Package 'sensiPhy'

October 14, 2022

Title Sensitivity Analysis for Comparative Methods

Version 0.8.5

Date 2020-03-31

Maintainer Gustavo Paterno <paternogbc@gmail.com>

Description An implementation of sensitivity analysis for phylogenetic comparative
methods. The package is an umbrella of statistical and graphical methods that
estimate and report different types of uncertainty in PCM:
(i) Species Sampling uncertainty (sample size; influential species and clades).
(ii) Phylogenetic uncertainty (different topologies and/or branch lengths).
(iii) Data uncertainty (intraspecific variation and measurement error).

Depends R (>= 3.4.0), ape (>= 3.3), phylolm (>= 2.4), ggplot2 (>=
2.1.0)

Imports caper (>= 0.5.2), phytools (>= 0.6), geiger (>= 2.0)

License GPL-2

URL https://github.com/paternogbc/sensiPhy

BugReports https://github.com/paternogbc/sensiPhy/issues

LazyData true

RoxygenNote 7.0.2

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Gustavo Paterno [cre, aut],
Gijsbert Werner [aut],
Caterina Penone [aut],
Pablo Martinez [ctb]

Repository CRAN

Date/Publication 2020-04-02 14:50:02 UTC

R **topics documented:**

---

alien   *Alien Mammals dataset: Example dataset for the package sensiPhy*

Description

A comparative dataset containing traits for 94 alien mammal species (alien.data) and a multiphylo object with 101 phylogenies matching the data (alien.phy). Tip labels are the binomial species names and match with data rownames. Data was taken from (Gonzalez-Suarez et al. 2015) and phylogenies from (Fritz et al 2009) and (Kuhn et al 2011).

Usage

```
data(alien)
```

Format

A data frame with 94 rows and 7 variables:

- family: Taxonomic family
- adultMass: Mean adult body mass (g)
- gestaLen: Mean gestation length (days)

- homeRange: Mean home range (km)
- SE_mass: Standard deviation (intraspecific) for mean adult body mass (g)
- SE_gesta: Standard deviation (intraspecific) for mean gestation length (days)
- SE_range: Standard deviation (intraspecific) for mean home range (km)

References

Alien mammal data: Gonzalez-Suarez, Manuela, Sven Bacher, and Jonathan M. Jeschke. "Intraspecific trait variation is correlated with establishment success of alien mammals." The American Naturalist 185.6 (2015): 737-746 DOI: 10.1086/681105

Downloaded from: Gonzalez-Surez M, Bacher S, Jeschke J (2015) Data from: Intraspecific trait variation is correlated with establishment success of alien mammals. Dryad Digital Repository. http://dx.doi.org/10.5061/dryad.sp963

Phylogeny: Kuhn, Tyler S., Arne O. Mooers, and Gavin H. Thomas. "A simple polytomy resolver for dated phylogenies." Methods in Ecology and Evolution 2.5 (2011): 427-436.

Fritz, Susanne A., Olaf RP Bininda-Emonds, and Andy Purvis. "Geographical variation in predictors of mammalian extinction risk: big is bad, but only in the tropics." Ecology letters 12.6 (2009): 538-549.

| alien.data | *Alien Mammals dataset: Example dataset for the package sensiPhy* |

Description

A comparative dataset containing traits for 94 alien mammal species (alien.data) and a multiphylo object with 101 phylogenies matching the data (alien.phy). Tip labels are the binomial species names and match with data rownames. Data was taken from (Gonzalez-Suarez et al. 2015) and phylogenies from (Fritz et al 2009) and (Kuhn et al 2011).

Usage

```
data(alien)
```

Format

A data frame with 94 rows and 7 variables:

- family: Taxonomic family
- adultMass: Mean adult body mass (g)
- gestaLen: Mean gestation length (days)
- homeRange: Mean home range (km)
- SE_mass: Standard deviation (intraspecific) for mean adult body mass (g)
- SE_gesta: Standard deviation (intraspecific) for mean gestation length (days)
- SE_range: Standard deviation (intraspecific) for mean home range (km)

References

Alien mammal data: Gonzalez-Suarez, Manuela, Sven Bacher, and Jonathan M. Jeschke. "Intraspecific trait variation is correlated with establishment success of alien mammals." The American Naturalist 185.6 (2015): 737-746 DOI: 10.1086/681105

Downloaded from: Gonzalez-Surez M, Bacher S, Jeschke J (2015) Data from: Intraspecific trait variation is correlated with establishment success of alien mammals. Dryad Digital Repository. http://dx.doi.org/10.5061/dryad.sp963

Phylogeny: Kuhn, Tyler S., Arne O. Mooers, and Gavin H. Thomas. "A simple polytomy resolver for dated phylogenies." Methods in Ecology and Evolution 2.5 (2011): 427-436.

Fritz, Susanne A., Olaf RP Bininda-Emonds, and Andy Purvis. "Geographical variation in predictors of mammalian extinction risk: big is bad, but only in the tropics." Ecology letters 12.6 (2009): 538-549.

---

alien.phy *Alien Mammals dataset: Example dataset for the package sensiPhy*

Description

A comparative dataset containing traits for 94 alien mammal species (alien.data) and a multiphylo object with 101 phylogenies matching the data (alien.phy). Tip labels are the binomial species names and match with data rownames. Data was taken from (Gonzalez-Suarez et al. 2015) and phylogenies from (Fritz et al 2009) and (Kuhn et al 2011).

Usage

```
data(alien)
```

Format

A data frame with 94 rows and 7 variables:

- family: Taxonomic family
- adultMass: Mean adult body mass (g)
- gestaLen: Mean gestation length (days)
- homeRange: Mean home range (km)
- SE_mass: Standard deviation (intraspecific) for mean adult body mass (g)
- SE_gesta: Standard deviation (intraspecific) for mean gestation length (days)
- SE_range: Standard deviation (intraspecific) for mean home range (km)

References

Alien mammal data: Gonzalez-Suarez, Manuela, Sven Bacher, and Jonathan M. Jeschke. "Intraspecific trait variation is correlated with establishment success of alien mammals." The American Naturalist 185.6 (2015): 737-746 DOI: 10.1086/681105

Downloaded from: Gonzalez-Surez M, Bacher S, Jeschke J (2015) Data from: Intraspecific trait variation is correlated with establishment success of alien mammals. Dryad Digital Repository. http://dx.doi.org/10.5061/dryad.sp963

Phylogeny: Kuhn, Tyler S., Arne O. Mooers, and Gavin H. Thomas. "A simple polytomy resolver for dated phylogenies." Methods in Ecology and Evolution 2.5 (2011): 427-436.

Fritz, Susanne A., Olaf RP Bininda-Emonds, and Andy Purvis. "Geographical variation in predictors of mammalian extinction risk: big is bad, but only in the tropics." Ecology letters 12.6 (2009): 538-549.

---

clade_continuous   *Influential Clade Detection - Trait Evolution Continuous Characters*

Description

Fits models for trait evolution of continuous characters, detecting influential clades

Usage

```
clade_continuous(
  data,
  phy,
  model,
  trait.col,
  clade.col,
  n.species = 5,
  n.sim = 20,
  bounds = list(),
  n.cores = NULL,
  track = TRUE,
  ...
)
```

Arguments

data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
model	The evolutionary model (see Details).
trait.col	The column in the provided data frame which specifies the trait to analyse (which should be a factor with two level)
clade.col	The column in the provided data frame which specifies the clades (a character vector with clade names).

n.species	Minimum number of species in a clade for the clade to be included in the leave-one-out deletion analysis. Default is 5.
n.sim	Number of simulations for the randomization test.
bounds	settings to constrain parameter estimates. See [fitContinuous]
n.cores	number of cores to use. If 'NULL', number of cores is detected.
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to [fitContinuous]

Details

This function sequentially removes one clade at a time, fits a model of continuous character evolution using [fitContinuous], repeats this many times (controlled by n.sim), stores the results and calculates the effects on model parameters Currently, only binary continuous traits are supported.

Additionally, to account for the influence of the number of species on each clade (clade sample size), this function also estimates a null distribution expected for the number of species in a given clade. This is done by fitting models without the same number of species as in the given clade.The number of simulations to be performed is set by 'n.sim'. To test if the clade influence differs from the null expectation for a clade of that size, a randomization test can be performed using 'summary(x)'.

Different evolutionary models from fitContinuous can be used, i.e. BM,OU, EB, trend, lambda, kappa, delta and drift.

See [fitContinuous] for more details on evolutionary models.

Output can be visualised using sensi_plot.

Value

The function tree_continuous returns a list with the following components:

call: The function call

data: The original full data frame.

full.model.estimates: Parameter estimates (rate of evolution sigsq and where applicable optpar), root state z0, AICc for the full model without deleted clades.

sensi.estimates: Parameter estimates (sigsq and optpar), (percentual) difference in parameter estimate compared to the full model (DIFsigsq, sigsq.perc, DIFoptpar, optpar.perc), AICc and z0 for each repeat with a clade removed.

null.dist: A data frame with estimates for the null distributions for all clades analysed.

errors: Clades where deletion resulted in errors.

clade.col: Which column was used to specify the clades?

optpar: Transformation parameter used (e.g. lambda, kappa etc.)

Author(s)

Gijsbert Werner & Gustavo Paterno

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Yang Z. 2006. Computational Molecular Evolution. Oxford University Press: Oxford.

Harmon Luke J, Jason T Weir, Chad D Brock, Richard E Glor, and Wendell Challenger. 2008. GEIGER: investigating evolutionary radiations. Bioinformatics 24:129-131.

See Also

[fitContinuous](#)

Examples

```
## Not run:
data("primates")
#Model trait evolution accounting for phylogenetic uncertainty
clade_cont<-clade_continuous(data=primates$data,phy = primates$phy[[1]], model="OU",
trait.col = "adultMass",clade.col="family",n.sim=30,n.species=10,n.cores = 2,track=TRUE)
#Print summary statistics
summary(clade_cont)
sensi_plot(clade_cont,graph="all")
sensi_plot(clade_cont,clade="Cercopithecidae",graph = "sigsq")
sensi_plot(clade_cont,clade="Cercopithecidae",graph = "optpar")
#Change the evolutionary model, tree transformation or minimum number of species per clade
clade_cont2<-clade_continuous(data=primates$data,phy = primates$phy[[1]],model="delta",
trait.col = "adultMass",clade.col="family",n.sim=30,n.species=5,n.cores = 2,track=TRUE)
summary(clade_cont2)
sensi_plot(clade_cont2)
clade_cont3<-clade_continuous(data=primates$data,phy = primates$phy[[1]],model="BM",
trait.col = "adultMass",clade.col="family",n.sim=30,n.species=5,n.cores = 2,track=TRUE)
summary(clade_cont3)
sensi_plot(clade_cont3,graph="sigsq")

## End(Not run)
```

---

clade_discrete   *Influential Clade Detection - Trait Evolution Discrete Characters*

Description

Fits models for trait evolution of discrete (binary) characters, detecting influential clades

Usage

```
clade_discrete(
  data,
  phy,
```

```
    model,
    transform = "none",
    trait.col,
    clade.col,
    n.species = 5,
    n.sim = 20,
    bounds = list(),
    n.cores = NULL,
    track = TRUE,
    ...
)
```

Arguments

data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
model	The Mkn model to use (see Details).
transform	The evolutionary model to transform the tree (see Details). Default is none.
trait.col	The column in the provided data frame which specifies the trait to analyse (which should be a factor with two level)
clade.col	The column in the provided data frame which specifies the clades (a character vector with clade names).
n.species	Minimum number of species in a clade for the clade to be included in the leave-one-out deletion analysis. Default is 5.
n.sim	Number of simulations for the randomization test.
bounds	settings to constrain parameter estimates. See [fitDiscrete](#)
n.cores	number of cores to use. If 'NULL', number of cores is detected.
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to [fitDiscrete](#)

Details

This function sequentially removes one clade at a time, fits a model of discrete character evolution using [fitDiscrete](#), repeats this this many times (controlled by n.sim), stores the results and calculates the effects on model parameters. Currently, only binary discrete traits are supported.

Additionally, to account for the influence of the number of species on each clade (clade sample size), this function also estimates a null distribution expected for the number of species in a given clade. This is done by fitting models without the same number of species as in the given clade.The number of simulations to be performed is set by 'n.sim'. To test if the clade influence differs from the null expectation for a clade of that size, a randomization test can be performed using 'summary(x)'.

Different character model from fitDiscrete can be used, including ER (equal-rates), SYM (symmetric), ARD (all-rates-different) and meristic (stepwise fashion).

All transformations to the phylogenetic tree from fitDiscrete can be used, i.e. none, EB, lambda, kappa anddelta.

See [fitDiscrete](#) for more details on character models and tree transformations.

Output can be visualised using sensi_plot.

Value

The function `tree_discrete` returns a list with the following components:

`call`: The function call

`data`: The original full data frame.

`full.model.estimates`: Parameter estimates (transition rates q12 and q21), AICc and the optimised value of the phylogenetic transformation parameter (e.g. `lambda`) for the full model without deleted clades.

`sensi.estimates`: Parameter estimates (transition rates q12 and q21),(percentual) difference in parameter estimate compared to the full model (DIFq12, sigsq.q12, DIFq21, optpar.q21), AICc and the optimised value of the phylogenetic transformation parameter (e.g. `lambda`) for each repeat with a clade removed.

`null.dist`: A data frame with estimates for the null distributions for all clades analysed.

`errors`: Clades where deletion resulted in errors.

`clade.col`: Which column was used to specify the clades?

`optpar`: Transformation parameter used (e.g. `lambda`, `kappa` etc.)

Author(s)

Gijsbert Werner & Gustavo Paterno

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Yang Z. 2006. Computational Molecular Evolution. Oxford University Press: Oxford.

Harmon Luke J, Jason T Weir, Chad D Brock, Richard E Glor, and Wendell Challenger. 2008. GEIGER: investigating evolutionary radiations. Bioinformatics 24:129-131.

See Also

fitDiscrete

Examples

```
## Not run:
#Load data:
data("primates")
#Create a binary trait factor
primates$data$adultMass_binary<-ifelse(primates$data$adultMass > 7350, "big", "small")
clade_disc<-clade_discrete(data=primates$data,phy = primates$phy[[1]],model="SYM",
trait.col = "adultMass_binary",clade.col="family",n.sim=30,n.species=10,n.cores = 2)
summary(clade_disc)
sensi_plot(clade_disc)
sensi_plot(clade_disc, clade = "Cebidae", graph = "q12")
#Change the evolutionary model, tree transformation or minimum number of species per clade
clade_disc_2<-clade_discrete(data=primates$data,phy = primates$phy[[1]],
model="ARD",transform="kappa",
```

```
      trait.col = "adultMass_binary",clade.col="family",n.sim=30,
      n.species=8,n.cores = 2)
    summary(clade_disc_2)
    sensi_plot(clade_disc_2)
    sensi_plot(clade_disc_2, graph = "q12")
    sensi_plot(clade_disc_2, graph = "q21")

    ## End(Not run)
```

---

clade_phyglm	*Estimate the impact on model estimates of phylogenetic logistic re-gression after removing clades from the analysis.*

Description

Estimate the impact on model estimates of phylogenetic logistic regression after removing clades from the analysis.

Usage

```
clade_phyglm(
  formula,
  data,
  phy,
  btol = 50,
  track = TRUE,
  clade.col,
  n.species = 5,
  n.sim = 100,
  ...
)
```

Arguments

formula	The model formula
data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
btol	Bound on searching space. For details see phyloglm
track	Print a report tracking function progress (default = TRUE)
clade.col	The column in the provided data frame which specifies the clades (a character vector with clade names).
n.species	Minimum number of species in a clade for the clade to be included in the leave-one-out deletion analysis. Default is 5.
n.sim	Number of simulations for the randomization test.
...	Further arguments to be passed to phyloglm

Details

This function sequentially removes one clade at a time, fits a phylogenetic logistic regression model using [phyloglm](#) and stores the results. The impact of of a specific clade on model estimates is calculated by a comparison between the full model (with all species) and the model without the species belonging to a clade.

To account for the influence of the number of species on each clade (clade sample size), this function also estimates a null distribution expected for the number of species in a given clade. This is done by fitting models without the same number of species as in the given clade. The number of simulations to be performed is set by 'n.sim'. To test if the clade influence differs from the null expectation for a clade of that size, a randomization test can be performed using 'summary(x)'.

Currently, only logistic regression using the "logistic_MPLE"-method from `phyloglm` is implemented.

`clade_phyglm` detects influential clades based on difference in intercept and/or estimate when removing a given clade compared to the full model including all species.

Additionally, to account for the influence of the number of species on each clade (clade sample size), this function also estimates a null distribution expected for the number of species in a given clade. This is done by fitting models without the same number of species in the given clade. The number of simulations to be performed is set by 'n.sim'. To test if the clade influence differs from the null expectation for a clade of that size, a randomization test can be performed using 'summary(x)'.

Currently, this function can only implement simple logistic models (i.e. $trait\ predictor$). In the future we will implement more complex models.

Output can be visualised using `sensi_plot`.

Value

The function `clade_phyglm` returns a list with the following components:

`formula`: The formula

`full.model.estimates`: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. `alpha`) for the full model without deleted species.

`sensi.estimates`: A data frame with all simulation estimates. Each row represents a deleted clade. Columns report the calculated regression intercept (`intercept`), difference between simulation intercept and full model intercept (`DIFintercept`), the percentage of change in intercept compared to the full model (`intercept.perc`) and intercept p-value (`pval.intercept`). All these parameters are also reported for the regression slope (`DIFestimate` etc.). Additionally, model aic value (`AIC`) and the optimised value (`optpar`) of the phylogenetic parameter are reported.

`null.dist`: A data frame with estimates for the null distributions for all clades analysed.

`data`: Original full dataset.

`errors`: Clades where deletion resulted in errors.

`clade.col`: Which column was used to specify the clades?

Author(s)

Gustavo Paterno & Gijsbert Werner

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phyloglm, clade_phylm, influ_phyglm, sensi_plot

Examples

```
## Not run:
# Simulate Data:
set.seed(6987)
phy = rtree(150)
x = rTrait(n=1,phy=phy)
X = cbind(rep(1,150),x)
y = rbinTrait(n=1,phy=phy, beta=c(-1,0.5), alpha=.7 ,X=X)
cla <- rep(c("A","B","C","D","E"), each = 30)
dat = data.frame(y, x, cla)
# Run sensitivity analysis:
clade <- clade_phyglm(y ~ x, phy = phy, data = dat, n.sim = 30, clade.col = "cla")
# To check summary results and most influential clades:
summary(clade)
# Visual diagnostics for clade removal:
sensi_plot(clade)
# Specify which clade removal to plot:
sensi_plot(clade, "B")
sensi_plot(clade, "C")
sensi_plot(clade, "D") #The clade with the largest effect on slope and intercept

## End(Not run)
```

---

clade_phylm   *Influential clade detection - Phylogenetic Linear Regression*

Description

Estimate the impact on model estimates of phylogenetic linear regression after removing clades from the analysis.

Usage

```
clade_phylm(
  formula,
  data,
  phy,
```

```
    model = "lambda",
    track = TRUE,
    clade.col,
    n.species = 5,
    n.sim = 100,
    ...
)
```

Arguments

formula	The model formula
data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
model	The phylogenetic model to use (see Details). Default is lambda.
track	Print a report tracking function progress (default = TRUE)
clade.col	The column in the provided data frame which specifies the clades (a character vector with clade names).
n.species	Minimum number of species in a clade for the clade to be included in the leave-one-out deletion analysis. Default is 5.
n.sim	Number of simulations for the randomization test.
...	Further arguments to be passed to phylolm

Details

This function sequentially removes one clade at a time, fits a phylogenetic linear regression model using phylolm and stores the results. The impact of of a specific clade on model estimates is calculated by a comparison between the full model (with all species) and the model without the species belonging to a clade.

Additionally, to account for the influence of the number of species on each clade (clade sample size), this function also estimates a null distribution expected for the number of species in a given clade. This is done by fitting models without the same number of species as in the given clade. The number of simulations to be performed is set by 'n.sim'. To test if the clade influence differs from the null expectation for a clade of that size, a randomization test can be performed using 'summary(x)'.

All phylogenetic models from phylolm can be used, i.e. BM, OUfixedRoot, OUrandomRoot, lambda, kappa, delta, EB and trend. See ?phylolm for details.

clade_phylm detects influential clades based on difference in intercept and/or estimate when removing a given clade compared to the full model including all species.

Currently, this function can only implement simple linear models (i.e. $y = a + bx$). In the future we will implement more complex models.

Output can be visualised using sensi_plot.

Value

The function `clade_phylm` returns a list with the following components:

`formula`: The formula

`full.model.estimates`: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. `lambda`) for the full model without deleted species.

`sensi.estimates`: A data frame with all simulation estimates. Each row represents a deleted clade. Columns report the calculated regression intercept (`intercept`), difference between simulation intercept and full model intercept (`DIFintercept`), the percentage of change in intercept compared to the full model (`intercept.perc`) and intercept p-value (`pval.intercept`). All these parameters are also reported for the regression slope (`DIFestimate` etc.). Additionally, model aic value (`AIC`) and the optimised value (`optpar`) of the phylogenetic parameter (e.g. `kappa` or `lambda`, depending on the phylogenetic model used) are reported.

`null.dist`: A data frame with estimates for the null distributions for all clades analysed.

`data`: Original full dataset.

`errors`: Clades where deletion resulted in errors.

`clade.col`: Which column was used to specify the clades?

Author(s)

Gustavo Paterno

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phylolm, samp_phylm, influ_phylm, sensi_plot, sensi_plot,clade_phyglm,

Examples

```
## Not run:
# Load data:
data(primates)
# run analysis:
clade <- clade_phylm(log(sexMaturity) ~ log(adultMass),
phy = primates$phy[[1]], data = primates$data, n.sim = 30, clade.col = "family")
# To check summary results and most influential clades:
summary(clade)
# Visual diagnostics for clade removal:
sensi_plot(clade)
# Specify which clade removal to plot:
sensi_plot(clade, "Cercopithecidae")
sensi_plot(clade, "Cebidae")
```

```
## End(Not run)
```

---

clade_physig   *Influential clade detection - Phylogenetic signal*

Description

Estimate the influence of clade removal on phylogenetic signal estimates

Usage

```
clade_physig(
  trait.col,
  data,
  phy,
  clade.col,
  n.species = 5,
  n.sim = 100,
  method = "K",
  track = TRUE,
  ...
)
```

Arguments

trait.col	The name of a column in the provided data frame with trait to be analyzed (e.g. "Body_mass").
data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
clade.col	The column in the provided data frame which specifies the clades (a character vector with clade names).
n.species	Minimum number of species in a clade for the clade to be included in the leave-one-out deletion analysis. Default is 5.
n.sim	Number of simulations for the randomization test.
method	Method to compute signal: can be "K" or "lambda".
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to phylosig

Details

This function sequentially removes one clade at a time, estimates phylogenetic signal (K or lambda) using [phylosig](#) and stores the results. The impact of a specific clade on signal estimates is calculated by the comparison between the full data (with all species) and reduced data estimates (without the species belonging to a clade).

To account for the influence of the number of species on each clade (clade sample size), this function also estimate a null distribution of signal estimates expected by the removal of the same number of species as in a given clade. This is done by estimating phylogenetic signal without the same number of species in the given clade. The number of simulations to be performed is set by 'n.sim'. To test if the clade influence differs from the null expectation for a clade of that size, a randomization test can be performed using 'summary(x)'.

Output can be visualised using `sensi_plot`.

Value

The function `clade_physig` returns a list with the following components:

`trait.col`: Column name of the trait analysed

`full.data.estimates`: Phylogenetic signal estimate (K or lambda) and the P value (for the full data).

`sensi.estimates`: A data frame with all simulation estimates. Each row represents a deleted clade. Columns report the calculated phylogenetic signal (K or lambda) (`estimate`), difference between simulation signal and full data signal (`DF`), the percentage of change in signal compared to the full data estimate (`perc`) and p-value of the phylogenetic signal with the reduced data (`pval`).

`null.dist`: A data frame with estimates for the null distribution of phylogenetic signal for all clades analysed.

`data`: Original full dataset.

Note

The argument "se" from [phylosig](#) is not available in this function. Use the argument "V" instead with [intra_physig](#) to indicate the name of the column containing the standard deviation or the standard error of the trait variable instead.

Author(s)

Gustavo Paterno

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Blomberg, S. P., T. Garland Jr., A. R. Ives (2003) Testing for phylogenetic signal in comparative data: Behavioral traits are more labile. Evolution, 57, 717-745.

Pagel, M. (1999) Inferring the historical patterns of biological evolution. Nature, 401, 877-884.

Kamilar, J. M., & Cooper, N. (2013). Phylogenetic signal in primate behaviour, ecology and life history. Philosophical Transactions of the Royal Society B: Biological Sciences, 368: 20120341.

See Also

phylosig, clade_phylm,sensi_plot

Examples

```
data(alien)
alien.data<-alien$data
alien.phy<-alien$phy
# Logtransform data
alien.data$logMass <- log(alien.data$adultMass)
# Run sensitivity analysis:
clade <- clade_physig(trait.col = "logMass", data = alien.data, n.sim = 20,
                phy = alien.phy[[1]], clade.col = "family", method = "K")
summary(clade)
sensi_plot(clade, "Bovidae")
sensi_plot(clade, "Sciuridae")
```

---

influ_continuous   *Influential Species Detection - Trait Evolution Continuous Characters*

Description

Fits models for trait evolution of continuous characters, detecting influential species.

Usage

```
influ_continuous(
  data,
  phy,
  model,
  bounds = list(),
  cutoff = 2,
  n.cores = NULL,
  track = TRUE,
  ...
)
```

Arguments

data	Data vector for a single continuous trait, with names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
model	The evolutionary model (see Details).
bounds	settings to constrain parameter estimates. See fitContinuous
cutoff	The cut-off parameter for influential species (see Details).
n.cores	number of cores to use. If 'NULL', number of cores is detected.
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to fitContinuous

Details

This function sequentially removes one species at a time, fits different models of continuous character evolution using `fitContinuous`, stores the results and calculates the effects on model parameters.

`influ_continuous` detects influential species based on the standardised difference in the rate parameter `sigsq` and the optimisation parameter `optpar` (e.g. lamda, kappa, alpha, depending on which `model` is set), when removing a given species compared to the full model including all species. Species with a standardised difference above the value of `cutoff` are identified as influential.

Different evolutionary models from `fitContinuous` can be used, i.e. BM,OU, EB, trend, lambda, kappa, delta and drift.

See `fitContinuous` for more details on evolutionary models.

Value

The function `tree_discrete` returns a list with the following components:

`call`: The function call

`cutoff`: The value selected for `cutoff`

`data`: The original full data vector

`optpar`: Transformation parameter used (e.g. `lambda`, `kappa` etc.)

`full.model.estimates`: Parameter estimates (rate of evolution `sigsq` and where applicable `optpar`), root state `z0`, AICc for the full model without deleted species.

`influential_species`: List of influential species, based on standardised difference in estimates for sigsq and optpar. Species are ordered from most influential to less influential and only include species with a standardised difference > `cutoff`.

`sensi.estimates`: Parameter estimates (sigsq and optpar),(percentual) difference in parameter estimate compared to the full model (DIFsigsq, sigsq.perc,sDIFsigsq, DIFoptpar, optpar.perc,sDIFoptpar), AICc and z0 for each repeat with a species removed.

Author(s)

Gijsbert Werner & Gustavo Paterno

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467.

Yang Z. 2006. Computational Molecular Evolution. Oxford University Press: Oxford.

Harmon Luke J, Jason T Weir, Chad D Brock, Richard E Glor, and Wendell Challenger. 2008. GEIGER: investigating evolutionary radiations. Bioinformatics 24:129-131.

See Also

`fitContinuous`

Examples

```
## Not run:
#Load data:
data("primates")
#Model trait evolution accounting for influential species
adultMass<-primates$data$adultMass
names(adultMass)<-rownames(primates$data)
influ_cont<-influ_continuous(data = adultMass,phy = primates$phy[[1]],
model = "OU",cutoff = 2,n.cores = 2,track = TRUE)
#Print summary statistics
summary(influ_cont)
sensi_plot(influ_cont)
sensi_plot(influ_cont,graphs="sigsq")
#' sensi_plot(influ_cont,graphs="optpar")
#Use a different evolutionary model or cutoff
influ_cont2<-influ_continuous(data = adultMass,phy = primates$phy[[1]],
model = "lambda",cutoff = 1.2,n.cores = 2,track = TRUE)
summary(influ_cont2)
sensi_plot(influ_cont2)
influ_cont3<-influ_continuous(data = adultMass,phy = primates$phy[[1]],
model = "BM",cutoff = 2,n.cores = 2,track = TRUE)
summary(influ_cont3)

## End(Not run)
```

---

influ_discrete   *Influential Species Detection - Trait Evolution Discrete Characters*

Description

Fits models for trait evolution of discrete (binary) characters, detecting influential species.

Usage

```
influ_discrete(
  data,
  phy,
  model,
  transform = "none",
  bounds = list(),
  cutoff = 2,
  n.cores = NULL,
  track = TRUE,
  ...
)
```

Arguments

data	Data vector for a single binary trait, with names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
model	The Mkn model to use (see Details).
transform	The evolutionary model to transform the tree (see Details). Default is none.
bounds	settings to constrain parameter estimates. See [fitDiscrete](#)
cutoff	The cut-off parameter for influential species (see Details).
n.cores	number of cores to use. If 'NULL', number of cores is detected.
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to [fitDiscrete](#)

Details

This function sequentially removes one species at a time, fits a model of discrete character evolution using [fitDiscrete](#), stores the results and calculates the effects on model parameters. Currently, only binary discrete traits are supported.

influ_discrete detects influential species based on the standardised difference in q12 or q21 when removing a given species compared to the full model including all species. Species with a standardised difference above the value of cutoff are identified as influential.

Different character model from fitDiscrete can be used, including ER (equal-rates), SYM (symmetric), ARD (all-rates-different) and meristic (stepwise fashion).

Different transformations to the phylogenetic tree from fitDiscrete can be used, i.e. none, EB, lambda, kappa anddelta.

See [fitDiscrete](#) for more details on character models and tree transformations.

Value

The function tree_discrete returns a list with the following components:

call: The function call

cutoff: The value selected for cutoff

data: The original full data vector

optpar: Transformation parameter used (e.g. lambda, kappa etc.)

full.model.estimates: Parameter estimates (transition rates q12 and q21), AICc and the optimised value of the phylogenetic transformation parameter (e.g. lambda) for the full model.

influential_species: List of influential species, based on standardised difference in estimates for q12 and q21. Species are ordered from most influential to less influential and only include species with a standardised difference > cutoff.

sensi.estimates: Parameter estimates (transition rates q12 and q21),,(percentual) difference in parameter estimate compared to the full model (DIFq12, sigsq.q12,sDIFq12, DIFq21, optpar.q21,sDIFq21), AICc and the optimised value of the phylogenetic transformation parameter (e.g. lambda) for each analysis with a species deleted.

Author(s)

Gijsbert Werner & Gustavo Paterno

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467.

Yang Z. 2006. Computational Molecular Evolution. Oxford University Press: Oxford.

Harmon Luke J, Jason T Weir, Chad D Brock, Richard E Glor, and Wendell Challenger. 2008. GEIGER: investigating evolutionary radiations. Bioinformatics 24:129-131.

See Also

fitDiscrete

Examples

```
## Not run:
#Load data:
data("primates")
#Create a binary trait factor
adultMass_binary<-ifelse(primates$data$adultMass > 7350, "big", "small")
adultMass_binary<-as.factor(as.factor(adultMass_binary))
names(adultMass_binary)<-rownames(primates$data)
#Model trait evolution accounting for influential species
influ_binary<-influ_discrete(data = adultMass_binary,phy = primates$phy[[1]],
model = "SYM",transform = "none",cutoff = 2,n.cores = 2,track = TRUE)
#Print summary statistics
summary(influ_binary)
sensi_plot(influ_binary) #q12 and q21 are, as expected, exactly the same in symmetrical model.
#Use a different evolutionary model.
influ_binary2<-influ_discrete(data = adultMass_binary,phy = primates$phy[[1]],
model = "SYM",transform = "delta",n.cores = 2,track = TRUE)
summary(influ_binary2)
sensi_plot(influ_binary2)
#Or change the cutoff and transformation
influ_binary3<-influ_discrete(data = adultMass_binary,phy = primates$phy[[1]],
model = "ARD",transform = "none",cutoff = 1.2,n.cores = 2,track = TRUE)
summary(influ_binary3)
sensi_plot(influ_binary3)

## End(Not run)
```

---

influ_phyglm   *Influential species detection - Phylogenetic Logistic Regression*

Description

Performs leave-one-out deletion analysis for phylogenetic logistic regression, and detects influential species.

Usage

```
influ_phyglm(formula, data, phy, btol = 50, cutoff = 2, track = TRUE, ...)
```

Arguments

formula	The model formula
data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
btol	Bound on searching space. For details see phyloglm
cutoff	The cutoff value used to identify for influential species (see Details)
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to phyloglm

Details

This function sequentially removes one species at a time, fits a phylogenetic logistic regression model using [phyloglm](), stores the results and detects influential species.

Currently only logistic regression using the "logistic_MPLE"-method from phyloglm is implemented.

influ_phyglm detects influential species based on the standardised difference in intercept and/or slope when removing a given species compared to the full model including all species. Species with a standardised difference above the value of cutoff are identified as influential. The default value for the cutoff is 2 standardised differences change.

Currently, this function can only implement simple logistic models (i.e. $trait\ predictor$). In the future we will implement more complex models.

Output can be visualised using sensi_plot.

Value

The function influ_phyglm returns a list with the following components:

cutoff: The value selected for cutoff

formula: The formula

full.model.estimates: Coefficients, aic and the optimised value of the phylogenetic parameter (i.e. alpha) for the full model without deleted species.

influential_species: List of influential species, both based on standardised difference in intercept and in the slope of the regression. Species are ordered from most influential to less influential and only include species with a standardised difference > cutoff.

sensi.estimates: A data frame with all simulation estimates. Each row represents a deleted clade. Columns report the calculated regression intercept (intercept), difference between simulation intercept and full model intercept (DIFintercept), the standardised difference (sDIFintercept),

the percentage of change in intercept compared to the full model (`intercept.perc`) and intercept p-value (`pval.intercept`). All these parameters are also reported for the regression slope (`DIFestimate` etc.). Additionally, model aic value (`AIC`) and the optimised value (`optpar`) of the phylogenetic parameter (i.e. `alpha`) are reported.

`data`: Original full dataset.

`errors`: Species where deletion resulted in errors.

Author(s)

Gustavo Paterno & Gijsbert D.A. Werner

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467.

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phyloglm, samp_phyglm, influ_phylm, sensi_plot

Examples

```
# Simulate Data:
set.seed(6987)
phy = rtree(100)
x = rTrait(n=1,phy=phy)
X = cbind(rep(1,100),x)
y = rbinTrait(n=1,phy=phy, beta=c(-1,0.5), alpha=.7 ,X=X)
dat = data.frame(y, x)
# Run sensitivity analysis:
influ <- influ_phyglm(y ~ x, data = dat, phy = phy)
# To check summary results and most influential species:
summary(influ)
# Visual diagnostics for clade removal:
sensi_plot(influ)
```

---

influ_phylm   *Influential species detection - Phylogenetic Linear Regression*

Description

Performs leave-one-out deletion analysis for phylogenetic linear regression, and detects influential species.

Usage

```
influ_phylm(
  formula,
  data,
  phy,
  model = "lambda",
  cutoff = 2,
  track = TRUE,
  ...
)
```

Arguments

formula	The model formula
data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
model	The phylogenetic model to use (see Details). Default is lambda.
cutoff	The cutoff value used to identify for influential species (see Details)
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to phylolm

Details

This function sequentially removes one species at a time, fits a phylogenetic linear regression model using phylolm, stores the results and detects influential species.

All phylogenetic models from phylolm can be used, i.e. BM, OUfixedRoot, OUrandomRoot, lambda, kappa, delta, EB and trend. See ?phylolm for details.

influ_phylm detects influential species based on the standardised difference in intercept and/or slope when removing a given species compared to the full model including all species. Species with a standardised difference above the value of cutoff are identified as influential. The default value for the cutoff is 2 standardised differences change.

Currently, this function can only implement simple linear models (i.e. $trait\ predictor$). In the future we will implement more complex models.

Output can be visualised using sensi_plot.

Value

The function influ_phylm returns a list with the following components:

cutoff: The value selected for cutoff

formula: The formula

full.model.estimates: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. lambda) for the full model without deleted species.

influential_species: List of influential species, both based on standardised difference in intercept and in the slope of the regression. Species are ordered from most influential to less influential and only include species with a standardised difference > cutoff.

sensi.estimates: A data frame with all simulation estimates. Each row represents a deleted clade. Columns report the calculated regression intercept (intercept), difference between simulation intercept and full model intercept (DIFintercept), the standardised difference (sDIFintercept), the percentage of change in intercept compared to the full model (intercept.perc) and intercept p-value (pval.intercept). All these parameters are also reported for the regression slope (DIFestimate etc.). Additionally, model aic value (AIC) and the optimised value (optpar) of the phylogenetic parameter (e.g. kappa or lambda, depending on the phylogenetic model used) are reported.

data: Original full dataset.

errors: Species where deletion resulted in errors.

Author(s)

Gustavo Paterno & Gijsbert D.A. Werner

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phylolm, samp_phylm, influ_phyglm, sensi_plot

Examples

```
# Load data:
data(alien)
# run analysis:
influ <- influ_phylm(log(gestaLen) ~ log(adultMass), phy = alien$phy[[1]],
data = alien$data)
# To check summary results:
summary(influ)
# Most influential speciesL
influ$influential.species
# Visual diagnostics
sensi_plot(influ)
# You can specify which graph and parameter ("estimate" or "intercept") to print:
sensi_plot(influ, param = "estimate", graphs = 2)
```

---

influ_physig   *Influential species detection - Phylogenetic signal*

Description

Performs leave-one-out deletion analysis for phylogenetic signal estimates, and detects influential species for K or lambda.

Usage

```
influ_physig(trait.col, data, phy, method = "K", cutoff = 2, track = TRUE, ...)
```

Arguments

trait.col	The name of a column in the provided data frame with trait to be analyzed (e.g. "Body_mass").
data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
method	Method to compute signal: can be "K" or "lambda".
cutoff	The cutoff value used to identify for influential species (see Details)
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to phylosig

Details

This function sequentially removes one species at a time, ans estimates phylogenetic signal (K or lambda) using phylosig, stores the results and detects the most influential species.

influ_physig detects influential species based on the standardised difference in signal estimate (K or lambda) when removing a given species compared to the full data estimate (with all species). Species with a standardised difference above the value of cutoff are identified as influential. The default value for the cutoff is 2 standardised differences in signal estimate.

Output can be visualised using sensi_plot.

Value

The function influ_physig returns a list with the following components:

cutoff: The value selected for cutoff

trait.col: Column name of the trait analysed

full.data.estimates: Phylogenetic signal estimate (K or lambda) and the P value (for the full data).

influential_species: List of influential species, based on standardised difference in K or lambda. Species are ordered from most influential to less influential and only include species with a standardised difference > cutoff.

`influ.physig.estimates`: A data frame with all simulation estimates. Each row represents a deleted species Columns report the calculated signal estimate (k) or (`lambda`), difference between signal estimation of the reduced and full data (`DF`), the percentage of change in signal compared to the full data signal (`perc`) and p-value for the phylogenetic signal test (`pval`)

`data`: Original full dataset.

Note

The argument "se" from [phylosig](#) is not available in this function. Use the argument "V" instead with [intra_physig](#) to indicate the name of the column containing the standard deviation or the standard error of the trait variable instead.

Author(s)

Gustavo Paterno

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Blomberg, S. P., T. Garland Jr., A. R. Ives (2003) Testing for phylogenetic signal in comparative data: Behavioral traits are more labile. Evolution, 57, 717-745.

Pagel, M. (1999) Inferring the historical patterns of biological evolution. Nature, 401, 877-884.

Kamilar, J. M., & Cooper, N. (2013). Phylogenetic signal in primate behaviour, ecology and life history. Philosophical Transactions of the Royal Society B: Biological Sciences, 368: 20120341.

See Also

[phylosig](#), [influ_phylm](#),[sensi_plot](#)

Examples

```
## Not run:
# Load data:
data(alien)
# Logtransform data
alien.data$logMass <- log(alien.data$adultMass)
# Run sensitivity analysis:
influ <- influ_physig("logMass", data = alien.data, phy = alien.phy[[1]])
# To check summary results:
summary(influ)
# Most influential speciesL
influ$influential.species
# Visual diagnostics
sensi_plot(influ)
# You can specify which graph to print:
sensi_plot(influ, graphs = 1)
sensi_plot(influ, graphs = 2)
```

```
## End(Not run)
```

| intra_clade_phyglm | *Interaction between intraspecific variability and influential clade detection - Phylogenetic Logistic Regression* |

Description

Estimate the impact on model estimates of phylogenetic logistic regression after removing clades from the analysis, while taking into account potential interactions with intraspecific variability.

Usage

```
intra_clade_phyglm(
  formula,
  data,
  phy,
  clade.col,
  n.species = 5,
  n.sim = 100,
  n.intra = 2,
  Vx = NULL,
  distrib = "normal",
  x.transf = NULL,
  btol = 50,
  track = TRUE,
  ...
)
```

Arguments

formula	The model formula
data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
clade.col	The column in the provided data frame which specifies the clades (a character vector with clade names).
n.species	Minimum number of species in a clade for the clade to be included in the leave-one-out deletion analysis. Default is 5.
n.sim	Number of simulations for the randomization test.
n.intra	Number of datasets resimulated taking into account intraspecific variation (see: "intra_phyglm")
Vx	Name of the column containing the standard deviation or the standard error of the predictor variable. When information is not available for one taxon, the value can be 0 or NA

distrib	A character string indicating which distribution to use to generate a random value for the response and/or predictor variables. Default is normal distribution: "normal" (function [rnorm](#)). Uniform distribution: "uniform" ([runif](#)) Warning: we recommend to use normal distribution with Vx or Vy = standard deviation of the mean.
x.transf	Transformation for the predictor variable (e.g. "log" or "sqrt"). Please use this argument instead of transforming data in the formula directly (see also details below).
btol	Bound on searching space. For details see phyloglm
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to phyloglm

Details

This function sequentially removes one clade at a time, fits a phylogenetic logistic regression model using [phyloglm](#) and stores the results. The impact of of a specific clade on model estimates is calculated by the comparison between the full model (with all species) and the model without the species belonging to a clade. This operation is repeated n.intra times for simulated values of the dataset, taking into account intraspecific variation. At each iteration, the function generates a random value for each row in the dataset using the standard deviation or errors supplied, and detect the influential species within that iteration.

Additionally, to account for the influence of the number of species on each clade (clade sample size), this function also estimate a null distribution expected for the number of species in a given clade. This is done by fitting models without the same number of species in the given clade. The number of simulations to be performed is set by 'n.sim'. To test if the clade influence differs from the null expectation for a clade of that size, a randomization test can be performed using 'summary(x)'.

All phylogenetic models from phyloglm can be used, i.e. BM, OUfixedRoot, OUrandomRoot, lambda, kappa, delta, EB and trend. See ?phyloglm for details.

clade_phyglm detects influential clades based on difference in intercept and/or slope when removing a given clade compared to the full model including all species.

Currently, this function can only implement simple logistic models (i.e. $y = a + bx$). In the future we will implement more complex models.

Output can be visualised using sensi_plot.

Value

The function intra_clade_phyglm returns a list with the following components:

formula: The formula

full.model.estimates: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. lambda) for the full model without deleted species.

sensi.estimates: A data frame with all simulation estimates. Each row represents a deleted clade. Columns report the calculated regression intercept (intercept), difference between simulation intercept and full model intercept (DIFintercept), the percentage of change in intercept compared to the full model (intercept.perc) and intercept p-value (pval.intercept). All these parameters are also reported for the regression slope (DIFestimate etc.). Additionally, model aic value (AIC)

and the optimised value (`optpar`) of the phylogenetic parameter (e.g. `kappa` or `lambda`, depending on the phylogenetic model used) are reported.

`null.dist`: A data frame with estimates for the null distributions for all clades analysed.

`data`: Original full dataset.

`errors`: Clades and/or iterations where deletion resulted in errors.

Author(s)

Gustavo Paterno, Caterina Penone

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

`phyloglm`, `intra_phyglm`, `clade_phyglm`, `intra_clade_phylm`, `sensi_plot`

Examples

```
## Not run:
set.seed(6987)
phy = rtree(100)
x = rTrait(n=1,phy=phy,parameters=list(ancestral.state=2,optimal.value=2,sigma2=1,alpha=1))
X = cbind(rep(1,100),x)
y = rbinTrait(n=1,phy=phy, beta=c(-1,0.5), alpha=.7 ,X=X)
z = rnorm(n = length(x),mean = mean(x),sd = 0.1*mean(x))
cla <- rep(c("A","B","C","D"), each = 25)
dat = data.frame(y, x, z, cla)
intra_clade <- intra_clade_phyglm(formula=y ~ x, data = dat, phy = phy,
                                  clade.col = "cla", n.sim = 30, n.intra = 3,
                                  x.transf = log, Vx = "z", distrib="normal")
sensi_plot(intra_clade)
sensi_plot(intra_clade, clade = "B", graphs = 2)

## End(Not run)
```

---

intra_clade_phylm	*Interaction between intraspecific variability and influential clade detection - Phylogenetic Linear Regression*

Description

Estimate the impact on model estimates of phylogenetic linear regression after removing clades from the analysis, while taking into account potential interactions with intraspecific variability.

Usage

```
intra_clade_phylm(
  formula,
  data,
  phy,
  clade.col,
  n.species = 5,
  n.sim = 100,
  n.intra = 2,
  Vy = NULL,
  Vx = NULL,
  distrib = "normal",
  y.transf = NULL,
  x.transf = NULL,
  model = "lambda",
  track = TRUE,
  ...
)
```

Arguments

formula	The model formula
data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
clade.col	The column in the provided data frame which specifies the clades (a character vector with clade names).
n.species	Minimum number of species in a clade for the clade to be included in the leave-one-out deletion analysis. Default is 5.
n.sim	Number of simulations for the randomization test.
n.intra	Number of datasets resimulated taking into account intraspecific variation (see: "intra_phylm")
Vy	Name of the column containing the standard deviation or the standard error of the response variable. When information is not available for one taxon, the value can be 0 or NA.
Vx	Name of the column containing the standard deviation or the standard error of the predictor variable. When information is not available for one taxon, the value can be 0 or NA
distrib	A character string indicating which distribution to use to generate a random value for the response and/or predictor variables. Default is normal distribution: "normal" (function [rnorm](#)). Uniform distribution: "uniform" ([runif](#)) Warning: we recommend to use normal distribution with Vx or Vy = standard deviation of the mean.
y.transf	Transformation for the response variable (e.g. "log" or "sqrt"). Please use this argument instead of transforming data in the formula directly (see also details below).

x.transf	Transformation for the predictor variable (e.g. "log" or "sqrt"). Please use this argument instead of transforming data in the formula directly (see also details below).
model	The phylogenetic model to use (see Details). Default is lambda.
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to phylolm

Details

This function sequentially removes one clade at a time, fits a phylogenetic linear regression model using [phylolm](#) and stores the results. The impact of of a specific clade on model estimates is calculated by the comparison between the full model (with all species) and the model without the species belonging to a clade. This operation is repeated n.intra times for simulated values of the dataset, taking into account intraspecific variation. At each iteration, the function generates a random value for each row in the dataset using the standard deviation or errors supplied, and detect the influential species within that iteration.

Additionally, to account for the influence of the number of species on each clade (clade sample size), this function also estimates a null distribution expected for the number of species in a given clade. This is done by fitting models without the same number of species in the given clade. The number of simulations to be performed is set by 'n.sim'. To test if the clade influence differs from the null expectation for a clade of that size, a randomization test can be performed using 'summary(x)'.

All phylogenetic models from phylolm can be used, i.e. BM, OUfixedRoot, OUrandomRoot, lambda, kappa, delta, EB and trend. See ?phylolm for details.

clade_phylm detects influential clades based on difference in intercept and/or slope when removing a given clade compared to the full model including all species.

Currently, this function can only implement simple linear models (i.e. $y = a + bx$). In the future we will implement more complex models.

Output can be visualised using sensi_plot.

Value

The function intra_clade_phylm returns a list with the following components:

formula: The formula

full.model.estimates: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. lambda) for the full model without deleted species.

sensi.estimates: A data frame with all simulation estimates. Each row represents a deleted clade. Columns report the calculated regression intercept (intercept), difference between simulation intercept and full model intercept (DIFintercept), the percentage of change in intercept compared to the full model (intercept.perc) and intercept p-value (pval.intercept). All these parameters are also reported for the regression slope (DIFestimate etc.). Additionally, model aic value (AIC) and the optimised value (optpar) of the phylogenetic parameter (e.g. kappa or lambda, depending on the phylogenetic model used) are reported.

null.dist: A data frame with estimates for the null distributions for all clades analysed.

data: Original full dataset.

errors: Clades and/or iterations where deletion resulted in errors.

Author(s)

Gustavo Paterno, Caterina Penone

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467.

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phylolm, intra_phylm, clade_phylm, intra_clade_phyglm, sensi_plot

Examples

```
## Not run:
#load data
data(alien)
intra_clade <- intra_clade_phylm(gestaLen ~ adultMass, phy = alien$phy[[1]],
 data = alien$data, clade.col = "family", n.sim = 30, n.intra = 5,
 y.transf = log, x.transf = log, Vy="SD_gesta")
summary(intra_clade)
sensi_plot(intra_clade)
sensi_plot(intra_clade, clade = "Bovidae", graphs = 2)
sensi_plot(intra_clade, clade = "Mustelidae", graphs = 2)

## End(Not run)
```

---

intra_influ_phyglm	*Interaction between intraspecific variability and influential species - Phylogenetic Logistic Regression*

Description

Performs leave-one-out deletion analysis for phylogenetic logistic regression, and detects influential species, while taking into account potential interactions with intraspecific variability.

Usage

```
intra_influ_phyglm(
  formula,
  data,
  phy,
  Vx = NULL,
  n.intra = 30,
  x.transf = NULL,
```

```
  distrib = "normal",
  cutoff = 2,
  btol = 50,
  track = TRUE,
  ...
)
```

Arguments

formula	The model formula: `response~predictor`.
data	Data frame containing species traits with row names matching tips in `phy`.
phy	A phylogeny (class 'phylo') matching `data`.
Vx	Name of the column containing the standard deviation or the standard error of the predictor variable. When information is not available for one taxon, the value can be 0 or `NA`
n.intra	Number of datasets resimulated taking into account intraspecific variation (see: "intra_phylgm")
x.transf	Transformation for the predictor variable (e.g. "log" or "sqrt"). Please use this argument instead of transforming data in the formula directly (see also details below).
distrib	A character string indicating which distribution to use to generate a random value for the response and/or predictor variables. Default is normal distribution: "normal" (function [rnorm](#)). Uniform distribution: "uniform" ([runif](#)) Warning: we recommend to use normal distribution with Vx = standard deviation of the mean.
cutoff	The cutoff value used to identify for influential species (see Details)
btol	Bound on searching space. For details see phyloglm
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to phylolm

Details

This function fits a phylogenetic linear regression model using [phylolm](#), and detects influential species by sequentially deleting one at a time. The regression is repeated `n.intra` times for simulated values of the dataset, taking into account intraspecific variation. At each iteration, the function generates a random value for each row in the dataset using the standard deviation or errors supplied, and detect the influential species within that iteration.

`influ_phylm` detects influential species based on the standardised difference in intercept and/or slope when removing a given species compared to the full model including all species. Species with a standardised difference above the value of cutoff are identified as influential. The default value for the cutoff is 2 standardised differences change.

Currently, this function can only implement simple models (i.e. $trait\ predictor$). In the future we will implement more complex models.

Output can be visualised using `sensi_plot`.

Value

The function `intra_influ_phylm` returns a list with the following components:

`cutoff`: The value selected for `cutoff`

`formula`: The formula

`full.model.estimates`: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. `lambda`) for the full model without deleted species.

`influential_species`: List of influential species, both based on standardised difference in intercept and in the slope of the regression. Species are ordered from most influential to less influential and only include species with a standardised difference > `cutoff`.

`sensi.estimates`: A data frame with all simulation estimates. Each row represents a deleted clade for an iteration of resimulated data. Columns report the calculated regression intercept (`intercept`), difference between simulation intercept and full model intercept (`DIFintercept`), the standardised difference (`sDIFintercept`), the percentage of change in intercept compared to the full model (`intercept.perc`) and intercept p-value (`pval.intercept`). All these parameters are also reported for the regression slope (`DIFestimate` etc.). Additionally, model aic value (`AIC`) and the optimised value (`optpar`) of the phylogenetic parameter (e.g. `kappa` or `lambda`, depending on the phylogenetic model used) are reported.

`data`: Original full dataset.

`errors`: Species where deletion resulted in errors.

Warning

When Vx exceeds X negative (or null) values can be generated, this might cause problems for data transformation (e.g. log-transformation). In these cases, the function will skip the simulation. This problem can be solved by increasing `n.intra`, changing the transformation type and/or checking the target species in output$sp.pb.

Setting `n.intra` at high values can take a long time to execute, since the total number of iterations equals `n.intra * nrow(data)`.

Author(s)

Gustavo Paterno, Caterina Penone & Gijsbert D.A. Werner

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467.

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

`phylolm`, `intra_phyglm`, `influ_phyglm`,`intra_influ_phylm`,`sensi_plot`.

Examples

```
## Not run:
#Generate data
set.seed(6987)
phy = rtree(100)
x = rTrait(n=1,phy=phy,parameters=list(ancestral.state=2,optimal.value=2,sigma2=1,alpha=1))
X = cbind(rep(1,100),x)
y = rbinTrait(n=1,phy=phy, beta=c(-1,0.5), alpha=.7 ,X=X)
z = rnorm(n = length(x),mean = mean(x),sd = 0.1*mean(x))
dat = data.frame(y, x, z)
# Run sensitivity analysis:
intra_influ <- intra_influ_phyglm(formula = y ~ x, data = dat, phy = phy,
                      Vx = "z", n.intra = 5,track = TRUE,distrib="normal",x.transf=NULL)
# To check summary results and most influential species:
summary(intra_influ)
# Visual diagnostics for clade removal:
sensi_plot(intra_influ)

## End(Not run)
```

---

intra_influ_phylm	*Interaction between intraspecific variability and influential species - Phylogenetic Linear Regression*

Description

Performs leave-one-out deletion analysis for phylogenetic linear regression, and detects influential species, while taking into account potential interactions with intraspecific variability.

Usage

```
intra_influ_phylm(
  formula,
  data,
  phy,
  Vy = NULL,
  Vx = NULL,
  y.transf = NULL,
  x.transf = NULL,
  n.intra = 10,
  distrib = "normal",
  model = "lambda",
  cutoff = 2,
  track = TRUE,
  ...
)
```

Arguments

formula	The model formula: `response~predictor`.
data	Data frame containing species traits with row names matching tips in `phy`.
phy	A phylogeny (class 'phylo') matching `data`.
Vy	Name of the column containing the standard deviation or the standard error of the response variable. When information is not available for one taxon, the value can be 0 or `NA`.
Vx	Name of the column containing the standard deviation or the standard error of the predictor variable. When information is not available for one taxon, the value can be 0 or `NA`
y.transf	Transformation for the response variable (e.g. `"log"` or `"sqrt"`). Please use this argument instead of transforming data in the formula directly (see also details below).
x.transf	Transformation for the predictor variable (e.g. `"log"` or `"sqrt"`). Please use this argument instead of transforming data in the formula directly (see also details below).
n.intra	Number of datasets resimulated taking into account intraspecific variation (see: `"intra_phylm"`)
distrib	A character string indicating which distribution to use to generate a random value for the response and/or predictor variables. Default is normal distribution: "normal" (function [rnorm](#)). Uniform distribution: "uniform" ([runif](#)) Warning: we recommend to use normal distribution with Vx or Vy = standard deviation of the mean.
model	The phylogenetic model to use (see Details). Default is `lambda`.
cutoff	The cutoff value used to identify for influential species (see Details)
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to `phylolm`

Details

This function fits a phylogenetic linear regression model using [phylolm](#), and detects influential species by sequentially deleting one at a time. The regression is repeated `n.intra` times for simulated values of the dataset, taking into account intraspecific variation. At each iteration, the function generates a random value for each row in the dataset using the standard deviation or errors supplied, and detect the influential species within that iteration.

All phylogenetic models from `phylolm` can be used, i.e. `BM`, `OUfixedRoot`, `OUrandomRoot`, `lambda`, `kappa`, `delta`, `EB` and `trend`. See `?phylolm` for details.

`influ_phylm` detects influential species based on the standardised difference in intercept and/or slope when removing a given species compared to the full model including all species. Species with a standardised difference above the value of `cutoff` are identified as influential. The default value for the cutoff is 2 standardised differences change.

Currently, this function can only implement simple linear models (i.e. $trait\ predictor$). In the future we will implement more complex models.

Output can be visualised using `sensi_plot`.

Value

The function `intra_influ_phylm` returns a list with the following components:

`cutoff`: The value selected for `cutoff`

`formula`: The formula

`full.model.estimates`: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. `lambda`) for the full model without deleted species.

`influential_species`: List of influential species, both based on standardised difference in intercept and in the slope of the regression. Species are ordered from most influential to less influential and only include species with a standardised difference > `cutoff`.

`sensi.estimates`: A data frame with all simulation estimates. Each row represents a deleted clade for an iteration of resimulated data. Columns report the calculated regression intercept (`intercept`), difference between simulation intercept and full model intercept (`DIFintercept`), the standardised difference (`sDIFintercept`), the percentage of change in intercept compared to the full model (`intercept.perc`) and intercept p-value (`pval.intercept`). All these parameters are also reported for the regression slope (`DIFestimate` etc.). Additionally, model aic value (`AIC`) and the optimised value (`optpar`) of the phylogenetic parameter (e.g. `kappa` or `lambda`, depending on the phylogenetic model used) are reported.

`data`: Original full dataset.

`errors`: Species where deletion resulted in errors.

Warning

When Vy or Vx exceed Y or X, respectively, negative (or null) values can be generated, this might cause problems for data transformation (e.g. log-transformation). In these cases, the function will skip the simulation.

Setting `n.intra` at high values can take a long time to execute, since the total number of iterations equals `n.intra * nrow(data)`.

Author(s)

Gustavo Paterno, Caterina Penone & Gijsbert D.A. Werner

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467.

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

`phylolm`, `intra_phylm`, `influ_phylm`, `intra_influ_phyglm`, `sensi_plot`.

Examples

```
## Not run:
# Load data:
data(alien)
# run analysis:
intra_influ <- intra_influ_phylm(formula = gestaLen ~ adultMass, phy = alien$phy[[1]],
data=alien$data,model="lambda",y.transf = log,x.transf = NULL,Vy="SD_gesta",Vx=NULL,
n.intra=30,distrib = "normal")
summary(intra_influ)
sensi_plot(intra_influ)

## End(Not run)
```

---

intra_phyglm   *Intraspecific variability - Phylogenetic Logistic Regression*

Description

Performs Phylogenetic logistic regression evaluating intraspecific variability in predictor variables.

Usage

```
intra_phyglm(
  formula,
  data,
  phy,
  Vx = NULL,
  n.intra = 30,
  x.transf = NULL,
  distrib = "normal",
  btol = 50,
  track = TRUE,
  ...
)
```

Arguments

formula	The model formula: `response~predictor`.
data	Data frame containing species traits with species as row names.
phy	A phylogeny (class 'phylo', see ?ape).
Vx	Name of the column containing the standard deviation or the standard error of the predictor variable. When information is not available for one taxon, the value can be 0 or `NA`
n.intra	Number of times to repeat the analysis generating a random value for the predictor variable. If NULL, `n.intra = 2`

x.transf	Transformation for the predictor variable (e.g. log or sqrt). Please use this argument instead of transforming data in the formula directly (see also details below).
distrib	A character string indicating which distribution to use to generate a random value for the predictor variable. Default is normal distribution: "normal" (function rnorm). Uniform distribution: "uniform" (runif) Warning: we recommend to use normal distribution with Vx = standard deviation of the mean.
btol	Bound on searching space. For details see phyloglm
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to phyloglm

Details

This function fits a phylogenetic logistic regression model using phyloglm. The regression is repeated n.intra times. At each iteration the function generates a random value for each row in the dataset using the standard deviation or error supplied and assuming a normal or uniform distribution. To calculate means and se for your raw data, you can use the summarySE function from the package Rmisc.

All phylogenetic models from phyloglm can be used, i.e. BM, OUfixedRoot, OUrandomRoot, lambda, kappa, delta, EB and trend. See ?phyloglm for details.

Currently, this function can only implement simple logistic models (i.e. $trait \; predictor$). In the future we will implement more complex models.

Output can be visualised using sensi_plot.

Value

The function intra_phyglm returns a list with the following components:

formula: The formula

data: Original full dataset

sensi.estimates: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. lambda) for each regression.

N.obs: Size of the dataset after matching it with tree tips and removing NA's.

stats: Main statistics for model parameters.CI_low and CI_high are the lower and upper limits of the 95

all.stats: Complete statistics for model parameters. sd_intra is the standard deviation due to intraspecific variation. CI_low and CI_high are the lower and upper limits of the 95

sp.pb: Species that caused problems with data transformation (see details above).

Warning

When Vx exceeds X negative (or null) values can be generated, this might cause problems for data transformation (e.g. log-transformation). In these cases, the function will skip the simulation. This problem can be solved by increasing n.intra, changing the transformation type and/or checking the target species in output$sp.pb.

Author(s)

Caterina Penone & Pablo Ariel Martinez

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Martinez, P. a., Zurano, J.P., Amado, T.F., Penone, C., Betancur-R, R., Bidau, C.J. & Jacobina, U.P. (2015). Chromosomal diversity in tropical reef fishes is related to body size and depth range. Molecular Phylogenetics and Evolution, 93, 1-4

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phyloglm, sensi_plot

Examples

```
# Simulate Data:
set.seed(6987)
phy = rtree(150)
x = rTrait(n=1,phy=phy)
x_sd = rnorm(150,mean = 0.8,sd=0.2)
X = cbind(rep(1,150),x)
y = rbinTrait(n=1,phy=phy, beta=c(-1,0.5), alpha=.7 ,X=X)
dat = data.frame(y, x, x_sd)

# Run phylogenetic logistic regression accounting for intraspecific variation:
intra_glm <- intra_phyglm(y~x,Vx = "x_sd",data = dat,phy=phy,distrib = "normal")

#Print summary of sensitivity analysis
summary(intra_glm)
head(intra_glm$sensi.estimates)
#Visual output
sensi_plot(intra_glm)
```

---

intra_phylm   *Intraspecific variability - Phylogenetic Linear Regression*

Description

Performs Phylogenetic linear regression evaluating intraspecific variability in response and/or predictor variables.

Usage

```
intra_phylm(
  formula,
  data,
  phy,
  Vy = NULL,
  Vx = NULL,
  y.transf = NULL,
  x.transf = NULL,
  n.intra = 30,
  distrib = "normal",
  model = "lambda",
  track = TRUE,
  ...
)
```

Arguments

formula	The model formula: `response~predictor`.
data	Data frame containing species traits and species names as row names.
phy	A phylogeny (class 'phylo', see ?ape).
Vy	Name of the column containing the standard deviation or the standard error of the response variable. When information is not available for one taxon, the value can be 0 or `NA`.
Vx	Name of the column containing the standard deviation or the standard error of the predictor variable. When information is not available for one taxon, the value can be 0 or `NA`
y.transf	Transformation for the response variable (e.g. `log` or `sqrt`). Please use this argument instead of transforming data in the formula directly (see also details below).
x.transf	Transformation for the predictor variable (e.g. `log` or `sqrt`). Please use this argument instead of transforming data in the formula directly (see also details below).
n.intra	Number of times to repeat the analysis generating a random value for response and/or predictor variables. If NULL, `n.intra` = 30
distrib	A character string indicating which distribution to use to generate a random value for the response and/or predictor variables. Default is normal distribution: "normal" (function [rnorm](#)). Uniform distribution: "uniform" ([runif](#)) Warning: we recommend to use normal distribution with Vx or Vy = standard deviation of the mean.
model	The phylogenetic model to use (see Details). Default is `lambda`.
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to `phylolm`

Details

This function fits a phylogenetic linear regression model using phylolm. The regression is repeated `n.intra` times. At each iteration the function generates a random value for each row in the dataset using the standard deviation or errors supplied and assuming a normal or uniform distribution. To calculate means and se for your raw data, you can use the `summarySE` function from the package `Rmisc`.

#' All phylogenetic models from `phylolm` can be used, i.e. `BM`, `OUfixedRoot`, `OUrandomRoot`, `lambda`, `kappa`, `delta`, `EB` and `trend`. See ?phylolm for details.

Currently, this function can only implement simple linear models (i.e. $trait \sim predictor$). In the future we will implement more complex models.

Output can be visualised using `sensi_plot`.

Value

The function `intra_phylm` returns a list with the following components:

`formula`: The formula

`data`: Original full dataset

`sensi.estimates`: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. `lambda`) for each regression.

`N.obs`: Size of the dataset after matching it with tree tips and removing NA's.

`stats`: Main statistics for model parameters.`CI_low` and `CI_high` are the lower and upper limits of the 95

`all.stats`: Complete statistics for model parameters. `sd_intra` is the standard deviation due to intraspecific variation. `CI_low` and `CI_high` are the lower and upper limits of the 95

`sp.pb`: Species that caused problems with data transformation (see details above).

Warning

When Vy or Vx exceed Y or X, respectively, negative (or null) values can be generated, this might cause problems for data transformation (e.g. log-transformation). In these cases, the function will skip the simulation. This problem can be solved by increasing `n.intra`, changing the transformation type and/or checking the target species in output$sp.pb.

Author(s)

Caterina Penone & Pablo Ariel Martinez

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Martinez, P. a., Zurano, J.P., Amado, T.F., Penone, C., Betancur-R, R., Bidau, C.J. & Jacobina, U.P. (2015). Chromosomal diversity in tropical reef fishes is related to body size and depth range. Molecular Phylogenetics and Evolution, 93, 1-4

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phylolm, sensi_plot

Examples

```
# Load data:
data(alien)
# run PGLS accounting for intraspecific variation:
intra <- intra_phylm(gestaLen ~ adultMass, y.transf = log, x.transf = log,
phy = alien$phy[[1]], data = alien$data, Vy = "SD_gesta", n.intra = 30)
# To check summary results:
summary(intra)
# Visual diagnostics
sensi_plot(intra)
```

---

intra_physig	*Intraspecific variability - Phylogenetic signal*

Description

Performs Phylogenetic signal estimates evaluating trait intraspecific variability

Usage

```
intra_physig(
  trait.col,
  data,
  phy,
  V = NULL,
  n.intra = 100,
  distrib = "normal",
  method = "K",
  track = TRUE
)
```

Arguments

trait.col	The name of a column in the provided data frame with trait to be analyzed (e.g. "Body_mass").
data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'phylo', see ?ape).
V	Name of the column containing the standard deviation or the standard error of the trait variable. When information is not available for one taxon, the value can be 0 or NA.
n.intra	Number of times to repeat the analysis generating a random trait value. If NULL, n.intra = 30

distrib   A character string indicating which distribution to use to generate a random value for the response and/or predictor variables. Default is normal distribution: "normal" (function [rnorm](#)). Uniform distribution: "uniform" ([runif](#)) Warning: we recommend to use normal distribution with Vx or Vy = standard deviation of the mean.

method   Method to compute signal: can be "K" or "lambda".

track   Print a report tracking function progress (default = TRUE)

Details

This function estimates phylogenetic signal using [phylosig](#). The analysis is repeated n.intra times. At each iteration the function generates a random value for each row in the dataset using the standard deviation or errors supplied and assuming a normal or uniform distribution. To calculate means and se for your raw data, you can use the summarySE function from the package Rmisc.

Output can be visualised using sensi_plot.

Value

The function intra_physig returns a list with the following components:

Trait: Column name of the trait analysed

data: Original full dataset

intra.physig.estimates: Run number, phylogenetic signal estimate (lambda or K) and the p-value for each run with a different simulated datset.

N.obs: Size of the dataset after matching it with tree tips and removing NA's.

stats: Main statistics for signal estimateCI_low and CI_high are the lower and upper limits of the 95

Note

The argument "se" from [phylosig](#) is not available in this function. Use the argument "V" instead with [intra_physig](#) to indicate the name of the column containing the standard deviation or the standard error of the trait variable instead.

Author(s)

Caterina Penone & Pablo Ariel Martinez & Gustavo Paterno

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Martinez, P. a., Zurano, J.P., Amado, T.F., Penone, C., Betancur-R, R., Bidau, C.J. & Jacobina, U.P. (2015). Chromosomal diversity in tropical reef fishes is related to body size and depth range. Molecular Phylogenetics and Evolution, 93, 1-4

Blomberg, S. P., T. Garland Jr., A. R. Ives (2003) Testing for phylogenetic signal in comparative data: Behavioral traits are more labile. Evolution, 57, 717-745.

Pagel, M. (1999) Inferring the historical patterns of biological evolution. Nature, 401, 877-884.

Kamilar, J. M., & Cooper, N. (2013). Phylogenetic signal in primate behaviour, ecology and life history. Philosophical Transactions of the Royal Society B: Biological Sciences, 368: 20120341.

See Also

[phylosig](), [sensi_plot]()

Examples

```
## Not run:
data(alien)
alien.data<-alien$data
alien.phy<-alien$phy
# Run sensitivity analysis:
intra <- intra_physig(trait.col = "gestaLen", V = "SD_gesta" ,
                      data = alien.data, phy = alien.phy[[1]])
summary(intra)
sensi_plot(intra)
sensi_plot(intra, graphs = 1)
sensi_plot(intra, graphs = 2)

## End(Not run)
```

---

intra_samp_phyglm	*Interaction between intraspecific variability and species sampling - Phylogenetic Logistic Regression*

Description

Performs analyses of sensitivity to species sampling by randomly removing species and detecting the effects on parameter estimates in a phylogenetic logistic regression, while taking into account potential interactions with intraspecific variability.

Usage

```
intra_samp_phyglm(
  formula,
  data,
  phy,
  n.sim = 10,
  n.intra = 3,
  breaks = seq(0.1, 0.5, 0.1),
  btol = 50,
  Vx = NULL,
  distrib = "normal",
```

```
    x.transf = NULL,
    track = TRUE,
    ...
)
```

Arguments

formula	The model formula
data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
n.sim	The number of times species are randomly deleted for each break.
n.intra	Number of datasets resimulated taking into account intraspecific variation (see: "intra_phyglm")
breaks	A vector containing the percentages of species to remove.
btol	Bound on searching space. For details see phyloglm
Vx	Name of the column containing the standard deviation or the standard error of the predictor variable. When information is not available for one taxon, the value can be 0 or NA
distrib	A character string indicating which distribution to use to generate a random value for the response and/or predictor variables. Default is normal distribution: "normal" (function [rnorm](#)). Uniform distribution: "uniform" ([runif](#)) Warning: we recommend to use normal distribution with Vx = standard deviation of the mean.
x.transf	Transformation for the predictor variable (e.g. "log" or "sqrt"). Please use this argument instead of transforming data in the formula directly (see also details below).
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to phylolm

Details

This function randomly removes a given percentage of species (controlled by breaks) from the full phylogenetic logistic regression, fits a phylogenetic logistic regression model without these species using [phylolm](#), repeats this many times (controlled by n.sim), stores the results and calculates the effects on model parameters. This operation is repeated n.intra times for simulated values of the dataset, taking into account intraspecific variation. At each iteration, the function generates a random value for each row in the dataset using the standard deviation or errors supplied, and evaluates the effects of sampling within that iteration.

All phylogenetic models from phylolm can be used, i.e. BM, OUfixedRoot, OUrandomRoot, lambda, kappa, delta, EB and trend. See ?phylolm for details.

Currently, this function can only implement simple logistic models (i.e. $trait\ predictor$). In the future we will implement more complex models.

Output can be visualised using sensi_plot.

Value

The function `samp_phyglm` returns a list with the following components:

`formula`: The formula

`full.model.estimates`: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. `lambda` or `kappa`) for the full model without deleted species.

`sensi.estimates`: A data frame with all simulation estimates. Each row represents a model re-run with a given number of species `n.remov` removed, representing `n.percent` of the full dataset. Columns report the calculated regression intercept (`intercept`), difference between simulation intercept and full model intercept (`DIFintercept`), the percentage of change in intercept compared to the full model (`intercept.perc`) and intercept p-value (`pval.intercept`). All these parameters are also reported for the regression slope (`DIFestimate` etc.). Additionally, model aic value (`AIC`) and the optimised value (`optpar`) of the phylogenetic parameter (e.g. `kappa` or `lambda`, depending on the phylogenetic model used) are reported. Lastly we reported the standardised difference in intercept (`sDIFintercept`) and slope (`sDIFestimate`).

`sign.analysis` For each break (i.e. each percentage of species removed) this reports the percentage of statistically significant (at p<0.05) intercepts (`perc.sign.intercept`) over all repetitions as well as the percentage of statisticaly significant (at p<0.05) slopes (`perc.sign.estimate`).

`data`: Original full dataset.

Note

Please be aware that dropping species may reduce power to detect significant slopes/intercepts and may partially be responsible for a potential effect of species removal on p-values. Please also consult standardised differences in the (summary) output.

Author(s)

Gustavo Paterno, Gijsbert D.A. Werner & Caterina Penone

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Werner, G.D.A., Cornwell, W.K., Sprent, J.I., Kattge, J. & Kiers, E.T. (2014). A single evolutionary innovation drives the deep evolution of symbiotic N2-fixation in angiosperms. Nature Communications, 5, 4087.

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phylolm, samp_phyglm, intra_phyglm, intra_samp_phylm, sensi_plot

Examples

```
## Not run:
set.seed(6987)
phy = rtree(100)
x = rTrait(n=1,phy=phy,parameters=list(ancestral.state=2,optimal.value=2,sigma2=1,alpha=1))
X = cbind(rep(1,100),x)
y = rbinTrait(n=1,phy=phy, beta=c(-1,0.5), alpha=.7 ,X=X)
z = rnorm(n = length(x),mean = mean(x),sd = 0.1*mean(x))
dat = data.frame(y, x, z)
#Run sensitivity analysis:
intra_samp <- intra_samp_phyglm(formula = y ~ x, data = dat, phy = phy,
                                n.sim=10, n.intra = 3,
                                breaks=seq(.1,.5,.1),
                                Vx = "z", distrib="normal", x.transf=NULL)
summary(intra_samp)
sensi_plot(intra_samp)

## End(Not run)
```

---

intra_samp_phylm	*Interaction between intraspecific variability and species sampling - Phylogenetic Linear Regression*

Description

Performs analyses of sensitivity to species sampling by randomly removing species and detecting the effects on parameter estimates in a phylogenetic linear regression, while taking into account potential interactions with intraspecific variability.

Usage

```
intra_samp_phylm(
  formula,
  data,
  phy,
  n.sim = 10,
  n.intra = 3,
  breaks = seq(0.1, 0.5, 0.1),
  model = "lambda",
  Vy = NULL,
  Vx = NULL,
  distrib = "normal",
  y.transf = NULL,
  x.transf = NULL,
  track = TRUE,
  ...
)
```

Arguments

`formula`	The model formula
`data`	Data frame containing species traits with row names matching tips in `phy`.
`phy`	A phylogeny (class 'phylo') matching `data`.
`n.sim`	The number of times species are randomly deleted for each `break`.
`n.intra`	Number of datasets resimulated taking into account intraspecific variation (see: `"intra_phylm"`)
`breaks`	A vector containing the percentages of species to remove.
`model`	The phylogenetic model to use (see Details). Default is `lambda`. #' @param Vy Name of the column containing the standard deviation or the standard error of the response variable. When information is not available for one taxon, the value can be 0 or `NA`.
`Vy`	Name of the column containing the standard deviation or the standard error of the response variable. When information is not available for one taxon, the value can be 0 or `NA`.
`Vx`	Name of the column containing the standard deviation or the standard error of the predictor variable. When information is not available for one taxon, the value can be 0 or `NA`
`distrib`	A character string indicating which distribution to use to generate a random value for the response and/or predictor variables. Default is normal distribution: "normal" (function [rnorm](#)). Uniform distribution: "uniform" ([runif](#)) Warning: we recommend to use normal distribution with Vx or Vy = standard deviation of the mean.
`y.transf`	Transformation for the response variable (e.g. `"log"` or `"sqrt"`). Please use this argument instead of transforming data in the formula directly (see also details below).
`x.transf`	Transformation for the predictor variable (e.g. `"log"` or `"sqrt"`). Please use this argument instead of transforming data in the formula directly (see also details below).
`track`	Print a report tracking function progress (default = TRUE)
`...`	Further arguments to be passed to `phylolm`

Details

This function randomly removes a given percentage of species (controlled by `breaks`) from the full phylogenetic linear regression, fits a phylogenetic linear regression model without these species using [phylolm](#), repeats this many times (controlled by `n.sim`), stores the results and calculates the effects on model parameters. This operation is repeated `n.intra` times for simulated values of the dataset, taking into account intraspecific variation. At each iteration, the function generates a random value for each row in the dataset using the standard deviation or errors supplied, and evaluates the effects of sampling within that iteration.

All phylogenetic models from `phylolm` can be used, i.e. `BM`, `OUfixedRoot`, `OUrandomRoot`, `lambda`, `kappa`, `delta`, `EB` and `trend`. See ?phylolm for details.

Currently, this function can only implement simple linear models (i.e. $trait\ predictor$). In the future we will implement more complex models.

Output can be visualised using `sensi_plot`.

Value

The function `samp_phylm` returns a list with the following components:

`formula`: The formula

`full.model.estimates`: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. `lambda` or `kappa`) for the full model without deleted species.

`sensi.estimates`: A data frame with all simulation estimates. Each row represents a model re-run with a given number of species `n.remov` removed, representing `n.percent` of the full dataset. Columns report the calculated regression intercept (`intercept`), difference between simulation intercept and full model intercept (`DIFintercept`), the percentage of change in intercept compared to the full model (`intercept.perc`) and intercept p-value (`pval.intercept`). All these parameters are also reported for the regression slope (`DIFestimate` etc.). Additionally, model aic value (`AIC`) and the optimised value (`optpar`) of the phylogenetic parameter (e.g. `kappa` or `lambda`, depending on the phylogenetic model used) are reported. Lastly we reported the standardised difference in intercept (`sDIFintercept`) and slope (`sDIFestimate`).

`sign.analysis` For each break (i.e. each percentage of species removed) this reports the percentage of statistically significant (at p<0.05) intercepts (`perc.sign.intercept`) over all repetitions as well as the percentage of statisticaly significant (at p<0.05) slopes (`perc.sign.estimate`).

`data`: Original full dataset.

Note

Please be aware that dropping species may reduce power to detect significant slopes/intercepts and may partially be responsible for a potential effect of species removal on p-values. Please also consult standardised differences in the (summary) output.

Author(s)

Gustavo Paterno, Gijsbert D.A. Werner & Caterina Penone

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Werner, G.D.A., Cornwell, W.K., Sprent, J.I., Kattge, J. & Kiers, E.T. (2014). A single evolutionary innovation drives the deep evolution of symbiotic N2-fixation in angiosperms. Nature Communications, 5, 4087.

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phylolm, samp_phylm, intra_phylm, intra_samp_phyglm, sensi_plot

Examples

```
## Not run:
# Load data:
data(alien)
# Run analysis:
samp <- intra_samp_phylm(gestaLen ~ adultMass, phy = alien$phy[[1]],
                         y.transf = log,x.transf = NULL,Vy="SD_gesta",Vx=NULL,
                         data = alien$data, n.intra = 5, n.sim=10)
summary(samp)
head(samp$sensi.estimates)
# Visual diagnostics
sensi_plot(samp)
# You can specify which graph and parameter ("estimate" or "intercept") to print:
sensi_plot(samp, graphs = 1)
sensi_plot(samp, graphs = 2)

## End(Not run)
```

match_dataphy	*Match data and phylogeny based on model formula*

Description

Combines phylogeny and data to ensure that tips in phylogeny match data and that observations with missing values are removed. This function uses variables provided in the 'formula' argument to:

- Remove NA's: Check if there is any row with NA in the variables included in the formula. All rows containing NA will be removed from the data

- Match data and phy: Check if tips from phylogeny match rownames in data. Tips not present in data and phy will be removed from the phylogeny and data

- Return matched data and phy: The returned data has no NA in the variables included in 'formula' and only rows that match phylogeny tips. Returned phy has only tips that match data

Used internally in samp_phylm, samp_phyglm, clade_phylm, clade_phyglm, intra_phylm, intra_phyglm, tree_phylm, tree_phyglm and all function analysing interactions. Users can also directly use this function to combine a phylogeny and a dataset.

Usage

```
match_dataphy(formula, data, phy, verbose = TRUE, ...)
```

Arguments

formula	The model formula
data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'phylo' or 'multiphylo')

verbose	Print the number of species that match data and phylogeny and warnings. We highly recommend to use the default (verbose = T), but warning and information can be silenced for advanced use.
...	Further arguments to be passed to `match_dataphy`

Details

This function uses all variables provided in the 'formula' to match data and phylogeny. To avoid cropping the full dataset, 'match_dataphy' searches for NA values only on variables provided by formula. Missing values on other variables, not included in 'formula', will not be removed from data. If no species names are provided as row names in the dataset but the number of rows in the dataset is the same as the number of tips in the phylogeny, the function assumes that the dataset and the phylogeny are in the same order.

This ensures consistency between data and phylogeny only for the variables that are being used in the model (set by 'formula').

If phy is a 'multiphylo' object, all phylogenies will be cropped to match data. But the dataset order will only match the first tree provided. The returned phylogeny will be a 'multiphylo' object.

Value

The function `match_dataphy` returns a list with the following components:

`data`: Cropped dataset matching phylogeny

`phy`: Cropped phylogeny matching data

`dropped`: Species dropped from phylogeny and removed from data.

Note

If tips are removed from the phylogeny and data or if rows containing missing values are removed from data, a message will be printed with the details. Further, the final number of species that match data and phy will always be reported by a message.

Author(s)

Caterina Penone & Gustavo Paterno

References

This function is largely inspired by the function `comparative.data` in caper package David Orme, Rob Freckleton, Gavin Thomas, Thomas Petzoldt, Susanne Fritz, Nick Isaac and Will Pearse (2013). caper: Comparative Analyses of Phylogenetics and Evolution in R. R package version 0.5.2. http://CRAN.R-project.org/package=caper

Examples

```
# Load data:
data(alien)
head(alien$data)
# Match data and phy based on model formula:
```

```
comp.data <- match_dataphy(gestaLen ~ homeRange, data = alien$data, alien$phy[[1]])
# Check data:
head(comp.data$data)
# Check phy:
comp.data$phy
# See species dropped from phy or data:
comp.data$dropped
# Example2:
# Match data and phy based on model formula:
comp.data2 <- match_dataphy(gestaLen ~ adultMass, data = alien$data, alien$phy)
# Check data (missing data on variables not included in the formula are preserved)
head(comp.data2$data)
# Check phy:
comp.data2$phy
# See species dropped from phy or data:
comp.data2$dropped
```

---

miss.phylo.d   *Phylogenetic signal in missing data*

Description

Calculates D statistic (Fritz & Purvis 2010), a measure of phylogenetic signal, for missing data. Missingness is recoded into a binary variable (1=missing, 0=non missing). This function is an adaptation of phylo.d for missing data.

Usage

```
miss.phylo.d(data, phy, ...)
```

Arguments

data	Data frame containing species traits with species as row names.
phy	A phylogeny (class 'phylo', see ?ape).
...	Further arguments to be passed to phylo.d.

Details

This function builds on phylo.d to calculate a phylogenetic signal in missing data. The variable of interest, usually a trait, is recoded into a binary variable (1=missing data, 0=non missing data). Then the phylo.d function tests the estimated D value for significant departure from both random association and the clumping expected under a Brownian evolution threshold model (Fritz & Purvis, 2010).

Output can be visualised using print() and plot()

Value

The function `miss.phylo.d` returns an object of class "phylo.d" with the following components, for complete list of arguments see [phylo.d](#):

`DEstimate`: The estimated D value

`Pval1`: A p value, giving the result of testing whether D is significantly different from one

`Pval0`: A p value, giving the result of testing whether D is significantly different from zero The function also prints the percentage of missing data per variable in the dataset.

Author(s)

Caterina Penone & Gustavo Paterno

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Fritz, S. A. and Purvis, A. (2010). Selectivity in mammalian extinction risk and threat types: a new measure of phylogenetic signal strength in binary traits. Conservation Biology, 24(4):1042-1051.

David Orme, Rob Freckleton, Gavin Thomas, Thomas Petzoldt, Susanne Fritz, Nick Isaac and Will Pearse (2013). caper: Comparative Analyses of Phylogenetics and Evolution in R. R package version 0.5.2. https://CRAN.R-project.org/package=caper

Examples

```
# Load caper:
library(caper)
# Load data
data(primates)
data<-alien$data
phy=alien$phy[[1]]

# Test phylogenetic signal for missing data:
sexNAsig <- miss.phylo.d(data,phy,binvar=homeRange)
print(sexNAsig)
plot(sexNAsig)

massNAsig <- miss.phylo.d(data,phy,binvar=adultMass)
print(massNAsig)
plot(massNAsig)
```

---

primates   *Primates dataset: Example dataset for the package sensiPhy*

Description

A comparative dataset containing traits for 95 Primates species (primates.data) and a multiphylo object with 101 phylogenies matching the data (primates.phy). Tip labels are the binomial species names and match with data rownames. Data was taken from (Jones et al. 2009) and phylogenies from (Fritz et al 2009) and (Kuhn et al 2011).

Usage

```
data(primates)
```

Format

A data frame with 95 rows and 3 variables:

- family: Taxonomic family
- adultMass: Mean adult body mass (g)
- sexMaturity: Age when individuals are first physically capable of reproducing (days)
- homeRange: Mean home range (km)

References

Data downloaded from: http://esapubs.org/archive/ecol/E090/184/

Jones, K. E., Bielby, J., Cardillo, M., Fritz, S. A., O'Dell, J., Orme, C. D. L., Safi, K., Sechrest, W., Boakes, E. H., Carbone, C., Connolly, C., Cutts, M. J., Foster, J. K., Grenyer, R., Habib, M., Plaster, C. A., Price, S. A., Rigby, E. A., Rist, J., Teacher, A., Bininda-Emonds, O. R. P., Gittleman, J. L., Mace, G. M., Purvis, A. (2009), PanTHERIA: a species-level database of life history, ecology, and geography of extant and recently extinct mammals. Ecology, 90: 2648. doi: 10.1890/08-1494.1

Phylogeny: Kuhn, Tyler S., Arne O. Mooers, and Gavin H. Thomas. "A simple polytomy resolver for dated phylogenies." Methods in Ecology and Evolution 2.5 (2011): 427-436.

Fritz, Susanne A., Olaf RP Bininda-Emonds, and Andy Purvis. "Geographical variation in predictors of mammalian extinction risk: big is bad, but only in the tropics." Ecology letters 12.6 (2009): 538-549.

---

primates.data	*Primates dataset: Example dataset for the package sensiPhy*

Description

A comparative dataset containing traits for 95 Primates species (primates.data) and a multiphylo object with 101 phylogenies matching the data (primates.phy). Tip labels are the binomial species names and match with data rownames. Data was taken from (Jones et al. 2009) and phylogenies from (Fritz et al 2009) and (Kuhn et al 2011).

Usage

```
data(primates)
```

Format

A data frame with 95 rows and 3 variables:

- family: Taxonomic family
- adultMass: Mean adult body mass (g)
- sexMaturity: Age when individuals are first physically capable of reproducing (days)
- homeRange: Mean home range (km)

References

Data downloaded from: http://esapubs.org/archive/ecol/E090/184/

Jones, K. E., Bielby, J., Cardillo, M., Fritz, S. A., O'Dell, J., Orme, C. D. L., Safi, K., Sechrest, W., Boakes, E. H., Carbone, C., Connolly, C., Cutts, M. J., Foster, J. K., Grenyer, R., Habib, M., Plaster, C. A., Price, S. A., Rigby, E. A., Rist, J., Teacher, A., Bininda-Emonds, O. R. P., Gittleman, J. L., Mace, G. M., Purvis, A. (2009), PanTHERIA: a species-level database of life history, ecology, and geography of extant and recently extinct mammals. Ecology, 90: 2648. doi: 10.1890/08-1494.1

Phylogeny: Kuhn, Tyler S., Arne O. Mooers, and Gavin H. Thomas. "A simple polytomy resolver for dated phylogenies." Methods in Ecology and Evolution 2.5 (2011): 427-436.

Fritz, Susanne A., Olaf RP Bininda-Emonds, and Andy Purvis. "Geographical variation in predictors of mammalian extinction risk: big is bad, but only in the tropics." Ecology letters 12.6 (2009): 538-549.

---

primates.phy   *Primates dataset: Example dataset for the package sensiPhy*

Description

A comparative dataset containing traits for 95 Primates species (primates.data) and a multiphylo object with 101 phylogenies matching the data (primates.phy). Tip labels are the binomial species names and match with data rownames. Data was taken from (Jones et al. 2009) and phylogenies from (Fritz et al 2009) and (Kuhn et al 2011).

Usage

```
data(primates)
```

Format

A data frame with 95 rows and 3 variables:

- family: Taxonomic family
- adultMass: Mean adult body mass (g)
- sexMaturity: Age when individuals are first physically capable of reproducing (days)
- homeRange: Mean home range (km)

References

Data downloaded from: http://esapubs.org/archive/ecol/E090/184/

Jones, K. E., Bielby, J., Cardillo, M., Fritz, S. A., O'Dell, J., Orme, C. D. L., Safi, K., Sechrest, W., Boakes, E. H., Carbone, C., Connolly, C., Cutts, M. J., Foster, J. K., Grenyer, R., Habib, M., Plaster, C. A., Price, S. A., Rigby, E. A., Rist, J., Teacher, A., Bininda-Emonds, O. R. P., Gittleman, J. L., Mace, G. M., Purvis, A. (2009), PanTHERIA: a species-level database of life history, ecology, and geography of extant and recently extinct mammals. Ecology, 90: 2648. doi: 10.1890/08-1494.1

Phylogeny: Kuhn, Tyler S., Arne O. Mooers, and Gavin H. Thomas. "A simple polytomy resolver for dated phylogenies." Methods in Ecology and Evolution 2.5 (2011): 427-436.

Fritz, Susanne A., Olaf RP Bininda-Emonds, and Andy Purvis. "Geographical variation in predictors of mammalian extinction risk: big is bad, but only in the tropics." Ecology letters 12.6 (2009): 538-549.

---

samp_continuous	*Species Sampling uncertainty - Trait Evolution Continuous Characters*

Description

Fits models for trait evolution of continuous characters, evaluating sampling uncertainty.

Usage

```
samp_continuous(
  data,
  phy,
  n.sim = 30,
  breaks = seq(0.1, 0.5, 0.1),
  model,
  n.cores = NULL,
  bounds = list(),
  track = TRUE,
  ...
)
```

Arguments

data	Data vector for a single binary trait, with names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
n.sim	The number of times species are randomly deleted for each break.
breaks	A vector containing the percentages of species to remove.
model	The evolutionary model (see Details).
n.cores	number of cores to use. If 'NULL', number of cores is detected.
bounds	settings to constrain parameter estimates. See fitContinuous
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to fitContinuous

Details

This function randomly removes a given percentage of species (controlled by breaks), fits different models of continuous character evolution using fitContinuous, repeats this this many times (controlled by n.sim), stores the results and calculates the effects on model parameters.

Different evolutionary models from fitContinuous can be used, i.e. BM,OU, EB, trend, lambda, kappa, delta and drift.

See fitContinuous for more details on character models and tree transformations.

Output can be visualised using sensi_plot.

Value

The function tree_continuous returns a list with the following components:

call: The function call

data: The original full data vector

optpar: Transformation parameter used (e.g. lambda, kappa etc.)

full.model.estimates: Parameter estimates (rate of evolution sigsq and where applicable optpar), root state z0, AICc for the full model without deleted species.

break.summary.tab: Summary per break of the mean and median effects of species removal on percentage and absolute change parameter estimates.

sensi.estimates: Parameter estimates (sigsq and optpar),(percentual) difference in parameter estimate compared to the full model (DIFsigsq, sigsq.perc,sDIFsigsq, DIFoptpar, optpar.perc,sDIFoptpar), AICc and z0 for each repeat with random species removed.

optpar: Transformation parameter used (e.g. lambda, kappa etc.)

Author(s)

Gijsbert Werner & Gustavo Paterno

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Yang Z. 2006. Computational Molecular Evolution. Oxford University Press: Oxford.

Harmon Luke J, Jason T Weir, Chad D Brock, Richard E Glor, and Wendell Challenger. 2008. GEIGER: investigating evolutionary radiations. Bioinformatics 24:129-131.

Werner, G.D.A., Cornwell, W.K., Sprent, J.I., Kattge, J. & Kiers, E.T. (2014). A single evolutionary innovation drives the deep evolution of symbiotic N2-fixation in angiosperms. Nature Communications, 5, 4087.

See Also

fitContinuous

Examples

```
## Not run:
#Load data:
data("primates")
#Model trait evolution accounting for sampling size
adultMass<-primates$data$adultMass
names(adultMass)<-rownames(primates$data)
samp_cont<-samp_continuous(data = adultMass,phy = primates$phy[[1]],
model = "OU",n.sim=25,breaks=seq(.05,.2,.05),n.cores = 2, track = TRUE)
#Print summary statistics
summary(samp_cont)
sensi_plot(samp_cont)
sensi_plot(samp_cont, graphs = 1)
#Use a different evolutionary model
samp_cont2<-samp_continuous(data = adultMass,phy = primates$phy[[1]],
model = "kappa",n.sim=25,breaks=seq(.05,.2,.05),n.cores = 2,track = TRUE)
summary(samp_cont2)
sensi_plot(samp_cont2)
sensi_plot(samp_cont2, graphs = 2)
samp_cont3<-samp_continuous(data = adultMass,phy = primates$phy[[1]],
model = "BM",n.sim=25,breaks=seq(.05,.2,.05),n.cores = 2,track = TRUE)
summary(samp_cont3)

## End(Not run)
```

---

samp_discrete   *Species Sampling uncertainty - Trait Evolution Discrete Characters*

Description

Fits models for trait evolution of discrete (binary) characters, evaluating sampling uncertainty.

Usage

```
samp_discrete(
  data,
  phy,
  n.sim = 30,
  breaks = seq(0.1, 0.5, 0.1),
  model,
  transform = "none",
  bounds = list(),
  n.cores = NULL,
  track = TRUE,
  ...
)
```

Arguments

`data`	Data vector for a single binary trait, with names matching tips in `phy`.
`phy`	A phylogeny (class 'phylo') matching `data`.
`n.sim`	The number of times species are randomly deleted for each `break`.
`breaks`	A vector containing the percentages of species to remove.
`model`	The Mkn model to use (see Details).
`transform`	The evolutionary model to transform the tree (see Details). Default is `none`.
`bounds`	settings to constrain parameter estimates. See [`fitDiscrete`](#)
`n.cores`	number of cores to use. If 'NULL', number of cores is detected.
`track`	Print a report tracking function progress (default = TRUE)
`...`	Further arguments to be passed to [`fitDiscrete`](#)

Details

This function randomly removes a given percentage of species (controlled by `breaks`), fits different models of discrete character evolution using [`fitDiscrete`](#), repeats this this many times (controlled by `n.sim`), stores the results and calculates the effects on model parameters Currently, only binary discrete traits are supported.

Different character model from `fitDiscrete` can be used, including ER (equal-rates), SYM (symmetric), ARD (all-rates-different) and `meristic` (stepwise fashion).

Transformations to the phylogenetic tree from `fitDiscrete` can be used, i.e. `none`, `EB`, `lambda`, `kappa` and`delta`.

See [`fitDiscrete`](#) for more details on character models and tree transformations.

Output can be visualised using `sensi_plot`.

Value

The function `tree_discrete` returns a list with the following components:

`call`: The function call

`data`: The original full data vector

`optpar`: Transformation parameter used (e.g. `lambda`, `kappa` etc.)

`full.model.estimates`: Parameter estimates (transition rates q12 and q21), AICc and the optimised value of the phylogenetic transformation parameter (e.g. `lambda`) for the full model without deleted species.

`break.summary.tab`: Summary per `break` of the mean and median effects of species removal on percentage and absolute change in parameters q12 and q21.

`sensi.estimates`: Parameter estimates (transition rates q12 and q21),(percentual) difference in parameter estimate compared to the full model (DIFq12, sigsq.q12,sDIFq12, DIFq21, optpar.q21,sDIFq21), AICc and the optimised value of the phylogenetic transformation parameter (e.g. `lambda`) for each analysis with a species deleted.

`optpar`: Transformation parameter used (e.g. `lambda`, `kappa` etc.)

Author(s)

Gijsbert Werner & Gustavo Paterno

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Yang Z. 2006. Computational Molecular Evolution. Oxford University Press: Oxford.

Harmon Luke J, Jason T Weir, Chad D Brock, Richard E Glor, and Wendell Challenger. 2008. GEIGER: investigating evolutionary radiations. Bioinformatics 24:129-131.

Werner, G.D.A., Cornwell, W.K., Sprent, J.I., Kattge, J. & Kiers, E.T. (2014). A single evolutionary innovation drives the deep evolution of symbiotic N2-fixation in angiosperms. Nature Communications, 5, 4087.

See Also

fitDiscrete

Examples

```
## Not run:
#Load data:
data("primates")
#Create a binary trait factor
adultMass_binary<-ifelse(primates$data$adultMass > 7350, "big", "small")
adultMass_binary<-as.factor(as.factor(adultMass_binary))
names(adultMass_binary)<-rownames(primates$data)
#Model trait evolution accounting for sampling size
samp_binary<-samp_discrete(data = adultMass_binary,phy = primates$phy[[1]],
n.sim=25,breaks=seq(.1,.3,.1),model = "SYM",transform = "none",n.cores = 2,track = TRUE)
#Print summary statistics
summary(samp_binary)
sensi_plot(samp_binary)
sensi_plot(samp_binary,graphs=1)
sensi_plot(samp_binary,graphs=2)
#Use a different evolutionary model or transformation
samp_binary2<-samp_discrete(data = adultMass_binary,phy = primates$phy[[1]],
n.sim=25,breaks=seq(.1,.3,.1),model = "ARD",transform = "lambda",n.cores = 2,track = TRUE)
summary(samp_binary2)
sensi_plot(samp_binary2)
sensi_plot(samp_binary2,graphs=1)
sensi_plot(samp_binary2,graphs=3)

## End(Not run)
```

samp_phyglm   *Sensitivity Analysis Species Sampling - Phylogenetic Logistic Regression*

Description

Performs analyses of sensitivity to species sampling by randomly removing species and detecting the effects on parameter estimates in phylogenetic logistic regression.

Usage

```
samp_phyglm(
  formula,
  data,
  phy,
  n.sim = 30,
  breaks = seq(0.1, 0.5, 0.1),
  btol = 50,
  track = TRUE,
  ...
)
```

Arguments

formula	The model formula
data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
n.sim	The number of times species are randomly deleted for each break.
breaks	A vector containing the percentages of species to remove.
btol	Bound on searching space. For details see phyloglm
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to phyloglm

Details

This function randomly removes a given percentage of species (controlled by breaks) from the full phylogenetic logistic regression, fits a phylogenetic logistic regression model without these species using [phyloglm](), repeats this many times (controlled by n.sim), stores the results and calculates the effects on model parameters.

Only logistic regression using the "logistic_MPLE"-method from phyloglm is implemented.

Currently, this function can only implement simple logistic models (i.e. $trait\ predictor$). In the future we will implement more complex models.

Output can be visualised using sensi_plot.

Value

The function `samp_phylm` returns a list with the following components:

`formula`: The formula

`full.model.estimates`: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. `lambda` or `kappa`) for the full model without deleted species.

`sensi.estimates`: A data frame with all simulation estimates. Each row represents a model re-run with a given number of species `n.remov` removed, representing `n.percent` of the full dataset. Columns report the calculated regression intercept (`intercept`), difference between simulation intercept and full model intercept (`DIFintercept`), the percentage of change in intercept compared to the full model (`intercept.perc`) and intercept p-value (`pval.intercept`). All these parameters are also reported for the regression slope (`DIFestimate` etc.). Additionally, model aic value (`AIC`) and the optimised value (`optpar`) of the phylogenetic parameter (e.g. `kappa` or `lambda`, depending on the phylogenetic model used) are reported. Lastly we reported the standardised difference in intercept (`sDIFintercept`) and slope (`sDIFestimate`).

`sign.analysis` For each break (i.e. each percentage of species removed) this reports the percentage of statistically significant (at p<0.05) intercepts (`perc.sign.intercept`) over all repetitions as well as the percentage of statisticaly significant (at p<0.05) slopes (`perc.sign.estimate`).

`data`: Original full dataset.

Note

Please be aware that dropping species may reduce power to detect significant slopes/intercepts and may partially be responsible for a potential effect of species removal on p-values. Please also consult standardised differences in the (summary) output.

Author(s)

Gustavo Paterno & Gijsbert D.A. Werner

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467.

Werner, G.D.A., Cornwell, W.K., Sprent, J.I., Kattge, J. & Kiers, E.T. (2014). A single evolutionary innovation drives the deep evolution of symbiotic N2-fixation in angiosperms. Nature Communications, 5, 4087.

#' Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phyloglm, samp_phylm, influ_phyglm, sensi_plot

Examples

```
# Simulate Data:
set.seed(6987)
phy = rtree(100)
x = rTrait(n=1,phy=phy)
X = cbind(rep(1,100),x)
y = rbinTrait(n=1,phy=phy, beta=c(-1,0.5), alpha=.7 ,X=X)
dat = data.frame(y, x)
# Run sensitivity analysis:
samp <- samp_phyglm(y ~ x, data = dat, phy = phy, n.sim = 10)
# To check summary results and most influential species:
summary(samp)
## Not run:
# Visual diagnostics for clade removal:
sensi_plot(samp)

## End(Not run)
```

---

samp_phylm   *Sensitivity Analysis Species Sampling - Phylogenetic Linear Regres-
                          sion*

Description

Performs analyses of sensitivity to species sampling by randomly removing species and detecting
the effects on parameter estimates in a phylogenetic linear regression.

Usage

```
samp_phylm(
  formula,
  data,
  phy,
  n.sim = 30,
  breaks = seq(0.1, 0.5, 0.1),
  model = "lambda",
  track = TRUE,
  ...
)
```

Arguments

formula	The model formula
data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
n.sim	The number of times species are randomly deleted for each break.
breaks	A vector containing the percentages of species to remove.

model	The phylogenetic model to use (see Details). Default is lambda.
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to phylolm

Details

This function randomly removes a given percentage of species (controlled by breaks) from the full phylogenetic linear regression, fits a phylogenetic linear regression model without these species using [phylolm](), repeats this many times (controlled by n.sim), stores the results and calculates the effects on model parameters.

All phylogenetic models from phylolm can be used, i.e. BM, OUfixedRoot, OUrandomRoot, lambda, kappa, delta, EB and trend. See ?phylolm for details.

Currently, this function can only implement simple linear models (i.e. $trait\ predictor$). In the future we will implement more complex models.

Output can be visualised using sensi_plot.

Value

The function samp_phylm returns a list with the following components:

formula: The formula

full.model.estimates: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. lambda or kappa) for the full model without deleted species.

sensi.estimates: A data frame with all simulation estimates. Each row represents a model re-run with a given number of species n.remov removed, representing n.percent of the full dataset. Columns report the calculated regression intercept (intercept), difference between simulation intercept and full model intercept (DIFintercept), the percentage of change in intercept compared to the full model (intercept.perc) and intercept p-value (pval.intercept). All these parameters are also reported for the regression slope (DIFestimate etc.). Additionally, model aic value (AIC) and the optimised value (optpar) of the phylogenetic parameter (e.g. kappa or lambda, depending on the phylogenetic model used) are reported. Lastly we reported the standardised difference in intercept (sDIFintercept) and slope (sDIFestimate).

sign.analysis For each break (i.e. each percentage of species removed) this reports the percentage of statistically significant (at p<0.05) intercepts (perc.sign.intercept) over all repetitions as well as the percentage of statisticaly significant (at p<0.05) slopes (perc.sign.estimate).

data: Original full dataset. #' @note Please be aware that dropping species may reduce power to detect significant slopes/intercepts and may partially be responsible for a potential effect of species removal on p-values. Please also consult standardised differences in the (summary) output.

Author(s)

Gustavo Paterno & Gijsbert D.A. Werner

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467.

Werner, G.D.A., Cornwell, W.K., Sprent, J.I., Kattge, J. & Kiers, E.T. (2014). A single evolutionary innovation drives the deep evolution of symbiotic N2-fixation in angiosperms. Nature Communications, 5, 4087.

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phylolm, samp_phyglm, influ_phylm,sensi_plot

Examples

```
# Load data:
data(alien)
# Run analysis:
samp <- samp_phylm(log(gestaLen) ~ log(adultMass), phy = alien$phy[[1]],
data = alien$data, n.sim = 10)
summary(samp)
head(samp$sensi.estimates)
# Visual diagnostics
## Not run:
sensi_plot(samp)
# You can specify which graph and parameter ("estimate" or "intercept") to print:
sensi_plot(samp, graphs = 1, param = "estimate")
sensi_plot(samp, graphs = 2, param = "intercept")

## End(Not run)
```

---

samp_physig   *Sensitivity Analysis Species Sampling - Phylogenetic signal*

Description

Performs analyses of sensitivity to species sampling by randomly removing species and detecting the effects on phylogenetic signal estimates

Usage

```
samp_physig(
  trait.col,
  data,
  phy,
  n.sim = 30,
  breaks = seq(0.1, 0.5, 0.1),
  method = "K",
  track = TRUE,
  ...
)
```

Arguments

`trait.col`	The name of a column in the provided data frame with trait to be analyzed (e.g. "Body_mass").
`data`	Data frame containing species traits with row names matching tips in phy.
`phy`	A phylogeny (class 'phylo') matching `data`.
`n.sim`	The number of times to repeat species random removal for each `break` interval.
`breaks`	A vector containing the percentages of species to remove.
`method`	Method to compute signal: can be "K" or "lambda".
`track`	Print a report tracking function progress (default = TRUE)
`...`	Further arguments to be passed to phylosig

Details

This function randomly removes a given percentage of species (controlled by `breaks`) from the full data, estimates phylogenetic signal for a given trait (K or lambda) without these species using phylosig, then repeats the analysis many times (controlled by `n.sim`), stores the results and calculates the effect of random species removal on phylogenetic signal estimates.

Output can be visualised using `sensi_plot`.

Value

The function `samp_phylosig` returns a list with the following components:

`Trait`: Column name of the trait analysed

`full.model.estimates`: Phylogenetic signal (K or lambda) and p-value using the full dataset (without deleted species). See phylosig for details.

`sensi.estimates`: A data frame with all simulation estimates. Each row represents a rerun with a given number of species `n.remov` removed, representing `n.percent` of the full dataset. Columns report the calculated signal estimate (`estimate`), difference between reduced data signal estimate and full data signal (`DF`), the percentage of change in signal compared to the full data estimate (`perc`) and signal p-value for the reduced data estimate(`pval`).

`sign.analysis` For each break (i.e. each percentage of species removed) this reports the percentage of statistically significant (at p<0.05) phylogenetic signal over all repetitions with reduced data sets.

`data`: Original full dataset used in the analysis. #' @note Please be aware that dropping species may reduce power to detect significant signal and may partially be responsible for a potential effect of species removal on p-values. Please also consult standardised differences in the (summary) output.

Note

The argument "se" from phylosig is not available in this function. Use the argument "V" instead with intra_physig to indicate the name of the column containing the standard deviation or the standard error of the trait variable instead.

Author(s)

Gustavo Paterno & Gijsbert D.A. Werner

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467.

Werner, G.D.A., Cornwell, W.K., Sprent, J.I., Kattge, J. & Kiers, E.T. (2014). A single evolutionary innovation drives the deep evolution of symbiotic N2-fixation in angiosperms. Nature Communications, 5, 4087.

Blomberg, S. P., T. Garland Jr., A. R. Ives (2003) Testing for phylogenetic signal in comparative data: Behavioral traits are more labile. Evolution, 57, 717-745.

Pagel, M. (1999) Inferring the historical patterns of biological evolution. Nature, 401, 877-884.

Kamilar, J. M., & Cooper, N. (2013). Phylogenetic signal in primate behaviour, ecology and life history. Philosophical Transactions of the Royal Society B: Biological Sciences, 368: 20120341.

See Also

phylosig, samp_phylm,sensi_plot

Examples

```
## Not run:
data(alien)
alien.data<-alien$data
alien.phy<-alien$phy
# Logtransform data
alien.data$logMass <- log(alien.data$adultMass)
# Run sensitivity analysis:
samp <- samp_physig(trait.col = "logMass", data = alien.data, n.sim = 30,
phy = alien.phy[[1]])
summary(samp)
sensi_plot(samp)
sensi_plot(samp, graphs = 1)
sensi_plot(samp, graphs = 2)

## End(Not run)
```

---

sensiPhy   *Sensitivity Analysis for Comparative Methods*

Description

An implementation of sensitivity analysis for phylogenetic comparative methods. The package is an umbrella of statistical and graphical methods that estimate and report different types of uncertainty in PCM: (i) Species Sampling uncertainty (sample size; influential species and clades). (ii) Phylogenetic uncertainty (different topologies and/or branch lengths). (iii) Data uncertainty (intraspecific variation and measurement error).

Bug reports

https://github.com/paternogbc/sensiPhy/issues.

How to contribute?

You can find instructions on how to contribute to **sensiPhy** at this link: https://github.com/paternogbc/sensiPhy/wiki/How-to-support-sensiPhy

A quick tutorial is available at this link: https://github.com/paternogbc/sensiPhy/wiki

Author(s)

Gustavo Paterno; Caterina Penone & Gijsbert D.A. Werner

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467.

---

sensi_plot	*Graphical sensitivity analysis for comparative methods*

Description

Generic function for plotting results from any sensitivity analysis performed with 'sensiPhy'

Usage

```
sensi_plot(x, ...)
```

Arguments

x	any output from the sensiPhy package.
...	further arguments to methods

Details

sensi_plot recognize and print different sets of graphs depending on the function that generated 'x'. See the links below for details about the graphs generated for each sensiPhy function:

PGLS regressions (single uncertainty):

- clade_phylm: `sensi_plot.sensiClade`
- influ_phylm: `sensi_plot.sensiInflu`
- samp_phylm: `sensi_plot.sensiSamp`
- intra_phylm: `sensi_plot.sensiIntra`
- tree_phylm: `sensi_plot.sensiTree`

PGLS regressions (interacting uncertainties):

- tree_intra_phylm: `sensi_plot.sensiTree_Intra`

- intra_clade_phylm: `sensi_plot.sensiIntra_Clade`

- intra_influ_phylm: `sensi_plot.sensiIntra_Influ`

- intra_samp_phylm: `sensi_plot.sensiIntra_Samp`

- tree_clade_phylm: `sensi_plot.sensiTree_Clade`

- tree_influ_phylm: `sensi_plot.sensiTree_Influ`

- tree_samp_phylm: `sensi_plot.sensiTree_Samp`

  Phylogenetic signal:

- clade_physig: `sensi_plot.clade.physig`

- influ_physig: `sensi_plot.influ.physig`

- samp_physig: `sensi_plot.samp.physig`

- tree_physig: `sensi_plot.tree.physig`

- intra_physig: `sensi_plot.intra.physig`

  trait evolution (continuous & discrete characters):

- clade_continuous & _discrete `sensi_plot.sensiClade.TraitEvol`

- influ_continuous & _discrete `sensi_plot.sensiInflu.TraitEvol`

- tree_continuous & _discrete `sensi_plot.sensiTree.TraitEvol`

- samp_continuous & _discrete `sensi_plot.sensiSamp.TraitEvol`

Author(s)

Gustavo Paterno

References

The function 'multiplot', developed by Winston Chang, is used inside sensi_plot to print multiple graphs in one frame. The source code is available here: http://www.cookbook-r.com/Graphs/Multiple_graphs_on_one_page_(ggplot2)/

---

sensi_plot.clade.physig

                              Graphical diagnostics for class 'clade.physig'

Description

`plot_clade_physig` Plot results from `clade_physig`

Usage

```
## S3 method for class 'clade.physig'
sensi_plot(x, clade = NULL, ...)
```

Arguments

x	output from `influ_physig`
clade	The name of the clade to be evaluated (see details)
...	further arguments to methods.

Details

For 'x' from clade_physig:

Graph 1: Distribution of the simulated signal estimates (Null distribution for a given clade sample size). The red dashed line represents the estimated signal for the reduced data (without the focal clade) and the black line represents the signal estimate for the full data.

Author(s)

Gustavo Paterno

See Also

[ggplot](), [phylosig]()

---

sensi_plot.influ.physig

Graphical diagnostics for class 'influ.physig'

Description

`sensi_plot.influ_physig` Plot results from `influ_physig`

Usage

```
## S3 method for class 'influ.physig'
sensi_plot(x, graphs = "all", ...)
```

Arguments

x	output from `influ_physig`
graphs	choose which graph should be printed on the output ("all", 1, 2)
...	further arguments to methods.

Details

For 'x' from influ_physig:

Graph 1: Distribution of estimated phylogenetic signal (K or lambda) for each simulation (leave-one-out deletion). Dashed red vertical line represents the original estimate of phylogenetic signal with the full data (with all species).

Graph 2: Distribution of P values for the phylogenetic signal (K or lambda) for each simulation (leave-one-out deletion). Red vertical line represents the alpha significance level = 0.05.

Author(s)

Gustavo Paterno

See Also

[ggplot](), [phylosig]()

---

sensi_plot.intra.physig

Graphical diagnostics for class 'intra.physig'

Description

sensi_plot_intra.physig Plot results from intra_physig.

Usage

```
## S3 method for class 'intra.physig'
sensi_plot(x, graphs = "all", ...)
```

Arguments

x	output from intra_physig
graphs	choose which graph should be printed in the output ("all", 1 or 2)
...	further arguments to methods

Details

For 'x' from intra_physig

Graphs 1: Distribution of estimated phylogenetic signal (lambda or K) for each simulation Red vertical line represents the average signal among all estimates.

Graph 2: Distribution of p-values for the phylogenetic signal (K or lambda) for each simulation. Red vertical line represents the alpha significance level = 0.05.

Author(s)

Caterina Penone and Gustavo Paterno

See Also

[ggplot](), [intra_phylm intra_phylm]()

sensi_plot.samp.physig

Graphical diagnostics for class 'samp.physig'

Description

`plot_samp_phylm` Plot results from `samp_physig`

Usage

```
## S3 method for class 'samp.physig'
sensi_plot(x, graphs = "all", ...)
```

Arguments

x	output from `samp_physig`
graphs	choose which graph should be printed on the output ("all", 1,2 and 3)
...	further arguments to methods

Details

For 'x' from samp_physig:

Graph 1: Estimated phylogenetic signal for each simulation across percentages of species removed. Colours represent percentage of change in comparison with the full data estimate (blue = lower than 5, orange = between 5 and 10 and red = higher than 10). The red horizontal line represents the original phylogenetic signal estimated from the full model (with all species).

Graph 2: The proportion of estimated signal in each category across the percentage of species removed.

Graph 3: The percentage of significant signal estimates across the percentage of species removed.

Author(s)

Gustavo Paterno

See Also

[ggplot](), [samp_phylm]() [samp_physig]()

---

sensi_plot.sensiClade   *Graphical diagnostics for class 'sensiClade'*

Description

Plot results from `clade_phylm` and `clade_phyglm`

Usage

```
## S3 method for class 'sensiClade'
sensi_plot(x, clade = NULL, ...)
```

Arguments

x	output from `clade_phylm` or `clade_phyglm`
clade	The name of the clade to be evaluated (see details)
...	further arguments to methods.

Details

For 'x' from clade_phylm or clade_phyglm:

Graph 1: The original scatterplot $y = a + bx$ (with the full dataset) and a comparison between the regression lines of the full dataset and the rerun without the selected clade (set by `clade`). For further details about this method, see `clade_phylm`.

Species from the selected clade are represented in red (removed species), black solid line represents the regression with the full model and red dashed line represents the regression of the model without the species from the selected clade. To check the available clades to plot, see `x$sensi.estimates$clade` in the object returned from `clade_phylm` or `clade_phyglm`.

Graph 2: Distribution of the simulated slopes (Null distribution for a given clade sample size). The red dashed line represents the estimated slope for the reduced model (without the focal clade) and the black line represents the slope for the full model.

Author(s)

Gustavo Paterno

See Also

`clade_phylm`

sensi_plot.sensiClade.TraitEvol

Graphical diagnostics for class 'sensiClade.TraitEvol'

Description

Plot results from `clade_discrete` and `clade_continuous`

Usage

```
## S3 method for class 'sensiClade.TraitEvol'
sensi_plot(x, clade = NULL, graph = "all", ...)
```

Arguments

x	output from `clade_discrete` or `clade_continuous`
clade	The name of the clade to be evaluated (see details)
graph	The graph to be printed. Default `all`, or for `clade_continuous` set `sigsq` or `optpar` and for `clade_discrete` set`q12` or `q21`.
...	further arguments to methods.

Details

For 'x' from clade_discrete or clade_continuous:

Graph 1: Distribution of the simulated parameter estimates (Null distribution for a given clade sample size). The red dashed line represents the estimated signal for the reduced data (without the focal clade) and the black line represents the signal estimate for the full data.

Author(s)

Gustavo Paterno & Gijsbert Werner

See Also

[clade_continuous](#) or [clade_discrete](#)

---

sensi_plot.sensiInflu   *Graphical diagnostics for class 'sensiInflu'*

Description

plot_influ_phylm Plot results from `influ_phylm` and `influ_phyglm`

Usage

```
## S3 method for class 'sensiInflu'
sensi_plot(x, graphs = "all", param = "estimate", ...)
```

Arguments

x	output from `influ_phylm`
graphs	choose which graph should be printed on the output ("all", 1,2,3 or 4)
param	choose which parameter ("intercept" or "estimate" should be printed)
...	further arguments to methods

Details

For 'x' from influ_phylm or influ_phyglm:

Graph 1: Distribution of estimated slopes (estimates) or intercepts for each simulation (leave-one-out deletion). Red vertical line represents the original slope or intercept from the full model (with all species).

Graph 2: Original regression plot ($trait\ predictor$). Standardized difference in slope or intercept is represented by a continuous size scale. The names of the most influential species (sDF > cutoff) are ploted in the graph.

Graph 3: Distribution of standardized difference in slope or intercept. Red colour indicates influential species (with a standardised difference above the value of `cutoff`).

Graph 4: Distribution of the percentage of change in slope or intercept.

Author(s)

Gustavo Paterno

See Also

ggplot

---

sensi_plot.sensiInflu.TraitEvol
Graphical diagnostics for class 'sensiInflu.TraitEvol'

Description

sensi_plot.sensiTree.TraitEvol Plot results from influ_discrete and influ_continuous.

Usage

```
## S3 method for class 'sensiInflu.TraitEvol'
sensi_plot(x, graphs = "all", ...)
```

Arguments

x   output from influ_discrete or influ_continuous

graphs   choose which graph should be printed in the output ("all", "q12", "q21", "aic"
            or" "optpar")

...   further arguments to methods

Author(s)

Gijsbert Werner

See Also

[ggplot](), [influ_discrete](#) [influ_continuous]()

The following graphs are printed.

Graph aicc: Distribution of estimated AICc-values across all single-species deletions. Red vertical line represents the mean signal among all estimates. Blue vertical line represents the median signal among all estimates.

Graph optpar: Distribution of estimated values for optimisation parameter specified using 'transform' (if applicable) Red vertical line represents the mean signal among all estimates. Blue vertical line represents the median signal among all estimates.

Additionally, only for tree_discrete the function creates the following graphs.

Graph q12: Distribution of estimated parameter values for transition rates q12 across all single-species deletions. Red vertical line represents the mean signal among all estimates. Blue vertical line represents the median signal among all estimates.

Graph q21: Distribution of estimated parameter values for transition rates q21. Red vertical line represents the mean signal among all estimates. Blue vertical line represents the median signal among all estimates.

While only for tree_continuous the function creates the following graphs.

Graph sigsq: Distribution of estimated parameter values for rate of evolution sigsq across all single-species deletions. . Red vertical line represents the mean signal among all estimates. Blue vertical line represents the median signal among all estimates.

Graph z0: Distribution of estimated parameter values for z0. Red vertical line represents the mean signal among all estimates. Blue vertical line represents the median signal among all estimates.

---

sensi_plot.sensiIntra   *Graphical diagnostics for class 'sensiIntra'*

Description

`plot_tree.intra_phylm` Plot results from `tree_phylm`, `intra_phylm` and `intra_phyglm`

Usage

```
## S3 method for class 'sensiIntra'
sensi_plot(x, graphs = "all", ...)
```

Arguments

x	output from `tree_phylm`, `tree_phyglm`, `intra_phylm` or `intra_phyglm`
graphs	choose which graph should be printed in the output ("all", 1, 2 or 3)
...	further arguments to methods

Details

For 'x' from `tree_phylm`, `tree_phyglm`, `intra_phylm` or `intra_phyglm`:

Graphs 1 and 2: Distribution of estimated slopes (estimates) and intercepts for each tree (for `tree_phylm`) or value generated within a given interval (`intra_phylm`) Red vertical line represents the mean estimate or intercept for all models.

Graph 3: Scatterplot with mean regression (black line) and standard deviation of the regression (red dotted lines).

Author(s)

Caterina Penone and Gustavo Paterno

See Also

ggplot, tree_phylm intra_phylm

```
sensi_plot.sensiIntra_Clade
```
Graphical diagnostics for class 'sensiIntra_Clade'

Description

Plot results from `intra_clade_phylm` and `intra_clade_phyglm`

Usage

```
## S3 method for class 'sensiIntra_Clade'
sensi_plot(x, clade = NULL, graphs = "all", ...)
```

Arguments

x	output from `intra_clade_phylm` or `intra_clade_phyglm`
clade	The name of the clade to be evaluated (see details)
graphs	Choose which graph should be printed on the output ("all", 1 or 2). Defaults to "all".
...	further arguments to methods.

Details

For 'x' from intra_clade_phylm or intra_clade_phyglm:

Graph 1: Estimated slopes after clade removal (reduced data) across multiple simulations. Small dots represent estimates reruns between simulations while larger dots represents the average estimate between all simulations for each clade. The solid black line represents the average slope estimate among trees using the full dataset.

Graph 2: The effect of clade removal on slope estimate across all individual simulations for each clade analyzed. The black line indicates estimates with the full dataset while the red line represent estimates without the focal clade (reduced data) across different simulation The blue dots represent null expectation estimates after removing the same number of species of the focal clade, with dots falling outside the red line area indicating a larger than expected absolute effect.

Author(s)

Gustavo Paterno, Caterina Penone

See Also

[intra_clade_phylm](#)

sensi_plot.sensiIntra_Influ

Graphical diagnostics for class 'sensiIntra_Influ'

Description

sensi_plot.intra_influ Plot results from `intra_influ_phylm` and `intra_influ_phyglm`

Usage

```
## S3 method for class 'sensiIntra_Influ'
sensi_plot(x, graphs = "all", ...)
```

Arguments

x	output from `influ_phylm` or `intra_influ_phyglm`
graphs	choose which graph should be printed on the output ("all", 1,2,3 or 4)
...	further arguments to methods

Details

For 'x' from intra_influ_phylm or intra_influ_phyglm:

Graph 1: Most common influential species on model estimates. Percentage of iterations (n.tree or n.intra) where the removal of individual species caused significant change in model estimate (sDIFestimate > cutoff).

Graph 2: Shift on model estimate after species removal for most influential species. Small red dots represent individual reruns between different trees or simulations while large red dots represent the average DIFestimate among all iterations.

Author(s)

Gustavo Paterno, Caterina Penone

See Also

[intra_influ_phylm](); [intra_influ_phyglm]()

---

sensi_plot.sensiIntra_Samp

Graphical diagnostics for class 'sensiIntra_Samp' and 'sensiTree_samp'

Description

`sensi_plot.sensiIntra_Samp` Plot results from `sensiIntra_Samp_phylm` and `sensiIntra_Samp_phyglm`

Usage

```
## S3 method for class 'sensiIntra_Samp'
sensi_plot(x, graphs = "all", ...)
```

Arguments

x	output from `sensiIntra_Samp_phylm` or `sensiIntra_Samp_phyglm`
graphs	choose which graph should be printed on the output ("all", 1,2,3 or 4)
...	further arguments to methods

Details

For 'x' from sensiIntra_Samp_phylm or sensiIntra_Samp_phyglm:

Graph 1: : Estimated estimates for each simulation across percentages of species removed. Small red dots represent individual estimates for each iteration (tree or intra) across percentages of removed species. Large dots represent the average among these estimates within each iteration and percentage of species removal. Different estimates within the same percentage interval represent different iterations (tree or intra).

Graph 2: : The percentage of significant estimates ($p < 0.05$) across the percentage of species removed. Small red dots represent individual iterations (tree or intra) while large dots represent the average among these iterations.

Author(s)

Gustavo Paterno

See Also

[ggplot](#), [samp_phylm](#) [samp_phyglm](#)

---

sensi_plot.sensiSamp   *Graphical diagnostics for class 'sensiSamp'*

Description

`plot_samp_phylm` Plot results from `samp_phylm` and `influ_phyloglm`

Usage

```
## S3 method for class 'sensiSamp'
sensi_plot(x, graphs = "all", param = "estimate", ...)
```

Arguments

x	output from `samp_phylm`
graphs	choose which graph should be printed on the output ("all", 1,2,3 or 4)
param	choose which model parameter should be ploted ("intercept" or "estimate")
...	further arguments to methods

Details

For 'x' from samp_phylm or samp_phyglm:

Graph 1: Estimated slopes or intercepts for each simulation across percentages of species removed. Colours represent percentage of change in comparison with the full model (blue = lower than 5, orange = between 5 and 10 and red = higher than 10). The red horizontal line represents the original slope or intercept from the full model (with all species).

Graph 2: The proportion of estimated slopes and intercepts in each category across the percentage of species removed.

Graph 3: Estimated phylogenetic model parameter for each simulation across the percentage of species removed.

Graph 4: The percentage of significant slopes or intercepts across the percentage of species removed.

Note

If model = "BM", only plots 1, 2 and 4 are printed. Plot 3, phylogenetic model parameter is not available for model = "BM"

Author(s)

Gustavo Paterno

See Also

[ggplot](#), [samp_phylm](#) [samp_phyglm](#)

```
sensi_plot.sensiSamp.TraitEvol
```
Graphical diagnostics for class 'sensiSamp.TraitEvol'

Description

sensi_plot.sensiSamp.TraitEvol Plot results from `samp_continuous` and `samp_discrete`

Usage

```
## S3 method for class 'sensiSamp.TraitEvol'
sensi_plot(x, graphs = "all", ...)
```

Arguments

x	output from `samp_continuous` or `samp_continuous`
graphs	choose which graph should be printed on the output ("all", 1,2,3 or 4)
...	further arguments to methods

Details

For 'x' from samp_continuous or samp_discrete:

Graph 1: Estimated q12 (discrete) or sigsq (discrete) for each simulation across percentages of species removed. Colours represent percentage of change in comparison with the full model (blue = lower than 5, orange = between 5 and 10 and red = higher than 10).The red horizontal line represents the original value from the full model (with all species).

Graph 2: The proportion of estimated q12 (discrete) or sigsq (discrete) in each category across the percentage of species removed.

Graph 3: Estimated q21 for each simulation across the percentage of species removed (only for `samp_discrete`).

Graph 4: The percentage of significant q21 across the percentage of species removed (only for `samp_discrete`).

Author(s)

Gustavo Paterno & Gijsbert Werner

See Also

[ggplot](), [samp_continuous]() [samp_discrete]()

---

sensi_plot.sensiTree   *Graphical diagnostics for class 'sensiTree'*

Description

`plot_tree.intra_phylm` Plot results from `tree_phylm`, `intra_phylm` and `intra_phyglm`

Usage

```
## S3 method for class 'sensiTree'
sensi_plot(x, graphs = "all", ...)
```

Arguments

x	output from `tree_phylm`, `tree_phyglm`, `intra_phylm` or `intra_phyglm`
graphs	choose which graph should be printed in the output ("all", 1, 2 or 3)
...	further arguments to methods

Details

For 'x' from `tree_phylm`, `tree_phyglm`, `intra_phylm` or `intra_phyglm`:

Graphs 1 and 2: Distribution of estimated estimates and intercepts for each tree (for `tree_phylm`) or value generated within a given interval (`intra_phylm`) Red vertical line represents the mean estimate or intercept for all models.

Graph 3: Scatterplot with mean regression (black line) and standard deviation of the regression (blue dotted lines).

Author(s)

Caterina Penone and Gustavo Paterno

See Also

[ggplot](), [tree_phylm]() [intra_phylm]()

---

sensi_plot.sensiTree.TraitEvol

Graphical diagnostics for class 'sensiTree.TraitEvol'

Description

sensi_plot.sensiTree.TraitEvol Plot results from `tree_discrete` and `tree_continuous`.

Usage

```
## S3 method for class 'sensiTree.TraitEvol'
sensi_plot(x, graphs = "all", ...)
```

Arguments

x	output from `tree_fitDiscrete` or `tree_fitContinuous`
graphs	choose which graph should be printed in the output ("all", "q12", "q21", "aic" or" "optpar")
...	further arguments to methods

Author(s)

Gijsbert Werner

See Also

[ggplot](), [tree_discrete]() [tree_continuous]()

The following graphs are printed.

Graph aicc: Distribution of estimated AICc-values across each tree. Red vertical line represents the mean value among all estimates. Blue vertical line represents the median value among all estimates.

Graph optpar: Distribution of estimated values for optimisation parameter specified using 'transform' (if applicable) Red vertical line represents the mean value among all estimates. Blue vertical line represents the median value among all estimates.

Additionally, only for `tree_discrete` the function creates the following graphs.

Graph q12: Distribution of estimated parameter values for transition rates q12 for each tree. Red vertical line represents the mean value among all estimates. Blue vertical line represents the median value among all estimates.

Graph q21: Distribution of estimated parameter values for transition rates q21 for each tree. Red vertical line represents the mean value among all estimates. Blue vertical line represents the median value among all estimates.

While only for `tree_continuous` the function creates the following graphs.

Graph sigsq: Distribution of estimated parameter values for rate of evolution sigsq for each tree. Red vertical line represents the mean value among all estimates. Blue vertical line represents the median value among all estimates.

Graph z0: Distribution of estimated parameter values for z0 for each tree. Red vertical line represents the mean value among all estimates. Blue vertical line represents the median value among all estimates.

---

sensi_plot.sensiTree_Clade

Graphical diagnostics for class 'sensiTree_Clade'

Description

Plot results from `tree_clade_phylm` and `tree_clade_phyglm`

Usage

```
## S3 method for class 'sensiTree_Clade'
sensi_plot(x, clade = NULL, graphs = "all", ...)
```

Arguments

x	output from `tree_clade_phylm` or `tree_clade_phyglm`
clade	The name of the clade to be evaluated (see details)
graphs	Choose which graph should be printed on the output ("all", 1 or 2). Defaults to "all".
...	further arguments to methods.

Details

For 'x' from tree_clade_phylm or tree_clade_phyglm:

Graph 1: Estimated slopes after clade removal (reduced data) across multiple trees. Small dots represent estimates reruns between phylogenetic trees while larger dots represents the average estimate between all trees for each clade. The solid black line represents the average slope estimate among trees using the full dataset.

Graph 2: The effect of clade removal on slope estimate across all individual phylogenetic trees for each clade analyzed. The black line indicates estimates with the full dataset while the red line represent estimates without the focal clade (reduced data) across different trees. The blue dots represent null expectation estimates after removing the same number of species of the focal clade, with dots falling outside the red line area indicating a larger than expected absolute effect.

Author(s)

Gustavo Paterno, Caterina Penone

See Also

[tree_clade_phylm](#)

sensi_plot.sensiTree_Influ

Graphical diagnostics for class 'sensiTree_Influ'

Description

sensi_plot.sensiTree_Influ Plot results from `tree_influ_phylm` and `tree_influ_phyglm`

Usage

```
## S3 method for class 'sensiTree_Influ'
sensi_plot(x, graphs = "all", ...)
```

Arguments

x	output from `tree_influ_phylm` or `tree_influ_phyglm`
graphs	choose which graph should be printed on the output ("all", 1,2,3 or 4)
...	further arguments to methods

Details

For 'x' from sensiTree_Influ_phylm or sensiTree_Influ_phyglm:

Graph 1: Most common influential species on model estimates. Percentage of iterations (n.tree or n.intra) where the removal of individual species caused significant change in model estimate (sDIFestimate > cutoff).

Graph 2: Shift on model estimate after species removal for most influential species. Small red dots represent individual reruns between different trees or simulations while large red dots represent the average DIFestimate among all iterations.

Author(s)

Gustavo Paterno, Caterina Penone

See Also

ggplot

---

sensi_plot.sensiTree_Intra

Graphical diagnostics for class 'sensiIntra_Tree'

Description

`plot_tree.intra_phylm` Plot results from `tree_intra_phylm`, `tree_intra_phylm` and `tree_intra_phyglm`

Usage

```
## S3 method for class 'sensiTree_Intra'
sensi_plot(x, graphs = "all", uncer.type = "all", ...)
```

Arguments

x	output from `tree_intra_phylm`, `tree_intra_phyglm`
graphs	choose which graph should be printed in the output ("all", 1, 2 or 3)
uncer.type	choose which uncertainty type should be printed ("all", "intra", "tree")
...	further arguments to methods

Details

For 'x' from `tree_intra_phylm` or `tree_intra_phyglm`:

Graphs 1 and 2: Distribution of estimated estimates and intercepts for each tree (for `tree_phylm`) or value generated within a given interval (`interaction_intra_tree_phylm`) Red vertical line represents the mean estimate or intercept for all models.

Graph 3: Scatterplot with mean regression (black line) and standard deviation of the regression (blue dotted lines).

Author(s)

Caterina Penone and Gustavo Paterno

See Also

[ggplot](), [tree_phylm]() [intra_phylm]()

`sensi_plot.sensiTree_Samp`

Graphical diagnostics for class 'sensiTree_samp'

Description

`sensi_plot.sensiTree_Samp` Plot results from `sensiTree_Samp_phylm` and `sensiTree_Samp_phyglm`

Usage

```
## S3 method for class 'sensiTree_Samp'
sensi_plot(x, graphs = "all", ...)
```

Arguments

x	output from `sensiTree_Samp_phylm` or `sensiTree_Samp_phyglm`
graphs	choose which graph should be printed on the output ("all", 1,2,3 or 4)
...	further arguments to methods

Details

For 'x' from sensiTree_Samp_phylm or sensiTree_Samp_phyglm:

Graph 1: : Estimated estimates for each simulation across percentages of species removed. Small red dots represent individual estimates for each iteration (tree or intra) across percentages of removed species. Large dots represent the average among these estimates within each iteration and percentage of species removal. Different estimates within the same percentage interval represent different iterations (tree or intra).

Graph 2: : The percentage of significant estimates ($p < 0.05$) across the percentage of species removed. Small red dots represent individual iterations (tree or intra) while large dots represent the average among these iterations.

Note

If model = "BM", only plots 1, 2 and 4 are printed. Plot 3, phylogenetic model parameter is not available for model = "BM"

Author(s)

Gustavo Paterno

See Also

[ggplot](), [samp_phylm]() [samp_phyglm]()

---

sensi_plot.tree.bd   *Graphical diagnostics for class 'tree.physig'*

Description

sensi_plot_tree.bd Plot results from tree_bd.

Usage

```
## S3 method for class 'tree.bd'
sensi_plot(x, graphs = "all", ...)
```

Arguments

x	output from tree_bd
graphs	choose which graph should be printed in the output ("all", 1 or 2)
...	further arguments to methods

Details

For 'x' from tree_bd

Graphs 1: Distribution of estimated diversification or speciation rate for each tree. Red vertical line represents the average signal among all estimates.

Graphs 2: Estimates across each phylogenetic tree.

Author(s)

Caterina Penone and Gustavo Paterno

See Also

[ggplot](), [tree_phylm]() [intra_phylm]()

---

sensi_plot.tree.physig

                    Graphical diagnostics for class 'tree.physig'

Description

sensi_plot_tree.physig Plot results from tree_physig.

Usage

```
## S3 method for class 'tree.physig'
sensi_plot(x, graphs = "all", ...)
```

Arguments

x	output from `tree_physig`
graphs	choose which graph should be printed in the output ("all", 1 or 2)
...	further arguments to methods

Details

For 'x' from `tree_physig`

Graphs 1: Distribution of estimated phylogenetic signal (lambda or K) for each tree Red vertical line represents the average signal among all estimates.

Graph 2: Distribution of p-values for the phylogenetic signal (K or lambda) for each tree. Red vertical line represents the alpha significance level = 0.05.

Author(s)

Caterina Penone and Gustavo Paterno

See Also

`ggplot`, `tree_phylm` `intra_phylm`

---

tree_bd	*Phylogenetic uncertainty - net diversification rate*

Description

Performs estimates of diversification rate evaluating uncertainty in trees topology.

Usage

```
tree_bd(phy, n.tree = "all", method = "ms", track = F, ...)
```

Arguments

phy	A phylogeny (class 'multiPhylo', see ?ape).
n.tree	Number of times to repeat the analysis with n different trees picked randomly in the multiPhylo file. (If `n.tree = "all"`, diversification will be estimated among the set of trees provided in phy)
method	the method for estimating diversification rate ("ms" or "km") (see Details).
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to `phylosig`

Details

This function estimates net diversification rate using `bd.ms` (Magallon and Sanderson (2000) method) or speciation rate using `bd.km` (Kendall-Moran method) for n trees, randomly picked from a multi-Phylo file.

Output can be visualised using `sensi_plot`.

Value

The function `tree_bd` returns a list with the following components:

`tree.bd.estimates`: Three number, diversification/speciation rate estimate ("Magallon and Sanderson" or "Kendall-Moran") for each run with a different phylogenetic tree.

`stats`: Main statistics for estimates across trees.`CI_low` and `CI_high` are the lower and upper limits of the 95

Author(s)

Gustavo Paterno

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Magallon S and MJ Sanderson. 2000. Absolute diversification rates in angiosperm clades. Evolution 55:1762-1780.

See Also

`bd.ms`, `tree_phylm`,`sensi_plot`

Examples

```
data("primates")
# To estimate diversification rate with Magallon and Sanderson method:
fit <- tree_bd(phy = primates.phy, n.tree = 30, method = "ms")
summary(fit)
sensi_plot(fit)
# To estimate speciation rate Kendall-Moran method
fit <- tree_bd(phy = primates.phy, n.tree = 30, method = "km")
summary(fit)
sensi_plot(fit)
```

| tree_clade_phyglm | *Interaction between phylogenetic uncertainty and influential clade detection - Phylogenetic Logistic Regression* |

Description

Estimate the impact on model estimates of phylogenetic logistic regression after removing clades from the analysis and evaluating uncertainty in trees topology.

Usage

```
tree_clade_phyglm(
  formula,
  data,
  phy,
  clade.col,
  n.species = 5,
  n.sim = 100,
  n.tree = 2,
  btol = 50,
  track = TRUE,
  ...
)
```

Arguments

formula	The model formula
data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'multiPhylo', see ?ape).
clade.col	The column in the provided data frame which specifies the clades (a character vector with clade names).
n.species	Minimum number of species in a clade for the clade to be included in the leave-one-out deletion analysis. Default is 5.
n.sim	Number of simulations for the randomization test.
n.tree	Number of times to repeat the analysis with n different trees picked randomly in the multiPhylo file. If NULL, n.tree = 2
btol	Bound on searching space. For details see phyloglm
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to phyloglm

Details

Currently only logistic regression using the "logistic_MPLE"-method from phyloglm is implemented.

This function sequentially removes one clade at a time, fits a phylogenetic logistic regression model using [phyloglm](#) and stores the results. The impact of of a specific clade on model estimates is calculated by the comparison between the full model (with all species) and the model without the species belonging to a clade. It repeats this operation using n trees, randomly picked in a multiPhylo file.

Additionally, to account for the influence of the number of species on each clade (clade sample size), this function also estimates a null distribution of slopes expected for the number of species in a given clade. This is done by fitting models without the same number of species in the given clade. The number of simulations to be performed is set by 'n.sim'. To test if the clade influence differs from the null expectation for a clade of that size, a randomization test can be performed using 'summary(x)'.

clade_phyglm detects influential clades based on difference in intercept and/or slope when removing a given clade compared to the full model including all species. This is done for n trees in the multiphylo file.

Currently, this function can only implements simple logistic models (i.e. $trait\ predictor$). In the future we will implement more complex models.

Output can be visualised using sensi_plot.

Value

The function clade_phyglm returns a list with the following components:

formula: The formula

full.model.estimates: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. lambda) for the full model without deleted species.

sensi.estimates: A data frame with all simulation estimates. Each row represents a deleted clade for a tree iteration. Columns report the calculated regression intercept (intercept), difference between simulation intercept and full model intercept (DIFintercept), the percentage of change in intercept compared to the full model (intercept.perc) and intercept p-value (pval.intercept). All these parameters are also reported for the regression slope (DIFestimate etc.). Additionally, model aic value (AIC) and the optimised value (optpar) of the phylogenetic parameter (e.g. kappa or lambda, depending on the phylogenetic model used) are reported.

null.dist: A data frame with estimates for the null distributions for all clades analysed.

data: Original full dataset.

errors: Clades and/or trees where deletion resulted in errors.

Author(s)

Gustavo Paterno, Caterina Penone & Gijsbert D.A. Werner

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phyloglm, tree_phyglm, clade_phyglm, tree_clade_phylm, sensi_plot

Examples

```
## Not run:
# Simulate Data:
set.seed(6987)
mphy = rmtree(150, N = 30)
x = rTrait(n=1,phy=mphy[[1]])
X = cbind(rep(1,150),x)
y = rbinTrait(n=1,phy=mphy[[1]], beta=c(-1,0.5), alpha=.7 ,X=X)
cla <- rep(c("A","B","C","D","E"), each = 30)
dat = data.frame(y, x, cla)
# Run sensitivity analysis:
tree_clade <- tree_clade_phyglm(y ~ x, phy = mphy, data = dat,
n.tree = 10, n.sim = 10, clade.col = "cla")
# To check summary results and most influential clades:
summary(tree_clade)
# Visual diagnostics for clade removal:
sensi_plot(tree_clade)
# Specify which clade removal to plot:
sensi_plot(tree_clade, "B")
sensi_plot(tree_clade, "C", graphs = 2)
sensi_plot(tree_clade, "D", graphs = 2)

## End(Not run)
```

tree_clade_phylm	*Interaction between phylogenetic uncertainty and influential clade detection - Phylogenetic Linear Regression*

Description

Estimate the impact on model estimates of phylogenetic linear regression after removing clades from the analysis and evaluating uncertainty in trees topology.

Usage

```
tree_clade_phylm(
  formula,
  data,
  phy,
  clade.col,
  n.species = 5,
  n.sim = 100,
  n.tree = 2,
  model = "lambda",
  track = TRUE,
  ...
)
```

Arguments

formula	The model formula
data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'multiPhylo', see ?ape).
clade.col	The column in the provided data frame which specifies the clades (a character vector with clade names).
n.species	Minimum number of species in a clade for the clade to be included in the leave-one-out deletion analysis. Default is 5.
n.sim	Number of simulations for the randomization test.
n.tree	Number of times to repeat the analysis with n different trees picked randomly in the multiPhylo file. If NULL, n.tree = 2
model	The phylogenetic model to use (see Details). Default is lambda.
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to phylolm

Details

This function sequentially removes one clade at a time, fits a phylogenetic linear regression model using phylolm and stores the results. The impact of of a specific clade on model estimates is calculated by the comparison between the full model (with all species) and the model without the species belonging to a clade. It repeats this operation using n trees, randomly picked in a multiPhylo file.

Additionally, to account for the influence of the number of species on each clade (clade sample size), this function also estimate a null distribution of slopes expected for the number of species in a given clade. This is done by fitting models without the same number of species in the given clade. The number of simulations to be performed is set by 'n.sim'. To test if the clade influence differs from the null expectation for a clade of that size, a randomization test can be performed using 'summary(x)'.

All phylogenetic models from phylolm can be used, i.e. BM, OUfixedRoot, OUrandomRoot, lambda, kappa, delta, EB and trend. See ?phylolm for details.

`clade_phylm` detects influential clades based on difference in intercept and/or slope when removing a given clade compared to the full model including all species. This is done for n trees in the multiphylo file.

Currently, this function can only implement simple linear models (i.e. $y = a + bx$). In the future we will implement more complex models.

Output can be visualised using `sensi_plot`.

Value

The function `clade_phylm` returns a list with the following components:

`formula`: The formula

`full.model.estimates`: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. `lambda`) for the full model without deleted species.

`sensi.estimates`: A data frame with all simulation estimates. Each row represents a deleted clade for a tree iteration. Columns report the calculated regression intercept (`intercept`), difference between simulation intercept and full model intercept (`DIFintercept`), the percentage of change in intercept compared to the full model (`intercept.perc`) and intercept p-value (`pval.intercept`). All these parameters are also reported for the regression slope (`DIFestimate` etc.). Additionally, model aic value (`AIC`) and the optimised value (`optpar`) of the phylogenetic parameter (e.g. `kappa` or `lambda`, depending on the phylogenetic model used) are reported.

`null.dist`: A data frame with estimates for the null distributions for all clades analysed.

`data`: Original full dataset.

`errors`: Clades and/or trees where deletion resulted in errors.

Author(s)

Gustavo Paterno, Caterina Penone & Gijsbert D.A. Werner

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

`phylolm`, `tree_phylm`, `clade_phylm`, `tree_clade_phyglm`, `sensi_plot`

Examples

```
## Not run:
# Load data:
data(primates)
# run analysis:
clade_tree <- tree_clade_phylm(log(sexMaturity) ~ log(adultMass),
phy = primates$phy, data = primates$data, clade.col = "family", n.sim = 50, n.tree = 5)
```

```
# To check summary results and most influential clades:
summary(clade_tree)
# Visual diagnostics for clade removal:
sensi_plot(clade_tree)
# Specify which clade removal to plot:
sensi_plot(clade_tree)
sensi_plot(clade_tree, "Cercopithecidae")
sensi_plot(clade_tree, clade = "Cebidae", graphs = 2)

## End(Not run)
```

---

tree_continuous   *Phylogenetic uncertainty - Trait Evolution Continuous Characters*

Description

Fits models for trait evolution of continuous characters, evaluating phylogenetic uncertainty.

Usage

```
tree_continuous(
  data,
  phy,
  n.tree = 10,
  model,
  bounds = list(),
  n.cores = NULL,
  track = TRUE,
  ...
)
```

Arguments

data	Data vector for a single continuous trait, with names matching tips in phy.
phy	Phylogenies (class 'multiPhylo', see ?ape).
n.tree	Number of times to repeat the analysis with n different trees picked randomly in the multiPhylo file. If NULL, n.tree = 10
model	The evolutionary model (see Details).
bounds	settings to constrain parameter estimates. See [fitContinuous](#)
n.cores	number of cores to use. If 'NULL', number of cores is detected.
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to [fitContinuous](#)

Details

This function fits different models of continuous character evolution using `fitContinuous` to n trees, randomly picked in a multiPhylo file.

Different evolutionary models from fitContinuous can be used, i.e. `BM`,`OU`, `EB`, `trend`, `lambda`, `kappa`, `delta` and `drift`.

See `fitContinuous` for more details on character models and tree transformations.

Output can be visualised using `sensi_plot`.

Value

The function `tree_continuous` returns a list with the following components:

`call`: The function call

`data`: The original full data vector

`sensi.estimates`: (rate of evolution `sigsq`, root state `z0` and where applicable `optpar`), AICc and the optimised value of the phylogenetic transformation parameter (e.g. `lambda`) for each analysis with a different phylogenetic tree.

`N.tree`: Number of trees `n.tree` analysed

`stats`: Main statistics for model parameters, i.e. minimum, maximum, mean, median and sd-values

`optpar`: Evolutionary model used (e.g. `lambda`, `kappa` etc.)

Author(s)

Gijsbert Werner & Caterina Penone

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Yang Z. 2006. Computational Molecular Evolution. Oxford University Press: Oxford.

Harmon Luke J, Jason T Weir, Chad D Brock, Richard E Glor, and Wendell Challenger. 2008. GEIGER: investigating evolutionary radiations. Bioinformatics 24:129-131.

See Also

`fitContinuous`

Examples

```
## Not run:
#Load data:
data("primates")
#Model trait evolution accounting for phylogenetic uncertainty
adultMass<-primates$data$adultMass
names(adultMass)<-rownames(primates$data)
tree_cont<-tree_continuous(data = adultMass,phy = primates$phy,
```

```
model = "OU",n.tree=30,n.cores = 2,track = TRUE)
#Print summary statistics
summary(tree_cont)
sensi_plot(tree_cont)
sensi_plot(tree_cont,graphs="sigsq")
sensi_plot(tree_cont,graphs="optpar")
#Use a different evolutionary model
tree_cont2<-tree_continuous(data = adultMass,phy = primates$phy,
model = "delta",n.tree=30,n.cores = 2,track = TRUE)
summary(tree_cont2)
sensi_plot(tree_cont2)

## End(Not run)
```

---

tree_discrete   *Phylogenetic uncertainty - Trait Evolution Discrete Characters*

Description

Fits models for trait evolution of discrete (binary) characters, evaluating phylogenetic uncertainty.

Usage

```
tree_discrete(
  data,
  phy,
  n.tree = 10,
  model,
  transform = "none",
  bounds = list(),
  n.cores = NULL,
  track = TRUE,
  ...
)
```

Arguments

data	Data vector for a single binary trait, with names matching tips in phy.
phy	Phylogenies (class 'multiPhylo', see ?ape).
n.tree	Number of times to repeat the analysis with n different trees picked randomly in the multiPhylo file. If NULL, n.tree = 10
model	The Mkn model to use (see Details).
transform	The evolutionary model to transform the tree (see Details). Default is none.
bounds	settings to constrain parameter estimates. See [fitDiscrete](#)
n.cores	number of cores to use. If 'NULL', number of cores is detected.
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to [fitDiscrete](#)

Details

This function fits different models of discrete character evolution using `fitDiscrete` to n trees, randomly picked in a multiPhylo file. Currently, only binary discrete traits are supported

Different character model from `fitDiscrete` can be used, including ER (equal-rates), SYM (symmetric), ARD (all-rates-different) and `meristic` (stepwise fashion).

All transformations to the phylogenetic tree from `fitDiscrete` can be used, i.e. none, EB, `lambda`, `kappa` and`delta`.

See `fitDiscrete` for more details on character models and tree transformations.

Output can be visualised using `sensi_plot`.

Value

The function `tree_discrete` returns a list with the following components:

`call`: The function call

`data`: The original full data vector

`sensi.estimates`: Parameter estimates (transition rates q12 and q21), AICc and the optimised value of the phylogenetic transformation parameter (e.g. `lambda`) for each analysis with a different phylogenetic tree.

`N.tree`: Number of trees n.tree analysed

`stats`: Main statistics for model parameters, i.e. minimum, maximum, mean, median and sd-values

`optpar`: Transformation parameter used (e.g. `lambda`, `kappa` etc.)

Author(s)

Gijsbert Werner & Caterina Penone

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Yang Z. 2006. Computational Molecular Evolution. Oxford University Press: Oxford.

Harmon Luke J, Jason T Weir, Chad D Brock, Richard E Glor, and Wendell Challenger. 2008. GEIGER: investigating evolutionary radiations. Bioinformatics 24:129-131.

See Also

`fitDiscrete`

Examples

```
## Not run:
#Load data:
data("primates")
#Create a binary trait factor
adultMass_binary<-ifelse(primates$data$adultMass > 7350, "big", "small")
adultMass_binary<-as.factor(as.factor(adultMass_binary))
```

```
names(adultMass_binary)<-rownames(primates$data)
#Model trait evolution accounting for phylogenetic uncertainty
tree_binary<-tree_discrete(data = adultMass_binary,phy = primates$phy,
model = "ARD",transform = "none",n.tree = 30,n.cores = 2,track = TRUE)
#Print summary statistics
summary(tree_binary)
sensi_plot(tree_binary)
sensi_plot(tree_binary,graphs="q12")
sensi_plot(tree_binary,graphs="q21")
#Use a different evolutionary model or transformation.
tree_binary_lambda<-tree_discrete(data = adultMass_binary,phy = primates$phy,
model = "SYM",transform = "lambda",n.tree = 30,n.cores = 2,track = TRUE)
summary(tree_binary_lambda) #Using Pagel's Lambda
sensi_plot(tree_binary_lambda)
#Symmetrical rates, with an Early Burst (EB) model of trait evolution
tree_binary_SYM_EB<-tree_discrete(data = adultMass_binary,phy = primates$phy,
model = "SYM",transform = "EB",n.tree = 30,n.cores = 2,track = TRUE)
summary(tree_binary_SYM_EB)
sensi_plot(tree_binary_SYM_EB)
sensi_plot(tree_binary_SYM_EB,graphs="optpar")

## End(Not run)
```

---

tree_influ_phyglm	*Interaction between phylogenetic uncertainty and influential species detection - Phylogenetic Logistic Regression*

Description

Performs leave-one-out deletion analysis for phylogenetic logistic regression, and detects influential species while evaluating uncertainty in trees topology.

Usage

```
tree_influ_phyglm(
  formula,
  data,
  phy,
  n.tree = 2,
  cutoff = 2,
  btol = 50,
  track = TRUE,
  ...
)
```

Arguments

formula	The model formula
data	Data frame containing species traits with row names matching tips in phy.

phy	A phylogeny (class 'phylo') matching `data`.
n.tree	Number of times to repeat the analysis with n different trees picked randomly in the multiPhylo file.
cutoff	The cutoff value used to identify for influential species (see Details)
btol	Bound on searching space. For details see `phyloglm`
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to `phyloglm`

Details

This function sequentially removes one species at a time, fits a phylogenetic logistic regression model using [phyloglm](), stores the results and detects influential species. It repeats this operation using n trees, randomly picked in a multiPhylo file.

Currently only logistic regression using the "logistic_MPLE"-method from `phyloglm` is implemented.

`influ_phyglm` detects influential species based on the standardised difference in intercept and/or slope when removing a given species compared to the full model including all species. Species with a standardised difference above the value of `cutoff` are identified as influential. The default value for the cutoff is 2 standardised differences change.

Currently, this function can only implement simple logistic models (i.e. $trait\ predictor$). In the future we will implement more complex models.

Output can be visualised using `sensi_plot`.

Value

The function `influ_phyglm` returns a list with the following components:

cutoff: The value selected for `cutoff`

formula: The formula

full.model.estimates: Coefficients, aic and the optimised value of the phylogenetic parameter (i.e. alpha) for the full model without deleted species.

influential_species: List of influential species, both based on standardised difference in interecept and in the slope of the regression. Species are ordered from most influential to less influential and only include species with a standardised difference > cutoff.

sensi.estimates: A data frame with all simulation estimates. Each row represents a deleted clade for a given random tree. Columns report the calculated regression intercept (intercept), difference between simulation intercept and full model intercept (DIFintercept), the standardised difference (sDIFintercept), the percentage of change in intercept compared to the full model (intercept.perc) and intercept p-value (pval.intercept). All these parameters are also reported for the regression slope (DIFestimate etc.). Additionally, model aic value (AIC) and the optimised value (optpar) of the phylogenetic parameter (i.e. alpha) are reported.

data: Original full dataset.

errors: Species where deletion resulted in errors.

Author(s)

Gustavo Paterno, Caterina Penone & Gijsbert D.A. Werner

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phyloglm, tree_phyglm, influ_phyglm, tree_influ_phyglm, sensi_plot

Examples

```
## Not run:
# Simulate Data:
set.seed(6987)
mphy = rmtree(100, N = 30)
x = rTrait(n=1,phy=mphy[[1]])
X = cbind(rep(1,100),x)
y = rbinTrait(n=1,phy=mphy[[1]], beta=c(-1,0.5), alpha=.7 ,X=X)
dat = data.frame(y, x)
# Run sensitivity analysis:
tree_influ <- tree_influ_phyglm(y ~ x, data = dat, phy = mphy, n.tree = 5)
summary(tree_influ)
sensi_plot(tree_influ)
sensi_plot(tree_influ, graphs = 1)
sensi_plot(tree_influ, graphs = 2)

## End(Not run)
```

---

tree_influ_phylm	*Interaction between phylogenetic uncertainty and influential species detection - Phylogenetic Linear Regression*

Description

Performs leave-one-out deletion analysis for phylogenetic linear regression, and detects influential species while evaluating uncertainty in trees topology.

Usage

```
tree_influ_phylm(
  formula,
  data,
  phy,
```

```
    n.tree = 2,
    cutoff = 2,
    model = "lambda",
    track = TRUE,
    ...
)
```

Arguments

formula	The model formula
data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
n.tree	Number of times to repeat the analysis with n different trees picked randomly in the multiPhylo file.
cutoff	The cutoff value used to identify for influential species (see Details)
model	The phylogenetic model to use (see Details). Default is lambda.
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to phylolm

Details

This function sequentially removes one species at a time, fits a phylogenetic linear regression model using [phylolm](), stores the results and detects influential species. It repeats this operation using n trees, randomly picked in a multiPhylo file.

All phylogenetic models from phylolm can be used, i.e. BM, OUfixedRoot, OUrandomRoot, lambda, kappa, delta, EB and trend. See ?phylolm for details.

influ_phylm detects influential species based on the standardised difference in intercept and/or slope when removing a given species compared to the full model including all species. Species with a standardised difference above the value of cutoff are identified as influential. The default value for the cutoff is 2 standardised differences change.

Currently, this function can only implement simple linear models (i.e. $trait\ predictor$). In the future we will implement more complex models.

Output can be visualised using sensi_plot.

Value

The function influ_phylm returns a list with the following components:

cutoff: The value selected for cutoff

formula: The formula

full.model.estimates: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. lambda) for the full model without deleted species.

influential_species: List of influential species, both based on standardised difference in intercept and in the slope of the regression. Species are ordered from most influential to less influential and only include species with a standardised difference > cutoff.

sensi.estimates: A data frame with all simulation estimates. Each row represents a deleted clade for a given random tree. Columns report the calculated regression intercept (intercept), difference between simulation intercept and full model intercept (DIFintercept), the standardised difference (sDIFintercept), the percentage of change in intercept compared to the full model (intercept.perc) and intercept p-value (pval.intercept). All these parameters are also reported for the regression slope (DIFestimate etc.). Additionally, model aic value (AIC) and the optimised value (optpar) of the phylogenetic parameter (e.g. kappa or lambda, depending on the phylogenetic model used) are reported.

data: Original full dataset.

errors: Species where deletion resulted in errors.

Author(s)

Gustavo Paterno, Caterina Penone & Gijsbert D.A. Werner

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phylolm, tree_phylm, influ_phylm, tree_influ_phyglm, sensi_plot

Examples

```
## Not run:
# Load data:
data(alien)
# run analysis:
tree_influ <- tree_influ_phylm(log(gestaLen) ~ log(adultMass), phy = alien$phy,
data = alien$data, n.tree = 5)
# To check summary results:
summary(tree_influ)
# Visual diagnostics
sensi_plot(tree_influ)
sensi_plot(tree_influ, graphs = 1)
sensi_plot(tree_influ, graphs = 2)

data(alien)
tree_influ <- tree_influ_phylm(log(gestaLen) ~ log(adultMass), phy = alien$phy,
data = alien$data[1:25, ], n.tree = 2)
summary(tree_influ)

## End(Not run)
```

---

tree_intra_phyglm | *Interaction between phylogenetic uncertainty and intraspecific variability - Phylogenetic logistic Regression*

Description

Performs Phylogenetic logistic regression evaluating intraspecific variability in response and/or predictor variables and uncertainty in trees topology.

Usage

```
tree_intra_phyglm(
  formula,
  data,
  phy,
  Vx = NULL,
  x.transf = NULL,
  n.intra = 10,
  n.tree = 2,
  distrib = "normal",
  track = TRUE,
  btol = 50,
  ...
)
```

Arguments

formula	The model formula: response~predictor.
data	Data frame containing species traits and species names as row names.
phy	A phylogeny (class 'phylo', see ?ape).
Vx	Name of the column containing the standard deviation or the standard error of the predictor variable. When information is not available for one taxon, the value can be 0 or NA
x.transf	Transformation for the predictor variable (e.g. log or sqrt). Please use this argument instead of transforming data in the formula directly (see also details below).
n.intra	Number of times to repeat the analysis generating a random value for response and/or predictor variables. If NULL, n.intra = 30
n.tree	Number of times to repeat the analysis with n different trees picked randomly in the multiPhylo file. If NULL, n.tree = 2
distrib	A character string indicating which distribution to use to generate a random value for the response and/or predictor variables. Default is normal distribution: "normal" (function [rnorm](#)). Uniform distribution: "uniform" ([runif](#)) Warning: we recommend to use normal distribution with Vx or Vy = standard deviation of the mean.

track	Print a report tracking function progress (default = TRUE)
btol	Bound on searching space. For details see phyloglm
...	Further arguments to be passed to phyloglm

Details

This function fits a phylogenetic logistic regression model using [phyloglm](#) to n trees (`n.tree`), randomly picked in a multiPhylo file. The regression is also repeated `n.intra` times. At each iteration the function generates a random value for each row in the dataset using the standard deviation or errors supplied and assuming a normal or uniform distribution. To calculate means and se for your raw data, you can use the `summarySE` function from the package Rmisc.

#' All phylogenetic models from phyloglm can be used, i.e. `BM`, `OUfixedRoot`, `OUrandomRoot`, `lambda`, `kappa`, `delta`, `EB` and `trend`. See ?phyloglm for details.

Currently, this function can only implement simple logistic models (i.e. $trait\ predictor$). In the future we will implement more complex models.

Output can be visualised using `sensi_plot`.

Value

The function `tree_intra_phylm` returns a list with the following components:

`formula`: The formula

`data`: Original full dataset

`sensi.estimates`: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. `lambda`) for each regression using a value in the interval of variation and a different phylogenetic tree.

`N.obs`: Size of the dataset after matching it with tree tips and removing NA's.

`stats`: Main statistics for model parameters.`CI_low` and `CI_high` are the lower and upper limits of the 95

`all.stats`: Complete statistics for model parameters. Fields coded using `all` describe statistics due to both intraspecific variation and phylogenetic uncertainty. Fields coded using `intra` describe statistics due to intraspecific variation only. Fields coded using `tree` describe statistics due to phylogenetic uncertainty only. `sd` is the standard deviation. `CI_low` and `CI_high` are the lower and upper limits of the 95

`sp.pb`: Species that caused problems with data transformation (see details above).

Warning

When Vy or Vx exceed Y or X, respectively, negative (or null) values can be generated, this might cause problems for data transformation (e.g. log-transformation). In these cases, the function will skip the simulation. This problem can be solved by increasing `times`, changing the transformation type and/or checking the target species in output$sp.pb.

Author(s)

Caterina Penone & Pablo Ariel Martinez

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Martinez, P. a., Zurano, J.P., Amado, T.F., Penone, C., Betancur-R, R., Bidau, C.J. & Jacobina, U.P. (2015). Chromosomal diversity in tropical reef fishes is related to body size and depth range. Molecular Phylogenetics and Evolution, 93, 1-4

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phyloglm, tree_phyglm, intra_phyglm, tree_intra_phylm, sensi_plot

Examples

```
# Simulate data
set.seed(6987)
mphy = ape::rmtree(150, N = 30)
x = phylolm::rTrait(n=1,phy=mphy[[1]])
x_sd = rnorm(150,mean = 0.8,sd=0.2)
X = cbind(rep(1,150),x)
y = rbinTrait(n=1,phy=mphy[[1]], beta=c(-1,0.5), alpha=.7 ,X=X)
dat = data.frame(y, x, x_sd)
intra.tree <- tree_intra_phyglm(y ~ x, data = dat, phy = mphy, n.intra = 3,
                                      n.tree = 3, Vx = "x_sd")
# summary results:
summary(intra.tree)
# Visual diagnostics for phylogenetic uncertainty:
sensi_plot(intra.tree, uncer.type = "all") #or uncer.type = "tree", uncer.type = "intra"
```

---

tree_intra_phylm	*Interaction between phylogenetic uncertainty and intraspecific variability - Phylogenetic Linear Regression*

Description

Performs Phylogenetic linear regression evaluating intraspecific variability in response and/or predictor variables and uncertainty in trees topology.

Usage

```
tree_intra_phylm(
  formula,
  data,
  phy,
  Vy = NULL,
  Vx = NULL,
```

```
    y.transf = NULL,
    x.transf = NULL,
    n.intra = 10,
    n.tree = 2,
    distrib = "normal",
    model = "lambda",
    track = TRUE,
    ...
)
```

Arguments

formula	The model formula: `response~predictor`.
data	Data frame containing species traits and species names as row names.
phy	A phylogeny (class 'phylo', see ?ape).
Vy	Name of the column containing the standard deviation or the standard error of the response variable. When information is not available for one taxon, the value can be 0 or `NA`.
Vx	Name of the column containing the standard deviation or the standard error of the predictor variable. When information is not available for one taxon, the value can be 0 or `NA`
y.transf	Transformation for the response variable (e.g. `log` or `sqrt`). Please use this argument instead of transforming data in the formula directly (see also details below).
x.transf	Transformation for the predictor variable (e.g. `log` or `sqrt`). Please use this argument instead of transforming data in the formula directly (see also details below).
n.intra	Number of times to repeat the analysis generating a random value for response and/or predictor variables. If NULL, `n.intra = 30`
n.tree	Number of times to repeat the analysis with n different trees picked randomly in the multiPhylo file. If NULL, `n.tree = 2`
distrib	A character string indicating which distribution to use to generate a random value for the response and/or predictor variables. Default is normal distribution: "normal" (function [rnorm](#)). Uniform distribution: "uniform" ([runif](#)) Warning: we recommend to use normal distribution with Vx or Vy = standard deviation of the mean.
model	The phylogenetic model to use (see Details). Default is `lambda`.
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to `phylolm`

Details

This function fits a phylogenetic linear regression model using [phylolm](#) to n trees (n.tree), randomly picked in a multiPhylo file. The regression is also repeated `n.intra` times. At each iteration the function generates a random value for each row in the dataset using the standard deviation or

errors supplied and assuming a normal or uniform distribution. To calculate means and se for your raw data, you can use the `summarySE` function from the package `Rmisc`.

#' All phylogenetic models from `phylolm` can be used, i.e. `BM`, `OUfixedRoot`, `OUrandomRoot`, `lambda`, `kappa`, `delta`, `EB` and `trend`. See ?phylolm for details.

Currently, this function can only implement simple linear models (i.e. $trait \ predictor$). In the future we will implement more complex models.

Output can be visualised using `sensi_plot`.

Value

The function `tree_intra_phylm` returns a list with the following components:

`formula`: The formula

`data`: Original full dataset

`sensi.estimates`: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. `lambda`) for each regression using a value in the interval of variation and a different phylogenetic tree.

`N.obs`: Size of the dataset after matching it with tree tips and removing NA's.

`stats`: Main statistics for model parameters.`CI_low` and `CI_high` are the lower and upper limits of the 95

`all.stats`: Complete statistics for model parameters. Fields coded using `all` describe statistics due to both intraspecific variation and phylogenetic uncertainty. Fields coded using `intra` describe statistics due to intraspecific variation only. Fields coded using `tree` describe statistics due to phylogenetic uncertainty only. `sd` is the standard deviation. `CI_low` and `CI_high` are the lower and upper limits of the 95

`sp.pb`: Species that caused problems with data transformation (see details above).

Warning

When Vy or Vx exceed Y or X, respectively, negative (or null) values can be generated, this might cause problems for data transformation (e.g. log-transformation). In these cases, the function will skip the simulation. This problem can be solved by increasing `times`, changing the transformation type and/or checking the target species in output$sp.pb.

Author(s)

Caterina Penone & Pablo Ariel Martinez

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Martinez, P. a., Zurano, J.P., Amado, T.F., Penone, C., Betancur-R, R., Bidau, C.J. & Jacobina, U.P. (2015). Chromosomal diversity in tropical reef fishes is related to body size and depth range. Molecular Phylogenetics and Evolution, 93, 1-4

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phylolm, tree_phylm, intra_phylm, tree_intra_phyglm, sensi_plot

Examples

```
# Load data:
data(alien)
# run PGLS accounting for intraspecific and phylogenetic variation:
intra.tree <- tree_intra_phylm(gestaLen ~ adultMass, data = alien$data, phy = alien$phy,
Vy = "SD_gesta", n.intra = 3, n.tree = 3, y.transf = log, x.transf = log)
# To check summary results:
summary(intra.tree)
# Visual diagnostics
sensi_plot(intra.tree, uncer.type = "all") #or uncer.type = "tree", uncer.type = "intra"
```

---

tree_phyglm   *Phylogenetic uncertainty - Phylogenetic Logistic Regression*

Description

Performs Phylogenetic logistic regression evaluating uncertainty in trees topology.

Usage

```
tree_phyglm(formula, data, phy, n.tree = 2, btol = 50, track = TRUE, ...)
```

Arguments

formula	The model formula
data	Data frame containing species traits with species as row names.
phy	A phylogeny (class 'multiPhylo', see ?ape).
n.tree	Number of times to repeat the analysis with n different trees picked randomly in the multiPhylo file. If NULL, n.tree = 2
btol	Bound on searching space. For details see phyloglm.
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to phyloglm

Details

This function fits a phylogenetic logistic regression model using phyloglm to n trees, randomly picked in a multiPhylo file.

Currently, this function can only implement simple logistic models (i.e. $trait\ predictor$). In the future we will implement more complex models.

Output can be visualised using sensi_plot.

Value

The function `tree_phyglm` returns a list with the following components:

`formula`: The formula

`data`: Original full dataset

`sensi.estimates`: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. `lambda`) for each regression with a different phylogenetic tree.

`N.obs`: Size of the dataset after matching it with tree tips and removing NA's.

`stats`: Main statistics for model parameters.`CI_low` and `CI_high` are the lower and upper limits of the 95

`all.stats`: Complete statistics for model parameters. `sd_intra` is the standard deviation due to intraspecific variation. `CI_low` and `CI_high` are the lower and upper limits of the 95

Author(s)

Caterina Penone & Pablo Ariel Martinez

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Donoghue, M.J. & Ackerly, D.D. (1996). Phylogenetic Uncertainties and Sensitivity Analyses in Comparative Biology. Philosophical Transactions: Biological Sciences, pp. 1241-1249.

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phyloglm, sensi_plot,tree_phylm

Examples

```
### Simulating Data:
set.seed(6987)
mphy = rmtree(150, N = 30)
x = rTrait(n=1,phy=mphy[[1]])
X = cbind(rep(1,150),x)
y = rbinTrait(n=1,phy=mphy[[1]], beta=c(-1,0.5), alpha=.7 ,X=X)
dat = data.frame(y, x)
# Run sensitivity analysis:
tree <- tree_phyglm(y ~ x, data = dat, phy = mphy, n.tree = 30)
# summary results:
summary(tree)
# Visual diagnostics for phylogenetic uncertainty:
sensi_plot(tree)
```

---

tree_phylm   *Phylogenetic uncertainty - Phylogenetic Linear Regression*

Description

Performs Phylogenetic linear regression evaluating uncertainty in trees topology.

Usage

```
tree_phylm(formula, data, phy, n.tree = 2, model = "lambda", track = TRUE, ...)
```

Arguments

formula	The model formula
data	Data frame containing species traits with species as row names.
phy	A phylogeny (class 'multiPhylo', see ?ape).
n.tree	Number of times to repeat the analysis with n different trees picked randomly in the multiPhylo file. If NULL, n.tree = 2
model	The phylogenetic model to use (see Details). Default is lambda.
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to phylolm

Details

This function fits a phylogenetic linear regression model using [phylolm](#) to n trees, randomly picked in a multiPhylo file.

All phylogenetic models from phylolm can be used, i.e. BM, OUfixedRoot, OUrandomRoot, lambda, kappa, delta, EB and trend. See ?phylolm for details.

Currently, this function can only implement simple linear models (i.e. $trait\ predictor$). In the future we will implement more complex models.

Output can be visualised using sensi_plot.

Value

The function tree_phylm returns a list with the following components:

formula: The formula

data: Original full dataset

sensi.estimates: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. lambda) for each regression with a different phylogenetic tree.

N.obs: Size of the dataset after matching it with tree tips and removing NA's.

stats: Main statistics for model parameters.CI_low and CI_high are the lower and upper limits of the 95

all.stats: Complete statistics for model parameters. sd_intra is the standard deviation due to intraspecific variation. CI_low and CI_high are the lower and upper limits of the 95

Author(s)

Caterina Penone & Pablo Ariel Martinez

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Donoghue, M.J. & Ackerly, D.D. (1996). Phylogenetic Uncertainties and Sensitivity Analyses in Comparative Biology. Philosophical Transactions: Biological Sciences, pp. 1241-1249.

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phylolm, sensi_plot, tree_phyglm

Examples

```
# Load data:
data(alien)
# This analysis needs a multiphylo file:
class(alien$phy)
alien$phy
# run PGLS accounting for phylogenetic uncertain:
tree <- tree_phylm(log(gestaLen) ~ log(adultMass), phy = alien$phy,
data = alien$data, n.tree = 30)
# To check summary results:
summary(tree)
# Visual diagnostics
sensi_plot(tree)
# You can specify which graph to print:
sensi_plot(tree, graphs = 3)
```

---

tree_physig	*Phylogenetic uncertainty - Phylogenetic signal*

Description

Performs phylogenetic signal estimates evaluating uncertainty in trees topology.

Usage

```
tree_physig(
  trait.col,
  data,
  phy,
  n.tree = "all",
  method = "K",
```

```
    track = TRUE,
    ...
)
```

Arguments

`trait.col`	The name of a column in the provided data frame with trait to be analyzed (e.g. "Body_mass").
`data`	Data frame containing species traits with row names matching tips in phy.
`phy`	A phylogeny (class 'phylo') matching `data`.
`n.tree`	Number of times to repeat the analysis with n different trees picked randomly in the multiPhylo file. (If `n.tree = "all"`, phylogenetic signal will be estimated among the all set of trees provided in phy)
`method`	Method to compute signal: can be "K" or "lambda".
`track`	Print a report tracking function progress (default = TRUE)
`...`	Further arguments to be passed to `phylosig`

Details

This function estimates phylogenetic signal using phylosig to n trees, randomly picked in a multi-Phylo file.

Output can be visualised using `sensi_plot`.

Value

The function `tree_physig` returns a list with the following components:

`Trait`: Column name of the trait analysed

`data`: Original full dataset

`tree.physig.estimates`: Three number, phylogenetic signal estimate (lambda or K) and the p-value for each run with a different phylogenetic tree.

`N.obs`: Size of the dataset after matching it with tree tips and removing NA's.

`stats`: Main statistics for phylogenetic estimates.`CI_low` and `CI_high` are the lower and upper limits of the 95

Note

The argument "se" from phylosig is not available in this function. Use the argument "V" instead with intra_physig to indicate the name of the column containing the standard deviation or the standard error of the trait variable instead.

Author(s)

Caterina Penone & Gustavo Paterno

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Donoghue, M.J. & Ackerly, D.D. (1996). Phylogenetic Uncertainties and Sensitivity Analyses in Comparative Biology. Philosophical Transactions: Biological Sciences, pp. 1241-1249.

Blomberg, S. P., T. Garland Jr., A. R. Ives (2003) Testing for phylogenetic signal in comparative data: Behavioral traits are more labile. Evolution, 57, 717-745.

Pagel, M. (1999) Inferring the historical patterns of biological evolution. Nature, 401, 877-884.

Kamilar, J. M., & Cooper, N. (2013). Phylogenetic signal in primate behaviour, ecology and life history. Philosophical Transactions of the Royal Society B: Biological Sciences, 368: 20120341.

See Also

phylosig, tree_phylm, sensi_plot

Examples

```
# Load data:
data(alien)
alien.data<-alien$data
alien.phy<-alien$phy
# Logtransform data
alien.data$logMass <- log(alien.data$adultMass)
# Run sensitivity analysis:
tree <- tree_physig(trait.col = "logMass", data = alien.data,
phy = alien.phy, n.tree = 10)
summary(tree)
sensi_plot(tree)
sensi_plot(tree, graphs = 1)
sensi_plot(tree, graphs = 2)
```

---

tree_samp_phyglm	*Interaction between phylogenetic uncertainty and sensitivity to species sampling - Phylogenetic Logistic Regression*

Description

Performs analyses of sensitivity to species sampling by randomly removing species and detecting the effects on parameter estimates in phylogenetic logistic regression, while evaluating uncertainty in trees topology.

Usage

```
tree_samp_phyglm(
  formula,
  data,
  phy,
```

```
    n.sim = 30,
    n.tree = 2,
    breaks = seq(0.1, 0.5, 0.1),
    btol = 50,
    track = TRUE,
    ...
)
```

Arguments

formula	The model formula
data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
n.sim	The number of times species are randomly deleted for each break.
n.tree	Number of times to repeat the analysis with n different trees picked randomly in the multiPhylo file.
breaks	A vector containing the percentages of species to remove.
btol	Bound on searching space. For details see phyloglm
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to phyloglm

Details

This function randomly removes a given percentage of species (controlled by breaks) from the full phylogenetic logistic regression, fits a phylogenetic logistic regression model without these species using [phyloglm](), repeats this many times (controlled by times), stores the results and calculates the effects on model parameters. It repeats this operation using n trees, randomly picked in a multiPhylo file.

Only logistic regression using the "logistic_MPLE"-method from phyloglm is implemented.

Currently, this function can only implement simple logistic models (i.e. $trait\ predictor$). In the future we will implement more complex models.

Output can be visualised using sensi_plot.

Value

The function samp_phylm returns a list with the following components:

formula: The formula

full.model.estimates: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. lambda or kappa) for the full model without deleted species.

sensi.estimates: A data frame with all simulation estimates. Each row represents a model rerun with a given number of species n.remov removed, representing n.percent of the full dataset. Columns report the calculated regression intercept (intercept), difference between simulation intercept and full model intercept (DIFintercept), the percentage of change in intercept compared to the full model (intercept.perc) and intercept p-value (pval.intercept). All these parameters are also reported for the regression slope (DIFestimate etc.). Additionally, model aic value (AIC)

and the optimised value (`optpar`) of the phylogenetic parameter (e.g. `kappa` or `lambda`, depending on the phylogenetic model used) are reported. Lastly we reported the standardised difference in intercept (`sDIFintercept`) and slope (`sDIFestimate`).

`sign.analysis` For each break (i.e. each percentage of species removed) this reports the percentage of statistically significant (at p<0.05) intercepts (`perc.sign.intercept`) over all repetitions as well as the percentage of statisticaly significant (at p<0.05) slopes (`perc.sign.estimate`).

`data`: Original full dataset.

Note

Please be aware that dropping species may reduce power to detect significant slopes/intercepts and may partially be responsible for a potential effect of species removal on p-values. Please also consult standardised differences in the (summary) output.

Author(s)

Gustavo Paterno, Gijsbert D.A. Werner & Caterina Penone

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Werner, G.D.A., Cornwell, W.K., Sprent, J.I., Kattge, J. & Kiers, E.T. (2014). A single evolutionary innovation drives the deep evolution of symbiotic N2-fixation in angiosperms. Nature Communications, 5, 4087.

#' Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phyloglm, samp_phyglm, tree_phyglm, tree_samp_phylm,sensi_plot

Examples

```
## Not run:
# Simulate Data:
set.seed(6987)
mphy = rmtree(100, N = 30)
x = rTrait(n=1,phy=mphy[[1]])
X = cbind(rep(1,100),x)
y = rbinTrait(n=1,phy=mphy[[1]], beta=c(-1,0.5), alpha=.7 ,X=X)
dat = data.frame(y, x)
# Run sensitivity analysis:
tree_samp <- tree_samp_phyglm(y ~ x, data = dat, phy = mphy, n.tree = 3, n.sim=10)
summary(tree_samp)
sensi_plot(tree_samp)
sensi_plot(tree_samp, graphs = 1)
sensi_plot(tree_samp, graphs = 2)

## End(Not run)
```

---

tree_samp_phylm   *Interaction between phylogenetic uncertainty and sensitivity to species*
                                sampling - Phylogenetic Linear Regression

---

Description

Performs analyses of sensitivity to species sampling by randomly removing species and detecting
the effects on parameter estimates in a phylogenetic linear regression, while evaluating uncertainty
in trees topology.

Usage

```
tree_samp_phylm(
  formula,
  data,
  phy,
  n.sim = 30,
  n.tree = 2,
  breaks = seq(0.1, 0.5, 0.1),
  model = "lambda",
  track = TRUE,
  ...
)
```

Arguments

formula	The model formula
data	Data frame containing species traits with row names matching tips in phy.
phy	A phylogeny (class 'phylo') matching data.
n.sim	The number of times species are randomly deleted for each break.
n.tree	Number of times to repeat the analysis with n different trees picked randomly in the multiPhylo file.
breaks	A vector containing the percentages of species to remove.
model	The phylogenetic model to use (see Details). Default is lambda.
track	Print a report tracking function progress (default = TRUE)
...	Further arguments to be passed to phylolm

Details

This function randomly removes a given percentage of species (controlled by breaks) from the
full phylogenetic linear regression, fits a phylogenetic linear regression model without these species
using [phylolm](#), repeats this many times (controlled by n.sim), stores the results and calculates the
effects on model parameters. It repeats this operation using n trees, randomly picked in a multiPhylo
file.

All phylogenetic models from `phylolm` can be used, i.e. BM, OUfixedRoot, OUrandomRoot, lambda, kappa, delta, EB and trend. See ?`phylolm` for details.

Currently, this function can only implement simple linear models (i.e. *trait predictor*). In the future we will implement more complex models.

Output can be visualised using `sensi_plot`.

Value

The function `samp_phylm` returns a list with the following components:

`formula`: The formula

`full.model.estimates`: Coefficients, aic and the optimised value of the phylogenetic parameter (e.g. `lambda` or `kappa`) for the full model without deleted species.

`sensi.estimates`: A data frame with all simulation estimates. Each row represents a model re-run with a given number of species `n.remov` removed, representing `n.percent` of the full dataset. Columns report the calculated regression intercept (`intercept`), difference between simulation intercept and full model intercept (`DIFintercept`), the percentage of change in intercept compared to the full model (`intercept.perc`) and intercept p-value (`pval.intercept`). All these parameters are also reported for the regression slope (`DIFestimate` etc.). Additionally, model aic value (`AIC`) and the optimised value (`optpar`) of the phylogenetic parameter (e.g. `kappa` or `lambda`, depending on the phylogenetic model used) are reported. Lastly we reported the standardised difference in intercept (`sDIFintercept`) and slope (`sDIFestimate`).

`sign.analysis` For each break (i.e. each percentage of species removed) this reports the percentage of statistically significant (at p<0.05) intercepts (`perc.sign.intercept`) over all repetitions as well as the percentage of statisticaly significant (at p<0.05) slopes (`perc.sign.estimate`).

`data`: Original full dataset. #' @note Please be aware that dropping species may reduce power to detect significant slopes/intercepts and may partially be responsible for a potential effect of species removal on p-values. Please also consult standardised differences in the (summary) output.

Author(s)

Gustavo Paterno, Gijsbert D.A. Werner & Caterina Penone

References

Paterno, G. B., Penone, C. Werner, G. D. A. sensiPhy: An r-package for sensitivity analysis in phylogenetic comparative methods. Methods in Ecology and Evolution 2018, 9(6):1461-1467

Werner, G.D.A., Cornwell, W.K., Sprent, J.I., Kattge, J. & Kiers, E.T. (2014). A single evolutionary innovation drives the deep evolution of symbiotic N2-fixation in angiosperms. Nature Communications, 5, 4087.

Ho, L. S. T. and Ane, C. 2014. "A linear-time algorithm for Gaussian and non-Gaussian trait evolution models". Systematic Biology 63(3):397-408.

See Also

phylolm, samp_phylm, tree_phylm, tree_samp_phyglm, sensi_plot

Examples

```
## Not run:
# Load data:
data(alien)
# Run analysis:
samp <- tree_samp_phylm(log(gestaLen) ~ log(adultMass), phy = alien$phy,
                                    data = alien$data, n.tree = 5, n.sim=10)
summary(samp)
head(samp$sensi.estimates)
# Visual diagnostics
sensi_plot(samp)
sensi_plot(samp, graphs = 1)
sensi_plot(samp, graphs = 2)

## End(Not run)
```

Index