Package ‘rKOMICS’

July 21, 2021

Title Minicircle Sequence Cluster (MSC) Analyses

Version 1.1

Date 2021-07-21

Description It establishes a critical framework to manipulate, explore and extract biologically relevant information from mitochondrial minicircle assemblies in tens to hundreds of samples simultaneously and efficiently. This should facilitate research that aims to develop new molecular markers for identifying species-specific minicircles, or to study the ancestry of parasites for complementary insights into their evolutionary history.

License GPL

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Imports ggplot2, ape, circlize, ComplexHeatmap, reshape2, utils, stats, dplyr, factoextra, FactoMineR, ggpubr, magrittr, stringr

Suggests viridis

biocViews

Depends R (>= 2.10)

NeedsCompilation no

Author Frederik Van den Broeck [aut],
Manon Geerts [aut, cre]

Maintainer Manon Geerts <mgeerts@itg.be>

Repository CRAN

Date/Publication 2021-07-21 11:40:02 UTC

R topics documented:

`exData` *Example dataset*

Description

A dataset containing example inputs.

Usage

```
exData
```

Format

A list with seven objects

strain a character vector containing the strain names.

subspecies a factor specifying to which subspecie the strains belong to.

HCN a numerical vector with their corresponding median genome wide coverage.

fastafiles a character vector containing the file names of the minicircle sequences in fasta format.

ucs a character vector containing the file names of the cluster information in uc format.

mapstats a character vector containing the files names of the mapping statistics in text format.

depthstats a character vector containing the files names of the depth statistics in text format.

`matrices` *Example cluster matrices*

Description

A list containing 15 example cluster matrices with percent identities of 80, 85 and 88:100.

Usage

```
matrices
```

Format

a list of different cluster matrices.

`msc.depth` *Check the read depth of assembled minicircles*

Description

The depth statistics, generated with KOMICS, include average, median, minimum and maximum per site read depth of every minicircle contig that has been assembled. The msc.depth function allows you to summarize those statistics and to estimate minicircle copy numbers by standardizing median read depths per minicircle contig to the median genome-wide read depths.

Usage

```
msc.depth(depthstats, groups, HCN = NULL)
```

Arguments

`depthstats`	a character vector containing the file names of the depth statistics (output of KOMICS), e.g. sampleA.depthstats.txt, sampleB.depthstats.txt,... .
`groups`	a vector specifying to which groups (e.g. species) the samples belong to.
`HCN`	a numeric vector containing haploid copy numbers of the corresponding samples (optional, by default set to null).

Value

`all`	a table merging depth statistics of all samples. Depth statistics include the average, median, minimum and maximum per site read depth.
`plots`	a plot per sample visualizing the median read depth distribution.
`medianRD`	one graph summarizing the median read depth distribution of all samples.
`CN`	one graph summarizing the copy number (if HCN is not null) of all samples.

Examples

```
require(ggpubr)
data(exData)

### run function

depth <- msc.depth(depthstats = system.file("extdata",
                  exData$depthstats, package = "rKOMICS"), groups = exData$species,
                  HCN = exData$medGWD/2)

### visualize results
hist(depth$all[,"MEDIAN.DEPTH"], breaks=100,
    main="Global median depth distribution",xlab = (''))

### alter plot
annotate_figure(depth$plots$CUM29A1, fig.lab = "CUM29A1",
               fig.lab.pos = "bottom.right", fig.lab.face = 'italic')
```

`msc.heatmap` *Visualization of cluster matrices*

Description

The msc.heatmap function generates a heatmap of the input cluster matrix that summarizes the presence or absence of Minicircle Cluster Sequences (MCSs) between groups of samples.

Usage

```
msc.heatmap(clustmatrix, samples, groups)
```

Arguments

`clustmatrix`	one cluster matrix generated with the msc.matrix function.
`samples`	a vector containing the sample names. This can include all samples or it can be a subset.
`groups`	a vector specifying to which groups (e.g. species) the samples belong to.

Value

a heatmap

Examples

```
data(exData)
data(matrices)

### run function
```

```
msc.heatmap(matrices[[”id80”]], groups = exData$species,
            samples = exData$samples )

### run function on every cluster matrix with subset of samples
### you will be asked to confirm
table(exData$species)
hybrid <- which(exData$species==”hybrid”)
# msc.heatmap(matrices[[”id97”]], groups = exData$species[hybrid],
#              samples = exData$samples[hybrid])
```

`msc.length` *Length of minicircles*

Description

The msc.length function allows you to check the length of minicircle sequences based on one fasta file.

Usage

```
msc.length(file, samples, groups)
```

Arguments

`file` the name of the fasta file containing all minicircle sequences, e.g. all.minicircles.circ.fasta.

`samples` a character vector containing the sample names.

`groups` a vector, of equal length as samples, specifying to which group (e.g. subspecies) the samples belong to.

Value

`length` a numerical vector containing the length of minicircle sequences.

`plot` a histogram visualizing the length frequency of minicircle sequences.

Examples

```
require(ggplot2)
require(ggpubr)

### run function
bf <- msc.length(file = system.file(”extdata”, ”all.minicircles.fasta”, package=”rKOMICS”),
                 samples = exData$samples, groups = exData$subspecies)
af <- msc.length(file = system.file(”extdata”, ”all.minicircles.circ.fasta”, package=”rKOMICS”),
                 samples = exData$samples, groups = exData$subspecies)

length(which(bf$length<800))
length(which(bf$length>1400))

### visualize results
hist(af$length, breaks=50)

### alter plot
ggarrange(bf$plot + labs(caption = "Before filtering"),
          af$plot + labs(caption = "After filtering"), nrow=2)
```

msc.matrix *Build cluster matrix*

Description

Clustering based on a percent identity, performed with the VSEARCH tool, will generate files in uc format. The msc.matrix function will transform every input file into a cluster matrix. The columns of the matrix correspond to the samples and the rows of the matrix correspond to the minicircle sequence cluster (MSC). The absence of a MSC in a sample is indicated with the value of zero while the presence of a MSC in a sample will be indicated with a value >= 1.

Usage

```
msc.matrix(files, samples, groups)
```

Arguments

files	a character vector containing the uc file names (output of the VSEARCH tool) e.g. all.minicircles.circ.id70.uc, all.minicircles.circ.id80.uc...
samples	a character vector containing the sample names. The vector should be order alphabetically.
groups	a vector, of equal length as samples, specifying to which group (e.g. species) the samples belong to.

Value

a list containing one cluster matrix per percent identity. The value 0 in the cluster matrix means the MSC doesn't occur in the sample. A value higher than 0 means the MSC is found at least once in the sample.

Examples

```
data(exData)

### run function

matrices <- msc.matrix(files = system.file("extdata", exData$ucs, package="rKOMICS"),
                          samples = sort(exData$samples),
                          groups = exData$species[order(exData$samples)])


### or:
data(matrices)

### show matrix with id 95%
matrices[[”id95”]]
rowSums(matrices[[”id95”]]) # --> frequency of MSC across all samples
colSums(matrices[[”id95”]]) # --> number of MSC per sample
```

---

msc.pca *Prinicple Component Analysis based on MSC*

Description

The msc.pca allows you to perform Principle Component Analysis to summarize MSCs variation in all samples or in a subset of samples.

Usage

```
msc.pca(clustmatrix, samples, groups, n = 20, labels = TRUE, title = NULL)
```

Arguments

`clustmatrix` a cluster matrix

`samples` a vector containing the names of the samples. This can include all samples or it can be a subset.

`groups` a vector specifying to which group (e.g. species) the samples belong to.

`n` number of clusters to select with highest contribution to PCA.

`labels` a logical parameter indicating whether to use labels on the PCA plot or not. Default is set to true.

`title` the title of the graph

Value

`plot` a PCA plot.

`eigenvalues` a barplot showing percentage of explained variances.

`clustnames` list of cluster names with highest contribution to PCA.

Examples

```
data(matrices)
data(exData)

### run function with all samples
res.pca <- lapply(matrices, function(x) msc.pca(x, samples = exData$samples,
                  groups = exData$species, n=30, labels=FALSE, title=NULL))

res.pca$id93$eigenvalues
res.pca$id93$plot

### use clusters with highest contribution to visualize in a heatmap
msc.heatmap(matrices[["id93"]][res.pca$id93$clustnames,], samples = exData$samples,
            groups = exData$species)

### run function with a subset of samples
### you will be asked to confirm
table(exData$species)
hybrid <- which(exData$species=="hybrid")
# pca.subset <- msc.pca(clustmatrix = matrices[["id97"]],
#                       samples = exData$samples[hybrid],
#                       groups = exData$species[hybrid], labels = TRUE,
#                       title = "PCA only with hybrids")
```

`msc.quality` *Check the quality of the assembly*

Description

The msc.quality function allows you to summarise mapping statistics generated by KOMCIS. To check whether minicircles have been assembled successfully, read, mapped read and high quality mapped read frequencies are inspected. The proportion of (near-)perfect alignments of CSB3-containing reads are herefor an important measure.

Usage

```
msc.quality(mapstats, groups)
```

Arguments

`mapstats`	a character vector containing the file names of mapping statistics (output of KOMICS).
`groups`	a vector specifying to which group (e.g. species) the samples belong to.

Value

all — a table merging mapping statistics of all samples. Mapping statistics include the number of mapped reads (MR), mapped reads with high quality (MR_HQ), CSB3-containing mapped reads (MR_CSB3) and CSB3-containing mapped read with high quality (MR_CSB3_HQ).

proportions — a list of tables containing the proportion of previous mentioned items.

plots — barplots visualizing previous results.

Examples

```
data(exData)

### run function
map <- msc.quality(mapstats = system.file("extdata", exData$mapstats, package = "rKOMICS"),
                   exData$species)

lapply(map$proportions, mean)$MR_HQ
lapply(map$proportions, mean)$MR_CSB3_HQ

### visualize results
barplot(map$proportions$MR)
```

msc.richness *Minicircle Sequence Cluster richness*

Description

The msc.richness function counts how many Minicircle Sequence Clusters (MSC) are present per sample across different percent identities.

Usage

```
msc.richness(clustmatrices, samples, groups)
```

Arguments

clustmatrices — a list of cluster matrices.

samples — a vector containing the names of the samples. This can include all samples or it can be a subset.

groups — a vector, of equal length as samples, specifying to which group (e.g. species) the samples belong to.

Value

table — a table containing the number of MSC per sample across different percent identities.

plot — a boxplot visualizing previous results.

Examples

```
require(ggplot2)
data(matrices)
data(exData)

#### run function
richness <- msc.richness(matrices, samples = exData$samples, groups = exData$species)

apply(richness$table[which(richness$table$group==”L. peruviana”),-(1:2)], 2, mean)
apply(richness$table[which(richness$table$group==”L. braziliensis”),-(1:2)], 2, mean)
apply(richness$table[which(richness$table$group==”hybrid”),-(1:2)], 2, mean)

#### visualize results
barplot(richness$table[,”id93”], names.arg = richness$table[,1],
        las=2, cex.names=0.4, main=”N of MSC at id 93”)

#### adjust plot
richness$plot + ggtitle(”MSC richness across % id”) +
                theme(axis.text.x = element_text(angle=45, hjust=1))

### show results of subset
table(exData$species)
hybrid <- which(exData$species==”hybrid”)
# richness.subset <- msc.richness(matrices, samples = exData$samples[hybrid],
#                                  groups = exData$species[hybrid])
```

msc.seqs *Retrieve sequences*

Description

The msc.seqs retrieves the DNA sequence of a minicircle sequence cluster (MSC) together with all its hit sequences.

Usage

```
msc.seqs(fastafile, ucfile, clustnumbers, writeDNA = TRUE)
```

Arguments

`fastafile` the name of the fasta file containing all minicircle sequences.

`ucfile` the name of the uc file.

`clustnumbers` a character vector containing the cluster numbers (in the format "C0", "C1", ...) of which cluster and hit sequences need to be extracted.

`writeDNA` a logical parameter which is by default set to TRUE. This will write fasta files to the current directory.

Value

a table which summarizes the number of hit sequences found in the MSC, the MSC name and where the MSC is present (strain names).

one fasta file per MSC with all its hits sequences.

Examples

```
data(exData)

### select a subset of MSC
Lpe <- which(exData$species == "L. peruviana")
specific <- msc.subset(matrices[[7]], subset = Lpe)

### run function
seq <- msc.seqs(fastafile = system.file("extdata", "all.minicircles.circ.fasta", package="rKOMICS"),
                ucfile = system.file("extdata", exData$ucs, package="rKOMICS")[7],
                clustnumbers = specific$clustnumbers, writeDNA = FALSE)
```

msc.similarity *Minicircle Sequence Cluster similarity*

Description

The msc.similarity function allows you to check the absolute and relative frequency of shared and unique MSC between different groups across different percent identities.

Usage

```
msc.similarity(clustmatrices, samples, groups)
```

Arguments

clustmatrices	a list of cluster matrices.
samples	a vector containing the names of the samples. This can include all samples or it can be a subset.
groups	a vector, of equal length as samples, specifying to which group (e.g. species) the samples belong to.

Value

absfreq	a list per percent identity containing absolute frequency values of shared and unique MSCs.
absfreq.plot	a list of barplots visualizing previous results.
relfreq	a list per percent identity containing relative frequency values of shared and unique MSCs.
relfreq.plot	one barplot visualizing previous results.

Examples

```
require(viridis)
data(matrices)
data(exData)

### run function
sim <- msc.similarity(matrices, samples = exData$samples,
                      groups = exData$species)

### visualize results (absolute frequencies)
barplot(sim$absfreq$id93)

### adjust plot (relative frequencies)
sim$relfreq.plot + scale_fill_viridis(discrete = TRUE)

sim$relfreq$id97["2"]*100
sim$relfreq$id97["3"]*100

### reduce number of groups
groups <- exData$species
levels(groups)[levels(groups)!='hybrid'] <- "non-hybrid"
sim.red <- msc.similarity(matrices, samples = exData$samples, groups = groups)
sim.red$relfreq.plot + scale_fill_viridis(discrete = TRUE)
```

`msc.subset` *Specific MSC*

Description

The msc.subset allows you to find specific MSC for a certain subset of samples.

Usage

```
msc.subset(clustmatrix, subset)
```

Arguments

`clustmatrix`	a cluster matrix
`subset`	a numerical vector indicating which subset of samples to include.

Value

`clustnumbers`	a vector containing the specific MSC names.
`freq`	frequency values of those specific MSC in the subset of samples.
`matrix`	a subset of the cluster matrix containing only those specific MSC. All samples, not in the subset, should have a value of 0 meaning the MSC is absent.
`sum`	the total number of MSC found in the indicated subset of samples.

Examples

```
data(matrices)
data(exData)

### selecting a group of samples e.g. all L. peruviana species
Lpe <- which(exData$species == ”L. peruviana”)

### run function
specific <- msc.subset(matrices[[”id97”]], subset = Lpe)

### visualize results (check if it is indeed specific)
heatmap(specific$matrix) # or:
msc.heatmap(specific$matrix, samples = exData$samples, groups = exData$species)

### find specific MSC with highest frequency
which.max(specific$freq)
```

`msc.uc` *Cluster Analyses*

Description

The msc.uc function allows you to check some clustering : insertions and deletions –> ask Fre. The total amount of minicircle sequence clusters (MSCs) found per percent identity will be calculated as well as the number of gaps.

Usage

```
msc.uc(files)
```

Arguments

`files` a character vector containing the uc file names (output of VSEARCH) e.g. all.minicircles.circ.id70.uc, all.minicircles.circ.id80.uc...

Value

`MSCs` a numerical vector containing the number of MSC per percent identity.

`perfect aligments` a numerical vector containing the proportions of perfect alignments per percent identity.

`insertions` a table showing the length and the number of insertions across different percent identities.

`deletions` a table showing the length and the number of deletions across different percent identities.

`plots` different plots showing previous results.

Examples

```
data(exData)

### run function

ucs <- msc.uc(files = system.file("extdata", exData$ucs, package="rKOMICS"))

ucs$MSCs["100"]
ucs$MSCs["97"]

### results
ucs$plots
```

`preprocess` *Filtering of minicircle sequences*

Description

Assembling minicircle sequences with KOMICS generates individual fasta files (one per sample). The preprocess function allows you to filter the minicircle sequences based on sequence length (as the size of minicircular kDNA is species-specific and variable) and circularization success. The function will write filtered individual fasta files in the current working directory.

Usage

```
preprocess(files, groups, circ = TRUE, min = 500, max = 1500, writeDNA = TRUE)
```

Arguments

`files` a character vector containing the fasta file names in the format sampleA.minicircles.fasta, sampleB.minicircles.fasta,... (output of KOMICS).

`groups` a factor specifying to which group (e.g. species) the samples belong to. It should have the same length as the list of files.

`circ` a logical parameter. By default non-circularized minicicle sequences will be excluded. If interested in non-circularized sequences as well, set the parameter to FALSE.

`min` a minimum value for the minicircle sequences length. Default value is set to 500.

`max` a maximum value for the minicircle sequences length. Default value is set to 1500.

`writeDNA` a logical parameter. By default filtered minicircle sequences will by written in fasta format to the current working directory. Set to FALSE if only interested in other output values like plots and summary.

Value

`samples`	the sample names (based on the input files).
`N_MC`	a table containing the sample name, which group it belongs to and the number of minicirce sequences (N_MC) before and after filtering.
`plot`	a barplot visualizing the number of minicircle sequences per sample before and after filtering.
`summary`	the total number of minicircle sequences before and after filtering.

Examples

```
require(ggplot2)
data(exData)

### setwd("")

### run function
table(exData$species)
pre <- preprocess(files = system.file("extdata", exData$fastafiles, package="rKOMICS"),
                  groups = exData$species,
                  circ = TRUE, min = 500, max = 1200, writeDNA = FALSE)

pre$summary

### visualize results
barplot(pre$N_MC[,"beforefiltering"],
        names.arg = pre$N_MC[,1], las=2, cex.names=0.4)

### alter plot
pre$plot + labs(caption = paste0('N of MC sequences before and after filtering, ', Sys.Date()))
```

`read.uc` *Read in uc files*

Description

Clustering based on a percent identity, performed with VSEARCH, generates files in uc format. The read.uc function allows you to read in an uc file and store cluster and hit records in individual tables.

Usage

```
read.uc(file)
```

Arguments

`file`	the name of the uc file e.g. all.minicircles.circ.id70.uc

Value

`hits`	a table containing all hit records.
`clusters`	a table containing all cluster records.
`clustnumbers`	a vector containing the cluster numbers (0-based).

`rKOMICS` *Minicircle Sequence Cluster (MSC) Analyses*

Description

It establishes a critical framework to manipulate, explore and extract biologically relevant information from mitochondrial minicircle assemblies in tens to hundreds of samples simultaneously and efficiently. This should facilitate research that aims to develop new molecular markers for identifying species-specific minicircles, or to study the ancestry of parasites for complementary insights into their evolutionary history.

Details

The DESCRIPTION file: This package was not yet installed at build time.

Index of help topics:

```
exData                  Example dataset
matrices                Example cluster matrices
msc.depth               Check the read depth of assembled minicircles
msc.heatmap             Visualization of cluster matrices
msc.length              Length of minicircles
msc.matrix              Build cluster matrix
msc.pca                 Prinicple Component Analysis based on MSC
msc.quality             Check the quality of the assembly
msc.richness            Minicircle Sequence Cluster richness
msc.seqs                Retrieve sequences
msc.similarity          Minicircle Sequence Cluster similarity
msc.subset              Specific MSC
msc.uc                  Cluster Analyses
preprocess              Filtering of minicircle sequences
rKOMICS                 Minicircle Sequence Cluster (MSC) Analyses
read.uc                 Read in uc files
```

Author(s)

Frederik Van den Broeck <`frederik.vandenbroeck@kuleuven.be`>

Manon Geerts <`mgeerts@itg.be`>

References

Van den Broeck F, Savill NJ, Imamura H, Sanders M, Maes I, Cooper S, et al. Ecological divergence and hybridization of Neotropical Leishmania parasites. Proc Natl Acad Sci U S A. 2020;117. doi: 10.1073/pnas.1920136117.

See Also

Github: `https://frebio.github.io/`

komics-suite: `https://frebio.github.io/komics/`

Index