Package ‘TDIagree’

May 7, 2026

Type Package

Title Assessment of Agreement using the Total Deviation Index

Version 0.2.0

Maintainer Anna Felip-Badia <`annafelipibadia@gmail.com`>

Description The total deviation index (TDI) is an unscaled statistical measure used to evaluate the deviation between paired quantitative measurements when assessing the extent of agreement between different raters. It describes a boundary such that a large specified proportion of the differences in paired measurements are within the boundary (Lin, 2000) <`https://pubmed.ncbi.nlm.nih.gov/10641028/`>.
This R package implements some methodologies existing in the literature for TDI estimation and inference in the case of two raters.

License GPL (>= 2)

Depends R (>= 4.1.0)

Encoding UTF-8

LazyData true

RoxygenNote 7.3.3

Imports boot, gt, multcomp, nlme, stats, plotfunctions, katex, furniture, graphics

NeedsCompilation no

Author Anna Felip-Badia [aut, cre] (ORCID: <`https://orcid.org/0009-0002-8751-5228`>),
Sara Perez-Jaume [aut] (ORCID: <`https://orcid.org/0000-0002-9805-4810`>),
Josep L Carrasco [aut] (ORCID: <`https://orcid.org/0000-0003-1184-0753`>)

Repository CRAN

Date/Publication 2025-10-29 11:20:02 UTC

Contents

`TDIagree-package` *Assessment of Agreement using the Total Deviation Index*

Description

The total deviation index (TDI) is an unscaled statistical measure used to evaluate the deviation between paired quantitative measurements when assessing the extent of agreement between different raters. It describes a boundary such that a large specified proportion of the differences in paired measurements are within the boundary (Lin, 2000). This R package implements some methodologies existing in the literature for TDI estimation and inference in the case of two raters reviewed in Perez-Jaume and Carrasco (2015).

Functions

`TDI`, `descriptive.plots`

Methods

`print.tdi`, `plot.tdi`

Datasets

`AMLad`

Author(s)

Maintainer: Anna Felip-Badia <`annafelipibadia@gmail.com`> (ORCID)

Authors:

- Sara Perez-Jaume (ORCID)
- Josep L Carrasco (ORCID)

References

Lin, L. I. K. (2000). Total deviation index for measuring individual agreement with applications in laboratory performance and bioequivalence. Statistics in Medicine, 19(2):255-270.

`AMLad` *Acute Myeloid Leukaemia agreement data*

Description

Acute myeloid leukaemia (AML) is a type of cancer that starts in the blood-forming cells of the bone marrow. While in adults it is the most common type of leukaemia, it is much rarer in children, accounting for 15-20% percent of paediatric leukaemia cases, which translates to 8 cases per year for every million children under the age of 15 years.

Minimal residual disease (MRD) is the percentage of cancer cells that remain in a person either during or after treatment when the patient is in remission (no symptoms or signs of disease). MRD aids in identifying high-risk patients so therapy can be intensified in them while deintensification of therapy can prevent long-term sequelae of chemotherapy in low-risk category patients.

MRD describes disease that can be detected using techniques other than traditional morphology, including molecular methods such as polymerase chain reaction (PCR) and immunological methods such as flow cytometry (FCM) (Chattaerjee *et al.*, 2016).

This dataset is adapted from the *Childhood Leukemia: Overcoming distance between South America and Europe Regions* (CLOSER) project, whose goal was to decrease the gap between Europe and Latin America in terms of the diagnosis, monitoring, survival, and quality of life of patients with childhood leukaemia and their caregivers. See **Source** for further information on the project. The dataset contains data from 116 paediatric patients diagnosed with AML, in which the MRD was measured twice after treatment initiation by the methods PCR and FCM.

Usage

```
AMLad
```

Format

A data frame in long format with the following columns:

id:	Patient identifier
met:	Method to quantify MRD (PCR or FCM)
rep:	Replicate (1 = first, 2 = second)
mrd:	MRD (%)

Source

https://closerleukemia.eu/

References

Chatterjee, T., Mallhi, R. S., & Venkatesan, S. (2016). Minimal residual disease detection using flow cytometry: Applications in acute leukemia. Medical Journal Armed Forces India, 72(2), 152-156.

`descriptive.plots` *Descriptive plots for the assessment of agreement*

Description

This function implements some basic descriptive plots that can be analysed in the context of agreement. The plots implemented are:

- a scatter plot of all the possible paired-measurements of one rater against the other
- a boxplot of the measurements per rater and, if it applies, replicate
- a histogram of the measurements per rater and, if it applies, replicate
- a normal quantile-quantile plot of the measurements per rater and, if it applies, replicate
- a histogram of the paired-differences
- a normal quantile-quantile plot of the paired-differences

Usage

```
descriptive.plots(data, y, id, met, rep = NA,
                  plots = c("scatterplot", "boxplot",
                            "histogram.y", "qqplot.y",
                            "histogram.d", "qqplot.d"),
                  rater.col = c("#cc6e5c", "#06402b"), d.col = "lavender",
                  scatterplot.xlab = NULL, scatterplot.ylab = NULL,
                  histogram.y.main = NULL, histogram.y.xlab = NULL,
                  qqplot.y.main = NULL,
                  line.col = "red",
                  histogram.d.density = TRUE, histogram.d.bw = "nrd0",
                  density.col = "red", ...)
```

Arguments

`data`	name of the dataset, of class `data.frame`, containing at least 3 columns (quantitative measurement, subject effect, rater effect).
`y`	quantitative measurement column name.
`id`	subject effect column name. The corresponding column of `data` must be a factor.
`met`	rater effect column name. The corresponding column of `data` must be a factor.
`rep`	replicate effect column name. When there are no replicates the user should use `rep = NA`. When there are replicates, the corresponding column of data must be a factor. The default value is `NA`.
`plots`	name of the descriptive plots to display. The options are: `"scatterplot"` (a scatter plot of all the possible paired-measurements of one rater against the other), `"boxplot"` (a boxplot of the measurements per rater and, if it applies,

	replicate), `"histogram.y"` (a histogram of the measurements per rater and, if it applies, replicate), `"qqplot.y"` (a QQ plot of the measurements per rater and, if it applies, replicate), `"histogram.d"` (a histogram of the paired-differences) and `"qqplot.d"` (a QQ plot of the paired-differences). This argument is not case-sensitive and is passed to `match.arg`. The default value is `c("scatterplot", "boxplot", "histogram.y", "qqplot.y", "histogram.d", "qqplot.d")`, so all plots are executed by default.
`rater.col`	colour palette to be used in the plots involving the measurements separated per rater. If neither `"boxplot"` nor `"histogram.y"` are selected in `plots`, this argument is ignored. The default value is `c("#cc6e5c", "#06402b")` for the first and second level, respectively, of the variable met in `data`.
`d.col`	colour to be used in the histogram involving the paired-differences. If `"histogram.d"` is not selected in `plots`, this argument is ignored. The default value is `"lavender"`.
`scatterplot.xlab`	a label to use for the x-axis in the scatterplot. If `"scatterplot"` is not selected in `plots`, this argument is ignored. The default value, `NULL`, indicates that the label *"*`y`*" from "1st level of* `met` *column"* should be used.
`scatterplot.ylab`	a label to use for the y-axis in the scatterplot. If `"scatterplot"` is not selected in `plots`, this argument is ignored. The default value, `NULL`, indicates that the label *"*`y`*" from "2nd level of* `met` *column"* should be used.
`histogram.y.main`	overall title for the histograms of the measurements (to be passed to `main` argument in `hist`). Notice that all histograms (one per rater and, if it applies, replicate) will have the same title. If `"histogram.y"` is not selected in `plots`, this argument is ignored. The default value, `NULL`, indicates that the label *"j-th level of* `met` *column"* should be used in the case of no replicates, where $j \in \{1,2\}$, and the label *"i-th level of* `rep` *column" from "j-th level of* `met` *column"* should be used in the case of replicates, where $i \in \{1,\ldots,\text{number of replicates}\}$ and $j \in \{1,2\}$.
`histogram.y.xlab`	a label to use for the x-axis in the histograms of the measurements (to be passed to `xlab` argument in `hist`). Notice that all histograms (one per rater and, if it applies, replicate) will have the same label. If `"histogram.y"` is not selected in `plots`, this argument is ignored. The default value, `NULL`, indicates that the label `"y"` should be used.
`qqplot.y.main`	overall title for the normal quantile-quantile plots of the measurements (to be passed to `main` argument in `qqplot`). Notice that all QQ plots (one per rater and, if it applies, replicate) will have the same title. If `"qqplot.y"` is not selected in `plots`, this argument is ignored. The default value, `NULL`, indicates that the label *"j-th level of* `met` *column"* should be used in the case of no replicates, where $j \in \{1,2\}$, and the label *"i-th level of* `rep` *column" from "j-th level of* `met` *column"* should be used in the case of replicates, where $i \in \{1,\ldots,\text{number of replicates}\}$ and $j \in \{1,2\}$.

`line.col`	colour to be used in the theoretical line added to the scatterplot or the normal quantile-quantile plots. If none of `"scatterplot"`, `"qqplot.y"`, or `"qqplot.d"` are selected in `plots`, this argument is ignored. The default value is `"red"`.
`histogram.d.density`	logical indicating whether the density should be added as a solid curve to the histogram of the paired-differences. If `"histogram.d"` is not selected in `plots`, this argument is ignored. The default value is `TRUE`.
`histogram.d.bw`	the smoothing bandwidth to be used in the density curve added to the histogram involving the paired differences (to be passed to bw argument in `density`). If `"histogram.d"` is not selected in `plots` or `histogram.d.density` is `FALSE`, this argument is ignored. The default value is `"nrd0"`.
`density.col`	colour to be used in the density curve added to the histogram involving the paired differences. If `"histogram.d"` is not selected in `plots` or `histogram.d.density` is `FALSE`, this argument is ignored. The default value is `"red"`.
`...`	other graphical parameters (to be passed to each plot). To be used calling only one plot for optimal performance.

Value

The desired plots are shown one after the other.

Examples

```
# normal data

set.seed(2025)

n <- 100

mu.ind <- rnorm(n, 0, 7)

epsA1 <- rnorm(n, 0, 3)
epsA2 <- rnorm(n, 0, 3)
epsB1 <- rnorm(n, 0, 3)
epsB2 <- rnorm(n, 0, 3)

y_A1 <- 50 + mu.ind + epsA1 # rater A, replicate 1
y_A2 <- 50 + mu.ind + epsA2 # rater A, replicate 2
y_B1 <- 40 + mu.ind + epsB1 # rater B, replicate 1
y_B2 <- 40 + mu.ind + epsB2 # rater B, replicate 2

ex_data <- data.frame(y = c(y_A1, y_A2, y_B1, y_B2),
                      rater = factor(rep(c("A", "B"), each = 2*n)),
                      replicate = factor(rep(rep(1:2, each = n), 2)),
                      subj = factor(rep(1:n, 4)))
```

```
descriptive.plots(ex_data, y, subj, rater, replicate, plots = "scatterplot")
descriptive.plots(ex_data, y, subj, rater, replicate, plots = "boxplot",
                 rater.col = c("blue", "red"), names = rep(c("1st rep.", "2nd rep."), 2),
                  xlab = "", main = "Boxplots per rater and replicate")
legend("topright", legend = c("A", "B"), fill = c("blue", "red"))

# non-normal data

# involving the measurements
def.par <- par(no.readonly = TRUE)
par(mfcol = c(4, 2), las = 1)
descriptive.plots(AMLad, mrd, id, met, rep, plots = "histogram.y",
                  ylim = c(0, 120), xlim = c(0, 100),
                  breaks = seq(from = 0, to = 100, by = 2.5))
descriptive.plots(AMLad, mrd, id, met, rep, plots = "qqplot.y",
                  ylim = c(0, 8), xlim = c(-3, 3))
par(def.par)

# inolving the paired-differences
def.par <- par(no.readonly = TRUE)
par(mfrow = c(1, 2), las = 1)
descriptive.plots(AMLad, mrd, id, met, rep, plots = "histogram.d",
                  histogram.d.bw = 2,
                  ylim = c(0, 0.15), breaks = seq(-20, 100, 5),
                  xlab = "Paired-differences", main = "",
                  line.col = "darkred", density.col = "darkred")
descriptive.plots(AMLad, mrd, id, met, rep, plots = "qqplot.d",
                  ylim = c(-2, 8), xlim = c(-4, 4), main = "")
par(def.par)
```

`plot.tdi` *Bland-Altman plot*

Description

This function creates a Bland-Altman plot from Altman and Bland (1983), which is used to evaluate the agreement among the quantitative measures taken by two raters. The plot displays the mean of the measurements from both raters in the x-axis and the differences between the measures taken by the two raters in the y-axis. It can also display the TDI and UB estimates from the call of the function `TDI` as well as the limits of agreement (LoA) from Bland and Altman (1986).

Usage

```
## S3 method for class 'tdi'
plot(
  x,
  tdi = FALSE,
  ub = FALSE,
```

```
  loa = FALSE,
  method = NULL,
  ub.pc = NULL,
  p = NULL,
  loess = FALSE,
  method.col = NULL,
  loa.col = "#c27d38",
  loess.col = "#cd2c35",
  loess.span = 2/3,
  legend = FALSE,
  inset = c(-0.24, 0),
  main = "Bland-Altman plot",
  xlab = "Mean",
  ylab = "Difference",
  xlim = NULL,
  ylim = NULL,
  ...
)
```

Arguments

`x`	input object of class `tdi` resulting from a call of the function `TDI`.
`tdi`	logical indicating whether the ±TDI estimate(s) should be added to the plot as solid lines. The default value is `FALSE`.
`ub`	logical indicating whether the ±UB estimate(s) should be added to the plot as dashed lines. The default value is `FALSE`.
`loa`	logical indicating whether the LoA should be added to the plot as dotted lines. The default value is `FALSE`.
`method`	name of the method(s) for which the TDI or the UB estimates will be added to the plot. If both `tdi` and `ub` are set to `FALSE`, this argument is ignored. This argument is not case-sensitive and is passed to `match.arg`. The default value, `NULL`, indicates that, for the measures specified, all the methods for which the TDI (and/or UB) has been computed in the call of the function `TDI` are to be added to the plot.
`ub.pc`	name of the technique for the estimated UB to be added from the method of Perez-Jaume and Carrasco (2015). Possible values are: `p_db`, `n_db`, `e_db`, `b_db`, `p_cb`, `n_cb`, `e_cb` and `b_cb`. The bootstrap approach (differences or cluster) is indicated with "db" and "cb" and the strategy (based on percentiles, the normal distribution, the empirical method or the BC_a) is indicated with "p", "n", "e" and "b". The default value, `NULL`, indicates that the first estimated UB is to be added to the plot.
`p`	value of the proportion for which the TDI and/or UB (depending on the value of the arguments `tdi` and `ub`) are to be added to the plot. If both `tdi` and `ub` are set to `FALSE`, this argument is ignored.

The default value, `NULL`, indicates that only the first proportion passed to the call of the function `TDI` is to be considered.

`loess`	logical indicating whether a smooth curve computed by `loess.smooth` should be added to the plot as a dotdashed curve. The default value is `FALSE`.
`method.col`	colour palette to be used in the drawing of TDIs and/or UBs. A colour should be indicated for every method asked. It is assumed that the colours are passed in the same order as the methods passed to `method`. If both `tdi` and `ub` are set to `FALSE`, this argument is ignored. The default value, `NULL`, indicates that the following palette should be used: `"#f3df6c"`, `"#9c964a"`, `"#f4b5bd"` and `"#85d4e3"` corresponding to the options `"Choudhary P"`, `"Escaramis et al."`, `"Choudhary NP"` and `"Perez-Jaume and Carrasco"` of `method`, respectively.
`loa.col`	colour to be used in the drawing of the LoA. If `loa` is set to `FALSE`, this argument is ignored. The default value is `"#c27d38"`.
`loess.col`	colour to be used in the drawing of the loess smooth curve. If `loess` is set to `FALSE`, this argument is ignored. The default value is `"#cd2c35"`.
`loess.span`	smoothness parameter for `loess.smooth`. The default value is 2/3.
`legend`	logical indicating whether a legend should be added outside the plot. If all `tdi`, `ub` and `loa` are set to `FALSE`, this argument is ignored. The default value is `FALSE`.
`inset`	specifies how far the legend is inset from the plot margins (to be passed to `inset` argument in `legend`). The default value is `c(-0.25, 0)`, recommended for 24” screens with default plot window. For 13” screens, `c(-0.34, 0)` is recommended.
`main`	overall title for the plot (to be passed to `main` argument in `plot`). The default value is `"Bland-Altman plot"`.
`xlab`	a label for the x-axis (to be passed to `xlab` argument in `plot`). The default value is `"Mean"`.
`ylab`	a label for the y-axis (to be passed to `ylab` argument in `plot`). The default value is `"Difference"`.
`xlim`	the x limits of the plot (to be passed to `xlim` argument in `plot`). The default value, `NULL`, indicates that the range of the mean values should be used.
`ylim`	the y limits of the plot (to be passed to `ylim` argument in `plot`). The default value, `NULL`, indicates that the range of the differences values should be used.
`...`	other graphical parameters (to be passed to `plot`).

Details

The LoA are computed using the formula $\bar{d} \pm z_{1-\frac{\alpha}{2}} \cdot \mathrm{sd}(d)$, where $z_{1-\frac{\alpha}{2}}$ is the $(1-\frac{\alpha}{2})$-th quantile of the standard normal distribution, d is the vector containing the differences between the two raters and $\bar{d}$ represents their mean.

Value

A Bland-Altman plot of the data in `x` with a solid black line at differences $= 0$, with differences considered as first level $-$ second level of the variable `met` in the call of the function `TDI`.

Note

A call to `par` is used in this method. Notice that the arguments `font.lab` and `las` are always set to 2 and 1 respectively. Moreover, if `legend` is `TRUE`, `mar` is set to `c(4, 4, 2, 9)`.

References

Altman, D. G., & Bland, J. M. (1983). Measurement in medicine: the analysis of method comparison studies. Journal of the Royal Statistical Society Series D: The Statistician, 32(3), 307-317.

Bland, J. M., & Altman, D. (1986). Statistical methods for assessing agreement between two methods of clinical measurement. The Lancet, 327(8476), 307-310.

Perez-Jaume, S., & Carrasco, J. L. (2015). A non-parametric approach to estimate the total deviation index for non-normal data. Statistics in Medicine, 34(25), 3318-3335.

Examples

```
# normal data

set.seed(2025)

n <- 100

mu.ind <- rnorm(n, 0, 7)

epsA1 <- rnorm(n, 0, 3)
epsA2 <- rnorm(n, 0, 3)
epsB1 <- rnorm(n, 0, 3)
epsB2 <- rnorm(n, 0, 3)

y_A1 <- 50 + mu.ind + epsA1 # rater A, replicate 1
y_A2 <- 50 + mu.ind + epsA2 # rater A, replicate 2
y_B1 <- 40 + mu.ind + epsB1 # rater B, replicate 1
y_B2 <- 40 + mu.ind + epsB2 # rater B, replicate 2

ex_data <- data.frame(y = c(y_A1, y_A2, y_B1, y_B2),
                      rater = factor(rep(c("A", "B"), each = 2*n)),
                      replicate = factor(rep(rep(1:2, each = n), 2)),
                      subj = factor(rep(1:n, 4)))

tdi <- TDI(ex_data, y, subj, rater, replicate, p = c(0.8, 0.9),
```

```
           method = c("Choudhary P", "Escaramis et al.",
                      "Choudhary NP", "Perez-Jaume and Carrasco"),
           boot.type = "cluster", R = 1000)
plot(tdi)

# enhance plot
plot(tdi, xlim = c(20, 70), ylim = c(-20, 30), tdi = TRUE, ub = TRUE,
     method = c("es", "pe"), ub.pc = "b_cb", loa = TRUE, loa.col = "red",
     legend = TRUE)


# non-normal data

tdi.aml <- TDI(AMLad, mrd, id, met, rep, p = c(0.85, 0.95), boot.type = "cluster",
               dec.est = 4, R = 1000)
plot(tdi.aml)

# enhance plot
plot(tdi.aml, method = c("choudhary p", "pe"), tdi = TRUE, ub = TRUE, legend = TRUE,
     main = "Bland-Altman plot of the MRD")
```

`print.tdi` *Printing tdi objects*

Description

A nice gt table containing the values computed with the function TDI.

Usage

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

Arguments

`x`	input object of class `tdi` resulting from a call of the function `TDI`.
`...`	currently not in use

Value

A nice **gt** table containing the values computed with the function `TDI`. The number of decimals for the estimates and the proportions correspond to the arguments `dec.est` and `dec.p` of the function `TDI`, respectively.

Examples

```
# normal data

set.seed(2025)

n <- 100

mu.ind <- rnorm(n, 0, 7)

epsA1 <- rnorm(n, 0, 3)
epsA2 <- rnorm(n, 0, 3)
epsB1 <- rnorm(n, 0, 3)
epsB2 <- rnorm(n, 0, 3)

y_A1 <- 50 + mu.ind + epsA1 # rater A, replicate 1
y_A2 <- 50 + mu.ind + epsA2 # rater A, replicate 2
y_B1 <- 40 + mu.ind + epsB1 # rater B, replicate 1
y_B2 <- 40 + mu.ind + epsB2 # rater B, replicate 2

ex_data <- data.frame(y = c(y_A1, y_A2, y_B1, y_B2),
                      rater = factor(rep(c("A", "B"), each = 2*n)),
                      replicate = factor(rep(rep(1:2, each = n), 2)),
                      subj = factor(rep(1:n, 4)))

tdi <- TDI(ex_data, y, subj, rater, replicate, p = c(0.8, 0.9),
           method = c("Choudhary P", "Escaramis et al.",
                      "Choudhary NP", "Perez-Jaume and Carrasco"),
           boot.type = "cluster", R = 1000)
tdi


# non-normal data

tdi.aml <- TDI(AMLad, mrd, id, met, rep, p = c(0.85, 0.95), boot.type = "cluster",
               dec.est = 4, R = 1000)
tdi.aml
```

`TDI` *TDI estimation and inference*

Description

This function implements the estimation of the TDI and its corresponding $100(1-\alpha)\%$ upper bound (UB), where α is the significance level, using the methods proposed by Choudhary (2007), Escaramis *et al.* (2010), Choudhary (2010) and Perez-Jaume and Carrasco (2015) in the case of two raters. See **Details** and **References** for further information about these methods.

Usage

```
TDI(data, y, id, met, rep = NA,
    method = c("Choudhary P", "Escaramis et al.",
               "Choudhary NP", "Perez-Jaume and Carrasco"),
    p = 0.9, ub = TRUE, boot.type = c("differences", "cluster"),
    type = 8, R = 10000, dec.p = 2, dec.est = 3,
    choose.model.ch.p = TRUE, var.equal = TRUE,
    choose.model.es = TRUE, int = FALSE, tol = 10^(-8), add.es = NULL,
    alpha = 0.05)
```

Arguments

`data`	name of the dataset, of class `data.frame`, containing at least 3 columns (quantitative measurement, subject effect, rater effect).
`y`	quantitative measurement column name.
`id`	subject effect column name. The corresponding column of `data` must be a factor.
`met`	rater effect column name. The corresponding column of `data` must be a factor.
`rep`	replicate effect column name. When there are no replicates the user should use `rep = NA`. When there are replicates, the corresponding column of data must be a factor. The default value is `NA`.
`method`	name of the method(s) to estimate the TDI and UB. The options are: `"Choudhary P"` (Choudhary, 2007), `"Escaramis et al."` (Escaramis *et al.*, 2010), `"Choudhary NP"` (Choudhary, 2010) and `"Perez-Jaume and Carrasco"` (Perez-Jaume and Carrasco, 2015). This argument is not case-sensitive and is passed to `match.arg`. The default value is `c("Choudhary P", "Escaramis et al.", "Choudhary NP", "Perez-Jaume and Carrasco")`, so all approaches are executed by default.
`p`	a value or vector of the proportion(s) for estimation of the TDI, where $0 < p < 1$. Commonly, $p \geq 0.80$. The default value is 0.90.
`ub`	logical asking whether the UBs should be computed. The default value is `TRUE`.
`boot.type`	name of the bootstrap approach(es) to be used in the method of Perez-Jaume and Carrasco (2015). There are two different options when there are replicates: to bootstrap the vector of the within-subject differences (`"differences"`) or to bootstrap at subject level (`"cluster"`). This is, not all the differences coming from the same subject need to be bootstrapped together in the first one but all the measurements from the same subjects have to be bootstrapped together in the second one. This argument is passed to `match.arg` The default value is `c("differences", "cluster")`, so all approaches are executed by default.
`type`	in the method of Perez-Jaume and Carrasco (2015), a quantile is calculated to obtain the estimation of the TDI. This argument is an integer between 1 and 9 selecting one of the nine quantile algorithms (to be passed to `quantile`). We

	recommend 8 for continuous data and 3 for discrete data. The default value is 8.
`R`	in the method of Perez-Jaume and Carrasco (2015), bootstrap is used for the estimation of the UB. This argument chooses the number of bootstrap replicates (to be passed to `boot`). The default value is 10000.
`dec.p`	number of decimals to display for `p` in the method `print.tdi`. The default value is 2.
`dec.est`	number of decimals to display for the estimates in the method `print.tdi`. Up to 4 decimals. The default value is 3.
`choose.model.ch.p`	in the parametric method of Choudhary (2007) two methods can be fit, one with equal residual homoscedasticity between raters and one with unequal residual homoscedasticity. This argument, if `TRUE`, chooses the model with the minimum AIC. If `FALSE`, the argument `var.equal` must be specified. The default value is `TRUE`.
`var.equal`	logical asking if there is residual homoscedasticity between raters to choose the model in the parametric method of Choudhary (2007). If `choose.model.ch.p` is set to `TRUE`, this argument is ignored. The default value is `TRUE`.
`choose.model.es`	in the method of Escaramis *et al.* (2010) two methods can be fit, one including the subject–rater interaction and one that does not. The model with interaction only applies to data with replicates. This argument, if `TRUE`, chooses the model with the minimum AIC. If `FALSE`, the argument `int` must be specified. The default value is `TRUE`.
`int`	logical asking if there is interaction between subjects and methods to choose the model in the method of Escaramis *et al.* (2010). The model with interaction only applies to data with replicates. If `choose.model.es` is set to `TRUE`, this argument is ignored. The default value is `FALSE`.
`tol`	tolerance to be used in the method of Escaramis *et al.* (2010). The default value is 10^(-8).
`add.es`	name of the columns in `data` that will be added to the model (as fixed effects) of the method of Escaramis *et al.* (2010). It must be passed as a column name or vector of column names. The default value, `NULL`, indicates that no extra variables are to be added in the model.
`alpha`	significance level for inference on the TDI. The default value is 0.05.

Details

The methods of Choudhary (2007) and Escaramis *et al.* (2010) are parametric methods based on linear mixed models that assume normality of the data and linearity between the response and the effects (subjects, raters and random errors). The linear mixed models are fitted using the function `lme` from the `nlme` package. The methods of Choudhary (2010) and Perez-Jaume and Carrasco (2015) are non-parametric methods based on the estimation of quantiles of the absolute value of the differences between raters. Non-parametric methods are recommended when dealing with skewed data or other non-normally distributed data, such as count data. In situations of normality, parametric methods are recommended. See **References** for further details.

Value

An object of class `tdi`, which is a list with five components:

`result` an object of class `data.frame` with the TDI estimates and UBs of the methods specified for every proportion.

`fitted.models` a list with the fitted models of the parametric methods of Choudhary (2007) and Escaramis *et al.* (2010).

`params` a list with the values `dec.est`, `dec.p`, `ub`, `method` and `alpha` to be used in the method `print.tdi` and in the method `plot.tdi`.

`data.long` an object of class `data.frame` with columns y, id, met (and rep if it applies) with the values of the measurement, subject identifiers, rater (and replicate number if it applies) from the original data frame provided.

`data.wide` an object of class `data.frame` with either:

- columns id, y.met1, y.met2 (in the case of no replicates) with the measurements of each method.
- columns id, y.met1.rep1,..., y.met1.rep$m$, y.met2.rep1,..., y.met2.rep$m$, with the measurements of each method and each replicate, where m is the number of replicates.

met1 and met2 correspond to the names of the first and second level of the column met in data, respectively. While rep1,..., rep$m$ correspond to the names of the first,..., m-th level of the column rep in data, respectively.

References

Efron, B., & Tibshirani, R. (1993). An Introduction to the Bootstrap; Chapman and Hall. Inc.: New York, NY, USA, 914.

Lin, L. I. K. (2000). Total deviation index for measuring individual agreement with applications in laboratory performance and bioequivalence. Statistics in Medicine, 19(2):255-270.

Choudhary, P. K. (2007). A tolerance interval approach for assessment of agreement with left censored data. Journal of Biopharmaceutical Statistics, 17(4), 583-594.

Escaramis, G., Ascaso, C., & Carrasco, J. L. (2010). The total deviation index estimated by tolerance intervals to evaluate the concordance of measurement devices. BMC Medical Research Methodology, 10, 1-12.

Choudhary, P. K. (2010). A unified approach for nonparametric evaluation of agreement in method comparison studies. The International Journal of Biostatistics, 6(1).

Perez-Jaume, S., & Carrasco, J. L. (2015). A non-parametric approach to estimate the total deviation index for non-normal data. Statistics in Medicine, 34(25), 3318-3335.

See Also

`print.tdi`, `plot.tdi`

Examples

```
# normal data, parametric methods more suitable

set.seed(2025)

n <- 100

mu.ind <- rnorm(n, 0, 7)

epsA1 <- rnorm(n, 0, 3)
epsA2 <- rnorm(n, 0, 3)
epsB1 <- rnorm(n, 0, 3)
epsB2 <- rnorm(n, 0, 3)

y_A1 <- 50 + mu.ind + epsA1 # rater A, replicate 1
y_A2 <- 50 + mu.ind + epsA2 # rater A, replicate 2
y_B1 <- 40 + mu.ind + epsB1 # rater B, replicate 1
y_B2 <- 40 + mu.ind + epsB2 # rater B, replicate 2

ex_data <- data.frame(y = c(y_A1, y_A2, y_B1, y_B2),
                      rater = factor(rep(c("A", "B"), each = 2*n)),
                      replicate = factor(rep(rep(1:2, each = n), 2)),
                      subj = factor(rep(1:n, 4)))

tdi <- TDI(ex_data, y, subj, rater, replicate, p = c(0.8, 0.9),
           method = c("Choudhary P", "Escaramis et al.",
                      "Choudhary NP", "Perez-Jaume and Carrasco"),
           boot.type = "cluster", R = 1000)

tdi$result
tdi$fitted.models
tdi$data.long
tdi$data.wide


# non-normal data, non-parametric methods more suitable

tdi.aml <- TDI(AMLad, mrd, id, met, rep, p = c(0.85, 0.95), boot.type = "cluster",
               dec.est = 4, R = 1000)
tdi.aml$result
tdi.aml$fitted.models
tdi.aml$data.long
tdi.aml$data.wide
```

Index