

# Package ‘GenVectors’

June 11, 2019

**Type** Package

**Title** GenVectors an Integrative Analytical Tool for Phylogeography

**Version** 0.0.1

**Date** 2019-06-11

**Author** Leandro da Silva Duarte, Jacqueline de Souza Lima, Renan Maestri, Vanderlei Julio Debastiani and Rosane Garcia Collevatti

**Maintainer** Vanderlei Julio Debastiani <[vanderleidebastiani@yahoo.com.br](mailto:vanderleidebastiani@yahoo.com.br)>

**Description** Function to extract eigenvectors and perform null model-based tests (available at <<https://github.com/vanderleidebastiani/GenVectors>>).

**License** GPL (>= 2)

**Encoding** UTF-8

**Depends** PCPS (>= 1.0.7)

**Imports** ape, pegas, adegenet

**RoxygenNote** 6.1.1

**Collate** 'SNPVectors.R' 'HaploVectors.R' 'GenVectors.R' 'HaploDist.R'  
'SNPDist.R' 'print.GenVectors.R'

**NeedsCompilation** no

## R topics documented:

|                      |          |
|----------------------|----------|
| HaploDist . . . . .  | 1        |
| segv . . . . .       | 2        |
| SNPDist . . . . .    | 3        |
| SNPVectors . . . . . | 3        |
| <b>Index</b>         | <b>7</b> |

---

|           |                  |
|-----------|------------------|
| HaploDist | <i>HaploDist</i> |
|-----------|------------------|

---

## Description

Function to extract haplotypes and compute pairwise distances between haplotypes. This function use the functions [haplotype](#) of package pagas and [dist.dna](#) of package ape.

**Usage**

```
HaploDist(x, dist.model = "N", ...)
```

**Arguments**

|                         |  |
|-------------------------|--|
| <code>x</code>          | A set of DNA sequences (as an object of class "DNABin" or "haplotype") as used by the function <a href="#">haplotype</a> .   |
| <code>dist.model</code> | A character string used by the function <a href="#">dist.dna</a> to specify the evolutionary model to be used to compute pairwise distances from DNA sequences (default <code>dist.model = "N"</code> ). |
| <code>...</code>        | Additional arguments to the function <a href="#">dist.dna</a> .  |

**Value**

A list with:

|                                       |   |
|---------------------------------------|---|
| <code>call</code>                     | Arguments used.   |
| <code>haplotypes</code>               | A list with haplotypes indices that identify each observation sharing the same haplotype. |
| <code>individual.per.haplotype</code> | A matrix with individuals per haplotype.  |
| <code>haplotype.distances</code>      | A matrix with pairwise distances between haplotypes.                                      |

**See Also**

[HaploVectors](#)

**Examples**

```
data(segv)
HaploDist(segv$segv.fas)
```

---

segv

*segv - Simple Example for GenVectors*

---

**Description**

Hypothetical data for running examples.

**Usage**

```
data(segv)
```

**Format**

A list with:

**segv.fas** An object of class DNABin with 18 DNA sequences.  
**segv.pi** A matrix describing the incidence of each individual (columns) in a given locality (rows).  
**segv.envir** A matrix with one environmental variable describing each locality.

**Examples**

```
data(segv)
segv
```

---

SNPDist

*SNPDist*


---

**Description**

Function to computes distances between alleles. This function use the function [propShared](#) of package adegenet.

**Usage**

```
SNPDist(x)
```

**Arguments**

**x** A set of individual genotypes (an object of class [genind](#)) as used by the function [propShared](#).

**Value**

A list with:

**call** Arguments used.  
**SNP.distances** A matrix of distances between alleles.

**See Also**

[SNPVectors](#)

---

SNPVectors

*HaploVectors, SNPVectors and GenVectors*


---

**Description**

Function to extract haplotypic/SNP/genetic eigenvectors and perform null model-based tests.

**Usage**

```
SNPVectors(x, pop, checkdata = TRUE, method = "euclidean",
  squareroot.dis = TRUE, choices = c(1, 2), analysis = "none", envir,
  formula, runs = 999, ...)
```

```
HaploVectors(x, pop, dist.model = "N", checkdata = TRUE,
  log.frequencies = TRUE, method = "euclidean",
  squareroot.dis = TRUE, choices = c(1, 2), analysis = "none", envir,
  formula, runs = 999, ...)
```

```
GenVectors(pop, distances, checkdata = TRUE, method = "euclidean",
  squareroot.dis = TRUE, choices = c(1, 2), analysis = "none", envir,
  formula, runs = 999, ...)
```

```
## S3 method for class 'GenVectors'
print(x, ...)
```

## Arguments

|                 |   |
|-----------------|---|
| x               | A set of DNA sequences (class "DNABin" or "haplotype") as used by the function <a href="#">haplotype</a> or a set of individual genotypes (an object of class <a href="#">genind</a> ) as used by the function <a href="#">propShared</a> . |
| pop             | A matrix describing the incidence of each individual (columns) in a given locality (rows).  |
| checkdata       | Logical argument (TRUE or FALSE) to check if individual sequences in the pop data follow the same order as in the set of DNA sequences (Default checkdata = TRUE).  |
| method          | Dissimilarity index to apply in matrix P, which describes localities by their haplotypic/SNP/genetic composition, as accepted by <code>vegdist</code> function in <code>vegan</code> package (Default method = "euclidean").                |
| squareroot.dis  | Logical argument (TRUE or FALSE) to specify if use square root of dissimilarity index in matrix P (Default squareroot.dis = TRUE).  |
| choices         | Axes for re-scaling and to used in analysis. Choices must have length equal to two (Default choices = c(1, 2)).   |
| analysis        | Type of analysis, partial match to "none", "adonis" or "glm" (Default analysis = "none").   |
| envir           | A matrix with environmental variables for each population, with variables as columns and localities as rows. See Details and Examples.  |
| formula         | An object of class <a href="#">formula</a> . Used in "adonis" or "glm" analysis. See Details and Examples.  |
| runs            | Number of permutations for assessing probability of type I error.   |
| ...             | Additional arguments to function <a href="#">matrix.p.sig</a> and <a href="#">pcps.sig</a> .  |
| dist.model      | A character string used by the function <a href="#">dist.dna</a> to specify the evolutionary model to be used to computes pairwise distances from DNA sequences (default dist.model = "N").   |
| log.frequencies | Logical argument (TRUE or FALSE) to specify if transformation of natural logarithms plus one in haplotype per locality data must be applied (Default log.frequencies = TRUE).   |
| distances       | Matrix containing genetic distances between individuals.  |

## Details

HaploVectors and SNPVectors are two complementary functions to extract haplotypic/SNP eigenvectors and perform null model-based tests.

HaploVectors function is based in set of DNA sequences (class "DNABin" or "haplotype") as used by the function [haplotype](#) and pairwise distances from DNA sequences as used in [dist.dna](#). SNPVectors function is based in set of individual genotypes (class "genind") as pairwise distances between alleles based in [propShared](#) function.

The function `HaploVectors` extract haplotypic eigenvectors and perform null model tests. The argument *analysis* specify the type of analysis performed. When *analysis* is equal "adonis" the analysis is performed in matrix of haplotypic composition (using `matrix.p.sig` function). The argument *formula* must be specified, where the left hand side gives the resemblance data, right hand side gives the variables. The resemblance data is internally named *p.dist*, thus formula is an expression of the form *p.dist ~ predictors*. If *analysis* is equal "glm" it is performed with `haplovector` (using `pcps.sig` function). In this case, the argument *formula* must be specified, where the left hand side gives the vectors used, right hand side gives the variables. The vectors are internally named sequentially *haplovector.1*, *haplovector.2*, *haplovector.3* and so on. Thus, formula is an expression of the form *haplovector.1 ~ predictors*.

The function `SNPVectors` work same way, however extract genetic eigenvectors based in distances between alleles. Similarly the argument *analysis* specify the type of analysis performed. When *analysis* is equal "adonis" the analysis is performed in matrix of genetic composition and the argument *formula* must be specified in the same way the `HaploVectors` function. If *analysis* is equal "glm" it is performed with `SNPvector` and the argument *formula* must be specified. This case the vectors are internally named sequentially *SNPvector.1*, *SNPvector.2*, *SNPvector.3* and so on. Thus, formula is an expression of the form *SNPvector.1 ~ predictors*.

A third function, called a `GenVectors`, is also available. In this case, the matrices of distances between individuals can be supplied directly. This function work same way that other function, but the vectors are internally named sequentially *geneticvector.1*, *geneticvector.2*, *geneticvector.3* and so on.

## Value

A list with:

|                                       |   |
|---------------------------------------|---|
| <code>call</code>                     | Arguments used.   |
| <code>haplotypes</code>               | A list with haplotypes index that identify each observation that share the same haplotype.  |
| <code>haplotype.distances</code>      | A matrix with pairwise distances between haplotypes.  |
| <code>SNP.distances</code>            | A matrix with pairwise distances between alleles.   |
| <code>genetic.distances</code>        | A matrix with pairwise genetic distances.   |
| <code>individual.per.haplotype</code> | A matrix with individuals per haplotype.  |
| <code>haplotype.per.locality</code>   | A matrix with frequency of each haplotype per locality ( <i>W</i> ).  |
| <code>vectors</code>                  | Haplotypic/SNP/genetic eigenvectors ( <code>haplovector</code> s, <code>snpvectors</code> or <code>geneticvec</code> - <code>tors</code> ). |
| <code>values</code>                   | Eigenvalues, relative eigenvalues and cumulative relative eigenvalues.  |
| <code>correlations</code>             | Correlations between haplotypic/SNP/genetic eigenvectors and haplotypes/alleles.  |
| <code>P</code>                        | Matrix of haplotypic/SNP/genetic composition ( <i>P</i> ).  |
| <code>scores</code>                   | Scores for biplots.   |
| <code>model</code>                    | The observed model.   |
| <code>fun</code>                      | The funtion used.   |
| <code>statistic.null.turnover</code>  | A matrix with null statistic for turnover null model.   |

|  |   |
|--|---|
| <code>statistic.null.divergence</code> | A matrix with null statistic for divergence null model. |
| <code>statistic.obs</code>             | Observed statistic, F value to predefined function.     |
| <code>p.turnover</code>                | The p value for the turnover null model.                |
| <code>p.divergence</code>              | The p value for the divergence null model.              |

**See Also**

[HaploDist](#), [SNPDist](#), [matrix.p.sig](#), [pcps.sig](#)

**Examples**

```
data(segv)

HaploVectors(segv$segv.fas, segv$segv.pi, envir = segv$segv.envir)

HaploVectors(segv$segv.fas, segv$segv.pi, analysis = "adonis",
             envir = segv$segv.envir, formula = p.dist~R, runs = 99)

HaploVectors(segv$segv.fas, segv$segv.pi, analysis = "glm",
             envir = segv$segv.envir, formula = haplovector.1~R, runs = 99)
```

# Index

\*Topic **datasets**

segv, [2](#)

dist.dna, [1](#), [2](#), [4](#)

formula, [4](#)

genind, [3](#), [4](#)

GenVectors (SNPVectors), [3](#)

HaploDist, [1](#), [6](#)

haplotype, [1](#), [2](#), [4](#)

HaploVectors, [2](#)

HaploVectors (SNPVectors), [3](#)

matrix.p.sig, [4–6](#)

pcps.sig, [4–6](#)

print.GenVectors (SNPVectors), [3](#)

propShared, [3](#), [4](#)

segv, [2](#)

SNPDist, [3](#), [6](#)

SNPVectors, [3](#), [3](#)