Package 'SMDIC'

June 5, 2020

Type Package

Title Identification of Somatic Mutation-Driven Immune Cells

Version 0.1.0

Author Baotong Zheng, Junwei Han

Maintainer Junwei Han <hanjunwei1981@163.com>

Description A computing tool is developed to automated identify somatic mutation-driven immune cells. The operation modes including: i) inferring the relative abundance matrix of tumor-infiltrating immune cells and integrating it with a particular gene mutation status, ii) detecting differential immune cells with respect to the gene mutation status and converting the abundance matrix of significant differential immune cell into two binary matrices (one for up-regulated and one for down-regulated), iii) identifying somatic mutation-driven immune cells by comparing the gene mutation status with each immune cell in the binary matrices across all samples, and iv) visualization of immune cell abundance of samples in different mutation status.

License GPL (>= 2)

Encoding UTF-8

LazyData true

Imports GSVA,
samr,
e1071,
parallel,
preprocessCore,
pheatmap,
maftools,
grDevices,
survival,
survminer,
MASS,
pracma

Suggests knitr,
rmarkdown

Depends R (>= 3.5.0)

RoxygenNote 7.1.0

VignetteBuilder knitr

R topics documented:

`SMDIC-package` *Identification of somatic mutation-driven immune cells*

Description

With the use of functions in this packages, users could identify the immune cells driven by somatic mutations in tumor microenvironment.

`cell24` *A data.frame of 24 immune cells name from Bindea et al*

Description

It's a built-in data. The first column represents the abbreviation of 24 immune cells, the second column represents the full name of 24 immune cells

Usage

```
cell24
```

Format

A data.frame with 24 rows and 2 column

References

Bindea G, Mlecnik B, Tosolini M, Kirilovsky A, Waldner M, Obenauf AC, et al. Spatiotemporal dynamics of intratumoral immune cells reveal the immune landscape in human cancer. Immunity. 2013;39:782–95.

cell64 *A data.frame of 64 immune cells name from xCell method*

Description

It's a built-in data. The first column represents the abbreviation of 64 immune cells, the second column represents the full name of 64 immune cells

Usage

```
cell64
```

Format

A data.frame with 64 rows and 2 column

References

Aran D , Hu Z , Butte A J . xCell: digitally portraying the tissue cellular heterogeneity landscape[J]. Genome Biology, 2017, 18(1):220.

envData *envData*

Description

The variables in the environment include an example expression profiles, a cell abundance matrix, a binary numerical matrix which shows the immune cells driven by somatic mutation, a binary mutations matrix.

Format

An environment variable

Details

The environment variable includes the variable `exp.example`, `cellmatrix`,`mutcell`,`mutmatrix`

Author(s)

Junwei Han<hanjunwei1981@163.com>,Baotong Zheng<btzheng1116@163.com>

exp2cell *exp2cell*

Description

Function ‘exp2cell‘ use gene expression profiles to quantify cell abundance matrix. ‘exp2cell‘ provides three methods for estimating the relative infiltration abundance of different cell types in the tumor microenvironment (TME), which including xCell, ssGSEA estimated method proposed by Şenbabaoğlu et al. and CIBERSORT.

Usage

```
exp2cell(exp, method = "xCell")
```

Arguments

exp	The gene expression data set. A matrix with row names as symbols and columns as samples. Gene expression profiles were used to quantify cell abundance matrix.
method	Method must be one of "xCell", "ssGSEA" and "CIBERSORT".

Value

Cell abundance matrix.

References

1. Aaron, M, Newman, et al. Robust enumeration of cell subsets from tissue expression profiles.[J]. Nature Methods, 2015. 2. Aran D , Hu Z , Butte A J . xCell: digitally portraying the tissue cellular heterogeneity landscape[J]. Genome Biology, 2017, 18(1):220. 3. Şenbabaoğlu, Yasin, Gejman R S , Winer A G , et al. Tumor immune microenvironment characterization in clear cell renal cell carcinoma identifies prognostic and immunotherapeutically relevant messenger RNA signatures[J]. Genome biology, 2016, 17(1).

Examples

```
exp.example<-GetExampleData("exp.example") # gene expression profiles
cellmatrix<-exp2cell(exp=exp.example,method="ssGSEA") #cell abundance matrix
```

gene2cellsummary *gene2cellsummary*

Description

Function ‘gene2cellsummary‘ is a generic function used to produce result summaries of the immune cells driven by a somatic mutation.

Usage

```
gene2cellsummary(gene, method = "xCell", mutcell)
```

Arguments

gene	Somatic mutant gene name
method	Method must be one of "xCell","ssGSEA" and "CIBERSORT".
mutcell	The result of ‘mutcorcell‘ funtion.

Value

A matrix shows the short name, full name, pvalue, fdr of the cells driven by a somatic mutation

Examples

```
mutcell<-GetExampleData("mutcell") # The result of `mutcorcell` funtion.
genecellsummary<-gene2cellsummary(gene="TP53",mutcell=mutcell)
```

GetExampleData *Get the example data*

Description

Get the example data from SMDIC package.

Usage

```
GetExampleData(exampleData)
```

Arguments

exampleData	A character, should be one of "exp.example", "cellmatrix", "mutcell", "mutmatrix", "surv".

Details

The function ‘GetExampleData(ExampleData = "mutmatrix)")‘ obtains the mutations matrix

References

Subramanian, A., Tamayo, P., Mootha, V.K., Mukherjee, S., Ebert, B.L., Gillette, M.A., Paulovich, A., Pomeroy, S.L., Golub, T.R., Lander, E.S. et al. (2005) Gene set enrichment analysis: a knowledgebased approach for interpreting genome-wide expression profiles. Proc Natl Acad Sci U S A, 102, 15545-15550.

`heatmapcell` *heatmapcell*

Description

A function to draw clustered heatmaps for the cells driven by a somatic mutation.

Usage

```
heatmapcell(gene, mutcell, cellmatrix, mutmatrix)
```

Arguments

`gene`	Somatic mutant gene name
`mutcell`	A list, mutcell is the result of function 'mutcorcell'.
`cellmatrix`	Cell abundance matrix, cellmatrix is the result of function 'exp2cell'.
`mutmatrix`	A binary mutations matrix, in which 1 represents any mutation occurs in a particular gene in a particular sample, otherwise the element is 0.

Examples

```
mutcell<-GetExampleData("mutcell") # The result of `mutcorcell` function.
cellmatrix<-GetExampleData("cellmatrix") # Cell abundance matrix
mutmatrix<-GetExampleData("mutmatrix") # A binary mutations matrix
heatmapcell(gene = "TP53",mutcell = mutcell,cellmatrix = cellmatrix,mutmatrix = mutmatrix)
```

`immunelist` *A large list of 24 immune cells type-specific gene signatures from Bindea et al*

Description

It's a built-in data. The name of the list represent 24 immune cells, the value of the list are 24 immune cells type-specific gene signatures from Bindea et al

Usage

```
immunelist
```

Format

A list

References

Bindea G, Mlecnik B, Tosolini M, Kirilovsky A, Waldner M, Obenauf AC, et al. Spatiotemporal dynamics of intratumoral immune cells reveal the immune landscape in human cancer. Immunity. 2013;39:782–95.

maf2matrix *maf2matrix*

Description

Function ‘maf2matrix‘ use mutation annotation file (MAF) format data to build a binary mutations matrix.

Usage

```
maf2matrix(maffile, percent = 0.01, nonsynonymous = TRUE)
```

Arguments

maffile	The name of mutation annotation file (MAF) format data. It must be an absolute path or the name relatived to the current working directory.
percent	A threshold value(one percent as the default value). The genes with a given mutation frequency equal or greater than the threshold value are retained for the following analysis.
nonsynonymous	Logical, tell if extract the non-silent somatic mutations (nonsense mutation, missense mutation, frame-shif indels, splice site, nonstop mutation, translation start site, inframe indels).

Value

A binary mutations matrix, in which 1 represents any mutation occurs in a particular gene in a particular sample, otherwise the element is 0.

Examples

```
#get path of the mutation annotation file.
maf = system.file('extdata', 'example.maf', package = 'SMDIC')
# perform function `maf2matrix`.
mutmatrix.example<-maf2matrix(maf)
```

mutcellsummary *mutcellsummary*

Description

Function ‘mutcellsummary‘ is a generic function used to produce summaries of the results of ‘mutcorcell‘ function.

Usage

```
mutcellsummary(mutcell, mutmatrix, cellmatrix)
```

Arguments

mutcell	The result of ‘mutcorcell‘ funtion.
mutmatrix	A binary mutations,in which 1 represents any mutation occurs in a particular gene in a particular sample, otherwise the element is 0. matrix
cellmatrix	Cell abundance matrix

Value

The result summaries have four columns. The first column is somatic mutant gene names, the second column is the immune cell names driven by the somatic mutation, the third column is the number of the immune cell, the fourth column is the mutation rate.

Examples

```
mutcell<-GetExampleData("mutcell") # The result of `mutcorcell` funtion
cellmatrix<-GetExampleData("cellmatrix") # Cell abundance matrix
mutmatrix<-GetExampleData("mutmatrix") # A binary mutations matrix
summary<-mutcellsummary(mutcell = mutcell,mutmatrix = mutmatrix,cellmatrix=cellmatrix)
```

---

mutcorcell *mutcorcell*

Description

Function ‘mutcorcell‘ identifies somatic mutation-driven immune cells by comparing the cell abundance matrix and binary mutations matrix.

Usage

```
mutcorcell(
  cellmatrix = cellmatrix,
  mutmatrix = mutmatrix,
  samfdr.cutoff = 0.05,
  nperms = 100,
  fisher.cutoff = 0.05,
  fisher.adjust = FALSE
)
```

Arguments

cellmatrix	Cell abundance matrix.
mutmatrix	A binary mutations matrix, in which 1 represents any mutation occurs in a particular gene in a particular sample, otherwise the element is 0.
samfdr.cutoff	False Discovery Rate cutoff for output in significant immune cells
nperms	Number of permutations used by SAM to estimate False Discovery Rates
fisher.cutoff	False Discovery Rate(fisher.adjust=TRUR) or P-Value(fisher.adjust=FALSE) cutoff for Fisher’s exact test
fisher.adjust	Logical,tell if corrects p-values

Value

A list of three matrices: a binary numerical matrix which shows the immune cells driven by somatic mutant gene; two numerical matrix which show the pvalue and fdr of the immune cells driven by somatic mutant gene.

Examples

```
cellmatrix<-GetExampleData("cellmatrix") # Cell abundance matrix
mutmatrix<-GetExampleData("mutmatrix") # A binary mutations matrix
mutcell<-mutcorcell(cellmatrix = cellmatrix,mutmatrix = mutmatrix)
# The summary for somatic mutations are produced by function `mutcellsummary`.
#summary<-mutcellsummary(mutcell = mutcell,mutmatrix = mutmatrix,cellmatrix=cellmatrix)
# The summary of the immune cells driven by a mutation are produced by function `gene2cellsummary`.
#genecellsummary<-gene2cellsummary(gene="TP53",mutcell=mutcell)
```

`plotCoocMutex` *plotCoocMutex*

Description

Function ‘plotCoocMutex‘ plots the co-occurrence and mutual exclusivity plots for mutation genes which drive immune cells.

Usage

```
plotCoocMutex(maffile, mutcell.summary, cellnumcuoff = 3)
```

Arguments

`maffile`	The name of mutation annotation file (MAF) format data. It must be an absolute path or the name relatived to the current working directory.
`mutcell.summary`	The result of ‘mutcellsummary‘ function
`cellnumcuoff`	A threshold value (4 as the default value). The mutation genes which drive at least "cellnumcuoff" cells are retained for drawing a co-occurrence and mutual exclusivity plots.

References

Gerstung M, Pellagatti A, Malcovati L, et al. Combining gene mutation with gene expression data improves outcome prediction in myelodysplastic syndromes. Nature Communications. 2015;6:5901. doi:10.1038/ncomms6901.

Examples

```
cellmatrix<-GetExampleData("cellmatrix") # Cell abundance matrix
mutmatrix<-GetExampleData("mutmatrix") # A binary mutations matrix
mutcell<-GetExampleData("mutcell") # The result of `mutcorcell` funtion
summary<-summary<-mutcellsummary(mutcell = mutcell,mutmatrix = mutmatrix,cellmatrix=cellmatrix)
file<-"dir" #dir must be an absolute path or the name  relatived to the current working directory.
## Not run: plotCoocMutex(maffile = dir,mutcell.summary = summary,cellnumcuoff =0)
```

plotwaterfall *plotwaterfall*

Description

Function‘plotwaterfall‘ plots the waterfall for mutation genes which drive immune cells.

Usage

```
plotwaterfall(maffile, mutcell.summary, cellnumcuoff = 3)
```

Arguments

maffile	The name of mutation annotation file (MAF) format data. It must be an absolute path or the name relatived to the current working directory.
mutcell.summary	The result of ‘mutcellsummary‘ function
cellnumcuoff	a threshold value (4 as the default value). The mutation genes which drive at least "cellnumcuoff" cells are retained for drawing an waterfall.

Examples

```
file<-"dir" #dir must be an absolute path or the name  relatived to the current working directory.
## Not run: plotwaterfall(maffile = dir,mutcell.summary = summary,cellnumcuoff =0)
```

survcell *survcell*

Description

Function ‘survcell‘ draws Kaplan–Meier curves for survival in the above-median and below-median groups for cell risk score. The cell risk score is calaulated by the weighted mean of cells driven by a gene mutation, where the weight of cells is estimated by the "Univariate" or "Multivariate" cox.

Usage

```
survcell(gene, mutcell, cellmatrix, surv, method = "Multivariate")
```

Arguments

gene	Somatic mutant gene name
mutcell	The result of ‘mutcorcell‘ function
cellmatrix	Cell abundance matrix
surv	Surv is the survival data, the first column is the sample name, the second column is the survival time, and the third is the survival event.
method	Method must be one of "Univariate" and "Multivariate". The coefficient of cells for risk score are estimated by "Univariate" or "Multivariate" cox proportional risk regression model on cell abundance matrix and overall survival data..

Value

Kaplan–Meier curves

Examples

```
mutcell<-GetExampleData("mutcell") # The result of `mutcorcell` function.
cellmatrix<-GetExampleData("cellmatrix") # Cell abundance matrix
surv<-GetExampleData("surv") # The survival data
survcell(gene ="TP53",mutcell=mutcell,cellmatrix=cellmatrix,surv=surv)
```

`xCell.data` *xCell datasets*

Description

xCell datasets. It's a built-in data.

Usage

```
xCell.data
```

Format

list:

spill spillover matrix and calibration parameters

signatures the signatures for calculating scores

genes genes to use to calculate xCell

Index