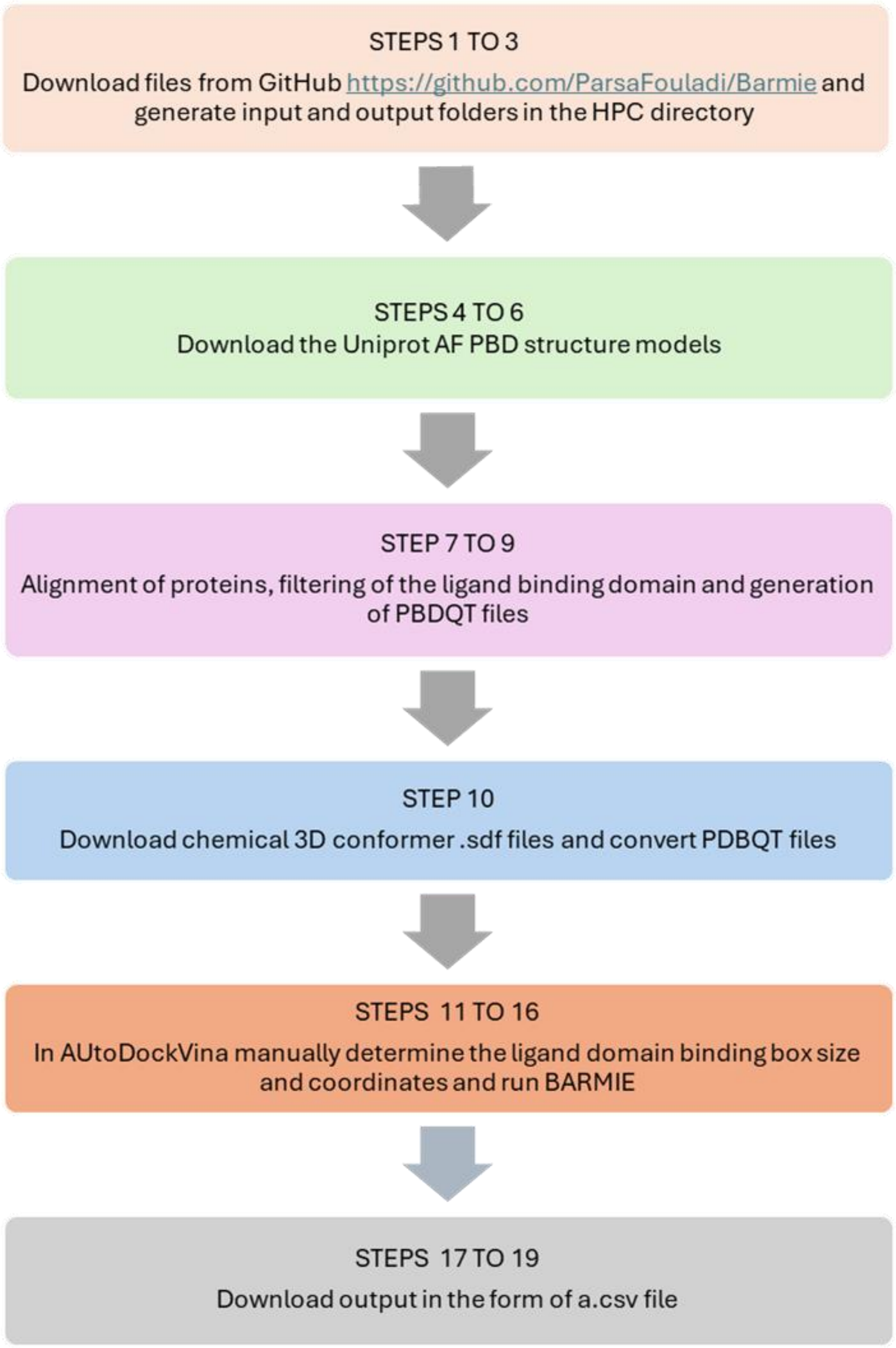
Running the Binding Affinity Ranking at the Molecular Initiating Event (BARMIE) pipeline

IMPORTANT: This procedure is an example using the University of Southampton IRIDIS5 HPC. Your local HPC may use different architecture and thus you may need to modify the instructions.

Flow diagram describing the sequential steps within the BARMIE pipeline



STEPS 1 TO 3

Download files from GitHub https://github.com/ParsaFouladi/Barmie and generate input and output folders in the HPC directory

STEPS 4 TO 6

Download the Uniprot AF PBD structure models

STEP 7 TO 9

Alignment of proteins, filtering of the ligand binding domain and generation of PBDQT files

STEP 10

Download chemical 3D conformer .sdf files and convert PDBQT files

STEPS 11 TO 16

In AUtoDockVina manually determine the ligand domain binding box size and coordinates and run BARMIE

STEPS 17 TO 19

Download output in the form of a.csv file

PREPARING PROTEIN AND LIGAND FORMATS.

1. Download files from GitHub (https://github.com/ParsaFouladi/Barmie) into your HPC directory
    - Insert command *cd and directory address where your script is placed*

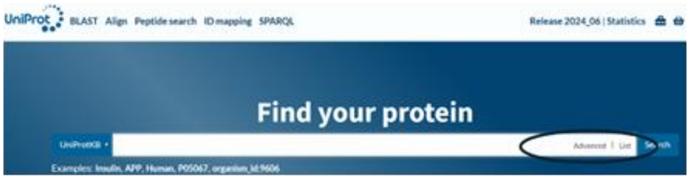
   This provides access to the script.
2. Generate the following folders:

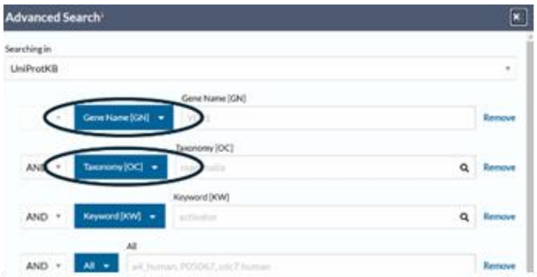
   PDB_files
   PDB_files_filtered
   PDB_files_aligned

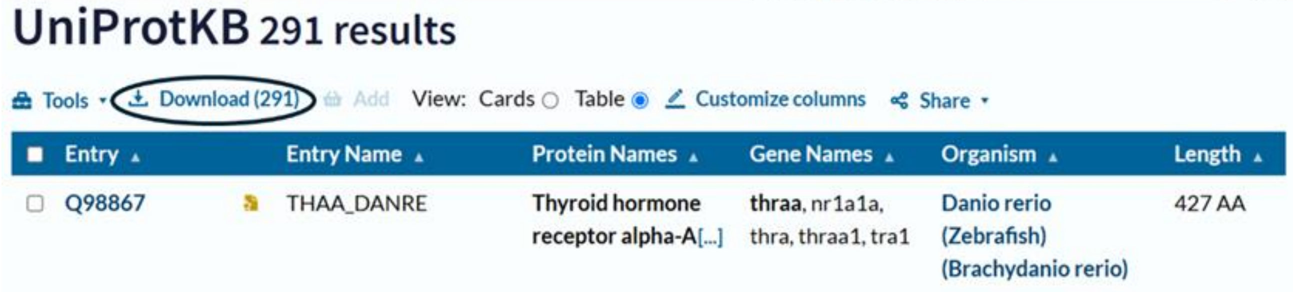
PDB_files_pdbqt
LIGAND_sdf
LiGAND_pdbqt

3. In the terminal enter the command:
   - *conda activate pymol_env*

   This provides access to PyMoL
4. To collect pdb files from Uniprot database Excel file function click on advance search then enter gene name and taxonomy, click on search and download.





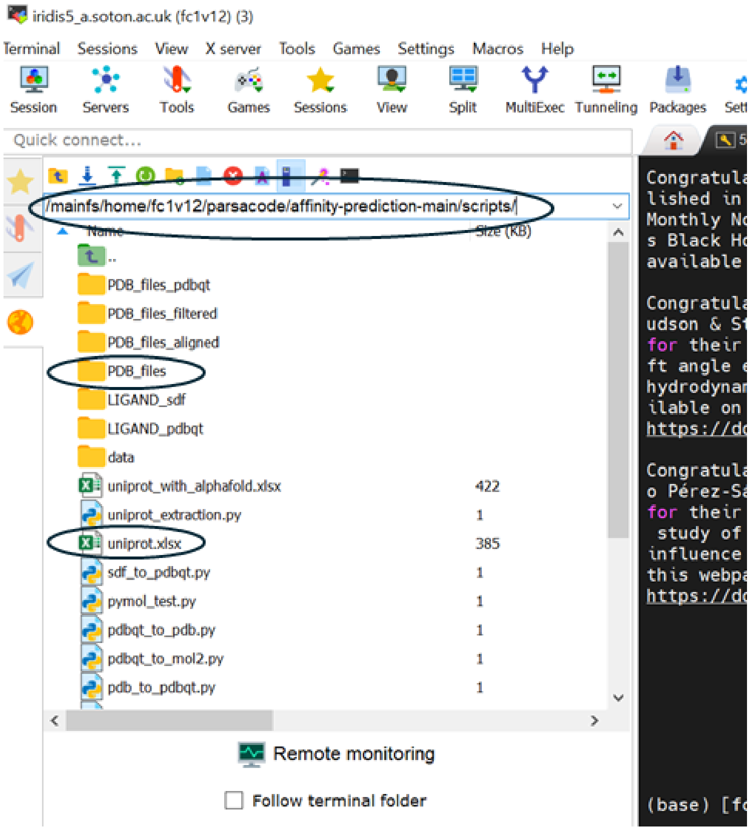


UniProtKB 291 results

Entry	Entry Name	Protein Names	Gene Names	Organism	Length
□ Q98867	THAA_DANRE	Thyroid hormone receptor alpha-A[...]	thraa, nr1a1a, thra, thraa1, tra1	Danio rerio (Zebrafish) (Brachydanio rerio)	427 AA

5. Rename the Excel file 'uniprot.xlsx' (this ensures recognition by the script) and move to the HPC directory.
6. Insert the command:
   - *python uniprot_extraction.py*.

   The *PDB_files* folder will contain the pdb files.



/mainfs/home/fc1v12/parsacode/affinity-prediction-main/scripts/

Name	Size (KB)
..	
PDB_files_pdbqt	
PDB_files_filtered	
PDB_files_aligned	
PDB_files	
LIGAND_sdf	
LIGAND_pdbqt	
data	
uniprot_with_alphafold.xlsx	422
uniprot_extraction.py	1
uniprot.xlsx	385
sdf_to_pdbqt.py	1
pymol_test.py	1
pdbqt_to_pdb.py	1
pdbqt_to_mol2.py	1
pdb_to_pdbqt.py	1

```
Congratula
lished in
Monthly No
s Black Ho
available

Congratula
udson & St
for their
ft angle e
hydrodynam
ilable on
https://do

Congratula
o Pérez-Sa
for their
 study of
influence
this webpa
https://do

(base) [fc
```

7. For the alignment of the glucocorticoid receptor protein structures and then extractions of LBD structure insert the command:
    - *python filter_pocket.py PDB_files PDB_files_filtered KAIEP 7*.

    This command is specific to this example and the teleost fish GRs. The amino acid sequence KAIEP conserved between species and is used as a guide, the code will need to be altered for your specific needs to isolate the ligand binding domain. Errors may occur if your sequence does not contain the conserved amino acids.

    PDB_files - This is the folder containing pdb files extracted from step #6.

    PDB_files_filtered - This is the folder where the trimmed structure will deposited.

    KAIEP 7 – This command is the amino acid sequence plus 7 amino acid residues upstream of the KAIEP motif where the LGB structures are cut.
8. To align the filtered proteins in the same spatial orientation inset the command:
    - *python alignment.py PDB_files_filtered PDB_files_aligned PDB_files_filtered/AF-P49843-F1-model_v4.pdb*.

    PDB_files_filtered - This is the directory where the protein used as reference is.

    PDB_files_aligned - This is the output folder that will contain the aligned pdb files.

    AF-P49843-F1-model_v4.pdb - is the reference protein structure for the alignment. For this example, we used the *Oncorhynchus mykiss* GR, thus, replace *AF-P49843-F1-model_v4.pdb* with your pdb file.
9. To convert aligned files from pdb to pdbqt formats insert the command:
    - *python pdb_to_pdbqt.py PDB_files_aligned PDB_files_pdbqt*.

    PDB_files_pdbqt - This is the output folder containing the pdbqt files.
10. For the chemical structures we downloaded ligand 3D conformer sdf files from PubChem and transferred to the LIGAND_sdf folder in the HPC directory. To convert the ligand file, insert the command:
    - *python sdf_to_pdbqt.py LIGAND_sdf LIGAND_pdbqt*.

    LIGAND_sdf This is the input folder.

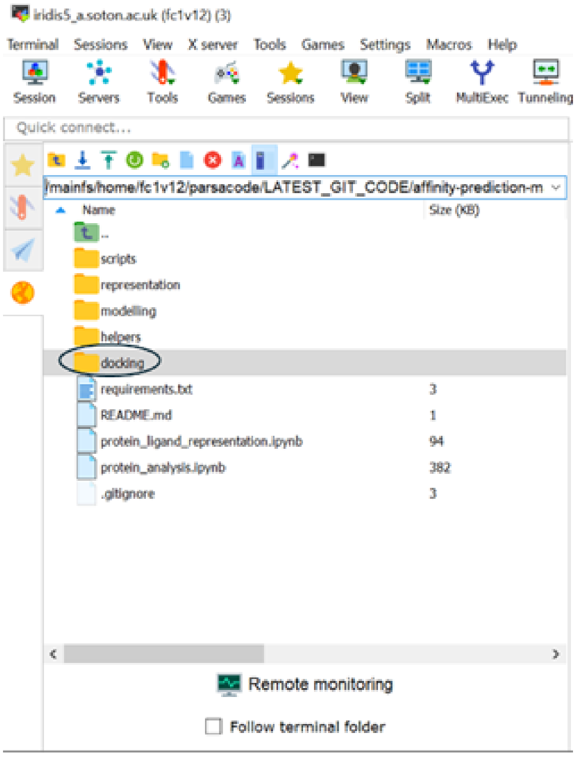
    LIGAND_pdbqt This is the output folder.

LIGAND/RECEPTOR DOCKING EXERCICES

11. Make an output folder in the HPC directory
12. Transfer the *submit_docking_job.slurm* from the *docs* folder to *docking* folder.
13. Write the command line:
    - *conda activate vina2*.

This provides access to the ligand docking program AutoDockVina.

14. Ensure you are in the correct directory containing the docking script. In this example the we have: *cd /mainfs/home/fc1v12/parsacode/LATEST_GIT_CODE/affinity-prediction-main/docking/*

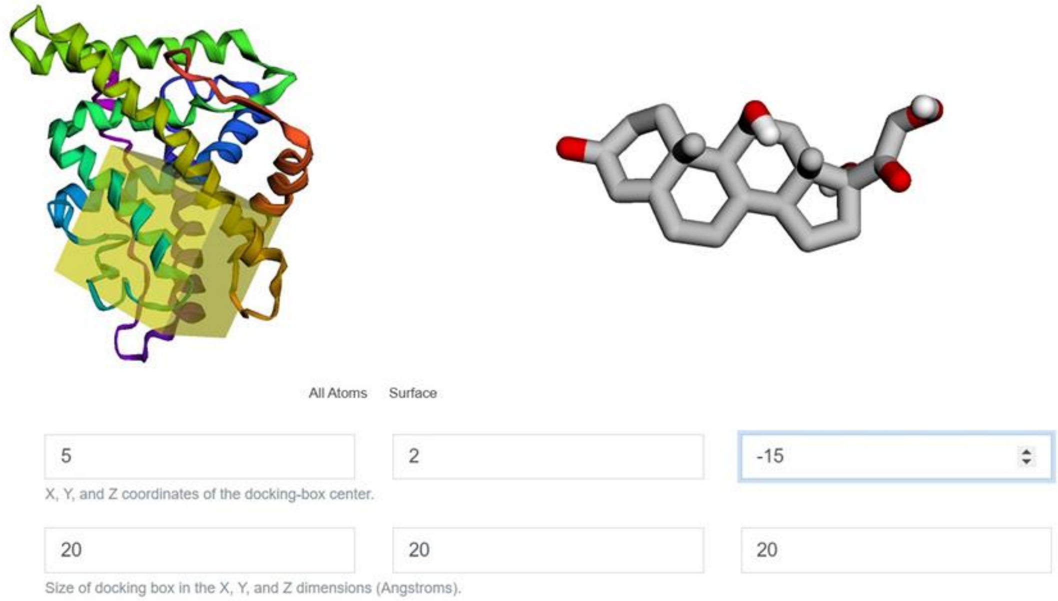

iridis5_a.soton.ac.uk (fc1v12) (3)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling
Quick connect...
Remote monitoring
Follow terminal folder

/mainfs/home/fc1v12/parsacode/LATEST_GIT_CODE/affinity-prediction-m

Name	Size (KB)
..	
scripts	
representation	
modelling	
helpers	
docking	
requirements.txt	3
README.md	1
protein_ligand_representation.ipynb	94
protein_analysis.ipynb	382
.gitignore	3

15. Double click on the submit_docking_job.slurm file. This is the file used for submission of docking exercise and edit the commands. Some of these are specific to this example:
    Line 6: tasks per node. 39 (40 is the maximum value).
    Line 8: Memory 1G - if error messages with memory then increase.
    Line 10: total time estimated to keep running the submitted job, we used 3 h, but our runs finished in less than 30 mins, it depends on the number of proteins and ligands.
    Line 17: always shown '*source activate vina2*'.
    Line 24: This is where the coordinates can be changed. First part– directory of input pdbqt folder; second- directory of ligand folder; third- directory of output folder. After " - -" symbols a specific output directory can be added, e.g output folder from STEP 11. '-e' to edit exhaustiveness; '-c' to edit box coordinates; '-b' for size of the box; '-n' to edit numbers of CPUs (maximum 40). When finishing the editing of slurm file, click on save – save all.

    TO DERIVE THE COORDINATES KNOWLEDGE OF THE LIGNAD BINDING POCKET IS REQUIRED. The example below is for GR.



Box Center	5	2	-15
	X, Y, and Z coordinates of the docking-box center.		
Box Size	20	20	20
	Size of docking box in the X, Y, and Z dimensions (Angstroms).		

16. Submitting the slurm file.

- In the terminal, write *cd and copy address of where the slurm file is located*
- Insert command *sbatch submit_docking_job.slurm*.
- The progress of the running can be checked by writing in the terminal *myqueue* time to time. In 'dockin_log_date.log' file the completed protein-ligand dockings can be checked.

EXTRACTING BINDING AFFINTY RESULTS

In the terminal write the following commands:

17. Insert command cd ..
18. Insert command cd scripts
    - *python extract_vina_results.py/(and add the address of the output folder from the docking exercise) [space] /(address of directory where you want to save the) .cvs*.
19. The results can be opened in Excell.