Practical-7 : Function Prediction

Siddharth Tomar

May 28, 2017

Summary

Sequences

Group 11

Input	Organism	Size (in basepairs)	Kingdom	Number of Genes
05.fa.txt	*Chlamydia trachomatis*	1042588	Bacteria	977
11.fa.txt	*Geobacter sulfurreducens*	4566144	Bacteria	4,172
15.fa.txt	*Pseudomonas aeruginosa*	6433441	Bacteria	5,938
19.fa.txt	*Thermodesulfovibrio yellowstonii*	2003803	Bacteria	2084
34.fa.txt	*Saccharomyces cerevisiae*	784333 (Chromosome)	Eukaryota	348(Verified ORFs)

Domain annotation

Task 1 & 2

Find the Pfam domain organization for the first 100 proteins encoded in your genomes.
1. The way to do this is to use the hmmscan program.
2. Easiest is to run it as

```
# Normally you would download library from Pfam
wget ftp://ftp.ebi.ac.uk/pub/databases/Pfam/current_release/Pfam-A.hmm.gz
# Each Pfam file is described by release notes
# ftp://ftp.ebi.ac.uk/pub/databases/Pfam/current_release/relnotes.txt
# Than you decompress is and prepare it for hmmscan tool
gzip -d Pfam-A.hmm.gz
hmmpress Pfam-A.hmm
# We have done that for you already, and the Pfam/ files are provided in
course directory
# Run the hmmscan
hmmscan --cut_ga --acc <hmm database file> <query protein file>
# Where hmm database file is Pfam-A.hmm and query protein file is your
proteome in multi-fasta format
```

Done using the above instructions

(a) What do the options mean?

```
hmmscan --cut_ga --acc <hmm database file> <query protein file>
--cut_ga : use profile's GA gathering cutoffs to set all thresholding /
GA thresholds are curated thresholds defining family membership
--acc : prefer accessions over names in output /
Trunicates the name to just the accessions
```

(b) As database file, use Pfam/Pfam-A.hmm. This may take a while.
Done using the above commands

(c) To parse the results, you can use the command

```
cat hmmscan-output.txt | perl hmmscan_parser.pl
```

Done using above command

Task 3

What kind of output does this give?
The output consists of the fasta header for that particular sequence combined with ID of the best matched sequence in PfamA database, along with the statistical values.

Task 4

Using the pfam2go map (http://geneontology.org/external2go/pfam2go), assign gene ontology terms to each of the genes.
Done using script:
`https://github.com/SiddharthTomar/ComparitiveGenomics/blob/master/Lab7/GeneOntology/goparser.py`

Task 5

Do the results from what you ran above differ from simple BLAST (hypothetically domain sequence vs proteome file)? How, why and to what extent?
Essentially, both HMM based search and text based BLAST search use different algorithms, and will yield different results accordingly. BLAST will categorize according the similarity between sequences and support it with statistical score, and will order them according to highest similarity. On the contrary, HMM based profiles of sequence against PFam HMM profiles for protein families will yield a more "generalized" overview and search, where each sequence is assigned to the large profile umbrella in PFam. For example, if the above pipeline uses BLAST as search algorithm, the gene ontology terms associated with the best hit will be used, which may give more specific functional annotation or no annotation at all, depending on the completeness of database. But PFam will give the closest family associated with input sequence, which will at least give a rough general overview, if nothing else.

Whole-proteome analysis with Phobius

Task 7

Now we want to run Phobius for all proteins in a genome. This assumes you have a fasta file with all proteins in each proteome from previous practicals. We need a script that launches Phobius for each sequence, and parses the output of Phobius. This is a very typical script in bioinformatics so it is a very general exercise. All we want to collect for now is the number of predicted signal peptides and TM (predicted transmembrane segments) segments for each protein, and find out for one proteome:

(a) The fraction of proteins with 0 TM segments.

(b) The fraction of proteins with > 0 TM segments.

(c) The average number of TM segments for those with > 0 segments.

(d) The fraction of proteins with > 0 signal peptide.

(e) The fraction of those (with > 0 signal peptide) with > 0 TM segment.

Done using script:
`https://github.com/SiddharthTomar/ComparitiveGenomics/blob/master/Lab7/Predictions/plot.py`
Output (for all proteoms in descending order):

```
The fraction of proteins with 0 TM segments in sequence of input file :
[693, 3316, 4600, 1599, 221]
The fraction of proteins with > 0 TM segments in sequence of input file :
[208, 802, 1230, 408, 66]
The average number of TM segments for those with >0 segments in sequence of input file :
```

```
[4.197115384615385, 4.339152119700748, 5.394308943089431, 4.654411764705882, 4.0606060606060606]
The fraction of proteins with > 0 signal peptide in sequence of input file :
[138, 776, 1303, 252, 16]
The fraction of those (with > 0 signal peptide) with > 0 TM segments in sequence of input file :
[30, 149, 273, 49, 5]
```

Comparative proteome analysis with Phobius

Task 8

Now run the previous analysis on all your (real) genomes. Make an xy scatter plot showing the fraction of TM proteins on one axis and the average nr of TM segments on the other axis. Is there a trend?
Please refer to Figure 1 for the plot. Yes, there is a increasing trend between TM element size and TM protein number, except for one proteom.

More protein localization analysis with targetP

What does Plant/Non-Plant parameter do ?
Non-plant parameter limits the prediction to mitochondrion, secretory pathway, and other. Plant parameter adds chloroplast as one of the possible location.

(a) What fraction are predicted mitochondrial proteins ?
0.087

(b) How many are both predicted mitochondrial and to have a signal peptide?
None. The biological accuracy of these predictions depends on the assumptions made and the dataset used to create the tool. Therefore the accuracy of TargetP is limited by organisms used to train it and the given library of *presequences*. Thus like any other tool which works on pre-existing data, TargetP will work optimally only on organisms which are evolutionary less distant.

Task 10

Would your run targetP for all of your genomes ? Why ?
No, I won't use targetP for all of my genome, since the algorithm is specifically designed for Eukaryotic genomes, and we also have Prokaryotic genomes in our dataset.

GIT

Additional files can be found in the address below:
`https://github.com/SiddharthTomar/ComparitiveGenomics/tree/master/Lab7`

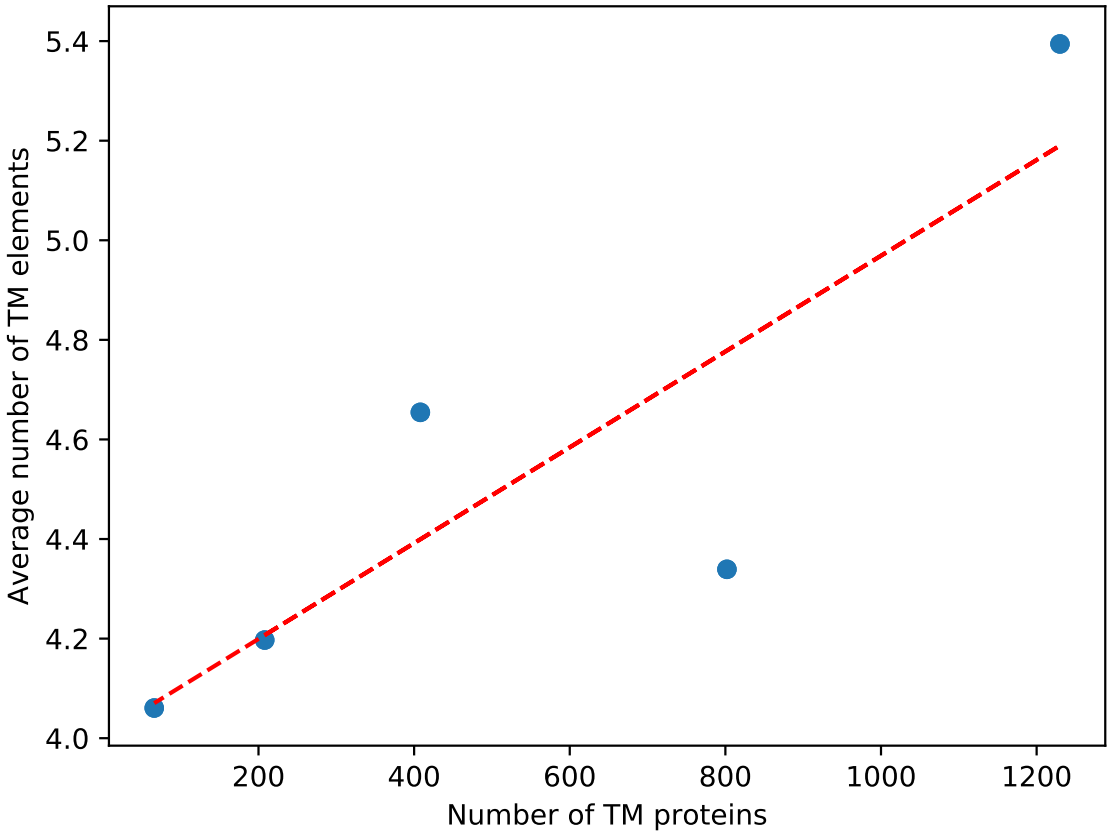

Figure 1: Number of TM proteins vs TM segments