Homology modeling with CPHmodels and HHpred.

GOAL: You will make a homology model of the H7N7 hemagglutinin (HA) from the influenza A virus.

Hemagglutinin (HA) is a glycoprotein found on the surface of influenza viruses. The main function of hemagglutinin is to facilitate the binding of the virus to human cells.

Download the most recent H7N7 HA protein sequence from the influenza sequence database (http://www.ncbi.nlm.nih.gov/genomes/FLU/FLU.html).

1. Under "Type" pick "A" to select influenza type A.
2. Under "Host" pick "any".
3. Under "Country/Region" select "any".
4. Under "Protein" Select "HA".
5. Under "Subtype" select H7 and N7.
6. Select "full-length only" to get only complete sequence.
7. Click "Add query" and then "Show results".
8. You can sort the result by clicking on the "date" to find the most recent protein.
9. The most recent hemagglutinin should be a protein with the accession number AQS25750
10. Find and save the sequence by clicking on the accession number AQS25750 select FASTA.

We will now use CPHmodels and HHpred to generate a homology model of our selected hemagglutinin sequence.

CPHmodels

CPHmodels was created by Ole Lund et al. from CBS. Its main plusses are that it is very simple to use (put a sequence in and get a model out) and very fast. It compares well with other servers in producing accurate models for high homology problems, but with regions of low homology, it is quite conservative and only models the part it is most confident about. (This can be considered an advantage.) CPHmodels will create a model based on the template it finds as the best, and it is not possible to force it to use another template, if you do not like its choice of template, or to manually edit the sequence alignment.

1) Make a homology model of your protein with CPHmodels by submitting the sequence of your protein to the server (http://www.cbs.dtu.dk/services/CPHmodels/).
2) Find and note the name of the template and the specific chain used for modeling.
3) Have a look at the alignment between the query and the target sequence in the lower part of result page. Are there any insertions or deletions in the query-template alignment and if so list the residue numbers?
4) The 3D model can be downloaded if you click on the link called query.pdb. Save the model and open it in PyMOL.
5) Rename the saved model to CPH-model. In the right side of PyMOL click 'A' => 'rename object' => write "CPH-model" and press enter.
6) Compare the model with the template in PyMOL. This can be done by using the "fetch" command in PyMOL followed by the name of the template (You should have found that the template model is 3M5G so to get this structure type "fetch 3M5G")
7) Use the following command to change the representation of the structures:
hide all; show cartoon
8) Make a new object called CPH-temp by selecting the chain used by CPH-model. Type the following command: create CPH-temp, 3M5G and chain A
9) hide the 3M5G selection by clicking on the name in the right site of PyMOL.

10) Align the model and the template by using the following command in PyMOL:
align CPH-model, CPH-temp
11) Click on the little “S” i the bottom right corner of PyMOL to see the sequence of the structures.

We know from the alignment above that the model and template does not have the same length. Find the difference using PyMOL. Hint, find the residues in the insertion and highlight them by using the sequence that should have appeared after you clicked on the little “S”, or select the residues using the command eg. ‘sele CPH-model and resi 1-10’ (but remember to change the numbers in this command!).

In what part of the structure does the template and the model differs? Why could it be a problem that the template and the model differs in this part of the protein?

HHpred

HHpred (https://toolkit.tuebingen.mpg.de/#/tools/hhpred) is one of the servers which performed best at the latest CASP experiments. It has a lot of nice features – one of the most important being the ability to modify the sequence alignment by hand, if necessary. It is also possible to select which PDB entry you want to use as a template for your structure – you don’t have to choose the one which comes out as the best in the analysis by HHpred. This gives you the possibility for deselecting “bad” structures as templates. (You have to check the quality of the structures somewhere else, e.g. using links from the PDB website to ProCheck, WHATIF or other validation sites.)

To use HHpred you will need the license key for the modeling program MODELLER. The license key is: MODELIRANJE

1) Make a homology model of your protein with HHpred by submitting the sequence of your protein to the server. The server will now find the best templates and give you a list of them.
2) Write down the name of the best template (including the name of the chain used for modeling) and select it. Then click on “Model using selection”.
3) Click “Forward to MODELLER” and insert the MODELLER license key to build the model.
4) Download the final model by clicking “Download PDB file” and save it on your computer.
5) Open the HHpred model in your PyMOL session with the CPH-model by pressing “File” => “Open...” => find the file on your computer.

Use the commands you learned above to show your result in PyMOL. Rename the HHpred model ‘HHpred-model’ and align it with the rest by typing ‘align HHpred-model, CPH-model’. Then show it as cortoon.

Analyse the quality of the models

1) Were the two models built on different templates?
2) Also have a look at each of the templates in PDB (https://www.rcsb.org/pdb/home/home.do)
Which of the templates do you consider to be of the best quality?
3) Compare the CPH model and the HHpred model in PyMOL. Do the models cover the same part of the input sequence? If not, then which parts of the structure is missing? (Hint, look up the function of your protein of interest).

Try to check the model quality using the ProQ web server
(http://www.sbc.su.se/~bjornw/ProQ/ProQ.html).

1) Open the ProQ web server. Click on “ProQ web server” and upload our model under “upload a PDB file”.
Which of the two models scores best in ProQ?

Now go to RAMPAGE (http://mordred.bioc.cam.ac.uk/~rapper/rampage.php). The RAMPAGE server generates a Ramachandran plot that can be used to analyse the backbone conformation of the residues in your model.

2) Upload the HHpred- and the CPH-model to RAMPAGE and make a Ramachandran plot for each model. To get the plot in a better resolution you can download the result by clicking on "PostScript".

How many of residues are found in the disallowed region of the Ramachandran plot for each of your models?

Have a look at a few of these outliers in the CPH-model. Do you understand why these are outliers?

Now load the structure 5XL9 into PyMOL (type "fetch 5XL9") and show the structure in cartoon. 5XL9 is a hemagglutinin from an H4N6 influenza virus and this structure have the native hemagglutinin ligand, sialic acid (in PyMOL this ligand in denoted, SIA).

Align 5XL9 to your HHpred model in PyMOL (type "align 5XL9, HHpred-model"). Then select the sialic acid by typing "create SIA, 5XL9 and SIA". Right next to the selection called SIA click 'S' => 'sticks' and color it red by clicking 'C' => 'reds' => 'red'.

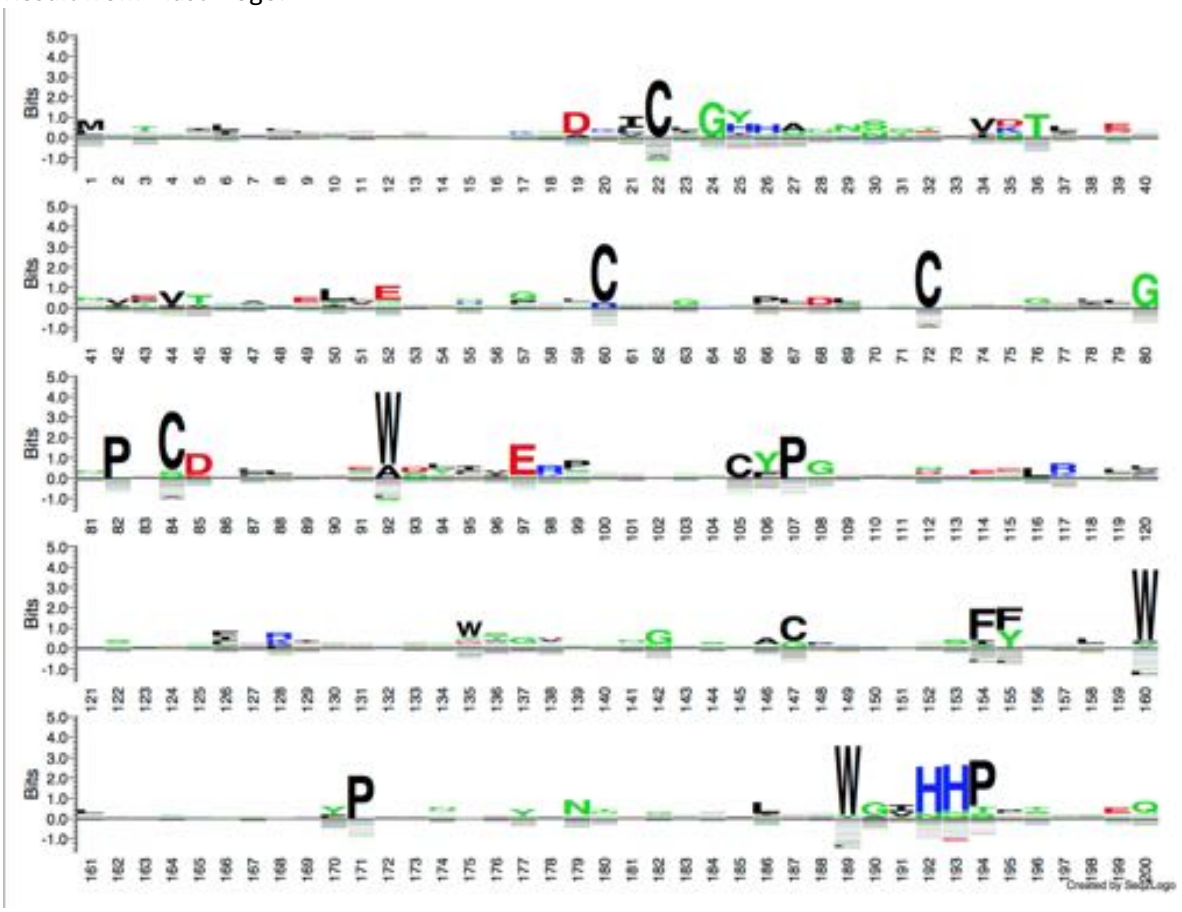
Now find the residues that are in close contact with the sialic acid, using the following steps:
1) select the sialic acid and rename the selection SIA
2) To select residues within 5Å write the following command "select nearSIA, br. all within 5 of SIA". This will generate a new selection called nearSIA.
3) Right next to the selection called nearSIA click 'S' => 'side chain' => 'as sticks'.

How many residues from the HHpred model are in contact with the sialic acid? (Hint. To count this look at the highlighted residues in the sequence show in the top part of PyMOL).

How did the HHpred model the binding site and are there any big clashes between the model and the sialic acid? how does this compare to the CPH model?

Now go to Blast2Logo (http://www.cbs.dtu.dk/biotools/Blast2logo-1.1/) and have a look at this website. Use your sequence as input set "Blast Database" to nr70 and "Number of Blast iterations" to 3 and click Submit. It takes a long time to run the job so use the result shown below.

Result from Blast2Logo.



By looking at the Blast2Logo can you guess which of the following residues will be in the binding site of the protein?

1. residue 60 and 72
2. residue 105-107
3. residue 130-133
4. residue 160

Have a look at the HHpred model in PyMOL, did you guess correctly? Hint, be aware that the Blast2Logo covers the original query sequence and the HHpred model does not, there should be around 16 residues in length difference so remember to account for this when finding the residues in the HHpred model.

Have a closer look at the two cysteines in the HHpred-model. Are they located close to each other? What kind of bond can be formed between these two cysteine residues and does this explain why they are important for the tertiary structure?

Good job, you have now created two different models of the hemagglutinin molecule and analysed the quality using different tools. After all this work, which of the two models do you think is the best?