

# Package ‘sidier’

December, 2012

**Version** 0.1

**Date** 2012-12-01

**Title** Substitution and Indel Distances to Infer Evolutionary Relationships

**Author** A. Jesús Muñoz-Pajares

**Maintainer** A. Jesús Muñoz-Pajares <ajesusmp@ugr.es>

**Imports** ape, igraph, network

**ZipData** no

**Description** sidier provides functions for reading and writing fasta sequences, finding unique haplotypes, estimating genetic distances based on gap positions and lengths, combining distance matrices and estimating and plotting percolation networks.

## R topics documented:

sidier-package . . . . .	2
FindHaplo . . . . .	2
GetHaplo . . . . .	3
HapPerPop . . . . .	4
MCIC . . . . .	6
nt.gap.comb . . . . .	7
perc.thr . . . . .	8
plot.network . . . . .	10
pop.dist . . . . .	11
Index . . . . .	12

---

sidier-package

The sidier package

---

### Description

sidier is a library and R package for evolutionary reconstruction based on substitutions and insertion-deletion (indels) analyses in a distance-based framework.

### References

Muñoz-Pajares, A.J., Abdelaziz, M., Gómez, J.M., Perfectti, F. Combining indels and substitutions information for the reconstruction of evolutionary haplotype relationships

Muñoz-Pajares, A.J., Abdelaziz, M., Herrador, M.B., Gómez, J.M., Perfectti, F. Phylogeography and colonization pathways of the *Erysimum nevadense* species complex based on a plastidial indel-rich region distance analysis

---

FindHaplo

Find equal haplotypes

---

### Description

This function assigns the same name to equal haplotypes in a sequence alignment.

### Usage

```
FindHaplo (readfile=T, input=NA, align=NA, saveFile=T, outname="FindHaplo.txt")
```

### Arguments

readfile	a logical; if TRUE (default) input alignment is provided as a fasta format in a text file. If FALSE, the alignment is provided as an R object.
input	the name of the fasta file to be analysed.
align	the name of the alignment to be analysed (if “readfile” is set to FALSE,). See “read.dna” in APE package for details about reading alignments.
saveFile	a logical; if TRUE (default), function output is saved as a text file.
outname	the name of the output file. If “SaveFile” is set to TRUE, but “outname” is not defined, the file is named "FindHaplo.txt" by default

**Value**

A matrix showing sequence names and the assigned haplotype name.

**See Also**

HapPerPop.

**Examples**

```
exampleAlign1<-"example1.fas"

FindHaplo(input=exampleAlign1)

alin<-read.dna(file=exampleAlign1,format="fasta")

FindHaplo(readfile=F,align=alin)

FH<-FindHaplo(readfile=F,align=alin,outname="FindHaplo_custom.txt")
```

---

GetHaplo	Get sequences of unique haplotypes
----------	------------------------------------

---

**Description**

This function returns the subset of haplotypes (that is, unique sequences) in a given alignment.

**Usage**

```
GetHaplo (readfile=T, input=NA, align=NA, saveFile=T, outname="Haplotypes.txt",
format="fasta", seqsNames=NA)
```

**Arguments**

readfile	a logical; if TRUE (default) input alignment is provided as a fasta format in a text file. If FALSE, the alignment is provided as an R object.
input	the name of the fasta file to be analysed.
align	the name of the alignment to be analysed (if “readfile” is set to FALSE,). See “read.dna” in APE package for details about reading alignments.
saveFile	a logical; if TRUE (default), function output is saved as a text file.
outname	the name of the output file. If “SaveFile” is set to TRUE, but “outname” is not defined, the file is named "Haplotypes.txt" by default.

format	format of the DNA sequences saved: "interleaved", "sequential", or "fasta" (default). See "write.dna" in APE package for details.
seqsNames	names for each DNA sequence saved: Three choices are possible: if n unique sequences are found, "Inf.Hap" assign names from H1 to Hn (according to input order). The second option is to define n names. By default (NA), names defined in input are used.

## Details

If two equal sequences are not identically aligned, they will be considered as different haplotypes. To avoid misleading results in uncertain alignments it is recommended to use as input the original unaligned sequences, including gaps after the last nucleotide of short sequences to make all sequence lengths equal.

## Value

a file containing unique sequences from the input file.

## Examples

```
exampleAlign1<-"example1.fas"
```

```
alin<-read.dna(file=exampleAlign1,format="fasta")
```

```
GetHaplo(input=exampleAlign1)
```

```
GetHaplo(readfile=F,align=alin,outname="Haplotypes_custom.txt",seqsNames="Inf.Hap")
```

---

HapPerPop	Returns the number of haplotypes per population.
-----------	--

---

## Description

Given a two column matrix, this function returns the number of haplotypes per population. The matrix must contain one row per individual. The first column must contain the population name, while the second must contain the name of the haplotype. The desired matrix can be obtained using "FindHaplo".

Two output matrices are estimated, one giving the abundance of each haplotype per population (named weighted matrix) and the other representing presence/absence of each haplotype per population by 1/0 (named interaction matrix).

## Usage

```
HapPerPop (readfile=T, sep=" ", header=F, inputFile=NA, input=NA, saveFile=T,
Wname=NA, Iname=NA)
```

## Arguments

<code>readfile</code>	a logical; if TRUE (default) the input matrix is provided in a text file. If FALSE, the matrix is provided as an R object.
<code>sep</code>	the character separating columns in the input matrix (space, by default).
<code>header</code>	a logical value indicating whether the file contains the names of the variables as its first line. (Default=FALSE).
<code>inputFile</code>	(if <code>readfile=TRUE</code> ) the name of the file containing the input matrix.
<code>input</code>	(if <code>readfile=FALSE</code> ) the name of the input matrix as an R object.
<code>saveFile</code>	a logical; if TRUE (default), the two output matrices computed is saved in two different text files.
<code>Wname</code>	the name given to the output weighted matrix file.
<code>Iname</code>	the name given to the output interaction matrix file

## Value

A list containing two matrices. The first matrix contains the weighted matrix, that is, the number of haplotypes (columns) found per population (rows). The second is the interaction matrix, containing information about the presence or absence of each haplotype (columns) per population (rows).

## See also

`FindHaplo`

## Examples

```
exampleAlign1<-"example1.fas"
```

```
alin<-read.dna(file=exampleAlign1,format="fasta")
```

```
HapPerPop(readfile=T,inputFile="FindHaplo_custom.txt",header=T,saveFile=F)
```

```
HapPerPop(readfile=F,input=FH,header=T,saveFile=F)
```

---

MCIC	Modified Complex Indel Coding as distance matrix
------	--

---

### Description

This function computes the insertion-deletion (indel) distance matrix following the rationale of the Modified Complex Indel Coding (Müller, 2006) to estimate transition matrices.

### Usage

```
MCIC (readfile = T, input = NA, align = NA, saveFile = T, outname = paste(input, "IndelDistanceMatrixMullerMod.txt"))
```

### Arguments

readfile	a logical; if TRUE (default) input alignment is provided as a fasta format in a text file. If FALSE, the alignment is provided as an R object.
input	the name of the fasta file to be analysed.
align	the name of the alignment to be analysed (if “readfile” is set to FALSE,). See “read.dna” in APE package for details about reading alignments.
saveFile	a logical; if TRUE (default), function output is saved as a text file.
outname	the name of the output file. If “SaveFile” is set to TRUE, but “outname” is not defined, the file is named by default.

### Value

A matrix containing the genetic distances estimated based in indels pairwise differences.

### Examples

```
alin<-read.dna(file=exampleAlign1,format="fasta")
```

```
MCIC (readfile = T, input=exampleAlign1, saveFile = F, onlyUniques = F)
```

### References

Müller K. (2006). Incorporating information from length-mutational events into phylogenetic analysis. *Molecular Phylogenetics and Evolution*, **38**, 667–676.

---

nt.gap.comb	substitution and indel distance combinations
-------------	--

---

## Description

This function obtains a lineal combination of two original matrices. The weight of each matrix in the combination must be defined. If it is a range of values, several matrices are computed.

## Usage

```
nt.gap.comb (DISTnuc=NA, DISTgap=NA, range=seq(0,1,0.1), method="Corrected",
saveFile=TRUE)
```

## Arguments

DISTnuc	a matrix containing substitution genetic distances. See “dist.dna” in “ape” package.
DISTgap	a matrix containing indel genetic distances.
range	a numeric between 0 and 1, is the weights given to the indel genetic distance matrix in the combination. By definition, the weight of the substitution genetic matrix is the complementary value.
method	a string defining whether each distance matrix must be divided by its maximum value before the combination ("Corrected") or not ("Uncorrected"). Consequently, if the “Corrected” method is chosen, both matrices will range between 0 and 1.
saveFile	a logical; if TRUE (default), each output matrix is saved in a different text file.

## Value

A list containing the estimated combination of substitution and indel distance matrices.

## Examples

```
exampleAlign1<-"example1.fas"

distGap<-MCIC(readfile=T,input=exampleAlign1,saveFile=F,onlyUniques==F)

align<-read.dna(exampleAlign1,format="fasta")

dist.nt<-dist.dna(align,model="raw",pairwise.deletion=T)

DISTnt<-as.matrix(dist.nt)
```

```
nt.gap.comb(DISTgap=distGap, range=seq(0,1,0.1), method="Corrected", saveFile=FALSE,
DISTnuc=DISTnt)
```

```
nt.gap.comb(DISTgap=distGap, range=0.5, method="Both", saveFile=FALSE,
DISTnuc=DISTnt)
```

## See also

MCIC

---

perc.thr      Percolation threshold network

---

## Description

This function computes the percolation network following Rozenfeld et al. (2008).

## Usage

```
perc.thr (dis, threshold = seq (0,1,0.01), ptPDF = TRUE, ptPDFname =
"PercolatedNetwork.pdf", estimPDF = TRUE, estimPDFname = "PercThr Estimation.pdf",
estimOutfile = TRUE, estimOutName = "PercThresholdEstimation.txt", appendOutfile =
TRUE, plotALL = FALSE, bgcol = "white", label.col = "black", label = colnames(dis),
modules = FALSE)
```

## Arguments

dis	the distance matrix to be represented
threshold	a numeric vector between 0 and 1, is the range of thresholds to be screened (from 0 to 1, by default).
ptPDF	a logical, must the percolated network be saved as pdf?
ptPDFname	if ptPDF=TRUE, the name to save the percolation network as pdf ("PercolatedNetwork.pdf", by default)
estimPDF	a logical, must the percolation threshold estimation be saved as pdf?
estimPDFname	if estimPDF=TRUE, the name to save the pdf ("PercThr Estimation.pdf", by default)
estimOutfile	a logical, must the matrix containing percolation threshold estimation variables be saved as pdf?
estimOutName	if estimOutfile=TRUE, the name to save the file ("PercThresholdEstimation.txt", by default).



<code>appendOutfile</code>	a logical, if <code>estimOutfile=TRUE</code> , it defines whether results must be appended to an existing file with the same name (TRUE) or not (FALSE).
<code>plotALL</code>	a logical, must all the networks calculated during the percolation threshold estimation be saved as pdf? (FALSE, by default). If TRUE, for each value in threshold, one file is generated. It will increase computation time.
<code>bgcol</code>	string, defining the colour of the background for each node in the network. Can be equal for all nodes (if only one colour is defined), can be customized (if several colours are defined), or can represent different modules (see <code>modules</code> option).
<code>label.col</code>	string, defining the colour of labels for each node in the network. Can be equal for all nodes (if only one colour is defined), can be customized (if several colours are defined),
<code>label</code>	string, labels for each node. By default are the column names of the distance matrix ( <code>dis</code> ). (See <code>substr</code> to automatically reduce name lengths).
<code>modules</code>	a logical, must nodes belonging to different modules be represented as different colours?

## Examples

```
exampleAlign1<-"example1.fas"

distGap<-MCIC(readfile=T,input=exampleAlign1,saveFile=F,onlyUniques==F)

align<-read.dna(exampleAlign1,format="fasta")

dist.nt<-dist.dna(align,model="raw",pairwise.deletion=T)

DISTnt<-as.matrix(dist.nt)

CombinedDistance<-nt.gap.comb(DISTgap=distGap, range=0.5, method="Corrected",
saveFile=FALSE, DISTnuc=DISTnt)

perc.thr(dis=as.data.frame(CombinedDistance$Corrected),label=paste(substr(row.names(as.data.frame(CombinedDistance$Corrected)),4,4),substr(row.names(as.data.frame(CombinedDistance$Corrected)),8,8),sep="-"))
```

## References

Rozenfeld AF, Arnaud-Haond S, Hernández-García E, Eguíluz VM, Serrão EA, Duarte CM. (2008). Network analysis identifies weak and strong links in a metapopulation system. *Proceedings of the National Academy of Sciences*, **105**, 18824 –18829.

---

`plot.network` Plot a network given a threshold or a range of thresholds.

---

### Description

Given a distance matrix, this function plots a network taking into account only distances shorter than a defined value (threshold). Multiple networks are estimated if a list of thresholds are provided.

The defined threshold must range between 0 and 1 and represents the percentage of the maximum value of the distance matrix. Consequently, if the threshold value is set to 0.5, only distances lower than half the maximum will be represented.

### Usage

```
plot.network(dis, threshold, bgcol = "white", label.col = "black", label = colnames(dis),
             modules = FALSE, PDF = TRUE, PDFname = paste("Network_thr = ", threshold,
             sep = ""))
```

### Arguments

<code>dis</code>	the distance matrix to be represented
<code>threshold</code>	a numeric or a vector between 0 and 1, is the threshold or range of thresholds to be plotted.
<code>PDF</code>	a logical, must the network be saved as pdf?
<code>PDFname</code>	if <code>PDF=TRUE</code> , the name to save the percolation network as pdf.
<code>bgcol</code>	string, defining the colour of the background for each node in the network. Can be equal for all nodes (if only one colour is defined), can customized (if several colours are defined), or can represent different modules (see <code>modules</code> option).
<code>label.col</code>	string, defining the colour of labels for each node in the network. Can be equal for all nodes (if only one colour is defined), can customized (if several colours are defined),
<code>label</code>	string, labels for each node. By default are the column names of the distance matrix ( <code>dis</code> ). (See <code>substr</code> to automatically reduce name lengths).
<code>modules</code>	a logical, must nodes belonging to different modules be represented as different colours?

## Examples

```
exampleAlign1<-"example1.fas"
```

```
matrixMCIC<-MCIC (readfile = T, input=exampleAlign1, saveFile = F)
```

---

pop.dist      Distances among populations

---

## Description

This function computes the among population distance matrix based on the frequency of haplotypes per population and the among haplotypes distance matrix.

## Usage

```
pop.dist (DistFile=T, inputDist=NA, distances=NA, HaploFile=T, inputHaplo=NA,
Haplos=NA, outType=O, logfile=TRUE, saveFile=TRUE)
```

## Arguments

DistFile	a logical; if TRUE (default) input distance matrix among haplotypes is provided as a matrix in a text file. If FALSE, the matrix must be provided as an R object.
inputDist	the name of the file containing the distance matrix among haplotypes to be analysed.
distances	the name of the distance matrix among haplotypes to be analysed (if “DistFile” is set to FALSE,).
HaploFile	a logical; if TRUE (default) the input matrix containing the number of haplotypes found per population is provided as a matrix in a text file. If FALSE, the matrix must be provided as an R object. See HapPerPop for details on how to estimate such matrix.
inputHaplo	the name of the file containing the matrix with the number of haplotypes found per population.
Haplos	the name of the matrix containing the number of haplotypes found per population (if “DistFile” is set to FALSE,).
outType	a string; the format of output matrix. “L” for lower diagonal hemi-matrix; “7” for upper diagonal hemi-matrix; "O" for both hemi-matrices (default).
logfile	a logical; if TRUE (default), it saves a file containing matrix names used (inputDist and HaploFile)
saveFile	a logical; if TRUE (default), function output is saved as a text file.

**Value**

A matrix containing the genetic distances among populations, based on the haplotype distances and their frequencies per populations.

**Examples**

```
alin<-read.dna(file=exampleAlign1,format="fasta")
```

```
Weighted<-
```

```
as.data.frame(HapPerPop(readfile=T,inputFile="FindHaplo_custom.txt",header=T,saveFile=F)[1])
```

```
CombinedDistance<-as.data.frame(nt.gap.comb(DISTgap=distGap, range=0.5,  
method="Corrected", saveFile=FALSE, DISTnuc=DISTnt)[2])
```

```
pop.dist (DistFile=F, distances=CombinedDistance, HaploFile=F, Haplos=Weighted, outType=O,  
logfile=F, saveFile=F)
```

**Index**

FindHaplo, [2](#)

GetHaplo, [3](#)

HapPerPop, [4](#)

MCIC, [6](#)

nt.gap.comb, [7](#)

perc.thr, [8](#)

plot.network, [10](#)

pop.dist, [11](#)

sidier-package, [2](#)