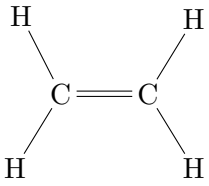
CS612 Homework Assignment 3

Due Thursday, March 30, 2023

1. **Internal and Cartesian coordinates:** The molecule ethene or ethylene looks like this:



The bond between the two carbons is double, and the molecule itself is all planar.

The Z-matrix looks as follows:

Atom	Bonded	Dist	Angle	Value	Dihe	Value
C						
C	1	1.31				
H	1	1.07	2	121.5		
H	1	1.07	2	121.5	3	180.0
H	2	1.07	1	121.5	3	180.0
H	2	1.07	1	121.5	4	180.0

Reconstruct the cartesian coordinates for ethylene. Have the first C (the left most in the figure) be the origin and the bond between the two carbons be the X-axis. Show your calculations. You may either calculate by hand or write a small piece of code to convert polar to cartesian coordinates. Notice that most software packages measure angles in radians, not degrees.

Note: Since the molecule is planar you can use x,y coordinates only.

2. **HP lattice model:** Given the following two sequences:
   - `S1 = HHPPPPHPPPH`
   - `S2 = HHPHPPHPHPH`

   (a) Find five possible high-scoring self-avoiding 2-D grid arrangements for S1 and calculate each one's "potential energy" according to the model discussed in class (award one point for each pair of H-H points that are one grid point apart (not diagonal), but not adjacent on the sequence. For more info look at the class notes – I'm referring to the white dashed lines. Attach the five arrangement (indicate which one is the start and which one is the end). Color P in blue and H in red or magenta, like in class. Notice that you don't have to find the absolute globally best structure, just arrangements that have a good score.

   (b) Find five possible high-scoring self-avoiding 2-D grid arrangements for S2 and calculate their score as in (a).

   (c) For the same grid arrangements as in (a), "thread" S2. That is, use the same arrangements as in (a) above but the amino acid sequence in S2. Calculate the score again.

(d) For the same grid arrangements as in (b), "thread" S1. That is, use the same arrangements as in (a) above but the amino acid sequence in S1. Calculate the score again.

(e) Explain the observed differences briefly.

(f) Small Bonus: You may also try the attached code to test your results – just change the sequence in the main function. It performs a simple search over an HP lattice and returns five arrangements. Attach the images for the two sequences above.

3. **Hands-on homology modeling exercise using a template:** In this assignment you will perform homology modeling using the SwissModel server.

   As a nice and easy example, let us start by modeling a short protein from a family called Crambin. Go to the SwissModel server at: `https://swissmodel.expasy.org/interactive` . There are three ways of doing homology modeling using the SWISS-MODEL server – a fully automated approach (called Automated mode), Alignment mode which allows you to submit a multiple sequence alignment of your target with one or more templates and a Project mode, which allows you to manually optimize your alignment. We will use the automated.

   Note: SwissProt contains a rather extensive help section, please take a look if you need a clarification: `swissmodel.expasy.org/docs/help` I suggest you create a login with them, so that your jobs will be saved on the server for about a week.

   On the top window, copy and paste the following sequence:

   `SVCCPSLVARTNYNVCRLPGTEAALCATFTGCIIIPGATCGGDYAN`

   Click on "search for templates" and wait. It should take a couple of minutes. When you are done, you can access the templates by clicking on "templates". The page contains information about each template – its PDB code, its coverage of the query sequence, its sequence identity etc.

   (a) How many templates did you get? What is the accession code, GMQE and identity of the top template? Notice that this is not a PDB file but an AlphaFold model.

   Now, check the boxes near the top three templates and click "build model" on the top right. The server will now try to build three models, based on the three templates. Again, you will probably have to wait a while.

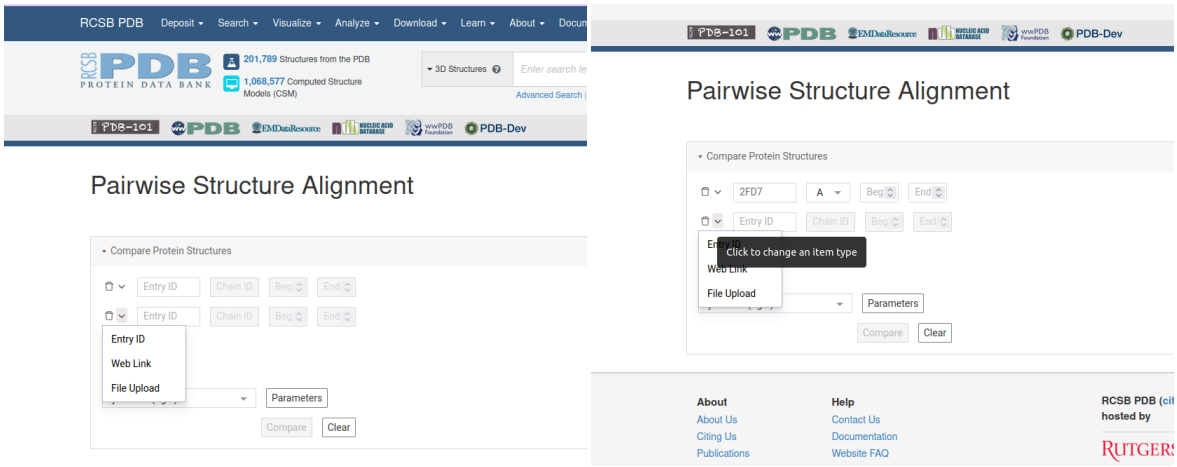
   (b) You can look at each model closely using the "Structure assessment" option. What is the QMean score of each of the models? (this is a score that combines various aspects of interactions in the molecule. This statistical score is logarithmic and the lower, the worse.

   (c) What is the overall score, GMQE (Global Model Quality Estimation) of each one of the models?

   The scoring functions combine the energy terms used for the modeling, which are shown on the model window on a blue (good) to red (bad) scale. Notice the three superimposed models on the right window. You will notice that the protein is depicted on a blue to orange scale which correspond to model quality. As you may tell, the orange parts correspond mostly to loop regions.

   At the top of the page you will see an icon of a page and if you hover the mouse over it, it will say "one page project report". Please attach it to your submission.

   (d) Now measure the RMSD and TM scores of each model to its respective template. The easiest way IMO is to save the three models to your computer by clicking on the button near the model and save as pdb. The names of your file will be model_01.pdb, model_02.pdb and model_03.pdb. For each model do the following: Go to `https://www.rcsb.org/alignment`. In one window put the pdb code (only the first four number-letter-letter-letter/number), and chain A. In the second window click "upload file", upload your respective model and hit "Compare" (see Figures).



Hit ”compare” and retrieve the RMSD and TM scores from the ”Scores” tab (see Figure):

(e) **Bonus:** Implement (d) with BioPython. You can use the Superimposer or cealign modules. Calculate only the RMSD, not the TM-align score (there is no easy way to calculate it out of the box).

Important! Notice that even though the models have the same coverage, the same sequence identity and they are overall quite good, model 2 beats the model 3 in all the categories. The reason is that the model 3 was generated by NMR and model 2 was generated by X-ray and has very high-resolution. NMR structures tend to be more fuzzy, and in the case of homology modeling selecting a good quality, high-res structure can be quite important. While all three models came out OK in this case, it can be quite critical in other cases If at all possible – aim for X-ray structures when you create your models, and try not to go below a 2Å resolution. Notice that model 1 was generated by AlphaFold and is the best of them all. AlphaFold does a very good job in modeling relatively small structures for which lots of similar sequences and structures are available.

4. **Basic protein folding exercise:** Use Protein Investigator (the software we demonstrated in class) at http://intro.bio.umb.edu/MOOC/jsPI/JsPI.html. It requires the Java running environment to run. On the upper folding window type the following sequence: IFMQSRTDAA (Ile-Phe-Met-Gln-Ser-Arg-Thr-Asp-Ala-Ala). Type ”Fold” and see the shape of the folded protein. The energy function is based on hydrophobic contacts, ionic interactions (opposite charges attract, similar charges repel each other), and hydrogen

bonds between polar amino acids. For the classification of hydrophobic, charged and polar amino acids see class notes.

(a) Create a mutant protein by changing **one** amino acid in the sequence above, such that the mutation has no effect on the shape of the mutant protein. Explain. Attach a screenshot of the resulting protein.

(b) Create a mutant protein by changing **one** amino acid in the sequence above, such that the mutation has a large effect on the shape of the mutant protein. Explain. Attach a screenshot of the resulting protein.

(c) Design a protein of at least 8 amino acids such that a salt bridge (an ionic interaction between charged amino acids) is critical to its shape. Explain and attach a screenshot.

5. Given the following point sets:

Point Set A:

0.9003	-0.3258	-0.2888
-0.5377	0.2196	-0.8140
0.2137	0.8614	-0.4608
-0.0280	-0.0740	-0.9969
0.7826	0.2782	0.5569
0.5242	-0.7065	0.4755
-0.0871	0.9154	-0.3929
-0.9630	0.2336	-0.1344
0.6428	-0.6475	0.4094
-0.1106	0.7801	-0.6158

Point Set B:

-0.8842	0.4649	0.0448
-0.2943	-0.0193	-0.9555
0.6263	-0.7336	0.2636
-0.9803	0.1798	-0.0821
-0.7222	-0.6759	0.1467
-0.5945	-0.7013	0.3934
-0.6026	0.4536	-0.6566
0.2076	-0.9660	-0.1540
-0.4556	0.2610	0.8511
-0.6024	-0.3751	-0.7046

(text versions available as set1.txt and set2.txt, enclosed).

(a) Determine the RMSD between the two point sets. Do not modify the coordinates (yet). DO NOT use the Superimposer module in Biopython (just use simple array functions). You can use the NumPy package. See more details here: https://www.tutorialspoint.com/matrix-manipulation-in-python. You can also use Matlab or R if you want, details below.

(b) Determine the optimal RMSD between the point sets given that they are allowed to translate but not rotate. Again, do not use the Superimposer module. It can be shown that the optimal RMSD is obtained when the two point sets are translated so that their centroids are at the same point. The centroid or center of mass of a set of points is a point whose x,y,z coordinates are the average of the x,y,z coordinates of the point set, respectively.

(c) Now determine the optimal RMSD between the point sets given that they are allowed to translate AND rotate. You can now use biopython. Please attach your source code.

Setup for Matlab

If you have access to Matlab you can do the calculation really easily using basic matrix operations. You can read a text file using set1=load('set1.txt'), same for set2.txt. The data will be read as a 10×3 matrix. Matlab supports a number of useful matrix operations. Some operations that can be useful here: set1-set2 subtracts the two matrices. set1(:,1) extracts the first column into a vector. set1(1,:) extracts the first row. mean(vec) returns the average value of the input vector. set1.*set2 does an element by element multiplication (as opposed to matrix multiplication). More can be found in the Matlab documentation at http://www.mathworks.com/products/matlab/ .

Setup for R

R is somwhat less user friendly IMHO but it's free, making it a lot more appealing... If Matlab is not available to you, you can use R. A nice GUI, RStudio, is also available for free. To read a data into an R variable use, for example, set1←read.table("set1.txt"). The data will be represented in this case as a 10×3 matrix. Some useful matrix operations supported by R: set[,1] extracts the first column into a vector. set[1,] extracts the first row. set1*set2 does an element by element multiplication of set1 and set2 (as opposed to matrix multiplication). The function colMeans gives the average value of the columns (for centroid calculation). To subtract a vector from a matrix row by row use the sweep function, as in sweep(mat,MARGIN=2,vec,FUN="-") (mat and vec are the names of the matrix and vector, respectively). MARGIN=2 means rows. See more here: http://www.r-project.org/other-docs.html