Package ‘deltaPlotR’

February 19, 2015

Type Package

Title Identification of dichotomous differential item functioning (DIF) using Angoff’s Delta Plot method

Version 1.5

Date 2014-11-20

Author David Magis (U Liege), Bruno Facon (Univ Lille-Nord de France)

Maintainer David Magis <david.magis@ulg.ac.be>

Depends R (>= 2.10.0), MASS

Description The deltaPlotR package proposes an implementation of Angoff’s Delta Plot method to detect dichotomous DIF. Several detection thresholds are included, either from multivariate normality assumption or by prior determination. Item purification is supported.

License GPL (>= 2)

LazyLoad yes

NeedsCompilation no

Repository CRAN

Date/Publication 2014-11-20 11:56:37

R topics documented:

deltaPlotR-package ......................................................... 2
adjustExtreme .............................................................. 2
deltaPlot ................................................................. 5
diagPlot ................................................................. 11
verbal ................................................................. 15

Index 17
Identification of dichotomous differential item functioning (DIF) using Angoff’s Delta Plot method

Description

The deltaPlotR package proposes an implementation of Angoff’s Delta Plot method to detect dichotomous DIF. Several detection thresholds are included, either from multivariate normality assumption, random simulation, or by prior determination. Item purification is supported.

Details

```
Package: deltaPlotR
Type: Package
Version: 1.5
Date: 2014-11-20
Depends: R (>= 2.10.0), MASS
License: GPL (>= 2)
```

Author(s)

David Magis
Post-doc Fellow of the National Funds for Scientific Research (FNRS, Belgium)
University of Liege
<David.Magis@ulg.ac.be>, http://ppw.kuleuven.be/okp/home/

Bruno Facon
Professor, Department of Psychology
Universite Lille-Nord de France
<bruno.facon@univ-lille3.fr>,

See Also

Other useful packages can be found in the R Psychometric task view. See http://cran.stat.sfu.ca/web/views/Psychometrics.html for further details.
Description

This command modifies the proportions of correct responses when these equal either zero or one, for compatibility with the Delta plot.

Usage

adjustExtreme(data = NULL, group = NULL, focal.name = NULL, prop,
method = "constraint", const.range = c(0.001, 0.999), nrAdd = 1)

Arguments

data numeric: the data matrix: one row per subject, one column per item, with possible entries 0, 1 or NA. Default value is NULL.
group numeric or character: a vector of the same length as nrow(data) with two different values, one for the reference group, one for the focal group. Values can be either numeric or character. Default value is NULL.
focal.name numeric or character: the value used in the group vector to refer to the focal group. Default value is NULL.
prop numeric: a matrix with one row per item and two columns. The first column holds the percentage of correct responses for each item in the reference group, and the second column holds the same percentages but for the focal group. Default value is NULL.
method character: the method used to modify the extreme proportions. Possible values are "constraint" (default) or "add". See Details.
const.range numeric: a vector of two constraining proportions. Default values are 0.001 and 0.999. Ignored if method is "add".

Details

The Delta plot method requires the computation of the proportions of correct responses per item and per group. However, these proportions must be strictly greater than zero and smaller than one, since they are transformed onto z-scores. Thus, extreme proportions must be adjusted for proper use of the Delta plot.

Two approaches are currently implemented and set to adjustExtreme by the method argument.

The first method is the constraint method and is set by method="constraint". It simply consists in constraining the proportions within a specified range of values in (0,1). This restricted range of values is set by the const.range argument and takes the default value c(0.001, 0.999).

The second method is the so-called add method and is specified by method="add". It consists in arbitrarily adding some successes and the same number of failures to the data, in order to get a modified proportion of successes. This number of extra successes is set by the nrAdd argument, with default value one. In sum, by default one success one failure is added to the item responses, so that the newly computed proportion of successes is not extreme anymore, yet close to the original value. This default values refers to the so-called Laplace rule (see e.g. Jaynes, 2003).
The input arguments are: the data matrix of item responses (ith possible entries 0, 1 and NA for missing data), the vector of group membership and the numeric (or character) value coding for the focal group. By default they take the NULL value so they can be left unassigned, but then only the "constraint" method can be applied. In any case, the matrix of proportions of correct responses per item and per group of respondents must be specified through the prop argument.

Value

A list with the following arguments:

adj.prop the restricted proportions, in the same format as the input prop matrix.
method the value of the method argument or "constraint" if the latter was applied by lack of data.
range the value of the const.range argument.
nrAdd the value of the nrAdd argument.

Author(s)

David Magis
Post-doc Fellow of the National Funds for Scientific Research (FNRS, Belgium)
University of Liege
<David.Magis@ulg.ac.be>, http://ppw.kuleuven.be/okp/home/
Bruno Facon
Professor, Department of Psychology
Universite Lille-Nord de France
<bruno.facon@univ-lille3.fr>,

References


See Also
deltaPlot
**deltaPlot**

*Delta Plot method for dichotomous DIF*

**Description**

This command computes the Delta plot statistics for dichotomous differential item functioning, with all associated output (Delta points, perpendicular distances). The modified Delta plot is also available, as well as several item purification techniques.

**Usage**

deltaPlot(data, type = "response", group, focal.name, thr = "norm", purified = FALSE, purType = "IPP1", maxIter = 10, alpha = 0.05, extreme = "constraint", const.range = c(0.001, 0.999), nrAdd = 1, ...)

---

**Examples**

```r
# Loading of the verbal data
data(verbal)
attach(verbal)

# Excluding the "Anger" variable
verbal <- verbal[!colnames(verbal) == "Anger"]

# Computing the proportions of correct answers per group
prop <- matrix(NA, 24, 2)
for (i in 1:24) {
  prop[i,1] <- mean(verbal[,25] == 0, na.rm=TRUE)
  prop[i,2] <- mean(verbal[,25] == 1, na.rm=TRUE)
}

# "constraint" method
adjustExtreme(data=verbal[,1:24], group=verbal[,25], focal.name=1, prop=prop)

# "constraint" method with differently specified range
adjustExtreme(data=verbal[,1:24], group=verbal[,25], focal.name=1, prop=prop, const.range=c(0.01, 0.99))

# "add" method
adjustExtreme(data=verbal[,1:24], group=verbal[,25], focal.name=1, prop=prop, method="add")

# "add" method with different number of successes added
adjustExtreme(data=verbal[,1:24], group=verbal[,25], focal.name=1, prop=prop, method="add", nrAdd=2)

# "constraint" method because of lack of provided data
adjustExtreme(prop=prop)
```
deltaPlot

save.output = FALSE, output = c("out", "default")
## S3 method for class 'deltaPlot'
print(x, only.final = TRUE, ...)

Arguments

data numeric: either (a) the data matrix with item responses and group membership, (b) the two-column matrix of proportions of correct responses per item and per group, or (c) the two-column matrix of Delta scores. See Details.
type character: the type of data argument. Possible values are "response" (default), "prop" and "delta". See Details.
group integer or character: a single value for locating the group membership column in the data argument. Ignored if type is not "response". See Details.
focal.name numeric or character: the value used in the group membership column to refer to the focal group. Ignored if type is not "response". See Details.
thr numeric or character: the threshold for flagging items as DIF. Can be a positive numeric value or "norm". See Details.
purify logical: should item purification be performed? (Default is codeFALSE). See Details.
purType character: the type of purification process to be run. Possible values are "IPP1" (default), "IPP2" and "IPP3". Ignored if purify is FALSE. See Details.
maxIter integer: the maximum number of iteration in the purification process (default is 10). Ignored if purify is FALSE.
able numeric: the significance level for calculating the detection threshold (default is 0.05). Ignored if thr is numeric.

extreme character: the method used to modify the extreme proportions. Possible values are "constraint" (default) or "add". See Details.

const.range numeric: a vector of two constraining proportions. Default values are 0.001 and 0.999. Ignored if extreme is "add".
nrAdd integer: the number of successes and the number of failures to add to the data in order to adjust the proportions. Default value is 1. Ignored if extreme is "constraint".

save.output logical: should the output be saved into a text file? (Default is FALSE).
output character: a vector of two components. The first component is the name of the output file ("out" by default), the second component is either the file path or "default" (default value). See Details.
x an object of class "deltaPlot", typically the output of the deltaPlot function.
only.final logical: should only the first and last steps of the purification process be printed? (default is TRUE. If FALSE all perpendicular distances, parameters of the major axis, and detection thresholds are printed additionally. Ignored if purify is FALSE.

... other generic parameters for the plot or the print functions.
Angoff’s Delta plot (Angoff and Ford, 1973) is a straightforward test-score method to detect DIF among dichotomously scored items. Proportions of correct responses are computed first per item and per group of respondents, and are successively transformed onto $z$-scores and then onto $\Delta$ scores. The pairs of $\Delta$ scores can be displayed onto a scatter plot, called the Delta plot, and the major axis of the ellipsoid of Delta points is derived. Eventually, items whose perpendicular distance (from the major axis) is too large are flagged as DIF. See Angoff and Ford (1973) for further details.

The data must be passed through the argument data and can be of three types. Each type is defined by the type argument and can take three values: "response", "prop" and "delta".

- If type is "response", the input data consist in a matrix with one row per respondent and $J + 1$ columns, where $J$ is the number of items. In the columns coding for the items, only possible entries are 0 (for incorrect responses), 1 (for correct responses) and NA (for missing values). The extra column is used to define group membership: all respondents of the reference group take the same value (either numeric or character), and all respondents in the focal group take the same (numeric or character) value but different from the reference group. Note that the group membership column can be located anywhere in the data set (not especially in first or last position).
- If type is "prop", the input data consist in a two-column matrix with one row per item. Each row contains the proportions of correct responses, respectively in the reference group (first column) and in the focal group (second column).
- If type is "delta", the input data consist in a two-column matrix that is similar to that provided with the "prop" type of input, but with the Delta scores provided instead of the proportions of correct responses.

If the type of input is either "prop" or "delta", not anymore input information is required and the arguments group and focal.nname are ignored. Otherwise, the group membership column in the data matrix is specified by giving to argument group either the column number (1 for first column, etc.) or the column name (provided the data matrix has argument names). Moreover, the focal group is specified by giving to the argument focal.nname the value that was used in the group membership column to code for the focal group.

If the input type is not "delta", then extreme proportions of correct responses (either provided when type is "prop" or computed from the data if type is "response") are adjusted by specifying the arguments extreme, const.range and nrAdd with appropriate values. See the adjustExtreme function for further details (note that the current extreme argument corresponds to the method argument in this function).

The threshold for flagging items as DIF can be of two types and is specified by the thr argument.

1. It can be fixed to some arbitrary positive value by the user, for instance 1.5 (Angoff and Ford, 1973). In this case, thr takes the required numeric threshold value.
2. Alternatively, it can be derived from the bivariate normal approximation of the Delta points (Magis and Facon, 2012). In this case, thr must be given the character value "norm" (which is the default value). This threshold equals

$$\Phi^{-1}(1-\alpha/2) \sqrt{\frac{b^2 s_0^2 - 2 b s_0 s_1 + s_1^2}{b^2 + 1}}$$

where $\Phi$ is the density of the standard normal distribution, $\alpha$ is the significance level (set by the argument alpha with default value 0.05), $b$ is the slope parameter of the major axis, $s_0$
and \( s_1 \) are the sample standard deviations of the Delta scores in the reference group and the focal group, respectively, and \( s_{01} \) is the sample covariance of the Delta scores (see Magis and Facon, 2012, for further details).

Item purification can be performed by setting the argument \texttt{purify} to \texttt{TRUE} (by default it is \texttt{FALSE} so no purification is performed). The item purification process (IPP) starts when at least one item was flagged as DIF after the first run of the Delta plot, and proceeds as follows.

1. The intercept and slope parameters of the major axis are re-calculated by removing all DIF that are currently flagged as DIF. This yields updated values \( a^*, b^*, s_0^*, s_1^* \) and \( s_{01}^* \) of the intercept and slope parameters, sample standard deviations and sample covariance of the Delta scores.
2. Perpendicular distances (for all items) are updated with respect to the updated major axis.
3. Detection threshold is also updated. Three possible updates are possible: see below.
4. All items are now tested for the presence of DIF, given the updated perpendicular distances and major axis.
5. If the set of items flagged as DIF is the same as the one from the previous loop, stop the process. Otherwise go back to step 1.

Unlike traditional DIF methods, the detection threshold may also be updated since it depends on the sample estimates (when the normal approximation is considered). Three approaches are currently implemented and are specified by the \texttt{purType} argument.

1. Method 1 (\texttt{purType="IPP1"): the same threshold is used throughout the purification process, it is not iteratively updated. The threshold is the one obtained after the first run of the Delta plot.
2. Method 2 (\texttt{purType="IPP2"): only the slope parameter is updated in the threshold formula. By this way, one keeps the full data structure (i.e. neither the sample variances nor the sample covariance of the Delta scores are modified) but only the slope parameter is adjusted to lessen the impact of DIF items.
3. Method 3 (\texttt{purType="IPP3"): all adjusted parameters are plugged in the threshold formula. This approach completely discards the effect of items flagged as DIF from the computation of the threshold.

See Magis and Facon (2013) for further details. Note that purification can also be performed with fixed threshold (i.e. specified by the user), but then only IPP1 process is performed.

In order to avoid possible infinite loops in the purification process, a maximal number of iterations must be specified through the argument \texttt{maxIter}. The default maximal number of iterations is 10.

The output contains all input information, the Delta scores and perpendicular distances, the parameter of the major axis and the items flagged as DIF (if none, a character sentence is returned). In addition, the detection threshold and the type of threshold (fixed or normal approximation) is provided.

If item purification was run, several additional elements are returned: the number of iterations, a logical indicator whether the convergence was reached (or not, meaning that the process stopped because of reaching the maximal number of allowed iterations), a matrix with indicators of which items were flagged as DIF at each iteration, and the type of item purification process. Moreover, perpendicular distances are returned in a matrix format (one column per iteration), as well as successive major axis parameters (one row per iteration) and successive thresholds (as a vector).
The output is managed and printed in a more user-friendly way. When item purification is performed, only the first and last steps are displayed. Specifying the argument `only.final` to `FALSE` prints in addition all intermediate steps of the process (successive perpendicular distances, parameters of the major axis, and detection thresholds).

The output can be saved into a text file by specifying the argument `save.output` to `TRUE` (by default the output is not captured). If so, the argument output can be specified as a vector of two character values. The first one gives the desired name of the text file, and the second one specifies the directory where the file will be saved (full path is required but without the final "/" symbol, see Examples below). By default, the output will be saved in the current working directory as "out.txt" file.

Value

A list of class "deltaPlot" with the following arguments:

- `Props` the matrix of proportions of correct responses, or `NA` if type is "delta".
- `adjProps` the restricted proportions, in the same format as the output `Props` matrix, or `NA` if type is "delta".
- `Deltas` the matrix of Delta scores.
- `Dist` a matrix with perpendicular distances, one row per item and one column per run of the Delta plot. If `purify` is `FALSE`, only a single column is returned.
- `axis.par` a matrix with two columns, holding respectively the intercepts and the slope parameters of the major axis. Each row refers to one step of the purification process. If `purify` is `FALSE`, only a single row is returned.
- `nIter` the number of iterations involved in the purification process. Returned only if `purify` is `TRUE`.
- `maxIter` the value of the `maxIter` argument. Returned only if `purify` is `TRUE`.
- `convergence` a logical value indicating whether convergence was reached in the purification process. Returned only if `purify` is `TRUE`.
- `difPur` a matrix with one column per item and one row per iteration in the purification process, holding zeros and ones to indicate which items were flagged as DIF or not at each step of the process. Returned only if `purify` is `TRUE`.
- `thr` a vector of successive threshold values used during the purification process. If `purify` is `FALSE`, a single value is returned.
- `rule` a character value indicating whether the threshold was “fixed” by the user (i.e. by setting `thr` to a numeric value) or whether it was computed by normal approximation (i.e. by setting `thr` to "norm").
- `purType` the value of the `purType` argument. Returned only if `purify` is `TRUE`.
- `DIFItems` either "No DIF item detected” or an integer vector with the items that were flagged as DIF.
- `adjust.extreme` the value of the `extreme` argument.
- `const.range` the value of the `const.range` argument.
- `nrAdd` the value of the `nrAdd` argument.
- `purify` the value of the `purify` argument.
alpha the value of the alpha argument.

save.output the value of the save.output argument.

output the value of the output argument.

Author(s)

David Magis
Post-doc Fellow of the National Funds for Scientific Research (FNRS, Belgium)
University of Liege
<David.Magis@ulg.ac.be>, http://ppw.kuleuven.be/okp/home/

Bruno Facon
Professor, Department of Psychology
Universite Lille-Nord de France
<bruno.facon@univ-lille3.fr>,

References


See Also

adjustExtreme

Examples

# Loading of the verbal data
data(verbal)
attach(verbal)

# Excluding the "Anger" variable
verbal <- verbal[, colnames(verbal)!="Anger"]

# Basic Delta plot, threshold 1.5, no item purification
res <- deltaPlot(data=verbal, type="response", group=25, focal.name=1, purify=FALSE, thr=1.5)

# Equivalent writing
res <- deltaPlot(data=verbal, type="response", group="Gender", focal.name=1,
# Using proportions of correct responses as input
dataRef <- verbal[verbal[,25]==0,1:24]
dataFoc <- verbal[verbal[,25]==1,1:24]
p0 <- colMeans(dataRef)
p1 <- colMeans(dataFoc)
res.1 <- deltaPlot(data=cbind(p0,p1), type="prop", purify=FALSE, thr=1.5)

# Using Delta values as input
Delta <- 4*qnorm(1-cbind(p0,p1)) + 13
res.2 <- deltaPlot(data=Delta, type="delta", purify=FALSE, thr=1.5)

# 'norm' threshold
res <- deltaPlot(data=verbal, type="response", group="Gender", focal.name=1,
    purify=FALSE, thr="norm")

# Keeping the first 10 items to exhibit DIF
data <- verbal[,c(1:10,25)]
deltaPlot(data=data, type="response", group=11, focal.name=1, purify=FALSE, thr="norm")
    # Item 8 is flagged as DIF

# Item purification with the three processes
res0 <- deltaPlot(data=data, type="response", group=11, focal.name=1, purify=TRUE,
    thr=1.5, purType="IPP1")
res0 # No DIF item detected
res1 <- deltaPlot(data=data, type="response", group=11, focal.name=1, purify=TRUE,
    thr="norm", purType="IPP1")
res1 # Item 8 flagged as DIF after 2 iterations
res2 <- deltaPlot(data=data, type="response", group=11, focal.name=1, purify=TRUE,
    thr="norm", purType="IPP2")
res2 # Item 8 flagged as DIF after 2 iterations
res3 <- deltaPlot(data=data, type="response", group=11, focal.name=1, purify=TRUE,
    thr="norm", purType="IPP3")
res3 # Items 6, 7 and 8 flagged as DIF after 4 iterations

# Printing the full results of item purification
print(res, only.final=FALSE)
print(res0, only.final=FALSE)
print(res1, only.final=FALSE)
print(res2, only.final=FALSE)
print(res3, only.final=FALSE)

---

**diagPlot**

*Plot of the Deltas points from the Delta Plot method*
**Description**

This command plots the output of the `deltaPlot` function as a diagonal plot of Deltas points. Several graphical options are available.

**Usage**

```r
diagPlot(x, pch = 2, pch.mult = 17, axis.draw = TRUE, thr.draw = FALSE, 
dif.draw = c(1,3), print.corr = FALSE, xlim = NULL, ylim = NULL, xlab = NULL, 
ylab = NULL, main = NULL, save.plot = FALSE, 
save.options = c("plot", "default", "pdf"))
```

**Arguments**

- `x`: an object of class "deltaPlot", typically the output of the `deltaPlot` function.
- `pch`: integer: the usual point character type for point display. Default value is 2, that is, Delta points are drawn as empty triangles.
- `pch.mult`: integer: the type of point to be used for superposing onto Delta points that correspond to several items. Default value is 17, that is, full black triangles are drawn onto existing Delta plots wherein multiple items are located.
- `axis.draw`: Logical: should the major axis be drawn? (default is TRUE). If so, it will be drawn as a solid line.
- `thr.draw`: logical: should the upper and lower bounds for DIF detection be drawn? (default is FALSE). If TRUE, they will be drawn as dashed lines.
- `dif.draw`: numeric: a vector of two integer values to specify how the DIF items should be displayed. The first component of `dif.draw` is the type of point (i.e. the usual `pch` argument) and the second component determines the point size (i.e. the usual `cex` argument). Default values are 1 and 3, meaning that empty circles of three times the usual size are drawn around the Delta points of items flagged as DIF.
- `print.corr`: Logical: should the sample correlation of Delta scores be printed? (default is FALSE). If TRUE, it is printed in upper-left corner of the plot.
- `xlim`, `ylim`, `xlab`, `ylab`, `main`: either the usual plot arguments `xlim`, `ylim`, `xlab`, `ylab` and `main`, or NULL (default value for all arguments). If NULL, the X and Y axis limits are computed from the range of Delta scores, the X and Y axis labels are "Reference group" and "Focal group" respectively, and no main title is produced.
- `save.plot`: logical: should the plot be saved in an external figure? (default is FALSE).
- `save.options`: character: a vector of three components. The first component is the name of the output file, the second component is either the file path (without final "/" symbol) or "default" (default), and the third component is the file extension, either "pdf" (default) or "jpeg". Ignored if `save.plot` is FALSE.

**Details**

The results of the Delta plot method can be graphically displayed using this function. Basically the Delta plot displays the items in a scatter plot by means of their Delta points, and the major axis is drawn. Several options permit to enhance this basic plot.
The input data \( x \) must be a list of class `deltaplot`, so typically the output of the `deltaplot` function. All other arguments are rather standard and for optimization of the graphical display.

The type of point is defined by the `cex` argument. It takes the default value 2, which means that items are displayed with empty triangles. If several items are located on exactly the same Delta point, the `pch_mult` argument defines the type of point to display over the existing point. The default value is 17, that is, a full black triangle. In this way, multiple items located at a single Delta point can easily be located on the plot.

Two types of axes can be drawn: the major axis and the upper and lower bounds for DIF detection. The major axis is drawn by default, while the upper and lower bounds are not. The major axis can be withdrawn by setting the argument `axis_draw` to `FALSE`, and the bounds can be displayed by setting the argument `thr_draw` to `TRUE`. The major axis is always drawn by a solid line, the bounds by dashed lines.

Items flagged as DIF are also clearly identified on the plot. The argument `dif_draw` defines both the type of point and the size of the point to draw over the existing Delta points (for items flagged as DIF only). The default value is \( c(1,3) \), meaning that empty circles three times larger than usual are drawn.

The sample correlation between the Delta scores can also be printed, in the upper-left corner of the plot. To do this, the argument `print_corr` must be set to `TRUE`.

Finally, the function will automatically determine the X and Y axis limits and specifies default labels for X and Y axes and the main title. These can also be specified by the user, using the usual `xlim`, `ylim`, `xlab`, `ylab` and `main` arguments.

The plot can be saved in an external file, either as PDF or JPEG format. First, the argument `save_plot` must be set to `TRUE` (default is `FALSE`). Then, the name of the figure, its location and format are specified through the argument `save_options`, all as character strings. See the Examples section for further information and a practical example.

**Author(s)**

David Magis  
Post-doc Fellow of the National Funds for Scientific Research (FNRS, Belgium)  
University of Liege  

Bruno Facon  
Professor, Department of Psychology  
Universite Lille-Nord de France  
<bruno.facon@univ-lille3.fr>,

**References**


See Also

deltaplot

Examples

```r
# Loading of the verbal data
data(verbal)
attach(verbal)

# Excluding the "Anger" variable
verbal <- verbal[!colnames(verbal) == "Anger"]

# Basic Delta plot, threshold 1.5, no item purification
res <- deltaplot(data=verbal, type="response", group=25, focal.name=1, purify=FALSE, thr=1.5)

# Keeping the first 10 items to exhibit DIF
data <- verbal[1:10,]
res0 <- deltaplot(data=data, type="response", group=11, focal.name=1, purify=FALSE, thr="norm")
res0 # Item 8 is flagged as DIF

res1 <- deltaplot(data=data, type="response", group=11, focal.name=1, purify=TRUE, thr="norm", purtype="IPP3")
res1 # Items 6, 7 and 8 flagged as DIF after 4 iterations

# Delta plot, default options
diagPlot(res)
diagPlot(res0)
diagPlot(res1)

# Drawing upper and lower bounds and removing the major axis
diagPlot(res, axis.draw=FALSE, thr.draw=TRUE)
diagPlot(res1, axis.draw=FALSE, thr.draw=TRUE)

# Modifying the type of points for all and for DIF items
diagPlot(res, pch=3, dif.draw=c(2,4))
diagPlot(res1, pch=3, dif.draw=c(2,4))

# Printing the correlation and modifying the axis limits
diagPlot(res, xlim=c(9,20), ylim=c(9,20), print.corr=TRUE)
diagPlot(res1, xlim=c(9,17), print.corr=TRUE)

# Saving the plots as PDF and JPEG files, default folder, specific names
diagPlot(res, save.plot=TRUE, save.options=c("res", "default", "pdf"))
diagPlot(res1, save.plot=TRUE, save.options=c("res1", "default", "jpeg"))
```
# Modifying the results to make two items be located on the same place
res2<-res1
res2$Deltas[9,]<-res2$Deltas[3,]
diagPlot(res2)

verbal Verbal Aggression Data Set

Description

The Verbal Aggression data set comes from Vansteelandt (2000) and is made of the responses of 316 subjects (243 women and 73 men) to a questionnaire of 24 items, about verbal aggression. All items describe a frustrating situation together with a verbal aggression response. A correct answer responses is coded as 0 and 1, a value of one meaning that the subject would (want to) respond to the frustrating situation in an aggressive way. In addition, the Trait Anger score (Spielberger, 1988) was computed for each subject.

Format

The verbal matrix consists of 316 rows (one per subject) and 26 columns.

The first 24 columns hold the responses to the dichotomously scored items. The 25th column holds the trait anger score for each subject. The 26th column is vector of the group membership; values 0 and 1 refer to women and men, respectively.

Each item name starts with S followed by a value between 1 and 4, referring to one of the situations below:

S1: A bus fails to stop for me.
S2: I miss a train because a clerk gave me faulty information.
S3: The grocery store closes just as I am about to enter.
S4: The operator disconnects me when I had used up my last 10 cents for a call.

The second part of the name is either Want or Do, and indicates whether the subject wanted to respond to the situation or actually did respond.

The third part of the name is one of the possible aggressive responses, either Curse, Scold or Shout.

For example, item S1WantShout refers to the sentence: "a bus fails to stop for me. I want to shout". The corresponding item response is 1 if the subject agrees with that sentence, and 0 if not.

Note

This data set was originally included in the difR package (Magis, Beland and Raiche, 2012). It is reproduced here for illustrative purposes.
Source

The Verbal aggression data set is taken originally from Vansteelandt (2000) and has been used as an illustrative example in De Boeck (2008), De Boeck and Wilson (2004) and Smits, De Boeck and Vansteelandt (2004), among others. The following URL \url{http://bear.soe.berkely.edu/EIRM/} permits to get access to the full data set.

References

Index

*Topic package
  deltaPlotR-package, 2

adjustExtreme, 2, 7, 10

deltaPlot, 4, 5, 12–14
deltaPlotR (deltaPlotR-package), 2
deltaPlotR-package, 2
diagPlot, 11

print.deltaPlot (deltaPlot), 5

verbal, 15