Package ‘psychtoolbox’

March 3, 2024

Title Tools for psychology research and psychometrics

Version 0.0.1

Description This package contains functions helping to analyse psychological data.

URL https://gitlab.com/lukas.novak/psychtoolbox

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Imports coin,
docxtractr,
dplyr,
equaltestMI,
foreign,
insight,
lavaan,
magrittr,
rmarkdown,
rstatix,
stats,
tidyr,
expss,
stringr,
tidyselect,
vctrs,
purrr,
reshape2,
janitor,
broom,
car

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Depends R (>= 2.10)

R topics documented:

`clin_sig_chang` *Clinically significant change*

Description

This easy function calculates Clinically significant change (clinical cut-off scores) as defined by Jacobson and Truax (1991).

Usage

```
clin_sig_chang(SD_0, SD_1, M_1, M_0)
```

Arguments

`SD_0`	standard deviation of the non-clinical population
`SD_1`	standard deviation of the clinical population
`M_1`	mean of the clinical population
`M_0`	mean of the non-clinical population

Format

numeric vector of values

Details

This function computes cut-off score differentiating between the clinical and non-clinical population based on the Jacobson and Truax (1991) formula (p. 13). The mathematical formula can be also found in Biescad & Timulak(2014, p. 150).

Value

numeric vector

Author(s)

Lukas Novak, <lukasjirinovak@gmail.com>

References

Jacobson, N. S., & Truax, P. (1991). Clinical significance: A statistical approach to defining meaningful change in psychotherapy research. Journal of Consulting and Clinical Psychology, 59(1), 12-19, DOI: https://doi.org/10.1037/0022-006X.59.1.12

Matus Biescad & Ladislav Timulak (2014). Measuring psychotherapy outcomes in routine practice: Examining Slovak versions of three commonly used outcome instruments, European Journal of Psychotherapy & Counselling, 16:2, 140-162, DOI: https://doi.org/10.1080/13642537.2014.895772

See Also

`RCI()` function for calculation of the Reliable Change Index

Examples

```
clin.cut.off=clin_sig_chang(SD_0 = 3.5,
                            SD_1 = 2.1,
                            M_0 = 4.2,
                            M_1 = 12.1)
clin.cut.off
```

---

lasy.lin.reg — *Lasy linear regression function*

Description

This function performs linear regression and print results in tibble output. This function aims to provide the results of the regression analysis in the format, which is frequently desired in academic journals.

Usage

```
lasy.lin.reg(
  independent.var,
  dependent.var,
  print.cov = FALSE,
  Z_score_independent = FALSE,
  check_multicolinearity = TRUE,
  covariates = NULL,
  data
)
```

Arguments

`independent.var`	independent variable/s
`dependent.var`	dependent variable/s
`print.cov`	Print effect of covariates, default is FALSE
`Z_score_independent`	Should independent variables be z-scored? Default is FALSE

`check_multicolinearity`
: Should multicolinearity assumption be checked? Default is TRUE

`covariates` covariates to be included in a model

`data` data frame or tibble object

Format

An object of class `"tibble"`

Details

Currently, this function does not provide model fit indicators such as AIC or BIC.

Value

data frame

Author(s)

Lukas Novak, <lukasjirinovak@gmail.com>

References

Welch, B. L. (1947). "The generalization of "Student's" problem when several different population variances are involved". Biometrika. 34 (1–2): 28–35.

Wilcoxon, F., Individual Comparisons by Ranking Methods, Biometrics Bulletin, Vol. 1, 1945, pp. 80–83. DOI:10.2307/3001968

Dunn, O. J. (1961) Multiple comparisons among means. Journal of the American Statistical Association. 56, 52–64.

Games, P. A., Keselman, H. J., & Clinch, J. J. Tests for homogeneity of variance in factorial designs. Psychological Bulletin, 86, 978–984

Examples

```
# data simulation
library(dplyr)
data.PAQ = tibble(.rows = 1000)
data.PAQ <- data.PAQ %>%
 mutate("multiple__exper_1" = rnorm(n = nrow(data.PAQ), mean = 20, sd = 5)) %>%
 mutate("exper_1" = rnorm(n = nrow(data.PAQ),  mean = 20, sd = 5)) %>%
 mutate("exper_1" = rnorm(n = nrow(data.PAQ),  mean = 20, sd = 5),
        "exper_2" = rnorm(n = nrow(data.PAQ),  mean = 20, sd = 5),
        "last_binary_vasdl" = rnorm(n = nrow(data.PAQ), mean = 20, sd = 20),
        "last_binary_val2" = rnorm(n = nrow(data.PAQ), mean = 5, sd = 5),
        "last_binary_val3" = rnorm(n = nrow(data.PAQ), mean = 10, sd = 40),
        "last_binary_val4" = rnorm(n = nrow(data.PAQ), mean = 50, sd = 5),
        "last_binary_val5" = rnorm(n = nrow(data.PAQ), mean = 5, sd = 4),
        "last_binary_val6" = rnorm(n = nrow(data.PAQ), mean = 65, sd = 5))

lin.reg.output <- lasy.lin.reg(independent.var = c("last_binary_vasdl","last_binary_val2"),
                              covariates = c("last_binary_val6"),
                              dependent.var = c("last_binary_val5","last_binary_val4"),
                              Z_score_independent = FALSE,
                              check_multicolinearity = TRUE,
```

```
                           data = data.PAQ,
                           print.cov = FALSE)

print(lin.reg.output)
```

lasy.log.reg *Lasy logistic regression function*

Description

This function performs logistic regression and print results in tibble output. This function aims to provide the results of the regression analysis in the format, which is frequently desired in academic journals.

Usage

```
lasy.log.reg(
  independent.var,
  dependent.var,
  print.cov = FALSE,
  covariates = NULL,
  check_multicolinearity = TRUE,
  data
)
```

Arguments

`independent.var`	independent variable/s
`dependent.var`	dependent variable/s
`print.cov`	Print effect of covariates, default is FALSE
`covariates`	covariates to be included in a model
`check_multicolinearity`	Should multicolinearity assumption be checked? Default is TRUE
`data`	data frame or tibble object

Format

An object of class `"tibble"`

Details

Currently, this function does not provide model fit indicators such as AIC or BIC.

Value

data frame

Author(s)

Lukas Novak, `<lukasjirinovak@gmail.com>`

References

Welch, B. L. (1947). "The generalization of "Student's" problem when several different population variances are involved". Biometrika. 34 (1–2): 28–35.

Wilcoxon, F., Individual Comparisons by Ranking Methods, Biometrics Bulletin, Vol. 1, 1945, pp. 80–83. DOI:10.2307/3001968

Dunn, O. J. (1961) Multiple comparisons among means. Journal of the American Statistical Association. 56, 52–64.

Games, P. A., Keselman, H. J., & Clinch, J. J. Tests for homogeneity of variance in factorial designs. Psychological Bulletin, 86, 978–984

Examples

```
# data loading
data(paq.validation.study)
regress.output <- lasy.log.reg(independent.var = c("TEQ","PAQ"),
                               covariates = c("Age"),
                               dependent.var = c("econom_stat_dich",
                                                 "family_status",
                                                 "edu_dich"),
                               data = paq.validation.study)

print(regress.output)
```

`mitab` *Measurement invariance table*

Description

Measurement invariance table

Usage

```
mitab(
  group1_nam,
  group2_nam,
  ordered,
  model,
  data,
  std.lv,
  meanstructure,
  group,
  yes_no_results,
  estimator,
  robust = FALSE,
  cfi.difference = TRUE,
  rmsea.difference = TRUE,
  ...
)
```

Arguments

`group1_nam`	name of the first group
`group2_nam`	name of the second group
`ordered`	logical, if set to TRUE items will be treated as ordered variables
`model`	lavaan model to test
`data`	data frame or tibble
`std.lv`	logical, if TRUE than standardized loadings are stored in temporal output
`meanstructure`	logical, if TRUE than model with meanstructure is estimated
`group`	name of grouping variable
`yes_no_results`	logical, if TRUE than lasy output indicating difference between models is added, currently working only based on CFI and RMSEA
`estimator`	name of estimator to be used during fitting procedure
`robust`	logical, if TRUE, than robust results are printed, working only with estimators providing robust results (e.g. MLR or DWLS)
`cfi.difference`	logical, if TRUE, delta of the CFI is printed in output
`rmsea.difference`	logical, if TRUE, delta of RMSEA is printed in output
`...`	optional arguments for CFA function

Format

An object of class `"tibble"`

Details

This function creates table with the key output from measurement invariance testing.

Value

data frame

Author(s)

Lukas Novak, `<lukasjirinovak@gmail.com>`

References

Myles Hollander and Douglas A. Wolfe (1973). Nonparametric Statistical Methods. New York: John Wiley & Sons. Pages 27–33 (one-sample), 68–75 (two-sample). Or second edition (1999).

Examples

```
# The famous Holzinger and Swineford (1939) example
HS.model <- ' visual  =~ x1 + x2 + x3
textual =~ x4 + x5 + x6
speed   =~ x7 + x8 + x9 '

library(lavaan)
dat <- HolzingerSwineford1939
res.tab.mi <- mitab(
```

```
group1_nam = "Grant-White",
group2_nam = "Pasteur",
ordered = FALSE,
model = HS.model,
data = dat,
std.lv = TRUE,
meanstructure = TRUE,
group = "school",
yes_no_results = TRUE,
estimator = "MLR",
robust = TRUE,
cfi.difference = TRUE,
rmsea.difference = TRUE
)

print(res.tab.mi)
```

`mult.g.comp` *Robust multi-group comparison*

Description

This function allows to compare multiple groups in multiple outcome variables with violated parametric assumptions.

Usage

```
mult.g.comp(
  df,
  outcome.var,
  groups,
  desc_only = FALSE,
  short_results = TRUE,
  remove_missings = FALSE,
  percent_decimals = 2
)
```

Arguments

`df`	data frame or tibble object
`outcome.var`	continuous variable/s
`groups`	grouping variable/s
`desc_only`	print only descriptive statistics, default is FALSE
`short_results`	prints only significance stars without numerical results, default is TRUE
`remove_missings`	remove missing values from a table, default is FALSE
`percent_decimals`	number of decimals used to round percenages, default is 2

Format

An object of class "`tibble`"

Details

Currently, this function does not report effect size from post-hoc tests.

Two group comparison:
If there is less than three groups, the Welch test or the Wilcoxon test depending on data distribution.

Three and more groups comparison:
If more than two groups are present in data, the Dunn test or Games-Howell test is performed.

Value

data frame

Author(s)

Lukas Novak, <lukasjirinovak@gmail.com>

References

Welch, B. L. (1947). "The generalization of "Student's" problem when several different population variances are involved". Biometrika. 34 (1–2): 28–35.

Wilcoxon, F., Individual Comparisons by Ranking Methods, Biometrics Bulletin, Vol. 1, 1945, pp. 80–83. DOI:10.2307/3001968

Dunn, O. J. (1961) Multiple comparisons among means. Journal of the American Statistical Association. 56, 52–64.

Games, P. A., Keselman, H. J., & Clinch, J. J. Tests for homogeneity of variance in factorial designs. Psychological Bulletin, 86, 978–984

Examples

```
# data loading
tab.1=mult.g.comp(df = paq.validation.study, outcome.var = c("PAQ","G_DIF","G_DDF","G_EOT"),
groups = c("economical_status",
          "Gender",
          "education",
          "family_status"))
# printing the output
print(tab.1)
```

---

paq.validation.study   *paq.validation.study*

Description

This dataset contains data which were used for validation of the Perth Alexithymia Questionnaire in the Czech Republic.

Usage

```
paq.validation.study
```

Format

A data frame with 848 rows and 53 variables:

P_DIF double COLUMN_DESCRIPTION

P_DDF double COLUMN_DESCRIPTION

P_DAF double COLUMN_DESCRIPTION

N_DIF double COLUMN_DESCRIPTION

N_DDF double COLUMN_DESCRIPTION

N_DAF double COLUMN_DESCRIPTION

G_EOT double COLUMN_DESCRIPTION

G_DIF double COLUMN_DESCRIPTION

G_DDF double COLUMN_DESCRIPTION

G_DAF double COLUMN_DESCRIPTION

family_status integer COLUMN_DESCRIPTION

Gender integer COLUMN_DESCRIPTION

TEQ_1 double COLUMN_DESCRIPTION

TEQ_3 double COLUMN_DESCRIPTION

TEQ_5 double COLUMN_DESCRIPTION

TEQ_16 double COLUMN_DESCRIPTION

TEQ_CON_2 double COLUMN_DESCRIPTION

TEQ_CON_4 double COLUMN_DESCRIPTION

TEQ_CON_14 double COLUMN_DESCRIPTION

TEQ double COLUMN_DESCRIPTION

Age double COLUMN_DESCRIPTION

age_group double COLUMN_DESCRIPTION

age.quality double COLUMN_DESCRIPTION

ethnicity integer COLUMN_DESCRIPTION

education integer COLUMN_DESCRIPTION

economical_status integer COLUMN_DESCRIPTION

PAQ_1 double COLUMN_DESCRIPTION

PAQ_2 double COLUMN_DESCRIPTION

PAQ_3 double COLUMN_DESCRIPTION

PAQ_4 double COLUMN_DESCRIPTION

PAQ_5 double COLUMN_DESCRIPTION

PAQ_6 double COLUMN_DESCRIPTION

PAQ_7 double COLUMN_DESCRIPTION

PAQ_8 double COLUMN_DESCRIPTION

PAQ_9 double COLUMN_DESCRIPTION

PAQ_10 double COLUMN_DESCRIPTION

PAQ_11 double COLUMN_DESCRIPTION

PAQ_12 double COLUMN_DESCRIPTION

PAQ_13 double COLUMN_DESCRIPTION

PAQ_14 double COLUMN_DESCRIPTION

PAQ_15 double COLUMN_DESCRIPTION

PAQ_16 double COLUMN_DESCRIPTION

PAQ_17 double COLUMN_DESCRIPTION

PAQ_18 double COLUMN_DESCRIPTION

PAQ_19 double COLUMN_DESCRIPTION

PAQ_20 double COLUMN_DESCRIPTION

PAQ_21 double COLUMN_DESCRIPTION

PAQ_22 double COLUMN_DESCRIPTION

PAQ_23 double COLUMN_DESCRIPTION

PAQ_24 double COLUMN_DESCRIPTION

PAQ double COLUMN_DESCRIPTION

edu_dich integer COLUMN_DESCRIPTION

econom_stat_dich integer COLUMN_DESCRIPTION

dichotomization of variables

paq.validation.study <- paq.validation.study %>% dplyr::mutate(edu_dich = as.factor(ifelse(education == "University master or higher", "University","lower_edu")), econom_stat_dich = as.factor(ifelse(economical_status == "Student", "Student","non_student")))

RCI *Reliable Change Index (RCI)*

Description

This function calculates Reliable Change Index (RCI) as modifed by Wiger and Solberg (2001, p.148).

Usage

```
RCI(SD_0, test.ret.rel)
```

Arguments

SD_0	standard deviation of the non-clinical population
test.ret.rel	test-retest reliability of the instrument

Format

numeric vector of values

Details

This function computes value corresponding to "the minimum amount of change that could not be attributed to the error of measurement" (Biescad & Timulak, 2014, p. 150). If score change from before to post treatment is lower that value resulting from this function, than change in client score can not be attributed to the effectiveness of the therapy but rather other factors such as a measurement error (Biescad & Timulak, 2014). This function is a result of modification of the original Jacobson and Truax (1991) formula by Wiger and Solberg (2001, p.148).

Value

numeric vector

Author(s)

Lukas Novak, <lukasjirinovak@gmail.com>

References

Jacobson, N. S., & Truax, P. (1991). Clinical significance: A statistical approach to defining meaningful change in psychotherapy research. Journal of Consulting and Clinical Psychology, 59(1), 12-19, DOI: https://doi.org/10.1037/0022-006X.59.1.12

Matus Biescad & Ladislav Timulak (2014). Measuring psychotherapy outcomes in routine practice: Examining Slovak versions of three commonly used outcome instruments, European Journal of Psychotherapy & Counselling, 16:2, 140-162, DOI: https://doi.org/10.1080/13642537.2014.895772

Wiger, D. E., & Solberg, K. B. (2001). Tracking Mental Health Outcomes: A Therapist's Guide to Measuring Client Progress, Analyzing Data, and Improving Your Practice (1., Vol. 2001). Wiley.

See Also

clin_sig_chang() function for calculation of the clinical cut-off scores

Examples

```
re.ch.in = RCI(SD_0 = 4.87, test.ret.rel = 0.66)
re.ch.in
```

---

small_boot_table   *Summarizing Bootstrapped Network Estimates from 'bootnet'*

Description

Summarizing Bootstrapped Network Estimates from 'bootnet'

Usage

```
small_boot_table(bootnet_output, predict_function_output = NULL)
```

Arguments

`bootnet_output` output from the 'bootnet' package after bootstrapping network analysis.

`predict_function_output`
optional, output from a predictability analysis function.

Format

An object of class `"tibble"`.

Details

This function provides a convenient summary of bootstrapped network estimates from the 'bootnet' package, including edge weights and their confidence intervals. Optionally, it can also include node predictability metrics if provided. It is tailored to enhance the usability of 'bootnet' output by summarizing critical metrics in a concise format.

Value

A tibble summarizing edge weights and, optionally, node predictability.

Author(s)

Lukas Novak, <lukasjirinovak@gmail.com>

References

Epskamp, S., Borsboom, D., & Fried, E. I. (2018). Network analysis: An integrative approach to the structure of psychopathology. Annual review of clinical psychology, 14, 91-121.

Examples

```
# Load required package
library(bootnet)

# Simulate data using base R
set.seed(123)
Sigma <- matrix(0.5, nrow = 5, ncol = 5) + diag(0.5, 5)
chol_decomp <- chol(Sigma)
z <- matrix(rnorm(100 * 5), nrow = 100, ncol = 5)
sim_data <- z %*% chol_decomp
sim_data_df <- as.data.frame(sim_data)

# Estimate a network using 'EBICglasso'
estimate <- bootnet::estimateNetwork(sim_data_df, default = "EBICglasso")

# Perform bootstrapping on the estimated network
boot_results <- bootnet::bootnet(estimate, nBoots = 100, nCores = 1)

# Summarize the bootstrapped network
summary <- small_boot_table(bootnet_output = boot_results)
print(summary)
```

`two.g.comp` *Automatic two-groups comparison*

Description

Automatic two-groups comparison

Usage

```
two.g.comp(df, y, group.var)
```

Arguments

`df`	data frame or tibble with one socio-demographic variable and one continuous variable
`y`	continuous variable
`group.var`	binary grouping variable

Format

An object of class "`tibble`"

Details

This function computes either Wilcox test or t-test depending on whether homogeneity of variances assumption is met or not.

Value

data frame

Author(s)

Lukas Novak, <lukasjirinovak@gmail.com>

References

Myles Hollander and Douglas A. Wolfe (1973). Nonparametric Statistical Methods. New York: John Wiley & Sons. Pages 27–33 (one-sample), 68–75 (two-sample). Or second edition (1999).

Examples

```
Sex = rbinom(1:100, size = 0:1, prob = .5)
test.dat = data.frame(cbind(Sex), Empathy = ifelse(Sex == 0,
                                              rnorm(n = 1:100, mean = 50, sd = 10),
                                              rnorm(n = 1:100, mean = 10, sd = 25)))


# running the function
two.g.comp.out.EC = two.g.comp(df = test.dat, y = "Empathy", group.var = "Sex")
# printing the output
print(two.g.comp.out.EC)
```

`word2pdf` *word to pdf*

Description

Conversion of word document to pdf using either R Markdown package or Libre office. The latter represents higher quality approach - in general.

Usage

```
word2pdf(imp_file, out_file)
```

Arguments

`imp_file`	name of the word document to convert - without docx suffix
`out_file`	name of output pdf file without - without pdf suffix

Format

An object of class "pdf"

Details

this function is currently running only on windows

Value

pdf file

Author(s)

Lukas Novak, <lukasjirinovak@gmail.com>

Examples

```
# example from word do pdf
#word2pdf(imp_file = "example.docx",out_file = "example1.pdf")
```

Index