Haji Mohammad Saleem
260508983

# Comp 598: Assignment 2

RNA bioinformatics

**Collaborators:** Faizy Ahsan

## Exercise 1

α-helices vs β-sheets

α-helices can be predicted with almost 10% more accuracy than β-sheets. The likely reasons for such a difference in prediction is as follows:

- Hydrogen bonding patterns for α-helices are among amino acids in close proximity to each other, and those for β-sheets are not.
- Shorter secondary structure elements are harder to predict, presumably because the signal is not strong enough from these fragments.
- β-sheets have high prevalence of nonlocal interactions between regions of the protein chain that are not necessarily consecutive in the amino acid sequence.

Giegerich, Robert. "Introduction to stochastic context free grammars." RNA Sequence, Structure, and Function: Computational and Bioinformatic Methods (2014): 85-106.
https://www.cs.princeton.edu/~mona/Chapter29.pdf

## Exercise 2

Protein secondary structure prediction

```
>1MBN:A|PDBID|CHAIN|SEQUENCE
VLSEGEWQLVLHVWAKVEADVAGHGQDILIRLFKSHPETLEKFDRFKHLKTEAEMKASEDLKKHGVTVLTALGAILKKKGHHEA
ELKPLAQSHATKHKIPIKYLEFISEAIIHVLHSRHPGDFGADAQGAMNKALELFRKDIAAKYKELGYQG
```

Myoglobin (1MBN chain A) FASTA file – *primary structure*

```
1      VLSEGEWQLV LHVWAKVEAD VAGHGQDILI RLFKSHPETL EKFDRFKHLK
          HHHHHHH HHHHHHHTTS HHHHHHHHHH HHHHH HHHH HT HHHHT

51     TEAEMKASED LKKHGVTVLT ALGAILKKKG HHEAELKPLA QSHATKHKIP
       SHHHHHH HH HHHHHHHHHH HHHHHHTTTT   HHHHHHHH HHHHHTT

101    IKYLEFISEA IIHVLHSRHP GDFGADAQGA MNKALELFRK DIAAKYKELG
       HHHHHHHHHH HHHHHHHH T TTTSHHHHHH HHHHHHHHHH HHHHHHHHHT

151    YQG
```

Myoglobin (1MBN chain A) – *Sequence and secondary structure*

| Amino Acid | 3-Letter | 1-Letter | Helical Penalty |
|---|---|---|---|
| *Alanine* | Ala | A | 0 |
| *Arginine* | Arg | R | 0.21 |
| *Asparagine* | Asn | N | 0.65 |
| *Aspartic acid* | Asp | D | 0.69 |
| *Cysteine* | Cys | C | 0.68 |
| *Glutamic acid* | Glu | E | 0.4 |
| *Glutamine* | Gln | Q | 0.39 |
| *Glycine* | Gly | G | 1 |
| *Histidine* | His | H | 0.61 |
| *Isoleucine* | Ile | I | 0.41 |
| *Leucine* | Leu | L | 0.21 |
| *Lysine* | Lys | K | 0.26 |
| *Methionine* | Met | M | 0.24 |
| *Phenylalanine* | Phe | F | 0.54 |
| *Proline* | Pro | P | 3.16 |
| *Serine* | Ser | S | 0.5 |
| *Threonine* | Thr | T | 0.66 |
| *Tryptophan* | Trp | W | 0.49 |
| *Tyrosine* | Tyr | Y | 0.53 |
| *Valine* | Val | V | 0.61 |

**Table 1:** *Standard amino acid alpha-helical propensities*

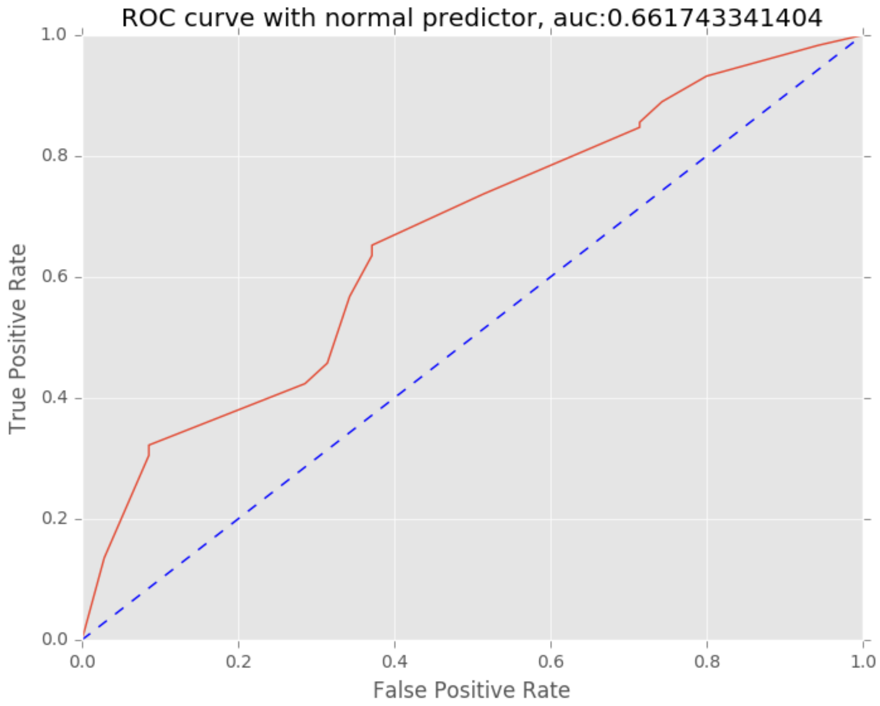
Pace, C. N., & Scholtz, J. M. (1998). A helix propensity scale based on experimental studies of peptides and proteins. Biophysical journal, 75(1), 422-427, Table 3.

Code file: q2.py

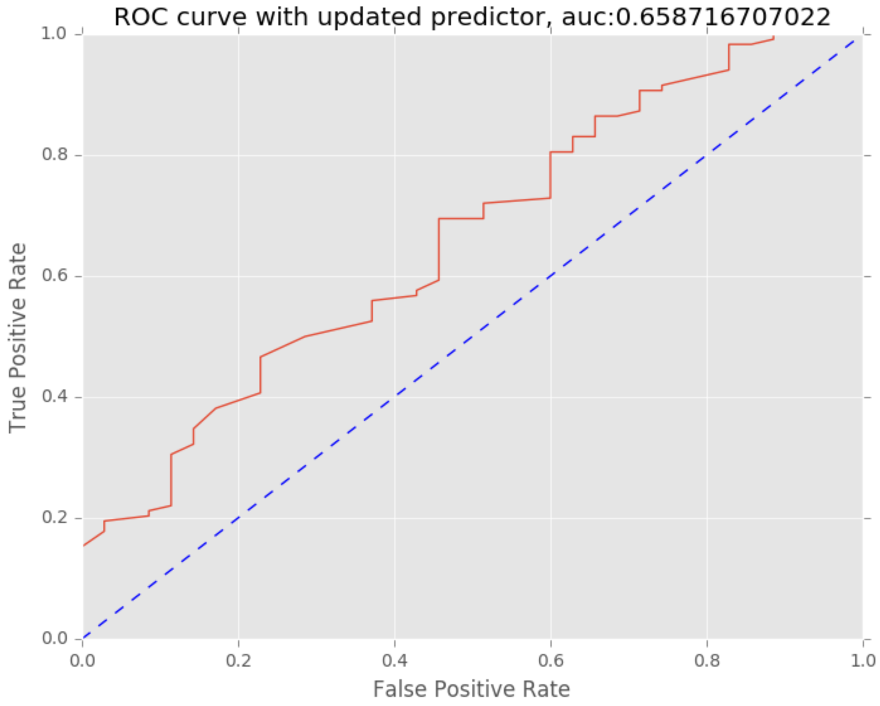
We find that the updated predictor decreases the AUC. We can therefore say that the alpha helix propensity is more a function of the reside than the neighborhood. A low propensity neighbor can cause the overall propensity of the neighborhood to go down, while they could have created the helices when neighbors are not considered. The result is surprising because intuitively, a helix is the result of a neighborhood of alleles and therefore collective propensity should be a factor in helix generation. However, it may not be just simple average and we might have to find a more sophisticated mechanism.

Haji Mohammad Saleem
260508983

## Normal Predictor



## Updated Predictor

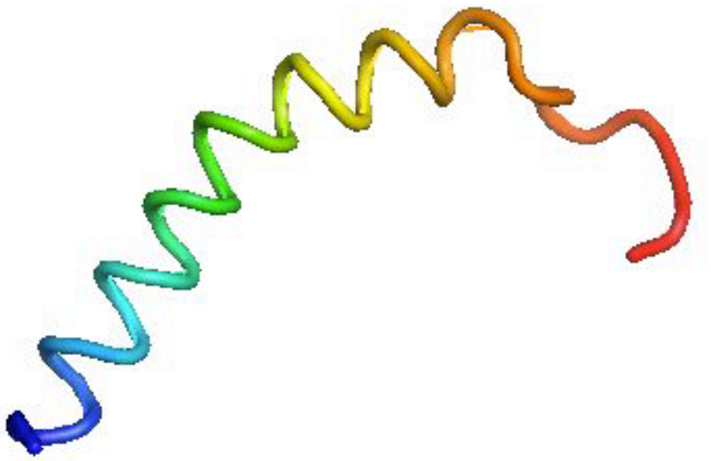


Haji Mohammad Saleem
260508983

# Exercise 3

Molecular Dynamics

## PYMOL Movie



Movie file: pymol.mov

Can also be find at:

EduPyMol_molecule-movie https://www.youtube.com/watch?v=flqxt8XAcb0

### RMSD Graph

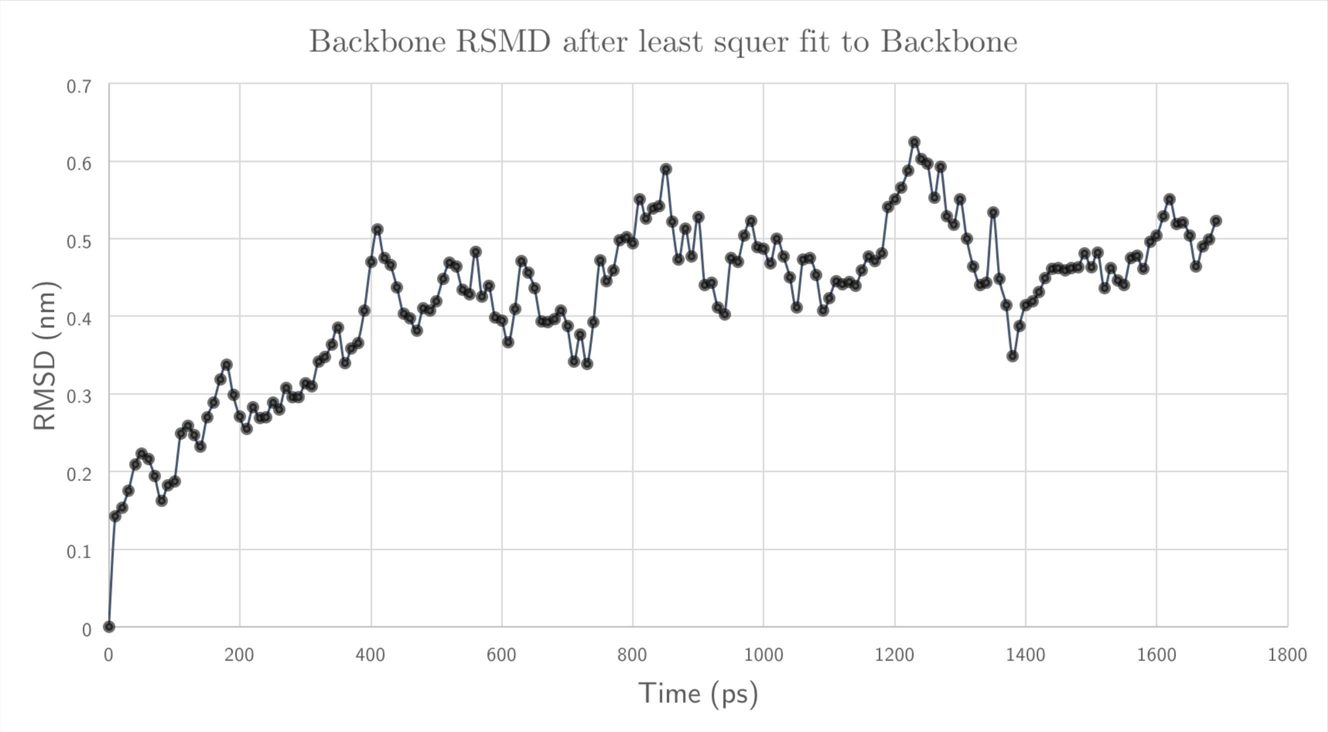


**Figure 1:** *Backbone RMSD after least square fit on Backbone*

## Exercise 4

ISORANK

### Methodology

For every combination of nodes $i \in \mathrm{N}_1$ and $j \in \mathrm{N}_2$, find all the neighbors of $i$ *N(i)* and $j$ *N(j)*. For every combination of nodes $u \in N(i)$ and $v \in N(j)$, we compute the neighborhood size *N(u)* and *N(v)*. Then A[$i$,$j$][$u$,$v$] becomes $1/|N(u)||N(v)|$. The rest of the values are 0.

After the matrix A has been calculated, We calculate the eigenvectors of matrix A. We select one of the rows of the matrix to obtain one set of the solution. For that row, we select the real values and align the pair with the highest value. All the pairs that contain those nodes are then removed and the second highest value pair is selected and so on.

We go on to do this until the remaining values are below a threshold, set in this case as 0.01 or no node pair are left.

Code file: q4.py

Haji Mohammad Saleem
260508983

We use the following version of python for the code:

```
Python 2.7.9 :: Anaconda custom (x86_64)
```

To run:

```
python q4.py <Network File 1> <Network File 2>
```

The output is a list node pairs that have been aligned together. They are provided in the following files: N1N2.nodes N2N3.nodes N1N3.nodes

Based on the number of aligned nodes, we find that N2-N3 are the most similar. We find that 53 nodes aligned for N2-N3 compared to 31 for N1-N3 and 17 for N1-N2.

In case of weighted graphs, we propose the use of *weighted degree* in the generation of matrix A. That is, instead of just computing the size of the neighborhood, the values of the degree would also be taken into consideration.

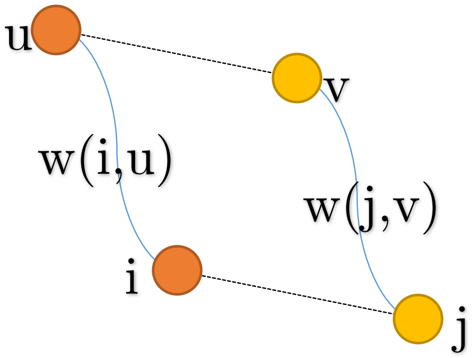
Therefore

$$R_{ij} = \sum\nolimits_{u\in N(i)} \sum\nolimits_{v\in N(j)} \frac{w(i,u)w(j,v)}{W(u)W(v)} R_{uv}$$

where

$$W(u) = \sum\nolimits_{x\in N(u)} w(x,u)$$

$w(x,u)$ stands for the weight of the edge between $x$ and $u$.


u
v
w(i,u)
w(j,v)
i
j