

Package ‘anchorpoint’

January 8, 2021

Title Anchor Point Selection Based on the Gini Inequality Criterion

Version 0.0.0.9000

Description This package implements a Anchor Point Selection method based on the paper 'Anchor Point Selection – Scale Alignment Based on an Inequality Criterion' by Strobl et al. (2020). It provides data generating processes and graphical decision support functions (criterion path, shifted item plots and a graphical test).

License `use_mit_license()`, `use_gpl3_license()` or friends to
pick a license

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1

Imports grDevices,
stats,
ineq,
graphics,
scales,
psychotools,
multcomp

Suggests qvcalc,
mirt

R topics documented:

anchorpoint	2
checkInput	3
clfCriterion	3
dgp_multi	4
dgp_single	5
diftests	6
generateGrid	6
getCriterionRes	7
getData	8
getItemDiscrimination	8
getWald	9
get_covmat	10
get_results	10

graphicalTest	11
plot.anchorpoint	11
plotCriterion	12
print.anchorpoint	13
print.plot.anchorpoint	13
print.summary.anchorpoint	14
print.waldtestpV	14
raschFit	15
shiftPlot	15
summary.anchorpoint	16
waldtestpV	17
Wald_test	17

Index 19

anchorpoint	<i>Function to produce anchorpoint objects</i>
-------------	------------------------------------------------

Description

Function to produce anchorpoint objects

Usage

```
anchorpoint(
  rm1,
  rm2,
  select = c("CLF Criterion", "Gini Index"),
  grid = c("symmetric", "sparse")
)
```

Arguments

rm1	Fitted Rasch Model object corresponding to the first group. Object is of class "raschmodel", produced by function <code>RaschModel.fit</code> of the package psychotools .
rm2	Fitted Rasch Model object corresponding to the second group. Object is of class "raschmodel", produced by function <code>RaschModel.fit</code> of the package psychotools .
select	a string, specifying the criterion that is evaluated ("CLF Criterion" or "Gini Index", abbreviations are accepted)
grid	a string, specifying the grid method that is used to generated the shifts for evaluation ("symmetric" or "sparse", abbreviations are accepted)

Value

an anchorpoint object containing:

- list with global optima
- list with all results (grids and criterion values)
- string with used criteria
- string with used grids
- list with the rm object

checkInput	<i>Function to check user-specific Input for the right format</i>
------------	-------------------------------------------------------------------

Description

Function to check user-specific Input for the right format

Usage

```
checkInput(manuelInput, resp.var, group.var)
```

Arguments

manuelInput	manual Data input as list with response matrix and grouping vector
resp.var	name of the binary response matrix
group.var	name of the binary grouping vector

Value

Data ready for anchorpoint::raschFit function

clfCriterion	<i>Calculates the CLF criterion</i>
--------------	-------------------------------------

Description

Calculates the CLF criterion

Usage

```
clfCriterion(dist, eps = 1e-04)
```

Arguments

dist	distance vector
eps	shift for numerical stability (see Muthen paper)

Value

criterion value

dgp_multi

Data generating process for multi dimensional rasch model (only two dimensions are supported at the moment)

Description

Data generating process for multi dimensional rasch model (only two dimensions are supported at the moment)

Usage

```
dgp_multi(
  nobs,
  tlength,
  DIFpercent,
  Nr.dim = 2,
  Theta = NULL,
  a.vec = NULL,
  d.vec = NULL,
  DIF_mode = "intersect",
  d_distr = list(mean = 0, sd = 0.2),
  MultiNorm = NULL,
  itemtype = "dich"
)
```

Arguments

nobs	positive interger, number of total observations (default 1000) or positive integer vector of length 2, number of group observations
tlength	positive interger, number of items (default 30)
DIFpercent	numeric array, proportion of items which have DIF per dimension (default 0,1/3)
Nr.dim	positive interger, number of dimensions (default 2)
Theta	matrix of the underlying ability parameters (optional)
a.vec	numeric nobs x Nr.dim-dimensional matrix, item discrimination parameter matrix (if NA, calculated according to DIFmode & DIFpercent)
d.vec	numeric one-dimensional vector, task difficulty matrix (if NA, calculated according to d_distr)
DIF_mode	positive interger, number of dimensions (default 2)
d_distr	distribution parameters normal distribution to calculate d.vec, default: mean = 0, sd = .2
MultiNorm	list with parameters for multivariate normal which models the abilities of group 1 and group 2, respectively
itemtype	type of items (default "dich" which corresponds to multidimensional Rasch model items)

Value

list consisting of:

- binary response matrix
- group vector (factor),
- a.vec,
- d.vec,
- DIFindex,
- Theta,
- DIFside

dgp_single

Data generating process for one dimensional rasch model

Description

Data generating process for one dimensional rasch model

Usage

```
dgp_single(
  nobs,
  tlength,
  DIFpercent,
  DIFpattern = "balanced",
  DIFeffect = "constant",
  DIFamount = 0.6,
  ability = TRUE,
  sigmaable = c(1, 1),
  itemref = c(-2.522, -1.902, -1.351, -1.092, -0.234, -0.317, 0.037, 0.268, -0.571,
    0.317, 0.295, 0.778, 1.514, 1.744, 1.951, -1.152, -0.526, 1.104, 0.961, 1.314,
    -2.198, -1.621, -0.761, -1.179, -0.61, -0.291, 0.067, 0.706, -2.713, 0.213, 0.116,
    0.273, 0.84, 0.745, 1.485, -1.208, 0.189, 0.345, 0.962, 1.592)
)
```

Arguments

nobs	number of observations in each group (here equal)
tlength	number of items the test contains
DIFpercent	rel. frequency of DIF items in the test
DIFpattern	"balanced": DIF balanced over groups "favorref", "favorfoc": all DIF items favor one group
DIFeffect	data generating process for DIF effect: "normal" -> Wang(2012), "uniform" -> alternative
DIFamount	magnitude of DIF
ability	ability
sigmaable	positive numeric vector of length two (two groups), default c(1,1), standard deviation for person parameter matrix (abilities)
itemref	numeric vector of length tlength (if shorter, then sampling with replacement is used), items difficulty parameters according to Hartmann

Value

list containing response matrix, group assignments, DIFindex, DIFside, itemref, itemfoc

diftests	<i>Extend diftests function of psychotools to include offset</i>
----------	------------------------------------------------------------------

Description

Extend diftests function of psychotools to include offset

Usage

```
diftests(obj1, obj2, anchor_items, adjust = "none", offset = 0)
```

Arguments

obj1	rasch model object 1
obj2	rasch model object 2
anchor_items	anchor items
adjust	p-value adjustment
offset	offset

Value

list containing test results, item paramters and covariance

generateGrid	<i>This function generates the grid values.</i>
--------------	-------------------------------------------------

Description

This function generates the grid values.

Usage

```
generateGrid(
  beta1,
  beta2,
  grid_method = c("symmetric", "sparse"),
  j.length = 1000
)
```

Arguments

beta1	Coefficients from first Rasch model fit
beta2	Coefficients from second Rasch model fit
grid_method	a string, specifying the grid method that is used to generated the shifts for evaluation ("symmetric" and/or "sparse", abbreviations are accepted)
j.length	positive integer, the granularity of the symmetric grid (default: 1000)

Value

A list with the selected grid methods each having two components:

1. A vector 'c_grid' with the grid values
2. A list of matrices 'betas_grid' for each grid value (length = j.length) each matrix has three columns: coefficient from group 1 (beta1), shifted coefficient from group 2 (newbeta2) and distance between them (beta1-newbeta2). The rows correspond to the items.

getCriterionRes	<i>Criterion function</i>
-----------------	---------------------------

Description

Criterion function

Usage

```
getCriterionRes(
  rm1,
  rm2,
  select = c("Gini Index", "CLF Criterion"),
  grid = c("symmetric", "sparse"),
  shift = NULL
)
```

Arguments

rm1	Fitted Rasch Model object corresponding to the first group. Object is of class "raschmodel", produced by function <code>raschmodel</code> of the package psychotools .
rm2	Fitted Rasch Model object corresponding to the second group. Object is of class "raschmodel", produced by function <code>raschmodel</code> of the package psychotools .
select	criterion: Gini Index or CLF Criterion
grid	grid method: symmetric or sparse
shift	desired shift. if NULL, then the criterion maximizing is used. Can also be numeric to get desired shift. Caution: must be within grid!

Value

a list which contains:

- a list with the results (grid values, criterion values, information about the optima)
- a rm object,

 getData

Simulate data from one dimensional or multidimensional DGP

Description

Simulate data from one dimensional or multidimensional DGP

Usage

```
getData(nobs, tlength, DIFpercent, type = c("single", "multi"), ...)
```

Arguments

nobs	positive integer, number of observations
tlength	positive integer, number of items
DIFpercent	non-negative double ≤ 1 , DIF percentage
type	string specifying which DGP to use ("single" or "multi")
...	additional arguments: <ul style="list-style-type: none"> • For single-dimensional DGP, check ?anchorpoint::dgp_single • For multi-dimensional DGP, check ?anchorpoint::dgp_multi

Value

a list containing

- DGP: simulated data
- RM: Rasch Model objects

 getItemDiscrimination *Function to create a item discrimination parameter matrix*

Description

Function to create a item discrimination parameter matrix

Usage

```
getItemDiscrimination(
  dimensions,
  DIFpercent,
  tlength,
  DIF_mode = c("intersect", "disjoint")
)
```


Arguments

dimensions	integer specifying the number of dimensions used in dgp (currently only 2 are allowed)
DIFpercent	numeric, quantifying DIF percentage
tlength	positive integer specifying the number of items
DIF_mode	character vector specifying the mode how to create the matrix: <ul style="list-style-type: none"> • "intersect": all items load on the first, length*DIFpercent items also on the second • "disjoint": ceiling(tlength*DIFpercent) items load on the first, the rest on the second, where ceiling rounds the number up to the next integer

Value

A binary item discrimination parameter matrix of dimension: tlength x dimensions

getWald	<i>Function which executes Wald test for given rm object and shift (with "min_dist" setting)</i>
---------	--------------------------------------------------------------------------------------------------

Description

Function which executes Wald test for given rm object and shift (with "min_dist" setting)

Usage

```
getWald(rm, shift)
```

Arguments

rm	A list containing the two Rasch Model objects of group 0 and group 1
shift	The desired shift of the second group

Value

A list containing the output of the function Wald_test():

- p: results from the test (p-values)
- vcov: the covariance matrices of the fit (from diftests function)

get_covmat	<i>Function that gives back a covariance matrix for n dimesnions</i>
------------	----------------------------------------------------------------------

Description

Function that gives back a covariance matrix for n dimesnions

Usage

```
get_covmat(Nr.dim, variances = 0.25, covariances = 0.125)
```

Arguments

Nr.dim	integer - the number of dimensions
variances	numeric, positive, <= 1, (same for all dimensions) or Nr.dim-dimensional vector - variance of each dimension
covariances	numeric, positive, <= 1, (same for all dimensions) or choose(Nr.dim,2)-dimensional vector - covariances between dimensions

Value

covariance matrix of dimension Nr.dim x Nr.dim

get_results	<i>Function to evaluate criterion values and obtain test results for a given grid and method</i>
-------------	--------------------------------------------------------------------------------------------------

Description

Function to evaluate criterion values and obtain test results for a given grid and method

Usage

```
get_results(grid, shift, getTestResults, rm, metric)
```

Arguments

grid	The grid values: output of the "generateGrid.R" function
shift	the desired shift
getTestResults	logic, whether test should be applied
rm	list containing the two Rasch Model corresponding two group 0 and 1
metric	criterion to evaluate as a function

Value

a list containing the criterion evaluated at grid points and the result of the wald test

graphicalTest	<i>Function to produce graphical test plot</i>
---------------	------------------------------------------------

Description

Function to produce graphical test plot

Usage

```
graphicalTest(
  object,
  shift = NULL,
  highlight = NULL,
  alpha = 0.05,
  testColors = list(`not significant` = "darkgreen", significant = "red3", `anchor\n
                    item` = "black"),
  TestResults = NULL,
  ask = TRUE,
  ...
)
```

Arguments

object	anchorpoint object as produced by the function anchorpoint
shift	numeric, shift which is applied, default NULL (global optimum)
highlight	positive integer, items to be highlighted (invalid items are excluded).
alpha	significance level for DIF test
testColors	list with colors for the items: <ul style="list-style-type: none"> • "not significant" = "darkgreen" • "significant" = "red3" • "anchor item" = "black"
TestResults	waldtest object from anchorpoint::getWald. If NULL, then they are computed within the function. Default: NULL.
ask	logical, ask for next plot. Default = TRUE
...	further arguments for plot() like lty, cex.axis, cex.main, cex.lab etc. credit: some code is adapted from the function plotGOF of the R package eRm.

plot.anchorpoint	<i>Plot function hand over location_picker = TRUE, to identify specific points in the plot to terminate the function, press any mouse button other than the first (X11 device) or press ESC key (quartz) see ?identify for help</i>
------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Description

Plot function hand over location_picker = TRUE, to identify specific points in the plot to terminate the function, press any mouse button other than the first (X11 device) or press ESC key (quartz) see ?identify for help

Usage

```
## S3 method for class 'anchorpoint'
plot(x, ask = T, location_picker = FALSE, ...)
```

Arguments

x	anchorpoint object as produced by the function anchorpoint
ask	logical, ask for next plot. Default = TRUE
location_picker	logical, use location picker. Default FALSE.
...	additional parameters for plot function as for standard plot function (e.g. col)

plotCriterion	<i>Function to produce criterion plot</i>
---------------	-------------------------------------------

Description

Function to produce criterion plot

Usage

```
plotCriterion(
  object,
  names,
  location_picker = FALSE,
  lty = 1,
  col = 1,
  cex.axis = 1,
  cex.lab = 1,
  cex.main = 1,
  cex = 1
)
```

Arguments

object	anchorpoint object as produced by the function anchorpoint
names	list, with criterion and grid: names of the methods used.
location_picker	use location picker
lty	line type
col	color
cex.axis	cex.axis
cex.lab	cex.lab
cex.main	cex.main
cex	cex

Value

selected points with additional information

print.anchorpoint	<i>Print function</i>
-------------------	-----------------------

Description

Print function

Usage

```
## S3 method for class 'anchorpoint'  
print(x, ...)
```

Arguments

x	anchorpoint object as produced by the function anchorpoint
...	further arguments passed to or from other methods (e.g. digits for rounding).

print.plot.anchorpoint	<i>Print function for plot.anchorpoint</i>
------------------------	--------------------------------------------

Description

Print function for plot.anchorpoint

Usage

```
## S3 method for class 'plot.anchorpoint'  
print(x, ...)
```

Arguments

x	plot.anchorpoint object
...	further arguments passed to or from other methods.

```
print.summary.anchorpoint
```

Print function for summary.anchorpoint

Description

Print function for summary.anchorpoint

Usage

```
## S3 method for class 'summary.anchorpoint'  
print(x, ...)
```

Arguments

x	summary.anchorpoint object
...	further arguments passed to or from other methods.

```
print.waldtestpV
```

Print function for waldtestpV object

Description

Print function for waldtestpV object

Usage

```
## S3 method for class 'waldtestpV'  
print(x, ...)
```

Arguments

x	waldtest object
...	further arguments passed to or from other methods.

raschFit	<i>Fits Rasch models for the reference group 0 and the focal group 1</i>
----------	--------------------------------------------------------------------------

Description

Fits Rasch models for the reference group 0 and the focal group 1

Usage

```
raschFit(data, resp.mat.name = "i", group.name = "groups")
```

Arguments

data	<ul style="list-style-type: none"> • data.frame - simulated or a real data. Must contain: • response item matrix (matrix), binary (0/1) input. • group (vector), the group of the test takers.
resp.mat.name	string vector, the name of the response matrix in 'data' input with 'i' as a default (as dgp).
group.name	string vector, the group name in the data frame 'data' (as dgp).

Value

two objects of class "raschmodel", produced by function RaschModel.fit of the package **psychotools**.

shiftPlot	<i>Function to produce shift Plot</i>
-----------	---------------------------------------

Description

Function to produce shift Plot

Usage

```
shiftPlot(
  object,
  shift = NULL,
  testColors = list(`not significant` = "darkgreen", significant = "red3", `anchor
    item` = "black"),
  testPCH = list(`not significant` = 21, significant = 22, `anchor item` = 23),
  addLegend = TRUE,
  highlight = NULL,
  digits = 3,
  cex.legend = 0.5,
  TestResults = NULL,
  ask = TRUE,
  ...
)
```

Arguments

object	anchorpoint object as produced by the function anchorpoint
shift	numeric, shift which is applied
testColors	list with colors for the items: <ul style="list-style-type: none"> • "not significant" = "darkgreen" • "significant" = "red3" • "anchor item" = "black"
testPCH	list with pch for the items (for color blind people): <ul style="list-style-type: none"> • "not significant" = 21 • "significant" = 22 • "anchor item" = 23
addLegend	logic, add a legend to the plot, default: False
highlight	positive integer, items to be highlighted (invalid items are excluded).
digits	positive integer, controls rounding of the shift in title
cex.legend	numeric, controls size of legend
TestResults	waldtest object from anchorpoint::getWald. If NULL, then they are computed within the function. Default: NULL.
ask	logical, ask for next plot. Default = TRUE
...	additional graphics arguments

summary.anchorpoint	<i>Summary function</i>
---------------------	-------------------------

Description

Summary function

Usage

```
## S3 method for class 'anchorpoint'
summary(object, ...)
```

Arguments

object	anchorpoint object as produced by the function anchorpoint
...	additional arguments affecting the summary produced.

waldtestpV	<i>Function to get waldtest p-value results</i>
------------	-------------------------------------------------

Description

Function to get waldtest p-value results

Usage

```
waldtestpV(object, shift = NULL, ...)
```

Arguments

object	anchorpoint object as produced by the function anchorpoint
shift	shift, default NULL (global optimum), else numeric
...	further arguments for signif(x,...) (digits)

Wald_test	<i>Function to create a item discrimination parameter matrix</i>
-----------	------------------------------------------------------------------

Description

Function to create a item discrimination parameter matrix

Usage

```
Wald_test(
  rm1,
  rm2,
  c_shift,
  alias_method = c("constant4_MPT", "quasi_var", "min_dist"),
  alias_anchor_items = NULL,
  adjust = "none"
)
```

Arguments

rm1	Fitted Rasch Model object corresponding to the first group. Object is of class "raschmodel", produced by function RaschModel.fit of the package psychotools .
rm2	Fitted Rasch Model object corresponding to the first group. Object is of class "raschmodel", produced by function RaschModel.fit of the package psychotools .
c_shift	The shift of the second group
alias_method	character vector specifying the aliasing method. One of "constant4_MPT", "quasi_var", "min_dist".
alias_anchor_items	integer in 1,...,#items. Default: NULL, will be chosen according to alias_method
adjust	p-value adjustment (multiple testing correction), Default: "none"

Value

list containing

- p: results from the test (p-values)
- vcov: the covariance matrices of the fit (from diftests function)

Index

anchorpoint, [2](#)

checkInput, [3](#)
clfCriterion, [3](#)

dgp_multi, [4](#)
dgp_single, [5](#)
diftests, [6](#)

generateGrid, [6](#)
get_covmat, [10](#)
get_results, [10](#)
getCriterionRes, [7](#)
getData, [8](#)
getItemDiscrimination, [8](#)
getWald, [9](#)
graphicalTest, [11](#)

plot.anchorpoint, [11](#)
plotCriterion, [12](#)
print.anchorpoint, [13](#)
print.plot.anchorpoint, [13](#)
print.summary.anchorpoint, [14](#)
print.waldtestpV, [14](#)

raschFit, [15](#)

shiftPlot, [15](#)
summary.anchorpoint, [16](#)

Wald_test, [17](#)
waldtestpV, [17](#)