

Package: tabledown (via r-universe)

August 30, 2024

Title Create Publication Quality Tables and Plots

Version 1.0.0

Description Create publication quality plots and tables for Item Response Theory and Classical Test theory based item analysis, exploratory and confirmatory factor analysis.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

BugReports <https://github.com/masiraji/tabledown/issues>

Language en-US

Depends R (>= 2.10)

Imports psych, MOTE, stats, dplyr, tibble, magrittr, data.table, tidyselect, lavaan, mirt, ggplot2, plotly, kutils, tidy

Suggests rmarkdown, knitr, markdown, spelling, testthat (>= 3.0.0)

URL <https://masiraji.github.io/tabledown/>

Repository <https://masiraji.r-universe.dev>

RemoteUrl <https://github.com/masiraji/tabledown>

RemoteRef HEAD

RemoteSha 6edef3330bf914cc5af3b0e64120773babf3cb8e

Contents

| | |
|-------------------------|---|
| bedTime | 2 |
| cfa.tab | 3 |
| cfa.tab.multi | 3 |
| des.tab | 4 |
| fac.tab | 4 |

| | |
|--------------------------------|----|
| FFMQ.CFA | 5 |
| FFMQ.Val | 7 |
| Gantt | 10 |
| ggicc | 11 |
| ggiteminfo | 11 |
| ggreliability | 12 |
| ggreliability_plotly | 13 |
| ggtestinfo | 13 |
| ggtestinfo_se | 14 |
| ggtestinfo_se_plotly | 15 |
| gt_tab | 15 |
| normality.loop | 16 |
| Rotter | 17 |
| Spot | 20 |
| tabledown | 20 |

Index 22

| | |
|---------|--|
| bedTime | <i>A Function for calculating time spent in bed.</i> |
|---------|--|

Description

This function will help you to calculate the time a person spent in bed based on their sleep log. This type of calculation is very common in sleep research. However, as one can guess, working with dates in R is a bit tricky. This function will ease the task. More importantly you do not require to entry the dates to calculate bed time. Just wake up time and time to go to bed is enough (24 hour format).

Usage

```
bedTime(x, y)
```

Arguments

| | |
|---|--|
| x | A vector containing time to do to bed. |
| y | A vector containing time of wake. |

Value

Calculates time spent in bed in hours. Output class is numeric.

Examples

```
#Please use 24 hour format.
#Easiest way is to enter the data as character.
bed <-c("20:00", "21:00", "23:00")
wake <-c("6:00", "7:00", "8:00")
bedtime <- bedTime(bed, wake)
```

| | |
|---------|---|
| cfa.tab | <i>A Function for Creating Publication Quality Tables with CFA fit indices.</i> |
|---------|---|

Description

This function will create publication worthy tables with CFA fit indices from lavaan class object.

Usage

```
cfa.tab(x, robust = FALSE)
```

Arguments

| | |
|--------|--|
| x | A lavaan class object. |
| robust | If TRUE, will provide robust fit indices when applicable instead of the default indices. |

| | |
|---------------|---|
| cfa.tab.multi | <i>A Function for Creating Publication Quality Tables with CFA fit indices from several lavaan objects.</i> |
|---------------|---|

Description

Often researchers are required to show fit indices from several CFA models. This function will create publication worthy tables with CFA fit indices from several lavaan class objects. #' To run this function successfully one need to provide at least two lavaan objects. This command supports up-to five lavaan models.

Usage

```
cfa.tab.multi(x, y, z = NULL, a = NULL, b = NULL, robust = FALSE)
```

Arguments

| | |
|--------|--|
| x | first object of class lavaan (Mandatory). |
| y | second object of class lavaan (Mandatory). |
| z | third object of class lavaan (Optional). |
| a | fourth object of class lavaan (Optional). |
| b | fifth object of class lavaan (Optional). |
| robust | If TRUE, will provide robust fit indices when applicable instead of the default indices. |

des.tab *A Function for Descriptive data for item analysis.*

Description

This function will create a publication ready essential descriptive table for item analysis. Normality is tested using shapiro.test from base stats with Bonferroni Correction.

Usage

```
des.tab(df, reverse = FALSE)
```

Arguments

| | |
|---------|--|
| df | A data frame. |
| reverse | If TRUE, will provide indicate which items had a negative correlation and reverse them |

Value

Returns a summary table of descriptives in a data frame structure.

Examples

```
data <- tabledown::Rotter[, 11:31]
table <- des.tab(data)
```

fac.tab *A Function for Creating Publication Quality Factor Tables.*

Description

This function will create publication worthy factor tables from objects created from psych pack. I have come across this beautiful piece of codes at <https://www.anthonyschmidt.co/post/2020-09-27-efa-tables-in-r/> and modified it a bit.

Usage

```
fac.tab(x, cut, complexity = TRUE)
```

Arguments

| | |
|------------|--|
| x | A psych package object. |
| cut | The value under which all factor loading will be suppressed. |
| complexity | To add complexity parameters. |

Value

A publication ready summary table for the Factor analysis conducted by psych Package. Output structure is data frame.

Examples

```
data <- tabledown::Rotter[, 11:31]
correlations <- psych::polychoric(data, correct = 0)
fa.5F.1 <- psych::fa(r=correlations$rho, nfactors = 5, fm= "pa", rotate = "varimax",
residuals = TRUE, SMC = TRUE, n.obs = 428)
table <- fac.tab(fa.5F.1, .3)
#always save the output into an object
```

FFMQ.CFA

*Structural Validity data of FFMQ***Description**

This is the structural validation data of Bangla Five Facet Mindfulness Questionnaire

Usage

```
FFMQ.CFA
```

Format

A data frame with 277 rows and 47 variables:

```
ID double COLUMN_DESCRIPTION
Gender character COLUMN_DESCRIPTION
Education character COLUMN_DESCRIPTION
Education Years double COLUMN_DESCRIPTION
Income double COLUMN_DESCRIPTION
Profession character COLUMN_DESCRIPTION
Marital Status character COLUMN_DESCRIPTION
Social_status double COLUMN_DESCRIPTION
item1 double COLUMN_DESCRIPTION
item2 double COLUMN_DESCRIPTION
Ritem3 double COLUMN_DESCRIPTION
item4 double COLUMN_DESCRIPTION
Ritem5 double COLUMN_DESCRIPTION
item6 double COLUMN_DESCRIPTION
item7 double COLUMN_DESCRIPTION
```

Ritem8 double COLUMN_DESCRIPTION
item9 double COLUMN_DESCRIPTION
Ritem10 double COLUMN_DESCRIPTION
item11 double COLUMN_DESCRIPTION
Ritem12 double COLUMN_DESCRIPTION
Ritem13 double COLUMN_DESCRIPTION
Ritem14 double COLUMN_DESCRIPTION
item15 double COLUMN_DESCRIPTION
Ritem16 double COLUMN_DESCRIPTION
Ritem17 double COLUMN_DESCRIPTION
Ritem18 double COLUMN_DESCRIPTION
item19 double COLUMN_DESCRIPTION
item20 double COLUMN_DESCRIPTION
item21 double COLUMN_DESCRIPTION
Ritem22 double COLUMN_DESCRIPTION
Ritem23 double COLUMN_DESCRIPTION
item24 double COLUMN_DESCRIPTION
Ritem25 double COLUMN_DESCRIPTION
item26 double COLUMN_DESCRIPTION
item27 double COLUMN_DESCRIPTION
Ritem28 double COLUMN_DESCRIPTION
item29 double COLUMN_DESCRIPTION
Ritem30 double COLUMN_DESCRIPTION
item31 double COLUMN_DESCRIPTION
item32 double COLUMN_DESCRIPTION
item33 double COLUMN_DESCRIPTION
Ritem34 double COLUMN_DESCRIPTION
Ritem35 double COLUMN_DESCRIPTION
item36 double COLUMN_DESCRIPTION
item37 double COLUMN_DESCRIPTION
Ritem38 double COLUMN_DESCRIPTION
Ritem39 double COLUMN_DESCRIPTION

Source

<https://github.com/masiraji/taledown/tree/main/data-raw>

 FFMQ.Val

Correlational based Validity evidence of FFMQ

Description

Correlational based Validity evidence of Bangla FFMQ

Usage

FFMQ.Val

Format

A data frame with 255 rows and 106 variables:

```

id double COLUMN_DESCRIPTION
Age double COLUMN_DESCRIPTION
Gender double COLUMN_DESCRIPTION
Education Years double COLUMN_DESCRIPTION
Profession character COLUMN_DESCRIPTION
Marital Status character COLUMN_DESCRIPTION
Social_Status double COLUMN_DESCRIPTION
item1 double COLUMN_DESCRIPTION
item2 double COLUMN_DESCRIPTION
Ritem3 double COLUMN_DESCRIPTION
item4 double COLUMN_DESCRIPTION
Ritem5 double COLUMN_DESCRIPTION
item6 double COLUMN_DESCRIPTION
item7 double COLUMN_DESCRIPTION
Ritem8 double COLUMN_DESCRIPTION
item9 double COLUMN_DESCRIPTION
Ritem10 double COLUMN_DESCRIPTION
item11 double COLUMN_DESCRIPTION
Ritem12 double COLUMN_DESCRIPTION
Ritem13 double COLUMN_DESCRIPTION
Ritem14 double COLUMN_DESCRIPTION
item15 double COLUMN_DESCRIPTION
Ritem16 double COLUMN_DESCRIPTION
Ritem17 double COLUMN_DESCRIPTION
Ritem18 double COLUMN_DESCRIPTION
  
```

item19 double COLUMN_DESCRIPTION
item20 double COLUMN_DESCRIPTION
item21 double COLUMN_DESCRIPTION
Ritem22 double COLUMN_DESCRIPTION
Ritem23 double COLUMN_DESCRIPTION
item24 double COLUMN_DESCRIPTION
Ritem25 double COLUMN_DESCRIPTION
item26 double COLUMN_DESCRIPTION
item27 double COLUMN_DESCRIPTION
Ritem28 double COLUMN_DESCRIPTION
item29 double COLUMN_DESCRIPTION
Ritem30 double COLUMN_DESCRIPTION
item31 double COLUMN_DESCRIPTION
item32 double COLUMN_DESCRIPTION
item33 double COLUMN_DESCRIPTION
Ritem34 double COLUMN_DESCRIPTION
Ritem35 double COLUMN_DESCRIPTION
item36 double COLUMN_DESCRIPTION
item37 double COLUMN_DESCRIPTION
Ritem38 double COLUMN_DESCRIPTION
Ritem39 double COLUMN_DESCRIPTION
EI1 character COLUMN_DESCRIPTION
EI2 character COLUMN_DESCRIPTION
EI3 character COLUMN_DESCRIPTION
EI4 character COLUMN_DESCRIPTION
EI5 character COLUMN_DESCRIPTION
EI6 character COLUMN_DESCRIPTION
EI7 character COLUMN_DESCRIPTION
EI8 character COLUMN_DESCRIPTION
EI9 character COLUMN_DESCRIPTION
EI10 character COLUMN_DESCRIPTION
EI11 character COLUMN_DESCRIPTION
EI12 character COLUMN_DESCRIPTION
EI13 character COLUMN_DESCRIPTION
EI14 character COLUMN_DESCRIPTION
EI15 character COLUMN_DESCRIPTION
EI16 character COLUMN_DESCRIPTION

EI17 character COLUMN_DESCRIPTION
EI18 character COLUMN_DESCRIPTION
EI19 character COLUMN_DESCRIPTION
EI20 character COLUMN_DESCRIPTION
EI21 character COLUMN_DESCRIPTION
EI22 character COLUMN_DESCRIPTION
EI23 character COLUMN_DESCRIPTION
EI24 character COLUMN_DESCRIPTION
EI25 character COLUMN_DESCRIPTION
EI26 character COLUMN_DESCRIPTION
EI27 character COLUMN_DESCRIPTION
EI28 character COLUMN_DESCRIPTION
EI29 character COLUMN_DESCRIPTION
EI30 character COLUMN_DESCRIPTION
EI31 character COLUMN_DESCRIPTION
EI32 character COLUMN_DESCRIPTION
EI33 character COLUMN_DESCRIPTION
EI34 character COLUMN_DESCRIPTION
01 character COLUMN_DESCRIPTION
02 character COLUMN_DESCRIPTION
03 character COLUMN_DESCRIPTION
04 character COLUMN_DESCRIPTION
05 character COLUMN_DESCRIPTION
06 character COLUMN_DESCRIPTION
07 character COLUMN_DESCRIPTION
08 character COLUMN_DESCRIPTION
09 character COLUMN_DESCRIPTION
010 character COLUMN_DESCRIPTION
E1 character COLUMN_DESCRIPTION
E2 character COLUMN_DESCRIPTION
E3 character COLUMN_DESCRIPTION
E4 character COLUMN_DESCRIPTION
E5 character COLUMN_DESCRIPTION
E6 character COLUMN_DESCRIPTION
E7 character COLUMN_DESCRIPTION
E8 character COLUMN_DESCRIPTION
N1 character COLUMN_DESCRIPTION

N2 character COLUMN_DESCRIPTION
N3 character COLUMN_DESCRIPTION
N4 character COLUMN_DESCRIPTION
N5 character COLUMN_DESCRIPTION
N6 character COLUMN_DESCRIPTION
N7 character COLUMN_DESCRIPTION
N8 character COLUMN_DESCRIPTION

Source

<https://github.com/masiraji/taledown/tree/main/data-raw>

Gantt

Gantt Data

Description

Demo project breakdown to create Gantt

Usage

Gantt

Format

A data frame with 25 rows and 4 variables:

wp character Main Component
activity character Activities
start_date character Start Date
end_date character End Date

Source

<https://github.com/masiraji/taledown/tree/main/data-raw>

| | |
|-------|---|
| ggicc | <i>A Function for Creating Publication Quality Item Response Theory based item characteristic plot.</i> |
|-------|---|

Description

This function will create publication worthy Item Response Theory based item characteristic plot using ggplot2 from objects created from mirt pack. Using ggplot2 will enable the user to modify the item characteristic plot.

Usage

```
ggicc(model, item, theta)
```

Arguments

| | |
|-------|---|
| model | A mirt package fitted object. |
| item | Item number (i.e. 1,2,3,4). |
| theta | Theta range. Put only one number. Theta =3 will be considered as theta range (-3 to 3). |

Value

A publication quality item characteristic plot. Output object is a ggplot object.

Examples

```
data <- tabledown::Rotter[, 11:31]
model <- mirt::mirt(data, model = 1, itemtype = '2PL', SE = TRUE, Se.type = 'MHRM')

plot <- tabledown::ggicc(model, 1, 3)
```

| | |
|------------|--|
| ggiteminfo | <i>A Function for Creating Publication Quality Item Response Theory based item information plot.</i> |
|------------|--|

Description

This function will create publication worthy Item Response Theory based item information plot. using ggplot2 from objects created from mirt pack.

Usage

```
ggiteminfo(model, item, theta)
```

Arguments

| | |
|-------|---|
| model | A mirt package fitted object. |
| item | Item number (i.e. 1,2,3,4). |
| theta | Theta range. Put only one number. Theta =3 will be considered as theta range (-3 to 3). |

Value

A publication quality item information plot. Output object is a ggplot object.

Examples

```
data <- tabledown::Rotter[, 11:31]
model <- mirt::mirt(data, model = 1, itemtype = '2PL')

plot <- ggiteminfo(model, 1, 3)
```

| | |
|---------------|---|
| ggreliability | <i>A Function for Creating Publication Quality Item Response Theory based reliability plot.</i> |
|---------------|---|

Description

This function will create publication worthy Item Response Theory based based reliability plot with standard error using ggplot2 from objects created from mirt pack. Using ggplot2 will enable the user to modify the Item plot.

Usage

```
ggreliability(dataframe, model)
```

Arguments

| | |
|-----------|-------------------------------|
| dataframe | your data. |
| model | A mirt package fitted object. |

Value

A publication quality reliability plot (dashed line). Output object is a ggplot object.

Examples

```
data <- tabledown::Rotter[, 11:31]
model <- mirt::mirt(data, model = 1, itemtype = '2PL')

plot <- ggreliability(data, model)
```

ggreliability_plotly *A Function for Creating Item Response Theory based reliability plot based on plotly.*

Description

This function will create Item Response Theory based based reliability plot with standard error using ggplot2 and plotly from objects created from mirt pack. Using ggplot2 will enable the user to modify the Item plot.

Usage

```
ggreliability_plotly(dataframe, model)
```

Arguments

dataframe your data.
model A mirt package fitted object.

Value

A publication quality reliability plot (dashed line). Output object is a ggplot object.

Examples

```
data <- tabledown::Rotter[, 11:31]  
model <- mirt::mirt(data, model = 1, itemtype = '2PL')  
  
plot <- ggreliability_plotly(data, model)
```

ggtestinfo *A Function for Creating Publication Quality Item Response Theory based test information plot.*

Description

This function will create publication worthy Item Response Theory based Test information plot using ggplot2 from objects created from mirt pack. Using ggplot2 will enable the user to modify the Item plot.

Usage

```
ggtestinfo(dataframe, model)
```

Arguments

dataframe your data.
model A mirt package fitted object.

Value

A publication quality Test information plot. Output object is a ggplot object.

Examples

```
data <- tabledown::Rotter[, 11:31]
model <- mirt::mirt(data, model = 1, itemtype = '2PL')

plot <- ggtestinfo(data, model)
```

| | |
|---------------|--|
| ggtestinfo_se | <i>A Function for Creating Publication Quality Item Response Theory based test information plot with standard error.</i> |
|---------------|--|

Description

This function will create publication worthy Item Response Theory based Test information plot with standard error using ggplot2 from objects created from mirt pack. Using ggplot2 will enable the user to modify the Item plot.

Usage

```
ggtestinfo_se(dataframe, model)
```

Arguments

dataframe your data.
model A mirt package fitted object.

Value

A publication quality Test information plot with standard error (dashed line). Output object is a ggplot object.

Examples

```
data <- tabledown::Rotter[, 11:31]
model <- mirt::mirt(data, model = 1, itemtype = '2PL')

plot <- ggtestinfo(data, model)
```

ggtestinfo_se_ploty *A Function for Creating Item Response Theory based test information plot with standard error with plotly.*

Description

This function will create Item Response Theory based Test information plot with standard error using ggplot2 and plotly from objects created from mirt pack. Using ggplot2 will enable the user to modify the Item plot.

Usage

```
ggtestinfo_se_ploty(dataframe, model)
```

Arguments

dataframe your data.
model A mirt package fitted object.

Value

A publication quality Test information plot with standard error (dashed line). Output object is a ggplot object.

Examples

```
data <- tabledown::Rotter[, 11:31]  
model <- mirt::mirt(data, model = 1, itemtype = '2PL')  
plot <- ggtestinfo_se_ploty(data, model)
```

gt_tab *A Function for gtExtra package friendly data summary.*

Description

This function will gtExtra package friendly data summary using the datafrmae provided psych pack.

Usage

```
gt_tab(dataframe, recode_code)
```

Arguments

dataframe Dataframe with all items.
recode_code Recode key

Value

A publication ready descriptive summary table in png format.

Examples

```
data <- tabledown::FFMQ.CFA[, c(9,10,12,14)]
recode_code <- c( "1" = "Never or very rarely true", "2" = "Rarely true",
"3"= "Sometimes true","4" = "Often true","5" = "Very often or always true")
sample_tab <- gt_tab(data,recode_code)
```

normality.loop

A Function for computing univariate normality test on data frame.

Description

This function will compute normality on entire data set. Sometime in dlookr package p values turns out to be null thus failing to test normality of the data set. This is a good alternative of dlookr function. Here normality is tested using shapiro.test from base stats.

Usage

```
normality.loop(df, bonf = TRUE, alpha = 0.05)
```

Arguments

| | |
|-------|--|
| df | A data frame. |
| bonf | If TRUE a bonferonni correction will be conducted. |
| alpha | Desired alpha. |

Value

Provides normality tests results for all columns in a wide data frame in a list format.

Examples

```
data <- tabledown::Rotter[, 11:31]
normality.loop(data)
```

 Rotter

 Validation Data of Bangla Rotter I-E Scale

Description

This is the validation data of Bangla Rotter's Internal and External Scale.

Usage

Rotter

Format

A data frame with 478 rows and 91 variables:

```

id double Id
sample character EFA or CEA
Age double Age
Gender character Gender
Educational Status character Educational Status
Education Years double COLUMN_DESCRIPTION
Income double COLUMN_DESCRIPTION
Religion double COLUMN_DESCRIPTION
Marital Status double COLUMN_DESCRIPTION
Social Stance double COLUMN_DESCRIPTION
item2 double COLUMN_DESCRIPTION
item3 double COLUMN_DESCRIPTION
item4 double COLUMN_DESCRIPTION
item5 double COLUMN_DESCRIPTION
item6 double COLUMN_DESCRIPTION
item7 double COLUMN_DESCRIPTION
item9 double COLUMN_DESCRIPTION
item10 double COLUMN_DESCRIPTION
item11 double COLUMN_DESCRIPTION
item12 double COLUMN_DESCRIPTION
item13 double COLUMN_DESCRIPTION
item15 double COLUMN_DESCRIPTION
item16 double COLUMN_DESCRIPTION
item17 double COLUMN_DESCRIPTION
item18 double COLUMN_DESCRIPTION
  
```

item20 double COLUMN_DESCRIPTION
item21 double COLUMN_DESCRIPTION
item22 double COLUMN_DESCRIPTION
item23 double COLUMN_DESCRIPTION
item25 double COLUMN_DESCRIPTION
item26 double COLUMN_DESCRIPTION
item28 double COLUMN_DESCRIPTION
item29 double COLUMN_DESCRIPTION
01 double COLUMN_DESCRIPTION
02 double COLUMN_DESCRIPTION
03 double COLUMN_DESCRIPTION
04 double COLUMN_DESCRIPTION
05 double COLUMN_DESCRIPTION
06 double COLUMN_DESCRIPTION
07 double COLUMN_DESCRIPTION
08 double COLUMN_DESCRIPTION
09 double COLUMN_DESCRIPTION
010 double COLUMN_DESCRIPTION
Total_0pennes double COLUMN_DESCRIPTION
E1 double COLUMN_DESCRIPTION
E2 double COLUMN_DESCRIPTION
E3 double COLUMN_DESCRIPTION
E4 double COLUMN_DESCRIPTION
E5 double COLUMN_DESCRIPTION
E6 double COLUMN_DESCRIPTION
E7 double COLUMN_DESCRIPTION
E8 double COLUMN_DESCRIPTION
Total_Extro double COLUMN_DESCRIPTION
N1 double COLUMN_DESCRIPTION
N2 double COLUMN_DESCRIPTION
N3 double COLUMN_DESCRIPTION
N4 double COLUMN_DESCRIPTION
N5 double COLUMN_DESCRIPTION
N6 double COLUMN_DESCRIPTION
N7 double COLUMN_DESCRIPTION
N8 double COLUMN_DESCRIPTION
Total_Neuro double COLUMN_DESCRIPTION

DIR1 double COLUMN_DESCRIPTION
DIR2 double COLUMN_DESCRIPTION
DI3 double COLUMN_DESCRIPTION
DIR4 double COLUMN_DESCRIPTION
DI5 double COLUMN_DESCRIPTION
DIR6 double COLUMN_DESCRIPTION
DI7 double COLUMN_DESCRIPTION
DIR8 double COLUMN_DESCRIPTION
DI9 double COLUMN_DESCRIPTION
DI10 double COLUMN_DESCRIPTION
DIR11 double COLUMN_DESCRIPTION
DI12 double COLUMN_DESCRIPTION
DI13 double COLUMN_DESCRIPTION
DIR14 double COLUMN_DESCRIPTION
DI15 double COLUMN_DESCRIPTION
DI16 double COLUMN_DESCRIPTION
DIR17 double COLUMN_DESCRIPTION
DI18 double COLUMN_DESCRIPTION
DIR19 double COLUMN_DESCRIPTION
DI20 double COLUMN_DESCRIPTION
DI21 double COLUMN_DESCRIPTION
DIR22 double COLUMN_DESCRIPTION
DIR23 double COLUMN_DESCRIPTION
DIR24 double COLUMN_DESCRIPTION
DI25 double COLUMN_DESCRIPTION
DIR26 double COLUMN_DESCRIPTION
DIR27 double COLUMN_DESCRIPTION
DI28 double COLUMN_DESCRIPTION
DI_Total double COLUMN_DESCRIPTION

Source

<https://github.com/masiraji/taledown/tree/main/data-raw>

| | |
|------|------------------|
| Spot | <i>Spot Data</i> |
|------|------------------|

Description

Additional demo data for GanTT

Usage

Spot

Format

A data frame with 29 rows and 3 variables:

activity character Activity

spot_type character Progress Status

spot_date character Date of Reporting Progress

Source

<https://github.com/masiraji/tabledown/tree/main/data-raw>

| | |
|-----------|---|
| tabledown | <i>Produce Publication Quality Tables and Plots</i> |
|-----------|---|

Description

The tabledown package provides necessary data frames used throughout the book and some neat functions.

tabledown Data-frames

1. Rotter: Psychometric validation data of Bangla Rotter's Internal- External Scale.
2. Gantt and Spot: Two sample data-frames for creating project management Gantt chart.
3. FFMQ.CFA: Structural Validation data of Bangla Five Factor Mindfulness Questionnaire.
4. FFMQ.Val: Correlational Validity evidences of Bangla Five Factor Mindfulness Questionnaire.

tabledown functions

This packages includes some neat and useful functions to create tables and figures suitable for journal submission:

1. `fac.tab()`: Creates a publication ready table from the output of "psych" package based factor analysis.
2. `des.tab()`: Creates a publication ready descriptive table of Item analysis with the reporting of normality assumptions.
3. `normality.loop()`: Compute normality test on the whole data frame. No grouping variable required.
4. `bedTime()`: Calculate total time spent in bed from the sleep log entry.
5. `cfa.tab()`:Creates a table with necessary fit indices from a "lavaan" class objects.
6. `cfa.tab/multi()`:creates a table with necessary fit indices from several lavaan class objects.
7. `ggicc`: Creates a ggplot2 based publication ready Item Characteristics Curve from the "mirt" package based item response theory estimations.
8. `ggiteminfo`: Creates a ggplot2 based publication ready Item Information Curve from the "mirt" package based item response theory estimations.
9. `ggtestinfo`: Creates a ggplot2 based publication ready Test Information Curve from the "mirt" package based item response theory estimations.
10. `ggtestinfo_se`: Creates a ggplot2 based publication ready Test Information Curve with standard error from the "mirt" package based item response theory estimations. It is advisable that you load **tidyverse** along with **tabledown**

Index

* datasets

FFMQ.CFA, [5](#)

FFMQ.Val, [7](#)

Gantt, [10](#)

Rotter, [17](#)

Spot, [20](#)

bedTime, [2](#)

cfa.tab, [3](#)

cfa.tab.multi, [3](#)

des.tab, [4](#)

fac.tab, [4](#)

FFMQ.CFA, [5](#)

FFMQ.Val, [7](#)

Gantt, [10](#)

ggicc, [11](#)

ggiteminfo, [11](#)

ggreliability, [12](#)

ggreliability_plotly, [13](#)

ggtestinfo, [13](#)

ggtestinfo_se, [14](#)

ggtestinfo_se_plotly, [15](#)

gt_tab, [15](#)

normality.loop, [16](#)

Rotter, [17](#)

Spot, [20](#)

tabledown, [20](#)