Package: mcqAnalysis (via r-universe)

October 9, 2026

Title Classical Test Theory Item Analysis for Multiple-Choice Tests

Version 0.1.0

Description A unified toolkit for classical test theory (CTT) item analysis of multiple-choice test data, including item difficulty (p-value), item discrimination (point-biserial correlation and upper-lower 27-percent discrimination index), per-distractor analysis (frequency, proportion, and discrimination), and Haladyna's distractor efficiency. A wrapper function returns a tidy 'mcq_analysis' object with print, plot (difficulty-discrimination scatter), and APA-style table methods for direct inclusion in journal manuscripts. Implemented in pure R with no compiled code and minimal dependencies.

License MIT + file LICENSE

URL https://rafhq1403.github.io/mcqAnalysis/, https://github.com/Rafhq1403/mcqAnalysis

BugReports https://github.com/Rafhq1403/mcqAnalysis/issues

Encoding UTF-8

Roxygen list(markdown = TRUE)

Depends R (>= 3.5)

Imports stats, graphics, grDevices

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

LazyData true

Config/testthat/edition 3

Config/roxygen2/version 8.0.0

Repository https://rafhq1403.r-universe.dev

Date/Publication 2026-09-09 21:16:02 UTC

RemoteUrl https://github.com/rafhq1403/mcqanalysis

RemoteRef HEAD

RemoteSha b39d479e0b07d8c1940208be38f38836605c8ed4

Contents

apa_table . . . 2
apa_table.mcq_analysis . . . 3
distractor_analysis . . . 4
distractor_efficiency . . . 5
item_difficulty . . . 6
item_discrimination . . . 7
mcq_analysis . . . 8
mcq_example . . . 9
plot.mcq_analysis . . . 10
point_biserial . . . 12

Index **13**

`apa_table` *Generic APA-style table formatter*

Description

S3 generic for converting analysis objects into publication-ready APA-style tables. The default behavior is dispatched to class-specific methods (e.g., `apa_table.mcq_analysis`). Output formats include data frame, markdown, HTML, and LaTeX for direct inclusion in manuscripts.

Usage

```
apa_table(x, format = c("data.frame", "markdown", "html", "latex"), ...)
```

Arguments

`x`	An object of an appropriate class (e.g., `mcq_analysis`).
`format`	One of `"data.frame"`, `"markdown"`, `"html"`, or `"latex"`.
`...`	Additional arguments passed to methods.

Value

A formatted table object whose type depends on `format`.

`apa_table.mcq_analysis` *APA-style table for an mcq_analysis object*

Description

Formats item-level results from an `mcq_analysis` object as a publication-ready APA-style table, with optional Interpretation columns based on conventional CTT cutoffs (Ebel & Frisbie, 1991).

Usage

```
## S3 method for class 'mcq_analysis'
apa_table(
  x,
  format = c("data.frame", "markdown", "html", "latex"),
  digits = 2,
  include_interpretation = TRUE,
  ...
)
```

Arguments

`x`	An object of class `mcq_analysis`.
`format`	Output format. One of `"data.frame"` (default), `"markdown"`, `"html"`, or `"latex"`.
`digits`	Number of decimal places to display. Default 2.
`include_interpretation`	Logical. If `TRUE` (default), includes columns interpreting difficulty and discrimination using conventional cutoffs.
`...`	Additional arguments passed to `knitr::kable()` for non data-frame formats.

Value

A data frame (when `format = "data.frame"`) or a character string formatted in the requested style.

References

Ebel, R. L., & Frisbie, D. A. (1991). *Essentials of educational measurement* (5th ed.). Prentice Hall.

Examples

```
data(mcq_example)
result <- mcq_analysis(mcq_example$responses, mcq_example$key)
apa_table(result, format = "data.frame")
```

`distractor_analysis` *Distractor analysis*

Description

For each item, summarizes the selection frequency, proportion, and point-biserial correlation with the total test score for every response option (the key and all distractors). Distractor analysis is a core classical test theory diagnostic for evaluating multiple-choice items: the key should be the most-selected option and should have a positive point-biserial correlation with total score, while each distractor should be selected by at least some examinees and should have a negative point-biserial correlation with total score (Haladyna, 2004).

Usage

```
distractor_analysis(responses, key, options = NULL)
```

Arguments

`responses`	A matrix or data frame of student responses, with students in rows and items in columns.
`key`	A vector of correct answers with length equal to the number of items.
`options`	Optional character vector listing all possible response options (e.g., `c("A", "B", "C", "D")`). If `NULL` (default), the set of options is inferred from the unique values present in `responses`.

Value

A data frame in long format with one row per item-option combination, containing:

- `item`: item identifier
- `option`: response option
- `is_key`: logical, `TRUE` if this option is the correct answer
- `frequency`: number of students selecting this option
- `proportion`: proportion of students selecting this option
- `point_biserial`: correlation between selecting this option and the total test score (using all items)

References

Haladyna, T. M. (2004). *Developing and validating multiple-choice test items* (3rd ed.). Lawrence Erlbaum Associates.

Examples

```
set.seed(1)
responses <- matrix(
  sample(c("A", "B", "C", "D"), 200, replace = TRUE),
  nrow = 50, ncol = 4,
  dimnames = list(NULL, paste0("Q", 1:4))
)
key <- c("A", "B", "C", "A")
distractor_analysis(responses, key)
```

`distractor_efficiency` *Distractor efficiency*

Description

Computes Haladyna's distractor efficiency for each item: the number of functioning distractors per item. A distractor is considered to be functioning if it meets two criteria: (a) it is selected by at least a threshold proportion of examinees (default 5 percent), and (b) it has a negative point-biserial correlation with the total test score (Haladyna & Downing, 1993). The key (correct answer) is excluded from the count.

Usage

```
distractor_efficiency(responses, key, options = NULL, min_proportion = 0.05)
```

Arguments

`responses`	A matrix or data frame of student responses, with students in rows and items in columns.
`key`	A vector of correct answers with length equal to the number of items.
`options`	Optional character vector listing all possible response options. If `NULL` (default), the set of options is inferred from the unique values present in `responses`.
`min_proportion`	Minimum proportion of examinees selecting a distractor for it to be considered functioning. Default is 0.05.

Details

Distractor efficiency provides a simple integer summary of item quality. A four-option multiple-choice item with three functioning distractors (distractor efficiency = 3) is performing optimally. Items with fewer functioning distractors waste examinee time and reduce the item's contribution to score variance, and they are candidates for revision.

Value

A named numeric vector of distractor efficiency values, one per item, representing the count of functioning distractors.

References

Haladyna, T. M., & Downing, S. M. (1993). How many options is enough for a multiple-choice test item? *Educational and Psychological Measurement*, 53(4), 999-1010.

Examples

```
set.seed(1)
responses <- matrix(
  sample(c("A", "B", "C", "D"), 400, replace = TRUE),
  nrow = 100, ncol = 4,
  dimnames = list(NULL, paste0("Q", 1:4))
)
key <- c("A", "B", "C", "A")
distractor_efficiency(responses, key)
```

---

item_difficulty *Item difficulty (p-value)*

Description

Computes the proportion of students who answered each item correctly, commonly called the item p-value in classical test theory.

Usage

```
item_difficulty(responses, key, na.rm = FALSE)
```

Arguments

responses	A matrix or data frame of student responses, with students in rows and items in columns. Entries may be character or numeric (e.g., "A", "B", "C", "D" or 1, 2, 3, 4).
key	A vector of correct answers with length equal to the number of items.
na.rm	Logical. If TRUE, missing responses are excluded from the calculation on a per-item basis. If FALSE (default), missing responses are scored as incorrect.

Details

Item difficulty is interpreted as the easiness of an item: values near 1 indicate an easy item (most students got it correct), while values near 0 indicate a hard item. Conventional interpretive guidelines suggest that well-functioning items typically have p-values between 0.30 and 0.90, with optimal difficulty around 0.50 for maximum discrimination (Crocker & Algina, 1986).

Value

A named numeric vector of item p-values, one per item.

References

Crocker, L., & Algina, J. (1986). *Introduction to classical and modern test theory*. Holt, Rinehart and Winston.

Examples

```
responses <- matrix(
  c("A", "A", "B", "C",
    "A", "B", "B", "C",
    "A", "A", "C", "D",
    "B", "A", "B", "C",
    "A", "A", "B", "A"),
  nrow = 5, byrow = TRUE,
  dimnames = list(NULL, c("Q1", "Q2", "Q3", "Q4"))
)
key <- c("A", "A", "B", "C")
item_difficulty(responses, key)
```

---

item_discrimination *Item discrimination*

Description

Computes a discrimination index for each item using one of two classical methods: the point-biserial correlation between item and total test score, or the upper-lower 27 percent discrimination index proposed by Kelley (1939).

Usage

```
item_discrimination(
  responses,
  key,
  method = c("point_biserial", "discrimination_index"),
  group_pct = 0.27
)
```

Arguments

responses	A matrix or data frame of student responses, with students in rows and items in columns.
key	A vector of correct answers with length equal to the number of items.
method	One of "point_biserial" (default) or "discrimination_index".
group_pct	For method = "discrimination_index", the proportion of students assigned to each extreme group. Default is 0.27 following Kelley (1939).

Details

The point-biserial method is the most widely used CTT discrimination index. The discrimination index `D` compares the proportion of the upper-scoring group (top 27 percent by total score) who answered the item correctly to the proportion of the lower-scoring group (bottom 27 percent) who answered it correctly. Kelley (1939) demonstrated that the 27 percent cutoff maximizes the difference between extreme groups under a normal distribution of ability.

Interpretive guidelines for `D` (Ebel & Frisbie, 1991):

- D >= 0.40: very good item
- 0.30 <= D < 0.40: good item, possibly subject to improvement
- 0.20 <= D < 0.30: marginal item, needs improvement
- D < 0.20: poor item, revise or discard

Value

A named numeric vector of discrimination values, one per item.

References

Ebel, R. L., & Frisbie, D. A. (1991). *Essentials of educational measurement* (5th ed.). Prentice Hall.

Kelley, T. L. (1939). The selection of upper and lower groups for the validation of test items. *Journal of Educational Psychology*, 30(1), 17-24.

Examples

```
set.seed(1)
responses <- matrix(
  sample(c("A", "B", "C", "D"), 200, replace = TRUE),
  nrow = 40, ncol = 5,
  dimnames = list(NULL, paste0("Q", 1:5))
)
key <- c("A", "B", "C", "A", "B")
item_discrimination(responses, key)
item_discrimination(responses, key, method = "discrimination_index")
```

`mcq_analysis` *Comprehensive multiple-choice item analysis*

Description

Runs the full classical test theory item analysis on a multiple-choice response matrix and returns a tidy `mcq_analysis` object containing per-item difficulty, discrimination (both point-biserial and the upper-lower 27 percent index), distractor efficiency, and the full per-option distractor analysis. The returned object has dedicated `print()`, `plot()`, and `apa_table()` methods.

Usage

```
mcq_analysis(responses, key, options = NULL, min_proportion = 0.05)
```

Arguments

responses	A matrix or data frame of student responses, with students in rows and items in columns.
key	A vector of correct answers with length equal to the number of items.
options	Optional character vector listing all possible response options. If NULL (default), inferred from the data.
min_proportion	Minimum proportion of examinees selecting a distractor for it to be considered functioning when computing distractor efficiency. Default 0.05.

Value

An object of class mcq_analysis (a list) with components:

items Data frame with one row per item summarizing difficulty, point-biserial, discrimination index, and distractor efficiency.

distractors Data frame with full per-option distractor analysis (one row per item-option combination).

total_scores Numeric vector of total test scores, one per student.

n_students Number of students.

n_items Number of items.

key Answer key.

Examples

```
data(mcq_example)
result <- mcq_analysis(mcq_example$responses, mcq_example$key)
result
```

---

mcq_example *Simulated multiple-choice test data*

Description

A simulated dataset for demonstrating the mcqAnalysis package. The test contains 30 four-option multiple-choice items administered to 200 students. The data are generated under a two-parameter logistic framework with a deliberately mixed mix of item quality:

- Items 1-8 are easy items with strong discrimination.
- Items 9-24 are medium-difficulty items, most discriminating well.
- Items 25-28 are harder items with progressively weaker discrimination.

- Items 29-30 are deliberately badly-written items with negative discrimination (high-ability students get them wrong more often).
- Item 30 additionally has a "trap" distractor disproportionately chosen by high-ability students, useful for demonstrating distractor analysis.

Usage

```
mcq_example
```

Format

A list with two components:

responses A 200 x 30 character matrix of student responses (values in `{"A", "B", "C", "D"}`).

key A named character vector of length 30 giving the correct answer for each item.

Examples

```
data(mcq_example)
str(mcq_example, max.level = 1)
mcq_example$key
head(mcq_example$responses)
```

---

plot.mcq_analysis *Plot a difficulty-discrimination scatter for an mcq_analysis object*

Description

Produces the classical item quality map: a scatterplot of item difficulty (x-axis) against item discrimination (y-axis), with reference lines marking conventional adequacy cutoffs. Items in the upper-middle region (medium difficulty, high discrimination) are performing well; items in the lower regions are candidates for revision.

Usage

```
## S3 method for class 'mcq_analysis'
plot(
  x,
  y = NULL,
  discrimination_metric = c("point_biserial", "discrimination_index"),
  label = c("flagged", "all", "none"),
  flag_threshold_difficulty = c(0.3, 0.9),
  flag_threshold_discrimination = 0.3,
  point_cex = 1.4,
  label_cex = 0.75,
  ...
)
```

Arguments

x	An object of class mcq_analysis.
y	Ignored. Present for S3 compatibility.
discrimination_metric	Which discrimination index to plot on the y-axis. One of "point_biserial" (default) or "discrimination_index".
label	One of "flagged" (default, label only problematic items), "all" (label every item), or "none" (no labels). Also accepts TRUE (= "all") or FALSE (= "none") for backwards compatibility.
flag_threshold_difficulty	Numeric vector of length 2 giving the informative difficulty range. Default c(0.30, 0.90).
flag_threshold_discrimination	Numeric. Discrimination cutoff below which an item is considered weak. Default 0.30.
point_cex	Numeric. Point size. Default 1.4.
label_cex	Numeric. Label text size. Default 0.75.
...	Additional graphical parameters passed to plot().

Details

By default, only flagged items (those falling outside the conventional adequacy region) are labeled, to keep the plot legible when many items cluster in the acceptable region. Use label = "all" to label every item, or label = "none" to suppress labels entirely.

Reference lines are drawn at conventional cutoffs from Ebel and Frisbie (1991): discrimination >= 0.30 (acceptable) and difficulty between 0.30 and 0.90 (informative range).

Value

The input mcq_analysis object, invisibly.

References

Ebel, R. L., & Frisbie, D. A. (1991). *Essentials of educational measurement* (5th ed.). Prentice Hall.

Examples

```
data(mcq_example)
result <- mcq_analysis(mcq_example$responses, mcq_example$key)
plot(result)
plot(result, label = "all")
plot(result, label = "none")
```

`point_biserial` *Point-biserial correlation*

Description

Computes the point-biserial correlation between each item and the total test score (excluding the item itself, i.e., corrected for item overlap). This is the standard classical test theory discrimination index based on the correlation between item performance and overall test performance.

Usage

```
point_biserial(responses, key, corrected = TRUE)
```

Arguments

`responses`	A matrix or data frame of student responses, with students in rows and items in columns.
`key`	A vector of correct answers with length equal to the number of items.
`corrected`	Logical. If `TRUE` (default), the total score is computed excluding the item being correlated, yielding the corrected item-total correlation. If `FALSE`, the total score includes the item.

Details

Items with point-biserial correlations of 0.30 or above are generally considered to discriminate well between high- and low-ability students. Values between 0.20 and 0.29 are marginal; values below 0.20 indicate poor discrimination, and negative values suggest a problem with the item (Ebel & Frisbie, 1991).

Value

A named numeric vector of point-biserial correlations, one per item.

References

Ebel, R. L., & Frisbie, D. A. (1991). *Essentials of educational measurement* (5th ed.). Prentice Hall.

Examples

```
set.seed(1)
responses <- matrix(
  sample(c("A", "B", "C", "D"), 100, replace = TRUE),
  nrow = 20, ncol = 5,
  dimnames = list(NULL, paste0("Q", 1:5))
)
key <- c("A", "B", "C", "A", "B")
point_biserial(responses, key)
```

Index

∗ **datasets**
mcq_example, 9

apa_table, 2
apa_table.mcq_analysis, 3

distractor_analysis, 4
distractor_efficiency, 5

item_difficulty, 6
item_discrimination, 7

mcq_analysis, 8
mcq_example, 9

plot.mcq_analysis, 10
point_biserial, 12