Package 'IPV'

December 14, 2021

Title Item Pool Visualization

Type Package

Version 0.2.0

Description Generate plots based on the Item Pool Visualization concept for latent constructs. Item Pool Visualizations are used to display the conceptual structure of a set of items (self-report or psychometric). Dantlgraber, Stieger, & Reips (2019) <doi:10.1177/2059799119884283>.

Depends R (>= 3.5.0)

Imports ggforce, ggplot2, stats, stringr, lavaan, Matrix, reshape2

Suggests knitr, rmarkdown, extrafont, readxl, cowplot, gplots

VignetteBuilder knitr

License GPL-2

LazyData true

Maintainer Nils Petras <nils.petras@mailbox.org>

RoxygenNote 7.1.2

URL https://github.com/NilsPetras/IPV

BugReports https://github.com/NilsPetras/IPV/issues

Encoding UTF-8

NeedsCompilation no

Author Nils Petras [aut, cre],
Michael Dantlgraber [aut],
Ulf-Dietrich Reips [ctb],
Matthias Bannert [ctb]

Repository CRAN

Date/Publication 2021-12-14 13:40:02 UTC

R topics documented:

`coord_facets` *Coord Facets*

Description

Generates the coordinates for a facet chart.

Usage

```
coord_facets(
  data,
  cd_method = "aggregate",
  facet_order = NULL,
  subradius = 0,
  tick = 0,
  rotate_tick_label = 0,
  rotate_radians = 0,
  rotate_degrees = 0,
  dist_test_label = 2/3,
  rotate_test_label_radians = 0,
  rotate_test_label_degrees = 0
)
```

Arguments

data	SEM estimates in the appropriate format, given by the input functions.
cd_method	character; method to summarize center distances, either "mean" or "aggregate", see details; defaults to "aggregate".
facet_order	character; vector of facet names in desired order (counter-clockwise); defaults to NULL, in which case the order is based on the correlation matrix columns in 'data'.
subradius	integer; same unit as center distances; radius of the facet circles; defaults to 0, in which case an appropriate value is estimated.
tick	numeric; axis tick position; defaults to 0, in which case an appropriate value is estimated.
rotate_tick_label	numeric; number of positions to move the tick label (counter-clockwise); defaults to 0.
rotate_radians	integer; radian angle to rotate the chart counter-clockwise by; use fractions of pi (e.g. pi/2 = 90 degrees).
rotate_degrees	integer; angle in degrees to rotate the chart counter-clockwise by.
dist_test_label	integer; position of the test label relative to the surrounding circle; defaults to 2/3, in which case the test label is displayed 2/3 of the way from the center to the surrounding circle.
rotate_test_label_radians	integer; radian angle to rotate the test label counter-clockwise by; use fractions of pi (e.g. pi/2 = 90 degrees).
rotate_test_label_degrees	integer; angle in degrees to rotate the global label counter-clockwise by.

Details

Use facet_chart to create facet charts.

Value

List containing coordinates of chart objects.

See Also

plot_facets facet_chart

---

coord_items *Coord Items*

Description

Generates the coordinates for an item chart.

Usage

```
coord_items(
  data,
  facet_order = NULL,
  rotate_radians = 0,
  rotate_degrees = 0,
  grid_limit = 0,
  dist_test_label = 0.5,
  rotate_test_label_radians = 0,
  rotate_test_label_degrees = 0,
  width_items = 1,
  length_items = 1,
  length_ratio_items = 1.5,
  dodge = 1
)
```

Arguments

data	SEM estimates in the appropriate format, given by the input functions.
facet_order	character; vector of facet names in desired order (counter-clockwise); defaults to NULL, in which case the order is based on the correlation matrix columns in 'data'.
rotate_radians	integer; radian angle to rotate the chart counter-clockwise by; use fractions of pi (e.g. pi/2 = 90 degrees).
rotate_degrees	integer; angle in degrees to rotate the chart counter-clockwise by.
grid_limit	integer; upper limit to which the grid lines should be drawn; defaults to 0, in which case an appropriate value is estimated.
dist_test_label	integer; position of the test label relative to the surrounding circle; defaults to .5, in which case the test label is displayed halfway from the center to the surrounding circle.

`rotate_test_label_radians`
: integer; radian angle to rotate the test label counter-clockwise by; use fractions of pi (e.g. pi/2 = 90 degrees).

`rotate_test_label_degrees`
: integer; angle in degrees to rotate the test label counter-clockwise by.

`width_items`
: integer; item bar width relative to default.

`length_items`
: integer; item bar length relative to default.

`length_ratio_items`
: integer; relative item bar length; defaults to 1.5.

`dodge`
: integer; horizontal outward dodge of facet labels relative to default.

Details

Use `item_chart` to create item charts.

Value

List containing coordinates of chart objects.

See Also

`plot_items` `coord_nested` `item_chart`

`coord_nested` *Coord Nested*

Description

Generates the coordinates for a nested chart and all other charts.

Usage

```
coord_nested(
  data,
  cd_method = "aggregate",
  test_order = NULL,
  facet_order = NULL,
  subradius = 0,
  tick = 0,
  rotate_tick_label = 0,
  rotate_radians = 0,
  rotate_degrees = 0,
  subrotate_radians = 0,
  subrotate_degrees = 0,
  dist_construct_label = 10,
  rotate_construct_label_radians = 0,
  rotate_construct_label_degrees = 0,
```

```
  dist_test_labels = 2/3,
  rotate_test_labels_radians = 0,
  rotate_test_labels_degrees = 0,
  prepare_item_charts = FALSE,
  correlations = TRUE,
  cor_spacing = 0,
  relative_scaling = 0,
  xarrows = NULL
)
```

Arguments

data	SEM estimates in the appropriate format, given by the input functions.
cd_method	character; method to summarize center distances, either "mean" or "aggregate", see details; defaults to "aggregate".
test_order	character; vector of test names in desired order (counter-clockwise); defaults to NULL, in which case the order is based on the correlation matrix columns in 'data'.
facet_order	character; vector of all facet names of all tests in desired order (counter-clockwise); defaults to NULL, in which case the order is based on the correlation matrix columns in 'data'.
subradius	integer; same unit as center distances; radius of the facet circles; defaults to 0, in which case an appropriate value is estimated.
tick	numeric; axis tick position; defaults to 0, in which case an appropriate value is estimated.
rotate_tick_label	numeric; number of positions to move the tick label (counter-clockwise); defaults to 0.
rotate_radians	integer; radian angle to rotate the chart counter-clockwise by; use fractions of pi (e.g. pi/2 = 90 degrees).
rotate_degrees	integer; angle in degrees to rotate the chart counter-clockwise by.
subrotate_radians	integer; radian angle or vector of radian angles to rotate the nested facet charts counter-clockwise by; use fractions of pi (e.g. pi/2 = 90 degrees).
subrotate_degrees	integer; angle in degrees or vector of angles in degrees to rotate the nested facet charts counter-clockwise by.
dist_construct_label	integer; position of the construct label relative to the surrounding circle; defaults to 10, in which case an appropriate value is estimated; a value of .5 would position the label halfway between the center and the surrounding circle.
rotate_construct_label_radians	integer; radian angle to rotate the construct label counter-clockwise by; use fractions of pi (e.g. pi/2 = 90 degrees).
rotate_construct_label_degrees	integer; angle in degrees to rotate the construct label counter-clockwise by.

`dist_test_labels`
: integer; position of the test labels relative to the surrounding circle; defaults to 2/3, in which case the test labels are displayed 2/3 of the way from the centers to the surrounding circles.

`rotate_test_labels_radians`
: integer; radian angle or vector of radian angles to rotate the test labels counter-clockwise by; use fractions of pi (e.g. pi/2 = 90 degrees).

`rotate_test_labels_degrees`
: integer; angle or vector of angle in degrees to rotate the test labels counter-clockwise by.

`prepare_item_charts`
: logical; if `TRUE`, generates the item chart coordinates for all factors by calling `coord_items`.

`correlations`
: logical; if `TRUE`, generates the coordinates for the latent correlations between tests. Sets up a ring to draw them in. If `FALSE`, the ring and the correlations are omitted, simplifying the chart significantly.

`cor_spacing`
: integer; if `correlations = TRUE`: width of the ring, the latent correlations between tests are drawn in; defaults to 0, in which case an appropriate value is estimated.

`relative_scaling`
: integer; relative size of the global chart scale compared to the nested facet chart scales; defaults to 0, in which case an appropriate value is estimated.

`xarrows`
: data frame containing information about additional correlation arrows between facets of different tests; see examples.

Details

Use `nested_chart` to create nested charts.

Value

List containing coordinates of chart objects.

See Also

`plot_nested` `nested_chart`

---

cormat *Cor(relation) Mat(rix)*

Description

Retrieve factor correlation matrix from lavaan model

Usage

```
cormat(fit)
```

Arguments

`fit`	fitted lavaan model

Value

matrix; correlation matrix

DSSEI *Domain Specific Self-Esteem Inventory*

Description

A set of center distances and latent correlations for items and facets of the DSSEI, completed by 2272 german speaking participants using the german version of the questionnaire. SEM estimation performed on the DSSEI data alone (differing from the estimation in "self_confidence").

Usage

```
DSSEI
```

Format

An object of class `list` of length 2.

Items

__ Social Competence (So) __

1. I usually feel as if I have handled myself well at social gatherings.

5. I feel secure in social situations.

9. I feel confident of my social behaviour.

13. I am often troubled with shyness. (R)

17. At social gatherings I am often withdrawn, not at all outgoing. (R)

__ Task-Related Abilities (Ab) __

2. I feel as if I lack the necessary skills to really succeed at the work I do. (R)

6. I am able to do things as well as most other people.

10. I usually expect to succeed at the things I do.

14. I almost always accomplish the goals I set for myself.

18. In general, I feel confident about my abilities.

__ Physical Appeal (Ph) __

3. I feel that others would consider me to be attractive.

7. I'm not as nice looking as most people. (R)

11. I feel confident that my physical appearance is appealing to others.

15. I am satisfied with the way I look.

19. I feel unattractive compared to most people my age. (R)

__ Public Presentation (Pb) __

4. When I speak in a large group discussion, I usually feel sure of myself.

8. I enjoy being in front of large audiences.

12. I feel quite confident when speaking before a group of my peers.

16. I find it very hard to talk in front of a group. (R)

20. When I talk in front of a group of people my own age, I am usually somewhat worried or afraid. (R)

(Hoyle, R. H. (1991). Evaluating measurement models in clinical research: Covariance structure analysis of latent variable models of self-conception. Journal of Consulting and Clinical Psychology, 59(1), 67.)

Source

Dantlgraber, M., Stieger, S., & Reips, U. D. (2019). Introducing Item Pool Visualization: A method for investigation of concepts in self-reports and psychometric tests. Methodological Innovations, 12(3), 2059799119884283.

`facet_chart` *Facet Chart*

Description

Creates a facet chart, showing the facets of a test.

Usage

```
facet_chart(
  data,
  cd_method = "aggregate",
  facet_order = NULL,
  subradius = 0,
  file_name = "none",
  size = 1,
  font = "sans",
  rotate_radians = 0,
  rotate_degrees = 0,
  file_width = 10,
  file_height = 10,
  zoom_x = NULL,
  zoom_y = NULL,
  dpi = 500,
  color = "#007AD6",
  fade = 85,
```

```
  tick = 0,
  rotate_tick_label = 0,
  cor_labels = TRUE,
  dist_test_label = 2/3,
  rotate_test_label_radians = 0,
  rotate_test_label_degrees = 0,
  title = NULL,
  size_title = 1,
  size_cor_labels = 1,
  size_test_label = 1,
  size_facet_labels = 1,
  width_axes = 1,
  width_circles = 1,
  width_tick = 1,
  size_tick_label = 1
)
```

Arguments

data	SEM estimates in the appropriate format, given by the input functions.
cd_method	character; method to summarize center distances, either "mean" or "aggregate", see details; defaults to "aggregate".
facet_order	character; vector of facet names in desired order (counter-clockwise); defaults to NULL, in which case the order is based on the correlation matrix columns in 'data'.
subradius	integer; same unit as center distances; radius of the facet circles; defaults to 0, in which case an appropriate value is estimated.
file_name	character; name of the file to save. Supported formats are: "pdf" (highest quality and smallest file size), "png", "jpeg"; defaults to "none".
size	integer; changes the size of most chart objects simultaneously.
font	character; text font, use extrafonts to access additional fonts; defaults to "sans", which is "Helvetica".
rotate_radians	integer; radian angle to rotate the chart counter-clockwise by; use fractions of pi (e.g. pi/2 = 90 degrees).
rotate_degrees	integer; angle in degrees to rotate the chart counter-clockwise by.
file_width	integer; file width in inches; defaults to 10.
file_height	integer; file height in inches; defaults to 10.
zoom_x	integer; vector with two values, the edges of the zoomed section on the x-axis; defaults to NULL.
zoom_y	integer; vector with two values, the edges of the zoomed section on the y-axis; defaults to NULL.
dpi	integer; resolution in dots per inch for "png" and "jpeg" files; defaults to 500.
color	accent color; defaults to blue ("#007AD6").
fade	integer; brightness of the gray tones between 0 = "black" and 100 = "white" in steps of 1; defaults to 85.

`tick` numeric; axis tick position; defaults to 0, in which case an appropriate value is estimated.

`rotate_tick_label` numeric; number of positions to move the tick label (counter-clockwise); defaults to 0.

`cor_labels` logical; if `TRUE`, shows latent correlations between facets; defaults to `TRUE`.

`dist_test_label` integer; position of the test label relative to the surrounding circle; defaults to 2/3, in which case the test label is displayed 2/3 of the way from the center to the surrounding circle.

`rotate_test_label_radians` integer; radian angle to rotate the test label counter-clockwise by; use fractions of pi (e.g. pi/2 = 90 degrees).

`rotate_test_label_degrees` integer; angle in degrees to rotate the global label counter-clockwise by.

`title` character; overall chart title; defaults to NULL.

`size_title` integer; title font size relative to default.

`size_cor_labels` integer; correlation font size relative to default.

`size_test_label` integer; test font size relative to default.

`size_facet_labels` integer; facet font size relative to default.

`width_axes` integer; radial axis width relative to default.

`width_circles` integer; facet circle outline width relative to default.

`width_tick` integer; axis tick line width relative to default.

`size_tick_label` integer; axis tick font size relative to default.

Details

To summarize center distances (`cd_method`), the "mean" method computes the average center distance (compute cds first, summarize across items second), while the "aggregate" method computes a center distance based on the sum of the squared loadings (summarize across items first, compute cds second). "Aggregate" (default) is recommended, because it is more meaningful in cases with heterogeneous factor loadings, while "mean" is the originally proposed method.

Pdf files will be vector based and can be scaled arbitrarily. For other formats use `file_width`, `file_height`, and `dpi` to avoid later rescaling and loss of quality.

Instead of using screenshots to crop the chart, it is highly recommendable to use `zoom_x` and `zoom_y`. This allows for vector-based graphics quality when showing sections of the chart. With this cropping method, use `file_width` to set the overall size of the file output, `file_height` will automatically adjust to retain the correct aspect ratio, if both `zoom_x` and `zoom_y` are provided.

Consider adding title and caption in your typesetting software (LaTeX, MS Word, ...), not here. The option to add a title is only a quick and dirty shurtcut. It reduces chart size and is inflexible. Adding the title manually will provide additional options, but requires you to save to a file manually. To manually add a title or caption use `labs`.

Value

Object of the class "ggplot".

See Also

item_chart nested_chart

Examples

```
# as simple as that:
facet_chart(SMTQ)
```

---

floads *Load(ing)s*

Description

Extract the standardized factor loadings from a fitted lavaan model.

Usage

```
floads(fit, vars = NULL)
```

Arguments

fit	fitted lavaan model
vars	character; variables for which loadings should be extracted; defaults to NULL, in which case all variables are considered

Value

numeric; vector of standardized factor loadings

get_names *Get names*

Description

Extract the names of tests, facets, and items from the variable names of a dataset.

Usage

```
get_names(dat)
```

Arguments

dat	data frame; dataset

Details

variable names in the data have to strictly match the following scheme: "test_facet_item" or "facet_item".

Value

data frame; names of tests, facets and items

get_xarrows *Get Xarrows*

Description

Creates a data frame for the drawing of arrows in nested charts, including all correlations between facets that exceed the correlation of the respective tests.

Usage

```
get_xarrows(cors, design)
```

Arguments

cors	list; list of latent correlation matrices of each model
design	data frame; each facet (column "facet") is matched with its superordinate test (column "test")

Value

data frame; data frame in the required format for the drawing of arrows in nested charts, including only those latent facet correlations, that exceed the correlation between the respective tests.

HEXACO *IPIP HEXACO Equivalent Scales*

Description

Cleaned-up data from an ad-hoc online sample of n = 22786 participants on the 240 items of the IPIP HEXACO Equivalent Scales. Data were collected before 21st June 2014 within the Open-Source Psychometrics Project (https://openpsychometrics.org/). After listwise deletion of missing values and including only those participants who did at least "agree" on the items "I understand the instructions for this test." and "I have answered all of these questions as accurately as possible.", data on n = 20174 participants remains.

Usage

```
HEXACO
```

Format

An object of class `data.frame` with 20174 rows and 245 columns.

Items

https://ipip.ori.org/newhexaco_pi_key.htm

Source

:

https://openpsychometrics.org/_rawdata/ (May 6th, 2020)

ind_lav *ind lav*

Description

create a lavaan model syntax based on a set of variable names and indicator names that comprise these variable names

Usage

```
ind_lav(vars, indicators)
```

Arguments

`vars`	character; variable names
`indicators`	character; indicator names, may include unused indicators

Details

Indicator names have to include the variable names like this: "...variable_...". Variable names have to be unique and cannot be contained in one another like this: "variable_" and "ariable_"

Value

character; lavaan model syntax

`input_excel` *Input Excel*

Description

Reads excel files containing factor loadings and latent correlations for IPV charts.

Usage

```
input_excel(global = NULL, tests)
```

Arguments

`global`	character; name of the excel file containing factor loadings from the global level and the test level, and latent correlations from the test level.
`tests`	character; name(s) of the excel file(s) containing factor loadings from the test level and the facet level, and latent correlations from the facet level.

Details

Note that the excel files need a very specific structure. Use the example files as templates.

The `global` argument defaults to NULL. This allows to only use the `tests` argument, resulting in a simple model with one test and its facets.

If you specify an element in `tests` as `NA`, this test will be treated as having no facets.

Value

List containing formatted data including center distances for `item_chart`, `facet_chart`, and `nested_chart`.

Examples

```
# read data for a simple model by ignoring the "global" parameter of
# input_excel
single_file <- system.file(
  "extdata",
  "DSSEI.xlsx",
  package = "IPV",
  mustWork = TRUE)
x <- input_excel(tests = single_file)
```

```
# read data for a nested model
# the estimates need to be split into several excel files as in the example
global <- system.file(
  "extdata",
  "IPV_global.xlsx",
  package = "IPV",
  mustWork = TRUE)
tests <- c(
  system.file(
    "extdata",
    "IPV_DSSEI.xlsx",
    package = "IPV",
    mustWork = TRUE),
  system.file(
    "extdata",
    "IPV_SMTQ.xlsx",
    package = "IPV",
    mustWork = TRUE),
  system.file(
    "extdata",
    "IPV_RSES.xlsx",
    package = "IPV",
    mustWork = TRUE))
x <- input_excel(global = global, tests = tests)
```

`input_excel_factor` *Input Excel Factor*

Description

Reads factor loadings and latent correlations from an excel file.

Usage

```
input_excel_factor(file)
```

Arguments

`file`	character; filename of the excel file

Details

Helper function of `input_excel`.

Value

list containing formatted data including center distances for `item_chart`, `facet_chart`

See Also

input_excel

`input_manual_nested` *Input Manual Nested*

Description

Generates manual data input for a nested model with several tests.

Usage

```
input_manual_nested(
  construct_name,
  test_names,
  items_per_test,
  item_names,
  construct_loadings,
  test_loadings,
  correlation_matrix
)
```

Arguments

`construct_name`	character; the name of the overall construct.
`test_names`	character; the names of the tests in correct order.
`items_per_test`	integer; number of items per test in correct order (determined by test_names), if all tests have the same number of items a single number can be used, e.g. 10 instead of c(10, 10, 10).
`item_names`	character or integer; the names of the items in correct order (determined by test_names).
`construct_loadings`	integer; vector of the factor loadings from the single factor model of the construct in correct order (determined by item_names).
`test_loadings`	integer; vector of the factor loadings on the test factors from the group factor model in correct order (determined by item_names).
`correlation_matrix`	matrix containing the latent correlations between tests, pay attention to the order of rows and columns, which is determined by test_names.

Details

Pay attention to the order of tests and items, it has to be coherent throughout the whole data. test_names and items_per_test determine which test is listed first and how many items are listed for that test. item_names, construct_loadings and test_loadings have to match that order. The correlation matrix uses the order in test_names for rows and columns.

This function only lists the name of the tests in output$tests. For each of those tests, the data on the facets needs to be added using `input_manual_simple`. Every test for which you do not provide this data will be treated as having no facets.

Visually inspect the returned object before continuing with `input_manual_process`!

Value

list containing "raw" data. The data on the facets of the tests needs to be added using `input_manual_simple`. Afterwards, the whole data needs to be pre-processed using `input_manual_process`.

See Also

`input_manual_simple` `input_manual_process`

Examples

```
# these data can also be seen in self_confidence, the example data of
# this package
mydata <- input_manual_nested(
construct_name = "Self-Confidence",
test_names = c("DSSEI", "SMTQ", "RSES"),
items_per_test = c(20, 14, 10),
item_names = c(
 1,  5,  9, 13, 17, # DSSEI
 3,  7, 11, 15, 19, # DSSEI
 16,  4, 12,  8, 20, # DSSEI
 2,  6, 10, 14, 18, # DSSEI
 11, 13, 14,  1,  5,  6, # SMTQ
 3, 10, 12,  8, # SMTQ
 7,  2,  4,  9, # SMTQ
 1,  3,  4,  7, 10, # RSES
 2,  5,  6,  8,  9), # RSES
construct_loadings = c(
 .5189, .6055, .618,  .4074, .4442,
 .5203, .2479, .529,  .554,  .5144,
 .3958, .5671, .5559, .4591, .4927,
 .3713, .5941, .4903, .5998, .6616,
 .4182, .2504, .4094, .3977, .5177, .4603,
 .3271,  .261, .3614, .4226,
 .2076, .3375, .5509, .3495,
 .5482, .4627, .4185, .4185, .5319,
 .4548, .4773, .4604, .4657, .4986),
test_loadings = c(
 .5694, .6794, .6615, .4142, .4584, # DSSEI
 .5554, .2165, .5675, .5649, .4752, # DSSEI
```

```
 .443 , .6517, .6421, .545 , .5266, # DSSEI
 .302 , .6067, .5178, .5878, .6572, # DSSEI
 .4486, .3282, .4738, .4567, .5986, .5416, # SMTQ
 .3602, .2955, .3648, .4814, # SMTQ
 .2593, .4053, .61  , .4121, # SMTQ
 .6005, .4932, .4476, .5033, .6431, # RSES
 .5806, .5907, .6179, .5899, .6559), # RSES
correlation_matrix = matrix(data = c(  1, .73, .62,
                                     .73,   1, .75,
                                     .62, .75,   1),
                            nrow = 3,
                            ncol = 3))
mydata
```

`input_manual_process` *Input Manual Process*

Description

Pre-processes the SEM estimates listed using `input_manual_simple` or `input_manual_nested` for the use of chart functions.

Usage

```
input_manual_process(data)
```

Arguments

`data`	list generated by `input_manual_simple` or `input_manual_nested` with complete data.

Value

List containing formatted data including center distances for `item_chart`, `facet_chart`, and `nested_chart`.

See Also

`input_manual_simple` `input_manual_nested`

Examples

```
# these RSES data can also be seen in self_confidence, the example data of
# this package
mydata <- input_manual_simple(
test_name = "RSES",
facet_names = c("Ns", "Ps"),
items_per_facet = 5,
item_names = c(2, 5, 6, 8, 9,
               1, 3, 4, 7, 10),
```

```
test_loadings = c(.5806, .5907, .6179, .5899, .6559,
                  .6005, .4932, .4476, .5033, .6431),
facet_loadings = c(.6484, .6011, .6988, .6426, .6914,
                   .6422, .5835, .536, .5836, .6791),
correlation_matrix = matrix(data = c(1, .69,
                                     .69, 1),
                            nrow = 2,
                            ncol = 2))
mydata
input_manual_process(mydata)
```

---

input_manual_process_factor *Input Manual Process Factor*

Description

Helper function of input_manual_process.

Usage

```
input_manual_process_factor(data)
```

Arguments

data	list generated by input_manual_simple with complete data.

Value

List containing formatted data including center distances for a single factor.

---

input_manual_simple *Input Manual Simple*

Description

Generates manual data input for a simple model with one test.

Usage

```
input_manual_simple(
  test_name,
  facet_names,
  items_per_facet,
  item_names,
  test_loadings,
  facet_loadings,
  correlation_matrix
)
```

Arguments

test_name	character; the name of the test.
facet_names	character; the names of the facets in correct order.
items_per_facet	integer; number of items per facet in correct order (determined by facet_names), if all facets have the same number of items a single number can be used, e.g. 5 instead of c(5, 5, 5, 5).
item_names	character or integer; the names of the items in correct order (determined by facet_names).
test_loadings	integer; vector of the factor loadings from the single factor model of the test or a group factor model of multiple tests in correct order (determined by item_names).
facet_loadings	integer; vector of the factor loadings on the facet factors from the group factor model in correct order (determined by item_names).
correlation_matrix	matrix containing the latent correlations between facets, pay attention to the order of rows and columns, which is determined by facet_names.

Details

Pay attention to the order of facets and items, it has to be coherent throughout the whole data. facet_names and items_per_facet determine which facet is listed first and how many items there are listed for that facet. item_names, test_loadings and facet_loadings have to match that order. The correlation matrix uses the order in facet_names for rows and columns.

Visually inspect the returned object before continuing with `input_manual_process`!

Value

list containing "raw" data, that needs to be pre-processed using `input_manual_process`.

See Also

`input_manual_nested` `input_manual_process`

Examples

```
# these RSES data can also be seen in self_confidence, the example data of
# this package
mydata <- input_manual_simple(
test_name = "RSES",
facet_names = c("Ns", "Ps"),
items_per_facet = 5,
item_names = c(2, 5, 6, 8, 9,
               1, 3, 4, 7, 10),
test_loadings = c(.5806, .5907, .6179, .5899, .6559,
                     .6005, .4932, .4476, .5033, .6431),
facet_loadings = c(.6484, .6011, .6988, .6426, .6914,
                       .6422, .5835, .536, .5836, .6791),
correlation_matrix = matrix(data = c(1, .69,
                                     .69, 1),
                            nrow = 2,
                            ncol = 2))
mydata
input_manual_process(mydata)
```

IPV *IPV: A package to create Item Pool Visualizations*

Description

The IPV package provides the following functions.

Estimation function

`ipv_est` uses raw data to estimate the IPV models and pre-format their estimates for chart creation. This is the easiest and recommended Workflow.

Chart Functions

Chart functions create a ggplot2 object (the chart). There are three types of charts. `item_chart` `facet_chart` `nested_chart`

Input Functions

The input functions prepare existing model estimates for the chart functions. This is not recommended, if the raw data are available. Read in vectors containing model estimates from within R by using `input_manual_simple`, `input_manual_nested`) and `input_manual_process`. Read in model estimates via MS Excel files and `input_excel`.

Miscellaneous functions

The function `item_overview` creates a grid of bar plots showing the (squared) factor loadings of all items in all models underlying a nested chart. Use this to inspect the absolute values underlying the charts.

The function `rename` enables quick changes of the labels for variables.

Basic Workflow

1. Prepare your raw data.
2. Generate the model estimates using the estimation function.
3. Select a chart function and use it with the estimates, a file name (.pdf), and otherwise default values.
4. Change the default values of the chart function arguments.
5. Check the chart's appearance by opening the created file (do not rely on the display of plots in R, results may differ).
6. Repeat until you are satisfied with the result.

`ipv_est` *IPV estimation*

Description

IPV estimation

Usage

```
ipv_est(
  dat,
  name,
  estimator = "ML",
  include_raw = TRUE,
  include_lav = TRUE,
  include_xarrow = TRUE
)
```

Arguments

`dat`	data frame; raw data (see details)
`name`	character; name of the overall construct or test that comprises all items used
`estimator`	character; estimator used by lavaan; defaults to "ML" (Maximum Likelihood)
`include_raw`	logical; should raw estimates of factor loadings be included in the output?; defaults to TRUE
`include_lav`	logical; should lavaan objects of the fitted models be included in the output?; defaults to TRUE
`include_xarrow`	logical; should an object for the drawing of arrows in nested plots be returned?; defaults to TRUE

Details

the data given to dat have to conform to the following rules: * no additional variables / columns * variables are named according to the following pattern: "test_facet_item". * If there is only one test in the data, the pattern is "facet_item". For tests without facets in a larger dataset also comprising tests with items, the pattern is "test_item". * Variable names have to be unique. Item names have to be unique at the level of the test (not only at the level of the facet) See example

Value

list; $est includes the center distances and all necessary input for the IPV chart functions, $est_raw includes the factor loadings and latent correlations, $lav includes the fitted models (class: lavaan), $xarrow includes a data frame for arrows between facets in nested charts, that can be passed on directly to nested_chart; by default, all three of these elements are provided.

$xarrow includes only those cases, where the estimate of the latent correlation between facets exceeds the estimate of the latent correlation between their respective tests, as recommended by the original authors.

Examples

```
# an IPV that comprises the honesty/humility and the agreeableness factor of
# the HEXACO (reduced to first 4 items per facet and first 1000 observations
# to reduce runtime)
res <- ipv_est(
  HEXACO[1:500, grep("^H_.*[1-4]$|^A_.*[1-4]$", names(HEXACO))],
  "HA")
nested_chart(res$est)
```

---

item_chart *Item Chart*

Description

Creates an item chart, showing the items of a test arranged by facets.

Usage

```
item_chart(
  data,
  facet_order = NULL,
  file_name = "none",
  size = 1,
  font = "sans",
  rotate_radians = 0,
  rotate_degrees = 0,
  grid_limit = 0,
  file_width = 12,
  file_height = 10,
```

```
  zoom_x = NULL,
  zoom_y = NULL,
  dpi = 500,
  color = "black",
  color2 = "black",
  fade_axes = 50,
  fade_grid_major = 15,
  fade_grid_minor = 65,
  dodge = 1,
  dist_test_label = 0.5,
  rotate_test_label_radians = 0,
  rotate_test_label_degrees = 0,
  width_items = 1,
  length_items = 1,
  length_ratio_items = 1.5,
  title = NULL,
  size_title = 1,
  size_tick_label = 1,
  size_test_label = 1,
  size_facet_labels = 1,
  width_axes = 1,
  size_arrow_heads = 1,
  width_grid = 1
)
```

Arguments

data	SEM estimates in the appropriate format, given by the input functions.
facet_order	character; vector of facet names in desired order (counter-clockwise); defaults to NULL, in which case the order is based on the correlation matrix columns in 'data'.
file_name	character; name of the file to save. Supported formats are: "pdf" (highest quality and smallest file size), "png", "jpeg"; defaults to "none".
size	integer; changes the size of most chart objects simultaneously.
font	character; text font, use extrafonts to access additional fonts; defaults to "sans", which is "Helvetica".
rotate_radians	integer; radian angle to rotate the chart counter-clockwise by; use fractions of pi (e.g. pi/2 = 90 degrees).
rotate_degrees	integer; angle in degrees to rotate the chart counter-clockwise by.
grid_limit	integer; upper limit to which the grid lines should be drawn; defaults to 0, in which case an appropriate value is estimated.
file_width	integer; file width in inches; defaults to 12.
file_height	integer; file height in inches; defaults to 10.
zoom_x	integer; vector with two values, the edges of the zoomed section on the x-axis; defaults to NULL.

`zoom_y` integer; vector with two values, the edges of the zoomed section on the y-axis; defaults to NULL.

`dpi` integer; resolution in dots per inch for "png" and "jpeg" files; defaults to 500.

`color` first accent color; defaults to "black".

`color2` second accent color; defaults to "black".

`fade_axes` integer; brightness of the gray tone of the axes between 0 = "black" and 100 = "white" in steps of 1; defaults to 50.

`fade_grid_major` integer; brightness of the gray tone of the major grid lines between 0 = "black" and 100 = "white" in steps of 1; defaults to 15.

`fade_grid_minor` integer; brightness of the gray tone of the minor grid lines between 0 = "black" and 100 = "white" in steps of 1; defaults to 65.

`dodge` integer; horizontal outward dodge of facet labels relative to default.

`dist_test_label` integer; position of the test label relative to the surrounding circle; defaults to .5, in which case the test label is displayed halfway from the center to the surrounding circle.

`rotate_test_label_radians` integer; radian angle to rotate the test label counter-clockwise by; use fractions of pi (e.g. pi/2 = 90 degrees).

`rotate_test_label_degrees` integer; angle in degrees to rotate the test label counter-clockwise by.

`width_items` integer; item bar width relative to default.

`length_items` integer; item bar length relative to default.

`length_ratio_items` integer; relative item bar length; defaults to 1.5.

`title` character; overall chart title; defaults to NULL.

`size_title` integer; title font size relative to default.

`size_tick_label` integer; axis tick label font size relative to default.

`size_test_label` integer; test label font size relative to default.

`size_facet_labels` integer; facet label font size relative to default.

`width_axes` integer; radial axis width relative to default.

`size_arrow_heads` integer; arrow head size relative to default.

`width_grid` integer; grid line width relative to default.

Details

When changing the size of objects, consider the `size` parameter first and make specific adjustments with the other `size_` and `width_` parameters after.

To better display overlapping item values, change the width of the item bars, or set the accent colors to different values, or change the ratio of item lengths.

Pdf files will be vector based and can be scaled arbitrarily. For other formats use `file_width`, `file_height`, and `dpi` to avoid later rescaling and loss of quality.

Instead of using screenshots to crop the chart, it is highly recommendable to use `zoom_x` and `zoom_y`. This allows for vector-based graphics quality when showing sections of the chart. With this cropping method, use `file_width` to set the overall size of the file output, `file_height` will automatically adjust to retain the correct aspect ratio, if both `zoom_x` and `zoom_y` are provided.

Consider adding title and caption in your typesetting software (LaTeX, MS Word, ...), not here. The option to add a title is only a quick and dirty shurtcut. It reduces chart size and is inflexible. Adding the title manually will provide additional options, but requires you to save to a file manually. To manually add a title or caption use `labs`.

Using a `grid_limit` higher than the default will re-scale the whole chart, while a value below the default will only remove grid lines.

Value

Object of the class "ggplot" and, by default, the same object saved as a file.

See Also

`facet_chart` `nested_chart`

Examples

```
# as simple as that
item_chart(SMTQ)
```

`item_overview` *Item Overview*

Description

Shows all (squared) factor loadings of all items in all models in a plot grid of bar plots.

Usage

```
item_overview(
  data,
  squared = TRUE,
  file_name = "none",
  dpi = 500,
```

```
  color = NULL,
  font = "mono"
)
```

Arguments

data	raw SEM estimates in the appropriate format, given by the input functions.
squared	logical; should factor loadings be squared?; defaults to TRUE
file_name	character; name of the file to save. Supported formats are: "pdf" (highest quality and smallest file size), "png", "jpeg"; defaults to "none".
dpi	integer; resolution in dots per inch for "png" and "jpeg" files; defaults to 500.
color	character; vector of hex codes for colors; defaults to the colors "#DAD8D8" (gray), "#11C1FF" (light blue), and "#007AD6" (blue)
font	character; font of the plot labels; defaults to "mono"

Details

File output produces much more reliable results than display within R. Display within R may scatter elements of the chart and distort the overall appearance.

Value

gg / ggplot object; plot grid with one bar plot per item showing (squared) factor loadings of that item in all IPV models, arranged by facets and tests

Examples

```
# Honesty/Humility and Agreeableness items
# the use of file output is recommended
# to prevent irregular placement of plot labels
res <- ipv_est(
  HEXACO[1:1000,grep("^H_.*[1-4]$|^A_.*[1-4]$", names(HEXACO))],
  "HA")
# reduced to first 4 items per facet and first 1000 observations to reduce
# runtime
item_overview(res$est_raw) # file output is recommended (see details)
```

---

nested_chart *Nested Chart*

Description

Creates a nested chart, showing several tests and their facets.

Usage

```
nested_chart(
  data,
  cd_method = "aggregate",
  test_order = NULL,
  facet_order = NULL,
  xarrows = NULL,
  subradius = 0,
  file_name = "none",
  size = 1,
  relative_scaling = 0,
  font = "sans",
  rotate_radians = 0,
  rotate_degrees = 0,
  subrotate_radians = 0,
  subrotate_degrees = 0,
  file_width = 10,
  file_height = 10,
  zoom_x = NULL,
  zoom_y = NULL,
  dpi = 500,
  color_global = "#11C1FF",
  color_nested = "#007AD6",
  fade = 85,
  cor_spacing = 0,
  tick = 0,
  rotate_tick_label = 0,
  dist_construct_label = 10,
  rotate_construct_label_radians = 0,
  rotate_construct_label_degrees = 0,
  dist_test_labels = 2/3,
  rotate_test_labels_radians = 0,
  rotate_test_labels_degrees = 0,
  cor_labels_tests = TRUE,
  cor_labels_facets = TRUE,
  title = NULL,
  size_title = 1,
  size_construct_label = 1,
  size_test_labels = 1,
  size_facet_labels = 1,
  width_axes = 1,
  width_axes_inner = 1,
  width_circles = 1,
  width_circles_inner = 1,
  width_tick = 1,
  width_tick_inner = 1,
  size_tick_label = 1,
  size_cor_labels = 1,
```

```
  size_cor_labels_inner = 1,
  width_xarrows = 1,
  size_xarrow_heads = 1,
  size_xarrow_labels = 1
)
```

Arguments

data	SEM estimates in the appropriate format, given by the input functions.
cd_method	character; method to summarize center distances, either "mean" or "aggregate", see details; defaults to "aggregate".
test_order	character; vector of test names in desired order (counter-clockwise); defaults to NULL, in which case the order is based on the correlation matrix columns in 'data'.
facet_order	character; vector of all facet names of all tests in desired order (counter-clockwise); defaults to NULL, in which case the order is based on the correlation matrix columns in 'data'.
xarrows	data frame containing information about additional correlation arrows between facets of different tests; see examples.
subradius	integer; same unit as center distances; radius of the facet circles; defaults to 0, in which case an appropriate value is estimated.
file_name	character; name of the file to save. Supported formats are: "pdf" (highest quality and smallest file size), "png", "jpeg"; defaults to "none".
size	integer; changes the size of most chart objects simultaneously.
relative_scaling	integer; relative size of the global chart scale compared to the nested facet chart scales; defaults to 0, in which case an appropriate value is estimated.
font	character; text font, use extrafonts to access additional fonts; defaults to "sans", which is "Helvetica".
rotate_radians	integer; radian angle to rotate the chart counter-clockwise by; use fractions of pi (e.g. pi/2 = 90 degrees).
rotate_degrees	integer; angle in degrees to rotate the chart counter-clockwise by.
subrotate_radians	integer; radian angle or vector of radian angles to rotate the nested facet charts counter-clockwise by; use fractions of pi (e.g. pi/2 = 90 degrees).
subrotate_degrees	integer; angle or vector of angles in degrees to rotate the nested facet charts counter-clockwise by.
file_width	integer; file width in inches; defaults to 10.
file_height	integer; file height in inches; defaults to 10.
zoom_x	integer; vector with two values, the edges of the zoomed section on the x-axis; defaults to NULL.
zoom_y	integer; vector with two values, the edges of the zoomed section on the y-axis; defaults to NULL.

`dpi` integer; resolution in dots per inch for "png" and "jpeg" files; defaults to 500.

`color_global` global accent color; defaults to light blue ("#11C1FF").

`color_nested` nested accent color; defaults to blue ("#007AD6").

`fade` integer; brightness of the gray tones between 0 (black) and 100 (white) in steps of 1; defaults to 85.

`cor_spacing` integer; if `correlations = TRUE`: width of the ring, the correlations between tests are drawn in; defaults to 0, in which case an appropriate value is estimated.

`tick` numeric; axis tick position; defaults to 0, in which case an appropriate value is estimated.

`rotate_tick_label` numeric; number of positions to move the tick label (counter-clockwise); defaults to 0.

`dist_construct_label` integer; position of the construct label relative to the surrounding circle; defaults to 10, in which case an appropriate value is estimated; a value of .5 would position the label halfway between the center and the surrounding circle.

`rotate_construct_label_radians` integer; radian angle to rotate the construct label counter-clockwise by; use fractions of pi (e.g. pi/2 = 90 degrees).

`rotate_construct_label_degrees` integer; angle in degrees to rotate the construct label counter-clockwise by.

`dist_test_labels` integer; position of the test labels relative to the surrounding circle; defaults to 2/3, in which case the test labels are displayed 2/3 of the way from the centers to the surrounding circles.

`rotate_test_labels_radians` integer; radian angle or vector of radian angles to rotate the test labels counter-clockwise by; use fractions of pi (e.g. pi/2 = 90 degrees).

`rotate_test_labels_degrees` integer; angle or vector of angle in degrees to rotate the test labels counter-clockwise by.

`cor_labels_tests` logical; if `TRUE`, shows the correlations between tests as text.

`cor_labels_facets` logical; if `TRUE`, shows the correlations between facets as text.

`title` character; overall chart title; defaults to NULL.

`size_title` integer; title font size relative to default.

`size_construct_label` integer; construct label font size relative to default.

`size_test_labels` integer; test label font size relative to default.

`size_facet_labels` integer; facet label font size relative to default.

`width_axes` integer; global radial axis width relative to default.

`width_axes_inner`
: integer; nested radial axis width relative to default.

`width_circles`
: integer; global circle outline width relative to default.

`width_circles_inner`
: integer; nested circle outline width relative to default.

`width_tick`
: integer; global axis tick line width relative to default.

`width_tick_inner`
: integer; nested axis tick line width relative to default.

`size_tick_label`
: integer; axis tick label font size relative to default.

`size_cor_labels`
: integer; font size of the correlations between tests relative to default.

`size_cor_labels_inner`
: integer; font size of the correlations between facets relative to default.

`width_xarrows`
: integer; extra arrow line width relative to default.

`size_xarrow_heads`
: integer; extra arrow head length relative to default.

`size_xarrow_labels`
: integer; font size of the correlations indicated by extra arrows relative to default.

Details

To summarize center distances (`cd_method`), the "mean" method computes the average center distance (compute cds first, summarize across items second), while the "aggregate" method computes a center distance based on the sum of the squared loadings (summarize across items first, compute cds second). "Aggregate" (default) is recommended, because it is more meaningful in cases with heterogeneous factor loadings, while "mean" is the originally proposed method.

To get tidy results, it is often required to use `rotate_` and `subrotate_` for better alignment.

If you set `subrotate_` to a single value, all nested facet charts will be rotated by the same amount. If you use a vector of values, the nested facet charts will be rotated one by one by the values from that vector.

Increase `relative_scaling` to avoid circle overlap. Decrease it to make small chart objects more visible.

`correlations` and `cor_spacing` add larger circles around the nested facet charts, but do not change these facet charts.

When changing the size of objects, consider the `size` parameter first and make specific adjustments with the other `size_` and `width_` parameters after.

Pdf files will be vector based and can be scaled arbitrarily. For other formats use `file_width`, `file_height`, and `dpi` to avoid later rescaling and loss of quality.

Instead of using screenshots to crop the chart, it is highly recommendable to use `zoom_x` and `zoom_y`. This allows for vector-based graphics quality when showing sections of the chart. With this cropping method, use `file_width` to set the overall size of the file output, `file_height` will automatically adjust to retain the correct aspect ratio, if both `zoom_x` and `zoom_y` are provided.

If facet1 or facet2 is NA for a given xarrow, the arrow will end on the test's circle. Note: this correlation is usually not part of the model.

Consider adding title and caption in your typesetting software (LaTeX, MS Word, ...), not here. The option to add a title is only a quick and dirty shurtcut. It reduces chart size and is inflexible. Adding the title manually will provide additional options, but requires you to save to a file manually. To manually add a title or caption use labs.

Value

Object of the class "ggplot".

See Also

item_chart facet_chart

Examples

```
# as simple as that
nested_chart(self_confidence, subradius = .6)

# adding xarrows, in this example for all cases where the correlation between
# facets exceeds the correlation between their respective tests.
x <- data.frame(
  test1 = rep(NA, 3),
  facet1 = NA,
  test2 = NA,
  facet2 = NA,
  value = NA)
x[1, ] <- c("DSSEI", "Ab", "RSES", "Ps", ".67")
x[2, ] <- c("DSSEI", "Ab", "SMTQ", "Cs", ".81")
x[3, ] <- c("SMTQ", "Ct", "RSES", "Ns", ".76")
nested_chart(self_confidence,
             subradius = .6,
             xarrows = x)

# rotating the nested facet charts one by one
nested_chart(self_confidence,
             subradius = .6,
             subrotate_radians = c(0, pi / 2, 0))

# test without facets

global <- system.file(
  "extdata", "IPV_global.xlsx", package = "IPV", mustWork = TRUE)
tests <- c(
  system.file("extdata", "IPV_DSSEI.xlsx", package = "IPV", mustWork = TRUE),
  system.file("extdata", "IPV_SMTQ.xlsx", package = "IPV", mustWork = TRUE),
  NA)
x <- input_excel(global = global, tests = tests)
nested_chart(x)
```

plot_facets *Plot Facets*

Description

Generates a facet chart from coordinates.

Usage

```
plot_facets(
  coord,
  title = NULL,
  size = 1,
  file_name = "none",
  file_width = 10,
  file_height = 10,
  zoom_x = NULL,
  zoom_y = NULL,
  dpi = 500,
  color = "black",
  fade = 85,
  font = "sans",
  cor_labels = TRUE,
  size_title = 1,
  size_cor_labels = 1,
  size_test_label = 1,
  size_facet_labels = 1,
  width_axes = 1,
  width_circles = 1,
  width_tick = 1,
  size_tick_label = 1
)
```

Arguments

coord	list generated by coord_facets or coord_nested.
title	character; overall chart title; defaults to NULL.
size	integer; changes the size of most chart objects simultaneously.
file_name	character; name of the file to save. Supported formats are: "pdf" (highest quality and smallest file size), "png", "jpeg"; defaults to "none".
file_width	integer; file width in inches; defaults to 10.
file_height	integer; file height in inches; defaults to 10.
zoom_x	integer; vector with two values, the edges of the zoomed section on the x-axis; defaults to NULL.

zoom_y	integer; vector with two values, the edges of the zoomed section on the y-axis; defaults to NULL.
dpi	integer; resolution in dots per inch for "png" and "jpeg" files; defaults to 500.
color	accent color; defaults to "black".
fade	integer; brightness of the gray tones between 0 = "black" and 100 = "white" in steps of 1; defaults to 85.
font	character; text font, use extrafonts to access additional fonts; defaults to "sans", which is "Helvetica".
cor_labels	logical; if TRUE, shows latent correlations between facets; defaults to TRUE.
size_title	integer; title font size relative to default.
size_cor_labels	integer; correlation font size relative to default.
size_test_label	integer; test font size relative to default.
size_facet_labels	integer; facet font size relative to default.
width_axes	integer; radial axis width relative to default.
width_circles	integer; facet circle outline width relative to default.
width_tick	integer; axis tick line width relative to default.
size_tick_label	integer; axis tick font size relative to default.

Details

Use facet_chart to create facet charts.

Value

Object of the class "ggplot".

See Also

coord_facets facet_chart

---

plot_items *Plot Items*

Description

Generates an item chart from coordinates.

Usage

```
plot_items(
  coord,
  size = 1,
  file_name = "none",
  file_width = 12,
  file_height = 10,
  zoom_x = NULL,
  zoom_y = NULL,
  dpi = 500,
  color = "black",
  color2 = "black",
  fade_axes = 50,
  fade_grid_major = 15,
  fade_grid_minor = 65,
  font = "sans",
  title = NULL,
  size_title = 1,
  size_tick_label = 1,
  size_test_label = 1,
  size_facet_labels = 1,
  width_axes = 1,
  size_arrow_heads = 1,
  width_items = 1,
  width_grid = 1
)
```

Arguments

coord	list generated by coord_items or coord_nested.
size	integer; changes the size of most chart objects simultaneously.
file_name	character; name of the file to save. Supported formats are: "pdf" (highest quality and smallest file size), "png", "jpeg"; defaults to "none".
file_width	integer; file width in inches; defaults to 12.
file_height	integer; file height in inches; defaults to 10.
zoom_x	integer; vector with two values, the edges of the zoomed section on the x-axis; defaults to NULL.
zoom_y	integer; vector with two values, the edges of the zoomed section on the y-axis; defaults to NULL.
dpi	integer; resolution in dots per inch for "png" and "jpeg" files; defaults to 500.
color	first accent color; defaults to "black".
color2	second accent color; defaults to "black".
fade_axes	integer; brightness of the gray tone of the axes between 0 = "black" and 100 = "white" in steps of 1; defaults to 50.

`fade_grid_major`
: integer; brightness of the gray tone of the major grid lines between 0 = "black" and 100 = "white" in steps of 1; defaults to 15.

`fade_grid_minor`
: integer; brightness of the gray tone of the minor grid lines between 0 = "black" and 100 = "white" in steps of 1; defaults to 65.

`font`
: character; text font, use extrafonts to access additional fonts; defaults to "sans", which is "Helvetica".

`title`
: character; overall chart title; defaults to NULL.

`size_title`
: integer; title font size relative to default.

`size_tick_label`
: integer; axis tick label font size relative to default.

`size_test_label`
: integer; test font size relative to default.

`size_facet_labels`
: integer; facet font size relative to default.

`width_axes`
: integer; radial axis width relative to default.

`size_arrow_heads`
: integer; arrow head size relative to default.

`width_items`
: integer; item bar width relative to default.

`width_grid`
: integer; grid line width relative to default.

Details

Use `item_chart` to create item charts.

Value

Object of the class "ggplot".

See Also

`coord_items` `item_chart`

---

plot_nested *Plot Nested*

Description

Generates a nested chart from coordinates.

Usage

```
plot_nested(
  coord,
  size = 1,
  file_name = "none",
  file_width = 10,
  file_height = 10,
  zoom_x = NULL,
  zoom_y = NULL,
  dpi = 500,
  cor_labels_tests = TRUE,
  cor_labels_facets = TRUE,
  color_global = "black",
  color_nested = "black",
  fade = 85,
  font = "sans",
  size_construct_label = 1,
  size_test_labels = 1,
  size_facet_labels = 1,
  width_axes = 1,
  width_axes_inner = 1,
  width_circles = 1,
  width_circles_inner = 1,
  width_tick = 1,
  width_tick_inner = 1,
  title = NULL,
  size_title = 1,
  size_tick_label = 1,
  size_cor_labels = 1,
  size_cor_labels_inner = 1,
  width_xarrows = 1,
  size_xarrow_heads = 1,
  size_xarrow_labels = 1
)
```

Arguments

coord	list generated by coord_nested.
size	integer; changes the size of most chart objects simultaneously.
file_name	character; name of the file to save. Supported formats are: "pdf" (highest quality and smallest file size), "png", "jpeg"; defaults to "none".
file_width	integer; file width in inches; defaults to 10.
file_height	integer; file height in inches; defaults to 10.
zoom_x	integer; vector with two values, the edges of the zoomed section on the x-axis; defaults to NULL.
zoom_y	integer; vector with two values, the edges of the zoomed section on the y-axis; defaults to NULL.

`dpi` integer; resolution in dots per inch for "png" and "jpeg" files; defaults to 500.

`cor_labels_tests`
: logical; if `TRUE`, shows the correlations between tests as text.

`cor_labels_facets`
: logical; if `TRUE`, shows the correlations between facets as text.

`color_global` global accent color; defaults to "black".

`color_nested` nested accent color; defaults to "black".

`fade` integer; brightness of the gray tones between 0 (black) and 100 (white) in steps of 1; defaults to 85.

`font` character; text font, use extrafonts to access additional fonts; defaults to "sans", which is "Helvetica".

`size_construct_label`
: integer; construct label font size relative to default.

`size_test_labels`
: integer; test label font size relative to default.

`size_facet_labels`
: integer; facet label font size relative to default.

`width_axes` integer; global radial axis width relative to default.

`width_axes_inner`
: integer; nested radial axis width relative to default.

`width_circles` integer; global circle outline width relative to default.

`width_circles_inner`
: integer; nested circle outline width relative to default.

`width_tick` integer; global axis tick line width relative to default.

`width_tick_inner`
: integer; nested axis tick line width relative to default.

`title` character; overall chart title; defaults to NULL.

`size_title` integer; title font size relative to default.

`size_tick_label`
: integer; axis tick label font size relative to default.

`size_cor_labels`
: integer; font size of the correlations between tests relative to default.

`size_cor_labels_inner`
: integer; font size of the correlations between facets relative to default.

`width_xarrows` integer; extra arrow line width relative to default.

`size_xarrow_heads`
: integer; extra arrow head length relative to default.

`size_xarrow_labels`
: integer; font size of the correlations indicated by extra arrows relative to default.

Details

Use `nested_chart` to create nested charts

Value

Object of the class "ggplot" and, by default, the same object saved as a file.

See Also

coord_nested nested_chart

---

rename *Rename*

Description

Renames tests, facets, or items in IPV estimates

Usage

```
rename(data, before, after)
```

Arguments

data	IPV estimates for chart creation or full output of ipv_est
before	character; a vector of names to replace
after	character; a vector of replacement names

Value

the same data with renamed values / variables

---

rename_est *Rename estimates*

Description

Renames tests, facets, or items in IPV data

Usage

```
rename_est(data, before, after)
```

Arguments

data	IPV chart creation data (nested or simple)
before	character; a vector of names to replace
after	character; a vector of replacement names

Value

the same data with renamed values / variables

rename_raw *Rename raw estimates*

Description

Renames tests, facets, or items in IPV raw estimates

Usage

```
rename_raw(data, before, after)
```

Arguments

data	IPV raw estimates (as provided by ipv_est as est_raw)
before	character; a vector of names to replace
after	character; a vector of replacement names

Value

the same data with renamed values / variables

rename_raw_simple *Rename Raw Simple*

Description

Rename Raw Simple

Usage

```
rename_raw_simple(data, before, after, regex = FALSE)
```

Arguments

data	IPV raw estimates (simple)
before	character; a vector of names to replace
after	character; a vector of replacement names
regex	logical; should items be renamed based on regular expressions for treatment of global section in nested data?; defaults to FALSE

Details

This function does not support regular expressions. Provide full names only.

Value

the same data with renamed values / variables

rename_simple *Rename Simple*

Description

Rename Simple

Usage

```
rename_simple(data, before, after, regex = FALSE)
```

Arguments

data	IPV estimates (simple)
before	character; a vector of names to replace
after	character; a vector of replacement names
regex	logical; should items be renamed based on regular expressions for treatment of global section in nested data?; defaults to FALSE

Details

This function does not support regular expressions. Provide full names only.

Value

the same data with renamed values / variables

rename_xarrow *Rename Xarrow*

Description

Rename Xarrow

Usage

```
rename_xarrow(data, before, after)
```

Arguments

data	IPV estimates for extra arrows (as provided e.g. by ipv_est)
before	character; a vector of names to replace
after	character; a vector of replacement names

Value

the same data with renamed values / variables

`self_confidence` *Self-Confidence*

Description

A set of center distances and latent correlations for items and facets of the Domain Specific Self-Esteem Inventory (DSSEI), Sports Mental Toughness Questionnaire (SMTQ), and Rosenberg Self-Esteem Scale (RSES) completed by 2272 german speaking participants using the german versions of the questionnaires. SEM estimation performed for all tests as one item pool.

Usage

```
self_confidence
```

Format

An object of class `list` of length 2.

Items

_____ RSES _____

__ Positive Self-Esteem (Ps) __

1. On the whole, I am satisfied with myself.

3. I feel that I have a number of good qualities.

4. I am able to do things as well as most other people.

7.I feel that I'm a person of worth.

10. I take a positive attitude toward myself.

__ Lack of Negative Self-Esteem (Ns) __

2. At times I think I am no good at all.

5. I feel I do not have much to be proud of.

6. I certainly feel useless at times.

8. I wish I could have more respect for myself.

9. All in all, I am inclined to think that I am a failure.

(Rosenberg, M. (1965). Rosenberg self-esteem scale (RSE). Acceptance and commitment therapy. Measures package, 61, 52.)

_____ SMTQ _____

`SMTQ`

_____ DSSEI _____

`DSSEI`

Source

Dantlgraber, M., Stieger, S., & Reips, U. D. (2019). Introducing Item Pool Visualization: A method for investigation of concepts in self-reports and psychometric tests. Methodological Innovations, 12(3), 2059799119884283.

`SMTQ` *Sports Mental Toughness Questionnaire*

Description

A set of center distances and latent correlations for items and facets of the SMTQ, completed by 2272 german speaking participants using the german version of the questionnaire. SEM estimation performed together with DSSEI and RSES (same as in "self_confidence").

Usage

```
SMTQ
```

Format

An object of class `list` of length 2.

Items

__ Confidence (Cf) __

13. I interpret potential threats as positive opportunities

5. I have an unshakeable confidence in my ability

11. I have qualities that set me apart from other competitors

6. I have what it takes to perform well while under pressure

14. Under pressure, I am able to make decisions with confidence and commitment

1. I can regain my composure if I have momentarily lost it

__ Constancy (Cs) __

3. I am committed to completing the tasks I have to do

12. I take responsibility for setting myself challenging targets

8. I give up in difficult situations

10. I get distracted easily and lose my concentration

__ Control (Ct) __

2. I worry about performing poorly

4. I am overcome by self-doubt

9. I get anxious by events I did not expect or cannot control

7. I get angry and frustrated when things do not go my way

(Sheard, M., Golby, J., & Van Wersch, A. (2009). Progress toward construct validation of the Sports Mental Toughness Questionnaire (SMTQ). European Journal of Psychological Assessment, 25(3), 186-193.)

Source

Dantlgraber, M., Stieger, S., & Reips, U. D. (2019). Introducing Item Pool Visualization: A method for investigation of concepts in self-reports and psychometric tests. Methodological Innovations, 12(3), 2059799119884283.

`write_IPV_syntax` *Write IPV syntax*

Description

Write lavaan model syntax of IPV models on the given dataset

Usage

```
write_IPV_syntax(dat, name)
```

Arguments

`dat`	data frame; correctly formatted raw data
`name`	character; name of the overall construct or test

Details

Variable names in dat have to conform to the pattern "test_facet_item". If there is only one test in the data, the pattern is "facet_item". For tests without facets in a larger dataset also comprising tests with items, the pattern is "test_item". Variable names have to be unique at the level of the test AND the facet. Item names have to be unique at the level of the test (not only at the level of the facet).

Value

list of character; lavaan model syntax

Index