Package 'ionr'

October 13, 2022

Type Package

Title Test for Indifference of Indicator

Version 0.3.0

Date 2016-10-02

Author Uku Vainik <ukuvainik@gmail.com>, Rene Mottus <rene.mottus@ed.ac.uk>

Maintainer Uku Vainik <ukuvainik@gmail.com>

Depends R (>= 3.0)

Imports gplots, psych

Suggests lavaan

Description Provides item exclusion procedure, which is a formal method to
test 'Indifference Of iNdicator' (ION). When a latent personality
trait-outcome association is assumed, then the association strength
should not depend on which subset of indicators (i.e. items) has been
chosen to reflect the trait. Personality traits are often measured
(reflected) by a sum-score of a certain set of indicators.
Item exclusion procedure randomly excludes items from a sum-score and
tests, whether the sum-score - outcome correlation changes. ION has been
achieved, when any item can be excluded from the sum-score without the
sum-score - outcome correlation substantially changing . For more details,
see Vainik, Mottus et. al, (2015) ``Are Trait-Outcome Associations Caused
by Scales or Particular Items? Example Analysis of Personality Facets and
BMI'',European Journal of Personality DOI: <10.1002/per.2009> .

License GPL (>= 2)

NeedsCompilation no

LazyData TRUE

RoxygenNote 5.0.1

Encoding UTF-8

URL www.ut.ee/uku.vainik/ion/

Repository CRAN

Date/Publication 2016-02-11 00:47:11

R topics documented:

---

ciplotter   *Plot confidence intervals*

Description

Used by [scenario_plot](#).

Usage

```
ciplotter(cix, ciy.lo, ciy.hi, eps, ...)
```

Arguments

cix	values of the x axis
ciy.lo	spread of confidence intervals
ciy.hi	spread of confidence intervals
eps	width of the whiskars
...	additional base plot arguments

---

ind_excl *Wrapper and scripts for indicator exclusion procedure*

Description

For each item, correlation between the scale's sum scores and outcome is calculated such that the particular item is excluded from the sum scores. Each of the obtained correlations will then be compared with the original scale–outcome correlation (sum score of all items). This comparison can be conducted with William's test for two dependent correlations that share one variable (Steiger, 1980), using `r.test` from **psych**. William's test characterises difference between correlations with a p-value—a small p-value indicates that the tested difference between correlations is unlikely to have happened by chance and could be considered a real difference. Thus, each item will receive a p-value characterising the 'significance' of difference between correlations—here called 'Significance Of iNdicator Exclusion' (SONE). When one item is excluded, another round is begun until no more items should be excluded.

Usage

```
ind_excl(indicators, indicators2 = vector(), outcome, covars = NULL,
  scalename = "scale", outcomename = "outcome",
  indicatornames = 1:ncol(indicators), pcrit, location1 = "topleft",
  location2 = "topright", draw = F, subset = vector(),
  coruse = "everything", multi = 1, verbose = F, ci = "estimate")
```

Arguments

indicators	Set of numeric indicators (items) in a matrix.
indicators2	An additional set of indicators (e.g. informant-report)
outcome	A numeric outcome vector. Indicators and outcome can be simulated with `scale_sim`
covars	A data frame with covariates to take into account. The outcome is residualised for these covariates. For instance, BMI was residualised for age, gender, and education in Vainik et al. (2015) EJP.
scalename	A string for labelling the scale
outcomename	A string for labelling the outcome
indicatornames	An array of strings for labelling the outcome. Default to numbers from 1 to n of indicators
pcrit	a p-value characterising the 'significance' of difference between correlations—here called 'significance of indicator exclusion' (SONE). Look it up from Table 2 in Vainik, Mõttus et al 2015, or simulate using `optimal_p` function
location1	Location for legends at left-side plot
location2	Location for legends at right-side plot
draw	TRUE plots the result to a .tiff file in the working directory. Defaults to FALSE
subset	Allows exluding certain indicators from the start. Use numbers

coruse	argument for function cor(). Defaults to 'everything', as simulations have no missing data.
multi	influences cex of certain plot variables. Defaults to 1
verbose	option for observing steps for debugging. Defaults to FALSE
ci	should output object and plot have 95 CI-s from corr.test. If you insert a number (e.g., ci=5000), then the CI-s are bootstrapped using cor.ci. Any other string results in no CI-s. r value in output matrix is taken from cor.

Value

Plots results using , using barplot2 from **gplots**. Also returns scale-outcome correlation magnitude(s) and their comparison, if appropriate

Examples

```
### Create a scale-outcome set that violates IOn_ Only 2 indicators out of 8
### relate to the outcome, the others just relate to the 2 indicators. This setting is
### similar to the N5: Impulsiveness - BMI association in Vainik et al (2015) EJP paper.
set.seed(466)
a<-scale_sim(n=2500, to_n=2, tn_n=6)
# Last 2 indicators have considerably higher correlation with the outcome
ind_excl(a[[1]], outcome=a[[2]], pcrit=0.0037)

## boostrapped confidence intervals
ind_excl(a[[1]], outcome=a[[2]], pcrit=0.0037, ci=100)

# no confidence intervals
ind_excl(a[[1]], outcome=a[[2]], pcrit=0.0037, ci="no")

## include covariates in the model
covx=rnorm(2500)
covy=rnorm(2500)
outcome=a[[2]]+0.3*covx+0.4*covy
covars=data.frame(covx=covx, covy=covy)

# ind_excl() with covariates taken into account
ind_excl(a[[1]],outcome=outcome,covars=covars, pcrit=0.0037)

#effect sizes are lower when noisy covariatse are not accounted for
ind_excl(a[[1]],outcome=outcome, pcrit=0.0037)

# just a single covariate also needs to be in data frame

covx=rnorm(2500)
outcome=a[[2]]+0.3*covx
covars=data.frame(covx=covx)

ind_excl(a[[1]],outcome=outcome,covars=covars, pcrit=0.0037)

### Create a scale-outcome set that has ION, all 8 indicators relate to the outcome
set.seed(466)
```

```
b<-scale_sim(n=2500, to_n=8, cor_to_outcome = 0.35)
# All indicators correlate largely on the same level with the outcome.
ind_excl(b[[1]], outcome=b[[2]], pcrit=1.7*10^-4)

#note that using cor_to_outcome=0.25, sometimes still indicators get wrongly flagged.
# Here, the method could probably be improved..

### Create a scale-outcome set that violates ION - only 1 indicator relates to the
### outcome. Include other-report.
set.seed(466)
c<-scale_sim(n=2500, to_n=1, tn_n=7, indicators2=TRUE)
# Last indicator has considerably higher correlation with the outcome
ind_excl(c[[1]], c[[3]], outcome=c[[2]], pcrit=0.0037)
```

---

ind_excl_inc   *Incrementally calls ind_excl_step*

Description

See [ind_excl](#) for details.

Usage

```
ind_excl_inc(indicators, outcome, indicatornames = 1:ncol(indicators),
  pcrit = 0.05, verbose = F, coruse = "everything")
```

Arguments

indicators	Set of numeric indicators (items) in a matrix.
outcome	A numeric outcome vector. Indicators and outcome can be simulated with [scale_sim](#)
indicatornames	An array of strings for labelling the outcome. Default to numbers from 1 to n of indicators
pcrit	a p-value characterising the 'significance' of difference between correlations—here called 'significance of indicator exclusion' (SONE). Look it up from Table 2 in Vainik, Mõttus et al 2015, or simulate using [optimal_p](#) function
verbose	option for observing steps for debugging. Defaults to FALSE
coruse	argument for function cor(). Defaults to 'everything', as simulations have no missing data.

Value

Provides the results of a single step in indicator exclusion procedure. See example for details

Examples

```
## Create a scale-outcome set that violates ION. Only 2 last indicators out of 8
## relate to the outcome, the others just relate to the 2 indicators
set.seed(466)
a<-scale_sim(n=2500, to_n=2, tn_n=6)
# run the exclusion procedure. Pcrit taken from Table 2 in Vainik et al., 2015,
# European Journal of Personality
res=ind_excl_inc(a[[1]],a[[2]], pcrit=0.0037)
# which indicators does the procedure exclude?
res
```

---

ind_excl_plot   *Plot indicator exlusion results with and without excluded indicators*

Description

Provides an overview of the indicator exclusion results. Marked(x) indicators are excluded in the indicator exclusion procedure. See `ind_excl` for details.

left correlations between single indicator and outcome

right correlations between sum-score and outcome with and without the marked indicators

Usage

```
ind_excl_plot(indicators, indicators2 = vector(), outcome,
  scalename = "scale", outcomename = "outcome",
  indicatornames = 1:ncol(indicators), tagged = vector(),
  tagged2 = vector(), location1 = "topleft", location2 = "topright",
  pcrit = 0.05, multi = 1, coruse = "everything", ci = "estimate")
```

Arguments

indicators	Set of numeric indicators (items) in a matrix.
indicators2	An additional set of indicators (e.g. informant-report)
outcome	A numeric outcome vector. Indicators and outcome can be simulated with `scale_sim`
scalename	A string for labelling the scale
outcomename	A string for labelling the outcome
indicatornames	An array of strings for labelling the outcome. Default to numbers from 1 to n of indicators
tagged	items to be marked as excluded by the indicator exclusion procedure
tagged2	same as 'tagged' for second scale (e.g., informant report)
location1	Location for legends at left-side plot
location2	Location for legends at right-side plot

pcrit	a p-value characterising the 'significance' of difference between correlations—here called 'significance of indicator exclusion' (SONE). Look it up from Table 2 in Vainik, Mõttus et al 2015, or simulate using optimal_p function
multi	influences cex of certain plot variables. Defaults to 1
coruse	argument for function cor(). Defaults to 'everything', as simulations have no missing data.
ci	should output object and plot have 95 CI-s from corr.test. If you insert a number (e.g., ci=5000), then the CI-s are bootstrapped using cor.ci. Any other string results in no CI-s. r value in output matrix is taken from cor.

Value

See ind_excl

---

ind_excl_step	*One step in indicator exclusion procedure*

Description

See ind_excl for details.

Usage

```
ind_excl_step(indicators, outcome, indicatornames = 1:ncol(indicators),
  exclude = vector(), coruse = "everything", round = F)
```

Arguments

indicators	Set of numeric indicators (items) in a matrix.
outcome	A numeric outcome vector. Indicators and outcome can be simulated with scale_sim
indicatornames	An array of strings for labelling the outcome. Default to numbers from 1 to n of indicators
exclude	Exclude an item excluded at previous step, e.g., as decided by ind_excl_inc
coruse	argument for function cor(). Defaults to 'everything', as simulations have no missing data.
round	Allows rounding of values in returned matrix.

Value

Provides the results of a single step in indicator exclusion procedure. See example for details

Examples

```
## Create a scale-outcome set that violates ION. Only 2 indicators out of 8 relate to
## the outcome, the others just relate to the 2 indicators
set.seed(466)
a<-scale_sim(n=2500, to_n=2, tn_n=6)
res=ind_excl_step(a[[1]],a[[2]])
print(res)

# note that the p-values for upper items (7 & 8 ) are much smaller than for the rest

#row number    indicator number
#r.test.t      t value of the r.test.
#t.test.p      p value of the r.test.
#cor_excl      correlation between outcome and sum-score when an item is excluded.
#cor_all       correlation between outcome and sum-score when all items are included
# (i.e., full scale).
#cor.excl_all correlation between two sum-scores.
```

---

optimal_p   *Find an optimal p-value for SONE*

Description

a wrapper that runs the maximum and minimum scenarios using [scenario_sim](#) and provides the optimal p -value

Usage

```
optimal_p(sizes, n_sim = 100, plotting = "", n_indicators = 8,
  to_min = (round((n_indicators/2), 0)) - 1, ...)
```

Arguments

sizes	An array of sample sizes to be simulated. Can be single value.
n_sim	number of simulations. 1000 is a start, 10000 was used in paper, but takes a long time
plotting	Plots the result with [optimal_p_out](#). Defaults to ". Possible options: " - no plot; 'yes' - a regular plot; 'file' – writes the plot to a tiff file in working directory. If sizes is a single value, plotting is disabled.
n_indicators	How many many indicators are there in a scale. The package is tested with 8 indicators (default), but should work with other number.
to_min	How many indicators relate to the outcome in the lack of ION condition. In [optimal_p](#) defaults to (round((n_indicators/2),0)) - 1, i.e close to half the number of indicators.
...	further tweaking of the scale simulator, see [scale_sim](#) for details.

Value

Returns the P criterion, as well as the p values for max and min scenario for each sample size. If min pvalue > max pvalue, then p criterion is NA.

Examples

```
set.seed(466)
n_sim=100
ptm <- proc.time()
a=optimal_p(sizes=750, n_sim=n_sim, n_indicators=8, cor_to_outcome=0.25)
stp=proc.time() - ptm
print(paste("Currently elapsed:",round(stp[3],1)))
print(paste("Time estimate for n_sim=5000:",round(stp[3]*5000/n_sim,1)))
```

---

optimal_p_out   *Table and plot the SONE values*

Description

Takes max and min scenarious and produces a table and optionally a plot. See scenario_sim or optimal_p.

Usage

```
optimal_p_out(scenario_max, scenario_min, sizes, n_sim, to_min, plotting = "",
  multi = 1)
```

Arguments

scenario_max	SONE data from scenario_sim output
scenario_min	SONE data from scenario_sim output
sizes	An array of sample sizes to be simulated. Can be single value.
n_sim	number of simulations. 1000 is a start, 10000 was used in paper, but takes a long time
to_min	How many indicators relate to the outcome in the lack of ION condition. In optimal_p defaults to (round((n_indicators/2),0)) - 1), i.e close to half the number of indicators.
plotting	Plots the result with optimal_p_out. Defaults to ". Possible options: " - no plot; 'yes' - a regular plot; 'file' – writes the plot to a tiff file in working directory. If sizes is a single value, plotting is disabled.
multi	influences cex of certain plot variables. Defaults to 1

Examples

```
set.seed(466)
sizes=c(500,1000)
n_sim=50  #  make bigger for more accurate estimates..
to_n=8
cor_to_outcome=0.25
ptm <- proc.time()  # timing
# takes a few seconds..
scen1=scenario_sim(sizes=sizes,n_sim=n_sim,to_n=to_n, cor_to_outcome=cor_to_outcome)
proc.time() - ptm
ptm <- proc.time()
# A scenario with 3 out of 8 items relating to outcome, 3 different samples
to_n=3
scen2=scenario_sim(sizes=sizes,n_sim=n_sim,to_n=to_n, cor_to_outcome=cor_to_outcome)
proc.time() - ptm

optimal_p_out(scen1[[1]],scen2[[1]],sizes = sizes,n_sim=n_sim,to_min = to_n, plot='yes', multi=1)

# Should be equivalent. Some variation can be expected when n_sim is below 1000
ptm <- proc.time()
a=optimal_p(sizes=sizes, n_sim=n_sim, n_indicators=8, plotting='yes', cor_to_outcome=cor_to_outcome)
proc.time() - ptm
print(a[[1]])
```

---

scale_sim   *Simulate personality scale(s) and an outcome*

Description

Simulates a personality scale which correlates to an outcome. The function can specify the number of indicators (i.e. indicators) truly relating to the outcome. Also, the function can create a secondary scale, for instance mimicing informant-report

Usage

```
scale_sim(n, to_n, tn_n = 0, indicators2 = FALSE, cor_to_tn = 0.3,
  cor_to_outcome = 0.4, to_min = 0.4, to_max = 0.7, tn_min = 0.4,
  tn_max = 0.7, n.cat = 5, sdev = 0.8)
```

Arguments

n	Number of participants
to_n	Number of indicators in a Trait relating to Outcome
tn_n	Number of indicators in a Trait Not relating to outcome.
indicators2	if TRUE, a secondary set of indicators is created, e.g. to mimic informant-report. Defaults to FALSE
cor_to_tn	Correlation between to and tn. Defaults to 0.3

cor_to_outcome	correlation between to and outcome. Defaults to 0.4
to_min	minimum factor loading for to_n. Defaults to 0.4
to_max	maximum factor loading for to_n. Defaults to 0.7
tn_min	minimum factor loading for tn_n. Defaults to 0.4
tn_max	maximum factor loading for tn_n. Defaults to 0.7
n.cat	number of response options. when you go larger than 5, update the standard deviation as well. Defaults to 5
sdev	standard deviation. Defaults to 0.8

Value

A list object, first object is indicators' matrix and second object is outcome vector. If indicators2=TRUE, then a third object is added, which is the secondary indicators' matrix.

Examples

```
## Create a scale-outcome set that violates ION. Only 2 indicators out of 8 relate
## to the outcome, the others just relate to the 2 indicators This setting is similar
## to the N5: Impulsiveness - BMI association in Vainik et al (2015) EJP paper.
set.seed(466)
a<-scale_sim(n=2500, to_n=2, tn_n=6)
# Last 2 indicators have considerably higher correlation with the outcome
cor(a[[1]],a[[2]])

## Create a scale-outcome set that has ION, all 8 indicators relate to the outcome
set.seed(466)
b<-scale_sim(n=2500, to_n=8)
# All indicators correlate largely on the same level with the outcome.
cor(b[[1]],b[[2]])

## Create a scale-outcome set that violates ION - only 1 indicator relates to
##the outcome. Include other-report.
set.seed(466)
c<-scale_sim(n=2500, to_n=1, tn_n=7, indicators2=TRUE)
# Last 2 indicators have considerably higher correlation with the outcome
cor(c[[1]],c[[2]])
cor(c[[3]],c[[2]])
```

---

scenario_plot	*Plot scenario simulation results*

Description

Plots the results like in Study 1 of the Vainik et al. paper an overview of the indicator exclusion results. Starred indicators are excluded in the indicator exclusion procedure. See scenario_sim for details. NB! Scenario with no ION violations needs *scenario_plot80()* See scenario_sim for examples

Usage

```
scenario_plot(dat, sizes, n_sim, to_n, tn_n = 8 - to_n, multi = 1,
  jitter = 0.05, letter = "", ...)

scenario_plot80(dat, sizes, n_sim, multi = 1, letter = "", ...)
```

Arguments

dat	simulated data
sizes	An array of sample sizes to be simulated. Can be single value.
n_sim	number of simulations. 1000 is a start, 10000 was used in paper, but takes a long time
to_n	Number of indicators in a Trait relating to Outcome
tn_n	Number of indicators in a Trait Not relating to outcome.
multi	influences cex of certain plot variables.
jitter	Avoid overalp between lines
letter	assigns plot a letter, useful for combining multiple plots
...	additional options for axis

Functions

- `scenario_plot80`: For plotting scenarios where ION is not violated

---

scenario_sim   *Simulate SONE values for scenario.*

Description

A wrapper that takes a scenario, and produces the Significance Of iNdicator Exclusion (SONE) values for each exclusion and calculates efficacy. Used by [optimal_p](#).

Usage

```
scenario_sim(sizes, n_sim, to_n, tn_n = 8 - to_n, ...)
```

Arguments

sizes	An array of sample sizes to be simulated. Can be single value.
n_sim	number of simulations. 1000 is a start, 10000 was used in paper, but takes a long time
to_n	Number of indicators in a Trait relating to Outcome
tn_n	Number of indicators in a Trait Not relating to outcome.
...	further tweaking of the scale simulator, see [scale_sim](#) for details.

Value

Returns a list of SONE values and related efficacy. See example for details

1. SONE results. Feed this to [scenario_plot](#) or [scenario_plot80](#) (see examples)
2. Summary efficacy. Such data comprises Table 1 in Vainik, Mottus et al., 2015 EJP
3. Full efficacy data.

Examples

```
#A scenario with 8 items relating to outcome, testing 2 different samples
sizes=c(250,500)
n_sim=100
to_n=8
scen1=scenario_sim(sizes,n_sim,to_n)  # takes a few seconds..
scenario_plot80(scen1[[1]],sizes,n_sim)

# A scenario with 2 out of 8 items relating to outcome, 2 different samples
to_n=2
scen2=scenario_sim(sizes,n_sim,to_n)  # takes a few seconds..
scenario_plot(scen2[[1]],sizes,n_sim,to_n)
```

Index