Package 'modelscompete4'

May 9, 2026

Title Compare Nested and Non-Nested Structural Equation Models

Version 0.2.6

Date 2026-04-23

Description A comprehensive package for comparing multiple Structural Equation Models (SEM). Supports both nested and non-nested model comparisons, chi-square difference tests, and extraction of multiple fit indices including AIC (Akaike Information Criterion), BIC (Bayesian Information Criterion), CFI (Comparative Fit Index), TLI (Tucker-Lewis Index), RMSEA (Root Mean Square Error of Approximation), and SRMR (Standardized Root Mean Square Residual). Built on top of the 'lavaan' package for seamless SEM model comparison workflows. The Vuong test (Vuong, 1989) for non-nested models is used as the statistical test.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.3

Imports lavaan (>= 0.6), stats, boot, ggplot2, nonnest2, tidyr

Suggests testthat (>= 3.0.0)

URL https://github.com/ssjerf-stack/modelscompete4

BugReports https://github.com/ssjerf-stack/modelscompete4/issues

NeedsCompilation no

Author Jerf Yeung [aut, cre]

Maintainer Jerf Yeung <ssjerf@gmail.com>

Repository CRAN

Date/Publication 2026-04-24 20:10:02 UTC

Contents

`.onAttach` *Package startup message*

Description

Package startup message

Usage

```
.onAttach(libname, pkgname)
```

Arguments

`libname`	library location
`pkgname`	package name

Value

No return value, called for side effects.

`.onLoad` *Package load*

Description

Package load

Usage

```
.onLoad(libname, pkgname)
```

Arguments

`libname`	library location
`pkgname`	package name

Value

No return value, called for side effects.

`bootstrap_lavaan_comparison` *Bootstrap Comparison for Lavaan Models*

Description

Perform bootstrap-based comparison of lavaan models

Usage

```
bootstrap_lavaan_comparison(
  model1,
  model2,
  R = 1000,
  parallel = "no",
  ncpus = 1
)
```

Arguments

`model1`	First lavaan model
`model2`	Second lavaan model
`R`	Number of bootstrap replications (default: 1000)
`parallel`	Type of parallel processing (if any)
`ncpus`	Number of CPUs to use for parallel processing

Value

A list containing bootstrap results

Examples

```
library(lavaan)
model1 <- 'F1 =~ x1 + x2 + x3'
model2 <- 'F1 =~ x1 + x2 + x3 + x4'
fit1 <- cfa(model1, data = HolzingerSwineford1939)
fit2 <- cfa(model2, data = HolzingerSwineford1939)
boot_result <- bootstrap_lavaan_comparison(fit1, fit2, R = 100)
print(boot_result)
```

compare_latent_models *Compare Latent Variable Models*

Description

Compare Latent Variable Models

Usage

```
compare_latent_models(
  ...,
  nested = FALSE,
  fit_measures = c("chisq", "df", "pvalue", "cfi", "tli", "rmsea", "srmr"),
  method = "default",
  verbose = TRUE
)
```

Arguments

...	lavaan model objects
nested	logical, whether models are nested
fit_measures	character vector of fit measures to extract
method	comparison method
verbose	Logical; if TRUE, progress messages are printed.

Value

A latent_comparison object

compare_models *Compare Multiple Nested or Non-Nested Structural Equation Models*

Description

This is the core function of the modelscompete4 package. It automatically fits a list of SEM models, determines their nesting relationship, and performs the appropriate statistical comparison (chi-square difference test for nested models, Vuong test for non-nested models).

Usage

```
compare_models(
  model_list,
  data,
  estimator = "ML",
  se = "standard",
  bootstrap = 1000,
  parallel = "no",
  verbose = TRUE,
  ...
)
```

Arguments

model_list	A named list. Each element is a character string specifying the model syntax in lavaan format.
data	A data.frame containing the observed variables used in the models.
estimator	The estimator to be used (e.g., "ML"). Passed to sem.
se	Type of standard errors. Default is "standard". Use "bootstrap" for bootstrapped standard errors and confidence intervals.
bootstrap	Number of bootstrap draws if se="bootstrap". Default is 1000.
parallel	Method for parallel processing for bootstrapping ("multicore", "snow", or "no"). Recommended for large samples.
verbose	Logical; if TRUE, progress messages are printed.
...	Additional arguments passed to sem.

Value

An object of class modelscompete4. This is a list containing:

- fit_list: The list of fitted lavaan objects.
- fit_table: A data.frame of key fit indices for all models.
- comparison_matrix: A matrix showing pairwise nesting relationships and test results.
- test_results: Detailed results of the statistical tests performed.
- bootstrap_summary: Summary of bootstrapped results if requested.

---

compare_models_advanced_lv

Advanced Model Comparison with Latent Variable Support

Description

Advanced Model Comparison with Latent Variable Support

Usage

```
compare_models_advanced_lv(
  models,
  model_names = NULL,
  model_types = NULL,
  criteria = c("AIC", "BIC", "CFI", "TLI", "RMSEA", "SRMR"),
  latent_indicators = NULL,
  bootstrap = FALSE,
  n_bootstrap = 1000
)
```

Arguments

models	List of model objects (lm, lavaan)
model_names	Character vector of model names
model_types	Character vector of model types ("lm", "lavaan")
criteria	Criteria to calculate
latent_indicators	List of latent variable indicators (for lavaan)
bootstrap	Logical, whether to perform bootstrapping
n_bootstrap	Number of bootstrap replications

Value

Comparison results with latent variable support

---

extract_latent_fit *Extract Latent Variable Fit Indices*

Description

Extract Latent Variable Fit Indices

Usage

```
extract_latent_fit(model)
```

Arguments

model	lavaan model object

Value

Comprehensive fit indices for latent variable model

`extract_latent_parameters` *Extract Parameters from Lavaan Models*

Description

This function extracts parameters (loadings, variances, etc.) from a lavaan model. P-values are formatted appropriately (e.g., <0.001 for very small values).

Usage

```
extract_latent_parameters(
  model,
  type = "loadings",
  standardized = FALSE,
  digits = 3,
  ...
)
```

Arguments

`model`	A fitted lavaan model.
`type`	Type of parameters to extract: "loadings", "variances", or "all". Default is "loadings".
`standardized`	Logical; if TRUE, returns standardized estimates. Default is FALSE.
`digits`	Number of decimal places for p-value formatting (default=3)
`...`	Additional arguments passed to `lavaan::parameterEstimates` or `lavaan::standardizedSolution`.

Value

A data frame containing the extracted parameters with formatted p-values.

Examples

```
library(lavaan)
model <- 'F1 =~ x1 + x2 + x3'
fit <- cfa(model, data = HolzingerSwineford1939)
extract_latent_parameters(fit, type = "loadings")
```

plot_latent_comparison

Plot Latent Model Comparison Results

Description

Creates visualization of model comparison results

Usage

```
plot_latent_comparison(x, type = "fit", ...)
```

Arguments

x	An object of class 'modelscompete4' or 'latent_comparison'
type	Type of plot: 'fit' for fit indices, 'diff' for differences
...	Additional arguments passed to plotting functions

Value

A ggplot object (if ggplot2 and tidyr are available), otherwise NULL

print.latent_comparison

Print method for latent_comparison objects

Description

Prints a summary of latent model comparison results.

Usage

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

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

Arguments

x	An object of class 'latent_comparison'
digits	Number of digits to display (default: 3)
...	Additional arguments passed to print method

Value

Invisibly returns the input object.

Invisibly returns the input object

`print.modelscompete4_advanced` *Print Method for modelscompete4_advanced Objects*

Description

Print Method for modelscompete4_advanced Objects

Usage

```
## S3 method for class 'modelscompete4_advanced'
print(x, ...)
```

Arguments

`x`	A modelscompete4_advanced object
`...`	Additional arguments passed to print

Value

Invisibly returns the input object `x`.

Index