

Package ‘APIS’

October 12, 2022

Type Package

Title Auto-Adaptive Parentage Inference Software Tolerant to Missing Parents

Version 1.0.1

Author Ronan Griot, François Allal, Romain Morvezen, Sophie Brard-Fudulea, Pierrick Haffray, Florence Phocas and Marc Vandeputte

Maintainer Ronan Griot <ronan.griot@gmail.com>

Description Parentage assignment package.

Parentage assignment is performed based on observed average Mendelian transmission probability distributions.

The main function of this package is the function APIS(), which is the parentage assignment function.

License GPL

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Depends R (>= 3.4.0)

Imports foreach, parallel, doParallel, ggplot2, gridExtra, methods

NeedsCompilation yes

Suggests knitr, rmarkdown

VignetteBuilder knitr

Repository CRAN

Date/Publication 2020-12-02 15:40:03 UTC

R topics documented:

allFreq	2
APIS	2
APIS_dam	3
APIS_offspring	4

APIS_sire	4
assignmentFortran	5
assignmentPower	6
personalThreshold	6
selectParents	7
setThreshold	8

Index	9
--------------	----------

allFreq	<i>Estimate the allele frequencies</i>
---------	--

Description

This function estimates allele frequencies

Usage

```
allFreq(genotype)
```

Arguments

genotype	A matrix of genotypes (n*p) n = number of individuals p = number of markers (coded as "All1/All2", ex: "A/A" or "NA/NA" for missing genotype)
----------	---

Value

allele frequencies

Examples

```
data("APIS_offspring")
freq <- allFreq(APIS_offspring)
```

APIS	<i>APIS function that assigns with observed data</i>
------	--

Description

This function performs the APIS procedure

Usage

```
APIS(off.genotype, sire.genotype, dam.genotype, error = 0,
     exclusion.threshold = ncol(off.genotype), preselect.Parent = FALSE,
     nb.cores = 2, verbose = TRUE)
```

Arguments

off.genotype	Offspring genotypes Matrix (n*p) where n = number of individuals p = number of markers rownames(offspring) = labels of offspring marker coding = "A11/A112" example: "A/A", "A/B", "NA/NA" (for missing genotype)
sire.genotype	Sire genotypes Matrix (n*p) where n = number of individuals p = number of markers rownames(sire) = labels of sires marker coding = "A11/A112" example: "A/A", "A/B", "NA/NA" (for missing genotype)
dam.genotype	Dam genotypes Matrix (n*p) where n = number of individuals p = number of markers rownames(dam) = labels of dams marker coding = "A11/A112" example: "A/A", "A/B", "NA/NA" (for missing genotype)
error	(default: 0) The assignment error rate accepted by the user
exclusion.threshold	(default: ncol(off.genotype)) Threshold for exclusion (number of mismatches allowed)
preselect.Parent	(default: FALSE) Preselection of parents. Can be FALSE, an integer or a vector of two integers (number of sires, numbers of dams)
nb.cores	(default: 2) Number of cores to use. If you have more than 2 cores, you can use the "parallel" function detectCores()
verbose	(default : TRUE) Display the process of the function on the console.

Value

pedigree
a log file

Examples

```
data("APIS_offspring")
data("APIS_sire")
data("APIS_dam")

result <- APIS(off.genotype = APIS_offspring[1:50, ],
              sire.genotype = APIS_sire,
              dam.genotype = APIS_dam,
              error = 0.05,
              verbose = FALSE)
```

APIS_dam

Example dam genotypes

Description

Example dam genotypes

Usage

APIS_dam

Format

A matrix with 14 rows (one row = one dam) and 100 columns (one column = one marker)

APIS_offspring	<i>Example offspring genotypes</i>
----------------	------------------------------------

Description

Example offspring genotypes

Usage

APIS_offspring

Format

A matrix with 1068 rows (one row = one offspring) and 100 columns (one column = one marker)

APIS_sire	<i>Example sire genotypes</i>
-----------	-------------------------------

Description

Example sire genotypes

Usage

APIS_sire

Format

A matrix with 39 rows (one row = one sire) and 100 columns (one column = one marker)

assignmentFortran	<i>Assignment function to obtain the average Mendelian transmission probabilities using a Fortran library</i>
-------------------	---

Description

This function calculates the average Mendelian transmission probabilities

Usage

```
assignmentFortran(offspring, sire, dam, thresh = ncol(offspring),
  preselect.Parent = FALSE, nb.cores = 2, verbose = TRUE)
```

Arguments

offspring	Offspring genotypes Matrix (n*p) where n = number of individuals, p = number of markers rownames(offspring) = labels of offspring marker coding = "A111/A112" example: "A/A", "A/B", "NA/NA" (for missing genotype)
sire	Sire genotypes Matrix (n*p) where n = number of individuals, p = number of markers rownames(sire) = labels of sires marker coding = "A111/A112" example: "A/A", "A/B", "NA/NA" (for missing genotype)
dam	Dam genotypes Matrix (n*p) where n = number of individuals, p = number of markers rownames(dam) = labels of dams marker coding = "A111/A112" example: "A/A", "A/B", "NA/NA" (for missing genotype)
thresh	(default: ncol(offspring)) Threshold for exclusion (number of mismatches allowed)
preselect.Parent	(default: FALSE) Preselection of parents. Can be FALSE, an integer or a vector of two integers (number of sires, numbers of dams)
nb.cores	(default: 2) Number of cores to use. If you have more than 2 cores, you can use the "parallel" function detectCores()
verbose	(default : TRUE) Display the process of the function on the console.

Value

intermediate pedigree
 log file for Mendelian transmission probabilities
 log file for exclusion

Examples

```
data("APIS_offspring")
data("APIS_sire")
data("APIS_dam")

assignment <- assignmentFortran(APIS_offspring[1:50, ], APIS_sire, APIS_dam, verbose = FALSE)
```

assignmentPower *calculate the theoretical assignment power*

Description

This function calculates the theoretical assignment power of the marker set

Usage

```
assignmentPower(sire, dam)
```

Arguments

sire Sire genotypes | Matrix (n*p) where n = number of individuals, p = number of markers rownames(sire) = labels of sires marker coding = "All1/All2" example: "A/A", "A/B", "NA/NA" (for missing genotype)

dam Dam genotypes | Matrix (n*p) where n = number of individuals, p = number of markers rownames(dam) = labels of dams marker coding = "All1/All2" example: "A/A", "A/B", "NA/NA" (for missing genotype)

Value

Theoretical assignment power of the marker set

Examples

```
data("APIS_sire")
data("APIS_dam")
assignmentPower(APIS_sire, APIS_dam)
```

personalThreshold *Establish personal threshold*

Description

This function allows the user to set up his own threshold

Usage

```
personalThreshold(APIS.result, method, threshold = NULL,
  verbose = TRUE)
```

Arguments

APIS.result	APIS function output
method	the method for the new threshold 'delta' for deltas, 'Pmendel' for Mendelian probabilities, 'exclusion' for mismatches
threshold	personal threshold default values are implemented
verbose	(default : TRUE) Display the process of the function on the console.

Value

new pedigree from the new threshold

Examples

```
data("APIS_offspring")
data("APIS_sire")
data("APIS_dam")

result <- APIS(off.genotype = APIS_offspring[1:50, ],
              sire.genotype = APIS_sire,
              dam.genotype = APIS_dam,
              error = 0.05,
              verbose = FALSE)

new.result <- personalThreshold(result, method = 'exclusion', threshold = 2, verbose = FALSE)
```

selectParents	<i>Select most likely parents for potent parent pairs tests</i>
---------------	---

Description

This function allows the selection of the most likely parents for assignment, reducing computation time

Usage

```
selectParents(off.genotype, parent.genotype, parent.sex, n.Parent)
```

Arguments

off.genotype	genotype of one offspring
parent.genotype	genotype matrix of parent genotypes
parent.sex	vector of parents sex
n.Parent	vector of number of sires and dams to select

Value

list of potential sires and dams

setThreshold	<i>Set the APIS threshold</i>
--------------	-------------------------------

Description

This function calculates the threshold for APIS

Usage

```
setThreshold(ped.log, ped.exclu, nb.mrk, error = NULL, verbose = TRUE)
```

Arguments

ped.log	log.like from assignment function
ped.exclu	log.exclu from assignment function
nb.mrk	Number of markers
error	(default: NULL) The assignment error rate accepted by the user
verbose	(default : TRUE) Display the process of the function on the console.

Value

pedigree
log file

Index

* APIS

APIS, 2
personalThreshold, 6
selectParents, 7

* allele

allFreq, 2

* assignment

APIS, 2
assignmentFortran, 5
assignmentPower, 6
personalThreshold, 6
selectParents, 7
setThreshold, 8

* datasets

APIS_dam, 3
APIS_offspring, 4
APIS_sire, 4

* exclusion

assignmentPower, 6

* frequencies

allFreq, 2

* power

assignmentPower, 6

* threshold

personalThreshold, 6
selectParents, 7

allFreq, 2

APIS, 2

APIS_dam, 3

APIS_offspring, 4

APIS_sire, 4

assignmentFortran, 5

assignmentPower, 6

personalThreshold, 6

selectParents, 7

setThreshold, 8