Introduction to Bioinformatics – AS 250.265
Laboratory Assignment 3

Few things are more exciting to a structural biophysicist than solving and publishing a novel crystal structure. Although solving our own protein structures is beyond the scope of our class, we do have the tools to analyze existing structures and examine how proteins perform some of the chemistry they perform. In this lab, we will explore some of the tools for predicting and visualizing molecular structure. Since the protein structure prediction often takes a few minutes to a few hours to complete, we will only submit our structures in this lab—you will work with these predicted structures in your homework assignment. Most of the lab will be spent teaching you about molecular visualization using the program PyMol.

This lab will be a large part of your assignment for the coming week. You will submit snapshots of various protein structures as well as the answers to the questions from this lab (which are highlighted in gray). All image files should be uploaded to the course web server under assignment six, but you may submit the answers to the lab questions either digitally or by hand at the start of class on March 31. Your answers from this lab will count as 20 points toward assignment number six. Then, the first part of assignment six will be constructing images using the skills you learn here.

Part 1: Structure Prediction Servers

In this part of the lab, we will be submitting a sequence to three different structure prediction servers. In your assignment you will compare these structures to the actual structure from which the sequence was generated.

The sequence you will use for this part of the assignment is a 72 amino acid protein that was derived from the PDB ID 1WM3, a structure of a human gene called SUMO-2. The sequence from this PDB entry was varied according to the PAM Markov model and a new sequence was generated with approximately 40% sequence identity to the original sequence. You can find this sequence on the laboratory three web site. It is called `protein.fasta`.

The SWISS-MODEL Protein Server

The SWISS-MODEL server is a homology modeling protein server. That is, it finds structures that are homologous to your own structure using its internal equivalent of BLAST or PSI-BLAST and attempts to construct a three-dimensional model by assuming aligned sequences will have similar structures.

Visit the SWISS-MODEL server at `http://swissmodel.expasy.org/`. At the left, choose "First Approach Mode." This is a rough mode that can be refined later on using the "Optimize Mode," although you don't have to do this. Enter your email address, fill in the requested information, and paste the sequence from your FASTA file into the server's sequence input box. As we will be looking at the results in PyMol, scroll down into the options and choose to receive "Normal" output—a PDB file with a log of what was done. Finally, submit your request. When the prediction is complete, the result files will be emailed to you.

The 3D-JIGSAW Server

3D-JIGSAW, like SWISS-MODEL, is a homology-modeling server. It provides the ability to perform an interactive modeling session, allowing you to identify domains and view homologous templates as you go along. Here, we will simply use the automated option.

The 3D-JIGSAW server is located at `http://www.bmm.icnet.uk/servers/3djigsaw/`. When you visit the site, click on the "Submission" link. As before, enter your email address and the sequence from the FASTA file, and click submit. The server will begin calculating its structure.

Secondary Structure Prediction

As a final test, submit your sequence to the secondary structure prediction server at Pôle BioInformatique Lyonnais. This web server applies several secondary structure prediction methods to your sequence that you will be able to compare to the actual structure as well as the three-dimensional structures that you get from SWISS-MODEL and 3D-JIGSAW. The sever is located at `http:// pbil.univ-lyon1.fr/`, and then you should select "Secondary Structure Prediction" from the column on the right.

Select PHD from the list of secondary structure prediction algorithms, and enter your sequence. PHD is a secondary structure technique that uses known secondary structures in proteins to predict secondary structures in unknown proteins. It is not a homology modeling technique in that no homologues of your sequence are identified using BLAST—instead, the information is predicted by "training" a neural network to look for patterns that often times are helices, strands, etc.

The result will be displayed immediately. Helices will show up in blue underneath your original protein sequence, strands will be displayed in red, and coil will be displayed in orange. Predictions where the algorithm is most confident will appear in capital letters. The plot at the bottom of the web page indicates the confidence of the algorithm in a chart format for each residue.

Because you will not analyze this information until assignment two, you should simply note the procedure you used to obtain PHD predictions. You can return to this website later when you are compiling your answers to the homework assignment.

Part 2: Obtaining PDB Files and Using PyMol

Once an experimentalist has determined a protein structure, tools are needed for the community at large to visualize and interpret the structure. While there are many such tools that have been developed for this cause, we will only focus on one here. PyMol is a visualization package that is very flexible and also produces publication-quality images. This is unique, as many programs are either good at analysis or display, but not both.

Before we begin, we must download the PDB files we need from the PDB website. This website is located at `http://www.rcsb.org/pdb/`. Visit this site, and search for the PDB ID 1RBP, retinol binding protein. Take a minute to browse the information on the main page.

a. What are the class, fold, superfamily, and family names for the SCOP classification of retinol binding protein (RBP)? What are the CATH class, archtitecture, topology, and homologous superfamiliy names? (4 points)

In another browser window, visit the SCOP website (`http://scop.berkeley.edu/`) and browse through the hierarchy to the RBP family level. You should see a list of different domains of proteins here, one of which should simply be "Retinol Binding Protein."

b. How many PDB entries are there for retinol binding proteins in humans? What are they? Recall that PDB accession numbers are a string of four letters and numbers. (3 points)

Return to your original browser window, and on the menu at the left select "Download Files → PDB File." Save the PDB file somewhere to your local account where you can access it later.

PyMol should be installed on the Jenkins 122 Macs under the applications directory. Open up Mac PyMol and load the PDB file you just downloaded from the website (File → Open …) The molecular structure will appear in the large black window.

Wait here for a brief overview of using the PyMol application.

Practice rotating the structure and changing your viewpoint. You can do this by left-clicking within the black region and dragging in various directions. You can zoom in and out by holding the right mouse button and moving the mouse. Finally, you can translate the molecule by holding down the middle mouse button and moving the molecule around. Because there are 6 degrees of freedom in real space and only two for your mouse, it isn't always obvious how moving the mouse will rotate your structure.

c. While keeping the structure stationary, move the scroll wheel on your mouse down several rotations. You can undo what you've just done by moving the wheel back up. What is this doing to the structural representation you see? How might this functionality be useful for large proteins? (Hint: This functionality is often called the "slab.") (4 points)

Selecting Atoms

One of the more powerful features of PyMol compared to other editors involves its command-line interface. Through this interface, it is possible to select different atoms that can then be rendered in many different formats. For example, type the following command into the PyMol command line (at the bottom of the display area):

```
show spheres, resi 1-10 and not resn RTL
```

You should see that the first ten residues in the structure are displayed as spacefilling spheres (also sometimes called CPK rendering). As you might have guessed, the first part of the command tells PyMol to render the second part of the command as spheres. The second part of the command selects the first 10 residues, excluding the retinol molecule. Here `resi` stands for "residue index," and `resn` stands for "residue name." You can hide the spheres you just displayed by typing:

```
hide spheres
```

The default selection (that is, if no comma is given after the `show` or `hide` command) is the entire molecule. Thus, you can quickly hide representations by leaving off the selection. This is generally true for the other commands we will learn, too.

For a list of selection options you can use within PyMol, type `help selection`. The molecular image will disappear, and you will see a listing of all the selection syntax currently available in PyMol. The top three are the ones we will be most concerned with, but some others are useful, too. `all`, for example, selects the entire molecule, and is useful for the final selection listed `within`. So, for example, the following command would show spheres for all the atoms within a 5 angstrom radius of residue 115. An angstrom is a distance unit equivalent to 10^{-10} m.

```
show spheres, all within 5 of resi 115
```

Try the command above out to see what the effects are (and to get the feel for how large the protein is, in angstroms. You can press "escape" to toggle between text and graphical modes.

d. If you were to model retinol binding protein as a sphere, what would the spherical radius be, based on the sphere of spheres you are looking at now with a radius of five angstroms? What is the volume of that sphere in cubic meters? (3 points)

Another useful selection string is `hetatm`. This selection string typically matches everything that isn't protein in the structure—the "hetero atoms." In our case, this matches the retinol as well as the red dots around the protein, which are water molecules crystallized with the protein structure.

Saved Selections

For many applications, it is often useful to be able to save the selections you create for later use. This way you don't have to continue to type the same thing over and over again. The following command saves the selection we used above to the selection name "subunit1."

```
select subunit1, all within 5 of resi 115
```

After entering the above command, you'll notice a new item appears on the list of objects at the right of your display. You'll also notice that little pink dots appear of the atoms that are selected in that particular selection. The new item is called "subunit1" and the dots should correspond to the atoms that are already shown as spheres in your molecule. Clicking on the gray box corresponding to "subunit1" will toggle whether the selection is displayed. You can also use this approach to display or hide the molecule, which is also represented as a box at the right (1RBP).

Since having the selection dots is often annoying and gets in the way, click on the selection box and hide it for now. You can display it later on if you like.

Let's create another selection to show how naming selections can be useful:

```
select subunit2, all within 5 of resi 165
```

When you've created this selection, hide it, too, by clicking on its box. Now, if we want to display all those atoms that are both within five angstroms of residue 115 and five angstroms of residue 165, we simply need to type:

```
hide spheres
select merged, subunit1 and subunit2
show spheres, merged
```

You'll see that only a few atom's pink dots now appear in the new selection, and the final command displays only those atoms as spheres. Constructing the intersection of two selections is a useful way to weed out unwanted atoms from your selection without making extremely long selection expressions.

How can you determine what those spherical atoms correspond to? Early versions of PyMol made this quite a difficult task, but in newer versions this is relatively easy to do. First, hide all of the selections so you see no pink dots on the screen. Then, double click on the blue spherical atom. You should see a menu appear when you double click indicating that you double clicked on the nitrogen atom from valine 136. The top of the menu should look something like this:

```
/1RBP/1RBP//VAL`136/N
```

This syntax corresponds to the internal representation PyMol uses to store atoms. The first two words (1RBP, 1RPB), indicate the file and molecule name of the object. The text we're interested is at the right of the string: "VAL`136" tells us the atom is a part of the valine (Val) residue at position 136 in the amino acid sequence. The N tells us this is the backbone nitrogen from this residue. Each atom in each residue has a standardized name that can be represented in the PDB text file. For example, the backbone atoms are named N, CA, C, and O, for the amino nitrogen, alpha carbon, carbonyl carbon, and carbonyl oxygen, respectively.

Exploring the menu for the atom will show lots of different options, including display options. Items under the "atom" menu will affect only that atom, whereas items under the "residue" menu will affect the residue associated with that atom. Clicking elsewhere on the screen will hide the menu.

e. How many other atoms are displayed as spheres at this point? What residues are represented in this selection (give both names and numbers)? (3 points)

Atoms don't need to be shown as spheres in order for you to be able to double-click on them. Any atom will display this menu, but atoms displayed as spheres will be easier to click on. Try double clicking a few other atoms to see what you get.

Saving Your Pictures

It is often very useful to be able to save the output of your work as an image file that can be published, used in a presentation, or submitted as part of your homework. PyMol provides a way to do this using the PNG image file format. To save your image, simply select "Save Image…" from the PyMol "File Menu." After you choose a filename for your image file and select the location, your image will be saved.

If you look in the directory where you saved your PNG file, you should now see the new file that you saved. Double clicking on this file will should display an exact replica of your PyMol screen at the time you saved the file. As PNGs are recognized by almost all publication and presentation software, you can now import the image into other software programs.

Of course, when preparing images for papers or your final project, you may wish to change the background color for your pictures from black to white. You can do this by selecting the "Display Menu" and choosing "White" under background. It's harder to visualize this particular screen mode on your monitor, but it does save ink and present well when included in papers.

For images that are really sharp, you can use the PyMol `ray` command. This performs what is called a "ray trace" of the display. Using the principles of the physics of scattering, `ray` assumes a light source and renders the image as though beams of light were actually hitting objects and reflecting to the camera view. Most images used in professional publications are ray-traced images. The image, once ray traced (you'll see a blue bar indicating the progress of your work), can be saved using the "Save Image…" menu option, as before. Because of the amount of time this takes, it is recommended that you take care when using the command: you can easily craft an image that takes several hours to ray trace. To illustrate ray tracing, try entering the following commands:

```
hide everything
show spheres, resi 1-16 and not resn RTL
ray
```

Note that if you try to rotate the structure before saving the PNG image, the ray-trace data will be lost, so make sure the image is exactly how you want it before issuing the `ray` command and then save the image immediately when it completes. Since we aren't keeping this image, you need not save it.

In addition to saving pictures, PyMol can save your session itself: Under the "File" menu, you can select "Save Session As" and save the PyMol session file (with a .pse extension). If you must stop using PyMol, you can save your session, quit, and load it again, finding yourself in exactly the same program state that you were in before you quit. PyMol session files, however, cannot be imported by most programs, and you must use the PNG option for exporting images.

Other Representations of the Protein

PyMol can display the protein in other representations than simply spheres. For a list, type in `help show`. The available representations for `show` are also available the command `hide`. Some of the notable choices are:

- `lines` – This representation is what is displayed at startup, showing molecules as simple lines with atoms at the intersection between segments. In the default mode, carbon atoms are displayed in green, nitrogen in blue, oxygen in red, and sulfur in orange.

- `sticks` – Similar to lines, but the lines are of finite size and look similar to tubing or pipes.

- `ribbon` – Displays only a trace through the alpha carbons in the backbone. The result is a line tracing through the protein chain itself.

- `cartoon` – This is a representation that shows alpha helices, beta strands, and coil regions using smoothed cartoon illustrations. Of all the representation styles this option is the least scientifically accurate, as the atomic positions are not explicitly displayed. However, this option most clearly displays the overall protein topology with the secondary structure overlaid.

- `everything` – This is a wildcard that matches all representations. For example, typing `hide everything, all` will hide every representation (sticks, lines, etc.) for the entire molecule.

f. Construct an image of retinol binding protein in cartoon representation. Display the retinol molecule (`resn RTL`) in sticks representation. Save your image as a PNG file and upload it to the course web server. (4 points)

Coloring Your Molecules

Using a syntax similar to the `show` command, you can change the color of various parts of the molecule. The following command colors the retinol yellow:

```
color yellow, resn RTL
```

There are many different colors available to you in PyMol, some of which can be displayed by selecting "Colors…" from the "Setting" menu. In this menu, you can also edit existing colors or create new ones.

If at any time you wish to revert to the default PyMol coloring, entering the following command will undo all the color changes you may apply.

```
util.cbag
```

Displaying Distances

One final tool remains for you to learn in our tutorial of PyMol. It is often useful to visualize how close (or far apart) two atoms are. This can be done using the `dist` command in PyMol. To determine the distance between two atoms in PyMol, use the following syntax:

```
dist <name> = <select 1>, <select 2>
```

In your homework assignment for this week, you will determine an average distance between alpha carbons (atom name CA) in a protein. To determine the distance between one single pair of alpha carbons, you can use the following example:

```
dist cadist = resi 49 and name CA, resi 50 and name CA
```

When you type this command, you will see a distance appear on the screen and on the list of objects at the right of your display. Your distance should be about 3.9 angstroms. As you rotate the molecule, the distance and the label will rotate with it.

You selections for the distance command need not be single atoms. Try entering the following commands:

```
hide everything
center resi 50
show sticks, resi 49-51
dist test = resi 49, resi 51, 4
```

You'll see that two distances are highlighted, but no distance is created that is greater than four angstroms. Thus, `dist` can draw several distances at once when the selections contain multiple atoms. Repeating the above commands while leaving off the cutoff specification of 4 angstroms will give you an idea why it's a good idea to include a cutoff when using multiple atom selections. Also note the utility of the `center` command: it centers your screen so that the rotation is about the selection you specify.

If you wish to remove distances once you have displayed them, you can either click on them at the left and hide them, or you can type `delete <name>` to remove them more permanently. The following command will delete the distance object you just created.

```
delete test
```

The delete command works for other things too, like named selections and molecule objects. Use it with care.

The Tools Menu

Many of the topics we discussed above are available through the menu at the right hand side of the display. Clicking the "A" next to the 1RBP object will display a list of actions, including delete, which was discussed above. The "S" will allow you to display how the molecule is displayed, and stands for "show." Similarly, "H" stands for "hide." "L" allows you to label your selections and objects, and "C" allows you to change the color.

On Your Own

You should have all the information necessary to produce basic analysis and image production in PyMol. If you need additional help, however, PyMol has a fairly good help system, available on the "Help" menu or at the PyMol website, `http://www.pymol.org/`.