Package 'PAICE'

October 27, 2023

Type Package

Title Phylogeographic Analysis of Island Colonization Events

Version 1.0.1

Author Alberto J. Coello, Mario Fernandez-Mazuecos, Ruben H. Heleno, Pablo Vargas

Maintainer Alberto J. Coello <albjcoello@gmail.com>

Depends R (>= 3.6.0)

Suggests testthat

Description Estimation of the number of colonization events between islands of the same archipelago for a species. It uses rarefaction curves to control for both field and genetic sample sizes.

License GPL-2

Encoding UTF-8

LazyData true

Date 2023-10-27

URL <https://github.com/PAICEcode/PAICE>

RoxygenNote 7.2.1

NeedsCompilation no

R topics documented:

PAICE-package	*Phylogeographic Analysis of Island Colonization Events*

Description

A package for inferring inter-island colonization events in island-like systems.

Details

Estimation of the number of infer inter-island colonization events in an island-like system by analyzing the geographic distribution of uniparentally inherited haplotypes and their genealogical relationships. Furthermore, by building rarefaction curves based on both genetic sampling (variable positions) and field sampling (populations/individuals), an estimation of the number of colonization events corrected by sampling effort could be done. The method used in the **PAICE** package is described in Coello et al. (2022).

PAICE functions

colonization to infer the minimun number of colonization events

geneticResampling to simplify the genealogy by deleting a variable position

maxCol to calculate asymptotic estimators considering genetic and field sampling

plot.maxCol to plot curves generated by maxCol

plot.rarecol to plot rarefaction curves

rarecol to generate rarefaction curves of colonization events

read.rarecol to read previously saved rarefaction curve files

PAICE datasets

CmonsData haplotype distribution of *Cistus monspeliensis* in the Canary Islands

CmonsNetwork genealogy of *Cistus monspeliensis*

CmonsRare example data of rarefaction curves for *Cistus monspeliensis*

Author(s)

Alberto J. Coello, Mario Fernandez-Mazuecos, Ruben H. Heleno and Pablo Vargas

Maintainer: Alberto J. Coello <albjcoello@gmail.com>

References

Coello, A.J., Fernandez-Mazuecos, M., Heleno, R.H., Vargas, P. (2022). PAICE: A new R package to estimate the number of inter-island colonizations considering haplotype data and sample size. *Journal of Biogeography*, 49(4), 577-589.DOI: 10.1111/jbi.14341

Examples

```
# Inference of minimum number of inter-island colonization events
data(CmonsData)
data(CmonsNetwork)
col <- colonization(data = CmonsData, network = CmonsNetwork)
col
summary(col)

# Asumptotic estimators of colonization events
# 25 replicates used in each sampling variable
## Note: The code is commented because 'CmonsRare' exists as an example
#set.seed(31)
#CmonsRare <- rarecol(data = CmonsData, network = CmonsNetwork,
#    replicates_field = 25, replicates_genetic = 25, monitor = TRUE,
#    mode = c(1, 2))
maxcol <- maxCol(data = CmonsRare, level = 0.95, del = 0.5, method = 1)
maxcol
summary(maxcol)

# Plotting results
par(mfrow = c(2, 2))
plot(CmonsRare)
par(fig = c(0, 1, 0, 0.5), new = TRUE)
plot(maxcol)
```

CmonsData	*Occurrence matrix of* Cistus monspeliensis *in the Canary Islands*

Description

Data of *Cistus monspeliensis* prepared to be used as example for the **PAICE** package.

Usage

```
data(CmonsData)
```

Format

A data frame containing a presence matrix of *Cistus monspeliensis* haplotypes in the Canary Islands extracted from Coello et al. (2021). Each row indicates the number of individuals of each haplotype occurring in each population. The first column indicates the island, the second column indicates the population and successive columns correspond to haplotypes in the island system. Missing haplotypes are also included but without any presence (haplotypes m1 and m2).

Details

Data containing occurrences of each haplotype of *Cistus monspeliensis* found in the Canary Islands. Data were taken from Coello et al. (2021). This dataset was constructed using three ptDNA regions and 37 populations from the Canarian archipelago.

References

Coello, A.J., Fernandez-Mazuecos, M., Garcia-Verdugo, C., Vargas, P. (2021). Phylogeographic sampling guided by species distribution modeling reveals the Quaternary history of the Mediterranean-Canarian *Cistus monspeliensis* (Cistaceae). *Journal of Systematics and Evolution, 59(2)*, 262-277. DOI: 10.1111/jse.12570

Examples

```
data(CmonsData)
CmonsData # Show data frame
```

CmonsNetwork	*Genealogical relationship of* Cistus monspeliensis *haplotypes*

Description

Genealogy of Canarian haplotypes of *Cistus monspeliensis*.

Usage

```
data("CmonsNetwork")
```

Format

A data frame containing the genalogy of *Cistus monspeliensis* in the Canary Islands. Each row indicates the connection between each haplotype and its ancestral haplotype. The first column is the name of the haplotype, the second column is the name of its ancestral haplotype and the third column indicates the number of variable positions that change between both haplotypes. The ancestral haplotype in the archipelago (haplotype C1) is connected to the outgroup ("OUT"), and is be located in the first row of the genealogy.

Details

This dataset was taken from Coello et al. (2021). It was constructed using three ptDNA regions and 37 populations from the Canarian archipelago.

References

Coello, A.J., Fernandez-Mazuecos, M., Garcia-Verdugo, C., Vargas, P. (2021). Phylogeographic sampling guided by species distribution modeling reveals the Quaternary history of the Mediterranean-Canarian *Cistus monspeliensis* (Cistaceae). *Journal of Systematics and Evolution, 59(2)*, 262-277. DOI: 10.1111/jse.12570

Examples

```
data(CmonsNetwork)
CmonsNetwork # Show data frame
```

CmonsRare *Simulated rarefaction curves of* Cistus monspeliensis

---

Description

Simulated rarefaction curves to be used as example data for estimation of colonization events.

Usage

```
data(CmonsRare)
```

Format

A list containing data of both genetic and field rarefaction curves. The first element corresponds to the genetic estimation and the second element corresponds to the field estimation.

Details

This dataset was constructed from CmonsData and CmonsNetwork with the following code:

```
set.seed(31)

CmonsRare <- rarecol(data = CmonsData, network = CmonsNetwork,
    replicates_field = 25, replicates_genetic = 25)
```

Examples

```
data(CmonsRare)
str(CmonsRare) # Structure of data
```

---

colonization *Inference of minimum number of colonization events*

Description

A inference of the minimum number of colonization events between islands of an archipelago considering both haplotype distributions and genealogy.

Usage

```
colonization(data, network)
```

Arguments

data   a data frame containing the matrix of occurrences of haplotypes in the islands of an archipelago (applicable to any island-like system). The first two columns indicate islands and populations sampled. Successive columns indicate haplotype occurrences (one column per haplotype). If present, missing haplotypes must also be included (i.e. columns without occurrences).

network   a data frame containing the genealogy of haplotypes. The first column indicates the haplotype, the second column indicates its ancestral haplotype and the third column indicates the variable position changed between an haplotype and its ancestral haplotype. If present, missing haplotypes must also be included. The ancestral haplotype must be connected to an outgroup named "OUT", located in the first row of the data frame, and has a variable position not shared with other connections.

Details

Colonization events are inferred following Coello et al. (2022).

Each haplotype produces a number of colonization events equal to the total number of islands in which the haplotype occurs minus one. These are type 1 colonization events (c1).

Additionally, colonization events between an ancestral haplotype and the derived haplotypes are also inferred if the ancestral haplotype occurs in different islands than the derived haplotypes. These inferred colonization events correspond to type 2 (c2) and type 3 (c3) colonization events.

A type 2 colonization events is that between a haplotype and its ancestral haplotype that can only be assigned to the connection between these two haplotypes. These colonization events are noted in the derived haplotype.

Type 3 colonization events are those (one or more) inferred between an ancestral haplotype and its derived haplotypes but that cannot be assigned to a specific connection, so colonization events are assigned to the ancestral haplotype.

Value

colonization returns an object of class "colonization".

The function print shows the total of colonization events inferred. The function summary returns a more detailed output showing a description of data used and inferred colonization events by haplotype and by type.

Note

colonization only considers the complete sampling. To correct the inference by field and genetic sampling use rarecol.

References

Coello, A.J., Fernandez-Mazuecos, M., Heleno, R.H., Vargas, P. (2022). PAICE: A new R package to estimate the number of inter-island colonizations considering haplotype data and sample size. *Journal of Biogeography*, 49(4), 577-589.DOI: 10.1111/jbi.14341

See Also

rarecol to build a rarefaction curve of colonization events. maxCol to calculate the asymptotic estimator for the number of colonization events from data generated by rarecol.

Examples

```
data(CmonsData)
data(CmonsNetwork)
col <- colonization(data = CmonsData, network = CmonsNetwork)
col # Total of colonization events inferred
summary(col) # Detailed description of inferred colonization events
```

---

geneticResampling   *Simulate genetic sampling effort reduction*

Description

A reduction of the resolution of the genealogy by supressing a variable position in the genealogy. It simulates a lower level of genetic sampling.

Usage

```
geneticResampling(data, network, position)
```

Arguments

data	a data frame containing the occurrence matrix of haplotypes in the islands of an archipelago (applicable to any island-like system). The two first columns indicate islands and populations sampled. Successive columns indicate haplotype occurrences (one column per haplotype). If present, missing haplotypes must also be included (i.e. columns without occurrences).
network	a data frame containing the genealogy of haplotypes. The first column indicates the haplotype, the second column indicates its ancestral haplotype and the third column indicates the variable position changed between an haplotype and its ancestral haplotype. If present, missing haplotypes must also be included. The ancestral haplotype must be connected to an outgroup named "OUT", located in the first row of the data frame, and has a variable position not shared with other connections.
position	numeric. Indicates the variable position that will be deleted in the simplified data.

Details

To simulate a lower level of genetic sampling, this function deletes a variable position from the original data and thus simplifies the genealogy. geneticResampling generates a new occurrence matrix of haplotypes and a new genealogy without the variable position previously indicated and merging ancestral and derived haplotypes separated by this variable position. If more than one connection are defined by the variable position indicated, this function deletes all connections with this variable position. This function works for both observed and missing haplotypes.

Value

geneticResampling returns a list containing the new occurrence matrix of haplotypes and the new genealogy after deleting the variable position indicated. The returned object contains the following components:

data	a data frame containing the new occurrence matrix of haplotypes after removing the variable position indicated.
network	a data frame containing the new genealogy after removing the variable position indicated.

Note

If the variable position corresponds to the connection between the ancestral haplotype in the archipelago and the outgroup (denoted as "OUT"), no change is effected as the ancestral haplotype stays connected to the outgroup.

This function works inside rarecol.

See Also

rarecol to build a rarefaction curve of colonization events.

Examples

```
data(CmonsData)
data(CmonsNetwork)
# Delete position 462 of Cistus monspeliensis data
newdata <- geneticResampling(CmonsData, CmonsNetwork, 462)
newdata$data # New presences matrix of haplotypes
newdata$network # New genealogy
```

---

maxCol	*Asymptotic estimation of the number of colonization events*

Description

A calculation of asymptotic estimators of colonization events from both curves generated using the rarecol function.

Usage

```
maxCol(data, level = 0.95, del = 0.05, method = 1)
```

Arguments

data	an object of class "rarecol" that contains output from rarecol.
level	numeric. Determines the confidence interval used to estimate error in Michaelis-Menten equation parameters. By default 0.95.
del	numeric. Determines the interval of values to be deleted to avoid the influence of extreme values. By default 0.05 (i.e. deleted values below 2.5 quantile and above 97.5 quantile).
method	numeric. Indicates if the algorithm should try to fit the curve by assigning a value to the intercept in genetic rarefaction curves (method = 1) or discard these cases when it is not possible to fit the curve with all values (method = 0). By default, method = 1.

Details

This function calculates the number of colonization events estimated by both resampling methods used in the function rarecol. The first estimation (genetic estimation) corresponds to resampling first at genetic level (number of variable positions) and then, per each variable position, a complete resampling of the number of populations is done. The second estimation (field estimation) corresponds to the opposite resampling, it is done first at field level (number of populations) and then, per each population, a complete resampling of the number of variable positions is done.

For each curve, the function first estimates the asymptote (estimated number of colonization events) for each level of the second resampling (populations in the first estimation and variable positions in the second estimation) using the mean value of all replicates at each point. Then, these estimations are used to build the final curve estimating the number of colonization events for each resampling methodology. This final curve uses estimations calculated previously, and the asymptote of the curve is calculated by using mean points for each value of the first resampling method (variable positions in the first estimation and populations in the second estimation). The asymptote is calculated by fitting the curve to a Michaelis-Menten equation following Coello et al. (2022).

The confidence interval for the estimated number of colonization events is calculated with the confint function. Curve fitting is done using the nls function.

Value

This function returns an object of class "maxCol" consisting in a list of the following elements:

DataGen	a data frame containing the mean estimated number of colonization events per number of variable positions in the genetic estimation.
FormulaGen	formula used to fit final curve in the genetic estimation.
DataField	a data frame containing the mean estimated number of colonization events per population in the field estimation.
FormulaField	formula used to fit the final curve in the field estimation.
Summary	a matrix containing the estimated number of colonization events of each estimation (genetic and field). Minimum and maximum are calculated using the confidence interval indicated.
ParametersGen	a matrix containing the value of each parameter to fit a Michaelis-Menten equation for genetic estimation. The minimum and maximum of each parameter according to the confidence interval indicated. This equation is described as: colonization events = M * positions / (K + positions) + c.
ParametersField	
	a matrix containing the value of each parameter to fit a Michaelis-Menten equation for field estimation. The minimum and maximum of each parameter according to the confidence interval indicated. This equation is described as: colonization events = M * (populations - 1) / (K + populations - 1).
ConfintLevel	a vector containing the confidence interval used to calculate minimum and maximum for each parameter.
DeletedData	a vector containing the interval of extreme values deleted to do the fit of the second accumulation curve.

The function print returns the number of colonization events inferred for each estimation (genetic and field) and the interval of confidence of these estimations.The function summary shows a detailed description of parameters used to fit both curves, the formula used to fit these curves and the confidence of interval of each parameter.

Note

To show a detailed description of inferred colonization events in the most complete case use the function `colonization`.

References

Coello, A.J., Fernandez-Mazuecos, M., Heleno, R.H., Vargas, P. (2022). PAICE: A new R package to estimate the number of inter-island colonizations considering haplotype data and sample size. *Journal of Biogeography*, 49(4), 577-589.DOI: 10.1111/jbi.14341

See Also

`rarecol` to build rarefaction curves of colonization events. To describe the number of colonization events inferred in the most complete case use the function `colonization`. `plot.maxCol` to plot the result of this function.

Examples

```
# Use 'CmonsRare' data, a dataset generated using 25 replicates
# in both genetic and field sampling
data(CmonsRare)
maxcol <- maxCol(data = CmonsRare)
maxcol # Number of colonization estimated in each curve
summary(maxcol) # Description of curves
plot(maxcol) # Plotting estimations
# Plot all the information
par(mfrow = c(2, 2))
plot(CmonsRare) # First two plots with rarefaction curves
par(fig = c(0, 1, 0, 0.5), new = TRUE)
plot(maxcol) # Third plot with estimations
```

---

plot.maxCol   *Plot asymptotic estimators of colonization events*

Description

Plots for the estimators calculated by `maxCol`.

Usage

```
## S3 method for class 'maxCol'
plot(x, xlim, ylim, col, xlabbotton, xlabtop, ylab, main,
     pch = 16, lty = 1, lwd = 2, cex = 1, estimation = TRUE,
     legend = TRUE, ...)
```

Arguments

x	an object of `class` "maxCol" returned by `maxCol` function.
xlim, ylim	numeric vector containing limits of x and y axis of the plot (min, max).
col	character vector containing colour of both estimation: genetic and field.
xlabbotton	a title of the x axis at the bottom of the plot. It correspond with genetic estimation.

xlabtop	a title of the x axis at the top of the plot. It correponds with field estimation.
ylab	a title of the y axis of the plot.
main	an overall title for the plot.
pch	indicate symbol used for points of the plot (by default pch = 16). See [par](#) for additional information.
lty	type of lines used in the plot for curve fitting (by default lty = 1). See [par](#) for additional information.
lwd	width of lines used in the plot for curve fitting (by default lwd = 2). See [par](#) for additional information.
cex	size of elements in the plot.
estimation	logical. If it is TRUE an estimation of estimation of number of colonization events is plotted at the right size of the plot.
legend	logial. If it is TRUE the legend is added.
...	aditional graphical parameters (see [par](#)) for aditional information.

Details

Genetic and field estimation are fitted to Michaelis-Menten equation following Coello et al. (2022).

Value

Plot returned by this function represent estimations calculated by [maxCol](#). The two curves representing both estimators: genetic and field. Each point represent the mean of number of colonization events inferred by all replicates at this sampling level. Curves represent Michaelis-Menten equation fitted to this dataset. If it is plotted, right side of the plot represent the number of colonization events estimated by this fitting curve for each estimation, including the conficende interval of this estimation.

References

Coello, A.J., Fernandez-Mazuecos, M., Heleno, R.H., Vargas, P. (2022). PAICE: A new R package to estimate the number of inter-island colonizations considering haplotype data and sample size. *Journal of Biogeography*, 49(4), 577-589.DOI: [10.1111/jbi.14341](#)

See Also

[maxCol](#) to fit the accumulation curve of colonization events and estimate the number of colonization events.

Examples

```
# Use 'CmonsRare' data, a dataset generated using 25 replicates
# in both genetic and field sampling
data(CmonsRare)
maxcol <- maxCol(data = CmonsRare)
plot(maxcol)
```

---

plot.rarecol   *Plot rarefaction curve of colonization events*

Description

Plots for the rarefaction curves produced by rarecol.

Usage

```
## S3 method for class 'rarecol'
plot(x, xlim1, xlim2, ylim, ylim1, ylim2, palette1, palette2, main1,
    main2, xlab1, xlab2, ylab1, ylab2, las1 = 1, las2 = 1,
    cextText = 0.75, legendbar = TRUE, ...)
```

Arguments

x	an object generated by the rarecol function.
xlim1, xlim2	x limits (min, max) of the two plots.
ylim1, ylim2	y limits (min, max) of the two plots.
ylim	y limits (min, max) of the two plots simultaneously. If ylim is defined, the function does not consider ylim1 and ylim2.
palette1, palette2	
	vector of color for lines in plot 1 and plot 2.
main1, main2	overall title of plot 1 and plot 2.
xlab1, xlab2	label of x axis of plot 1 and plot 2.
ylab1, ylab2	label of y axis of plot 1 and plot 2
las1, las2	numeric. Corresponds to the style of axis labels in plot 1 and plot 2. Values: 0 (always parallel to the axis), 1 (always horizontal), 2 (always perpendicular to the axis), 3 (always vertical). See par for more information.
cextText	size of legend text.
legendbar	logical. If TRUE, it shows a legend bar indicating the color value of each variable position.
...	aditional graphical parameters (see par) for additional information.

Details

The first plot corresponds to the genetic estimation. This plot shows accumulation of colonization events as a function of population number. Each curve was created for each number of variable positions in the dataset.

The second plot corresponds to the field estimation. This plot shows accumulation of colonization events as a function of the number of variable positions. Each curve is created for each number of populations in the dataset.

Value

This function returns two plots corresponding to the two resampling methods used in rarecol. The first curve corresponds to the "genetic estimation" in which a genetic resampling of every possible number of variable position is done and, for each resample, a complete resampling of population is done. The second curve represents the opposite method corresponding to the "field estimation": it first resamples every possible number of populations and, for each case, a complete resampling of variable positions is done.

References

Coello, A.J., Fernandez-Mazuecos, M., Heleno, R.H., Vargas, P. (2022). PAICE: A new R package to estimate the number of inter-island colonizations considering haplotype data and sample size. *Journal of Biogeography*, 49(4), 577-589.DOI: 10.1111/jbi.14341

See Also

rarecol to build a rarefection curve of colonization events.

Examples

```
# Use 'CmonsRare' data, a dataset generated using 25 replicates
# in both genetic and field sampling
data(CmonsRare)
plot(CmonsRare)
```

---

rarecol   *Rarefaction curve of colonization events*

Description

A creation of rarefaction curves considering both genetic and field data. First, the function samples variable positions and then samples populations for each variable position. Second, it samples populations and then samples variable positions for each population.

Usage

```
rarecol(data, network, replicates_field = 10,
        replicates_genetic = 10, mode = c(1, 2), monitor = TRUE,
        file = NULL)
```

Arguments

data   a data frame containing the occurrence matrix of haplotypes in the islands of an archipelago (applicable to any island-like system). The first two columns indicate islands and populations sampled. Successive columns indicate haplotype occurrences (one column per haplotype). If present, missing haplotypes must also be included (i.e. columns without occurrences).

network   a data frame containing the genealogy of haplotypes. The first column indicates the haplotype, the second column indicates its ancestral haplotype and the third column indicates the variable position changed between an haplotype and its ancestral haplotype. If present, missing haplotypes must also be included. The

ancestral haplotype must be connected to an outgroup named "OUT", located in the first row of the data frame, and has a variable position not shared with other connections.

replicates_field

numeric. Number of replicates for field resampling. Each replicate adds populations from one to the total number of populations in the dataset and infers the corresponding number of colonization events.

replicates_genetic

numeric. Number of replicates for genetic resampling. Each replicate adds variable positions from none (chorology) to the total number of variable positions in the dataset and infers the corresponding number of colonization events.

monitor   logical. If TRUE it shows progress in the console.

mode   numeric vector. Indicates which estimations must be conducted. 1 for genetic estimation and 2 for field estimation. By default the function conducts both processes.

file   character string determining the name of the file to save rarefaction curves built by this function. Two files are created, one for genetic estimation and one for field estimation. If a file name is indicated, the function does not return any result, all the results are saved in the indicated files. If set to NULL, the data are showed as output.

Value

rarecol returns an object of class "rarecol". The return is a list containing information about the two rarefaction curves generated.

References

Coello, A.J., Fernandez-Mazuecos, M., Heleno, R.H., Vargas, P. (2022). PAICE: A new R package to estimate the number of inter-island colonizations considering haplotype data and sample size. *Journal of Biogeography*, 49(4), 577-589.DOI: 10.1111/jbi.14341

See Also

To describe the number of colonization events observed in the most complete case, use the function colonization. maxCol, which estimates the number of colonization events of data generated by this funtion. plot.rarecol to plot the result of this function. read.rarecol to import files generated from this function.

Examples

```
data(CmonsData)
data(CmonsNetwork)
# Build rarefaction curves with 5 field and genetic replicates
## Note: more replicates are needed to build accurate curves
## Note: 5 replicates are relatively fast and adequate to
##       explore the data
rcol <- rarecol(data = CmonsData, network = CmonsNetwork,
                replicates_field = 5, replicates_genetic = 5,
                monitor = TRUE, mode = c(1, 2))
par(mfrow = c(1, 2))
plot(rcol) # Plotting results
```

---

read.rarecol | *Read files containing rarefaction curves of colonization events*

Description

An import method for data generated by rarecol.

Usage

```
read.rarecol(gen, field)
```

Arguments

gen, field   filenames of genetic and field estimation data.

Details

This function uses read.table to import both files created by rarecol.

Value

This function returns an object of class rarecol. This object is a list in which each element is a data.frame containing information about colonization inference.

References

Coello, A.J., Fernandez-Mazuecos, M., Heleno, R.H., Vargas, P. (2022). PAICE: A new R package to estimate the number of inter-island colonizations considering haplotype data and sample size. *Journal of Biogeography*, 49(4), 577-589.DOI: 10.1111/jbi.14341

See Also

rarecol for building of rarefaction curves of colonization events.

Examples

```
data(CmonsData)
data(CmonsNetwork)
# Make rarefaction curves and save it in working directory,
## Note: only one replicate per sampling to it quickly
rarecol(data = CmonsData, network = CmonsNetwork,
        replicates_field = 1, replicates_genetic = 1,
        monitor = TRUE, file = "rareData")
# Genetic estimation has the suffix "_gen" and the field "_field"
raredata <- read.rarecol(gen = "rareData_gen.csv",
                         field = "rareData_field.csv")
str(raredata) # Show structure of data imported
# Remove files created
file.remove("rareData_gen.csv", "rareData_field.csv")
```

Index