

# Package ‘jmv’

October 12, 2023

**Type** Package

**Title** The 'jamovi' Analyses

**Version** 2.4.11

**Date** 2023-10-12

**Maintainer** Jonathon Love <jon@thon.cc>

**Description** A suite of common statistical methods such as descriptives, t-tests, ANOVAs, regression, correlation matrices, proportion tests, contingency tables, and factor analysis. This package is also useable from the 'jamovi' statistical spreadsheet (see <<https://www.jamovi.org>> for more information).

**BugReports** <https://github.com/jamovi/jmv/issues>

**License** GPL (>= 2)

**Depends** R (>= 3.2)

**Imports** jmvcore (>= 2.4.2), R6, car (>= 3.0.0), multcomp, ggplot2 (>= 2.2.1), PMCMR, emmeans (>= 1.4.2), vcd, vcdExtra, GGally, BayesFactor, psych (>= 1.7.5), GPArotation, afex (>= 0.28-0), mvnormtest, lavaan, ggridges, ROCR, nnet, MASS, ggrepel, dplyr, magrittr

**Suggests** exact2x2, testthat (>= 3.1.5), semPlot, carData, knitr, rmarkdown

**Encoding** UTF-8

**RoxygenNote** 6.1.1

**NeedsCompilation** no

**Author** Ravi Selker [aut, cph],  
Jonathon Love [aut, cre, cph],  
Damian Dropmann [aut, cph],  
Victor Moreno [ctb, cph],  
Maurizio Agosti [ctb, cph]

**Repository** CRAN

**Date/Publication** 2023-10-12 13:20:02 UTC

**R topics documented:**

ancova . . . . .	2
ANOVA . . . . .	5
anovaNP . . . . .	7
anovaOneW . . . . .	8
anovaRM . . . . .	10
anovaRMNP . . . . .	13
Big5 . . . . .	15
bugs . . . . .	15
cfa . . . . .	15
contTables . . . . .	18
contTablesPaired . . . . .	21
corrMatrix . . . . .	23
corrPart . . . . .	24
descriptives . . . . .	26
efa . . . . .	29
iris . . . . .	31
linReg . . . . .	31
logLinear . . . . .	34
logRegBin . . . . .	36
logRegMulti . . . . .	39
logRegOrd . . . . .	42
mancova . . . . .	44
pca . . . . .	46
propTest2 . . . . .	47
propTestN . . . . .	49
reliability . . . . .	50
ToothGrowth . . . . .	52
ttestIS . . . . .	52
ttestOneS . . . . .	54
ttestPS . . . . .	56
<b>Index</b>	<b>58</b>

---

ancova

*ANCOVA*


---

**Description**

The Analysis of Covariance (ANCOVA) is used to explore the relationship between a continuous dependent variable, one or more categorical explanatory variables, and one or more continuous explanatory variables (or covariates). It is essentially the same analysis as ANOVA, but with the addition of covariates.

**Usage**

```
ancova(data, dep, factors = NULL, covs = NULL, effectSize = NULL,
       modelTest = FALSE, modelTerms = NULL, ss = "3", homo = FALSE,
       norm = FALSE, qq = FALSE, contrasts = NULL, postHoc = NULL,
       postHocCorr = list("tukey"), postHocES = list(),
       postHocEsCi = FALSE, postHocEsCiWidth = 95, emMeans = list(list()),
       emmPlots = TRUE, emmPlotData = FALSE, emmPlotError = "ci",
       emmTables = FALSE, emmWeights = TRUE, ciWidthEmm = 95, formula)
```

**Arguments**

<code>data</code>	the data as a data frame
<code>dep</code>	the dependent variable from data, variable must be numeric (not necessary when providing a formula, see examples)
<code>factors</code>	the explanatory factors in data (not necessary when providing a formula, see examples)
<code>covs</code>	the explanatory covariates (not necessary when providing a formula, see examples)
<code>effectSize</code>	one or more of 'eta', 'partEta', or 'omega'; use eta <sup>2</sup> , partial eta <sup>2</sup> , and omega <sup>2</sup> effect sizes, respectively
<code>modelTest</code>	TRUE or FALSE (default); perform an overall model test
<code>modelTerms</code>	a formula describing the terms to go into the model (not necessary when providing a formula, see examples)
<code>ss</code>	'1', '2' or '3' (default), the sum of squares to use
<code>homo</code>	TRUE or FALSE (default), perform homogeneity tests
<code>norm</code>	TRUE or FALSE (default), perform Shapiro-Wilk tests of normality
<code>qq</code>	TRUE or FALSE (default), provide a Q-Q plot of residuals
<code>contrasts</code>	a list of lists specifying the factor and type of contrast to use, one of 'deviation', 'simple', 'difference', 'helmert', 'repeated' or 'polynomial'
<code>postHoc</code>	a formula containing the terms to perform post-hoc tests on (see the examples)
<code>postHocCorr</code>	one or more of 'none', 'tukey', 'scheffe', 'bonf', or 'holm'; provide no, Tukey, Scheffé, Bonferroni, and Holm Post Hoc corrections respectively
<code>postHocES</code>	a possible value of 'd'; provide cohen's d measure of effect size for the post-hoc tests
<code>postHocEsCi</code>	TRUE or FALSE (default), provide confidence intervals for the post-hoc effect sizes
<code>postHocEsCiWidth</code>	a number between 50 and 99.9 (default: 95), the width of confidence intervals for the post-hoc effect sizes
<code>emMeans</code>	a formula containing the terms to estimate marginal means for (see the examples)
<code>emmPlots</code>	TRUE (default) or FALSE, provide estimated marginal means plots
<code>emmPlotData</code>	TRUE or FALSE (default), plot the data on top of the marginal means

<code>emmPlotError</code>	'none', 'ci' (default), or 'se'. Use no error bars, use confidence intervals, or use standard errors on the marginal mean plots, respectively
<code>emmTables</code>	TRUE or FALSE (default), provide estimated marginal means tables
<code>emmWeights</code>	TRUE (default) or FALSE, weigh each cell equally or weigh them according to the cell frequency
<code>ciWidthEmm</code>	a number between 50 and 99.9 (default: 95) specifying the confidence interval width for the estimated marginal means
<code>formula</code>	(optional) the formula to use, see the examples

## Value

A results object containing:

<code>results\$main</code>	a table of ANCOVA results
<code>results\$model</code>	The underlying aov object
<code>results\$assump\$homo</code>	a table of homogeneity tests
<code>results\$assump\$norm</code>	a table of normality tests
<code>results\$assump\$qq</code>	a q-q plot
<code>results\$contrasts</code>	an array of contrasts tables
<code>results\$postHoc</code>	an array of post-hoc tables
<code>results\$emm</code>	an array of the estimated marginal means plots + tables
<code>results\$residsOV</code>	an output

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$main$asDF
as.data.frame(results$main)
```

## Examples

```
data('ToothGrowth')

ancova(formula = len ~ supp + dose, data = ToothGrowth)

#
# ANCOVA
#
# ANCOVA
# -----
#              Sum of Squares    df    Mean Square    F      p
# -----
#   supp              205         1         205.4       11.4    0.001
#   dose              2224         1         2224.3      124.0    < .001
#   Residuals         1023        57          17.9
# -----
#
ancova(
```

```

formula = len ~ supp + dose,
data = ToothGrowth,
postHoc = ~ supp,
emMeans = ~ supp)

```

ANOVA

ANOVA

### Description

The Analysis of Variance (ANOVA) is used to explore the relationship between a continuous dependent variable, and one or more categorical explanatory variables.

### Usage

```

ANOVA(data, dep, factors = NULL, effectSize = NULL,
modelTest = FALSE, modelTerms = NULL, ss = "3", homo = FALSE,
norm = FALSE, qq = FALSE, contrasts = NULL, postHoc = NULL,
postHocCorr = list("tukey"), postHocES = list(),
postHocEsCi = FALSE, postHocEsCiWidth = 95, emMeans = list(list()),
emmPlots = TRUE, emmPlotData = FALSE, emmPlotError = "ci",
emmTables = FALSE, emmWeights = TRUE, ciWidthEmm = 95, formula)

```

### Arguments

data	the data as a data frame
dep	the dependent variable from data, variable must be numeric (not necessary when providing a formula, see examples)
factors	the explanatory factors in data (not necessary when providing a formula, see examples)
effectSize	one or more of 'eta', 'partEta', or 'omega'; use eta <sup>2</sup> , partial eta <sup>2</sup> , and omega <sup>2</sup> effect sizes, respectively
modelTest	TRUE or FALSE (default); perform an overall model test
modelTerms	a formula describing the terms to go into the model (not necessary when providing a formula, see examples)
ss	'1', '2' or '3' (default), the sum of squares to use
homo	TRUE or FALSE (default), perform homogeneity tests
norm	TRUE or FALSE (default), perform Shapiro-Wilk tests of normality
qq	TRUE or FALSE (default), provide a Q-Q plot of residuals
contrasts	a list of lists specifying the factor and type of contrast to use, one of 'deviation', 'simple', 'difference', 'helmert', 'repeated' or 'polynomial'
postHoc	a formula containing the terms to perform post-hoc tests on (see the examples)

<code>postHocCorr</code>	one or more of 'none', 'tukey', 'scheffe', 'bonf', or 'holm'; provide no, Tukey, Scheffe, Bonferroni, and Holm Post Hoc corrections respectively
<code>postHocES</code>	a possible value of 'd'; provide cohen's d measure of effect size for the post-hoc tests
<code>postHocEsCi</code>	TRUE or FALSE (default), provide confidence intervals for the post-hoc effect sizes
<code>postHocEsCiWidth</code>	a number between 50 and 99.9 (default: 95), the width of confidence intervals for the post-hoc effect sizes
<code>emMeans</code>	a formula containing the terms to estimate marginal means for (see the examples)
<code>emmPlots</code>	TRUE (default) or FALSE, provide estimated marginal means plots
<code>emmPlotData</code>	TRUE or FALSE (default), plot the data on top of the marginal means
<code>emmPlotError</code>	'none', 'ci' (default), or 'se'. Use no error bars, use confidence intervals, or use standard errors on the marginal mean plots, respectively
<code>emmTables</code>	TRUE or FALSE (default), provide estimated marginal means tables
<code>emmWeights</code>	TRUE (default) or FALSE, weigh each cell equally or weigh them according to the cell frequency
<code>ciWidthEmm</code>	a number between 50 and 99.9 (default: 95) specifying the confidence interval width for the estimated marginal means
<code>formula</code>	(optional) the formula to use, see the examples

### Details

ANOVA assumes that the residuals are normally distributed, and that the variances of all groups are equal. If one is unwilling to assume that the variances are equal, then a Welch's test can be used instead (However, the Welch's test does not support more than one explanatory factor). Alternatively, if one is unwilling to assume that the data is normally distributed, a non-parametric approach (such as Kruskal-Wallis) can be used.

### Value

A results object containing:

<code>results\$main</code>	a table of ANOVA results
<code>results\$model</code>	The underlying aov object
<code>results\$assump\$homo</code>	a table of homogeneity tests
<code>results\$assump\$norm</code>	a table of normality tests
<code>results\$assump\$qq</code>	a q-q plot
<code>results\$contrasts</code>	an array of contrasts tables
<code>results\$postHoc</code>	an array of post-hoc tables
<code>results\$emm</code>	an array of the estimated marginal means plots + tables
<code>results\$residsOV</code>	an output

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$main$asDF
as.data.frame(results$main)
```

## Examples

```
data('ToothGrowth')

ANOVA(formula = len ~ dose * supp, data = ToothGrowth)

#
# ANOVA
#
# ANOVA
# -----
#               Sum of Squares    df    Mean Square    F      p
# -----
# dose                2426         2         1213.2    92.00 < .001
# supp                 205         1          205.4    15.57 < .001
# dose:supp            108         2           54.2     4.11  0.022
# Residuals           712        54           13.2
# -----
#

ANOVA(
  formula = len ~ dose * supp,
  data = ToothGrowth,
  emMeans = ~ supp + dose:supp, # est. marginal means for supp and dose:supp
  emmPlots = TRUE,              # produce plots of those marginal means
  emmTables = TRUE)            # produce tables of those marginal means
```

---

anovaNP

*One-Way ANOVA (Non-parametric)*

---

## Description

The Kruskal-Wallis test is used to explore the relationship between a continuous dependent variable, and a categorical explanatory variable. It is analogous to ANOVA, but with the advantage of being non-parametric and having fewer assumptions. However, it has the limitation that it can only test a single explanatory variable at a time.

## Usage

```
anovaNP(data, deps, group, es = FALSE, pairs = FALSE, formula)
```

**Arguments**

data	the data as a data frame
deps	a string naming the dependent variable in data
group	a string naming the grouping or independent variable in data
es	TRUE or FALSE (default), provide effect-sizes
pairs	TRUE or FALSE (default), perform pairwise comparisons
formula	(optional) the formula to use, see the examples

**Value**

A results object containing:

results\$table	a table of the test results
results\$comparisons	an array of pairwise comparison tables

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$table$asDF
as.data.frame(results$table)
```

**Examples**

```
data('ToothGrowth')

anovaNP(formula = len ~ dose, data=ToothGrowth)

#
# ONE-WAY ANOVA (NON-PARAMETRIC)
#
# Kruskal-Wallis
# -----
#           X2    df    p
# -----
# len    40.7    2    < .001
# -----
#
```

---

anovaOneW

*One-Way ANOVA*

---

**Description**

The Analysis of Variance (ANOVA) is used to explore the relationship between a continuous dependent variable, and one or more categorical explanatory variables. This 'One-Way ANOVA' is a



simplified version of the 'normal' ANOVA, allowing only a single explanatory factor, however also providing a Welch's ANOVA. The Welch's ANOVA has the advantage that it need not assume that the variances of all groups are equal.

### Usage

```
anovaOneW(data, deps, group, welchs = TRUE, fishers = FALSE,
  miss = "perAnalysis", desc = FALSE, descPlot = FALSE,
  norm = FALSE, qq = FALSE, eqv = FALSE, phMethod = "none",
  phMeanDif = TRUE, phSig = TRUE, phTest = FALSE, phFlag = FALSE,
  formula)
```

### Arguments

data	the data as a data frame
deps	a string naming the dependent variables in data
group	a string naming the grouping or independent variable in data
welchs	TRUE (default) or FALSE, perform Welch's one-way ANOVA which does not assume equal variances
fishers	TRUE or FALSE (default), perform Fisher's one-way ANOVA which assumes equal variances
miss	'perAnalysis' or 'listwise', how to handle missing values; 'perAnalysis' excludes missing values for individual dependent variables, 'listwise' excludes a row from all analyses if one of its entries is missing.
desc	TRUE or FALSE (default), provide descriptive statistics
descPlot	TRUE or FALSE (default), provide descriptive plots
norm	TRUE or FALSE (default), perform Shapiro-Wilk test of normality
qq	TRUE or FALSE (default), provide a Q-Q plot of residuals
eqv	TRUE or FALSE (default), perform Levene's test for homogeneity of variances
phMethod	'none', 'gamesHowell' or 'tukey', which post-hoc tests to provide; 'none' shows no post-hoc tests, 'gamesHowell' shows Games-Howell post-hoc tests where no equivalence of variances is assumed, and 'tukey' shows Tukey post-hoc tests where equivalence of variances is assumed
phMeanDif	TRUE (default) or FALSE, provide mean differences for post-hoc tests
phSig	TRUE (default) or FALSE, provide significance levels for post-hoc tests
phTest	TRUE or FALSE (default), provide test results (t-value and degrees of freedom) for post-hoc tests
phFlag	TRUE or FALSE (default), flag significant post-hoc comparisons
formula	(optional) the formula to use, see the examples

### Details

For convenience, this method allows specifying multiple dependent variables, resulting in multiple independent tests.

Note that the Welch's ANOVA is the same procedure as the Welch's independent samples t-test.

**Value**

A results object containing:

<code>results\$anova</code>	a table of the test results
<code>results\$desc</code>	a table containing the group descriptives
<code>results\$assump\$norm</code>	a table containing the normality tests
<code>results\$assump\$eqv</code>	a table of homogeneity of variances tests
<code>results\$plots</code>	an array of groups of plots
<code>results\$postHoc</code>	an array of post-hoc tables

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$anova$asDF
as.data.frame(results$anova)
```

**Examples**

```
data('ToothGrowth')
dat <- ToothGrowth
dat$dose <- factor(dat$dose)

anovaOneW(formula = len ~ dose, data = dat)

#
# ONE-WAY ANOVA
#
# One-Way ANOVA (Welch's)
# -----
#           F         df1    df2     p
# -----
#   len    68.4         2    37.7  < .001
# -----
#
```

---

anovaRM

*Repeated Measures ANOVA*

---

**Description**

The Repeated Measures ANOVA is used to explore the relationship between a continuous dependent variable and one or more categorical explanatory variables, where one or more of the explanatory variables are 'within subjects' (where multiple measurements are from the same subject). Additionally, this analysis allows the inclusion of covariates, allowing for repeated measures ANCOVAs as well.

**Usage**

```
anovaRM(data, rm = list(list(label = "RM Factor 1", levels =
  list("Level 1", "Level 2"))), rmCells = NULL, bs = NULL,
  cov = NULL, effectSize = NULL, depLabel = "Dependent",
  rmTerms = NULL, bsTerms = NULL, ss = "3", spherTests = FALSE,
  spherCorr = list("none"), leveneTest = FALSE, qq = FALSE,
  contrasts = NULL, postHoc = NULL, postHocCorr = list("tukey"),
  emMeans = list(list()), emmPlots = TRUE, emmTables = FALSE,
  emmWeights = TRUE, ciWidthEmm = 95, emmPlotData = FALSE,
  emmPlotError = "ci", groupSumm = FALSE)
```

**Arguments**

<code>data</code>	the data as a data frame
<code>rm</code>	a list of lists, where each list describes the label (as a string) and the levels (as vector of strings) of a particular repeated measures factor
<code>rmCells</code>	a list of lists, where each list describes a repeated measure (as a string) from data defined as measure and the particular combination of levels from <code>rm</code> that it belongs to (as a vector of strings) defined as <code>cell</code>
<code>bs</code>	a vector of strings naming the between subjects factors from data
<code>cov</code>	a vector of strings naming the covariates from data. Variables must be numeric
<code>effectSize</code>	one or more of 'eta', 'partEta', or 'omega'; use eta <sup>2</sup> , partial eta <sup>2</sup> , and omega <sup>2</sup> effect sizes, respectively
<code>depLabel</code>	a string (default: 'Dependent') describing the label used for the dependent variable throughout the analysis
<code>rmTerms</code>	a list of character vectors describing the repeated measures terms to go into the model
<code>bsTerms</code>	a list of character vectors describing the between subjects terms to go into the model
<code>ss</code>	'2' or '3' (default), the sum of squares to use
<code>spherTests</code>	TRUE or FALSE (default), perform sphericity tests
<code>spherCorr</code>	one or more of 'none' (default), 'GG', or HF; use no p-value correction, the Greenhouse-Geisser p-value correction, and the Huynh-Feldt p-value correction for sphericity, respectively
<code>leveneTest</code>	TRUE or FALSE (default), test for homogeneity of variances (i.e., Levene's test)
<code>qq</code>	TRUE or FALSE (default), provide a Q-Q plot of residuals
<code>contrasts</code>	in development
<code>postHoc</code>	a list of character vectors describing the post-hoc tests that need to be computed
<code>postHocCorr</code>	one or more of 'none', 'tukey' (default), 'scheffe', 'bonf', or 'holm'; use no, Tukey, Scheffe, Bonferroni and Holm posthoc corrections, respectively
<code>emMeans</code>	a list of lists specifying the variables for which the estimated marginal means need to be calculate. Supports up to three variables per term.
<code>emmPlots</code>	TRUE (default) or FALSE, provide estimated marginal means plots

emmTables	TRUE or FALSE (default), provide estimated marginal means tables
emmWeights	TRUE (default) or FALSE, weigh each cell equally or weigh them according to the cell frequency
ciWidthEmm	a number between 50 and 99.9 (default: 95) specifying the confidence interval width for the estimated marginal means
emmPlotData	TRUE or FALSE (default), plot the data on top of the marginal means
emmPlotError	'none', 'ci' (default), or 'se'. Use no error bars, use confidence intervals, or use standard errors on the marginal mean plots, respectively
groupSumm	TRUE or FALSE (default), report a summary of the different groups

### Details

This analysis requires that the data be in 'wide format', where each row represents a subject (as opposed to long format, where each measurement of the dependent variable is represented as a row).

A non-parametric equivalent of the repeated measures ANOVA also exists; the Friedman test. However, it has the limitation of only being able to test a single factor.

### Value

A results object containing:

results\$rmTable	a table
results\$bsTable	a table
results\$assump\$spherTable	a table
results\$assump\$leveneTable	a table
results\$assump\$qq	a q-q plot
results\$contrasts	an array of tables
results\$postHoc	an array of tables
results\$emm	an array of the estimated marginal means plots + tables
results\$groupSummary	a summary of the groups

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$rmTable$asDF
as.data.frame(results$rmTable)
```

### Examples

```
data('bugs', package = 'jmv')

anovaRM(
  data = bugs,
  rm = list(
    list(
```

```

        label = 'Frightening',
        levels = c('Low', 'High'))),
rmCells = list(
  list(
    measure = 'LDLF',
    cell = 'Low'),
  list(
    measure = 'LDHF',
    cell = 'High')),
rmTerms = list(
  'Frightening'))

#
# REPEATED MEASURES ANOVA
#
# Within Subjects Effects
# -----
#           Sum of Squares   df   Mean Square   F     p
# -----
# Frightening             126     1       126.11    44.2  < .001
# Residual                 257    90         2.85
# -----
# Note. Type 3 Sums of Squares
#
#
# Between Subjects Effects
# -----
#           Sum of Squares   df   Mean Square   F     p
# -----
# Residual                 954    90        10.6
# -----
# Note. Type 3 Sums of Squares
#
#

```

anovaRMNP

*Repeated Measures ANOVA (Non-parametric)***Description**

The Friedman test is used to explore the relationship between a continuous dependent variable and a categorical explanatory variable, where the explanatory variable is 'within subjects' (where multiple measurements are from the same subject). It is analogous to Repeated Measures ANOVA, but with the advantage of being non-parametric, and not requiring the assumptions of normality or homogeneity of variances. However, it has the limitation that it can only test a single explanatory variable at a time.

**Usage**

```
anovaRMNP(data, measures, pairs = FALSE, desc = FALSE, plots = FALSE,
           plotType = "means")
```

**Arguments**

data	the data as a data frame
measures	a vector of strings naming the repeated measures variables
pairs	TRUE or FALSE (default), perform pairwise comparisons
desc	TRUE or FALSE (default), provide descriptive statistics
plots	TRUE or FALSE (default), provide a descriptive plot
plotType	'means' (default) or 'medians', the error bars to use in the plot

**Value**

A results object containing:

results\$table	a table of the Friedman test results
results\$comp	a table of the pairwise comparisons
results\$desc	a table containing the descriptives
results\$plot	a descriptives plot

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$table$asDF
as.data.frame(results$table)
```

**Examples**

```
data('bugs', package = 'jmv')

anovaRMNP(bugs, measures = vars(LDLF, LDHF, HDLF, HDHF))

#
# REPEATED MEASURES ANOVA (NON-PARAMETRIC)
#
# Friedman
# -----
#   X2    df    p
# -----
#   55.8    3    < .001
# -----
#
```

---

 Big5

*Big 5*


---

**Description**

Big 5

---

 bugs

*bugs*


---

**Description**

bugs

**Author(s)**

Ryan, Wilde &amp; Crist (2013)

**References**

<https://faculty.kutztown.edu/rryan/RESEARCH/PUBS/Ryan,%20Wilde,%20%26%20Crist%202013%20Web%20exp%20vs%20lab.pdf>

---

 cfa

*Confirmatory Factor Analysis*


---

**Description**

Confirmatory Factor Analysis

**Usage**

```
cfa(data, factors = list(list(label = "Factor 1", vars = list())),
    resCov, miss = "fiml", constrain = "facVar", estTest = TRUE,
    ci = FALSE, ciWidth = 95, stdEst = FALSE, factCovEst = TRUE,
    factInterceptEst = FALSE, resCovEst = FALSE,
    resInterceptEst = FALSE, fitMeasures = list("cfi", "tli", "rmsea"),
    modelTest = TRUE, pathDiagram = FALSE, corRes = FALSE,
    hlCorRes = 0.1, mi = FALSE, h1MI = 3)
```

**Arguments**

<code>data</code>	the data as a data frame
<code>factors</code>	a list containing named lists that define the label of the factor and the vars that belong to that factor
<code>resCov</code>	a list of lists specifying the residual covariances that need to be estimated
<code>miss</code>	'listwise' or 'fiml', how to handle missing values; 'listwise' excludes a row from all analyses if one of its entries is missing, 'fiml' uses a full information maximum likelihood method to estimate the model.
<code>constrain</code>	'facVar' or 'facInd', how to constrain the model; 'facVar' fixes the factor variances to one, 'facInd' fixes each factor to the scale of its first indicator.
<code>estTest</code>	TRUE (default) or FALSE, provide 'Z' and 'p' values for the model estimates
<code>ci</code>	TRUE or FALSE (default), provide a confidence interval for the model estimates
<code>ciWidth</code>	a number between 50 and 99.9 (default: 95) specifying the confidence interval width that is used as 'ci'
<code>stdEst</code>	TRUE or FALSE (default), provide a standardized estimate for the model estimates
<code>factCovEst</code>	TRUE (default) or FALSE, provide estimates for the factor (co)variances
<code>factInterceptEst</code>	TRUE or FALSE (default), provide estimates for the factor intercepts
<code>resCovEst</code>	TRUE (default) or FALSE, provide estimates for the residual (co)variances
<code>resInterceptEst</code>	TRUE or FALSE (default), provide estimates for the residual intercepts
<code>fitMeasures</code>	one or more of 'cfi', 'tli', 'srmr', 'rmsea', 'aic', or 'bic'; use CFI, TLI, SRMR, RMSEA + 90% confidence interval, adjusted AIC, and BIC model fit measures, respectively
<code>modelTest</code>	TRUE (default) or FALSE, provide a chi-square test for exact fit that compares the model with the perfect fitting model
<code>pathDiagram</code>	TRUE or FALSE (default), provide a path diagram of the model
<code>corRes</code>	TRUE or FALSE (default), provide the residuals for the observed correlation matrix (i.e., the difference between the expected correlation matrix and the observed correlation matrix)
<code>hlCorRes</code>	a number (default: 0.1), highlight values in the 'corRes' table above this value
<code>mi</code>	TRUE or FALSE (default), provide modification indices for the parameters not included in the model
<code>hIMI</code>	a number (default: 3), highlight values in the 'modIndices' tables above this value

**Value**

A results object containing:

<code>results\$factorLoadings</code>	a table containing the factor loadings
<code>results\$factorEst\$factorCov</code>	a table containing factor covariances estimates
<code>results\$factorEst\$factorIntercept</code>	a table containing factor intercept estimates



<code>results\$resEst\$resCov</code>	a table containing residual covariances estimates
<code>results\$resEst\$resIntercept</code>	a table containing residual intercept estimates
<code>results\$modelFit\$test</code>	a table containing the chi-square test for exact fit
<code>results\$modelFit\$fitMeasures</code>	a table containing fit measures
<code>results\$modelPerformance\$corRes</code>	a table containing residuals for the observed correlation matrix
<code>results\$modelPerformance\$modIndices</code>	a group
<code>results\$pathDiagram</code>	an image containing the model path diagram
<code>results\$modelSyntax</code>	the lavaan syntax used to fit the model

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$factorLoadings$asDF
as.data.frame(results$factorLoadings)
```

## Examples

```
data <- lavaan::HolzingerSwineford1939

jmv::cfa(
  data = data,
  factors = list(
    list(label="Visual", vars=c("x1", "x2", "x3")),
    list(label="Textual", vars=c("x4", "x5", "x6")),
    list(label="Speed", vars=c("x7", "x8", "x9")),
    resCov = NULL)

#
# CONFIRMATORY FACTOR ANALYSIS
#
# Factor Loadings
# -----
#   Factor   Indicator   Estimate   SE       Z       p
# -----
#   Visual   x1           0.900     0.0832   10.81   < .001
#           x2           0.498     0.0808    6.16   < .001
#           x3           0.656     0.0776    8.46   < .001
#   Textual  x4           0.990     0.0567   17.46   < .001
#           x5           1.102     0.0626   17.60   < .001
#           x6           0.917     0.0538   17.05   < .001
#   Speed    x7           0.619     0.0743    8.34   < .001
#           x8           0.731     0.0755    9.68   < .001
#           x9           0.670     0.0775    8.64   < .001
# -----
#
#
# FACTOR ESTIMATES
#
# Factor Covariances
# -----
#           Estimate   SE       Z       p
# -----
```

```

# Visual Visual 1.000 a
# Textual Textual 0.459 0.0635 7.22 < .001
# Speed Speed 0.471 0.0862 5.46 < .001
# Textual Textual 1.000 a
# Speed Speed 0.283 0.0715 3.96 < .001
# Speed Speed 1.000 a
# -----
# a fixed parameter
#
#
# MODEL FIT
#
# Test for Exact Fit
# -----
# X2 df p
# -----
# 85.3 24 < .001
# -----
#
#
# Fit Measures
# -----
# CFI TLI RMSEA Lower Upper
# -----
# 0.931 0.896 0.0921 0.0714 0.114
# -----
#
#

```

---

contTables

*Contingency Tables*


---

## Description

The  $X^2$  test of association (not to be confused with the  $X^2$  goodness of fit) is used to test whether two categorical variables are independent or associated. If the p-value is low, it suggests the variables are not independent, and that there is a relationship between the two variables.

## Usage

```

contTables(data, rows, cols, counts = NULL, layers = NULL,
  chiSq = TRUE, chiSqCorr = FALSE, zProp = FALSE, likeRat = FALSE,
  fisher = FALSE, contCoef = FALSE, phiCra = FALSE,
  diffProp = FALSE, logOdds = FALSE, odds = FALSE, relRisk = FALSE,
  ci = TRUE, ciWidth = 95, compare = "rows",
  hypothesis = "different", gamma = FALSE, taub = FALSE,
  mh = FALSE, obs = TRUE, exp = FALSE, pcRow = FALSE,
  pcCol = FALSE, pcTot = FALSE, barplot = FALSE, yaxis = "ycounts",
  yaxisPc = "total_pc", xaxis = "xrows", bartype = "dodge", formula)

```

**Arguments**

data	the data as a data frame
rows	the variable to use as the rows in the contingency table (not necessary when providing a formula, see the examples)
cols	the variable to use as the columns in the contingency table (not necessary when providing a formula, see the examples)
counts	the variable to use as the counts in the contingency table (not necessary when providing a formula, see the examples)
layers	the variables to use to split the contingency table (not necessary when providing a formula, see the examples)
chiSq	TRUE (default) or FALSE, provide $X^2$
chiSqCorr	TRUE or FALSE (default), provide $X^2$ with continuity correction
zProp	TRUE or FALSE (default), provide a z test for differences between two proportions
likeRat	TRUE or FALSE (default), provide the likelihood ratio
fisher	TRUE or FALSE (default), provide Fisher's exact test
contCoef	TRUE or FALSE (default), provide the contingency coefficient
phiCra	TRUE or FALSE (default), provide Phi and Cramer's V
diffProp	TRUE or FALSE (default), provide the differences in proportions (only available for 2x2 tables)
logOdds	TRUE or FALSE (default), provide the log odds ratio (only available for 2x2 tables)
odds	TRUE or FALSE (default), provide the odds ratio (only available for 2x2 tables)
relRisk	TRUE or FALSE (default), provide the relative risk (only available for 2x2 tables)
ci	TRUE or FALSE (default), provide confidence intervals for the comparative measures
ciWidth	a number between 50 and 99.9 (default: 95), width of the confidence intervals to provide
compare	columns or rows (default), compare columns/rows in difference of proportions or relative risks (2x2 tables)
hypothesis	'different' (default), 'oneGreater' or 'twoGreater', the alternative hypothesis; group 1 different to group 2, group 1 greater than group 2, and group 2 greater than group 1 respectively
gamma	TRUE or FALSE (default), provide gamma
taub	TRUE or FALSE (default), provide Kendall's tau-b
mh	TRUE or FALSE (default), provide Mantel-Haenszel test for trend
obs	TRUE or FALSE (default), provide the observed counts
exp	TRUE or FALSE (default), provide the expected counts
pcRow	TRUE or FALSE (default), provide row percentages
pcCol	TRUE or FALSE (default), provide column percentages
pcTot	TRUE or FALSE (default), provide total percentages

barplot	TRUE or FALSE (default), show barplots
yaxis	ycounts (default) or ypc. Use respectively counts or percentages for the bar plot y-axis
yaxisPc	total_pc (default), column_pc, or row_pc. Use respectively percentages of total, within columns, or within rows for the bar plot y-axis.
xaxis	rows (default), or columns in bar plot X axis
bartype	stack or side by side (default), barplot type
formula	(optional) the formula to use, see the examples

### Value

A results object containing:

results\$freqs	a table of proportions
results\$chiSq	a table of $X^2$ test results
results\$odds	a table of comparative measures
results\$nom	a table of the 'nominal' test results
results\$gamma	a table of the gamma test results
results\$taub	a table of the Kendall's tau-b test results
results\$mh	a table of the Mantel-Haenszel test for trend
results\$barplot	an image

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$freqs$asDF
as.data.frame(results$freqs)
```

### Examples

```
data('HairEyeColor')
dat <- as.data.frame(HairEyeColor)

contTables(formula = Freq ~ Hair:Eye, dat)

#
# CONTINGENCY TABLES
#
# Contingency Tables
# -----
#   Hair      Brown      Blue      Hazel      Green      Total
# -----
#   Black      68       20       15        5       108
#   Brown     119       84       54       29       286
#   Red        26       17       14       14        71
#   Blond        7       94       10       16       127
#   Total     220      215       93       64       592
# -----
#
```

```

#
# X2 Tests
# -----
#           Value   df   p
# -----
#   X2      138     9   < .001
#   N         592
# -----
#

# Alternatively, omit the left of the formula (`Freq`) if each row
# represents a single observation:

contTables(formula = ~ Hair:Eye, dat)

```

---

contTablesPaired      *Paired Samples Contingency Tables*

---

### Description

McNemar test

### Usage

```

contTablesPaired(data, rows, cols, counts = NULL, chiSq = TRUE,
  chiSqCorr = FALSE, exact = FALSE, pcRow = FALSE, pcCol = FALSE,
  formula)

```

### Arguments

data	the data as a data frame
rows	the variable to use as the rows in the contingency table (not necessary when providing a formula, see the examples)
cols	the variable to use as the columns in the contingency table (not necessary when providing a formula, see the examples)
counts	the variable to use as the counts in the contingency table (not necessary when providing a formula, see the examples)
chiSq	TRUE (default) or FALSE, provide X <sup>2</sup>
chiSqCorr	TRUE or FALSE (default), provide X <sup>2</sup> with continuity correction
exact	TRUE or FALSE (default), provide an exact log odds ratio (requires exact2x2 to be installed)
pcRow	TRUE or FALSE (default), provide row percentages
pcCol	TRUE or FALSE (default), provide column percentages
formula	(optional) the formula to use, see the examples

**Value**

A results object containing:

results\$freqs	a proportions table
results\$test	a table of test results

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$freqs$asDF
as.data.frame(results$freqs)
```

**Examples**

```
dat <- data.frame(
  `1st survey` = c('Approve', 'Approve', 'Disapprove', 'Disapprove'),
  `2nd survey` = c('Approve', 'Disapprove', 'Approve', 'Disapprove'),
  `Counts` = c(794, 150, 86, 570),
  check.names=FALSE)

contTablesPaired(formula = Counts ~ `1st survey`:`2nd survey`, data = dat)

#
# PAIRED SAMPLES CONTINGENCY TABLES
#
# Contingency Tables
# -----
#   1st survey   Approve   Disapprove   Total
# -----
#   Approve      794        150        944
#   Disapprove    86         570        656
#   Total        880        720       1600
# -----
#
#
# McNemar Test
# -----
#                                     Value   df   p
# -----
#   X2                                17.4    1   < .001
#   X2 continuity correction          16.8    1   < .001
# -----
#

# Alternatively, omit the left of the formula (`Counts`) from the
# formula if each row represents a single observation:

contTablesPaired(formula = ~ `1st survey`:`2nd survey`, data = dat)
```

---

 corrMatrix                      *Correlation Matrix*


---

**Description**

Correlation matrices are a way to examine linear relationships between two or more continuous variables.

**Usage**

```
corrMatrix(data, vars, pearson = TRUE, spearman = FALSE,
           kendall = FALSE, sig = TRUE, flag = FALSE, n = FALSE,
           ci = FALSE, ciWidth = 95, plots = FALSE, plotDens = FALSE,
           plotStats = FALSE, hypothesis = "corr")
```

**Arguments**

data	the data as a data frame
vars	a vector of strings naming the variables to correlate in data
pearson	TRUE (default) or FALSE, provide Pearson's R
spearman	TRUE or FALSE (default), provide Spearman's rho
kendall	TRUE or FALSE (default), provide Kendall's tau-b
sig	TRUE (default) or FALSE, provide significance levels
flag	TRUE or FALSE (default), flag significant correlations
n	TRUE or FALSE (default), provide the number of cases
ci	TRUE or FALSE (default), provide confidence intervals
ciWidth	a number between 50 and 99.9 (default: 95), the width of confidence intervals to provide
plots	TRUE or FALSE (default), provide a correlation matrix plot
plotDens	TRUE or FALSE (default), provide densities in the correlation matrix plot
plotStats	TRUE or FALSE (default), provide statistics in the correlation matrix plot
hypothesis	one of 'corr' (default), 'pos', 'neg' specifying the alternative hypothesis; correlated, correlated positively, correlated negatively respectively.

**Details**

For each pair of variables, a Pearson's  $r$  value indicates the strength and direction of the relationship between those two variables. A positive value indicates a positive relationship (higher values of one variable predict higher values of the other variable). A negative Pearson's  $r$  indicates a negative relationship (higher values of one variable predict lower values of the other variable, and vice-versa). A value of zero indicates no relationship (whether a variable is high or low, does not tell us anything about the value of the other variable).

More formally, it is possible to test the null hypothesis that the correlation is zero and calculate a  $p$ -value. If the  $p$ -value is low, it suggests the correlation co-efficient is not zero, and there is a linear (or more complex) relationship between the two variables.

**Value**

A results object containing:

results\$matrix	a correlation matrix table
results\$plot	a correlation matrix plot

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$matrix$asDF
as.data.frame(results$matrix)
```

**Examples**

```
data('mtcars')

corrMatrix(mtcars, vars = vars(mpg, cyl, disp, hp))

#
# CORRELATION MATRIX
#
# Correlation Matrix
# -----
#                mpg      cyl      disp      hp
# -----
# mpg  Pearson's r      -0.852  -0.848  -0.776
#      p-value          < .001  < .001  < .001
#
# cyl  Pearson's r              0.902   0.832
#      p-value                < .001  < .001
#
# disp Pearson's r              0.791
#      p-value                < .001
#
# hp   Pearson's r
#      p-value
# -----
#
```

---

corrPart

*Partial Correlation*

---

**Description**

Partial correlation matrices are a way to examine linear relationships between two or more continuous variables while controlling for other variables



**Usage**

```
corrPart(data, vars, controls, pearson = TRUE, spearman = FALSE,
         kendall = FALSE, type = "part", sig = TRUE, flag = FALSE,
         n = FALSE, hypothesis = "corr")
```

**Arguments**

<code>data</code>	the data as a data frame
<code>vars</code>	a vector of strings naming the variables to correlate in data
<code>controls</code>	a vector of strings naming the control variables in data
<code>pearson</code>	TRUE (default) or FALSE, provide Pearson's R
<code>spearman</code>	TRUE or FALSE (default), provide Spearman's rho
<code>kendall</code>	TRUE or FALSE (default), provide Kendall's tau-b
<code>type</code>	one of 'part' (default) or 'semi' specifying the type of partial correlation to calculate; partial or semipartial correlation.
<code>sig</code>	TRUE (default) or FALSE, provide significance levels
<code>flag</code>	TRUE or FALSE (default), flag significant correlations
<code>n</code>	TRUE or FALSE (default), provide the number of cases
<code>hypothesis</code>	one of 'corr' (default), 'pos', 'neg' specifying the alternative hypothesis; correlated, correlated positively, correlated negatively respectively.

**Details**

For each pair of variables, a Pearson's  $r$  value indicates the strength and direction of the relationship between those two variables. A positive value indicates a positive relationship (higher values of one variable predict higher values of the other variable). A negative Pearson's  $r$  indicates a negative relationship (higher values of one variable predict lower values of the other variable, and vice-versa). A value of zero indicates no relationship (whether a variable is high or low, does not tell us anything about the value of the other variable).

More formally, it is possible to test the null hypothesis that the correlation is zero and calculate a  $p$ -value. If the  $p$ -value is low, it suggests the correlation co-efficient is not zero, and there is a linear (or more complex) relationship between the two variables.

**Value**

A results object containing:

<code>results\$matrix</code>	a (semi)partial correlation matrix table
------------------------------	--

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$matrix$asDF
as.data.frame(results$matrix)
```

**Examples**

```

data('mtcars')

corrPart(mtcars, vars = vars(mpg, cyl, disp), controls = vars(hp))

#
# PARTIAL CORRELATION
#
# Partial Correlation
# -----
#
#                mpg      cyl      disp
# -----
#   mpg   Pearson's r
#         p-value
#
#   cyl   Pearson's r   -0.590
#         p-value       < .001
#
#   disp   Pearson's r   -0.606   0.719
#         p-value       < .001   < .001
# -----
# Note. controlling for 'hp'
#

```

descriptives

*Descriptives***Description**

Descriptives are an assortment of summarising statistics, and visualizations which allow exploring the shape and distribution of data. It is good practice to explore your data with descriptives before proceeding to more formal tests.

**Usage**

```

descriptives(data, vars, splitBy = NULL, freq = FALSE,
  desc = "columns", hist = FALSE, dens = FALSE, bar = FALSE,
  barCounts = FALSE, box = FALSE, violin = FALSE, dot = FALSE,
  dotType = "jitter", boxMean = FALSE, boxLabelOutliers = TRUE,
  qq = FALSE, n = TRUE, missing = TRUE, mean = TRUE,
  median = TRUE, mode = FALSE, sum = FALSE, sd = TRUE,
  variance = FALSE, range = FALSE, min = TRUE, max = TRUE,
  se = FALSE, ci = FALSE, ciWidth = 95, iqr = FALSE,
  skew = FALSE, kurt = FALSE, sw = FALSE, pcEqGr = FALSE,
  pcNEqGr = 4, pc = FALSE, pcValues = "25,50,75", extreme = FALSE,
  extremeN = 5, formula)

```

**Arguments**

<code>data</code>	the data as a data frame
<code>vars</code>	a vector of strings naming the variables of interest in data
<code>splitBy</code>	a vector of strings naming the variables used to split vars
<code>freq</code>	TRUE or FALSE (default), provide frequency tables (nominal, ordinal variables only)
<code>desc</code>	'rows' or 'columns' (default), display the variables across the rows or across the columns (default)
<code>hist</code>	TRUE or FALSE (default), provide histograms (continuous variables only)
<code>dens</code>	TRUE or FALSE (default), provide density plots (continuous variables only)
<code>bar</code>	TRUE or FALSE (default), provide bar plots (nominal, ordinal variables only)
<code>barCounts</code>	TRUE or FALSE (default), add counts to the bar plots
<code>box</code>	TRUE or FALSE (default), provide box plots (continuous variables only)
<code>violin</code>	TRUE or FALSE (default), provide violin plots (continuous variables only)
<code>dot</code>	TRUE or FALSE (default), provide dot plots (continuous variables only)
<code>dotType</code>	.
<code>boxMean</code>	TRUE or FALSE (default), add mean to box plot
<code>boxLabelOutliers</code>	TRUE (default) or FALSE, add labels with the row number to the outliers in the box plot
<code>qq</code>	TRUE or FALSE (default), provide Q-Q plots (continuous variables only)
<code>n</code>	TRUE (default) or FALSE, provide the sample size
<code>missing</code>	TRUE (default) or FALSE, provide the number of missing values
<code>mean</code>	TRUE (default) or FALSE, provide the mean
<code>median</code>	TRUE (default) or FALSE, provide the median
<code>mode</code>	TRUE or FALSE (default), provide the mode
<code>sum</code>	TRUE or FALSE (default), provide the sum
<code>sd</code>	TRUE (default) or FALSE, provide the standard deviation
<code>variance</code>	TRUE or FALSE (default), provide the variance
<code>range</code>	TRUE or FALSE (default), provide the range
<code>min</code>	TRUE or FALSE (default), provide the minimum
<code>max</code>	TRUE or FALSE (default), provide the maximum
<code>se</code>	TRUE or FALSE (default), provide the standard error
<code>ci</code>	TRUE or FALSE (default), provide confidence intervals for the mean
<code>ciWidth</code>	a number between 50 and 99.9 (default: 95), the width of confidence intervals
<code>iqr</code>	TRUE or FALSE (default), provide the interquartile range
<code>skew</code>	TRUE or FALSE (default), provide the skewness
<code>kurt</code>	TRUE or FALSE (default), provide the kurtosis

sw	TRUE or FALSE (default), provide Shapiro-Wilk p-value
pcEqGr	TRUE or FALSE (default), provide quantiles
pcNEqGr	an integer (default: 4) specifying the number of equal groups
pc	TRUE or FALSE (default), provide percentiles
pcValues	a comma-separated list (default: 25,50,75) specifying the percentiles
extreme	TRUE or FALSE (default), provide N most extreme (highest and lowest) values
extremeN	an integer (default: 5) specifying the number of extreme values
formula	(optional) the formula to use, see the examples

### Value

A results object containing:

results\$descriptives	a table of the descriptive statistics
results\$descriptivesT	a table of the descriptive statistics
results\$freqencies	an array of frequency tables
results\$extremeValues	an array of extreme values tables
results\$plots	an array of descriptive plots

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$descriptives$asDF
as.data.frame(results$descriptives)
```

### Examples

```
data('mtcars')
dat <- mtcars

# frequency tables can be provided for factors
dat$gear <- as.factor(dat$gear)

descriptives(dat, vars = vars(mpg, cyl, disp, gear), freq = TRUE)

#
# DESCRIPTIVES
#
# Descriptives
# -----
#          mpg    cyl    disp    gear
# -----
# N          32     32     32     32
# Missing     0      0      0      0
# Mean       20.1   6.19   231   3.69
# Median     19.2   6.00   196   4.00
# Minimum    10.4   4.00   71.1   3
# Maximum    33.9   8.00   472   5
```

```

# -----
#
#
# FREQUENCIES
#
# Frequencies of gear
# -----
#   Levels   Counts
# -----
#     3         15
#     4         12
#     5          5
# -----
#

# splitting by a variable
descriptives(formula = disp + mpg ~ cyl, dat,
              median=F, min=F, max=F, n=F, missing=F)

# providing histograms
descriptives(formula = mpg ~ cyl, dat, hist=T,
              median=F, min=F, max=F, n=F, missing=F)

# splitting by multiple variables
descriptives(formula = mpg ~ cyl:gear, dat,
              median=F, min=F, max=F, missing=F)

```

---

efa

*Exploratory Factor Analysis*


---

## Description

Exploratory Factor Analysis

## Usage

```

efa(data, vars, nFactorMethod = "parallel", nFactors = 1,
     minEigen = 0, extraction = "minres", rotation = "oblimin",
     hideLoadings = 0.3, sortLoadings = FALSE, screePlot = FALSE,
     eigen = FALSE, factorCor = FALSE, factorSummary = FALSE,
     modelFit = FALSE, kmo = FALSE, bartlett = FALSE,
     factorScoreMethod = "Thurstone")

```

## Arguments

data	the data as a data frame
vars	a vector of strings naming the variables of interest in data

**nFactorMethod** 'parallel' (default), 'eigen' or 'fixed', the way to determine the number of factors  
**nFactors** an integer (default: 1), the number of factors in the model  
**minEigen** a number (default: 0), the minimal eigenvalue for a factor to be included in the model  
**extraction** 'minres' (default), 'ml', or 'pa' use respectively 'minimum residual', 'maximum likelihood', or 'principal axis' as the factor extraction method  
**rotation** 'none', 'varimax', 'quartimax', 'promax', 'oblimin' (default), or 'simplimax', the rotation to use in estimation  
**hideLoadings** a number (default: 0.3), hide factor loadings below this value  
**sortLoadings** TRUE or FALSE (default), sort the factor loadings by size  
**screePlot** TRUE or FALSE (default), show scree plot  
**eigen** TRUE or FALSE (default), show eigenvalue table  
**factorCor** TRUE or FALSE (default), show inter-factor correlations  
**factorSummary** TRUE or FALSE (default), show factor summary  
**modelFit** TRUE or FALSE (default), show model fit measures and test  
**kmo** TRUE or FALSE (default), show Kaiser-Meyer-Olkin (KMO) measure of sampling adequacy (MSA) results  
**bartlett** TRUE or FALSE (default), show Bartlett's test of sphericity results  
**factorScoreMethod** 'Thurstone' (default), 'Bartlett', 'tenBerge', 'Anderson', or 'Harman' use respectively 'Thurstone', 'Bartlett', 'ten Berge', 'Anderson & Rubin', or 'Harman' method for estimating factor scores

## Value

A results object containing:

results\$text                      a preformatted

## Examples

```

data('iris')

efa(iris, vars = vars(Sepal.Length, Sepal.Width, Petal.Length, Petal.Width))

#
# EXPLORATORY FACTOR ANALYSIS
#
# Factor Loadings
# -----
#           1      2      Uniqueness
# -----
# Sepal.Length  0.993      0.10181
# Sepal.Width   0.725      0.42199

```

```
# Petal.Length    0.933                0.00483
# Petal.Width    0.897                0.07088
# -----
# Note. 'oblimin' rotation was used
#
```

---

iris	<i>iris</i>
------	-------------

---

### Description

iris

---

linReg	<i>Linear Regression</i>
--------	--------------------------

---

### Description

Linear regression is used to explore the relationship between a continuous dependent variable, and one or more continuous and/or categorical explanatory variables. Other statistical methods, such as ANOVA and ANCOVA, are in reality just forms of linear regression.

### Usage

```
linReg(data, dep, covs = NULL, factors = NULL, weights = NULL,
        blocks = list(list()), refLevels = NULL, intercept = "refLevel",
        r = TRUE, r2 = TRUE, r2Adj = FALSE, aic = FALSE, bic = FALSE,
        rmse = FALSE, modelTest = FALSE, anova = FALSE, ci = FALSE,
        ciWidth = 95, stdEst = FALSE, ciStdEst = FALSE,
        ciWidthStdEst = 95, norm = FALSE, qqPlot = FALSE,
        resPlots = FALSE, durbin = FALSE, collin = FALSE, cooks = FALSE,
        emMeans = list(list()), ciEmm = TRUE, ciWidthEmm = 95,
        emmPlots = TRUE, emmTables = FALSE, emmWeights = TRUE)
```

### Arguments

data	the data as a data frame
dep	the dependent variable from data, variable must be numeric
covs	the covariates from data
factors	the fixed factors from data
weights	the (optional) weights from data to be used in the fitting process
blocks	a list containing vectors of strings that name the predictors that are added to the model. The elements are added to the model according to their order in the list

<code>refLevels</code>	a list of lists specifying reference levels of the dependent variable and all the factors
<code>intercept</code>	'refLevel' (default) or 'grandMean', coding of the intercept. Either creates contrast so that the intercept represents the reference level or the grand mean
<code>r</code>	TRUE (default) or FALSE, provide the statistical measure R for the models
<code>r2</code>	TRUE (default) or FALSE, provide the statistical measure R-squared for the models
<code>r2Adj</code>	TRUE or FALSE (default), provide the statistical measure adjusted R-squared for the models
<code>aic</code>	TRUE or FALSE (default), provide Akaike's Information Criterion (AIC) for the models
<code>bic</code>	TRUE or FALSE (default), provide Bayesian Information Criterion (BIC) for the models
<code>rmse</code>	TRUE or FALSE (default), provide RMSE for the models
<code>modelTest</code>	TRUE (default) or FALSE, provide the model comparison between the models and the NULL model
<code>anova</code>	TRUE or FALSE (default), provide the omnibus ANOVA test for the predictors
<code>ci</code>	TRUE or FALSE (default), provide a confidence interval for the model coefficients
<code>ciWidth</code>	a number between 50 and 99.9 (default: 95) specifying the confidence interval width
<code>stdEst</code>	TRUE or FALSE (default), provide a standardized estimate for the model coefficients
<code>ciStdEst</code>	TRUE or FALSE (default), provide a confidence interval for the model coefficient standardized estimates
<code>ciWidthStdEst</code>	a number between 50 and 99.9 (default: 95) specifying the confidence interval width
<code>norm</code>	TRUE or FALSE (default), perform a Shapiro-Wilk test on the residuals
<code>qqPlot</code>	TRUE or FALSE (default), provide a Q-Q plot of residuals
<code>resPlots</code>	TRUE or FALSE (default), provide residual plots where the dependent variable and each covariate is plotted against the standardized residuals.
<code>durbin</code>	TRUE or FALSE (default), provide results of the Durbin- Watson test for autocorrelation
<code>collin</code>	TRUE or FALSE (default), provide VIF and tolerance collinearity statistics
<code>cooks</code>	TRUE or FALSE (default), provide summary statistics for the Cook's distance
<code>emMeans</code>	a formula containing the terms to estimate marginal means for, supports up to three variables per term
<code>ciEmm</code>	TRUE (default) or FALSE, provide a confidence interval for the estimated marginal means
<code>ciWidthEmm</code>	a number between 50 and 99.9 (default: 95) specifying the confidence interval width for the estimated marginal means
<code>emmPlots</code>	TRUE (default) or FALSE, provide estimated marginal means plots
<code>emmTables</code>	TRUE or FALSE (default), provide estimated marginal means tables
<code>emmWeights</code>	TRUE (default) or FALSE, weigh each cell equally or weigh them according to the cell frequency



**Value**

A results object containing:

results\$modelFit	a table
results\$modelComp	a table
results\$models	an array of model specific results
results\$predictOV	an output
results\$residsOV	an output
results\$cooksOV	an output

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$modelFit$asDF
as.data.frame(results$modelFit)
```

**Examples**

```
data('Prestige', package='carData')

linReg(data = Prestige, dep = income,
       covs = vars(education, prestige, women),
       blocks = list(list('education', 'prestige', 'women')))
```

```
#
# LINEAR REGRESSION
#
# Model Fit Measures
# -----
#   Model    R      R²
# -----
#       1    0.802  0.643
# -----
#
# MODEL SPECIFIC RESULTS
#
# MODEL 1
#
# Model Coefficients
# -----
# Predictor   Estimate   SE      t      p
# -----
# Intercept   -253.8    1086.16 -0.234  0.816
# women       -50.9     8.56    -5.948 < .001
# prestige    141.4    29.91    4.729 < .001
# education   177.2    187.63  0.944  0.347
# -----
#
```

logLinear

*Log-Linear Regression***Description**

Log-Linear Regression

**Usage**

```
logLinear(data, factors = NULL, counts = NULL, blocks = list(list()),
  refLevels = NULL, modelTest = FALSE, dev = TRUE, aic = TRUE,
  bic = FALSE, pseudoR2 = list("r2mf"), omni = FALSE, ci = FALSE,
  ciWidth = 95, RR = FALSE, ciRR = FALSE, ciWidthRR = 95,
  emMeans = list(list()), ciEmm = TRUE, ciWidthEmm = 95,
  emmPlots = TRUE, emmTables = FALSE, emmWeights = TRUE)
```

**Arguments**

data	the data as a data frame
factors	a vector of strings naming the factors from data
counts	a string naming a variable in data containing counts, or NULL if each row represents a single observation
blocks	a list containing vectors of strings that name the predictors that are added to the model. The elements are added to the model according to their order in the list
refLevels	a list of lists specifying reference levels of the dependent variable and all the factors
modelTest	TRUE or FALSE (default), provide the model comparison between the models and the NULL model
dev	TRUE (default) or FALSE, provide the deviance (or $-2\text{LogLikelihood}$ ) for the models
aic	TRUE (default) or FALSE, provide Akaike's Information Criterion (AIC) for the models
bic	TRUE or FALSE (default), provide Bayesian Information Criterion (BIC) for the models
pseudoR2	one or more of 'r2mf', 'r2cs', or 'r2n'; use McFadden's, Cox & Snell, and Nagelkerke pseudo- $R^2$ , respectively
omni	TRUE or FALSE (default), provide the omnibus likelihood ratio tests for the predictors
ci	TRUE or FALSE (default), provide a confidence interval for the model coefficient estimates
ciWidth	a number between 50 and 99.9 (default: 95) specifying the confidence interval width

RR	TRUE or FALSE (default), provide the exponential of the log-rate ratio estimate, or the rate ratio estimate
ciRR	TRUE or FALSE (default), provide a confidence interval for the model coefficient rate ratio estimates
ciWidthRR	a number between 50 and 99.9 (default: 95) specifying the confidence interval width
emMeans	a list of lists specifying the variables for which the estimated marginal means need to be calculate. Supports up to three variables per term.
ciEmm	TRUE (default) or FALSE, provide a confidence interval for the estimated marginal means
ciWidthEmm	a number between 50 and 99.9 (default: 95) specifying the confidence interval width for the estimated marginal means
emmPlots	TRUE (default) or FALSE, provide estimated marginal means plots
emmTables	TRUE or FALSE (default), provide estimated marginal means tables
emmWeights	TRUE (default) or FALSE, weigh each cell equally or weigh them according to the cell frequency

### Value

A results object containing:

results\$modelFit	a table
results\$modelComp	a table
results\$models	an array of model specific results

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$modelFit$asDF
as.data.frame(results$modelFit)
```

### Examples

```
data('mtcars')

tab <- table('gear'=mtcars$gear, 'cyl'=mtcars$cyl)
dat <- as.data.frame(tab)

logLinear(data = dat, factors = vars(gear, cyl), counts = Freq,
          blocks = list(list("gear", "cyl", c("gear", "cyl"))),
          refLevels = list(
            list(var="gear", ref="3"),
            list(var="cyl", ref="4")))

#
# LOG-LINEAR REGRESSION
#
# Model Fit Measures
```

```

# -----
#   Model   Deviance   AIC   R2-McF
# -----
#         1   4.12e-10   41.4   1.000
# -----
#
#
# MODEL SPECIFIC RESULTS
#
# MODEL 1
#
# Model Coefficients
# -----
#   Predictor           Estimate      SE        Z        p
# -----
#   Intercept           -4.71e-16    1.00    -4.71e-16    1.000
#   gear:
#   4 3                   2.079      1.06     1.961    0.050
#   5 3                   0.693      1.22     0.566    0.571
#   cyl:
#   6 4                   0.693      1.22     0.566    0.571
#   8 4                   2.485      1.04     2.387    0.017
#   gear:cyl:
#   (4 3):(6 4)          -1.386      1.37    -1.012    0.311
#   (5 3):(6 4)          -1.386      1.73    -0.800    0.423
#   (4 3):(8 4)         -26.867    42247.17  -6.36e -4  0.999
#   (5 3):(8 4)          -2.485      1.44    -1.722    0.085
# -----
#
#
#

```

---

logRegBin

*Binomial Logistic Regression*


---

## Description

Binomial Logistic Regression

## Usage

```

logRegBin(data, dep, covs = NULL, factors = NULL,
  blocks = list(list()), refLevels = NULL, modelTest = FALSE,
  dev = TRUE, aic = TRUE, bic = FALSE, pseudoR2 = list("r2mf"),
  omni = FALSE, ci = FALSE, ciWidth = 95, OR = FALSE,
  ciOR = FALSE, ciWidthOR = 95, emMeans = list(list()),
  ciEmm = TRUE, ciWidthEmm = 95, emmPlots = TRUE,
  emmTables = FALSE, emmWeights = TRUE, class = FALSE, acc = FALSE,
  spec = FALSE, sens = FALSE, auc = FALSE, rocPlot = FALSE,
  cutOff = 0.5, cutOffPlot = FALSE, collin = FALSE,
  boxTidwell = FALSE, cooks = FALSE)

```

**Arguments**

data	the data as a data frame
dep	a string naming the dependent variable from data, variable must be a factor
covs	a vector of strings naming the covariates from data
factors	a vector of strings naming the fixed factors from data
blocks	a list containing vectors of strings that name the predictors that are added to the model. The elements are added to the model according to their order in the list
refLevels	a list of lists specifying reference levels of the dependent variable and all the factors
modelTest	TRUE or FALSE (default), provide the model comparison between the models and the NULL model
dev	TRUE (default) or FALSE, provide the deviance (or $-2\text{LogLikelihood}$ ) for the models
aic	TRUE (default) or FALSE, provide Aikaike's Information Criterion (AIC) for the models
bic	TRUE or FALSE (default), provide Bayesian Information Criterion (BIC) for the models
pseudoR2	one or more of 'r2mf', 'r2cs', or 'r2n'; use McFadden's, Cox & Snell, and Nagelkerke pseudo- $R^2$ , respectively
omni	TRUE or FALSE (default), provide the omnibus likelihood ratio tests for the predictors
ci	TRUE or FALSE (default), provide a confidence interval for the model coefficient estimates
ciWidth	a number between 50 and 99.9 (default: 95) specifying the confidence interval width
OR	TRUE or FALSE (default), provide the exponential of the log-odds ratio estimate, or the odds ratio estimate
ciOR	TRUE or FALSE (default), provide a confidence interval for the model coefficient odds ratio estimates
ciWidthOR	a number between 50 and 99.9 (default: 95) specifying the confidence interval width
emMeans	a list of lists specifying the variables for which the estimated marginal means need to be calculate. Supports up to three variables per term.
ciEmm	TRUE (default) or FALSE, provide a confidence interval for the estimated marginal means
ciWidthEmm	a number between 50 and 99.9 (default: 95) specifying the confidence interval width for the estimated marginal means
emmPlots	TRUE (default) or FALSE, provide estimated marginal means plots
emmTables	TRUE or FALSE (default), provide estimated marginal means tables
emmWeights	TRUE (default) or FALSE, weigh each cell equally or weigh them according to the cell frequency

class	TRUE or FALSE (default), provide a predicted classification table (or confusion matrix)
acc	TRUE or FALSE (default), provide the predicted accuracy of outcomes grouped by the cut-off value
spec	TRUE or FALSE (default), provide the predicted specificity of outcomes grouped by the cut-off value
sens	TRUE or FALSE (default), provide the predicted sensitivity of outcomes grouped by the cut-off value
auc	TRUE or FALSE (default), provide the rea under the ROC curve (AUC)
rocPlot	TRUE or FALSE (default), provide a ROC curve plot
cutOff	TRUE or FALSE (default), set a cut-off used for the predictions
cutOffPlot	TRUE or FALSE (default), provide a cut-off plot
collin	TRUE or FALSE (default), provide VIF and tolerance collinearity statistics
boxTidwell	TRUE or FALSE (default), provide Box-Tidwell test for linearity of the logit
cooks	TRUE or FALSE (default), provide summary statistics for the Cook's distance

### Value

A results object containing:

results\$modelFit	a table
results\$modelComp	a table
results\$models	an array of model specific results
results\$predictOV	an output
results\$residsOV	an output
results\$cooksOV	an output

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$modelFit$asDF
as.data.frame(results$modelFit)
```

### Examples

```
data('birthwt', package='MASS')

dat <- data.frame(
  low = factor(birthwt$low),
  age = birthwt$age,
  bwt = birthwt$bwt)

logRegBin(data = dat, dep = low,
  covs = vars(age, bwt),
  blocks = list(list("age", "bwt")),
  refLevels = list(list(var="low", ref="0")))
```

```

#
# BINOMIAL LOGISTIC REGRESSION
#
# Model Fit Measures
# -----
#   Model   Deviance   AIC   R2-McF
# -----
#       1     4.97e-7    6.00    1.000
# -----
#
#
# MODEL SPECIFIC RESULTS
#
# MODEL 1
#
# Model Coefficients
# -----
#   Predictor   Estimate     SE       Z       p
# -----
#   Intercept   2974.73225   218237.2   0.0136   0.989
#   age         -0.00653     482.7     -1.35e-5  1.000
#   bwt         -1.18532     87.0      -0.0136   0.989
# -----
#   Note. Estimates represent the log odds of "low = 1"
#   vs. "low = 0"
#
#
#

```

---

logRegMulti

*Multinomial Logistic Regression*


---

## Description

Multinomial Logistic Regression

## Usage

```

logRegMulti(data, dep, covs = NULL, factors = NULL,
  blocks = list(list()), refLevels = NULL, modelTest = FALSE,
  dev = TRUE, aic = TRUE, bic = FALSE, pseudoR2 = list("r2mf"),
  omni = FALSE, ci = FALSE, ciWidth = 95, OR = FALSE,
  ciOR = FALSE, ciWidthOR = 95, emMeans = list(list()),
  ciEmm = TRUE, ciWidthEmm = 95, emmPlots = TRUE,
  emmTables = FALSE, emmWeights = TRUE)

```

## Arguments

data            the data as a data frame

dep	a string naming the dependent variable from data, variable must be a factor
covs	a vector of strings naming the covariates from data
factors	a vector of strings naming the fixed factors from data
blocks	a list containing vectors of strings that name the predictors that are added to the model. The elements are added to the model according to their order in the list
refLevels	a list of lists specifying reference levels of the dependent variable and all the factors
modelTest	TRUE or FALSE (default), provide the model comparison between the models and the NULL model
dev	TRUE (default) or FALSE, provide the deviance (or $-2\text{LogLikelihood}$ ) for the models
aic	TRUE (default) or FALSE, provide Akaike's Information Criterion (AIC) for the models
bic	TRUE or FALSE (default), provide Bayesian Information Criterion (BIC) for the models
pseudoR2	one or more of 'r2mf', 'r2cs', or 'r2n'; use McFadden's, Cox & Snell, and Nagelkerke pseudo-R <sup>2</sup> , respectively
omni	TRUE or FALSE (default), provide the omnibus likelihood ratio tests for the predictors
ci	TRUE or FALSE (default), provide a confidence interval for the model coefficient estimates
ciWidth	a number between 50 and 99.9 (default: 95) specifying the confidence interval width
OR	TRUE or FALSE (default), provide the exponential of the log-odds ratio estimate, or the odds ratio estimate
ciOR	TRUE or FALSE (default), provide a confidence interval for the model coefficient odds ratio estimates
ciWidthOR	a number between 50 and 99.9 (default: 95) specifying the confidence interval width
emMeans	a list of lists specifying the variables for which the estimated marginal means need to be calculate. Supports up to three variables per term.
ciEmm	TRUE (default) or FALSE, provide a confidence interval for the estimated marginal means
ciWidthEmm	a number between 50 and 99.9 (default: 95) specifying the confidence interval width for the estimated marginal means
emmPlots	TRUE (default) or FALSE, provide estimated marginal means plots
emmTables	TRUE or FALSE (default), provide estimated marginal means tables
emmWeights	TRUE (default) or FALSE, weigh each cell equally or weigh them according to the cell frequency

**Value**

A results object containing:



```

results$modelFit      a table
results$modelComp    a table
results$models        an array of model specific results

```

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```

results$modelFit$asDF
as.data.frame(results$modelFit)

```

## Examples

```

data('birthwt', package='MASS')

dat <- data.frame(
  race = factor(birthwt$race),
  age = birthwt$age,
  low = factor(birthwt$low))

logRegMulti(data = dat, dep = race,
  covs = age, factors = low,
  blocks = list(list("age", "low")),
  refLevels = list(
    list(var="race", ref="1"),
    list(var="low", ref="0")))

#
# MULTINOMIAL LOGISTIC REGRESSION
#
# Model Fit Measures
# -----
#   Model   Deviance   AIC   R2-McF
# -----
#       1         360    372    0.0333
# -----
#
#
# MODEL SPECIFIC RESULTS
#
# MODEL 1
#
# Model Coefficients
# -----
#   race   Predictor   Estimate   SE       Z       p
# -----
# 2 - 1   Intercept    0.8155    1.1186    0.729    0.466
#         age          -0.1038   0.0487   -2.131    0.033
#         low:
#         1 0          0.7527    0.4700    1.601    0.109
# 3 - 1   Intercept    1.0123    0.7798    1.298    0.194
#         age          -0.0663   0.0324   -2.047    0.041
#         low:

```

```
#           1  0           0.5677   0.3522   1.612   0.107
# -----
#
#
```

---

logRegOrd

*Ordinal Logistic Regression*


---

### Description

Ordinal Logistic Regression

### Usage

```
logRegOrd(data, dep, covs = NULL, factors = NULL,
  blocks = list(list()), refLevels = NULL, modelTest = FALSE,
  dev = TRUE, aic = TRUE, bic = FALSE, pseudoR2 = list("r2mf"),
  omni = FALSE, thres = FALSE, ci = FALSE, ciWidth = 95,
  OR = FALSE, ciOR = FALSE, ciWidthOR = 95)
```

### Arguments

data	the data as a data frame
dep	a string naming the dependent variable from data, variable must be a factor
covs	a vector of strings naming the covariates from data
factors	a vector of strings naming the fixed factors from data
blocks	a list containing vectors of strings that name the predictors that are added to the model. The elements are added to the model according to their order in the list
refLevels	a list of lists specifying reference levels of the dependent variable and all the factors
modelTest	TRUE or FALSE (default), provide the model comparison between the models and the NULL model
dev	TRUE (default) or FALSE, provide the deviance (or $-2\text{LogLikelihood}$ ) for the models
aic	TRUE (default) or FALSE, provide Aikaike's Information Criterion (AIC) for the models
bic	TRUE or FALSE (default), provide Bayesian Information Criterion (BIC) for the models
pseudoR2	one or more of 'r2mf', 'r2cs', or 'r2n'; use McFadden's, Cox & Snell, and Nagelkerke pseudo- $R^2$ , respectively
omni	TRUE or FALSE (default), provide the omnibus likelihood ratio tests for the predictors

thres	TRUE or FALSE (default), provide the thresholds that are used as cut-off scores for the levels of the dependent variable
ci	TRUE or FALSE (default), provide a confidence interval for the model coefficient estimates
ciWidth	a number between 50 and 99.9 (default: 95) specifying the confidence interval width
OR	TRUE or FALSE (default), provide the exponential of the log-odds ratio estimate, or the odds ratio estimate
ciOR	TRUE or FALSE (default), provide a confidence interval for the model coefficient odds ratio estimates
ciWidthOR	a number between 50 and 99.9 (default: 95) specifying the confidence interval width

### Value

A results object containing:

results\$modelFit	a table
results\$modelComp	a table
results\$models	an array of model specific results

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$modelFit$asDF
as.data.frame(results$modelFit)
```

### Examples

```
set.seed(1337)

y <- factor(sample(1:3, 100, replace = TRUE))
x1 <- rnorm(100)
x2 <- rnorm(100)

df <- data.frame(y=y, x1=x1, x2=x2)

logRegOrd(data = df, dep = y,
          covs = vars(x1, x2),
          blocks = list(list("x1", "x2")))

#
# ORDINAL LOGISTIC REGRESSION
#
# Model Fit Measures
# -----
#   Model   Deviance   AIC   R2-McF
# -----
#       1       218     226   5.68e-4
```

```

# -----
#
#
# MODEL SPECIFIC RESULTS
#
# MODEL 1
#
# Model Coefficients
# -----
# Predictor      Estimate    SE        Z        p
# -----
# x1              0.0579    0.193    0.300    0.764
# x2              0.0330    0.172    0.192    0.848
# -----
#
#
#

```

---

mancova

*MANCOVA*


---

## Description

Multivariate Analysis of (Co)Variance (MANCOVA) is used to explore the relationship between multiple dependent variables, and one or more categorical and/or continuous explanatory variables.

## Usage

```

mancova(data, deps, factors = NULL, covs = NULL,
         multivar = list("pillai", "wilks", "hotel", "roy"), boxM = FALSE,
         shapiro = FALSE, qqPlot = FALSE)

```

## Arguments

data	the data as a data frame
deps	a string naming the dependent variable from data, variable must be numeric
factors	a vector of strings naming the factors from data
covs	a vector of strings naming the covariates from data
multivar	one or more of 'pillai', 'wilks', 'hotel', or 'roy'; use Pillai's Trace, Wilks' Lambda, Hotelling's Trace, and Roy's Largest Root multivariate statistics, respectively
boxM	TRUE or FALSE (default), provide Box's M test
shapiro	TRUE or FALSE (default), provide Shapiro-Wilk test
qqPlot	TRUE or FALSE (default), provide a Q-Q plot of multivariate normality

## Value

A results object containing:

```

results$multivar      a table
results$univar       a table
results$assump$boxM  a table
results$assump$shapiro a table
results$assump$qqPlot an image
    
```

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```

results$multivar$asDF
as.data.frame(results$multivar)
    
```

**Examples**

```

data('iris')

mancova(data = iris,
        deps = vars(Sepal.Length, Sepal.Width, Petal.Length, Petal.Width),
        factors = Species)

#
# MANCOVA
#
# Multivariate Tests
# -----
#               value      F      df1      df2      p
# -----
# Species  Pillai's Trace      1.19    53.5      8      290    < .001
#           Wilks' Lambda      0.0234    199      8      288    < .001
#           Hotelling's Trace    32.5     581      8      286    < .001
#           Roy's Largest Root   32.2    1167      4      145    < .001
# -----
#
# Univariate Tests
# -----
#               Dependent Variable  Sum of Squares  df  Mean Square  F      p
# -----
# Species  Sepal.Length           63.21      2      31.6061    119.3    < .001
#           Sepal.Width            11.34      2       5.6725     49.2    < .001
#           Petal.Length          437.10      2     218.5514   1180.2    < .001
#           Petal.Width            80.41      2      40.2067    960.0    < .001
# Residuals  Sepal.Length           38.96     147       0.2650
#           Sepal.Width            16.96     147       0.1154
#           Petal.Length           27.22     147       0.1852
#           Petal.Width             6.16     147       0.0419
# -----
#
    
```

pca

*Principal Component Analysis***Description**

Principal Component Analysis

**Usage**

```
pca(data, vars, nFactorMethod = "parallel", nFactors = 1,
     minEigen = 1, rotation = "varimax", hideLoadings = 0.3,
     sortLoadings = FALSE, screePlot = FALSE, eigen = FALSE,
     factorCor = FALSE, factorSummary = FALSE, kmo = FALSE,
     bartlett = FALSE)
```

**Arguments**

data	the data as a data frame
vars	a vector of strings naming the variables of interest in data
nFactorMethod	'parallel' (default), 'eigen' or 'fixed', the way to determine the number of factors
nFactors	an integer (default: 1), the number of components in the model
minEigen	a number (default: 1), the minimal eigenvalue for a component to be included in the model
rotation	'none', 'varimax' (default), 'quartimax', 'promax', 'oblimin', or 'simplimax', the rotation to use in estimation
hideLoadings	a number (default: 0.3), hide loadings below this value
sortLoadings	TRUE or FALSE (default), sort the factor loadings by size
screePlot	TRUE or FALSE (default), show scree plot
eigen	TRUE or FALSE (default), show eigenvalue table
factorCor	TRUE or FALSE (default), show inter-factor correlations
factorSummary	TRUE or FALSE (default), show factor summary
kmo	TRUE or FALSE (default), show Kaiser-Meyer-Olkin (KMO) measure of sampling adequacy (MSA) results
bartlett	TRUE or FALSE (default), show Bartlett's test of sphericity results

**Value**

A results object containing:

results\$loadings	a table
results\$factorStats\$factorSummary	a table
results\$factorStats\$factorCor	a table

results\$modelFit\$fit	a table
results\$assump\$bartlett	a table
results\$assump\$kmo	a table
results\$eigen\$initEigen	a table
results\$eigen\$screePlot	an image
results\$factorScoresOV	an output

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$loadings$asDF
as.data.frame(results$loadings)
```

### Examples

```
data('iris')

pca(iris, vars = vars(Sepal.Length, Sepal.Width, Petal.Length, Petal.Width))

#
# PRINCIPAL COMPONENT ANALYSIS
#
# Component Loadings
# -----
#           1           Uniqueness
# -----
# Sepal.Length    0.890    0.2076
# Sepal.Width     -0.460    0.7883
# Petal.Length    0.992    0.0168
# Petal.Width     0.965    0.0688
# -----
# Note. 'varimax' rotation was used
#
```

---

propTest2	<i>Proportion Test (2 Outcomes)</i>
-----------	-------------------------------------

---

### Description

The Binomial test is used to test the Null hypothesis that the proportion of observations match some expected value. If the p-value is low, this suggests that the Null hypothesis is false, and that the true proportion must be some other value.

### Usage

```
propTest2(data, vars, areCounts = FALSE, testValue = 0.5,
  hypothesis = "notequal", ci = FALSE, ciWidth = 95, bf = FALSE,
  priorA = 1, priorB = 1, ciBayes = FALSE, ciBayesWidth = 95,
  postPlots = FALSE)
```

**Arguments**

data	the data as a data frame
vars	a vector of strings naming the variables of interest in data
areCounts	TRUE or FALSE (default), the variables are counts
testValue	a number (default: 0.5), the value for the null hypothesis
hypothesis	'notequal' (default), 'greater' or 'less', the alternative hypothesis
ci	TRUE or FALSE (default), provide confidence intervals
ciWidth	a number between 50 and 99.9 (default: 95), the confidence interval width
bf	TRUE or FALSE (default), provide Bayes factors
priorA	a number (default: 1), the beta prior 'a' parameter
priorB	a number (default: 1), the beta prior 'b' parameter
ciBayes	TRUE or FALSE (default), provide Bayesian credible intervals
ciBayesWidth	a number between 50 and 99.9 (default: 95), the credible interval width
postPlots	TRUE or FALSE (default), provide posterior plots

**Value**

A results object containing:

results\$table	a table of the proportions and test results
results\$postPlots	an array of the posterior plots

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$table$asDF
as.data.frame(results$table)
```

**Examples**

```
dat <- data.frame(x=c(8, 15))

propTest2(dat, vars = x, areCounts = TRUE)

#
# PROPORTION TEST (2 OUTCOMES)
#
# Binomial Test
# -----
#      Level  Count  Total  Proportion  p
# -----
# x    1      8     23      0.348    0.210
#      2     15     23      0.652    0.210
# -----
# Note. Ha is proportion != 0.5
```



#

---

propTestN *Proportion Test (N Outcomes)*


---

**Description**

The  $X^2$  Goodness of fit test (not to be confused with the  $X^2$  test of independence), tests the Null hypothesis that the proportions of observations match some expected proportions. If the p-value is low, this suggests that the Null hypothesis is false, and that the true proportions are different to those tested.

**Usage**

```
propTestN(data, var, counts = NULL, expected = FALSE, ratio = NULL,
           formula)
```

**Arguments**

data	the data as a data frame
var	the variable of interest in data (not necessary when using a formula, see the examples)
counts	the counts in data
expected	TRUE or FALSE (default), whether expected counts should be displayed
ratio	a vector of numbers: the expected proportions
formula	(optional) the formula to use, see the examples

**Value**

A results object containing:

results\$props	a table of the proportions
results\$tests	a table of the test results

Tables can be converted to data frames with `asDF` or [as.data.frame](#). For example:

```
results$props$asDF
as.data.frame(results$props)
```

**Examples**

```
data('HairEyeColor')
dat <- as.data.frame(HairEyeColor)
```

```

propTestN(formula = Freq ~ Eye, data = dat, ratio = c(1,1,1,1))

#
# PROPORTION TEST (N OUTCOMES)
#
# Proportions
# -----
#   Level   Count   Proportion
# -----
#   Brown   220     0.372
#   Blue    215     0.363
#   Hazel   93      0.157
#   Green   64      0.108
# -----
#
#
# X2 Goodness of Fit
# -----
#   X2   df   p
# -----
#   133    3   < .001
# -----
#
#

```

---

reliability

*Reliability Analysis*


---

## Description

Reliability Analysis

## Usage

```

reliability(data, vars, alphaScale = TRUE, omegaScale = FALSE,
  meanScale = FALSE, sdScale = FALSE, corPlot = FALSE,
  alphaItems = FALSE, omegaItems = FALSE, meanItems = FALSE,
  sdItems = FALSE, itemRestCor = FALSE, revItems = NULL)

```

## Arguments

data	the data as a data frame
vars	a vector of strings naming the variables of interest in data
alphaScale	TRUE (default) or FALSE, provide Cronbach's alpha
omegaScale	TRUE or FALSE (default), provide McDonald's omega
meanScale	TRUE or FALSE (default), provide the mean
sdScale	TRUE or FALSE (default), provide the standard deviation

corPlot	TRUE or FALSE (default), provide a correlation plot
alphaItems	TRUE or FALSE (default), provide what the Cronbach's alpha would be if the item was dropped
omegaItems	TRUE or FALSE (default), provide what the McDonald's omega would be if the item was dropped
meanItems	TRUE or FALSE (default), provide item means
sdItems	TRUE or FALSE (default), provide item standard deviations
itemRestCor	TRUE or FALSE (default), provide item-rest correlations
revItems	a vector containing strings naming the variables that are reverse scaled

**Value**

A results object containing:

results\$scale	a table
results\$items	a table
results\$corPlot	an image
results\$meanScoreOV	an output
results\$sumScoreOV	an output

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$scale$asDF
as.data.frame(results$scale)
```

**Examples**

```
data('iris')

reliability(iris, vars = c('Sepal.Length', 'Sepal.Width', 'Petal.Length', 'Petal.Width'),
            omegaScale = TRUE)

#
# RELIABILITY ANALYSIS
#
# Scale Reliability Statistics
# -----
#           Cronbach's alpha   McDonald's omega
# -----
#   scale           0.708         0.848
# -----
#
```

---

 ToothGrowth
 

---

*Tooth Growth***Description**

Tooth Growth

---

 ttestIS
 

---

*Independent Samples T-Test***Description**

The Student's Independent samples t-test (sometimes called a two-samples t-test) is used to test the null hypothesis that two groups have the same mean. A low p-value suggests that the null hypothesis is not true, and therefore the group means are different.

**Usage**

```
ttestIS(data, vars, group, students = TRUE, bf = FALSE,
        bfPrior = 0.707, welchs = FALSE, mann = FALSE,
        hypothesis = "different", norm = FALSE, qq = FALSE, eqv = FALSE,
        meanDiff = FALSE, ci = FALSE, ciWidth = 95, effectSize = FALSE,
        ciES = FALSE, ciWidthES = 95, desc = FALSE, plots = FALSE,
        miss = "perAnalysis", formula)
```

**Arguments**

data	the data as a data frame
vars	the dependent variables (not necessary when using a formula, see the examples)
group	the grouping variable with two levels (not necessary when using a formula, see the examples)
students	TRUE (default) or FALSE, perform Student's t-tests
bf	TRUE or FALSE (default), provide Bayes factors
bfPrior	a number between 0.01 and 2 (default 0.707), the prior width to use in calculating Bayes factors
welchs	TRUE or FALSE (default), perform Welch's t-tests
mann	TRUE or FALSE (default), perform Mann-Whitney U tests
hypothesis	'different' (default), 'oneGreater' or 'twoGreater', the alternative hypothesis; group 1 different to group 2, group 1 greater than group 2, and group 2 greater than group 1 respectively
norm	TRUE or FALSE (default), perform Shapiro-Wilk tests of normality
qq	TRUE or FALSE (default), provide Q-Q plots of residuals

eqv	TRUE or FALSE (default), perform Levene's tests for homogeneity of variances
meanDiff	TRUE or FALSE (default), provide means and standard errors
ci	TRUE or FALSE (default), provide confidence intervals
ciWidth	a number between 50 and 99.9 (default: 95), the width of confidence intervals
effectSize	TRUE or FALSE (default), provide effect sizes
ciES	TRUE or FALSE (default), provide confidence intervals for the effect-sizes
ciWidthES	a number between 50 and 99.9 (default: 95), the width of confidence intervals for the effect sizes
desc	TRUE or FALSE (default), provide descriptive statistics
plots	TRUE or FALSE (default), provide descriptive plots
miss	'perAnalysis' or 'listwise', how to handle missing values; 'perAnalysis' excludes missing values for individual dependent variables, 'listwise' excludes a row from all analyses if one of its entries is missing.
formula	(optional) the formula to use, see the examples

### Details

The Student's independent t-test assumes that the data from each group are from a normal distribution, and that the variances of these groups are equal. If unwilling to assume the groups have equal variances, the Welch's t-test can be used in its place. If one is additionally unwilling to assume the data from each group are from a normal distribution, the non-parametric Mann-Whitney U test can be used instead (However, note that the Mann-Whitney U test has a slightly different null hypothesis; that the distributions of each group is equal).

### Value

A results object containing:

results\$ttest	a table containing the t-test results
results\$assum\$norm	a table containing the normality tests
results\$assum\$eqv	a table containing the homogeneity of variances tests
results\$desc	a table containing the group descriptives
results\$plots	an array of groups of plots

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$ttest$asDF
as.data.frame(results$ttest)
```

### Examples

```
data('ToothGrowth')
ttestIS(formula = len ~ supp, data = ToothGrowth)
```

```

#
# INDEPENDENT SAMPLES T-TEST
#
# Independent Samples T-Test
# -----
#               statistic    df      p
# -----
# len  Student's t      1.92   58.0   0.060
# -----
#
#

```

---

ttestOneS

*One Sample T-Test*


---

### Description

The Student's One-sample t-test is used to test the null hypothesis that the true mean is equal to a particular value (typically zero). A low p-value suggests that the null hypothesis is not true, and therefore the true mean must be different from the test value.

### Usage

```

ttestOneS(data, vars, students = TRUE, bf = FALSE, bfPrior = 0.707,
  wilcoxon = FALSE, testValue = 0, hypothesis = "dt", norm = FALSE,
  qq = FALSE, meanDiff = FALSE, ci = FALSE, ciWidth = 95,
  effectSize = FALSE, ciES = FALSE, ciWidthES = 95, desc = FALSE,
  plots = FALSE, miss = "perAnalysis", mann = FALSE)

```

### Arguments

data	the data as a data frame
vars	a vector of strings naming the variables of interest in data
students	TRUE (default) or FALSE, perform Student's t-tests
bf	TRUE or FALSE (default), provide Bayes factors
bfPrior	a number between 0.5 and 2.0 (default 0.707), the prior width to use in calculating Bayes factors
wilcoxon	TRUE or FALSE (default), perform Wilcoxon signed rank tests
testValue	a number specifying the value of the null hypothesis
hypothesis	'dt' (default), 'gt' or 'lt', the alternative hypothesis; different to testValue, greater than testValue, and less than testValue respectively
norm	TRUE or FALSE (default), perform Shapiro-wilk tests of normality
qq	TRUE or FALSE (default), provide a Q-Q plot of residuals
meanDiff	TRUE or FALSE (default), provide means and standard deviations
ci	TRUE or FALSE (default), provide confidence intervals for the mean difference

ciWidth	a number between 50 and 99.9 (default: 95), the width of confidence intervals
effectSize	TRUE or FALSE (default), provide Cohen's d effect sizes
ciES	TRUE or FALSE (default), provide confidence intervals for the effect-sizes
ciWidthES	a number between 50 and 99.9 (default: 95), the width of confidence intervals for the effect sizes
desc	TRUE or FALSE (default), provide descriptive statistics
plots	TRUE or FALSE (default), provide descriptive plots
miss	'perAnalysis' or 'listwise', how to handle missing values; 'perAnalysis' excludes missing values for individual dependent variables, 'listwise' excludes a row from all analyses if one of its entries is missing.
mann	deprecated

### Details

The Student's One-sample t-test assumes that the data are from a normal distribution – in the case that one is unwilling to assume this, the non-parametric Wilcoxon signed-rank can be used in it's place (However, note that the Wilcoxon signed-rank has a slightly different null hypothesis; that the \*median\* is equal to the test value).

### Value

A results object containing:

results\$ttest	a table containing the t-test results
results\$normality	a table containing the normality test results
results\$descriptives	a table containing the descriptives
results\$plots	an image of the descriptive plots
results\$qq	an array of Q-Q plots

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$ttest$asDF
as.data.frame(results$ttest)
```

### Examples

```
data('ToothGrowth')

ttestOneS(ToothGrowth, vars = vars(len, dose))

#
# ONE SAMPLE T-TEST
#
# One Sample T-Test
# -----
#               statistic    df    p
# -----
```

```
#   len   Student's t      19.1   59.0   < .001
#   dose  Student's t      14.4   59.0   < .001
# -----
#
```

ttestPS

*Paired Samples T-Test***Description**

The Student's paired samples t-test (sometimes called a dependent-samples t-test) is used to test the null hypothesis that the difference between pairs of measurements is equal to zero. A low p-value suggests that the null hypothesis is not true, and that the difference between the measurement pairs is not zero.

**Usage**

```
ttestPS(data, pairs, students = TRUE, bf = FALSE, bfPrior = 0.707,
        wilcoxon = FALSE, hypothesis = "different", norm = FALSE,
        qq = FALSE, meanDiff = FALSE, ci = FALSE, ciWidth = 95,
        effectSize = FALSE, ciES = FALSE, ciWidthES = 95, desc = FALSE,
        plots = FALSE, miss = "perAnalysis")
```

**Arguments**

data	the data as a data frame
pairs	a list of lists specifying the pairs of measurement in data
students	TRUE (default) or FALSE, perform Student's t-tests
bf	TRUE or FALSE (default), provide Bayes factors
bfPrior	a number between 0.5 and 2 (default 0.707), the prior width to use in calculating Bayes factors
wilcoxon	TRUE or FALSE (default), perform Wilcoxon signed rank tests
hypothesis	'different' (default), 'oneGreater' or 'twoGreater', the alternative hypothesis; measure 1 different to measure 2, measure 1 greater than measure 2, and measure 2 greater than measure 1 respectively
norm	TRUE or FALSE (default), perform Shapiro-wilk normality tests
qq	TRUE or FALSE (default), provide a Q-Q plot of residuals
meanDiff	TRUE or FALSE (default), provide means and standard errors
ci	TRUE or FALSE (default), provide confidence intervals
ciWidth	a number between 50 and 99.9 (default: 95), the width of confidence intervals
effectSize	TRUE or FALSE (default), provide effect sizes
ciES	TRUE or FALSE (default), provide confidence intervals for the effect-sizes



ciWidthES	a number between 50 and 99.9 (default: 95), the width of confidence intervals for the effect sizes
desc	TRUE or FALSE (default), provide descriptive statistics
plots	TRUE or FALSE (default), provide descriptive plots
miss	'perAnalysis' or 'listwise', how to handle missing values; 'perAnalysis' excludes missing values for individual dependent variables, 'listwise' excludes a row from all analyses if one of its entries is missing

## Details

The Student's paired samples t-test assumes that pair differences follow a normal distribution – in the case that one is unwilling to assume this, the non-parametric Wilcoxon signed-rank can be used in it's place (However, note that the Wilcoxon signed-rank has a slightly different null hypothesis; that the two groups of measurements follow the same distribution).

## Value

A results object containing:

results\$ttest	a table containing the t-test results
results\$norm	a table containing the normality test results
results\$desc	a table containing the descriptives
results\$plots	an array of the descriptive plots

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$ttest$asDF
as.data.frame(results$ttest)
```

## Examples

```
data('bugs', package = 'jmv')

ttestPS(bugs, pairs = list(
  list(i1 = 'LDLF', i2 = 'LDHF')))

#
# PAIRED SAMPLES T-TEST
#
# Paired Samples T-Test
# -----
#
#               statistic    df    p
# -----
# LDLF    LDHF    Student's t    -6.65    90.0    < .001
# -----
#
```

# Index

- \* **data**
  - Big5, 15
  - iris, 31
  - ToothGrowth, 52
- ancova, 2
- ANOVA, 5
- anovaNP, 7
- anovaOneW, 8
- anovaRM, 10
- anovaRMNP, 13
- as.data.frame, 4, 7, 8, 10, 12, 14, 17, 20, 22, 24, 25, 28, 33, 35, 38, 41, 43, 45, 47–49, 51, 53, 55, 57
- Big5, 15
- bugs, 15
- cfa, 15
- contTables, 18
- contTablesPaired, 21
- corrMatrix, 23
- corrPart, 24
- descriptives, 26
- efa, 29
- iris, 31
- linReg, 31
- logLinear, 34
- logRegBin, 36
- logRegMulti, 39
- logRegOrd, 42
- mancova, 44
- pca, 46
- propTest2, 47
- propTestN, 49
- reliability, 50
- ToothGrowth, 52
- ttestIS, 52
- ttestOneS, 54
- ttestPS, 56