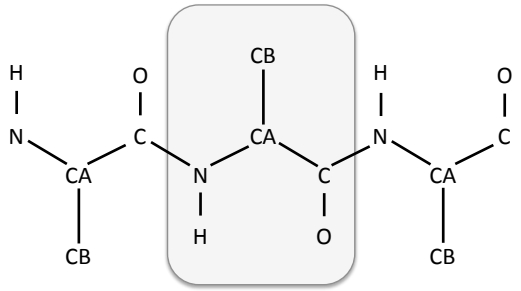
Homework #1 Recapitulating Ramachandran statistics with Rosetta's energy terms

In this homework you are going to write a PyRosetta script to evaluate the energetics of tripeptides at various values of phi and psi.



Part 1. Write a function that will:

a) generate a tripeptide of the sequence 'AXA' where X may be any one of the twenty amino acids

b) set the phi and psi angles of the center amino acid to the input values

c) set the phi/psi values of the first and third residues to "extended," phi=-120 and psi=120

d) "repack" the center residue, finding the sidechain conformation that optimizes total energy

e) evaluate the Rosetta energy of the tripeptide.

Note: you should use a modified energy function with only the following terms:

```
scorefxn.set_weight(rosetta.core.scoring.fa_atr, 1.0)
scorefxn.set_weight(rosetta.core.scoring.fa_rep, 1.0)
scorefxn.set_weight(rosetta.core.scoring.fa_elec, 1.0)
```

The PyRosetta tutorial notebooks (http://www.pyrosetta.org/tutorials) might be useful in implementing these functions. See the notebooks corresponding to chapters 2, 3, and 6. For this assignment, you might find the following functions helpful:

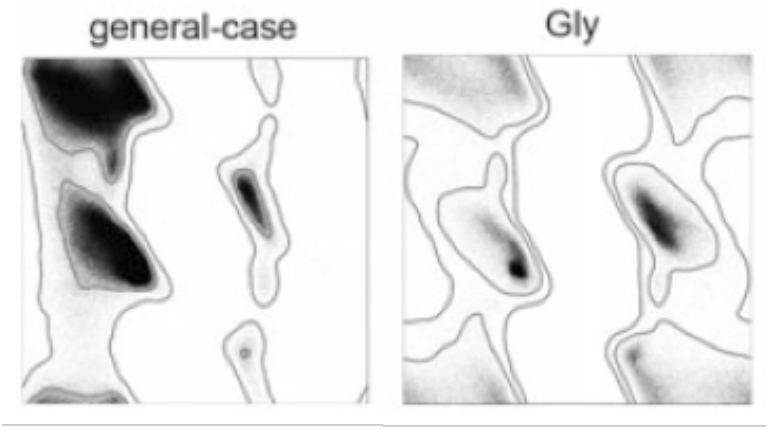
```
## generate a model with a given sequence
pose = pose_from_sequence('AAA')

## pack a single residue
task_pack = standard_packer_task(pose)
task_pack.restrict_to_repacking()
task_pack.temporarily_fix_everything()
task_pack.temporarily_set_pack_residue(2, True)
pack_mover = (
   rosetta.protocols.minimization_packing.PackRotamersMover(
     scorefxn, task_pack))
pack_mover.apply(pose)
```

As a simple test, you should see energy values of the following with a center residue of 'W' and a phi/psi of -60 and -50 as **about -2.116.**

Part 2. You are going to use the code from part 1 to evaluate this energy over all of phi/psi space. For the middle residue, we will scan over all possible phi-psi combinations at 10 degree increments (so 36x36=1296 conformations). Set the identity of all residues to 'A' (alanine), and

the phi and psi angles of the terminal residues (1st and 3rd) to an extended conformation, phi=-120 and psi=120. Evaluate the energy of each of these 1296 conformations. Plot these energies as a function of phi and psi, where phi is on the x-axis (from -180 to 180) and psi is on the y axis (from -180 to 180). How does this compare to the "general case" Ramachandran plot shown below? If it differs, why do you think is it different?

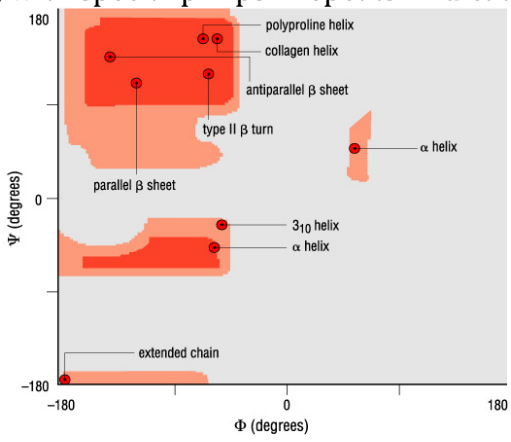
Now change the middle residue to 'G' (glycine) and rerun the analysis. Plot the data in the same way as before. How does the plot differ from before? How does it compare to the Ramachandran plot of glycine, shown below? If it differs, why do you think is it different?



Do the same analysis for three other amino acids: proline (P), isoleucine (I) and one other of your choosing. How do these compare to alanine and glycine? Chemically, why do you think they are the same/different?

Part 3. You are now going to look at the energetics of *repeating* phi/psi values. Modify your code from part 1 so that you generate a 9-mer of all alanines and set the phi/psi values over *all* residues.

What does the plot look like in this case? How does it differ from the "AAA" case in part 1? We showed a plot like this in class, with special phi-psi "repeats" indicated:



Do you tend to find low-energy states at these repeats? Why or why not? Using PyMol with the following command to dump structures:

```
pose.dump_pdb("outpath.pdb")
```

Look at some of the low- and high-energy points in this plot. Does this make sense physically? How could this simulation be further improved?

Finally, the simplified energy function we have used does not account for hydrogen bonding. However, we learned in class that hydrogen bonds are a major component in both helices and sheets. Try performing the same analysis with hydrogen bonding terms enabled:

```
scorefxn.set_weight(rosetta.core.scoring.hbond_sr_bb, 1.0)
scorefxn.set_weight(rosetta.core.scoring.hbond_lr_bb, 1.0)
```

Does this change your analysis? Why or why not? Using PyMol, visualize structures with the largest relative change in energies. What is special about these conformations?

What to hand in: The code from parts 1-3, the plots generated in part 2 and 3, and (at most) a one-page summary answering the questions proposed in this homework.