

# Assignment 1 – Protein Structure and Visualization

BIOE/BIOMEDIN/BIOPHYS/CME/CS 279

Due: October 12, 2017 at 3:00 PM

The goal of this assignment is to familiarize yourself with the visualization software PyMOL and the basics of protein structure.

**Acknowledgements:** Portions of this assignment are based off of the PyRosetta Tutorials.

## 1 Preliminaries

- Download `assn1.zip` from the course website.
- Download and install PyMOL (<https://www.pymol.org/>) on your computer. If you cannot get PyMOL to run on your computer, please contact the TAs.

## 2 Visualizing Proteins

To start, we'll use the program PyMOL to visualize the molecular structure of some peptides and folded proteins. First, unzip the contents of `assn1.zip`, then open PyMOL. PyMOL allows you to execute instructions through their GUI or through their built-in command line.

First, in the command line, change to the assignment directory.

```
PyMOL> cd <path to assignment directory>/pds
PyMOL> load hras.pdb
```

When you load a structure, the default visualization is an all-atom depiction. While there are times when this is necessary/helpful, frequently, we want a coarser view of the protein. In particular, the `cartoon` view allows us to quickly learn about the secondary structures that are present in the folded 3D conformation.

To access this mode, we will hide the current depiction and then show the cartoon depiction. These commands are available through the GUI buttons in the right-most panel. In the row of the molecule we wish to manipulate (in this case `hras`), select the following commands.

H → everything      S → cartoon

If you prefer, you can also control the depiction from the command line.

```
PyMOL> hide everything, hras
PyMOL> show cartoon, hras
```

In this visual mode, alpha helices are depicted by twisting coils and beta strands are depicted as fat arrows.

**Question 1:**

(a) *How many sections of hras are folded as an alpha helix?*

(b) *How many strands of hras are folded as a beta strand?*

*Note: You may find it useful to color by secondary structure.*

C → by ss → Helix Sheet Loop

Alternatively, you can use the command:

```
color red, ss h; color yellow, ss s
```

## 2.1 Backbone Geometry

Next, we will take a closer look at the typical secondary structures, the alpha helix and beta sheets. Start by reinitializing PyMOL and then running the following commands.

```
PyMOL> reinit
PyMOL> load helix.pdb
PyMOL> hide everything
PyMOL> show cartoon
```

**Question 2:** *Looking down the helix and moving away from the viewer, which direction is the helix coiled (clockwise or counterclockwise)?*

The geometry of an alpha helix (and a beta strand) refers to a specific orientation of the atoms in the polypeptide backbone. In general, the orientation of the backbone atoms tells us a lot about the overall structure of the protein. To select just the backbone atoms, execute the following commands.

```
PyMOL> hide everything
PyMOL> select bb, name c+o+n+ca
PyMOL> show lines, bb
```

Note: “bb” is not a keyword in this command. The PyMol selection syntax is

```
select <name to be give selection>, selection
```

Here we are selecting all atoms named ‘n’, ‘o’, ‘c’, or ‘ca’ which are the standard names for atoms in the backbone.

You should see the backbone atoms represented as connected lines, with carbons in green, oxygens

in red, and nitrogens in blue.

The backbone geometry can be succinctly described by the *dihedral angles* (also known as torsional angles). In general, a dihedral angle is an angle between two planes. In the context of backbone molecular geometry, we are concerned with the angle between atoms while “looking down” a bond.

**Question 3:** *We will be looking at residue 159. Which amino acid does this correspond to?*

(a) *Compute the dihedral angle  $\phi_{159}$  (between  $C_{i-1} - N_i - C_i^\alpha - C_i$ ).*

(b) *Compute the dihedral angle  $\psi_{159}$  (between  $N_i - C_i^\alpha - C_i - N_{i+1}$ ).*

(c) *Compute the dihedral angle  $\omega_{159}$  (between  $C_i^\alpha - C_i - N_{i+1} - C_{i+1}^\alpha$ ).*

*Select the atoms in the correct order, and then use the following command to compute the appropriate dihedral angles.*

```
PyMOL> get_dihedral (pk1), (pk2), (pk3), (pk4)
```

*Note: Use the `pkAt` mouse command (by default double-clicking with the right click).*

*Alternatively, PyMol has a powerful atom selection language. You can select by residue number using*

```
PyMOL> select resid 159
```

*and by atom name using*

```
PyMOL> select name ca
```

*Combining these into one command selects the  $C^\alpha$  in residue 159 using*

```
PyMOL> select resid 159 and name ca
```

*or the equivalent short form*

```
PyMOL> select 159/ca
```

*You can then combine these selections to compute a dihedral*

```
PyMOL> get_dihedral resid1/name1, resid2/name2, resid3/name3,  
resid4/name4
```

Due to the chemistry of the peptide bond between  $C_n - C_{n+1}$ ,  $\omega$  is generally very close to  $180^\circ$  (and rarely,  $0^\circ$ ).

Next, perform the same computations for the beta sheet segment (contained in `pdbs/bstrand.pdb`).

**Question 4:** *Are the beta-strands running parallel or antiparallel to one another?*

**Question 5:**

(a) Compute  $\phi_{41}$ .

(b) Compute  $\psi_{41}$ .

(c) Compute  $\omega_{41}$ .

Now that we have computed the dihedral angles for two amino acid residues, it is natural to ask whether these measurements are typical. In fact, one way of characterizing proteins' secondary structures is by their  $\phi$  and  $\psi$  angles.

In particular, one way to represent the distribution of backbone dihedral angles is a Ramachandran plot. Ramachandran plots are two-dimensional heat maps, which represent the number of residues present for a given  $\phi$  and  $\psi$ . Note that the  $\omega$  angle is excluded from these plots as it is constrained to be close to  $180^\circ$ .

**Question 6:** *Generate a Ramachandran plot for the alpha helices (saved in `helices.pdb`), the beta strands (saved in `bstrands.pdb`), and full hras protein. Use the RAMPAGE server to generate the plots (<http://mordred.bioc.cam.ac.uk/~rapper/rampage.php>). Identify the regions in the plot for hras that correspond to alpha helices and beta strands.*

*Note: The plot of interest will be the first one at the top. Your input data corresponds to the black marks, whereas the shading corresponds to the angles observed in most structures.*

## 2.2 Hydrogen Bonds

A *hydrogen bond* is an attractive force between molecular dipoles involving hydrogen. While the hydrogen atoms are not drawn in the backbone (because they were not resolved in the experimental structure), there is a partially positive hydrogen bonded to (almost) every nitrogen involved in the peptide backbone, which tends to be attracted to the partially negative oxygens in the backbone. To visualize these “bonds”, we can measure the lengths between groups of atoms, specifically where polar contacts occur. First, we will add hydrogens into the structure in PyMOL and then show the backbone H-bond distances. Reinitialize PyMOL, and load `helix.pdb` from the `pdbs` subdirectory in the `assn1` folder. You should hide everything, then enter the following commands.

```
PyMOL> select bb, name c+o+n+ca
PyMOL> show lines, bb
PyMOL> h_add
PyMOL> distance hbonds, name o, name n, mode=2
```

Note: “hbonds” is not a keyword in this command, i.e. you could replace it with “foo” and see the

same results. The specifier `mode=2` selects polar contact distances.

**Question 7:** For this question, please use *helix.pdb*. What is the average length of H-bonds in this alpha helix (round to the nearest tenth)? What is the **integer** offset between residues that are interacting (ie if two residues share a H-bond in an alpha helix, what is their positions in the protein backbone relative to each other)?

*Note: By default, the distance command has an extremely liberal hydrogen bond angle cut-off. Among other constraints, the hydrogen bond donor (the atom the hydrogen is covalently bound to), the hydrogen, and the hydrogen bond acceptor should be approximately in line. Please ignore interactions not meeting this criteria.*

**Question 8:** Next use *bstrand.pdb*. What is the average length of H-bonds between these two beta-strands? Does the average H-bond length differ significantly if the strands are parallel or anti-parallel (hint: use the *bstrands.pdb*)?

## 2.3 Statistics of non-bonded interactions

As you saw (or will see) in lecture, a major class of energy functions, called knowledge-based or statistical potentials, are typically created using patterns observed in solved crystal structures, as opposed to physics-based models. These potentials rely on the simple hypothesis that, on average, atoms in solved structures will be positioned in energetically favorable conformations. For example, earlier in this problem set, you saw that the  $\phi$  and  $\psi$  angles often favor particular values. This information could be used to create an energy function that favored combinations of angles that are observed most often.

In this problem, we will examine statistics of specific non-bonded interactions. In particular, we will generate statistics about the distance between positive and negatively charged residues as compared to positive and neutrally charged residues. We have provided a small set of randomly selected structures from the PDB to use as training data, as well as code to parse the files. While our analysis will lack the rigor and completeness needed to create the most widely used knowledge-based potentials, we hope that this problem will show you the intuition behind knowledge-based

potentials and serve as a gentle introduction to Python programming.

**Question 9:** *Open the file `nonbonded_distances.py` in your favorite text editor (think `emacs`, `vim`, `sublime text`). Examine the contents of the file, first concentrating on the code and comments about overall workflow at the bottom, then moving on to the functions at the top. The only thing you need to do is fill in the function “`get_minimum_distances`”. When you are ready, you can run the script by opening your terminal, changing to the `asn1` directory and executing*

```
python nonbonded_distances.py nonbonded_pdb/*
```

*The script will create a file “`nonbonded_dists.png`” containing a histogram of the minimum distance from each positively charged nitrogen in arginine or lysine in the set of structures to the closest negatively charged oxygen in aspartate or glutamate, as compared to the neutrally charged terminal carbons in valine and leucine. Include this histogram in your submission.*

*Do the results agree with your intuition?*

### 3 Feedback

You will receive full-credit for this portion for providing any response. We encourage constructive criticism.

**Question 10:** *What was your favorite aspect of the assignment? What was your least favorite aspect of the assignment? Why? Any suggestions for improvement?*

**Question 11:** *Approximately how long did this assignment take you? Where did you spend most of this time?*

### 4 Submission Instructions

Please submit this homework on Canvas. Let the course staff know if you have any issues.