CS612 Homework Assignment 5 – Updating

Due May 9, 2025, on Gradescope

1. **Term project:** Based on the one-hot encoding from last HW assignment, write code to read a file with a multiple sequence alignment (such as the ones I linked on the course webpage), and create a matrix with the one-hot encoded sequences. Attach your code.

2. **Term project:** Write a (concise, but not too short) progress report about your term project. Describe what you have done so far, what difficulties (if any) you ran into. Detail your choice of programming language, implementation etc. Even if you haven't started yet (which you should have!) write about your plans for starting.

3. MD Tutorial: OpenMM is a platform for molecular simulations. You can find them here: `https://openmm.org/documentation`.It provides a very convenient python platform to run simulations. To start, you should install openmm – see here `http://docs.openmm.org/latest/userguide/application.html#installing-openmm`. I recommend either a virtual environment or conda. Start out from the tutorial linked here: `https://openmm.github.io/openmm-cookbook/latest/notebooks/tutorials/protein_in_water.html`. In this tutorial you will conduct an MD simulation of Lysosyme. The tutorial uses the AMBER force field and a water box.

   The protocol does everything we discussed – adds hydrogens and water with a 10Å padding and setting up the simulation parameters. Then you perform energy minimization, equilibration and production. Notice that the code contains the objects `PDBReporter` and two `StateDataReporter` objects - one that outputs into `stdout` and one to a log file.

   (a) Follow the tutorial with the following modification: Add a minimization reporter that outputs the energy of the system during minimization. You can use the class `MinimizationReporter` for your code - you would need to define a class that inherits from `MinimizationReporter`. Have it output the energies every 100 steps. Plot the energy (y-axis) during the minimization as a function of the number of steps (x-axis). You can use the matplotlib library in a similar way to the tutorial. Attach the energy plot as part of your solution. What is the starting energy? Final minimum energy? Number of steps?

   (b) The equilibration stage (NVT – constant volume and temperature) runs for 10000 steps and the production, (NPT – constant pressure and temperature) and reports the results to a file named `md_log.txt` What are the temperature, volume and potential energy after 100 steps? After 20,000 steps? Attach the three plots of the evolution of these variables to your solution.

   (c) Notice that these properties are supposed to reach an equilibrium before the production (step 10,000), so they don't change much afterwards. Looking at the data and the plots, is this the case?

   (d) Here you will again have to write your own code. The resulting file, output.pdb, is quite large. It contains the equilibration and production trajectory written every 1,000 steps, so 20 snapshots overall. The reason the file is so big is that it has the waterbox. If you want you can open it with a molecular viewer and look (it may be really slow and not part of the HW). Align the first and last snapshots – marked model1 and model20, and calculate the RMSD between them. Attach the superimposed image as part of your solution. You can use Biopython for this part.