Package ‘seraphim’

May 31, 2016

Type Package

Title Studying environmental rasters and phylogenetic informed movements

Version 1.0

Date 2014-11-06

Author Simon Dellicour, Rebecca Rose, Nuno Faria, Philippe Lemey, Oliver G. Pybus

Maintainer Simon Dellicour <simon.dellicour@zoo.ox.ac.uk>

Depends ape, doMC, fields, gdistance, gstat, ks, MASS, raster, rgdal, yhat, vegan

Description The source code for SERAPHIM is distributed under a GNU Lesser General Public License

License GPL-3

R topics documented:

seraphim-package .......................................................... 1
circuitScape ................................................................. 2
rasterSimulation ............................................................. 3
spreadFactors ................................................................. 4
spreadGraphic ................................................................. 7
spreadStatistics .............................................................. 9
torusRandomisation ......................................................... 10
treeExtractions .............................................................. 11
variogramModel .............................................................. 12
vectorRandomisation ......................................................... 13

seraphim-package  A package to analyse environmental raster and spatiotemporally-referenced phylogenetic trees.
Description

The package "seraphim" (for "studying environmental rasters and phylogenetic informed movements"; Dellicour et al., submitted) can be used to (i) extract the spatio-temporal information contained in spatiotemporally-referenced trees inferred, for instance, by the continuous phylogeographic method implemented in BEAST (Lemey et al. 2010), (ii) estimate a series of dispersion/epidemiological statistics based on these trees, (iii) compute the environmental "weights" associated with each phylogenetic branch for given environmental rasters, (iv) estimate the correlation between the dispersal duration and these weights associated with each branch, and (v) test the statistical significance of these correlations using null models generated by randomisation procedures.

Author(s)

Simon Dellicour, Rebecca Rose, Nuno Faria, Philippe Lemey, Oliver G. Pybus

References


See Also

http://beast.bio.ed.ac.uk
http://www.circuitscape.org

circuitScape

A function to launch Circuitscape analyses.

Description

The function calls the external Python library "circuitscape" to compute resistance or conductance weight associated with each phylogenetic branch.

Usage

circuitScape(envVariable, envVariableName, resistance, avgResistance, fourCells, fromCoor, toCoor, OS, prefix, ID)

Arguments

envVariable raster file on which the resistance or conductance has to be computed.
envVariableName name (string) of the raster file. This name can be different from the name saved in the raster object and will be used to name the temporary file generated for the Circuitscape analysis.
### circuitScape

**resistance** boolean variable specifying if the raster will be treated as a resistance or conductance factor.

**avgResistance** boolean variable specifying if the raster cells have to be connected by average resistance (TRUE) or average conductance (FALSE).

**fourCells** boolean variable specifying if a cell is connected to its four (TRUE) or eight (FALSE) neighbours.

**fromCoor** matrix containing the coordinates of the starting nodes of each phylogenetic branch coming from one tree (one line per node, 1st column with the longitude and the 2nd column with the latitude).

**toCoor** matrix containing the coordinates of the ending nodes of each phylogenetic branch coming from one tree (one line per node, 1st column with the longitude and the 2nd column with the latitude).

**OS** operating system on which the function is used ("Unix" or "Windows").

**prefix** a character string that will be used as a prefix to name the temporary folders generated for Circuitscape.

**ID** a number to identify the Circuitscape operation.

### Value

The function returns a vector containing the resistance or conductance estimated for each phylogenetic branch.

### Note

This function requires the preliminary installation of the software Circuitscape as a Python package. Instructions on how to install this Python package are available on Circuitscape website: [http://docs.circuitscape.org/circuitscape_4_0_user_guide.html](http://docs.circuitscape.org/circuitscape_4_0_user_guide.html). Also note that fromCoor and toCoor matrices have to display the same dimensions and lines in the same order: coordinates in line i in fromCoor have to be the coordinates of the starting node connected to the ending node for which geographic coordinates are on line i in toCoor.

This function can be used the Python library "circuitscape" to compute resistance or conductance weight associated to any movement vectors for which start and end locations are respectively specified in the "fromCoor" and "toCoor" arguments.

### Author(s)

Simon Dellicour

### References


rasterSimulation  

*Function to simulate an environmental raster*

**Description**

This function simulates an environmental raster based on a variogram model that can be preliminary estimated with the "variogramModel" function. After the simulation step, a n-score transformation is automatically performed on the simulated raster cell values in order to obtain a distribution of values matching the one observed on the original raster file.

**Usage**

```
rasterSimulation(rast, variogramModel)
```

**Arguments**

- `rast`: raster template for the raster simulation.
- `variogramModel`: estimated variogram model for the environmental variable.

**Details**

This function can be called by the "spreadFactor" function.

**Value**

The function returns a raster simulated under a specific variogram model.

**Author(s)**

Simon Dellicour

**References**

Description
This function estimates and tests the correlation between environmental weight(s) and dispersal duration associated with each phylogenetic branch. The function performs three steps: (1) for each environmental factor, computation of the environmental weights associated with each phylogenetic branch (using the "straight-line path", "least-cost path" or "random walk" model, see the details below), (2) estimation of the correlation between the dispersal durations and the corresponding environmental weights using a linear regression approach, and (3) optional randomisations to test the level of significance of the different correlation measures. To generate null models, three different randomisation procedures are implemented: a torus translation and reflection of the raster, a simulation of a random raster based on a previously estimated variogram model, and a randomisation of the nodes position while maintaining the three topology and the estimated position of the most ancestral node (root).

Usage
spreadFactors(localTreesDirectory = ".",
nberOfExtractionFiles = 1,
envVariables,
pathModel = 1,
resistances = list(),
avgResistances = list(),
fourCells = FALSE,
nberOfRandomisations = 0,
randomProcedure = 1,
outputName = "",
variogramModels = list(),
showingPlots = FALSE,
nberOfCores = 1,
OS = "Unix")

Arguments
localTreesDirectory
directory where the extracted matrices are saved (see the "treeExtractions" function).
nberOfExtractionFiles
number of files created with the "treeExtractions" function.
envVariables list of environmental rasters to test.
pathModel path taken model to use for computing environmental weights: "1" (straight-line path model), "2" (least-cost path model, Dijkstra 1959) or "3" (random walk model, McRae 2006).
resistances list of boolean variables corresponding to the "envVariables" list. These boolean variables specify if the rasters of the "envVariables" list have to be treated as resistance or conductance factors (only useful for the least-cost and random walk path models).
avgResistances list of boolean variables corresponding to the "envVariables" list. These boolean variables specify if the raster cells have to be connected by average resistance (TRUE) or average conductance (FALSE). This option is only useful for the random walk path model (Circuitscape).
spreadFactors

fourCells  boolean variable specifying if a cell is connected to its four (TRUE) or eight (FALSE) neighbours. This option is only important for the random walk path model (Circuitscape).

nberOfRandomisations  number of randomisation steps to perform in order to test the correlation between environmental weights and dispersal durations.

randomProcedure  randomisation procedure to use: "1" (torus translations and reflections), "2" (raster simulations according to preliminary estimated variogram model(s)) or "3" (randomisations of nodes position while maintaining the branches length, the tree topology and the location of the most ancestral node).

outputName  name (prefix) to give to the different output files.

variogramModels  estimated variogram models for each environmental variables (used if "randomProcedure" = "2").

showingPlots  boolean variable specifying if the different plots have to be displayed or not.

nberOfCores  number of available cores to parallelise the randomisations.

OS  operating system on which the function is used ("Unix" or "Windows").

Details

Three different path taken models are available to compute the environmental "weights" associated with each phylogenetic branch: (1) the straight-line path model only considering the raster cells below each phylogenetic branch. In this model, the environmental weight associated with a given branch for a given environmental variable will be the sum of the values of the raster cells below the straight-line segment connecting the starting and ending nodes of the branch. (2) The least-cost path model (Dijkstra 1959). In this second model, environmental weight associated with a given branch for a given environmental variable will be the sum of the values of the raster cells selected by a shortest-path algorithm while considering the environmental variable as a resistance or a conductance factor. (3) The random walk model based on circuit theory (McRae 2006). This latter model considers random walk dispersal behaviour, simultaneously integrating the contribution of multiple possible pathways to compute the environmental weight (McRae 2006). The least-cost path is here computed with the "costDistance" function available in the "gdistance" R package (van Etten 2012) and the random-walk weights are computed using the Python package "circuitscape" (McRae 2006).

Regarding the significance tests of the correlation measures, three randomisation procedures have been implemented to create null models: (1) a torus translation and reflection of the original environmental rasters (e.g. Harms et al. 2001), (2) a simulation of random environmental rasters based on preliminary estimated variogram model(s) (one per raster file), and (3) a randomisation of the nodes position while maintaining the branches length, the tree topology and the location of the ancestral node. Torus randomisations (translations and reflections) are performed with the "torusTransformation" function. Raster simulations are performed with the "rasterSimulation" function that requires a variogram model that has to be preliminary estimated with the "variogramModel" function. Note that the "rasterSimulation" function automatically performs an n-score transformation on the simulated raster cell values in order to obtain a distribution of values matching the one observed on the original raster file.

Finally, the function also returns the results based on a multivariate standard GLM (generalized linear model). In this GLM analysis, the dispersal duration remains the response variable and all the environmental weights computed for each distinct environmental factors are the response variables. As in the GLM approach of Faria et al. (2013), all variable were log transformed and standardised.
Value

The function creates several output files. If the number of randomisations to perform is set to zero, the function will create a text file listing the linear regression results for each analysed phylogenetic trees and a pdf file containing histograms displaying linear regression outputs. If the number of randomisations to perform is higher than zero, the function will return a text file listing the p-values obtained for each tree and for each correlation measures and, if there are at least 50 trees, a pdf file containing histograms displaying distributions of these p-values. Currently, several correlation measures are computed and used: the linear regression determination coefficient $R^2_{\text{env}}$ (obtained from the univariate linear regression between the dispersal durations and the environmental weights computed for a given environmental factor), its corresponding p-value, the difference $D$ (difference between $R^2_{\text{env}}$ and $R^2_{\text{null}}$, the determination coefficient obtained from the linear regression between the dispersal durations and the geographical distances associated with each branch). Note that the geographical distance is computed using the selected environmental weight computation method on a “null” raster with uniform cell values equal to 1.

Note

The computation of random walk weights requires the preliminary installation of the software Circuitscape as a Python package.
Instructions on how to install this Python package are available on Circuitscape website:
http://docs.circuitscape.org/circuitscape_4_0_user_guide.html?id=gsite

Author(s)

Simon Dellicour

References


spreadGraphic

A function to generate a raster file estimating the dispersal history

Description

This function generates a raster made by the superimposition of distinct layers corresponding to successive time slices. Each time slice layer is build by estimating the HPD (highest posterior density) region based on all the ending positions of phylogenetic branches whose ending time falls within the considered time slice. See also the software SPREAD of Bielejec et al. (2011) for a very similar approach that also allows to display the outputs in Google Earth.

Usage

spreadGraphic(localTreesDirectory, nberOfExtractionFiles, rast, prob = 0.05, startDatum, precision = 1, showingPlots = F, nberOfCores = 1, origin = F, timeLayers = F)

Arguments

localTreesDirectory
directory where the extracted matrices are saved (see the "treeExtractions" function).
nberOfExtractionFiles
number of files created with the "treeExtractions" function.
rast
template raster that will be used to map the dispersal history (raster cell values will not be taken into account).
prob
probability corresponding to the HPD (highest posterior density) regions.
startDatum
number in a decimal format (e.g. "2007.4") defining the beginning of the dispersal history.
precision
number, time interval that will be used to define the successive time slices.
showingPlots
boolean variable specifying if the final raster object has to be displayed or not.
nberOfCores
number of available cores to parallelise the estimation of the different time slice layers.
origin
boolean variable specifying if the function has to rather estimate the HPD region of the spread origin (default value is "FALSE": see the details).
timeLayers
boolean variable specifying if the function has to rather return a list of raster objects corresponding to the successive combination of time slice layers (default value is "FALSE"; useful e.g. to generate a movie).
spreadStatistics

Details

If the "origin" boolean variable is set to "TRUE", the function will only estimate the HPD region of the spread origin. In that particular case, the function will only focus on the position of the most ancestral node of each extraction matrix, no matter their associated time. This latter aspect differs from the main option ("origin = FALSE", i.e. estimation of the dispersal history).

Value

The function returns a raster object or a list of raster objects in the case where "timeLayers" equals "TRUE".

Author(s)

Simon Dellicour, Nuno Faria

References


See Also

http://www.kuleuven.be/aidslab/phylogeography/SPREAD.html

---

spreadStatistics

A function to compute several dispersion/epidemiological statistics

Description

This function estimates several dispersion/epidemiological statistics based on the spatio-temporal information extracted from phylogenetic trees by the "treeExtractions" function.

Usage

```r
spreadStatistics(localTreesDirectory, nberOfExtractionFiles, onlyTipBranches = F, showingPlots = T, name = gsub(" ","_",date()))
```

Arguments

- **localTreesDirectory**: directory where the extracted matrices are saved (see the "treeExtractions" function).
- **nberOfExtractionFiles**: number of files created with the "treeExtractions" function.
onlyTipBranches

boolean variable defining if the statistic estimations have to be performed only while considering the tip branches of the trees.

showingPlots

boolean variable specifying if the different plots have to be displayed or not.

outputName

name (prefix) to give to the different output files.

Value

The function generates several outputs. The first output is a text file containing the mean dispersal velocity and diffusion coefficient, as well as dispersal velocity and diffusion coefficient variation among lineages, estimated for each phylogenetic tree. See Pybus et al. (2012) for the formula used to estimate diffusion coefficients. The function also generates and automatically saves figures displaying the kernel density estimates of dispersal velocity and diffusion coefficient parameters (coefficient variations against mean values). Finally, if the parameter onlyTipBranches = FALSE, the function will also generate and save graphs displaying the evolution of the maximal wavefront distance. Two types of wavefront distance are estimated: a spatial distance computed as the straight-line distance from the estimated location of the ancestral node, and a patristic distance computed as the sum of geographical distances of each branch connecting a given node to the ancestral node. On these graphs, grey area corresponds to the 95% credible region of the estimated wavefront position. Note that geographic distances used to estimate the different statistics are great circle distances computed with the function "rdist.earth" from the R package "fields" (Nychka et al. 2014).

Author(s)

Simon Dellicour

References


torusRandomisation

A function to perform the torus randomisation of an original raster

Description

This function generates a randomised raster by performing random torus translation and reflection of an original raster (vertical and horizontal reflections both having a probability of occurrence equals to 0.5). Torus translation and reflection randomisation can be used to break down the association between variables while keeping their respective spatial autocorrelation pattern intact, as far as possible (Harms et al. 2001, Hardy 2009).

Usage

torusRandomisation(rast)
Arguments

rast  raster to randomise with a torus translation and reflection.

Details

This function can be called by the "spreadFactor" function.

Value

The function returns a raster object with cell values randomized by a torus translation and reflection.

Author(s)

Simon Dellicour

References


Hardy OJ (2009). TOROCOR: a program to assess the association between spatially autocorrelated variables using a torus-translation test on multiple grids.

treeExtractions  A function to extract the spatio-temporal information contained in spatiotemporally-referenced phylogenetic trees.

Description

This function extracts the spatio-temporal information contained in the phylogenetic trees inferred e.g. by the continuous phylogeographic method implemented in BEAST (Lemey et al. 2010).

Usage

treeExtractions(localTreesDirectory, allTrees, burnIn, randomSampling, nberOfTreesToSample, mostRecentSamplingDatum, coordinateAttributeName)

Arguments

localTreesDirectory  name of the (new) directory where the extracted matrices have to be saved.

allTrees  all the trees in a Nexus format.

burnIn  number of trees to discard as burn-in.
randomSampling

- boolean variable specifying if the trees have to be randomly sampled in the list. If FALSE, the trees will be sampled at a regular interval (at the largest possible interval).

numberOfTreesToSample

- number of trees to sample.

mostRecentSamplingDatum

- most recent datum of sampling in decimal format (e.g. "2007.4").

coordinateAttributeName

- attribute name used to indicate the geographic coordinates within the trees file.

Value

The function generates "csv" files (one per phylogenetic tree) containing the spatio-temporal information extracted from phylogenetic trees.

Author(s)

Rebecca Rose, Simon Dellicour

References


---

### variogramModel

A function to estimate variogram model(s)

#### Description

Using the functions "vgm" and "fit.variogram" of the "gstat" package (Pebesma 2004) to estimate the best variogram model(s) for one or several raster files.

#### Usage

```r
variogramModel(envVariables)
```

#### Arguments

- `envVariables` list of environmental rasters for which a variogram model has to be estimated.

#### Value

The function returns a list of variogram models, one per considered raster file.
vectorRandomisation

Note

This function involves graphics generation and thus cannot be launched from a terminal window. It has to be called from a R interface.

Author(s)

Simon Dellicour

References


vectorRandomisation

*A function to randomise the position of a set of movement vectors*

Description

This function returns randomised coordinates for set of vectors each defined by a start and end positions. The position of each vector is randomised on a template raster while avoiding that new start and/or new end position fall on a cell with a NODATA ("NA") value. During the randomization, vectors length (i.e. euclidian distance between start and end locations) is not modified.

Usage

```r
vectorRandomisation(rast, fromCoor, toCoor, showingPlots = FALSE)
```

Arguments

- `rast` template raster on which the position of vectors will be randomised.
- `fromCoor` matrix containing the coordinates of the start position of each movement vector (one line per vector, 1st column with the longitude and the 2nd column with the latitude).
- `toCoor` matrix containing the coordinates of the end position of each movement vector (one line per vector, 1st column with the longitude and the 2nd column with the latitude).
- `showingPlots` boolean variable specifying if the plot displaying the randomised vectors has to be displayed or not.

Details

This function can be used to randomise the movement vectors obtained, e.g., by CMR (capture-mark-recapture) or GPS data.
**Value**

The function returns a matrix of randomised vector coordinates with one line per vector: new starting longitude, new starting latitude, new ending longitude and new ending latitude.

**Author(s)**

Simon Dellicour

**References**
