Speptide. Application to find amino acids substitutions by spectra comparison.

version: 0.92.

Dmitry Ischenko, Dmitry Alexeev, Ilya Altukhov

June 29, 2016

0.1 Intro

Spectide written at C/C++ language. The application allows you to determine the amino acid substitutions based on a comparison of the spectra. The basic idea of the spectrum as a vector, shifts of the peaks and the calculation of the angle between the vectors.

0.2 Installation and usage

For installation just run:

```
$ make
```

in folder with project.

To search for a substitutions you should have two sets of spectra. One – experiment set of spectra and another – database set of spectra (with SEQ= tag for each spectra). Both in Mascot Generic Format (.mgf).

To create database spectra from **.mgf** and mascot result **.csv** file (it must contain columns with spectrum name "pep_scan_title" and peptide sequence "pep_seq") – use **mgfe.pl** script (utils/mgfe.pl):

```
$ perl mgfe.pl mseq <mascot csv> <mgf> > <db mgf>
```

Next. To run application:

```
$ speptide <exp mgf> <db mgf> <config>
```

Results in tab separated format:

```
[exp spectrum id] [db spectrum id] [position] [exp ami] [db ami] [db seq] [cos(theta)]
```

```
<exp spectrum id> # experiment spectrum title
<db spectrum id>  # database spectrum title
<position>        # positions of substitution (if several space as delimiter)
<exp ami>         # aminoacid in experiment spectrum (substitution)
<db ami>          # aminoacid in database spectrum
<db seq>          # database spectrum sequence
<cos(theta)>      # cos(angle) between spectra
```

0.3 Params

File with parameters (**.ini**) consists of several keys:

```
# Default settings for algorithm

# Part for algorithm of idenitcal spectras
[ident]
ppm = 10;     # ppm accuracy for MS1 peak intersection
Da = 0.5;     # Da accuracy for MS2 peak intersection
N = 100;      # value for top (m / N) intensity peak selection
trans = b;    # algorightm for transfromation (a : sqrt, b : ln, c : none)
norm = y;     # normalize intensity
const = 0;    # add constant after normalization and transfromation
cos01 = 0.47; # value of threshold of cos(theta) (FDR <= 0.01)
cos05 = 0.3;  # value of threshold of cos(theta) (FDR <= 0.05)

# Part for algorithm for finding sap
[sap]
ppm = 10;     # ppm accuracy for MS1 peak intersection
Da = 0.5;     # Da accuracy for MS2 peak intersection
N = 3.2;      # value for top (N * S) intensity peak selection (S number of annotated peaks)
trans = b;    # algorightm for transfromation (a : sqrt, b : ln, c : none)
norm = y;     # normalize intensity
const = 0;    # add constant after normalization and transfromation
mcos = 0.3;   # value of threshold of cos(theta) for modification filtration
cos = 0;      # value of threshold of cos(theta) for printing
addions = n;  # additional ions (-H2O, -NH3)
refdiv = 1;   # value for top (mz / N) intensity peak in reference (1 mean all)
fident = y;   # filter identical spectra from SP
fmod = y;     # filter modifications from SP

# Annotation params
[annot]
Da = 0.5      # Da accuracy for MS2 peak annotation
Ch = 2,3      # Add defalt charges (if didn't set in mgf)
```

Training results:

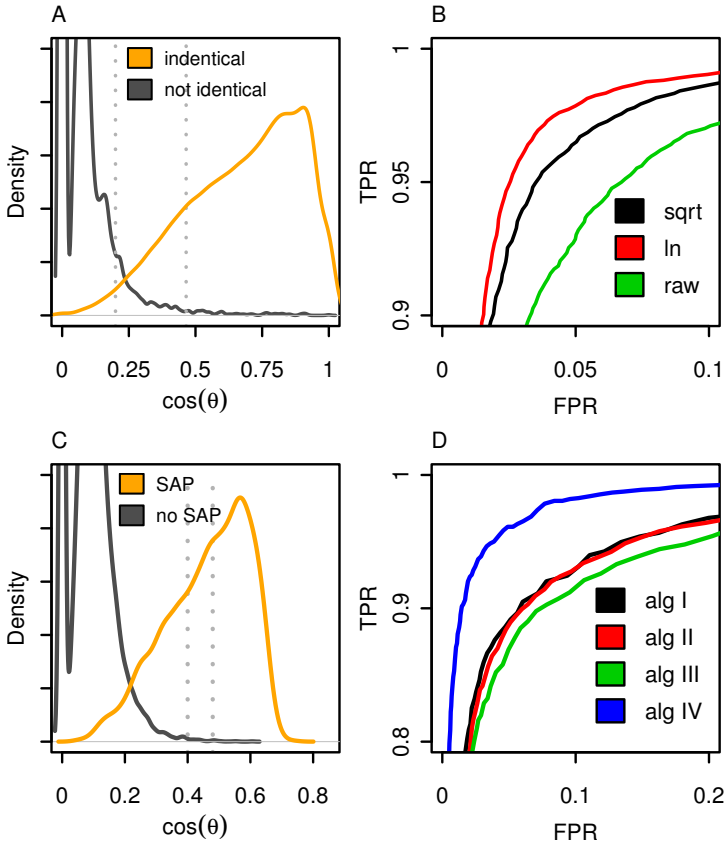

Figure 1: **A**. Probability density of $\cos\Theta$ between the spectra, corresponding to similar and different peptide sequences ($\ln I$ transformation, top $\frac{m}{100}$ peaks). **B**. ROC curves for different methods of intensity transformation. **C**. Probability density of $\cos\Theta$ between the spectra, corresponding to true SAP and random match (IV algorithm, $\ln I$ transformation, top $3.2\dot{S}$ peaks). **D**. ROC curves for different methods of reference spectra transformation.

0.4 Pipeline

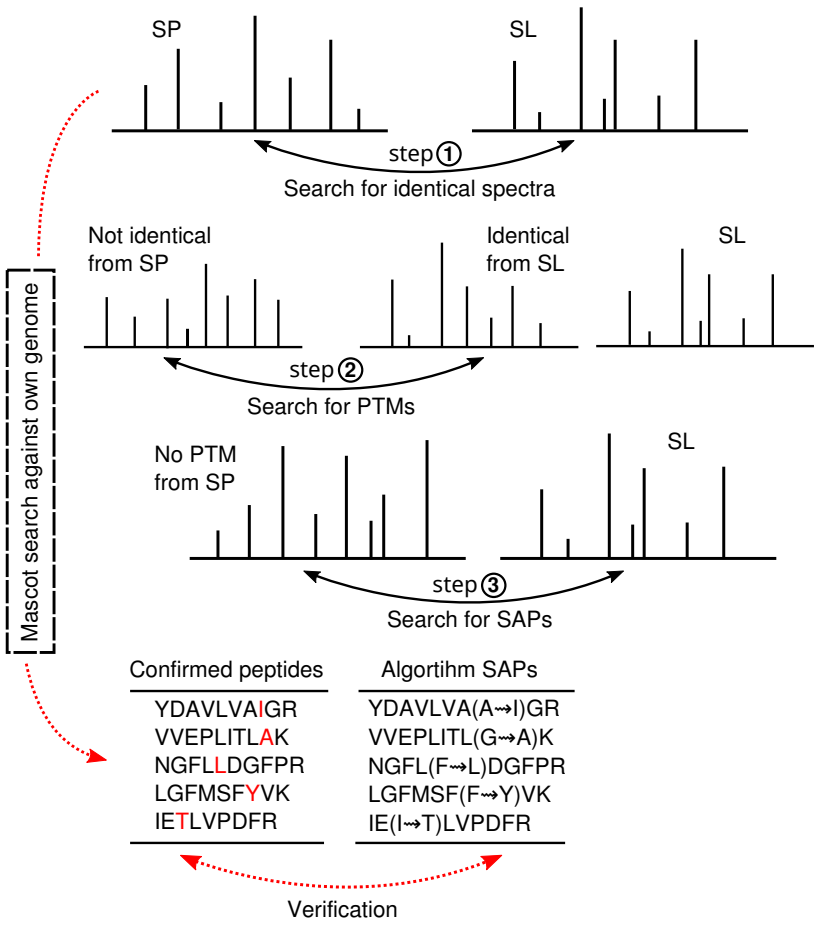

Figure 2: Flowchart of algorithm for detecting the spectra with single amino acid substitution (steps 1-3) and additional verification procedure.

Application algorithm consists of several steps:

1. Search for identical spectra in SP and SL. Filtration identical spectra from SP.
2. Search for PTM spectra in SP against identical form SL. Filtration PTM spectra from SP.
3. Search for SAPs.

0.5 Example

Graphical representation of the results of algorithm for several bacterial spectra datasets:

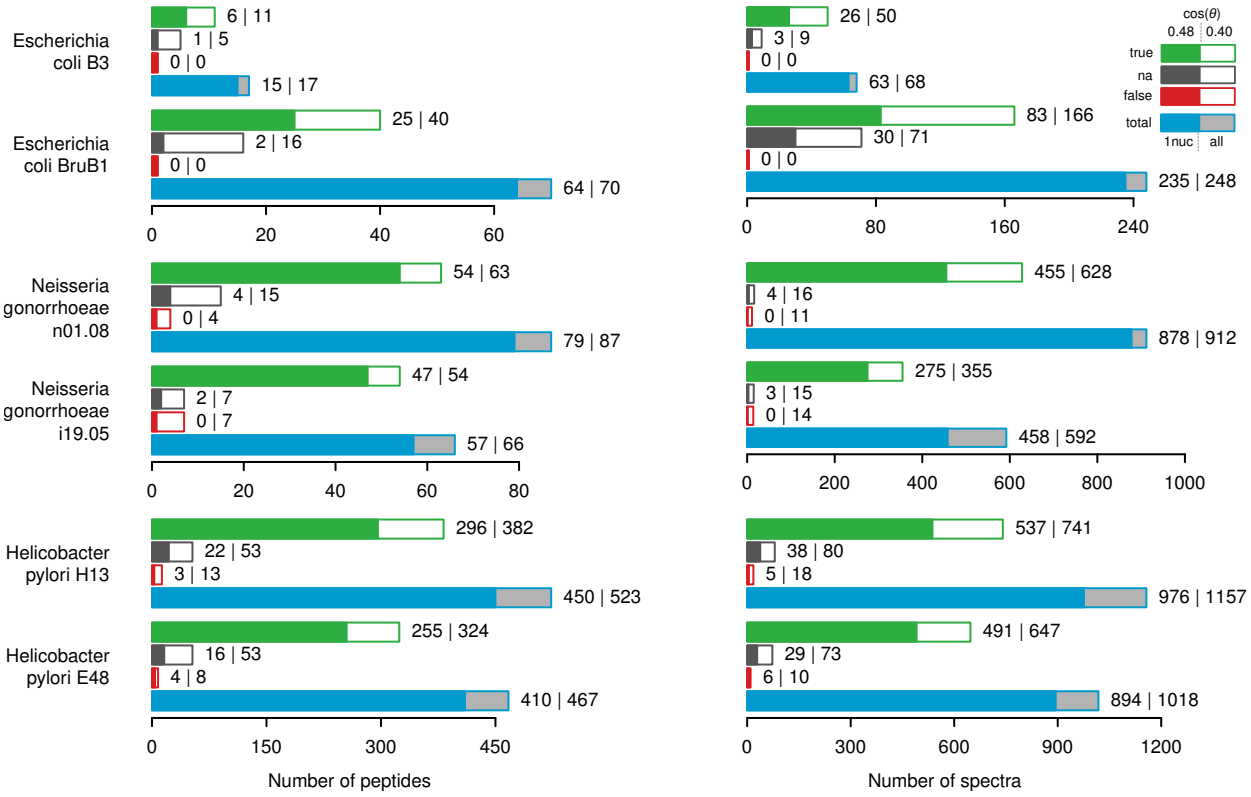

Figure 3: Example of result of application for different bacterial datasets.

0.6 Contribution

For comments and requests, send an email to:
Dima Ischenko (ischenko.dmitry@gmail.com)