Package 'dStruct'

February 25, 2019

Type Package

Title Identifying differentially reactive regions from RNA structurome profiling data

Version 0.1.0

Author Krishna Choudhary <krishna.choudhary@gladstone.ucsf.edu>

Maintainer Krishna Choudhary <krishna.choudhary@gladstone.ucsf.edu>, Sharon Aviran <saviran@ucdavis.edu>

Description More about what it does (maybe more than one line)
Use four spaces when indenting paragraphs within the Description.

Imports zoo,
ggplot2,
purrr,
reshape2,
parallel

License MIT

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

Suggests knitr,
rmarkdown

VignetteBuilder knitr

R topics documented:

calcDis *Calculates d score.*

Description

Calculates d score.

Usage

```
calcDis(x)
```

Arguments

x	A numeric vector or matrix.

Value

If input is a numeric vector, a number is returned. For a matrix, a numeric vector is returned.

dCombs *Assesses within-group or between-group variation.*

Description

Assesses within-group or between-group variation.

Usage

```
dCombs(rdf, combs)
```

Arguments

rdf	Data.frame of reactivities for each sample.
combs	Data.frame with each column containing groupings of samples.

Value

Nucleotide-wise d score.

dStruct *Performs de novo discovery of differentially reactive regions.*

Description

Performs de novo discovery of differentially reactive regions.

Usage

```
dStruct(rdf, reps_A, reps_B, batches = F, min_length = 11,
  check_signal_strength = T, check_nucs = T, check_quality = T,
  quality = "auto", evidence = 0, signal_strength = 0.1,
  within_combs = NULL, between_combs = NULL, ind_regions = T, gap = 1,
  get_FDR = T, proximity_assisted = F, proximity = 10,
  proximity_defined_length = 30)
```

Arguments

rdf	Dataframe of reactivities for each sample.
reps_A	Number of replicates of group A.
reps_B	Number of replicates of group B.
batches	Logical suggesting if replicates of group A and B were performed in batches and are labelled accordingly. If TRUE, a heterogeneous/homogeneous subset may not have multiple samples from the same batch.
min_length	Minimum length of constructed regions.
check_signal_strength	Logical, if TRUE, construction of regions must be based on nucleotides that have a minimum absolute value of reactivity.
check_nucs	Logical, if TRUE, constructed regions must have a minimum number of nucleotides participating in Wilcoxon signed rank test.
check_quality	Logical, if TRUE, check constructed regions for quality.
quality	Worst allowed quality for a region to be tested.
evidence	Minimum evidence of increase in variation from within-group comparisons to between-group comparisons for a region to be tested.
signal_strength	Threshold for minimum signal strength.
within_combs	Data.frame with each column containing groupings of replicates of groups A or B, which will be used to assess within-group variation.
between_combs	Dataframe with each column containing groupings of replicates of groups A and B, which will be used to assess between-group variation.
ind_regions	Logical, if TRUE, test each region found in the transcript separately.
gap	Integer. Join regions if they are separated by these many nucleotides.
get_FDR	Logical, if FALSE, FDR is not reported.
proximity_assisted	Logical, if TRUE, proximally located regions are tested together.

`proximity` Maximum distance between constructed regions for them to be considered proximal.

`proximity_defined_length` If performing a "proximity-assisted" test, minimum end-to-end length of a region to be tested.

Value

Constructs regions, reports p-values and FDR for them.

`dStruct.guided` *Performs guided discovery of differentially reactive regions.*

Description

Performs guided discovery of differentially reactive regions.

Usage

```
dStruct.guided(rdf, reps_A, reps_B, batches = F, within_combs = NULL,
  between_combs = NULL, check_quality = TRUE, quality = "auto",
  evidence = 0)
```

Arguments

`rdf`	Dataframe of reactivities for each sample. Each column must be labelled as A1, A2, ..., B1, B2, ...
`reps_A`	Number of replicates of group A.
`reps_B`	Number of replicates of group B.
`batches`	Logical suggesting if replicates of group A and B were performed in batches and are labelled accordingly. If TRUE, a heterogeneous/homogeneous subset may not have multiple samples from the same batch.
`within_combs`	Data.frame with each column containing groupings of replicates of groups A or B, which will be used to assess within-group variation.
`between_combs`	Dataframe with each column containing groupings of replicates of groups A and B, which will be used to assess between-group variation.
`check_quality`	Logical, if TRUE, check regions for quality.
`quality`	Worst allowed quality for a region to be tested.
`evidence`	Minimum evidence of increase in variation from within-group comparisons to between-group comparisons for a region to be tested.

Value

p-value for the tested region, estimated using one-sided Wilcoxon signed rank test.

dStructome *Performs de novo discovery of differentially reactive regions for transcriptome-wide data.*

Description

Performs de novo discovery of differentially reactive regions for transcriptome-wide data.

Usage

```
dStructome(rl, reps_A, reps_B, batches = F, min_length = 11,
  check_signal_strength = T, check_nucs = T, check_quality = T,
  quality = "auto", evidence = 0, signal_strength = 0.1,
  within_combs = NULL, between_combs = NULL, ind_regions = T, gap = 1,
  processes = "auto", method = "denovo", proximity_assisted = F,
  proximity = 10, proximity_defined_length = 30)
```

Arguments

rl	List of dataframes of reactivities for each sample.
reps_A	Number of replicates of group A.
reps_B	Number of replicates of group B.
batches	Logical suggesting if replicates of group A and B were performed in batches and are labelled accordingly. If TRUE, a heterogeneous/homogeneous subset may not have multiple samples from the same batch.
min_length	Minimum length of constructed regions.
check_signal_strength	Logical, if TRUE, construction of regions must be based on nucleotides that have a minimum absolute value of reactivity.
check_nucs	Logical, if TRUE, constructed regions must have a minimum number of nucleotides participating in Wilcoxon signed rank test.
check_quality	Logical, if TRUE, check constructed regions for quality.
quality	Worst allowed quality for a region to be tested.
evidence	Minimum evidence of increase in variation from within-group comparisons to between-group comparisons for a region to be tested.
signal_strength	Threshold for minimum signal strength.
within_combs	Data.frame with each column containing groupings of replicates of groups A or B, which will be used to assess within-group variation.
between_combs	Dataframe with each column containing groupings of replicates of groups A and B, which will be used to assess between-group variation.
ind_regions	Logical, if TRUE, test each region found in the transcript separately.
gap	Integer. Join regions if they are separated by these many nucleotides.
processes	Number of parallel processes to use.
method	Character specifying either guided or de novo discovery approach.

`proximity_assisted`
: Logical, if TRUE, proximally located regions are tested together.

`proximity`
: Maximum distance between constructed regions for them to be considered proximal.

`proximity_defined_length`
: If performing a "proximity-assisted" test, minimum end-to-end length of a region to be tested.

Value

Constructs regions, reports p-values and FDR for them.

`getCombs` *Identifies subgroupings of replicates for assessing within-group and between-group variation.*

Description

Identifies subgroupings of replicates for assessing within-group and between-group variation.

Usage

```
getCombs(reps_A, reps_B, batches = F, between_combs = NULL,
  within_combs = NULL)
```

Arguments

`reps_A`
: Number of replicates of group A.

`reps_B`
: Number of replicates of group B.

`batches`
: Logical suggesting if replicates of group A and B were performed in batches and are labelled accordingly. If TRUE, a heterogeneous/homogeneous subset may not have multiple samples from the same batch.

`between_combs`
: Dataframe with each column containing groupings of replicates of groups A and B, which will be used to assess between-group variation.

`within_combs`
: Data.frame with each column containing groupings of replicates of groups A or B, which will be used to assess within-group variation.

Value

List of two dataframes, containing groupings for within-group and between-group variation.

getContigRegions *Identifies contiguous regions from a list of nucleotide indices.*

Description

Identifies contiguous regions from a list of nucleotide indices.

Usage

```
getContigRegions(x, gap = 0)
```

Arguments

x	A vector of integers.
gap	Allowed gap to merge regions.

Value

Dataframe storing start and stop sites of continguous regions.

getRegions *Constructs potential differentially reactive regions.*

Description

Constructs potential differentially reactive regions.

Usage

```
getRegions(d_within, d_spec, rdf, min_length = 11,
  check_signal_strength = T, check_nucs = T, check_quality = T,
  quality = 0.5, evidence = 0, signal_strength = 0.1)
```

Arguments

d_within	Nucleotide-wise d score for within-group variation.
d_spec	Nucleotide-wise d score for between-group variation.
rdf	Dataframe of reactivities for each sample.
min_length	Minimum length of constructed regions.
check_signal_strength	Logical, if TRUE, construction of regions must be based on nucleotides that have a minimum absolute value of reactivity.
check_nucs	Logical, if TRUE, constructed regions must have a minimum number of nucleotides participating in Wilcoxon signed rank test.
check_quality	Logical, if TRUE, check constructed regions for quality.
quality	Worst allowed quality for a region to be tested.
evidence	Minimum evidence of increase in variation from within-group comparisons to between-group comparisons for a region to be tested.
signal_strength	Threshold for minimum signal strength.

normalizer *Returns normalizer for reactivity vector.*

Description

Assesses normalization factor for raw reactivities using 2-8 % method.

Usage

```
normalizer(raw.estimates)
```

Arguments

raw.estimates	A vector of raw reactivities.

Value

The normalization factor.

plot_dStructurome *Plots differentially reactive regions.*

Description

Plots differentially reactive regions.

Usage

```
plot_dStructurome(rl, diff_regions, outfile, fdr = 0.05, ylim = c(-0.05, 3))
```

Arguments

rl	List of dataframes of reactivities for each sample.
diff_regions	Dataframe of regions with significance of differentially reactivity.
outfile	The name for pdf file which will be saved.
fdr	FDR threshold for plotted regions.
ylim	Y-axis limits for plots.

Value

Saves a PDF for all differentially reactive regions. Returns NULL.

`two.eight.normalize` *Normalizes reactivity vector.*

Description

Normalizes raw reactivities using 2-8 % method.

Usage

```
two.eight.normalize(raw.estimates)
```

Arguments

`raw.estimates`	A vector of raw reactivities.

Value

A vector of normalized reactivities.

Index