

## **Treephyler**

### **Fast Taxonomic profiling of metagenomes**

This software is freely available at  
<http://www.gobics.de/fabian/software/treephyler.php>

Please read the COPYRIGHT file before using this software.

If you use Treephyler please cite:  
*F. Schreiber, P. Gumrich, R. Daniel and P. Meinicke (2009)*  
*Treephyler: fast taxonomic profiling of metagenomes*  
*Bioinformatics (in press), doi:10.1093/bioinformatics/btq070*

#### **Installation**

- 1) Checking system requirements
  - Perl (version 5.8 or higher)
  - BioPerl (version 1.4 or higher)
  - HMMER Version 2.3.2 (October 2003)  
(<http://hmmmer.janelia.org/#download>)
  - Pfam database (Release 23.0)
  - FastTree Version 2.0.1

Installing the Pfam database (Release 23)

- a) The following Pfam files are required:
  - Pfam\_fs                   The Pfam fragment HMM library
  - Pfam-A.full               The full alignments of all curated families
  - Pfam-A.fasta             A fasta version of Pfam's underlying sequence database
  - pfamseq.txt              Contains the taxonomic information (Make sure to not mistake the pfamseq.txt with the pfamseq file)  
(Note that PFAM might change the data format of the above mentioned files)

These files can be downloaded from:

<ftp://ftp.sanger.ac.uk/pub/databases/Pfam/releases/Pfam23.0/>

- b) Create an index for the Pfam\_fs database:

- hmminindex Pfam\_fs

- c) Extract all files.

#### **INSTALLING TREEPHYLER**

- 4) Getting the most recent treephyler tarball from  
<http://www.gobics.de/fabian/software/treephyler.php>

- 5) Unpacking the tarball in a temporary directory:

```
tar -xvzf treephyler.tar
```

- 6) Configure treephyler

All global configurations are stored in the perl file  
Configurations.pm.

Adapt all parameters in the section "System parameters" to your  
system.

- 7) Use the script "treephyler.pl" to prepare all PFAM-related files  
(alignments, hmms, taxonomy):

```
perl treephyler.pl -m prepare -p Pfam-A.full -pfamT pfamseq.txt  
-pfamH Pfam_fs -pfamF Pfam-A.fasta
```

This will put all pfam alignments, hmms, and taxonomy entries  
into single files. The directory structure will be:

```
pfam/alignments  
pfam/hmms  
pfam/taxonomy
```

- 8) Testing the installation

Execute treephyler: `perl treephyler.pl`

A usage message describing the available parameters should appear.

### **Prediction workflow**

The whole prediction workflow consists of two parts, the generation of Pfam assignments using e.g. UFO and the taxonomic classification using Treephyler.

### **UFO Predictions**

PFAM predictions for input sequences can be made using the UFO web server at [www.ufo.gobics.de](http://www.ufo.gobics.de). You can use treephyler to translate your dataset into protein sequences and split it into equal chunks, so that you can easily analyse it using UFO.

### **Taxonomic classification using treephyler**

Adapt the configuration file ("Configuration.pm") to match your type of input data (nucleotide or protein) and the computer you are using (single-/multi core or computer cluster using a Sun Grid Engine).

With all sequences in "sequences.fa" and all UFO predictions in

"UFO\_predictions.txt" you can start the analysis typing:

```
perl treephyler.pl -m analyse -i sequences.fa -a
Ufo_predictions.txt
```

Treephyler will put all output files in project directory specified in the configuration file ("Configuration.pm").

You will find the taxonomic classifications in the file "taxonomic\_assignments.txt" in the project directory. In case you start the analysis in a parallel environment, results will be in "taxonomic\_assignments1.txt, taxonomic\_assignments2.txt, taxonomic\_assignments3.txt", usw. You can easily put all assignments in one file using

```
cat taxonomic_assignments* > taxonomic_assignments.txt
```

## **Additional methods**

### **Splitting file into equal chunks**

The following command takes the input file "sequences.fa" and splits into files of e.g. 50 mb size each.

```
perl treephyler.pl -m split -i sequences.fa -msize 50
```

### **Translating file to protein format**

The following command takes the input file "sequences.fa", translates it in all six reading frames, and saves it in "sequences.fa\_translated".

```
perl treephyler.pl -m translate -i sequences.fa
```

### **Generating statistics**

The following command generates a statistic file showing the percentage of assigned sequences for each bacterial phylum from all predictions in file "all\_predictions.fa".

```
perl treephyler.pl -m statistics -I all_predictions.fa
```

## **Appendix**

Input format:

-sequences: Input sequences have to be in multiple fasta format and can be either nucleotide or protein sequences  
-predictions: The required format is:

```
FASTA_HEADER_OF_SEQUENCE  
PFXXXX
```

```
FASTA_HEADER_OF_SEQUENCE  
PFXXXX  
PFYYYY
```

```
FASTA_HEADER_OF_SEQUENCE  
no assignments
```

where "FASTA\_HEADER\_OF\_SEQUENCE" is the fasta header that can also be found in the sequence file and "PFXXXX" is the PFAM family the sequence has been assigned to. This is the standard UFO output format. But you can adapt the output of different prediction methods and use them as input.

## Synopsis:

Parameters of treephyler.pl:

```
perl treephyler.pl -m ("statistics|"analyse|"prepare|"split|"translate") -i Fastafile -a Assignment_file  
[-nsplits|-pfamA|-pfamT|-pfamH|-pfamF]
```

Required parameters

-i : Contains the query sequences in Fasta format

-a: Contains PFAM predictions in UFO format

-m: modus, available options are:

"statistics" - Compute statistics

e.g. perl treephyler.pl -m statistics -i assignments.fa

"prepare" - Extracts information about alignments, taxonomy, and hmms from PFAM files and saves them in the subfolder "pfam"

e.g. perl treephyler.pl -m prepare -p Pfam-A.full -pfamT pfamseq.txt -pfamH Pfam\_fs -pfamF Pfam-A.fasta

"analyse" - Start analysis

e.g. perl treephyler.pl -m analyse -i gletscher.fas -a Ufo.out

"split" - Splits a file into several smaller chunks

e.g. perl treephyler.pl -m split -i input\_sequences.fa -nsplits 50000

"translate" - Translates sequences in all six reading frames

e.g. perl treephyler.pl -m translate -i input\_sequences.fa

"help" - print this help message

Example call:

```
perl treephyler.pl -m analyse -i gletscher.fas -a Ufo.out
```

-----