

# Package ‘descr’

July 22, 2025

**Version** 1.1.8

**Date** 2023-11-27

**Title** Descriptive Statistics

**Author** Jakson Aquino. Includes R source code and/or documentation written by DirkENZmann, Marc Schwartz, Nitin Jain, and Stefan Kraft

**Maintainer** Jakson Aquino <jalvesaq@gmail.com>

**Imports** xtable, utils, grDevices, graphics, stats

**Description** Weighted frequency and contingency tables of categorical variables and of the comparison of the mean value of a numerical variable by the levels of a factor, and methods to produce xtable objects of the tables and to plot them. There are also functions to facilitate the character encoding conversion of objects, to quickly convert fixed width files into csv ones, and to export a data.frame to a text file with the necessary R and SPSS codes to reread the data.

**License** GPL (>= 2)

**URL** <https://github.com/jalvesaq/descr>

**NeedsCompilation** yes

**Repository** CRAN

**Date/Publication** 2023-11-27 19:20:11 UTC

## Contents

compmeans . . . . .	2
crosstab . . . . .	3
CrossTable . . . . .	6
data.frame2txt . . . . .	9
descr . . . . .	10
file.head . . . . .	11
forODFTable . . . . .	12
freq . . . . .	13
fromUTF8 . . . . .	14

fwf2csv . . . . .	15
histkdnc . . . . .	16
labels2R . . . . .	17
LogRegR2 . . . . .	19
plot.CrossTable . . . . .	19
plot.freqtable . . . . .	20
toUTF8 . . . . .	21
xtable.CrossTable . . . . .	22

## Index 23

---

compmeans	<i>Means of a numerical vector according to a factor</i>
-----------	--

---

### Description

Calculates the means of a numerical vector according to a factor.

### Usage

```
compmeans(x, f, w, sort = FALSE, maxlevels = 60,
          user.missing, missing.include = FALSE,
          plot = getOption("descr.plot"),
          relative.widths = TRUE, col = "lightgray",
          warn = getOption("descr.warn"), ...)
```

### Arguments

x	A numeric vector.
f	A factor.
w	Optional vector with weights.
sort	If TRUE, sorts the lines by the means values.
maxlevels	Maximum number of levels that x converted into factor should have.
user.missing	Character vector, indicating what levels of f must be treated as missing values.
missing.include	If TRUE, then NA values, if present in f, are included as level "NA". You can change the new level label by setting the value of descr.na.replacement option. Example: options(descr.na.replacement = "Missing").
plot	Logical: if TRUE (default), a boxplot is produced. You may put options(descr.plot = FALSE) in your '.Rprofile' to change the default function behavior.
relative.widths	If TRUE, the boxes widths will be proportional to the number of elements in each level of f.
col	Vector with the boxes colors.

warn	Warn if conversion from factor into numeric or from numeric into factor was performed and if missing values were dropped (default: TRUE).
...	Further arguments to be passed to either <a href="#">boxplot</a> (if w is missing) or <a href="#">bxp</a> (for w weighted boxplot).

### Value

A matrix with class `c("matrix", "meanscomp")` with labels attributes for x and f. The returned object can be plotted, generating a [boxplot](#) of x grouped by f.

### Author(s)

Jakson A. Aquino <jalvesaq@gmail.com>, with code for weighted boxplots written by Stefan Kraft for simPopulation package.

### See Also

[boxplot](#).

### Examples

```
sex <- factor(c(rep("F", 900), rep("M", 900)))
income <- 100 * (rnorm(1800) + 5)
weight <- rep(1, 1800)
weight[sex == "F" & income > 500] <- 3
attr(income, "label") <- "Income"
attr(sex, "label") <- "Sex"
compmeans(income, sex, col = "lightgray", ylab = "income", xlab = "sex")
comp <- compmeans(income, sex, weight, plot = FALSE)
plot(comp, col = c("pink", "lightblue"), ylab = "income", xlab = "sex")

library(xtable)
# If the decimal separator in your country is a comma:
# options(OutDec = ",")
print(xtable(comp, caption = "Income according to sex", label = "tab:incsx"))
```

---

crosstab

---

*Cross tabulation with mosaic plot*


---

### Description

This function is a wrapper for [CrossTable](#), adding a mosaic plot and making it easier to do a weighted cross-tabulation.

**Usage**

```
crosstab(dep, indep, weight = NULL,
         digits = list(expected = 1, prop = 3, percent = 1, others = 3),
         max.width = NA,
         expected = FALSE, prop.r = FALSE, prop.c = FALSE, prop.t = FALSE,
         prop.chisq = FALSE, chisq = FALSE, fisher = FALSE, mcnemar = FALSE,
         resid = FALSE, sresid = FALSE, asresid = FALSE,
         missing.include = FALSE, drop.levels = TRUE, format = "SPSS",
         cell.layout = TRUE, row.labels = !cell.layout,
         percent = (format == "SPSS" && !row.labels),
         total.r, total.c, dnn = "label", xlab = NULL,
         ylab = NULL, main = "", user.missing.dep, user.missing.indep,
         plot = getOption("descr.plot"), ...)
```

**Arguments**

dep, indep	Vectors in a matrix or a dataframe. dep should be the dependent variable, and indep should be the independent one.
weight	An optional vector for a weighted cross tabulation.
digits	See <a href="#">CrossTable</a> .
max.width	See <a href="#">CrossTable</a> .
expected	See <a href="#">CrossTable</a> .
prop.r	See <a href="#">CrossTable</a> .
prop.c	See <a href="#">CrossTable</a> .
prop.t	See <a href="#">CrossTable</a> .
prop.chisq	See <a href="#">CrossTable</a> .
chisq	See <a href="#">CrossTable</a> .
fisher	See <a href="#">CrossTable</a> .
mcnemar	See <a href="#">CrossTable</a> .
resid	See <a href="#">CrossTable</a> .
sresid	See <a href="#">CrossTable</a> .
asresid	See <a href="#">CrossTable</a> .
missing.include	See <a href="#">CrossTable</a> .
drop.levels	See <a href="#">CrossTable</a> .
format	See <a href="#">CrossTable</a> .
cell.layout	See <a href="#">CrossTable</a> .
row.labels	See <a href="#">CrossTable</a> .
percent	See <a href="#">CrossTable</a> .
total.r	See <a href="#">CrossTable</a> .
total.c	See <a href="#">CrossTable</a> .

dnn	See <a href="#">CrossTable</a> . If dnn = "label", then the "'label'" attribute of 'dep' and 'indep' will be used as the dimension names.
xlab	See <a href="#">plot.default</a> .
ylab	See <a href="#">plot.default</a> .
main	An overall title for the plot (see <a href="#">plot.default</a> and <a href="#">title</a> ).
user.missing.dep	An optional character vector with the levels of dep that should be treated as missing values.
user.missing.indep	An optional character vector with the levels of indep that should be treated as missing values.
plot	Logical: if TRUE (default), a mosaic plot is produced. You may put <code>options(descr.plot = FALSE)</code> in your '.Rprofile' to change the default function behavior.
...	Further arguments to be passed to <a href="#">mosaicplot</a> .

### Details

crosstab invokes the [CrossTable](#) with all boolean options set to FALSE and "SPSS" as the default format option. The returned CrossTable object can be plotted as a [mosaicplot](#). Note that the gray scale colors used by default in the mosaic plot do not have any statistical meaning. The colors are used only to ease the plot interpretation.

Differently from [CrossTable](#), this function requires both dep and indep arguments. If you want an univariate tabulation, you should try either [CrossTable](#) or [freq](#).

By default, if weight has decimals, the result of xtabs is rounded before being passed to [CrossTable](#). If you prefer that the results are not rounded, add to your code:

```
options(descr.round.xtabs = FALSE)
```

### Author(s)

Jakson A. Aquino <jalvesaq@gmail.com>

### See Also

[CrossTable](#), [plot.CrossTable](#), [xtable.CrossTable](#).

### Examples

```
educ <- sample(c(1, 2), 200, replace = TRUE, prob = c(0.3, 0.7))
educ <- factor(educ, levels = c(1, 2), labels = c("Low", "High"))
opinion <- sample(c(1, 2, 9), 200, replace = TRUE,
  prob = c(0.4, 0.55, 0.05))
opinion <- factor(opinion, levels = c(1, 2, 9),
  labels = c("Disagree", "Agree", "Don't know"))
attr(educ, "label") <- "Education level"
attr(opinion, "label") <- "Opinion"
weight <- sample(c(10, 15, 19), 200, replace = TRUE)
```

```

crosstab(opinion, educ, xlab = "Education", ylab = "Opinion")
ct <- crosstab(opinion, educ, weight,
               dnn = c("Opinion", "Education"),
               user.missing.dep = "Don't know",
               expected = TRUE, prop.c = TRUE, prop.r = TRUE,
               plot = FALSE)

ct
plot(ct, inv.y = TRUE)

# Get the table of observed values as an object of class "table"
tab <- ct$tab
class(tab)
tab

# Get the complete cross table as "matrix"
complete.tab <- descr::CreateNewTab(ct)
class(complete.tab)
complete.tab

## xtable support
library(xtable)

# Print ugly table
print(xtable(ct))

# Print pretty table
# Add to the preamble of your Rnoweb document:
# \usepackage{booktabs}
# \usepackage{multirow}
# \usepackage{dcolumn}
# \newcolumntype{d}{D{.}{.}{-1}}
print(xtable(ct, align = "l1ddd", multirow = TRUE, hline = TRUE,
             row.labels = TRUE, percent = FALSE,
             caption = "Opinion according to level of education"),
      booktabs = TRUE, include.rownames = FALSE,
      sanitize.text.function = function(x) x)

```

---

CrossTable

---

*Cross tabulation with tests for factor independence*


---

## Description

An implementation of a cross-tabulation function with output similar to S-Plus `crosstabs()` and SAS Proc Freq (or SPSS format) with Chi-square, Fisher and McNemar tests of the independence of all table factors.

## Usage

```
CrossTable(x, y,
```

```

digits = list(expected = 1, prop = 3, percent = 1, others = 3),
max.width = NA, expected = FALSE,
prop.r = TRUE, prop.c = TRUE, prop.t = TRUE,
prop.chisq = TRUE, chisq = FALSE, fisher = FALSE,
mcnemar = FALSE, resid = FALSE, sresid = FALSE,
asresid = FALSE, missing.include = FALSE,
drop.levels = TRUE, format = c("SAS", "SPSS"),
dnn = NULL, cell.layout = TRUE,
row.labels = !cell.layout,
percent = (format == "SPSS" && !row.labels),
total.r, total.c, xlab = NULL, ylab = NULL, ...)

```

### Arguments

<code>x</code>	A vector or a matrix. If <code>y</code> is specified, <code>x</code> must be a vector.
<code>y</code>	A vector in a matrix or a dataframe.
<code>digits</code>	Named list with number of digits after the decimal point for four categories of statistics: expected values, cell proportions, percentage and others statistics. It can also be a numeric vector with a single number if you want the same number of digits in all statistics.
<code>max.width</code>	In the case of a 1 x n table, the default will be to print the output horizontally. If the number of columns exceeds <code>max.width</code> , the table will be wrapped for each successive increment of <code>max.width</code> columns. If you want a single column vertical table, set <code>max.width</code> to 1.
<code>prop.r</code>	If TRUE, row proportions will be included.
<code>prop.c</code>	If TRUE, column proportions will be included.
<code>prop.t</code>	If TRUE, table proportions will be included.
<code>expected</code>	If TRUE, expected cell counts from the $\chi^2$ will be included.
<code>prop.chisq</code>	If TRUE, chi-square contribution of each cell will be included.
<code>chisq</code>	If TRUE, the results of a chi-square test will be printed after the table.
<code>fisher</code>	If TRUE, the results of a Fisher Exact test will be printed after the table.
<code>mcnemar</code>	If TRUE, the results of a McNemar test will be printed after the table.
<code>resid</code>	If TRUE, residual (Pearson) will be included.
<code>sresid</code>	If TRUE, standardized residual will be included.
<code>asresid</code>	If TRUE, adjusted standardized residual will be included.
<code>missing.include</code>	If TRUE, then NA values, if present, are included as level "NA" of both <code>x</code> and <code>y</code> . You can change the new level label by setting the value of <code>descr.na.replacement</code> option. Example: <code>options(descr.na.replacement = "Missing")</code> .
<code>drop.levels</code>	If TRUE, then remove any unused factor levels.
<code>format</code>	Either SAS (default) or SPSS, depending on the type of output desired.
<code>dnn</code>	The names to be given to the dimensions in the result (the <code>dimnames</code> names).
<code>cell.layout</code>	If TRUE, print the cell layout.

<code>row.labels</code>	If TRUE, add labels to rows of calculated statistics.
<code>percent</code>	A logical value indicating whether to add the percentage symbol 'prop.r', 'prop.c' and 'prop.t' if 'format' is "SPSS".
<code>.</code>	
<code>total.r</code>	If TRUE, print row totals.
<code>total.c</code>	If TRUE, print column totals.
<code>xlab</code>	A title for the x axis when plotting the CrossTable object (see <a href="#">title</a> ). If missing, <code>dnn[1]</code> is used if not NULL.
<code>ylab</code>	A title for the y axis when plotting the CrossTable object (see <a href="#">title</a> ). If missing, <code>dnn[2]</code> is used if not NULL.
<code>...</code>	Optional arguments passed to <a href="#">chisq.test</a> .

### Details

A summary table will be generated with cell row, column and table proportions and marginal totals and proportions. Expected cell counts can be printed if desired. In the case of a 2 x 2 table, both corrected and uncorrected values will be included for appropriate tests. In the case of tabulating a single vector, cell counts and table proportions will be printed.

Note 1: If 'x' is a vector and 'y' is not specified, no statistical tests will be performed, even if any are set to TRUE.

Note 2: 'x' and 'y' labels will be truncated if the table is not going to fit to the screen, according to the value of `getOption("width")`.

If both arguments 'total.c' and 'total.r' are missing, both will be TRUE. If only one of them is missing, the other will have the same value of the not missing one.

### Value

A list of class `CrossTable` containing parameters used by the `print.CrossTable` method and the following components:

`tab`: An n by m matrix containing table cell counts.

`prop.row`: An n by m matrix containing cell row proportions.

`prop.col`: An n by m matrix containing cell column proportions.

`prop.tbl`: An n by m matrix containing cell table proportions.

`chisq`: Results from the Chi-Square test. A list with class 'htest'. See [chisq.test](#) for details.

`chisq.corr`: Results from the corrected Chi-Square test. A list with class 'htest'. See [chisq.test](#) for details. ONLY included in the case of a 2 x 2 table.

`fisher.ts`: Results from the two-sided Fisher Exact test. A list with class 'htest'. See [fisher.test](#) for details. ONLY included if 'fisher' = TRUE.

`fisher.lt`: Results from the Fisher Exact test with HA = "less". A list with class 'htest'. See [fisher.test](#) for details. ONLY included if 'fisher' = TRUE and in the case of a 2 x 2 table.

`fisher.gt`: Results from the Fisher Exact test with HA = "greater". A list with class 'htest'. See [fisher.test](#) for details. ONLY included if 'fisher' = TRUE and in the case of a 2 x 2 table.



mcnemar: Results from the McNemar test. A list with class 'htest'. See [mcnemar.test](#) for details. ONLY included if 'mcnemar' = TRUE.

mcnemar.corr: Results from the corrected McNemar test. A list with class 'htest'. See [mcnemar.test](#) for details. ONLY included if 'mcnemar' = TRUE and in the case of a 2 x 2 table.

resid/sresid/asresid: Pearson Residuals (from chi-square tests).

### Author(s)

Jakson Aquino <jalvesaq@gmail.com> has splited the function CrossTable (from the package gmodels) in two: CrossTable and print.CrossTable. The gmodels's function was developed by Marc Schwartz (original version posted to r-devel on Jul 27, 2002. SPSS format modifications added by Nitin Jain based upon code provided by Dirk Enzmann).

### See Also

[crosstab](#) (a wrapper to 'CrossTable' that makes it easier to do a weighted contingency table), [plot.CrossTable](#), [forODFTable](#), [table](#), [prop.table](#), [xtabs](#).

### Examples

```
# Simple cross tabulation of education versus prior induced
# abortions using infertility data
data(warpbreaks, package = "datasets")
ct <- CrossTable(warpbreaks$wool, warpbreaks$tension,
                 dnn = c("Wool", "Tension"))
data(esoph, package = "datasets")
ct <- CrossTable(esoph$alcgp, esoph$agegp, expected = TRUE,
                 chisq = FALSE, prop.chisq = FALSE,
                 dnn = c("Alcohol consumption", "Tobacco consumption"))
plot(ct, inv.y = TRUE)
print(ct)

# While printing the object, you can replace some (but not all)
# arguments previously passed to CrossTable
print(ct, format = "SPSS", cell.layout = FALSE, row.labels = TRUE)

# For better examples, including the use of xtable,
# see the documentation of crosstab().
```

---

data.frame2txt

---

*Export a data.frame and create scripts to input the data again.*


---

### Description

Export a data.frame to a tab delimited text and create R and SPSS/PSPP scripts to input the data again.

**Usage**

```
data.frame2txt(x, datafile = "x.txt", r.codefile = "x.R",
               sps.codefile = "x.sps", df.name = "x",
               user.missing)
```

**Arguments**

<code>x</code>	The data.frame to be exported.
<code>datafile</code>	The name of the tab delimited file to be created.
<code>r.codefile</code>	The name of the R script to read the data file.
<code>sps.codefile</code>	The name of the SPSS/PSPP script to read the data file.
<code>df.name</code>	The name of the data.frame object to be created by the R script.
<code>user.missing</code>	Labels of levels that must be coded as user missing in the sps script.

**Details**

Logical vectors are converted into numeric before being saved.

**Value**

The return value of `write.table`.

**Author(s)**

Jakson A. Aquino <jalvesaq@gmail.com>

**Examples**

```
## Not run:
data(CO2)
data.frame2txt(CO2)

## End(Not run)
```

---

 descr

*Summary of an object*


---

**Description**

Wrapper for the function `summary` of **base** package, including information about variable label. The function prints the `label` attribute of the object and, then, invokes `summary(object)`. If the object is a data frame, the function prints the `label` and invokes `summary` for each variable in the data frame.

**Usage**

```
descr(x)
```

**Arguments**

x                      The object to be described.

**Value**

Null.

**Author(s)**

Jakson Aquino <jalvesaq@gmail.com>

**See Also**

[summary](#)

---

file.head	<i>Prints first lines of a file.</i>
-----------	--------------------------------------

---

**Description**

The function prints the first lines of a file, optionally truncating the lines according to the screen width. The lines are truncated at `getOption("width") - 2`.

**Usage**

```
file.head(file, n, truncate.cols = TRUE)
```

**Arguments**

file                      Character: The name of the file whose first lines should be printed.

n                          The number of lines to show.

truncate.cols      Logical: if TRUE truncate the lines.

**Value**

NULL.

**Author(s)**

Jakson A. Aquino <jalvesaq@gmail.com>

---

forODFTable

---

*Convert an object of class CrossTable into a matrix for odfTable*


---

### Description

The function converts an object of class `CrossTable` into a matrix to be printed by `'odfTable()'` of **odfWeave** package.

### Usage

```
forODFTable(x, digits = 1, ...)
```

### Arguments

<code>x</code>	A object of class <code>'CrossTable'</code> .
<code>digits</code>	See <a href="#">round</a> .
<code>...</code>	Optional arguments passed to <a href="#">format</a> .

### Value

A matrix.

### Author(s)

Jakson A. Aquino <jalvesaq@gmail.com>.

### See Also

[CrossTable](#)

### Examples

```
## Not run:
library(odfWeave)
data(infert, package = "datasets")
x <- crosstab(infert$education, infert$induced, expected = TRUE)

# Use the function directly:
odfTable(forODFTable(x))

# Create a method for odfTable:
odfTable.CrossTable <- function(x) odfTable(forODFTable(x))
odfTable(x)
methods(odfTable)

## End(Not run)
```

---

freq	<i>Frequency table</i>
------	------------------------

---

**Description**

Prints a frequency table of the selected object. Optionally, the frequency might be weighted.

**Usage**

```
freq(x, w, user.missing, plot = getOption("descr.plot"), ...)
```

**Arguments**

x	The factor from which the frequency of values is desired.
w	An optional vector for a weighted frequency table.
user.missing	Character vector, indicating what levels must be treated as missing values while calculating valid percents. Levels representing user missing values are not shown in the <a href="#">barplot</a> .
plot	Logical: if TRUE (default), a barplot is produced. You may put <code>options(descr.plot = FALSE)</code> in your <code>‘.Rprofile’</code> to change the default function behavior.
...	Further arguments to be passed to <a href="#">plot.freqtable</a> if <code>plot = TRUE</code> .

**Details**

A column with cumulative percents are added to the frequency table if x is an ordered factor.

**Value**

A matrix with class `c("matrix", "freqtable")` with the attribute `"xlab"` which is a character string corresponding to either the attribute `"label"` of x or, if x does not have this attribute, the name of x. The returned object can be plotted, generating a [barplot](#).

**Author(s)**

Jakson A. Aquino <jalvesaq@gmail.com>, based on function written by DirkENZmann

**Examples**

```
x <- c(rep(1, 100), rep(2, 120), rep(3, 10), rep(NA, 12))
w <- c(rep(1.1, 122), rep(0.9, 120))
x <- factor(x, levels = c(1, 2, 3),
            labels = c("No", "Yes", "No answer"))
attr(x, "label") <- "Do you agree?"

freq(x, y.axis = "percent")
f <- freq(x, w, user.missing = "No answer", plot = FALSE)
```

```
f
plot(f)

# If the decimal separator in your country is a comma:
# options(OutDec = ",")
library(xtable)
print(xtable(f))
```

---

fromUTF8

*Conversion from UTF-8 encoding*

---

## Description

Converts the encoding of some attributes of an object from UTF-8 into other encoding.

## Usage

```
fromUTF8(x, to = "WINDOWS-1252")
```

## Arguments

x	A R object, usually a variable of a data frame or a data frame.
to	A string indicating the desired encoding. Common values are "LATIN1" and "WINDOWS-1252". Type <code>iconvlist()</code> for the complete list of available encodings.

## Details

The function converts the attribute `label` of `x` from UTF-8 into the specified encoding. If `x` is a factor, the levels are converted as well. If `x` is a `data.frame`, the function makes the conversions in all of its variables.

## Value

The object with its label and levels converted.

## Author(s)

Jakson A. Aquino <jalvesaq@gmail.com>.

## See Also

[iconv](#)

---

fwf2csv*Fast conversion of a fwf file into a csv one*

---

## Description

Convert fixed width formatted file into a tab separated one.

## Usage

```
fwf2csv(fwffile, csvfile, names, begin, end,  
        verbose = getOption("verbose"))
```

## Arguments

fwffile	The fixed width format file.
csvfile	The csv file to be created. The fields will be separated by tab characters and there will be no quotes around strings.
names	A character vector with column names.
begin	A numeric vector with the begin offset of values in the fixed width format file.
end	A numeric vector with the end offset of values in the fixed width format file.
verbose	Logical: if TRUE a message about the number of saved lines is printed.

## Details

The return value is NULL, but csvfile is created if the function is successful. The file is a text table with fields separated by tabular characters without quotes around the strings.

This function is useful if you have a very big fixed width formatted file to read and [read.fwf](#) would be too slow. The function that does the real job is very fast because it is written in C, and the use of RAM is minimum.

## Value

NULL.

## Author(s)

Jakson A. Aquino <jalvesaq@gmail.com>

## See Also

For an efficient way of reading a csv file, see the function ‘`fread()`’ from **data.table** package.

**Examples**

```

txt_file <- tempfile()
csv_file <- tempfile()

# Column:      12345678901234567
writeLines(c("CE  1  11M43 2000",
             "CE  1  12F40 1800",
             "CE  1  13F  9   0",
             "CE  1  13M  6   0",
             "CE  2  21F36 1200",
             "CE  2  23M  6   0",
             "BA  1  11M33 2100",
             "BA  1  12F34 2300",
             "BA  1  13M10   0",
             "BA  1  13F  7   0",
             "BA  2  21F26 3600",
             "BA  2  22M27 3200",
             "BA  2  23F  2   0"),
           con = txt_file)

tab <- rbind(c("state",  1,  2),
             c("municp", 3,  5),
             c("house",  6,  8),
             c("cond",   9,  9),
             c("sex",    10, 10),
             c("age",    11, 12),
             c("income", 13, 17))

fwf2csv(txt_file, csv_file,
        names = tab[, 1],
        begin = as.numeric(tab[, 2]),
        end = as.numeric(tab[, 3]))
d <- read.table(csv_file, header = TRUE,
               sep = "\t", quote = "")
d$cond <- factor(d$cond, levels = c(1, 2, 3),
               labels = c("Reference", "Spouse", "Child"))
d$sex <- factor(d$sex)
d

```

---

histkdnc

*Histogram with kernel density and normal curve*


---

**Description**

Plots a histogram with kernel density and normal curve.

**Usage**

```
histkdnc(v, breaks = 0, include.lowest = TRUE, right = TRUE,
```



```

main = "Histogram with kernel density and normal curve",
xlab = deparse(substitute(v)), col = grey(0.90),
col.cur = c("red", "blue"), lty.cur = c(1, 1),
xlim = NULL, ylim = NULL, ...)

```

### Arguments

<code>v</code>	The object from which the histogram is desired.
<code>breaks</code>	See <a href="#">hist</a> .
<code>include.lowest</code>	See <a href="#">hist</a> .
<code>right</code>	See <a href="#">hist</a> .
<code>main</code>	See <a href="#">hist</a> .
<code>xlab</code>	See <a href="#">hist</a> .
<code>col</code>	See <a href="#">hist</a> .
<code>col.cur</code>	Vector of size two with the colors of, respectively, kernel density and normal curve.
<code>lty.cur</code>	Vector of size two with line type of, respectively, kernel density and normal curve.
<code>xlim</code>	See <a href="#">plot.default</a> and <a href="#">hist</a> .
<code>ylim</code>	See <a href="#">plot.default</a> and <a href="#">hist</a> .
<code>...</code>	Further arguments to be passed to <a href="#">hist</a> .

### Details

The function plots a histogram of the object `x` with its kernel density and a normal curve with the same mean and standard deviation of `x`.

### Value

NULL.

### Author(s)

Dirk Enzmann (modified by Jakson Aquino<jalvesaq@gmail.com>).

### Description

Convert a specially written text file with information on variable labels and value labels into R code that converts integer vectors into factor variables.

**Usage**

```
labels2R(lfile, rfile, dfname = "b", echo = FALSE)
```

**Arguments**

<code>lfile</code>	The path to the text file to be converted.
<code>rfile</code>	The path to the file to be created.
<code>dfname</code>	Name of data.frame where the variables are.
<code>echo</code>	If TRUE, then lines of <code>lfile</code> are printed in the R Console while the file is parsed. This may be useful debugging.

**Details**

The return value is NULL, but `rfile` is created if the function is successful. The file is an R code that converts numeric vectors into factors. The text file must have a format as in the example below:

```
v1 Sex
1 Female
2 Male

v2 Household income

v3 Taking all things together, would you say you are...
1 Very happy
2 Rather happy
3 Not very happy
4 Not at all happy
```

The above code would be converted into:

```
b$v1 <- factor(b$v1, levels=c(1, 2), labels=c("Female", "Male"))
attr(b$v1, "label") <- "Sex"
attr(b$v2, "label") <- "Household income"
b$v3 <- factor(b$v3, levels=c(1, 2, 3, 4),
              labels=c("Very happy", "Rather happy",
                      "Not very happy", "Not at all happy"))
attr(b$v3, "label") <- "Taking all things together, would you say you are..."
```

**Value**

NULL.

**Author(s)**

Jakson A. Aquino <jalvesaq@gmail.com>

---

LogRegR2

*Pseudo R2 of logistic regression*


---

**Description**

The function calculates multiple R2 analogues (pseudo R2) of logistic regression.

**Usage**

```
LogRegR2(model)
```

**Arguments**

`model`                      A logistic regression model.

**Details**

The function calculates McFaddens R2, Cox & Snell Index, and Nagelkerke Index of a logistic regression model.

**Value**

A object of class `list` with the calculated indexes.

**Author(s)**

Dirk Enzmann

---

plot.CrossTable

*Mosaic plot from object of class CrossTable*


---

**Description**

This function receives a [CrossTable](#) object as its main argument and produces a mosaicplot.

**Usage**

```
## S3 method for class 'CrossTable'
plot(x, xlab, ylab, main = "", col,
      inv.x = FALSE, inv.y = FALSE, ...)
```

**Arguments**

x	A object of class CrossTable.
xlab	See <a href="#">plot.default</a> .
ylab	See <a href="#">plot.default</a> .
main	See <a href="#">plot.default</a> and <a href="#">title</a> .
col	A specification for the default plotting color. (See section ‘Color Specification’ of <a href="#">par</a> ). If the argument is missing, a gray scale is used to make the plot easier to interpret.
inv.x	A logical value indicating whether the order of the levels of the x variable should be inverted.
inv.y	A logical value indicating whether the order of the levels of the y variable should be inverted.
...	Further arguments to be passed to <a href="#">mosaicplot</a> .

**Author(s)**

Jakson A. Aquino <jalvesaq@gmail.com>

**See Also**

[CrossTable](#), [crosstab](#).

---

plot.freqtable	<i>Bar plot from object of class freqtable</i>
----------------	--

---

**Description**

This function receives a freqtable object as its main argument and produces a barplot.

**Usage**

```
## S3 method for class 'freqtable'
plot(x, y.axis = "count", ...)
```

**Arguments**

x	A object of class freqtable.
y.axis	Character string, indicating what variable to use in the y axis, "count" or "percent", when plotting the frequency table.
...	Further arguments to be passed to <a href="#">barplot</a> .

**Author(s)**

Jakson A. Aquino <jalvesaq@gmail.com>

---

toUTF8	<i>Conversion to UTF-8 encoding</i>
--------	-------------------------------------

---

**Description**

Converts the encoding of some attributes of an object to UTF-8

**Usage**

```
toUTF8(x, from = "WINDOWS-1252")
```

**Arguments**

x	A R object, usually a variable of a data frame or a data frame.
from	A string indicating the original encoding. Common values are "LATIN1" and "WINDOWS-1252". Type <code>iconvlist()</code> for the complete list of available encodings.

**Details**

The function converts the attribute `label` of `x` from the specified encoding into UTF-8. If `x` is a factor, the levels are converted as well. If `x` is a `data.frame`, the function makes the conversions in all of its variables.

**Value**

The object with its label and levels converted.

**Author(s)**

Jakson A. Aquino <jalvesaq@gmail.com>.

**See Also**

[iconv](#)

---

xtable.CrossTable	<i>CrossTable method for xtable</i>
-------------------	-------------------------------------

---

## Description

The method creates an object of class xtable.

## Usage

```
## S3 method for class 'CrossTable'
xtable(x, caption = NULL, label = NULL,
       align = NULL, digits = NULL, display = NULL,
       auto = FALSE, multirow = FALSE, hline = FALSE, ...)
```

## Arguments

x	A object of class CrossTable.
caption	See <a href="#">xtable</a> .
label	See <a href="#">xtable</a> .
align	See <a href="#">xtable</a> .
display	See <a href="#">xtable</a> .
digits	See <a href="#">xtable</a> .
auto	See <a href="#">xtable</a> .
multirow	A logical value indicating whether the command \multirow should be added to the table. See the Details section below.
hline	A logical value indicating whether the command \hline should be added to the table. See the Details section below.
...	Further arguments to be passed to <a href="#">format</a> or to replace arguments previously passed to <a href="#">CrossTable</a> .

## Details

If either multirow or hline is TRUE, the sanitize.text.function argument of [print.xtable](#) must be defined. You will also have to add \usepackage{multirow} to your Rnoweb document. See the Example section of [crosstab](#).

## Author(s)

Jakson A. Aquino <jalvesaq@gmail.com>

## See Also

[CrossTable](#), [crosstab](#), [print.xtable](#).

# Index

barplot, [13](#), [20](#)  
boxplot, [3](#)  
bxp, [3](#)  
  
chisq.test, [8](#)  
compmeans, [2](#)  
crosstab, [3](#), [9](#), [20](#), [22](#)  
CrossTable, [3–5](#), [6](#), [12](#), [19](#), [20](#), [22](#)  
  
data.frame2txt, [9](#)  
descr, [10](#)  
  
file.head, [11](#)  
fisher.test, [8](#)  
format, [12](#), [22](#)  
forODFTable, [9](#), [12](#)  
freq, [5](#), [13](#)  
fromUTF8, [14](#)  
fwf2csv, [15](#)  
  
hist, [17](#)  
histkdnc, [16](#)  
  
iconv, [14](#), [21](#)  
  
labels2R, [17](#)  
LogRegR2, [19](#)  
  
mcnemar.test, [9](#)  
mosaicplot, [5](#), [20](#)  
  
par, [20](#)  
plot.CrossTable, [5](#), [9](#), [19](#)  
plot.default, [5](#), [17](#), [20](#)  
plot.freqtable, [13](#), [20](#)  
print.xtable, [22](#)  
prop.table, [9](#)  
  
read.fwf, [15](#)  
round, [12](#)  
  
summary, [11](#)  
  
table, [9](#)  
title, [5](#), [8](#), [20](#)  
toUTF8, [21](#)  
  
write.table, [10](#)  
  
xtable, [22](#)  
xtable.CrossTable, [5](#), [22](#)  
xtabs, [9](#)