Package ‘highMLR’

May 11, 2021

Title Feature Selection for High Dimensional Survival Data

Version 0.1.0

Date 2021-05-08

Depends R (>= 3.5.0)

Imports mlr3, mlr3proba, mlr3learners, survival, gtools, tibble, dplyr, utils, coxme, missForest

LazyData Yes

LazyDataCompression xz

ByteCompile Yes

Description Perform high dimensional Feature Selection in the presence of survival outcome. Based on Feature Selection method and different survival analysis, it will obtain the best markers with optimal threshold levels according to their effect on disease progression and produce the most consistent level according to those threshold values. The functions’ methodology is based on by Sonabhend et al (2021) <doi:10.1093/bioinformatics/btab039> and Bhattacharjee et al (2021) <arXiv:2012.02102>.

License GPL-3

Encoding UTF-8

NeedsCompilation no

Maintainer Atanu Bhattacharjee <atanustat@gmail.com>

RoxygenNote 7.1.1

Author Atanu Bhattacharjee [aut, cre, ctb], Gajendra K. Vishwakarma [aut, ctb], Souvik Banerjee [aut, ctb]

Repository CRAN

Date/Publication 2021-05-11 09:40:08 UTC

R topics documented:

hnscc .......................................................... 2
High dimensional head and neck cancer survival and gene expression data

ID "Column/Variable name" consisting id of subjects
Death "Column/Variable name" consisting survival event
OS "Column/Variable name" consisting duration of overall survival
PFS "Column/Variable name" consisting duration of progression free survival
Prog "Column/Variable name" consisting progression event
GJB1,...,HMGCS2 High dimensional covariates

Examples
data(hnscc)
mlclassCox

Applications of machine learning in survival analysis by prognostic classification of genes by CoxPH model.

Description
Applications of machine learning in survival analysis by prognostic classification of genes by CoxPH model.

Usage
mlclassCox(m, n, idSurv, idEvent, Time, s_ID, per = 20, fold = 3, data)

Arguments
- m: Starting column number from where high dimensional variates to be selected.
- n: Ending column number till where high dimensional variates to be selected.
- idSurv: "Column/Variable name" consisting duration of survival.
- idEvent: "Column/Variable name" consisting survival event.
- Time: "Column/Variable name" consisting Times of repeated observations.
- s_ID: "Column/Variable name" consisting unique identification for each subject.
- per: Percentage value for ordering, default=20.
- fold: Number of folds for re-sampling, default=3.
- data: High dimensional data containing survival observations with multiple covariates.

Value
A list of genes as per their classifications

- **GeneClassification**: List of genes classified using Cox proportional hazard model
- **GeneClassification$Positive_Gene**: Sublist of genes classified as positive genes
- **GeneClassification$Negative_Gene**: Sublist of genes classified as negative genes
- **GeneClassification$Volatile_Gene**: Sublist of genes classified as volatile genes

Result: A dataframe consisting threshold values with corresponding coefficients and p-values.

Examples

```r
data(srdata)
mlclassCox(m=50, n=59, idSurv="OS", idEvent="event", Time="Visit", s_ID="ID", per=20, fold=3, data=srdata)
```
mlclassKap

Applications of machine learning in survival analysis by prognostic classification of genes by Kaplan-Meier estimator.

Description
Applications of machine learning in survival analysis by prognostic classification of genes by Kaplan-Meier estimator.

Usage
mlclassKap(m, n, idSurv, idEvent, Time, s_ID, per = 20, fold = 3, data)

Arguments
m Starting column number from where high dimensional variates to be selected.
n Ending column number till where high dimensional variates to be selected.
idSurv "Column/Variable name" consisting duration of survival.
idEvent "Column/Variable name" consisting survival event.
Time "Column/Variable name" consisting timepoints of repeated observations.
s_ID "Column/Variable name" consisting unique identification for each subject.
per Percentage value for ordering, default=20.
fold Number of fold for resampling, default=3.
data High dimensional data containing survival observations and high dimensional covariates.

Value
A list of genes as per their classifications
GeneClassification List of genes classified using Cox proportional hazard model
GeneClassification$Positive_Gene Sublist of genes classified as positive genes
GeneClassification$Negative_Gene Sublist of genes classified as negative genes
GeneClassification$Volatile_Gene Sublist of genes classified as volatile genes
Result A dataframe consisting threshold values with corresponding coefficients and p-values.

Examples
##
mlclassKap(m=50,n=59,idSurv="OS",idEvent="event",Time="Visit",s_ID="ID",per=20,fold=3,data=srdata)
##
**Description**

This function extracts desired number of features based on minimum log-Loss function using Cox proportional hazard model as learner method on a high dimensional survival data.

**Usage**

```r
mlhighCox(cols, idSurv, idEvent, per = 20, fold = 3, data)
```

**Arguments**

- `cols` A numeric vector of column numbers indicating the features for which the log Loss functions are to be computed
- `idSurv` The name of the survival time variable
- `idEvent` The name of the survival event variable
- `per` Percentage of total features to be selected, default value 20
- `fold` An integer denoting number of folds in cross validation, default value 3
- `data` A data frame that contains the survival and covariate information for the subjects

**Details**

Performs feature Selection using Cox PH on high-dimensional data

Using the Cox proportional hazard model on the given survival data, this function selects the most significant feature based on a performance measure. The performance measure is considered as logarithmic loss function. It is defined as,

\[ L(f, t) = -\log(f(t)) \]

. The features with minimum log-loss function are extracted.

**Value**

A dataframe containing desired number of features and the corresponding log Loss function.

**Author(s)**

Atanu Bhattacharjee, Gajendra K. Vishwakarma & Souvik Banerjee

**References**

mlhighFrail

See Also
mlhighKap, mlhighFrail

Examples

```r
data(hnscc)
mlhighCox(cols=c(6:15), idSurv="OS", idEvent="Death", per=20, fold = 3, data=hnscc)
```

Description

This function extracts features based on minimum log-Loss function using Cox proportional hazard
model as learner method on a high dimensional survival data. For those genes, we obtain frailty
variances using CoxPH.

Usage

```r
mlhighFrail(
  cols, 
idSurv, 
idEvent, 
idFrail, 
dist = "gaussian", 
per = 20, 
fold = 3, 
data
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cols</td>
<td>A numeric vector of column numbers indicating the features for which the log Loss functions are to be computed</td>
</tr>
<tr>
<td>idSurv</td>
<td>The name of the survival time variable</td>
</tr>
<tr>
<td>idEvent</td>
<td>The name of the survival event variable</td>
</tr>
<tr>
<td>idFrail</td>
<td>The name of the frailty variable</td>
</tr>
<tr>
<td>dist</td>
<td>The name of the frailty distribution. Options are &quot;gamma&quot;, &quot;gaussian&quot; or &quot;t&quot;, default is &quot;gaussian&quot;</td>
</tr>
<tr>
<td>per</td>
<td>Percentage of features to be selected, default value 20</td>
</tr>
<tr>
<td>fold</td>
<td>An integer denoting number of folds in cross validation, default value 3</td>
</tr>
<tr>
<td>data</td>
<td>A data frame that contains the survival and covariate information for the subjects</td>
</tr>
</tbody>
</table>
mlhighHet

Details

Performs CoxPH frailty on high dimensional survival data

Using the Cox proportional hazard model on the given survival data, this function selects the most significant feature based on minimum logarithmic loss function. The logarithmic loss function is defined as,

\[ L(f, t) = -\log(f(t)) \]

After selecting the most significant features, a Cox proportional hazard frailty model is fitted on the selected features. The CoxPH frailty model is defined as,

\[ \lambda(t) = \lambda_0(t) \nu \exp X^T \beta \]

where \( \nu \) is called the frailty component. The variance of the frailty term is considered as the heterogeneity among the subjects or patients. The distribution of frailty component is considered as either Gaussian, Gamma or t distribution.

Value

A dataframe containing desired number of features with corresponding frailty variances.

Author(s)

Atanu Bhattacharjee, Gajendra K. Vishwakarma & Souvik Banerjee

See Also

mlhighHet, mlhighCox

Examples

data(hnscc)
mlhighFrail(cols=c(10:20), idSurv="OS", idEvent="Death", idFrail="ID", dist="gaussian", per=20, fold = 3, data=hnscc)

mlhighHet

mlhighHet Performs heterogeneity analysis in gene expression

Description

This function extracts features based on ML method, finds optimal cut-off values of features using sequential Cox PH model and obtain the most consistent level according to the cut-offs.

Usage

mlhighHet(cols, idSurv, idEvent, idFrail, num, fold = 3, data)
Arguments

cols A numeric vector of column numbers indicating the features for which the log Loss functions are to be computed
idSurv The name of the survival time variable
idEvent The name of the survival event variable
idFrail The name of the frailty variable
num Number of features to be selected
fold An integer denoting number of folds in cross validation, default value 3
data A data frame that contains the survival and covariate information for the subjects

Details

This function extracts features based on minimum log-Loss function using Cox proportional hazard model as learner method on a high dimensional survival data. For those selected genes, we obtain optimal cutoff values using minimum p-value in a Cox PH model. The Cox PH model is used sequentially for each combination of genes and all possible gene combinations are tested to obtain best possible combination with minimum BIC value. The subjects are classified according to different levels of those genes. Using a Cox PH frailty model, we obtain the most consistent level for which the frailty variance is minimum. The data is splited using cross validation technique. The performance measure is considered as logarithmic loss function. It is defined as:

\[ L(f, t) = -\log(f(t)) \]

The CoxPH frailty model is defined as,

\[ \lambda(t) = \lambda_0(t) \nu \exp X' \beta \]

where \( \nu \) is called the frailty. The variance of the frailty term is considered as the heterogeneity among the subjects or patients. Gaussian distribution with mean 0 is considered for the distribution of frailty component.

Value
dataframes containing optimal gene cutoff values and most consistent level according to those cutoffs with frailty variance.

Author(s)
Atanu Bhattacharjee, Gajendra K. Vishwakarma & Souvik Banerjee

See Also
mlhighCox, mlhighFrail

Examples

data(hnscc)
mlhighHet(cols=c(27:32), idSurv="OS", idEvent="Death", idFrail="ID", num=2, fold = 3, data=hnscc)
Description

This function extracts desired number of features based on minimum log-Loss function using Kaplan Meier model as learner method on a high dimensional survival data.

Usage

mlhighKap(cols, idSurv, idEvent, per = 20, fold = 3, data)

Arguments

cols A numeric vector of column numbers indicating the features for which the log Loss functions are to be computed
idSurv The name of the survival time variable
idEvent The name of the survival event variable
per Percentage of features to be selected, default value 20
fold An integer denoting number of folds in cross validation, default value 3
data A data frame that contains the survival and covariate information for the subjects

Details

Performs feature selection using Kaplan Meier method

Using the Kaplan Meier method on the given survival data, this function selects the most significant feature based on a performance measure. The performance measure is considered as logarithmic loss function. It is defined as,

\[ L(f, t) = -\log(f(t)) \]

The features with minimum log-loss function are extracted.

Value

A dataframe containing desired number of features based on minimum log Loss function

Author(s)

Atanu Bhattacharjee, Gajendra K. Vishwakarma & Souvik Banerjee

See Also

mlhighCox
Examples

```
data(hnscc)
mlhighKap(cols=c(6:15), idSurv="OS", idEvent="Death", per=20, fold = 3, data=hnscc)
```

---

**srdata**

*High dimensional protein gene expression data*

**Description**

High dimensional protein gene expression data

**Usage**

```
srdata
```

**Format**

A dataframe with 288 rows and 250 variables

- **ID** "Column/Variable name" consisting id of subjects
- **Visit** "Column/Variable name" consisting number of times observations recorded
- **event** "Column/Variable name" consisting survival event
- **OS** "Column/Variable name" consisting duration of overall survival
- **C6kine,.....,GFRalpha4** High dimensional covariates

**Examples**

```
data(srdata)
```
Index

* datasets
  * hnscc, 2
  * srdata, 10

hnscc, 2

mlclassCox, 3
mlclassKap, 4
mlhighCox, 5
mlhighFrail, 6
mlhighHet, 7
mlhighKap, 9

srdata, 10