Package: survtype (via r-universe)

November 25, 2026

Type Package

Title Subtype Identification with Survival Data

Description Subtypes are defined as groups of samples that have distinct molecular and clinical features. Genomic data can be analyzed for discovering patient subtypes, associated with clinical data, especially for survival information. This package is aimed to identify subtypes that are both clinically relevant and biologically meaningful.

Version 1.30.0

Date 2019-12-18

Author Dongmin Jung

Maintainer Dongmin Jung <dmdmjung@gmail.com>

Depends SummarizedExperiment, pheatmap, survival, survminer, clustvarsel, stats, utils

Suggests maftools, scales, knitr, rmarkdown

License Artistic-2.0

biocViews Software, StatisticalMethod, GeneExpression, Survival, Clustering, Sequencing, Coverage

NeedsCompilation no

VignetteBuilder knitr

Config/pak/sysreqs cmake make libicu-dev libjpeg-dev libpng-dev libxml2-dev libssl-dev zlib1g-dev

Repository Bioconductor 3.24

Date/Publication 2026-11-23 22:42:29 UTC

RemoteUrl https://github.com/bioc/survtype

RemoteRef RELEASE_3_24

RemoteSha bb7ca86b71d81fa8d6649c68ba6d09d58eaeebe7

Contents

Exprs.survtype 2
gene.clust 4
MAF.survgroup 5
maf2matrix 6
plot.survtype 7
quantile_normalization 8
Single.survgroup 8
Surv.survtype 10

Index **12**

`Exprs.survtype` *Sample subtype identification via survival information and gene expression data*

Description

For discovery of subtypes of samples that are both clinically relevant and biologically meaningful, the Cox regession model and hierarchical clustering are combined.

Usage

```
Exprs.survtype(surv.data, time, status, exprs.data, K = 2, num.genes = 100,
               gene.sel = FALSE, gene.sel.opt = list(verbose = FALSE), ...)
```

Arguments

`surv.data`	survival data
`time`	survival time
`status`	status indicator
`exprs.data`	expression data
`K`	the number of clusters (default: 2)
`num.genes`	the number of top genes based on the Cox score, before variable selection (default: 100)
`gene.sel`	a logical value indicating whether or not gene selection for clustring is applied (default: FALSE)
`gene.sel.opt`	a list of options for the gene selection function "clustvarsel". "verbose" is set to FALSE as default.
`...`	additional parameters for the "pheatmap"

Value

n	the number of subjects in each group
obs	the weighted observed number of events in each group
exp	the weighted expected number of events in each group
chisq	the chi-squared statistic for a test of equality
call	the matched call
fit	fitted survival curves
cluster	a vector of integers indicating the cluster to which each sample is assigned
time	survival time
status	status indicator
surv.data	survival data
exprs.data	expression data

Author(s)

Dongmin Jung

References

Bair, E., & Tibshirani, R. (2004). Semi-supervised methods to predict patient survival from gene expression data. PLoS biology, 2(4), e108.

See Also

survival::Surv, survival::survfit, survival::survdiff, survival::coxph, clustvarsel::clustvarsel, pheatmap::pheatmap

Examples

```
set.seed(1)
nrows <- 5
ncols <- nrow(ovarian)
counts <- matrix(runif(nrows * ncols, 1, 1e4), nrows)
colnames(counts) <- paste("X", 1:ncols, sep = "")
rownames(ovarian) <- paste("X", 1:ncols, sep = "")
SE <- SummarizedExperiment(assays = SimpleList(counts = counts))
ovarian.survtype <- Exprs.survtype(ovarian, time = "futime", status = "fustat",
                                   assay(SE), num.genes = 2, scale = "row",
                                   clustering_method = "ward.D2")
plot(ovarian.survtype, pval = TRUE)
```

gene.clust *Plots of the heatmap for each cluster of expression data*

Description

Heatmaps of clustered genes for subtypes of samples can be drawn.

Usage

```
gene.clust(object, K, ...)
```

Arguments

object	the result of "Exprs.survtype"
K	the number of clusters
...	additional parameters for the "pheatmap"

Value

Heatmap for each cluster

Author(s)

Dongmin Jung

See Also

pheatmap::pheatmap

Examples

```
set.seed(1)
nrows <- 5
ncols <- nrow(ovarian)
counts <- matrix(runif(nrows * ncols, 1, 1e4), nrows)
colnames(counts) <- paste("X", 1:ncols, sep = "")
rownames(ovarian) <- paste("X", 1:ncols, sep = "")
SE <- SummarizedExperiment(assays = SimpleList(counts = counts))
ovarian.survtype <- Exprs.survtype(ovarian, time = "futime", status = "fustat",
                                   assay(SE), num.genes = 5, scale = "row",
                                   clustering_method = "ward.D2")
plot(ovarian.survtype, pval = TRUE)
gene.clust(ovarian.survtype, 2, scale = "row", clustering_method = "ward.D2")
```

`MAF.survgroup` *Patient group identification via survival data and mutation annotation information*

Description

The groups of patients are identified according to whether the varints exist on a single gene. Survival difference between patients with and without mutations is compared.

Usage

```
MAF.survgroup(surv.data, time, status, maf, variants = NULL,
              sample.name = "Tumor_Sample_Barcode", gene.name = "Hugo_Symbol",
              variant.type = "Variant_Classification", num.genes = 10,
              siginificant.genes = 1, ...)
```

Arguments

`surv.data`	survival data
`time`	survival time
`status`	status indicator
`maf`	a MAF file
`variants`	types of varints on a single gene for mutated samples. samples with any mutations, defined as mutated samples by default.
`sample.name`	the column name containing sample names (defult: "Tumor_Sample_Barcode")
`gene.name`	the column name containing gene names (defult: "Hugo_Symbol")
`variant.type`	the column name containing variant types (defult: "Variant_Classification")
`num.genes`	the number of top genes based on the number of mutated genes (default: 10)
`siginificant.genes`	the number of top genes based on the statistical siginificance of mutated genes (default: 1)
`...`	additional parameters for the "ggsurvplot" for the statistically siginificant genes

Value

`time`	survival time
`status`	status indicator
`surv.data`	survival data
`maf.matrix`	a mutation matrix
`summary`	a list of number of samples with variants, chi-squared statistics and p-values
`cluster`	a vector of integers indicating the cluster to which each sample is assigned, for the most significant gene
`fit`	fitted survival curves, for the most significant gene

Author(s)

Dongmin Jung

See Also

survival::Surv, survival::survfit, survival::survdiff, survminer::ggsurvplot

Examples

```
library(maftools)
laml.maf <- system.file('extdata', 'tcga_laml.maf.gz', package = 'maftools', mustWork = TRUE)
laml.clin <- system.file('extdata', 'tcga_laml_annot.tsv', package = 'maftools', mustWork = TRUE)
laml.maf <- read.csv(laml.maf, sep = "\t")
laml.clinical.data <- read.csv(laml.clin, sep = "\t", row.names = 1)
index <- which(laml.clinical.data$days_to_last_followup == -Inf)
laml.clinical.data <- laml.clinical.data[-index,]
laml.clinical.data <- data.frame(laml.clinical.data)
laml.survgroup <- MAF.survgroup(laml.clinical.data, time = "days_to_last_followup",
                                status = "Overall_Survival_Status", laml.maf,
                                num.genes = 3, siginificant.genes = 1,
                                pval = TRUE)
```

---

maf2matrix *Convert a MAF file to a mutation matrix*

Description

Create a mutation matrix using variant types

Usage

```
maf2matrix(maf, surv.data = NULL, sample.name = "Tumor_Sample_Barcode",
           gene.name = "Hugo_Symbol", variant.type = "Variant_Classification")
```

Arguments

maf	a MAF file
surv.data	survival data for sample names (default: NULL)
sample.name	the column name containing sample names (defult: "Tumor_Sample_Barcode")
gene.name	the column name containing gene names (defult: "Hugo_Symbol")
variant.type	the column name containing variant types (defult: "Variant_Classification")

Value

a mutation matrix

Author(s)

Dongmin Jung

Examples

```
laml.maf <- system.file("extdata", "tcga_laml.maf.gz", package = "maftools")
laml.maf <- read.csv(laml.maf, sep = "\t")
laml.mat <- maf2matrix(laml.maf)
```

---

plot.survtype *Plot of survival curves of sample subtypes*

Description

Survival curves for subtypes of samples can be drawn.

Usage

```
## S3 method for class 'survtype'
plot(object, ...)
```

Arguments

object	object of class "survtype"
...	additional parameters for the "ggsurvplot"

Value

Survival curves

Author(s)

Dongmin Jung

See Also

survminer::ggsurvplot

Examples

```
data(ovarian)
ovarian.survtype <- Surv.survtype(ovarian, time = "futime", status = "fustat")
plot(ovarian.survtype, pval = TRUE)
```

quantile_normalization *Normalize a gene expression profile*

Description

Normalize expression data using quantile normalization

Usage

```
quantile_normalization(x)
```

Arguments

x	an expression profile

Value

a normalized matrix

Author(s)

Dongmin Jung

Examples

```
set.seed(1)
nrows <- 10
ncols <- 5
counts <- matrix(runif(nrows * ncols, 1, 1e4), nrows)
normalized.matrix <- quantile_normalization(counts)
```

Single.survgroup *Patient group identification via survival information and expression of a single gene*

Description

All midpoints of the expression level or real-valued statistic are investigated to find the best threshold giving the most significant difference between two groups. Any patients having the value greater than the best cutoff are assigned as the "high-score" class. Otherwise, the others belong to the "low-score" class.

Usage

```
Single.survgroup(surv.data, time, status, single.gene, intermediate = FALSE,
                 group.names = c("High", "Intermediate", "Low"))
```

Arguments

surv.data	survival data
time	survival time
status	status indicator
single.gene	expression level of a single gene
intermediate	a logical value indicating whether or not the intermediate class is considered (default: FALSE)
group.names	the name of clusters for "high-score", "intermediate-score", and "low-score" classes (defult: "High", "Intermediate", "Low")

Value

time	survival time
status	status indicator
surv.data	survival data
score	a vector of scores
summary	a list of thresholds, chi-squared statistics and p-values
cluster	a vector of clusters to which samples are assigned
fit	fitted survival curves

Author(s)

Dongmin Jung

See Also

survival::Surv, survival::survfit, survival::survdiff

Examples

```
set.seed(1)
nrows <- 1
ncols <- nrow(ovarian)
counts <- matrix(runif(nrows * ncols, 1, 1e4), nrows)
colnames(counts) <- paste("X", 1:ncols, sep = "")
rownames(ovarian) <- paste("X", 1:ncols, sep = "")
Single.ovarian <- Single.survgroup(ovarian, time = "futime", status = "fustat", counts[1,])
plot(Single.ovarian, pval = TRUE)
```

`Surv.survtype` *Sample subtype identification via survival information*

Description

Any patient who lived longer than the median was considered to be a "low-risk" patient, and any patient that lived less than the median was considered to be a "high-risk"" patient. In this manner, we assigned a class label to each observation. For censored data, we can estimate the probability that each censored observation belongs to the "low-risk" and "high-risk" classes, respectively.

Usage

```
Surv.survtype(surv.data, time, status)
```

Arguments

`surv.data`	survival data
`time`	survival time
`status`	status indicator

Value

`n`	the number of subjects in each group
`obs`	the weighted observed number of events in each group
`exp`	the weighted expected number of events in each group
`chisq`	the chi-squared statistic for a test of equality
`call`	the matched call
`cluster`	a vector of clusters to which samples are assigned
`time`	survival time
`status`	status indicator
`surv.data`	survival data
`fit`	fitted survival curves

Author(s)

Dongmin Jung

References

Bair, E., & Tibshirani, R. (2004). Semi-supervised methods to predict patient survival from gene expression data. PLoS biology, 2(4), e108.

See Also

survival::Surv, survival::survfit, survival::survdiff

Examples

```
data(ovarian)
ovarian.survtype <- Surv.survtype(ovarian, time = "futime", status = "fustat")
```

Index

Exprs.survtype, 2

gene.clust, 4

MAF.survgroup, 5
maf2matrix, 6

plot *(*plot.survtype*)*, 7
plot.survtype, 7

quantile_normalization, 8

Single.survgroup, 8
Surv.survtype, 10