Homework Exercise 5

1)

It is claimed that the **ring finger protein (C3H2C3 type) 6** human protein has a homolog in the **Herpesviridae** viral family. Find the most likely candidate to be that homolog.

If you want to make the search go faster, you may limit the search to strains that infect human.

Do you feel convinced that you have found a true homologue? In general, how can virus-human homologues come about?

2)

Researchers at the Institute of Pointless Studies have recently sequenced a meta-genomic sample taken from the depths of the Atlantic Ocean. They compared DNA sequences of that sample to the genomes of human and *E. coli*. They found 98% of the sequences to match better to the human genome than to the *E. coli* genome, so they concluded that the sample is more likely to have originated from a eukaryote than a prokaryote genome.

Do you think this conclusion is justified? Why or why not? What additional information or analysis might be required for accepting or rejecting it?

3)

The relative mass of a protein is defined by its molecular mass divided by the protein length (i.e. the average mass per residue). The frequency of a secondary structure in a given protein is defined by its number of residues with that secondary structure divided by its length.

Which of the three secondary-structures (**helix**, **beta strand** or **turn**) gives the strongest correlation between secondary structure frequency to relative mass in human proteins?

Use **Bio.SeqUtils.molecular_weight** to calculate molecular weight and **scipy.stats.pearsonr** to calculate Pearson's correlation.

Does molecular weight seem substantially associated with the secondary structures of proteins?

If you need to refresh your memory on Pearson's correlation, consider watching this video.

Bonus Questions

1)

A) [**Introduction to command-line interfaces: 1-2 slides**]

What is the command-line interface? How can one run it in the Windows operating system? How is it different than Graphical User Interface (GUI) and what are the benefits of each type of interface?

B) [**Clustal Omega: 1-2 slides**]

We have seen that Biopython's sequence alignment implementation is quite slow. Fortunately, there are many better alignment tools out there. Clustal Omega is one of the popular choices. Download the Clustal Omega tool (http://www.clustal.org/omega/) and the BLOSUM62 matrix in *.mat* format (http://www.inf.fu-berlin.de/lehre/WS05/aldabi/P2/blosum62.mat).

You can run Clustal Omega in command line as follows:

clustalo -i <FASTA_INPUT_FILE> --distmat-in=<DISTANCE_MATRIX_FILE>

where *<FASTA_INPUT_FILE>* is the path of the file with the input sequences to align, and *<DISTANCE_MATRIX_FILE>* is the path of the distance matrix file (e.g. BLOSUM62).

C) [**Python's *subprocess* module & Clustal Omega as an example: the main part**]

How does Python's ***subprocess*** module allow you to wrap command-line tools?

Use it to wrap Clustal Omega into a Python interface by implementing the following function:

```
def run_ pairwise_clustal_omega_with_blosum64(protein_seq1, protein_seq2):
    # TODO call clustalo using subprocess, and return the result of the alignment as a
    # simple Python object
```

IMPORTANT: Implement this function by yourself using the ***subprocess*** module. Don't use an existing Python interface (e.g. Biopython's), although normally it is a better approach to use existing code. Our purpose here is educational: to show that the ***subprocess*** module can be used to wrap any command-line interface on our own (if no existing wrapper exists).

How is the performance of the function you created compared to Biopython's default pairwise alignment we learned in class? Use it to repeat question (1) (looking for a human-virus homolog) more efficiently.