Package 'BPEC'

April 3, 2018

Type Package

Title Bayesian Phylogeographic and Ecological Clustering

Version 1.2.1

Date 2018-04-03

Author Ioanna Manolopoulou, Axel Hille

Maintainer Ioanna Manolopoulou <ioanna.manolopoulou@gmail.com>

Description Model-based clustering for phylogeographic data comprising mtDNA sequences and geographical locations along with optional environmental characteristics, aiming to identify migration events that led to homogeneous population clusters.

Depends

Imports
ggplot2,igraph,mvtnorm,maptools,sp,R2G2,phytools,fields,coda,methods,OpenStreetMap,ggmap,ape

License GPL-2

LazyLoad yes

NeedsCompilation yes

Repository CRAN

Date/Publication 2018-04-03 20:28:24 UTC

R topics documented:

---

bpec *Bayesian Phylogeographic and Ecological Clustering (BPEC)*

Description

Bayesian Phylogeographic and Ecological Clustering (BPEC) is aimed at identifying genetically, geographically and ecologically distinct population clusters and drawing inferences about ancestral locations. Given a dataset of DNA sequences (non-recombinant, typically mtDNA) and their respective geographical locations (longitude and latitude) along with additional environmental or phenotypic characteristics, the algorithm simultaneously draws inferences about the genealogy (in the form of a haplotype tree) and the population clustering. In addition, the algorithm identifies locations with high posterior probability of being ancestral.

Usage

```
bpec(seqsFile, coordsFile, dims = 2, iter = 100000, postSamples = 100,
     maxMig = 5, ds = 0, colorCode = c(7,5,6,3,2,8))

## S3 method for class 'bpec'
plot(x, GoogleEarth = 0, colorCode = c(7,5,6,3,2,8),...)
## S3 method for class 'bpec'
summary(object,...)
## S3 method for class 'bpec'
print(x,...)
## S3 method for class 'bpec'
mean(x,...)
```

Arguments

x	R object from bpec.mcmc run.
object	R object from bpec.mcmc run.
colorCode	A vector of color codes to use, ideally the same ones used in bpec.contourPlot.
seqsFile	The name of the NEXUS file in full, eg "SeqsFile.nex".
coordsFile	the name of the coordinate and sequence file in full, eg "CoordsLocs.txt".

maxMig	The maximum number of migration events (this means that the maximum number of clusters will be maxmig+1). The number you enter here is just an upper bound, so start with maxmig=6 and only increase it if you are really getting 7 clusters in return (which could mean that more clusters are appropriate). If, say, 4 clusters are needed, whether you use maxmig=6 or maxmig=10 (or similar), the number of clusters will collapse down to 4.
iter	The number of iterations for the MCMC sampler, must be a multiple of 10. You will need quite a large number here, like 100,000. Two MCMC chains will run, after which convergence is checked. If convergence has not been reached, the output will say "NO CONVERGENCE" and you should increase the number of iterations.
ds	This represents the parsimony relaxation parameter, with 0 being the minimum. Generally the higher ds, the more candidate trees are considered, but this comes at a computational cost. Start with ds=0 and increase to ds=1, etc, observing any changes.
postSamples	How many posterior samples per chain to save for use post-processing. A value of at least PostSamples=1000 would provide a reasonable assessment of posterior uncertainty. PostSamples must not be greater than iter/10. Also, only up to 20 saved (thinned) samples are used in the bpec.contourPlot function.
dims	The dimension, 2 for purely geographical data, +1 for each covariate (for example if environmental or phenotypic characteristics are also available).
GoogleEarth	If 1, .kml files are produced which can be opened with GoogleEarth.
...	Other default options.

Details

bpec requires 2 input files in order to run:

haplotypes.nex : The sequence file in NEXUS format. Sequence labels should either be integers, or contain unique integers which correspond to the labels in the CoordsLocsFile.txt. For example, '1', '1_label', 'label1_label' will all be treated as haplotype 1. NOTE: bpec will currently ignore unknown nucleotides in the inference.

coordsLocsFile.txt : For each observation, the coordinates (latitude and longitude, please use a +/- to indicate W or E), any other environmental or phenotypic covariates (the latitude and longitude MUST come first), plus the ID number of the haplotype (must match the number in the sequence NEXUS file). If more than one haplotype were found at a single location, these can be entered one after the other, eg:

```
36.88 -5.42  24  25
37.00 -3.98 245 251 243 142 143 244 246 247
43.35  1.48 153
```

so, in the first location (lat/long 36.88, -5.42) you have 2 sampled individuals with haplotypes 24,25, in the second location eight individuals etc. Sequences don't necessarily need to be collapsed onto haplotypes, the program should take care of it.

The main function is bpec.mcmc and runs a Markov chain Monte Carlo sampler in order to obtain posterior samples for all the parameters simultaneously: the haplotype tree, the ancestral nodes, the number of population clusters as well as their means and covariances.

Three plotting functions are available. `bpec.contourPlot` shows the inferred population clusters superimposed on a world map. `bpec.treePlot` shows the Maximum A Posteriori rooted haplotype tree, indicating posterior cluster membership and number of times each haplotype was sampled. It can also output .kml files which can be loaded into Google Earth. `bpec.CovariatesPlot` shows the posterior density estimates for each additional covariate of each cluster.

Value

`bpec` object which can be analysed using `input()`, `data()`, `clust()`, `tree()` and `mcmc()`.

Note

`bpec.mcmc` uses `cexcept.h 2.0.1` (an interface for exception-handling in ANSI C) developed by Adam M. Costello and Cosmin Truta.

Author(s)

Ioanna Manolopoulou <ioanna.manolopoulou@gmail.com>, Axel Hille <axel.hille@gmx.net>

References

I. Manolopoulou and B.C. Emerson (2012). Phylogeographic ancestral inference using the coalescent model on haplotype trees. *Journal of Computational Biology*, 19(6), 745-755.

I. Manolopoulou, L. Legarreta, B.C. Emerson, S. Brooks, and S. Tavar? (2011). A Bayesian approach to phylogeographic clustering. *Interface focus*, rsfs20110054.

S.P. Brooks, I. Manolopoulou, and B.C. Emerson (2007). *Assessing the Effect of Genetic Mutation - A Bayesian Framework for Determining Population History from DNA Sequence Data*. Bayesian Statistics 8. Oxford University Press.

Adam M. Costello and Cosmin Truta (2008) `cexcept.h` exception handling interface in C, available at website http://www.nicemice.net/cexcept/.

Examples

```
## if you want to load the `mini' example Brown Frog dataset
data(MacrocnemisRawSeqs)
data(MacrocnemisCoordsLocsMini)
rawSeqs <- MacrocnemisRawSeqs
coordsLocs <- MacrocnemisCoordsLocsMini

dims <- 3 #this is 2 if you only have geographical longitude/latitude.
#(add 1 for each environmental or phenotypic covariate)
maxMig <- 2 #you will need a higher maximum number of migrations, suggest 7
ds <- 0 #start with ds=0 and increase to 1 and then to 2
iter <- 1000 #you will need far more iterations for convergence, start with 100,000
postSamples <- 100 #you will need at least 100 saved posterior samples

#run the Markov chain Monte Carlo sampler
bpecout <- bpec.mcmc(rawSeqs,coordsLocs,maxMig,iter,ds,postSamples,dims)

par(mar=c(0,0,0,0),pty="m",mfrow=c(1,2))  #no plot margins, plot contours and tree side-by-side
```

```
# plot geographical cluster contour map
bpec.contourPlot(bpecout,GoogleEarth=0)

# plot tree network with cluster indicators
bpec.Tree <- bpec.treePlot(bpecout)

# now also plot the environmental covariates
 bpec.covariatesPlot(bpecout)

 bpec.Geo <- bpec.geoTree(bpecout,file="GoogleEarthTree.kml")


## Not run:
# if you want to load the example burnet moth dataset
data(TransalpinaRawSeqs)
data(TransalpinaCoordsLocs)
rawSeqs <- TransalpinaRawSeqs
coordsLocs <- TransalpinaCoordsLocs

##if you want to use your own dataset, use setwd() to enter the correct folder,
##then run the command below, changing the input parameters if necessary
 #rawSeqs <- bpec.loadSeq('haplotypes.nex')
 #coordsLocs <- bpec.loadCoords("coordsLocsFile.txt")

 ## to set phenotypic/environmental covariate names manually, use (as appropriate)
# colnames(CoordsLocs)[1:dims] <- c('lat','long','cov1','cov2','cov3')
## where dims is the corresponding number of measurements available
## (2 for latitude and longitude only, add one for each additional available measurement)

dims <- 2 #this is 2 if you only have geographical longitude/latitude.
#(add 1 for each environmental or phenotypic covariate)
maxMig <- 5 #you will need a higher maximum number of migrations, suggest 7
ds <- 0 #start with ds=0 and increase to 1 and then to 2
iter <- 10000 #you will need far more iterations for convergence, start with 100,000
postSamples <- 2 #you will need at least 100 saved posterior samples

#run the Markov chain Monte Carlo sampler
bpecout <- bpec.mcmc(rawSeqs,coordsLocs,maxMig,iter,ds,postSamples,dims)

par(mar=c(0,0,0,0),pty="m",mfrow=c(1,2)) #No plot margins. Contours and tree side-by-side
# plot geographical cluster contour map
bpec.contourPlot(bpecout, GoogleEarth=0, mapType = 'plain')

# plot tree network with cluster indicators
bpec.Tree <- bpec.treePlot(bpecout)

## if you want to load the example Brown Frog dataset
data(MacrocnemisRawSeqs)
data(MacrocnemisCoordsLocs)
rawSeqs <- MacrocnemisRawSeqs
coordsLocs <- MacrocnemisCoordsLocs

dims <- 8 #this is 2 if you only have geographical longitude/latitude.
```

```
#(add 1 for each environmental or phenotypic covariate)
maxMig <- 4 #you will need a higher maximum number of migrations, suggest 7
ds <- 2 #start with ds=0 and increase to 1 and then to 2
iter <- 10000 #you will need far more iterations for convergence, start with 100,000
postSamples <- 2 #you will need at least 100 saved posterior samples

#run the Markov chain Monte Carlo sampler
bpecout <- bpec.mcmc(rawSeqs,coordsLocs,maxMig,iter,ds,postSamples,dims)

par(mar=c(0,0,0,0),pty="m",mfrow=c(1,2))  #no plot margins, plot contours and tree side-by-side
# plot geographical cluster contour map
bpec.contourPlot(bpecout,GoogleEarth=0)

# plot tree network with cluster indicators
bpec.Tree <- bpec.treePlot(bpecout)

# now also plot the environmental covariates
 par(mfrow=c(2,3)) #split the plot window into 2x3 to fit all the covariates
 bpec.covariatesPlot(bpecout)

 bpec.Geo <- bpec.geoTree(bpecout,file="GoogleEarthTree.kml")

## End(Not run)
```

`bpec.contourPlot` *Plot contours of inferred clusters on geographical map*

Description

Plots the contours for a set of population clusters, adding points for sampling locations, with larger points representing inferred ancestral locations. Assuming that the data input is in latitude and longitude form, the contour is superimposed onto the map of the corresponding region.

Usage

```
bpec.contourPlot(bpecout, GoogleEarth = 0,
colorCode = c(7,5,6,3,2,8,4,9), mapType = 'plain', mapCentre = NULL, zoom = 6)
```

Arguments

`bpecout`	R object from `bpec.mcmc` run
`GoogleEarth`	If 1, .kml files are produced which can be opened with GoogleEarth.
`colorCode`	(optional) A vector of color codes.
`mapType`	Options are 'none' (no map shown in the background), 'plain' (outline shown only), 'google' (google maps), 'osm' (openstreetview maps).

mapCentre	(optional) The longitude and latitude to use as the centre of the map. If not provided, the midpoint of sampling range will be used.
zoom	The level of zoom into the map, default 6.

Details

Contours with level 0.5 of the posterior mean centres amd covariances of the clusters are shown on the map, with colour representing cluster index. Posterior samples are shown with semi-transparent filled contours, so that uncertainty can be represented directly on the map together with the posterior means. The sampling locations are shown as black dots, with the top three (in terms of posterior probability) ancestral locations represented by larger dots. If `GoogleEarth=1`, this same plot is exported in files that can be directly loaded into Google Earth.

Value

Produces a contour plot overlayed on a map. If `GoogleEarth=1`, files which may be loaded directly into Google Earth are produced.

Author(s)

Ioanna Manolopoulou, Axel Hille and Steve Brooks

References

I. Manolopoulou, L. Legarreta, B.C. Emerson, S. Brooks, and S. Tavaré (2011). A Bayesian approach to phylogeographic clustering. *Interface focus*, rsfs20110054.

S.P. Brooks, I. Manolopoulou, and B.C. Emerson (2007). *Assessing the Effect of Genetic Mutation - A Bayesian Framework for Determining Population History from DNA Sequence Data*. Bayesian Statistics 8. Oxford University Press.

Examples

```
#to use example dataset:
data(MacrocnemisRawSeqs)
data(MacrocnemisCoordsLocs)
rawSeqs <- MacrocnemisRawSeqs
coordsLocs <- MacrocnemisCoordsLocs

##to use your own dataset:
#rawSeqs <- bpec.loadSeq('Haplotypes.nex')
#coordsLocs <- bpec.loadCoords("coordsLocsFile.txt")

## to set phenotypic/environmental covariate names manually, use (as appropriate)
# colnames(coordsLocs)[1:dims] <- c('lat','long','cov1','cov2','cov3')
## where dims is the corresponding number of measurements available
## (2 for latitude and longitude only, add one for each additional available measurement)

#for the analysis:
#check the helpfile of bpec.mcmc using ?bpec.mcmc
colorCode <- c(7,5,6,3,2,8) #default colour scheme
bpecout <- bpec.mcmc(rawSeqs, coordsLocs, maxMig = 2, iter = 20, ds = 0, postSamples = 2, dims = 8)
```

```
bpec.contourPlot(bpecout, GoogleEarth = 0, colorCode, mapType = 'plain')
```

`bpec.covariatesPlot` *Plot posterior marginal distribution of each covariate for the inferred clusters*

Description

For each covariate, the posterior marginal distribution for each cluster is shown, with colours representing cluster index. The posterior median as well as .10 and .90 quantiles are shown on the same plot in order to provide a measure of uncertainty.

Usage

```
bpec.covariatesPlot(bpecout, colorCode=c(7,5,6,3,2,8,4,9))
```

Arguments

`bpecout`	R object from `bpec.mcmc` run
`colorCode`	A vector of color codes

Details

Unlike the contour plot, where all posterior samples are shown on the map, in this case quantiles are used pointwise along the axis of each covariate. The plots are shown in the original scale of the covariates and axis labels can be fed into the function to aid interpretation.

Value

Produces plots for all additional (environmental or phenotypic) covariate, where posterior medians as well as credible intervals are shown for each cluster.

Author(s)

Ioanna Manolopoulou

References

I. Manolopoulou, L. Legarreta, B.C. Emerson, S. Brooks, and S. Tavaré (2011). A Bayesian approach to phylogeographic clustering. *Interface focus*, rsfs20110054.

Examples

```
#to use example dataset:
data(MacrocnemisRawSeqs)
data(MacrocnemisCoordsLocs)
rawSeqs <- MacrocnemisRawSeqs
coordsLocs <- MacrocnemisCoordsLocs

##to use your own dataset
#rawSeqs <- bpec.loadSeq('Haplotypes.nex')
#coordsLocs <- bpec.loadCoords("coordsLocsFile.txt")

## to set phenotypic/environmental covariate names manually, use (as appropriate)
# colnames(CoordsLocs)[1:dims] <- c('lat','long','cov1','cov2','cov3')
## where dims is the corresponding number of measurements available
## (2 for latitude and longitude only, add one for each additional available measurement)

#for the analysis:
#check the helpfile of bpec.mcmc using ?bpec.mcmc

bpecout <- bpec.mcmc(rawSeqs, coordsLocs, maxMig = 2, iter = 100, ds = 0, postSamples = 5, dims = 8)

#if there are also environmental covariates available:
par(mfrow=c(2,3)) #this splits the plot window into 2x3 to fit all the covariates
bpec.covariatesPlot(bpecout)
```

---

bpec.geoTree *Phylogeographic GoogleEarth plot*

Description

Given the output of an MCMC run, outputs phylogeographic code to show migration on a map through GoogleEarth.

Usage

```
bpec.geoTree(bpecout,file="GoogleEarthTree.kml")
```

Arguments

bpecout	R object from bpec.mcmc run
file	Filename for the .kml file to be created

Details

Additionally to the output variables, a .kml file is created which can be loaded into GoogleEarth.

Value

phy	Set of nodes and branches that can be loaded into Phylo2GE()
geo	Set of haplotypes and locations that can be loaded into Phylo2GE()

Author(s)

Axel Hille

References

I. Manolopoulou and B.C. Emerson (2012). Phylogeographic ancestral inference using the coalescent model on haplotype trees. *Journal of Computational Biology*, 19(6), 745-755.

G. Valiente (2009). *Combinational Pattern Matching Algorithms in Computational Biology Using Perl and R*. CRC Press.

N. Arrigo, L.P. Albert, P.G. Mickelson and M.S. Barker (2012). Quantitative visualization of biological data in Google Earth using R2G2, an R CRAN package. *Molecular Ecology Resources*, 12(6), 1177-1179.

Examples

```
## Not run:
#to use example dataset:
data(MacrocnemisRawSeqs)
data(MacrocnemisCoordsLocs)
coordsLocs <- MacrocnemisCoordsLocs
rawSeqs <- MacrocnemisRawSeqs

## to use your own dataset
# rawSeqs <- bpec.loadSeq('Haplotypes.nex')
# coordsLocs <- bpec.loadCoords("coordsLocsFile.txt")

## to set phenotypic/environmental covariate names manually, use (as appropriate)
# colnames(coordsLocs)[1:dims] <- c('lat','long','cov1','cov2','cov3')
## where dims is the corresponding number of measurements available
## (2 for latitude and longitude only, add one for each additional available measurement)

#to run the MCMC sampler:
bpecout <- bpec.mcmc(rawSeqs, coordsLocs, maxMig = 2, iter = 50, ds = 0, postSamples = 5, dims = 8)
bpec.Geo <- bpec.geoTree(bpecout, file = "GoogleEarthTree.kml")

## End(Not run)
```

---

bpec.loadCoords *Load location data*

Description

Create R objects of locations from corresponding file.

Usage

```
bpec.loadCoords(coordsFile, header = FALSE)
```

Arguments

coordsFile	The name of the coordinate and sequence file in full, eg "coordsLocs.txt".
header	If TRUE, then the first row of the file is assumed to be variable names.

Details

Given the names of the file, this command will create an R objects of the coordinates and locations that can be fed into bpec.mcmc.

Value

coordsLocs	R object of locations and corresponding haplotypes.

Author(s)

Ioanna Manolopoulou & Axel Hille

Examples

```
#to load existing example file from BPEC
coordsFile <- system.file("coordsLocsFile.txt",package = "BPEC")
coordsLocs <- bpec.loadCoords(coordsFile, header = TRUE)

## Not run:
#to load a different input file "coordsLocs.txt"
coordsLocs <- bpec.loadCoords("coordsLocs.txt", header = TRUE)

## End(Not run)
```

---

bpec.loadSeq *Load sequence data*

Description

Create R objects of sequences from NEXUS file.

Usage

```
bpec.loadSeq(seqsFile)
```

Arguments

seqsFile	The name of the NEXUS file in full, eg "SeqsFile.nex".

Details

Given the names of the two files, this command will create an R objects of the sequences that can be fed into `bpec.MCMCout`.

Value

`rawSeqs` R object of sequences.

Author(s)

Ioanna Manolopoulou & Axel Hille

Examples

```
#to load existing example file from BPEC
seqsFile <- system.file("haplotypes.nex",package = "BPEC")
rawSeqs <- bpec.loadSeq(seqsFile)

## Not run:
#to load a different input file "haplotypes.nex"
rawSeqs <- bpec.loadSeq('haplotypes.nex')

## End(Not run)
```

`bpec.mcmc` *Markov chain Monte Carlo sampler for BPEC*

Description

Markov chain Monte Carlo for Bayesian Phylogeographic and Ecological Clustering, implemented in C. Given a dataset of DNA sequences (non-recombinant, typically mtDNA) and their respective geographical locations (longitude and latitude), the algorithm simultaneously draws inferences about the genealogy (in the form of a haplotype tree) and the population clustering (with an unknown number of clusters). In addition, the algorithm identifies locations with high posterior probability of being ancestral. In case where additional covariates are available (e.g. climate data), these may be added to the 2-dimensional data and inserted into the analysis.

Usage

```
bpec.mcmc(rawSeqs,coordsLocs,maxMig,iter,ds,postSamples,dims=-1)
```

Arguments

`rawSeqs`	The input DNA sequences.
`coordsLocs`	A matrix where each row shows latitude, longitude (plus any additional covariates), plus all the haplotype numbers found at each location.

maxMig	The maximum number of migration events (this means that the maximum number of clusters will be maxmig+1). The number you enter here is just an upper bound, so start with maxmig=6 and only increase it if you are really getting 7 clusters in return (which could mean that more clusters are appropriate). If, say, 4 clusters are needed, whether you use maxMig=6 or maxMig=10 (or similar), the number of clusters will collapse down to 4.
iter	The number of iterations for the MCMC sampler, must be a multiple of 10. You will need quite a large number here, like 100,000. Two MCMC chains will run, after which convergence is checked. If convergence has not been reached, the output will say "NO CONVERGENCE" and you should increase the number of iterations.
ds	This represents the parsimony relaxation parameter, with 0 being the minimum. Generally the higher ds, the more candidate trees are considered, but this comes at a computational cost. Start with ds=0 and increase to ds=1, etc, observing any changes.
postSamples	How many posterior samples per chain to save for use post-processing. A value of at least PostSamples=1000 would provide a reasonable assessment of posterior uncertainty. PostSamples must not be greater than iter/10. Also, only up to 20 saved (thinned) samples are used in the bpec.contourPlot function.
dims	The dimension, 2 for purely geographical data, +1 for each covariate (for example if environmental or phenotypic characteristics are also available).

Details

This is the main command for BPEC, the details of which can be found in the provided references. In short, the model is such that, for a given haplotype tree, clusterings are the result of the migration of an individual, such that all the descendants of that individual belong to the new cluster (unless they subsequently migrate themselves). The number of migrating individuals is itself a parameter. Conditional on the clustering allocations for each sequence (note: not all samples of a haplotype need to belong to the same cluster), the geographical (and covariate) distribution for each cluster is assumed to be Gaussian. The distributions of longitude and latitude in each cluster are assumed to have an unknown covariance, whereas additional covariates are assumed to be independent.

Posterior samples are obtained through Markov chain Monte Carlo (MCMC) sampling. Two Markov chain Monte Carlo runs are carried out. In each, the sampler sweeps through the updates of all the parameters: the haplotype tree, the root of the tree, the number of clusters, the precise clustering (as a result of migrations) the locations and variances of the clusters. Metropolis-Hastings updates are performed for most of the parameters. In the case of the number of clusters, Metropolis-Hastings Reversible Jump updates are performed.

The space of possible clusterings is vast and highly multi-modal. The migration of an individual to a new cluster implies that all of their descendants will belong to the new cluster. This results in a combinatorial parameter space which is challenging to explore. A number of sophisticated tricks are used in order to overcome this challenge, alternating between local and global MCMC proposal moves. The biggest challenge is to converge to the region of high posterior probability in terms of the number of clusters and the cluster allocation. As such, the first 90 percent of the iterations are discarded as burn-in and only the final 10 percent are used as potential posterior samples.

bpec.mcmc requires 2 input files in order to run:

haplotypes.nex : The sequence file in NEXUS format. Sequence labels should either be integers, or contain unique integers which correspond to the labels in the CoordsLocsFile.txt. For example, '1', '1_label', 'label1_label' will all be treated as haplotype 1. NOTE: BPEC will currently ignore unknown nucleotides in the inference.

coordsLocsFile.txt : For each observation, the coordinates (latitude and longitude, please use a +/- to indicate W or E), any other environmental or phenotypic covariates (the latitude and longitude MUST come first), plus the ID number of the haplotype (must match the number in the sequence NEXUS file). If more than one haplotype were found at a single location, these can be entered one after the other, eg:

```
36.88 -5.42  24  25
37.00 -3.98 245 251 243 142 143 244 246 247
43.35  1.48 153
```

so, in the first location (lat/long 36.88, -5.42) you have 2 sampled individuals with haplotypes 24,25, in the second location eight individuals etc. Sequences don't necessarily need to be collapsed onto haplotypes, the program should take care of it.

Value

bpec object which can be analysed and summarised using the accessor functions input(), dataSummary(), clust(), tree(), mcmc().

Note

bpec.mcmc uses cexcept.h 2.0.1 (an interface for exception-handling in ANSI C) developed by Adam M. Costello and Cosmin Truta.

Author(s)

Ioanna Manolopoulou

References

I. Manolopoulou and B.C. Emerson (2012). Phylogeographic ancestral inference using the coalescent model on haplotype trees. *Journal of Computational Biology*, 19(6), 745-755.

I. Manolopoulou, L. Legarreta, B.C. Emerson, S. Brooks, and S. Tavaré (2011). A Bayesian approach to phylogeographic clustering. *Interface focus*, rsfs20110054.

Adam M. Costello and Cosmin Truta (2008) cexcept.h exception handling interface in C http://www.nicemice.net/cexcept/.

Examples

```
#to use the example dataset:
data(MacrocnemisRawSeqs)
data(MacrocnemisCoordsLocs)
coordsLocs <- MacrocnemisCoordsLocs
rawSeqs <- MacrocnemisRawSeqs

##to use your own dataset
```

```
#rawSeqs <- bpec.loadSeq('Haplotypes.nex')
#coordsLocs <- bpec.loadCoords("coordsLocsFile.txt")

## to set phenotypic/environmental covariate names manually, use (as appropriate)
# colnames(coordsLocs)[1:dims] <- c('lat','long','cov1','cov2','cov3')
## where dims is the corresponding number of measurements available
## (2 for latitude and longitude only, add one for each additional available measurement)

#to run the MCMC sampler:
bpecout <- bpec.mcmc(rawSeqs, coordsLocs, maxMig = 2, iter = 50, ds = 0, postSamples = 5, dims = 8)
```

`bpec.treePlot` *Treeplot indicating cluster membership*

Description

Given a set of haplotype nodes in a tree structure, creates a haplotype tree plot, starting from the root at the top. The size of each haplotype node represents the number of times it was sampled, and color represents population cluster. Unsampled haplotypes are represented by black dots. Thickness of tree edges represents posterior certainty, with thinner edges corresponding to higher uncertainty.

Usage

```
bpec.treePlot(bpecout,colorCode=c(7,5,6,3,2,8,4,9))
```

Arguments

`bpecout`	R object from `bpec.mcmc` run.
`colorCode`	A vector of color codes to use, ideally the same ones used in bpec.ContourPlot.

Details

The algorithm will try to plot the tree such that branches don't cross. The label which will be shown correspond to the original labels provided by the user. In cases where 2 sequences (with different labels) were collapsed to the same haplotype, the smallest (in number) of the two labels will be shown.

Value

`graphEdgesSub`	Set of nodes and branch that can be loaded into `plot.igraph()`.
`graphEdgesTree`	Phylogenetic tree representation that can be plotted directly using `plot()`.

Author(s)

Ioanna Manolopoulou & Axel Hille

References

G. Csardi, T. Nepusz (2006). The igraph software package for complex network research. *InterJournal, Complex Systems*, 1695(5), 1-9.

I. Manolopoulou and B.C. Emerson (2012). Phylogeographic ancestral inference using the coalescent model on haplotype trees. *Journal of Computational Biology*, 19(6), 745-755.

G. Valiente (2009). *Combinational Pattern Matching Algorithms in Computational Biology Using Perl and R*. CRC Press.

Examples

```
#to use example dataset:
data(MacrocnemisRawSeqs)
data(MacrocnemisCoordsLocs)
coordsLocs <- MacrocnemisCoordsLocs
rawSeqs <- MacrocnemisRawSeqs

##to use your own dataset
#rawSeqs <- bpec.loadSeq('haplotypes.nex')
#coordsLocs <- bpec.loadCoords("coordsLocsFile.txt")

## to set phenotypic/environmental covariate names manually, use (as appropriate)
# colnames(coordsLocs)[1:dims] <- c('lat','long','cov1','cov2','cov3')
## where dims is the corresponding number of measurements available
## (2 for latitude and longitude only, add one for each additional available measurement)

#to run the MCMC sampler:

bpecout <- bpec.mcmc(rawSeqs, coordsLocs, maxMig = 2, iter = 50, ds = 0, postSamples = 1, dims = 8)
bpec.Tree <- bpec.treePlot(bpecout)
```

CladeBrownFrogsCoordsLocs

Coordinates and haplotypes for the Brown Frog dataset

Description

Dataset of all known taxa of the "western" and "eastern" lineage of the Near Eastern brown frogs (*Rana macrocnemis macrocnemis, R. m. camerani, R. holtzi, R. m. pseudodalmatina, R. m. tavasensis*, Amphibia, Ranidae). Contains the coordinates of each sampling location along with the haplotypes found in each location. It is called using `data(CladeBrownFrogsCoordsLocs)`.

Usage

```
CladeBrownFrogsCoordsLocs
```

Format

In total 40 sampling locations throughout the Near Eastern (Turkey, Georgia, Daghestan, Northern Iran) with 21 (19 + 2) haplotypes.

References

M. Veith, J.F. Schmidtler, J. Kosuch, I. Baran and A. Seitz (2003) Palaeoclimatic changes explain Anatolian mountain frog evolution: a test for alternating vicariance and dispersal events. Molecular Ecology.

M. Veith, J. Kosuch and M. Vences (2003) Climatic oscillations triggered post-Messinian speciation of Western Palearctic brown frogs (Amphibia, Ranidae). Molecular Phylogenetics and Evolution.

`CladeBrownFrogsRawSeqs`

Sequences for the Brown Frog dataset

Description

Dataset of all known taxa of the "western" and "eastern" lineage of the Near Eastern brown frogs (*Rana macrocnemis macrocnemis, R. m. camerani, R. holtzi, R. m. pseudodalmatina, R. m. tavasensis, Amphibia, Ranidae, R. m. tavasensis*, Amphibia, Ranidae) for the exploration of deeper phylogeographic patterns. Contains the 16S ribosomal mtDNA sequences (n=19) listed in the Appendix II of Veith et al (2003). It is called using `data(CladeBrownFrogsRawSeqs)`.

Usage

```
CladeBrownFrogsRawSeqs
```

Format

In total 40 sampling locations throughout the Near Eastern (Turkey, Georgia, Daghestan, Northern Iran) with 21 (19 + 2) haplotypes of length 536.

References

M. Veith, J.F. Schmidtler, J. Kosuch, I. Baran and A. Seitz (2003) Palaeoclimatic changes explain Anatolian mountain frog evolution: a test for alternating vicariance and dispersal events. Molecular Ecology.

M. Veith, J. Kosuch and M. Vences (2003) Climatic oscillations triggered post-Messinian speciation of Western Palearctic brown frogs (Amphibia, Ranidae). Molecular Phylogenetics and Evolution.

clust *Load location data*

Description

Posterior output for the clustering parameters.

Usage

```
clust(bpecout)

## S3 method for class 'bpec'
clust(bpecout)
```

Arguments

bpecout	R object from bpec.mcmc run

Value

sampleMeansR	A set of posterior samples of the cluster means (i.e. centres).
sampleCovsR	A set of posterior samples of the cluster covariances (i.e. shapes).
sampleIndicesR	A set of posterior samples of the cluster allocations of each observation.
clusterProbsR	For each haplotype, posterior probabilities that it belongs to each cluster.

Author(s)

Ioanna Manolopoulou & Axel Hille

Examples

```
## if you want to load the `mini' example Brown Frog dataset
data(MacrocnemisRawSeqs)
data(MacrocnemisCoordsLocsMini)
rawSeqs <- MacrocnemisRawSeqs
coordsLocs <- MacrocnemisCoordsLocsMini

dims <- 3 #this is 2 if you only have geographical longitude/latitude.
#(add 1 for each environmental or phenotypic covariate)
maxMig <- 2 #you will need a higher maximum number of migrations, suggest 7
ds <- 0 #start with ds=0 and increase to 1 and then to 2
iter <- 1000 #you will need far more iterations for convergence, start with 100,000
postSamples <- 100 #you will need at least 100 saved posterior samples

#run the Markov chain Monte Carlo sampler
bpecout <- bpec.mcmc(rawSeqs,coordsLocs,maxMig,iter,ds,postSamples,dims)
clust(bpecout)
```

input *Load location data*

Description

Shows all input files and settings used in a BPEC run.

Usage

```
input(bpecout)

## S3 method for class 'bpec'
input(bpecout)
```

Arguments

bpecout	R object from bpec.mcmc run

Value

seqCountOrig	The number of input sequences.
seqLengthOrig	The length of the input sequences.
iterR	The number of MCMC iterations.
dsR	The parsimony relaxation parameter.
coordsLocsR	The input coordinate and observation file.
coordsDimsR	The input dimension (2 for purely geographical data).
locNoR	The number of distinct sampling locations.
locDataR	The list of coordinates of each observation.

Author(s)

Ioanna Manolopoulou & Axel Hille

Examples

```
## if you want to load the `mini' example Brown Frog dataset
data(MacrocnemisRawSeqs)
data(MacrocnemisCoordsLocsMini)
rawSeqs <- MacrocnemisRawSeqs
coordsLocs <- MacrocnemisCoordsLocsMini

dims <- 3 #this is 2 if you only have geographical longitude/latitude.
#(add 1 for each environmental or phenotypic covariate)
maxMig <- 2 #you will need a higher maximum number of migrations, suggest 7
ds <- 0 #start with ds=0 and increase to 1 and then to 2
iter <- 1000 #you will need far more iterations for convergence, start with 100,000
```

```
postSamples <- 100 #you will need at least 100 saved posterior samples

#run the Markov chain Monte Carlo sampler
bpecout <- bpec.mcmc(rawSeqs,coordsLocs,maxMig,iter,ds,postSamples,dims)
input(bpecout)
```

`MacrocnemisCoordsLocs` *Coordinates and haplotypes for the R.macrocnemis Brown Frog dataset*

Description

Rana macrocnemis (Amphibia, Ranidae) dataset, the Near Eastern brown frog, from a transition zone of landscapes in Georgia, Caucasus. Contains the coordinates and ecological characteristics of each sampling location along with the haplotypes found in each location. It is called using `data(MacrocnemisCoordsLocs)`.

Usage

```
MacrocnemisCoordsLocs
```

Format

In total 40 sampling locations with one haplotype in each. For each observation, longitude, latitude, plus 6 additional environmental coordinates are available: Four bioclimatic variables from the www.worldclim.org database (Annual Mean Temperature (degrees Celsius x 10), Temperature Annual Range (100 x standard deviation of monthly mean temperature), Annual Precipitation (in mm), Precipitation Seasonality), Coefficient of Variation (CV), altitude in meters as a proxy for a digital elevation model (DEM) and the land cover map (GLC2000) from worldgrids.org. In fact some sampling locations may be identical in terms of longitude and latitude, but are differentiated because of their additional covariates.

References

D. Tarkhnishvili, A. Hille and W. Böhme (2001) Humid forest refugia, speciation and secondary introgression between evolutionary lineages: differentiation in a Near Eastern brown frog, Rana macrocnemis. Biological Journal of the Linnean Society.

A. Hille, I. Manolopoulou and D. Tarkhnisvili (2015) Detecting landscape-dependent selection with Bayesian phylogeographic and ecological clustering: comparison of recombinant microsatellite and mtDNA haplotype data (unpublished).

MacrocnemisCoordsLocsMini *Coordinates and haplotypes for the R.macrocnemis Brown Frog dataset*

Description

A reduced version of the *Rana macrocnemis* (Amphibia, Ranidae) dataset, the Near Eastern brown frog, from a transition zone of landscapes in Georgia, Caucasus. Contains the coordinates and ecological characteristics of each sampling location along with the haplotypes found in each location. It is called using `data(MacrocnemisCoordsLocsMini)`.

Usage

```
MacrocnemisCoordsLocsMini
```

Format

In total 21 sampling locations with one haplotype in each. For each observation, longitude, latitude, plus Annual Mean Temperature (degrees Celsius x 10) from the www.worldclim.org database.

References

D. Tarkhnishvili, A. Hille and W. Böhme (2001) Humid forest refugia, speciation and secondary introgression between evolutionary lineages: differentiation in a Near Eastern brown frog, Rana macrocnemis. Biological Journal of the Linnean Society.

A. Hille, I. Manolopoulou and D. Tarkhnisvili (2015) Detecting landscape-dependent selection with Bayesian phylogeographic and ecological clustering: comparison of recombinant microsatellite and mtDNA haplotype data (unpublished).

MacrocnemisRawSeqs *Sequences for the R.macrocnemis Brown Frog dataset*

Description

Rana macrocnemis (Amphibia, Ranidae) dataset, the Near Eastern brown frog, from a transition zone of landscapes in Georgia, Caucasus. Contains the mtDNA sequences (cytochrome b) of the Brown Frog dataset used in the BPEC examples. It is called using `data(MacrocnemisRawSeqs)`.

Usage

```
MacrocnemisRawSeqs
```

Format

In total 40 mtDNA sequences of length 504 nucleotides each.

References

D. Tarkhnishvili, A. Hille and W. Böhme (2001) Humid forest refugia, speciation and secondary introgression between evolutionary lineages: differentiation in a Near Eastern brown frog, Rana macrocnemis. Biological Journal of the Linnean Society.

A. Hille, I. Manolopoulou and D. Tarkhnisvili (2015) Detecting landscape-dependent selection with Bayesian phylogeographic and ecological clustering: comparison of recombinant microsatellite and mtDNA haplotype data, (unpublished).

`mcmc` *Load location data*

Description

Provides various MCMC tuning parameters, as well as posterior samples for convergence assessment.

Usage

```
mcmc(bpecout)

## S3 method for class 'bpec'
mcmc(bpecout)
```

Arguments

`bpecout`	R object from `bpec.mcmc` run

Value

`MCMCparamsR`	Various MCMC tuning parameters, useful for development.
`codaInput`	Posterior samples from the two MCMC chains for the cluster means, cluster covariance entries, as well as the root haplotype. Note that, since the number of clusters varies from iteration to iteration, some samples are simply draws from the prior (corresponding to empty clusters). This variable can be loaded directly into the `coda` package for convergence analysis.

Author(s)

Ioanna Manolopoulou & Axel Hille

Examples

```
## if you want to load the `mini' example Brown Frog dataset
data(MacrocnemisRawSeqs)
data(MacrocnemisCoordsLocsMini)
rawSeqs <- MacrocnemisRawSeqs
coordsLocs <- MacrocnemisCoordsLocsMini

dims <- 3 #this is 2 if you only have geographical longitude/latitude.
#(add 1 for each environmental or phenotypic covariate)
maxMig <- 2 #you will need a higher maximum number of migrations, suggest 7
ds <- 0 #start with ds=0 and increase to 1 and then to 2
iter <- 1000 #you will need far more iterations for convergence, start with 100,000
postSamples <- 100 #you will need at least 100 saved posterior samples

#run the Markov chain Monte Carlo sampler
bpecout <- bpec.mcmc(rawSeqs,coordsLocs,maxMig,iter,ds,postSamples,dims)
mcmc(bpecout)
```

---

preproc *Load location data*

Description

Provides various MCMC tuning parameters, as well as posterior samples for convergence assessment.

Usage

```
preproc(bpecout)

## S3 method for class 'bpec'
preproc(bpecout)
```

Arguments

bpecout	R object from bpec.mcmc run

Value

seqR	The output DNA sequences of distinct haplotypes, collapsed to effective nucleotide sites (both sampled and missing sequences which were inferred).
seqsFileR	A vector of the numerical labels of each haplotype.
seqLabelsR	Correspondence vector for each of the processed observations to the original haplotype labels.
seqIndicesR	Correspondence vector for each of the original observations to the resulting haplotype labels.

seqLengthR	The effective length of the input sequences, given by the number of variable nucleotide sites which are informative. In other words, if two different nucleotide sites are variable in exactly the same haplotypes, then they effectively provide information of a single site.
noSamplesR	The number of times each haplotype was observed in the sample.
countR	The number of output sequences.

Author(s)

Ioanna Manolopoulou & Axel Hille

Examples

```
## if you want to load the `mini' example Brown Frog dataset
data(MacrocnemisRawSeqs)
data(MacrocnemisCoordsLocsMini)
rawSeqs <- MacrocnemisRawSeqs
coordsLocs <- MacrocnemisCoordsLocsMini

dims <- 3 #this is 2 if you only have geographical longitude/latitude.
#(add 1 for each environmental or phenotypic covariate)
maxMig <- 2 #you will need a higher maximum number of migrations, suggest 7
ds <- 0 #start with ds=0 and increase to 1 and then to 2
iter <- 1000 #you will need far more iterations for convergence, start with 100,000
postSamples <- 100 #you will need at least 100 saved posterior samples

#run the Markov chain Monte Carlo sampler
bpecout <- bpec.mcmc(rawSeqs,coordsLocs,maxMig,iter,ds,postSamples,dims)
preproc(bpecout)
```

TransalpinaCoordsLocs *Coordinates and haplotypes for the burnet moth dataset.*

Description

The dataset of three closely related forms of a European burnet moth with a complicated micro-vicariant distribution pattern, the *Zygaena transalpina* complex (Lepidoptera, Zygaenidae). Contains the coordinates of each sampling location along with the haplotypes found in each location. It is called using data(TransalpinaCoordsLocs).

Usage

```
TransalpinaCoordsLocs
```

Format

In total 55 sampling locations with rich population samples per location revealing plenty of distinct haplotypes.

Author(s)

compiled by Axel Hille using supplementary material of von Reumont et al (2012).

References

B.M. von Reumont, J.-F. Struwe, J. Schwarzer and B. Misof (2012) Phylogeography of the burnet moth Zygaena transalpina-complex: molecular and morphometrical differentiation suggests glacial refugia in Southern France, Western Europe and micro refugia within the Alps. Journal of Zoological Systematics and Evolutionary Research.

`TransalpinaRawSeqs` *Sequences for the burnet moth dataset.*

Description

The dataset of three closely related forms of a European burnet moth with a complicated microvicariant distribution pattern, the *Zygaena transalpina* complex (Lepidoptera, Zygaenidae). Contains the coordinates of each sampling location along with the haplotypes found in each location. It is called using data(TransalpinaRawSeqs).

Usage

```
TransalpinaRawSeqs
```

Format

In total 472 mtDNA sequences of length 497 nucleotides each.

Author(s)

compiled by Axel Hille using supplementary material of von Reumont et al (2012).

References

B.M. von Reumont, J.-F. Struwe, J. Schwarzer and B. Misof (2012) Phylogeography of the burnet moth Zygaena transalpina-complex: molecular and morphometrical differentiation suggests glacial refugia in Southern France, Western Europe and micro refugia within the Alps. Journal of Zoological Systematics and Evolutionary Research.

tree *Load location data*

Description

Posterior output for the tree model.

Usage

```
tree(bpecout)
## S3 method for class 'bpec'
tree(bpecout)
```

Arguments

bpecout	R object from bpec.mcmc run

Value

cladoR	The MAP adjacency matrix for the tree in vectorised format: this means that for two haplotypes i,j, the (i,j)th entry of the matrix is 1 if the haplotypes are connected in the network and 0 otherwise.
levelsR	Starting from the root (level 0) all the way to the tips, the discrete depth for the Maximum A Posteriori tree plot.
edgeTotalProbR	Posterior probabilities of each edge being present, i.e. corresponding to a mutation which occurred.
rootProbsR	The posterior probability per chain that each haplotype was the root of the tree.
treeEdgesR	The set of edges (from and to haplotypes) of the Maximum A Posteriori haplotype tree (could be used in another program if needed).
rootLocProbsR	Vector of posterior probabilities of each sampling location being the ancestral location.
migProbsR	The posterior probability of 0...maxMig migrations.

Author(s)

Ioanna Manolopoulou & Axel Hille

Examples

```
## if you want to load the `mini' example Brown Frog dataset
data(MacrocnemisRawSeqs)
data(MacrocnemisCoordsLocsMini)
rawSeqs <- MacrocnemisRawSeqs
coordsLocs <- MacrocnemisCoordsLocsMini

dims <- 3 #this is 2 if you only have geographical longitude/latitude.
```

```
#(add 1 for each environmental or phenotypic covariate)
maxMig <- 2 #you will need a higher maximum number of migrations, suggest 7
ds <- 0 #start with ds=0 and increase to 1 and then to 2
iter <- 1000 #you will need far more iterations for convergence, start with 100,000
postSamples <- 100 #you will need at least 100 saved posterior samples

#run the Markov chain Monte Carlo sampler
bpecout <- bpec.mcmc(rawSeqs,coordsLocs,maxMig,iter,ds,postSamples,dims)
tree(bpecout)
```

Index