Package 'IDMIR'

January 20, 2025

Type Package

Title Identification of Dysregulated MiRNAs Based on MiRNA-MiRNA
Interaction Network

Version 0.1.0

Maintainer Junwei Han <hanjunwei1981@163.com>

Description A systematic biology tool was developed to identify dysregulated miRNAs via a miRNA-
miRNA interaction network. 'IDMIR' first constructed a weighted miRNA interaction net-
work through integrating miRNA-target interaction information, molecular func-
tion data from Gene Ontology (GO) database and gene transcriptomic data in specific-
disease context, and then, it used a network propagation algorithm on the network to identify sig-
nificantly dysregulated miRNAs.

License GPL (>= 2)

Encoding UTF-8

LazyData true

Imports egg, fastmatch, forestplot, ggplot2, grDevices, grid, igraph,
pheatmap, survival, survminer

RoxygenNote 7.2.3

Depends R (>= 2.10)

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Junwei Han [aut, cre, cph],
Xilong Zhao [aut],
Jiashuo Wu [aut]

Repository CRAN

Date/Publication 2023-11-09 15:30:02 UTC

Contents

Index   **10**

GetData_Mirna   *GetData_Mirna*

Description

Get the example data

Usage

```
GetData_Mirna(Data)
```

Arguments

Data   A character should be one of"survival", "GEP", "MF_Target", "MiRNA_Target",
              "matrix_mirna_go_inter", "matrix_mirna_go_jaccard"

Value

data

GetGDEscore   *GetGDEscore*

Description

Function "GetGDEscore" is used to calculate gene differential expression levels.

Usage

```
GetGDEscore(ExpData,Label)
```

Arguments

ExpData	A gene expression profile of interest (rows are genes, columns are samples).
Label	A character vector consists of "0" and "1" which represent sample class in the gene expression profile. "0" means normal sample and "1" means disease sample.

Value

A matrix with one column of GDEscore.

Examples

```
# Obtain the example data
GEP<-GetData_Mirna("GEP")
label<-GetData_Mirna("label")
# Run the function
GDEscore<-GetGDEscore(GEP,label)
```

IdentifyMiRNA	*IdentifyMiRNA*

Description

The function "IdentifyMiRNA" is used to identify significantly dysregulated miRNAs by calculating the eigenvector centrality of miRNAs.

Usage

```
IdentifyMiRNA(GDEscore.table,nperm=1000,damping=0.90)
```

Arguments

GDEscore.table	A matrix with one column of GDEscore.
nperm	The Number of random permutations (default: 100).
damping	Restart the probability of the random-walk algorithm (default: 0.9).

Value

A data frame with seven columns those are "MiRNA", "Target", "Number" (number of targets), "Score" (Centrality score), "P-value", and "FDR".

Examples

```
# Obtain the example data
GEP<-GetData_Mirna("GEP")
label<-GetData_Mirna("label")
# Run the function
GDEscore<-GetGDEscore(GEP,label)
MiRNAScore<-IdentifyMiRNA(GDEscore,nperm=5, damping=0.90)
```

MirnaData   *An environment variable that includes some example data*

Description

An environment variable that includes some example data. matirx_mirna_go_inter:A matrix of Jaccard score between MiRNAs and GOMF. matirx_mirna_go_jaccard:A matrix consisting of genes shared by MiRNAs targets and GOMF. MiRNAScore:a ranked list of strong and weak associations with cancer. MF_Target:GOMF and corresponding targets. MiRNA_Target:MiRNAs and corresponding targets. zscore_BRCA:An example gene expression profile. label:A vector representing the label of the sample of BRCA, where "1" is the disease sample and "0" is the normal sample. survival:A dataframe including three columns which are sample, status, and time.

Usage

    MirnaData

Format

An environment variable

 MutiMiRNA_CRModel   *MutiMiRNA_CRModel*

Description

Function "MutiMiRNA_CRModel" can build a multivariate Cox model through integrating the models constructed separately based on different mirna targets.

Usage

    MutiMiRNA_CRModel(ExpData, MiRNAs,SurvivalData,cutoff.point=NULL)

Arguments

ExpData	A gene expression profile of interest (rows are genes, columns are samples).
MiRNAs	An interest miRNA vector.
SurvivalData	Survival data (the column names are: "sample", "status", and "time") corresponding to the samples in gene expression profile of interest.
cutoff.point	A numeric value used to divide high-risk and low-risk groups.

Value

A list includes a data frame with seven parts those are "sample", "status", "time", "target gene expression", "risk score", "group", and a data frame with five columns those are "Gene", "HR", "HR.95L", "HR.95H", "beta", and "P-value".

Examples

```
# Obtain the example data
GEP<-GetData_Mirna("GEP")
survival<-GetData_Mirna("survival")
MiRNAs<-c("hsa-miR-21-5p","hsa-miR-26a-5p","hsa-miR-369-5p","hsa-miR-1238-3p","hsa-miR-10b-5p")
# Run the function
MutiMiRNA_CRData<-MutiMiRNA_CRModel(GEP,
MiRNAs,survival,cutoff.point=NULL)
```

PlotForest   *PlotForest*

Description

Function "PlotForest" can visualize the result of Cox regression analysis through forest plot.

Usage

```
PlotForest(MiRNA_CRData,g.pos = 2,b.size = 3,col = c("#FE0101", "#1C61B6", "#A4A4A4"),
lwd.zero = 2,lwd.ci = 3,x.lab = "Hazard Ratio Plot")
```

Arguments

MiRNA_CRData	A list includes a data frame with seven parts those are "sample", "status", "time", "target genes expression", "risk score", "group", and a data frame with five columns those are "Gene", "HR", "HR.95L", "HR.95H", "beta", and "P-value".
g.pos	The position of the graph element within the table of text. The position can be 1-(ncol(labeltext) + 1). You can also choose set the position to "left" or "right".
b.size	Override the default box size based on precision.
col	Set the colors for all the elements in the plot.
lwd.zero	lwd for the vertical line that gives the no-effect line, see gpar.
lwd.ci	lwd for the confidence bands, see gpar.
x.lab	x-axis label.

Value

Forest maps associated with the Cox risk model.

Examples

```
# Obtain the example data
GEP<-GetData_Mirna("GEP")
survival<-GetData_Mirna("survival")
MiRNAs<-c("hsa-miR-21-5p","hsa-miR-26a-5p","hsa-miR-369-5p","hsa-miR-1238-3p","hsa-miR-10b-5p")
# Run the function
SingleMiRNA_CRData<-SingleMiRNA_CRModel(GEP,
```

```
"hsa-miR-21-5p",survival,cutoff.point=NULL)
PlotForest(SingleMiRNA_CRData)
MutiMiRNA_CRData<-MutiMiRNA_CRModel(GEP,
MiRNAs,survival,cutoff.point=NULL)
PlotForest(MutiMiRNA_CRData)
```

PlotHeatmap   *PlotHeatmap*

Description

The function "PlotHeatmap" is used to plot a heat map of miRNA targets expression.

Usage

```
PlotHeatmap(MiRNA_CRData,yaxis=c(-2,2),scale="row",
cluster.rows=FALSE,cluster.cols=FALSE,show.colnames=FALSE,
ann_colors=c("#ef6d6d","#5470c6"),col=c("#ef6d6d","#5470c6"))
```

Arguments

MiRNA_CRData	A list includes a data frame with seven parts those are "sample", "status", "time", "target genes expression", "risk score", "group", and a data frame with five columns those are "Gene", "HR", "HR.95L", "HR.95H", "beta", and "P-value".
yaxis	The upper and lower limits of this heat map.
scale	character indicating if the values should be centered and scaled in either the row direction or the column direction, or none. Corresponding values are "row", "column" and "none".
cluster.rows	A logical value that represents whether row clustering is used.
cluster.cols	A logical value that represents whether col clustering is used.
show.colnames	This parameter controls whether column names are displayed.
ann_colors	Vector of colors used to define groups.
col	Vector of colors used in the heatmap.

Value

A heat map of miRNA targets expression.

Examples

```
# Obtain the example data
GEP<-GetData_Mirna("GEP")
survival<-GetData_Mirna("survival")
MiRNAs<-c("hsa-miR-21-5p","hsa-miR-26a-5p","hsa-miR-369-5p","hsa-miR-1238-3p","hsa-miR-10b-5p")
# Run the function
SingleMiRNA_CRData<-SingleMiRNA_CRModel(GEP,
```

```
"hsa-miR-21-5p",survival,cutoff.point=NULL)
PlotHeatmap(SingleMiRNA_CRData)
MutiMiRNA_CRData<-MutiMiRNA_CRModel(GEP,
MiRNAs,survival,cutoff.point=NULL)
PlotHeatmap(MutiMiRNA_CRData)
```

PlotScatter   *PlotScatter*

Description

Function "PlotScatter" is used to plot a scatter diagram.

Usage

```
PlotScatter(MiRNA_CRData,status.0='Alive',status.1='Dead',
TitleYlab_A='Risk Score',TitleYlab_B='Survival Time',TitleXlab='Rank',
TitleLegend_A='Risk Group',TitleLegend_B='Status',
color.A=c(low='blue',high='red'),color.B=c(status.0='blue',status.1='red'))
```

Arguments

MiRNA_CRData	A list includes a data frame with seven parts those are "sample", "status", "time", "target genes expression", "risk score", "group", and a data frame with five columns those are "Gene", "HR", "HR.95L", "HR.95H", "beta", and "P-value".
status.0	string. Code for event 0. Default is 'Alive'
status.1	string. Code for event 1. Default is 'Dead'
TitleYlab_A	string, y-lab title for figure A. Default is 'Riskscore'
TitleYlab_B	string, y-lab title for figure B. Default is 'Survival Time'
TitleXlab	string, x-lab title for figure B. Default is 'Rank'
TitleLegend_A	string, legend title for figure A. Default is 'Risk Group'
TitleLegend_B	string, legend title for figure B. Default is 'Status'
color.A	color for figure A. Default is low = 'blue', high = 'red'
color.B	color for figure B. Default is status.0 = 'blue', status.1 = 'red'

Value

A riskscore picture

Examples

```
# Obtain the example data
GEP<-GetData_Mirna("GEP")
survival<-GetData_Mirna("survival")
MiRNAs<-c("hsa-miR-21-5p","hsa-miR-26a-5p","hsa-miR-369-5p","hsa-miR-1238-3p","hsa-miR-10b-5p")
# Run the function
SingleMiRNA_CRData<-SingleMiRNA_CRModel(GEP,
"hsa-miR-21-5p",survival,cutoff.point=NULL)
PlotScatter(SingleMiRNA_CRData)
MutiMiRNA_CRData<-MutiMiRNA_CRModel(GEP,
MiRNAs,survival,cutoff.point=NULL)
PlotScatter(MutiMiRNA_CRData)
```

PlotSurvival *PlotSurvival*

Description

Function "PlotSurvival" is used to draw a Kaplan-Meier curve.

Usage

```
PlotSurvival(MiRNA_CRData,colors=c("#ef6d6d","#5470c6"))
```

Arguments

MiRNA_CRData	A list includes a data frame with seven parts those are "sample", "status", "time", "target genes expression", "risk score", "group", and a data frame with five columns those are "Gene", "HR", "HR.95L", "HR.95H", "beta", and "P-value".
colors	Vector of colors used to define groups.

Value

A survival curve of a data set.

Examples

```
# Obtain the example data
GEP<-GetData_Mirna("GEP")
survival<-GetData_Mirna("survival")
MiRNAs<-c("hsa-miR-21-5p","hsa-miR-26a-5p","hsa-miR-369-5p","hsa-miR-1238-3p","hsa-miR-10b-5p")
# Run the function
SingleMiRNA_CRData<-SingleMiRNA_CRModel(GEP,
"hsa-miR-21-5p",survival,cutoff.point=NULL)
PlotSurvival(SingleMiRNA_CRData)
MutiMiRNA_CRData<-MutiMiRNA_CRModel(GEP,
MiRNAs,survival,cutoff.point=NULL)
PlotSurvival(MutiMiRNA_CRData)
```

SingleMiRNA_CRModel   *SingleMiRNA_CRModel*

Description

Function "SingleMiRNA_CRModel" uses survival data to build a multivariate Cox model using the targets of a single miRNA.

Usage

```
SingleMiRNA_CRModel(ExpData,MiRNA,cutoff.point=NULL,SurvivalData)
```

Arguments

ExpData   A gene expression profile of interest (rows are genes, columns are samples).

MiRNA   A miRNA ID.

cutoff.point   A numeric value is used to divide high-risk and low-risk groups.

SurvivalData   Survival data (the column names are: "sample", "status", "time") corresponding to samples in the gene expression profile of interest.

Value

A list includes a data frame with seven parts those are "sample", "status", "time", "target genes expression", "risk score", "group", and a dataframe with five columns those are "Gene", "HR", "HR.95L", "HR.95H", "beta", and "P-value".

Examples

```
# Obtain the example data
GEP<-GetData_Mirna("GEP")
survival<-GetData_Mirna("survival")
# Run the function
SingleMiRNA_CRData<-SingleMiRNA_CRModel(GEP,
"hsa-miR-21-5p",cutoff.point=NULL,survival)
```

Index