Package 'MetabolicSurv'

June 11, 2021

Type Package

Title A Biomarker Validation Approach for Classification and Predicting Survival Using Metabolomics Signature

Version 1.1.2

Description An approach to identifies metabolic biomarker signature for metabolic data by discovering predictive metabolite for predicting survival and classifying patients into risk groups. Classifiers are constructed as a linear combination of predictive/important metabolites, prognostic factors and treatment effects if necessary.
Several methods were implemented to reduce the metabolomics matrix such as the principle component analysis of Wold Svante et al. (1987) <doi:10.1016/0169-7439(87)80084-9> , the LASSO method by Robert Tibshirani (1998) <doi:10.1002/(SICI)1097-0258(19970228)16:4%3C385::AID-SIM380%3E3.0.CO;2-3>, the elastic net approach by Hui Zou and Trevor Hastie (2005) <doi:10.1111/j.1467-9868.2005.00503.x>.
Sensitivity analysis on the quantile used for the classification can also be accessed to check the deviation of the classification group based on the quantile specified.
Large scale cross validation can be performed in order to investigate the mostly selected predictive metabolites and for internal validation. During the evaluation process, validation is accessed using the hazard ratios (HR) distribution of the test set and inference is mainly based on resampling and permutations technique.

URL https://github.com/OlajumokeEvangelina/MetabolicSurv

BugReports https://github.com/OlajumokeEvangelina/MetabolicSurv/issues/new

Depends R (>= 4.1.0)

License GPL-3

Encoding UTF-8

LazyData true

Imports superpc, glmnet, matrixStats, survminer, survival, rms, tidyr, pls, Rdpack, methods, stats, ggplot2, dplyr

RoxygenNote 7.1.1

RdMacros Rdpack

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Olajumoke Evangelina Owokotomo [aut, cre],
Ziv Shkedy [ctb]

Maintainer Olajumoke Evangelina Owokotomo <olajumoke.owokotomo@uhasselt.be>

Repository CRAN

Date/Publication 2021-06-11 08:30:02 UTC

R topics documented:

`CVLasoelacox` *Cross Validations for Lasso Elastic Net Survival predictive models and Classification*

Description

The function does cross validation for Lasso, Elastic net and Ridge regressions models before the survial analysis and classification. The survival analysis is based on the selected metabolites in the presence or absene of prognostic factors.

Usage

```
CVLasoelacox(
  Survival,
  Censor,
  Mdata,
  Prognostic,
  Quantile = 0.5,
  Metlist = NULL,
  Standardize = TRUE,
  Reduce = TRUE,
  Select = 15,
  Alpha = 1,
  Fold = 4,
```

```
  Ncv = 10,
  nlambda = 100
)
```

Arguments

Survival	A vector of survival time with length equals to number of subjects
Censor	A vector of censoring indicator
Mdata	A large or small metabolic profile matrix. A matrix with metabolic profiles where the number of rows should be equal to the number of metabolites and number of columns should be equal to number of patients.
Prognostic	A dataframe containing possible prognostic(s) factor and/or treatment effect to be used in the model.
Quantile	The cut off value for the classifier, default is the median cutoff
Metlist	A list of metabolites to be considered in the model usually smaller than the metabolites in the Mdata . Default is to use all metabolites available and it is advisable to be greater than 17.
Standardize	A Logical flag for the standardization of the metabolite matrix, prior to fitting the model sequence. The coefficients are always returned on the original scale. Default is standardize=TRUE.
Reduce	A boolean parameter indicating if the metabolic profile matrix should be reduced, default is TRUE and larger metabolic profile matrix is reduced by supervised pca approach and first pca is extracted from the reduced matrix to be used in the classifier.
Select	Number of metabolites (default is 15) to be selected from supervised PCA. This is valid only if the argument Reduce=TRUE
Alpha	The mixing parameter for glmnet (see glmnet). The range is 0<= Alpha <= 1. The Default is 1
Fold	number of folds to be used for the cross validation. Its value ranges between 3 and the numbe rof subjects in the dataset
Ncv	Number of validations to be carried out. The default is 25.
nlambda	The number of lambda values - default is 100 as in glmnet.

Details

The function performs the cross validations for Lasso, Elastic net and Ridge regressions models for Cox proportional hazard model. Metabolites are selected at each iteration and then use for the classifier. This implies that predictive metabolites signature is varied from one cross validation to the other depending on selection. The underline idea is to investigate the Hazard Ratio for the train and test data based on the optimal lambda selected for the non-zero shrinkage coefficients, the nonzero selected metabolites will thus be used in the survival analysis and in calculation of the risk scores for each sets of data.

Value

A object of class cvle is returned with the following values

- Coef.mat :A matrix of coefficients with rows equals to number of cross validations and columns equals to number of metabolites.
- RuntimeA vector of runtime for each iteration measured in seconds.
- lambdaA vector of estimated optimum lambda for each iterations.
- nA vector of the number of selected metabolites
- HRTrainA matrix of survival information for the training dataset. It has three columns representing the estimated HR, the 95% lower confidence interval and the 95% upper confidence interval.
- HRTestA matrix of survival information for the test dataset. It has three columns representing the estimated HR, the 95% lower confidence interval and the 95% upper confidence interval.
- pldA vector of partial likelihood deviance at each cross validations.
- Met.matA matrix with 0 and 1. Number of rows equals to number of iterations and number of columns equals to number of metabolites. 1 indicates that the particular metabolite was selected or had nonzero coefficient and otherwise it is zero.
- MdataThe Metabolite data matrix that was used for the analysis either same as Mdata or a reduced version.

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

See Also

coxph, EstimateHR, glmnet, Lasoelacox

Examples

```
## FIRSTLY SIMULATING A METABOLIC SURVIVAL DATA
Data = MSData(nPatients = 100, nMet = 150, Prop = 0.5)

## USING THE FUNCTION
Results = CVLasoelacox(Survival = Data$Survival,Censor = Data$Censor,
Mdata = t(Data$Mdata),Prognostic = Data$Prognostic, Quantile = 0.5,
Metlist = NULL,Standardize = TRUE, Reduce=FALSE, Select=15,
Alpha = 1,Fold = 4,Ncv = 10,nlambda = 100)

## NUMBER OF SELECTED METABOLITES PER CV
Results@n

## GET THE MATRIX OF COEFFICIENTS
Results@Coef.mat
```

```
## SURVIVAL INFORMATION OF THE TRAIN DATASET
Results@HRTrain

## SURVIVAL INFORMATION OF THE TEST DATASET
Results@HRTest
```

`cvle` *Constructor for the cvle class*

Description

Constructor for the cvle class

Usage

```
cvle(Coef.mat, Runtime, lambda, n, Met.mat, HRTrain, HRTest, pld, Mdata)
```

Arguments

`Coef.mat`	A matrix of coefficients with rows equals to number of cross validations and columns equals to number of metabolites.
`Runtime`	A vector of runtime for each iteration measured in seconds.
`lambda`	A vector of estimated optimum lambda for each iterations.
`n`	A vector of the number of selected metabolites
`Met.mat`	A matrix with 0 and 1. Number of rows equals to number of iterations and number of columns equals to number of metabolites. 1 indicates that the particular metabolite was selected or had nonzero coefficient and otherwise it is zero.
`HRTrain`	A matrix of survival information for the training dataset. It has three columns representing the estimated HR, the 95% lower confidence interval and the 95% upper confidence interval.
`HRTest`	A matrix of survival information for the test dataset. It has three columns representing the estimated HR, the 95% lower confidence interval and the 95% upper confidence interval.
`pld`	A vector of partial likelihood deviance at each cross validations.
`Mdata`	The metabolite matrix that was used for the analysis which can either be the full the full data or a reduced supervised PCA version.

Value

object of class cvle

cvle-class *The cvle Class.*

Description

The cvle Class.

Slots

`Coef.mat` A matrix of coefficients with rows equals to number of cross validations and columns equals to number of metabolites.

`Runtime` A vector of runtime for each iteration measured in seconds.

`lambda` A vector of estimated optimum lambda for each iterations.

`n` A vector of the number of selected metabolites

`Met.mat` A matrix with 0 and 1. Number of rows equals to number of iterations and number of columns equals to number of metabolites. 1 indicates that the particular metabolite was selected or had nonzero coefficient and otherwise it is zero.

`HRTrain` A matrix of survival information for the training dataset. It has three columns representing the estimated HR, the 95% lower confidence interval and the 95% upper confidence interval.

`HRTest` A matrix of survival information for the test dataset. It has three columns representing the estimated HR, the 95% lower confidence interval and the 95% upper confidence interval.

`pld` A vector of partial likelihood deviance at each cross validations.

`Mdata` The metabolite matrix that was used for the analysis which can either be the full the full data or a reduced supervised PCA version.

Author(s)

Olajumoke Evangelina Owokotomo, `<olajumoke.owokotomo@uhasselt.be>`

Ziv Shkedy

See Also

`EstimateHR`, `glmnet`, `Lasoelacox`

Examples

```
## GENERATE SOME METABOLIC SURVIVAL DATA WITH PROGNOSTIC FACTORS
Data<-MSData(nPatients=100,nMet=150,Prop=0.5)

## USE THE FUNCTION
Eg = CVLasoelacox(Survival = Data$Survival,Censor = Data$Censor,
Mdata = t(Data$Mdata),Prognostic = Data$Prognostic, Quantile = 0.5,
Metlist = NULL,Standardize = TRUE, Reduce=FALSE, Select=15,
Alpha = 1,Fold = 4,Ncv = 10,nlambda = 100)
```

```
## GET THE CLASS OF THE OBJECT
class(Eg)     # An "cvle" Class

##  METHOD THAT CAN BE USED FOR THIS CLASS
show(Eg)
summary(Eg)
plot(Eg, type =3)
```

`CVMajorityvotes` *Cross validation for majority votes*

Description

This function does cross validation for the Majority votes based classification.

Usage

```
CVMajorityvotes(
  Survival,
  Censor,
  Prognostic = NULL,
  Mdata,
  Reduce = TRUE,
  Select = 15,
  Fold = 3,
  Ncv = 100
)
```

Arguments

`Survival`	A vector of survival time with length equals to number of subjects
`Censor`	A vector of censoring indicator
`Prognostic`	A dataframe containing possible prognostic(s) factor and/or treatment effect to be used in the model.
`Mdata`	A large or small metabolic profile matrix. A matrix with metabolic profiles where the number of rows should be equal to the number of metabolites and number of columns should be equal to number of patients.
`Reduce`	A boolean parameter indicating if the metabolic profile matrix should be reduced, default is TRUE and larger metabolic profile matrix is reduced by supervised pca approach and first pca is extracted from the reduced matrix to be used in the classifier.
`Select`	Number of metabolites (default is 15) to be selected from supervised PCA. This is valid only if th argument Reduce=TRUE

Fold	Number of times in which the dataset is divided. Default is 3 which implies dataset will be divided into three groups and 2/3 of the dataset will be the train datset and 1/3 will be to train the results.
Ncv	The Number of cross validation loop. Default is 50 but it is recommended to have at least 100.

Details

This function does cross validation for the Majority votes based classification which is a cross validated approach to Majorityvotes.

Value

A object of class cvmv is returned with the following values

HRTrain	A matrix of survival information for the training dataset. It has three columns representing the estimated HR, the 95% lower confidence interval and the 95% upper confidence interval.
HRTest	A matrix of survival information for the test dataset. It has three columns representing the estimated HR, the 95% lower confidence interval and the 95% upper confidence interval.
Ncv	The number of cross validation used
Mdata	The Metabolite data matrix that was used for the analysis either same as Mdata or a reduced version.
Progfact	The names of prognostic factors used

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

See Also

Majorityvotes

Examples

```
## FIRSTLY SIMULATING A METABOLIC SURVIVAL DATA
Data = MSData(nPatients = 100, nMet = 150, Prop = 0.5)

## USING THE FUNCTION
Result = CVMajorityvotes(Survival=Data$Survival,Censor=Data$Censor,
Prognostic=Data$Prognostic, Mdata=t(Data$Mdata), Reduce=FALSE,
Select=15, Fold=3, Ncv=10)

## GET THE CLASS OF THE OBJECT
class(Result)     # An "cvmv" Class
```

```
##  METHOD THAT CAN BE USED FOR THE RESULT
show(Result)
summary(Result)
```

CVMetSpecificCoxPh *Cross validation for the Metabolite specific analysis*

Description

The function performs cross validation for each metabolite depending the number of fold which guides the division into the train and testing dataset. The classifier is then obtained on the training dataset to be validated on the test dataset

Usage

```
CVMetSpecificCoxPh(
  Fold = 3,
  Survival,
  Mdata,
  Censor,
  Reduce = TRUE,
  Select = 150,
  Prognostic = NULL,
  Quantile = 0.5,
  Ncv = 3
)
```

Arguments

Fold	Number of times in which the dataset is divided. Default is 3 which implies dataset will be divided into three groups and 2/3 of the dataset will be the train datset and 1/3 will be to train the results.
Survival	A vector of survival time with length equals to number of subjects
Mdata	A large or small metabolic profile matrix. A matrix with metabolic profiles where the number of rows should be equal to the number of metabolites and number of columns should be equal to number of patients.
Censor	A vector of censoring indicator
Reduce	A boolean parameter indicating if the metabolic profile matrix should be reduced, default is TRUE and larger metabolic profile matrix is reduced by supervised pca approach and first pca is extracted from the reduced matrix to be used in the classifier.
Select	Number of metabolites (default is 15) to be selected from supervised PCA. This is valid only if th argument Reduce=TRUE

Prognostic	A dataframe containing possible prognostic(s) factor and/or treatment effect to be used in the model.
Quantile	The cut off value for the classifier, default is the median cutoff
Ncv	The Number of cross validation loop. Default is 50 but it is recommended to have at least 100.

Details

This function performs the cross validation for metabolite by metabolite analysis. The data will firstly be divided into data train dataset and test datset. Furthermore, a metabolite-specific model is fitted on train data and a classifier is built. In addition, the classifier is then evaluated on test dataset for each particular metabolite. The Process is repeated for all the full or reduced metabolites to obtaind the HR statistics of the low risk group. The following steps depends on the number of cross validation specified.

Value

A object of class cvmm is returned with the following values

HRTrain	The Train dataset HR statistics for each metabolite by the number of CV
HRTest	The Test dataset HR statistics for each metabolite by the number of CV
train	The selected subjects for each CV in the train dataset
train	The selected subjects for each CV in the test dataset
n.mets	The number of metabolite used in the analysis
Ncv	The number of cross validation performed
Rdata	The Metabolite data matrix that was used for the analysis either same as Mdata or a reduced version.

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

See Also

coxph, EstimateHR, MSpecificCoxPh,

Examples

```
## FIRSTLY SIMULATING A METABOLIC SURVIVAL DATA
Data = MSData(nPatients = 100, nMet = 150, Prop = 0.5)

## USING THE FUNCTION
Result = CVMetSpecificCoxPh(Fold=3,Survival=Data$Survival,
Mdata=t(Data$Mdata),Censor= Data$Censor,Reduce=TRUE,
Select=150,Prognostic=Data$Prognostic,Quantile = 0.5,Ncv=3)
```

```
## GET THE CLASS OF THE OBJECT
class(Result)     # An "cvmm" Class

##  METHOD THAT CAN BE USED FOR THE RESULT
show(Result)
summary(Result)
plot(Result)
```

---

cvmm *Constructor for the cvmm class*

Description

Constructor for the cvmm class

Usage

```
cvmm(HRTrain, HRTest, train, test, n.mets, Ncv, Rdata)
```

Arguments

HRTrain	A 3-way array, The first dimension is the number of metabolites, the second dimension is the HR statistics for the low risk group in the train dataset (HR,1/HR LCI, UCI) while the third dimension is the number of cross validation performed.
HRTest	A 3-way array, The first dimension is the number of metabolites, the second dimension is the HR statistics for the low risk group in the test dataset (HR,1/HR LCI, UCI) while the third dimension is the number of cross validation performed.
train	The selected subjects for each CV in the train dataset
test	The selected subjects for each CV in the test dataset
n.mets	The number of metabolite used in the analysis
Ncv	The number of cross validation performed
Rdata	The Metabolite data matrix that was used for the analysis either same as Mdata or a reduced version

Value

object of class cvmm

cvmm-class *The cvmm Class.*

Description

The cvmm Class.

Slots

`HRTrain` A 3-way array, The first dimension is the number of metabolites, the second dimension is the HR statistics for the low risk group in the train dataset (HR,1/HR LCI, UCI) while the third dimension is the number of cross validation performed.

`HRTest` A 3-way array, The first dimension is the number of metabolites, the second dimension is the HR statistics for the low risk group in the test dataset (HR,1/HR LCI, UCI) while the third dimension is the number of cross validation performed.

`train` The selected subjects for each CV in the train dataset

`test` The selected subjects for each CV in the test dataset

`n.mets` The number of metabolite used in the analysis

`Ncv` The number of cross validation performed

`Rdata` The Metabolite data matrix that was used for the analysis either same as Mdata or a reduced version

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

See Also

CVMetSpecificCoxPh

Examples

```
## GENERATE SOME METABOLIC SURVIVAL DATA WITH PROGNOSTIC FACTORS
Data<-MSData(nPatients=100,nMet=150,Prop=0.5)

## USING THE FUNCTION
Result = CVMetSpecificCoxPh(Fold=3,Survival=Data$Survival,
Mdata=t(Data$Mdata),Censor= Data$Censor,Reduce=TRUE,
Select=150,Prognostic=Data$Prognostic,Quantile = 0.5,Ncv=3)

## GET THE CLASS OF THE OBJECT
class(Result)     # An "cvmm" Class

##  METHOD THAT CAN BE USED FOR THIS CLASS
```

```
show(Result)
summary(Result)
plot(Result)
```

`cvmv` *Constructor for the cvmv class*

Description

Constructor for the cvmv class

Usage

```
cvmv(HRTrain, HRTest, Ncv, Mdata, Progfact)
```

Arguments

`HRTrain`	A matrix of survival information for the training dataset. It has three columns representing the estimated HR, the 95% lower confidence interval and the 95% upper confidence interval.
`HRTest`	A matrix of survival information for the test dataset. It has three columns representing the estimated HR, the 95% lower confidence interval and the 95% upper confidence interval.
`Ncv`	The number of cross validation used
`Mdata`	The Metabolite data matrix that was used for the analysis either same as Mdata or a reduced version.
`Progfact`	The names of prognostic factors used

Value

object of class cvmv

`cvmv-class` *The cvmv Class.*

Description

The cvmv Class.

Slots

HRTrain A matrix of survival information for the training dataset. It has three columns representing the estimated HR, the 95% lower confidence interval and the 95% upper confidence interval.

HRTest A matrix of survival information for the test dataset. It has three columns representing the estimated HR, the 95% lower confidence interval and the 95% upper confidence interval.

Ncv The number of cross validation used

Mdata The Metabolite data matrix that was used for the analysis either same as Mdata or a reduced version.

Progfact The names of prognostic factors used

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

See Also

Majorityvotes, CVPcaPls, SurvPcaClass, SurvPlsClass

Examples

```
## GENERATE SOME METABOLIC SURVIVAL DATA WITH PROGNOSTIC FACTORS
Data<-MSData(nPatients=100,nMet=150,Prop=0.5)

## USING THE FUNCTION
Result = CVMajorityvotes(Survival=Data$Survival,Censor=Data$Censor,
Prognostic=Data$Prognostic, Mdata=t(Data$Mdata), Reduce=FALSE,
Select=15, Fold=3, Ncv=10)

## GET THE CLASS OF THE OBJECT
class(Result)     # A "cvmv" Class

##  METHOD THAT CAN BE USED FOR THE RESULT
show(Result)
summary(Result)
```

CVPcaPls *Cross Validations for PCA and PLS based methods*

Description

This function does cross validation for the analysis performs by SurvPcaClass and SurvPlsClass functions where the dimension reduction methods can either be PCA and PLS.

Usage

```
CVPcaPls(
  Fold = 3,
  Survival,
  Mdata,
  Censor,
  Reduce = TRUE,
  Select = 15,
  Prognostic = NULL,
  Ncv = 5,
  DR = "PCA"
)
```

Arguments

Fold	Number of times in which the dataset is divided. Default is 3 which implies dataset will be divided into three groups and 2/3 of the dataset will be the train datset and 1/3 will be to train the results.
Survival	A vector of survival time with length equals to number of subjects
Mdata	A large or small metabolic profile matrix. A matrix with metabolic profiles where the number of rows should be equal to the number of metabolites and number of columns should be equal to number of patients.
Censor	A vector of censoring indicator
Reduce	A boolean parameter indicating if the metabolic profile matrix should be reduced, default is TRUE and larger metabolic profile matrix is reduced by supervised pca approach and first pca is extracted from the reduced matrix to be used in the classifier.
Select	Number of metabolites (default is 15) to be selected from supervised PCA. This is valid only if th argument Reduce=TRUE
Prognostic	A dataframe containing possible prognostic(s) factor and/or treatment effect to be used in the model.
Ncv	The Number of cross validation loop. Default is 50 but it is recommended to have at least 100.
DR	The dimension reduction method. It can be either "PCA" for Principle components analysis or "PLS" for Partial least squares.

Details

This function does cross validation for the analysis using two reduction method. The reduction method can be PCA or PLS. If it is PCA then the SurvPcaClass is internally used for the cross validation and SurvPlsClass otherwise.

Value

A object of class cvpp is returned with the following values

Result	A dataframe containg the estimated Hazard ratio of the test dataset and the training dataset
Ncv	The number of cross validation performed
Method	The dimesion reduction method used
CVtrain	The training dataset indices matrix used for the cross validation
CVtest	The test dataset indices matrix used for the cross validation
Select	The number of metabolite used for the dimesion reduction method used

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

References

Bair E, Hastie T, Debashis P, Tibshirani R (2006). "Prediction by supervised principal components." *American Statistics Association,*, **101**(473), 119–137.

See Also

SurvPlsClass, SurvPcaClass

Examples

```
## FIRSTLY SIMULATING A METABOLIC SURVIVAL DATA
Data = MSData(nPatients = 100, nMet = 150, Prop = 0.5)

## USING THE FUNCTION
Result = CVPcaPls(Fold = 4, Survival = Data$Survival,
Mdata = t(Data$Mdata), Censor = Data$Censor, Reduce=TRUE,
Select=19, Prognostic= Data$Prognostic,Ncv=55,DR ="PLS")

## GET THE CLASS OF THE OBJECT
class(Result)     # An "cvpp" Class

##  METHOD THAT CAN BE USED FOR THE RESULT
show(Result)
summary(Result)
plot(Result)
```

cvpp *Constructor for the cvpp class*

Description

Constructor for the cvpp class

Usage

```
cvpp(Results, Ncv, Method, CVtrain, CVtest, Nmet)
```

Arguments

Results	A dataframe containg the estimated Hazard ratio of the test dataset and the training dataset
Ncv	The number of cross validation performed
Method	The dimesion reduction method used
CVtrain	The training dataset indices matrix used for the cross validation
CVtest	The test dataset indices matrix used for the cross validation
Nmet	The number of metabolite used for the dimesion reduction method used

Value

object of class cvpp

cvpp-class *The cvpp Class.*

Description

The cvpp Class.

Slots

Results A dataframe containg the estimated Hazard ratio of the test dataset and the training dataset

Ncv The number of cross validation performed

Method The dimesion reduction method used

CVtrain The training dataset indices matrix used for the cross validation

CVtest The test dataset indices matrix used for the cross validation

Nmet The number of metabolite used for the dimesion reduction method used

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

See Also

CVPcaPls, SurvPcaClass, SurvPlsClass

Examples

```
## GENERATE SOME METABOLIC SURVIVAL DATA WITH PROGNOSTIC FACTORS
Data<-MSData(nPatients=100,nMet=150,Prop=0.5)

## USING THE FUNCTION
Result = CVPcaPls(Fold = 4, Survival = Data$Survival,
Mdata = t(Data$Mdata), Censor = Data$Censor, Reduce=TRUE,
Select=19, Prognostic= Data$Prognostic,Ncv=55,DR ="PLS")

## GET THE CLASS OF THE OBJECT
class(Result)     # A "cvpp" Class

##  METHOD THAT CAN BE USED FOR THE RESULT
show(Result)
summary(Result)
plot(Result)
```

---

cvsim *Constructor for the cvsim class*

Description

Constructor for the cvsim class

Usage

```
cvsim(HRpca, HRpls, Nmets, Ncv, Top)
```

Arguments

HRpca	A 3-way array in which first, second, and third dimensions correspond to number of metabolites, Hazard ratio infromation(Estimated HR, LowerCI and UpperCI), and number of cross validation respectively. This contains the estimated HR on test data and dimension reduction method is PCA.
HRpls	A 3-way array in which first, second, and third dimensions correspond to number of metabolites, Hazard ratio infromation(Estimated HR, LowerCI and UpperCI), and number of cross validation respectively. This contains the estimated HR on test data and dimension reduction method is PLS.

Nmets	The number of metabolites in the reduced matrix
Ncv	The number of cross validation done
Top	A sequence of top k metabolites considered. Default is Top=seq(5,100,by=5)

Value

object of class cvsim

---

cvsim-class *The cvsim Class.*

Description

The cvsim Class.

Slots

HRpca A 3-way array in which first, second, and third dimensions correspond to number of metabolites, Hazard ratio infromation(Estimated HR, LowerCI and UpperCI), and number of cross validation respectively. This contains the estimated HR on test data and dimension reduction method is PCA.

HRpls A 3-way array in which first, second, and third dimensions correspond to number of metabolites, Hazard ratio infromation(Estimated HR, LowerCI and UpperCI), and number of cross validation respectively. This contains the estimated HR on test data and dimension reduction method is PLS.

Nmets The number of metabolites in the reduced matrix

Ncv The number of cross validation done

Top A sequence of top k metabolites considered. Default is Top=seq(5,100,by=5)

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

See Also

CVPcaPls, SurvPcaClass, SurvPlsClass

Examples

```
## GENERATE SOME METABOLIC SURVIVAL DATA WITH PROGNOSTIC FACTORS
Data<-MSData(nPatients=100,nMet=150,Prop=0.5)

## FIRST IS THE NETABOLITE BY METABOLITE ANALYSIS
w = CVMetSpecificCoxPh(Fold=3,Survival=Data$Survival,
```

```
Mdata=t(Data$Mdata),Censor= Data$Censor,Reduce=TRUE,
Select=150,Prognostic=Data$Prognostic,Quantile = 0.5,Ncv=3)

## USING THE FUNCTION
Result = CVSimet(w, Top = seq(5, 100, by = 5), Survival=Data$Survival,
 Censor=Data$Censor, Prognostic = Data$Prognostic)

## GET THE CLASS OF THE OBJECT
class(Result)     # A "cvsim" Class

##  METHOD THAT CAN BE USED FOR THE RESULT
show(Result)
summary(Result)
plot(Result, type =2)
```

`CVSimet` *Cross validation for sequentially increases metabolites*

Description

This function does cross validation for the metabolite by metabolite analysis while sequentially increasing the number of metabolites as specified.

Usage

```
CVSimet(Object, Top = seq(5, 100, by = 5), Survival, Censor, Prognostic = NULL)
```

Arguments

`Object`	An object of class `cvmm`
`Top`	The Top k number of metabolites to be used
`Survival`	A vector of survival time with length equals to number of subjects
`Censor`	A vector of censoring indicator
`Prognostic`	A dataframe containing possible prognostic(s) factor and/or treatment effect to be used in the model.

Details

This function firstly processes the cross validation for the metabolite by metabolite analysis results, and then sequentially considers top k metabolites. The function recompute first PCA or PLS on train data and estimate risk scores on both test and train data only on the metabolite matrix with top k metabolites. Patients are then classified as having low or high risk based on the test data where the cutoff used is median of the risk score. The process is repeated for each top K metabolite sets.

Value

A object of class cvsim is returned with the following values

HRpca	A 3-way array in which first, second, and third dimensions correspond to number of metabolites, Hazard ratio infromation(Estimated HR, LowerCI and UpperCI), and number of cross validation respectively. This contains the estimated HR on test data and dimension reduction method is PCA.
HRpls	A 3-way array in which first, second, and third dimensions correspond to number of metabolites, Hazard ratio infromation(Estimated HR, LowerCI and UpperCI), and number of cross validation respectively. This contains the estimated HR on test data and dimension reduction method is PLS.
Nmets	The number of metabolites in the reduced matrix
Ncv	The number of cross validation done
Top	A sequence of top k metabolites considered. Default is Top=seq(5,100,by=5)

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

See Also

MSpecificCoxPh

Examples

```
## FIRSTLY SIMULATING A METABOLIC SURVIVAL DATA
Data = MSData(nPatients = 100, nMet = 150, Prop = 0.5)

## GETTING THE cvmm OBJECT
Result = CVMetSpecificCoxPh(Fold=3,Survival=Data$Survival,
Mdata=t(Data$Mdata),Censor= Data$Censor,Reduce=TRUE,Select=150,
Prognostic=Data$Prognostic,Quantile = 0.5,Ncv=3)

## USING THE FUNCTION
 Result2 = CVSimet(Result, Top = seq(5, 100, by = 5), Data$Survival,
 Data$Censor,Prognostic = Data$Prognostic)

## GET THE CLASS OF THE OBJECT
class(Result2)     # An "cvsim" Class
```

DataHR *Survival and Prognostic Data .*

Description

A dataset containing the riskscore, survival parameters (Overall survival and censoring indicator) and other pronostic factors of 149 subjects.

Usage

```
data(DataHR)
```

Format

A data frame with 149 rows and 5 variables:

Riskscore Riskscores of the subjects

Survival Overall survival of the subjects

Censor Censoring indicator for all the patients; 1= Dead and 0 = Alive

Gender The first prognostic factor which is the gender of all the patients; 1=Male and 0 = Female

Stage The second prognostic factor which is the cancer stage of all the patients; 1= Early stage and 0= Advanced stage ...

Source

https://bmccancer.biomedcentral.com/articles/10.1186/s12885-018-4755-1

Examples

```
data(DataHR)
summary(DataHR[,1:2])
```

DistHR *Null Distribution of the Estimated HR*

Description

This function generates the null distribution of the HR by permutation approach. Several ways of permutation setting can be implemented. That is, function can be used to generate null distributions for four different validation schemes, PLS based, PCA based, Majority votes based and Lasso based.

Usage

```
DistHR(
  Survival,
  Censor,
  Mdata,
  Prognostic = NULL,
  Quantile = 0.5,
  Reduce = FALSE,
  Select = 15,
  nperm = 100,
  case = 2,
  Validation = c("PLSbased", "PCAbased", "L1based", "MVbased")
)
```

Arguments

Survival	A vector of survival time with length equals to number of subjects
Censor	A vector of censoring indicator
Mdata	A large or small metabolic profile matrix. A matrix with metabolic profiles where the number of rows should be equal to the number of metabolites and number of columns should be equal to number of patients.
Prognostic	A dataframe containing possible prognostic(s) factor and/or treatment effect to be used in the model.
Quantile	The cut off value for the classifier, default is the median cutoff
Reduce	A boolean parameter indicating if the metabolic profile matrix should be reduced, default is TRUE and larger metabolic profile matrix is reduced by supervised pca approach and first pca is extracted from the reduced matrix to be used in the classifier.
Select	Number of metabolites (default is 15) to be selected from supervised PCA. This is valid only if the argument Reduce=TRUE
nperm	Number of permutations to be used and default 100
case	There are seven different ways on how to call this argument: 1. Permute survival only. 2. Permute survival and rows of data frame of the prognostic factors. 3. Permute survival, rows of data frame of the prognostic factors, columns of metabolite matrix independently. 4. Permute metabolite matrix only.
Validation	There are four different validation schemes where the null distribution can be estimated. That is c("PLSbased","PCAbased","L1based","MVbased").

Details

This function generates the null distribution of the HR by permutation approach either using a large metabolite matrix or a reduced version by supervised pca approach. Several ways of permutation setting can be implemented. That is, the function can be used to generate null distributions for four

different validation schemes which are PLS based, PCA based, Majority votes based and Lasso based. Note this function internally calls function SurvPcaClass, SurvPlsClass, Majorityvotes, and Lasoelacox.

Value

A object of class perm is returned with the following values

HRobs	Estimated HR for low risk group on the original data
HRperm	Estimated HR for low risk group on the permuted data
nperm	Number of permutations carried out
Validation	The validation scheme that was used

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

See Also

coxph, EstimateHR, SurvPcaClass, SurvPlsClass, Majorityvotes, Lasoelacox, EstimateHR, Lasoelacox

Examples

```
## FIRSTLY SIMULATING A METABOLIC SURVIVAL DATA
Data = MSData(nPatients = 100, nMet = 150, Prop = 0.5)

## USING THE FUNCTION
Example <- DistHR(Survival = Data$Survival,Mdata = t(Data$Mdata),
Censor = Data$Censor,Reduce=FALSE,Select=15,Prognostic=Data$Prognostic,
Quantile = 0.5, nperm=10, case=2, Validation=c("L1based"))
```

EstimateHR *Classification, Survival Estimation and Visualization*

Description

The Function classifies subjects into low and high risk group using the risk scores based on the cut-off percentile specified. It also visualize survival fit along with HR estimates.

Usage

```
EstimateHR(
  Risk.Scores,
  Data.Survival,
  Prognostic = NULL,
  Plots = FALSE,
  Quantile = 0.5
)
```

Arguments

Risk.Scores	A vector of risk scores with size equals to number of subjects
Data.Survival	A dataframe in which the first column is the survival time and the second column is the Censoring indicator for each subject.
Prognostic	A dataframe containing possible prognostic(s) factor and/or treatment effect
Plots	A boolean parameter indicating if plots should be shown. Default is FALSE
Quantile	The cut off value for the classifier, default is the median cutoff

Details

The risk scores obtained using the signature is used to generate the risk group by dividing subjects into low and high risk group. A Cox model is then fitted with the risk group as covariate in the presence or absence of prognostic factors and or treatment effect. The extent of survival in the risk groups is known.

Value

An object of is returned, which is a list with the results of the cox regression and some informative plot concerning survival of the risk group.

SurvResult	The cox proportional regression result
Riskgroup	The riskgroup based on the riskscore and the cut off value and length is equal to number of subjects
KMplot	The Kaplan-Meier survival plot of the riskgroup
SurvBPlot	The distribution of the survival in the riskgroup

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

See Also

coxph

Examples

```
### Classification and estimating with prognostic factors
data(DataHR)
Result = EstimateHR(Risk.Scores=DataHR[,1],Data.Survival=DataHR[,2:3]
,Prognostic=DataHR[,4:5],Plots=FALSE,Quantile=0.50)

### Classification and estimating without prognostic factors
data(DataHR)
Result = EstimateHR(Risk.Scores=DataHR[,1],Data.Survival=DataHR[,2:3]
,Prognostic=NULL,Plots=FALSE,Quantile=0.50)
```

---

fcv *Constructor for the fcv class*

Description

Constructor for the fcv class

Usage

```
fcv(Runtime, Fold, Ncv, Nicv, TopK, HRInner, HRTest, Weight)
```

Arguments

Runtime	A vector of runtime for each iteration measured in seconds.
Fold	Number of folds used.
Ncv	Number of outer cross validations used.
Nicv	Number of inner cross validations used.
TopK	The Top metabolites used
HRInner	A 3-way array in which first, second, and third dimensions correspond to Nicv, 1, and Ncv respectively. This contains estimated HR for low risk group on the out of bag data.
HRTest	A matrix of survival information for the test dataset based on the out of bag data. It has three columns representing the estimated HR, the 95% lower confidence interval and the 95% upper confidence interval.
Weight	A matrix with columns equals number of TopK metabolites and rows Ncv. Note that Weights are estimated as colMeans of coefficients matrix return from the inner cross validations.

Value

object of class fcv

fcv-class *The fcv Class.*

Description

The fcv Class.

Slots

`Runtime` A vector of runtime for each iteration measured in seconds.

`Fold` Number of folds used.

`Ncv` Number of outer cross validations used.

`Nicv` Number of inner cross validations used.

`TopK` The Top metabolites used

`HRInner` A 3-way array in which first, second, and third dimensions correspond to Nicv, 1, and Ncv respectively. This contains estimated HR for low risk group on the out of bag data.

`HRTest` A matrix of survival information for the test dataset based on the out of bag data. It has three columns representing the estimated HR, the 95% lower confidence interval and the 95% upper confidence interval.

`Weight` A matrix with columns equals number of TopK metabolites and rows Ncv. Note that Weights are estimated as colMeans of coefficients matrix return from the inner cross validations.

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

See Also

CVLasoelacox, EstimateHR, glmnet, Lasoelacox

Examples

```
## GENERATE SOME METABOLIC SURVIVAL DATA WITH PROGNOSTIC FACTORS
Data<-MSData(nPatients=100,nMet=150,Prop=0.5)

## USE THE FUNCTION
Eg = Icvlasoel(Data$Survival, Data$Censor, Data$Prognostic,
t(Data$Mdata), Fold = 3,Ncv = 5, Nicv = 7, Alpha = 1,
TopK = colnames(Data$Mdata[,80:100]), Weights = FALSE)

## GET THE CLASS OF THE OBJECT
class(Eg)     # An "fcv" Class
```

```
##  METHOD THAT CAN BE USED FOR THIS CLASS
show(Eg)
summary(Eg)
plot(Eg, type =1)
```

`Icvlasoel` *Inner and Outer Cross Validations for Lasso Elastic Net Survival predictive models and Classification*

Description

The function does cross validation for Lasso, Elastic net and Ridge regressions models based on fixed or top selected metabolites from `CVLasoelacox` with classifier validated on a independent sample for the survial analysis and classification. The survival analysis is based on the selected metabolites in the presence or absene of prognostic factors.

Usage

```
Icvlasoel(
  Survival,
  Censor,
  Prognostic = NULL,
  Mdata,
  Fold = 3,
  Ncv = 50,
  Nicv = 100,
  Alpha = 0.1,
  TopK,
  Weights = FALSE
)
```

Arguments

`Survival`	A vector of survival time with length equals to number of subjects
`Censor`	A vector of censoring indicator
`Prognostic`	A dataframe containing possible prognostic(s) factor and/or treatment effect to be used in the model.
`Mdata`	A large or small metabolic profile matrix. A matrix with metabolic profiles where the number of rows should be equal to the number of metabolites and number of columns should be equal to number of patients.
`Fold`	number of folds to be used for the cross validation. Its value ranges between 3 and the numbe rof subjects in the dataset
`Ncv`	Number of validations to be carried out. The default is 25.
`Nicv`	Number of validations to be carried out for the inner loop. The default is 5.

Alpha	The mixing parameter for glmnet (see glmnet). The range is 0<= Alpha <= 1. The Default is 1
TopK	Top list of metabolites. Usually this can be mostly selected metabolites by function CVLasoelacox.
Weights	A logical flag indicating if a fixed or non-fixed weights should be used during the classifier evaluations. Default is FALSE.

Details

The function does cross validation for Lasso, Elastic net and Ridge regressions models based on fixed or top selected metabolites from CVLasoelacox with classifier validated on a independent sample for the survial analysis and classification. The survival analysis is based on the selected metabolites in the presence or absene of prognostic factors. The classifier is built on the weights obtain from the inner cross validations results and it is tested on out-of-bag data. These weights can be fixed or can be updated at each outer iteration. If weights are not fixed then patients are classified using majority votes. Otherwise, weights obtained from the inner cross validations are summarized by mean weights and used in the classifier. Inner cross validations are performed by calling to function CVLasoelacox. Hazard ratio for low risk group is estimated using out-of-bag data.

Value

A object of class fcv is returned with the following values

Runtime	A vector of runtime for each iteration measured in seconds.
Fold	Number of folds used.
Ncv	Number of outer cross validations used.
Nicv	Number of inner cross validations used.
TopK	The Top metabolites used
HRInner	A 3-way array in which first, second, and third dimensions correspond to Nicv, 1, and Ncv respectively. This contains estimated HR for low risk group on the out of bag data.
HRTest	A matrix of survival information for the test dataset based on the out of bag data. It has three columns representing the estimated HR, the 95% lower confidence interval and the 95% upper confidence interval.
Weight	A matrix with columns equals number of TopK metabolites and rows Ncv. Note that Weights are estimated as colMeans of coefficients matrix return from the inner cross validations.

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

See Also

CVLasoelacox, EstimateHR, glmnet, Lasoelacox

Examples

```
## FIRSTLY SIMULATING A METABOLIC SURVIVAL DATA
Data = MSData(nPatients = 100, nMet = 150, Prop = 0.5)

## USING THE FUNCTION
Results = Icvlasoel(Data$Survival, Data$Censor, Data$Prognostic,
t(Data$Mdata), Fold = 3,Ncv = 5, Nicv = 7, Alpha = 1,
TopK = colnames(Data$Mdata[,80:100]), Weights = FALSE)

## NUMBER OF Outer CV
Results@Ncv
## NUMBER OF Inner CV
Results@Nicv

## HR of low risk group for the Inner CV
Results@HRInner

## HR of low risk group for the out of bag dataset
Results@HRTest

## The weight for the analysis
Results@Weight
```

`Lasoelacox` *Wapper function for glmnet*

Description

The function uses the glmnet function to firstly do the variable selection either with Lasso, Elastic net or ridge regressions before the survial analysis. The survival analysis is based on the selected metabolites in the presence or absence of prognostic factors.

Usage

```
Lasoelacox(
  Survival,
  Censor,
  Mdata,
  Prognostic,
  Quantile = 0.5,
  Metlist = NULL,
  Plots = FALSE,
  Standardize = TRUE,
  Alpha = 1,
  Fold = 4,
  nlambda = 100
)
```

Arguments

Survival	A vector of survival time with length equals to number of subjects
Censor	A vector of censoring indicator
Mdata	A large or small metabolic profile matrix. A matrix with metabolic profiles where the number of rows should be equal to the number of metabolites and number of columns should be equal to number of patients.
Prognostic	A dataframe containing possible prognostic(s) factor and/or treatment effect to be used in the model.
Quantile	The cut off value for the classifier, default is the median cutoff
Metlist	A list of metabolites to be considered in the model usually smaller than the metabolites in the Mdata . Default is to use all metabolites available
Plots	A boolean parameter indicating if plots should be shown. Default is FALSE. If TRUE, the first plot is the partial likelihood deviance against the logarithmn of each lambda while the second is the coefficients versus the lamdas
Standardize	A Logical flag for the standardization of the metabolite matrix, prior to fitting the model sequence. The coefficients are always returned on the original scale. Default is standardize=TRUE.
Alpha	The mixing parameter for glmnet (see glmnet). The range is 0<= Alpha <= 1. The Default is 1
Fold	number of folds to be used for the cross validation. Its value ranges between 3 and the numbe rof subjects in the dataset
nlambda	The number of lambda values - default is 100 as in glmnet.

Details

This is a wrapper function for glmnet and it fits models using either Lasso, Elastic net and Ridge regressions. This is done in the presence or absence of prognostic factors. The prognostic factor when avaialable will always be forced to be in the model so no penalty for it. Optimum lambda will be used to select the non-zero shrinkage coefficients, the nonzero selected metabolites will thus be used in the survival analysis and in calculation of the risk scores.

Value

A object is returned with the following values

Coefficients.NonZero	The coefficients of the selected metabolites
Selected.Mets	The selected metabolites
n	The number of selected metabolites
Risk.scores	The risk scores of the subjects
Risk.group	The risk classification of the subjects based on the specified quantile
SurvFit	The cox analysis of the riskgroup based on the selected metabolites and the prognostic factors
Select	A Boolean argument indicating if there was selection or not

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

See Also

coxph, EstimateHR, glmnet,

Examples

```
## FIRSTLY SIMULATING A METABOLIC SURVIVAL DATA
Data = MSData(nPatients = 100, nMet = 150, Prop = 0.5)

## USING THE FUNCTION
Results = Lasoelacox(Survival=Data$Survival, Censor=Data$Censor,
Mdata=t(Data$Mdata), Prognostic = Data$Prognostic, Quantile = 0.5,
Metlist = NULL, Plots = FALSE, Standardize = TRUE, Alpha = 1)

## VIEW THE SELECTED METABOLITES
Results$Selected.mets
## NUMBER OF SELECTED METABOLITES
Results$n

## VIEW THE CLASSIFICATION GROUP OF EACH SUBJECT
Results$Risk.Group

## VIEW THE SURVIVAL ANALYSIS RESULT
Results$SurvFit

## TO CHECK IF THERE WAS ANY SELECTION
Results$Select
```

Majorityvotes *Classifiction for Majority Votes*

Description

The Function fits cox proportional hazard model and does classification based on the majority votes.

Usage

```
Majorityvotes(Result, Prognostic, Survival, Censor, J = 1)
```

Arguments

Result	An object obtained from the metabolite specific analysis (MSpecificCoxPh) which is of class "ms"

Prognostic	A dataframe containing possible prognostic(s) factor and/or treatment effect to be used in the model.
Survival	A vector of survival time with length equals to number of subjects
Censor	A vector of censoring indicator
J	The jth set of patients required for the visualization. The default is J=1 which is the first set of patients. For visualization, J should be less than the number of patients divided by 25

Details

The Function fits cox proportional hazard model and does classification based on the majority votes while estimating the Hazard ratio of the low risk group. The function firstly count the number of low risk classification for each subject based on the metabolite specific analysis which determines the majority votes. In addition, It visualizes the metabolic specific calssification for the subjects. 25 subjects is taken for visualization purpose.

Value

A list is returned with the following values

Model.result	The cox proportional regression result based on the majority vote classification
N	The majority vote for each subject
Classif	The majority vote classification for each subjects
Group	The classification of the subjects based on each metabolite analysis

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

References

Hastie T, Tibshirani R, Friedman J (2001). *The elements of statistical learning: data mining, inference, and prediction: with 200 full-color illustrations*. New York: Springer-Verlag.

See Also

MSpecificCoxPh, coxph, EstimateHR

Examples

```
## FIRSTLY SIMULATING A METABOLIC SURVIVAL DATA
Data = MSData(nPatients = 100, nMet = 150, Prop = 0.5)

## RUNNING THE METABOLITE SPECIFIC FUNCTION
Example1 = MSpecificCoxPh(Survival = Data$Survival,
Mdata = t(Data$Mdata), Censor = Data$Censor, Reduce = FALSE,
Select = 15,Prognostic = Data$Prognostic, Quantile = 0.5)
```

```
## USING THE FUNCTION
Result2 = Majorityvotes(Example1,Data$Prognostic, Data$Survival,Data$Censor,J=2)

## THE SURVIVAL ANALYSIS FOR MAJORITY VOTE RESULT
 Result2$Model.result

### THE MAJORITY VOTE FOR EACH SUBJECT
Result2$N

### THE MAJORITY VOTE CLASSIFICATION FOR EACH SUBJECT
Result2$Classif

### THE GROUP FOR EACH SUBJECT BASED ON THE METABOLITE SPPECIFIC ANALYSIS
Result2$Group
```

`MetabolicSurv` *MetabolicSurv: A biomarker validation approach for predicting survival using metabolic signature.*

Description

This package develope biomarker signature for metabolic data. It contains a set of functions and cross validation methods to validate and select biomarkers when the outcome of interest is survival. The package can handle prognostic factors and mainly metabolite matrix as input, the package can served as biomarker validation tool.

MetabolicSurv functions

1. It can be used with any form of high dimensional/omics data such as: Metabolic data, Gene expression matrix, incase you dont have a data it can simulate hypothetical scinerio of a high dimensional data based on the desired biological parameters
2. It developed any form of signature from the high dimensional data to be used for other purpose
3. It also employs data reduction techniques such as PCA, PLS and Lasso
4. It classifies subjects based on the signatures into Low and high risk group
5. It incorporate the use of subject prognostic information for the to enhance the biomarker for classification
6. It gives information about the survival rate of subjects depending on the classification

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

MetFreq *Frequency of Selected Metabolites from the LASSO, Elastic-net Cross-Validation*

Description

The function selects the frquency of selection from the shrinkage method (LASSO, Elastic-net) based on cross validation, that is the number of times each metabolite occur during the cross-validation process. In case of large metabolomic matrix then the N argument can be used to select metabolites occurence at a particular frequency.

Usage

```
MetFreq(Object, TopK = NULL, N = 3)
```

Arguments

Object	An object of class cvle returned from the function CVLasoelacox.
TopK	The number of Top K metabolites (5 by default) to be displayed in the frequency of selection graph.
N	The metqbolites with the specified frequency should be displayed in the frequency of selection graph.

Details

This function outputs the mostly selected metabolites during the LASSO and Elastic-net cross validation. Selected top metabolites are ranked based on frequency of selection and also a particular frequency cqn be selected. In addition, it visualizes the selected top metabolites based on the minimum frequency specified.

Value

A vector of metabolites and their frequency of selection. Also, a graphical representation is displayed.

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

See Also

cvmm, coxph, EstimateHR,CVLasoelacox

Examples

```
## FIRSTLY SIMULATING A METABOLIC SURVIVAL DATA
Data = MSData(nPatients = 100, nMet = 150, Prop = 0.5)

## CROSS-VALIDATION FOR LASSO AND ELASTIC-NET
Result = CVLasoelacox(Survival = Data$Survival,
Censor = Data$Censor, Mdata = t(Data$Mdata),
Prognostic = Data$Prognostic, Quantile = 0.5,
Metlist = NULL,Standardize = TRUE, Reduce=FALSE, Select=15,
Alpha = 1,Fold = 4,Ncv = 10,nlambda = 100)

## CONFIRMING THE CLASS
class(Result)

## USING THE FUNCTION
MetFreq(Result,TopK = 5, N=5)
```

---

ms *The ms class*

Description

The ms class

Constructor for the ms class

Usage

```
ms(Result, HRRG, Group, Metnames)

ms(Result, HRRG, Group, Metnames)
```

Arguments

Result	A list of dataframes of each output object of coxph for the metabolites.
HRRG	A dataframe with estimated metabolite-specific HR for low risk group and 95 percent CI.
Group	A matrix of the classification group a subject belongs to for each of the metabolite analysis. The metabolites are on the rows and the subjects are the columns
Metnames	The names of the metabolites for the analysis

Value

object of class ms

Slots

`Result` A list of dataframes of each output object of coxph for the metabolites.

`HRRG` A dataframe with estimated metabolite-specific HR for low risk group and 95 percent CI.

`Group` A matrix of the classification group a subject belongs to for each of the metabolite analysis. The metabolites are on the rows and the subjects are the columns

`Metnames` The names of the metabolites for the analysis

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

See Also

MSpecificCoxPh

Examples

```
## GENERATE SOME METABOLIC SURVIVAL DATA WITH PROGNOSTIC FACTORS
Data<-MSData(nPatients=100,nMet=150,Prop=0.5)

## DO THE METABOLITE BY METABOLITE ANALYSIS
Eg = MSpecificCoxPh(Survival=Data$Survival, Mdata=t(Data$Mdata),
Censor=Data$Censor, Reduce = FALSE, Select = 15,
Prognostic=Data$Prognostic, Quantile = 0.5)

## GET THE CLASS OF THE OBJECT
class(Eg)     # An "ms" Class

##  METHOD THAT CAN BE USED FOR THIS CLASS
show(Eg)
summary(Eg)
plot(Eg)
```

MSData *Generate Artificial Metabolic Survival Data*

Description

The Function generates metabolic profile and survival dataset of any number of patients and also their survival information.

Usage

```
MSData(nPatients = 100, nMet = 150, Prop = 0.5)
```

Arguments

nPatients	The number of patients
nMet	The number of metabolites
Prop	The proportion of patients having low risk

Details

The function generates the metabolic profile where small set of metabolites (30) are informative and rest of them are set as noisy metabolites. Also, the survival time and Censoring information are generated based on first right singular vectors of svd of the metabolic profile matrix. It also generates other prognostic factors such as Age, Stage and Gender which are slightly correlated with survival time.

Value

An object of class list is returned with the following items .

Censor	The censoring/event indicator
Survival	The Survival time
Met.names	The vector of metabolites name
Mdata	The metabolic profile matrix
Prognostic	A data frame with prognostic factors.

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

See Also

coxph

Examples

```
#GENERATE SOME METABOLIC SURVIVAL DATA WITH PROGNOSTIC FACTORS

Data<-MSData(nPatients=100,nMet=150,Prop=0.5)

SurvTime<-Data$Survival
Censor<-Data$Censor
ProgFact<-Data$Prognostic
MetData<-Data$Mdata
Metnames<-Data$Met.names
```

`MSpecificCoxPh` *Metabolite by metabolite Cox proportional analysis*

Description

The Function fits cox proportional hazard model and does classification for each metabolite

Usage

```
MSpecificCoxPh(
  Survival,
  Mdata,
  Censor,
  Reduce = FALSE,
  Select = 15,
  Prognostic = NULL,
  Quantile = 0.5
)
```

Arguments

`Survival`	A vector of survival time with length equals to number of subjects
`Mdata`	A large or small metabolic profile matrix. A matrix with metabolic profiles where the number of rows should be equal to the number of metabolites and number of columns should be equal to number of patients.
`Censor`	A vector of censoring indicator
`Reduce`	A boolean parameter indicating if the metabolic profile matrix should be reduced, default is TRUE and larger metabolic profile matrix is reduced by supervised pca approach and first pca is extracted from the reduced matrix to be used in the classifier.
`Select`	Number of metabolites (default is 15) to be selected from supervised PCA. This is valid only if the argument Reduce=TRUE
`Prognostic`	A dataframe containing possible prognostic(s) factor and/or treatment effect to be used in the model.
`Quantile`	The cut off value for the classifier, default is the median cutoff

Details

This function fits metabolite by metabolite Cox proportional hazard model and perform the classification based on a specified quantile. Risk score will be been estimated using a single metabolite. Function is useful for majority vote classification method and metabolite by metabolite analysis and also for top K metabolites.

Value

A object of class ms is returned with the following values

Result	The cox proportional regression result for each metabolite
HRRG	The hazard ratio statistics (Hazard-ratio, Lower confidence interval and upper confidence interval) of the riskgroup based on the riskscore and the cut off value for each metabolite
Group	The classification of the subjects based on each metabolite analysis
Metnames	The names of the metabolites for the analysis

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

See Also

coxph, EstimateHR

Examples

```
## FIRSTLY SIMULATING A METABOLIC SURVIVAL DATA
Data = MSData(nPatients = 100, nMet = 150, Prop = 0.5)

## USING THE FUNCTION
Example1 = MSpecificCoxPh(Survival = Data$Survival,
Mdata = t(Data$Mdata), Censor = Data$Censor, Reduce = FALSE,
Select = 15,Prognostic = Data$Prognostic, Quantile = 0.5)

## KNOWLING THE CLASS OF THE OUTPUT
class(Example1)

## EXTRACTING THE COMPONENT OF THE FUNCTION
### HAZARD RATIO INFORMATION FOR EACH METABOLITES
Example1@HRRG

### COX MODEL RESULT FOR EACH METABOLITES
Example1@Result

### CLASSIFICATION FOR EACH METABOLITES
Example1@Group
```

perm *Constructor for the perm class*

Description

Constructor for the perm class

Usage

```
perm(HRobs, HRperm, nperm, Validation)
```

Arguments

HRobs	Estimated HR for low risk group on the original data.
HRperm	Estimated HR for low risk group on the permuted data
nperm	Number of permutations carried out.
Validation	The validation scheme that was used.

Value

object of class perm

perm-class *The perm Class.*

Description

The perm Class.

Slots

HRobs Estimated HR for low risk group on the original data.

HRperm Estimated HR for low risk group on the permuted data

nperm Number of permutations carried out.

Validation The validation scheme that was used.

Note

The first, third and last vertical line on the plot are the lower, median and upper CI of the permuted data estimated HR while the red line is the estimated HR of the original data

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

See Also

DistHR, EstimateHR, SurvPcaClass, SurvPlsClass, Majorityvotes, Lasoelacox, EstimateHR, Lasoelacox

Examples

```
## GENERATE SOME METABOLIC SURVIVAL DATA WITH PROGNOSTIC FACTORS
Data<-MSData(nPatients=100,nMet=150,Prop=0.5)

## USING THE FUNCTION
Example <- DistHR(Survival = Data$Survival,Mdata = t(Data$Mdata),
Censor = Data$Censor,Reduce=FALSE,Select=15,Prognostic=Data$Prognostic,
Quantile = 0.5, nperm=10, case=2, Validation=c("L1based"))

## GET THE CLASS OF THE OBJECT
class(Example)     # A "perm" Class

##  METHOD THAT CAN BE USED FOR THIS CLASS
show(Example)
summary(Example)
plot(Example)
```

---

plot,cvle,missing-method

Plot method for cvle class

Description

Plot method for cvle class

Usage

```
## S4 method for signature 'cvle,missing'
plot(x, y, type = 1, ...)
```

Arguments

x	A cvle class object
y	missing
type	Plot type. 1 distribution of the HR under training and test set. 2 HR vs number selected metabolites.
...	The usual extra arguments to generic functions — see plot.default

Value

Cross Valdiated Results for Lasso and Elastic Net based Predictive Metabolite plots

plot,cvmm,ANY-method *Plot method for cvmm class*

Description

Plot method for cvmm class

Usage

```
## S4 method for signature 'cvmm,ANY'
plot(x, y, which = 1, ...)
```

Arguments

x	A cvmm class object
y	missing
which	This specify which metabolite for which estimated HR information need to be visualized. By default results of the first metabolite is used.
...	The usual extra arguments to generic functions — see plot.default

Value

Cross Valdiated Metabolic Specific CoxPh plots

plot,cvmv,ANY-method *Plot method for cvmv class*

Description

Plot method for cvmv class

Usage

```
## S4 method for signature 'cvmv,ANY'
plot(x, y, ...)
```

Arguments

x	A cvmv class object
y	missing
...	The usual extra arguments to generic functions — see plot.default

Value

Cross validation for Majority Votes Based Classification Analysis plots

plot,cvpp,missing-method

Plot method for cvpp class

Description

Plot method for cvpp class

Usage

```
## S4 method for signature 'cvpp,missing'
plot(x, y, ...)
```

Arguments

x	A cvpp class object
y	missing
...	The usual extra arguments to generic functions — see plot.default

Value

Cross Validations for PCA and PLS based plots

plot,cvsim,missing-method

Plot method for cvsim class

Description

Plot method for cvsim class

Usage

```
## S4 method for signature 'cvsim,missing'
plot(x, y, type = 1, ...)
```

Arguments

x	A cvsim class object
y	missing
type	Plot type. 1 distribution of the HR under test For the Top K metabolites using PCA. 2 distribution of the HR under test For the Top K metabolites using PLS.
...	The usual extra arguments to generic functions — see plot.default

Value

Cross validation for sequentially increases metabolites plots

---

`plot,fcv,missing-method` *Plot method for fcv class*

Description

Plot method for fcv class

Usage

```
## S4 method for signature 'fcv,missing'
plot(x, y, type = 1, ...)
```

Arguments

`x`	A fcv class object
`y`	missing
`type`	Plot type. 1 is the distribution of the inner cross validated HR under test data for each outer iterations and estimated HR on the out of bag data are superimposed. 2 Estimated HR Density for low Risk Group.
`...`	The usual extra arguments to generic functions — see `plot.default`

Value

Inner and Outer Cross Validations for Lasso Elastic Net Survival predictive models and Classification plots

---

`plot,ms,ANY-method` *Plot method for ms class*

Description

Plot method for ms class

Usage

```
## S4 method for signature 'ms,ANY'
plot(x, y, ...)
```

Arguments

`x`	A ms class object
`y`	missing
`...`	The usual extra arguments to generic functions — see `plot.default`

Value

Metabolite by Metabolite CoxPh plots

`plot,perm,ANY-method` *Plot method for perm class*

Description

Plot method for perm class

Usage

```
## S4 method for signature 'perm,ANY'
plot(x, y, ...)
```

Arguments

`x`	A perm class object
`y`	missing
`...`	The usual extra arguments to generic functions — see `plot.default`

Value

Null Distribution of the Estimated HR plots

`show,cvle-method` *Show method for cvle class*

Description

Show method for cvle class

Usage

```
## S4 method for signature 'cvle'
show(object)
```

Arguments

`object`	An object of class cvle

Value

Cross Valdiated Results for Lasso and Elastic Net based Predictive Metabolite signature.

show,cvmm-method *Show method for cvmm class*

Description

Show method for cvmm class

Usage

```
## S4 method for signature 'cvmm'
show(object)
```

Arguments

object	An object of class cvmm

Value

Cross Valdiated Metabolic Specific CoxPh information

show,cvmv-method *Show method for cvmv class*

Description

Show method for cvmv class

Usage

```
## S4 method for signature 'cvmv'
show(object)
```

Arguments

object	An object of class cvmv

Value

Cross validation for Majority Votes Based Classification Analysis information

show,cvpp-method *Show method for cvpp class*

Description

Show method for cvpp class

Usage

```
## S4 method for signature 'cvpp'
show(object)
```

Arguments

object	An object of class cvpp

Value

CCross Validations for PCA and PLS based information

show,cvsim-method *Show method for cvsim class*

Description

Show method for cvsim class

Usage

```
## S4 method for signature 'cvsim'
show(object)
```

Arguments

object	An object of class cvsim

Value

Cross validation for sequentially increases metabolites information

show,fcv-method *Show method for fcv class*

Description

Show method for fcv class

Usage

```
## S4 method for signature 'fcv'
show(object)
```

Arguments

object	An object of class fcv

Value

Inner and Outer Cross Validations for Lasso Elastic Net Survival predictive models and Classification information

show,ms-method *Show method for ms class*

Description

Show method for ms class

Usage

```
## S4 method for signature 'ms'
show(object)
```

Arguments

object	An object of class ms

Value

Metabolite by Metabolite CoxPh Model and the number of metabolites used.

show,perm-method *Show method for perm class*

Description

Show method for perm class

Usage

```
## S4 method for signature 'perm'
show(object)
```

Arguments

object	An object of class perm

Value

Null Distribution of the Estimated HR information

SIMet *Sequential Increase in Metabolites for the PCA or PLS classifier*

Description

The Function fits cox proportional hazard model and does classification by sequentially increasing the metabolites using either PCA or PLS based on the topK metabolites specified.

Usage

```
SIMet(
  TopK = 15,
  Survival,
  Mdata,
  Censor,
  Reduce = TRUE,
  Select = 50,
  Prognostic = NULL,
  Plot = FALSE,
  DimMethod = c("PLS", "PCA"),
  ...
)
```

Arguments

TopK	Top K metabolites (15 by default) to be used in the sequential analysis.
Survival	A vector of survival time with length equals to number of subjects
Mdata	A large or small metabolic profile matrix. A matrix with metabolic profiles where the number of rows should be equal to the number of metabolites and number of columns should be equal to number of patients.
Censor	A vector of censoring indicator
Reduce	A boolean parameter indicating if the metabolic profile matrix should be reduced, default is TRUE and larger metabolic profile matrix is reduced by supervised pca approach and first pca is extracted from the reduced matrix to be used in the classifier.
Select	Number of metabolites to be selected from supervised PCA. This is valid only if the argument Reduce=TRUE
Prognostic	A dataframe containing possible prognostic(s) factor and/or treatment effect to be used in the model.
Plot	A boolean parameter indicating if Plot should be shown. Default is FALSE
DimMethod	Dimension reduction method which can either be PLS or PCA.
...	Additinal arguments for plotting and only valid if Plot=TRUE

Details

This function sequentially increase the number of top K metabolites to be used in the PCA or PLS methods in order to obtain the risk score. This function internally calls MSpecificCoxPh to rank the metabolites based on HR. Therefore metabolites can be ordered based on increasing order of the HR for low risk group. Thereafter, the function takes few top K (15 is the default) to be used in the sequential analysis.

Value

A list containing a data frame with estimated HR along with 95% CI at each TopK value for the sequential analysis.

Result	The hazard ratio statistics (HR, Lower confidence interval and upper confidence interval) of the lower riskgroup based for each sequential metabolite analysis
TopKplot	A graphical representation of the Result containing the hazard ratio statistics

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

References

Vinzi VE, Chin WW, Henseler J, Wang H (2010). *Handbook of Partial Least Squares: Concepts, Methods and Applications*, 1st edition. Springer Publishing Company, Incorporated.

Bair E, Hastie T, Debashis P, Tibshirani R (2006). "Prediction by supervised principal components." *American Statistics Association,*, **101**(473), 119–137.

See Also

coxph, EstimateHR, MSpecificCoxPh, SurvPcaClass, SurvPcaClass

Examples

```
## FIRSTLY SIMULATING A METABOLIC SURVIVAL DATA
Data = MSData(nPatients = 100, nMet = 150, Prop = 0.5)

## USING THE FUNCTION
Example1 = SIMet(TopK = 10, Survival=Data$Survival,
Mdata=t(Data$Mdata), Censor=Data$Censor, Reduce = TRUE,
Select = 50,Prognostic = Data$Prognostic, Plot = TRUE, DimMethod ="PLS")

## FOR THE HR STATISTICS
Example1$Result

## FOR THE GRAPHICAL OUTPUT
Example1$TopKplot
```

---

summary,cvle-method *Summary method for cvle class*

Description

Summary method for cvle class

Usage

```
## S4 method for signature 'cvle'
summary(object)
```

Arguments

object	An object of class cvle

Value

Cross Valdiated Results for Lasso and Elastic Net based Predictive Metabolite signature summary information

summary,cvmm-method *Summary method for cvmm class*

Description

Summary method for cvmm class

Usage

```
## S4 method for signature 'cvmm'
summary(object, which = 1)
```

Arguments

object	An object of class cvmm
which	This specify which metabolite for which estimated HR

Value

Cross Valdiated Metabolic Specific CoxPh summary

summary,cvmv-method *Summary method for cvmv class*

Description

Summary method for cvmv class

Usage

```
## S4 method for signature 'cvmv'
summary(object)
```

Arguments

object	An object of class cvmv

Value

Cross validation for Majority Votes Based Classification Analysis summary

summary,cvpp-method *Summary method for cvpp class*

Description

Summary method for cvpp class

Usage

```
## S4 method for signature 'cvpp'
summary(object)
```

Arguments

object	An object of class cvpp

Value

Cross Validations for PCA and PLS based summary

summary,cvsim-method *Summary method for cvsim class*

Description

Summary method for cvsim class

Usage

```
## S4 method for signature 'cvsim'
summary(object)
```

Arguments

object	An object of class cvsim

Value

Cross validation for sequentially increases metabolites summary

summary,fcv-method *Summary method for fcv class*

Description

Summary method for fcv class

Usage

```
## S4 method for signature 'fcv'
summary(object)
```

Arguments

object	An object of class fcv

Value

Inner and Outer Cross Validations for Lasso Elastic Net Survival predictive models and Classification summary

summary,ms-method *Summary method for ms class*

Description

Summary method for ms class

Usage

```
## S4 method for signature 'ms'
summary(object)
```

Arguments

object	An object of class ms

Value

Metabolite by Metabolite CoxPh summary information

summary,perm-method *Summary method for perm class*

Description

Summary method for perm class

Usage

```
## S4 method for signature 'perm'
summary(object)
```

Arguments

object	An object of class perm

Value

Null Distribution of the Estimated HR summary

SurvPcaClass *Survival PCA and Classification for metabolic data*

Description

The function performs principal component analysis (PCA) on Metabolomics matrix and fit Cox proportional hazard model with covariates using also the first PCA as covariates.

Usage

```
SurvPcaClass(
  Survival,
  Mdata,
  Censor,
  Reduce = TRUE,
  Select = 150,
  Prognostic = NULL,
  Plots = FALSE,
  Quantile = 0.5
)
```

Arguments

Survival	A vector of survival time with length equals to number of subjects
Mdata	A large or small metabolic profile matrix. A matrix with metabolic profiles where the number of rows should be equal to the number of metabolites and number of columns should be equal to number of patients.
Censor	A vector of censoring indicator
Reduce	A boolean parameter indicating if the metabolic profile matrix should be reduced, default is TRUE and larger metabolic profile matrix is reduced by supervised pca approach and first pca is extracted from the reduced matrix to be used in the classifier.
Select	Number of metabolites (default is 15) to be selected from supervised PCA. This is valid only if th argument Reduce=TRUE
Prognostic	A dataframe containing possible prognostic(s) factor and/or treatment effect to be used in the model.
Plots	A boolean parameter indicating if the plots should be shown. Default is FALSE
Quantile	The cut off value for the classifier, default is the median cutoff

Details

This function can also be used to perform the grid analysis where the grid will be several quantile values and default is 0.5 which is the median cut-off. This function can handle single and multiple metabolites. For larger Metabolomics matrix, this function will reduce larger Metabolomics matrix to smaller version using supervised pca approach and this is by default done and can be control by using the argument Reduce. Other prognostic factors can be included to the model.

Value

A object of class SurvPca is returned with the following values

Survfit	The cox proportional regression result using the first PCA
Riskscores	A vector of risk scores which is equal to the number of patents.
Riskgroup	The classification of the subjects based on the PCA into low or high risk group
pc1	The First PCA scores based on either the reduced Metabolite matrix or the full matrix
KMplot	The Kaplan-Meier survival plot of the riskgroup
SurvBPlot	The distribution of the survival in the riskgroup
Riskpca	The plot of Risk scores vs first PCA

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

References

Bair E, Hastie T, Debashis P, Tibshirani R (2006). "Prediction by supervised principal components." *American Statistics Association,*, **101**(473), 119–137.

See Also

coxph, EstimateHR, princomp, SurvPlsClass

Examples

```
## FIRSTLY SIMULATING A METABOLIC SURVIVAL DATA
Data = MSData(nPatients = 100, nMet = 150, Prop = 0.5)

## USING THE FUNCTION
Result = SurvPcaClass(Survival=Data$Survival, Mdata=t(Data$Mdata),
Censor=Data$Censor, Reduce = FALSE, Select = 150,
Prognostic = Data$Prognostic, Plots = FALSE, Quantile = 0.5)

## GETTING THE SURVIVAL REGRESSION OUTPUT
Result$SurvFit

## GETTING THE RISKSCORES
Result$Riskscores

### GETTING THE RISKGROUP
Result$Riskgroup

### OBTAINING THE FIRST PRINCIPAL COMPONENT SCORES
Result$pc1
```

SurvPlsClass *Survival PLS and Classification for metabolic data*

Description

The function performs partial least squares (PLS) and principal component regression on Metabolomics matrix and fit Cox proportional hazard model with covariates using the first PLS scores as covariates.

Usage

```
SurvPlsClass(
  Survival,
  Mdata,
  Censor,
  Reduce = TRUE,
  Select = 150,
  Prognostic = NULL,
```

```
  Plots = FALSE,
  Quantile = 0.5
)
```

Arguments

Survival	A vector of survival time with length equals to number of subjects
Mdata	A large or small metabolic profile matrix. A matrix with metabolic profiles where the number of rows should be equal to the number of metabolites and number of columns should be equal to number of patients.
Censor	A vector of censoring indicator
Reduce	A boolean parameter indicating if the metabolic profile matrix should be reduced, default is TRUE and larger metabolic profile matrix is reduced by supervised pca approach and first pca is extracted from the reduced matrix to be used in the classifier.
Select	Number of metabolites (default is 15) to be selected from supervised PCA. This is valid only if th argument Reduce=TRUE
Prognostic	A dataframe containing possible prognostic(s) factor and/or treatment effect to be used in the model.
Plots	A boolean parameter indicating if the plots should be shown. Default is FALSE
Quantile	The cut off value for the classifier, default is the median cutoff

Details

This function reduces larger metabolomics matrix to smaller version using supervised pca approach. The function performs the PLS on the reduced metabolomics matrix and fit Cox proportional hazard model with first PLS scores as a covariate afterwards. And classifier is then built based on the first PLS scores multiplied by its estimated regression coefficient. Patients are classified using median of the risk scores. The function can also perform grid analysis where the grid will be several quantiles but the default is median. This function can handle single and multiple metabolites. Prognostic factors can also be included to enhance classification.

Value

A object is returned with the following values

Survfit	The cox proportional regression result using the first PCA
Riskscores	A vector of risk scores which is equal to the number of patents.
Riskgroup	The classification of the subjects based on the PCA into low or high risk group
pc1	The First PCA scores based on either the reduced Metabolite matrix or the full matrix
KMplot	The Kaplan-Meier survival plot of the riskgroup
SurvBPlot	The distribution of the survival in the riskgroup
Riskpca	The plot of Risk scores vs first PCA

Author(s)

Olajumoke Evangelina Owokotomo, <olajumoke.owokotomo@uhasselt.be>

Ziv Shkedy

References

Bair E, Hastie T, Debashis P, Tibshirani R (2006). "Prediction by supervised principal components." *American Statistics Association,*, **101**(473), 119–137.

See Also

coxph, EstimateHR, plsr, SurvPcaClass

Examples

```
## FIRSTLY SIMULATING A METABOLIC SURVIVAL DATA
Data = MSData(nPatients = 100, nMet = 150, Prop = 0.5)

## USING THE FUNCTION
Result = SurvPlsClass(Survival=Data$Survival, Mdata=t(Data$Mdata),
Censor=Data$Censor, Reduce = FALSE, Select = 150,
Prognostic = Data$Prognostic, Plots = FALSE, Quantile = 0.5)

## GETTING THE SURVIVAL REGRESSION OUTPUT
Result$SurvFit

## GETTING THE RISKSCORES
Result$Riskscores

### GETTING THE RISKGROUP
Result$Riskgroup

### OBTAINING THE FIRST PRINCIPAL COMPONENT SCORES
Result$pc1
```

Index