

Package ‘jrt’

April 13, 2023

Title Item Response Theory Modeling and Scoring for Judgment Data

Version 1.1.2

Description Psychometric analysis and scoring of judgment data using polytomous Item-Response Theory (IRT) models, as described in Myszkowski and Storme (2019) <doi:10.1037/aca0000225> and Myszkowski (2021) <doi:10.1037/aca0000287>. The package is used to automatically compare and select models, as well as to present a variety of model-based statistics. Plotting functions are used to present category curves, as well as information, reliability and standard error functions.

Depends R (>= 3.5.0), directlabels (>= 2021.1.13)

License GPL-3

Encoding UTF-8

LazyData true

Imports mirt (>= 1.34), psych (>= 2.1.9), irr (>= 0.84.1), methods, utils, stats, dplyr (>= 1.0.7), tidyr (>= 1.1.4), ggplot2 (>= 3.3.5), ggsci (>= 2.9)

RoxygenNote 7.2.3

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Nils Myszkowski [aut, cre]

Maintainer Nils Myszkowski <nilsmyszkowskiscience@gmail.com>

Repository CRAN

Date/Publication 2023-04-13 18:10:02 UTC

R topics documented:

anova,jrt-method	2
info.plot	2
jcc.plot	5
jrt	9
jrt-class	13

ratings	14
ratings_missing	14

Index	15
--------------	-----------

anova, jrt-method	<i>anova method for objects returned by the jrt function.</i>
-------------------	---

Description

anova method for objects returned by the [jrt](#) function.

Usage

```
## S4 method for signature 'jrt'
anova(object)
```

Arguments

object An object returned by [jrt](#).

info.plot	<i>Plot the information function for a judge or for the entire set of judges.</i>
-----------	---

Description

This function returns the Judge Information Function plot from a [jrt](#) object and the judge number. Information can be plotted as such, as reliability or as standard error. The function may also be used for the information of the entire set of judges. This is a wrapper function and adaptation of the `itemplot` function in the package `mirt` (Chalmers, 2012). It also uses the plotting functions of the packages `directlabels` and `ggplot2`.

Usage

```
info.plot(
  jrt.object,
  judge = "all",
  type = "information",
  title = "auto",
  column.names = "auto",
  theta.span = 3.5,
  y.limits = NULL,
  y.line = NULL,
  name.for.y.line = "Threshold",
  y.line.refers.to.secondary.axis = TRUE,
  greyscale = FALSE,
  color.palette = "D3",
```

```

line.type = 1,
line.width = 1,
key.width = 3,
legend.position = "right",
legend.columns = "",
theme = "bw",
text.size = 10,
title.size = text.size + 4,
remove.gridlines = TRUE,
font.family = "sans",
precision = 20,
mirt.object.input = F,
item = NULL
)

```

Arguments

jrt.object	A object of the jrt class (created by the function jrt).
judge	A numerical to indicate which judge to plot. Be careful : If a (Generalized) Rating Scale Model was used, then judges may have been removed for the model to be fitted. Provide "all" to get the information plot for all judges.
type	A character to indicate what to plot on the y axis, "information" for Information, "reliability" for Reliability, or "SE" for standard errors. Alternatively, use "infose" (or ise) to plot information and standard error of measurement in the same plot. Use "inforxx" (or ir) to plot information and reliability in the same plot.
title	A character title for the plot. By default it is created automatically based on the judge number.
column.names	A character to indicate how a judge should be called (Defaults to "auto", which uses what was set in the estimation function jrt, whose default is "judge", but you may use "Rater", "Expert", etc.). This is used to create automatic titles.
theta.span	A numeric indicating the maximum θ . The minimum is automatically adjusted to $-\theta$.span.
y.limits	A numeric vector to manually adjust the minimum and maximum of the y axis (may notably be useful if using reliability). Set to NULL (default) to automatically set with ggplot2.
y.line	A numeric to add a (dashed) horizontal line on the plot at the y value indicated (for example for a threshold of acceptable reliability). Defaults to NULL, which does not plot the line. Note that the y level is in reference to the primary axis, if there are two y axes.
name.for.y.line	A character to indicate how to call the y line in the legend. Default is "Threshold".
y.line.refers.to.secondary.axis	A logical to indicate if the y.line should refer to a value on the secondary axis (TRUE, default) or the primary (FALSE). Only used if there is a secondary axis. The default is TRUE because threshold values for interpretation are more often used for reliability or standard error than information.

greyscale	A logical (default is FALSE) to indicate whether to use greyscale graphics (useful for publication). Uses variations in linetype as opposed to variations in line colors.
color.palette	A character value to indicate the colour palette to use. Defaults to "D3" from "ggsci". Use "" for the default of ggplot2. The palettes are supplied as arguments in the scale_fill_brewer() function of ggplot2. In addition, most palettes from the package ggsci are available (e.g., "npg", "aas", "nejm", "lancet", "jama", "d3"). Use vignette("ggsci") for details.
line.type	A numeric indicating the line type for the information function curve (default is 1 for a plain line. This would be used if overlaying multiple plots.
line.width	A numeric indicating the width for the information function curve (default is 2.5).
key.width	A numeric to indicate the width of the legend key (default is 3).
legend.position	A character string or vector of coordinates to position the legend key. Defaults to "right". Other possibilities include notably "bottom".
legend.columns	A numeric to indicate after how many legend key elements to add a line break. Especially useful if using legend.position = "bottom" if you want line breaks between each key. Defaults to "", which automatically saves space based on the legend position (line breaks are used if the legend is positioned on the side of the graph).
theme	A character value to indicate the background color theme used by ggplot2. Defaults to "bw". Can be "light", "dark", "minimal", "classic", "gray", "bw" or "linedraw".
text.size	A numeric value to control the size of the text on the plot.
title.size	A numeric value to control the size of the plot title (defaults to text.size+4).
remove.gridlines	A logical value to remove the gridlines (default is TRUE).
font.family	A character value to control the font family used on the graph. Defaults to "sans". Other possible values include "serif" or "mono".
precision	A numeric to indicate the degree of precision used to plot the curves. Higher values will increase the accuracy of the graph and make the curves look smoother, but the data generated to plot the graph will be bigger, which will slow down the function. Lower values will do the opposite. Values between 10 and 100 are recommended, 20 is the default and sufficient for most uses.
mirt.object.input	A logical allowing to input directly an mirt object as a jrt.object argument, even though this should be detected automatically. See mirt package documentation, and note that this is a secondary use that may lead to inconsistent results at this point.
item	For convenience, this argument, more standard to IRT packages, can be used instead of the judge argument.

Value

A plot of the information, reliability or standard error function.

References

- Chalmers, R., P. (2012). mirt: A Multidimensional Item Response Theory Package for the R Environment. *Journal of Statistical Software*, 48(6), 1-29. doi:10.18637/jss.v048.i06
- Myszkowski, N., & Storme, M. (2019). Judge Response Theory? A call to upgrade our psychometrical account of creativity judgments. *Psychology of Aesthetics, Creativity and the Arts*, 13(2), 167-175. doi:10.1037/aca0000225
- Myszkowski, N. (2021). Development of the R library “jrt”: Automated item response theory procedures for judgment data and their application with the consensual assessment techniques. *Psychology of Aesthetics, Creativity and the Arts*, 15(3), 426-438. doi:10.1037/aca0000287

Examples

```
# Load dataset
data <- jrt::ratings

# Fit model
fit <- jrt(data, irt.model = "PCM")

# Information function of the first judge
info.plot(fit, 1)

# Reliability function of the second judge
info.plot(fit, 2, type = "reliability")

# Standard error function of the entire set of judges
info.plot(fit, "all", type = "SE")

# See vignette for more options
```

jcc.plot

Plot the category curves for a judge.

Description

This function returns the Judge Category Curves (JCC) plot from a `jrt` object and the judge number. This is a wrapper function and adaptation of the `itemplot` function in the package `mirt` (Chalmers, 2012). It also uses the plotting functions of the packages `directlabels` and `ggplot2`.

Usage

```
jcc.plot(
  jrt.object,
  judge = "all",
  labelled = T,
  greyscale = F,
```

```

vertical.labels = F,
title = "auto",
column.names = "auto",
manual.facet.names = "auto",
manual.line.names = "auto",
overlay.reliability = F,
color.palette = "D3",
category.name.for.legend = "",
name.for.reliability = "auto",
theta.span = 3.5,
line.width = 0.8,
line.opacity = 1,
key.width = 3,
legend.position = "right",
legend.columns = "",
theme = "bw",
text.size = 10,
title.size = text.size + 4,
font.family = "sans",
remove.gridlines = T,
facet.rows = NULL,
facet.cols = NULL,
facet.title.position = "top",
precision = 20,
debug = F,
mirt.object.input = F,
item = NULL
)

```

Arguments

<code>jrt.object</code>	A object of the <code>jrt</code> class (created by the function <code>jrt</code>).
<code>judge</code>	A numerical to indicate which judge(s) to plot. Default is <code>all</code> which plots all category curves of all judges. Alternatively, a single integer may be used to plot the JCC for one judge, or a vector of integers to plot multiple judges in a faceted plot. Note that, if a (Generalized) Rating Scale Model was used, then judges may have been removed for the model to be fitted.
<code>labelled</code>	A logical to indicate whether the curves should be labelled with boxed labels (<code>TRUE</code> , default) or whether a legend should be used instead (<code>FALSE</code>). This uses the package <code>directlabels</code> . Note that the rendering is slower (and may take more time to show in GUI) when the plot is labelled.
<code>greyscale</code>	A logical to indicate whether to plot in greyscale (<code>TRUE</code> , default) as opposed to color (<code>FALSE</code>).
<code>vertical.labels</code>	A logical to indicate whether the labels should be vertically oriented (<code>TRUE</code>), as opposed to oriented in the angle of the trace curve (<code>FALSE</code> , the default).
<code>title</code>	A character title for the plot. By default it is created automatically based on the judge number.

<code>column.names</code>	A character to indicate what a column corresponds to (Defaults to "auto", which uses what was set in the estimation function <code>jr t</code> , whose default is "judge", but you may use "Rater", "Expert", "Item", etc.). This is used to create automatic titles.
<code>manual.facet.names</code>	A vector to indicate the names to give to the different facets. Defaults to "auto", which will automatically name them. If not using "auto", the vector length should be equal to the total number of items/judges (not the total in the plot but the total in the dataset).
<code>manual.line.names</code>	A vector to indicate the individual names to give to the different response categories (or different category curves). Defaults to "auto", which names categories from 1 to the number of categories. If not using "auto", the vector supplied should be of the same length as the number of response categories (use the <code>name.for.reliability</code> argument to change it for reliability).
<code>overlay.reliability</code>	A logical to indicate whether to overlay the reliability function of the item (default is FALSE). If overlaid (TRUE), the reliability function will be contrast with the category curves by being in color if the category curves are in blackandwhite, and in black dashed if the category curves are in color.
<code>color.palette</code>	A character value to indicate the colour palette to use. Defaults to "D3" from "ggsci". Use "" for the default of ggplot2. The palettes are supplied as arguments to ggplot2. See here for a list of palettes. In addition, most palettes from the package ggsci are available (e.g., "npg", "aas", "nejm", "lancet", "jama", "d3"). Use <code>vignette("ggsci")</code> for details. Make sure there are enough colors in the palette. Alternatively, you can pass a vector of colors.
<code>category.name.for.legend</code>	A character to indicate how to call categories in the legend. Default to "Category" but for example you may try "Cat." or even "" to save space.
<code>name.for.reliability</code>	A character to indicate a preferred name for reliability in the legend or labels. Defaults to "auto", which adapts to whether labels are used.
<code>theta.span</code>	A numeric indicating the maximum θ .
<code>line.width</code>	A numeric indicating the width of the trace lines (default is 2.5).
<code>line.opacity</code>	A numeric vector to indicate opacities for the different category lines. Defaults to 1. Must be of length equal to the number of categories + 1 (for the reliability line, even if not plotted). For example if there are 5 response categories this vector should be of length 6.
<code>key.width</code>	A numeric to indicate the width of the legend key (default is 3).
<code>legend.position</code>	A character string or vector of coordinates to position the legend key. Defaults to "right". Other possibilities include notably "bottom".
<code>legend.columns</code>	A numeric to indicate after how many legend key elements to add a line break. Especially useful if using <code>legend.position = "bottom"</code> if you want line breaks between each key. Defaults to "", which automatically saves space based on the legend position (line breaks are used if the legend is positioned on the side of the graph).

<code>theme</code>	A character value to indicate the background color theme used by <code>ggplot2</code> . Defaults to "bw". Can be "light", "dark", "minimal", "classic", "gray", "bw" or "linedraw".
<code>text.size</code>	A numeric value to control the size of the text on the plot.
<code>title.size</code>	A numeric value to control the size of the plot title (defaults to <code>text.size+4</code>).
<code>font.family</code>	A character value to control the font family used on the graph. Defaults to "sans". Other possible values include "serif" or "mono".
<code>remove.gridlines</code>	A logical value to remove the gridlines (default is TRUE).
<code>facet.rows</code>	A numeric to change the number of rows for faceted plots. Use this one or <code>facet.cols</code> , not both. Defaults to NULL, which uses <code>ggplot2</code> 's automatic layout.
<code>facet.cols</code>	A numeric to change the number of columns for faceted plots. Use this one or <code>facet.rows</code> , not both. Defaults to NULL, which uses <code>ggplot2</code> 's automatic layout.
<code>facet.title.position</code>	A character string to indicate the position of the facet titles for faceted plots. Defaults to "top", but can be "bottom", "left", or "right".
<code>precision</code>	A numeric to indicate the degree of precision used to plot the category curves. Higher values will increase the accuracy of the graph and make the curves look smoother, but the data generated to plot the graph will be bigger, which will slow down the function. Lower values will do the opposite. Values between 10 and 100 are recommended, 20 is the default and sufficient for most uses.
<code>debug</code>	A logical to report debug messages (used in development). Defaults to FALSE.
<code>mirt.object.input</code>	A logical allowing to input directly an <code>mirt</code> object as a <code>jrt.object</code> argument, even though this should be detected automatically. See <code>mirt</code> package documentation, and note that this is a secondary use that may lead to inconsistent results at this point.
<code>item</code>	For convenience, this argument, more standard to IRT packages, can be used instead of the <code>judge</code> argument.

Value

A plot of the category curves.

References

- Chalmers, R., P. (2012). `mirt`: A Multidimensional Item Response Theory Package for the R Environment. *Journal of Statistical Software*, 48(6), 1-29. doi:10.18637/jss.v048.i06
- Myszkowski, N., & Storme, M. (2019). Judge Response Theory? A call to upgrade our psychometrical account of creativity judgments. *Psychology of Aesthetics, Creativity and the Arts*, 13(2), 167-175. doi:10.1037/aca0000225
- Myszkowski, N. (2021). Development of the R library "jrt": Automated item response theory procedures for judgment data and their application with the consensual assessment techniques. *Psychology of Aesthetics, Creativity and the Arts*, 15(3), 426-438. doi:10.1037/aca0000287

Examples

```
# Load dataset
data <- jrt::ratings

# Fit model
fit <- jrt(data, irt.model = "PCM")

# JCC of the first judge
jcc.plot(fit, 1)

# See vignette for more options
```

jrt	<i>Fit ordinal IRT models on judgment data and return factor scores and statistics.</i>
-----	---

Description

This function automatically selects appropriate polytomous IRT models based on an information criterion (e.g. Corrected AIC), then returns factor scores, standard errors and various IRT psychometric information, as well as more traditional ("CTT") psychometric information. All IRT estimation procedures are executed with the package `mirt` (Chalmers, 2012). The non-IRT procedures use packages `psych` and `irr`.

Usage

```
jrt(
  data,
  irt.model = "auto",
  summary = T,
  selection.criterion = "AIC",
  response.categories = "auto",
  remove.judges.with.unobserved.categories = F,
  additional.stats = F,
  method.factor.scores = "EAP",
  return.mean.scores = T,
  prefix.for.outputs = "Judgments",
  column.names = "Judge",
  maximum.iterations = 2000,
  convergence.threshold = 0.001,
  estimation.algorithm = "EM",
  status.verbose = F,
  estimation.package.warnings = F,
  digits = 3,
  plots = T,
```

```

greyscale = F,
progress.bar = T,
method.item.fit = "X2",
select.variables.that.contain = NULL,
silent = F,
show.calls = F,
debug = F
)

```

Arguments

<code>data</code>	A dataframe or matrix including the judgments to be scored. Note that so far missing data are not supported. This is the only required argument for the function.
<code>irt.model</code>	<p>A string value with the name of the model to fit. It can be:</p> <ul style="list-style-type: none"> • "auto" (default) or NULL : Empirically select the model based on an information criterion (see <code>selection.criterion</code> argument). Difference models (more or less constrained versions of the Graded Response Model) <ul style="list-style-type: none"> – "GRM": Graded Response Model – "CGRM": Constrained Graded Response Model (equal discriminations) – "GrRSM": Graded Rating Scale Model (same category structures) – "CGrRSM": Constrained Graded Rating Scale Model (same category structures and equal discriminations) Divide-by-total models (more or less constrained versions of the Generalized Partial Credit Model) <ul style="list-style-type: none"> – "GPCM": Generalized Partial Credit Model – "PCM": Partial Credit Model (equal discriminations) – "GRSM": Generalized Rating Scale Model (same category structures) – "RSM": Rating Scale Model (same category structures and equal discriminations) <p>For convenience, models can also be called by their full names (e.g. "Generalized Rating Scale Model" or "Generalized Rating Scale" work.)</p> <ul style="list-style-type: none"> • Note: Models where judges are constrained to same category structures (Graded Rating Scale Model, Constrained Graded Rating Scale Model, Generalized Rating Scale Model and Rating Scale Model) cannot be fit if judges have different observed categories. Judges with unobserved categories are automatically removed if these models are called. If the automatic model selection is used, these models are ignored in the comparison by default, but this behavior can be modified to removing judges in the comparison with <code>remove.judges.with.unobserved.categories = T</code>.
<code>summary</code>	A logical to indicate if summary statistics should be displayed as messages (default is TRUE).
<code>selection.criterion</code>	A string with the criterion for the automatic selection. The default is the Akaike Information Criterion (AIC), but other criteria may be used (HQ for Hannan-

Quinn Criterion, BIC for Bayesian Information Criterion and SABIC for Sample-Adjusted Bayesian Information Criterion).

`response.categories`

A numeric vector to indicate the possible score values. For example, use `1:7` for a Likert-type score from 1 to 7. The default, `auto` automatically detects the possible values based on the dataset provided.

`remove.judges.with.unobserved.categories`

A logical value to indicate whether to only keep the judges with all categories observed (based on the `response.categories` argument). The Rating Scale Model (RSM) and Graded Rating Scale Model (GRSM) can only be estimated if the same categories are observed for all judges. If set to `TRUE`, "incomplete judges" are removed only to fit models that require it (RSM and GRSM), and for other models when they are compared to them (to allow meaningful model comparisons). It defaults to `FALSE` to keep all the data available, and has no effect if models that do not require "complete judges" are called.

`additional.stats`

A logical to indicate whether to report other ("non-IRT") reliability statistics (based on computations from packages 'psych' and 'irr'). Defaults to `FALSE`.

`method.factor.scores`

A string to indicate the method used to compute the factor scores. Bayesian methods (EAP, MAP) are recommended. Defaults to Expected A Posteriori (EAP) based on a Standard Normal $N(0, 1)$ prior distribution. Alternatively, Maximum A Posteriori (MAP) with a Standard Normal $N(0, 1)$ prior may be used. Maximum Likelihood (ML) is also possible (it is equivalent to using a uniform prior), but it is discouraged as can produce $-\text{Inf}$ and $+\text{Inf}$ factor scores (for which standard errors will be missing). Alternatively, Weighted Likelihood Estimation (WLE) may be used.

`return.mean.scores`

A logical to indicate whether to return the mean scores in the output (defaults to `TRUE`).

`prefix.for.outputs`

A character used as prefix to name the vectors in the output data frames. Default is "Judgments".

`column.names`

A character to indicate the preferred name to give to a Judge. Defaults to "Judge".

`maximum.iterations`

A numeric indicating the maximum number of iterations used to fit the model (default is 2000).

`convergence.threshold`

A numeric to indicate the threshold used to tolerate convergence (default is `.001`). Reduce for increased precision (but slower or non convergent results).

`estimation.algorithm`

A string indicating the estimation algorithm. Can notably be EM for Bock and Atkin's Expected-Maximization (default) or MHRM for the Metropolis-Hastings Robbins-Monro algorithm (usually slower for unidimensional models).

`status.verbose`

A logical to indicate whether to output messages indicating what the package is doing. Defaults to `FALSE`.

<code>estimation.package.warnings</code>	A logical to indicate whether to output the warnings and messages of the estimation package. Defaults to FALSE for a cleaner output, but set to TRUE if experiencing issues with the estimation.
<code>digits</code>	A numeric to indicate the number of digits to round output statistics by (default is 3).
<code>plots</code>	A logical to indicate whether to plot the total information plot and judge category curves (TRUE, default) or not (FALSE).
<code>greyscale</code>	A logical to indicate whether the plots should be in greyscale (TRUE) or color (FALSE, default).
<code>progress.bar</code>	A logical to indicate whether to show a progress bar during the automatic model selection. Defaults to TRUE.
<code>method.item.fit</code>	A character value to indicate which fit statistic to use for the item fit output. Passed to the <code>itemfit</code> function of the <code>mirt</code> package. Can be <code>S_X2</code> , <code>Zh</code> , <code>X2</code> , <code>G2</code> , <code>PV_Q1</code> , <code>PV_Q1</code> , <code>X2*</code> , <code>X2*_df</code> , <code>infit</code> . Note that some are not be computable if there are missing data.
<code>select.variables.that.contain</code>	A character string to use as data the variables in the original dataset that contain the string. Based on the <code>select</code> function of <code>dplyr</code> . For example, if all your judgment data includes "Rater", use "Rater" to filter your dataset here.
<code>silent</code>	A logical (defaults to FALSE) to ask no output (no message or plot) but the <code>jrt</code> object. This uses other parameters (<code>progress.bar</code> , <code>estimation.package.warnings</code> , <code>plots</code> , <code>summary</code>) in order to return a silent output. Useful if only using the package for factor scoring, for example.
<code>show.calls</code>	A logical to report the calls made to fit the different models. This is meant as a didactic options for users who may be interested in switching over to <code>mirt</code> directly. Defaults to FALSE.
<code>debug</code>	A logical to report debug messages (used in development). Defaults to FALSE.

Value

An object of S4-class `jrt`. The factor scores can be accessed in slot `@output.data`.

References

- Chalmers, R., P. (2012). `mirt`: A Multidimensional Item Response Theory Package for the R Environment. *Journal of Statistical Software*, 48(6), 1-29. doi:10.18637/jss.v048.i06
- Myszkowski, N., & Storme, M. (2019). Judge Response Theory? A call to upgrade our psychometrical account of creativity judgments. *Psychology of Aesthetics, Creativity and the Arts*, 13(2), 167-175. doi:10.1037/aca0000225
- Myszkowski, N. (2021). Development of the R library `jrt`: Automated item response theory procedures for judgment data and their application with the consensual assessment techniques. *Psychology of Aesthetics, Creativity and the Arts*, 15(3), 426-438. doi:10.1037/aca0000287

Examples

```

# Load dataset
data <- jrt::ratings

# Fit models
fit <- jrt(data,
  irt.model = "GRM", # to manually select a model
  plots = FALSE) # to remove plots

# Extract the factor scores
fit@factor.scores # In a dataframe with standard errors
fit@factor.scores.vector # As a numeric vector

# See vignette for more options

```

jrt-class

Object returned by the [jrt](#) function.

Description

Object returned by the [jrt](#) function.

Slots

The original data
 The output data with factor scores.
 The selected model.
 The count of response categories.
 The method used to compute factor scores.
 The data with imputation.
 The factor scores with standard errors as a data.frame.
 The factor scores as a vector.
 The standard errors as a vector.
 The mean scores as a vector.
 The empirical reliability.
 The marginal reliability.
 Tests of item fit.
 Tests of person fit.
 Tests of local dependence.
 The sample size used in the model.
 The number of judges (or items) in the model.
 The name used for the columns.
 The mirt object of the model.

ratings	<i>A simulated dataset with 300 products judged by 6 judges.</i>
---------	--

Description

A simulated dataset with 300 products judged by 6 judges.

Usage

```
ratings
```

Format

A data.frame with 300 rows and 6 columns:

Judge_1 Judgments of judge 1

Judge_2 Judgments of judge 2

Judge_3 Judgments of judge 3

Judge_4 Judgments of judge 4

Judge_5 Judgments of judge 5

Judge_6 Judgments of judge 6

ratings_missing	<i>A simulated dataset with 350 cases judged by 5 judges, using a planned missingness design.</i>
-----------------	---

Description

A simulated dataset with 350 cases judged by 5 judges, using a planned missingness design.

Usage

```
ratings_missing
```

Format

A data.frame with 350 rows (cases) and 5 columns (judges):

Judge_1 Judgments of judge 1

Judge_2 Judgments of judge 2

Judge_3 Judgments of judge 3

Judge_4 Judgments of judge 4

Judge_5 Judgments of judge 5

Index

* datasets

ratings, [14](#)

ratings_missing, [14](#)

anova, jrt-method, [2](#)

info.plot, [2](#)

jcc.plot, [5](#)

jrt, [2](#), [9](#), [13](#)

jrt-class, [13](#)

ratings, [14](#)

ratings_missing, [14](#)