

**Arguments**

| | |
|---|---|
| exprTrain | matrix/data frame of gene expression data for training samples. Rows are genes and columns are arrays. Its column names must be provided. |
| covTrain | matrix/data frame of clinical covariates from the training samples. Rows are arrays and columns are clinical covariates. Default is NULL. |
| exprTest | matrix/data frame of gene expression data for new samples. Rows are genes and columns are arrays. Its column names must be provided. |
| geneid | matrix/data frame of gene IDs. |
| status | vector specifying survival status (1 = death, 0 = censored). |
| tme | vector specifying survival time. |
| geneSelect | character string of a gene selection method: "pc" for principal component and "pcr" for penalized Cox regression. Default is "pc". |
| nriskgroups | numeric of the number of risk groups. It can only be 2 or 3. Default is 2. |
| progIndexPerc | |
| | numeric of prognostic index percentile. In 2-risk group case, one percentile values needs to be provided, whereas, in the 3-risk group case, two percentile values need to be provided. Default is 50 for 2-risk group case. |
| cvMethod | character string of cross validation method. "loocv" for leave-one-out cross validation, and "10fold" for 10-fold cross validation. |
| nperm | numeric of number of permutation tests. If nperm == 0, no permutation tests will be conducted; otherwise, permutation tests based on log rank statistics will be run. Default is 0. |
| landmarktime | numeric of landmark time for the ROC curve. If landmarktime=0, median of survival time will be used. |
| alpha | numeric of the threshold significance level of penalized Cox regression model for selecting the genes to be used in computing the principal components. Default is 0.001. |

| | |
|---|---|
| `ncomp` | numeric of number of principal components if `geneSelect = "pc"`. |
| `mixing` | numeric of the mixing parameter in penalized Cox regression. |
| `pcrgenes` | numeric of number of desired genes if `geneSelect = "pcr"`. |
| `projectPath` | character string specifying the full project path. |
| `outputName` | character string specifying the output folder name. Default is "SurvivalRiskPrediction". |
| `generateHTML` | logical. If `TRUE`, an HTML page will be generated with detailed class prediction results saved in <projectPath>/Output/<outputName>/<outputName>.html; otherwise not. |
| `CovTest` | matrix/data frame of clinical covariates from new samples. Rows are arrays and columns are clinical covariates. Default is NULL. |

### Details

Please see the BRB-ArrayTools manual for details.

### Value

A list that may include the following objects:

- `predictNewSamplesModE`: a data frame with risk group predictions of new samples when the model with gene expression only is considered.

- `genesInClassifierModE`: a data frame with genes selected by fitting penalized Cox proportional hazards model when the model with gene expression only is considered.

- `loadingMatrixModE`: a data frame with loading matrix of the significant genes and the correlations between the principal components and the significant genes when the model with gene expression only is considered.

- `predictRiskTrainingModE`: a data frame with Table of arrays, survival time, censoring indicator and survival risk prediction of training samples when the model with gene expression only is considered.

- `predictNewSamplesModCo`: a data frame with risk group predictions of new samples when the model with both covariates and gene expression is considered.

- `genesInClassifierModCo`: a data frame with genes selected by fitting penalized Cox proportional hazards model when the model with both covariates and gene expression is considered.

- `loadingMatrixModCo`: a data frame with loading matrix of the significant genes and the correlations between the principal components and the significant genes when the model with both covariates and gene expression is considered.

- `predictRiskTrainingModCo`: a data frame with Table of arrays, survival time, censoring indicator and survival risk prediction of training samples when the model with both covariates and gene expression is considered.

- `predictNewSamplesModCl`: a data frame with risk group predictions of new samples when the model with covariates only is considered.

- `predictRiskTrainingModCl`: a data frame with Table of arrays, survival time, censoring indicator and survival risk prediction of training samples when the model with covariates only is considered.

- `ppval`: a vector with permutation p-values for the models with gene expression only, covariates only or the combination of gene expression and covariates.
- `workPath`: the path for Fortran and other intermediate outputs.

## Examples

```
#example 1
resList1 <- test.survRiskPredict("Pomeroy")
#example2
data("Pomeroy")
projectPath <- tempdir()
outputName <- "SurvivalRiskPrediction"
generateHTML <- TRUE
resList2 <- survRiskPredict(exprTrain, covTrain, exprTest, covTest,
                            geneId, status, tme, geneSelect = "pc",
                            nriskgroups = 2, progIndexPerc = 50,
                            cvMethod = "10fold", nperm = 0, landmarktime = 0,
                            alpha = .001, ncomp = 2, mixing = 1, pcrgenes = 10,
                            projectPath = projectPath,
                            outputName = outputName, generateHTML)
if (generateHTML)
  browseURL(file.path(projectPath, "Output", outputName,
            paste0(outputName, ".html")))
#example3
data("Perou")
projectPath <- file.path(Sys.getenv("HOME"),"Pomeroy")
outputName <- "SurvivalRiskPrediction"
generateHTML <- TRUE
resList3 <- survRiskPredict(exprTrain, covTrain, exprTest, covTest,
                            geneId, status, tme, geneSelect = "pc",
                            nriskgroups = 2, progIndexPerc = 50,
                            cvMethod = "10fold", nperm = 0, landmarktime = 0,
                            alpha = .001, ncomp = 2, mixing = 1, pcrgenes = 10,
                            projectPath = projectPath,
                            outputName = outputName, generateHTML)
if (generateHTML)
  browseURL(file.path(projectPath, "Output", outputName,
            paste0(outputName, ".html")))
```

---

```
test.survRiskPredict
```
*Test survRiskPredict() function*

---

### Description

This function will load a test dataset and use it run `survRiskPredict` function.

### Usage

```
test.survRiskPredict(dataset = c("Pomeroy", "Perou"), projectPath,
   outputName = "SurvivalRiskPrediction", generateHTML = FALSE)
```

**Arguments**

| | |
|---|---|
| `dataset` | character string specifying one of "Perou" or "Pomeroy" datasets. |
| `projectPath` | character string specifying the project path. Default is C:/Users/UserName/Documents/<dataset>. |
| `outputName` | character string for the output folder name. |
| `generateHTML` | logical. If `TRUE`, an html page will be generated with detailed prediction results. |

**Value**

A list as returned by `survRiskPredict`.

**See Also**

[survRiskPredict](#)

**Examples**

```
test.survRiskPredict("Pomeroy")
```

# Index