

Package: magmaan (via r-universe)

October 27, 2026

Title Structural Equation Modeling with One Justified Default Policy

Version 0.2.2

License MIT + file LICENSE

Encoding UTF-8

Imports magmaanlab (>= 0.2.2), stats

Suggests testthat (>= 3.0.0), lavaan

Config/testthat/edition 3

Roxygen list(markdown = TRUE)

Description One call estimates a structural equation model and computes inference under a single documented policy chosen by magmaan, with lavaan model syntax and a few familiar options. Pure R: all computation happens in 'magmaanlab', the methods-development package over the magmaan C++ core, which also offers every alternative convention. Not CRAN-bound.

Config/roxygen2/version 8.0.0

Config/pak/sysreqs cmake

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

Date/Publication 2026-10-27 15:59:30 UTC

RemoteUrl <https://github.com/JonasMoss/magmaan>

RemoteRef v0.2.2

RemoteSha c0dc6e296b515a3aff24353c390fd53e964de027

RemoteSubdir r-magmaan

Contents

anova.magmaan	2
as_lab_fit	3
barrier	4
fit_measures	4
fitted.magmaan	5

infer	6
magmaan	6
magmaan_methods	10
magmaan_model	11
modindices	13
summary.magmaan	14
Index	16

anova.magmaan	<i>Compare two nested magmaan fits</i>
---------------	--

Description

Score and likelihood-ratio tests of the restricted fit against the other, each calibrated with PEBA4, as the global tests are. The score test is primary and comes first: it calibrates better, especially when N is small relative to the df, where the likelihood-ratio test tends to over-reject. Select rows by test rather than position. The restricted model may constrain, fix or drop paths from the other model (for example, a shared label, `b == 0`, or a loading fixed to zero), fitted to the same observations with the same estimator and covariance policy. For a single restriction, a Wald test is the z-statistic of a defined parameter such as `d := a - b` in `coef(summary(fit))`.

Usage

```
## S3 method for class 'magmaan'
anova(object, ..., lavaan_compat = NULL, references = NULL)

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

Arguments

object, ...	Two magmaan() fits, in either order.
lavaan_compat	NULL (default) reports the policy's score and LR tests. "ML", "MLM" and "MLR" report lavaan's default difference test for complete-data ML. "WLSMV" and "ULSMV" report the ordinal Satorra-2000 scaled-shifted difference; plain "DWLS"/"ULS" retain the statistic without a p-value, and "WLS" reports the standard difference. FIML supports "ML" (standard) and "MLR" (SB2001 using Yuan-Bentler Mplus scales); "MLM" is inapplicable.
references	Character vector of reference laws, or NULL for the policy defaults.
x	A nested-test result.
digits	Number of digits printed after the decimal point.

Details

If the larger model fits worse than the restricted model, it is refitted from the embedded restricted estimate with its own fitting options. A better endpoint is used only when its native convergence verdict passes. The input fits are unchanged; `attr(result, "refit")` records the old and new objective and printing reports the recovery. Internal refit warnings are captured in `attr(result, "reseed")$warnings` and printed as notes, including when a retry is unsuccessful. An unsuccessful retry retains the failure.

Value

A data frame of class `magmaan_anova`.

Simulation studies

See `summary.magmaan()` for the reference grammar and uniform base columns. Each available nested test has one row per reference; an unavailable test retains one row with its typed reason and missing reference/pvalue. Defaults are PEBA4 for ML/FIML and All for DWLS. `recommended` is TRUE only for the policy's recommended p-values: score with PEBA4 for ML/FIML, or `fit_function_difference` with All for DWLS. The likelihood-ratio test is reported because it is the standard statistic, not recommended; `references = "std"` gives its plain chi-square p-value. Alternatives use the same policy statistic and retained nested spectrum (`spectra` attribute), including after recovery. Select by test and reference rather than position. References cannot be combined with `lavaan_compat`. A PSD-boundary input retains the `psd_boundary` attribute and marks computed rows with that reason; inference is computed assuming an interior population. Improper unrestricted fits also receive inference when the required tangents have full rank; tangent rank loss is `boundary_nesting`.

as_lab_fit

The magmaanlab fit behind a magmaan fit

Description

Use it to run any alternative estimator option or inference method from magmaanlab without refitting.

Usage

```
as_lab_fit(fit)
```

Arguments

`fit` A `magmaan()` fit.

Value

The underlying `magmaan_fit` from magmaanlab.

barrier	<i>A barrier covariance policy</i>
---------	------------------------------------

Description

The covariance argument of `magmaan()` for a penalized estimate in the interior of the PSD domain; see its Covariance policies section.

Usage

```
barrier(lambda = 0.25)
```

Arguments

lambda	The penalty weight, one finite non-negative number: the LKJ(1 + lambda) density's exponent on the log-likelihood scale. Zero is the unrestricted fit. The default, 0.25, is provisional.
--------	--

Value

An object of class `magmaan_covariance`.

Examples

```
if (requireNamespace("lavaan", quietly = TRUE)) {
  d <- lavaan::HolzingerSwineford1939
  sweep <- lapply(c(0, 0.1, 1), function(lambda)
    coef(magmaan("visual =~ x1 + x2 + x3", d, covariance = barrier(lambda),
      inference = FALSE)))
}
```

fit_measures	<i>Policy fit measures</i>
--------------	----------------------------

Description

Computes fit indices on request, fitting an independence baseline with the same estimator and data treatment. ML/FIML target likelihood discrepancy; continuous ULS targets squared covariance discrepancy; ML2S-NT uses Stage-1 Gamma; ordinal and mixed DWLS target discrepancy in the DWLS metric. Misspecification-robust trace corrections subtract estimated sampling bias before computing RMSEA, CFI and TLI. SRMR/CRMR subtract residual influence traces. Multigroup RMSEA includes $\sqrt{G}$; residuals pool across groups.

Usage

```
fit_measures(fit, lavaan_compat = NULL)
```

Arguments

<code>fit</code>	A <code>magmaan()</code> fit.
<code>lavaan_compat</code>	Optional named lavaan inference bundle. Returns lavaan's standard, scaled and robust families, intervals and close-fit p-values. Unsupported estimator/bundle combinations error. FIML ML/MLR compatibility uses saturated H1 EM parameter tolerance 1e-10.

Details

DWLS indices describe misfit in the DWLS metric: conventional ML cutoffs do not transfer (Savalei, 2021; Xia and Yang, 2019). See Brosseau-Liard, Savalei and Li (2012) and Savalei (2018, 2021) for robust corrections. These component checks do not establish finite-sample bias or coverage. Policy intervals and close-fit p-values await evaluation. With `lavaan_compat`, intervals and close-fit p-values are provided for supported bundles, including FIML ML/MLR.

Results have a fixed row set per estimator, including unavailable indices with NA estimates and typed reasons. Likelihood criteria appear only for likelihood estimators; CRMR appears for ordinal and mixed models. Computation includes a second fit and is performed each time this function is called.

Value

A data frame with columns `index`, `estimate` and `reason`. Stable index codes are `rmsea`, `cfi`, `tli`, `srmr`, `crmr`, `logl`, `unrestricted_logl`, `aic` and `bic`. The `details` attribute retains the lab composer's diagnostic ingredients.

<code>fitted.magmaan</code>	<i>Model-implied moments of a magmaan fit</i>
-----------------------------	---

Description

The covariance matrix and, with a mean structure, the mean vector that the fitted model implies, as lavaan's `fitted()` gives them: one list for a single group, a list per group otherwise. A model whose parameters are all fixed is evaluated at those values, so this also gives the moments of a population written in model syntax.

Usage

```
## S3 method for class 'magmaan'
fitted(object, ...)
```

Arguments

<code>object</code>	A <code>magmaan()</code> fit.
<code>...</code>	Unused.

Value

`list(cov, mean)`, or a named list of them per group.

infer	<i>Compute inference for a magmaan fit</i>
-------	--

Description

Runs the selected inference bundle on an existing fit without refitting. A component that cannot be computed is recorded with a reason; the estimates remain usable.

Usage

```
infer(fit, lavaan_compat = NULL)
```

Arguments

fit	A magmaan() fit.
lavaan_compat	NULL computes the default policy. A lavaan bundle such as "MLM", "MLR" or "WLSMV" stores an additional compatibility bundle for reuse by the reporting methods; it leaves the policy intact.

Value

The fit with its inference results attached.

magmaan	<i>Fit a structural equation model with magmaan's default inference</i>
---------	---

Description

Estimates the model and, unless `inference = FALSE`, computes inference under magmaan's single documented policy. The arguments are the choices that define the estimate; structural choices belong to [magmaan_model\(\)](#) and optimization details to `options`.

Usage

```
magmaan(
  model,
  data,
  estimator = "ML",
  covariance = "unrestricted",
  inference = TRUE,
  options = NULL,
  ...
)
```

Arguments

model	A model from <code>magmaan_model()</code> , or lavaan model syntax in one string (or a <code>magmaanlab::model_spec()</code> specification) as a shortcut that constructs the model with data as its prototype on every call. A syntax string has the default structural choices: one group, continuous variables and marker identification. Data with ordered factors need <code>magmaan_model(ordered =)</code> . Unfit-table Mplus specifications raise one <code>magmaan_mplus_error</code> listing all input edits, with a character vector reason and named character vector edit (names equal to reason); printing the constructed model shows its fitability and edits.
data	A data frame of raw observations. Mplus ESTIMATOR settings are reported, not imported: choose the ordinary estimator explicitly. Mplus ML/MLR use ML on complete data or FIML with missing data; WLSMV corresponds to DWLS with ordered outcomes. Inference follows magmaan's ordinary policy.
estimator	"ML", "FIML", "ML2S", "GLS" or "ULS" for continuous variables; "DWLS", "WLS", "ULS" or association "ML" for ordered variables. Omission selects ML for continuous variables and DWLS for ordered variables. Association ML requires complete rows; other estimators except FIML and ML2S delete incomplete rows listwise.
covariance	The covariance policy: "unrestricted", "psd" (every model-implied covariance matrix positive semidefinite), or "barrier" or <code>barrier()</code> (a penalized estimate in the interior of the PSD domain; experimental).
inference	Compute inference now. With FALSE, call <code>infer()</code> later.
options	Optimization details, a named list with any of start, optimizer, convergence, marker and preset; omitted entries use magmaan's defaults. • start: "default"; "fabin3", the FABIN3 start (Hägglund, 1982), which ML, ML2S and GLS used before 2026-09-26 and every other fit, including every PSD fit, uses by default (not for ordered variables); "lavaan-0.7.2", lavaan 0.7.2's starts; a previous <code>magmaan()</code> fit; or a data frame with columns lhs, op, rhs and est (and group for several groups). A fit or table sets the start of every free parameter it matches, as lavaan's <code>start = fit</code>; unmatched parameters use the resolved start convention (including the preset). In <code>as_lab_fit(fit)\$fitting</code>, <code>requested\$starts = "table"</code> records both tables and previous fits; <code>effective\$starts</code> appends "+table" to that convention, describing the first attempt (for example "lavaan-0.7.2+table"). • optimizer: "default", "port" (PORT, as R's <code>nlminb()</code>) or "lavaan-0.7.2" (lavaan 0.7.2's PORT search and retries). • convergence: "default" or "newton" (magmaan's convergence check) or "lavaan-0.7.2" (lavaan 0.7.2's acceptance rule). • marker: "default" keeps requested markers; "lavaan-0.7.2" applies lavaan's weak-marker rule for continuous ML, FIML and ordinal/mixed DWLS, including its original-model retry when the switched fit is rejected. • preset: "lavaan-0.7.2" selects lavaan 0.7.2's marker, start, search and acceptance rule; explicit entries override it. <code>optimizer</code>, <code>convergence</code>, <code>preset</code> and <code>start = "lavaan-0.7.2"</code> are available for continuous ML or FIML, or ordinal or complete mixed DWLS, with unrestricted covariance and supported linear equality constraints. Inspect <code>as_lab_fit(fit)\$fitting</code>

for the resolved settings, attempts and marker_switch table. The marker rule reads each route's saturated H1 covariance. Conditional-x residual H1 and composite adaptation are unsupported. Fitting computes magmaan's inference policy; reporting methods can select an explicit lavaan inference convention. The policy covers complete-data ML, observed-data FIML and ordinal or mixed DWLS. DWLS reports one global test, the fit-function statistic (equal to its score statistic), and marks the likelihood-ratio test "inapplicable"; its covariance includes the influence of the estimated weight.

... Arguments removed in magmaan 0.2.0; each raises an error naming its replacement.

Value

An object of class magmaan.

Ordered association ML

With all indicators declared ordered, estimator = "ML" fits Gaussian ML on the estimated polychoric correlations. The ordinal default remains DWLS. Association ML requires complete rows, saturated thresholds, fixed response scales and linear constraints. Its policy covariance uses exact empirical first-stage influence and observed sensitivity; global and nested fit-function tests use the All weighted-chi-square reference, with SB and PEBA4 available as references. Modification indices use robust one-df scores. Finite-sample global tests can be liberal: in decisions/08 sparse binary two-factor Gaussian models, All rejected 10.9% at nominal 5% for N = 1000 and 6.1% for N = 4000. This caveat applies even with Gaussian latent responses.

Identification

Structurally unidentified models are refused before fitting, including ordinary refits, with a magmaan_identification_error condition. Gauge freedoms (scale, location and rotation) carry specific suggested fixes; information deficits name their parameters without an automatic fix. Its fields are status, reason, direction_types, direction_factors, suggested_fixes, and directions: a coefficient matrix with parameter labels as row names and one column per free direction. The message names these directions. Add identifying restrictions to the model; there is no ordinary override. Deliberate unidentified fits remain available through magmaanlab::fit_model(). Unchecked models fit normally.

Covariance policies

"psd" constrains every model-implied covariance matrix to be positive semidefinite. A PSD estimate on the boundary gets inference for an interior population, and says so. This applies to global and nested tests: a comparison with either PSD fit on the boundary retains psd_boundary and reports that inference is computed assuming an interior population. An improper unrestricted null fit also receives nested inference when its required tangents have full rank. boundary_nesting means a required tangent loses rank at the embedded null point, not that its fitted factor covariance is improper or singular.

barrier(lambda) maximizes the log-likelihood plus lambda times the sum, over groups, of the log determinant of the model-implied correlation matrix of all latent and observed variables: the log density of an LKJ(1 + lambda) distribution. The estimate lies in the interior of the PSD domain

and moves by $O(\lambda / N)$ at interior points. "barrier" is `barrier(0.25)`; `barrier(0)` is the unrestricted fit. Barrier fitting is experimental: the first barrier fit in a session shows a message, and the inference of every barrier fit with positive λ is unavailable with reason "penalized".

Simulation extraction

`coef(fit)` is a named vector in free-parameter order, aligned with both dimensions of `vcov(fit)`. Equality constraints can produce repeated labels; use numeric indices to distinguish those entries. `coef(summary(fit))` is a data frame with `lhs`, `op`, `rhs`, `label`, `logical free`, `est`, `se`, `z`, `pvalue`, `ci.lower`, and `ci.upper`; multi-group fits also have `group`. Defined estimates are available independently of inference. Every model has intercepts (`op` is `"~1"`), so extract parameters by name.

`fit$rows` has `group`, `rows`, `used`, and `deleted` per group. `as_lab_fit(fit)$converged` is `TRUE`, `FALSE`, or `NA` for an unchecked fit; use `isTRUE()` to count successful fits. `fit$inference` is `NULL` before inference. Otherwise `fit$inference$status` has `component`, `available`, `reason`, and `detail` for covariance, global score, and global LR. Reasons include `available`, `not_converged`, `saturated`, `unsupported_model`, `penalized` and `numeric_failure`. `fit$inference$convergence` has `rule` (the acceptance rule that set converged, "newton" by default), `converged`, `magmaan` (magmaan's own check: "passed", "failed" or "unchecked"), and `disagree`. Under a compatibility rule such as `preset = "lavaan-0.7.2"`, `disagree` is `TRUE` when the rule and magmaan's check reach different verdicts; inference follows the rule. `identified` is `TRUE` when the model is structurally (generically locally) identified, `FALSE` when it is not, and `NA` when the check does not cover the model. Ordinary fits refuse structurally unidentified models; `as_lab_fit(fit)$diagnostics$identification` holds the rank report and the parameter combinations that leave the implied moments unchanged.

`summary(fit)$tests` is `NULL` before inference is computed, otherwise a data frame with `test`, `statistic`, `df`, `reference`, `pvalue`, `recommended`, and `reason`. Unavailable tests retain a row with their typed reason. See [summary.magmaan\(\)](#) for alternative reference laws in simulation studies. The score test is primary and comes first. `recommended` is `TRUE` only for the policy's recommended p-values: score with PEBA4 for ML/FIML, `fit_function` with All for DWLS global tests, and `fit_function_difference` with All for DWLS nested tests. Use `references = c("sb", "peba4")` in `summary()` or `anova()` to include SB alongside the ML/FIML default. The likelihood-ratio test is reported because it is the standard statistic, not recommended; `references = "std"` gives its plain chi-square p-value. The likelihood-ratio test follows and tends to over-reject when N is small relative to its df . Store both package versions with simulation results. Saved fits retain their estimates and inference; reusing them with another package version is not a compatibility promise.

Examples

```
if (requireNamespace("lavaan", quietly = TRUE)) {
  fit <- magmaan("visual =~ x1 + x2 + x3",
               lavaan::HolzingerSwineford1939)
  coef(fit)
  coef(summary(fit))
}
```

Description

`coef(fit)` returns free estimates in the same order as the rows and columns of `vcov(fit)`. `coef(summary(fit))` returns the full parameter table. `vcov()` and `confint()` raise a `magmaan_inference_unavailable` condition when covariance inference is unavailable; its component and reason fields identify the missing result. `nobs()` counts observations used, summed over groups.

Usage

```
## S3 method for class 'magmaan'
coef(object, ...)

## S3 method for class 'magmaan'
vcov(object, lavaan_compat = NULL, ...)

## S3 method for class 'magmaan'
confint(object, parm, level = 0.95, test = "wald", lavaan_compat = NULL, ...)

## S3 method for class 'magmaan'
nobs(object, ...)

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

## S3 method for class 'summary.magmaan'
coef(object, ...)

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

Arguments

<code>object, x</code>	A magmaan fit or its summary, as appropriate.
<code>...</code>	Unused.
<code>lavaan_compat</code>	Lavaan compatibility bundle: <code>NULL</code> (default policy), <code>"ML"</code> , <code>"MLM"</code> , <code>"MLR"</code> , <code>"DWLS"</code> , <code>"WLSMV"</code> , <code>"ULS"</code> , <code>"ULSMV"</code> or <code>"WLS"</code> , compatible with the fitted estimator. Lavaan bundles compute on demand without refitting. For parity, compare lavaan fits with the same model and estimation settings, including <code>meanstructure = TRUE</code> and <code>fixed.x = FALSE</code> . FIML supports <code>"ML"</code> and <code>"MLR"</code> with lavaan's missing = <code>"m1"</code> recipes. Complete mixed DWLS supports <code>"WLSMV"</code> , with NACOV covariance and scaled-shifted global/Satorra-2000 nested tests. Mixed delta/theta reporting is validated at identical parameter points; grouped/theta retained end-points have limited validation. Other mixed bundles and missing mixed observations remain unavailable.

parm	Parameter names or positive integer indices; omitted for all free parameters. Indices distinguish parameters with repeated labels.
level	One confidence level strictly between zero and one.
test	Interval test, currently only "wald".
digits	Number of digits printed after the decimal point.

Value

`coef()` returns a named numeric vector for fits and a data frame for summaries. `vcov()` returns the parameter covariance matrix; `confint()` a matrix with two columns; `nobs()` the total count. Printing returns its input invisibly.

magmaan_model	<i>Construct a model for magmaan()</i>
---------------	--

Description

Builds the model once: its parameters, structural choices and data schema. Fit it to any number of datasets with `magmaan()`; fitting never modifies it. Grouped and ordinal models need prototype to declare their groups and categories.

Usage

```
magmaan_model(
  model,
  prototype = NULL,
  ordered = NULL,
  group = NULL,
  group.equal = NULL,
  group.partial = NULL,
  identification = "marker",
  parameterization = "delta"
)
```

Arguments

model	lavaan model syntax, one string, or a model specification from <code>magmaanlab::model_spec()</code> carrying lavaan syntax. A specification supplies its own ordered variables, grouping, identification, parameterization and equality constraints; explicit arguments must agree with it. A specification that fixes exogenous covariates (<code>fixed_x = TRUE</code> with observed covariates) is rejected. Explicit <code>magmaanlab::mplus_model()</code> specifications preserve their source and tables; plain strings are never detected as Mplus. Mplus construction records <code>fittable</code> and <code>mplus_refusals</code> ; fitting refuses conditional observed X, <code>NOMEANSTRUCTURE</code> , and summary inputs without <code>MEANS</code> with a <code>magmaan_mplus_error</code> condition listing every needed edit, with a character vector <code>reason</code> and a named character vector <code>edit</code>
-------	---

	(names equal to reason). Printing the model shows fittability and all needed edits. Add all X variance mentions (x_1 x_2 ;) to specify the joint model, remove NOMEANSTRUCTURE, or supply raw observations through <code>magmaanlab::mplus_data()</code> , respectively.
prototype	A data frame that declares the data schema: the model's variables, the grouping column's groups and their order, and the categories of the ordered variables and their order. Factor levels declare groups and categories completely, including levels without rows, so a zero-row data frame suffices. Otherwise groups follow their first appearance and categories their sorted values. Only the schema is used: no estimates, starts or weights come from the prototype's rows.
ordered	Names of the ordered (categorical) variables.
group	Name of the grouping column.
group.equal, group.partial	Cross-group equality constraints, as in lavaan.
identification	"marker" (first loading fixed to one) or "std.lv" (latent variances fixed to one).
parameterization	"delta" or "theta", for ordered variables.

Details

Every model has a mean structure: when the syntax does not restrict the means, each observed variable gets a free intercept, which changes no other estimate, standard error or test. Exogenous observed covariates are random variables with free variances and covariances (there is no fixed x).

Value

An object of class `magmaan_model`.

Identification

Construction computes a data-free check of generic local identification and stores it in `identification_report`. Printing shows `identified`, `NOT identified`, or `unchecked` with its reason beside the `marker/std.lv` setting. Fits reuse the report. Construction always returns the model; `magmaan()` refuses an unidentified model before fitting. Unchecked models fit normally. The check does not assess global uniqueness or weak identification at an estimate, and never adds identifying constraints.

Schema checks

Each fit checks its data against the frozen schema. Data with an undeclared group or category, changed factor levels, a declared group without rows or a declared category without complete observations raise a condition of class `magmaan_schema_error` whose reason is `"undeclared_group"`, `"undeclared_category"`, `"changed_levels"`, `"empty_group"` or `"empty_category"`. Construct the model again to change its schema.

Group order follows the prototype's factor levels; lavaan instead orders groups by first appearance. Group-specific modifiers such as `c(a, b)*x` follow the model's order, and so does the reference group whose latent means are fixed under `group.equal = "intercepts"`.

Examples

```
if (requireNamespace("lavaan", quietly = TRUE)) {
  d <- lavaan::HolzingerSwineford1939
  m <- magmaan_model("visual =~ x1 + x2 + x3", prototype = d,
                    group = "school", group.equal = "loadings")
  fit <- magmaan(m, d)
}
```

modindices

Modification indices and equality releases

Description

Reports misspecification-robust one-df score statistics with chi-square(1) limiting p-values. There is no reference-distribution selector.

Usage

```
modindices(fit, test = c("score", "lr"), candidates = NULL, releases = TRUE)
```

Arguments

<code>fit</code>	A <code>magmaan()</code> fit.
<code>test</code>	Robust one-df score tests (default) or refit-based LR tests.
<code>candidates</code>	NULL for all rows, operator codes <code>=~</code> , <code>~~</code> , <code>~</code> , <code>==</code> , or a data frame with lhs, op, rhs and optional group and kind. Selection preserves the full table's order.
<code>releases</code>	Include univariate equality releases.

Details

ML uses observed sensitivity and empirical likelihood scores; FIML uses observed pattern-score geometry. All-ordinal and mixed DWLS use exact first-stage sampling rows and the estimated-weight channel with observed sensitivity. Shared labels, explicit affine equalities and `group.equal` restrictions are tested one at a time.

LR refits one augmented model per candidate from the embedded fitted estimate and uses the same nested policy as `anova()`: ML/FIML LR with PEBA4 (equal to All at one df), or DWLS fit-function differences with All. LR is optional because these refits are costly. EPC columns retain the one-step score EPC rather than a refitted parameter change.

The generator covers loadings, residual/latent covariances and absent regressions among variables already participating in structural equations, including reverse paths. Unidentified regression directions retain typed unavailable rows. ML/FIML score statistics use observed nuisance sensitivity and an expected-information metric; EPCs use expected information. Singular nuisance sensitivity or nonpositive score variance produces `numeric_failure` and NA values; LR may still refit these candidates. Failed refits carry `refit_failed` or `not_converged`. Penalized, non-converged and unsupported fit regimes keep typed rows.

Value

A data frame with kind (fixed or release), lhs, op, rhs, group, test, statistic, df, pvalue, epc, sepc.lv, sepc.all and reason. Rows remain in candidate order. Equality rows describe native affine restrictions in parameter-label coordinates.

See Also

[anova.magmaan](#), [magmaan](#)

summary.magmaan

Summary of a magmaan fit

Description

The fit, its parameter table and its global tests. `coef()` on the summary returns the parameter table: one row per model parameter, fixed and free, including defined ($:=$) parameters, with the estimate and, when inference is available, its selected standard error, z-statistic, p-value and Wald interval. `coef(fit)` stays the vector of free estimates that matches `vcov(fit)`.

Usage

```
## S3 method for class 'magmaan'
summary(
  object,
  level = 0.95,
  lavaan_compat = NULL,
  references = NULL,
  standardized = FALSE,
  fit_measures = FALSE,
  ...
)
```

Arguments

object	A magmaan() fit.
level	Confidence level of the intervals.
lavaan_compat	Inference bundle, as in vcov.magmaan() .
references	Character vector of reference laws, or NULL for the policy defaults.
standardized	Add std.lv, std.lv.se, std.all and std.all.se columns and an R-squared table. Delta-method SEs use the selected covariance. Without covariance, estimates remain available and SEs are NA with the inference reason attached to the tables. Defined parameters have no declared standardized scale and are NA (unsupported_defined_scale).
fit_measures	Logical; compute and attach fit_measures() on request. With lavaan_compat, returns the selected bundle's standard, scaled and robust measures, intervals and close-fit p-values, including FIML ML/MLR where supported.
...	Unused.

Value

An object of class `summary.magmaan`. With `standardized = TRUE`, `r2` is a data frame with variable, `r2` and `r2.se` columns (and `group` for grouped models), covering endogenous observed and latent variables.

Simulation studies

`references` accepts case-insensitive `std`, `sb`, `ss`, `mv`, `scaled_f`, `all`, `pall`, `peba<k>` and `eba<k>` (integer $k \geq 1$, up to the C++ integer limit). Output names are lower case. Each available test has one row per requested reference, using its policy statistic and stored spectrum; covariance and fitting choices stay fixed. Defaults are PEBA4 for ML/FIML and All for DWLS global and nested tests. Use `references = c("sb", "peba4")` to include the SB comparator. The base columns are `test`, `statistic`, `df`, `reference`, `pvalue`, `recommended`, `reason`. `test` uses stable codes `score`, `lr`, `fit_function`, `fit_function_difference`. Unavailable tests retain one row with missing reference and p-value and a typed reason. Select rows by `test` and `reference`; migrate `p.sb/p.peba4` to `pvalue` on the corresponding rows. `recommended` is `TRUE` only for the policy's recommended p-values: `score` with PEBA4 for ML/FIML, or `fit_function` with All for DWLS global tests. The likelihood-ratio test is reported because it is the standard statistic, not recommended; `references = "std"` gives its plain chi-square p-value. All rows are printed, including LR rows with the small-sample caveat. EBA/pEBA partition the retained spectrum (including zero eigenvalues) into blocks of size `ceiling(df / k)`; when $k \geq df$, blocks are singletons. EBA then equals All, and pEBA equals pAll. References cannot be combined with `lavaan_compat`; compatibility rows append `unscaled.statistic`, `scale` and `shift` and are never marked recommended.

Index

`anova.magmaan`, [2](#), [14](#)
`as_lab_fit`, [3](#)

`barrier`, [4](#)
`barrier()`, [7](#)

`coef.magmaan` (`magmaan_methods`), [10](#)
`coef.summary.magmaan` (`magmaan_methods`),
 [10](#)
`confint.magmaan` (`magmaan_methods`), [10](#)

`fit_measures`, [4](#)
`fit_measures()`, [14](#)
`fitted.magmaan`, [5](#)

`infer`, [6](#)
`infer()`, [7](#)

`magmaan`, [6](#), [14](#)
`magmaan()`, [2–7](#), [11](#), [12](#), [14](#)
`magmaan_methods`, [10](#)
`magmaan_model`, [11](#)
`magmaan_model()`, [6](#), [7](#)
`modindices`, [13](#)

`nobs.magmaan` (`magmaan_methods`), [10](#)

`print.magmaan` (`magmaan_methods`), [10](#)
`print.magmaan_anova` (`anova.magmaan`), [2](#)
`print.summary.magmaan`
 (`magmaan_methods`), [10](#)

`summary.magmaan`, [14](#)
`summary.magmaan()`, [3](#), [9](#)

`vcov.magmaan` (`magmaan_methods`), [10](#)
`vcov.magmaan()`, [14](#)