Package ‘GenoTriplo’

October 13, 2026

Type Package

Title Genotyping Triploids/Tetraploids (or Diploids) from Luminescence Data

Version 2.2.2

Description Genotyping of triploid individuals from luminescence data (marker probe-set A and B). Works also for diploids and tetraploids.
Three main functions: Create_Dataset_from_file() to build dataset ; Clustering_parallele_from_dir() that regroups individuals with a same genotype based on proximity and Genotyping_parallele_from_dir() that assigns a genotype to each cluster. For Shiny interface use: launch_GenoShiny().

License GPL

Encoding UTF-8

Imports cowplot, doParallel, dplyr, DT, foreach, ggplot2, htmltools, parallel, processx, rlang, Rmixmod, shiny, shinythemes, data.table, shinyFiles, bslib, stringr

Depends R (>= 3.5.0), shinyBS

NeedsCompilation no

Author Julien Roche [aut, cre],
Florence Phocas [aut],
Mathieu Besson [aut],
Pierre Patrice [aut],
Marc Vandeputte [aut],
François Allal [aut],
Pierrick Haffray [aut]

Maintainer Julien Roche <julien.roche@inrae.fr>

Config/roxygen2/version 8.0.0

Repository CRAN

Date/Publication 2026-10-13 10:10:07 UTC

Contents

Clustering 2

Clustering_from_dir 3
Clustering_from_file 4
Clustering_parallele_from_dir 5
Create_Dataset_from_file 6
create_genofiles 7
Genotyping_from_file 7
Genotyping_parallele_from_dir 9
launch_GenoShiny 10

Index **11**

`Clustering` *Clustering function*

Description

Clustering function to run clustering with no parallelization process nor auto save

Usage

```
Clustering(data_clustering, nb_clust_possible, n_iter = 5, Dmin = 0.28)
```

Arguments

`data_clustering`
: dataset with Contrast and SigStren for each individuals (as SampleName) and each markers (as MarkerName)

`nb_clust_possible`
: number of cluster possible (ploidy+1)

`n_iter` : number of iterations to perform for clustering

`Dmin` : minimal distance between two clusters

Value

list of results of clustering

Examples

```
## Not run:
file.create = system.file("extdata/output_create/1","1_1_to_clust.Rdata", package = "GenoTriplo")
load(file.create)
ploidy=3
Clustering(data_clustering=data_clustering,nb_clust_possible=ploidy+1)

## End(Not run)
```

Clustering_from_dir *Clustering function from directory*

Description

Clustering function to run clustering with no parallelization process and autosave

Usage

```
Clustering_from_dir(
  dir_from_create,
  new.dir = NULL,
  ploidy,
  n_iter = 5,
  Dmin = 0.28
)
```

Arguments

dir_from_create	directory of the file to clusterize.
new.dir	new directory name for saving in './output_clustering'
ploidy	ploidy of individuals
n_iter	number of iterations to perform for clustering
Dmin	minimal distance between two clusters

Value

Autosave of the results in new.dir

Examples

```
## Not run:
dir.create = system.file("extdata/output_create/1", package = "GenoTriplo")
ploidy = 3
Clustering_from_dir(dir_from_create=dir.create,ploidy=ploidy)

## End(Not run)
```

Clustering_from_file *Clustering function from file*

Description

Clustering function to run clustering with no parallelization process and autosave Used by Clustering_from_dir. Shouldn't be used else.

Usage

```
Clustering_from_file(
  dirname,
  filename,
  ploidy,
  n_iter = 5,
  Dmin = 0.28,
  new.dir = NULL
)
```

Arguments

dirname	directory of the file to clusterize.
filename	filename of the data to clusterize
ploidy	ploidy of individuals
n_iter	number of iterations to perform for clustering
Dmin	minimal distance between two clusters
new.dir	new directory name for saving

Value

Autosave of the results

Examples

```
## Not run:
dir.create = system.file("extdata/output_create/1", package = "GenoTriplo")
file.create = system.file("extdata/output_create/1","1_1_to_clust.Rdata", package = "GenoTriplo")
ploidy=3
Clustering_from_file(dirname=dir.create,filename=file.create,ploidy=ploidy)

## End(Not run)
```

Clustering_parallele_from_dir

Launch parallel clustering from directory

Description

Launch the clustering phase in parallel from the dataset with SampleName, Contrast and SigStren for each markers (MarkerName).

Usage

```
Clustering_parallele_from_dir(
  dir_from_create,
  new.dir = NULL,
  ploidy,
  n_iter = 5,
  Dmin = 0.28,
  n_core = 1
)
```

Arguments

dir_from_create	directory name with dataset files saved in './output_create'
new.dir	directory name where output is saved as in './output_clustering'
ploidy	ploidy of individuals
n_iter	number of iterations of clustering
Dmin	threshold distance between two clusters
n_core	number of cores used for parallelization

Value

the result of clustering or automatically save a list of objects if a saving name has been provided

Examples

```
## Not run:
dir.create = system.file("extdata/output_create/1", package = "GenoTriplo")
Clustering_parallele_from_dir(dir_from_create=dir.create,ploidy=3)

## End(Not run)
```

Create_Dataset_from_file

Create dataset from filename in appropriate format

Description

Create SigStren and Contrast variables from luminescence values of probeset A and B of each markers and return a dataframe to be used for clustering or save the result if a saving name is given

Usage

```
Create_Dataset_from_file(
  filename = "./AxiomGT1.summary.txt",
  dir.name = NULL,
  indiv_name = NULL,
  marker_name = NULL,
  n_marker_chunk = 4000,
  simplify.names = TRUE
)
```

Arguments

`filename`	filename from AXAS : `AxiomGT1.summary.txt`
`dir.name`	saving name (the number of the chunk and _to_clust is automatically added)
`indiv_name`	vector with name of individuals to keep
`marker_name`	vector with name of marker to keep
`n_marker_chunk`	number of lines by chunk (have to be even as markers goes by pair)
`simplify.names`	Samplenames might be like so : IndivXX_a1_a1.CEL (if from Axiom) -> allows to select only IndivXX as SampleName

Value

save file in `./output_create/dir.name` as `k_to_clust.Rdata` with k the chunk number.

Examples

```
## Not run:
filepath = system.file("extdata", "AxiomGT1.summary_examples.txt", package = "GenoTriplo")
Create_Dataset_from_file(filename=filepath)

## End(Not run)
```

create_genofiles — *Create files of genotype and indiv/marker stats from genotype .Rdata files*

Description

Create files of genotype and indiv/marker stats from genotype .Rdata files

Usage

```
create_genofiles(dir.name.save, new.dir, ploidy, corres_ATCG = NULL)
```

Arguments

dir.name.save	directory with .Rdata files with genotype
new.dir	new directory for saving files
ploidy	ploidy of the population
corres_ATCG	dataframe with correspondance between Allele_A/Allele_B to ATCG

Value

names of the indivCR and markerCR files

Examples

```
## Not run:
ploidy = 3
dir.name.save = system.file("extdata/output_genotyping/1", package = "GenoTriplo")
new.dir = "1"
create_genofiles(dir.name.save=dir.name.save,new.dir=new.dir,ploidy=ploidy)

## End(Not run)
```

Genotyping_from_file — *Run genotyping from file*

Description

Run genotyping from file

Usage

```
Genotyping_from_file(
  dirname,
  filename,
  new.dir = NULL,
  ploidy,
  SeuilNoCall = 0,
  SeuilNbSD = 2.8,
  SeuilSD = 0.28,
  cr_marker = 0.97,
  fld_marker = 3.4,
  hetso_marker = -0.3,
  same.pop = TRUE,
  tryEqHW = TRUE
)
```

Arguments

`dirname`	name of the directory where the file is
`filename`	name of the file with the data for genotyping
`new.dir`	new directory name for saving
`ploidy`	ploidy of individuals
`SeuilNoCall`	threshold of the probability of belonging to a cluster
`SeuilNbSD`	threshold for the distance between an individuals and his cluster (x=Contrast)
`SeuilSD`	threshold for the standard deviation of a cluster (`SeuilSD*(1+0.5*abs(mean_contrast_cluster))`)
`cr_marker`	call rate threshold
`fld_marker`	FLD threshold
`hetso_marker`	HetSO threshold
`same.pop`	Boolean : are individuals from a same population
`tryEqHW`	Only if same.pop==TRUE else ignore. If TRUE, will delete cluster that shouldn't happen under H-W equilibrium (heterozygote with less individuals than both homozygote in triploid for example). Additionnaly, the cluster should contain less than 5% of the individuals. Else it stays.

Value

Void. Save the results

Examples

```
## Not run:
ploidy=3
dir_from_clustering=system.file("extdata/output_clustering/1", package = "GenoTriplo")
file_from_clustering=system.file(
                      "extdata/output_clustering/1",
                      "1_1_to_geno.Rdata", package = "GenoTriplo")
```

```
Genotyping_from_file(dirname=dir_from_clustering,ploidy=ploidy,filename=file_from_clustering)

## End(Not run)
```

Genotyping_parallele_from_dir

Launch genotyping phase in parallel from directory

Description

Function that launch the genotyping phase from the dataset with SampleName, Contrast and SigStren for each markers and the result of the 'Run_clustering' function.

Usage

```
Genotyping_parallele_from_dir(
  dir_from_clustering,
  new.dir = NULL,
  ploidy,
  SeuilNoCall = 0,
  SeuilNbSD = 2.8,
  SeuilSD = 0.28,
  n_core = 1,
  corres_ATCG = NULL,
  same.pop = TRUE,
  cr_marker = 0.97,
  fld_marker = 3.4,
  hetso_marker = -0.3,
  tryEqHW = TRUE
)
```

Arguments

dir_from_clustering	directory with files from clustering phase
new.dir	name of the new directory for saving results in './output_genotyping'
ploidy	ploidy of individuals
SeuilNoCall	threshold of the probability of belonging to a cluster
SeuilNbSD	threshold for the distance between an individuals and his cluster (x=Contrast)
SeuilSD	threshold for the standard deviation of a cluster (SeuilSD*(1+0.5*abs(mean_contrast_cluster)))
n_core	number of cores used for parallelization
corres_ATCG	dataframe with the correspondence between A/B of AXAS and A/T/C/G (three columns : probeset_id, Allele_A, Allele_B)
same.pop	Boolean : are individuals from a same population
cr_marker	call rate threshold

`fld_marker`	FLD threshold
`hetso_marker`	HetSO threshold
`tryEqHW`	Only if same.pop==TRUE else ignore. If TRUE, will delete cluster that shouldn't happen under H-W equilibrium (heterozygote with less individuals than both homozygote in triploid for example). Additionnaly, the cluster should contain less than 5% of the individuals. Else it stays.

Value

Void. Save the results.

Examples

```
## Not run:
ploidy=3
dir_from_clustering=system.file("extdata/output_clustering/1", package = "GenoTriplo")
Genotyping_parallele_from_dir(dir_from_clustering=dir_from_clustering,ploidy=ploidy)

## End(Not run)
```

---

launch_GenoShiny *Shiny App for genotyping*

Description

Launch a shiny interface to use GenoTriplo. Really easy to use and user friendly, this will help you gain time !

Usage

```
launch_GenoShiny()
```

Value

void : most results are automatically saved

Examples

```
## Not run:
launch_GenoShiny()

## End(Not run)
```

Index

Clustering, 2
Clustering_from_dir, 3
Clustering_from_file, 4
Clustering_parallele_from_dir, 5
Create_Dataset_from_file, 6
create_genofiles, 7

Genotyping_from_file, 7
Genotyping_parallele_from_dir, 9

launch_GenoShiny, 10