Veronica	*Bedin*	*2097013*

Midterm test No. 2

03 / 05 / 2023

Please complete the following tasks and submit the document in **PDF format** to damiano.piovesan@unipd.it by **12:30 PM on May 17, 2022**, two weeks after the deadline. Please include your **surname** in the **file name**.

Each student has been assigned a different **protein structure (<PDB ID>_<chain ID>)**. The assessment consists of two parts: open questions and an analysis of the assigned structure. You can find the **assignment file here**.

Please answer the following questions concisely, with a maximum of **500 words** in total:

1. What is the relationship between sequence similarity and structure similarity in biological proteins?

   Proteins exhibit different structure/sequence similarity relationships, but those with high structure and high sequence similarity often share functional similarity and evolutionary relationships. Considering the fact that any random pair of natural sequences have at least 15% sequence identity, several studies have shown that proteins with approximately 30% or more identical residues tend to have similar structures. However, for shorter alignments, the threshold for similarity is higher. There exists also a "twilight zone", where proteins with less than 20% sequence identity still have the same fold.

2. What are the main steps involved in homology modelling?

   Homology modeling predicts the structure of a protein (the **target**) by using experimentally derived structures (e.g. from X-ray crystallography, NMR spectroscopy) as **templates**.

   As mentioned earlier, proteins with similar sequences tend to have similar structures. Therefore, this prediction method is applicable when the sequence identity between the template protein and the target protein exceeds 30%.

   The first step is a **database search**, to find homologous sequences with known structure, typically performed using BLAST. Once the template is selected, a pairwise alignment is conducted to align the target and template proteins. A model is then built for the target protein, incorporating information from the template structure, and subsequently evaluated. The model may be accepted or rejected. If rejected, alternative alignments or, in extreme cases, alternative templates will be attempted.

   One of the challenges of homology modeling is the management of the variable regions (generally loops) and in particular of positions near indels because they have to be predicted (e.g. via fragment-based building or restraint-based building).

3. How can you measure the quality of a structural alignment?

   The quality of a structural alignment can be assessed using various measures and scoring methods. Here are some commonly employed approaches:

   - **Root Mean Square Deviation (RMSD)**: RMSD measures the average distance between corresponding atoms (e.g. Alpha/Beta Carbons) in the aligned structures. A lower RMSD value indicates a higher quality alignment.
   - **TM-score**: TM-score is a structural similarity measure that ranges from 0 to 1, with higher values indicating better alignment. It weighs the residue pairs at smaller distances relatively stronger than those at larger distances. TM-score is used as an alternative to RMSD because a small number of local deviations could result in a high RMSD even if the global topologies of the compared structures are similar.

4. What are the differences, in terms of amino acid composition, between globular and intrinsically disordered proteins?

   IDPs (Intrinsically Disordered Proteins) have a higher content (compared to globular proteins) of disorder-promoting amino acids, such as glycine, proline (known as "structure breakers"), glutamine, and serine, compared to globular proteins. These residues are more flexible and lack specific secondary structure preferences.

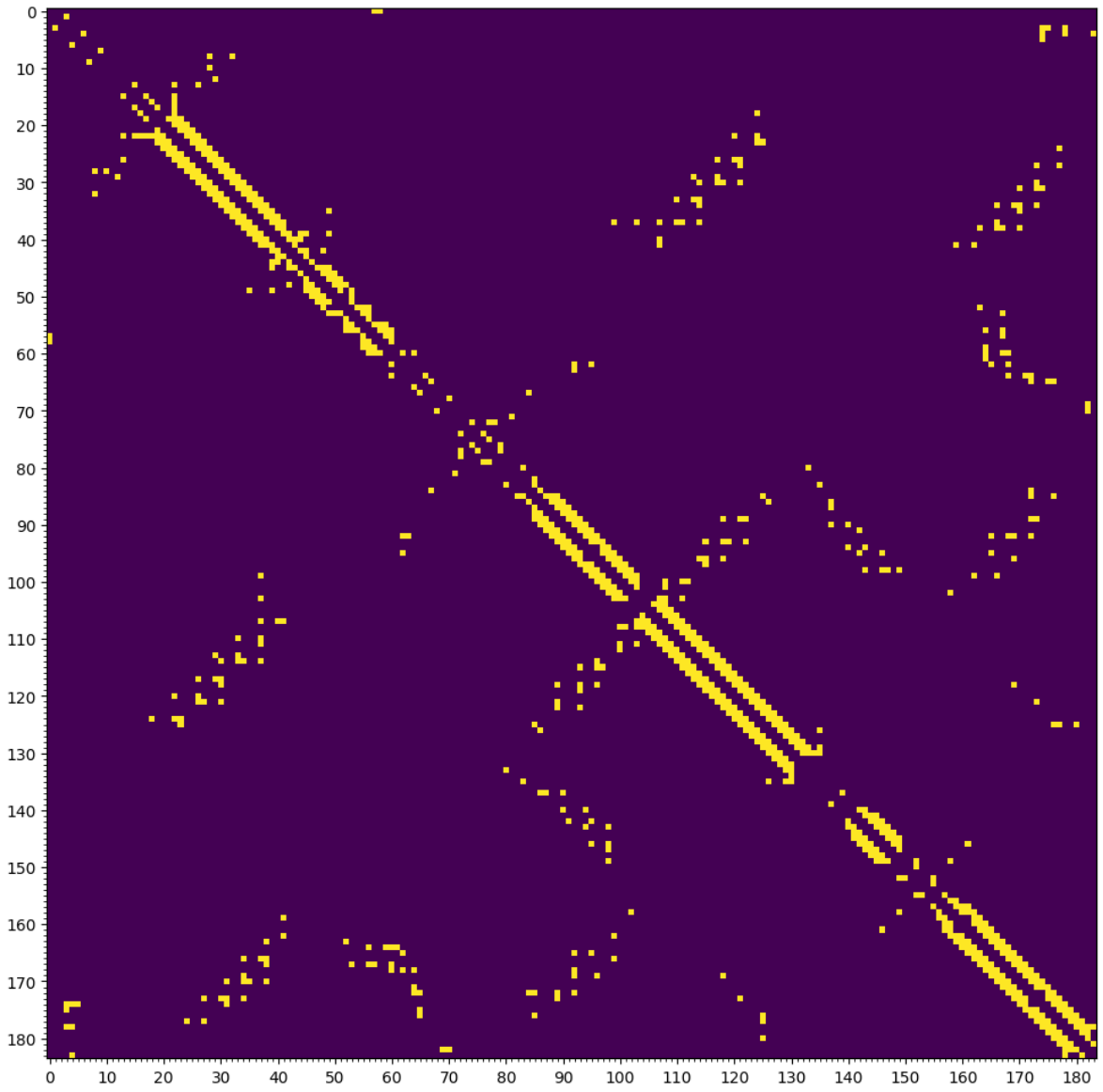
   On the other hand, globular proteins typically have a balanced distribution of hydrophobic and hydrophilic amino acids, which contributes to their spherical conformation.

   A good way to distinguish globular proteins from IDPs is hence to evaluate the mean hydrophobicity and the mean net charge (properties deriving from the amino acids composition): IDPs are characterized by small and hydrophilic amino acids.

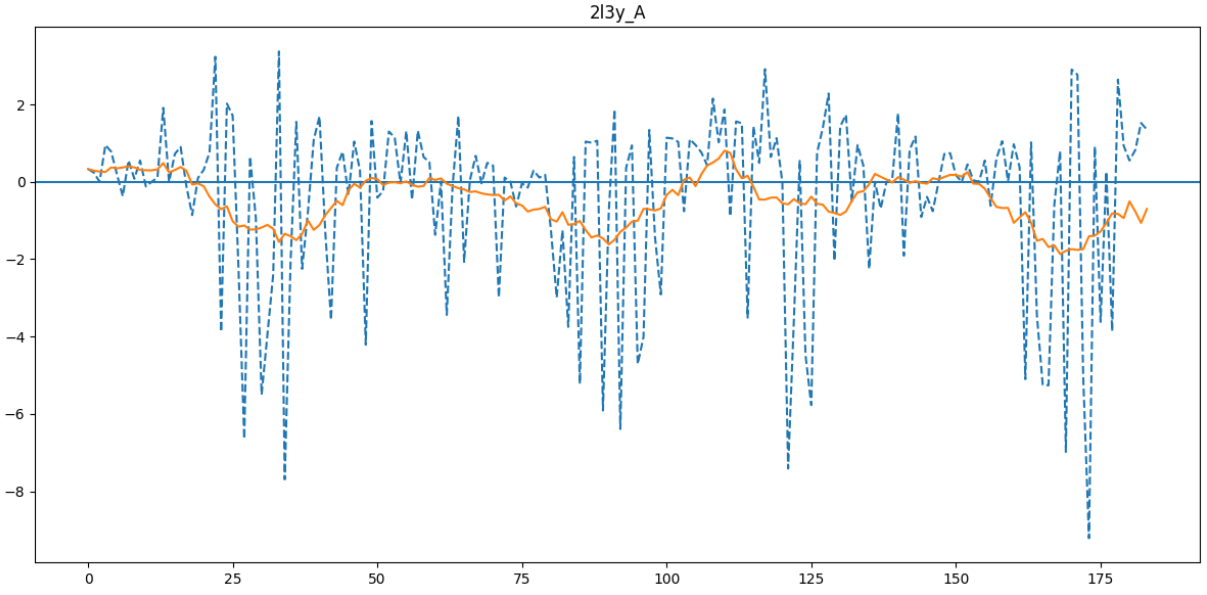
Download the assigned PDB structure and consider only **standard (non-hetero) residues** of the specified chain (<PDB ID>_<chain ID>). Calculate the contact map (question 1) and the conformational energy (questions 2 and 3) as described in the IUPRED paper. The M and P matrices are available from the ***iupred_data.py***. The smoothed energy is the moving average of the raw energy over a window of 21 residues (±10 residues around the current position).

Complete the following tasks:

5. Calculate and plot the contact map of your chain. Use the ***NeighborSearch*** module and the ***search_all(3.5, level="R")*** method. Consider only contacts between positions with a **sequence separation ≥ 2**.



6. Calculate the **exact energy** of each residue based on the weighted contribution of its **contacts** (as calculated above) and plot the raw and smoothed energy for each residue on the same figure. Use the ***M matrix*** to calculate the contact energy.

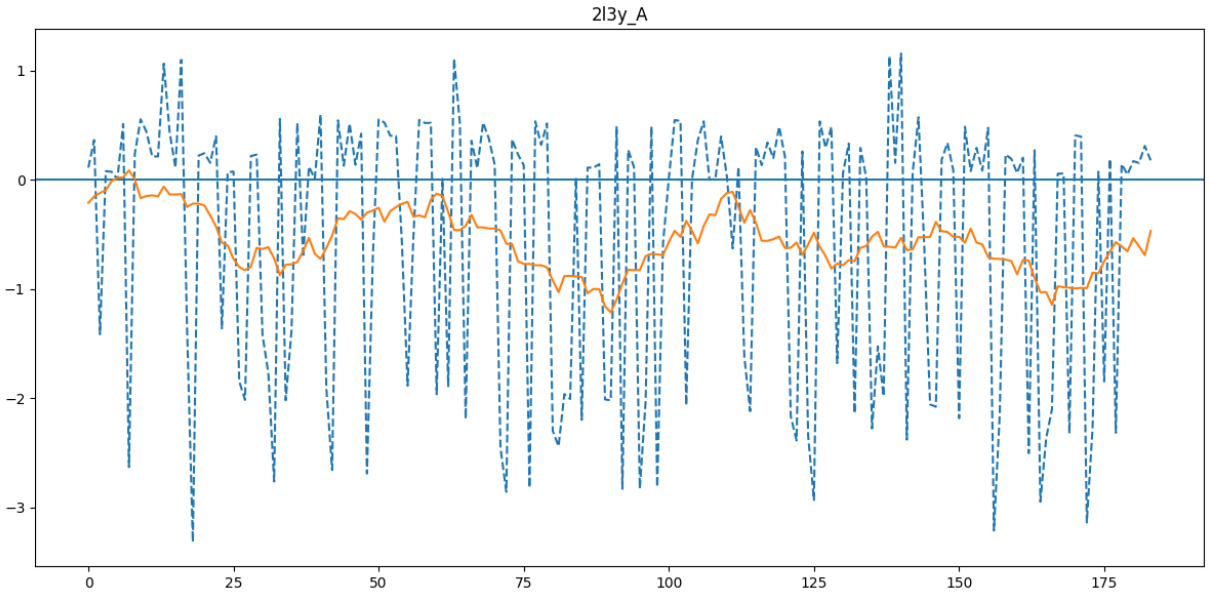


```
residues = [residue for residue in structure[0][chain_id] if residue.id[0] == " "]
pred_m   = []

#Transform sequence into indexes
seq      = "".join([seq1(residue.get_resname()) for residue in residues])
print(seq)

raw_count=0

# Calculate the exact energy of each residue based on the weighted contribution of its contacts
for i in range(len(seq)):
  energy = 0
  for j in range(len(seq)):
    if i!=j:
      energy += m_matrix[aa_list.index(seq[i])][aa_list.index(seq[j])]*contact_map_nb[i][j]
  if energy > 0:
    raw_count = raw_count+1
    fraction  = raw_count/len(seq)
  pred_m.append(energy)

print("raw count :", raw_count, "disordered count :", fraction)

window_size         = 100
window_size_smooth  = 10
pred_smooth         = []
sequence_separation = 2
indices             = [aa_list.index(aa) for aa in list(seq)]

# Get the slice i-100/i+100 excluding adjacent positions (+/-1)
start_before = max(0, i - window_size)
end_before = max(0, i - sequence_separation)
start_after = min(len(indices) - 1, i + sequence_separation)
end_after = min(len(indices) - 1, i + window_size)
indices_local = indices[start_before : end_before] + indices[start_after : end_after]

# Smooth the prediction (moving average)
for i in range(len(pred_m)):
    frag = pred_m[max(0, i - window_size_smooth) : min(i + window_size_smooth, len(pred_m))]
    pred_smooth.append(sum(frag) / len(frag))
```

This snippet of code contains also the disorder content calculation (**raw_count** and **fraction**). The same implementation for the disorder calculation will be used for the P matrix.

7. Calculate the **estimated energy** of each residue based on the weighted contribution of the **frequency of neighboring amino acids** in the sequence and plot the raw and smoothed energy for each residue on the same figure. Use the ***P matrix*** to calculate the estimated energy. Neighboring residues are those 2-100 residues apart from the current position.

   Here the**IUPRED ad-hoc** implementation was used.



8. Report the **disorder content** for the two different calculations. Disorder content can be calculated as the fraction of **residues with positive energy** (≥ 0) over the length of the sequence. Please report both the fraction and the raw count of residues with positive energy.

8.1. **M:** raw count : 104 disordered count : 0.5652173913043478

8.2. **P:** raw count : 117 disordered count : 0.6358695652173914