Assignment 2 — Challenge Question

Designing Disulfide Bridges to Stabilize Proteins

CS/BIOE/CME/BIOPHYS/BIOMEDIN 279

Due: November 8 at 3:00 PM

Note: This problem is strictly optional, but if you submit by the due date, we will review your solutions and consider them for extra credit. This problem is open-ended, and you're welcome to submit a partial solution.

This problem was inspired by a current research project led by Dr. A.J. Venkatakrishnan (a postdoctoral scholar in the Kobilka and Dror groups at Stanford) and composed by Anthony Ma.

Introduction

In many cases, one wishes to stabilize a protein, reducing its ability to change conformation. For example, this makes it easier to do various experiments on the protein (including solving crystal structures of the protein with many different bound ligands, which is very useful for structure-based drug design). One might also want to lock a protein into one particular conformation.

One effective method for stabilizing proteins involves introducing cysteine residues that can form disulfide bridges (also known as disulfide bonds) with one another. This is generally done by mutating some of the amino acids in the protein to cysteine.

In this problem, we will explore computational techniques for designing disulfide bridges into a protein. We will work with an adrenaline receptor, and in particular with PDB entry 2RH1 (the β_2-adrenergic G protein-coupled receptor). This receptor and closely related adrenaline receptors are the targets of beta-blockers, which are used to treat cardiac diseases (heart attack, heart failure, high blood pressure), and beta agonists, which are used to treat respiratory diseases (asthma, emphysema).

Part 1

Propose five pairs of residues that can be mutated to cysteine in order to form disulfide bridges that can potentially stabilize the overall protein conformation. (Some of these pairs may include one residue that is already a cysteine in the wild-type protein, so that only one residue in the pair needs to be mutated.)

The following pseudocode can help you get started on your own implementation of the disulfide bridge designer. Feel free to design your own alternative method as well. Recall that a rotamer is a particular conformation of an amino acid side chain.

For every pair of amino acids (aa1, aa2) in Protein:
- Mutate aa1, and aa2 to Cysteine

for every rotamer of aa1:

for every rotamer of aa2:
Test whether both of the following conditions hold:

1) Distance between gamma sulfur atoms (SG) is less than 3 Å.

2) Dihedral angle chi = (rotamer_1 "CB", rotamer_1 "SG", rotamer_2 "SG", rotamer_2 "CB")
and ((chi > –117° and chi < –57°) or (chi > 67° and chi < 127°))

Some helpful commands in PyMol for this implementation are

- cmd.wizard("mutagenesis") for mutating specific residues
- cmd.count_states to determine the number of rotamers for each residue
- cmd.frame(frame_number) to access a rotamer
- cmd.create() creates a rotamer for a specified residue
- cmd.get_distance() computes the distance between two atoms
- cmd.get_dihedral() computes the dihedral angle determined by four specified atoms

Part 2

The approach of part 1 does not take into account the fact that mutating residues will often change the local geometry of the protein, particularly the nearby sidechains. (In the absence of such changes, mutations might introduce clashes between atoms that are very unfavorable energetically.)

How might you improve the algorithm you used in part 1 to take such geometric changes into account?

You may wish to use the side-chain packing function from section 3.3 of the main assignment. If you wish to introduce a disulfide bond in PyMol, see:
https://pymolwiki.org/index.php/Modeling_and_Editing_Structures#Adding_disulfide_bonds

Part 3

In part 1, we suggested criteria for formation of a disulfide bridge. How could you determine such criteria if they weren't given? Can you suggest additional criteria (or types of criteria) that might improve the design methodology? Do you have other ideas for improving the method?

Submission

Your write-up to these questions should also include a description of your approach and the results for each of your methods (i.e., which residues to mutate to cysteine). Please save your write-up as **`challenge1.pdf`**. Please save your code and all the commands you used on PyMOL as **`disulfide.py`**. Be sure to comment your code such that we can reproduce your results. Upload your files to corn, change into the directory containing the files, and execute the command below to submit.

```
/afs/ir/class/cs279/scripts/submit.py assn2-challenge .
```