Package ‘cloneid’

March 8, 2019

Type Package

Title Clone Identity Inference Framework

Version 1.0.1

Date 2017-03-03

Author Noemi Andor

Maintainer Noemi Andor <cloneid.r@gmail.com>

Description Takes on a ``Manager's'' role, coordinating between perspectives, interacting with a MySQL database and enabling users to: i) view individual clone perspectives; ii) measure how well they agree and iii) merge clone perspectives to characterize a clone's identity (e.g. its genomic instability).

License GPL-2

URL https://github.com/noemiandor/cloneid

Depends R (>= 3.0)

Imports RMySQL, qualV, gplots(>= 3.0.1), gdata (>= 2.17.0), RColorBrewer, gtools, flexclust, Matrix, expands, liayson (>= 1.0.0), rJava (>= 0.5-0), matlab (>= 1.0.2)

NeedsCompilation no

Repository CRAN

Date/Publication 2017-04-03 08:11:04

RoxygenNote 6.0.1

R topics documented:

compare *Comparing clone perspectives*

Description

Compares two clones with respect to user-specified perspectives.

Usage

```
compare(cloneID1,cloneID2,perspective1="GenomePerspective",perspective2="GenomePerspective", col
```

Arguments

cloneID1	The ID of the first clone (integer).
cloneID2	The ID of the second clone (integer).
perspective1	The perspective on the first clone: GenomePerspective or TranscriptomePerspective.
perspective2	The perspective on the second clone: GenomePerspective or TranscriptomePerspective.
col	The colors according to which to code clones.
plotType	Whether to make a scatter plot (plotType = "scatter") or heatmap (plotType = "heatmap").

Author(s)

Noemi Andor

Examples

```
  #par(mfrow=c(3,1))
  #display(cloneID_or_sampleName = "KATOIII",whichP = "GenomePerspective")
  #display(cloneID_or_sampleName = "KATOIII",whichP = "TranscriptomePerspective")
#compare(4,1,perspective1 = "GenomePerspective",perspective2 = "TranscriptomePerspective")
```

display *Visualization of a biosample's/clone's subclonal composition*

Description

Given the name of a biosample or the ID of a clone, the method displays its subclonal composition.

Usage

```
display(cloneID_or_sampleName,whichP="GenomePerspective", colorBy = NULL, deep = F, save = F)
```

Arguments

`cloneID_or_sampleName`
: Clone ID (integer) or biosample name (character).

`whichP`
: What populations to vizualize: GenomePerspective (default), TranscriptomePerspective or Identity.

`colorBy`
: What aspect of the populations vizualize: GenomePerspective>XX, TranscriptomePerspective>XX or Identity>XX, where XX is a specific locus within the chosen perspective and is optional.

`deep`
: Whether to display the direct subclones or go one level deeper and display the subclones' subclones.

`save`
: Whether to save the plot as .tif file.

Author(s)

Noemi Andor

`getPathThroughRegions` *Naive traveling salesman heuristic without return*

Description

Finds a minimum length Hamiltonian path through the sampled regions of a tumor, given their coordinates.

Usage

```
getPathThroughRegions(coord)
```

Arguments

`coord`
: Matrix with one row per sample and two columns containing the sample's x- and y-coordinates.

Author(s)

Noemi Andor

`getState` *Retrieving state associated with a clone*

Description

The state in which a clone was sequenced

Usage

```
getState(cloneID, whichP="TranscriptomePerspective"
```

Arguments

`cloneID` Clone ID (integer).

`whichP` What to vizualize: GenomePerspective (default), TranscriptomePerspective or Identity.

Value

A string describing subclones' state.

Author(s)

Noemi Andor

`getSubclones` *Retrieving subclones*

Description

Given the name of a biosample or the ID of a clone, the method retrieves all its subclones.

Usage

```
getSubclones(cloneID_or_sampleName,whichP="GenomePerspective")
```

Arguments

`cloneID_or_sampleName` Clone ID (integer) or biosample name (character).

`whichP` What to vizualize: GenomePerspective (default), TranscriptomePerspective or Identity.

Value

A map of each clone to its unique ID.

Author(s)

Noemi Andor

getSubProfiles *Retrieving subclone profiles*

Description

Given the name of a biosample or the ID of a clone, the method retrieve the profiles of all its subclones.

Usage

```
getSubProfiles(cloneID_or_sampleName,whichP="TranscriptomePerspective", includeRoot = FALSE)
```

Arguments

`cloneID_or_sampleName`
Clone ID (integer) or biosample name (character).

`whichP` What to vizualize: GenomePerspective (default), TranscriptomePerspective or Identity.

`includeRoot` Whether or not to include the parent clone's profile into the output.

Value

A matrix with rows corresponding to features and columns corresponding to subclones.

Author(s)

Noemi Andor

Examples

```
pm=getSubProfiles(cloneID_or_sampleName = "LGG2T1",whichP = "GenomePerspective")
```

hyper *Subpopulation relatedness assessment*

Description

Calculates probability that two subpopulations are related, modelling overlapping mutations as hypergeometric distribution.

Usage

```
hyper(p,r=NULL)
```

Arguments

`p` A numeric vector or matrix holding the mutation profile of one or multiple subpopulations (0 denotes absence; values >0 denote presence).

`r` The mutation profile of the other subpopulation. Can be NULL if p is a matrix.

Details

Let SP_P be a subpopulation within one perspective and SP_R , a subpopulation within another perspective of the same tumor (perspectives may be of same type). Further, let M_P, M_R be the set of SNVs assigned to SP_P and SP_R, while M_{PR} is the set of overlapping SNVs between SP_P and SP_R. We calculate how likely is it to observe at least | M_{PR} | common SNVs between SP_P and SP_R just by chance, by calculating: **a**) how likely it is to observe at least | M_{PR} | common SNVs in SP_P **b**) how likely it is to observe at least | M_{PR} | common SNVs in SP_R Both probabilities are modeled as Hypergeometric distributions: For (**b**), we draw x $\in M_R$ and each time x $\in$ MP we count the draw as success. Conversely, if x $\notin M_P$, the draw is considered a failure. For (**a**) we proceed in the same way, but reverse the role of SP_R with that of SP_P. The likelihood that SP_P is related to SP_R is calculated as the minimum among the two probabilities (**a,b**).

Value

Probability that SP_P is related to SP_R.

Author(s)

Noemi Andor

`merge` *Merging perspectives into clones' identities*

Description

Merges either 2 different perspectives on the clonal composition of the same specimen OR the same perspective on the clonal composition of 2 or more different specimens to approximate each clone's identity as the consensus across perspectives/specimens.

Usage

```
merge(perspectives, specimens, simM = "euclidean", t=-Inf)
```

Arguments

`perspectives` Minimal one and maximal two perspectives on a specimen. Two perspectives are required if only one specimen is provided.

`specimens` The name of the specimen(s). Exactly one specimen is required if more than one perspectives are provided. Two or more specimens are required if only one perspective is provided.

`simM` What similarity measure to use in order to match clonal components across perspectives/specimens. Options are: inverse of euclidean distance (euclidean - default), correlation coefficient (either pearson or spearman), mutation overlap significance as assessed by hypergeometric distribution (hyper).

`t` Minimum similarity threshold below which two subpopulations will no longer be merged.

Details

Let S_i be the set of subpopulations detected in sample i and $S := \bigcup S_i$ – the set of clones detected across all biopsies of a given patient. Further let L be the set of all non-private loci, in which an SNV is detected in at least two samples and $M_x \subset L$ the set of loci mutated in $x \in S_i$.

Next, subpopulations S are grouped into categories by hierarchical cluster analysis of their SNV profiles M_S, using a distance metric defined by the hypergeometric probability calculated above (agglomeration method: "single"). Subpopulations from distinct samples, falling within the same category (hypergeometric $P \geq t$) are considered different perspectives on the same clone.

Value

List with four fields:

`sp2clone`	Matrix with rows denoting clones and columns holding the different perspectives on a clone. Entries contain the size of each clone and its ID in the database. Last column contains the Identity of each clone calculated across the preceding columns.
`sp2clone_sim`	Matrix with rows denoting clones and columns holding the different perspectives on a clone. Entries contain a measure of how confidently a clone could be assigned to the clone from the preceding column.
`consdat`	The consensus profile of each clone.
`usedOrder`	The path taken through persepctives to match clones.

Author(s)

Noemi Andor

Examples

```
 #par(mfrow=c(4,1))
 #display(cloneID_or_sampleName = "KATOIII",whichP = "GenomePerspective")
 #display(cloneID_or_sampleName = "KATOIII",whichP = "TranscriptomePerspective")
#merge(perspectives=c("GenomePerspective", "TranscriptomePerspective"), specimens="KATOIII")
 ##compare(4,1,perspective1 = "GenomePerspective",perspective2 = "Identity")
```

`parseLOCUS` *Parsing locus string to numeric*

Description

Transforms locus string "chr:start-end" to three corresponding numeric values.

Usage

```
parseLOCUS(loci)
```

Arguments

`loci`	Vector of strings holding genomic coordinates.

Value

Numeric matrix with columns "chr", "startpos", "endpos".

Author(s)

Noemi Andor

`viewPerspective` *Biosample's subclonal composition inference*

Description

Calculates user-specified perspective of the subclonal composition of a biosample and saves it to the database.

Usage

```
viewPerspective(pathToSample, whichP, tissue = NULL, expandsD ="expands", liaysonD = "liayson", suf
```

Arguments

`pathToSample`	The path towards the perspective to be viewed (character). If no such path is found, cloneID will call the corresponding algorithm to create it.
`whichP`	What to calculate: GenomePerspective or TranscriptomePerspective.
`tissue`	The tissue type of the specimen.
`expandsD`	The output directory for the GenomePerspective.
`liaysonD`	The output directory for the TranscriptomePerspective.
`suffix`	The suffix of the file within the output directory, containing the desired perspective.
`xy`	Two-dimensional vector containing the geographic location of the specimen.

Author(s)

Noemi Andor

Index