fastcov - Fast Covariant Mutation Detector v1.01

Fast Covariant Mutation Detector. http://yanlilab.github.io/fastcov

Introduction

Covariant mutations are very important to maintain the structural characteristics and consequently to maintain the protein conformational and functional stability. In this study, we developed a novel algorithm to identify correlated changes by using an independent pair model followed by a correlated tandem model.

Fastcov is based on a correlation idea of interaction restriction among site-residue elements, which is very suitable for natural co-variances analysis. In contrast to other complex methods, the lightweight and fast Fastcov algorithm significantly improves the processing efficiency.

By tests on the genotyping, phylogeny and divergence analysis, the results demonstrated that this approach has an excellent performance on detecting covariant residue patterns. Based on the covariant pattern clustering, the genotyping performance reached a sensitivity of 99.42%, a specificity of 99.94% and an accuracy of 99.77%; The covariant patterns displayed co-evolutionary modes corresponding to the phylogeny tree; Moreover, it found an important evidence involving in the structural stability of protein during the evolution. As an original algorithm, Fastcov provides not only a fast and reliable approach to achieve the data analysis, but also much more powerful functions including multiple variance detection and evolutionary classification.

Installation

`fastcov` is implemented in Golang programming language, executable binary files for most popular operating system are freely available in release page.

Just download executable file of your operating system and rename it to `fastcov.exe` (Windows) or `fastcov` (other operating systems) for convenience, and then run it in command-line interface, no any dependency are needed.

You can also add the directory of the executable file to environment variable `PATH`, so you can run `fastcov` anywhere.

1). For windows, the simplest way is copy it to `C:\WINDOWS\system32`.

2). For Linux, type:

```
chmod a+x /PATH/OF/FASTCOV/fastcov
echo export PATH=\$PATH:/PATH/OF/FASTCOV >> ~/.bashrc
```

or simply copy it to `/usr/local/bin`

Usage

```
usage: fastcov [-p ] [-d ] [-r ] [-c ] [-n ] [-o ] [-j ] [-h] INPUTFILE

positional arguments:
  inputfile

options:
  -p      minimum pairing purity of two sites [0.7]
  -d      minimum associated degree [0.6]
  -r      minimum matching ratio of to the pattern [0.45]
  -n      minimum residue number at each site [5]
  -c      minimum proportion of any sequence identical to the consensus [0.33]
  -o      prefix of output files [inputfile]
  -j      CPU number [CPU number of your computer]
  -h      show help
```

Positional arguments

- `inputfile` should be aligned protein sequences in FASTA format file, produced by multi sequence alignment softwares. Case is not sensitive.

  One-seq-per-line format could be converted to FASTA format by `for f in *.aln; do cat -n $f | awk '{print ">"$1"\n"$2}' > "$f.fas"`

Options

Main algorithm parameters

- `-p` defines the minimum pairing purity of two sites. Default is 0.7.
- `-d` defined the minimum associated degree of one group of covariant mutation elements. Default is 0.6.
- `-r` defines the minimum matching ratio of to the pattern at clustering stage. Default is 0.45.

Sequences filter criteria

- `-n` is the minimum residue number at each site. Default value is 5.
- `-c` is the minimum proportion of any sequence identical to the consensus. Default value is 0.33, i.e. the number of residues identical to the that of the same position of consensus sequences should be at least one third of the length of consensus. Sequences that fail to reach this criteria will be discarded.

Output

- `-o` defines the prefix of output files, default value is the same as input file. e.g, for a input file `test.fa`, output files will be:

```
test.aligned.fa.pairs
test.aligned.fa.clusters
test.aligned.fa.patterns
test.aligned.fa.seq2patterns
```

Performance

- `-j` is the number of CPU. `fastcov` detects your computer and set the default value with the maximum CPU number. The bigger the value is, the faster `fastcov` runs.

Examples

Taking `examples/ABCD_RT_M.aligned.fas` for example.

Quik run:

```
fastcov ABCD_RT_M.aligned.fas
```

Terminal Output:

```
Input: ABCD_RT_M.aligned.fas

Step 1/5: Reading sequences


Done

Step 2/5: Searching candidate sites

Done

Step 3/5: Searching independent pairs
21115 / 21115 [================================================================================

Covariant site pairs saved to file: ABCD_RT_M.aligned.fas.pairs

Done

Step 4/5: Searching covariant patterns
52 / 52 [======================================================================================

Covariant patterns saved to file: ABCD_RT_M.aligned.fas.patterns

Done

Step 5/5: Clustering by covariant patterns
Covariant patterns assigned to sequences: ABCD_RT_M.aligned.fas.seq2patterns
Sequences clustered by covariant patterns: ABCD_RT_M.aligned.fas.clusters
```

The most time-consuming stage is `step 3`, so we add a process bar.

Output files:

```
ABCD_RT_M.aligned.fas.pairs           # covariant pairs information
ABCD_RT_M.aligned.fas.patterns        # covariant patterns
ABCD_RT_M.aligned.fas.clusters        # sequence clusters by covariant patterns
ABCD_RT_M.aligned.fas.seq2patterns    # covariant patterns of every sequence
```

More examples

Errors and Solutions

1) No input file given. Please feed `fastcov` a aligned amino acids sequences in FASTA format.

```
$ fastcov
[Error] no input file (aligned amino acids sequences in FASTA format) given.
type "fastcov -h" for help
```

2) Input file is not aligned.

```
[Error] sequence length not equal: 343 (AB014392_Pol-C) != 344.
input file should be aligned amino acids sequences in FASTA format
```

3) Illegal characters in sequence. FASTA parsing module of `fastcov` strictly check the sequences, you may check input sequence according according to the IUPAC nucleotide code (http://www.bioinformatics.org/sms2/iupac.html). It may also be caused by unmatch of sequence type (PROTEIN) and actual sequence type (DNA) in FASTA file.

```
Input: test.fa

Step 1/5: Reading sequences
error when reading AB014367_Pol-C: invalid Protein sequence: AB014367_Pol-C
```

FAQ

Please don't hesitate to email us.

Authors

Yan Li liyan.com@gmail.com, Wei Shen shenwei356@gmail.com

Copyright