

Package ‘saas’

December 7, 2016

Title An implementation of the Search All, Asses Subset strategy for FDR estimation shotgun proteomics.

Version 0.0.0.9000

Description An implementation of the Search All, Asses Subset strategy for FDR estimation in shotgun proteomics.

Depends R (>= 3.3.0)

License Apache License, Version 2.0

URL <https://github.com/compomics/search-all-assess-subset>

BugReports <https://github.com/compomics/search-all-assess-subset/issues>

Encoding UTF-8

LazyData true

Imports tidyverse (>= 1.0.0),
cowplot (>= 0.7.0),
markdown (>= 0.7.0)

Suggests mzR (>= 2.8.0)

RoxygenNote 5.0.1

R topics documented:

calculate_fdr	2
dpi0	2
id_is_present	3
parse_msgf_mzid	3
pi0plot	4
plot_diag	4
plot_theo_dist	5
PPplot	5
preprocess	6
rpi0	6
saas_gui	7
simulate_subset	7

Index	8
--------------	----------

calculate_fdr	<i>Calculate qvalues on the subset PSMs.</i>
---------------	--

Description

Calculate qvalues on the subset PSMs.

Usage

```
calculate_fdr(df, score_higher = TRUE)
```

Arguments

df	dataframe with at least 3 columns: score score assigned to the peptide to spectrum match (PSM). subset TRUE if PSM belongs to the subset in interest, FALSE or otherwise. decoy TRUE if decoy PSM, FALSE otherwise. Additional columns are allowed but ignored. Target and decoy PSMs are assumed to be from a competitive target decoy database search.
score_higher	TRUE if a higher score means a better PSM.

Value

A data frame containing all columns in “df”. Following columns are added:

pi_0_cons conservative estimation of π_0 .
FDR estimated subset PSM qvalues calculated according the competitive target decoy approach.
FDR_BH estimated subset PSM qvalues calculated according the Benjamini Hochbergh procedure. When provided, non-subset decoy PSMs are used to stabilize estimates in small subsets
FDR_stable estimated subset PSM qvalues calculated with “pi_0_cons”. When provided, non-subset decoy PSMs are used to stabilize estimates in small subsets

dpi0	<i>Density function for the pi0 distribution</i>
------	--

Description

Density function for the pi0 distribution

Usage

```
dpi0(pi0, n_targets, n_decoys)
```

Arguments

pi0	vector of quantiles.
n_targets	vector of observed target PSMs.
n_decoys	vector of observed decoy PSMs.

Value

vector of densities. The length is the maximum length of the numerical arguments. Returns 'NaN' for ' $\pi_0 < 0$ ' and ' $\pi_1 > 1$ '.

id_is_present	<i>Checks if protein id appears in the headers of a fasta file.</i>
---------------	---

Description

Checks if protein id appears in the headers of a fasta file.

Usage

```
id_is_present(protein_id, fastapath)
```

Arguments

protein_id	Vector of protein ids.
fastapath	Location of the fasta file.

Value

Logical vector, TRUE if protein id is present in provided fasta file, FALSE otherwise.

parse_msgf_mzid	<i>Parses a mzID file generated by MS-GF+.</i>
-----------------	--

Description

See <https://omics.pnl.gov/software/ms-gf> for more info on how to perform a database search on MSMS dataset with MS-GF+ and how to generate a mzID file. Note that most functions in these package require data from a competitive target decoy search.

Usage

```
parse_msgf_mzid(mzid_path)
```

Arguments

mzid_path	Location of the mzID file.
-----------	----------------------------

Value

A data frame containing the following 7 columns:

spec_id Id of the spectrum from the searched dataset file.
sequence Amino acid sequence matching the spectra.
protein_id Id of the sequence from the database file.
score score assigned to the peptide to spectrum match (PSM).
database Name of the database file used to search the spectra.
decoy TRUE if decoy PSM, FALSE otherwise.
database_size Number of sequences in the database file.

pi0plot	<i>Creates density plot of the pi0 distribution</i>
---------	---

Description

Creates density plot of the pi0 distribution

Usage

```
pi0plot(n_targets, n_decoys)
```

Arguments

n_targets vector of observed target PSMs.
n_decoys vector of observed decoy PSMs.

Value

ggplot object.

plot_diag	<i>Plot diagnostic plots to evaluate assumptions from the search all, search subset strategy.</i>
-----------	---

Description

Plot diagnostic plots to evaluate assumptions from the search all, search subset strategy.

Usage

```
plot_diag(df)
```

Arguments

df dataframe with at least 3 columns:
score score assigned to the peptide to spectrum match (PSM).
subset TRUE if PSM belongs to the subset in interest, FALSE otherwise.
decoy TRUE if decoy PSM, FALSE otherwise.
 Additional columns are allowed but ignored. Target and decoy PSMs are assumed to be from a competitive target decoy database search.

plot_theo_dist	<i>plots the theoretical distribution of all components in the PSM distribution</i>
----------------	---

Description

plots the theoretical distribution of all components in the PSM distribution

Usage

```
plot_theo_dist(H0_mean = 2.75, H1_mean = 3.31, H0_sd = 0.13,  
              H1_sd = 0.28, decoy_mean = H0_mean, decoy_sd = H0_sd,  
              decoy_extra_mean = H0_mean, decoy_extra_sd = H0_sd)
```

PPplot	<i>Creates PP plot of two empirical distributions</i>
--------	---

Description

Creates PP plot of two empirical distributions

Usage

```
PPplot(score, label, pi0 = 0, title = "PP plot of target PSMs",  
       xlab = "Decoy percentile", ylab = "Target\npercentile")
```

Arguments

score	vector of quantiles of distribution 1 and 2
label	vector of logical values. TRUE if score belongs to distribution 1
pi0	mixture coefficient of distribution 1 in distribution 2
title	main title.
xlab	label on x-axis.
ylab	label on y-axis.

Value

ggplot object

preprocess	<i>Preprocess data from a MS-GF mzID file.</i>
------------	--

Description

The parsed data frame from `saas::parse_msgf_mzid` function contains sometimes multiple entries for a spectrum. (eg. if sequence can be assigned to multiple protein ids). This function takes care of this by default.

Usage

```
preprocess(dat, remove_target_decoy_PSM = TRUE,
           remove_multiple_proteins_PSM = FALSE)
```

Arguments

<code>dat</code>	Data frame generated by the <code>saas::parse_msgf_mzid</code> function.
<code>remove_target_decoy_PSM</code>	TRUE to remove PSMs that match both a target and decoy sequence.
<code>remove_multiple_proteins_PSM</code>	TRUE to remove PSMs that can be assigned to multiple protein ids.

Value

Data frame with the same columns as “dat” and 1 column is added:

non_subset TRUE if sequence can be assigned to a non subset protein id

Every spectrum has only 1 row in the data frame.

rpi0	<i>Random generation for the pi0 distribution</i>
------	---

Description

Random generation for the pi0 distribution

Usage

```
rpi0(n, n_targets, n_decoys)
```

Arguments

<code>n</code>	number of observations.
<code>n_targets</code>	number of observed target PSMs.
<code>n_decoys</code>	number of observed decoy PSMs.

Value

vector of random deviates. The length equals ‘n’.

saas_gui	<i>Launches the GUI version of saas.</i>
----------	--

Description

To easily launch the GUI outside an R session (eg. on a server), you can run `R -e "library(saas);saas_gui()"` from the terminal (on linux/mac).

Usage

```
saas_gui(options = list(port = 3320, host = "0.0.0.0"))
```

Arguments

options	See help of shiny::shinyApp for more details on available options
---------	---

simulate_subset	<i>Random generation of a dataset after TDA.</i>
-----------------	--

Description

Random generation of number of decoy, correct target and incorrect target PSMs target PSMs after a competitive target-decoy search.

Usage

```
simulate_subset(n, pi0, sims = 1)
```

Arguments

n	number of total PSMs.
pi0	theoretical π_0 .
sims	number of observations.

Value

A data frame with “sims” rows and 6 rows:

n number of PSMs.

pi0 theoretical π_0 .

decoy_n number of decoy PSMs.

target_n number of target PSMs.

H0_n number of incorrect target PSMs.

H1_n number of correct target PSMs.

Index

`calculate_fdr`, [2](#)
`dpi0`, [2](#)
`id_is_present`, [3](#)
`parse_msgf_mzid`, [3](#)
`pi0plot`, [4](#)
`plot_diag`, [4](#)
`plot_theo_dist`, [5](#)
`PPplot`, [5](#)
`preprocess`, [6](#)
`rpi0`, [6](#)
`saas_gui`, [7](#)
`simulate_subset`, [7](#)