MutCombinator

Release: Mar 26, 2020

Hanyang University, Seoul, Korea

1. Requirement

MutCombinator requires Java Running Environment (JRE) 1.7.0_45 or later. JRE can be installed by visiting http://www.oracle.com/technetwork/java/index.html.

2. Usage

1) Construct a variant graph and output to a file

java -jar variantgraph_gen.jar <options>

a) Options

-i <input file_path>: Config file path for search parameters [Required].

-o <output file_path>: Output file path for the variant graph [Optional].

b) Examples

java -jar variantgraph_gen.jar -i param.txt -o variantgraph.txt

2) Peptide identification using a variant graph

java -jar MutCombinator.jar <options>

a) Options

-i <input file_path>: Config file path for search parameters [Required].

-o <output ***folder***_path>: Output file path for search results [Optional].

* If there is no such a folder, it automatically generates the folder.

-@ <thread number>: The number of threads for search [Optional].

* If not specified, it automatically applies the maximum number of threads available in a system environment.

b) Examples

java -jar MutCombinator.jar -i param.txt -o output -@ 3

java -jar MutCombinator.jar -i param.txt

3) False discovery rate estimation

java –jar fdr_estimator.jar <options>

a) Options

-i < MutCombinator_result file_path>: Peptide identification result file name [Required]

-p <parameter file_path>: The same search parameter file used in the search [Required].

-fdr <fdr ratio>: Desired false discovery rate (0.00 ~ 1.00) [Required]

-o <output folder_path>: Output folder path [Optional].

* If there is no such a folder, it automatically generates the folder.

-s <l|g> : FDR strategy such as l for local fdr and g for global fdr. Default value is local fdr.

b) Examples

java -jar fdr_estimator.jar -i result.MutCombinator.txt -p param.txt -fdr 0.01 -o fdr_output -s g

3. Parameters for search

To run MutCombinator, a parameter file should be given as a text file.

Each line of the file has a form of “parameter=[VALUE]”.

Example)

Spectra=sample.mgf

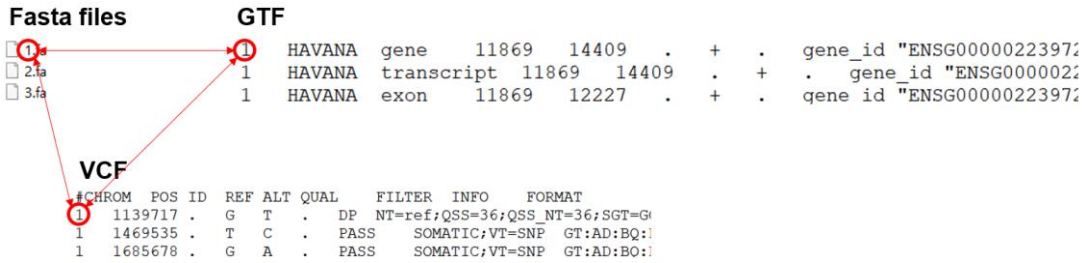
1) GTF=[FILEPATH] : Specifies a GTF file for gene models to construct a variant graph.

2) FASTA=[FOLDERPATH] : Specifies a folder that contains Fasta files to construct a variant graph.

3) VCF=[FILEPATH|FOLDERPATH] : Specifies a file/folder that contains VCF files representing mutations.

**** Caution! ****

Three inputs refer each other via chromosome index, thus a user must follow below rule.



The red circle indicates chromosome index in each file. They refer each other via the index. If the index is inconsistent, then a user may not get an expected result.

4) Spectra=[FILEPATH] : Specifies a path to the spectra file to search. Supported formats are *.mgf, *.pkl, *.dta, and *.mzXML.

5) VariantGraph=[FILEPATH] : Specifies a path to the variant graph file to search.

6) Instrument=[TYPE] : Specifies the type of MS/MS instrument used (or best matched) for peptide fragmentation. According to the instrument type, a different fragmentation model is applied. TYPE=(ESI-TRAP | ESI-QTOF). Default value is ESI-TRAP.

7) PPMTolerance=[VALUE] : Sets a parent mass tolerance in ppm. Default value is 10.

8) AutoPMCorrection=[0|1] : The default value is 0. If this parameter is set to 1 (this requires more search time), the program will automatically find the optimal parent mass for the input spectrum, regardless of the specified PeptTolerance. This parameter would be useful when some spectra have bigger mass error than what you specify in PeptTolerance parameter.

9) FragTolerance=[MASS] : Sets a fragment ion mass tolerance in Dalton. Default value is 0.025.

10) MinPeptLen=[LENGTH] : Sets a minimum peptide length to consider in search. Default value is 8.

11) MinTagLen=[3|4] : Sets a length of tag. Note that although length 4 is much faster than length 3, it can loss some of peptides. Default value is 3.

12) BlindMode=[0|1] : Sets the number of modifications per peptide, default value is 0. To be specific, 0 allows no modification per peptide, and 1 allows one modification per peptide.

13) Min/MaxModSize=[MASS] : Sets the minimum/maximum modification mass in Dalton to consider, default values are 0.

14) Mut=[NUMBER] : Sets the number of maximum mutations per peptide. Default value is 3. Note that the number must be integer greater than or equal to 0.

15) Enzyme=[NAME]{,[CLEAVAGE/TERMINUS]}* : Specifies the reagent used for protein digestion. CLEAVAGE specifies residues specific to the enzyme. Several residues can be concatenated. TERMINUS is either N or C, designating at which side of the residues the enzyme cleaves. 'N'=amino side, 'C'=carboxyl side. Users are free to define an enzyme.

    Ex) 'myenzyme, D/N, D/C, FL/C'. Separating multiple cleavage rules with a comma can be added. This enables the use of multiple enzymes in searching. Ex) 'Trypsin+GluC+LysN, KR/C, E/C, K/N'.

Default value is 'NONE(for non-enzyme search)'.

16) enzyme_constraint_min_number_termini=[0|1|2] : Set the number of enzymatic termini allowed during search.

17) MissedCleavage=[NUMBER] : Sets the number of allowed missed cleavage sites. Default value is 2.

18) ADD=[RESIDUE],[MASS]: Specifies the mass (in Dalton) of a fixed (static) modification on a residue. (e.g. cysteine alkylation, isotope labeling)

19) Protocol=[iTRAQ4plex/iTRAQ8plex/TMT2plex/TMT6plex/TMT10plex/NONE] : Before peptide spectrum match, MutCombinator removes iTRAQ or TMT reporter ion peaks, isobaric tag peaks & isobaric tag complementary peaks, so that those peaks are not used for fragment ion match.

20) HighResolution=[ON|OFF] : When HighResolution is turned on and fragment tolerance is larger than precursor tolerance, fragment tolerance is reduced to precursor tolerance.

21) DecoySearch=[ON|OFF] : Reverse decoy of the target sequence is automatically generated and searched when set as ON. Default value is ON.

22) SearchRegion=[CODING|NONCODING] : Specifies the search region of interests. It searches non-coding regions in the variant graph using three frame translation when set as NONCODING. Default value is CODING.

4. Output (.MutCombinator.txt)

Peptide identification file(.MutCombinator.txt) is tab-separated.

>>spectrum_file index observed_MW charge_state (scan_number)

[Rank 1] calculated_MW delta_mass score probability peptide geneid_with_mutations

[Rank 2] calculated_MW delta_mass score probability peptide geneid_with_mutations

…

- spectrum_file : name of the input spectrum file
- index : indices for ms/ms scans in the input spectrum file, starting from 1
- observed_MW : molecular weight of the precursor ion
- charge_state : charge state of the precursor ion
- scan_number : (only applicable for mgf & mzxml file formats) spectrum scan number indicated in the spectrum file

- calculated_MW : molecular weight of the peptide
- delta_mass : observed_MW minus calculated_MW
- score : score of the peptide-spectrum match
- probability : probability that the peptide-spectrum match is correct
- peptide : identified peptide
- geneid_with_mutations : reported geneid. If there are mutations in the peptide, mutations are also reported by appending after geneid. For example, ENSG1@1:4:T>C. @ denotes information about mutation occurrence as (chr:locus:refseq>altseq).

5. Citation

Not available now.

6. License

7. Contact

For feedback, questions and comments, contact us at prix@hanyang.ac.kr.