Package 'RTIGER'

January 20, 2025

Type Package

Title HMM-Based Model for Genotyping and Cross-Over Identification

Version 2.1.0

Description Our method integrates information from all sequenced samples, thus avoiding loss of alleles due to low coverage. Moreover, it increases the statistical power to uncover sequencing or alignment errors <doi:10.1093/plphys/kiad191>.

Depends R (>= 3.6), GenomicRanges, GenomeInfoDb

License GPL (>= 2)

Encoding UTF-8

LazyData true

LazyDataCompression gzip

Imports methods, e1071, extraDistr, reshape2, ggplot2, TailRank, JuliaCall, IRanges, qpdf, grDevices, graphics, stats, utils

RoxygenNote 7.2.3

VignetteBuilder knitr

Suggests knitr, rmarkdown, markdown, Gviz, rtracklayer

biocViews GenomeAnnotation, HiddenMarkovModel, Sequencing

NeedsCompilation no

Author Rafael Campos-Martin [cre] (<https://orcid.org/0000-0002-1395-8571>),
Sophia Schmickler [aut],
Manish Goel [ctb],
Korbinian Schneeberger [aut],
Achim Tresch [aut]

Maintainer Rafael Campos-Martin <rfael.mpi@gmail.com>

Repository CRAN

Date/Publication 2023-03-29 09:20:02 UTC

Contents

`ATseqlengths` *The autosome chromosome lengths for Arabidopsis Thaliana.*

Description

The autosome chromosome lengths for Arabidopsis Thaliana.

Author(s)

Rafael Campos-Martin

`calcCOnumber` *Obtain number of Cross-Over events per sample and chromosome.*

Description

Obtain number of Cross-Over events per sample and chromosome.

Usage

```
calcCOnumber(object)
```

Arguments

`object`	a RViterbi object.

Value

Matrix m x n. M number of samples and N chromosomes.

#' @return a matrix with n chromosomes and m samples (n x m) and the number of CO events.

Examples

```
data("fittedExample")
co.num = calcCOnumber(myDat)
```

`dev` *Function to developers. It runs one EM step*

Description

Function to developers. It runs one EM step

Usage

```
dev(psi, rigidity = NULL, nstates = 3, transition = NULL, start = NULL)
```

Arguments

`psi`	list of psi probabilities.
`rigidity`	Rigidity value.
`nstates`	Number of states.
`transition`	transition matrix
`start`	initial probabilities

Value

List with updates probabilites

`fit` *Call Julia code to fit the values*

Description

Call Julia code to fit the values

Usage

```
fit(rtigerobj, max.iter , eps,
trace, all = TRUE, random = FALSE,
specific = FALSE, nsamples = 20,
post.processing = TRUE)
```

Arguments

rtigerobj	an RTIGER object.
max.iter	maximum number of iterations to acomplish by the EM.
eps	differnece threshold to halt the EM.
trace	logical value whether to trace the changes in the parameters along the iterations.
all	logical value whether to use all data to fit the model.
random	if all FALSE use random samples.
specific	if all FALSE use specific samples.
nsamples	if random TRUE, how many samples to use.
post.processing	logical value, whether to run post.processing process.

Value

RTIGER object

Examples

```
## Not run:
data("fittedExample")
sourceJulia()
myfit = fit(myDat, max.iter = 2, eps=0.01,
            trace = TRUE, all = TRUE,
            random = FALSE, specific = FALSE,
            nsamples = 20, post.processing = TRUE)


## End(Not run)
```

---

generateObject *Load data*

Description

Load data

Usage

```
generateObject(experimentDesign = NULL,nstates = 3, rigidity=NULL,
seqlengths = NULL, verbose = TRUE)
```

Arguments

experimentDesign	a data Frame that contains minimum a column with the files direction (name of the column files) and another with a shorter name to be used inside the function.
nstates	the number of states to be fitted in the model. A standard setting would use 3 states (Homozygous1, Heterozygous, and Homozygous2).
rigidity	an integer number specifying the rigidity parameter to be used.
seqlengths	a named vector with the chromosome lenghts of the organism that the user is working with.
verbose	logical value. Whether to print info messages.

Value

RTIGER object

Examples

```
data("ATseqlengths")
path = system.file("extdata",  package = "RTIGER")
files = list.files(path, full.names = TRUE)
nam = sapply(list.files(path ), function(x) unlist(strsplit(x, split = "[.]"))[1])
expDesign = data.frame(files = files, name = nam)
names(ATseqlengths) = paste0("Chr", 1:5)
myres = generateObject(experimentDesign = expDesign,
              seqlengths = ATseqlengths,
              rigidity = 10
)
```

---

myDat *A fitted example using three own samples of Arabidopsis. More information in publication:*

Description

A fitted example using three own samples of Arabidopsis. More information in publication:

Author(s)

Rafael Campos-Martin

`optimize_R` *Find the otimum R value for a given data set*

Description

Find the otimum R value for a given data set

Usage

```
optimize_R(object,
max_rigidity = 2^9, average_coverage = NULL, crossovers_per_megabase = NULL,
save_it = FALSE, savedir = NULL)
```

Arguments

`object`	an RTIGER object
`max_rigidity`	R values will be explored up the value given in this parameter. Default = 2^9
`average_coverage`	For conservative results set it to the lowest average coverage of a sample in your experiment, or evne to the lowest average coverage in a (sufficiently large) region in one of your samples. The lower the value, the more conservative (higher) our estimates of the false positive segments rates. If it is not provided it will be computed as the average of all data points.
`crossovers_per_megabase`	For conservative results set it to the highest ratio of a sample in your experiment. The higher the value, the more conservative (higher) our estimates of the false positive segments rates. If it is not provided it will be computed as the average of all samples.
`save_it`	logical values if the results should be saved. Plots might be complicated to interpret. We suggest to read the manuscript to understand them (https://doi.org/10.1093/plphys/kiad191)
`savedir`	if results are saved, in which directory.

Value

A value with the optimum rigidity for the data set.

Examples

```
data("fittedExample")
bestR = optimize_R(myDat)
```

plotCOs *Obtain number of Cross-Over events per sample and chromosome.*

Description

Obtain number of Cross-Over events per sample and chromosome.

Usage

```
plotCOs(object, file = NULL)
```

Arguments

object	a RViterbi object.
file	file where to save the plot for CO numbers

Value

a plot

Examples

```
data("fittedExample")
co.num = calcCOnumber(myDat)
```

RTIGER *Load, Fit, and plot*

Description

Load, Fit, and plot

Usage

```
RTIGER(expDesign, rigidity=NULL, outputdir=NULL, nstates = 3,
seqlengths = NULL, eps=0.01, max.iter=50, autotune = FALSE,
max_rigidity = 2^9, average_coverage = NULL,
crossovers_per_megabase = NULL, trace = FALSE,
tiles = 4e5, all = TRUE, random = FALSE, specific = FALSE,
nsamples = 20, post.processing = TRUE, save.results = TRUE, verbose = TRUE)
```

Arguments

expDesign	a data Frame that contains minimum a column with the files direction (name of the column files) and another with a shorter name to be used inside the function.
rigidity	an integer number specifying the rigidity parameter to be used.
outputdir	a character string that specifies the directory in which to save the results form the function.
nstates	the number of states to be fitted in the model. A standard setting would use 3 states (Homozygous1, Heterozygous, and Homozygous2).
seqlengths	a named vector with the chromosome lenghts of the organism that the user is working with.
eps	the threshold of the difference between the parameters value between the previous and actuay iteration to stope de EM algorithm.
max.iter	maximum number of iterations of the EM algorithm before to stop in case that eps has not been achieved.
autotune	Logical value if the R-value should be tuned by our algorithm. This will take longer as it needs a first training with the rigidity value provided by the user and then the optimization step is carried. Finally, a training using the optimum R will be performed and results for the optimum R will be returned.
max_rigidity	If autotune true, R values will be explored up the value given in this parameter. Default = 2^9
average_coverage	If autotune true, for conservative results set it to the lowest average coverage of a sample in your experiment, or evne to the lowest average coverage in a (sufficiently large) region in one of your samples. The lower the value, the more conservative (higher) our estimates of the false positive segments rates. If it is not provided it will be computed as the average of all data points.
crossovers_per_megabase	If autotune true, for conservative results set it to the highest ratio of a sample in your experiment. The higher the value, the more conservative (higher) our estimates of the false positive segments rates. If it is not provided it will be computed as the average of all samples.
trace	logical value. Whether or not to keep track of the parameters for the HMM along the iterations. Deafault FALSE
tiles	length of the tiles by which the genome will be segmented in order to compute the ratio of COs in the complete dataset.
all	logical value. Whether to use the complete data set to fit the rHMM. default TRUE.
random	Logical value. Choose randomly a subset of the complete dataset to fit the rHMM. Default FALSE
specific	Logical value to specify which samples to take.
nsamples	if random TRUE, how many samples should be taken randomly.
post.processing	Logical value. Whether to run an extra step that fine maps the segment borthers. Default TRUE

save.results	Logical value, whether to generate and save the plots and igv files.
verbose	Logical, whether to print info to console.

Value

Matrix m x n. M number of samples and N chromosomes.

RTIGER object

Examples

```
## Not run:
data("ATseqlengths")
sourceJulia()
path = system.file("extdata",  package = "RTIGER")
files = list.files(path, full.names = TRUE)
nam = sapply(list.files(path ), function(x) unlist(strsplit(x, split = "[.]"))[1])
expDesign = data.frame(files = files, name = nam)
names(ATseqlengths) = paste0("Chr", 1:5)
myres = RTIGER(expDesign = expDesign,
               outputdir = "/home/campos/Documents/outputjjulia/",
               seqlengths = ATseqlengths,
               rigidity = 4,
               max.iter = 2,
               trace = FALSE,
               save.results = TRUE)

## End(Not run)
```

RTIGER-class *This class is a generic container for RTIGER analysis*

Description

This class is a generic container for RTIGER analysis

Slots

matobs Nested lists. the first level is a list of samples. For each sample there are 5 matrices that contains the allele counts for each position.

params a list with the parameters after training.

info List with phenotipic data of the samples.

Viterbi List of chromosomes with the viterbi path per sample.

Probabilities Computed probabilites for the EM algorithm.

num.iter Number of iterations needed to stop the EM algorithm.

`setupJulia` *Installs the needed packages in JULIA to run the EM algorithm for rHMM.*

Description

Installs the needed packages in JULIA to run the EM algorithm for rHMM.

Usage

```
setupJulia(JULIA_HOME = NULL)
```

Arguments

`JULIA_HOME`	the file folder which contains julia binary, if not set, JuliaCall will look at the global option JULIA_HOME, if the global option is not set, JuliaCall will then look at the environmental variable JULIA_HOME, if still not found, JuliaCall will try to use the julia in path.

Value

empty

`sourceJulia` *Function needed before using RTIGER() function. It loads the scripts in Julia that fit the rHMM.*

Description

Function needed before using RTIGER() function. It loads the scripts in Julia that fit the rHMM.

Usage

```
sourceJulia()
```

Value

empty

Index