Package 'rifiComparative'

October 16, 2023

Title 'rifiComparative' compares the outputs of 'rifi' under two different conditions.

Version 1.0.1

Description 'rifiComparative' is an extension of the 'rifi' package. It is designed to compare the outputs of 'rifi' under two different conditions by utilizing the half-life and mRNA at time 0 segments. To perform the segmentation, it takes into account the difference in half-life between the two conditions and the log2FC (fold change) of the mRNA at time 0. This package offers various functionalities such as segmentation, statistical analysis, summary tables, visualization of fragments, and additional plots that can be helpful for further analysis.

Depends R (>= 4.2)

Imports cowplot, doMC, parallel, dplyr, egg, foreach, ggplot2, ggrepel, graphics, grDevices, grid, methods, nnet, rlang, S4Vectors, scales, stats, stringr, tibble, rtracklayer, utils, writexl, DTA, LSD, reshape2, devtools, SummarizedExperiment

Suggests DescTools, devtools, knitr, rmarkdown, BiocStyle

VignetteBuilder knitr

biocViews RNASeq, DifferentialExpression, GeneRegulation, Transcriptomics, Microarray, Software

BugReports https://github.com/CyanolabFreiburg/rifiComparative

License GPL-3 + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Language en-US

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

git_branch RELEASE_3_17

git_last_commit fd7c64c

git_last_commit_date 2023-06-04

Date/Publication 2023-10-15

Author Loubna Youssar [aut, cre],
Jens cre Georg [aut]

Maintainer Loubna Youssar <lyoussar@gmail.com>

R topics documented:

`adjusting_HLToInt` *adjusting_HLToInt Creates one table merging HL and intensity fragments with genome annotation*

Description

'adjusting_HLToInt' merges HL and intensity segments adapting the positions to each other and combining to the genome annotation. To make HL and intensity segments comparable, log2FC(HL) is used to generate the data frame instead of distance. The fragments should have a significant p_value from t-test at least from one segmentation, either HL or intensity.

Usage

```
adjusting_HLToInt(data, Strand = c("+", "-"), annotation)
```

Arguments

data	data frame: data frame combined data by column
Strand	string: either "+" or "-"
annotation	data frame: data frame from processed gff3 file.

Details

The functions used are:

1. p_value_function extracts and return the p_values of HL and intensity segments respectively.
2. eliminate_outlier_hl eliminates outliers from HL fragments.
3. eliminate_outlier_int eliminates outliers from intensity fragments.
4. mean_length_int calculates the mean of the log2FC(intensity) fragments adapted to HL_fragments and their lengths
5. mean_length_hl calculates the mean of log2FC(HL) fragments adapted to the intensity fragments and their lengths.
6. calculating_rate calculates decay rate and log2FC(intensity). Both are used to calculate synthesis rate.

Value

The data frame with the corresponding columns:

position: Integer, position of the first fragment

region: String, region annotation covering the fragments

gene: String, gene annotation covering the fragments

locus_tag: String, locus_tag annotation covering the fragments

strand: Boolean. The bin/probe specific strand (+/-)

fragment_HL: String, HL fragments

fragment_int: String, intensity fragments

position_frg_int: Integer, position of the first fragment and the last position of the last fragment

mean_HL_fragment: Integer, mean of the HL of the fragments involved

mean_int_fragment: Integer, mean of the intensity of the fragments involved

log2FC(decay_rate): Integer, log2FC(decay(condition1)/ decay(condition2))

log2FC(synthesis_rate): Integer, sum of log2FC(decay_rate) and log2FC(intensity)

intensity_FC: Integer, log2FC(mean(intensity(condition1))/mean(intensity(condition2)))

Log2FC(HL)+Log2FC(int): Integer, sum of log2FC(decay_rate) and log2FC(intensity)

p_value: String, indicated by "*" means at least one fragment either HL fragment or intensity fragment has a significant p_value

Examples

```
data(stats_df_comb_minimal)
data(annot_g)
df_mean_minimal <- adjusting_HLToInt(data = stats_df_comb_minimal,
annotation = annot_g[[1]])
```

---

annot_g *The result of gff3_preprocessing of gff3 file A list containing all necessary information from a gff file for adjusting_HLToInt and visualization.*

Description

The result of gff3_preprocessing of gff3 file A list containing all necessary information from a gff file for adjusting_HLToInt and visualization.

Usage

```
data(annot_g)
```

Format

A list with 2 items:

data annotation: a data frame with 5853 rows and 6 variables

- **region:** the region from the gff file
- **start:** the start of the annotation
- **end:** the end of the annotation
- **strand:** the strand of the annotation
- **gene:** the annotated gene name
- **locus_tag:** the annotated locus tag

genome length: a numeric vector containing the length of the genome

Source

https://github.com/CyanolabFreiburg/rifiComparative

`data_combined_minimal` *The result of joining_by_row for inp_s and inp_f example data A data frame containing the output of joining_by_row as a data frame*

Description

The result of joining_by_row for inp_s and inp_f example data A data frame containing the output of joining_by_row as a data frame

Usage

```
data(data_combined_minimal)
```

Format

A data frame with 600 rows and 49 variables:

strand: The strand specific

position: The bin/probe specific position

ID: The bin/probe specific ID

FLT: The bin/probe flag for background level

intensity: The relative intensity at time point 0

probe_TI: An internal value to determine which fitting model is applied

flag: Information on which fitting model is applied

position_segment: The position based segment

delay: The delay value of the bin/probe

half_life: The half-life of the bin/probe

TI_termination_factor: The termination factor of the bin/probe

delay_fragment: The delay fragment the bin belongs to

velocity_fragment: The velocity value of the respective delay fragment

intercept: The vintercept of fit through the respective delay fragment

slope: The slope of the fit through the respective delay fragment

HL_fragment: The half-life fragment the bin belongs to

HL_mean_fragment: The mean half-life value of the respective half-life fragment

intensity_fragment: The intensity fragment the bin belongs to

intensity_mean_fragment: The mean intensity value of the respective intensity fragment

TU: The overarching transcription unit

TI_termination_fragment: The TI fragment the bin belongs to

TI_mean_termination_factor: The mean termination factor of the respective TI fragment

seg_ID: The combined ID of the fragment

pausing_site: presence of pausing site indicated by +/-

iTSS_I: presence of iTSS_I indicated by +/-

ps_ts_fragment: The fragments involved in pausing site or iTSS_I

event_ps_itss_p_value_Ttest: p_value of pausing site or iTSS_I

p_value_slope: p_value of the slope

delay_frg_slope: the slope value of the respective delay fragment

velocity_ratio: Integer, ratio of velocity between 2 delay fragments

event_duration: Integer, the duration between two delay fragments

event_position: Integer, the position middle between 2 fragments with an event

FC_fragment_HL: Integer, the fold change value of 2 intensity fragments

FC_HL: Integer, the fold change value of 2 HL fragments#'

p_value_HL: p_value of the fold change of HL fragments

FC_intensity: Integer, the fold change value of 2 intensity fragments

FC_fragment_intensity: String, fragments involved in fold change between 2 intensity fragments

p_value_intensity: p_value of the fold change of intensity fragments

FC_HL_intensity: ratio of fold change between 2 half-life fragments and fold change between 2 intensity fragments

FC_HL_intensity_fragment: fragments involved on ratio of fold change between 2 half-life fragments and fold change between 2 intensity fragments

FC_HL_adapted: Integer, the fold change of half-life/ fold change of intensity, position of the half-life fragment is adapted to intensity fragment

synthesis_ratio: Integer, the value correspomding to synthesis rate

synthesis_ratio_event: String, the event assigned by synthesis rate either Termination or iTSS

p_value_Manova: p_value of the variance between two fold-changes, HL and intensity

p_value_TI: p_value of TI fragment

TI_fragments_p_value: p_value of 2 TI fragments

cdt: The condition assigned to the experiment here cdt2

logFC_int: The bin/probe log2 fold change of intensity at time 0

P.Value: The bin/probe p_value adjusted

Source

https://github.com/CyanolabFreiburg/rifi

`df_comb_minimal` *The result of joining_by_column for data_combined_minimal example data A data frame containing the output of joining_by_row as a data frame*

Description

The result of joining_by_column for data_combined_minimal example data A data frame containing the output of joining_by_row as a data frame

Usage

```
data(df_comb_minimal)
```

Format

A data frame with 300 rows and 18 variables:

strand: The strand specific

position: The bin/probe specific position

ID: The bin/probe specific ID

intensity.cdt1: The relative intensity at time point 0 for condition 1

position_segment: The position based segment

half_life.cdt1: The half-life of the bin/probe condition 1

TI_termination_factor.cdt1: The termination factor of the bin/probe condition 1

HL_fragment.cdt1: The half-life fragment the bin belongs to condition 1

intensity_fragment.cdt1: The intensity fragment the bin belongs to condition 1

TI_termination_fragment.cdt1: The TI fragment the bin belongs to condition 1

logFC_int: The bin/probe log2 fold change of intensity at time 0

P.Value: The bin/probe p_value adjusted

intensity.cdt2: The relative intensity at time point 0 condition 2

half_life.cdt2: The half-life of the bin/probe condition 2

TI_termination_factor.cdt2: The termination factor of the bin/probe condition 2

HL_fragment.cdt2: The half-life fragment the bin belongs to condition 2

intensity_fragment.cdt2: The intensity fragment the bin belongs to condition 2

TI_termination_fragment.cdt2: The TI fragment the bin belongs to condition 2

Source

https://github.com/CyanolabFreiburg/rifiComparative

`df_mean_minimal` *The result of adjusting_HLToInt for stats_df_comb_minimal and annotation example data A data frame containing the output of adjusting_HLToInt as a data frame*

Description

The result of adjusting_HLToInt for stats_df_comb_minimal and annotation example data A data frame containing the output of adjusting_HLToInt as a data frame

Usage

```
data(df_mean_minimal)
```

Format

A data frame with 52 rows and 15 variables:

position: The bin/probe specific position

region: the region from the gff file

gene: the annotated gene name

locus_tag: the annotated locus tag

strand: The strand specific

fragment_HL: The half-life fragment the bin belongs

fragment_int: The intensity fragment the bin belongs

position_frg_int: The position of the first fragment and the last position of the last fragment

mean_HL_fragment: The mean half-life value of the respective half-life fragments

mean_int_fragment: The mean intensity value of the respective intensity fragments

log2FC(decay_rate): log2FC(decay(condition1)/decay(condition2))

Log2FC(HL)-Log2FC(int): log2FC(decay_rate/intensity)

log2FC(synthesis_rate): log2FC(decay_rate) + log2FC(intensity)

intensity_FC: log2FC(mean(intensity(condition1))/mean(intensity(condition2)))

p_value: indicated by "*" means at least one fragment either HL fragment or intensity fragment has a significant p_value

Source

https://github.com/CyanolabFreiburg/rifiComparative

`differential_expression` *An example data frame from Synechosystis PCC 6803 differential probes expression obtained from limma package and only interesting variables were selected. The data frame was used entirely.*

Description

An example data frame from Synechosystis PCC 6803 differential probes expression obtained from limma package and only interesting variables were selected. The data frame was used entirely.

Usage

```
data(differential_expression)
```

Format

A data frame of differential_expression with 55508 rows and 4 variables:

position: The bin/probe specific position

strand: The strand specific

logFC_int: The bin/probe differential expression

P.Value: The bin/probe p_value adjusted

Source

https://github.com/CyanolabFreiburg/rifiComparative

`figures_fun` *'figures_fun': generates several plots*

Description

'figures_fun' plots at one the density of HL, the HL category as histogram, log2FC of decay and synthesis rate and their heatscatter. Scatter plot of HL and volcano plot. The function uses the four output generated previously.

Usage

```
figures_fun(
  data.1,
  data.2,
  input.1,
  input.2,
  cdt1,
  cdt2,
  y = 30,
  x = 30,
  limits = c(0, 20)
)
```

Arguments

data.1	data frame output of statistic
data.2	data frame joining two outputs from rifi_stats by row
input.1	data frame joining two outputs from rifi_stats by column
input.2	data frame of differential gene expression at time 0
cdt1	string for the first condition
cdt2	string for the second condition
y	integer to break the scaling in scatter plot for y_axis
x	integer to break the scaling in scatter plot for x_axis
limits	vector to limit the scaling in scatter plot for both axis

Details

The functions used are:

plot_decay_synt: plots the changes in RNA decay rates versus the changes in RNA synthesis rates

plot_heatscatter: plots the changes in RNA decay rates versus the changes in RNA synthesis rates with density.

plot_volcano: plots statistical significance versus magnitude of change .

plot_histogram: plot a histogram of probe/bin half-life categories from 2 to 20 minutes in both conditions.

plot_density: plots the probe/bin half-life density in both conditions.

plot_scatter: plots of the bin/probe half-life in one condition1 vs. condition2.

extract the object of rifi_statistics from both conditions. The differential gene expression at time 0 needs to be run separately. The columns log2FC, p_value adjusted, position and strand are extracted and saved to a data frame. loading_fun_fig joins the differential gene expression table and the output from rifi statistics into one data frame.

Value

several plots

Examples

```
data(data_combined_minimal)
data(df_comb_minimal)
data(differential_expression)
data(df_mean_minimal)
figures_fun(data.1 = df_mean_minimal, data.2 = data_combined_minimal,
input.1 = df_comb_minimal, input.2 = differential_expression, cdt1 = "sc",
cdt2 = "fe")
```

`fragmentation` *fragmentation: Conveniently wraps all fragmentation steps*

Description

fragmentation fragments the half-life and intensity into segments using the penalties output.

Usage

```
fragmentation(data, pen_HL, pen_int, cores = 2)
```

Arguments

`data`	data frame: data frame combined data by column
`pen_HL`	list: list of the penalties set optimal for the fragmentation for half-life
`pen_int`	list: list of the penalties set optimal for the fragmentation for intensity
`cores`	integer: the number of assigned cores for the task. It needs to be increased in case of big data.

Value

Two data frames with half-life and intensity fragments and the mean of the coefficient fragment based.

Examples

```
data(penalties_df)
data(pen_HL)
data(pen_int)
df_comb_minimal <- fragmentation(data = penalties_df, pen_HL,
pen_int)
```

`fragment_int` *The result of fragmentation for df_comb_minimal example data A data frame containing the output of fragmentation as a data frame*

Description

The result of fragmentation for df_comb_minimal example data A data frame containing the output of fragmentation as a data frame

Usage

```
data(fragment_int)
```

Format

A data frame with 500 rows and 24 variables:

strand: The strand specific

position: The bin/probe specific position

ID: The bin/probe specific ID

intensity.cdt1: The relative intensity at time point 0 for condition 1

position_segment: The position based segment

half_life.cdt1: The half-life of the bin/probe condition 1

TI_termination_factor.cdt1: The termination factor of the bin/probe condition 1

HL_fragment.cdt1: The half-life fragment the bin belongs to condition 1

intensity_fragment.cdt1: The intensity fragment the bin belongs to condition 1

TI_termination_fragment.cdt1: The TI fragment the bin belongs to condition 1

logFC_int: The bin/probe log2 fold change of intensity at time 0

P.Value: The bin/probe p_value adjusted

intensity.cdt2: The relative intensity at time point 0 condition 2

half_life.cdt2: The half-life of the bin/probe condition 2

TI_termination_factor.cdt2: The termination factor of the bin/probe condition 2

HL_fragment.cdt2: The half-life fragment the bin belongs to condition 2

intensity_fragment.cdt2: The intensity fragment the bin belongs to condition 2

TI_termination_fragment.cdt2: The TI fragment the bin belongs to condition 2

distance_HL: The bin/probe difference of half-life from both conditions

distance_int: The bin/probe log2 fold change of intensity at time 0

HL_comb_fragment: The half-life fragment the bin belongs to both conditions

HL_mean_comb_fragment: The half-life mean of the fragment the bin belongs to both conditions

intensity_comb_fragment: The intensity fragment the bin belongs to both conditions

intensity_mean_comb_fragment: The intensity mean of the fragment the bin belongs to both conditions

Source

https://github.com/CyanolabFreiburg/rifiComparative

`gff3_preprocess` *gff3_preprocess processes gff3 file from database*

Description

gff3_preprocess processes the gff3 file extracting gene names and locus_tag from all coding regions (CDS). UTRs/ncRNA/asRNA if available, are also extracted. The resulting dataframe contains region, positions, strand, gene and locus_tag.

Usage

```
gff3_preprocess(path)
```

Arguments

`path`	path: path to the directory containing the gff3 file.

Value

A list with 2 items:

data annotation: **region:** String, the region from the gff file

start: Integer, the start of the annotation

end: Integer, the end of the annotation

strand: Boolean, the strand of the annotation

gene: String, the annotated gene name

locus_tag: String, the annotated locus tag

genome length: a numeric vector containing the length of the genome

Examples

```
gff3_preprocess(
path = gzfile(system.file("extdata", "gff_synechocystis_6803.gff.gz",
package = "rifiComparative"))
)
```

inp_f *The result of loading_fun for stats_se_cdt2 example data Two data frame containing the output of loading_fun as second element of a list.*

Description

The result of loading_fun for stats_se_cdt2 example data Two data frame containing the output of loading_fun as second element of a list.

Usage

```
data(inp_f)
```

Format

A data frame with 500 rows and 49 variables:

strand: The strand specific

position: The bin/probe specific position

ID: The bin/probe specific ID

FLT: The bin/probe flag for background level

intensity: The relative intensity at time point 0

probe_TI: An internal value to determine which fitting model is applied

flag: Information on which fitting model is applied

position_segment: The position based segment

delay: The delay value of the bin/probe

half_life: The half-life of the bin/probe

TI_termination_factor: The termination factor of the bin/probe

delay_fragment: The delay fragment the bin belongs to

velocity_fragment: The velocity value of the respective delay fragment

intercept: The vintercept of fit through the respective delay fragment

slope: The slope of the fit through the respective delay fragment

HL_fragment: The half-life fragment the bin belongs to

HL_mean_fragment: The mean half-life value of the respective half-life fragment

intensity_fragment: The intensity fragment the bin belongs to

intensity_mean_fragment: The mean intensity value of the respective intensity fragment

TU: The overarching transcription unit

TI_termination_fragment: The TI fragment the bin belongs to

TI_mean_termination_factor: The mean termination factor of the respective TI fragment

seg_ID: The combined ID of the fragment

pausing_site: presence of pausing site indicated by +/-

iTSS_I: presence of iTSS_I indicated by +/-

ps_ts_fragment: The fragments involved in pausing site or iTSS_I

event_ps_itss_p_value_Ttest: p_value of pausing site or iTSS_I

p_value_slope: p_value of the slope

delay_frg_slope: the slope value of the respective delay fragment

velocity_ratio: Integer, ratio of velocity between 2 delay fragments

event_duration: Integer, the duration between two delay fragments

event_position: Integer, the position middle between 2 fragments with an event

FC_HL: Integer, the fold change value of 2 HL fragments

FC_fragment_HL: Integer, the fold change value of 2 intensity fragments

p_value_HL: p_value of the fold change of HL fragments

FC_intensity: Integer, the fold change value of 2 intensity fragments

FC_fragment_intensity: String, fragments involved in fold change between 2 intensity fragments

p_value_intensity: p_value of the fold change of intensity fragments

FC_HL_intensity: ratio of fold change between 2 half-life fragments and fold change between 2 intensity fragments

FC_HL_intensity_fragment: fragments involved on ratio of fold change between 2 half-life fragments and fold change between 2 intensity fragments

FC_HL_adapted: Integer, the fold change of half-life/ fold change of intensity, position of the half-life fragment is adapted to intensity fragment

synthesis_ratio: Integer, the value correspomding to synthesis rate

synthesis_ratio_event: String, the event assigned by synthesis rate either Termination or iTSS

p_value_Manova: p_value of the variance between two fold-changes, HL and intensity

p_value_TI: p_value of TI fragment

TI_fragments_p_value: p_value of 2 TI fragments

cdt: The condition assigned to the experiment here cdt2

logFC_int: The bin/probe log2 fold change of intensity at time 0

P.Value: The bin/probe p_value adjusted

Source

https://github.com/CyanolabFreiburg/rifiComparative

inp_s *The result of loading_fun for stats_se_cdt1 example data Two data frame containing the output of loading_fun as first element of a list.*

Description

The result of loading_fun for stats_se_cdt1 example data Two data frame containing the output of loading_fun as first element of a list.

Usage

```
data(inp_s)
```

Format

A data frame with 500 rows and 49 variables:

strand: The strand specific

position: The bin/probe specific position

ID: The bin/probe specific ID

FLT: The bin/probe flag for background level

intensity: The relative intensity at time point 0

probe_TI: An internal value to determine which fitting model is applied

flag: Information on which fitting model is applied

position_segment: The position based segment

delay: The delay value of the bin/probe

half_life: The half-life of the bin/probe

TI_termination_factor: The termination factor of the bin/probe

delay_fragment: The delay fragment the bin belongs to

velocity_fragment: The velocity value of the respective delay fragment

intercept: The vintercept of fit through the respective delay fragment

slope: The slope of the fit through the respective delay fragment

HL_fragment: The half-life fragment the bin belongs to

HL_mean_fragment: The mean half-life value of the respective half-life fragment

intensity_fragment: The intensity fragment the bin belongs to

intensity_mean_fragment: The mean intensity value of the respective intensity fragment

TU: The overarching transcription unit

TI_termination_fragment: The TI fragment the bin belongs to

TI_mean_termination_factor: The mean termination factor of the respective TI fragment

seg_ID: The combined ID of the fragment

pausing_site: presence of pausing site indicated by +/-

iTSS_I: presence of iTSS_I indicated by +/-

ps_ts_fragment: The fragments involved in pausing site or iTSS_I

event_ps_itss_p_value_Ttest: p_value of pausing site or iTSS_I

p_value_slope: p_value of the slope

delay_frg_slope: the slope value of the respective delay fragment

velocity_ratio: Integer, ratio of velocity between 2 delay fragments

event_duration: Integer, the duration between two delay fragments

event_position: Integer, the position middle between 2 fragments with an event

FC_HL: Integer, the fold change value of 2 HL fragments

FC_fragment_HL: Integer, the fold change value of 2 intensity fragments

p_value_HL: p_value of the fold change of HL fragments

FC_intensity: Integer, the fold change value of 2 intensity fragments

FC_fragment_intensity: String, fragments involved in fold change between 2 intensity fragments

p_value_intensity: p_value of the fold change of intensity fragments

FC_HL_intensity: ratio of fold change between 2 half-life fragments and fold change between 2 intensity fragments

FC_HL_intensity_fragment: fragments involved on ratio of fold change between 2 half-life fragments and fold change between 2 intensity fragments

FC_HL_adapted: Integer, the fold change of half-life/ fold change of intensity, position of the half-life fragment is adapted to intensity fragment

synthesis_ratio: Integer, the value correspomding to synthesis rate

synthesis_ratio_event: String, the event assigned by synthesis rate either Termination or iTSS

p_value_Manova: p_value of the variance between two fold-changes, HL and intensity

p_value_TI: p_value of TI fragment

TI_fragments_p_value: p_value of 2 TI fragments

cdt: The condition assigned to the experiment here cdt1

logFC_int: The bin/probe log2 fold change of intensity at time 0

P.Value: The bin/probe p_value adjusted

Source

https://github.com/CyanolabFreiburg/rifiComparative

joining_data_column *joining_data_column joins two data frames by column*

Description

'joining_data_column': joins two data frames from different conditions by column.

Usage

```
joining_data_column(data)
```

Arguments

data	data frame with joined columns from both conditions

Value

The data frame with joined columns from both conditions with the corresponding columns: strand, position, ID, intensity.cdt1, position_segment, half_life.cdt1, TI_termination_factor.cdt1", HL_fragment.cdt1, intensity_fragment.cdt1, TI_termination_fragment.cdt1, logFC_int, P.Value, intensity.cdt2, half_life.cdt2, TI_termination_factor.cdt2, HL_fragment.cdt2, intensity_fragment.cdt2, TI_termination_fragment.cdt2.

cdt1: first condition, cdt2: second condition.

Examples

```
data(data_combined_minimal)
df_comb_minimal <- joining_data_column(data = data_combined_minimal)
```

joining_data_row *joining_data_row joins two data frames by row*

Description

joining_data_row joins two data frames from different conditions by row.

Usage

```
joining_data_row(input1, input2)
```

Arguments

input1	data frame from SummarizedExperiment output of rifi_stats from rifi package of the first condition.
input2	data frame from SummarizedExperiment output of rifi_stats from rifi package of the second condition.

Value

The data frame with joined rows from both conditions with the corresponding columns: ID with position, strand, intensity, probe_TI, flag, position_segment, delay, half_life, TI_termination_factor, delay_fragment, velocity_fragment, intercept, slope, HL_fragment, HL_mean_fragment, intensity_fragment, intensity_mean_fragment, TU, TI_termination_fragment, TI_mean_termination_factor, seg_ID, pausing_site, iTSS_I, ps_ts_fragment, event_ps_itss_p_value_Ttest, p_value_slope, delay_frg_slope, velocity_ratio, event_duration, event_position, FC_HL, FC_fragment_HL, p_value_HL, FC_intensity, FC_fragment_intensity, p_value_intensity, FC_HL_intensity, FC_HL_intensity_fragment, FC_HL_adapted, synthesis_ratio, synthesis_ratio_event, p_value_Manova, p_value_TI, cdt (condition), logFC_int (log2FC(intensity)), P.Value of log2FC(intensity)

Examples

```
data(inp_s)
data(inp_f)
data_combined_minimal <-
joining_data_row(input1 = inp_s, input2 = inp_f)
```

---

loading_fun *loading_fun loads the data of all conditions*

Description

loading_fun extract outputs from rifi_stats of all conditions and join each data to the differential expression table. The differential gene expression at time 0 needs to be run separately. The columns log2FC, p_value adjusted, position and strand are extracted and saved to a data frame. loading_fun joins the differential gene expression table and the output from rifi statistics into one data frame.

Usage

```
loading_fun(data1, data2, data3)
```

Arguments

data1	data frame from rifi_stats of one condition
data2	data frame from rifi_stats of other condition
data3	data frame from differential gene expression at time 0

Value

A list of two data frames with joined columns from differential expression and output of rifi_stats with the corresponding columns: ID with position, strand, intensity, probe_TI, flag, position_segment, delay, half_life, TI_termination_factor, delay_fragment, velocity_fragment, intercept, slope, HL_fragment, HL_mean_fragment, intensity_fragment, intensity_mean_fragment, TU, TI_termination_fragment, TI_mean_termination_factor, seg_ID, pausing_site, iTSS_I, ps_ts_fragment, event_ps_itss_p_value_Ttest, p_value_slope, delay_frg_slope, velocity_ratio, event_duration, event_position, FC_HL, FC_fragment_HL, p_value_HL, FC_intensity, FC_fragment_intensity, p_value_intensity, FC_HL_intensity, FC_HL_intensity_fragment,

FC_HL_adapted, synthesis_ratio, synthesis_ratio_event, p_value_Manova, p_value_TI, cdt (condition), logFC_int (log2FC(intensity)), P.Value of log2FC(intensity).

Examples

```
data(stats_se_cdt1)
data(stats_se_cdt2)
data(differential_expression)
inp_s <-
loading_fun(stats_se_cdt1, stats_se_cdt2, differential_expression)[[1]]
inp_f <-
loading_fun(stats_se_cdt1, stats_se_cdt2, differential_expression)[[2]]
```

---

make_pen *make_pen assigns automatically penalties*

Description

make_pen calls one of four available penalty functions to automatically assign penalties for the dynamic programming. The two functions to be called are:

1. fragment_HL_pen
2. fragment_inty_pen

Usage

```
make_pen(
  probe,
  FUN,
  cores = 1,
  logs,
  dpt = 1,
  smpl_min = 10,
  smpl_max = 100,
  sta_pen = 0.5,
  end_pen = 4.5,
  rez_pen = 9,
  sta_pen_out = 0.5,
  end_pen_out = 3.5,
  rez_pen_out = 7
)
```

Arguments

probe	data frame: data frame combined data by column
FUN	function: one of the four bottom level functions (see details)
cores	integer: the number of assigned cores for the task

logs	numeric vector: the logbook vector.
dpt	integer: the number of times a full iteration cycle is repeated with a more narrow range based on the previous cycle.
smpl_min	integer: the smaller end of the sampling size.
smpl_max	integer: the larger end of the sampling size.
sta_pen	numeric: the lower starting penalty.
end_pen	numeric: the higher starting penalty.
rez_pen	numeric: the number of penalties iterated within the penalty range.
sta_pen_out	numeric: the lower starting outlier penalty.
end_pen_out	numeric: the higher starting outlier penalty.
rez_pen_out	numeric: the number of outlier penalties iterated within the outlier penalty range.

Details

The two functions called return the amount of statistically correct and statistically wrong splits at a specific pair of penalties. 'make_pen' iterates over many penalty pairs and picks the most suitable pair based on the difference between wrong and correct splits. The sample size, penalty range and resolution as well as the number of cycles can be customized. The primary start parameters create a matrix with n = rez_pen rows and n = rez_pen_out columns with values between sta_pen/sta_pen_out and end_pen/end_pen_out. The best penalty pair is picked. If dept is bigger than 1 the same process is repeated with a new matrix of the same size based on the result of the previous cycle. Only position segments with length within the sample size range are considered for the penalties to increase run time.

Value

A list with 4 items:

logbook: The logbook vector containing all penalty information

penalties: a vector with the respective penalty and outlier penalty

correct: a matrix of the correct splits

wrong: a matrix of the incorrect splits

Examples

```
data(df_comb_minimal)

df_comb_minimal$distance_HL <- df_comb_minimal$half_life.cdt1 -
df_comb_minimal$half_life.cdt2

df_comb_minimal$distance_int <- df_comb_minimal$logFC_int

pen_HL <- make_pen(
  probe = df_comb_minimal, FUN = rifiComparative:::fragment_HL_pen,
  cores = 2, logs = as.numeric(rep(NA, 8)), dpt = 1, smpl_min = 10,
  smpl_max = 50, sta_pen = 0.5, end_pen = 4.5, rez_pen = 9, sta_pen_out = 0.5,
```

```
  end_pen_out = 3.5, rez_pen_out = 7
)

pen_int <- make_pen(
  probe = df_comb_minimal, FUN = rifiComparative:::fragment_inty_pen,
  cores = 2, logs = as.numeric(rep(NA, 8)), dpt = 1, smpl_min = 10,
  smpl_max = 50, sta_pen = 0.5, end_pen = 4.5, rez_pen = 9, sta_pen_out = 0.5,
  end_pen_out = 3.5, rez_pen_out = 7
)
```

`penalties` *penalties wraps conveniently all penalty steps*

Description

penalties finds the best set of penalties for half-life and intensity fragmentation using dynamic programming. The segmentation of the HL uses the difference between 2 conditions and the segmentation of the intensity uses the log2FC.

Usage

```
penalties(data, cores = 2)
```

Arguments

`data`	data frame with the joined columns from differential expression and output of rifi_stats.
`cores`	integer: the number of assigned cores for the task. It needs to be increased in case of big data.

Details

The function uses 4 functions:

score_fun_ave.r

make_pen.r

fragment_HL_pen.r

fragment_inty_pen.r

Value

A list of data frame and penalties, The first element is data frame with 2 more variables, second and third are HL and intensity penalties respectively.

Examples

```
data(df_comb_minimal)
penalties_df <- penalties(df_comb_minimal)[[1]]
pen_HL <- penalties(df_comb_minimal)[[2]]
pen_int <- penalties(df_comb_minimal)[[3]]
```

`penalties_df` *The result of penalties for df_comb_minimal example data A data frame containing the output of penalties as a data frame*

Description

The result of penalties for df_comb_minimal example data A data frame containing the output of penalties as a data frame

Usage

```
data(penalties_df)
```

Format

A data frame with 300 rows and 20 variables:

strand: The strand specific

position: The bin/probe specific position

ID: The bin/probe specific ID

intensity.cdt1: The relative intensity at time point 0 for condition 1

position_segment: The position based segment

half_life.cdt1: The half-life of the bin/probe condition 1

TI_termination_factor.cdt1: The termination factor of the bin/probe condition 1

HL_fragment.cdt1: The half-life fragment the bin belongs to condition 1

intensity_fragment.cdt1: The intensity fragment the bin belongs to condition 1

TI_termination_fragment.cdt1: The TI fragment the bin belongs to condition 1

logFC_int: The bin/probe log2 fold change of intensity at time 0

P.Value: The bin/probe p_value adjusted

intensity.cdt2: The relative intensity at time point 0 condition 2

half_life.cdt2: The half-life of the bin/probe condition 2

TI_termination_factor.cdt2: The termination factor of the bin/probe condition 2

HL_fragment.cdt2: The half-life fragment the bin belongs to condition 2

intensity_fragment.cdt2: The intensity fragment the bin belongs to condition 2

TI_termination_fragment.cdt2: The TI fragment the bin belongs to condition 2

distance_HL: The bin/probe difference of half-life from both conditions

distance_int: The bin/probe log2 fold change of intensity at time 0

Source

https://github.com/CyanolabFreiburg/rifiComparative

---

pen_HL — *The result of penalties for df_comb_minimal example data. A list containing the output from penalties including the logbook and two penalty objects.*

Description

The result of penalties for df_comb_minimal example data. A list containing the output from penalties including the logbook and two penalty objects.

Usage

```
data(pen_HL)
```

Format

A list with 5 items:

pen_obj_HL: A list with 4 items:

logbook: The logbook vector containing half-life penalty information

HL_penalties: a vetor with the half-life penalty and half-life outlier penalty

correct: a matrix of the correct splits

wrong: a matrix of the incorrect splits

Source

https://github.com/CyanolabFreiburg/rifi

---

pen_int — *The result of penalties for df_comb_minimal example data. A list containing the output from penalties including the logbook and two penalty objects.*

Description

The result of penalties for df_comb_minimal example data. A list containing the output from penalties including the logbook and two penalty objects.

Usage

```
data(pen_int)
```

Format

A list with 5 items:

pen_int: A list with 4 items:

logbook: The logbook vector containing intensity penalty information

int_penalties: a vector with the intensity penalty and intensity outlier penalty

correct: a matrix of the correct splits

wrong: a matrix of the incorrect splits

Source

https://github.com/CyanolabFreiburg/rifi

```
rifiComparative        %
```

==========

rifiComparative

rifiComparative a successor package of rifi. It compares 2 rifi outputs from 2 different conditions of the same organism.

Description

rifiComparative was developed to compare 2 rifi outputs from 2 conditions. The rifi output may differ significantly from 2 conditions. The complexity of the segments number, position, length and the events make the comparison between 2 conditions nearly impossible. rifiComparative uses a simple strategy to generate single segments for both conditions, extract the features and make them comparable.

Details

Five major steps ate described in rifiComparative:

1. Joining data
2. penalties
3. fragmentation
4. statistics
5. visualization

Author(s)

Loubna Youssar <lyoussar@gmail.com>

Jens Georg <jens.georg@biologie.uni-freiburg.de>

rifi_visualization_comparison *rifi_visualization_comparison plots the segments and genome annotation*

Description

rifi_visualization_comparison plots the genome annotation, half-life difference (HL), log2FC(intensity) fragments. It uses several functions to plot TUs and genes including small-RNAs. Additionally it plots the statistical t-test between the neighboring fragment, significant p-values from t-test are assigned with '*'.

Usage

```
rifi_visualization_comparison(
  data,
  data_c,
  genomeLength = annot_g[[2]],
  annot = annot_g[[1]],
  condition = c("cdt1", "cdt2"),
  Strand = c("+", "-"),
  region = c("CDS", "asRNA", "5'UTR", "ncRNA", "3'UTR", "tRNA"),
  color_region = c("grey0", "red", "blue", "orange", "yellow", "green", "white",
    "darkseagreen1", "grey50", "black"),
  color_TU = c("cyan", "yellow", "orange"),
  scaling_TU = c(0, 3.4, 6.6),
  color_text.1 = "grey0",
  color_text.2 = "black",
  Alpha = 0.5,
  size_tu = 1.6,
  size_locusTag = 1.6,
  size_gene = 1.6,
  Limit = 10,
  shape = 22,
  face = "bold",
  tick_length = 0.3,
  arrow.color = "darkseagreen1",
  col_above20 = "#00FFFF",
  fontface = "plain",
  shape_above20 = 14,
  axis_text_y_size = 3,
  axis_title_y_size = 6,
  iTSS_threshold = 1.2,
  p_value_manova = 0.05,
  termination_threshold = 0.8
)
```

Arguments

data	dataframe: the probe based dataframe with joined columns.
data_c	dataframe: the probe based dataframe with joined rows.
genomeLength	integer: genome length output of gff3_preprocess function.
annot	dataframe: the annotation file.
condition	string: assigned as cdt1 (condition 1) and cdt2 (condition2), it could be adapted to any name.
Strand	string: either ("+" or "-").
region	dataframe: gff3 features of the genome.
color_region	string vector: vector of colors.
color_TU	string. TU color
scaling_TU	vector: values to adjusted termination and iTSSs to TUs.
color_text.1	string: TU color text
color_text.2	string: genes color text
Alpha	integer: color transparency degree.
size_tu	integer: TU size
size_locusTag	integer: locus_tag size
size_gene	integer: font size for gene annotation.
Limit	integer: value for y-axis limit.
shape	integer: value for shape.
face	string: label font.
tick_length	integer: value for ticks.
arrow.color	string: arrows color.
col_above20	string: color for probes/bin above value 20.
fontface	integer: font type
shape_above20	integer: shape for probes/bins above value 20.
axis_text_y_size	integer: text size for y-axis.
axis_title_y_size	integer: title size for y-axis.
iTSS_threshold	integer: threshold for iTSS_II selected to plot, default 1.2.
p_value_manova	integer: p_value of manova test fragments to plot, default 0.05.
termination_threshold	integer: threshold for termination to plot, default .8.

Details

The functions used are:

strand_selection: plots HL, intensity fragments from both strands.

splitGenome_function: splits the genome into fragments.

annotation_plot_comp: plots the corresponding annotation.

indice_function: assign a new column to the data to distinguish between fragments, outliers from delay or HL or intensity.

empty_data_positive: plots empty boxes in case no data is available for positive strand

empty_data_negative: plots empty boxes in case no data is available for negative strand

TU_annotation: designs the segments border for the genes and TUs annotation.

gene_annot_function: it requires gff3 file, returns a dataframe adjusting each fragment according to its annotation. It allows as well the plot of genes and TUs shared into two pages.

secondaryAxis: adjusts the half-life or delay to 20 in case of the dataframe row numbers is equal to 1 and the half-life or delay exceed the limit, they are plotted with different shape and color.

my_arrow: creates an arrow for the annotation.

arrange_byGroup: selects the last row for each segment and add 40 nucleotides in case of negative strand for a nice plot.

my_segment_T: plots terminals and pausing sites labels.

Value

The plot.

Examples

```
data(data_combined_minimal)
data(stats_df_comb_minimal)
data(annot_g)
rifi_visualization_comparison(
    data = data_combined_minimal,
    data_c = stats_df_comb_minimal,
    genomeLength = annot_g[[2]],
    annot = annot_g[[1]]
    )
```

`statistics` *statistics check segments significance using statistical test*

Description

statistics uses t-test to check HL and intensity segments significance. The function returns the data frame with p_value and p_value adjusted. The function used is t_test_function.

Usage

```
statistics(data)
```

Arguments

data	data frame: data frame output of fragmentation

Value

A list of two data frames, the first one contains all segments with p_value and p_value adjusted. The second one removes the duplicated segments from intensity and could be saved as an excel file.

Examples

```
data(fragment_int)
stats_df_comb_minimal <- statistics(data= fragment_int)[[1]]
df_comb_uniq_minimal <- statistics(data= fragment_int)[[2]]
```

---

stats_df_comb_minimal *The result of statistics for fragment_int example data A data frame containing the output of statistics as a data frame*

Description

The result of statistics for fragment_int example data A data frame containing the output of statistics as a data frame

Usage

```
data(stats_df_comb_minimal)
```

Format

A data frame with 500 rows and 26 variables:

strand: The strand specific

position: The bin/probe specific position

ID: The bin/probe specific ID

intensity.cdt1: The relative intensity at time point 0 for condition 1

position_segment: The position based segment

half_life.cdt1: The half-life of the bin/probe condition 1

TI_termination_factor.cdt1: The termination factor of the bin/probe condition 1

HL_fragment.cdt1: The half-life fragment the bin belongs to condition 1

intensity_fragment.cdt1: The intensity fragment the bin belongs to condition 1

TI_termination_fragment.cdt1: The TI fragment the bin belongs to condition 1

logFC_int: The bin/probe log2 fold change of intensity at time 0

P.Value: The bin/probe p_value adjusted

intensity.cdt2: The relative intensity at time point 0 condition 2

half_life.cdt2: The half-life of the bin/probe condition 2

TI_termination_factor.cdt2: The termination factor of the bin/probe condition 2

HL_fragment.cdt2: The half-life fragment the bin belongs to condition 2

intensity_fragment.cdt2: The intensity fragment the bin belongs to condition 2

TI_termination_fragment.cdt2: The TI fragment the bin belongs to condition 2

distance_HL: The bin/probe difference of half-life from both conditions

distance_int: The bin/probe log2 fold change of intensity at time 0

HL_comb_fragment: The half-life fragment the bin belongs to both conditions

HL_mean_comb_fragment: The half-life mean of the fragment the bin belongs to both conditions

intensity_comb_fragment: The intensity fragment the bin belongs to both conditions

intensity_mean_comb_fragment: The intensity mean of the fragment the bin belongs to both conditions

p_value_distance_HL: The p_value adjusted of the half-life fragment the bin belongs to both conditions

p_value_distance_intensity: The p_value adjusted of the intensity fragment the bin belongs to both conditions

Source

https://github.com/CyanolabFreiburg/rifiComparative

`stats_se_cdt1` *An example SummarizedExperiment from Synechosystis PCC 6803 first condition obtained from rifi_statistics and used as input for rifiComparative*

Description

An example SummarizedExperiment from Synechosystis PCC 6803 first condition obtained from rifi_statistics and used as input for rifiComparative

Usage

```
data(stats_se_cdt1)
```

Format

A rowRanges of SummarizedExperiment with 500 rows and 50 variables:

seqnames: The sequence name chromosome
start: The bin/probe start position
end: The bin/probe end position
width: The bin/probe length
strand: The strand specific
position: The bin/probe specific position
ID: The bin/probe specific ID
FLT: The bin/probe flag for background level
intensity: The relative intensity at time point 0
probe_TI: An internal value to determine which fitting model is applied
flag: Information on which fitting model is applied
position_segment: The position based segment
delay: The delay value of the bin/probe
half_life: The half-life of the bin/probe
TI_termination_factor: The termination factor of the bin/probe
delay_fragment: The delay fragment the bin belongs to
velocity_fragment: The velocity value of the respective delay fragment
intercept: The vintercept of fit through the respective delay fragment
slope: The slope of the fit through the respective delay fragment
HL_fragment: The half-life fragment the bin belongs to
HL_mean_fragment: The mean half-life value of the respective half-life fragment
intensity_fragment: The intensity fragment the bin belongs to
intensity_mean_fragment: The mean intensity value of the respective intensity fragment
TU: The overarching transcription unit
TI_termination_fragment: The TI fragment the bin belongs to
TI_mean_termination_factor: The mean termination factor of the respective TI fragment
seg_ID: The combined ID of the fragment
pausing_site: presence of pausing site indicated by +/-
iTSS_I: presence of iTSS_I indicated by +/-
ps_ts_fragment: The fragments involved in pausing site or iTSS_I
event_ps_itss_p_value_Ttest: p_value of pausing site or iTSS_I#'
delay_frg_slope: the slope value of the respective delay fragment
p_value_slope: p_value of the slope
velocity_ratio: Integer, ratio of velocity between 2 delay fragments
event_duration: Integer, the duration between two delay fragments

event_position: Integer, the position middle between 2 fragments with an event

FC_HL: Integer, the fold change value of 2 HL fragments

FC_fragment_HL: Integer, the fold change value of 2 intensity fragments

p_value_HL: p_value of the fold change of HL fragments

FC_intensity: Integer, the fold change value of 2 intensity fragments

FC_fragment_intensity: String, fragments involved in fold change between 2 intensity fragments

p_value_intensity: p_value of the fold change of intensity fragments

FC_HL_intensity: ratio of fold change between 2 half-life fragments and fold change between 2 intensity fragments

FC_HL_intensity_fragment: fragments involved on ratio of fold change between 2 half-life fragments and fold change between 2 intensity fragments

FC_HL_adapted: Integer, the fold change of half-life/ fold change of intensity, position of the half-life fragment is adapted to intensity fragment

synthesis_ratio: Integer, the value correspomding to synthesis rate

synthesis_ratio_event: String, the event assigned by synthesis rate either Termination or iTSS

p_value_Manova: p_value of the variance between two fold-changes, HL and intensity

p_value_TI: p_value of TI fragment

TI_fragments_p_value: p_value of 2 TI fragments

Source

https://github.com/CyanolabFreiburg/rifiComparative

`stats_se_cdt2` *An example SummarizedExperiment from Synechosystis PCC 6803 second condition obtained from rifi_statistics and used as input for rifiComparative*

Description

An example SummarizedExperiment from Synechosystis PCC 6803 second condition obtained from rifi_statistics and used as input for rifiComparative

Usage

```
data(stats_se_cdt2)
```

Format

A rowRanges of SummarizedExperiment with 500 rows and 50 variables:

seqnames: The sequence name chromosome
start: The bin/probe start position
end: The bin/probe end position
width: The bin/probe length
strand: The strand specific
position: The bin/probe specific position
ID: The bin/probe specific ID
FLT: The bin/probe flag for background level
intensity: The relative intensity at time point 0
probe_TI: An internal value to determine which fitting model is applied
flag: Information on which fitting model is applied
position_segment: The position based segment
delay: The delay value of the bin/probe
half_life: The half-life of the bin/probe
TI_termination_factor: The termination factor of the bin/probe
delay_fragment: The delay fragment the bin belongs to
velocity_fragment: The velocity value of the respective delay fragment
intercept: The vintercept of fit through the respective delay fragment
slope: The slope of the fit through the respective delay fragment
HL_fragment: The half-life fragment the bin belongs to
HL_mean_fragment: The mean half-life value of the respective half-life fragment
intensity_fragment: The intensity fragment the bin belongs to
intensity_mean_fragment: The mean intensity value of the respective intensity fragment
TU: The overarching transcription unit
TI_termination_fragment: The TI fragment the bin belongs to
TI_mean_termination_factor: The mean termination factor of the respective TI fragment
seg_ID: The combined ID of the fragment
pausing_site: presence of pausing site indicated by +/-
iTSS_I: presence of iTSS_I indicated by +/-
ps_ts_fragment: The fragments involved in pausing site or iTSS_I
event_ps_itss_p_value_Ttest: p_value of pausing site or iTSS_I
p_value_slope: p_value of the slope
delay_frg_slope: the slope value of the respective delay fragment
velocity_ratio: Integer, ratio of velocity between 2 delay fragments
event_duration: Integer, the duration between two delay fragments

event_position: Integer, the position middle between 2 fragments with an event

FC_HL: Integer, the fold change value of 2 HL fragments

FC_fragment_HL: Integer, the fold change value of 2 intensity fragments

p_value_HL: p_value of the fold change of HL fragments

FC_intensity: Integer, the fold change value of 2 intensity fragments

FC_fragment_intensity: String, fragments involved in fold change between 2 intensity fragments

p_value_intensity: p_value of the fold change of intensity fragments

FC_HL_intensity: ratio of fold change between 2 half-life fragments and fold change between 2 intensity fragments

FC_HL_intensity_fragment: fragments involved on ratio of fold change between 2 half-life fragments and fold change between 2 intensity fragments

FC_HL_adapted: Integer, the fold change of half-life/ fold change of intensity, position of the half-life fragment is adapted to intensity fragment

synthesis_ratio: Integer, the value correspomding to synthesis rate

synthesis_ratio_event: String, the event assigned by synthesis rate either Termination or iTSS

p_value_Manova: p_value of the variance between two fold-changes, HL and intensity

p_value_TI: p_value of TI fragment

TI_fragments_p_value: p_value of 2 TI fragments

Source

https://github.com/CyanolabFreiburg/rifiComparative

Index