

Package ‘dStruct’

June 5, 2018

Type Package

Title Identifying differentially reactive regions from RNA structurome profiling data

Version 0.1.0

Author Krishna Choudhary <kchoudhary@ucdavis.edu>

Maintainer Krishna Choudhary <kchoudhary@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	2
dCombs	2
dStruct	3
dStruct.guided	4
dStructome	4
getCombs	5
getContigRegions	6
getRegions	6
normalizer	7
plot_dStructurome	7
two.eight.normalize	8
Index	9

calcDis	<i>Calculates d score.</i>
---------	----------------------------

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	<i>Assesses within-group variation.</i>
--------	---

Description

Assesses within-group variation.

Usage

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

Arguments

rdf	Dataframe 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)
```

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.
min_length	Minimum length of constructed regions.
check_signal_strength	Logical, if TRUE, constructed regions must have a minimum signal strength.
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 between-group variation 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.

Value

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

dStruct.guided	<i>Performs guided discovery of differentially reactive regions.</i>
----------------	--

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.
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.
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 between-group variation for a region to be tested.

Value

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

dStructome	<i>Performs de novo discovery of differentially reactive regions for transcriptome-wide data.</i>
------------	---

Description

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

Usage

```
dStructome(r1, 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")
```

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.
min_length	Minimum length of constructed regions.
check_signal_strength	Logical, if TRUE, constructed regions must have a minimum signal strength.
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 between-group variation 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.

Value

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

getCombs	<i>Identifies subgroupings of replicates for assessing within-group and between-group variation.</i>
----------	--

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.
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	<i>Performs guided discovery of differentially reactive regions.</i>
------------------	--

Description

Performs guided discovery of differentially reactive regions.

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 contiguous regions.

getRegions	<i>Constructs potential differentially reactive regions.</i>
------------	--

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, constructed regions must have a minimum signal strength.
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 between-group variation for a region to be tested.
signal_strength	Threshold for minimum signal strength.

normalizer	<i>Returns normalizer for reactivity vector.</i>
------------	--

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	<i>Plots differentially reactive regions.</i>
-------------------	---

Description

Plots differentially reactive regions.

Usage

```
plot_dStructurome(r1, diff_regions, outfile, fdr = 0.05, ylim = c(-0.05, 3))
```

Arguments

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

Value

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

<code>two.eight.normalize</code>	<i>Normalizes reactivity vector.</i>
----------------------------------	--------------------------------------

Description

Normalizes raw reactivities using 2-8 % method.

Usage

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

Arguments

<code>raw.estimates</code>	A vector of raw reactivities.
----------------------------	-------------------------------

Value

A vector of normalized reactivities.

Index

`calcDis`, [2](#)

`dCombs`, [2](#)

`dStruct`, [3](#)

`dStruct.guided`, [4](#)

`dStructome`, [4](#)

`getCombs`, [5](#)

`getContigRegions`, [6](#)

`getRegions`, [6](#)

`normalizer`, [7](#)

`plot_dStructurome`, [7](#)

`two.eight.normalize`, [8](#)