

# Package ‘pathwayTMB’

October 10, 2021

**Type** Package

**Title** Pathway Based Tumor Mutational Burden

**Version** 0.1.1

**Maintainer** Junwei Han <hanjunwei1981@163.com>

**Description** A systematic bioinformatics tool to develop a new pathway-based gene panel for tumor mutational burden (TMB) assessment (pathway-based tumor mutational burden, PTMB), using somatic mutations files in an efficient manner from either The Cancer Genome Atlas sources or any in-house studies as long as the data is in mutation annotation file (MAF) format. Besides, we develop a multiple machine learning method using the sample's PTMB profiles to identify cancer-specific dysfunction pathways, which can be a biomarker of prognostic and predictive for cancer immunotherapy.

**License** GPL (>= 2)

**Depends** R (>= 4.1.0)

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.2

**biocViews**

**Imports** BiocGenerics, GenomicFeatures, purrr, utils, glmnet, randomForest, stats, survival, survminer, caret, data.table, RColorBrewer, grDevices, pROC, graphics, maftools, clusterProfiler

**VignetteBuilder** knitr,rmarkdown,qpdf

**Suggests** stringi, knitr, rmarkdown, testthat, BiocManager, xfun, e1071, qpdf, tinytex, spelling

**Language** en-US

**NeedsCompilation** no

**Author** Junwei Han [aut, cre, cph],  
Xiangmei Li [aut]

**Repository** CRAN

**Date/Publication** 2021-10-10 11:10:02 UTC

**R topics documented:**

final_character . . . . .	2
GenePathwayOncoplots . . . . .	2
genesmbol . . . . .	5
gene_path . . . . .	5
get_final_signature . . . . .	6
get_gene_length . . . . .	6
get_mut_matrix . . . . .	7
get_PTMB . . . . .	8
mut_matrix . . . . .	9
plotKMcurves . . . . .	9
plotMutInteract . . . . .	11
plotROC . . . . .	13
PTMB_matrix . . . . .	15
sur . . . . .	15
<b>Index</b>	<b>16</b>

---

final_character	<i>final_character, the example's final signature</i>
-----------------	---

---

**Description**

final\_character, a potential marker for cancer prognostic and immunotherapy, generated by 'get\_final\_signature'

**Usage**

```
final_character
```

**Format**

An object of class character of length 2.

---

GenePathwayOncoplots	<i>draw an GenePathwayOncoplots</i>
----------------------	-------------------------------------

---

**Description**

takes output generated by read.maf and draws an GenePathwayOncoplots.

**Usage**

```
GenePathwayOncoplots(
  maffile,
  gene_path,
  freq_matrix,
  risk_score,
  cut_off,
  final_character,
  isTCGA = FALSE,
  top = 20,
  clinicalFeatures = "sample_group",
  annotationColor = c("red", "green"),
  sortByAnnotation = TRUE,
  removeNonMutated = FALSE,
  drawRowBar = TRUE,
  drawColBar = TRUE,
  leftBarData = NULL,
  leftBarLims = NULL,
  rightBarData = NULL,
  rightBarLims = NULL,
  topBarData = NULL,
  logColBar = FALSE,
  draw_titv = FALSE,
  showTumorSampleBarcodes = FALSE,
  fill = TRUE,
  showTitle = TRUE,
  titleText = NULL
)
```

**Arguments**

<code>maffile</code>	an MAF object generated by <code>read.maf</code> .
<code>gene_path</code>	User input pathways geneset list.
<code>freq_matrix</code>	The mutations matrix, generated by <code>'get_mut_matrix'</code> .
<code>risk_score</code>	Samples' PTMB-related risk score, which could be a biomarker for survival analysis and immunotherapy prediction.
<code>cut_off</code>	A threshold value (the median risk score as the default value). Using this value to divide the sample into high and low risk groups with different overall survival.
<code>final_character</code>	The pathway signature, use to map gene in the GenePathwayOncoplots.
<code>isTCGA</code>	Is input MAF file from TCGA source. If TRUE uses only first 12 characters from <code>Tumor_Sample_Barcode</code> .
<code>top</code>	how many top genes to be drawn, genes are arranged from high to low depending on the frequency of mutations. defaults to 20.
<code>clinicalFeatures</code>	columns names from <code>'clinical.data'</code> slot of MAF to be drawn in the plot. Default <code>"sample_group"</code> .

annotationColor	Custom colors to use for sample annotation-"sample_group". Must be a named list containing a named vector of colors. Default "red" and "green".
sortByAnnotation	logical sort oncomatrix (samples) by provided 'clinicalFeatures'. Sorts based on first 'clinicalFeatures'. Defaults to TRUE. column-sort.
removeNonMutated	Logical. If TRUE removes samples with no mutations in the GenePathwayOncoplots for better visualization. Default FALSE.
drawRowBar	logical. Plots right barplot for each gene. Default TRUE.
drawColBar	logical plots top barplot for each sample. Default TRUE.
leftBarData	Data for leftside barplot. Must be a data.frame with two columns containing gene names and values. Default 'NULL'.
leftBarLims	limits for 'leftBarData'. Default 'NULL'.
rightBarData	Data for rightside barplot. Must be a data.frame with two columns containing to gene names and values. Default 'NULL' which draws distribution by variant classification. This option is applicable when only 'drawRowBar' is TRUE.
rightBarLims	limits for 'rightBarData'. Default 'NULL'.
topBarData	Default 'NULL' which draws absolute number of mutation load for each sample. Can be overridden by choosing one clinical indicator(Numeric) or by providing a two column data.frame containing sample names and values for each sample. This option is applicable when only 'drawColBar' is TRUE.
logColBar	Plot top bar plot on log10 scale. Default FALSE.
draw_titv	logical Includes TiTv plot. Default FALSE
showTumorSampleBarcodes	logical to include sample names.
fill	Logical. If TRUE draws genes and samples as blank grids even when they are not altered.
showTitle	Default TRUE.
titleText	Custom title. Default 'NULL'.

### Value

No return value

### Examples

```
#get the path of the mutation annotation file and samples' survival data
maf<-system.file("extdata","data_mutations_extended.txt",package = "pathwayTMB")
sur_path<-system.file("extdata","sur.csv",package = "pathwayTMB")
sur<-read.csv(sur_path,header=TRUE,row.names = 1)
#perform the function 'get_mut_matrix'
mut_matrix<-get_mut_matrix(maffile=maf,mu_fre = 0.01,is.TCGA=FALSE,sur=sur)
#perform the function `get_PTMB`
PTMB_matrix<-get_PTMB(freq_matrix=mut_matrix,genesmbol=genesmbol,gene_path=gene_path)
set.seed(1)
```

```
final_character<-get_final_signature(PTMB=PTMB_matrix,sur=sur)
#calculate the riskscore
riskscore<-plotKMcurves(t(PTMB_matrix[final_character,]),sur=sur,plots=FALSE)$risk_score
cut<-median(riskscore)
GenePathwayOncoplots(maf, gene_path, mut_matrix, riskscore, cut, final_character)
```

---

genesmbol	<i>genesmbol, the coding genes' length</i>
-----------	--

---

### Description

genesmbol, a list of coding genes' length, generated by 'get\_gene\_length'.

### Usage

```
genesmbol
```

### Format

An object of class list of length 34931.

---

gene_path	<i>gene_path, the pathways geneset</i>
-----------	--

---

### Description

gene\_path, a list of KEGG Non-metabolic pathways geneset.

### Usage

```
gene_path
```

### Format

An object of class list of length 27.

---

get\_final\_signature      *Filter cancer-specific dysfunction pathways.*

---

### Description

The function 'get\_final\_signature', using to filter cancer-specific dysfunction pathways (a potential marker for cancer prognostic and immunotherapy), is the main function of our analysis.

### Usage

```
get_final_signature(PTMB, sur)
```

### Arguments

PTMB	The pathway tumor mutation burden matrix, generated by 'get_PTMB'.
sur	A nx2 data frame of samples' survival data, the first line is samples' survival event and the second line is samples' overall survival.

### Value

Return the final PTMB signature, could be a potential marker for prognostic and immunotherapy prediction.

### Examples

```
#get the path of the mutation annotation file and samples' survival data
maf<-system.file("extdata","data_mutations_extended.txt",package = "pathwayTMB")
sur_path<-system.file("extdata","sur.csv",package = "pathwayTMB")
sur<-read.csv(sur_path,header=TRUE,row.names = 1)
#perform the function 'get_mut_matrix'
mut_matrix<-get_mut_matrix(maffile=maf,mu_fre = 0.01,is.TCGA=FALSE,sur=sur)
#perform the function 'get_PTMB'
PTMB_matrix<-get_PTMB(freq_matrix=mut_matrix,genesbol=genesbol,gene_path=gene_path)
set.seed(1)
final_character<-get_final_signature(PTMB=PTMB_matrix,sur=sur)
```

---

get\_gene\_length      *Extract coding genes' length from gene transfer format(GTF) file.*

---

### Description

Extract coding genes' length from gene transfer format(GTF) file.

### Usage

```
get_gene_length(filepath)
```

**Arguments**

filepath            Input gene transfer format file. It must be an absolute path or the name related to the current working directory.

**Value**

Return a list of genes' length.

---

get\_mut\_matrix            *Converts MAF into mutation matrix.*

---

**Description**

The function 'get\_mut\_matrix' converts mutation annotation file (MAF) format data into a mutations matrix. Then use the fisher exact test to select the geneset with higher mutation frequency in alive sample group. Finally return the higher mutation frequency matrix.

**Usage**

```
get_mut_matrix(
  maffile,
  is.TCGA = TRUE,
  mut_fre = 0,
  nonsynonymous = TRUE,
  cut_fisher.pval = 1,
  cut_oddsRatio = 1,
  sur
)
```

**Arguments**

maffile            Input mutation annotation file (MAF) format data. It must be an absolute path or the name related to the current working directory.

is.TCGA            Is input MAF file from TCGA source. If TRUE uses only first 15 characters from Tumor\_Sample\_Barcode.

mut\_fre            A threshold value(zero 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-synonymous somatic mutations (nonsense mutation, missense mutation, frame-shif indels, splice site, nonstop mutation, translation start site, inframe indels).

cut\_fisher.pval    The significant cut\_off pvalue for the fisher test.

cut\_oddsRatio     The cut\_off oddsRatio for the fisher test,uses to select the geneset with higher mutation frequency in alive sample group.

sur                A nx2 data frame of samples' survival data,the first line is samples' survival event and the second line is samples' overall survival.

**Value**

The survival-related mutations matrix.

**Examples**

```
#get the path of the mutation annotation file and samples' survival data
maf<-system.file("extdata","data_mutations_extended.txt",package = "pathwayTMB")
sur_path<-system.file("extdata","sur.csv",package = "pathwayTMB")
sur<-read.csv(sur_path,header=TRUE,row.names = 1)
#perform the function 'get_mut_matrix'
mut_matrix<-get_mut_matrix(maffile=maf,mut_fre = 0.01,is.TCGA=FALSE,sur=sur)
```

---

get\_PTMB

*Calculate the Pathway-based Tumor Mutational Burden.*

---

**Description**

The function 'get\_PTMB' uses to calculate the Pathway-based Tumor Mutational Burden (PTMB). PTMB is defined as pathway-based tumor mutational burden corrected by genes' length and number.

**Usage**

```
get_PTMB(freq_matrix, genesmbol, path_mut_cutoff = 0, gene_path)
```

**Arguments**

freq_matrix	The mutations matrix,generated by 'get_mut_matrix'.
genesmbol	The genes' length matrix,generated by 'get_gene_length'.
path_mut_cutoff	A threshold value(zero percent as the default value).Pathways with a given mutation frequency equal or greater than the threshold value are retained for the following analysis.
gene_path	User input pathways geneset list.

**Value**

Return the Pathway-based Tumor Mutational Burden matrix.

**Examples**

```
#get the path of the mutation annotation file and samples' survival data
maf<-system.file("extdata","data_mutations_extended.txt",package = "pathwayTMB")
sur_path<-system.file("extdata","sur.csv",package = "pathwayTMB")
sur<-read.csv(sur_path,header=TRUE,row.names = 1)
#perform the function 'get_mut_matrix'
mut_matrix<-get_mut_matrix(maffile=maf,mut_fre = 0.01,is.TCGA=FALSE,sur=sur)
#perform the function 'get_PTMB'
PTMB_matrix<-get_PTMB(freq_matrix=mut_matrix,genesmbol=genesmbol,gene_path=gene_path)
```



---

mut_matrix	<i>mut_matrix, mutations matrix</i>
------------	-------------------------------------

---

**Description**

mut\_matrix, the mutations matrix, generated by 'get\_mut\_matrix'

**Usage**

```
mut_matrix
```

**Format**

An object of class matrix (inherits from array) with 1214 rows and 35 columns.

---

plotKMcurves	<i>Drawing Kaplan Meier Survival Curves Using the final survival-related PTMB.</i>
--------------	--

---

**Description**

The function 'plotKMcurves' uses to draw Kaplan-Meier Survival Curves based on PTMB-related riskscore. The riskscore is generated by the signature's PTMB and the coefficient of "Univariate" or "Multivariate" cox regression.

**Usage**

```
plotKMcurves(
  sig_PTMB,
  sur,
  method = "Multivariate",
  returnAll = TRUE,
  pval = TRUE,
  color = NULL,
  plots = TRUE,
  palette = NULL,
  linetype = 1,
  conf.int = FALSE,
  pval.method = FALSE,
  test.for.trend = FALSE,
  surv.median.line = "none",
  risk.table = FALSE,
  cumevents = FALSE,
  cumcensor = FALSE,
  tables.height = 0.25,
  add.all = FALSE,
  ggtheme = theme_survminer()
)
```

**Arguments**

sig_PTMB	The signature's PTMB matrix, which rows are samples and columns are pathways.
sur	A nx2 data frame of samples' survival data, the first line is samples' survival event and the second line is samples' overall survival.
method	Method must be one of "Univariate" and "Multivariate".
returnAll	Logical value. Default is TRUE. If TRUE, return the riskscore and the coefficient of cox regression.
pval	Logical value, a numeric or a string. If logical and TRUE, the p-value is added on the plot. If numeric, then the computed p-value is substituted with the one passed with this parameter. If character, then the customized string appears on the plot.
color	Color to be used for the survival curves. If the number of strata/group (n.strata) = 1, the expected value is the color name. For example color = "blue". If n.strata > 1, the expected value is the grouping variable name. By default, survival curves are colored by strata using the argument color = "strata", but you can also color survival curves by any other grouping variables used to fit the survival curves. In this case, it's possible to specify a custom color palette by using the argument palette.
plots	logical value. Default is TRUE. If TRUE, plot the Kaplan Meier Survival Curves.
palette	the color palette to be used. Allowed values include "hue" for the default hue color scale; "grey" for grey color palettes; brewer palettes e.g. "RdBu", "Blues", ...; or custom color palette e.g. c("blue", "red"); and scientific journal palettes from ggsci R package, e.g.: "npg", "aaas", "lancet", "jco", "ucscgb", "uchicago", "simpsons" and "rickandmarty". See details section for more information. Can be also a numeric vector of length(groups); in this case a basic color palette is created using the function palette.
linetype	line types. Allowed values includes i) "strata" for changing linetypes by strata (i.e. groups); ii) a numeric vector (e.g., c(1, 2)) or a character vector c("solid", "dashed").
conf.int	logical value. If TRUE, plots confidence interval.
pval.method	whether to add a text with the test name used for calculating the pvalue, that corresponds to survival curves' comparison - used only when pval=TRUE
test.for.trend	logical value. Default is FALSE. If TRUE, returns the test for trend p-values. Tests for trend are designed to detect ordered differences in survival curves. That is, for at least one group. The test for trend can be only performed when the number of groups is > 2.
surv.median.line	character vector for drawing a horizontal/vertical line at median survival. Allowed values include one of c("none", "hv", "h", "v"). v: vertical, h:horizontal.
risk.table	Allowed values include:(1)TRUE or FALSE specifying whether to show or not the risk table. Default is FALSE.(2)"absolute" or "percentage". Shows the absolute number and the percentage of subjects at risk by time, respectively.(3)"abs_pct" to show both absolute number and percentage.(4)"nrisk_cumcensor" and "nrisk_cumevents".

	Show the number at risk and, the cumulative number of censoring and events, respectively.
cumevents	logical value specifying whether to show or not the table of the cumulative number of events. Default is FALSE.
cumcensor	logical value specifying whether to show or not the table of the cumulative number of censoring. Default is FALSE.
tables.height	numeric value (in [0 - 1]) specifying the general height of all tables under the main survival plot.
add.all	a logical value. If TRUE, add the survival curve of pooled patients (null model) onto the main plot.
ggtheme	function, ggplot2 theme name. Default value is theme_survminer. Allowed values include ggplot2 official themes: see theme.

### Value

Return a list of riskscore and coefficient of cox regression.

### Examples

```
#get the path of the mutation annotation file and samples' survival data
maf<-system.file("extdata","data_mutations_extended.txt",package = "pathwayTMB")
sur_path<-system.file("extdata","sur.csv",package = "pathwayTMB")
sur<-read.csv(sur_path,header=TRUE,row.names = 1)
#perform the function 'get_mut_matrix'
mut_matrix<-get_mut_matrix(maffile=maf,mut_fre = 0.01,is.TCGA=FALSE,sur=sur)
#perform the function `get_PTMB`
PTMB_matrix<-get_PTMB(freq_matrix=mut_matrix,genesmbol=genesmbol,gene_path=gene_path)
set.seed(1)
final_character<-get_final_signature(PTMB=PTMB_matrix,sur=sur)
#plot the K-M survival curve
plotKMcurves(t(PTMB_matrix[final_character,]),sur=sur,risk.table = TRUE)
```

---

plotMutInteract	<i>Exact tests to detect mutually exclusive, co-occurring and altered genes or pathways.</i>
-----------------	--

---

### Description

Performs Pair-wise Fisher's Exact test to detect mutually exclusive or co-occurring events.

### Usage

```
plotMutInteract(
  freq_matrix,
  genes,
  pvalue = c(0.05, 0.01),
```

```

returnAll = TRUE,
fontSize = 0.8,
showSigSymbols = TRUE,
showCounts = FALSE,
countStats = "all",
countType = "all",
countsFontSize = 0.8,
countsFontColor = "black",
colPal = "BrBG",
nShiftSymbols = 5,
sigSymbolsSize = 2,
sigSymbolsFontSize = 0.9,
pvSymbols = c(46, 42),
limitColorBreaks = TRUE
)

```

### Arguments

<code>freq_matrix</code>	The mutations matrix, generated by <code>'get_mut_matrix'</code> .
<code>genes</code>	List of genes or pathways among which interactions should be tested.
<code>pvalue</code>	Default <code>c(0.05, 0.01)</code> p-value threshold. You can provide two values for upper and lower threshold.
<code>returnAll</code>	If TRUE returns test statistics for all pair of tested genes. Default FALSE, returns for only genes below pvalue threshold.
<code>fontSize</code>	cex for gene names. Default 0.8.
<code>showSigSymbols</code>	Default TRUE. Highlight significant pairs.
<code>showCounts</code>	Default TRUE. Include number of events in the plot.
<code>countStats</code>	Default 'all'. Can be 'all' or 'sig'.
<code>countType</code>	Default 'cooccur'. Can be 'all', 'cooccur', 'mutexcl'.
<code>countsFontSize</code>	Default 0.8.
<code>countsFontColor</code>	Default 'black'.
<code>colPal</code>	<code>colPalBrewer</code> palettes. See <code>RColorBrewer::display.brewer.all()</code> for details.
<code>nShiftSymbols</code>	shift if positive shift <code>SigSymbols</code> by <code>n</code> to the left, default = 5.
<code>sigSymbolsSize</code>	size of symbols in the matrix and in legend.
<code>sigSymbolsFontSize</code>	size of font in legends.
<code>pvSymbols</code>	vector of pch numbers for symbols of p-value for upper and lower thresholds <code>c(upper, lower)</code> .
<code>limitColorBreaks</code>	limit color to extreme values. Default TRUE.

### Value

list of data.tables

**Examples**

```
#get the path of the mutation annotation file and samples' survival data
maf<-system.file("extdata","data_mutations_extended.txt",package = "pathwayTMB")
sur_path<-system.file("extdata","sur.csv",package = "pathwayTMB")
sur<-read.csv(sur_path,header=TRUE,row.names = 1)
#perform the function 'get_mut_matrix'
mut_matrix<-get_mut_matrix(maffile=maf,mut_fre = 0.01,is.TCGA=FALSE,sur=sur)
#perform the function `get_PTMB`
PTMB_matrix<-get_PTMB(freq_matrix=mut_matrix,genesmbol=genesmbol,gene_path=gene_path)
set.seed(1)
final_character<-get_final_signature(PTMB=PTMB_matrix,sur=sur)
plotMutInteract(freq_matrix=PTMB_matrix, genes=final_character,nShiftSymbols =0.3)
```

plotROC

*plot the ROC curve***Description**

This function uses to plot a ROC curve.

**Usage**

```
plotROC(
  riskscore,
  response,
  main,
  add = FALSE,
  col = par("col"),
  legacy.axes = TRUE,
  print.auc = FALSE,
  grid = FALSE,
  auc.polygon = FALSE,
  auc.polygon.col = "skyblue",
  max.auc.polygon = FALSE,
  max.auc.polygon.col = "#EEEEEE"
)
```

**Arguments**

riskscore	a numeric vector of the same length than response, containing the predicted value of each observation.
response	a factor, numeric or character vector of responses (true class), typically encoded with 0 (controls) and 1 (cases). Only two classes can be used in a ROC curve.
main	the title of the ROC curve
add	if TRUE, the ROC curve will be added to an existing plot. If FALSE (default), a new plot will be created.

col	the color of the ROC curve
legacy.axes	a logical indicating if the specificity axis (x axis) must be plotted as as decreasing “specificity” (FALSE) or increasing “1 - specificity” (TRUE, the default) as in most legacy software. This affects only the axis, not the plot coordinates.
print.auc	boolean. Should the numeric value of AUC be printed on the plot?
grid	boolean or numeric vector of length 1 or 2. Should a background grid be added to the plot? Numeric: show a grid with the specified interval between each line; Logical: show the grid or not. Length 1: same values are taken for horizontal and vertical lines. Length 2: grid value for vertical (grid[1]) and horizontal (grid[2]). Note that these values are used to compute grid.v and grid.h. Therefore if you specify a grid.h and grid.v, it will be ignored.
auc.polygon	boolean. Whether or not to display the area as a polygon.
auc.polygon.col	color (col) for the AUC polygon.
max.auc.polygon	boolean. Whether or not to display the maximal possible area as a polygon.
max.auc.polygon.col	color (col) for the maximum AUC polygon.

## Value

No return value

## Examples

```
#get the path of the mutation annotation file and samples' survival data
maf<-system.file("extdata","data_mutations_extended.txt",package = "pathwayTMB")
sur_path<-system.file("extdata","sur.csv",package = "pathwayTMB")
sur<-read.csv(sur_path,header=TRUE,row.names = 1)
#perform the function 'get_mut_matrix'
mut_matrix<-get_mut_matrix(maffile=maf,mut_fre = 0.01,is.TCGA=FALSE,sur=sur)
#perform the function `get_PTMB`
PTMB_matrix<-get_PTMB(freq_matrix=mut_matrix,genesmbol=genesmbol,gene_path=gene_path)
set.seed(1)
final_character<-get_final_signature(PTMB=PTMB_matrix,sur=sur)
#calculate the riskscore
riskscore<-plotKMcurves(t(PTMB_matrix[final_character,]),sur=sur,plots=FALSE)$risk_score
#get the path of samples' immunotherapy response data
res_path<- system.file("extdata","response.csv",package = "pathwayTMB")
response<-read.csv(res_path,header=TRUE,stringsAsFactors =FALSE,row.name=1)
plotROC(riskscore=riskscore,response=response,main="Objective Response",print.auc=TRUE)
```

---

PTMB_matrix	<i>PTMB_matrix, the Pathway-based Tumor Mutational Burden matrix</i>
-------------	--

---

**Description**

PTMB\_matrix, the pathway tumor mutation burden matrix, generated by 'get\_PTMB'

**Usage**

PTMB\_matrix

**Format**

An object of class data.frame with 27 rows and 35 columns.

---

sur	<i>sur, the samples' survival data</i>
-----	--

---

**Description**

sur, a nx2 data frame, the samples' survival data

**Usage**

sur

**Format**

An object of class data.frame with 110 rows and 2 columns.

# Index

## \* datasets

- final\_character, [2](#)
- gene\_path, [5](#)
- genesmbol, [5](#)
- mut\_matrix, [9](#)
- PTMB\_matrix, [15](#)
- sur, [15](#)

final\_character, [2](#)

- gene\_path, [5](#)
- GenePathwayOncoplots, [2](#)
- genesmbol, [5](#)
- get\_final\_signature, [6](#)
- get\_gene\_length, [6](#)
- get\_mut\_matrix, [7](#)
- get\_PTMB, [8](#)

mut\_matrix, [9](#)

- plotKMcurves, [9](#)
- plotMutInteract, [11](#)
- plotROC, [13](#)
- PTMB\_matrix, [15](#)

sur, [15](#)