Package: PsychoMatic (via r-universe)

June 30, 2026

Type Package

Title Automated Psychometric Workflows and Reporting Tools

Version 0.3.0

Description Automates common psychometric workflows for applied researchers, including item descriptives, inter-item correlations, exploratory and confirmatory factor analysis, reliability, multi-group measurement invariance, and alignment optimization. Decision heuristics are informed by procedures such as parallel analysis (Horn, 1965, <doi:10.1007/BF02289447>), multivariate normality diagnostics (Mardia, 1970, <doi:10.1093/biomet/57.3.519>), measurement-invariance fit-change rules (Chen, 2007, <doi:10.1080/10705510701301834>), and alignment optimization (Asparouhov and Muthen, 2014, <doi:10.1080/10705511.2014.919210>), among others. Results can be returned as structured R objects and exported as bilingual reports for transparent analytical documentation.

License MIT + file LICENSE

URL https://github.com/gmoncayoj/PsychoMatic

BugReports https://github.com/gmoncayoj/PsychoMatic/issues

Encoding UTF-8

Depends R (>= 4.1.0)

Imports GPArotation, lavaan, Matrix, psych, stats, utils

Suggests flextable, knitr, mirt, MVN, officer, openxlsx, rmarkdown, semTools, sirt, testthat (>= 3.0.0), writexl

Roxygen list(markdown = TRUE)

Config/testthat/edition 3

VignetteBuilder knitr

LazyData true

Config/roxygen2/version 8.0.0

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

Date/Publication 2026-06-23 01:54:48 UTC

RemoteUrl https://github.com/gmoncayoj/psychomatic

RemoteRef HEAD

RemoteSha e14e08071de7c3294cb492fc24adf9c9d66c2ef9

Contents

`cfa_auto` *Automated CFA with psychometric reporting*

Description

Runs a confirmatory factor analysis with heuristic estimator selection, multivariate normality diagnostics, fit indices, reliability estimates, and model respecification suggestions.

Usage

```
cfa_auto(
  data,
  model,
  ordered = NULL,
  estimator = NULL,
  std.lv = FALSE,
  mi_threshold = 10,
  n_mi = 10,
  alpha_norm = 0.05,
  language = "esp"
)

## S3 method for class 'cfa_auto'
print(x, digits = 3, ...)
```

Arguments

data	Data frame containing the observed variables.
model	Model syntax written for lavaan.
ordered	Optional vector indicating which items should be treated as ordinal.
estimator	Optional estimator. If NULL, the function selects one automatically.
std.lv	Whether latent variables should be standardized.
mi_threshold	Minimum threshold for reporting modification indices.
n_mi	Maximum number of modification indices to display.
alpha_norm	Alpha level used when evaluating multivariate normality.
language	Report language: "esp" or "eng".
x	Object returned by cfa_auto().
digits	Number of decimal places shown in the printed report.
...	Additional unused arguments.

Value

Object of class cfa_auto.

Examples

```
if (identical(Sys.getenv("PSYCHOMATIC_RUN_SLOW_EXAMPLES"), "true") &&
    requireNamespace("semTools", quietly = TRUE) &&
    requireNamespace("MVN", quietly = TRUE)) {
  model <- "visual =~ x1 + x2 + x3"
  items <- lavaan::HolzingerSwineford1939[1:80, c("x1", "x2", "x3")]
  result <- cfa_auto(items, model = model, estimator = "ML", language = "eng")
}
```

`compare_models_auto` *Compare alternative CFA models*

Description

Fits a named set of lavaan CFA models and returns a compact comparison table. The recommendation is descriptive only and favors converged models with stronger comparative fit indices. Information criteria are reported when available, but they are not required and are not used for WLSMV-style categorical estimators.

Usage

```
compare_models_auto(
  data,
  models,
  ordered = NULL,
  estimator = "ML",
  language = c("eng", "esp")
)
```

Arguments

`data`	Data frame containing observed variables.
`models`	Named list or named character vector of lavaan model strings.
`ordered`	Optional ordered variables passed to `lavaan::cfa()`. If `NULL` and a categorical estimator such as WLSMV is used, observed items referenced in `models` are treated as ordered.
`estimator`	Estimator passed to `lavaan::cfa()`.
`language`	Recommendation language: `"eng"` or `"esp"`.

Value

A list of class `psychomatic_model_comparison`.

Examples

```
if (requireNamespace("lavaan", quietly = TRUE)) {
  models <- list(one = "visual =~ x1 + x2 + x3")
  compare_models_auto(lavaan::HolzingerSwineford1939, models, estimator = "ML")
}
```

cormat *Inter-item correlation matrix*

Description

Computes an inter-item correlation matrix using Pearson, polychoric, or tetrachoric correlations. Optionally exports the matrix to an Excel file.

Usage

```
cormat(
  data,
  type = c("pearson", "poly", "tetra"),
  report = FALSE,
  file_name = NULL,
  digits = 3
)
```

Arguments

data	Data frame or matrix containing the items.
type	Type of correlation to compute. Options are "pearson", "poly" for polychoric correlations, and "tetra" for tetrachoric correlations.
report	If TRUE, exports the matrix to an Excel file.
file_name	Optional Excel file name. If NULL, a timestamped name is generated automatically.
digits	Number of decimal places used in the Excel report.

Value

A correlation matrix.

Examples

```
items <- data.frame(
  item1 = c(1, 2, 3, 4, 5),
  item2 = c(2, 2, 3, 4, 5),
  item3 = c(5, 4, 3, 2, 1)
)
cormat(items, type = "pearson")
if (requireNamespace("openxlsx", quietly = TRUE)) {
  path <- tempfile("correlations-", fileext = ".xlsx")
  cormat(items, type = "pearson", report = TRUE, file_name = path)
}
```

desc_auto *Automated descriptives for numeric items*

Description

Computes descriptive statistics and response percentages for numeric item columns in a data frame. Results can optionally be exported to Excel.

Usage

```
desc_auto(data, digits = 2, language = "esp", report = FALSE)
```

Arguments

data	Data frame containing the item responses.
digits	Number of decimal places used in the report.
language	Report language: "esp" or "eng".
report	If TRUE, exports an Excel file.

Value

Data frame with descriptive statistics and category percentages.

Examples

```
items <- data.frame(
  item1 = c(1, 2, 3, 4, 5),
  item2 = c(2, 2, 3, 4, 5)
)
desc_auto(items, language = "eng")
```

dif_auto *Screen differential item functioning by group*

Description

Performs a lightweight DIF screening using logistic regression. Polytomous items are dichotomized at their median for this screening, so flagged results should be followed up with a dedicated IRT or ordinal DIF analysis.

Usage

```
dif_auto(
  data,
  group,
  items = NULL,
  alpha = 0.01,
  language = c("eng", "esp"),
  digits = 3
)
```

Arguments

data	Data frame or matrix containing item responses. If group is a column name in data, that column is used as the grouping variable.
group	Grouping vector or the name of the grouping column in data.
items	Optional item names. Defaults to all numeric item columns except the grouping column.
alpha	Significance level used for DIF flags.
language	Interpretation language: "eng" or "esp".
digits	Number of decimal places used in interpretation tables.

Value

A list of class psychomatic_dif with screening results, item-level interpretation (dif_type, decision, severity, and recommendation), an overall DIF summary, and supporting APA 7 references.

Examples

```
set.seed(1)
group <- rep(c("A", "B"), each = 20)
theta <- rnorm(40)
items <- data.frame(
  item1 = as.integer(theta + rnorm(40) > 0),
  item2 = as.integer(theta + rnorm(40) > 0)
)
dif_auto(items, group = group)
```

---

efa_auto *Automated exploratory factor analysis*

Description

Runs an automated exploratory factor analysis workflow with estimator selection, parallel analysis, rotation, iterative item screening, reliability, and structured output tables.

Usage

```
efa_auto(
  data,
  rotation = c("oblique", "orthogonal"),
  min_loading = 0.3,
  min_communality = 0.3,
  min_cross_loading_diff = 0.15,
  max_iter = 20,
  verbose = TRUE,
  export_format = NULL,
  file_name = NULL,
  language = "esp",
  seed = NULL,
  ...
)
```

Arguments

`data`	Data frame or matrix containing numeric item responses.
`rotation`	Rotation family: `"oblique"` (oblimin) or `"orthogonal"` (varimax).
`min_loading`	Minimum acceptable factor loading.
`min_communality`	Minimum acceptable communality.
`min_cross_loading_diff`	Minimum difference between the two largest loadings required to avoid a cross-loading flag.
`max_iter`	Maximum number of iterative item-removal steps.
`verbose`	If `TRUE`, prints a console report.
`export_format`	Optional export format passed to `export_efa()`.
`file_name`	Optional base file name for exported files.
`language`	Report language: `"esp"` or `"eng"`.
`seed`	Optional single integer seed for reproducible random operations. Defaults to `NULL`, so no seed is set unless explicitly requested.
`...`	Deprecated legacy aliases kept temporarily for backward compatibility. New code should use the English argument names documented above.

Value

Object of class `efa_auto` with model results and tidy tables.

Examples

```
set.seed(1)
factor_score <- rnorm(80)
items <- data.frame(
  item1 = factor_score + rnorm(80, sd = 0.4),
```

```
  item2 = factor_score + rnorm(80, sd = 0.4),
  item3 = factor_score + rnorm(80, sd = 0.4),
  item4 = factor_score + rnorm(80, sd = 0.4)
)
result <- efa_auto(items, max_iter = 1, verbose = FALSE, language = "eng")
```

`export_cfa` *Export CFA results to Excel or Word*

Description

Export CFA results to Excel or Word

Usage

```
export_cfa(result, format = c("excel", "word"), file_name = NULL, digits = 3)
```

Arguments

`result`	Object of class `"cfa_auto"` returned by `cfa_auto()`.
`format`	`"excel"` or `"word"` (you can provide both).
`file_name`	Optional base file name without extension. If `NULL`, a timestamped name is generated automatically.
`digits`	Number of decimal places used when exporting the Word report.

Value

The input `result`, invisibly.

Examples

```
if (identical(Sys.getenv("PSYCHOMATIC_RUN_SLOW_EXAMPLES"), "true") &&
    requireNamespace("semTools", quietly = TRUE) &&
    requireNamespace("MVN", quietly = TRUE) &&
    requireNamespace("openxlsx", quietly = TRUE)) {
  model <- "visual =~ x1 + x2 + x3"
  items <- lavaan::HolzingerSwineford1939[1:80, c("x1", "x2", "x3")]
  result <- cfa_auto(items, model = model, estimator = "ML", language = "eng")
  export_cfa(result, format = "excel", file_name = tempfile("cfa-", fileext = ".xlsx"))
}
```

export_efa *Export EFA results to Excel or Word*

Description

Exports the structured tables from an `efa_auto()` result. The function uses the package-level report writer so EFA exports stay consistent with the integrated `psychomatic_report()` workflow.

Usage

```
export_efa(result, format = c("excel", "word"), file_name = NULL)
```

Arguments

`result`	Object of class `"efa_auto"` returned by `efa_auto()`.
`format`	Output format: `"excel"`, `"word"`, or both.
`file_name`	Optional output path or base path. If `NULL`, a temporary file path is generated.

Value

The input `result`, invisibly.

Examples

```
if (identical(Sys.getenv("PSYCHOMATIC_RUN_SLOW_EXAMPLES"), "true") &&
    requireNamespace("openxlsx", quietly = TRUE)) {
  set.seed(1)
  factor_score <- rnorm(80)
  items <- data.frame(
    item1 = factor_score + rnorm(80, sd = 0.4),
    item2 = factor_score + rnorm(80, sd = 0.4),
    item3 = factor_score + rnorm(80, sd = 0.4),
    item4 = factor_score + rnorm(80, sd = 0.4)
  )
  result <- efa_auto(items, max_iter = 1, verbose = FALSE, language = "eng")
  export_efa(result, format = "excel", file_name = tempfile("efa-", fileext = ".xlsx"))
}
```

`factorial_invariance_auto` *Automated multi-group measurement invariance*

Description

Sequentially fits configural, metric, scalar, and strict invariance models to evaluate the highest level of measurement invariance supported by the data.

Usage

```
factorial_invariance_auto(
  data,
  group,
  model,
  language = c("esp", "eng"),
  estimator = "auto",
  ordered = NULL,
  report = FALSE
)
```

Arguments

`data`	Data frame containing the items and grouping variable.
`group`	Name or expression identifying the grouping variable.
`model`	Model syntax written for `lavaan`.
`language`	Report language: `"esp"` or `"eng"`.
`estimator`	Requested estimator or `"auto"` for automatic selection.
`ordered`	Optional ordered-variable configuration.
`report`	`FALSE`, `TRUE`, or a file path for exporting a report.

Value

Object of class `factorial_invariance_auto`, returned invisibly.

Examples

```
if (identical(Sys.getenv("PSYCHOMATIC_RUN_SLOW_EXAMPLES"), "true")) {
  model <- "visual =~ x1 + x2 + x3"
  rows <- c(1:25, 160:184)
  items <- lavaan::HolzingerSwineford1939[rows, c("school", "x1", "x2", "x3")]
  factorial_invariance_auto(
    items,
    group = "school",
    model = model,
    estimator = "ML",
```

```
    language = "eng"
  )
}
```

---

inv_align_auto *Automated alignment invariance*

Description

Runs alignment optimization from loading and intercept matrices (lambda and nu) or from item-level data with a grouping vector.

Usage

```
inv_align_auto(
  lambda = NULL,
  nu = NULL,
  data = NULL,
  group = NULL,
  sampling_weights = NULL,
  wgt = NULL,
  language = c("esp", "eng"),
  model = c("2PM", "1PM"),
  align.scale = c(lambda = 0.4, nu = 0.2),
  align.pow = c(lambda = 0.25, nu = 0.25),
  parm_tol = NULL,
  noninvariance_cutoff = 25,
  center = FALSE,
  optimizer = "optim",
  fixed = NULL,
  meth = 1,
  eps = 0.001,
  vcov = NULL,
  digits = 3,
  config_args = list(),
  alignment_args = list()
)

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

## S3 method for class 'inv_align_auto_result'
summary(object, ...)
```

Arguments

lambda, nu	Parameter matrices for loadings and intercepts.
data	Item-level data used when lambda and nu are not supplied.
group	Grouping vector associated with data.
sampling_weights	Optional sampling weights, one per row.
wgt	Optional weights for the alignment procedure.
language	Report language: "esp" or "eng".
model	Model type for the configural alignment step.
align.scale, align.pow, parm_tol	Alignment tuning parameters.
noninvariance_cutoff	Percentage used to flag non-invariance.
center, optimizer, fixed, meth, eps, vcov	Advanced algorithm parameters.
digits	Number of decimal places used in the report.
config_args	Additional arguments for the configural step.
alignment_args	Additional arguments for the alignment step.
x	Object returned by inv_align_auto().
...	Additional unused arguments.
object	Object returned by inv_align_auto().

Value

Object of class inv_align_auto_result.

Examples

```
if (requireNamespace("sirt", quietly = TRUE)) {
  lambda <- matrix(c(0.70, 0.80, 0.75, 0.72, 0.78, 0.74), nrow = 2, byrow = TRUE)
  nu <- matrix(c(0.00, 0.00, 0.00, 0.10, 0.05, 0.08), nrow = 2, byrow = TRUE)
  res <- inv_align_auto(lambda = lambda, nu = nu, language = "eng")
  summary(res)
}
```

`irt_auto` *Fit basic IRT models automatically*

Description

Fits a one-dimensional Rasch, 2PL, or graded response model using `mirt`. The dependency is optional and is loaded only when this function is called.

Usage

```
irt_auto(
  data,
  model = c("auto", "rasch", "2pl", "grm"),
  factors = 1,
  itemtype = NULL,
  technical = list(),
  language = c("eng", "esp"),
  digits = 3
)
```

Arguments

`data`	Data frame or matrix containing item responses.
`model`	IRT model: `"auto"`, `"rasch"`, `"2pl"`, or `"grm"`.
`factors`	Number of latent dimensions passed to `mirt::mirt()`.
`itemtype`	Optional `mirt` item type. If `NULL`, it is selected from `model`.
`technical`	Optional technical arguments passed to `mirt::mirt()`.
`language`	Interpretation language: `"eng"` or `"esp"`.
`digits`	Number of decimal places used in interpretation tables.

Value

A list of class `psychomatic_irt` with fitted model, item parameters, item-level interpretation, a model-level summary, and supporting APA 7 references.

Examples

```
if (identical(Sys.getenv("PSYCHOMATIC_RUN_SLOW_EXAMPLES"), "true") &&
    requireNamespace("mirt", quietly = TRUE)) {
  data(psychomatic_ordinal)
  irt_auto(psychomatic_ordinal[1:60, 1:4], model = "grm")
}
```

psychomatic_continuous

Simulated continuous item responses

Description

A small continuous dataset for examples, tests, and vignettes. The variables approximate six positively related psychometric items generated from one common factor with residual noise.

Usage

```
psychomatic_continuous
```

Format

A data frame with 120 rows and 6 variables:

item1,item2,item3,item4,item5,item6 Continuous item responses.

psychomatic_multigroup

Simulated multigroup item responses

Description

A small multigroup dataset for measurement-invariance examples. It contains two groups and six continuous item responses generated from a common latent variable with a small group mean difference.

Usage

```
psychomatic_multigroup
```

Format

A data frame with 140 rows and 7 variables:

mg1,mg2,mg3,mg4,mg5,mg6 Continuous item responses.

group Grouping factor with levels `A` and `B`.

psychomatic_ordinal *Simulated ordinal item responses*

Description

A small ordinal Likert-type dataset for reproducible examples. The variables are five-category transformations of the continuous example items.

Usage

```
psychomatic_ordinal
```

Format

A data frame with 120 rows and 6 variables:

ord1,ord2,ord3,ord4,ord5,ord6 Ordinal item responses from 1 to 5.

psychomatic_report *Create an integrated PsychoMatic report*

Description

Collects tables from PsychoMatic result objects and writes an Excel workbook and/or Word document. The function is intentionally conservative: it only writes to the user-supplied path, or to `tempdir()` when `file_name = NULL`.

Usage

```
psychomatic_report(
  ...,
  format = "excel",
  file_name = NULL,
  language = c("eng", "esp"),
  digits = 3
)
```

Arguments

`...`	Named PsychoMatic result objects or data frames. Supported objects include outputs from `desc_auto()`, `screen_items()`, `score_scale()`, `efa_auto()`, `cfa_auto()`, `factorial_invariance_auto()`, `inv_align_auto()`, `irt_auto()`, `dif_auto()`, and `compare_models_auto()`.
`format`	Output format: `"excel"`, `"word"`, or both.
`file_name`	Output path or base path. If `NULL`, a temporary file is created.
`language`	Report language: `"eng"` or `"esp"`.
`digits`	Number of decimal places for numeric tables.

Value

Output path, invisibly. When both formats are requested, returns a named character vector.

Examples

```
if (requireNamespace("openxlsx", quietly = TRUE)) {
  items <- data.frame(item1 = 1:5, item2 = c(2, 2, 3, 4, 5))
  path <- tempfile("psychomatic-report-", fileext = ".xlsx")
  psychomatic_report(screening = screen_items(items), format = "excel", file_name = path)
}
```

---

reverse_items *Reverse-score selected items*

Description

Reverses item scores using either a user-defined score range or the observed range of the selected items. Items can be selected explicitly or detected from negative item-rest correlations.

Usage

```
reverse_items(
  data,
  items = NULL,
  min_score = NULL,
  max_score = NULL,
  suffix = "_r",
  overwrite = FALSE,
  detect = FALSE
)
```

Arguments

data	Data frame or matrix containing item responses.
items	Character vector of item names to reverse. Required unless detect = TRUE.
min_score, max_score	Optional theoretical score range. If omitted, the observed range among selected items is used.
suffix	Suffix used for reversed columns when overwrite = FALSE.
overwrite	If TRUE, replace the original columns. If FALSE, append reversed columns.
detect	If TRUE, reverse items flagged by screen_items() as probable reverse-keyed items.

Value

A data frame with attributes reversed_items and reverse_range.

Examples

```
items <- data.frame(item1 = 1:5, item2 = 5:1)
reverse_items(items, items = "item2", min_score = 1, max_score = 5)
```

`score_scale` *Score one or more psychometric scales*

Description

Computes scale scores using sum or mean scoring, optional reverse scoring, optional person-mean imputation, and a minimum valid-response rule.

Usage

```
score_scale(
  data,
  keys = NULL,
  items = NULL,
  method = c("mean", "sum"),
  min_valid = 0.8,
  impute = c("none", "person_mean"),
  reverse = NULL,
  min_score = NULL,
  max_score = NULL
)
```

Arguments

`data`	Data frame or matrix containing item responses.
`keys`	Named list where each element contains the item names for one scale. If `NULL`, `items` are scored as a single scale named `total`.
`items`	Optional item names used when `keys = NULL`.
`method`	Scoring method: `"mean"` or `"sum"`.
`min_valid`	Minimum number or proportion of valid item responses required to compute a scale score.
`impute`	Missing-data handling: `"none"` or `"person_mean"`.
`reverse`	Optional character vector of items to reverse before scoring.
`min_score, max_score`	Optional score range used when `reverse` is provided.

Value

A list of class `psychomatic_scores` with score and reliability tables.

Examples

```
items <- data.frame(item1 = c(1, 2, NA), item2 = c(2, 3, 4), item3 = c(3, 4, 5))
score_scale(items, items = names(items), min_valid = 2 / 3)
```

`screen_items` *Screen item-level data quality*

Description

Computes a compact item-screening table for numeric psychometric items: missingness, variance, distribution shape, floor and ceiling proportions, item-rest correlations, and probable reverse-keyed items.

Usage

```
screen_items(
  data,
  digits = 3,
  floor_cutoff = 0.8,
  ceiling_cutoff = 0.8,
  min_item_rest = -0.1
)
```

Arguments

`data`	Data frame or matrix containing item responses.
`digits`	Number of decimal places used in the returned table.
`floor_cutoff`	Proportion at the minimum score used to flag floor effects.
`ceiling_cutoff`	Proportion at the maximum score used to flag ceiling effects.
`min_item_rest`	Item-rest correlation below which an item is flagged as probably reverse keyed or inconsistent with the remaining items.

Value

A data frame of class `psychomatic_item_screen`.

Examples

```
items <- data.frame(
  item1 = c(1, 2, 3, 4, 5),
  item2 = c(5, 4, 3, 2, 1),
  item3 = c(1, 1, 2, 2, 3)
)
screen_items(items)
```

summary.cfa_auto *Summarize an automated CFA result*

Description

Summarize an automated CFA result

Usage

```
## S3 method for class 'cfa_auto'
summary(object, ...)
```

Arguments

object	Object returned by cfa_auto().
...	Additional unused arguments.

Value

A list of tidy tables.

summary.efa_auto *Summarize an automated EFA result*

Description

Summarize an automated EFA result

Usage

```
## S3 method for class 'efa_auto'
summary(object, ...)
```

Arguments

object	Object returned by efa_auto().
...	Additional unused arguments.

Value

A list of tidy tables.

summary.factorial_invariance_auto *Summarize a factorial invariance result*

Description

Summarize a factorial invariance result

Usage

```
## S3 method for class 'factorial_invariance_auto'
summary(object, ...)
```

Arguments

object	Object returned by factorial_invariance_auto().
...	Additional unused arguments.

Value

A list with fit, decisions, and conclusion.

Index