General Project Idea

Publicly available structure-activity data, docking, and machine learning methods can be leveraged to develop relative binding free energy (RBFE) type scoring functions. In early drug discovery campaigns compound series are developed to study the structure-activity relationship (SAR). From those studies binding affinity data for all synthesized compounds is generated - the compounds can be grouped in compound series - compounds that share a common scaffold. For some of the compounds x-ray crystallography is done to obtain the 3D structure of the ligand-protein complex. A common hypothesis in medicinal chemistry is that similar compounds share the same binding mode; this opens the doors for the usability of template-based docking as a tool to generate 3D interaction models for the elements of the compound series that do not have a corresponding x-ray crystal structure available. This results in an expansion of the structure-activity space available.

Structure-Activity Data Enrichment

Used databases:

ChEMBL: https://www.ebi.ac.uk/chembl/ - bioactivity database

BindingMOAD: http://bindingmoad.org/ - high-quality and biologically relevant database of ligand-protein complexes

SIFTS: https://www.ebi.ac.uk/pdbe/docs/sifts/quick.html - bioinformatics resource to map between Uniprot and PDB

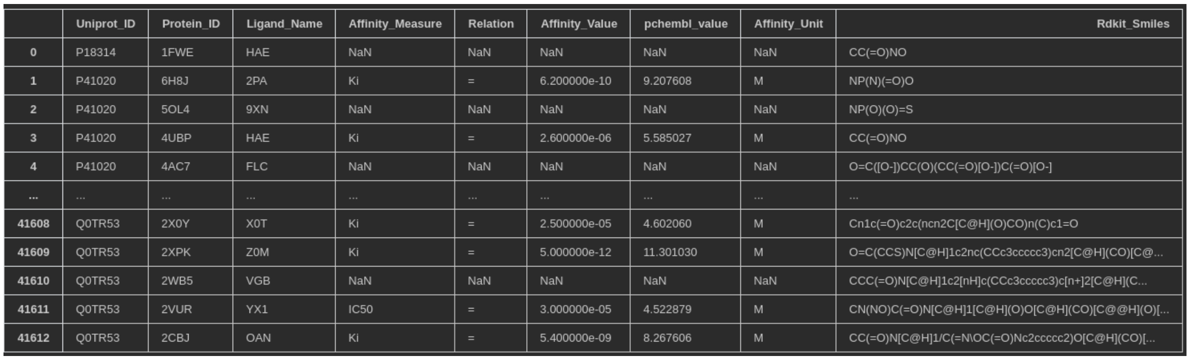
PDB: https://www.rcsb.org/ - public repository of biomolecular structural data

1st: BindingMOAD as the seed for the structure-activity data expansion:

BindingMOAD was transformed to a tidy dataset format for ease of use.

What is a tidy dataset? Every column is a variable. Every row is an observation.

The data set was filtered to only contain ligands defined as biologically relevant ligands according to BindingMOAD and identified by a 3 characters code (HET code) - small molecule ligands.



	Uniprot_ID	Protein_ID	Ligand_Name	Affinity_Measure	Relation	Affinity_Value	pchembl_value	Affinity_Unit	Rdkit_Smiles
0	P18314	1FWE	HAE	NaN	NaN	NaN	NaN	NaN	CC(=O)NO
1	P41020	6H8J	2PA	Ki	=	6.200000e-10	9.207608	M	NP(N)(=O)O
2	P41020	5OL4	9XN	NaN	NaN	NaN	NaN	NaN	NP(O)(O)=S
3	P41020	4UBP	HAE	Ki	=	2.600000e-06	5.585027	M	CC(=O)NO
4	P41020	4AC7	FLC	NaN	NaN	NaN	NaN	NaN	O=C([O-])CC(O)(CC(=O)[O-])C(=O)[O-]
...	...	...	...	...	...	...	...	...	...
41608	Q0TR53	2X0Y	X0T	Ki	=	2.500000e-05	4.602060	M	Cn1c(=O)c2c(ncn2C[C@H](O)CO)n(C)c1=O
41609	Q0TR53	2XPK	Z0M	Ki	=	5.000000e-12	11.301030	M	O=C(CCS)N[C@H]1c2nc(CCc3ccccc3)cn2[C@H](CO)[C@...
41610	Q0TR53	2WB5	VGB	NaN	NaN	NaN	NaN	NaN	CCC(=O)N[C@H]1c2[nH]c(CCc3ccccc3)c[n+]2[C@H](C...
41611	Q0TR53	2VUR	YX1	IC50	=	3.000000e-05	4.522879	M	CN(NO)C(=O)N[C@H]1[C@H](O)O[C@H](CO)[C@@H](O)[...
41612	Q0TR53	2CBJ	OAN	Ki	=	5.400000e-09	8.267606	M	CC(=O)N[C@H]1/C(=N\OC(=O)Nc2ccccc2)O[C@H](CO)[...

Size of data set:

- Total number of rows: 41613
- Total number of different protein targets: 8235

- Total number of different protein-ligand complexes: 28355
- Total number of different protein-ligand complexes that have activity data: 11691

Location of folder with code and data relative to this task:

/home/fol007/PhD_Project/BindingMOAD_2019

Location of the data set:

/home/fol007/PhD_Project/BindingMOAD_2019/processing/bm_2019_valid_activities_processed.csv

2nd: Retrieve from ChEMBL all entries that correspond to targets in BindingMOAD

I used ChEMBL release 26.

- Only compounds binding to proteins present in the BindingMOAD data set
- Filters used:
- chembl_26.assays.confidence_score = 9
- chembl_26.activities.potential_duplicate = 0
- ((chembl_26.activities.data_validity_comment is null)
  or (chembl_26.activities.data_validity_comment = 'Manually validated'))
- chembl_26.molecule_dictionary.chirality != 0
- chembl_26.activities.pchembl_value is not null
- ((chembl_26.activities.standard_type = 'IC50') or (chembl_26.activities.standard_type = 'Kd') or (chembl_26.activities.standard_type = 'Ki'))

Add location of data and code for this task:

/home/fol007/Documents/GitHub/ChEMBL_plus_BindingMOAD

Relevant folders inside the directory above:

ChEMBL; ChEMBL_plus_BindingMOAD

ChEMBL/retrieve_activities_from_ChEMBL.ipynb -> notebook with the SQL statements used to retrieve activities from ChEMBL 26.

ChEMBL/ChEMBL_activities -> inside I have a .csv file for each different protein target in BindingMOAD

Size of data set:

Number of targets that have activity data in ChEMBL: 1071

Number of targets that do not have activity data in ChEMBL: 6866

How many of the activities correspond to complexes already present in BindingMOAD?

How many ligands in ChEMBL can be grouped together with ligands in BindingMOAD as being part of the same compound series?

Can a ML model automatically select which ligands in BindingMOAD serve as templates for template-based docking of ChEMBL ligands?

How many new structure-activity data points can we generate with template-based docking if we use the data in BindingMOAD as templates for the ligands in ChEMBL?

Template-based docking as a tool for high accuracy docking pose generation.

Questions:

What is the domain of application of template-based docking?

How to pair a template to each compound to dock automatically?

Potential solution: Train a Machine Learning model to select a template for each compound that must be docked.

Initial training data: use the targets in Astex Diverse Set as the starting set of proteins to generate training data. Training data is generated with an all-against-all cross-template-based docking approach.

1st: which targets? Targets from Astex Diverse Set

Astex Diverse Set can be found in https://www.ccdc.cam.ac.uk/support-and-resources/downloads/ under “Validation Test Sets”.

For each target protein in Astex Diverse Set retrieve all the instances found in BindingMOAD. There are 80 different protein targets in Astex Diverse Set.

- Resolution <= 2.5 Å
- No engineered mutations
- Just one instance of the ligand per chain
- If there is more than one PDB entry for the same protein-ligand complex that pass the above mentioned filters, choose the one with the highest resolution (to avoid redundancy)
- RDKit must be able to process the ligand

After expanding we obtained 2278 entries. The number of protein-ligand complexes per protein target is not constant. The minimum is 1 and the maximum is 303. The average is 28 and the median is 12.

Location of folder with code and data relative to this task:

/home/fol007/PhD_Project/Template_Based_Docking_Project_GitRepo/Expanded_Astex_Diverse_Set

expanded_Astex_with_smiles.csv -> expanded file

pdb_files -> folder with the .pdb files downloaded from RCSB PDB

reference_ligands -> folder with the .sdf files generated from the ligand HET entries in the PDB files

smiles_to_genconformer -> folder with .smi files with the smiles of the ligands

2nd: filter and process the structural data:

- If the biological assembly plays a role in ligand binding drop it – using the asymmetric unit directly makes it easier (this should only be the case if very few cases are found, otherwise use the following approach https://github.com/f-krull/pdb-merge-bio to generate a biological assembly).

Location of folder with code and data relative to this task:

/home/fol007/PhD_Project/Template_Based_Docking_Project_GitRepo/check_if_assymetric_unit_is_valid

validity_of_AU_checked.csv -> file that allows to filter out the protein-ligand complexes for which the asymmetric unit model is not valid.

- If there are ligands that are not biologically relevant that are interacting with the ligands of interest, they should be removed

Location of folder with code and data relative to this task:

/home/fol007/PhD_Project/Template_Based_Docking_Project_GitRepo/notebook_to_remove_bad_hets

/Home/siv32/fol007/new_cross_docking_protocol/data/pdb_files_edited -> folder with the edited .pdb files without biologically irrelevant ligands.

- The quality of the interaction between the ligand-protein that is going to be used as the confirmed pose is accessed by HyDE optimization of the ligand pose. If the pose deviates from the original by less than 1 Å then it is kept.

Location of folder with code and data relative to this task:

/home/fol007/PhD_Project/Template_Based_Docking_Project_GitRepo/hyde_optimize_templates

It was run in the cluster (FRAM): /cluster/home/fol007/optimize_confirmed_with_HYDE/07_03_2021

/Home/siv32/fol007/new_cross_docking_protocol/data/optimized_ligands -> folder with HyDE optimized ligands

/home/fol007/PhD_Project/Template_Based_Docking_Project_GitRepo/notebook_filter_bad_optimized_poses

/Home/siv32/fol007/new_cross_docking_protocol/data/expanded_Astex_with_smiles.csv -> The result was adding a column to the file '../data/expanded_Astex_with_smiles.csv' with the values of the RMSD between the original pose and the HyDE optimized pose - column name: 'hyde_rmsd'.

3rd: matching up template – ligand-to-dock for cross-docking:

- Each bind to the same target and in the same pocket – to verify that they bind to the same pocket, alignment of the receptors of the two co-crystals is done with PyMOL align function: if the alignment rmsd is less than 1 Å and the distance between the two closest atom of the ligands is less than 1 Å, then the pair is accepted for template-based docking.

Location of folder with code and data relative to this task:

/home/fol007/PhD_Project/Template_Based_Docking_Project_GitRepo/Align_Receptors

/home/fol007/PhD_Project/ML_template_based_docking/template_based_docking_15_03_21/check_which_receptor_pairs_match/dictionary.csv -> file with rmsd between receptors and distance between the two closest atoms of the ligands.

4th: preparing inputs for template-based docking with FlexX:

- For each template – ligand-to-dock, constrained conformation of the ligand-to-dock was generated so that the MCS of both ligands superposes with a RMSD of less than 0.5 Å (the current applied algorithm does not guarantee that).
- The template – ligand-to-dock pair maximum common substructure (MCS) has minimum 5 heavy atoms (mcs is found via FindMCS from RDKit).

Location of folder with code and data relative to this task:

/home/fol007/PhD_Project/Template_Based_Docking_Project_GitRepo/self_cross_TBD_cluster_run

generate_conformers_cl.py -> script to generate conformers to template-based dock with

The task above was run on FRAM (fol007@fram.sigma2.no):

/cluster/projects/nn9376k/template_based_docking_15_03_21/conformers_to_dock -> folder with the adequate conformers for each template

5th: template-based docking and scoring:

- Generate docking and scoring definition files to be used by FlexX and HyDE – allows to specify a larger docking site and scoring site.

Location of folder with code and data relative to this task:

/home/fol007/PhD_Project/Template_Based_Docking_Project_GitRepo/Create_Docking_Definitions_SeeSAR

/home/fol007/PhD_Project/Template_Based_Docking_Project_GitRepo/notebook_to_create_definition_files

/Home/siv32/fol007/new_cross_docking_protocol/data/definition_files -> folder with the generated definition files

- Use FlexX for template-based docking (number of conformers to use per ligand should be more than 1, but I only used 1) - output a maximum of 3 poses.
- Use HyDE to score the complexes
- Choose the highest scoring complex according to HyDE score (if the HyDE score is the same choose the best docking score)
- Align the pose of the docked ligand to the crystallographic pose of the same ligand

Location of folder with code and data relative to this task:

/home/fol007/PhD_Project/Template_Based_Docking_Project_GitRepo/self_cross_TBD_cluster_run

The task above was run on FRAM (fol007@fram.sigma2.no):

/cluster/projects/nn9376k/template_based_docking_15_03_21

/cluster/projects/nn9376k/template_based_docking_15_03_21/aligned -> folder with final template-based docked ligands aligned to the respective validation poses and RMSD values calculated with DockRMSD.

- The only valid template-based docking are the ones for which the mcs of the template and docked-ligand superpose (what is the threshold? For example, less than 1 Å?).

Location of folder with code and data relative to this task:

/home/fol007/PhD_Project/Template_Based_Docking_Project_GitRepo/filter_out_failed_tbd

The task above was run on FRAM (fol007@fram.sigma2.no):

/cluster/projects/nn9376k/template_based_docking_15_03_21

The original output is: /cluster/projects/nn9376k/template_based_docking_15_03_21/good_pairs

It was copied to:
/home/fol007/PhD_Project/ML_template_based_docking/template_based_docking_15_03_21/data/rmsd_between_template_docked_mcs

And then processed to the following output:
/home/fol007/PhD_Project/ML_template_based_docking/template_based_docking_15_03_21/data/rmsd_between_template_docked_mcs.csv

- Initially I used DockRMSD to calculate the RMSD between the docking pose and the validation pose, but I changed to using RDKit (for consistency, then I can always use .sdf file format)

Location of folder with code and data relative to this task:

/cluster/projects/nn9376k/calculate_RDKitRMSD_of_template_based_docking_15_03_21

/home/fol007/PhD_Project/ML_template_based_docking/template_based_docking_15_03_21/data/rmsd_rdkit_values_base.csv -> .csv file with the RDKit calculated RMSD for all template-based docking pairs.

- Free cross-docking is also performed to be able to compare the performance of template-based cross-docking against a baseline

Location of folder with code and data relative to this task:

/home/fol007/PhD_Project/Template_Based_Docking_Project_GitRepo/cross_free_docking_cluster_run

The task above was run on FRAM:

/cluster/projects/nn9376k/cross_free_docking_cluster_run

Output copied to the folder:
/home/fol007/PhD_Project/ML_template_based_docking/free_docking_13_04_21/done

6th: train a classification model to select the template ligand

- Data featurization for "classical ML":
    - Lipinsky type features (https://www.rdkit.org/docs/source/rdkit.Chem.Lipinski.html)
    - Fingerprint Similarity type features
    - Receptor-Template interaction type features (a proxy can be SASA, but the features can also be explicitly counted with tools like PLIP)

  Location of folder with code and data relative to this task:

  /home/fol007/PhD_Project/ML_template_based_docking/SASA_calculation

  /home/fol007/PhD_Project/ML_template_based_docking/
  DecisionTree_TBD_RMSD_prediction_2/notebooks/featurization.ipynb

  /home/fol007/PhD_Project/ML_template_based_docking/
  DecisionTree_TBD_RMSD_prediction_2/notebooks/featurization_add_sasa.ipynb

  /home/fol007/PhD_Project/ML_template_based_docking/PLIF

- Divide dataset into train, validate and test sets

  Location of folder with code and data relative to this task:

/home/fol007/PhD_Project/ML_template_based_docking/DecisionTree_TBD_RMSD_prediction_2/
notebooks/train_and_validation_sets_version2.ipynb

- What model to use? Start with the Random Forest Model.

  Location of folder with code and data relative to this task:

/home/fol007/PhD_Project/ML_template_based_docking/DecisionTree_TBD_RMSD_prediction_2/
notebooks/RandomForest_Classification_inconclusives_2-3.ipynb

- How to assess model performance? Test the model on a held-out data set. Compare with free cross-docking results.
- How to assess feature importance? Permutation importance, Leaf purity, …?
- How to assess the impact of the "fuzzyness" of the classification barrier? Ideally, we could have the ground truth (but that implies visual check), otherwise choose data points for which the proxies work better. Another solution is to use multiple proxies that complement each other, for example SuCOS + RDKit_RMSD (requires calibration).