

Package: nomologR (via r-universe)

September 28, 2026

Type Package

Title Guided Scale Development and Construct Validation

Version 0.9.0

Description Provides an evidence-guided workflow for empirical scale development and construct validation, designed for graduate students and researchers. The package helps users examine item performance, dimensionality, measurement models, reliability, convergent and discriminant evidence, measurement invariance, and theory-specified nomological networks in the sense of Cronbach and Meehl (1955) <[doi:10.1037/h0040957](https://doi.org/10.1037/h0040957)>. Methods are documented from historical to contemporary practice with literature references, and simulated teaching datasets with known population models accompany the examples. Following calls for transparent measurement practice such as Flake, Pek, and Hehman (2017) <[doi:10.1177/1948550617693063](https://doi.org/10.1177/1948550617693063)>, it emphasizes transparent diagnostics, reproducible decision logging, and teaching-oriented guidance rather than automatic item deletion or binary declarations that a measure is valid.

License GPL-3

Encoding UTF-8

Language en-US

LazyData true

Roxygen list(markdown = TRUE)

Depends R (>= 4.2)

Imports dplyr, EFAtools (>= 1.0.0), ggplot2, lavaan (>= 0.6-21), methods, psych, semTools (>= 0.5-9), tibble

Suggests covr, knitr, parallel, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

URL <https://github.com/JUhalt/nomologR>,
<https://juhalt.github.io/nomologR/>

BugReports <https://github.com/JUhalt/nomologR/issues>

Config/roxygen2/version 8.1.0

Repository <https://juhalt.r-universe.dev>

Date/Publication 2026-09-28 18:10:23 UTC

RemoteUrl <https://github.com/JUhalt/nomologR>

RemoteRef v0.9.0

RemoteSha b72bc6d98933083151cabf9b0b30619307afc8d1

Contents

nomo_apa_table	3
nomo_cfa	5
nomo_compare	8
nomo_defaults	11
nomo_demo_continuous	12
nomo_demo_network	13
nomo_demo_ordinal	14
nomo_demo_walkthrough	15
nomo_efa	17
nomo_expectations	19
nomo_factors	21
nomo_hierarchical	24
nomo_hypotheses	27
nomo_invariance	28
nomo_methods	31
nomo_missing	33
nomo_model	35
nomo_network	37
nomo_partial	41
nomo_reliability	42
nomo_report	45
nomo_revise	46
nomo_run	49
nomo_scores	52
nomo_screen	55
nomo_split	58
nomo_table	59
nomo_validity	61
plot.nomo_cfa	63
plot.nomo_compare	64
plot.nomo_efa	64
plot.nomo_factors	65
plot.nomo_hierarchical	66
plot.nomo_invariance	66
plot.nomo_network	67
plot.nomo_reliability	67

<i>nomo_apa_table</i>	3
plot.nomo_screen	68
plot.nomo_validity	69
print.nomo_factors	69
print.nomo_screen	70
print.summary_nomo_factors	70
print.summary_nomo_screen	71
summary.nomo_cfa	71
summary.nomo_compare	72
summary.nomo_efa	72
summary.nomo_factors	73
summary.nomo_reliability	73
summary.nomo_screen	74
summary.nomo_validity	74
Index	75

nomo_apa_table	<i>Manuscript-ready tables in APA style</i>
----------------	---

Description

Formats the evidence in a nomologR result as a table ready for a thesis, dissertation, or manuscript, following APA 7 conventions: a bold table number, an italic title, no vertical rules, and notes below the table in the order general, specific, probability.

Usage

```
nomo_apa_table(x, type = NULL, number = NULL, title = NULL, ...)
```

Arguments

x	A result object: nomo_cfa, nomo_reliability, nomo_validity, nomo_invariance, or nomo_network.
type	Which table to build. For nomo_cfa: "loadings", "fit", or "factor_correlations". For nomo_validity: "discriminant" or "convergent". For nomo_network: "hypotheses" or "fit". Other objects have one table each.
number	Optional table number, printed in bold as "Table 1".
title	Optional title; a descriptive default is supplied.
...	Unused.

Details

Leading zeros follow the statistic, not the value. APA 7 drops the leading zero only for statistics that *cannot* exceed 1. So *p* values, correlations, reliability coefficients, and CFI are written .95, while TLI, RMSEA, SRMR, and standardized loadings keep it, 0.95, because each can exceed 1 in principle: TLI is not bounded above, and a standardized loading does in an improper (Heywood)

solution. Many published tables print standardized loadings without the zero; this follows the rule as written.

No verdicts. Table notes keep the package's reference-value language. No cell reads PASS or FAIL, and fit indices are not labeled good or poor.

Discriminant evidence without the Fornell-Larcker matrix. For a `nomo_validity` result, the "discriminant" table gives each pair of constructs one row: the latent correlation with its 95% confidence interval (Rönkkö & Cho, 2022), and the heterotrait-monotrait ratios HTMT2 (Roemer et al., 2021) and HTMT (Henseler et al., 2015) when they were computed. It does not print the correlation matrix with the square root of AVE on its diagonal, because that comparison often misses discriminant-validity problems (Henseler et al., 2015). AVE is convergent evidence and has its own "convergent" table.

The rules come from the *Publication Manual of the American Psychological Association* (7th ed.), checked against Purdue OWL's APA 7 guides. Journal-specific templates are out of scope.

Experimental layout. The layout of these tables is experimental until `nomologR` 1.0.0 and may be adjusted as APA style is applied to more tables. Any change will be described in NEWS; see `?nomologR` for the stability policy.

Value

A `nomo_apa_table` object, which prints in the console and renders as a formatted table, with its title and notes, when knitted.

References

- American Psychological Association. (2020). *Publication manual of the American Psychological Association* (7th ed.). doi:10.1037/0000165000
- Fornell, C., & Larcker, D. F. (1981). Evaluating structural equation models with unobservable variables and measurement error. *Journal of Marketing Research*, 18(1), 39-50. doi:10.2307/3151312
- Henseler, J., Ringle, C. M., & Sarstedt, M. (2015). A new criterion for assessing discriminant validity in variance-based structural equation modeling. *Journal of the Academy of Marketing Science*, 43(1), 115-135. doi:10.1007/s1174701404038
- Roemer, E., Schuberth, F., & Henseler, J. (2021). HTMT2—An improved criterion for assessing discriminant validity in structural equation modeling. *Industrial Management & Data Systems*, 121(12), 2637-2650. doi:10.1108/IMDS0220210082
- Rönkkö, M., & Cho, E. (2022). An updated guideline for assessing discriminant validity. *Organizational Research Methods*, 25(1). doi:10.1177/1094428120968614

Examples

```
model <- '
  visual =~ x1 + x2 + x3
  textual =~ x4 + x5 + x6
  speed  =~ x7 + x8 + x9
'

cfa <- nomo_cfa(model, data = lavaan::HolzingerSwineford1939)

nomo_apa_table(cfa, "loadings", number = 1)
```

```

nomo_apr_table(cfa, "fit", number = 2)
nomo_apr_table(nomo_reliability(cfa), number = 3)

```

nomo_cfa

Guided confirmatory factor analysis

Description

`nomo_cfa()` fits a researcher-specified confirmatory factor model with `lavaan::cfa()` and adds a transparent guidance layer around estimator choice, convergence, standardized loadings, global fit, local residuals, Heywood diagnostics, and modification indices.

Usage

```

nomo_cfa(
  model,
  data,
  ordered = NULL,
  estimator = NULL,
  missing = NULL,
  std.lv = FALSE,
  control = NULL,
  modification_indices = TRUE,
  mi_top = 10L,
  guidance = nomo_defaults()
)

```

Arguments

<code>model</code>	A researcher-specified lavaan measurement-model syntax string or an object created by <code>nomo_model()</code> .
<code>data</code>	A data frame containing the model indicators.
<code>ordered</code>	Optional character vector naming binary/ordinal indicators. When supplied and <code>estimator = NULL</code> , <code>nomo_cfa()</code> explicitly requests WLSMV.
<code>estimator</code>	Optional lavaan estimator. For continuous indicators, leaving this as <code>NULL</code> preserves <code>lavaan::cfa()</code> 's default. Robust ML variants such as "MLR" remain explicit researcher choices.
<code>missing</code>	Optional lavaan missing-data option such as "fiml" for continuous ML models or "pairwise" where supported.
<code>std.lv</code>	Logical. Passed directly to <code>lavaan::cfa()</code> .
<code>control</code>	Optional named list passed to lavaan's optimizer through <code>lavaan::cfa(control = ...)</code> . This is an advanced troubleshooting/reproducibility option; non-default optimizer controls are recorded in the decision log.

modification_indices	Logical; if TRUE, compute modification indices as quarantined diagnostics. They never trigger automatic respecification.
mi_top	Number of largest modification indices to retain in the compact \$top_modification_indices view. The full table remains available in \$modification_indices.
guidance	Guidance settings from <code>nomo_defaults()</code> .

Details

The fitted lavaan object is retained unchanged in \$fit. nomologR does not free parameters, add residual covariances, delete indicators, or refit a different model in response to fit statistics or modification indices.

Value

A nomo_cfa object. The fields to read are:

- fit: the unchanged lavaan fit, for anything else lavaan reports.
- model: the model syntax fitted.
- converged, estimator, data_n, n_used, and sample_summary.
- fit_evidence: global fit indices with their teaching references.
- standardized_loadings: one row per loading, with its interval, flag, and explanation.
- factor_correlations: latent correlations with intervals.
- heywood: improper-solution signals, if any.
- parameter_estimates and standardized_solution: lavaan's parameter tables, as tibbles.
- residual_matrix and residual_pairs: residual correlations, the pairs largest first.
- modification_indices and top_modification_indices: diagnostics only; nothing is freed.
- engine_warnings: warnings lavaan raised.
- decision_log.

Other fields record the call, the settings used, and intermediate engine results. They may change between releases and are not part of the stable interface (see ?nomologR).

References

Historical foundations:

Bentler, P. M., & Bonett, D. G. (1980). Significance tests and goodness of fit in the analysis of covariance structures. *Psychological Bulletin*, 88(3), 588-606. doi:10.1037/00332909.88.3.588

Jöreskog, K. G. (1969). A general approach to confirmatory maximum likelihood factor analysis. *Psychometrika*, 34(2), 183-202. doi:10.1007/BF02289343

Fit evaluation:

Bentler, P. M. (1990). Comparative fit indexes in structural models. *Psychological Bulletin*, 107(2), 238-246. doi:10.1037/00332909.107.2.238

Browne, M. W., & Cudeck, R. (1992). Alternative ways of assessing model fit. *Sociological Methods & Research*, 21(2), 230-258. doi:10.1177/0049124192021002005

Hu, L., & Bentler, P. M. (1999). Cutoff criteria for fit indexes in covariance structure analysis: Conventional criteria versus new alternatives. *Structural Equation Modeling*, 6(1), 1-55. doi:10.1080/10705519909540118

Marsh, H. W., Hau, K.-T., & Wen, Z. (2004). In search of golden rules: Comment on hypothesis-testing approaches to setting cutoff values for fit indexes and dangers in overgeneralizing Hu and Bentler's (1999) findings. *Structural Equation Modeling*, 11(3), 320-341. doi:10.1207/s15328007sem1103_2

Estimation, respecification, and improper solutions:

Flora, D. B., & Curran, P. J. (2004). An empirical evaluation of alternative methods of estimation for confirmatory factor analysis with ordinal data. *Psychological Methods*, 9(4), 466-491. doi:10.1037/1082989X.9.4.466

Kolenikov, S., & Bollen, K. A. (2012). Testing negative error variances: Is a Heywood case a symptom of misspecification? *Sociological Methods & Research*, 41(1), 124-167. doi:10.1177/0049124112442138

MacCallum, R. C., Roznowski, M., & Necowitz, L. B. (1992). Model modifications in covariance structure analysis: The problem of capitalization on chance. *Psychological Bulletin*, 111(3), 490-504. doi:10.1037/00332909.111.3.490

Rhemtulla, M., Brosseau-Liard, P. É., & Savalei, V. (2012). When can categorical variables be treated as continuous? A comparison of robust continuous and categorical SEM estimation methods under suboptimal conditions. *Psychological Methods*, 17(3), 354-373. doi:10.1037/a0029315

Rosseel, Y. (2012). lavaan: An R package for structural equation modeling. *Journal of Statistical Software*, 48(2), 1-36. doi:10.18637/jss.v048.i02

Examples

```
model <- '
  visual =~ x1 + x2 + x3
  textual =~ x4 + x5 + x6
  speed   =~ x7 + x8 + x9
'

out <- nomo_cfa(model, data = lavaan::HolzingerSwineford1939)
summary(out)

# Declared ordered indicators request WLSMV rather than ML
ordinal_model <- nomo_model(list(
  A = c("a1", "a2", "a3", "a4", "a5"),
  B = c("b1", "b2", "b3", "b4", "b5")
))
out_ord <- nomo_cfa(
  ordinal_model,
  data = nomo_demo_ordinal,
  ordered = names(nomo_demo_ordinal)
)
out_ord$fit_evidence
```

 nomo_compare

 Compare confirmatory measurement models

Description

nomo_compare() places two or more fitted [nomo_cfa\(\)](#) models side by side and reports the evidence that bears on the comparison: a difference test matched to the estimator when the models are nested, changes in global fit indices, information criteria when they are defined, and standardized loadings, reliability, and construct-separation evidence for each model. A researcher rationale is required and recorded. The function never selects a "winning" model.

Usage

```
nomo_compare(
  ...,
  rationale,
  origin = c("a_priori", "post_hoc"),
  nested = c("auto", "yes", "no"),
  reference = 1L,
  method = "default",
  evidence = TRUE,
  guidance = nomo_defaults()
)
```

Arguments

...	Two or more nomo_cfa objects. Argument names become model labels, for example <code>nomo_compare(full = cfa_full, reduced = cfa_reduced, rationale = "...")</code> .
rationale	Required character scalar recording why these models are compared (for example, the theoretical question each model represents).
origin	Whether the comparison was planned before seeing results ("a_priori") or prompted by results such as modification indices or residuals ("post_hoc"). Post-hoc comparisons are flagged in the decision log.
nested	"auto" (default) checks nesting automatically; "yes" declares the models nested; "no" declares them non-nested and skips the difference test.
reference	The model every other model is compared with, as an index or label. Defaults to the first model.
method	Difference-test method passed to lavaan::lavTestLRT() . "default" lets lavaan choose the method appropriate to the estimator.
evidence	Logical. If TRUE (default), compute side-by-side reliability (nomo_reliability()) and convergent/discriminant (nomo_validity()) evidence for each model.
guidance	Guidance settings from nomo_defaults() .

Details

Requirements. All models must be converged `nomo_cfa` objects fitted with the same estimator and missing-data handling to the same cases and data. Otherwise the comparison is refused with an explanation, because test statistics and fit indices would not be comparable.

Nesting. With `nested = "auto"`, nesting is checked from the models' implied moments with `semTools::net()` (Bentler & Satorra, 2010). The difference test is reported only for nested models. If the check cannot run (for example for some categorical models), set `nested = "yes"` when one model is obtained from the other by fixing or constraining parameters. A declaration that the check contradicts is recorded as a concern and no test is reported.

Difference tests. Tests come from `lavaan::lavTestLRT()`: the ordinary chi-square difference test for ML, the scaled difference test for robust ML estimators (Satorra & Bentler, 2001), and the scaled-and-shifted test for categorical estimators such as WLSMV (Satorra, 2000). The method lavaan used is reported. A test that lavaan cannot compute, or a negative scaled statistic, is reported as unavailable with lavaan's message rather than replaced by a different method; `method` can request an alternative explicitly.

Information criteria. AIC and BIC are reported for likelihood-based estimators when the models contain the same observed variables. They are not defined for WLSMV and are reported as unavailable, not substituted.

Removing an item. Models with different observed variables describe different data, so neither a difference test nor information criteria applies; only descriptive evidence is shown. To test whether an item is needed, keep it in both models and fix its loading to zero in the reduced model (for example $B \sim b_1 + b_2 + b_3 + b_4 + 0 \cdot b_5$). Reliability for such a model still includes the zero-loading item in the composite, and the evidence table says so.

Value

A `nomo_compare` object. The fields to read are:

- `models`: fit and information criteria for each model.
- `comparisons`: for each model against the reference, the nesting relation, the difference test, changes in fit and information criteria, and an interpretation.
- `loadings`: standardized loadings side by side.
- `evidence`: reliability, AVE, and HTMT2 by model, when `evidence = TRUE`.
- `fits`: the fitted `nomo_cfa` objects, and `engine_warnings`, the warnings lavaan raised for each.
- `reference`, `rationale`, `origin`, `references`, and `decision_log`.

Other fields record the call, the settings used, and intermediate engine results. They may change between releases and are not part of the stable interface (see `?nomologR`).

References

- Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6), 716-723. doi:10.1109/TAC.1974.1100705
- Bentler, P. M., & Satorra, A. (2010). Testing model nesting and equivalence. *Psychological Methods*, 15(2), 111-123. doi:10.1037/a0019625

- Burnham, K. P., & Anderson, D. R. (2004). Multimodel inference: Understanding AIC and BIC in model selection. *Sociological Methods & Research*, 33(2), 261-304. doi:10.1177/0049124104268644
- Chen, F. F. (2007). Sensitivity of goodness of fit indexes to lack of measurement invariance. *Structural Equation Modeling*, 14(3), 464-504. doi:10.1080/10705510701301834
- Cheung, G. W., & Rensvold, R. B. (2002). Evaluating goodness-of-fit indexes for testing measurement invariance. *Structural Equation Modeling*, 9(2), 233-255. doi:10.1207/S15328007SEM0902_5
- MacCallum, R. C., Roznowski, M., & Necowitz, L. B. (1992). Model modifications in covariance structure analysis: The problem of capitalization on chance. *Psychological Bulletin*, 111(3), 490-504. doi:10.1037/00332909.111.3.490
- Raftery, A. E. (1995). Bayesian model selection in social research. *Sociological Methodology*, 25, 111-163. doi:10.2307/271063
- Satorra, A. (2000). Scaled and adjusted restricted tests in multi-sample analysis of moment structures. In *Innovations in multivariate statistical analysis* (pp. 233-247). Springer. doi:10.1007/9781461546030_17
- Satorra, A., & Bentler, P. M. (2001). A scaled difference chi-square test statistic for moment structure analysis. *Psychometrika*, 66(4), 507-514. doi:10.1007/BF02296192
- Satorra, A., & Bentler, P. M. (2010). Ensuring positiveness of the scaled difference chi-square test statistic. *Psychometrika*, 75(2), 243-248. doi:10.1007/s113360099135y
- Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, 6(2), 461-464. doi:10.1214/aos/1176344136

See Also

`nomo_cfa()` to fit the models.

Examples

```
full <- nomo_cfa(
  "A =~ a1 + a2 + a3 + a4 + a5\nB =~ b1 + b2 + b3 + b4 + b5",
  data = nomo_demo_continuous
)
# Keep b5 in the data but fix its loading to zero to test whether it is needed
no_b5 <- nomo_cfa(
  "A =~ a1 + a2 + a3 + a4 + a5\nB =~ b1 + b2 + b3 + b4 + 0*b5",
  data = nomo_demo_continuous
)

cmp <- nomo_compare(
  full = full,
  no_b5 = no_b5,
  rationale = "Evaluate whether the weakly loading item b5 contributes to factor B.",
  evidence = FALSE
)
cmp
nomo_table(cmp, "comparisons")

# Side-by-side loadings, reliability, AVE, and HTMT2 for each model
```

```
cmp_evidence <- nomo_compare(
  full = full,
  no_b5 = no_b5,
  rationale = "Evaluate whether the weakly loading item b5 contributes to factor B."
)
summary(cmp_evidence)
plot(cmp_evidence, type = "loadings")
```

nomo_defaults

Default guidance settings for nomologR

Description

The numerical values returned by this function are teaching/reference points, not universal pass/fail criteria. They are intended to trigger inspection and explanation in downstream nomologR functions.

Usage

```
nomo_defaults(profile = "teaching")
```

Arguments

profile	Guidance profile. Only "teaching" is implemented during the initial development series.
---------	---

Details

Each value is a commonly taught reference point from the literature cited in the corresponding analysis function (for example, [nomo_efa\(\)](#), [nomo_cfa\(\)](#), [nomo_reliability\(\)](#), and [nomo_validity\(\)](#)). Changing a reference changes which evidence is flagged for review; it never deletes items, respecifies models, or declares validity.

Value

A named list of guidance settings.

Examples

```
guidance <- nomo_defaults()
guidance$efa_loading_reference
guidance$fit_reference

# A deliberately stricter loading reference flags more items for review.
stricter <- nomo_defaults()
stricter$efa_loading_reference <- 0.60
efa <- nomo_efa(nomo_demo_continuous, factors = 2, guidance = stricter)
efa$item_summary[, c("item", "primary_loading", "attention")]
```

nomo_demo_continuous *Simulated two-factor item data with known teaching features*

Description

A simulated scale-development dataset with two correlated latent factors and five candidate indicators per factor. Because the population model is known, learners can compare nomologR evidence with the structure that actually generated the data.

Usage

```
nomo_demo_continuous
```

Format

A data frame with 500 rows and 10 numeric columns:

a1, a2, a3, a4, a5 Indicators written for factor A. Population standardized loadings are .80, .75, .70, .72, and .45 (with a .35 cross-loading on factor B for a5).

b1, b2, b3, b4, b5 Indicators written for factor B. Population standardized loadings are .78, .74, .80, .70, and .30.

Details

The data are deliberately imperfect in ways that scale developers routinely encounter:

- a5 cross-loads on both factors (population loadings .45 and .35);
- b5 is a weak indicator (population loading .30);
- a2 (15 cases) and b3 (12 cases) contain values missing completely at random.

These features are included so that item-audit, factor-retention, and EFA output have something substantive to explain. They are not instructions to delete a5 or b5; whether such items remain depends on theory, content coverage, and later evidence.

Population model: factors A and B are standard normal with correlation .40. Each indicator is a linear function of its factor(s) plus normal unique variance chosen so that the indicator has unit population variance. Scores are reported on a continuous rating metric (population mean 4, SD 1) and rounded to two decimals; the linear transformation does not affect correlations or standardized estimates.

Source

Simulated with a fixed seed by `data-raw/nomo_demo.R` in the package source repository.

See Also

[nomo_demo_ordinal](#) for five-category ordered versions of the same latent responses and [nomo_demo_network](#) for a multi-construct validation study.

Examples

```
str(nomo_demo_continuous)
colSums(is.na(nomo_demo_continuous))

scr <- nomo_screen(nomo_demo_continuous)
scr
```

nomo_demo_network	<i>Simulated multi-construct validation study</i>
-------------------	---

Description

A simulated construct-validation dataset for confirmatory measurement, measurement-invariance, and theory-specified nomological-network examples. Three latent constructs are measured by multiple indicators, an observed outcome is available, and responses were collected in two administration groups.

Usage

```
nomo_demo_network
```

Format

A data frame with 800 rows and 13 columns:

ag1, ag2, ag3, ag4 Agency indicators. Population standardized loadings .80, .75, .70, .78.
pe1, pe2, pe3, pe4 Persistence indicators. Population standardized loadings .78, .72, .76, .70.
sd1, sd2, sd3 Social desirability indicators. Population standardized loadings .70, .75, .65.
Performance Observed outcome score (population mean 70, SD 10).
group Administration group: a factor with levels "online" and "paper" (400 cases each).

Details

Within each group, the population structural model is:

- Persistence is regressed on Agency with a standardized coefficient of .45;
- Performance is regressed on Agency (.40) and **not** on Persistence (0);
- Agency and Social desirability are uncorrelated (0).

Because Persistence is correlated with Agency, Persistence and Performance are still correlated marginally (about .18 in the population) even though the direct Persistence-to-Performance path is zero. This makes the dataset useful for teaching the difference between a marginal association and a theory-specified structural path.

Measurement-invariance teaching features: all loadings are equal across groups, the latent Agency mean is .25 SD higher in the paper group, and the intercept of ag3 is .50 higher in the paper group. The ag3 intercept is therefore a known source of scalar non-invariance. Indicators are reported on a continuous rating metric (population mean 4, SD 1 in the online group) and rounded to two decimals.

Source

Simulated with a fixed seed by `data-raw/nomo_demo.R` in the package source repository.

See Also

[nomo_network\(\)](#), [nomo_invariance\(\)](#), [nomo_run\(\)](#)

Examples

```
str(nomo_demo_network)
table(nomo_demo_network$group)
```

nomo_demo_ordinal	<i>Simulated five-category ordered item data</i>
-------------------	--

Description

An ordered-categorical version of [nomo_demo_continuous](#). The same latent item responses (before missing values were introduced) were cut into five ordered response categories, as is typical of Likert-type rating scales. This dataset supports polychoric factor-retention, ordinal EFA, WLSMV CFA, and ordinal reliability examples.

Usage

```
nomo_demo_ordinal
```

Format

A data frame with 500 rows and 10 ordered factors with levels "1" < "2" < "3" < "4" < "5":

a1, a2, a3, a4, a5 Ordered indicators written for factor A (a5 cross-loads on factor B).

b1, b2, b3, b4, b5 Ordered indicators written for factor B (b5 is weak).

Details

Latent responses were thresholded at -1.80, -0.80, 0.20, and 1.20, so the observed distributions lean toward the upper categories. The dataset has no missing values, which keeps the first ordinal examples focused on estimator and correlation choices rather than on missing-data handling for categorical models.

Source

Simulated with a fixed seed by `data-raw/nomo_demo.R` in the package source repository.

See Also

[nomo_demo_continuous](#), [nomo_demo_network](#)

Examples

```
str(nomo_demo_ordinal)
table(nomo_demo_ordinal$a1)
```

nomo_demo_walkthrough *Shared walkthrough data from content review to empirical screening*

Description

Simulated responses of 400 students to twelve candidate items for a two-facet Study Persistence scale. These are the shared teaching data of the joint walkthrough with contentvalidR: the same twelve items went through an expert item sort there, and the ten it carried forward are screened here. The data are simulated. No participant was involved, and the construct, its facets, and every item stem were written for the example.

Usage

```
nomo_demo_walkthrough
nomo_demo_walkthrough_items
```

Format

nomo_demo_walkthrough is a data frame with 400 rows and 14 columns:

respondent Respondent number, 1 to 400.

cohort A factor with levels "A" and "B", 200 respondents each.

EF1, EF2, EF3, EF4, EF5, EF6 Effort Regulation items, integer responses 1 to 5, as answered (EF2 is not recoded).

TF1, TF2, TF3, TF4, TF5, TF6 Task Focus items, integer responses 1 to 5, as answered (TF2 is not recoded).

nomo_demo_walkthrough_items is a data frame with 12 rows and 5 columns: item; facet ("EF" or "TF"); stem, the item's wording; reverse_worded, a logical; and role, what the item was built to do.

An object of class data.frame with 12 rows and 5 columns.

Details

Each item was built to behave in a particular way at one stage or both, so the two stages can be seen to disagree. nomo_demo_walkthrough_items states what each item was built to do, in its role column:

- EF1, TF1, TF3: ordinary items, which both stages accept.
- EF2, TF2: reverse-worded. A high answer means less persistence, so until they are recoded (6 minus the response) each correlates negatively with its own facet. That is a coding matter, not evidence against the item.

- EF3: endorsed by almost everyone, so its answers pile up at the top of the scale and every correlation it has is attenuated. An empirical screen flags it; it is also the only item about finishing required work, so dropping it narrows the domain the panel defined.
- EF4: passes content review (18 of 20 judges) and carries almost no common variance. Content review cannot see this.
- EF5, TF5: fail content review, so a handoff holds them back. They stay in the data so a reader can see what keeping them would have done.
- EF6: meets the content criterion by one judge, then behaves well.
- TF4: loads on both facets.
- TF6: shifted upward in cohort B (a classroom rule about phones), a known source of non-invariance between cohorts.

Population model, per respondent:

- Two facets, Effort Regulation and Task Focus, standard normal with correlation .45.
- Each item's latent response is its loadings times the facets plus normal error, scaled to unit variance, and cut into five categories at the normal quantiles .10, .30, .60, and .85.
- Standardized loadings on EF: EF1 .72, EF2 .68, EF3 .60, EF4 .15, EF5 .55, EF6 .60, and TF4 .45.
- Loadings on TF: TF1 .70, TF2 .66, TF3 .62, TF4 .40, TF5 .58, TF6 .64.
- EF3's latent response is shifted up by 1.75, and TF6's by .35 in cohort B. EF2 and TF2 are answered in the opposite direction (6 minus the category).

The content-review half of the example, the expert sort of these items, and the handoff it produced are in `contentvalidR`. The handoff for these items is stored with this package as `system.file("extdata", "content-handoff-walkthrough.rds", package = "nomologR")`.

Source

Generated by `contentvalidR`'s `data-raw/build-walkthrough-data.R` (seed 20260921), copied unchanged from its v0.9.0 release into this package's `data-raw/walkthrough/`, and converted by `data-raw/nomo_demo_walkthrough.R`.

See Also

`nomo_screen()`, `nomo_run()`, and `vignette("content-review", package = "nomologR")`.

Examples

```
str(nomo_demo_walkthrough)
nomo_demo_walkthrough_items[, c("item", "role")]

# Answered as written, a reverse-worded item runs against its facet.
cor(nomo_demo_walkthrough$EF1, nomo_demo_walkthrough$EF2)
cor(nomo_demo_walkthrough$EF1, 6 - nomo_demo_walkthrough$EF2)
```

Description

`nomo_efa()` estimates a common-factor exploratory factor analysis while keeping the consequential analytical choices visible. Oblique rotation is the default, item-level diagnostics are framed as prompts for inspection, and no item is automatically deleted or model silently refit.

Usage

```
nomo_efa(
  data,
  items = NULL,
  factors,
  factor_count = NULL,
  rotation = "oblimin",
  fm = "minres",
  correlation = NULL,
  types = NULL,
  missing = NULL,
  smooth = NULL,
  guidance = nomo_defaults()
)
```

Arguments

<code>data</code>	A data frame containing candidate items.
<code>items</code>	Optional character vector identifying item columns. If <code>factors</code> is a <code>nomo_factors</code> object and <code>items = NULL</code> , its item set is inherited. Otherwise all columns are used when <code>items = NULL</code> .
<code>factors</code>	A positive integer number of factors to extract, or a <code>nomo_factors</code> object. For a <code>nomo_factors</code> object, the selected parallel-analysis factor count is used unless <code>factor_count</code> is supplied.
<code>factor_count</code>	Optional positive integer. This may be supplied only when <code>factors</code> is a <code>nomo_factors</code> object. It keeps that result's items, modeling types, correlations, and missing-data handling while recording a researcher-selected factor count, rather than implying that the parallel-analysis suggestion was adopted.
<code>rotation</code>	Rotation passed to <code>psych::fa()</code> . The default is "oblimin". Orthogonal rotations are allowed but are recorded as a researcher choice.
<code>fm</code>	Common-factor extraction method passed to <code>psych::fa()</code> . The default is "minres". Supported values are "minres", "uls", "ols", "wls", "gls", "pa", "ml", "minchi", "minrank", "alpha", and "old.min".
<code>correlation</code>	Optional correlation strategy: "auto", "pearson", "polychoric", "tetrachoric", or "mixed". If <code>NULL</code> , the value is inherited from a supplied <code>nomo_factors</code> object when possible; otherwise "auto" is used.

types	Optional named character vector declaring selected items as "continuous", "ordinal", or "binary". If NULL, modeling types are inherited from a supplied nomo_factors object when possible; otherwise they are inferred using the same conservative rules as <code>nomo_factors()</code> .
missing	Optional missing-data strategy: "pairwise" or "complete". If NULL, the value is inherited from a supplied nomo_factors object when possible; otherwise "pairwise" is used.
smooth	Optional logical. If NULL (default), smoothing is inherited from a supplied nomo_factors object when that object explicitly used smoothing; otherwise FALSE. If FALSE, a non-positive-definite correlation matrix stops the analysis. If TRUE, <code>psych::cor.smooth()</code> is used explicitly and the intervention is recorded.
guidance	Guidance settings from <code>nomo_defaults()</code> .

Details

The factor count may be supplied directly as a positive integer or by passing a nomo_factors object. When a nomo_factors object is supplied, its primary parallel-analysis suggestion is used as the requested EFA factor count and, unless overridden, its item set, modeling types, correlation model, and missing-data strategy are carried forward.

Value

An object of class nomo_efa. The fields to read are:

- items, n_factors, and factor_source, which records whether the count was the researcher's or taken from a nomo_factors result.
- correlation_method and correlation_matrix: the correlations analyzed.
- pattern_matrix, structure_matrix, and factor_correlations.
- communalities, uniquenesses, and complexity.
- item_summary: one row per item, with its primary and secondary loadings, communality, flags, and the explanation of any flag.
- residual_matrix, residual_pairs, and rmsr: local misfit.
- sample_adequacy: sample size, KMO, and Bartlett's test.
- decision_log.

Other fields record the call, the settings used, and intermediate engine results. They may change between releases and are not part of the stable interface (see ?nomologR).

References

- Browne, M. W. (2001). An overview of analytic rotation in exploratory factor analysis. *Multivariate Behavioral Research*, 36(1), 111-150. doi:10.1207/S15327906MBR3601_05
- Conway, J. M., & Huffcutt, A. I. (2003). A review and evaluation of exploratory factor analysis practices in organizational research. *Organizational Research Methods*, 6(2), 147-168. doi:10.1177/1094428103251541

Costello, A. B., & Osborne, J. W. (2005). Best practices in exploratory factor analysis: Four recommendations for getting the most from your analysis. *Practical Assessment, Research, and Evaluation*, 10, Article 7. doi:[10.7275/jyj14868](https://doi.org/10.7275/jyj14868)

Fabrigar, L. R., Wegener, D. T., MacCallum, R. C., & Strahan, E. J. (1999). Evaluating the use of exploratory factor analysis in psychological research. *Psychological Methods*, 4(3), 272-299. doi:[10.1037/1082989X.4.3.272](https://doi.org/10.1037/1082989X.4.3.272)

Kaiser, H. F. (1958). The varimax criterion for analytic rotation in factor analysis. *Psychometrika*, 23(3), 187-200. doi:[10.1007/BF02289233](https://doi.org/10.1007/BF02289233)

Watkins, M. W. (2018). Exploratory factor analysis: A guide to best practice. *Journal of Black Psychology*, 44(3), 219-246. doi:[10.1177/0095798418771807](https://doi.org/10.1177/0095798418771807)

Examples

```
efa <- nomo_efa(nomo_demo_continuous, factors = 2)
efa
efa$item_summary[, c("item", "primary_loading", "secondary_loading", "attention")]

# Carry retention evidence and its modeling decisions into the EFA
fac <- nomo_factors(nomo_demo_continuous, n_iter = 20, seed = 2026)
efa_from_evidence <- nomo_efa(nomo_demo_continuous, factors = fac)
summary(efa_from_evidence)
```

nomo_expectations	<i>Specify directional or negligible theoretical expectations</i>
-------------------	---

Description

These helpers create machine-readable theoretical expectations for use with `nomo_hypotheses()`. They describe what theory predicts before the empirical network is evaluated.

Usage

```
positive(
  min = NULL,
  max = NULL,
  scale = c("standardized", "unstandardized"),
  origin = c("a_priori", "post_hoc")
)

negative(
  max = NULL,
  min = NULL,
  scale = c("standardized", "unstandardized"),
  origin = c("a_priori", "post_hoc")
)
```

```
negligible(
  within = NULL,
  scale = c("standardized", "unstandardized"),
  origin = c("a_priori", "post_hoc")
)
```

Arguments

min	Optional finite lower bound. For <code>positive()</code> it must be greater than zero. For <code>negative()</code> it must be less than zero.
max	Optional finite upper bound. For <code>positive()</code> it must be greater than zero. For <code>negative()</code> it must be less than zero.
scale	Scale on which the expectation is defined. Standardized coefficients are the default because magnitude expectations such as <code>.20</code> are otherwise not portable across arbitrary raw units.
origin	Whether the expectation was specified <code>a_priori</code> or added <code>post_hoc</code> . Post-hoc expectations remain machine-readable but are never presented as confirmatory evidence.
within	Optional length-two finite numeric vector defining the negligible-effect region. The interval must contain zero.

Details

`positive()` and `negative()` can express direction alone or add a researcher-specified magnitude boundary. `negligible()` deliberately distinguishes an unquantified null-like expectation from a quantitative smallest effect size of interest (SESOI) region.

A bare `negligible()` expectation is **not confirmable as negligible** from a non-significant p-value. Supply `within = c(lower, upper)` when theory or the study design provides a defensible negligible-effect region.

Value

An object of class `nomo_expectation`.

Examples

```
positive()
positive(min = .20)
negative(max = -.20)
negligible()
negligible(within = c(-.10, .10))
```

nomo_factors

Evaluate evidence about the number of latent factors

Description

`nomo_factors()` combines multiple pieces of factor-retention evidence rather than treating any single rule as definitive. The primary retention evidence is a common-factor parallel analysis, complemented by Velicer's original and revised MAP criteria and, under the default `criterion_set = "core"`, the empirical Kaiser criterion (EKC). Extended criterion sets can add NEST, Hull (CAF), comparison data, and the legacy Kaiser-Guttman rule when their assumptions are compatible with the analyzed data. Scree information, Kaiser-Meyer-Olkin (KMO) sampling adequacy, and Bartlett's test are returned as supporting diagnostics.

Usage

```
nomo_factors(
  data,
  items = NULL,
  correlation = c("auto", "pearson", "polychoric", "tetrachoric", "mixed"),
  types = NULL,
  missing = c("pairwise", "complete"),
  criterion_set = c("core", "minimal", "extended", "all"),
  parallel_rule = c("percentile", "mean", "crawford"),
  n_iter = NULL,
  quantile = NULL,
  max_factors = NULL,
  seed = 1234L,
  fm = "minres",
  smooth = FALSE,
  guidance = nomo_defaults()
)
```

Arguments

<code>data</code>	A data frame containing candidate items.
<code>items</code>	Optional character vector identifying item columns. If <code>NULL</code> , all columns are treated as candidate items.
<code>correlation</code>	Correlation strategy: "auto", "pearson", "polychoric", "tetrachoric", or "mixed".
<code>types</code>	Optional named character vector overriding modeling types for selected items. Allowed values are "continuous", "ordinal", and "binary". Explicit overrides are applied before default-type rejection, so researchers can intentionally model otherwise ambiguous storage (for example, an unordered factor whose levels already encode a substantive order). Overrides do not reorder, relabel, or recode the supplied data. For example, <code>c(item1 = "ordinal", item2 = "ordinal")</code> .

missing	Missing-data handling for correlation estimation. "pairwise" uses pairwise-complete observations; "complete" restricts the analysis to cases complete on all selected items.
criterion_set	Retention-criterion bundle. "minimal" uses parallel analysis plus original MAP; "core" (default) adds revised MAP and EKC; "extended" adds NEST and Hull where supported; "all" additionally requests comparison data and the legacy Kaiser-Guttman rule. Criteria whose assumptions are not compatible with the current data are explicitly marked as skipped rather than silently substituted.
parallel_rule	Parallel-analysis decision rule: "percentile" (default), "mean", or "crawford". All three rules are computed from the same null simulations and retained in the result as sensitivity evidence.
n_iter	Number of null-data iterations used for parallel analysis. If NULL, the value in <code>guidance\$factor_parallel_iterations</code> is used.
quantile	Quantile of null eigenvalues used as the parallel-analysis reference. If NULL, the value in <code>guidance\$factor_parallel_quantile</code> is used.
max_factors	Maximum number of factors/components evaluated for MAP. If NULL, up to 10 or $p - 1$, whichever is smaller, are evaluated.
seed	Integer seed for the null-data simulation. The caller's random number state is restored before return.
fm	Common-factor extraction method passed to <code>psych::fa()</code> when obtaining factor eigenvalues. The default is "minres".
smooth	Logical. If FALSE (default), a non-positive-definite observed correlation matrix blocks factor-retention analysis. If TRUE, smoothing is explicit, recorded, and performed with <code>psych::cor.smooth()</code> .
guidance	Guidance settings from <code>nomo_defaults()</code> .

Details

Correlation choice is explicit and visible. With `correlation = "auto"`, continuous indicators use Pearson correlations, all-binary indicators use tetrachoric correlations, ordinal/binary indicators use polychoric correlations, and genuinely mixed indicator sets use mixed correlations. Numeric variables with a small number of integer-like response values are treated conservatively as continuous unless the user explicitly overrides their modeling type through types.

The function does not claim that a scale "has exactly" a particular number of factors. It reports which factor counts deserve investigation and records disagreements among retention methods.

Value

An object of class `nomo_factors`. The fields to read are:

- `items`, `n_cases`, and `n_items`.
- `item_types`: each item's screened and modeling type.
- `correlation_method` and `correlation_matrix`: the correlations analyzed.
- `kmo` and `bartlett`: sampling-adequacy evidence.
- `parallel`, `map`, and `scree`: each criterion's result.

- `criterion_status`: which criteria ran, and why any did not.
- `evidence`, `family_evidence`, and `family_concordance`: the suggested factor counts by method and by family of related methods.
- `plausible_factors` and `recommendation`: the synthesis, which is evidence for a researcher's choice, not a choice.
- `decision_log`.

Other fields record the call, the settings used, and intermediate engine results. They may change between releases and are not part of the stable interface (see `?nomologR`).

References

Historical retention rules shown as context:

Cattell, R. B. (1966). The scree test for the number of factors. *Multivariate Behavioral Research*, 1(2), 245-276. doi:10.1207/s15327906mbr0102_10

Guttman, L. (1954). Some necessary conditions for common-factor analysis. *Psychometrika*, 19(2), 149-161. doi:10.1007/BF02289162

Kaiser, H. F. (1960). The application of electronic computers to factor analysis. *Educational and Psychological Measurement*, 20(1), 141-151. doi:10.1177/001316446002000116

Contemporary retention evidence:

Achim, A. (2017). Testing the number of required dimensions in exploratory factor analysis. *The Quantitative Methods for Psychology*, 13(1), 64-74. doi:10.20982/tqmp.13.1.p064

Braeken, J., & van Assen, M. A. L. M. (2017). An empirical Kaiser criterion. *Psychological Methods*, 22(3), 450-466. doi:10.1037/met0000074

Crawford, A. V., Green, S. B., Levy, R., Lo, W.-J., Scott, L., Svetina, D., & Thompson, M. S. (2010). Evaluation of parallel analysis methods for determining the number of factors. *Educational and Psychological Measurement*, 70(6), 885-901. doi:10.1177/0013164410379332

Horn, J. L. (1965). A rationale and test for the number of factors in factor analysis. *Psychometrika*, 30(2), 179-185. doi:10.1007/BF02289447

Lorenzo-Seva, U., Timmerman, M. E., & Kiers, H. A. L. (2011). The Hull method for selecting the number of common factors. *Multivariate Behavioral Research*, 46(2), 340-364. doi:10.1080/00273171.2011.564527

Ruscio, J., & Roche, B. (2012). Determining the number of factors to retain in an exploratory factor analysis using comparison data of known factorial structure. *Psychological Assessment*, 24(2), 282-292. doi:10.1037/a0025697

Velicer, W. F. (1976). Determining the number of components from the matrix of partial correlations. *Psychometrika*, 41(3), 321-327. doi:10.1007/BF02293557

Velicer, W. F., Eaton, C. A., & Fava, J. L. (2000). Construct explication through factor or component analysis: A review and evaluation of alternative procedures for determining the number of factors or components. In R. D. Goffin & E. Helmes (Eds.), *Problems and solutions in human assessment* (pp. 41-71). Springer. doi:10.1007/9781461543978_3

Supporting adequacy diagnostics:

Bartlett, M. S. (1950). Tests of significance in factor analysis. *British Journal of Statistical Psychology*, 3(2), 77-85. doi:10.1111/j.20448317.1950.tb00285.x

Kaiser, H. F. (1974). An index of factorial simplicity. *Psychometrika*, 39(1), 31-36. doi:10.1007/BF02291575

Examples

```
fac <- nomo_factors(nomo_demo_continuous, n_iter = 20, seed = 2026)
fac
summary(fac)

# Ordered five-category items are analyzed with polychoric correlations
fac_ord <- nomo_factors(nomo_demo_ordinal, n_iter = 20, seed = 2026)
fac_ord$correlation
```

nomo_hierarchical	<i>Evaluate general-factor strength in bifactor and higher-order models</i>
-------------------	---

Description

`nomo_hierarchical()` answers the question behind many published scales that report both a total score and subscale scores: how much of each score reflects a general factor, and how much reliable variance does each subscale carry beyond it? It evaluates a fitted bifactor or higher-order CFA and reports omega total, omega hierarchical, explained common variance (ECV), the percentage of uncontaminated correlations (PUC), and subscale omega and omega hierarchical subscale, each with its estimand stated.

Usage

```
nomo_hierarchical(
  fit,
  general = NULL,
  obs.var = TRUE,
  guidance = nomo_defaults()
)
```

Arguments

<code>fit</code>	A <code>nomo_cfa</code> object or fitted lavaan model containing a bifactor or higher-order measurement model. Single-group, single-level models only.
<code>general</code>	Optional name of the general (bifactor) or second-order (higher-order) factor. If NULL, it is identified from the model and the call is refused if that is ambiguous.
<code>obs.var</code>	Logical. TRUE (default, matching <code>nomo_reliability()</code>) uses observed covariances in each omega's denominator; FALSE uses model-implied covariances, which reproduces the formulas in Rodriguez, Reise, and Haviland (2016).
<code>guidance</code>	Guidance settings returned by <code>nomo_defaults()</code> .

Details

Structure. The structure is read from the fitted model, so syntax written by hand works as well as syntax from `nomo_model()`:

- *Higher-order*: one second-order factor measured by first-order factors, each measured by its own items. The first-order factors' disturbances are the group sources. Their loadings are the Schmid-Leiman decomposition of the higher-order solution, which is exact for a confirmatory model.
- *Bifactor*: one general factor measured by every item, plus group factors measured by disjoint subsets of items. Items may load on the general factor alone.

The general and group sources must be orthogonal, because variance can be attributed to a source only when sources do not covary. A model that allows them to correlate is refused with an explanation.

Estimands. Every omega is the proportion of the variance of a unit-weighted composite that a set of sources explains. With continuous indicators the composite is the observed sum score. With ordered indicators the indices describe the *latent-response* composite, an upper bound on the reliability of the observed ordinal sum score; the observed ordinal-scale version is not provided. ECV and PUC describe the item set and do not depend on the denominator.

No verdicts. The indices are reported with plain-language interpretation, but no value is treated as a pass/fail threshold. Reise (2012) notes that no benchmark value of ECV establishes when a general factor is strong enough, and that when PUC is very high, even modest ECV can yield relatively unbiased estimates from a unidimensional model.

Choosing a structure. A bifactor model will usually fit at least as well as a correlated-factors or higher-order model of the same items, including when it is not the model that generated the data (Reise, 2012), and a higher-order model is a constrained version of the bifactor model (Yung, Thissen, & McLeod, 1999). Bonifay, Lane, and Reise (2017) treat that tendency to show superior goodness of fit as a particular concern, and note that the superior performance "may be a symptom of overfitting", capturing unwanted noise as well as real trends. Murray and Johnson (2013) compared these two structures directly and found the comparison itself biased: unless there was essentially no unmodeled complexity, their simulation favored the bifactor model even when a higher-order model generated the data, and they concluded that the choice "should not rely on which is better fitting". Compare the alternatives with `nomo_compare()` and choose on substantive grounds, not on fit alone. `nomo_hierarchical()` does not choose.

Factor scores: two indices that answer different questions. The factors table reports, for the general factor and each group factor, factor determinacy and construct replicability.

Factor determinacy (Beauducel, 2011; Rodriguez, Reise, & Haviland, 2016) is the correlation between a factor and its estimated factor score. It is computed from the whole model-reproduced correlation matrix, so a group factor's score estimate can use the other items to partial out the general factor. `determinacy_r2` is its square, the proportion of variance in factor scores explained by the factor, and `min_competing_r` is the minimum possible correlation between two equally valid sets of factor scores, twice the squared determinacy minus one. A negative value there means two researchers scoring the same data could rank people in opposite orders and both be consistent with the model.

Construct replicability H (Hancock & Mueller, 2001) is the proportion of variance in a factor explainable by its own indicators when optimally weighted. It uses only that factor's loadings and treats the remainder of each item as uncorrelated residual.

The two are the same quantity when the data are unidimensional, and can differ under a bifactor model, where an item's residual with respect to one factor contains the other factors and is correlated across items. Rodriguez et al. (2016) state this and decline to prefer either, so both are reported. Determinacy is always computed from the model-reproduced matrix, whatever `obs.var` is set to, because that is what the formula is defined on.

Gorsuch (1983) recommended using factor score estimates only when determinacy exceeds .90, with competing score sets correlating above .70, and Hancock and Mueller (2001) proposed .70 as a standard for H. These are reported as their authors' recommendations where a value falls below them, as with fixed fit-index cutoffs elsewhere in the package, and are never applied as rules.

Value

A `nomo_hierarchical` object with the detected structure, the general factor, the groups and their items, an indices table, a subscales table, an item-level loadings table, identification and estimand notes, and a `decision_log`.

References

- Beauducel, A. (2011). Indeterminacy of factor score estimates in slightly misspecified confirmatory factor models. *Journal of Modern Applied Statistical Methods*, 10(2), 583-598. doi:10.22237/jmasm/1320120900
- Bonifay, W., Lane, S. P., & Reise, S. P. (2017). Three concerns with applying a bifactor model as a structure of psychopathology. *Clinical Psychological Science*, 5(1), 184-186. doi:10.1177/2167702616657069
- Gorsuch, R. L. (1983). *Factor analysis* (2nd ed.). Lawrence Erlbaum.
- Hancock, G. R., & Mueller, R. O. (2001). Rethinking construct reliability within latent variable systems. In R. Cudeck, S. du Toit, & D. Sorbom (Eds.), *Structural equation modeling: Present and future* (pp. 195-216). Scientific Software International.
- Holzinger, K. J., & Swineford, F. (1937). The bi-factor method. *Psychometrika*, 2(1), 41-54. doi:10.1007/BF02287965
- Murray, A. L., & Johnson, W. (2013). The limitations of model fit in comparing the bi-factor versus higher-order models of human cognitive ability structure. *Intelligence*, 41(5), 407-422. doi:10.1016/j.intell.2013.06.004
- Reise, S. P. (2012). The rediscovery of bifactor measurement models. *Multivariate Behavioral Research*, 47(5), 667-696. doi:10.1080/00273171.2012.715555
- Reise, S. P., Bonifay, W. E., & Haviland, M. G. (2013). Scoring and modeling psychological measures in the presence of multidimensionality. *Journal of Personality Assessment*, 95(2), 129-140. doi:10.1080/00223891.2012.725437
- Rodriguez, A., Reise, S. P., & Haviland, M. G. (2016). Evaluating bifactor models: Calculating and interpreting statistical indices. *Psychological Methods*, 21(2), 137-150. doi:10.1037/met0000045
- Schmid, J., & Leiman, J. M. (1957). The development of hierarchical factor solutions. *Psychometrika*, 22(1), 53-61. doi:10.1007/BF02289209
- Yung, Y.-F., Thissen, D., & McLeod, L. D. (1999). On the relationship between the higher-order factor model and the hierarchical factor model. *Psychometrika*, 64(2), 113-128. doi:10.1007/BF02294531

Examples

```
set.seed(2026)
n <- 500
g <- rnorm(n)
s <- matrix(rnorm(n * 3), n, 3)
dat <- as.data.frame(sapply(1:9, function(i) {
  .6 * g + .45 * s[, ceiling(i / 3)] + rnorm(n, sd = .65)
}))
names(dat) <- paste0("x", 1:9)

factors <- list(
  A = c("x1", "x2", "x3"),
  B = c("x4", "x5", "x6"),
  C = c("x7", "x8", "x9")
)

bf <- nomo_cfa(nomo_model(factors, structure = "bifactor"), data = dat)
h <- nomo_hierarchical(bf)
h
nomo_table(h, "subscales")
```

nomo_hypotheses	<i>Specify a priori expectations for a nomological network</i>
-----------------	--

Description

`nomo_hypotheses()` converts named theoretical predictions into a machine-readable object that can later be evaluated by `nomo_network()`. Relations use $A \rightarrow B$ for directed structural paths and $A \leftrightarrow B$ for associations/covariances.

Usage

```
nomo_hypotheses(...)
```

Arguments

... Named `nomo_expectations` created by `positive()`, `negative()`, or `negligible()`. Names must specify relations using $\rightarrow$ or $\leftrightarrow$.

Details

The function records theory; it does not inspect data, fit a model, or infer predictions from statistical significance.

Value

A `nomo_hypotheses` object whose `hypotheses` field is one row per hypothesis: its relation, prediction, theoretical region, scale, and origin. `n` is the number of hypotheses.

References

- Cronbach, L. J., & Meehl, P. E. (1955). Construct validity in psychological tests. *Psychological Bulletin*, 52(4), 281-302. doi:10.1037/h0040957
- Lakens, D., Scheel, A. M., & Isager, P. M. (2018). Equivalence testing for psychological research: A tutorial. *Advances in Methods and Practices in Psychological Science*, 1(2), 259-269. doi:10.1177/2515245918770963
- Nosek, B. A., Ebersole, C. R., DeHaven, A. C., & Mellor, D. T. (2018). The preregistration revolution. *Proceedings of the National Academy of Sciences*, 115(11), 2600-2606. doi:10.1073/pnas.1708274114

See Also

`nomo_network()` to evaluate the hypotheses.

Examples

```
h <- nomo_hypotheses(
  "GSE -> Spirituality" = positive(),
  "GSE -> Religiosity" = negligible(within = c(-.10, .10)),
  "Religiosity <-> Spirituality" = positive(min = .20)
)
h
```

nomo_invariance

Evaluate measurement invariance across groups

Description

`nomo_invariance()` evaluates increasingly constrained multi-group CFA models while retaining every generated `semTools::measEq.syntax()` object and every fitted lavaan model.

Usage

```
nomo_invariance(
  model,
  data,
  group,
  ordered = NULL,
  levels = NULL,
  partial = NULL,
  localize = TRUE,
  estimator = NULL,
  missing = NULL,
  ID.fac = "std.lv",
  ID.cat = "Wu.Estabrook.2016",
  parameterization = "theta",
  guidance = nomo_defaults()
)
```

Arguments

model	A researcher-specified lavaan measurement-model syntax string or an object created by <code>nomo_model()</code> .
data	A non-empty data frame.
group	Character scalar naming the grouping variable in data.
ordered	Optional character vector naming ordered indicators.
levels	Optional invariance levels. NULL uses the sequence implied by the indicator category structure.
partial	Optional researcher-specified partial-invariance releases from <code>nomo_partial()</code> .
localize	Logical. If TRUE, retain equality-constraint score tests as diagnostic evidence. Diagnostics never trigger automatic release/refitting.
estimator	Optional lavaan estimator. Ordered models default to WLSMV.
missing	Optional lavaan missing-data option.
ID.fac	Factor-identification method passed to <code>semTools::measEq.syntax()</code> . "std.lv" is the default.
ID.cat	Ordered-indicator identification method passed to <code>semTools::measEq.syntax()</code> . Wu-Estabrook is the default.
parameterization	Lavaan categorical parameterization. "theta" is the default for ordered indicators.
guidance	Guidance settings from <code>nomo_defaults()</code> .

Details

The sequence is adapted to the observed category structure. Continuous indicators use configural -> metric -> scalar -> strict. Ordered indicators with four or more categories permit a separate threshold step. Three-category indicators require threshold equality as part of the metric step. When any binary indicator is present, threshold, loading, and intercept restrictions are imposed together at the first equality step (strong) because those restrictions cannot be treated as independent tests under Wu-Estabrook identification.

Partial invariance is researcher controlled. Supply an object from `nomo_partial()` to request specific equality-constraint releases. Releases are carried forward to more restrictive levels and their rationales are retained. `nomo_invariance()` never searches for a combination of releases that makes a fit rule pass.

When `localize = TRUE`, univariate score tests for equality constraints are retained as diagnostic evidence. They are explicitly not used to modify the fitted model.

Value

A `nomo_invariance` object. The fields to read are:

- `groups`, `requested_levels`, and `completed_levels`.
- `fit_evidence`: one row per level, with its fit, its change from the level before, the likelihood-ratio test, and any warning or error.

- `local_strain`: score diagnostics for each equality constraint, which localize strain without releasing anything.
- `partial`: the researcher-specified releases, when given.
- `fits`: the fitted lavaan model at each level.
- `syntax_text`: the model syntax at each level.
- `engine_warnings`: warnings lavaan raised at each level.
- `indicator_type`, `identification_note`, and `decision_log`.

Other fields record the call, the settings used, and intermediate engine results. They may change between releases and are not part of the stable interface (see `?nomologR`).

References

Historical foundations:

Jöreskog, K. G. (1971). Simultaneous factor analysis in several populations. *Psychometrika*, 36(4), 409-426. doi:[10.1007/BF02291366](https://doi.org/10.1007/BF02291366)

Meredith, W. (1993). Measurement invariance, factor analysis and factorial invariance. *Psychometrika*, 58(4), 525-543. doi:[10.1007/BF02294825](https://doi.org/10.1007/BF02294825)

Vandenberg, R. J., & Lance, C. E. (2000). A review and synthesis of the measurement invariance literature: Suggestions, practices, and recommendations for organizational research. *Organizational Research Methods*, 3(1), 4-70. doi:[10.1177/109442810031002](https://doi.org/10.1177/109442810031002)

Change-in-fit evidence:

Chen, F. F. (2007). Sensitivity of goodness of fit indexes to lack of measurement invariance. *Structural Equation Modeling*, 14(3), 464-504. doi:[10.1080/10705510701301834](https://doi.org/10.1080/10705510701301834)

Cheung, G. W., & Rensvold, R. B. (2002). Evaluating goodness-of-fit indexes for testing measurement invariance. *Structural Equation Modeling*, 9(2), 233-255. doi:[10.1207/S15328007SEM0902_5](https://doi.org/10.1207/S15328007SEM0902_5)

Putnick, D. L., & Bornstein, M. H. (2016). Measurement invariance conventions and reporting: The state of the art and future directions for psychological research. *Developmental Review*, 41, 71-90. doi:[10.1016/j.dr.2016.06.004](https://doi.org/10.1016/j.dr.2016.06.004)

Ordered-categorical indicators:

Svetina, D., Rutkowski, L., & Rutkowski, D. (2020). Multiple-group invariance with categorical outcomes using updated guidelines: An illustration using Mplus and the lavaan/semTools packages. *Structural Equation Modeling*, 27(1), 111-130. doi:[10.1080/10705511.2019.1602776](https://doi.org/10.1080/10705511.2019.1602776)

Wu, H., & Estabrook, R. (2016). Identification of confirmatory factor analysis models of different levels of invariance for ordered categorical outcomes. *Psychometrika*, 81(4), 1014-1045. doi:[10.1007/s1133601695060](https://doi.org/10.1007/s1133601695060)

See Also

[nomo_partial\(\)](#) for researcher-specified releases.

Examples

```
inv <- nomo_invariance(
  "Agency =~ ag1 + ag2 + ag3 + ag4",
  data = nomo_demo_network,
  group = "group",
  levels = c("configural", "metric", "scalar")
)
inv
nomo_table(inv, "fit")
head(nomo_table(inv, "local_strain"))

# A release is a documented researcher decision, never an automatic search.
release <- nomo_partial(
  level = "scalar",
  syntax = "ag3 ~ 1",
  rationale = "Prior evidence anticipated a mode difference in ag3 wording."
)
inv_partial <- nomo_invariance(
  "Agency =~ ag1 + ag2 + ag3 + ag4",
  data = nomo_demo_network,
  group = "group",
  levels = c("configural", "metric", "scalar"),
  partial = release
)
nomo_table(inv_partial, "fit")
nomo_table(inv_partial, "partial")
```

nomo_methods	<i>The methods registry: what nomologR computes, and where it comes from</i>
--------------	--

Description

`nomo_methods()` returns a machine-readable registry of the statistical methods `nomologR` implements. Each entry records the workflow stage, where the method sits in the literature, what it estimates, its key assumptions, the function that implements it, the computational engine, and DOI-verified references.

Usage

```
nomo_methods(x = NULL, stage = NULL, lineage = NULL, references = FALSE)
```

Arguments

x	Optional <code>nomologR</code> result object. If supplied, only the methods actually used in producing that object are returned, in registry order. If <code>NULL</code> (default), the whole registry is returned.
---	---

stage	Optional character vector restricting the result to these workflow stages. One or more of "screen", "factors", "efa", "cfa", "compare", "reliability", "validity", "invariance", "network", "workflow".
lineage	Optional character vector restricting the result to "historical", "contemporary", or "emerging" methods.
references	If FALSE (default), each row is one method and the references column holds short in-text citations. If TRUE, the result is expanded to one row per method per reference, with full citation and doi columns, suitable for a reference list.

Details

The registry answers a question learners ask constantly and software rarely answers: *why this method, and where did it come from?* Every entry carries a lineage label:

- "historical": techniques a reader will meet in published work, retained so they can be recognized and interpreted. They are labeled and qualified.
- "contemporary": what the current methodological literature recommends.
- "emerging": recent proposals with a smaller evidence base.

A lineage label summarizes where a method sits in the literature. It is a teaching aid, not a claim that an older method is always wrong.

Two columns record how practice changed:

- introduced: the year of the publication that introduced the method. It is given only when the registry cites that publication, and is NA when the registry cites only a later review or critique. It is never dated from a secondary source.
- contemporary_practice: for a historical method, the registry's contemporary methods that now address the question it answered, such as parallel analysis for the eigenvalue-greater-than-one rule, or omega for coefficient alpha.

vignette("research-basis") draws a timeline from these columns.

The role column says how nomologR uses the method: "primary" evidence, "supporting" evidence, or "context". A "context" method is displayed for recognition and is deliberately excluded from any synthesis: the eigenvalue-greater-than-one rule and fixed fit-index cutoffs are shown so a reader can interpret older reports, never so nomologR can act on them.

The registry describes only what the package actually computes. Methods that are planned but not implemented are discussed in the research-basis article and tracked in issues; they are deliberately absent here, so that nomo_methods(run) can never credit a run with a method it did not use.

Passing a nomologR result object returns only the methods that object actually used, which is what `nomo_report()` cites.

Value

A tibble with columns id, stage, method, lineage, role, estimand, assumptions, implemented_by, engine, introduced, and contemporary_practice. With references = FALSE there is one row per method, and references holds short in-text citations. With references = TRUE there is one row per method-reference pair, with citation_key, citation, and doi.

Examples

```
# The whole registry
nomo_methods()

# What is shown only as historical context, and why
nomo_methods(lineage = "historical")[, c("method", "role", "assumptions")]

# How practice changed: historical methods and what now does their work
old <- nomo_methods(lineage = "historical")
old[!is.na(old$contemporary_practice),
  c("introduced", "method", "contemporary_practice")]

# A reference list for one stage
nomo_methods(stage = "reliability", references = TRUE)[, c("method", "citation")]
```

nomo_missing	<i>Missing-data sensitivity: does a result depend on how missing data were handled?</i>
--------------	---

Description

`nomo_missing()` refits the same prespecified model under alternative missing-data strategies. It reports whether the cases used, the estimates, fit, reliability, or theory evidence change. It does not choose a strategy and does not change the model.

Usage

```
nomo_missing(fit, data, strategies = NULL, ...)

## S3 method for class 'nomo_cfa'
nomo_missing(fit, data, strategies = NULL, reliability = TRUE, ...)

## S3 method for class 'nomo_network'
nomo_missing(fit, data, strategies = NULL, ...)
```

Arguments

<code>fit</code>	A <code>nomo_cfa</code> or <code>nomo_network</code> object.
<code>data</code>	The data the model was fitted to, including any cases listwise deletion removed. For a network fitted to a <code>nomo_split</code> , supply the same <code>nomo_split</code> ; the comparison uses its calibration sample. <code>data</code> is checked by refitting the original strategy, which must reproduce the fitted model.
<code>strategies</code>	Optional character vector of lavaan missing options to compare. The default compares listwise deletion with FIML for continuous indicators, and with pairwise deletion for ordered indicators. lavaan's aliases "fiml" and "direct" are treated as "ml".
<code>...</code>	Unused.

reliability For a `nomo_cfa`, whether to compare reliability across strategies with `nomo_reliability()`. Default TRUE. Reliability for ordered indicators is slow to compute, so FALSE is useful when only the estimates are of interest.

Details

Which strategies. For continuous indicators, listwise deletion is compared with full-information maximum likelihood (FIML), and FIML is the reference. `lavaan` does not offer FIML for ordered indicators estimated with WLSMV. For those, listwise deletion is compared with pairwise deletion, and pairwise deletion is the reference. The strategy the model was fitted with is always included.

Each strategy records what was requested and what `lavaan` actually used, because the two can differ, and differently across `lavaan` versions. When `missing = "ml"` is requested with ULS, `lavaan` 0.7 runs its two-stage method instead, as it does with GLS, while `lavaan` 0.6-21 refuses; with MLM it is refused. A refused strategy is reported as unavailable, with `lavaan`'s reason. If the reference itself cannot be fitted, another strategy that was fitted becomes the reference, and the decision log says so.

What each strategy assumes. Listwise and pairwise deletion require data missing completely at random (MCAR). FIML requires data missing at random (MAR). Enders and Bandalos (2001) found all three unbiased under MCAR, with FIML the most efficient. Under MAR, FIML remained unbiased, while listwise and pairwise deletion were biased. FIML also assumes multivariate normality, as complete-data maximum likelihood does.

What a comparison cannot show. Whether data are MAR cannot in general be tested from the data at hand (Schafer & Graham, 2002). Agreement between strategies shows that a result does not depend on the choice between them. It does not show that either strategy is unbiased; neither is guaranteed to be when data are missing not at random.

When a difference is flagged. Each estimate is compared with the reference strategy's estimate and expressed in units of the reference standard error. Schafer and Graham (2002) treat a bias larger than about half a standard error as practically important. Beyond that size it noticeably degrades the coverage of confidence intervals. A difference of that size is flagged for review, with two qualifications:

- The difference estimates listwise deletion's bias only if the data are MAR and the model is correct, since only then is FIML consistent. With ordered indicators, both strategies require MCAR, so a difference cannot be attributed to either one.
- The strategies analyze different cases, so part of any difference is sampling variability.

A hypothesis whose concordance with its prediction differs between strategies is also flagged for review.

Experimental. This function is experimental until `nomologR` 1.0.0. Its rule for flagging a difference may be refined to separate sampling variability from bias. Any change will be described in NEWS; see the package help page, `?nomologR`, for the stability policy.

Not implemented. Mean substitution is not offered. It understates variances and distorts covariances. Schafer and Graham (2002) show that even under MCAR it narrows confidence intervals below their nominal coverage. Multiple imputation and models for data missing not at random are outside this function.

Value

A `nomo_missing` object containing:

- `pattern`: missingness in the modeled variables.
- `variables`: missing values per variable.
- `strategies`: one row per strategy.
- `fit`: fit indices by strategy.
- `estimates`: standardized loadings and factor correlations for a `nomo_cfa`, or hypothesis estimates for a `nomo_network`, compared with the reference.
- `reliability`: for a `nomo_cfa`, reliability by strategy, unless `reliability = FALSE`.
- `decision_log`.
- `fits`: the refitted objects.

References

Enders, C. K., & Bandalos, D. L. (2001). The relative performance of full information maximum likelihood estimation for missing data in structural equation models. *Structural Equation Modeling*, 8(3), 430-457. doi:[10.1207/S15328007SEM0803_5](https://doi.org/10.1207/S15328007SEM0803_5)

Schafer, J. L., & Graham, J. W. (2002). Missing data: Our view of the state of the art. *Psychological Methods*, 7(2), 147-177. doi:[10.1037/1082989X.7.2.147](https://doi.org/10.1037/1082989X.7.2.147)

See Also

[nomo_cfa\(\)](#), [nomo_network\(\)](#), and [nomo_screen\(\)](#) for describing missingness before a model is fitted.

Examples

```
model <- "A =~ a1 + a2 + a3 + a4 + a5\nB =~ b1 + b2 + b3 + b4 + b5"
fit <- nomo_cfa(model, data = nomo_demo_continuous)

sensitivity <- nomo_missing(fit, data = nomo_demo_continuous, reliability = FALSE)
sensitivity
nomo_table(sensitivity, "strategies")
```

nomo_model

Build confirmatory factor-analysis syntax

Description

`nomo_model()` is a small convenience helper for reflective CFA models. It converts a named list of factor-to-indicator assignments into lavaan measurement-model syntax. It deliberately does not add residual covariances, cross-loadings, equality constraints, or other post-hoc changes.

Usage

```
nomo_model(
  factors,
  structure = c("correlated", "higher_order", "bifactor"),
  general = "G"
)
```

Arguments

factors	A named list. Each element name is a latent-factor name and each element value is a character vector of observed indicators. For hierarchical structures these are the first-order or group factors.
structure	One of "correlated", "higher_order", or "bifactor".
general	Name of the second-order or general factor. Used only when structure is "higher_order" or "bifactor".

Details

Three structures are available, and the researcher chooses among them. The helper never derives a structure from exploratory results.

- "correlated" (default): one factor per element of factors, with factors free to correlate.
- "higher_order": the elements of factors become first-order factors, and a second-order factor named by general explains their correlations. At least three first-order factors are required, because with two the second-order loadings are not identified without further constraints. With exactly three, the second-order part is just identified: the model fits exactly as well as the correlated-factors model, so fit cannot distinguish the two.
- "bifactor": a general factor named by general is measured by every indicator, and each element of factors becomes a group factor. All factors are orthogonal. Identification is written into the syntax (each factor's first loading is freed and its variance fixed to 1), so the model is identified the same way whatever `std.lv` is used to fit it. At least two group factors with at least two indicators each are required; configurations known to be fragile are noted.

Any identification notes are attached as the "notes" attribute and printed with the syntax. Use [nomo_hierarchical\(\)](#) to evaluate a fitted higher-order or bifactor model.

Value

A character scalar of class `nomo_model` that can be passed directly to [nomo_cfa\(\)](#) or to `lavaan::cfa()`. Attributes record the factors, structure, general factor, and any identification notes.

References

- Holzing, K. J., & Swineford, F. (1937). The bi-factor method. *Psychometrika*, 2(1), 41-54. [doi:10.1007/BF02287965](#)
- Reise, S. P. (2012). The rediscovery of bifactor measurement models. *Multivariate Behavioral Research*, 47(5), 667-696. [doi:10.1080/00273171.2012.715555](#)

Yung, Y.-F., Thissen, D., & McLeod, L. D. (1999). On the relationship between the higher-order factor model and the hierarchical factor model. *Psychometrika*, 64(2), 113-128. doi:10.1007/BF02294531

Examples

```
factors <- list(
  engagement = c("e1", "e2", "e3"),
  belonging = c("b1", "b2", "b3"),
  efficacy = c("f1", "f2", "f3")
)

nomo_model(factors)
nomo_model(factors, structure = "higher_order", general = "Wellbeing")
nomo_model(factors, structure = "bifactor", general = "Wellbeing")
```

nomo_network	<i>Evaluate a theory-specified nomological network</i>
--------------	--

Description

`nomo_network()` combines a researcher-specified measurement/SEM model with a machine-readable object from `nomo_hypotheses()`. Hypothesized relations that are not already present in model can be added transparently before fitting, so the theory object itself can define the structural portion of the network.

Usage

```
nomo_network(
  model,
  data,
  hypotheses,
  validation_data = NULL,
  add_missing = TRUE,
  ordered = NULL,
  estimator = NULL,
  missing = NULL,
  std.lv = TRUE,
  control = NULL,
  equivalence_alpha = 0.05,
  guidance = nomo_defaults()
)
```

Arguments

<code>model</code>	One non-empty lavaan SEM/measurement-model syntax string or an object created by <code>nomo_model()</code> .
--------------------	--

<code>data</code>	A non-empty data frame or a <code>nomo_split</code> object. For a <code>nomo_split</code> , the calibration subset is the primary sample and the validation subset is reserved for replication.
<code>hypotheses</code>	A <code>nomo_hypotheses</code> object.
<code>validation_data</code>	Optional independent validation data frame. Do not use this together with a <code>nomo_split</code> object.
<code>add_missing</code>	Logical. If TRUE (default), theory-specified relations absent from <code>model</code> are appended transparently before estimation.
<code>ordered</code>	Optional character vector naming ordered indicators.
<code>estimator</code>	Optional lavaan estimator. When ordered indicators are declared and <code>estimator = NULL</code> , WLSMV is requested.
<code>missing</code>	Optional lavaan missing-data option.
<code>std.lv</code>	Logical passed to <code>lavaan::sem()</code> .
<code>control</code>	Optional optimizer-control list passed to <code>lavaan::sem()</code> .
<code>equivalence_alpha</code>	One number strictly between 0 and .5. For quantitative negligible predictions, the equivalence confidence level is $1 - 2 * \text{equivalence_alpha}$.
<code>guidance</code>	Guidance settings returned by <code>nomo_defaults()</code> .

Details

Directed $A \rightarrow B$ hypotheses map to lavaan regression paths $B \sim A$. Association $A \leftrightarrow B$ hypotheses map to covariance paths $A \sim B$.

For quantitative negligible(`within = ...`) predictions, `nomo_network()` evaluates the SESOI using a normal-approximation equivalence confidence interval. With the default `equivalence_alpha = .05`, this is a 90 percent interval, corresponding to the usual two one-sided tests logic. A bare negligible() prediction remains non-confirmable from $p > .05$.

The function can also fit the same prespecified model in a validation sample. Pass `validation_data` explicitly, or pass a `nomo_split` object as `data` to use its calibration and validation subsets. No model relation is added or removed on the basis of validation results.

Value

A `nomo_network` object. The fields to read are:

- `hypothesis_evidence`: one row per hypothesis, with its prediction, estimate, interval, concordance with the prediction, and interpretation.
- `replication_evidence`: the same comparison in `validation_data`, when given.
- `fit_evidence`: global fit of the fitted model.
- `parameter_estimates` and `standardized_solution`: lavaan's parameter tables, as tibbles.
- `measurement_context`: the measurement model's loadings and fit, which qualify the structural evidence.
- `model_fitted` and `model_relations`: the syntax fitted and the relations added from the hypotheses.

- `fit`: the lavaan fit, and validation, the validation fit, when given.
- `converged`, `engine_warnings`, and `decision_log`.

Other fields record the call, the settings used, and intermediate engine results. They may change between releases and are not part of the stable interface (see `?nomologR`).

Replication status when the sign changes

When a directional prediction's primary and validation point estimates have opposite signs, `replication_status` is decided by the 95 percent confidence intervals, not by the point estimates alone:

- `"sign_reversal"`: both intervals exclude zero, on opposite sides. Each sample on its own supports a relation in a different direction.
- `"direction_not_replicated"`: exactly one interval excludes zero. The other sample does not support that direction, but it does not establish the opposite direction either.
- `"sign_change_within_uncertainty"`: neither interval excludes zero, or an interval is unavailable. Neither sample distinguishes the relation from zero, and the sign change is compatible with sampling variability around a small or null relation.

Point estimates scattered around a null relation differ in sign about half the time, so a sign change without interval evidence is not treated as a substantive discrepancy. Requiring both intervals to exclude zero is the interval counterpart of each sample separately rejecting a zero relation in its own direction. The rule does not turn a non-significant result into evidence of no relation: that claim needs a `negligible(within = ...)` prediction with a researcher-specified equivalence region (Lakens, Scheel, & Isager, 2018).

Relationships estimated between observed variables

A hypothesis whose endpoints are latent variables is estimated with their measurement error modeled. When an endpoint is an observed variable, that error enters unmodeled. If the observed variable is a composite of several items, such as a sum, mean, or factor score, the estimated relationship carries the discrepancy `nomo_scores()` reports as correlational accuracy, which can be substantial and runs in either direction depending on the scoring method and the model.

`nomo_network()` cannot tell from the model syntax whether an observed variable is a composite or a single measured quantity, so it does not guess. It classifies each hypothesis by whether its endpoints are latent, and the decision log discloses observed endpoints: for review when both ends are observed, and for information when one is. A single measured variable, such as a criterion recorded without items, is not a composite, and the disclosure says so.

Where the observed variables are composites, modeling their items as indicators of latent variables removes the discrepancy, and `lavaan::sam()` estimates the structural relationships after the measurement model (Rosseel & Loh, 2024).

Where they are factor scores and the hypothesis is a linear regression, Skrondal and Laake (2001) showed that one scoring design gives consistent estimates of the regression coefficients: regression-method scores for the predictors and Bartlett scores for the outcome, each block scored from a measurement model of its own. Scoring both with the same method, or scoring all the factors from one model, does not, and `vignette("scoring", package = "nomologR")` shows both failures. The standard errors still treat the scores as observed, and Skrondal and Laake note that corrected ones may require resampling. The result does not extend to nonlinear models. No correction is applied automatically.

References

- Anderson, J. C., & Gerbing, D. W. (1988). Structural equation modeling in practice: A review and recommended two-step approach. *Psychological Bulletin*, 103(3), 411-423. doi:10.1037/0033-2909.103.3.411
- Cronbach, L. J., & Meehl, P. E. (1955). Construct validity in psychological tests. *Psychological Bulletin*, 52(4), 281-302. doi:10.1037/h0040957
- Lakens, D., Scheel, A. M., & Isager, P. M. (2018). Equivalence testing for psychological research: A tutorial. *Advances in Methods and Practices in Psychological Science*, 1(2), 259-269. doi:10.1177/2515245918770963
- Messick, S. (1995). Validity of psychological assessment: Validation of inferences from persons' responses and performances as scientific inquiry into score meaning. *American Psychologist*, 50(9), 741-749. doi:10.1037/0003066X.50.9.741
- Rosseel, Y., & Loh, W. W. (2024). A structural after measurement approach to structural equation modeling. *Psychological Methods*, 29(3), 561-588. doi:10.1037/met0000503
- Schuurmann, D. J. (1987). A comparison of the two one-sided tests procedure and the power approach for assessing the equivalence of average bioavailability. *Journal of Pharmacokinetics and Biopharmaceutics*, 15(6), 657-680. doi:10.1007/BF01068419
- Skrondal, A., & Laake, P. (2001). Regression among factor scores. *Psychometrika*, 66(4), 563-575. doi:10.1007/BF02296196

Examples

```
model <- nomo_model(list(
  Agency = c("ag1", "ag2", "ag3", "ag4"),
  Persistence = c("pe1", "pe2", "pe3", "pe4"),
  SocialDesirability = c("sd1", "sd2", "sd3")
))

h <- nomo_hypotheses(
  "Agency -> Persistence" = positive(min = .20),
  "Agency <-> SocialDesirability" = negligible(within = c(-.15, .15)),
  "Agency -> Performance" = positive()
)

net <- nomo_network(model, data = nomo_demo_network, hypotheses = h)
net
nomo_table(net, "hypotheses")

# Evaluate the same prespecified network in calibration and validation rows
s <- nomo_split(nomo_demo_network, validation_prop = 0.40, seed = 2026)
net_rep <- nomo_network(model, data = s, hypotheses = h)
nomo_table(net_rep, "replication")
```

nomo_partial	<i>Specify researcher-controlled partial-invariance releases</i>
--------------	--

Description

`nomo_partial()` records equality constraints that the researcher has decided to release at a specific invariance level. It does not inspect modification indices, search for a better-fitting model, or choose releases automatically.

Usage

```
nomo_partial(level, syntax, rationale)
```

Arguments

<code>level</code>	Character vector naming the first invariance level at which each release should apply. Supported labels are <code>thresholds</code> , <code>metric</code> , <code>scalar</code> , <code>strong</code> , and <code>strict</code> ; the applicable subset depends on the indicator category structure.
<code>syntax</code>	Character vector of lavaan parameter expressions to exclude from the corresponding equality constraints.
<code>rationale</code>	Character vector documenting why each release was chosen. A single rationale may be recycled across multiple releases.

Details

Releases are written using ordinary lavaan parameter syntax accepted by the `group.partial` argument of `semTools::measEq.syntax()`, for example `"F =~ x2"`, `"x3 ~ 1"`, or `"u2 | t1"`.

A release first requested at one level is carried forward to later, more-restrictive levels so that a loading freed at the metric level does not silently become constrained again at the scalar level.

Value

A `nomo_partial` object whose `releases` field is one row per release: its identifier, level, syntax, and rationale. `n` is the number of releases.

References

- Byrne, B. M., Shavelson, R. J., & Muthén, B. (1989). Testing for the equivalence of factor covariance and mean structures: The issue of partial measurement invariance. *Psychological Bulletin*, 105(3), 456-466. doi:10.1037/00332909.105.3.456
- Putnick, D. L., & Bornstein, M. H. (2016). Measurement invariance conventions and reporting: The state of the art and future directions for psychological research. *Developmental Review*, 41, 71-90. doi:10.1016/j.dr.2016.06.004

Examples

```
partial <- nomo_partial(
  level = c("metric", "scalar"),
  syntax = c("F =~ x2", "x3 ~ 1"),
  rationale = c(
    "Loading difference was theoretically anticipated.",
    "Intercept difference was prespecified from prior evidence."
  )
)
partial
```

nomo_reliability	<i>Model-based reliability evidence</i>
------------------	---

Description

`nomo_reliability()` estimates score reliability from a fitted first-order CFA measurement model. The primary estimate is a model-based omega-type composite reliability from `semTools::compRelSEM()`. Coefficient alpha is available as a familiar secondary statistic and is explicitly qualified by its stronger assumptions.

Usage

```
nomo_reliability(
  fit,
  obs.var = TRUE,
  ordinal_scale = TRUE,
  include_alpha = TRUE,
  ci = c("none", "bootstrap"),
  ci_level = 0.95,
  ci_boot = 1000L,
  ci_seed = NULL,
  ci_ncpus = 1L,
  guidance = nomo_defaults()
)
```

Arguments

<code>fit</code>	A fitted <code>nomo_cfa()</code> object or a fitted lavaan CFA model.
<code>obs.var</code>	Logical passed to <code>semTools::compRelSEM()</code> . TRUE (default) uses observed covariances in the denominator; FALSE uses model-implied covariances.
<code>ordinal_scale</code>	Logical passed as <code>ord.scale</code> to <code>semTools::compRelSEM()</code> . For an all-ordered composite, TRUE (default) estimates reliability on the actual ordinal-score scale using the Green-Yang correction implemented by <code>semTools</code> ; FALSE estimates the latent-response-scale coefficient.

include_alpha	Logical. If TRUE (default), also return coefficient alpha as a secondary statistic where the requested estimand is defined. For ordered indicators with ordinal_scale = TRUE, observed-score alpha is deliberately reported as unavailable rather than silently switching to latent-response ("ordinal alpha") or numeric-score alpha. Alpha is not treated as the preferred reliability estimate for a general congeneric CFA.
ci	Character. "none" (default) returns point estimates only. "bootstrap" adds nonparametric percentile confidence intervals by repeatedly refitting the same CFA with <code>lavaan::bootstrapLavaan()</code> .
ci_level	Confidence level for bootstrap intervals. Default is 0.95.
ci_boot	Number of ordinary bootstrap resamples when ci = "bootstrap". Default is 1000.
ci_seed	Optional integer seed for reproducible bootstrap intervals.
ci_ncpus	Number of worker processes for the bootstrap. The default, 1, runs serially. Values above one use lavaan's snow backend, which works on Windows, macOS, and Linux. The draws depend on the worker count as well as the seed, so a result is reproducible for a given seed <i>and</i> worker count, and both are recorded. More workers are not always faster: each worker must start and load its packages, and in the evaluation recorded in the package's issue tracker, eight workers were slower than four. Workers must be able to load nomologR, lavaan, and semTools from the library.
guidance	Guidance settings from <code>nomo_defaults()</code> . The configured reliability reference is a review prompt, not a pass/fail criterion.

Details

The function is deliberately measurement-first: it refuses nonconverged models, structural SEMs, higher-order models, and cross-loaded first-order indicators in the v0.1 workflow. When global CFA strain or an improper solution is present, reliability is still inspectable but the decision log warns that model-based reliability can be distorted by misspecification.

Average variance extracted (AVE) is not a reliability coefficient and is intentionally handled by `nomo_validity()` instead.

Value

A `nomo_reliability` object. The fields to read are:

- omega: model-based omega per construct, with bootstrap intervals when ci = "bootstrap".
- alpha: coefficient alpha, reported as secondary.
- evidence: both coefficients with their references and interpretations.
- alpha_status and ci_status: what was computed, and why anything was not.
- model_strain and improper_solution: whether the measurement model showed strain that qualifies the coefficients.
- references and decision_log.

Other fields record the call, the settings used, and intermediate engine results. They may change between releases and are not part of the stable interface (see `?nomologR`).

References

Historical context:

Cronbach, L. J. (1951). Coefficient alpha and the internal structure of tests. *Psychometrika*, 16(3), 297-334. doi:10.1007/BF02310555

Contemporary model-based reliability:

Bell, S. M., Chalmers, R. P., & Flora, D. B. (2024). The impact of measurement model misspecification on coefficient omega estimates of composite reliability. *Educational and Psychological Measurement*, 84(1), 5-39. doi:10.1177/00131644231155804

Dunn, T. J., Baguley, T., & Brunsdon, V. (2014). From alpha to omega: A practical solution to the pervasive problem of internal consistency estimation. *British Journal of Psychology*, 105(3), 399-412. doi:10.1111/bjop.12046

Flora, D. B. (2020). Your coefficient alpha is probably wrong, but which coefficient omega is right? A tutorial on using R to obtain better reliability estimates. *Advances in Methods and Practices in Psychological Science*, 3(4), 484-501. doi:10.1177/2515245920951747

Green, S. B., & Yang, Y. (2009). Reliability of summed item scores using structural equation modeling: An alternative to coefficient alpha. *Psychometrika*, 74(1), 155-167. doi:10.1007/s11336-00890993

Kelley, K., & Pornprasertmanit, S. (2016). Confidence intervals for population reliability coefficients: Evaluation of methods, recommendations, and software for composite measures. *Psychological Methods*, 21(1), 69-92. doi:10.1037/a0040086

McNeish, D. (2018). Thanks coefficient alpha, we'll take it from here. *Psychological Methods*, 23(3), 412-433. doi:10.1037/met0000144

Sijtsma, K. (2009). On the use, the misuse, and the very limited usefulness of Cronbach's alpha. *Psychometrika*, 74(1), 107-120. doi:10.1007/s1133600891010

Examples

```
model <- '
  visual =~ x1 + x2 + x3
  textual =~ x4 + x5 + x6
  speed  =~ x7 + x8 + x9
'

cfa <- nomo_cfa(model, data = lavaan::HolzingerSwineford1939)
rel <- nomo_reliability(cfa)
summary(rel)
rel$decision_log

# Bootstrap intervals refit the same CFA repeatedly; use more resamples
# (for example 1000) for final reporting.
rel_ci <- nomo_reliability(cfa, ci = "bootstrap", ci_boot = 100, ci_seed = 2026)
summary(rel_ci)
```

nomo_report	<i>Render a reproducible nomologR analysis report</i>
-------------	---

Description

`nomo_report()` renders a self-contained HTML document from a `nomo_run` object. The report archives researcher inputs, sample roles, item and factor evidence, EFA/CFA results, reliability, convergent/discriminant evidence, optional invariance and nomological-network results, researcher decisions, deviations/post-hoc decisions, method citations, an evidence trace, and session information.

Usage

```
nomo_report(
  x,
  file = "nomologR-report.html",
  title = "nomologR reproducible analysis report",
  include_plots = TRUE,
  include_session = TRUE,
  max_table_rows = 50L,
  overwrite = FALSE,
  quiet = TRUE,
  apa_tables = FALSE
)
```

Arguments

<code>x</code>	An object created by <code>nomo_run()</code> .
<code>file</code>	Output path. The extension chooses the format: <code>.html</code> or <code>.htm</code> for a self-contained HTML report, <code>.docx</code> for a Word document. The Word report carries the same tables, figures, and interpretation contract as the HTML one; collapsible sections are shown expanded, and it uses Word's default styles.
<code>title</code>	Report title.
<code>include_plots</code>	Logical; include a compact set of component plots when those plots are available.
<code>include_session</code>	Logical; include full <code>sessionInfo()</code> output.
<code>max_table_rows</code>	Positive integer used for ordinary display tables. The final evidence-trace appendix is not truncated.
<code>overwrite</code>	Logical; replace an existing file.
<code>quiet</code>	Logical passed to <code>rmarkdown::render()</code> .
<code>apa_tables</code>	Logical; if TRUE, append a <i>Manuscript tables</i> appendix with the <code>nomo_apa_table()</code> tables for the results the run holds. These are the CFA loadings and fit; reliability; the validity stage's average variance extracted and construct pairs (without a validity stage, the CFA factor correlations take the pairs' place, after the fit);

and, when present, invariance and the network's hypotheses and fit. They are numbered in that order. Default FALSE, which leaves the report unchanged.

Details

The report is a presentation and provenance layer. It does not refit models, alter data, free parameters, remove items, or manufacture additional statistical conclusions.

Reports can be rendered from complete, paused, or blocked `nomo_run` objects. Incomplete stages are labeled as such rather than silently omitted.

`nomo_report()` behaves the same from the console, a script, or a chunk inside another R Markdown or Quarto document, such as a thesis chapter. Rendering from inside a document nests two renders that share knitr's global state, so the report is rendered with knitr's default chunk options, which its own template then sets, and the calling document's options are restored afterwards. The calling document's chunk options therefore cannot change the report, and rendering the report does not change the calling document.

Value

The normalized report file path, invisibly.

Examples

```
# Rendering requires pandoc, which RStudio and Quarto installations include.
if (rmarkdown::pandoc_available()) {
  run <- nomo_run(
    data = nomo_demo_network,
    scales = list(Agency = c("ag1", "ag2", "ag3", "ag4")),
    settings = list(factors = list(n_iter = 20, seed = 2026)),
    decisions = list(
      factor_count = 1L,
      cfa_model = "Agency =~ ag1 + ag2 + ag3 + ag4",
      measurement_model = "proceed"
    )
  )

  report_file <- nomo_report(
    run,
    file = tempfile(fileext = ".html")
  )
  file.exists(report_file)
}
```

Description

`nomo_revise()` creates a child workflow from a parent `nomo_run()` with a revised measurement model, a revised item set, or both. The child records what changed, why, whether the change was prespecified or post hoc, and a `nomo_compare()` result for the parent and revised measurement models. The parent workflow is not modified.

Usage

```
nomo_revise(
  run,
  cfa_model = NULL,
  items = NULL,
  rationale,
  origin = c("post_hoc", "a_priori"),
  decisions = list(),
  compare = TRUE
)
```

Arguments

<code>run</code>	A parent <code>nomo_run</code> object with a fitted measurement model.
<code>cfa_model</code>	Optional revised measurement model: a lavaan syntax string or an object from <code>nomo_model()</code> .
<code>items</code>	Optional named list of revised item vectors, one element per revised scale. Scales that are not named keep the parent's items. The measurement model must agree with the revised items: a revision is refused if the model still includes an item it drops, or omits an item it adds. Supply <code>cfa_model</code> together with <code>items</code> when the change requires it; <code>nomo_revise()</code> never rewrites the model.
<code>rationale</code>	Required character scalar recording why the workflow is revised.
<code>origin</code>	"post_hoc" (default) when the revision was prompted by results, or "a_priori" when it was planned in advance.
<code>decisions</code>	Optional named list of decisions for the child workflow, for example <code>list(factor_count = c(WellBeing = 1))</code> . Supplied decisions override inherited ones.
<code>compare</code>	Logical. If TRUE (default), compare the parent and revised measurement models with <code>nomo_compare()</code> .

Details

Revising is always a researcher decision: a rationale is required and no item is removed or parameter freed automatically. The child workflow reruns the staged evidence from screening onward with the revised scales and model, then pauses at the measurement review so the revised evidence is inspected before any downstream branch runs.

The factor-count decision is inherited from the parent unless decisions supplies a new one, so the revision changes only what the researcher changed.

Because a revision prompted by results is evaluated on the data that prompted it, the decision log records whether the change was "post_hoc" and recommends confirming the revised model in

independent data (Simmons, Nelson, & Simonsohn, 2011; Wicherts et al., 2016; Flake & Fried, 2020).

Removing an item changes the observed variables, so parent and revised models are not nested and `nomo_compare()` reports descriptive evidence only. To test whether an item is needed, keep it and fix its loading to zero in the revised model; see `nomo_compare()`.

Value

A new `nomo_run` object for the revised workflow, carrying `$lineage` (one row per revision), `$revision_comparison` (the `nomo_compare()` result, when available), and `$parent_summary`.

References

- Flake, J. K., & Fried, E. I. (2020). Measurement schmeasurement: Questionable measurement practices and how to avoid them. *Advances in Methods and Practices in Psychological Science*, 3(4), 456-465. doi:10.1177/2515245920952393
- MacCallum, R. C., Roznowski, M., & Necowitz, L. B. (1992). Model modifications in covariance structure analysis: The problem of capitalization on chance. *Psychological Bulletin*, 111(3), 490-504. doi:10.1037/00332909.111.3.490
- Simmons, J. P., Nelson, L. D., & Simonsohn, U. (2011). False-positive psychology: Undisclosed flexibility in data collection and analysis allows presenting anything as significant. *Psychological Science*, 22(11), 1359-1366. doi:10.1177/0956797611417632
- Wicherts, J. M., Veldkamp, C. L. S., Augusteijn, H. E. M., Bakker, M., van Aert, R. C. M., & van Assen, M. A. L. M. (2016). Degrees of freedom in planning, running, analyzing, and reporting psychological studies: A checklist to avoid p-hacking. *Frontiers in Psychology*, 7, 1832. doi:10.3389/fpsyg.2016.01832

See Also

`nomo_run()`, `nomo_compare()`

Examples

```
scales <- list(Agency = c("ag1", "ag2", "ag3", "ag4"))

run <- nomo_run(
  data = nomo_demo_network,
  scales = scales,
  settings = list(factors = list(n_iter = 20, seed = 2026)),
  decisions = list(
    factor_count = c(Agency = 1),
    cfa_model = "Agency =~ ag1 + ag2 + ag3 + ag4"
  )
)

revised <- nomo_revise(
  run,
  cfa_model = "Agency =~ ag1 + ag2 + ag3 + ag4\nag1 ~~ ag2",
  rationale = paste(
```

```

    "Residual diagnostics and item wording suggest ag1 and ag2 share",
    "method variance beyond the common factor."
  ),
  origin = "post_hoc"
)

nomo_table(revised, "lineage")
nomo_table(revised$revision_comparison, "comparisons")

```

nomo_run

Run the guided nomologR workflow

Description

`nomo_run()` coordinates the package's staged evidence workflow while stopping at consequential researcher decisions. Evidence can be computed automatically when doing so does not change the researcher's data or model, but `nomo_run()` does not silently select factor counts, remove items, construct a CFA model, free parameters, or respecify a model.

Usage

```

nomo_run(
  data = NULL,
  scales = NULL,
  mode = c("teaching", "research"),
  guidance = nomo_defaults(),
  decisions = list(),
  settings = list(),
  resume = NULL
)

```

Arguments

data	A non-empty data frame or a <code>nomo_split()</code> object. A split uses calibration rows for exploratory stages and validation rows for CFA, reliability, validity, and invariance. A network branch receives the split object so the same prespecified network can be evaluated across calibration and validation samples.
scales	A non-empty named list. Each element is a character vector of candidate item-column names for one scale/construct. May also be a handoff from <code>contentvalidR</code> 's <code>content_handoff()</code> , whose carried items and construct mapping then define the scales, as described for <code>nomo_screen()</code> . A handoff from a review with no construct mapping is refused, because a guided run needs scales and <code>nomologR</code> does not invent them.
mode	Presentation mode: "teaching" or "research". Mode changes presentation, not statistical behavior.
guidance	Guidance settings from <code>nomo_defaults()</code> .

decisions	Named list of explicit researcher decisions. Decisions may be supplied one pause at a time or all at once for a fully prespecified one-call run. A structured decision can be written as <code>list(value = ..., rationale = "...")</code> .
settings	Named list of stage-specific argument lists. Examples include <code>list(factors = list(n_iter = 200, seed = 2026))</code>, <code>list(invariance = list(group = "group"))</code>, or <code>list(network = list(hypotheses = h))</code>. Pipeline-controlled arguments such as component data/model inputs cannot be overridden through settings. When resuming, settings for future stages may be supplied without recomputing completed stages. <code>list(screen = list(effort = TRUE))</code> adds careless-responding indices (see nomo_screen()). They describe a respondent across the whole instrument, and even-odd consistency cannot be computed within one scale. So they are computed once, over every item in the run with the run's scales, and never inside the per-scale item audits. <code>reverse</code>, <code>scale_range</code>, <code>pair_magnitude</code>, and <code>scales</code> may be given alongside <code>effort</code>. When the scales came from a <code>contentvalidR</code> handoff that declares keying, its keying is used unless <code>reverse</code> or <code>scale_range</code> is given here. Two further requests attach evidence to the measurement model: • <code>list(scores = list(method = "sum"))</code> scores it with nomo_scores(), using a method the researcher names; <code>nomologR</code> does not choose one. • <code>list(missing = list())</code> compares missing-data strategies with nomo_missing(). strategies and reliability may be given. The comparison covers the network too when one is requested. Both run after convergent and discriminant evidence, so they are in view when the researcher decides whether to carry the model forward. Neither is a stage of its own. Requested evidence that cannot be computed is recorded in the design log, and the workflow continues, since no later stage depends on it.
resume	Optional prior <code>nomo_run</code> object. When supplied, the existing source data, scales, guidance, completed component results, decisions, and provenance are reused.

Details

The guided workflow is resumable. Completed component objects are retained rather than recomputed. Future-stage settings may be added or revised until that stage has completed; settings for completed/blocked stages are locked.

Consequential decisions are currently:

1. `factor_count`: the EFA factor count for each named scale;
2. `cfa_model`: the prespecified confirmatory measurement model;
3. `measurement_model`: "proceed" or "revise" after reviewing CFA, reliability, and convergent/discriminant evidence.

Optional invariance and nomological-network branches are requested explicitly through `settings$invariance` and `settings$network`. A network branch must include a `nomo_hypotheses` object. Invariance diagnostics never free parameters automatically, and network results never collapse to a one-number validity score.

Value

A `nomo_run` object. The fields to read are:

- `status` and `next_stage`: where the run is.
- `results`: each component's result, by stage and then by scale; for example `results$screen$Agency` is a `nomo_screen` object.
- `stage_status`: one row per stage.
- `decision_requests`: the decisions the run is waiting for.
- `decision_log`: the workflow's decisions, with their rationales and sources.
- `scales`, `mode`, `sample_design`, `sample_n`, `decisions`, and `settings`.

`nomo_table()` returns the run's tables, including the component recipe, the component decision logs, and the revision lineage. Other fields hold the source data and state needed to resume or revise the run. They may change between releases and are not part of the stable interface (see `?nomologR`).

References

- Boateng, G. O., Neilands, T. B., Frongillo, E. A., Melgar-Quinonez, H. R., & Young, S. L. (2018). Best practices for developing and validating scales for health, social, and behavioral research: A primer. *Frontiers in Public Health*, 6, 149. doi:10.3389/fpubh.2018.00149
- Clark, L. A., & Watson, D. (1995). Constructing validity: Basic issues in objective scale development. *Psychological Assessment*, 7(3), 309-319. doi:10.1037/10403590.7.3.309
- Flake, J. K., & Fried, E. I. (2020). Measurement schmeasurement: Questionable measurement practices and how to avoid them. *Advances in Methods and Practices in Psychological Science*, 3(4), 456-465. doi:10.1177/2515245920952393
- Flake, J. K., Pek, J., & Hehman, E. (2017). Construct validation in social and personality research: Current practice and recommendations. *Social Psychological and Personality Science*, 8(4), 370-378. doi:10.1177/1948550617693063
- Hinkin, T. R. (1998). A brief tutorial on the development of measures for use in survey questionnaires. *Organizational Research Methods*, 1(1), 104-121. doi:10.1177/109442819800100106
- Simmons, J. P., Nelson, L. D., & Simonsohn, U. (2011). False-positive psychology: Undisclosed flexibility in data collection and analysis allows presenting anything as significant. *Psychological Science*, 22(11), 1359-1366. doi:10.1177/0956797611417632

Examples

```
scales <- list(
  Agency = c("ag1", "ag2", "ag3", "ag4"),
  Persistence = c("pe1", "pe2", "pe3", "pe4")
)

# Evidence is computed, then the run pauses for the factor-count decision.
run <- nomo_run(
  data = nomo_demo_network,
  scales = scales,
  settings = list(factors = list(n_iter = 20, seed = 2026))
)
```

```

run
nomo_table(run, "requests")

run <- nomo_run(
  resume = run,
  decisions = list(
    factor_count = list(
      value = c(Agency = 1, Persistence = 1),
      rationale = "Each scale was written to measure one construct."
    )
  )
)

run <- nomo_run(
  resume = run,
  decisions = list(
    cfa_model = list(
      value = "Agency =~ ag1 + ag2 + ag3 + ag4; Persistence =~ pe1 + pe2 + pe3 + pe4",
      rationale = "Prespecified two-construct measurement model."
    )
  )
)

run <- nomo_run(
  resume = run,
  decisions = list(
    measurement_model = list(
      value = "proceed",
      rationale = "Measurement evidence reviewed before downstream analyses."
    )
  )
)
nomo_table(run, "stages")

```

nomo_scores

Score a measurement model, with the evidence for the scoring choice

Description

Computes scores from a fitted measurement model and reports what those scores are and are not. Scoring is a modeling decision, and `nomo_scores()` documents the decision rather than making it.

Usage

```

nomo_scores(
  fit,
  method = c("sum", "mean", "regression", "bartlett"),
  guidance = nomo_defaults()
)

```

Arguments

fit	A <code>nomo_cfa</code> object or fitted lavaan measurement model. Single-group, single-level, with no regressions among latent variables.
method	Scoring method. "sum" and "mean" are unit weighted; "regression" and "bartlett" are model weighted.
guidance	A <code>nomo_guidance</code> object from <code>nomo_defaults()</code> .

Details

Unit weighting is a model. McNeish and Wolf (2020) show that adding items is not a model-free arithmetic calculation but a *parallel* factor model, assuming equal unstandardized loadings and equal residual variances. For `method = "sum"` and `method = "mean"`, that constrained model is fitted and compared with the model supplied, so a researcher can see whether the assumption their sum score makes is consistent with their data.

A score is not the latent variable. Grice (2001) evaluates factor scores on three criteria, all reported here and all computed from the fitted model:

- `validity`, the correlation between a score and the factor it estimates. For `method = "regression"` this equals the factor determinacy coefficient reported by `nomo_hierarchical()`, because that method maximizes it.
- `univocality`, its correlation with the factors it does not represent.
- `correlational_accuracy`, how far correlations among scores sit from the correlations among the factors they stand in for.

The third deserves attention before scores are used in later analyses. A relationship estimated from scores carries that discrepancy as bias, and its direction depends on the scoring method and the model rather than being a constant that can be corrected for. Where a question can be asked of the latent variables instead, asking it of scores replaces an unbiased answer with a biased one.

One design recovers a regression. For a linear regression among factors, Skrondal and Laake (2001) proved that regression-method scores for the predictors and Bartlett scores for the outcome, each block scored from a measurement model of its own, give consistent estimates of the regression coefficients. Both conditions matter: scoring every factor from one model, or using the same method for both blocks, does not. Standard errors that treat the scores as observed are not corrected by this, and the result does not extend to nonlinear models. `vignette("scoring", package = "nomologR")` works through the design.

Thresholds are context. Gorsuch's (1983) recommendation that validity reach .80, and above .90 for scores serving as substitutes for the factors themselves, is reported where a value falls below it and is never applied as a rule.

The supplied data is never modified, and scores are computed only for the cases the model used.

Value

An object of class `nomo_scores`. The fields to read are:

- `scores`: one column per factor, one row per case used.
- `method` and `weighting`: how the scores were computed.

- diagnostics: Grice's validity, univocality, and correlational accuracy per factor.
- unit_weighting: the loading evidence on whether unit weights suit the model, and parallel_test, the test of the parallel model that unit weighting assumes.
- score_correlations and factor_correlations: for comparing the two.
- notes: what the scores do and do not estimate.

Other fields record the call, the settings used, and intermediate engine results. They may change between releases and are not part of the stable interface (see ?nomologR).

References

- Gorsuch, R. L. (1983). *Factor analysis* (2nd ed.). Lawrence Erlbaum.
- Grice, J. W. (2001). Computing and evaluating factor scores. *Psychological Methods*, 6(4), 430-450. doi:10.1037/1082989X.6.4.430
- McNeish, D., & Wolf, M. G. (2020). Thinking twice about sum scores. *Behavior Research Methods*, 52(6), 2287-2305. doi:10.3758/s13428020013980
- Skrondal, A., & Laake, P. (2001). Regression among factor scores. *Psychometrika*, 66(4), 563-575. doi:10.1007/BF02296196

See Also

[nomo_hierarchical\(\)](#) for factor determinacy and construct replicability.

Examples

```
model <- '
  visual  =~ x1 + x2 + x3
  textual  =~ x4 + x5 + x6
  speed    =~ x7 + x8 + x9
'

cfa <- nomo_cfa(model, data = lavaan::HolzingerSwineford1939)

# A sum score, with the parallel model it assumes fitted and compared
summed <- nomo_scores(cfa, method = "sum")
summed
summed$unit_weighting

# Regression-method factor scores, with Grice's criteria
refined <- nomo_scores(cfa, method = "regression")
refined$diagnostics
head(refined$scores)
```

nomo_screen

Audit item-level data before factor modeling

Description

nomo_screen() performs a conservative audit of candidate item data before factor-retention, EFA, or CFA decisions are made. It summarizes item storage and observed response patterns, missingness, response concentration, and case-level completeness. It also creates a decision log that distinguishes observations from recommendations.

Usage

```
nomo_screen(
  data,
  items = NULL,
  guidance = nomo_defaults(),
  effort = FALSE,
  scales = NULL,
  reverse = NULL,
  scale_range = NULL,
  pair_magnitude = 0.6
)
```

Arguments

data	A data frame containing candidate items.
items	Optional character vector identifying item columns. If NULL, all columns are audited and the decision log reminds the user to verify that identifiers, demographics, and other non-item columns were not included. May also be a handoff from contentvalidR's content_handoff(); see Items from content review .
guidance	Guidance settings from nomo_defaults() .
effort	Logical. If TRUE, case-level indices of careless or insufficient-effort responding are added. See Careless responding .
scales	Optional named list of character vectors assigning items to scales. Needed for even-odd consistency and for the within-scale versions of long-string and inter-item standard deviation. With two or more scales, each item's corrected item-rest correlation is also computed against the rest of its own scale, and the item review uses that value. Negative inter-item correlations are then reviewed only within a scale, since items of different constructs need not correlate positively. A contentvalidR handoff supplies its scales here.
reverse	Optional character vector naming reverse-keyed items. Used only to recode an internal copy for the indices that need it; the data is never recoded.
scale_range	Numeric c(min, max) of the response scale. Required whenever reverse is supplied, and never inferred from the data.

`pair_magnitude` Minimum absolute between-person correlation for an item pair to count as a psychometric antonym or synonym. Curran (2016) suggests .60 while saying there is no firm basis for it, so it is an argument rather than a constant.

Details

The function never removes rows or items, changes scores, reverse-keys items, or decides whether a scale is valid.

Item-type labels are descriptive, not modeling decisions. In particular, `numeric_discrete` means that the observed numeric values are integer-like with 10 or fewer distinct observed values. It does **not** automatically mean that the item should be treated as ordinal in later analyses.

A constant item has only one distinct observed value. An `all_missing` item has no observed values. These are hard data conditions rather than psychometric cutoff rules.

Response concentration and near-zero-variance flags use configurable teaching references from `nomo_defaults()`. They are screening heuristics, not psychometric laws or automatic item-retention rules. Ordered and numeric-discrete items also receive descriptive boundary concentration summaries. Continuous-like numeric indicators receive descriptive skewness and excess-kurtosis summaries without a pass/fail normality judgment.

Careless responding. With `effort = TRUE`, each case receives the indices Meade and Craig (2012), Huang et al. (2012), and Curran (2016) describe: long-string, inter-item standard deviation (Marjanovic et al., 2015), Mahalanobis distance, even-odd consistency, and psychometric antonym and synonym correlations. Curran recommends these be used in series because each has blind spots, and the output is built to show where they disagree rather than to combine them into one score.

They disagree for a reason. Inter-item standard deviation detects random responding and gives a respondent who answers every item identically the best possible score; long-string detects exactly that respondent. When the two disagree about a case, the decision log says so.

A case is flagged only where a source states a rule, and each flag carries the source's own qualification: long-string at half the number of items, which Curran offers as a conservative starting point and says is not the best cut score for every scale, and a positive antonym or negative synonym correlation. The other indices have no stated cut score and are reported without a flag. Huang et al. found the indices they recommended identified attentive respondents well and random responders poorly, so an unflagged case is not thereby shown to be attentive.

Each respondent's antonym, synonym, and even-odd value is a correlation whose N is the number of pairs or scales. With two, every value is exactly +1 or -1, so at least three are required; with fewer than five the log says the flags are coarse. Cases are flagged, never removed.

Items from content review. items may be the handoff that `contentvalidR`'s `content_handoff()` produces after content review. Only items it marks as carried are screened. Every item it held back is listed in the decision log with its status and recommendation quoted in `contentvalidR`'s own words, and is never analyzed or reinstated here. The log also records the producing version, workflow, and carry rule, and that item membership came from content review rather than from these data.

A carried item that is not a column of data is refused, never dropped. Where the call leaves scales, reverse, or `scale_range` unset, the handoff's scales and declared keying fill them. An item is never treated as forward keyed because keying was undeclared, and a response scale the handoff did not record is never inferred. A handoff with a schema version this release does not read

is refused, naming both package versions. The interface is specified in nomologR issue #46, and contentvalidR is not needed to read it.

Value

An object of class `nomo_screen`. The fields to read are:

- `items`: the items screened, and `n_cases`, the number of rows.
- `item_summary`: one row per item, with its storage, inferred type, missingness, most common response, descriptive statistics, and near-zero-variance indicators.
- `response_distribution`: counts and proportions of each response.
- `case_summary`: missingness per row.
- `relationship_summary`: each item's corrected item-rest correlation and summary of its inter-item correlations. With two or more declared scales, `scale`, `scale_item_rest_r`, and `scale_item_rest_n` give each item's scale and its item-rest correlation within that scale, and `scale_negative_interitem_n` counts its negative correlations with items of the same scale; otherwise they are NA.
- `inter_item_correlations`: one row per item pair.
- `decision_log`: the evidence and its explanations (see `nomo_table()`).
- `effort`, `effort_pairs`, and `effort_settings`: the careless-responding indices per row, the pairs they used, and their settings, when `effort = TRUE`.
- `handoff`: the content-review handoff read from `items`, when one was supplied.

Other fields record the call, the settings used, and intermediate engine results. They may change between releases and are not part of the stable interface (see `?nomologR`).

References

- Curran, P. G. (2016). Methods for the detection of carelessly invalid responses in survey data. *Journal of Experimental Social Psychology*, 66, 4-19. doi:10.1016/j.jesp.2015.07.006
- Huang, J. L., Curran, P. G., Keeney, J., Poposki, E. M., & DeShon, R. P. (2012). Detecting and deterring insufficient effort responding to surveys. *Journal of Business and Psychology*, 27(1), 99-114. doi:10.1007/s1086901192318
- Marjanovic, Z., Holden, R., Struthers, W., Cribbie, R., & Greenglass, E. (2015). The inter-item standard deviation (ISD): An index that discriminates between conscientious and random responders. *Personality and Individual Differences*, 84, 79-83. doi:10.1016/j.paid.2014.08.021
- Meade, A. W., & Craig, S. B. (2012). Identifying careless responses in survey data. *Psychological Methods*, 17(3), 437-455. doi:10.1037/a0028085
- Clark, L. A., & Watson, D. (2019). Constructing validity: New developments in creating objective measuring instruments. *Psychological Assessment*, 31(12), 1412-1427. doi:10.1037/pas0000626
- Kuhn, M., & Johnson, K. (2013). *Applied predictive modeling*. Springer. doi:10.1007/97814614-68493
- Nunnally, J. C., & Bernstein, I. H. (1994). *Psychometric theory* (3rd ed.). McGraw-Hill.

Examples

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

out <- nomo_screen(dat)
out$item_summary
out$decision_log

# Simulated scale-development data with known teaching features
scr <- nomo_screen(nomo_demo_continuous)
summary(scr)
```

nomo_split

Create a reproducible calibration/validation split

Description

`nomo_split()` creates an explicit random split for workflows in which EFA and CFA should be evaluated on different observations when the available sample permits it. The function does not claim that sample splitting is always preferable: dividing a modest sample reduces precision in both subsets, and an external validation sample is generally stronger evidence when one is available.

Usage

```
nomo_split(
  data,
  validation_prop = 0.5,
  seed = 1234L,
  guidance = nomo_defaults()
)
```

Arguments

<code>data</code>	A non-empty data frame.
<code>validation_prop</code>	Proportion of rows assigned to the validation sample. Must be strictly between 0 and 1.
<code>seed</code>	Integer seed used to make the split reproducible.
<code>guidance</code>	Guidance settings from <code>nomo_defaults()</code> .

Details

The caller's random-number-generator state is restored after the split so that using `nomo_split()` does not silently alter later stochastic analyses.

Value

A `nomo_split` object. The fields to read are:

- calibration and validation: the two samples.
- assignment: which sample each row went to.
- `n_total`, `n_calibration`, `n_validation`, and `validation_prop_realized`.
- seed and `decision_log`.

Other fields record the call, the settings used, and intermediate engine results. They may change between releases and are not part of the stable interface (see `?nomologR`).

References

Fokkema, M., & Greiff, S. (2017). How performing PCA and CFA on the same data equals trouble. *European Journal of Psychological Assessment*, 33(6), 399-402. doi:[10.1027/10155759/a000460](https://doi.org/10.1027/10155759/a000460)

MacCallum, R. C., Roznowski, M., & Necowitz, L. B. (1992). Model modifications in covariance structure analysis: The problem of capitalization on chance. *Psychological Bulletin*, 111(3), 490-504. doi:[10.1037/00332909.111.3.490](https://doi.org/10.1037/00332909.111.3.490)

Examples

```
split <- nomo_split(nomo_demo_continuous, validation_prop = 0.40, seed = 2026)
split
nrow(split$calibration)
nrow(split$validation)
```

nomo_table	<i>Extract report-ready evidence tables</i>
------------	---

Description

`nomo_table()` returns compact tibbles intended for manuscripts, audit appendices, teaching materials, and `nomo_report()`. It never rounds away the underlying object: full engine results remain stored in the original `nomologR` object.

Usage

```
nomo_table(x, ...)
```

Arguments

- `x` A supported `nomologR` result object.
- `...` Additional arguments passed to methods, usually `type`.

Details

Supported objects and type values. The first value listed is the default.

- `nomo_screen`: "items", "distribution", "cases", "relationships", "effort", "decision_log"; see `nomo_screen()`.
- `nomo_factors`: "evidence", "criteria", "adequacy", "concordance", "decision_log"; see `nomo_factors()`.
- `nomo_efa`: "items", "pattern", "factor_correlations", "residuals", "decision_log". The pattern and factor-correlation matrices are returned as tables with the item or factor in the first column; see `nomo_efa()`.
- `nomo_cfa`: "fit", "loadings", "factor_correlations", "heywood", "residuals", "modification_indices", "decision_log"; see `nomo_cfa()`.
- `nomo_reliability`: "coefficients", "alpha_status", "ci_status", "decision_log"; see `nomo_reliability()`.
- `nomo_validity`: "convergent", "discriminant", "htmt_status", "decision_log". "discriminant" has one row per construct pair, with the latent correlation and the HTMT-family values together; see `nomo_validity()`.
- `nomo_scores`: "scores", "diagnostics", "unit_weighting", "notes"; see `nomo_scores()`.
- `nomo_hypotheses`: the machine-readable hypothesis table (no type).
- `nomo_network`: "hypotheses" (default), "fit", "measurement", "relations", "replication", "decision_log".
- `nomo_invariance`: "fit" (default), "categories", "partial", "local_strain", "decision_log". The local-strain table keeps lavaan's internal constraint label and adds a human-readable `constraint_display` column (for example, Intercept: ag3 (online vs. paper)).
- `nomo_compare`: "comparisons" (default), "models", "loadings", "evidence", "decision_log"; see `nomo_compare()`.
- `nomo_missing`: "strategies" (default), "estimates", "fit", "reliability", "pattern", "variables", "decision_log"; see `nomo_missing()`.
- `nomo_hierarchical`: "indices" (default), "subscales", "loadings", "notes", "decision_log"; see `nomo_hierarchical()`.
- `nomo_run`: "stages" (default), "requests", "decisions", "component_log", "scales", "recipe", "settings", "lineage", "methods"; see `nomo_run()` and `nomo_revise()`. "lineage" returns one row per revision, or no rows for a workflow that was never revised. "methods" returns the registry entries for the methods the run actually used, and is equivalent to `nomo_methods(x)`; see `nomo_methods()`.

Value

A tibble.

Examples

```
h <- nomo_hypotheses(
  "Agency -> Persistence" = positive(min = .20),
  "Agency <-> SocialDesirability" = negligible(within = c(-.15, .15))
)
```

```
)
nomo_table(h)

inv <- nomo_invariance(
  "Agency =~ ag1 + ag2 + ag3 + ag4",
  data = nomo_demo_network,
  group = "group",
  levels = c("configural", "metric", "scalar")
)
nomo_table(inv, "fit")
head(nomo_table(inv, "local_strain"))
```

nomo_validity

Convergent and discriminant construct-validity evidence

Description

`nomo_validity()` summarizes several complementary forms of measurement evidence from a fitted first-order CFA. Convergent evidence includes standardized loadings and average variance extracted (AVE). Discriminant evidence includes latent-factor correlations and, where appropriate, HTMT2 as the primary heterotrait-monotrait summary, with the original HTMT available as a sensitivity/teaching comparison.

Usage

```
nomo_validity(
  fit,
  ave_obs_var = TRUE,
  htmt = c("both", "htmt2", "htmt", "none"),
  htmt_missing = "default",
  fornell_larcker = FALSE,
  guidance = nomo_defaults()
)
```

Arguments

<code>fit</code>	A fitted <code>nomo_cfa()</code> object or fitted lavaan CFA model.
<code>ave_obs_var</code>	Logical passed to <code>semTools::AVE()</code> . TRUE (default) uses observed variances in the denominator; FALSE uses model-implied variances. For ordinal indicators, <code>semTools</code> calculates AVE using the polychoric correlation structure.
<code>htmt</code>	Which heterotrait-monotrait summaries to request: "both" (default), "htmt2", "htmt", or "none". HTMT2 is prioritized because its geometric-mean formulation is designed for congeneric indicators; original HTMT assumes tau-equivalence.
<code>htmt_missing</code>	Missing-data option passed to <code>semTools::htmt()</code> . The default "default" delegates the unrestricted-correlation missing-data handling to <code>lavaan::lavCor()</code> rather than silently imposing listwise deletion. Other supported values are "listwise", "pairwise", "direct", "ml", and "fiml".

fornell_larcker	Logical. If TRUE, also produce a legacy Fornell-Larcker matrix and pairwise comparison where available. It is not used as the primary discriminant-validity criterion.
guidance	Guidance settings from <code>nomo_defaults()</code> . The AVE and HTMT references are review prompts rather than universal pass/fail cutoffs.

Details

The function deliberately avoids a binary declaration that a construct is "valid" or "invalid". Numerical references trigger inspection and explanation. HTMT-family results are considered together with theory, indicator content, CFA fit, and latent-factor overlap. The Fornell-Larcker comparison is available only by explicit request and is labeled legacy/ supporting evidence because simulation work has shown that it can miss discriminant-validity problems that HTMT detects.

Value

A `nomo_validity` object. The fields to read are:

- `ave`: average variance extracted per construct, as convergent evidence.
- `latent_correlations`: construct correlations with intervals.
- `htmt2` and `htmt`: heterotrait-monotrait ratios per pair.
- `discriminant`: each pair's separation evidence with its reference and interpretation.
- `htmt_status`: which HTMT variants were computed, and why any was not.
- `fornell_larcker_pairs`: the historical comparison, when requested.
- `standardized_loadings`, `references`, and `decision_log`.

Other fields record the call, the settings used, and intermediate engine results. They may change between releases and are not part of the stable interface (see `?nomologR`).

References

Historical context:

Campbell, D. T., & Fiske, D. W. (1959). Convergent and discriminant validation by the multitrait-multimethod matrix. *Psychological Bulletin*, 56(2), 81-105. doi:10.1037/h0046016

Fornell, C., & Larcker, D. F. (1981). Evaluating structural equation models with unobservable variables and measurement error. *Journal of Marketing Research*, 18(1), 39-50. doi:10.2307/3151312

Contemporary construct-separation evidence:

Henseler, J., Ringle, C. M., & Sarstedt, M. (2015). A new criterion for assessing discriminant validity in variance-based structural equation modeling. *Journal of the Academy of Marketing Science*, 43(1), 115-135. doi:10.1007/s1174701404038

Roemer, E., Schuberth, F., & Henseler, J. (2021). HTMT2—An improved criterion for assessing discriminant validity in structural equation modeling. *Industrial Management & Data Systems*, 121(12), 2637-2650. doi:10.1108/IMDS0220210082

Rönkkö, M., & Cho, E. (2022). An updated guideline for assessing discriminant validity. *Organizational Research Methods*, 25(1). doi:10.1177/1094428120968614

Voorhees, C. M., Brady, M. K., Calantone, R., & Ramirez, E. (2016). Discriminant validity testing in marketing: An analysis, causes for concern, and proposed remedies. *Journal of the Academy of Marketing Science*, 44(1), 119-134. doi:[10.1007/s1174701504554](https://doi.org/10.1007/s1174701504554)

Examples

```
model <- '
  visual  =~ x1 + x2 + x3
  textual =~ x4 + x5 + x6
  speed   =~ x7 + x8 + x9
'

cfa <- nomo_cfa(model, data = lavaan::HolzingerSwineford1939)
val <- nomo_validity(cfa)
summary(val)
val$decision_log
```

plot.nomo_cfa	<i>Plot confirmatory factor-analysis evidence</i>
---------------	---

Description

Plot confirmatory factor-analysis evidence

Usage

```
## S3 method for class 'nomo_cfa'
plot(x, type = c("loadings", "fit", "residuals", "modification_indices"), ...)
```

Arguments

x	A nomo_cfa object.
type	Plot type: "loadings", "fit", "residuals", or "modification_indices".
...	Unused.

Value

A ggplot2 object.

plot.nomo_compare	<i>Plot a measurement-model comparison</i>
-------------------	--

Description

Plot a measurement-model comparison

Usage

```
## S3 method for class 'nomo_compare'
plot(x, type = c("fit", "loadings"), ...)
```

Arguments

x	A nomo_compare object.
type	"fit" shows CFI, TLI, RMSEA, and SRMR for each model with the configured teaching references; "loadings" shows standardized loadings for each model side by side.
...	Unused.

Value

A ggplot2 object.

plot.nomo_efa	<i>Plot exploratory factor-analysis evidence</i>
---------------	--

Description

Plot exploratory factor-analysis evidence

Usage

```
## S3 method for class 'nomo_efa'
plot(x, type = c("pattern", "items", "residuals", "factor_correlations"), ...)
```

Arguments

x	A nomo_efa object.
type	Plot type: "pattern", "items", "residuals", or "factor_correlations".
...	Unused.

Value

A ggplot2 object.

plot.nomo_factors	<i>Plot factor-retention evidence</i>
-------------------	---------------------------------------

Description

Plot factor-retention evidence

Usage

```
## S3 method for class 'nomo_factors'
plot(
  x,
  y = NULL,
  type = c("retention", "parallel_rules", "scree", "map", "evidence", "concordance",
    "kmo"),
  show_values = TRUE,
  ...
)
```

Arguments

x	A nomo_factors object.
y	Unused; included for the base plot() generic.
type	Plot type. "retention" compares observed common-factor eigenvalues with the selected parallel-analysis reference; "parallel_rules" compares PA factor counts under mean, percentile, and Crawford rules; "scree" displays component and common-factor eigenvalues; "map" displays original TR2 and revised TR4 MAP curves; "evidence" compares available retention criteria; "concordance" groups related variants into criterion families before showing support for each factor count; and "kmo" displays item-level KMO/MSA values.
show_values	Logical; label values where useful.
...	Additional arguments, currently ignored.

Value

A ggplot object.

```
plot.nomo_hierarchical
```

Plot hierarchical measurement evidence

Description

Plot hierarchical measurement evidence

Usage

```
## S3 method for class 'nomo_hierarchical'
plot(x, type = c("variance", "loadings"), ...)
```

Arguments

x	A nomo_hierarchical object.
type	"variance" (default) shows, for the total score and each subscale composite, how its variance divides among the general factor, the group factor, and everything else. "loadings" shows each item's standardized general and group loadings.
...	Unused.

Value

A ggplot object.

```
plot.nomo_invariance
```

Plot measurement-invariance evidence

Description

Plot measurement-invariance evidence

Usage

```
## S3 method for class 'nomo_invariance'
plot(x, type = c("fit", "change", "local_strain"), ...)
```

Arguments

x	A nomo_invariance object.
type	Plot type: "fit", "change", or "local_strain".
...	Unused.

Value

A ggplot2 object.

plot.nomo_network	<i>Plot nomological-network evidence</i>
-------------------	--

Description

Plot nomological-network evidence

Usage

```
## S3 method for class 'nomo_network'
plot(x, type = c("effects", "concordance", "fit", "replication"), ...)
```

Arguments

x	A nomo_network object.
type	Plot type: "effects", "concordance", "fit", or "replication".
...	Unused.

Value

A ggplot2 object.

plot.nomo_reliability	<i>Plot model-based reliability evidence</i>
-----------------------	--

Description

Plot model-based reliability evidence

Usage

```
## S3 method for class 'nomo_reliability'
plot(x, ...)
```

Arguments

x	A nomo_reliability object.
...	Unused.

Value

A ggplot2 object. The conceptual 0-to-1 reliability range is shown by default and expands only if an estimate or interval lies outside it.

plot.nomo_screen	<i>Plot a nomo_screen audit</i>
------------------	---------------------------------

Description

Visual diagnostics complement the numerical screening output. The default evidence map integrates multiple diagnostic signals without converting them into a pass/fail scale. Additional plot types display item-rest relationships, inter-item correlations, response-category use, or missingness.

Usage

```
## S3 method for class 'nomo_screen'
plot(
  x,
  y = NULL,
  type = c("evidence", "item_rest", "interitem", "responses", "missingness"),
  items = NULL,
  show_values = TRUE,
  ...
)
```

Arguments

<code>x</code>	A <code>nomo_screen</code> object.
<code>y</code>	Ignored; included for compatibility with the base <code>plot()</code> generic.
<code>type</code>	Plot type: "evidence", "item_rest", "interitem", "responses", or "missingness". "responses" draws a bar for each response category of a categorical item, and a histogram of observed values for a continuous item (<code>item_type</code> "numeric_continuous"). When the selected items mix the two, the categorical items are drawn and the caption names the continuous ones, which can be plotted by passing them as <code>items</code> .
<code>items</code>	Optional character vector of candidate items to display.
<code>show_values</code>	Logical; add numerical labels where useful.
<code>...</code>	Additional arguments, currently ignored.

Value

A `ggplot2` plot object.

plot.nomo_validity	<i>Plot convergent or discriminant validity evidence</i>
--------------------	--

Description

Plot convergent or discriminant validity evidence

Usage

```
## S3 method for class 'nomo_validity'  
plot(x, type = c("ave", "discriminant"), ...)
```

Arguments

x	A nomo_validity object.
type	Plot type: "ave" or "discriminant".
...	Unused.

Value

A ggplot2 object. The conceptual 0-to-1 coefficient range is shown by default and expands only if an empirical value lies outside it.

print.nomo_factors	<i>Print factor-retention evidence</i>
--------------------	--

Description

Print factor-retention evidence

Usage

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

Arguments

x	A nomo_factors object.
...	Additional arguments, currently ignored.

Value

x, invisibly.

print.nomo_screen	<i>Print a nomo_screen object</i>
-------------------	-----------------------------------

Description

Print a nomo_screen object

Usage

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

Arguments

x	A nomo_screen object.
...	Additional arguments, currently ignored.

Value

x, invisibly.

print.summary_nomo_factors	<i>Print a factor-retention summary</i>
----------------------------	---

Description

Print a factor-retention summary

Usage

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

Arguments

x	A summary_nomo_factors object.
...	Additional arguments, currently ignored.

Value

x, invisibly.

```
print.summary_nomo_screen
```

Print a summary_nomo_screen object

Description

Print a summary_nomo_screen object

Usage

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

Arguments

x	A summary_nomo_screen object.
...	Additional arguments, currently ignored.

Value

x, invisibly.

```
summary.nomo_cfa
```

Summarize a guided confirmatory factor analysis

Description

Summarize a guided confirmatory factor analysis

Usage

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

Arguments

object	A nomo_cfa object.
...	Unused.

Value

An object of class summary_nomo_cfa.

summary.nomo_compare	<i>Summarize a measurement-model comparison</i>
----------------------	---

Description

Summarize a measurement-model comparison

Usage

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

Arguments

object	A nomo_compare object.
...	Unused.

Value

An object of class summary_nomo_compare containing model fit, comparisons with interpretations, side-by-side standardized loadings, and measurement evidence.

summary.nomo_efa	<i>Summarize a guided exploratory factor analysis</i>
------------------	---

Description

Summarize a guided exploratory factor analysis

Usage

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

Arguments

object	A nomo_efa object.
...	Unused.

Value

An object of class summary_nomo_efa.

summary.nomo_factors *Summarize factor-retention evidence*

Description

Summarize factor-retention evidence

Usage

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

Arguments

object	A nomo_factors object.
...	Additional arguments, currently ignored.

Value

An object of class summary_nomo_factors.

summary.nomo_reliability
 Summarize model-based reliability evidence

Description

Summarize model-based reliability evidence

Usage

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

Arguments

object	A nomo_reliability object.
...	Unused.

Value

An object of class summary_nomo_reliability.

summary.nomo_screen	<i>Summarize a nomo_screen audit</i>
---------------------	--------------------------------------

Description

summary.nomo_screen() integrates the descriptive, relationship, and decision-log evidence from [nomo_screen\(\)](#) into an item-level review table. The resulting attention field is deliberately phrased as none, review, or concern; it is not an item-retention decision.

Usage

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

Arguments

object	A nomo_screen object.
...	Additional arguments, currently ignored.

Value

An object of class summary_nomo_screen containing an overview, integrated item-review table, decision log, relationship-method note, and guidance settings.

summary.nomo_validity	<i>Summarize convergent and discriminant measurement evidence</i>
-----------------------	---

Description

Summarize convergent and discriminant measurement evidence

Usage

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

Arguments

object	A nomo_validity object.
...	Unused.

Value

An object of class summary_nomo_validity. \$convergent combines AVE with a compact loading summary, while \$discriminant aligns latent correlations, HTMT2, and original HTMT without repeating the full CFA loading table.

Index

* datasets

- [nomo_demo_continuous](#), [12](#)
 - [nomo_demo_network](#), [13](#)
 - [nomo_demo_ordinal](#), [14](#)
 - [nomo_demo_walkthrough](#), [15](#)
- [lavaan::bootstrapLavaan\(\)](#), [43](#)
- [lavaan::lavTestLRT\(\)](#), [8](#), [9](#)
- [negative\(nomo_expectations\)](#), [19](#)
- [negative\(\)](#), [27](#)
- [negligible\(nomo_expectations\)](#), [19](#)
- [negligible\(\)](#), [27](#)
- [nomo_apa_table](#), [3](#)
- [nomo_apa_table\(\)](#), [45](#)
- [nomo_cfa](#), [5](#)
- [nomo_cfa\(\)](#), [8](#), [10](#), [11](#), [35](#), [36](#), [42](#), [60](#), [61](#)
- [nomo_compare](#), [8](#)
- [nomo_compare\(\)](#), [25](#), [47](#), [48](#), [60](#)
- [nomo_defaults](#), [11](#)
- [nomo_defaults\(\)](#), [6](#), [8](#), [18](#), [22](#), [24](#), [43](#), [49](#), [53](#), [55](#), [56](#), [58](#), [62](#)
- [nomo_demo_continuous](#), [12](#), [14](#)
- [nomo_demo_network](#), [12](#), [13](#), [14](#)
- [nomo_demo_ordinal](#), [12](#), [14](#)
- [nomo_demo_walkthrough](#), [15](#)
- [nomo_demo_walkthrough_items](#)
([nomo_demo_walkthrough](#)), [15](#)
- [nomo_efa](#), [17](#)
- [nomo_efa\(\)](#), [11](#), [60](#)
- [nomo_expectations](#), [19](#), [27](#)
- [nomo_factors](#), [21](#)
- [nomo_factors\(\)](#), [18](#), [60](#)
- [nomo_hierarchical](#), [24](#)
- [nomo_hierarchical\(\)](#), [36](#), [53](#), [54](#), [60](#)
- [nomo_hypotheses](#), [27](#)
- [nomo_hypotheses\(\)](#), [19](#)
- [nomo_invariance](#), [28](#)
- [nomo_invariance\(\)](#), [14](#)
- [nomo_methods](#), [31](#)
- [nomo_methods\(\)](#), [60](#)
- [nomo_missing](#), [33](#)
- [nomo_missing\(\)](#), [50](#), [60](#)
- [nomo_model](#), [35](#)
- [nomo_model\(\)](#), [5](#), [25](#), [47](#)
- [nomo_network](#), [37](#)
- [nomo_network\(\)](#), [14](#), [27](#), [28](#), [35](#)
- [nomo_partial](#), [41](#)
- [nomo_partial\(\)](#), [30](#)
- [nomo_reliability](#), [42](#)
- [nomo_reliability\(\)](#), [8](#), [11](#), [24](#), [34](#), [60](#)
- [nomo_report](#), [45](#)
- [nomo_report\(\)](#), [32](#), [59](#)
- [nomo_revise](#), [46](#)
- [nomo_revise\(\)](#), [60](#)
- [nomo_run](#), [49](#)
- [nomo_run\(\)](#), [14](#), [16](#), [45](#), [48](#), [60](#)
- [nomo_scores](#), [52](#)
- [nomo_scores\(\)](#), [39](#), [50](#), [60](#)
- [nomo_screen](#), [55](#)
- [nomo_screen\(\)](#), [16](#), [35](#), [49](#), [50](#), [60](#), [74](#)
- [nomo_split](#), [58](#)
- [nomo_split\(\)](#), [49](#)
- [nomo_table](#), [59](#)
- [nomo_table\(\)](#), [51](#), [57](#)
- [nomo_validity](#), [61](#)
- [nomo_validity\(\)](#), [8](#), [11](#), [43](#), [60](#)
- [plot.nomo_cfa](#), [63](#)
- [plot.nomo_compare](#), [64](#)
- [plot.nomo_efa](#), [64](#)
- [plot.nomo_factors](#), [65](#)
- [plot.nomo_hierarchical](#), [66](#)
- [plot.nomo_invariance](#), [66](#)
- [plot.nomo_network](#), [67](#)
- [plot.nomo_reliability](#), [67](#)
- [plot.nomo_screen](#), [68](#)
- [plot.nomo_validity](#), [69](#)
- [positive\(nomo_expectations\)](#), [19](#)
- [positive\(\)](#), [27](#)

`print.nomo_factors`, [69](#)
`print.nomo_screen`, [70](#)
`print.summary_nomo_factors`, [70](#)
`print.summary_nomo_screen`, [71](#)
`psych::cor.smooth()`, [18](#), [22](#)
`psych::fa()`, [17](#), [22](#)

`rmarkdown::render()`, [45](#)

`semTools::AVE()`, [61](#)
`semTools::compRelSEM()`, [42](#)
`semTools::hmt()`, [61](#)
`semTools::net()`, [9](#)
`summary.nomo_cfa`, [71](#)
`summary.nomo_compare`, [72](#)
`summary.nomo_efa`, [72](#)
`summary.nomo_factors`, [73](#)
`summary.nomo_reliability`, [73](#)
`summary.nomo_screen`, [74](#)
`summary.nomo_validity`, [74](#)