Package ‘piecewiseSEM’

July 23, 2025

Type Package

Title Piecewise Structural Equation Modeling

Version 2.3.0.1

Date 2024-06-06

Maintainer Jon Lefcheck <`jslefche@gmail.com`>

Description Implements piecewise structural equation modeling from a single list of structural equations, with new methods for non-linear, latent, and composite variables, standardized coefficients, query-based prediction and indirect effects. See <`http://jslefche.github.io/piecewiseSEM/`> for more.

Depends R (>= 4.4.0)

URL `https://github.com/jslefche/`

BugReports `https://github.com/jslefche/piecewiseSEM/issues`

Imports car, DiagrammeR, emmeans, igraph, lme4, multcomp, MuMIn, MASS, methods, nlme, performance

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.3.1

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Jon Lefcheck [aut, cre],
Jarrett Byrnes [aut],
James Grace [aut]

Repository CRAN

Date/Publication 2024-06-11 14:10:03 UTC

Contents

`piecewiseSEM-package` *The 'piecewiseSEM' package*

Description

Piecewise structural equation modeling

Fitting and evaluation of piecewise structural equation models, complete with goodness-of-fit tests, estimates of (standardized) path coefficients, and evaluation of individual model fits (e.g., through R-squared values). Compared with traditional variance-covariance based SEM, piecewise SEM allows for fitting of models to different distributions through GLM and/or hierarchical/nested random structures through (G)LMER. Supported model classes include: `lm`, `glm`, `gls`, `Sarlm`, `lme`, `glmmPQL`, `lmerMod`, `merModLmerTest`,`glmerMod`, `glmmTMB`, `gam`.

Package:	piecewiseSEM
Type:	Package
Version:	2.3.0.1
Date:	2024-06-11
Depends:	R (>= 4.4.0), car, DiagrammeR, emmeans, igraph, lme4, multcomp, MuMIn, MASS, methods, nlme
License:	MIT

The primary functions in the package are `psem` which unites structural equations in a single model, and `summary.psem` can be used on an object of class `psem` to provide various summary statistics for evaluation and interpretation.

Author(s)

Jon Lefcheck <jslefche@gmail.com>

References

Shipley, Bill. "A new inferential test for path models based on directed acyclic graphs." Structural Equation Modeling 7.2 (2000): 206-218.

Shipley, Bill. Cause and correlation in biology: a user's guide to path analysis, structural equations and causal inference. Cambridge University Press, 2002.

Shipley, Bill. "Confirmatory path analysis in a generalized multilevel context." Ecology 90.2 (2009): 363-368.

Shipley, Bill. "The AIC model selection method applied to path analytic models compared using a d-separation test." Ecology 94.3 (2013): 560-564.

Shipley, Bill, and Jacob C. Douma. "Generalized AIC and chi-squared statistics for path models consistent with directed acyclic graphs." Ecology 101.3 (2020): e02960.

Grace, J.B., Johnson, D.A., Lefcheck, J.S., and Byrnes, J.E. "Standardized Coefficients in Regression and Structural Models with Binary Outcomes." Ecosphere 9(6): e02283.

Nakagawa, Shinichi, Paul CD Johnson, and Holger Schielzeth. "The coefficient of determination R2 and intra-class correlation coefficient from generalized linear mixed-effects models revisited and expanded." Journal of the Royal Society Interface 14.134 (2017): 20170213.

See Also

Useful links:

- `https://github.com/jslefche/`
- Report bugs at `https://github.com/jslefche/piecewiseSEM/issues`

`AIC.psem` *Generic function for SEM AIC(c) score*

Description

Generic function for SEM AIC(c) score

Usage

```
## S3 method for class 'psem'
AIC(object, ..., AIC.type = "loglik", aicc = FALSE)
```

Arguments

`object`	a psem object
`...`	additional arguments to AIC
`AIC.type`	whether the log-likelihood `"loglik"` or d-sep `"dsep"` AIC score should be reported. Default is `"loglik"`
`aicc`	whether correction for small sample size should be applied. Default is `FALSE`

Examples

```
mod <- psem(
lm(rich ~ cover, data = keeley),
lm(cover ~ firesev, data = keeley),
lm(firesev ~ age, data = keeley),
data = keeley
)

# Get log-likelihood based AIC
AIC(mod)

# Get d-sep based AIC
AIC(mod, AIC.type = "dsep")
```

`AIC_psem` *Information criterion values for SEM*

Description

Information criterion values for SEM

Usage

```
AIC_psem(
  modelList,
  AIC.type = "loglik",
  Cstat = NULL,
  add.claims = NULL,
  basis.set = NULL,
  direction = NULL,
  interactions = FALSE,
  conserve = FALSE,
  conditional = FALSE,
  .progressBar = FALSE
)
```

Arguments

`modelList`	a list of structural equations
`AIC.type`	whether the log-likelihood `"loglik"` or d-sep `"dsep"` AIC score should be reported. Default is `"loglik"`
`Cstat`	Fisher's C statistic obtained from `fisherC`
`add.claims`	an optional vector of additional independence claims (P-values) to be added to the basis set
`basis.set`	An optional list of independence claims.
`direction`	a vector of claims defining the specific directionality of any independence claim(s)
`interactions`	whether interactions should be included in independence claims. Default is FALSE
`conserve`	whether the most conservative P-value should be returned (See Details) Default is FALSE
`conditional`	whether the conditioning variables should be shown in the table. Default is FALSE
`.progressBar`	an optional progress bar. Default is FALSE

Value

a data.frame of AIC, AICc, d.f., and sample size

Author(s)

Jon Lefcheck <LefcheckJ@si.edu>, Jim Grace

References

Shipley, Bill, and Jacob C. Douma. "Generalized AIC and chi-squared statistics for path models consistent with directed acyclic graphs." Ecology 101.3 (2020): e02960.

Shipley, Bill. "The AIC model selection method applied to path analytic models compared using a d-separation test." Ecology 94.3 (2013): 560-564.

`anova.psem` *ANOVA and chi-squared difference test for model comparison*

Description

Compute analysis of variance table for one or more structural equation models.

Usage

```
## S3 method for class 'psem'
anova(object, ..., digits = 3, anovafun = "Anova")
```

Arguments

`object`	a `psem` object
`...`	additional objects of the same type
`digits`	number of digits to round results. Default is 3
`anovafun`	The function used for ANOVA. Defaults to `Anova`

Details

Additional models will be tested against the first model using a Chi-squared difference test.

Value

an F, LRT, or other table for a single model, or a list of comparisons between multiple models

Author(s)

Jon Lefcheck <lefcheckj@si.edu>, Jarrett Byrnes <jarrett.byrnes@umb.edu>

See Also

`Anova`

Examples

```
data(keeley)

mod1 <- psem(
lm(rich ~ cover, data = keeley),
lm(cover ~ firesev, data = keeley),
lm(firesev ~ age, data = keeley),
data = keeley
)

# get type II Anova
anova(mod1)
```

```
# conduct LRT
mod2 <- psem(
  lm(rich ~ cover, data = keeley),
  lm(cover ~ firesev, data = keeley),
  age ~ 1,
  data = keeley
)

anova(mod1, mod2)
```

`as.psem` *Convert list to psem object*

Description

Convert list to psem object

Usage

```
as.psem(object, Class = "psem")
```

Arguments

`object`	any R object
`Class`	the name of the class to which `object` should be coerced

`basisSet` *Derivation of the basis set*

Description

Acquires the set of independence claims–or the 'basis set'–for use in evaluating the goodness-of-fit for piecewise structural equation models.

Usage

```
basisSet(modelList, direction = NULL, interactions = FALSE)
```

Arguments

`modelList`	A list of structural equations
`direction`	a vector of claims defining the specific directionality of any independence claim(s)
`interactions`	whether interactions should be included in independence claims. Default is FALSE

Details

This function returns a list of independence claims. Each claim is a vector of the predictor of interest, followed by the response, and, if present, any conditioning variables.

Relationships among exogenous variables are omitted from the basis set because the directionality is unclear–e.g., does temperature cause latitude or does latitude cause temperature?–and the assumptions of the variables are not specified in the list of structural equations, so evaluating the relationship becomes challenging without further input from the user. This creates a circular scenario whereby the user specifies relationships among exogenous variables, raising the issue of whether they should be included as directed paths if they can be assigned directional relationships.

Paths can be omitted from the basis set by specifying them as correlated errors using `%~~%` or by assigning a directionality using the argument `direction`, e.g. `direction = c("X <- Y")`. This can be done if post hoc examination of the d-sep tests reveals nonsensical independence claims (e.g., arthropod abundance predicting photosynthetically-active radiation) that the user may wish to exclude from evaluation.

Value

A `list` of independence claims.

Author(s)

Jon Lefcheck <LefcheckJ@si.edu>

References

Shipley, Bill. "A new inferential test for path models based on directed acyclic graphs." Structural Equation Modeling 7.2 (2000): 206-218.

See Also

`dSep`

`cbind_fill` *Bind data.frames of differing dimensions*

Description

From: https://stackoverflow.com/a/31678079

Usage

```
cbind_fill(...)
```

Arguments

`...`	data.frames to be bound, separated by commas @keywords internal

`cerror` *Correlated errors*

Description

Calculates partial correlations and partial significance tests.

Usage

```
cerror(formula., modelList, data = NULL)
```

Arguments

`formula.`	A formula specifying the two correlated variables using `%~~%`.
`modelList`	A list of structural equations.
`data`	A `data.frame` containing the data used in the list of equations.

Details

If the variables are exogenous, then the correlated error is the raw bivariate correlation.

If the variables are endogenous, then the correlated error is the partial correlation, accounting for the influence of any predictors.

The significance of the correlated error is conducted using `cor.test` if the variables are exogenous. Otherwise, a t-statistic is constructed and compared to a t-distribution with N - k - 2 degrees of freedom (where N is the total number of replicates, and k is the total number of variables informing the relationship) to derive a P-value.

Value

Returns a `data.frame` containing the (partial) correlation and associated significance test.

Author(s)

Jon Lefcheck <lefcheckj@si.edu>

See Also

`%~~%`

Examples

```
# Generate example data
dat <- data.frame(x1 = runif(50),
  x2 = runif(50), y1 = runif(50),
    y2 = runif(50))

# Create list of structural equations
```

```
sem <- psem(
  lm(y1 ~ x1 + x2, dat),
  lm(y2 ~ y1 + x1, dat)
)

# Look at correlated error between x1 and x2
# (exogenous)
cerror(x1 %~~% x2, sem, dat)

# Same as cor.test
with(dat, cor.test(x1, x2))

# Look at correlatde error between x1 and y1
# (endogenous)
cerror(y1 %~~% x1, sem, dat)

# Not the same as cor.test
# (accounts for influence of x1 and x2 on y1)
with(dat, cor.test(y1, x1))

# Specify in psem
sem <- update(sem, x1 %~~% y1)

coefs(sem)
```

`coefs` *Extract path coefficients*

Description

Extracts (standardized) path coefficients from a `psem` object.

Usage

```
coefs(
  modelList,
  standardize = "scale",
  standardize.type = "latent.linear",
  test.statistic = "F",
  test.type = "II",
  intercepts = FALSE
)
```

Arguments

`modelList`	A list of structural equations, or a model.
`standardize`	The type of standardization: `none`, `scale`, `range`. Default is `scale`.

`standardize.type`	The type of standardized for non-Gaussian responses: `latent.linear`, `Menard.OE`. Default is `latent.linear` for binomial; otherwise it is `Menard.OE`.
`test.statistic`	the type of test statistic generated by `Anova`
`test.type`	the type of test for significance of categorical variables from `Anova`. Default is type "II".
`intercepts`	Whether intercepts should be included in the coefficients table. Default is FALSE.

Details

P-values for models constructed using `lme4` are obtained using the Kenward-Roger approximation of the denominator degrees of freedom as implemented in the `Anova` function.

Different forms of standardization can be implemented using the `standardize` argument:

- `none` No standardized coefficients are reported.
- `scale` Raw coefficients are scaled by the ratio of the standard deviation of x divided by the standard deviation of y. See below for cases pertaining to GLM.
- `range` Raw coefficients are scaled by a pre-selected range of x divided by a preselected range of y. The default argument is `range` which takes the two extremes of the data, otherwise the user must supply must a named `list` where the names are the variables to be standardized, and each entry contains a vector of length == 2 to the ranges to be used in standardization.

For non-Gaussian responses, standardized coefficients are obtained in one of two ways:

- `latent.linear` Referred to in Grace et al. 2019 as the standard form of the latent-theoretic (LT) approach. In this method, there is assumed to be a continuous latent propensity, y*, that underlies the observed binary responses. The standard deviation of y* is computed as the square-root of the variance of the predictions (on the linear or 'link' scale) plus the distribution-specific theoretical variance in the case of binomial responses (for logit links: pi^2/3, for probit links: 1).
- `Menard.OE` Referred to in Grace et al. 2019 as the standard form of the observed-empirical (OE) approach. In this method, error variance is based on the differences between predicted scores and the observed binary data. The standard deviation used for standardization is computed as the square-root of the variance of the predictions (on the linear scale) plus the correlation between the observed and predicted (on the original or 'response' scale) values of y.

For categorical predictors: significance is determined using ANOVA (or analysis of deviance). Because n-1 coefficients are reported for n levels, the output instead reports model-estimated means in the `Estimate` column. This is done so all n paths in the corresponding path diagram have assignable values.

The means are generated using function `emmeans` in the `emmeans` package. Pairwise contrasts are further conducted among all levels using the default correction for multiple testing. The results of those comparisons are given in the significance codes (e.g., "a", "b", "ab") as reported in the `multcomp::cld` function.

For non-linear variables (i.e., smoothing functions from `mgcv::gam`), there are no linear estimates reported.

Value

Returns a `data.frame` of coefficients, their standard errors, degrees of freedom, and significance tests.

Author(s)

Jon Lefcheck <LefcheckJ@si.edu>, Jim Grace

References

Grace, J.B., Johnson, D.A., Lefcheck, J.S., and Byrnes, J.E. "Standardized Coefficients in Regression and Structural Models with Binary Outcomes." Ecosphere 9(6): e02283.

See Also

`Anova`, `emmeans`, `cld`

Examples

```
mod <- psem(
lm(rich ~ cover, data = keeley),
lm(cover ~ firesev, data = keeley),
lm(firesev ~ age, data = keeley),
data = keeley
)

coefs(mod)
```

`dSep` *Tests of directed separation*

Description

Evaluation of conditional independence claims to be used in determining the goodness-of-fit for piecewise structural equation models.

Usage

```
dSep(
  modelList,
  basis.set = NULL,
  direction = NULL,
  interactions = FALSE,
  conserve = FALSE,
  conditioning = FALSE,
  .progressBar = TRUE
)
```

Arguments

`modelList`	A list of structural equations created using `psem`.
`basis.set`	An optional list of independence claims.
`direction`	A `vector` of claims defining the specific directionality of independence claims; for use in special cases (see Details).
`interactions`	whether interactions should be included in independence claims. Default is FALSE
`conserve`	Whether the most conservative P-value should be returned; for use in special cases (see Details). Default is FALSE.
`conditioning`	Whether the conditioning variables should be shown in the summary table. Default is FALSE.
`.progressBar`	An optional progress bar. Default is TRUE.

Details

In cases involving non-normally distributed responses in the independence claims that are modeled using generalized linear models, the significance of the independence claim is not reversible (e.g., the P-value of Y ~ X is not the same as X ~ Y). This is due to the transformation of the response via the link function. In extreme cases, this can bias the goodness-of-fit tests. `summary.psem` will issue a warning when this case is present and provide guidance for solutions.

One solution is to specify the directionality of the relationship using the `direction` argument, e.g. `direction = c("X <- Y")`. Another is to run both tests (Y ~ X, X ~ Y) and return the most conservative (i.e., lowest) P-value, which can be toggled using the `conserve = TRUE` argument.

Value

Returns a `data.frame` of independence claims and their significance values.

Author(s)

Jon Lefcheck <lefcheckj@si.edu>, Jarrett Byrnes

References

Shipley, Bill. "A new inferential test for path models based on directed acyclic graphs." Structural Equation Modeling 7.2 (2000): 206-218.

See Also

`basisSet`

`evaluateClasses` *Evaluate model classes and stop if unsupported model class*

Description

Evaluate model classes and stop if unsupported model class

Usage

```
evaluateClasses(x)
```

Arguments

`x`	a list of structural equations or a model object

`fisherC` *Summarize tests of directed separation using Fisher's C statistic*

Description

Summarize tests of directed separation using Fisher's C statistic

Usage

```
fisherC(
  dTable,
  add.claims = NULL,
  basis.set = NULL,
  direction = NULL,
  interactions = FALSE,
  conserve = FALSE,
  conditional = FALSE,
  .progressBar = FALSE
)
```

Arguments

`dTable`	a `data.frame` containing tests of directed separation from `dSep`
`add.claims`	an optional vector of additional independence claims (i.e., P-values) to be added to the basis set
`basis.set`	An optional list of independence claims.
`direction`	a vector of claims defining the specific directionality of any independence claim(s)
`interactions`	whether interactions should be included in independence claims. Default is FALSE

`conserve`	whether the most conservative P-value should be returned. Default is FALSE
`conditional`	whether the conditioning variables should be shown in the table. Default is FALSE
`.progressBar`	an optional progress bar. Default is FALSE

Value

a data.frame corresponding to the C statistic, d.f., and P-value

---

`getDAG` *Generate adjacency matrix from list of structural equations*

Description

Generate adjacency matrix from list of structural equations

Usage

```
getDAG(modelList)
```

Arguments

`modelList`	A list of structural equations

`getSortedPsem` *Get a sorted psem object in DAG order*

Description

Takes a [psem] object, pulls out the DAG, and then sorts the psem object into the order of the DAG (from exogenous to terminal endogenous variable) for use by other functions. Note: removes correlated errors.

Usage

```
getSortedPsem(object, keepdata = TRUE)
```

Arguments

`object`	A fit [psem] object
`keepdata`	Defaults to TRUE. Should the data with the psem be included in the returned object?

Value

A new [psem] object, without the data.

`import` *Functions to import from dependencies*

Description

Functions to import from dependencies

`keeley` *Data set from Grace & Keeley (2006)*

Description

Data set from Grace & Keeley (2006)

Usage

```
keeley
```

Format

A `data.frame` with 90 observations of 8 variables.

distance Distance to coast

elev Elevation from sea level

abiotic Abiotic favorability

age Age of stand before fire

hetero Plot heterogeneity

firesev Severity of fire

cover Cover of plants

rich Plant species richness

`LLchisq` *Generalized chi-squared for piecewise SEM*

Description

Derivation of log-likelihoods to be used in determining the goodness-of-fit for piecewise structural equation models.

Usage

```
LLchisq(
  modelList,
  basis.set = NULL,
  direction = NULL,
  interactions = FALSE,
  conserve = FALSE
)
```

Arguments

`modelList`	A list of structural equations created using `psem`.
`basis.set`	An optional list of independence claims.
`direction`	A `vector` of claims defining the specific directionality of independence claims; for use in special cases (see `dSep`.
`interactions`	whether interactions should be included in basis set. Default is FALSE
`conserve`	Whether the most conservative log-likelihood should be returned; for use in special cases (see Details). Default is FALSE.

Details

Here, a list of saturated models is first derived from the list of structured equations using the basis set. Then, the differences in summed log-likelihoods are computed and used to calculate the Chi-squared statistic.

Value

a data.frame corresponding to the Chi-squared statistic, d.f., and P-value

Author(s)

Jon Lefcheck <LefcheckJ@si.edu>

References

Shipley, Bill, and Jacob C. Douma. "Generalized AIC and chi-squared statistics for path models consistent with directed acyclic graphs." Ecology 101.3 (2020): e02960.

See Also

basisSet, dSep

Examples

```
mod <- psem(
lm(rich ~ cover, data = keeley),
lm(cover ~ firesev, data = keeley),
lm(firesev ~ age, data = keeley),
data = keeley
)

LLchisq(mod)
```

`meadows` *Data set from Grace & Jutila (1999)*

Description

Data set from Grace & Jutila (1999)

Usage

```
meadows
```

Format

A `data.frame` with 354 observations of 4 variables.

grazed Whether meadows were exposed to grazing: 0 = no, 1 = yes

mass Plant biomass in g m[-2]

elev Elevation of the plot above mean sea level

rich Plant species richness per m[2]

`multigroup` *Multigroup Analysis for Piecewise SEM*

Description

Multigroup Analysis for Piecewise SEM

Usage

```
multigroup(
  modelList,
  group,
  standardize = "scale",
  standardize.type = "latent.linear",
  test.type = "III"
)
```

Arguments

`modelList`	a list of structural equations
`group`	the name of the grouping variable in quotes
`standardize`	The type of standardization: `none`, `scale`, `range`. Default is `scale`.
`standardize.type`	The type of standardized for non-Gaussian responses: `latent.linear`, `Menard.OE`. Default is `latent.linear`.
`test.type`	what kind of ANOVA should be reported. Default is type III

Author(s)

Jon Lefcheck <lefcheckj@si.edu>

Examples

```
data(meadows)

jutila <- psem(
lm(rich ~ elev + mass, data = meadows),
lm(mass ~ elev, data = meadows)
)

jutila.multigroup <- multigroup(jutila, group = "grazed")

jutila.multigroup
```

`partialResid` *Computing partial effects*

Description

Extracts partial residuals from a model or `psem` object for a given `x` and `y`.

Usage

```
partialResid(formula., modelList, data = NULL)
```

Arguments

`formula.`	A formula where the `lhs` is the response and the `rhs` is the predictor whose partial effect is desired.
`modelList`	A list of structural equations.
`data`	A `data.frame` used to fit the equations.

Details

This function computes the partial residuals of `y ~ x + Z` in a two-step procedure to remove the variation explained by `Z`: (1) remove `x` from the equation and model `y ~ Z`, and (2) replace `y` with `x` and model `x ~ Z`.

Value

Returns a `data.frame` of residuals of `y ~ Z` called `yresids`, of `x ~ Z` called `xresids`.

Author(s)

Jon Lefcheck <lefcheckj@si.edu>

See Also

`cerror`

Examples

```
# Generate data
dat <- data.frame(y = rnorm(100), x1 = rnorm(100), x2 = rnorm(100))

# Build model
model <- lm(y ~ x1 + x2, dat)

# Compute partial residuals of y ~ x1
yresid <- resid(lm(y ~ x2, dat))

xresid <- resid(lm(x1 ~ x2, dat))
```

```
plot(xresid, yresid)

# Use partialResid
presid <- partialResid(y ~ x1, model)

with(presid, plot(xresid, yresid)) # identical plot!
```

`plot.psem` *Plotting of Piecewise Structural Equation Models*

Description

plot.psem uses [DiagrammeR] to generate path diagrams of ‘piecewiseSEM“ fits within R.

Usage

```
## S3 method for class 'psem'
plot(
  x,
  return = FALSE,
 node_attrs = data.frame(shape = "rectangle", color = "black", fillcolor = "white"),
  edge_attrs = data.frame(style = "solid", color = "black"),
  ns_dashed = T,
  alpha = 0.05,
  show = "std",
  digits = 3,
  add_edge_label_spaces = TRUE,
  ...
)
```

Arguments

`x`	a [psem()] object
`return`	whether to return the output from [DiagrammeR::create_graph()] for modification and later plotting
`node_attrs`	List of node attributes to override defaults of rectangular nodes with black outline and white fill. See [here](http://visualizers.co/diagrammer/articles/node-edge-data-frames.html) and [here](http://visualizers.co/diagrammer/articles/graphviz-mermaid.html) for a more complete rundown of options.
`edge_attrs`	List of edge attributes to override defaults of solid black arrows. See [here](http://visualizers.co/diagramm edge-data-frames.html) and [here](http://visualizers.co/diagrammer/articles/graphviz-mermaid.html) for a more complete rundown of options.
`ns_dashed`	If TRUE, paths that are not different from 0 will be dashed rather than solid, unless the whole is overridden in ‘edge_attrs‘

`alpha`	The alpha level for assessing whether a path is different from 0
`show`	What types of path coefficients are shown? Default ‘"std"‘ is standardized coefficients. For undstandardized, use ‘"unstd"‘
`digits`	How many significant digits should be shown?
`add_edge_label_spaces`	Should spaces by added on either side of edge labels? Default is ‘TRUE‘ as otherwise paths too often overlap edges.
`...`	Other arguments to [DiagrammeR::render_graph()]

Value

Returns an object of class [DiagrammeR::dgr_graph]

Author(s)

Jarrett Byrnes <jarrett.byrnes@umb.edu>

Examples

```
data(keeley)

mod <- psem(
  lm(rich ~ cover, data=keeley),
  lm(cover ~ firesev, data=keeley),
  lm(firesev ~ age, data=keeley),
  data = keeley
)

plot(mod)

### More customized plot

plot(mod, node_attrs = list(
  shape = "rectangle", color = "black",
  fillcolor = "orange", x = 3, y=1:4))
```

`print.anova.psem` *Print anova*

Description

Print anova

Usage

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

Arguments

`x`	an object of class anova.psem
`...`	further arguments passed to or from other methods

`print.basisSet` *Print basis set*

Description

Print basis set

Usage

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

Arguments

`x`	a basis set
`...`	further arguments passed to or from other methods

`print.multigroup.psem` *Print multigroup*

Description

Print multigroup

Usage

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

Arguments

`x`	an object to print
`...`	additional arguments to print

`print.psem` *Print psem*

Description

Print psem

Usage

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

Arguments

`x`	an object of class psem
`...`	further arguments passed to or from other methods

`print.summary.psem` *Print summary*

Description

Print summary

Usage

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

Arguments

`x`	an object of class summary.psem
`...`	further arguments passed to or from other methods

`psem` *Fitting piecewise structural equation models*

Description

`psem` is used to unite a list of structural equations into a single structural equation model.

Usage

```
psem(...)
```

Arguments

`...`	A list of structural equations

Details

`psem` takes a list of structural equations, which can be model objects of classes: `lm`, `glm`, `gls`, `pgls`, `Sarlm`, `lme`, `glmmPQL`, `lmerMod`,`lmerModLmerTest`, `glmerMod`, `glmmTMB`, `gam`.

It also takes objects of class `formula`, `formula.cerror`, corresponding to additional variables to be included in the tests of directed separation (`X ~ 1`) or correlated errors (`X1 %~~% X2`).

The function optionally accepts data objects of classes: `matrix`,`data.frame`, `SpatialPointsDataFrame`, `comparative.data`, or these are derived internally from the structural equations.

Value

Returns an object of class `psem`

Author(s)

Jon Lefcheck <LefcheckJ@si.edu>

See Also

`summary.psem`, `%~~%`

Examples

```
mod <- psem(
lm(rich ~ cover, data = keeley),
lm(cover ~ firesev, data = keeley),
lm(firesev ~ age, data = keeley),
data = keeley
)

summary(mod)
```

`residuals.psem` *Residual values from fit models*

Description

Residual values from fit models

Usage

```
## S3 method for class 'psem'
residuals(object, ...)
```

Arguments

`object`	a `psem` object
`...`	additional arguments to residuals

Value

a `data.frame` of residuals for endogenous variables as columns

`rsquared` *R-squared for linear regression*

Description

Returns (pseudo)-R^2 values for all linear, generalized linear, and generalized linear mixed effects models.

Usage

```
rsquared(modelList, method = NULL)
```

Arguments

`modelList`	a regression, or a list of structural equations.
`method`	The method used to compute the R2 value (See Details)

Details

For mixed models, marginal R2 considers only the variance by the fixed effects, and the conditional R2 by both the fixed and random effects.

For generalized additive models fit to gaussian distribution, the function returns ths adjusted-R2. For all other distributions, it returns the proportion of deviance explained.

For GLMs (`glm`), supported methods include:

- `mcfadden` 1 - ratio of likelihoods of full vs. null models
- `coxsnell` McFadden's R2 but raised to 2/N. Upper limit is < 1
- `nagelkerke` Adjusts Cox-Snell R2 so that upper limit = 1. The DEFAULT method

For GLMERs fit to Poisson, Gamma, and negative binomial distributions (`glmer`, `glmmPQL`, `glmer.nb`), supported methods include

- `delta` Approximates the observation variance based on second-order Taylor series expansion. Can be used with many families and link functions
- `lognormal` Observation variance is the variance of the log-normal distribution
- `trigamma` Provides most accurate estimate of the observation variance but is limited to only the log link. The DEFAULT method

For GLMERs fit to the binomial distribution (`glmer`, `glmmPQL`), supported methods include:

- `theoretical` Assumes observation variance is pi^2/3
- `delta` Approximates the observation variance as above. The DEFAULT method

Value

Returns a `data.frame` with the response, its family and link, the method used to estimate R2, and the R2 value itself. Mixed models also return marginal and conditional R2 values.

Author(s)

Jon Lefcheck <lefcheckj@si.edu>

References

Nakagawa, Shinichi, Paul CD Johnson, and Holger Schielzeth. "The coefficient of determination R 2 and intra-class correlation coefficient from generalized linear mixed-effects models revisited and expanded." Journal of the Royal Society Interface 14.134 (2017): 20170213.

Examples

```
## Not run:
  # Create data
  dat <- data.frame(
    ynorm = rnorm(100),
    ypois = rpois(100, 100),
    x1 = rnorm(100),
    random = letters[1:5]
```

```
    )

    # Get R2 for linear model
    rsquared(lm(ynorm ~ x1, dat))

    # Get R2 for generalized linear model
    rsquared(glm(ypois ~ x1, "poisson", dat))

    rsquared(glm(ypois ~ x1, "poisson", dat), method = "mcfadden") # McFadden R2

    # Get R2 for generalized least-squares model
    rsquared(gls(ynorm ~ x1, dat))

    # Get R2 for linear mixed effects model (nlme)
    rsquared(nlme::lme(ynorm ~ x1, random = ~ 1 | random, dat))

    # Get R2 for linear mixed effects model (lme4)
    rsquared(lme4::lmer(ynorm ~ x1 + (1 | random), dat))

    # Get R2 for generalized linear mixed effects model (lme4)
    rsquared(lme4::glmer(ypois ~ x1 + (1 | random), family = poisson, dat))

   rsquared(lme4::glmer(ypois ~ x1 + (1 | random), family = poisson, dat), method = "delta")

    # Get R2 for generalized linear mixed effects model (glmmPQL)
    rsquared(MASS::glmmPQL(ypois ~ x1, random = ~ 1 | random, family = poisson, dat))

    # Get R2 for generalized additive models (gam)
    rsquared(mgcv::gam(ynorm ~ x1, dat))

## End(Not run)
```

`shipley` *Data set from Shipley (2006)*

Description

Data set from Shipley (2006)

Usage

```
shipley
```

Format

A `data.frame` with 1900 observations of 9 variables.

site Site of observation

tree Individual tree of observation

lat	Latitude
year	Year of observation
Date	Julian date of first bud burst
DD	Cumulative degree days until first bud burst
Growth	Increase in stem diameter
Survival	Proportional survival
Live	Alive (1) or dead (0)

`summary.psem` *Summarizing piecewise structural equation models*

Description

Returns information necessary to interpret piecewise structural equation models, including tests of directed separation, path coefficients, information criterion values, and R-squared values of individual models.

Usage

```
## S3 method for class 'psem'
summary(
  object,
  ...,
  basis.set = NULL,
  direction = NULL,
  interactions = FALSE,
  conserve = FALSE,
  conditioning = FALSE,
  add.claims = NULL,
  standardize = "scale",
  standardize.type = "latent.linear",
  test.statistic = "F",
  test.type = "II",
  intercepts = FALSE,
  AIC.type = "loglik",
  .progressBar = TRUE
)
```

Arguments

`object`	a list of structural equations
`...`	additional arguments to summary
`basis.set`	an optional basis set
`direction`	a vector of claims defining the specific directionality of any independence claim(s)

`interactions`	whether interactions should be included in independence claims. Default is FALSE
`conserve`	whether the most conservative P-value should be returned (See Details) Default is FALSE
`conditioning`	whether all conditioning variables should be shown in the table Default is FALSE
`add.claims`	an optional vector of additional independence claims (P-values) to be added to the basis set
`standardize`	whether standardized path coefficients should be reported Default is "scale"
`standardize.type`	the type of standardized for non-Gaussian responses: `latent.linear` (default), `Mendard.OE`
`test.statistic`	the type of test statistic generated by `Anova`
`test.type`	the type of test ("II" or "III") for significance of categorical variables (from `car::Anova`)
`intercepts`	whether intercepts should be included in the coefficient table Default is FALSE
`AIC.type`	whether the log-likelihood `"loglik"` or d-sep `"dsep"` AIC score should be reported. Default is `"loglik"`
`.progressBar`	an optional progress bar. Default is TRUE

Details

The forthcoming argument `groups` splits the analysis based on an optional grouping factor, conducts separate d-sep tests, and reports goodness-of-fit and path coefficients for each submodel. The procedure is approximately similar to a multigroup analysis in traditional variance-covariance SEM. Coming in version 2.1.

In cases involving non-normally distributed responses in the independence claims that are modeled using generalized linear models, the significance of the independence claim is not reversible (e.g., the P-value of Y ~ X is not the same as X ~ Y). This is due to the transformation of the response via the link function. In extreme cases, this can bias the goodness-of-fit tests. `summary.psem` will issue a warning when this case is present and provide guidance for solutions. One solution is to specify the directionality of the relationship using the `direction` argument, e.g. `direction = c("X <- Y")`. Another is to run both tests (Y ~ X, X ~ Y) and return the most conservative (i.e., lowest) P-value, which can be toggled using the `conserve = TRUE` argument.

In some cases, additional claims that were excluded from the basis set can be added back in using the argument `add.claims`. These could be, for instance, independence claims among exogenous variables. See Details in `basisSet`.

Standardized path coefficients are scaled by standard deviations.

Value

The function `summary.psem` returns a list of summary statistics:

`dTable`	A summary table of the tests of directed separation, from `dSep`.
`CStat`	Fisher's C statistic, degrees of freedom, and significance value based on a Chi-square test.

`AIC`	Information criterion (Akaike, corrected Akaike) as well as degrees of freedom and sample size.
`coefficients`	A summary table of the path coefficients, from `link{coefs}`.
`R2`	(Pseudo)-R2 values, from `rsquared`.

Author(s)

Jon Lefcheck <lefcheckj@si.edu>

References

Shipley, Bill. "A new inferential test for path models based on directed acyclic graphs." Structural Equation Modeling 7.2 (2000): 206-218.

Shipley, Bill. Cause and correlation in biology: a user's guide to path analysis, structural equations and causal inference. Cambridge University Press, 2002.

Shipley, Bill. "Confirmatory path analysis in a generalized multilevel context." Ecology 90.2 (2009): 363-368.

Shipley, Bill. "The AIC model selection method applied to path analytic models compared using a d-separation test." Ecology 94.3 (2013): 560-564.

See Also

The model fitting function `psem`.

`update.psem` *Update psem model object with additional values.*

Description

Update psem model object with additional values.

Usage

```
## S3 method for class 'psem'
update(object, ...)
```

Arguments

`object`	a psem object
`...`	additional arguments to update

Examples

```
mod <- psem(
lm(rich ~ cover, data = keeley),
lm(cover ~ firesev, data = keeley),
lm(firesev ~ age, data = keeley),
data = keeley
)

update(mod, firesev ~ age + cover)
```

`%~~%` *Correlated error operator*

Description

Specifies correlated errors among predictors

Usage

```
e1 %~~% e2
```

Arguments

`e1`	first variable involved in correlated error
`e2`	second variable involved in correlated error

Details

For use in `psem` to identify correlated sets of variables.

Author(s)

Jon Lefcheck <LefcheckJ@si.edu>, Jarrett Byrnes

See Also

`cerror`

Examples

```
# Generate example data
dat <- data.frame(x1 = runif(50),
  x2 = runif(50), y1 = runif(50),
    y2 = runif(50))

# Create list of structural equations
sem <- psem(
  lm(y1 ~ x1 + x2, dat),
```

```
  lm(y2 ~ y1 + x1, dat)
)

# Look at correlated error between x1 and x2
# (exogenous)
cerror(x1 %~~% x2, sem, dat)

# Same as cor.test
with(dat, cor.test(x1, x2))

# Look at correlatde error between x1 and y1
# (endogenous)
cerror(y1 %~~% x1, sem, dat)

# Not the same as cor.test
# (accounts for influence of x1 and x2 on y1)
with(dat, cor.test(y1, x1))

# Specify in psem
sem <- update(sem, x1 %~~% y1)

coefs(sem)
```

Index