Package: packHV (via r-universe)

May 31, 2026

Type Package

Title A few Useful Functions for Statisticians

Version 2.4

Date 2025-03-24

Maintainer Hugo Varet <varethugo@gmail.com>

Depends survival

Imports stats, graphics, WriteXLS (>= 2.3.0)

Description Various useful functions for statisticians: describe data, plot Kaplan-Meier curves with numbers of subjects at risk, compare data sets, display spaghetti-plot, build multi-contingency tables...

License GPL (>= 2)

RoxygenNote 7.3.2

Config/pak/sysreqs perl

Repository https://hvaret.r-universe.dev

Date/Publication 2025-03-24 10:44:42 UTC

RemoteUrl https://github.com/hvaret/packhv

RemoteRef HEAD

RemoteSha 3d5c0739d8734a9794079e07d44848c0048645b5

Contents

`packHV-package` *A few useful functions for statisticians*

Description

Various useful functions for statisticians: describe data, plot Kaplan-Meier curves with numbers of subjects at risk, compare data sets, display spaghetti-plot, build multi-contingency tables...

Author(s)

Hugo Varet

`compare` *Comparing two databases assumed to be identical*

Description

Compares two data frames assumed to be identical, prints the differences in the console and also returns the results in a data frame

Usage

```
compare(d1, d2, id, file.export = NULL)
```

Arguments

`d1`	first data frame
`d2`	second data frame
`id`	character string, primary key of the two data bases
`file.export`	character string, name of the XLS file exported

Value

A data frame containing the differences between the two data bases

Author(s)

Hugo Varet

Examples

```
N=100
data1=data.frame(id=1:N,a=rnorm(N),
                         b=factor(sample(LETTERS[1:5],N,TRUE)),
                         c=as.character(sample(LETTERS[1:5],N,TRUE)),
                         d=as.Date(32768:(32768+N-1),origin="1900-01-01"))
data1$c=as.character(data1$c)
data2=data1
data2$id[3]=4654
data2$a[30]=NA
data2$a[31]=45
data2$b=as.character(data2$b)
data2$d=as.character(data2$d)
data2$e=rnorm(N)
compare(data1,data2,"id")
```

---

convert_factor *Convert variables of a data frame in factors*

Description

Converts variables of a data frame in factors

Usage

```
convert_factor(data, vars)
```

Arguments

data	the data frame in which we can find vars
vars	vector of character string of covariates

Value

The modified data frame

Author(s)

Hugo Varet

Examples

```
cgd$steroids
cgd$status
cgd=convert_factor(cgd,c("steroids","status"))
```

convert_zero_NA *Convert 0s in NA*

Description

Converts 0s in NA

Usage

```
convert_zero_NA(data, vars)
```

Arguments

data	the data frame in which we can find vars
vars	a character vector of covariates for which to transform 0s in NA

Value

The modified data frame

Author(s)

Hugo Varet

Examples

```
my.data=data.frame(x=rbinom(20,1,0.5),y=rbinom(20,1,0.5),z=rbinom(20,1,0.5))
my.data=convert_zero_NA(my.data,c("y","z"))
```

cut_quanti *Cut a quantitative variable in n equal parts*

Description

Cuts a quantitative variable in n equal parts

Usage

```
cut_quanti(x, n, ...)
```

Arguments

x	a numeric vector
n	numeric, the number of parts: 2 to cut according to the median, and so on...
...	other arguments to be passed in cut

Value

A factor vector

Author(s)

Hugo Varet

Examples

```
cut_quanti(cgd$height, 3)
```

`desc` *Making descriptive statistics*

Description

Makes descriptive statistics of a data frame according to a group covariate or not, can export the results

Usage

```
desc(
  data,
  vars,
  group = NULL,
  whole = TRUE,
  vars.labels = vars,
  group.labels = NULL,
  type.quanti = "mean",
  test.quanti = "param",
  test = TRUE,
  noquote = TRUE,
  justify = TRUE,
  digits = 2,
  file.export = NULL,
  language = "english"
)
```

Arguments

`data`	data frame to describe in which we can find `vars` and `group`
`vars`	vector of character strings of the covariates to describe
`group`	character string, statistics created for each levels of this covariate
`whole`	boolean, `TRUE` to add a column with the whole statistics when comparing groups (set to `FALSE` if `group=NULL`)
`vars.labels`	vector of character string for sweeter names of covariates in the output

group.labels	vector of character string for sweeter column names
type.quanti	character string, "med" to compute median [Q1;Q3], "mean" to compute mean (sd), "mean_med" to compute both mean (sd) and median [Q1;Q3] or "med_mm", "mean_mm" or "mean_med_mm" to add (min;max)
test.quanti	character string, "param" to compute parametric tests for quantitative covariates (t-test or ANOVA) or "nonparam" for non parametric tests (Wilcoxon test or Kruskal-Wallis test)
test	boolean, TRUE to perform tests (FALSE if group is NULL): Khi-2 or Fisher exact test for categorical covariates, t-test/ANOVA or Wilcoxon/Kruskal-Wallis Rank Sum Test for numerical covariates
noquote	boolean, TRUE to hide quotes when printing the table
justify	boolean, TRUE to justify columns on right or left (FALSE if export)
digits	number of digits of the statistics (mean, sd, median, min, max, Q1, Q3, %), p-values always have 3 digits
file.export	character string, name of the XLS file exported
language	character string, "french" or "english"

Value

A matrix of the descriptive statistics

Author(s)

Hugo Varet

Examples

```
cgd$steroids=factor(cgd$steroids)
cgd$status=factor(cgd$status)
desc(cgd,vars=c("center","sex","age","height","weight","steroids","status"),group="treat")
```

---

hist_boxplot *Plot a histogram with a boxplot below*

Description

Plots a histogram with a boxplot below

Usage

```
hist_boxplot(
  x,
  freq = TRUE,
  density = FALSE,
  main = NULL,
```

```
  xlab = NULL,
  ymax = NULL,
  col.hist = "lightblue",
  col.boxplot = "lightblue",
  ...
)
```

Arguments

x	a numeric vector
freq	boolean, TRUE for frequency or FALSE probability on the y axis
density	boolean, TRUE to plot the estimated density
main	character string, main title of the histogram
xlab	character string, label of the x axis
ymax	numeric value, maximum of the y axis
col.hist	color of the histogram
col.boxplot	color of the boxplot
...	other arguments to be passed in hist()

Value

None

Author(s)

Hugo Varet

Examples

```
par(mfrow=c(1,2))
hist_boxplot(rnorm(100),col.hist="lightblue",col.boxplot="red",freq=TRUE)
hist_boxplot(rnorm(100),col.hist="lightblue",col.boxplot="red",freq=FALSE,density=TRUE)
```

IC_OR_glm *OR and their confidence intervals for logistic regressions*

Description

Computes odd ratios and their confidence intervals for logistic regressions

Usage

```
IC_OR_glm(model, alpha = 0.05)
```

Arguments

model	a glm object
alpha	type I error, 0.05 by default

Value

A matrix with the estimaed coefficients of the logistic model, their s.e., z-values, p-values, OR and CI of the OR

Author(s)

Hugo Varet

Examples

```
IC_OR_glm(glm(inherit~sex+age,data=cgd,family="binomial"))
```

IC_RR_coxph *RR and their confidence intervals for Cox models*

Description

Computess risk ratios and their confidence intervals for Cox models

Usage

```
IC_RR_coxph(model, alpha = 0.05, sided = 2)
```

Arguments

model	a coxph object
alpha	type I error, 0.05 by default
sided	1 or 2 for one or two-sided

Value

A matrix with the estimaed coefficients of the Cox model, their s.e., z-values, p-values, RR and CI of the RR

Author(s)

Hugo Varet

Examples

```
cgd$time=cgd$tstop-cgd$tstart
IC_RR_coxph(coxph(Surv(time,status)~sex+age,data=cgd),alpha=0.05,sided=1)
```

multi.table *Multi cross table*

Description

Builds a big cross table between several covariates

Usage

```
multi.table(data, vars)
```

Arguments

data	the data frame in which we can find vars
vars	vector of character string of covariates

Value

A matrix containing all the contingency tables between the covariates

Author(s)

Hugo Varet

See Also

plot_multi.table

Examples

```
multi.table(cgd,c("treat","sex","inherit"))
```

plot_km *Kaplan-Meier plot with number of subjects at risk below*

Description

Kaplan-Meier plot with number of subjects at risk below

Usage

```
plot_km(
  formula,
  data,
  test = TRUE,
  xy.pvalue = NULL,
  conf.int = FALSE,
  times.print = NULL,
  nrisk.labels = NULL,
  legend = NULL,
  xlab = NULL,
  ylab = NULL,
  ylim = c(0, 1.02),
  left = 4.5,
  bottom = 5,
  cex.mtext = par("cex"),
  lwd = 2,
  lty = 1,
  col = NULL,
  ...
)
```

Arguments

formula	same formula than in survfit (Surv(time,cens)~group or Surv(time,cens)~1), where cens must equal to 0 (censorship) or 1 (failure)
data	data frame with time, cens and group
test	boolean, TRUE to compute and display the p-value of the log-rank test
xy.pvalue	numeric vector of length 2, coordinates where to display the p-value of the log-rank test
conf.int	boolean, TRUE to display the confidence interval of the curve(s)
times.print	numeric vector, times at which to display the numbers of subjects at risk
nrisk.labels	character vector to modify the levels of group in the table below the curve(s)
legend	character string ("bottomright" for example) or numeric vector (c(x,y)), where to place the legend of the curve(s)
xlab	character string, label of the time axis
ylab	character string, label of the y axis
ylim	numeric vector of length 2, minimum and maximum of the y-axis
left	integer, size of left margin
bottom	integer, number of lines in addition of the table below the graph
cex.mtext	numeric, size of the numbers of subjects at risk
lwd	width of the Kaplan-Meier curve(s)
lty	type of the Kaplan-Meier curve(s)
col	color(s) of the Kaplan-Meier curve(s)
...	other arguments to be passed in plot.survfit

Value

None

Author(s)

Hugo Varet

Examples

```
cgd$time=cgd$tstop-cgd$tstart
plot_km(Surv(time,status)~sex,data=cgd,col=c("blue","red"))
```

`plot_mm` *Spaghetti plot and plot of the mean at each time*

Description

Spaghetti plot and plot of the mean at each time

Usage

```
plot_mm(
  formula,
  data,
  col.spag = 1,
  col.mean = 1,
  type = "spaghettis",
  tick.times = TRUE,
  xlab = NULL,
  ylab = NULL,
  main = "",
  lwd.spag = 1,
  lwd.mean = 4,
  ...
)
```

Arguments

`formula`	`obs~time+(group|id)` or `obs~time+(1|id)`
`data`	data frame in which we can find `obs`, `time`, `group` and `id`
`col.spag`	vector of length `nrow(data)` with colors (one for each individual)
`col.mean`	vector of length `length(levels(group))` with colors (one for each group)
`type`	`"spaghettis"`, `"mean"` or `"both"`
`tick.times`	boolean, `TRUE` to display ticks at each observation time on the x-axis
`xlab`	character sring, label of the time axis

ylab	character string, label of the y axis
main	character string, main title
lwd.spag	numeric, width of the spaghetti lines, 1 by default
lwd.mean	numeric, width of the mean lines, 4 by default
...	Other arguments to be passed in plot

Value

None

Author(s)

Hugo Varet on Anais Charles-Nelson's idea

Examples

```
N=10
time=rep(1:4,N)
obs=1.1*time + rep(0:1,each=2*N) + rnorm(4*N)
my.data=data.frame(id=rep(1:N,each=4),time,obs,group=rep(1:2,each=N*2))
par(xaxs="i",yaxs="i")
plot_mm(obs~time+(group|id),my.data,col.spag=my.data$group,
        col.mean=c("blue","red"),type="both",main="Test plot_mm")
```

---

plot_multi.table *Plot a multi cross table*

Description

Plots a multi cross table on a graph

Usage

```
plot_multi.table(data, vars, main = "")
```

Arguments

data	the data frame in which we can find vars
vars	vector of character string of covariates
main	main title of the plot

Value

None

Author(s)

Hugo Varet

See Also

multi.table

Examples

```
plot_multi.table(cgd,c("treat","sex","inherit"))
```

---

plot_reg *Plot points with the corresponding linear regression line*

Description

Plots points with the corresponding linear regression line

Usage

```
plot_reg(x, y, pch = 19, xlab = NULL, ylab = NULL, ...)
```

Arguments

x	numeric vector
y	numeric vector
pch	type of points
xlab	character string, label of the x axis, NULL by default
ylab	character string, label of the y axis, NULL by default
...	other arguments to be passed in plot

Value

None

Author(s)

Hugo Varet

Examples

```
plot_reg(cgd$age, cgd$height, xlab="Age (years)", ylab="Height")
```

Index