Package 'clustcurv'

March 21, 2020

Type Package

Title Determining Groups in Multiples Curves

URL https://github.com/noramvillanueva/clustcurv

BugReports http://github.com/noramvillanueva/clustcurv/issues

Version 2.0.0

Date 2020-03-21

Maintainer Nora M. Villanueva <nmvillanueva@uvigo.es>

Description A method for determining groups in multiple
curves with an automatic selection of their number based on k-means or
k-medians algorithms. The selection of the optimal number is provided by
bootstrap methods. The methodology can be applied both in regression and survival framework.
Implemented methods are:
Grouping multiple survival curves described by Villanueva et al. (2018) <doi:10.1002/sim.8016>.

Depends R (>= 3.5.0)

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports doParallel, foreach, ggplot2, ggfortify, doRNG, Gmedian,
survival, wesanderson, npregfast, tidyr, RColorBrewer,
KernSmooth, data.table

Suggests testthat, usethis, condSURV, covr

RoxygenNote 6.1.1

NeedsCompilation no

Author Nora M. Villanueva [aut, cre] (<https://orcid.org/0000-0001-8085-2745>),
Marta Sestelo [aut]

Repository CRAN

Date/Publication 2020-03-21 16:00:02 UTC

R topics documented:

`autoclustcurv` *Clustering multiple curves*

Description

Function for grouping survival or regression curves based on the k-means or k-medians algorithm. It returns the number of groups and the assignement.

Usage

```
autoclustcurv(y, x, z, weights = NULL, method = "survival",
  kvector = NULL, kbin = 50, h = -1, nboot = 100,
  algorithm = "kmeans", alpha = 0.05, cluster = FALSE,
  ncores = NULL, seed = NULL, multiple = FALSE,
  multiple.method = "holm")
```

Arguments

`y`	Survival time (method = 'survival') or response variable (method = 'regression').
`x`	Only for method = 'regression'. Dependent variable.
`z`	Categorical variable indicating the population to which the observations belongs.
`weights`	Only for method = 'survival'. Censoring indicator of the survival time of the process; 0 if the total time is censored and 1 otherwise.
`method`	A character string specifying which method is used, 'survival' or 'regression'.
`kvector`	A vector specifying the number of groups of curves to be checking.
`kbin`	Size of the grid over which the survival functions are to be estimated.
`h`	The kernel bandwidth smoothing parameter (for method = 'regression').
`nboot`	Number of bootstrap repeats.
`algorithm`	A character string specifying which clustering algorithm is used, i.e., k-means(`"kmeans"`) or k-medians (`"kmedians"`).
`alpha`	Significance level of the testing procedure. Defaults to 0.05.

`cluster` A logical value. If `TRUE` (default), the testing procedure is parallelized. Note that there are cases (e.g., a low number of bootstrap repetitions) that R will gain in performance through serial computation. R takes time to distribute tasks across the processors also it will need time for binding them all together later on. Therefore, if the time for distributing and gathering pieces together is greater than the time need for single-thread computing, it does not worth parallelize.

`ncores` An integer value specifying the number of cores to be used in the parallelized procedure. If `NULL` (default), the number of cores to be used is equal to the number of cores of the machine - 1.

`seed` Seed to be used in the procedure.

`multiple` A logical value. If `TRUE` (not default), the resulted pvalues are adjunted by using one of several methods for multiple comparisons.

`multiple.method` Correction method. See Details.

Details

The adjustment methods include the Bonferroni correction ("bonferroni") in which the p-values are multiplied by the number of comparisons. Less conservative corrections are also included by Holm (1979) ('holm'), Hochberg (1988) ('hochberg'), Hommel (1988) ('hommel'), Benjamini & Hochberg (1995) ('BH' or its alias 'fdr'), and Benjamini & Yekutieli (2001) ('BY'), respectively. A pass-through option ('none') is also included.

Value

A list containing the following items:

`table` A data frame containing the null hypothesis tested, the values of the test statistics and the obtained pvalues.

`levels` Original levels of the variable z.

`cluster` A vector of integers (from 1:k) indicating the cluster to which each curve is allocated.

`centers` An object containing the centroids (mean of the curves pertaining to the samet group).

`curves` An object containing the fitted curves for each population.

Author(s)

Nora M. Villanueva and Marta Sestelo.

Examples

```
library(clustcurv)
library(survival)
library(condSURV)
data(veteran)
data(colonCS)
```

```
# Survival framework
res <- autoclustcurv(y = veteran$time, z = veteran$celltype,
weights = veteran$status, method = 'survival', algorithm = 'kmeans')

# Regression framework
res2 <- autoclustcurv(y = barnacle5$DW, x = barnacle5$RC, z = barnacle5$F,
method = 'regression', algorithm = 'kmeans', nboot = 20)
```

`autoplot.clustcurv` *Visualization of* `clustcurv` *objects with ggplot2 graphics*

Description

Useful for drawing the estimated functions grouped by color and the centroids (mean curve of the curves pertaining to the same group).

Usage

```
## S3 method for class 'clustcurv'
autoplot(object = object, groups_by_colour = TRUE,
  centers = FALSE, conf.int = FALSE, censor = FALSE, xlab = "Time",
  ylab = "Survival", ...)
```

Arguments

`object`	Object of `clustcurv` class.
`groups_by_colour`	A specification for the plotting groups by color.
`centers`	Draw the centroids (mean of the curves pertaining to the same group) into the plot. By default it is `FALSE`.
`conf.int`	Only for method = "survival". Logical flag indicating whether to plot confidence intervals.
`censor`	Only for method = "survival". Logical flag indicating whether to plot censors.
`xlab`	A title for the `x` axis.
`ylab`	A title for the `y` axis.
`...`	Other options.

Details

See help page of the function `autoplot.survfit`.

Value

A ggplot object, so you can use common features from ggplot2 package to manipulate the plot.

Author(s)

Nora M. Villanueva and Marta Sestelo.

Examples

```
library(survival)
library(clustcurv)
library(condSURV)
library(ggplot2)
library(ggfortify)

# Survival

data(veteran)
data(colonCS)

cl2 <- kclustcurv(y = veteran$time, weights = veteran$status,
z = veteran$celltype, k = 2, method = "survival", algorithm = "kmeans")

autoplot(cl2)
autoplot(cl2, groups_by_colour = FALSE)
autoplot(cl2, centers = TRUE)

# Regression

r2 <- kclustcurv(y = barnacle5$DW, x = barnacle5$RC,
z = barnacle5$F, k = 2, method = "regression", algorithm = "kmeans")

autoplot(r2)
autoplot(r2, groups_by_colour = FALSE)
autoplot(r2, centers = TRUE)


colonCSm <- data.frame(time = colonCS$Stime, status = colonCS$event,
                       nodes = colonCS$nodes)

table(colonCSm$nodes)
colonCSm$nodes[colonCSm$nodes == 0] <- NA
colonCSm <- na.omit(colonCSm)
colonCSm$nodes[colonCSm$nodes >= 10] <- 10
table(colonCSm$nodes) # ten levels

res <- autoclustcurv(y = colonCSm$time, weights = colonCSm$status,
       z = colonCSm$nodes, method = "survival", algorithm = "kmeans",
       nboot = 20)

autoplot(res)
autoplot(res, groups_by_colour = FALSE)
autoplot(res, centers = TRUE)
```

`barnacle5` *Barnacle data*

Description

This barnacle data set gives the measurements of the variables dry weight (in g.) and rostro-carinal lenght (in mm) for 5000 barnacles collected along the intertidal zone from five sites of the Atlantic coast of Galicia (Spain).

Usage

```
barnacle5
```

Format

`barnacle5` is a data frame with 5000 cases (rows) and 3 variables (columns).

Note that `barnacle` data set from the `npregfast` package gives the same three variables (columns) but for two sites, thus 2000 cases (rows).

DW Dry weight (in g.)

RC Rostro-carinal lenght (in mm).

F Factor indicating the sites of harvest: `laxe`, `lens`, `barca`, `laxe`, and `lens`.

Author(s)

Marta Sestelo

References

Sestelo, M. and Roca-Pardinas, J. (2011). A new approach to estimation of length-weight relationship of $Pollicipes\ pollicipes$ (Gmelin, 1789) on the Atlantic coast of Galicia (Northwest Spain): some aspects of its biology and management. Journal of Shellfish Research, 30(3), 939–948.

Sestelo, M., Villanueva, N.M., Meira-Machado, L., Roca-Pardinas, J. (2017). npregfast: An R Package for Nonparametric Estimation and Inference in Life Sciences. Journal of Statistical Software, 82(12), 1-27.

Examples

```
data(barnacle5)
head(barnacle5)
```

clustcurv clustcurv*: Determining Groups in Multiple Curves.*

Description

This package provides a method for determining groups in multiple curves with an automatic selection of their number based on k-means or k-medians algorithms. The selection of the optimal number is provided by bootstrap methods. The methodology can be applied both in regression and survival framework.

Details

Package:	clustcurv
Type:	Package
License:	MIT + file LICENSE

`clustcurv` is designed along lines similar to those of other `R` packages. This software helps the user determine groups in multiple curves (survival and regression curves). In addition, it enables both numerical and graphical outputs to be displayed (by means of ggplot2). The package provides the `kclustcurv()` function that groups the curves given a number k and the `autoclustcurv()` function that selects the optimal number of groups automatically through a boostrap-based test. The `autoplot()` function let the user draws the resulted estimated curves coloured by groups.

For a listing of all routines in the clustcurv package type: `library(help="clustcurv")`.

Author(s)

Nora M. Villanueva and Marta Sestelo

References

Villanueva, N. M., Sestelo, M., and Meira-Machado, J. (2019). A method for determining groups in multiple survival curves. Statistics in Medicine, 8(5):866-877

See Also

Useful links:

- https://github.com/noramvillanueva/clustcurv
- Report bugs at http://github.com/noramvillanueva/clustcurv/issues

`kclustcurv` *k-groups of multiple curves*

Description

Function for grouping survival or regression curves, given a number k, based on the k-means or k-medians algorithm.

Usage

```
kclustcurv(y, x, z, weights = NULL, k, method = "survival",
  kbin = 50, h = -1, algorithm = "kmeans", seed = NULL)
```

Arguments

`y`	Survival time (method = "survival") or response variable (method = "regression").
`x`	Only for method = "regression". Dependent variable.
`z`	Categorical variable indicating the population to which the observations belongs.
`weights`	Only for method = "survival". Censoring indicator of the survival time of the process; 0 if the total time is censored and 1 otherwise.
`k`	An integer specifying the number of groups of curves to be performed.
`method`	A character string specifying which method is used, "survival" or "regression".
`kbin`	Size of the grid over which the survival functions are to be estimated.
`h`	The kernel bandwidth smoothing parameter (for method = "regression").
`algorithm`	A character string specifying which clustering algorithm is used, i.e., k-means(`"kmeans"`) or k-medians (`"kmedians"`).
`seed`	Seed to be used in the procedure.

Value

A list containing the following items:

`measure`	A measure of...
`levels`	Original levels of the variable `fac`.
`cluster`	A vector of integers (from 1:k) indicating the cluster to which each curve is allocated.
`centers`	An object of class `survfit` containing the centroids (mean of the curves pertaining to the same group).
`curves`	An object of class `survfit` containing the survival curves for each population.

Author(s)

Nora M. Villanueva and Marta Sestelo.

Examples

```
library(clustcurv)
library(survival)
data(veteran)

# Survival: 2 groups k-means
s2 <- kclustcurv(y = veteran$time, weights = veteran$status,
z = veteran$celltype, k = 2, method = "survival", algorithm = "kmeans")

data.frame(level = s2$level, cluster = s2$cluster)


# Survival: 2 groups k-medians
s22 <- kclustcurv(y = veteran$time, weights = veteran$status,
z = veteran$celltype, k = 2, method = "survival", algorithm = "kmedians")

data.frame(level = s22$level, cluster = s22$cluster)


# Regression: 2 groups k-means
r2 <- kclustcurv(y = barnacle5$DW, x = barnacle5$RC,
z = barnacle5$F, k = 2, method = "regression", algorithm = "kmeans")

data.frame(level = r2$level, cluster = r2$cluster)
```

Index