**## Split up a string (with separators) into a character vector.**

cc(..., sep = "auto", trim = TRUE)

**## Set working directory to the path of currently opened file.**

bruceR::set.wd(ask=TRUE)

**## Import data from a file (TXT, CSV, Excel, SPSS, Stata, ...) or clipboard.**

import(

file,

encoding = NULL,

header = "auto",

sheet = NULL,

range = NULL,

pkg = c("haven", "foreign"),

value.labels = FALSE,

as = "data.frame",

verbose = FALSE

)

**## Export data to a file (TXT, CSV, Excel, SPSS, Stata, ...) or clipboard.**

export(

x,

file,

encoding = NULL,

header = "auto",

sheet = NULL,

overwrite = TRUE,

verbose = FALSE

)

**## Check dependencies of R packages.**

pkg\_depend(pkgs, excludes = NULL)

**## Format numeric values.**

formatF(x, digits = 3)

**## Format "1234" to "1,234".**

formatN(x, mark = ",")

**## Print a three-line table (to R Console and Microsoft Word).**

print\_table(

x,

digits = 3,

nspaces = 1,

row.names = TRUE,

col.names = TRUE,

title = "",

note = "",

append = "",

line = TRUE,

file = NULL,

file.align.head = "auto",

file.align.text = "auto"

)

**## Create, modify, and delete variables.**

add(data, expr, when, by, drop = FALSE)

**## Multivariate computation.**

COUNT(data, var = NULL, items = NULL, vars = NULL, varrange = NULL, value = NA)

MODE(data, var = NULL, items = NULL, vars = NULL, varrange = NULL)

SUM(

data,

var = NULL,

items = NULL,

vars = NULL,

varrange = NULL,

rev = NULL,

range = likert,

likert = NULL,

na.rm = TRUE

)

.sum(

var = NULL,

items = NULL,

vars = NULL,

varrange = NULL,

rev = NULL,

range = likert,

likert = NULL,

na.rm = TRUE

)

MEAN(

data,

var = NULL,

items = NULL,

vars = NULL,

varrange = NULL,

rev = NULL,

range = likert,

likert = NULL,

na.rm = TRUE

)

.mean(

var = NULL,

items = NULL,

vars = NULL,

varrange = NULL,

rev = NULL,

range = likert,

likert = NULL,

na.rm = TRUE

)

STD(

data,

var = NULL,

items = NULL,

vars = NULL,

varrange = NULL,

rev = NULL,

range = likert,

likert = NULL,

na.rm = TRUE

)

CONSEC(

data,

var = NULL,

items = NULL,

vars = NULL,

varrange = NULL,

values = 0:9

)

**## Recode a variable.**

RECODE(var, recodes)

**## Rescale a variable (e.g., from 5-point to 7-point).**

RESCALE(var, from = range(var, na.rm = T), to)

**## Search, match, and look up values (like Excel's functions INDEX + MATCH).**

LOOKUP(

data,

vars,

data.ref,

vars.ref,

vars.lookup,

return = c("new.data", "new.var", "new.value")

)

**## Reliability analysis (Cronbach's αα and McDonald's ωω).**

Alpha(data, var, items, vars = NULL, varrange = NULL, rev = NULL, digits = 3)

**## Principal Component Analysis (PCA) and Exploratory Factor analysis (EFA).**

EFA or PCA(

data,

var,

items,

vars = NULL,

varrange = NULL,

rev = NULL,

method = c("pca", "pa", "ml", "minres", "uls", "ols", "wls", "gls", "alpha"),

rotation = c("none", "varimax", "oblimin", "promax", "quartimax", "equamax"),

nfactors = c("eigen", "parallel", "(any number >= 1)"),

sort.loadings = TRUE,

hide.loadings = 0,

plot.scree = TRUE,

kaiser = TRUE,

max.iter = 25,

min.eigen = 1,

digits = 3,

file = NULL

)

**## Confirmatory Factor Analysis (CFA).**

CFA(

data,

model = "A =~ a[1:5]; B =~ b[c(1,3,5)]; C =~ c1 + c2 + c3",

estimator = "ML",

highorder = "",

orthogonal = FALSE,

missing = "listwise",

digits = 3,

file = NULL

)

**## Descriptive statistics.**

Describe(

data,

all.as.numeric = TRUE,

digits = 2,

file = NULL,

plot = FALSE,

upper.triangle = FALSE,

upper.smooth = "none",

plot.file = NULL,

plot.width = 8,

plot.height = 6,

plot.dpi = 500

)

**## Frequency statistics.**

Freq(x, varname, labels, sort = "", digits = 1, file = NULL)

**## Correlation analysis.**

Corr(

data,

method = "pearson",

p.adjust = "none",

all.as.numeric = TRUE,

digits = 2,

file = NULL,

plot = TRUE,

plot.r.size = 4,

plot.colors = NULL,

plot.file = NULL,

plot.width = 8,

plot.height = 6,

plot.dpi = 500

)

**## Test the difference between two correlations.**

cor\_diff(r1, n1, r2, n2, n = NULL, rcov = NULL)

**## Multilevel correlations (within-level and between-level).**

cor\_multilevel(data, group, digits = 3)

**## One-sample, independent-samples, and paired-samples t-test.**

TTEST(

data,

y,

x = NULL,

paired = FALSE,

paired.d.type = "dz",

var.equal = TRUE,

mean.diff = TRUE,

test.value = 0,

test.sided = c("=", "<", ">"),

factor.rev = TRUE,

bayes.prior = "medium",

digits = 2,

file = NULL

)

**## Multi-factor ANOVA.**

MANOVA(

data,

subID = NULL,

dv = NULL,

dvs = NULL,

dvs.pattern = NULL,

between = NULL,

within = NULL,

covariate = NULL,

ss.type = "III",

sph.correction = "none",

aov.include = FALSE,

digits = 3,

file = NULL

)

**## Simple-effect analysis and post-hoc multiple comparison.**

EMMEANS(

model,

effect = NULL,

by = NULL,

contrast = "pairwise",

reverse = TRUE,

p.adjust = "bonferroni",

sd.pooled = NULL,

model.type = "multivariate",

digits = 3,

file = NULL

)

**## Tidy report of regression models.**

model\_summary(

model.list,

std = FALSE,

digits = 3,

file = NULL,

check = TRUE,

zero = ifelse(std, FALSE, TRUE),

modify.se = NULL,

modify.head = NULL,

line = TRUE,

bold = 0,

...

)

**## Tidy report of lavaan model.**

lavaan\_summary(

lavaan,

ci = c("raw", "boot", "bc.boot", "bca.boot"),

nsim = 100,

seed = NULL,

digits = 3,

print = TRUE,

covariance = FALSE,

file = NULL

)

**## Tidy report of GLM (lm and glm models).**

GLM\_summary(model, robust = FALSE, cluster = NULL, digits = 3, ...)

**## Tidy report of HLM (lmer and glmer models).**

HLM\_summary(model = NULL, test.rand = FALSE, digits = 3, ...)

**## Tidy report of HLM indices: ICC(1), ICC(2), and rWG/rWG(J).**

HLM\_ICC\_rWG(

data,

group,

icc.var,

rwg.vars = icc.var,

rwg.levels = 0,

digits = 3

)

**## Regression analysis.**

regress(

formula,

data,

family = NULL,

digits = 3,

robust = FALSE,

cluster = NULL,

test.rand = FALSE

)

**## PROCESS for mediation and/or moderation analyses.**

PROCESS(

data,

y = "",

x = "",

meds = c(),

mods = c(),

covs = c(),

clusters = c(),

hlm.re.m = "",

hlm.re.y = "",

hlm.type = c("1-1-1", "2-1-1", "2-2-1"),

med.type = c("parallel", "serial"),

mod.type = c("2-way", "3-way"),

mod.path = c("x-y", "x-m", "m-y", "all"),

cov.path = c("y", "m", "both"),

mod1.val = NULL,

mod2.val = NULL,

ci = c("boot", "bc.boot", "bca.boot", "mcmc"),

nsim = 100,

seed = NULL,

center = TRUE,

std = FALSE,

digits = 3,

file = NULL

)

**## Tidy report of mediation analysis.**

med\_summary(model, digits = 3, file = NULL)

**## Grand-mean centering.**

grand\_mean\_center(data, vars = names(data), std = FALSE, add.suffix = "")

**## Group-mean centering.**

group\_mean\_center(

data,

vars = setdiff(names(data), by),

by,

std = FALSE,

add.suffix = "",

add.group.mean = "\_mean"

)

**## Cross-correlation analysis.**

ccf\_plot(

formula,

data,

lag.max = 30,

sig.level = 0.05,

xbreaks = seq(-100, 100, 10),

ybreaks = seq(-1, 1, 0.2),

ylim = NULL,

alpha.ns = 1,

pos.color = "black",

neg.color = "black",

ci.color = "blue",

title = NULL,

subtitle = NULL,

xlab = "Lag",

ylab = "Cross-Correlation"

)

**## Granger causality test (bivariate).**

granger\_test(formula, data, lags = 1:5, test.reverse = TRUE, file = NULL, ...)

**## Granger causality test (multivariate).**

granger\_causality(

varmodel,

var.y = NULL,

var.x = NULL,

test = c("F", "Chisq"),

file = NULL,

check.dropped = FALSE

)

**## Show colors.**

show\_colors(colors)