Package 'survClust'

November 29, 2024

Title Identification Of Clinically Relevant Genomic Subtypes Using
Outcome Weighted Learning

Version 1.0.0

Date 2024-04-16

Description survClust is an outcome weighted integrative clustering algorithm used to classify multi-
omic samples on their available time to event information. The resulting clusters are cross-
validated to avoid over overfitting and output classification of samples that are molecularly dis-
tinct and clinically meaningful. It takes in binary (mutation) as well as continu-
ous data (other omic types).

VignetteBuilder knitr

License MIT + file LICENSE

Imports Rcpp, MultiAssayExperiment, pdist, survival

LinkingTo Rcpp

URL https://github.com/arorarshi/survClust

BugReports https://support.bioconductor.org/t/survClust

biocViews Software, Clustering, Survival, Classification

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

LazyData true

Depends R (>= 3.5.0)

Suggests knitr, testthat (>= 3.0.0), gplots, htmltools, BiocParallel

Config/testthat/edition 3

git_url https://git.bioconductor.org/packages/survClust

git_branch RELEASE_3_20

git_last_commit bb5b1d7

git_last_commit_date 2024-10-29

Repository Bioconductor 3.20

Date/Publication 2024-11-28

Author Arshi Arora [aut, cre] (<https://orcid.org/0000-0002-4040-1787>)

Maintainer Arshi Arora <arshiaurora@gmail.com>

Contents

---

combineDist   *Integrates weighted distance matrices*

Description

combineDist integrates weighted distances matrices from getDist. All data types are now collapsed into one NxN matrix.

Usage

```
combineDist(dist.dat)
```

Arguments

dist.dat   list of weighted data matrices from getDist

Details

combineDist integrates and does cleaning of missing pair of samples. if datasets list had non-overlapping samples, then combineDist retains only those samples that have full information after accounting for all data types.

Value

- combMatFullA matrix. Combine normalized information across m genomic data types into NxN matrix, where N is the union of all samples across m data types/ or samples with complete pairwise information. Final matrix should not have any NAs

Author(s)

Arshi Arora

Examples

```
library(survClust)
dd <- getDist(simdat, simsurvdat)
cc <- combineDist(dd)
```

---

cv_survclust	*performs cross validation on supervised clustering,* survClust *for a particular* k. cv_survclust *runs*

Description

cv_survclust performs k fold cross-validation, runs survClust on each training and hold out test fold and return cross-validated supervised cluster labels.

Usage

```
cv_survclust(datasets, survdat = NULL, k, fold, cmd.k = NULL, type = NULL)
```

Arguments

datasets	A list object containing m data matrices representing m different genomic data types measured in a set of N~m samples. OR [MultiAssayExperiment](#) object of desired types of data. For list of matrices, each matrix, the rows represent samples, and the columns represent genomic features. Each data matrix is allowed to have different samples
survdat	A matrix, containing two columns - 1st column time and 2nd column containing events information. OR this information can be provided as a part of colData MultiAssayExperiment
k	integer, choice of k to perform clustering on samples
fold	integer, number of folds to run cross validation
cmd.k	integer, number of dimensions used by cmdscale to perform clustering on samples. Defaults is n-1
type	Specify type="mut", if datasets is of length 1 and contains binary data only.

Value

- cv.labelsreturns cross validated class labels for k cluster
- cv.logranklogrank test statistic of cross validated label
- cv.spwssstandardized pooled within-cluster sum of squares calculated from cross-validation class labels

Author(s)

Arshi Arora

Examples

```
library(survClust)
cv.fit <- cv_survclust(datasets = simdat, survdat = simsurvdat, k = 3, fold=3 )
```

---

cv_voting   *For a* survClust *fit, return consolidated labels across rounds of cross validation for a specific* k. *Note that cv.fit already has consolidated class labels across folds*

Description

For a survClust fit, return consolidated labels across rounds of cross validation for a specific k. Note that cv.fit already has consolidated class labels across folds

Usage

```
cv_voting(
  cv.fit,
  dat.dist,
  pick_k,
  cmd.k = NULL,
  pick_k.test = TRUE,
  minlabel.test = TRUE
)
```

Arguments

cv.fit	fit objects as returned from cv_survclust
dat.dist	weighted distance matrices from getDist
pick_k	choice of k cluster to summarize over rounds of cross validation
cmd.k	number of dimensions used by cmdscale to perform clustering on. Defaults is n-1
pick_k.test	logical, only selects cv.fit solutions where the resulting solution after consolidation contains pick_k classes. Default TRUE. Avoids edge cases, but in some cases FALSE might be desirable
minlabel.test	logical, only selects cv.fit solutions where classes have a minimum of 5 samples. Default TRUE. Avoids edge cases, but in some cases FALSE might be desirable

Value

final.labels consolidated class labels over rounds of cross-validation

Author(s)

Arshi Arora

Examples

```
library(survClust)
k4 <- cv_voting(uvm_survClust_cv.fit, getDist(uvm_dat, uvm_survdat), pick_k = 4)
table(k4)
```

---

getDist   *Calculates weighted distance matrix of multiple genomic data types*

Description

Given multiple genomic data types (e.g., gene expression, copy number, DNA methylation, miRNA expression (continuous) and mutation (binary)) measured across samples, allowing for missing values (NA) and missing samples, getDist calculates the survival weighted distance metric among samples. Used as an input to, combineDist().

Usage

```
getDist(datasets, survdat = NULL, cv = FALSE, train.snames = NULL, type = NULL)
```

Arguments

datasets	A list object containing m data matrices representing m different genomic data types measured in a set of N~m samples. OR [MultiAssayExperiment](#) object of desired types of data. For list of matrices, each matrix, the rows represent samples, and the columns represent genomic features. Each data matrix is allowed to have different samples
survdat	A matrix, containing two columns - 1st column time and 2nd column containing events information. OR this information can be provided as a part of colData MultiAssayExperiment
cv	logical. If TRUE, train.names cannot be NULL. Cross-validation will be performed on train.names samples, and the dataset will be split into training and test, and each respective matrices will be returned.
train.snames	required if cv=TRUE. A vector of sample names treated as training samples.
type	NULL. Specify type="mut", if datasets is of length 1 and contains binary data only. See details

Details

getDist allows for continuous and binary data type(s) in a matrix passed as a list. If the list only has a binary matrix data type. Set type="mut". All data types are standardized internally. All data types are not expected to have common samples. Non-overlapping samples within data types are replaced with NA, and returned weighted matrix consists of union of all the samples.

Value

- cv=FALSE,dist.datreturns a list of weighted data matrix/matrices, dist.dat

- cv=TRUE,dist.dat=list(train, all) returns a list of training train weighted data matrix. And the whole matrix weighed according to the weights computed on the training dataset all.

Author(s)

Arshi Arora

Examples

```
library(survClust)
dd <- getDist(simdat, simsurvdat)
```

---

getStats   *Compute fit statistics after cross validation via* cv_survclust

---

Description

Compute fit statistics after cross validation via `cv_survclust`

Usage

```
getStats(cv.fit, kk = 8, cvr = 50)
```

Arguments

cv.fit	output from `cv_survclust` object
kk	number of k clusters on which `cv_survclust` was run, default is 8
cvr	round of cross-validation on which `cv_survclust` was run, default is 50

Details

`getStats` calculates Logrank statistic and standardized pooled within sum of squares (SPWSS) across cross-validated labels. Visualize it via `plotStats`

Value

A list of the following

- lr log rank statistic
- spwss standardized pooled within sum of squares
- bad.sol number of solutions for each kk that have cluster class <5 samples

Author(s)

Arshi Arora

Examples

```
library(survClust)
ss_stats <- getStats(uvm_survClust_cv.fit, kk=7, cvr=10)
```

---

plotStats   *Plot the output from* getStats

Description

Plot the output from getStats

Usage

```
plotStats(out.getStats, labels = NULL, ...)
```

Arguments

out.getStats	list output from getStats
labels	labels to print on the boxplot. Default is 2:8
...	additional arguments as passed to boxplot function

Details

plots boxplots summarizing output of cv.survclust calculated via getStats. Use this to pick optimal k. Optimal k maximized logrank and minimizes SPWSS similar to the elbow method. Use consensus_summary to pick the best k and arrive at unique consolidated class labels

Value

a plot with three boxplots summarizing logrank, standardized pooled within sum of squares (SP-WSS) and if any class label has less than 5 samples

Author(s)

Arshi Arora

Examples

```
library(survClust)
ss_stats <- getStats(uvm_survClust_cv.fit, kk=7, cvr=10)
plotStats(ss_stats, 2:7)
```

---

simdat   *Simulated dataset with 3-class solution*

Description

A list of length 1 with a matrix simulated with 150 samples x 150 features with a 3-class structure such that 15 features are distinct and associated with survival, other 15 features are just distinct and not associated with survival and remaining 120 are noise. See how this dataset was generated in the vignette

Usage

```
data(simdat)
```

Format

An object of class `"list"`

Examples

```
data(simdat)
class(simdat)
dim(simdat[[1]])
simdat[[1]][1:5,1:5]
```

---

simsurvdat   *Simulated survival dataset with accompanying* simdat

---

Description

A matrix with simulate time-event data with 150 samples x 2 columns with a 3-class structure with median survival of 4.5, 3.25 and 2 yrs respectively. such that 15 features are distinct and associated with survival, other 15 features are just distinct and not associated with survival and remaining 120 are noise. See how this dataset was generated in the vignette

Usage

```
data(simsurvdat)
```

Format

An object of class `"matrix"`

Examples

```
data(simsurvdat)
dim(simsurvdat)
head(simsurvdat)
```

---

survClust   *perform supervised clustering for a particular* k

---

Description

`survClust` function performs supervised clustering on a `combineDist` output for a particular k. It uses all `n-1` dimensions for clustering.

`survClust` is an outcome weighted integrative clustering algorithm used to classify multi-omic samples on their available time to event information.

Usage

```
survClust(combine.dist, survdat, k, cmd.k = NULL)
```

Arguments

combine.dist	integrated weighted distance matrix from combineDist
survdat	A nx2 matrix consisting of survival data with n samples and first column as time and second column as events, with samples as rownames
k	choice of k to perform clustering on samples
cmd.k	number of dimensions used by cmdscale to perform clustering on samples. Defaults is n-1

Value

- fit returns a list , fit consisting of all clustering samples as in kmeans fit.lr, computed logrank statistic between k clusters

Author(s)

Arshi Arora

Maintainer: Arshi Arora <arshiaurora@gmail.com> (ORCID)

See Also

Useful links:

- https://github.com/arorarshi/survClust
- Report bugs at https://support.bioconductor.org/t/survClust

Examples

```
library(survClust)
dd <- getDist(datasets = simdat, survdat = simsurvdat)
cc <- combineDist(dd)
survclust_fit <- survClust(combine.dist = cc, survdat = simsurvdat, k = 3)
```

---

uvm_dat   *TCGA UVM Mutation and Copy Number datasets*

Description

A list of length 2 with TCGA UVM Mutation data with 80 samples and 87 genes TCGA UVM Copy Number data with 80 samples and 749 segments. See Appendix in vignette for more details. Teh data is downloaded from here https://gdc.cancer.gov/about-data/publications/pancanatlas

Usage

```
data(uvm_dat)
```

Format

An object of class "list"

Examples

```
data(uvm_dat)
uvm_dat[[1]][1:5,1:5]
uvm_dat[[2]][1:5,1:5]
```

---

uvm_survClust_cv.fit   *survClust cv.survclust output of integrated TCGA UVM Mutation and Copy Number datasets.*

Description

The output is a list object consisting of 6 sub-lists for k = 2:7, with 10 cv.survclust outputs (for each round of cross-validation), each consisting of cv.labels, cv.logrank, cv.spwss for 3 folds.

Usage

```
data(uvm_survClust_cv.fit)
```

Format

An object of class "list"

Examples

```
data(uvm_survClust_cv.fit)
names(uvm_survClust_cv.fit[[1]][[1]])
```

---

uvm_survdat   *TCGA UVM Clinical file*

Description

A matrix with 2 columns, with first column as OS.time and second column as OS events. The data is downloaded from - https://gdc.cancer.gov/about-data/publications/pancanatlas

Usage

```
data(uvm_survdat)
```

Format

An object of class "matrix"

Examples

```
data(uvm_survdat)
head(uvm_survdat)
```

Index