

Package: aberrance (via r-universe)

October 1, 2024

Title Detect Aberrant Behavior in Test Data
Version 0.1.1
Description Detect several types of aberrant behavior, including answer copying, answer similarity, nonparametric misfit, parametric misfit, preknowledge, rapid guessing, and test tampering.
License GPL (>= 3)
URL <https://github.com/kyliegorney/aberrance>
BugReports <https://github.com/kyliegorney/aberrance/issues>
Imports graphics, grDevices, MASS, stats, utils
Encoding UTF-8
RoxygenNote 7.3.2
NeedsCompilation no
Author Kylie Gorney [aut, cph, cre]
(<<https://orcid.org/0000-0002-8924-0726>>), Jiayi Deng [aut, cph] (<<https://orcid.org/0000-0002-1962-2956>>)
Maintainer Kylie Gorney <kgorney@msu.edu>
Repository CRAN
Date/Publication 2024-07-02 21:20:01 UTC

Contents

detect_ac	2
detect_as	5
detect_nm	11
detect_pk	14
detect_pm	18
detect_rg	22
detect_tt	26
sim	30
Index	36

detect_ac	<i>Detect answer copying</i>
-----------	------------------------------

Description

Detect answer copying for all possible source-copier pairs.

Usage

```
detect_ac(
  method,
  psi,
  xi = NULL,
  x = NULL,
  d = NULL,
  r = NULL,
  interval = c(-4, 4),
  alpha = 0.05
)
```

Arguments

method	<p>The answer copying statistic(s) to compute. Options for score-based statistics are:</p> <ul style="list-style-type: none"> • "OMG_S" for the conditional ω statistic (Wollack, 1997). • "GBT_S" for the conditional <i>GBT</i> statistic (van der Linden & Sotaridona, 2006). <p>Options for score and distractor-based statistics are:</p> <ul style="list-style-type: none"> • "OMG_SD" for the conditional ω statistic (Wollack, 1997). • "GBT_SD" for the conditional <i>GBT</i> statistic (van der Linden & Sotaridona, 2006). <p>Options for response-based statistics are:</p> <ul style="list-style-type: none"> • "OMG_R" for the conditional ω statistic (Wollack, 1997). • "GBT_R" for the conditional <i>GBT</i> statistic (van der Linden & Sotaridona, 2006).
psi	A matrix of item parameters.
xi	A matrix of person parameters. If NULL (default), person parameters are estimated using maximum likelihood estimation.
x, d, r	Matrices of raw data. x is for the item scores, d the item distractors, and r the item responses.
interval	The interval to search for the person parameters. Default is c(-4, 4).
alpha	Value(s) between 0 and 1 indicating the significance level(s) used for flagging. Default is 0.05.

Value

A list is returned with the following elements:

stat	A matrix of answer copying statistics.
pval	A matrix of p -values.
flag	An array of flagging results. The first dimension corresponds to source-copier pairs, the second dimension to methods, and the third dimension to significance levels.

References

van der Linden, W. J., & Sotaridona, L. (2006). Detecting answer copying when the regular response process follows a known response model. *Journal of Educational and Behavioral Statistics*, 31(3), 283–304.

Wollack, J. A. (1997). A nominal response model approach for detecting answer copying. *Applied Psychological Measurement*, 21(4), 307–320.

See Also

[detect_as\(\)](#) to detect answer similarity.

Examples

```
# Setup for Examples 1 to 3 -----

# Settings
set.seed(0)      # seed for reproducibility
N <- 50          # number of persons
n <- 40          # number of items

# Randomly select 10% sources and 10% copiers
s <- sample(1:N, size = N * 0.10)
c <- sample(setdiff(1:N, s), size = N * 0.10)

# Create vector of indicators (1 = copying pair, 0 = non-copying pair)
pair <- t(combn(N, 2))
pair <- rbind(pair, pair[, 2:1])
ind <- ifelse(1:nrow(pair) %in% apply(
  rbind(cbind(s, c), cbind(c, s)), 1, function(p)
  which(pair[, 1] == p[1] & pair[, 2] == p[2])), 1, 0)
names(ind) <- paste(pair[, 1], pair[, 2], sep = "-")

# Example 1: Item Scores -----

# Generate person parameters for the 3PL model
xi <- cbind(theta = rnorm(N, mean = 0.00, sd = 1.00))

# Generate item parameters for the 3PL model
psi <- cbind(
  a = rlnorm(n, meanlog = 0.00, sdlog = 0.25),
```

```

    b = rnorm(n, mean = 0.00, sd = 1.00),
    c = runif(n, min = 0.05, max = 0.30)
  )

  # Simulate uncontaminated data
  x <- sim(psi, xi)$x

  # Modify contaminated data by replacing 40% of the copier scores with source
  # scores
  for (v in 1:length(c)) {
    ci <- sample(1:n, size = n * 0.40)
    x[c[v], ci] <- x[s[v], ci]
  }

  # Detect answer copying
  out <- detect_ac(
    method = c("OMG_S", "GBT_S"),
    psi = psi,
    x = x
  )

  # Example 2: Item Scores and Distractors -----

  # Generate person parameters for the nested logit model
  xi <- MASS::mvrnorm(
    N,
    mu = c(theta = 0.00, eta = 0.00),
    Sigma = matrix(c(1.00, 0.80, 0.80, 1.00), ncol = 2)
  )

  # Generate item parameters for the nested logit model
  psi <- cbind(
    a = rlnorm(n, meanlog = 0.00, sdlog = 0.25),
    b = rnorm(n, mean = 0.00, sd = 1.00),
    c = runif(n, min = 0.05, max = 0.30),
    lambda1 = rnorm(n, mean = 0.00, sd = 1.00),
    lambda2 = rnorm(n, mean = 0.00, sd = 1.00),
    lambda3 = rnorm(n, mean = 0.00, sd = 1.00),
    zeta1 = rnorm(n, mean = 0.00, sd = 1.00),
    zeta2 = rnorm(n, mean = 0.00, sd = 1.00),
    zeta3 = rnorm(n, mean = 0.00, sd = 1.00)
  )

  # Simulate uncontaminated data
  dat <- sim(psi, xi)
  x <- dat$x
  d <- dat$d

  # Modify contaminated data by replacing 40% of the copier scores and
  # distractors with source scores and distractors
  for (v in 1:length(c)) {
    ci <- sample(1:n, size = n * 0.40)
    x[c[v], ci] <- x[s[v], ci]
  }

```

```

    d[c[v], ci] <- d[s[v], ci]
  }

  # Detect answer copying
  out <- detect_ac(
    method = c("OMG_S", "GBT_S", "OMG_SD", "GBT_SD"),
    psi = psi,
    x = x,
    d = d
  )

  # Example 3: Item Responses -----

  # Generate person parameters for the nominal response model
  xi <- cbind(eta = rnorm(N, mean = 0.00, sd = 1.00))

  # Generate item parameters for the nominal response model
  psi <- cbind(
    lambda1 = rnorm(n, mean = -0.50, sd = 0.50),
    lambda2 = rnorm(n, mean = -0.50, sd = 0.50),
    lambda3 = rnorm(n, mean = -0.50, sd = 0.50),
    lambda4 = rnorm(n, mean = 1.50, sd = 0.50),
    zeta1 = rnorm(n, mean = -0.50, sd = 0.50),
    zeta2 = rnorm(n, mean = -0.50, sd = 0.50),
    zeta3 = rnorm(n, mean = -0.50, sd = 0.50),
    zeta4 = rnorm(n, mean = 1.50, sd = 0.50)
  )

  # Simulate uncontaminated data
  r <- sim(psi, xi)$r

  # Modify contaminated data by replacing 40% of the copier responses with
  # source responses
  for (v in 1:length(c)) {
    ci <- sample(1:n, size = n * 0.40)
    r[c[v], ci] <- r[s[v], ci]
  }

  # Detect answer copying
  out <- detect_ac(
    method = c("OMG_R", "GBT_R"),
    psi = psi,
    r = r
  )

```

detect_as

Detect answer similarity

Description

Detect answer similarity for all possible pairs.

Usage

```
detect_as(
  method,
  psi,
  xi = NULL,
  x = NULL,
  d = NULL,
  r = NULL,
  y = NULL,
  interval = c(-4, 4),
  alpha = 0.05
)
```

Arguments

method	<p>The answer similarity statistic(s) to compute. Options for score-based statistics are:</p> <ul style="list-style-type: none"> • "OMG_S" for the unconditional ω statistic (Romero et al., 2015). • "GBT_S" for the unconditional <i>GBT</i> statistic (van der Linden & Sotaridona, 2006). • "M4_S" for the <i>M4</i> statistic (Maynes, 2014). <p>Options for score and distractor-based statistics are:</p> <ul style="list-style-type: none"> • "OMG_SD" for the unconditional ω statistic (Romero et al., 2015). • "GBT_SD" for the unconditional <i>GBT</i> statistic (van der Linden & Sotaridona, 2006). • "M4_SD" for the <i>M4</i> statistic (Maynes, 2014). <p>Options for response-based statistics are:</p> <ul style="list-style-type: none"> • "OMG_R" for the unconditional ω statistic (Romero et al., 2015). • "GBT_R" for the unconditional <i>GBT</i> statistic (van der Linden & Sotaridona, 2006). • "M4_R" for the <i>M4</i> statistic (Maynes, 2014). <p>Options for score and response time-based statistics are:</p> <ul style="list-style-type: none"> • "OMG_ST" for the unconditional ω statistic (Gorney & Wollack, 2024). • "GBT_ST" for the unconditional <i>GBT</i> statistic (Gorney & Wollack, 2024). <p>Options for score, distractor, and response time-based statistics are:</p> <ul style="list-style-type: none"> • "OMG_SDT" for the unconditional ω statistic (Gorney & Wollack, 2024). • "GBT_SDT" for the unconditional <i>GBT</i> statistic (Gorney & Wollack, 2024). <p>Options for response and response time-based statistics are:</p> <ul style="list-style-type: none"> • "OMG_RT" for the unconditional ω statistic (Gorney & Wollack, 2024). • "GBT_RT" for the unconditional <i>GBT</i> statistic (Gorney & Wollack, 2024).
psi	A matrix of item parameters.
xi	A matrix of person parameters. If NULL (default), person parameters are estimated using maximum likelihood estimation.

x, d, r, y	Matrices of raw data. x is for the item scores, d the item distractors, r the item responses, and y the item log response times.
interval	The interval to search for the person parameters. Default is $c(-4, 4)$.
alpha	Value(s) between 0 and 1 indicating the significance level(s) used for flagging. Default is 0.05.

Value

A list is returned with the following elements:

stat	A matrix of answer similarity statistics.
pval	A matrix of p -values.
flag	An array of flagging results. The first dimension corresponds to pairs, the second dimension to methods, and the third dimension to significance levels.

References

- Gorney, K., & Wollack, J. A. (2024). Using response times in answer similarity analysis. *Journal of Educational and Behavioral Statistics*. Advance online publication.
- Maynes, D. (2014). Detection of non-independent test taking by similarity analysis. In N. M. Kingston & A. K. Clark (Eds.), *Test fraud: Statistical detection and methodology* (pp. 53–80). Routledge.
- Romero, M., Riascos, Á., & Jara, D. (2015). On the optimality of answer-copying indices: Theory and practice. *Journal of Educational and Behavioral Statistics*, 40(5), 435–453.
- van der Linden, W. J., & Sotaridona, L. (2006). Detecting answer copying when the regular response process follows a known response model. *Journal of Educational and Behavioral Statistics*, 31(3), 283–304.

See Also

[detect_ac\(\)](#) to detect answer copying.

[detect_pk\(\)](#) to detect preknowledge.

Examples

```
# Setup for Examples 1 and 2 -----

# Settings
set.seed(0)      # seed for reproducibility
N <- 50          # number of persons
n <- 40          # number of items

# Randomly select 10% examinees with preknowledge and 40% compromised items
cv <- sample(1:N, size = N * 0.10)
ci <- sample(1:n, size = n * 0.40)

# Create vector of indicators (1 = similar pair, 0 = non-similar pair)
pair <- t(combn(N, 2))
```

```

ind <- ifelse((pair[, 1] %in% cv) & (pair[, 2] %in% cv), 1, 0)
names(ind) <- paste(pair[, 1], pair[, 2], sep = "-")

# Example 1: Item Scores and Response Times -----

# Generate person parameters for the 3PL model and lognormal model
xi <- MASS::mvrnorm(
  N,
  mu = c(theta = 0.00, tau = 0.00),
  Sigma = matrix(c(1.00, 0.25, 0.25, 0.25), ncol = 2)
)

# Generate item parameters for the 3PL model and lognormal model
psi <- cbind(
  a = rlnorm(n, meanlog = 0.00, sdlog = 0.25),
  b = NA,
  c = runif(n, min = 0.05, max = 0.30),
  alpha = runif(n, min = 1.50, max = 2.50),
  beta = NA
)

# Generate positively correlated difficulty and time intensity parameters
psi[, c("b", "beta")] <- MASS::mvrnorm(
  n,
  mu = c(b = 0.00, beta = 3.50),
  Sigma = matrix(c(1.00, 0.20, 0.20, 0.15), ncol = 2)
)

# Simulate uncontaminated data
dat <- sim(psi, xi)
x <- dat$x
y <- dat$y

# Modify contaminated data by changing the item scores and reducing the log
# response times
x[cv, ci] <- rbinom(length(cv) * length(ci), size = 1, prob = 0.90)
y[cv, ci] <- y[cv, ci] * 0.75

# Detect answer similarity
out <- detect_as(
  method = c("OMG_S", "GBT_S", "OMG_ST", "GBT_ST"),
  psi = psi,
  x = x,
  y = y
)

# Example 2: Polytomous Item Scores -----

# Generate person parameters for the generalized partial credit model
xi <- cbind(theta = rnorm(N, mean = 0.00, sd = 1.00))

# Generate item parameters for the generalized partial credit model
psi <- cbind(

```



```

    a = rlnorm(n, meanlog = 0.00, sdlog = 0.25),
    c0 = 0,
    c1 = rnorm(n, mean = -1.00, sd = 0.50),
    c2 = rnorm(n, mean = 0.00, sd = 0.50),
    c3 = rnorm(n, mean = 1.00, sd = 0.50)
  )

# Simulate uncontaminated data
x <- sim(psi, xi)$x

# Modify contaminated data by changing the item scores to the maximum score
x[cv, ci] <- 3

# Detect answer similarity
out <- detect_as(
  method = c("OMG_S", "GBT_S"),
  psi = psi,
  x = x
)

# Setup for Examples 3 and 4 -----

# Settings
set.seed(0)      # seed for reproducibility
N <- 50           # number of persons
n <- 40           # number of items

# Randomly select 10% sources and 10% copiers
s <- sample(1:N, size = N * 0.10)
c <- sample(setdiff(1:N, s), size = N * 0.10)

# Create vector of indicators (1 = similar pair, 0 = non-similar pair)
pair <- t(combn(N, 2))
ind <- ifelse(1:nrow(pair) %in% apply(
  rbind(cbind(s, c), cbind(c, s)), 1, function(p)
  which(pair[, 1] == p[1] & pair[, 2] == p[2])), 1, 0)
names(ind) <- paste(pair[, 1], pair[, 2], sep = "-")

# Example 3: Item Scores and Distractors -----

# Generate person parameters for the nested logit model
xi <- MASS::mvrnorm(
  N,
  mu = c(theta = 0.00, eta = 0.00),
  Sigma = matrix(c(1.00, 0.80, 0.80, 1.00), ncol = 2)
)

# Generate item parameters for the nested logit model
psi <- cbind(
  a = rlnorm(n, meanlog = 0.00, sdlog = 0.25),
  b = rnorm(n, mean = 0.00, sd = 1.00),
  c = runif(n, min = 0.05, max = 0.30),
  lambda1 = rnorm(n, mean = 0.00, sd = 1.00),

```

```

    lambda2 = rnorm(n, mean = 0.00, sd = 1.00),
    lambda3 = rnorm(n, mean = 0.00, sd = 1.00),
    zeta1 = rnorm(n, mean = 0.00, sd = 1.00),
    zeta2 = rnorm(n, mean = 0.00, sd = 1.00),
    zeta3 = rnorm(n, mean = 0.00, sd = 1.00)
  )

  # Simulate uncontaminated data
  dat <- sim(psi, xi)
  x <- dat$x
  d <- dat$d

  # Modify contaminated data by replacing 40% of the copier scores and
  # distractors with source scores and distractors
  for (v in 1:length(c)) {
    ci <- sample(1:n, size = n * 0.40)
    x[c[v], ci] <- x[s[v], ci]
    d[c[v], ci] <- d[s[v], ci]
  }

  # Detect answer similarity
  out <- detect_as(
    method = c("OMG_S", "GBT_S", "OMG_SD", "GBT_SD"),
    psi = psi,
    x = x,
    d = d
  )

  # Example 4: Item Responses -----

  # Generate person parameters for the nominal response model
  xi <- cbind(eta = rnorm(N, mean = 0.00, sd = 1.00))

  # Generate item parameters for the nominal response model
  psi <- cbind(
    lambda1 = rnorm(n, mean = -0.50, sd = 0.50),
    lambda2 = rnorm(n, mean = -0.50, sd = 0.50),
    lambda3 = rnorm(n, mean = -0.50, sd = 0.50),
    lambda4 = rnorm(n, mean = 1.50, sd = 0.50),
    zeta1 = rnorm(n, mean = -0.50, sd = 0.50),
    zeta2 = rnorm(n, mean = -0.50, sd = 0.50),
    zeta3 = rnorm(n, mean = -0.50, sd = 0.50),
    zeta4 = rnorm(n, mean = 1.50, sd = 0.50)
  )

  # Simulate uncontaminated data
  r <- sim(psi, xi)$r

  # Modify contaminated data by replacing 40% of the copier responses with
  # source responses
  for (v in 1:length(c)) {
    ci <- sample(1:n, size = n * 0.40)
    r[c[v], ci] <- r[s[v], ci]
  }

```

```

}

# Detect answer similarity
out <- detect_as(
  method = c("OMG_R", "GBT_R"),
  psi = psi,
  r = r
)

```

detect_nm

Detect nonparametric misfit

Description

Detect nonparametric misfit using person-fit statistics.

Usage

```
detect_nm(method, x = NULL, y = NULL)
```

Arguments

method

The person-fit statistic(s) to compute. Options for score-based statistics are:

- "G_S" for the number of Guttman errors (Guttman, 1944; see also Molenaar, 1991).
- "NC_S" for the norm conformity index (Tatsuoka & Tatsuoka, 1983). *Note:* This statistic cannot be computed for polytomous item scores.
- "U1_S" for the $U1$ statistic, also known as the G^* statistic (van der Flier, 1977; see also Emons, 2008).
- "U3_S" for the $U3$ statistic (van der Flier, 1982; see also Emons, 2008).
- "ZU3_S" for the $ZU3$ statistic (van der Flier, 1982). *Note:* This statistic cannot be computed for polytomous item scores.
- "A_S" for the agreement index (Kane & Brennan, 1980). *Note:* This statistic cannot be computed for polytomous item scores.
- "D_S" for the disagreement index (Kane & Brennan, 1980). *Note:* This statistic cannot be computed for polytomous item scores.
- "E_S" for the dependability index (Kane & Brennan, 1980). *Note:* This statistic cannot be computed for polytomous item scores.
- "C_S" for the caution index (Sato, 1975). *Note:* This statistic cannot be computed for polytomous item scores.
- "MC_S" for the modified caution index, also known as the C^* statistic (Harnisch & Linn, 1981). *Note:* This statistic cannot be computed for polytomous item scores.
- "PC_S" for the personal point-biserial correlation (Donlon & Fischer, 1968). *Note:* This statistic cannot be computed for polytomous item scores.

- "HT_S for the H^T statistic (Sijtsma, 1986). *Note:* This statistic cannot be computed for polytomous item scores.

Options for response time-based statistics are:

- "KL_T" for the Kullback-Leibler divergence (Man et al., 2018).

x, y Matrices of raw data. x is for the item scores and y the item log response times.

Value

A list is returned with the following elements:

stat A matrix of nonparametric person-fit statistics.

References

- Donlon, T. F., & Fischer, F. E. (1968). An index of an individual's agreement with group-determined item difficulties. *Educational and Psychological Measurement*, 28(1), 105–113.
- Emons, W. H. M. (2008). Nonparametric person-fit analysis of polytomous item scores. *Applied Psychological Measurement*, 32(3), 224–247.
- Guttman, L. (1944). A basis for scaling qualitative data. *American Sociological Review*, 9(2), 139–150.
- Harnisch, D. L., & Linn, R. L. (1981). Analysis of item response patterns: Questionable test data and dissimilar curriculum practices. *Journal of Educational Measurement*, 18(3), 133–146.
- Kane, M. T., & Brennan, R. L. (1980). Agreement coefficients as indices of dependability for domain referenced tests. *Applied Psychological Measurement*, 4(1), 105–126.
- Man, K., Harring, J. R., Ouyang, Y., & Thomas, S. L. (2018). Response time based nonparametric Kullback-Leibler divergence measure for detecting aberrant test-taking behavior. *International Journal of Testing*, 18(2), 155–177.
- Molenaar, I. W. (1991). A weighted Loevinger H-coefficient extending Mokken scaling to multi-category items. *Kwantitatieve Methoden*, 12(37), 97–117.
- Sato, T. (1975). *The construction and interpretation of S-P tables*.
- Sijtsma, K. (1986). A coefficient of deviance of response patterns. *Kwantitatieve Methoden*, 7(22), 131–145.
- Tatsuoka, K. K., & Tatsuoka, M. M. (1983). Spotting erroneous rules of operation by the individual consistency index. *Journal of Educational Measurement*, 20(3), 221–230.
- van der Flier, H. (1977) Environmental factors and deviant response patterns. In Y. H. Poortinga (Ed.), *Basic problems in cross-cultural psychology*. Swets & Zeitlinger Publishers.
- van der Flier, H. (1982). Deviant response patterns and comparability of test scores. *Journal of Cross-Cultural Psychology*, 13(3), 267–298.

See Also

[detect_pm\(\)](#) to detect parametric misfit.

Examples

```

# Setup for Examples 1 to 3 -----

# Settings
set.seed(0)      # seed for reproducibility
N <- 500          # number of persons
n <- 40           # number of items

# Randomly select 10% examinees with preknowledge and 40% compromised items
cv <- sample(1:N, size = N * 0.10)
ci <- sample(1:n, size = n * 0.40)

# Create vector of indicators (1 = misfitting, 0 = fitting)
ind <- ifelse(1:N %in% cv, 1, 0)

# Example 1: Dichotomous Item Scores -----

# Generate person parameters for the 3PL model
xi <- cbind(theta = rnorm(N, mean = 0.00, sd = 1.00))

# Generate item parameters for the 3PL model
psi <- cbind(
  a = rlnorm(n, meanlog = 0.00, sdlog = 0.25),
  b = rnorm(n, mean = 0.00, sd = 1.00),
  c = runif(n, min = 0.05, max = 0.30)
)

# Simulate uncontaminated data
x <- sim(psi, xi)$x

# Modify contaminated data by changing the item scores
x[cv, ci] <- rbinom(length(cv) * length(ci), size = 1, prob = 0.90)

# Detect nonparametric misfit
out <- detect_nm(
  method = c("G_S", "NC_S", "U1_S", "U3_S", "ZU3_S", "A_S", "D_S", "E_S",
             "C_S", "MC_S", "PC_S", "HT_S"),
  x = x
)

# Example 2: Polytomous Item Scores -----

# Generate person parameters for the generalized partial credit model
xi <- cbind(theta = rnorm(N, mean = 0.00, sd = 1.00))

# Generate item parameters for the generalized partial credit model
psi <- cbind(
  a = rlnorm(n, meanlog = 0.00, sdlog = 0.25),
  c0 = 0,
  c1 = rnorm(n, mean = -1.00, sd = 0.50),
  c2 = rnorm(n, mean = 0.00, sd = 0.50),
  c3 = rnorm(n, mean = 1.00, sd = 0.50)
)

```

```

)

# Simulate uncontaminated data
x <- sim(psi, xi)$x

# Modify contaminated data by changing the item scores to the maximum score
x[cv, ci] <- 3

# Detect nonparametric misfit
out <- detect_nm(
  method = c("G_S", "U1_S", "U3_S"),
  x = x
)

# Example 3: Item Response Times -----

# Generate person parameters for the lognormal model
xi <- cbind(tau = rnorm(N, mean = 0.00, sd = sqrt(0.25)))

# Generate item parameters for the lognormal model
psi <- cbind(
  alpha = runif(n, min = 1.50, max = 2.50),
  beta = rnorm(n, mean = 3.50, sd = sqrt(0.15))
)

# Simulate uncontaminated data
y <- sim(psi, xi)$y

# Modify contaminated data by reducing the log response times
y[cv, ci] <- y[cv, ci] * 0.75

# Detect nonparametric misfit
out <- detect_nm(
  method = "KL_T",
  y = y
)

```

detect_pk

Detect preknowledge

Description

Detect preknowledge under the assumption that the set of compromised items is known.

Usage

```

detect_pk(
  method,
  ci,
  psi,

```

```

xi = NULL,
xi_c = NULL,
xi_s = NULL,
x = NULL,
y = NULL,
interval = c(-4, 4),
alpha = 0.05,
cutoff = 0.05
)

```

Arguments

method	<p>The preknowledge detection statistic(s) to compute. Options for score-based statistics are:</p> <ul style="list-style-type: none"> • "L_S" for the signed likelihood ratio test statistic (Sinharay, 2017). • "ML_S" for the modified signed likelihood ratio test statistic (Sinharay & Jensen, 2019). For numerical stability, an absolute cutoff value can be specified using cutoff. <i>Note:</i> This statistic cannot be computed under the 3PL model or the graded response model. • "LR_S" for the Lugannani-Rice approximation (Sinharay & Jensen, 2019). For numerical stability, an absolute cutoff value can be specified using cutoff. <i>Note:</i> This statistic cannot be computed under the 3PL model or the graded response model. • "S_S" for the signed score test statistic (Sinharay, 2017). • "W_S" for the Wald test statistic (Sinharay & Jensen, 2019). <p>Options for response time-based statistics are:</p> <ul style="list-style-type: none"> • "L_T" for the signed likelihood ratio test statistic, or equivalently, "W_T" for the Wald test statistic (Sinharay, 2020). <p>Options for score and response time-based statistics are:</p> <ul style="list-style-type: none"> • "L_ST" for the constrained likelihood ratio test statistic (Sinharay & Johnson, 2020).
ci	A vector of compromised item positions. All other items are presumed secure.
psi	A matrix of item parameters.
xi, xi_c, xi_s	Matrices of person parameters. xi is based on all items, xi_c is based on the compromised items, and xi_s is based on the secure items. If NULL (default), person parameters are estimated using maximum likelihood estimation.
x, y	Matrices of raw data. x is for the item scores and y the item log response times.
interval	The interval to search for the person parameters. Default is c(-4, 4).
alpha	Value(s) between 0 and 1 indicating the significance level(s) used for flagging. Default is 0.05.
cutoff	Use with the modified signed likelihood ratio test statistic and the Lugannani-Rice approximation. If the absolute value of the signed likelihood ratio test statistic is less than the cutoff (default is 0.05), then the modified signed likelihood ratio test statistic is replaced with the signed likelihood ratio test statistic and the Lugannani-Rice approximation is replaced with the <i>p</i> -value of the signed likelihood ratio test statistic.

Value

A list is returned with the following elements:

stat	A matrix of preknowledge detection statistics.
pval	A matrix of p -values.
flag	An array of flagging results. The first dimension corresponds to persons, the second dimension to methods, and the third dimension to significance levels.

References

Sinharay, S. (2017). Detection of item preknowledge using likelihood ratio test and score test. *Journal of Educational and Behavioral Statistics*, 42(1), 46–68.

Sinharay, S. (2020). Detection of item preknowledge using response times. *Applied Psychological Measurement*, 44(5), 376–392.

Sinharay, S., & Jensen, J. L. (2019). Higher-order asymptotics and its application to testing the equality of the examinee ability over two sets of items. *Psychometrika*, 84(2), 484–510.

Sinharay, S., & Johnson, M. S. (2020). The use of item scores and response times to detect examinees who may have benefited from item preknowledge. *British Journal of Mathematical and Statistical Psychology*, 73(3), 397–419.

See Also

[detect_as\(\)](#) to detect answer similarity.

Examples

```
# Setup for Examples 1 and 2 -----

# Settings
set.seed(0)      # seed for reproducibility
N <- 500         # number of persons
n <- 40          # number of items

# Randomly select 10% examinees with preknowledge and 40% compromised items
cv <- sample(1:N, size = N * 0.10)
ci <- sample(1:n, size = n * 0.40)

# Create vector of indicators (1 = preknowledge, 0 = no preknowledge)
ind <- ifelse(1:N %in% cv, 1, 0)

# Example 1: Item Scores and Response Times -----

# Generate person parameters for the 2PL model and lognormal model
xi <- MASS::mvrnorm(
  N,
  mu = c(theta = 0.00, tau = 0.00),
  Sigma = matrix(c(1.00, 0.25, 0.25, 0.25), ncol = 2)
)
```



```

# Generate item parameters for the 2PL model and lognormal model
psi <- cbind(
  a = rlnorm(n, meanlog = 0.00, sdlog = 0.25),
  b = NA,
  c = 0,
  alpha = runif(n, min = 1.50, max = 2.50),
  beta = NA
)

# Generate positively correlated difficulty and time intensity parameters
psi[, c("b", "beta")] <- MASS::mvrnorm(
  n,
  mu = c(b = 0.00, beta = 3.50),
  Sigma = matrix(c(1.00, 0.20, 0.20, 0.15), ncol = 2)
)

# Simulate uncontaminated data
dat <- sim(psi, xi)
x <- dat$x
y <- dat$y

# Modify contaminated data by changing the item scores and reducing the log
# response times
x[cv, ci] <- rbinom(length(cv) * length(ci), size = 1, prob = 0.90)
y[cv, ci] <- y[cv, ci] * 0.75

# Detect preknowledge
out <- detect_pk(
  method = c("L_S", "ML_S", "LR_S", "S_S", "W_S", "L_T", "L_ST"),
  ci = ci,
  psi = psi,
  x = x,
  y = y
)

# Example 2: Polytomous Item Scores -----

# Generate person parameters for the generalized partial credit model
xi <- cbind(theta = rnorm(N, mean = 0.00, sd = 1.00))

# Generate item parameters for the generalized partial credit model
psi <- cbind(
  a = rlnorm(n, meanlog = 0.00, sdlog = 0.25),
  c0 = 0,
  c1 = rnorm(n, mean = -1.00, sd = 0.50),
  c2 = rnorm(n, mean = 0.00, sd = 0.50),
  c3 = rnorm(n, mean = 1.00, sd = 0.50)
)

# Simulate uncontaminated data
x <- sim(psi, xi)$x

# Modify contaminated data by changing the item scores to the maximum score

```

```

x[cv, ci] <- 3

# Detect preknowledge
out <- detect_pk(
  method = c("L_S", "ML_S", "LR_S", "S_S", "W_S"),
  ci = ci,
  psi = psi,
  x = x
)

```

detect_pm

*Detect parametric misfit***Description**

Detect parametric misfit using person-fit statistics.

Usage

```

detect_pm(
  method,
  psi,
  xi = NULL,
  x = NULL,
  d = NULL,
  r = NULL,
  y = NULL,
  interval = c(-4, 4),
  alpha = 0.05
)

```

Arguments

method

The person-fit statistic(s) to compute. Options for score-based statistics are:

- "ECI2_S_*" for the second standardized extended caution index, also known as ζ_1 (Tatsuoka, 1984; see also Sinharay, 2018b).
- "ECI4_S_*" for the fourth standardized extended caution index, also known as ζ_2 (Tatsuoka, 1984; see also Sinharay, 2018b).
- "L_S_*" for the standardized log-likelihood statistic (Dragow et al., 1985).

Options for distractor-based statistics are:

- "L_D_*" for the standardized log-likelihood statistic (Gorney & Wollack, 2023).

Options for score and distractor-based statistics are:

- "L_SD_*" for the log-likelihood statistic (Gorney & Wollack, 2023).

Options for response-based statistics are:

- "L_R_*" for the standardized log-likelihood statistic (Drasgow et al., 1985).

Options for response time-based statistics are:

- "L_T" for the log-likelihood statistic (Sinharay, 2018a).

Options for score and response time-based statistics are:

- "Q_ST_*" for the log-likelihood statistic that combines "L_S_*" and "L_T" (Gorney, Sinharay, & Liu, 2024).
- "L_ST_*" for the standardized log-likelihood statistic (Gorney, Sinharay, & Liu, 2024). *Note:* This statistic cannot be computed using the "CF", "CS", "EW", "TSCF", "TSCS", or "TSEW" corrections.

Options for response and response time-based statistics are:

- "Q_RT_*" for the log-likelihood statistic that combines "L_R_*" and "L_T" (Gorney, Sinharay, & Liu, 2024).
- "L_RT_*" for the standardized log-likelihood statistic (Gorney, Sinharay, & Liu, 2024). *Note:* This statistic cannot be computed using the "CF", "CS", "EW", "TSCF", "TSCS", or "TSEW" corrections.

Statistics ending in "*" can be computed using various corrections. Options are:

- "*" for all possible corrections.
- "NO" for no correction.
- "CF" for the Cornish-Fisher expansion (Molenaar & Hoijtink, 1990).
- "CS" for the chi-squared approximation (Molenaar & Hoijtink, 1990).
- "EW" for the Edgeworth expansion (Bedrick, 1997).
- "TS" for the Taylor series expansion (Snijders, 2001; see also Sinharay, 2016a, 2016b).
- "TSCF" for the Taylor series expansion and Cornish-Fisher expansion (Gorney, Sinharay, & Eckerly, 2024; see also Gorney, 2024).
- "TSCS" for the Taylor series expansion and chi-squared approximation (Gorney Sinharay, & Eckerly, 2024; see also Gorney, 2024).
- "TSEW" for the Taylor series expansion and Edgeworth expansion (Gorney Sinharay, & Eckerly, 2024; see also Gorney, 2024).

psi	A matrix of item parameters.
xi	A matrix of person parameters. If NULL (default), person parameters are estimated using maximum likelihood estimation.
x, d, r, y	Matrices of raw data. x is for the item scores, d the item distractors, r the item responses, and y the item log response times.
interval	The interval to search for the person parameters. Default is c(-4, 4).
alpha	Value(s) between 0 and 1 indicating the significance level(s) used for flagging. Default is 0.05.

Value

A list is returned with the following elements:

stat	A matrix of parametric person-fit statistics.
pval	A matrix of <i>p</i> -values.
flag	An array of flagging results. The first dimension corresponds to persons, the second dimension to methods, and the third dimension to significance levels.

References

- Bedrick, E. J. (1997). Approximating the conditional distribution of person fit indexes for checking the Rasch model. *Psychometrika*, 62(2), 191–199.
- Drasgow, F., Levine, M. V., & Williams, E. A. (1985). Appropriateness measurement with polychotomous item response models and standardized indices. *British Journal of Mathematical and Statistical Psychology*, 38(1), 67–86.
- Gorney, K. (2024). Three new corrections for standardized person-fit statistics for tests with polytomous items. *British Journal of Mathematical and Statistical Psychology*. Advance online publication.
- Gorney, K., Sinharay, S., & Eckerly, C. (2024). Efficient corrections for standardized person-fit statistics. *Psychometrika*, 89(2), 569–591.
- Gorney, K., Sinharay, S., & Liu, X. (2024). Using item scores and response times in person-fit assessment. *British Journal of Mathematical and Statistical Psychology*, 77(1), 151–168.
- Gorney, K., & Wollack, J. A. (2023). Using item scores and distractors in person-fit assessment. *Journal of Educational Measurement*, 60(1), 3–27.
- Molenaar, I. W., & Hoijtink, H. (1990). The many null distributions of person fit indices. *Psychometrika*, 55(1), 75–106.
- Sinharay, S. (2016a). Asymptotic corrections of standardized extended caution indices. *Applied Psychological Measurement*, 40(6), 418–433.
- Sinharay, S. (2016b). Asymptotically correct standardization of person-fit statistics beyond dichotomous items. *Psychometrika*, 81(4), 992–1013.
- Sinharay, S. (2018a). A new person-fit statistic for the lognormal model for response times. *Journal of Educational Measurement*, 55(4), 457–476.
- Sinharay, S. (2018b). Extension of caution indices to mixed-format tests. *British Journal of Mathematical and Statistical Psychology*, 71(2), 363–386.
- Snijders, T. A. B. (2001). Asymptotic null distribution of person fit statistics with estimated person parameter. *Psychometrika*, 66(3), 331–342.
- Tatsuoka, K. K. (1984). Caution indices based on item response theory. *Psychometrika*, 49(1), 95–110.

See Also

[detect_nm\(\)](#) to detect nonparametric misfit.

Examples

```
# Setup for Examples 1 and 2 -----

# Settings
set.seed(0)      # seed for reproducibility
N <- 500         # number of persons
n <- 40          # number of items

# Randomly select 10% examinees with preknowledge and 40% compromised items
cv <- sample(1:N, size = N * 0.10)
```

```

ci <- sample(1:n, size = n * 0.40)

# Create vector of indicators (1 = misfitting, 0 = fitting)
ind <- ifelse(1:N %in% cv, 1, 0)

# Example 1: Item Scores and Response Times -----

# Generate person parameters for the 3PL model and lognormal model
xi <- MASS::mvrnorm(
  N,
  mu = c(theta = 0.00, tau = 0.00),
  Sigma = matrix(c(1.00, 0.25, 0.25, 0.25), ncol = 2)
)

# Generate item parameters for the 3PL model and lognormal model
psi <- cbind(
  a = rlnorm(n, meanlog = 0.00, sdlog = 0.25),
  b = NA,
  c = runif(n, min = 0.05, max = 0.30),
  alpha = runif(n, min = 1.50, max = 2.50),
  beta = NA
)

# Generate positively correlated difficulty and time intensity parameters
psi[, c("b", "beta")] <- MASS::mvrnorm(
  n,
  mu = c(b = 0.00, beta = 3.50),
  Sigma = matrix(c(1.00, 0.20, 0.20, 0.15), ncol = 2)
)

# Simulate uncontaminated data
dat <- sim(psi, xi)
x <- dat$x
y <- dat$y

# Modify contaminated data by changing the item scores and reducing the log
# response times
x[cv, ci] <- rbinom(length(cv) * length(ci), size = 1, prob = 0.90)
y[cv, ci] <- y[cv, ci] * 0.75

# Detect parametric misfit
out <- detect_pm(
  method = c("L_S_TS", "L_T", "Q_ST_TS", "L_ST_TS"),
  psi = psi,
  x = x,
  y = y
)

# Example 2: Polytomous Item Scores -----

# Generate person parameters for the generalized partial credit model
xi <- cbind(theta = rnorm(N, mean = 0.00, sd = 1.00))

```

```

# Generate item parameters for the generalized partial credit model
psi <- cbind(
  a = rlnorm(n, meanlog = 0.00, sdlog = 0.25),
  c0 = 0,
  c1 = rnorm(n, mean = -1.00, sd = 0.50),
  c2 = rnorm(n, mean = 0.00, sd = 0.50),
  c3 = rnorm(n, mean = 1.00, sd = 0.50)
)

# Simulate uncontaminated data
x <- sim(psi, xi)$x

# Modify contaminated data by changing the item scores to the maximum score
x[cv, ci] <- 3

# Detect parametric misfit
out <- detect_pm(
  method = c("ECI2_S_TSCF", "ECI4_S_TSCF", "L_S_TSCF"),
  psi = psi,
  x = x
)

```

detect_rg

Detect rapid guessing

Description

Detect rapid guessing using item-level response time information.

Usage

```

detect_rg(
  method,
  t,
  x = NULL,
  outlier = 100,
  chance = 0.25,
  thr = 3,
  nt = 10,
  limits = c(0, Inf),
  min_item = 1
)

```

Arguments

method	<p>The rapid guessing detection method to apply. Options for visual inspection methods are:</p> <ul style="list-style-type: none"> "VI" for the visual inspection method (Schnipke, 1995). Each plot contains a histogram of the item response times.
--------	--

- "VITP" for the visual inspection with proportion correct method (Lee & Jia, 2014; Ma et al., 2011). Each plot contains a histogram of the item response times, a dashed red line indicating the proportion correct, and a solid red line indicating the chance rate of success (see chance).

Options for threshold methods are:

- "CT" for the custom threshold method (Wise et al., 2004; Wise & Kong, 2005). The thresholds can be specified using thr.
- "NT" for the normative threshold method (Martinez & Rios, 2023; Wise & Ma, 2012). The percentage(s) of the mean item response time can be specified using nt.

Options for visual inspection and threshold methods are:

- "CUMP" for the cumulative proportion correct method (Guo et al., 2016). Each plot contains a histogram of the item response times, a dashed red line indicating the cumulative proportion correct, and a solid red line indicating the chance rate of success (see chance). *Note:* No thresholds are returned for items for which the cumulative proportion correct is consistently above or below chance.

t, x	Matrices of raw data. t is for the item response times and x the item scores.
outlier	The percentile(s) above which to delete outliers in t. Length must be equal to 1 or equal to the total number of items. Default is 100, for which no response times are identified as outliers to be deleted.
chance	Use with the visual inspection with proportion correct method and the cumulative proportion correct method. Value(s) indicating the chance rate(s) of success. Length must be equal to 1 or equal to the total number of items. Default is 0.25.
thr	Use with the custom threshold method. Value(s) indicating the response time thresholds. Length must be equal to 1 or equal to the total number of items. Default is 3.
nt	Use with the normative threshold method. Value(s) indicating the percentage(s) of the mean item response time to be used as thresholds. If length is equal to 1, one normative threshold is applied to all items (Wise et al., 2004). Else if length is greater than 1, multiple normative thresholds are applied to all items (Martinez & Rios, 2023). Default is 10, for NT10.
limits	Use with threshold methods. A vector of length 2 indicating the minimum and maximum possible thresholds. Default is c(0, Inf).
min_item	The minimum number of items used to identify unmotivated persons. Default is 1.

Value

A list is returned. If a visual inspection method is used, the list contains the following elements:

plots A list containing one plot per item.

If a threshold method is used, the list contains the following elements:

thr A vector or matrix of response time thresholds.

flag	A matrix or array of flagging results.
rte	A vector or matrix of response time effort, equal to 1 minus the proportion of flagged responses per person (Wise & Kong, 2005).
rtf	A vector or matrix of response time fidelity, equal to 1 minus the proportion of flagged responses per item (Wise, 2006).
unmotivated	The proportion of unmotivated persons.

References

- Guo, H., Rios, J. A., Haberman, S., Liu, O. L., Wang, J., & Paek, I. (2016). A new procedure for detection of students' rapid guessing responses using response time. *Applied Measurement in Education*, 29(3), 173–183.
- Lee, Y.-H., & Jia, Y. (2014). Using response time to investigate students' test-taking behaviors in a NAEP computer-based study. *Large-Scale Assessments in Education*, 2, Article 8.
- Ma, L., Wise, S. L., Thum, Y. M., & Kingsbury, G. (2011, April). *Detecting response time threshold under the computer adaptive testing environment* [Paper presentation]. National Council of Measurement in Education, New Orleans, LA, United States.
- Martinez, A. J., & Rios, J. A. (2023, April). *The impact of rapid guessing on model fit and factor-analytic reliability* [Paper presentation]. National Council on Measurement in Education, Chicago, IL, United States.
- Schnipke, D. L. (1995, April). *Assessing speededness in computer-based tests using item response times* [Paper presentation]. National Council on Measurement in Education, San Francisco, CA, United States.
- Wise, S. L. (2006). An investigation of the differential effort received by items on a low-stakes computer-based test. *Applied Measurement in Education*, 19(2), 95–114.
- Wise, S. L., Kingsbury, G. G., Thomason, J., & Kong, X. (2004, April). *An investigation of motivation filtering in a statewide achievement testing program* [Paper presentation]. National Council on Measurement in Education, San Diego, CA, United States.
- Wise, S. L., & Kong, X. (2005). Response time effort: A new measure of examinee motivation in computer-based tests. *Applied Measurement in Education*, 18(2), 163–183.
- Wise, S. L., & Ma, L. (2012, April). *Setting response time thresholds for a CAT item pool: The normative threshold method* [Paper presentation]. National Council on Measurement in Education, Vancouver, BC, Canada.

Examples

```
# Setup for Examples 1 to 3 -----

# Settings
set.seed(0)      # seed for reproducibility
N <- 5000        # number of persons
n <- 40          # number of items

# Randomly select 20% unmotivated persons
cv <- sample(1:N, size = N * 0.20)
```



```

# Create vector of indicators (1 = unmotivated, 0 = motivated)
ind <- ifelse(1:N %in% cv, 1, 0)

# Generate person parameters for the 3PL model and lognormal model
xi <- MASS::mvrnorm(
  N,
  mu = c(theta = 0.00, tau = 0.00),
  Sigma = matrix(c(1.00, 0.25, 0.25, 0.25), ncol = 2)
)

# Generate item parameters for the 3PL model and lognormal model
psi <- cbind(
  a = rlnorm(n, meanlog = 0.00, sdlog = 0.25),
  b = NA,
  c = runif(n, min = 0.05, max = 0.30),
  alpha = runif(n, min = 1.50, max = 2.50),
  beta = NA
)

# Generate positively correlated difficulty and time intensity parameters
psi[, c("b", "beta")] <- MASS::mvrnorm(
  n,
  mu = c(b = 0.00, beta = 3.50),
  Sigma = matrix(c(1.00, 0.20, 0.20, 0.15), ncol = 2)
)

# Simulate item scores and response times
dat <- sim(psi, xi)
x <- dat$x
t <- exp(dat$y)

# Modify contaminated data by guessing on 20% of the items
for (v in cv) {
  ci <- sample(1:n, size = n * 0.20)
  x[v, ci] <- rbinom(length(ci), size = 1, prob = 0.25)
  t[v, ci] <- runif(length(ci), min = 1, max = 10)
}

# Example 1: Visual Inspection Methods -----

# Detect rapid guessing using the visual inspection method
out <- detect_rg(
  method = "VI",
  t = t,
  outlier = 90
)

# Detect rapid guessing using the visual inspection with proportion correct
# method
out <- detect_rg(
  method = "VITP",
  t = t,
  x = x,

```

```

    outlier = 90
  )

# Example 2: Threshold Methods -----

# Detect rapid guessing using the custom threshold method with a common
# three-second threshold
out <- detect_rg(
  method = "CT",
  t = t,
  thr = 3
)

# Detect rapid guessing using the custom threshold method with 10% of the
# median item response time
out <- detect_rg(
  method = "CT",
  t = t,
  thr = apply(t, 2, function(i) 0.10 * median(i))
)

# Detect rapid guessing using the normative threshold method with 10% of the
# mean item response time
out <- detect_rg(
  method = "NT",
  t = t,
  nt = 10
)

# Detect rapid guessing using the normative threshold method with 5 to 35% of
# the mean item response time
out <- detect_rg(
  method = "NT",
  t = t,
  nt = seq(5, 35, by = 5)
)

# Example 3: Visual Inspection and Threshold Methods -----

# Detect rapid guessing using the cumulative proportion correct method
out <- detect_rg(
  method = "CUMP",
  t = t,
  x = x,
  outlier = 90
)

```

Description

Detect test tampering at the person level or at the group level.

Usage

```
detect_tt(
  method,
  psi,
  xi = NULL,
  xi_c = NULL,
  xi_s = NULL,
  x = NULL,
  d = NULL,
  r = NULL,
  x_0 = NULL,
  d_0 = NULL,
  r_0 = NULL,
  interval = c(-4, 4),
  alpha = 0.05,
  group = NULL,
  c = -0.5
)
```

Arguments

method The test tampering statistic(s) to compute. Options for score and distractor-based statistics are:

- "EDI_SD_*" for the erasure detection index (Wollack et al., 2015; Wollack & Eckerly, 2017).
- "GBT_SD" for the generalized binomial test statistic (Sinharay & Johnson, 2017). *Note:* This statistic cannot be computed at the group level.
- "L_SD" for the signed likelihood ratio test statistic (Sinharay et al., 2017). *Note:* This statistic cannot be computed at the group level.

Options for response-based statistics are:

- "EDI_R_*" for the erasure detection index (Wollack et al., 2015; Wollack & Eckerly, 2017).
- "GBT_R" for the generalized binomial test statistic (Sinharay & Johnson, 2017). *Note:* This statistic cannot be computed at the group level.
- "L_R" for the signed likelihood ratio test statistic (Sinharay et al., 2017). *Note:* This statistic cannot be computed at the group level.

Statistics ending in "*" can be computed using various corrections. Options are:

- "*" for all possible corrections.
- "NO" for no correction (Sinharay, 2018; Wollack et al., 2015).
- "CO" for the continuity correction (Wollack et al., 2015; Wollack & Eckerly, 2017). The value of the continuity correction can be specified using *c*.
- "TS" for the Taylor series expansion (Sinharay, 2018).

psi	A matrix of item parameters.
xi, xi_c, xi_s	Matrices of person parameters. xi is based on all items, xi_c is based on items with changed answers, and xi_s is based on items with the same answers. If NULL (default), person parameters are estimated using maximum likelihood estimation.
x, d, r	Matrices of final data. x is for the item scores, d the item distractors, and r the item responses.
x_0, d_0, r_0	Matrices of initial data. x_0 is for the item scores, d_0 the item distractors, and r_0 the item responses.
interval	The interval to search for the person parameters. Default is c(-4, 4).
alpha	Value(s) between 0 and 1 indicating the significance level(s) used for flagging. Default is 0.05.
group	A vector indicating group membership. If NULL (default), statistics are computed at the person level.
c	Use with the erasure detection index. A value indicating the continuity correction. Default is -0.5.

Value

A list is returned with the following elements:

stat	A matrix of test tampering detection statistics.
pval	A matrix of <i>p</i> -values.
flag	An array of flagging results. The first dimension corresponds to persons/groups, the second dimension to methods, and the third dimension to significance levels.

References

- Sinharay, S., Duong, M. Q., & Wood, S. W. (2017). A new statistic for detection of aberrant answer changes. *Journal of Educational Measurement*, 54(2), 200–217.
- Sinharay, S., & Johnson, M. S. (2017). Three new methods for analysis of answer changes. *Educational and Psychological Measurement*, 77(1), 54–81.
- Sinharay, S. (2018). Detecting fraudulent erasures at an aggregate level. *Journal of Educational and Behavioral Statistics*, 43(3), 286–315.
- Wollack, J. A., Cohen, A. S., & Eckerly, C. A. (2015). Detecting test tampering using item response theory. *Educational and Psychological Measurement*, 75(6), 931–953.
- Wollack, J. A., & Eckerly, C. A. (2017). Detecting test tampering at the group level. In G. J. Cizek & J. A. Wollack (Eds.), *Handbook of quantitative methods for detecting cheating on tests* (pp. 214–231). Routledge.

Examples

```
# Setup for Examples 1 and 2 -----
# Settings
set.seed(0)      # seed for reproducibility
```

```

N <- 500      # number of persons
n <- 40       # number of items
G <- 20       # number of groups

# Create groups
group <- rep(1:G, each = N / G)

# Randomly select 20% tampered groups with 20% tampered persons
cg <- sample(1:G, size = G * 0.20)
cv <- NULL
for (g in cg) {
  cv <- c(cv, sample(which(group == g), size = N / G * 0.20))
}

# Create vectors of indicators (1 = tampered, 0 = non-tampered)
group_ind <- ifelse(1:G %in% cg, 1, 0)
person_ind <- ifelse(1:N %in% cv, 1, 0)

# Generate person parameters for the nested logit model
xi <- MASS::mvrnorm(
  N,
  mu = c(theta = 0.00, eta = 0.00),
  Sigma = matrix(c(1.00, 0.80, 0.80, 1.00), ncol = 2)
)

# Generate item parameters for the nested logit model
psi <- cbind(
  a = rlnorm(n, meanlog = 0.00, sdlog = 0.25),
  b = rnorm(n, mean = 0.00, sd = 1.00),
  c = runif(n, min = 0.05, max = 0.30),
  lambda1 = rnorm(n, mean = 0.00, sd = 1.00),
  lambda2 = rnorm(n, mean = 0.00, sd = 1.00),
  lambda3 = rnorm(n, mean = 0.00, sd = 1.00),
  zeta1 = rnorm(n, mean = 0.00, sd = 1.00),
  zeta2 = rnorm(n, mean = 0.00, sd = 1.00),
  zeta3 = rnorm(n, mean = 0.00, sd = 1.00)
)

# Simulate uncontaminated data
dat <- sim(psi, xi)
x_0 <- x <- dat$x
d_0 <- d <- dat$d

# Simulate 5% random erasures for non-tampered persons
r_0 <- r <- ifelse(x == 1, 4, d)
for (v in setdiff(1:N, cv)) {
  ci <- sample(1:n, size = n * 0.05)
  for (i in ci) {
    r_0[v, i] <- sample((1:4)[-r[v, i]], size = 1)
  }
  x_0[v, ci] <- ifelse(r_0[v, ci] == 4, 1, 0)
  d_0[v, ci] <- ifelse(r_0[v, ci] == 4, NA, r_0[v, ci])
}

```

```

rm(r_0, r)

# Modify contaminated data by tampering with 20% of the scores and
# distractors
for (v in cv) {
  ci <- sample(1:n, size = n * 0.20)
  x[v, ci] <- 1
  d[v, ci] <- NA
}

# Example 1: Person-Level Statistics -----

# Detect test tampering
out <- detect_tt(
  method = c("EDI_SD_*", "GBT_SD", "L_SD"),
  psi = psi,
  x = x,
  d = d,
  x_0 = x_0,
  d_0 = d_0
)

# Example 2: Group-Level Statistics -----

# Detect test tampering
out <- detect_tt(
  method = "EDI_SD_*",
  psi = psi,
  x = x,
  d = d,
  x_0 = x_0,
  d_0 = d_0,
  group = group
)

```

sim

Simulate data

Description

Simulate data using item response theory (IRT) models.

Usage

```
sim(psi, xi)
```

Arguments

psi	A matrix of item parameters.
xi	A matrix of person parameters.

Value

A list is returned. Possible elements include:

x	A matrix of item scores.
d	A matrix of item distractors.
r	A matrix of item responses.
y	A matrix of item log response times.

Models for Item Scores

The **Rasch**, **2PL**, and **3PL models** (Birnbbaum, 1968; Rasch, 1960) are given by

$$P(X_{vi} = 1|\theta_v, a_i, b_i, c_i) = c_i + \frac{1 - c_i}{1 + \exp\{-a_i(\theta_v - b_i)\}}.$$

- psi must contain columns named "a", "b", and "c" for the item discrimination, difficulty, and pseudo-guessing parameters, respectively.
- xi must contain a column named "theta" for the person ability parameters.

The **partial credit model** (PCM; Masters, 1982) and the **generalized partial credit model** (GPCM; Muraki, 1992) are given by

$$P(X_{vi} = j|\theta_v, a_i, \mathbf{c}_i) = \frac{\exp\{\sum_{k=0}^j a_i(\theta_v - c_{ik})\}}{\sum_{l=0}^{m_i} \exp\{\sum_{k=0}^l a_i(\theta_v - c_{ik})\}}.$$

- psi must contain columns named "a" for the item discrimination parameter and "c0", "c1", ..., for the item category parameters.
- xi must contain a column named "theta" for the person ability parameters.

The **graded response model** (GRM; Samejima, 1969) is given by

$$P(X_{vi} = j|\theta_v, a_i, \mathbf{b}_i) = P(X_{vi} \geq j|\theta_v, a_i, \mathbf{b}_i) - P(X_{vi} \geq j+1|\theta_v, a_i, \mathbf{b}_i),$$

where

$$P(X_{vi} \geq j|\theta_v, a_i, \mathbf{b}_i) = \begin{cases} 1 & \text{if } j = 0, \\ \frac{1}{1 + \exp\{-a_i(\theta_v - b_{ij})\}} & \text{if } 1 \leq j \leq m_i, \\ 0 & \text{if } j = m_i + 1. \end{cases}$$

- psi must contain columns named "a" for the item discrimination parameter and "b1", "b2", ..., for the item location parameters listed in increasing order.
- xi must contain a column named "theta" for the person ability parameters.

Models for Item Distractors

The **nested logit model** (NLM; Bolt et al., 2012) is given by

$$P(D_{vi} = j|\theta_v, \eta_v, a_i, b_i, c_i, \boldsymbol{\lambda}_i, \boldsymbol{\zeta}_i) = [1 - P(X_{vi} = 1|\theta_v, a_i, b_i, c_i)] \times P(D_{vi} = j|X_{vi} = 0, \eta_v, \boldsymbol{\lambda}_i, \boldsymbol{\zeta}_i),$$

where

$$P(D_{vi} = j|X_{vi} = 0, \eta_v, \boldsymbol{\lambda}_i, \boldsymbol{\zeta}_i) = \frac{\exp(\lambda_{ij}\eta_v + \zeta_{ij})}{\sum_{k=1}^{m_i-1} \exp(\lambda_{ik}\eta_v + \zeta_{ik})}.$$

- psi must contain columns named "a", "b", and "c" for the item discrimination, difficulty, and pseudo-guessing parameters, respectively, "lambda1", "lambda2", . . . , for the item slope parameters, and "zeta1", "zeta2", . . . , for the item intercept parameters.
- xi must contain columns named "theta" and "eta" for the person parameters that govern response correctness and distractor selection, respectively.

Models for Item Responses

The **nominal response model** (NRM; Bock, 1972) is given by

$$P(R_{vi} = j | \eta_v, \lambda_i, \zeta_i) = \frac{\exp(\lambda_{ij}\eta_v + \zeta_{ij})}{\sum_{k=1}^{m_i} \exp(\lambda_{ik}\eta_v + \zeta_{ik})}.$$

- psi must contain columns named "lambda1", "lambda2", . . . , for the item slope parameters and "zeta1", "zeta2", . . . , for the item intercept parameters. If there is a correct response category, its parameters should be listed last.
- xi must contain a column named "eta" for the person parameters that govern response selection.

Models for Item Log Response Times

The **lognormal model** (van der Linden, 2006) is given by

$$f(Y_{vi} | \tau_v, \alpha_i, \beta_i) = \frac{\alpha_i}{\sqrt{2\pi}} \exp\left\{-\frac{1}{2}[\alpha_i(Y_{vi} - (\beta_i - \tau_v))]^2\right\}.$$

- psi must contain columns named "alpha" and "beta" for the item time discrimination and time intensity parameters, respectively.
- xi must contain a column named "tau" for the person speed parameters.

References

- Birnbaum, A. (1968). Some latent trait models and their use in inferring an examinee's ability. In F. M. Lord & M. R. Novick (Eds.), *Statistical theories of mental test scores* (pp. 397–479). Addison-Wesley.
- Bock, R. D. (1972). Estimating item parameters and latent ability when responses are scored in two or more nominal categories. *Psychometrika*, 37(1), 29–51.
- Bolt, D. M., Wollack, J. A., & Suh, Y. (2012). Application of a multidimensional nested logit model to multiple-choice test items. *Psychometrika*, 77(2), 339–357.
- Masters, G. N. (1982). A Rasch model for partial credit scoring. *Psychometrika*, 47(2), 149–174.
- Muraki, E. (1992). A generalized partial credit model: Application of an EM algorithm. *Applied Psychological Measurement*, 16(2), 159–176.
- Rasch, G. (1960). *Probabilistic models for some intelligence and attainment tests*. Danish Institute for Educational Research.
- Samejima, F. (1969). Estimation of latent ability using a response pattern of graded scores. *Psychometrika*, 34(S1), 1–97.
- van der Linden, W. J. (2006). A lognormal model for response times on test items. *Journal of Educational and Behavioral Statistics*, 31(2), 181–204.

Examples

```
# Setup for Examples 1 to 5 -----

# Settings
set.seed(0)      # seed for reproducibility
N <- 500          # number of persons
n <- 40           # number of items

# Example 1: 3PL Model and Lognormal Model -----

# Generate person parameters
xi <- MASS::mvrnorm(
  N,
  mu = c(theta = 0.00, tau = 0.00),
  Sigma = matrix(c(1.00, 0.25, 0.25, 0.25), ncol = 2)
)

# Generate item parameters
psi <- cbind(
  a = rlnorm(n, meanlog = 0.00, sdlog = 0.25),
  b = NA,
  c = runif(n, min = 0.05, max = 0.30),
  alpha = runif(n, min = 1.50, max = 2.50),
  beta = NA
)

# Generate positively correlated difficulty and time intensity parameters
psi[, c("b", "beta")] <- MASS::mvrnorm(
  n,
  mu = c(b = 0.00, beta = 3.50),
  Sigma = matrix(c(1.00, 0.20, 0.20, 0.15), ncol = 2)
)

# Simulate item scores and log response times
dat <- sim(psi, xi)
x <- dat$x
y <- dat$y

# Example 2: Generalized Partial Credit Model -----

# Generate person parameters
xi <- cbind(theta = rnorm(N, mean = 0.00, sd = 1.00))

# Generate item parameters
psi <- cbind(
  a = rlnorm(n, meanlog = 0.00, sdlog = 0.25),
  c0 = 0,
  c1 = rnorm(n, mean = -1.00, sd = 0.50),
  c2 = rnorm(n, mean = 0.00, sd = 0.50),
  c3 = rnorm(n, mean = 1.00, sd = 0.50)
)
```

```

# Simulate item scores
x <- sim(psi, xi)$x

# Example 3: Graded Response Model -----

# Generate person parameters
xi <- cbind(theta = rnorm(N, mean = 0.00, sd = 1.00))

# Generate item parameters
psi <- cbind(
  a = rlnorm(n, meanlog = 0.00, sdlog = 0.25),
  b1 = rnorm(n, mean = -1.00, sd = 0.50),
  b2 = rnorm(n, mean = 0.00, sd = 0.50),
  b3 = rnorm(n, mean = 1.00, sd = 0.50)
)

# Sort item location parameters in increasing order
psi[, paste0("b", 1:3)] <- t(apply(psi[, paste0("b", 1:3)], 1, sort))

# Simulate item scores
x <- sim(psi, xi)$x

# Example 4: Nested Logit Model -----

# Generate person parameters
xi <- MASS::mvrnorm(
  N,
  mu = c(theta = 0.00, eta = 0.00),
  Sigma = matrix(c(1.00, 0.80, 0.80, 1.00), ncol = 2)
)

# Generate item parameters
psi <- cbind(
  a = rlnorm(n, meanlog = 0.00, sdlog = 0.25),
  b = rnorm(n, mean = 0.00, sd = 1.00),
  c = runif(n, min = 0.05, max = 0.30),
  lambda1 = rnorm(n, mean = 0.00, sd = 1.00),
  lambda2 = rnorm(n, mean = 0.00, sd = 1.00),
  lambda3 = rnorm(n, mean = 0.00, sd = 1.00),
  zeta1 = rnorm(n, mean = 0.00, sd = 1.00),
  zeta2 = rnorm(n, mean = 0.00, sd = 1.00),
  zeta3 = rnorm(n, mean = 0.00, sd = 1.00)
)

# Simulate item scores and distractors
dat <- sim(psi, xi)
x <- dat$x
d <- dat$d

# Example 5: Nominal Response Model -----

# Generate person parameters
xi <- cbind(eta = rnorm(N, mean = 0.00, sd = 1.00))

```

```
# Generate item parameters
psi <- cbind(
  lambda1 = rnorm(n, mean = -0.50, sd = 0.50),
  lambda2 = rnorm(n, mean = -0.50, sd = 0.50),
  lambda3 = rnorm(n, mean = -0.50, sd = 0.50),
  lambda4 = rnorm(n, mean = 1.50, sd = 0.50),
  zeta1 = rnorm(n, mean = -0.50, sd = 0.50),
  zeta2 = rnorm(n, mean = -0.50, sd = 0.50),
  zeta3 = rnorm(n, mean = -0.50, sd = 0.50),
  zeta4 = rnorm(n, mean = 1.50, sd = 0.50)
)

# Simulate item responses
r <- sim(psi, xi)$r
```

Index

`detect_ac`, [2](#)
`detect_ac()`, [7](#)
`detect_as`, [5](#)
`detect_as()`, [3](#), [16](#)
`detect_nm`, [11](#)
`detect_nm()`, [20](#)
`detect_pk`, [14](#)
`detect_pk()`, [7](#)
`detect_pm`, [18](#)
`detect_pm()`, [12](#)
`detect_rg`, [22](#)
`detect_tt`, [26](#)

`sim`, [30](#)