Package ‘SIGN’

July 21, 2025

Type Package

Title Similarity Identification in Gene Expression

Version 0.1.0

Date 2018-12-02

Description Provides a classification framework to use expression patterns of pathways as features to identify similarity between biological samples. It provides a new measure for quantifying similarity between expression patterns of pathways.

Depends R(>= 3.4)

License GPL (>= 3)

Imports stats, utils, survcomp, survival, GSVA

Encoding UTF-8

LazyData true

NeedsCompilation yes

biocViews GeneExpression, Classification, Clustering, Survival

RoxygenNote 6.1.1

Author Seyed Ali Madani Tonekaboni [aut],
Gangesh Beri [aut],
Janosch Ortmann [aut],
Benjamin Haibe-Kains [aut, cre]

Maintainer Benjamin Haibe-Kains <`benjamin.haibe.kains@utoronto.ca`>

Repository CRAN

Date/Publication 2018-12-18 23:11:47 UTC

Contents

`BubbleSort` *BubbleSort is a function for calculating bubble sort correlation between two vectors*

Description

BubbleSort is a function for calculating bubble sort correlation between two vectors

Usage

```
BubbleSort(Vec1, Vec2)
```

Arguments

`Vec1`	Vector of values of 1st feature across samples
`Vec2`	Vector of values of 2nd feature across samples

Value

Bubble sort similarity between the two vectors

`EventRenaming` *EventRenaming is a function for changing annotation of censored samples to 0 and dead samples to 1 for survival analysis*

Description

EventRenaming is a function for changing annotation of censored samples to 0 and dead samples to 1 for survival analysis

Usage

```
EventRenaming(EventVec, Censored_Annot)
```

Arguments

`EventVec`	Status vector for all of the samples (patients) including both samples undergone an event or censored
`Censored_Annot`	Index of samples censored in the dataset

Value

Vector of events including 0 for censoring and 1 for death

`ExpPheno_Categorize` *ExpPheno_Categorize is a function for grouping samples based on their survival to 3 groups of poor, good, and intermediate*

Description

ExpPheno_Categorize is a function for grouping samples based on their survival to 3 groups of poor, good, and intermediate

Usage

```
ExpPheno_Categorize(ExpMeta_List, Time_ID, Event_ID, Mad_Factor,
  MinNum_ExClass, Expression_Log2 = FALSE)
```

Arguments

`ExpMeta_List`	List containing expression matrix and metadata matrix
`Time_ID`	Index of time to death in metadata matrix
`Event_ID`	Index of event in metadata matrix
`Mad_Factor`	Threshold of mad in time to death values to determine poor survival group

`MinNum_ExClass`	Minimum number of samples that has to be kept in poor and good group (if number of samples is lower than this threhold, more samples will be addedd in order of survival)
`Expression_Log2`	Parameter for gene expsression value transformation to logarithmic scale (log2(expression value+1))

Value

List of expression matrices, and time to event as well as event for the patients within each category of poor, intermediate or good survival

---

`ExpPhen_Matching` *ExpPhen_Matching is a function for matching samples between expression matrices and metadata matrix (clinical feature matrix)*

Description

ExpPhen_Matching is a function for matching samples between expression matrices and metadata matrix (clinical feature matrix)

Usage

```
ExpPhen_Matching(ExpMat, MetaMat, SamID_Meta)
```

Arguments

`ExpMat`	Matrix of expression of genes (samples in columns and genes in rows)
`MetaMat`	Matrix of clinical features (samples in columns)
`SamID_Meta`	Sample ID in MetaMat

Value

List of expression matrix and metadata of the clinical information after matching patiend IDs between the expression and clinical information matrices

---

`ExpPhen_Subdividing` *ExpPhen_Subdividing is a function for grouping samples based on a clinical feature available in metadata matrix (clinical feature matrix)*

Description

ExpPhen_Subdividing is a function for grouping samples based on a clinical feature available in metadata matrix (clinical feature matrix)

Usage

```
ExpPhen_Subdividing(ExpMeta_List, SubDiv_ID)
```

Arguments

`ExpMeta_List`	List containing expression matrix and metadata matrix
`SubDiv_ID`	Index of the target clinical feature in metadata matrix for samples grouping

Value

List of expression and clinical information of patients grouped based on the specified clinical feature

---

`GeneMatching` *GeneMatching is a function to remove uncommon genes between a list of expression matrices*

Description

GeneMatching is a function to remove uncommon genes between a list of expression matrices

Usage

```
GeneMatching(ExpList)
```

Arguments

`ExpList`	List of expression matrices

Value

List of expression matrices restricted to the common genes between them

---

`Genes_SimCal` *Genes_SimCal is a function to calculate similarity between a set of samples and 2 reference groups of samples*

Description

Genes_SimCal is a function to calculate similarity between a set of samples and 2 reference groups of samples

Usage

```
Genes_SimCal(ExpMat_Test, ExpMat_Ref1, ExpMat_Ref2, RefIDs, TestClassIter,
  SampleIter)
```

Arguments

`ExpMat_Test`	Expression matrix for the test samples for which SIGN will indetify the similarity with the 2 reference sataset
`ExpMat_Ref1`	Expression matrix for the 1st reference set fo samples
`ExpMat_Ref2`	Expression matrix for the 2nd reference set fo samples
`RefIDs`	Annotations corresponding to the 2 expression matrices (1st and 2nd names are associated with the 1st and 2nd expression matrix and)
`TestClassIter`	Index to be matched with RefIDs for removal of test samples from reference expression matrices
`SampleIter`	Index of samples in the test expression matrix exist in referencece expression matrix 1 or 2

Value

Vector of similarity between the target samples and the 2 reference sets

---

`GSVA_Calculation` *GSVA_Calculation is a function for Calculating correlation between expression level of pathways between 2 groups using GSVA*

Description

GSVA_Calculation is a function for Calculating correlation between expression level of pathways between 2 groups using GSVA

Usage

```
GSVA_Calculation(ExpMat1, ExpMat2, GeneVec, GeneSets,
  Name = "SampleComparison")
```

Arguments

`ExpMat1`	Expression matrix of genes in the 1st group of sampls
`ExpMat2`	Expression matrix of genes in the 2nd group of sampls
`GeneVec`	Name of genes in the same order as considered in ExpMat1 and ExpMat2
`GeneSets`	List of genes within pathways
`Name`	Name used for naming the columns of output matrix of correlation between the 2 groups

Value

Similarity of the pathway between the two expression matrices based on pearson correlation, bubble sort, and wilcoxon paaired rank test using GSVA enrichment scores of pathways

---

`Pathway_Grouping` *Pathway_Grouping is a function to make a pathway list from files containing genes within each pathway*

Description

Pathway_Grouping is a function to make a pathway list from files containing genes within each pathway

Usage

```
Pathway_Grouping(PathwayDir, Pattern)
```

Arguments

`PathwayDir`	Path of directory including the files of pathways
`Pattern`	Pattern should be used to select the files of pathway genes from PathwayDir

Value

List of genes within the pathway

`Pathway_similarity` *Pathway_similarity is a function for calculating correlation between expression level of pathways between 2 groups using all the available approaches in SIGN*

Description

Pathway_similarity is a function for calculating correlation between expression level of pathways between 2 groups using all the available approaches in SIGN

Usage

```
Pathway_similarity(ExpMat1, ExpMat2, GeneVec, GeneSets, Name)
```

Arguments

`ExpMat1`	Expression matrix of genes in the 1st group of sampls
`ExpMat2`	Expression matrix of genes in the 2nd group of sampls
`GeneVec`	Name of genes in the same order as considered in ExpMat1 and ExpMat2
`GeneSets`	List of genes within pathways
`Name`	Name used for naming the columns of output matrix of correlation between the 2 groups

Value

Similarity of the pathway between the two expression matrices using pearson correlation, bubble sort, and wilcoxon paaired rank test

---

`SIGN_Aggregate` *SIGN_Aggregate is a function to reshape the list of pathway scoring, time to death, and event and return a summary list*

Description

SIGN_Aggregate is a function to reshape the list of pathway scoring, time to death, and event and return a summary list

Usage

```
SIGN_Aggregate(ScoreList, TimeList, EventList)
```

Arguments

`ScoreList`	List of similarity scores identified using different methodologies
`TimeList`	List of time to event (death) for different groups of patients
`EventList`	List of event vectors (death or censored) for different groups of patients

Value

List of scores identified for each sample as well as time to death and event of that sample

---

`SIGN_Ensemble_SimCal` *SIGN_Ensemble_SimCal is a function for Generating list fo similarities based on different pathway quantification methods and similarity measures*

Description

SIGN_Ensemble_SimCal is a function for Generating list fo similarities based on different pathway quantification methods and similarity measures

Usage

```
SIGN_Ensemble_SimCal(ExpList, RefClassID, TestClassID, GeneID, PathwaySets)
```

Arguments

`ExpList`	List of expression matrices for different groups of samples used in the centroid classification scheme
`RefClassID`	Names of the matrices in the ExpList
`TestClassID`	ID of a matrix in ExpList to be used as test set
`GeneID`	Parameter to determine if gene annotations are provided as Symbols or EntrezIDs
`PathwaySets`	List of pathways containing gene annotations for each pathways

Value

List of similarities identified in both gene and pathway level

---

`Similarities_Wrapper` *Similarities_Wrapper is wrapper to identify similarities between the expression of genes in target sample and the reference expression matrix*

Description

Similarities_Wrapper is wrapper to identify similarities between the expression of genes in target sample and the reference expression matrix

Usage

```
Similarities_Wrapper(ExpMat_Test, ExpMat_Ref, GeneVec, PathwaySet, RefID,
  TestClassIter, SampleIter)
```

Arguments

`ExpMat_Test`	Expression matrix of test samples
`ExpMat_Ref`	Expression matrix of reference samples
`GeneVec`	Vector of gene names
`PathwaySet`	List of pathways containing gene annotations for each pathways
`RefID`	Class of the reference set
`TestClassIter`	Class of the test set (if it is the same as reference set, the target test sample will be removed fro the reference set)
`SampleIter`	Target test sample in ExpMat_Testto be used for comparison with ExpMat_Ref

Value

List of similarities between the target sample and the expression matrix of reference samples

---

`SimSummary_2Class` *SimSummary_2Class is a function to calculating similarity between two set of samples*

Description

SimSummary_2Class is a function to calculating similarity between two set of samples

Usage

```
SimSummary_2Class(SimMat1, SimMat2)
```

Arguments

`SimMat1`	Matrix of similarity of the target samples with the 1st reference matrix
`SimMat2`	Matrix of similarity of the target samples with the 2nd reference matrix

Value

Matrix of similarities of samples

`SurvivalStat_PostProcess` *SurvivalStat_PostProcess is a function to Extract summary statistics of the built cox model*

Description

SurvivalStat_PostProcess is a function to Extract summary statistics of the built cox model

Usage

```
SurvivalStat_PostProcess(StatList)
```

Arguments

`StatList`	Summary lists of the cox models built using all the

Value

A list including Cindex, Cindex_std and LogTest_pval

`Survival_Stats` *Survival_Stats is a function for building cox model using all the features and each feature as a separate model*

Description

Survival_Stats is a function for building cox model using all the features and each feature as a separate model

Usage

```
Survival_Stats(ScoreMat, TimeVec, EventVec)
```

Arguments

`ScoreMat`	Matrix of feature values used for survival predition
`TimeVec`	Vectore of time to death of samples (patients)
`EventVec`	Vector of events for the samples (patients) as being dead or censored

Value

A list containing summary of a cox model using all of the features and separate cox models for each feature

---

`TSC` *TSC is a function to calculate transcriprtional similarity coefficient between two biological pathways*

Description

TSC is a function to calculate transcriprtional similarity coefficient between two biological pathways

Usage

```
TSC(PathwayExp1, PathwayExp2)
```

Arguments

`PathwayExp1`	Expression matrix of genes within the chosen pathway in the 1st set of samples
`PathwayExp2`	Expression matrix of genes within the chosen pathway in the 2nd set of samples

Value

Transcriptional similarity coefficient

Examples

```
Pathway1_ExpMat <- matrix(runif(100,0,10), ncol = 10)
Pathway2_ExpMat <- matrix(runif(100,0,10), ncol = 10)
TSC(Pathway1_ExpMat, Pathway2_ExpMat)
```

Index