Package: ItemRest (via r-universe)

October 7, 2026

Title Automated Item Removal Strategies for Exploratory Factor Analysis

Version 1.0.0

Description Identifies candidate item sets through threshold-driven iterative removal searches in exploratory factor analysis. Provides a no-removal baseline, numerical diagnostics, factor reliability, holdout evaluation, and bootstrap search stability to support transparent screening and documented content review rather than automatic measurement decisions. The loading criteria are based on best practices and established heuristics (e.g., Costello & Osborne (2005) <doi:10.7275/jyj1-4868>, Howard (2016) <doi:10.1080/10447318.2015.1087664>). Includes flexible thresholds for factor loadings (min_loading) and cross-loading differences (loading_diff).

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 4.1)

Imports clue, GPArotation, gtools, psych, qgraph, stats, tools, utils

URL https://github.com/ahmetcaliskan1987/ItemRest

BugReports https://github.com/ahmetcaliskan1987/ItemRest/issues

Suggests testthat (>= 3.2.0), knitr, rmarkdown, lavaan

VignetteBuilder knitr

Roxygen list(markdown = TRUE)

Config/testthat/edition 3

NeedsCompilation no

Config/roxygen2/version 8.1.0

Config/pak/sysreqs
cmake libglpk-dev make libicu-dev libjpeg-dev libpng-dev libuv1-dev libxml2-dev

Repository https://ahmetcaliskan1987.r-universe.dev

Date/Publication 2026-10-06 22:26:00 UTC

RemoteUrl https://github.com/ahmetcaliskan1987/itemrest

RemoteRef HEAD

RemoteSha 79d13132d25f7c432b2e68c8ee4d990f6132998f

Contents

itemrest 2
itemrest_bootstrap 5
itemrest_split 6
itemrest_validate 6
print.itemrest_result 8

Index **9**

`itemrest` *Identify Candidate Item Sets for Exploratory Factor Analysis*

Description

Reassesses remaining items after every removal and evaluates combinations of flagged items. This threshold-driven search in one sample does not establish an optimal, valid, or independently replicated measurement solution. Factor count is determined once at baseline and stays fixed. The original correlation matrix and observation counts are computed once for all baseline items, after missing-data handling and scoring keys. Each retained set uses the corresponding rows and columns. Positive-definiteness checks and any requested smoothing are applied separately to each subset.

Usage

```
itemrest(
  data,
  cor_method = "pearson",
  n_factors = NULL,
  extract = "uls",
  rotate = "oblimin",
  min_loading = 0.3,
  loading_diff = 0.1,
  missing = c("listwise", "pairwise", "fail"),
  min_items_per_factor = 3L,
  max_factor_correlation = 0.85,
  pd_action = c("fail", "smooth"),
  reliability = c("alpha_omega", "alpha", "none"),
  rank_by = c("none", "n_removed", "explained_variance"),
  retain_items = character(),
  item_reasons = NULL,
  max_solutions = 10000L,
  seed = NULL,
```

```
  verbose = TRUE,
  keys = NULL,
  parallel_iterations = 100L,
  ordinal_categories = 7L,
  store_fits = FALSE,
  parallel_method = c("fa", "pc")
)
```

Arguments

data	Numeric data.frame or matrix, with unique item names.
cor_method	"pearson", "spearman", "kendall", or "polychoric". The last uses qgraph mixed correlations after detecting integer-valued ordinal items up to ordinal_categories observed categories (excluding missing values).
n_factors	Fixed factor count, or NULL for baseline parallel analysis.
extract	Extraction method passed to psych::fa; default "uls".
rotate	Rotation passed to psych::fa; default "oblimin".
min_loading	Minimum absolute primary loading; default 0.30.
loading_diff	Minimum primary-secondary difference, or "howard".
missing	"listwise" (default), "pairwise", or "fail". Listwise handling is applied once across all baseline items, fixing observations across sets. Pairwise counts are recorded; their minimum is used as conservative n.obs. This does not resolve missing-data bias.
min_items_per_factor	Minimum qualifying primary items per factor; default 3. Flagged items do not count toward this threshold.
max_factor_correlation	Absolute factor correlation requiring review; default 0.85, a configurable screening threshold rather than a validity rule.
pd_action	"fail" or "smooth" for nonpositive definite correlations. Smoothed sets require review and are excluded from candidate screening.
reliability	"alpha_omega", "alpha", or "none". Raw, standardized, and selected-correlation alpha are distinguished. Model omega total uses sum(L Phi L') / (sum(L Phi L') + sum(uniquenesses)) for a unit-weighted standardized sum. It includes all common factors, is not omega hierarchical, and does not establish unidimensionality. No items are automatically reversed.
rank_by	"none" (discovery order), "n_removed" (ascending), or "explained_variance" (descending). No winner is selected. Explained variance is mean model communality within the retained set; values across different sets do not establish superiority.
retain_items	Item names protected from removal for content reasons. Protection does not waive screening criteria. A flagged protected item can prevent all candidate solutions; inspect Problem_Items and content_decisions.
item_reasons	Named character vector documenting item decisions.

`max_solutions` Maximum queued/evaluated sets including baseline; default 10000. A bounded search is explicitly labelled incomplete.

`seed` Integer seed. NULL uses the execution host's local calendar date in DDMMYYYY format (e.g. 05102026 is used numerically as 5102026). Both the formatted label and numeric seed are recorded. RNG state is restored on exit.

`verbose` Print a brief search summary; default TRUE.

`keys` Named numeric vector of 1 or -1 for specified items. Minus one explicitly reverses scoring by negating values. No automatic reverse scoring occurs; offsets do not affect correlations or reliability.

`parallel_iterations`
Number of parallel-analysis replications; default 100.

`ordinal_categories`
Maximum number of integer-valued categories detected as ordinal by the polychoric backend; default 7.

`store_fits` Keep full psych EFA fit objects in solution_details; default FALSE retains loadings, Phi, communalities, reliability, and diagnostics.

`parallel_method`
"fa" (default) for reduced-matrix eigenvalues based on a one-factor minres fit, or "pc" for full-matrix component eigenvalues. Polychoric analysis permutes observed values within each item, preserving categories and missing positions, and computes the same correlation type for each reference sample. The 95th percentile is used for comparison.

Value

An itemrest_result containing candidate_solutions, removal_summary (baseline, failures, and skipped sets), solution_details (EFA, diagnostics, assignments, factor reliability, and first-discovered paths), initial_efa, problem_items, descriptive_stats, settings, search, content_decisions, data_summary, provenance, correlation_matrix (the original unsmoothed baseline matrix), and pairwise_n. correlation_matrix is NULL when no correlation could be computed. candidate_solutions may have zero rows. correlation_warnings and correlation_messages belong to the source matrix; they are recorded once and not attributed to every retained set. Estimator messages are recorded per set, but informational messages do not require review; explicit nonconvergence, unavailable rotation, variance problems, and matrix repair reports do. parallel_analysis records automatic factor-count diagnostics. Full EFA fit objects are included only when store_fits = TRUE. Screening eligibility is not substantive validity. The old optimal_strategy field is replaced by candidate_solutions. search$complete is FALSE for a limit or numerical branch failure; skipped underidentified sets remain documented terminal attempts.

Examples

```
set.seed(4)
f <- rnorm(150)
d <- data.frame(I1 = f + rnorm(150), I2 = f + rnorm(150),
                I3 = f + rnorm(150), I4 = f + rnorm(150))
result <- itemrest(d, n_factors = 1, seed = 10, verbose = FALSE)
print(result)
```

`itemrest_bootstrap` *Bootstrap the Entire Threshold-Driven Candidate Search*

Description

Resamples observations and reruns discovery for each replicate, holding the original baseline factor count, scoring keys, and screening criteria fixed. Each replicate computes its own correlation matrix once and reuses its submatrices throughout that replicate's removal search. This measures search stability rather than merely refitting a selected set. It does not replace independent validation or estimate probabilities of validity.

Usage

```
itemrest_bootstrap(
  x,
  data,
  n_boot = 100L,
  seed = NULL,
  progress = interactive()
)
```

Arguments

`x`	An itemrest_result defining search settings.
`data`	Original numeric discovery data, including every discovery item.
`n_boot`	Number of replicates, default 100.
`seed`	Integer seed; NULL uses the local date in DDMMYYYY format.
`progress`	Show a text progress bar; default interactive().

Value

A list with item_stability, solution_stability, replicates, settings, and seed. Any/all-candidate frequencies divide by complete successful searches; searches with no candidates count as zero. Mean candidate retention divides only by complete replicates with candidates and weights replicates equally. Failed/incomplete searches are excluded from frequency denominators and reported separately. Solution frequencies use exact retained sets, not ranks. Additional *_All_Attempts columns divide by n_boot, conservatively treating failed and incomplete searches as having no candidate solutions. This is a sensitivity summary, not an estimate of what those searches would yield.

itemrest_split *Split Observations into Discovery and Validation Samples*

Description

Split Observations into Discovery and Validation Samples

Usage

```
itemrest_split(
  data,
  train_fraction = 0.7,
  seed = NULL,
  validation_method = c("efa", "cfa"),
  ...
)
```

Arguments

`data`	Numeric data.frame or matrix.
`train_fraction`	Fraction assigned to discovery, default 0.70.
`seed`	Integer seed; NULL uses the local date in DDMMYYYY format.
`validation_method`	"efa" or "cfa".
`...`	Arguments to itemrest, excluding data and seed.

Value

A list with discovery, validation, disjoint discovery_rows and validation_rows, and seed. Splitting precedes missing-data handling.

itemrest_validate *Reevaluate Prespecified Candidate Sets on Validation Data*

Description

Refits discovery item sets on user-supplied validation observations, without selecting or removing additional items. The user is responsible for supplying independent observations. EFA replication and optional simple-structure CFA are distinct checks; neither automatically establishes validity. EFA computes one new correlation matrix over the union of requested retained items and uses its submatrices for those sets. Constant items fail only the sets containing them and are excluded from correlation estimation. Missing data preparation fixes rows across all discovery items before this selection. Factor congruence maximizes total absolute congruence over one-to-one factor assignments and aligns signs; Min_Congruence is descriptive, not a validity test. Identical response values are detected regardless of row names or numeric storage types or row order. Nonidentical data may still overlap; independence is not proven. Bootstrap's original-data check remains sensitive to row order.

Usage

```
itemrest_validate(
  x,
  data,
  solution_ids = NULL,
  method = c("efa", "cfa"),
  ordered_items = NULL,
  seed = NULL
)
```

Arguments

x	An itemrest_result from discovery data.
data	Numeric validation data containing every discovery item.
solution_ids	Solution_ID values. NULL evaluates candidates plus baseline.
method	"efa" (default) or "cfa". CFA requires optional lavaan and fixes each retained item's primary factor to its discovery assignment; all other cross-loadings are constrained to zero. Factors remain correlated.
ordered_items	For CFA, names of ordinal items. NULL uses integer-valued discovery items identified under cor_method = "polychoric". A character() value explicitly requests continuous treatment. Ordinal CFA uses WLSMV; continuous CFA uses MLR. Pairwise EFA uses pair counts; continuous CFA with missing = "pairwise" uses FIML, explicitly labelled in the output.
seed	Integer seed; NULL uses the local date in DDMMYYYY format.

Value

A list containing validation_summary, solution_details, data_summary, source_solution_ids, method, seed, and independence_note. CFA fit indices are descriptive and are not converted to an automatic pass/fail verdict. correlation_items, constant_items, correlation_warnings, and correlation_messages describe the validation EFA source matrix.

Examples

```
set.seed(1)
f <- rnorm(300)
d <- as.data.frame(replicate(4, f + rnorm(300)))
discovery <- itemrest(d[1:150, ], n_factors = 1, verbose = FALSE)
check <- itemrest_validate(discovery, d[151:300, ])
```

`print.itemrest_result` *Print Candidate Solutions from ItemRest*

Description

Print Candidate Solutions from ItemRest

Usage

```
## S3 method for class 'itemrest_result'
print(x, report = c("candidates", "all"), ...)
```

Arguments

`x`	An itemrest_result.
`report`	"candidates" (default) or "all". The old "optimal" value is a deprecated alias for "candidates".
`...`	Unused.

Value

x, invisibly.

Index

itemrest, 2
itemrest_bootstrap, 5
itemrest_split, 6
itemrest_validate, 6

print.itemrest_result, 8