Package ‘sitePath’

April 11, 2024

Type    Package
Title   Phylogeny-based sequence clustering with site polymorphism
Version 1.18.0

Description

Using site polymorphism is one of the ways to cluster DNA/protein sequences but it is possible for the sequences with the same polymorphism on a single site to be genetically distant. This package is aimed at clustering sequences using site polymorphism and their corresponding phylogenetic trees. By considering their location on the tree, only the structurally adjacent sequences will be clustered. However, the adjacent sequences may not necessarily have the same polymorphism. So a branch-and-bound like algorithm is used to minimize the entropy representing the purity of site polymorphism of each cluster.

License MIT + file LICENSE

Depends R (>= 4.1)
Imports RColorBrewer, Rcpp, ape, aplot, ggplot2, ggrepel, ggtree, graphics, grDevices, gridExtra, methods, parallel, seqinr, stats, tidytree, utils
Suggests BiocStyle, devtools, knitr, magick, rmarkdown, testthat

LinkingTo Rcpp
RoxygenNote 7.1.2

Encoding UTF-8
VignetteBuilder knitr

URL https://wuaipinglab.github.io/sitePath/

BugReports https://github.com/wuaipinglab/sitePath/issues
biocViews Alignment, MultipleSequenceAlignment, Phylogenetics, SNP, Software

git_url https://git.bioconductor.org/packages/sitePath

git_branch RELEASE_3_18

git_last_commit b07adee

git_last_commit_date 2023-10-24
Repository  Bioconductor 3.18
Date/Publication  2024-04-10
Author  Chengyang Ji [aut, cre, cph] ([https://orcid.org/0000-0001-9258-5453>]),
        Hangyu Zhou [ths],
        Aiping Wu [ths]
Maintainer  Chengyang Ji <chongyang.ji12@alumni.xjtlu.edu.cn>

R topics documented:

allSitesName .................................................. 3
as.data.frame.fixationSites .................................... 4
extractSite ....................................................... 5
extractTips ...................................................... 5
fixationIndels .................................................. 7
fixationPath ..................................................... 7
fixationSites .................................................... 8
groupTips ....................................................... 9
h3n2_align ...................................................... 10
h3n2_tree ........................................................ 11
lineagePath ..................................................... 11
paraFixSites ..................................................... 14
parallelSites ..................................................... 16
phyMSAmatched .................................................. 17
plot.phyMSAmatched ............................................ 18
plotFixationSites ............................................... 20
plotMutSites ..................................................... 21
plotParallelSites ............................................... 22
plotSingleSite ................................................. 23
reexports ....................................................... 24
sars2_align ..................................................... 24
sars2_tree ....................................................... 25
setSiteNumbering ............................................... 25
similarityMatrix ............................................... 26
sitePath-deprecated ............................................ 27
sitesMinEntropy ............................................... 27
SNPsites ......................................................... 28
zikv_align ....................................................... 29
zikv_tree ........................................................ 29

Index 30
allSitesName

Retrieve position of all the sites

Description

The function is a way to get position of the resulting sites from `SNPsites`, `fixationSites` and `parallelSites`. The numbering is consistent with what’s being set by `setSiteNumbering`

Usage

```r
allSitesName(x, ...) # S3 method for class 'SNPsites'
allSitesName(x, ...) # S3 method for class 'sitesMinEntropy'
allSitesName(x, ...) # S3 method for class 'fixationSites'
allSitesName(x, ...) # S3 method for class 'parallelSites'
allSitesName(x, type = c("paraFix", "fixation", "parallel"), ...)```

Arguments

- `x` The object containing the sites from analysis
- `...` Other arguments
- `type` Return fixation or parallel sites

Value

An integer vector for sites position

Examples

```r
data(zikv_tree)
msaPath <- system.file('extdata', 'ZIKV.fasta', package = 'sitePath')
tree <- addMSA(zikv_tree, msaPath = msaPath, msaFormat = 'fasta')
snp <- SNPsites(tree)
allSitesName(snp)```
as.data.frame.fixationSites

Convert results to Data Frame

Description

Convert return of functions in sitePath package to a \code{data.frame} so can be better worked with. The group name for each tip is the same as \code{groupTips}.

A \code{fixationSites} object will output the mutation name of the fixation and the cluster name before and after the mutation.

An \code{SNPsites} object will output the tip name with the SNP and its position.

An \code{parallelSites} object will output the tip name with the group name and mutation info.

Usage

\begin{verbatim}
## S3 method for class 'fixationSites'
as.data.frame(x, row.names = NULL, optional = FALSE, ...)

## S3 method for class 'SNPsites'
as.data.frame(x, row.names = NULL, optional = FALSE, ...)

## S3 method for class 'parallelSites'
as.data.frame(x, row.names = NULL, optional = FALSE, ...)
\end{verbatim}

Arguments

- \code{x} The object to be converted to \code{data.frame}.
- \code{row.names} Unimplemented.
- \code{optional} Unimplemented.
- \code{...} Other arguments.

Value

A \code{data.frame} object.

Examples

\begin{verbatim}
data(zikv_tree_reduced)
data(zikv_align_reduced)
tree <- addMSA(zikv_tree_reduced, alignment = zikv_align_reduced)
fixations <- fixationSites(lineagePath(tree))
as.data.frame(fixations)
\end{verbatim}
**extractSite**

*Extract tips for a single site*

**Description**

The functions in sitePath usually include the results on more than one site. The function extractSite can be used to extract the predicted result on a single site.

**Usage**

```r
extractSite(x, site, ...)
```

### S3 method for class 'fixationSites'

```r
estractSite(x, site, ...)
```

**Arguments**

- `x` A fixationSites or a parallelSites object. More type will be supported in the later version.
- `site` A site included in the result.
- `...` Other arguments

**Value**

The predicted result of a single site

**Examples**

```r
data(zikv_tree_reduced)
data(zikv_align_reduced)
tree <- addMSA(zikv_tree_reduced, alignment = zikv_align_reduced)
mutations <- fixationSites(lineagePath(tree))
eextractSite(mutations, 139)
```

**extractTips**

*Extract grouped tips for a single site*

**Description**

The result of fixationSites and sitePath contains all the possible sites with fixation mutation. The function extractTips retrieves the name of the tips involved in the fixation.

For lineagePath, the function extractTips groups all the tree tips according to the amino acid/nucleotide of the site.

For parallelSites and sitePara object, the function extractTips retrieve all the tips with parallel mutation.
Usage

extractTips(x, ...)

## S3 method for class 'lineagePath'
extractTips(x, site, ...)

## S3 method for class 'sitesMinEntropy'
extractTips(x, site, ...)

## S3 method for class 'fixationSites'
extractTips(x, site, select = 1, ...)

## S3 method for class 'sitePath'
extractTips(x, select = 1, ...)

## S3 method for class 'parallelSites'
extractTips(x, site, ...)

## S3 method for class 'sitePara'
extractTips(x, ...)

Arguments

x A fixationSites or a sitePath object.
...

site A site predicted to experience fixation.
select For a site, there theoretically might be more than one fixation on different lineages. You may use this argument to extract for a specific fixation of a site. The default is the first fixation of the site.

Value

Tree tips grouped as list

Examples

data(zikv_tree_reduced)
data(zikv_align_reduced)
tree <- addMSA(zikv_tree_reduced, alignment = zikv_align_reduced)
migrations <- fixationSites(lineagePath(tree))
extractTips(migrations, 139)
**fixationIndels** | *Fixation indels prediction*

**Description**

The fixation of insertions of deletions.

**Usage**

```r
fixationIndels(x, ...)
```

```r
## S3 method for class 'sitesMinEntropy'
fixationIndels(x, ...)
```

**Arguments**

- `x`: The return from `sitesMinEntropy` function.
- `...`: Other arguments.

**Value**

A `fixationIndels` object.

**Examples**

```r
data(zikv_tree_reduced)
data(zikv_align_reduced)
tree <- addMSA(zikv_tree_reduced, alignment = zikv_align_reduced)
fixationIndels(sitesMinEntropy(tree))
```

---

**fixationPath** | *Accumulation of fixed mutation as a tree*

**Description**

The tips are clustered according to the fixation sites. The transition of fixation sites will be plotted as a phylogenetic tree. The length of each branch represents the number of fixation mutation between two clusters. The name of the tree tips indicate the number of sequences in the cluster.

**Usage**

```r
fixationPath(x, ...)
```

```r
## S3 method for class 'sitesMinEntropy'
fixationPath(x, minEffectiveSize = NULL, ...)
```

```r
## S3 method for class 'fixationSites'
fixationPath(x, minEffectiveSize = NULL, ...)
```
Arguments

- **x**  The return from `fixationSites` function.
- **...** Further arguments passed to or from other methods.
- **minEffectiveSize**
  
  The minimum size for a tip cluster.

Value

An `fixationPath` object

Examples

```r
data(zikv_tree_reduced)
data(zikv_align_reduced)
tree <- addMSA(zikv_tree_reduced, alignment = zikv_align_reduced)
paths <- lineagePath(tree)
mutations <- fixationSites(paths)
fixationPath(mutations)
```

Description

After finding the `lineagePath` of a phylogenetic tree, `fixationSites` uses the result to find those sites that show fixation on some, if not all, of the lineages. The number of tips before and after the fixation mutation is expected to be more than `minEffectiveSize`. Also, the fixation will be skipped if the amino acid/nucleotide is gap or ambiguous character. A lineage has to have at least one fixation mutation to be reported.

Usage

```r
fixationSites(paths, ...)

## S3 method for class 'lineagePath'
fixationSites(
  paths,
  minEffectiveSize = NULL,
  searchDepth = 1,
  method = c("compare", "insert", "delete"),
  ...
)

## S3 method for class 'sitesMinEntropy'
fixationSites(paths, ...)

## S3 method for class 'paraFixSites'
fixationSites(paths, ...)
```
Arguments

paths A lineagePath object returned from `lineagePath` function.

... further arguments passed to or from other methods.

minEffectiveSize The minimum number of tips in a group.

searchDepth The function uses heuristic search but the termination of the search cannot be intrinsically decided. `searchDepth` is needed to tell the search when to stop.

method The strategy for predicting the fixation. The basic approach is entropy minimization and can be achieved by adding or removing fixation point, or by comparing the two.

Value

A fixationSites object.

See Also

`as.data.frame.fixationSites`

Examples

data(zikv_tree_reduced)
data(zikv_align_reduced)
tree <- addMSA(zikv_tree_reduced, alignment = zikv_align_reduced)
fixationSites(lineagePath(tree))

datadput("groupTips", fixture"

Description

The grouping of tree tips

Usage

groupTips(tree, ...)

# S3 method for class 'phyMSAmatched'
groupTips(
  tree,
  similarity = NULL,
  simMatrix = NULL,
  forbidTrivial = TRUE,
  tipnames = TRUE,
  ...
)

---
## S3 method for class 'lineagePath'

```r
groupTips(tree, tipnames = TRUE, ...)
```

## S3 method for class 'sitesMinEntropy'

```r
groupTips(tree, tipnames = TRUE, ...)
```

## S3 method for class 'fixationSites'

```r
groupTips(tree, tipnames = TRUE, ...)
```

## S3 method for class 'fixationPath'

```r
groupTips(tree, tipnames = TRUE, ...)
```

### Arguments

- **tree**
  The return from `addMSA, lineagePath, sitesMinEntropy` or other functions.
- **...**
  Other arguments.
- **similarity**
  This decides how minor SNPs are to remove. If provided as fraction between 0 and 1, then the minimum number of SNP will be total tips times similarity. If provided as integer greater than 1, the minimum number will be similarity. The default similarity is 0.05 for `lineagePath`.
- **simMatrix**
  Deprecated and will not have effect.
- **forbidTrivial**
  Does not allow trivial trimming.
- **tipnames**
  If return tips as integer or tip names.

### Value

`groupTips` returns grouping of tips.

### Examples

```r
data(zikv_tree)
data(zikv_align)
tree <- addMSA(zikv_tree, alignment = zikv_align)
groupTips(tree)
```

---

**h3n2_align**

*Multiple sequence alignment of H3N2’s HA protein*

### Description

The raw protein sequences were downloaded from NCBI database and aligned using MAFFT ([https://mafft.cbrc.jp/alignment/software/](https://mafft.cbrc.jp/alignment/software/)).

`h3n2_align_reduced` is a truncated version of `h3n2_align`
**h3n2_tree**

**Usage**

- `data(h3n2_align)`
- `data(h3n2_align_reduced)`

**Format**

- an alignment object
- an alignment object

---

**Description**

Tree was built from `h3n2_align` using RAxML ([http://www.exelixis-lab.org/](http://www.exelixis-lab.org/)) with default settings.

`h3n2_tree_reduced` is a truncated version of `h3n2_tree`.

**Usage**

- `data(h3n2_tree)`
- `data(h3n2_tree_reduced)`

**Format**

- a phylo object
- a phylo object

---

**lineagePath**

**Resolving lineage paths using SNP**

**Description**

`lineagePath` finds the lineages of a phylogenetic tree providing the corresponding sequence alignment. This is done by finding 'major SNPs' which usually accumulate along the evolutionary pathways.

`sneakPeek` is intended to plot 'similarity' (actually the least percentage of 'major SNP') as a threshold against number of output `lineagePath`. This plot is intended to give user a rough view about how many lineages they could expect from the 'similarity' threshold in the function `lineagePath`. The number of `lineagePath` is preferably not be too many or too few. The result excludes where the number of `lineagePath` is greater than number of tips divided by 20 or user-defined `maxPath`. The zero `lineagePath` result will also be excluded.
When used on the return of sneakPeek, a lineagePath with the closest similarity will be retrieved from the returned value. similarity has no effect when using on paraFixSites object

Usage

lineagePath(tree, similarity, ...)

## S3 method for class 'phylo'
lineagePath(  
  tree,  
  similarity = NULL,  
  alignment = NULL,  
  seqType = c("AA", "DNA", "RNA"),  
  reference = NULL,  
  gapChar = ",",  
  minSkipSize = NULL,  
  ...  
)

## S3 method for class 'treedata'
lineagePath(tree, ...)

## S3 method for class 'phyMSAmatched'
lineagePath(  
  tree,  
  similarity = NULL,  
  simMatrix = NULL,  
  forbidTrivial = TRUE,  
  ...  
)

sneakPeek(tree, step = 9, maxPath = NULL, minPath = 0, makePlot = TRUE)

## S3 method for class 'sneakPeekedPaths'
lineagePath(tree, similarity, ...)

## S3 method for class 'paraFixSites'
lineagePath(tree, similarity = NULL, ...)

Arguments

tree The return from addMSA or sneakPeek function.
similarity The parameter for identifying phylogenetic pathway using SNP. If provided as fraction between 0 and 1, then the minimum number of SNP will be total tips times Nmin. If provided as integer greater than 1, the minimum number will be Nmin.
...

Other arguments.
alignment  An alignment object. This commonly can be from sequence parsing function in the seqinr package. Sequence names in the alignment should include all tip.label in the tree

seqType  The type of the sequence in the alignment file. The default is "AA" for amino acid. The other options are "DNA" and "RNA".

reference  Name of reference for site numbering. The name has to be one of the sequences’ name. The default uses the intrinsic alignment numbering

gapChar  The character to indicate gap. The numbering will skip the gapChar for the reference sequence.

minSkipSize  The minimum number of tips to have gap or ambiguous amino acid/nucleotide for a site to be ignored in other analysis. This will not affect the numbering. The default is 0.8.

simMatrix  Deprecated and will not have effect.

forbidTrivial  Does not allow trivial trimming.

step  the 'similarity' window for calculating and plotting. To better see the impact of threshold on path number. The default is 10.

maxPath  maximum number of path to return show in the plot. The number of path in the raw tree can be far greater than trimmed tree. To better see the impact of threshold on path number. This is preferably specified. The default is one 20th of tree tip number.

minPath  minimum number of path to return show in the plot. To better see the impact of threshold on path number. The default is 1.

makePlot  Whether make a plot when return.

Value

Lineage path represent by node number.

sneakPeek return the similarity threhold against number of lineagePath. There will be a simple dot plot between threshold and path number if makePlot is TRUE.

Examples

data('zikv_tree')
data('zikv_align')
tree <- addMSA(zikv_tree, alignment = zikv_align)
lineagePath(tree)
sneakPeek(tree, step = 3)
x <- sneakPeek(tree, step = 3)
lineagePath(x, similarity = 0.05)
paraFixSites

The fixation sites with mutation on parallel lineage

Description

The operation between the results of fixationSites and parallelSites.

Usage

paraFixSites(x, ...)

## S3 method for class 'phylo'
paraFixSites(
  x,
  alignment = NULL,
  seqType = c("AA", "DNA", "RNA"),
  Nmin = NULL,
  reference = NULL,
  gapChar = "-",
  minSkipSize = NULL,
  ...
)

## S3 method for class 'treedata'
paraFixSites(x, ...)

## S3 method for class 'lineagePath'
paraFixSites(
  x,
  minEffectiveSize = NULL,
  searchDepth = 1,
  method = c("compare", "insert", "delete"),
  ...
)

## S3 method for class 'sitesMinEntropy'
paraFixSites(
  x,
  category = c("intersect", "union", "parallelOnly", "fixationOnly"),
  minSNP = NULL,
  mutMode = c("all", "exact", "pre", "post"),
  ...
)

Arguments

x  A lineagePath object returned from lineagePath function.
... further arguments passed to or from other methods.

alignment An alignment object. This commonly can be from sequence parsing function in the seqinr package. Sequence names in the alignment should include all tip.label in the tree.

seqType The type of the sequence in the alignment file. The default is "AA" for amino acid. The other options are "DNA" and "RNA".

Nmin The parameter for identifying phylogenetic pathway using SNP. If provided as fraction between 0 and 1, then the minimum number of SNP will be total tips times Nmin. If provided as integer greater than 1, the minimum number will be Nmin.

reference Name of reference for site numbering. The name has to be one of the sequences’ name. The default uses the intrinsic alignment numbering.

gapChar The character to indicate gap. The numbering will skip the gapChar for the reference sequence.

minSkipSize The minimum number of tips to have gap or ambiguous amino acid/nucleotide for a site to be ignored in other analysis. This will not affect the numbering. The default is 0.8.

minEffectiveSize The minimum number of tips in a group.

searchDepth The function uses heuristic search but the termination of the search cannot be intrinsically decided. searchDepth is needed to tell the search when to stop.

method The strategy for predicting the fixation. The basic approach is entropy minimization and can be achieved by adding or removing fixation point, or by comparing the two.

category Could be parallelOnly, fixationOnly, intersect or union.

minSNP The minimum number of mutations to be qualified as parallel on at least two lineages. The default is 1.

mutMode The strategy for finding parallel site. The default all is to consider any mutation regardless of the amino acid/nucleotide before and after mutation; Or exact to force mutation to be the same; Or pre/post to select the site having amino acid/nucleotide before/after mutation.

Value

A paraFixSites object.

Examples

data(zikv_tree_reduced)

data(zikv_align_reduced)

paraFixSites(zikv_tree_reduced, alignment = zikv_align_reduced)
parallelSites  Mutation across multiple phylogenetic lineages

Description

A site may have mutated on parallel lineages. Mutation can occur on the same site across the phylogenetic lineages solved by lineagePath. The site will be considered mutated in parallel if the mutation occurs on the non-overlap part of more than two lineages. The amino acid/nucleotide before and after the mutation can be allowed different on different lineages or only the exact same mutations are considered.

Usage

parallelSites(x, ...)

## S3 method for class 'lineagePath'
parallelSites(
  x,
  minSNP = NULL,
  mutMode = c("all", "exact", "pre", "post"),
  ...
)

## S3 method for class 'sitesMinEntropy'
parallelSites(
  x,
  minSNP = NULL,
  mutMode = c("all", "exact", "pre", "post"),
  ...
)

## S3 method for class 'paraFixSites'
parallelSites(x, ...)

Arguments

x  
A lineagePath or a sitesMinEntropy object.

...  
The arguments in sitesMinEntropy.

minSNP  
The minimum number of mutations to be qualified as parallel on at least two lineages. The default is 1.

mutMode  
The strategy for finding parallel site. The default all is to consider any mutation regardless of the amino acid/nucleotide before and after mutation; Or exact to force mutation to be the same; Or pre/post to select the site having amino acid/nucleotide before/after mutation.
**Value**

A parallelSites object

**Examples**

```r
data(zikv_tree_reduced)
data(zikv_align_reduced)
tree <- addMSA(zikv_tree_reduced, alignment = zikv_align_reduced)
paths <- lineagePath(tree)
x <- sitesMinEntropy(paths)
parallelSites(x)
```

---

**phyMSAmatched**  
*Add matching sequence alignment to the tree*

**Description**

addMSA wraps `read.alignment` function in seqinr package and helps match names in tree and sequence alignment. Either provide the file path to an alignment file and its format or an alignment object from the return of `read.alignment` function. If both the file path and alignment object are given, the function will use the sequence in the alignment file.

**Usage**

```r
addMSA(tree, ...)
```

```r
## S3 method for class 'phylo'
addMSA(
  tree,
  msaPath = "",
  msaFormat = c("fasta", "clustal", "phylip", "mase", "msf"),
  alignment = NULL,
  seqType = c("AA", "DNA", "RNA"),
  ...
)
```

```r
## S3 method for class 'treedata'
addMSA(tree, ...)
```

**Arguments**

- `tree`: A phylo object. This commonly can be from tree parsing function in ape or ggtree. All the tip.label should be found in the sequence alignment. The tree is supposed to be fully resolved (bifurcated) and will be resolved by multi2di if is.binary gives FALSE.
- `...`: Other arguments.
- `msaPath`: The file path to the multiple sequence alignment file.
The format of the multiple sequence alignment file. The internal uses the `read.alignment` from `seqinr` package to parse the sequence alignment. The default is “fasta” and it also accepts ”clustal”, “phylip”, “mase”, “msf”.

An alignment object. This commonly can be from sequence parsing function in the `seqinr` package. Sequence names in the alignment should include all `tip.label` in the tree.

The type of the sequence in the alignment file. The default is ”AA” for amino acid. The other options are ”DNA” and ”RNA”.

Since 1.5.12, the function returns a `phyMSAmatched` object to avoid S3 methods used on `phylo` (better encapsulation).

The function name `plot` is used to keep the compatibility with previous versions, but they do not behave like the generic `plot` function since 1.5.4.

A `phyMSAmatched` object will be plotted as a tree diagram.

A `lineagePath` object will be plotted as a tree diagram and paths are black solid line while the trimmed nodes and tips will use gray dashed line.

A `parallelSites` object will be plotted as original phylogenetic tree marked with parallel mutations attached as dot plot.

A `fixationSites` object will be plotted as original phylogenetic tree marked with fixation substitutions.

A `sitePath` object can be extracted by using `extractSite` on the return of `fixationSites`.

A `fixationIndels` object will be plotted as original phylogenetic tree marked with indel fixation.

A `fixationPath` object will be plotted as a `phylo` object. The tips are clustered according to the fixation sites. The transition of fixation sites will be plotted as a phylogenetic tree. The length of each branch represents the number of fixation mutation between two clusters.
Usage

## S3 method for class 'phyMSAmatched'
plot(x, y = TRUE, ...)

## S3 method for class 'lineagePath'
plot(x, y = TRUE, showTips = FALSE, ...)

## S3 method for class 'parallelSites'
plot(x, y = TRUE, ...)

## S3 method for class 'fixationSites'
plot(x, y = TRUE, tipsGrouping = NULL, ...)

## S3 method for class 'sitePath'
plot(x, y = NULL, select = NULL, showTips = FALSE, ...)

## S3 method for class 'fixationIndels'
plot(x, y = TRUE, ...)

## S3 method for class 'fixationPath'
plot(x, y = TRUE, ...)

Arguments

- **x**  
The object to plot.
- **y**  
  Whether to show the fixation mutation between clusters. For lineagePath object and sitePath object, it is deprecated and no longer have effect since 1.5.4.
- **...**  
  Other arguments. Since 1.5.4, the function uses ggtree as the base function to make plots so the arguments in plot.phylo will no longer work.
- **showTips**  
  Whether to plot the tip labels. The default is FALSE.
- **tipsGrouping**  
  A list to hold the grouping of tips for how the tree will be colored.
- **select**  
  For a sitePath object, it can have result on more than one evolution pathway. This is to select which path to plot. The default is NULL which will plot all the paths. It is the same as select in plotSingleSite.

Value

A ggplot object to make the plot.

Examples

data(zikv_tree)
data(zikv_align)
tree <- addMSA(zikv_tree, alignment = zikv_align)
plot(tree)
paths <- lineagePath(tree)
plot(paths)
parallel <- parallelSites(paths)
```r
plot(parallel)
fixations <- fixationSites(paths)
plot(fixations)
sp <- extractSite(fixations, 139)
plot(sp)
x <- fixationPath(fixations)
plot(x)
```

---

## plotFixationSites

**Plot the result of fixation sites**

### Description

Visualize the results of `paraFixSites`

### Usage

```r
plotFixationSites(x, ...)
```

#### S3 method for class 'fixationSites'

```r
plotFixationSites(x, site = NULL, ...)
```

#### S3 method for class 'paraFixSites'

```r
plotFixationSites(x, site = NULL, ...)
```

### Arguments

- **x**: return from `paraFixSites`
- **...**: further arguments passed to or from other methods.
- **site**: the number of the site according to `setSiteNumbering`. If not provided, all sites will be plotted as labels on the tree.

### Value

A `ggplot` object.

### Examples

```r
data(zikv_tree_reduced)
data(zikv_align_reduced)
paraFix <- paraFixSites(zikv_tree_reduced, alignment = zikv_align_reduced)
plotFixationSites(paraFix)
```
plotMutSites

Plot tree and mutation sites

Description

The mutated sites for each tip in a phylogenetic tree will be represented as colored dots positioned by their site number.

Usage

plotMutSites(x, ...)

## S3 method for class 'SNPsites'
plotMutSites(x, showTips = FALSE, ...)

## S3 method for class 'lineagePath'
plotMutSites(x, ...)

## S3 method for class 'parallelSites'
plotMutSites(x, ...)

## S3 method for class 'fixationSites'
plotMutSites(x, ...)

## S3 method for class 'paraFixSites'
plotMutSites(
  x,
  widthRatio = 0.75,
  fontSize = 3.88,
  dotSize = 1,
  lineSize = 0.5,
  ...
)

Arguments

x An SNPsites object.
...
showTips Whether to plot the tip labels. The default is FALSE.
widthRatio The width ratio between tree plot and SNP plot
fontSize The font size of the mutation label in tree plot
dotSize The dot size of SNP in SNP plot
lineSize The background line size in SNP plot
plotParallelSites

Value

A tree plot with SNP as dots for each tip.

Examples

data(zikv_tree_reduced)
data(zikv_align_reduced)
tree <- addMSA(zikv_tree_reduced, alignment = zikv_align_reduced)
plotMutSites(SNPsites(tree))

plotParallelSites

Plot the result of fixation sites

Description

Visualize the results of paraFixSites

Usage

plotParallelSites(x, ...)

## S3 method for class 'parallelSites'
plotParallelSites(x, site = NULL, ...)

## S3 method for class 'paraFixSites'
plotParallelSites(x, site = NULL, ...)

Arguments

x return from paraFixSites
...

further arguments passed to or from other methods.
site the number of the site according to setSiteNumbering

Value

A ggplot object.

Examples

data(zikv_tree)
data(zikv_align)
paraFix <- paraFixSites(zikv_tree, alignment = zikv_align)
plotParallelSites(paraFix)
plotSingleSite

Description

Plot and color the tree according to amino acid/nucleotide of the selected site. The color scheme depends on the seqType set in addMSA function.

For lineagePath, the tree will be colored according to the amino acid of the site. The color scheme tries to assign distinguishable color for each amino acid.

For parallelSites, the tree will be colored according to the amino acid of the site if the mutation is not fixed.

For fixationSites, it will color the ancestral tips in red, descendant tips in blue and excluded tips in grey.

Usage

plotSingleSite(x, site, ...)

## S3 method for class 'lineagePath'
plotSingleSite(x, site, showPath = TRUE, showTips = FALSE, ...)

## S3 method for class 'sitesMinEntropy'
plotSingleSite(x, site, ...)

## S3 method for class 'parallelSites'
plotSingleSite(x, site, showPath = TRUE, ...)

## S3 method for class 'fixationSites'
plotSingleSite(x, site, select = NULL, ...)

Arguments

x  The object to plot.

site  For lineagePath, it can be any site within sequence length. For fixationSites and parallelSites, it is restrained to a predicted fixation site. The numbering is consistent with the reference defined by setSiteNumbering.

...  Other arguments. Since 1.5.4, the function uses ggtree as the base function to make plots so the arguments in plot.phylo will no longer work.

showPath  If plot the lineage result from lineagePath. The default is TRUE.

showTips  Whether to plot the tip labels. The default is FALSE.

select  Select which fixation path in to plot. The default is NULL which will plot all the fixations.
Value

Since 1.5.4, the function returns a ggplot object so on longer behaviors like the generic `plot` function.

See Also

`plot.sitePath`

Examples

```r
data(zikv_tree)
data(zikv_align)
tree <- addMSA(zikv_tree, alignment = zikv_align)
paths <- lineagePath(tree)
plotSingleSite(paths, 139)
fixations <- fixationSites(paths)
plotSingleSite(fixations, 139)
```

---

sars2_align

Multiple sequence alignment of SARS-CoV-2 genome CDS

Description


Usage

```r
data(sars2_align)
```

Format

an alignment object
**Description**

Tree was built from `sars2_align` using RAxML (http://www.exelixis-lab.org/) with default settings. The tip EPI_ISL_402125 was used as the outgroup to root the tree.

**Usage**

```r
data(sars2_tree)
```

**Format**

a phylo object

---

**setSiteNumbering**

*Set site numbering to the reference sequence*

**Description**

A reference sequence can be used to define a global site numbering scheme for multiple sequence alignment. The gap in the reference sequence will be skipped for the numbering. Also, the site that is gap or amino acid/nucleotide for too many tips will be ignored but won’t affect numbering.

**Usage**

```r
setSiteNumbering(x, reference, gapChar, ...)
```

## S3 method for class 'phyMSAmatched'
`setSiteNumbering(x, reference = NULL, gapChar = "-", minSkipSize = NULL, ...)`

**Arguments**

- `x`: The object to set site numbering. It could be a `phyMSAmatched` or a `lineagePath` object.
- `reference`: Name of reference for site numbering. The name has to be one of the sequences’ name. The default uses the intrinsic alignment numbering.
- `gapChar`: The character to indicate gap. The numbering will skip the gapChar for the reference sequence.
- `...`: Further arguments passed to or from other methods.
- `minSkipSize`: The minimum number of tips to have gap or ambiguous amino acid/nucleotide for a site to be ignored in other analysis. This will not affect the numbering. The default is 0.8.
similarityMatrix

Value

The input \( x \) with numbering mapped to reference.

Examples

data(zikv_tree)
msaPath <- system.file('extdata', 'ZIKV.fasta', package = 'sitePath')
tree <- addMSA(zikv_tree, msaPath = msaPath, msaFormat = 'fasta')
setSiteNumbering(tree)

<table>
<thead>
<tr>
<th>similarityMatrix</th>
<th>Similarity between sequences</th>
</tr>
</thead>
</table>

Description

Get similarity between aligned sequences with gap ignored.

Usage

similarityMatrix(tree)

Arguments

tree  
The return from addMSA function.

Value

A diagonal matrix of similarity between sequences.

Examples

data(zikv_tree)
data(zikv_align)
tree <- addMSA(zikv_tree, alignment = zikv_align)
simMatrix <- similarityMatrix(tree)
Description

These functions are provided for compatibility with older versions of ‘sitePath’ only, and will be
defunct at the next release.

Details

The following functions are deprecated and will be made defunct; use the replacement indicated
below:

- multiFixationSites: fixationSites

sitesMinEntropy  Fixation sites prediction

Description

After finding the lineagePath of a phylogenetic tree, sitesMinEntropy perform entropy mini-
mization on every site of the sequence to group the tips according to amino acid/nucleotide.

Usage

sitesMinEntropy(x, ...)

## S3 method for class 'lineagePath'
sitesMinEntropy(
x,
  minEffectiveSize = NULL,
searchDepth = 1,
  method = c("compare", "insert", "delete"),
  ...
)

Arguments

- x  A lineagePath object returned from lineagePath function.
- ... further arguments passed to or from other methods.
- minEffectiveSize  The minimum number of tips in a group.
- searchDepth  The function uses heuristic search but the termination of the search cannot be
  intrinsically decided. searchDepth is needed to tell the search when to stop.
- method  The strategy for predicting the fixation. The basic approach is entropy minimiza-
  tion and can be achieved by adding or removing fixation point, or by comparing
  the two.
Value

A `sitesMinEntropy` object.

Examples

```r
data(zikv_tree_reduced)
data(zikv_align_reduced)
tree <- addMSA(zikv_tree_reduced, alignment = zikv_align_reduced)
sitesMinEntropy(lineagePath(tree))
```

---

SNPsites | Finding sites with variation

Description

Single nucleotide polymorphism (SNP) in the whole package refers to variation of amino acid. SNPsites will try to find SNP in the multiple sequence alignment. A reference sequence and gap character may be specified to number the site.

Usage

```r
SNPsites(tree, ...)  
```

### S3 method for class 'phyMSAmatched'

```r
SNPsites(tree, minSNP = NULL, ...)
```

Arguments

- `tree`: A `phyMSAmatched` object.
- `...`: Other arguments
- `minSNP`: Minimum number of a mutation to be a SNP. The default is 10th of the total tree tips.

Value

A SNPsites object.

Examples

```r
data(zikv_tree_reduced)
data(zikv_align_reduced)
tree <- addMSA(zikv_tree_reduced, alignment = zikv_align_reduced)
SNPsites(tree)
```
**zikv_align**

*Multiple sequence alignment of Zika virus polyprotein*

**Description**


zikv_align_reduced is a truncated version of zikv_align

**Usage**

```r
data(zikv_align)
data(zikv_align_reduced)
```

**Format**

- an alignment object
- an alignment object

**zikv_tree**

*Phylogenetic tree of Zika virus polyprotein*

**Description**

Tree was built from zikv_align using RAxML ([http://www.exelixis-lab.org/](http://www.exelixis-lab.org/)) with default settings. The tip ANK57896 was used as outgroup to root the tree.

zikv_tree_reduced is a truncated version of zikv_tree

**Usage**

```r
data(zikv_tree)
data(zikv_tree_reduced)
```

**Format**

- a phylo object
- a phylo object
Index

* datasets
  h3n2_align, 10
  h3n2_tree, 11
  sars2_align, 24
  sars2_tree, 25
  zikv_align, 29
  zikv_tree, 29

* internal
  reexports, 24

addMSA, 10, 12, 23, 26
addMSA (phyMSAmatched), 17
allSitesName, 3
ape, 17
as.data.frame.fixationSites, 4, 9
as.data.frame.parallelSites
  (as.data.frame.fixationSites), 4
as.data.frame.SNPsites
  (as.data.frame.fixationSites), 4
as.phylo, 24
as.phylo (reexports), 24
as.treedata, 24
as.treedata (reexports), 24
data.frame, 4
extractSite, 5, 18
extractTips, 5
fixationIndels, 7, 18
fixationPath, 7, 18
fixationSites, 3–5, 8, 8, 14, 18, 23, 27
ggplot2, 18
ggtree, 17, 19, 23
groupTips, 4, 9
h3n2_align, 10, 11
h3n2_align_reduced (h3n2_align), 10
h3n2_tree, 11
h3n2_tree_reduced (h3n2_tree), 11
is.binary, 17
lineagePath, 5, 8–11, 11, 14, 16, 18, 23, 25, 27
list, 6
multi2di, 17
multiFixationSites
  (sitePath-deprecated), 27
paraFixSites, 12, 14, 20, 22
parallelSites, 3–5, 14, 16, 18, 23
phylo, 17
phyMSAmatched, 17, 18, 25, 28
plot, 18, 24
plot.fixationIndels
  (plot.phyMSAmatched), 18
plot.fixationPath
  (plot.phyMSAmatched), 18
plot.fixationSites
  (plot.phyMSAmatched), 18
plot.lineagePath
  (plot.phyMSAmatched), 18
plot.parallelSites
  (plot.phyMSAmatched), 18
plot.phyMSAmatched, 18
plot.sitePath, 24
plot.sitePath
  (plot.phyMSAmatched), 18
plotFixationSites, 20
plotMutSites, 21
plotParallelSites, 22
plotSingleSite, 19, 23
read.alignment, 17, 18, 24
read.alignment (reexports), 24
read.tree, 24
read.tree (reexports), 24
reexports, 24
INDEX

sars2_align, 24, 25
sars2_tree, 25
seqinr, 13, 15, 17, 18
setSiteNumbering, 3, 20, 22, 23, 25
similarityMatrix, 26
sitePath-deprecated, 27
sitesMinEntropy, 7, 10, 16, 27
sneakPeek (lineagePath), 11
SNPsites, 3, 4, 21, 28

zikv_align, 29, 29
zikv_align_reduced (zikv_align), 29
zikv_tree, 29
zikv_tree_reduced (zikv_tree), 29