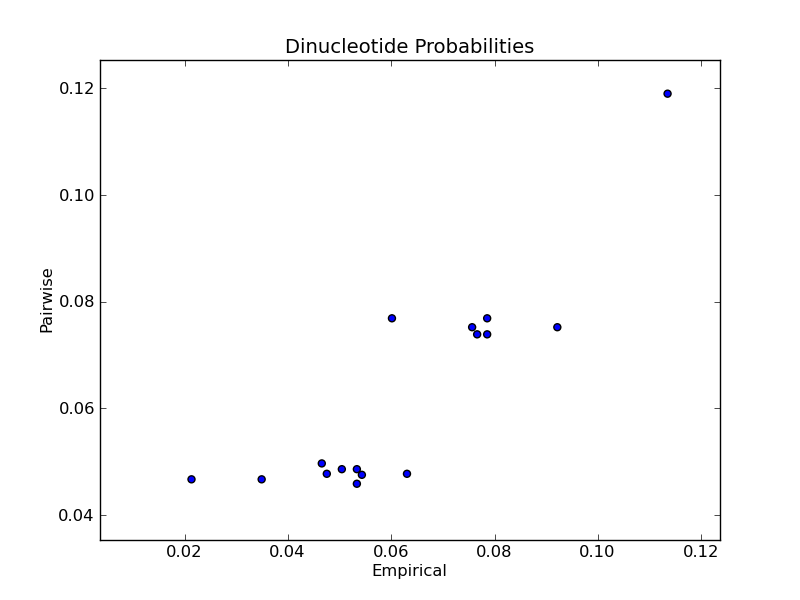
Bioinformatics

Lab 2

1a. This function reads in a filename provided as the argument after the call ot your python script. It then prints out the filename to confirm it obtained it.

1b. The for loop created the auxilary variables seq and header which it will return. It then goes through for each line in the fasta file and adds the lines to the ‘seq’ string. There is a check (which should occur on the first line of a valid fasta file) that pulls the ‘title line’ beginning with a ‘>’ into it’s own string, the auxilary vairable header. It the returns these two variables , header and seq. The strip() function of the string class appears to remove the new line character at the end of the line, I’m sure it can take an argument and modify that behavior to strip what you choose. If I were to remove it then the sequence would be riddled with \n which is not what we expect and would render the squence invalid or unusbale in that state. The [1:] on the string slices it by slicing at the first character removing it. In the case of our method it removes that initial ‘>’ character.

If the argument file does not exist, since we do not account for this exception in our read method, it throws an IOError saying there is no such file or directory.

2. 

I notice that the predicted (pairwise) probability is only really close for a few of the probabilities, the rest seem to have happened more often than predicted.

3. This function could probably be a bit more useful if you used something like an optional variable which defaults to 3 for the dna (maybe translation can occur on more than 3 nucleotides as a key or something, I don’t know. GUG corresponds to the ‘Valine’ amino acid. Evolutionarily using AUG as the start codon makes sense since it’s the single codon that encodes it’s amino acid (Methionine). If GUG were to be used as the start sequence in an organism in such a rare case I don’t see any disadvantages immediately. GUG is not a single codon, maybe it has something to do with that, Valine can be encoded by GUU, GUC, GUA and GUG.

4. It appears to be the Influenza A virus, the translation I produced was this:

MKAKLLVLLCAFTATYADTICIGYHANNSTDTVDTVLEKNVTVTHSVNLLEDSHNGKLCRLKGIAPLQLGNCSVAGWILG

NPECESLFSKESWSYIAETPNPENGTCYPGYFADYEELREQLSSVSSFERFEIFPKESSWPNHTVTKGVTASCSHNGKSS

FYRNLLWLTEKNGLYPNLIKSYVNNKEKEVLVLWGVHHPSNIGDQRAIYHTENAYVSVVSSHYSRRFTPEIAKRPKVRDQ

EGRINYYWTLLEPGDTIIFEANGNLIAPWYAFALSRGFGSGIITSNASMNECDAKCQTPQGAINSSLPFQNVHPVTIGEC

PKYVRSTKLRMVTGLRNIPSIQSR

The translation on the BLAST database is:

MKAKLLVLLCAFTATYADTICIGYHANNSTDTVDTVLEKNVTVTHSVNLLEDSHNGKLCRLKGIAPLQLGNCSVAGWILGNPECESLFSKESWSYIAETPNPENGTCYPGYFADYEELREQLSSVSSFERFEIFPKESSWPNHTVTKGVTASCSHNGKSSFYRNLLWLTEKNGLYPNLIKSYVNNKEKEVLVLWGVHHPSNIGDQRAIYHTENAYVSVVSSHYSRRFTPEIAKRPKVRDQEGRINYYWTLLEPGDTIIFEANGNLIAPWYAFALSRGFGSGIITSNASMNECDAKCQTPQGAINSSLPFQNVHPVTIGECPKYVRSTKLRMVTGLRNIPSIQSR

I made a quick python method to compare the accuracy and included it into my python script. The result was that our translation was 100% accurate when compared to the BLAST sequence.