**Iterative Knowledge-Based Protein-Ligand Scoring Function: Implementation and Improvements**

Thanh Lai

Protein-ligand binding is a process that occurs in all biological scope—from endogenous ligand-receptor binding for signal transduction, to selective drugging of a protein target. It is of utmost importance to understand the thermodynamics and kinetics that govern such process, as doing so has implications for fields such as drug discovery. One can gauge the stability of a protein-ligand complex by considering only the thermodynamics of the binding, however this consideration has multiple level of theory, from empirically-derived scoring functions to *ab inito* calculations. In the early stages of a computational drug discovery program, scoring functions are favored as a quick method of filtering thousands of protein-ligand (ligand = drug candidate) complexes. However, as scoring functions trade accuracy for computing speed, it is not far-fetched to say that the trajectory of a drug discovery program greatly hinges on identifying strong drug candidates from a pool of several thousands. Therefore, there is a need to improve on scoring functions as it has the ability to shape the trajectory of these drug discovery programs.

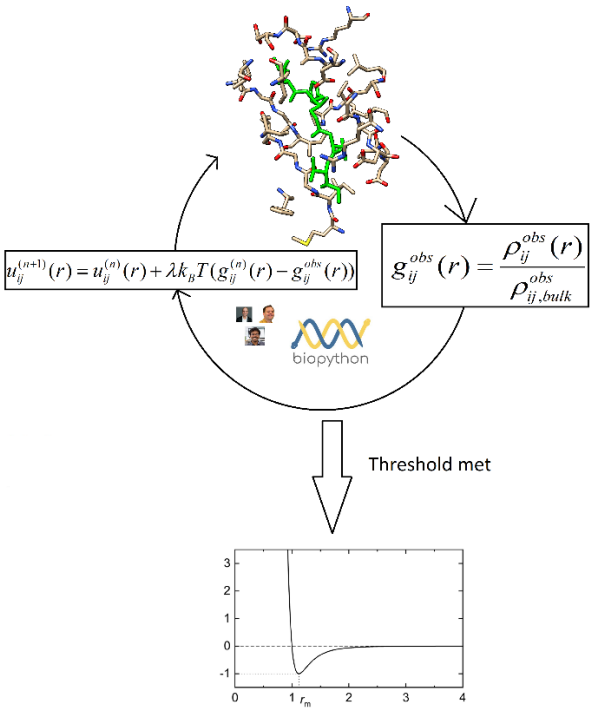
There are four classifications of scoring functions: physics-based, empirical-based, descriptor-based, and knowledge-based. Physics-based scoring functions generally score protein-ligand complexes by the summation of the following energy terms: van der Waals, electrostatic, hydrogen bonding, and the free energy of de-solvation. This scoring type is limited by the approximations in some of the energy terms and neglecting the entropy contribution. Descriptor-based scoring functions are machine learning models that are trained to predict either a binary result (real vs. decoy) or a regression result (free energy). While descriptor-based scoring functions are known to be accurate, they require massive training data and is considered as a "black box" method. Empirical-based scoring functions implements intuitive reward and penalty scores based on interaction patterns (hydrophobicity, hydrogen bonding, etc.). As the method is based on the summation of terms describing specific interaction patterns, it is difficult to capture all forms of protein-ligand interactions. Lastly, knowledge-based scoring is a statistical approach that creates pairwise interaction potentials by counting the number density of pairwise atoms relative to a reference. A large limitation of this approach is obtaining the reference state.

The iterative knowledge-based protein ligand scoring function (ITScore), as described by Huang et. Al., avoids the calculation of the reference state by iteratively improving the trial potentials until native complexes are discriminated from decoy complexes. The main idea is as follows: from a set of protein-ligand complexes, calculate the pair distribution function for each ligand atom protein atom pair, which is a function of the number density in each spherical shell centered at the ligand. The pair distribution functions, as well as AMBER forcefield parameters, are used to derive initial interaction potentials. Next, through a docking program, generate an ensemble of ligand-protein poses for each complex. Boltzmann-weighted pair distribution functions for each ligand protein atom pairs are calculated for all poses (similar functional form to the pair distribution function described beforehand except scaled by a Boltzmann factor). At the beginning, the Boltzmann factor uses the initial potential to calculate the energy, but in future iterations it uses the trial/updated potentials. During the iterative procedure, use the current trial interaction potential to determine the best ligand-protein complexes from the training set (decoys + natives). If the success rate fails to meet a threshold, recalculate the Boltzmann-weighted distribution function, update the trial potential based on the distribution function, and use the updated trial potential to predict the best complexes from the training set. This procedure repeats until the success rate meets a convergence criterion. The first (and main) part of my project is to replicate this procedure on python. I will use the open-source

python library Biopython for most of the algorithm implementation. Biopython's ease of use will ensure the success of the project, as it makes it easy to iterate through atoms in a PDB file. The replicated code base will be trained and tested on the same PDB set that is used by Huang et. Al. If I have time, I want to do several more things that may improve the model: use updated AMBER forcefield parameters for the initial potentials, train the model on a bigger training set (more natives from PDBBind data set, more decoys from "Blur" ensemble generator algorithm developed by the Merz group) with a stricter convergence criterion, and parallelize the code. I believe these changes will improve the model since the model is very dependent on the quality of the training PDBs, as well as the forcefield parameters used to derive the initial potentials. Parallelizing the code is mainly for fun/bonus (and synergizes well with what I am currently learning in CMSE401) but could also be useful if the model ever needs to be retrained on a better/larger PDB set in the future.

The success of this project will have several implications. Firstly, replicating the code base means the code for the algorithm is now open source and can be readily modified in the future. Improving the model by the ideas mentioned above could be very useful for computational drug discovery programs. As mentioned previously, these programs screen thousands of potential drug candidates through docking and scoring. An improved scoring function, even if only marginal, could be the difference between finding a truly effective drug molecule or discarding it under the pretense of inactivity.

**Graphical Abstract:**




**Papers:**

(1) Huang, S.-Y.; Zou, X. An Iterative Knowledge-Based Scoring Function to Predict ProteinLigand Interactions: I. Derivation of Interaction Potentials. J. Comput. Chem. 2006, 27 (15), 1866–1875. https://doi.org/10.1002/jcc.20504.

(2) Ucisik, M. N.; Zheng, Z.; Faver, J. C.; Merz, K. M. Bringing Clarity to the Prediction of Protein Ligand Binding Free Energies via “Blurring.” J. Chem. Theory Comput. 2014, 10 (3), 1314 1325. https://doi.org/10.1021/ct400995c.

(3) Tian, C.; Kasavajhala, K.; Belfon, K. A. A.; Raguette, L.; Huang, H.; Migues, A. N.; Bickel, J.; Wang, Y.; Pincay, J.; Wu, Q.; Simmerling, C. Ff19SB: Amino-Acid-Specific Protein Backbone Parameters Trained against Quantum Mechanics Energy Surfaces in Solution. J. Chem. Theory Comput. 2020, 16 (1), 528–552. https://doi.org/10.1021/acs.jctc.9b00591.

(4) Cock, P. J. A.; Antao, T.; Chang, J. T.; Chapman, B. A.; Cox, C. J.; Dalke, A.; Friedberg, I.; Hamelryck, T.; Kauff, F.; Wilczynski, B.; De Hoon, M. J. L. Biopython: Freely Available Python Tools for Computational Molecular Biology and Bioinformatics. Bioinformatics 2009, 25 (11), 1422–1423. https://doi.org/10.1093/bioinformatics/btp163.

(5) Thomas, P. D.; Dill, K. A. Statistical Potentials Extracted from Protein Structures: How Accurate Are They? J. Mol. Biol. 1996, 257 (2), 457–469. https://doi.org/10.1006/jmbi.1996.0175.

(6) Liu, J.; Wang, R. Classification of Current Scoring Functions. J. Chem. Inf. Model. 2015, 55 (3), 475–482. https://doi.org/10.1021/ci500731a.

(7) Huang, S.-Y.; Zou, X. An Iterative Knowledge-Based Scoring Function to Predict ProteinLigand Interactions: II. Validation of the Scoring Function. J. Comput. Chem. 2006, 27 (15), 1876–1882. https://doi.org/10.1002/jcc.20505.