Protein-Protein Complexes Binding Affinity Prediction using Random Forest Regressor

Master's Degree Thesis Summary

Beryl Ramadhian Aribowo
Department of Computational Science
Kanazawa University
Kanazawa, Japan
berylramadhian@gmail.com

Supervisor I: Hidemi Nagao
Department of Computational Science
Kanazawa University
Kanazawa, Japan
nagao@wriron1.s.kanazawa-u.ac.jp

Supervisor II: Kawaguchi Kazutomo
Department of Computational Science
Kanazawa University
Kanazawa, Japan
kkawa@wriron1.s.kanazawa-u.ac.jp

***Abstract*—Understanding the structural information of a protein complex is important to infer its biological functions. The structure of protein complexes are composed of multiple chains bound together. The success of protein docking computationally relies on the accuracy of a *Scoring Function*. Molecular scoring functions rank and judge the docking process and quality of the results. Within this research, the scoring function modelled using Random Forest is applied to predict binding affinity values of protein-protein complexes. Investigation on the combination of biophysical features are also conducted, which yields different type of models. The best prediction model is the beta-type model which incorporates the interactions of hydrophobics and electrostatics as features. The devised model framework should be applicable as the baseline for binding affinity prediction on diverse protein-protein complexes set.**

***Index Terms*—protein-protein complexes, molecular scoring function, binding affinity, machine learning algorithm**

I. Introduction

Molecular docking refers to the process of predicting the binding pose of two or more molecules. Protein-ligand and protein-protein complexes are prominent molecular docking objects due to the importance for the advancement of drug discovery. Computational molecular docking in particular is applied on the prediction of quaternary protein structures. Molecular docking consists of two main processes, the first is the docking of two or more molecules, which often represented as searching-problem or mathematical optimization problem, then the second is the process of scoring the docking results, which is referred as the scoring function [1]. Scoring Function (SF) scores the results of docking process using pre-determined mathematical functions. SF takes the outputs of the earlier docking process, such as the three-dimensional structure conformations of the docked molecules, then evaluate and judge whether the docked conformation is correct, the conformation results from docking-prediction process are not always necessarily successful if the docking process is not able to differentiate the correct poses from the incorrect ones [1]. SF is useful on several tasks on computational drug discovery building blocks, such as virtual screening (selecting molecules that are likely to bind), lead optimization (predicting the best docking position) [2], and binding affinity prediction.

Binding affinity prediction is the process of predicting how well the the binding pose between the target molecule and ligand molecules according to binding affinity values such as inhibition constant K_i, dissociation constant K_d, and IC_{50}, where higher value corresponds to better affinity between the molecules. Within this research, we propose an investigation on protein-protein binding affinity prediction by using machine learning approach, in particular Random Forest Algorithm (RF) is employed on performing the regression task. The feature extraction process is inspired by the framework of protein-ligand binding affinity prediction by Ballester and Mitchell (2010), referred as RF-score [2], with several modifications to accommodate the higher number of interacting amino acid chains such as modification on the sum of interactions across all protein chains combination. This research also investigates the effects of additional interaction features such as the backbone carbon interaction on the predictive capability. Several binding affinity values of experimental and docked structures are also inferred, particularly on the protein-protein structures which are docked using Zdock [3], to capture the comparison between the prediction model on unseen data.

II. Methodology

A. Dataset

The dataset used for the purpose of model-building and validation is PDBBind [4]. PDBBbind compiles experimentally determined structures alongside the binding affinity values. On the 2018 version of PDBbind, 19,588 bio-molecular complexes are available, including protein-ligand (16,151), nucleic acid-ligand (125), protein-nucleic acid (896), and protein-protein complexes (2,416). Currently PDBbind contains the highest number of protein-protein complexes with experimental binding affinity data compared to several other protein-protein complexes data.

B. Feature Extraction

The feature is represented by the distance of inter-molecular interactions between the most common heavy atom-types found in amino-acid chains (Carbon, Nitrogen, Oxygen, Sulphur). The inter-molecular interaction is defined as the total occurrences of atom-atom distance between protein chains within a cut-off constant. This definition and formula of inter-molecular interaction was first employed for protein-ligand interaction by Ballester, et al. [2]. The formula modification applied for protein-protein inter-molecular interaction is as follows:

$$x_{Z(P_q(j)),Z(P_r(i))} \equiv \sum_{k=1}^{K_j}\sum_{l=1}^{L_i}\Theta(d_{cutoff} - d_{kl}) \tag{1}$$

this function counts the occurrences of interaction between k-th atom with i-type in q-th chain and l-th atom with j-type in r-th chain where the distance d_{kl} is within d_{cutoff} (in Angstrom unit), Z is a function that returns the atom number, and Θ is a Heaviside step function. The total interactions of each features within a protein-protein complex is the sum of that particular feature value from each amino acid chains pairing combinations. For example, suppose there are four amino acid chains $S = \{P_1, P_2, P_3, P_4\}$ present in a protein-protein complex, the total of available amino acid chains pairing combinations of the complex is $\binom{|S|}{2} = 6$.

Other than common heavy atom-types, hydrophobic and electrostatic patches are also considered, as the biochemical events of protein chains are steered by several forces including hydrophobic and electrostatic forces. Hydrophobic patches consist of eight amino-acid types {Ala, Val, Ile, Leu, Met, Phe, Tyr, Trp} and charged patches which consist of five amino-acid types {Arg, His, Lys, Asp, Glu} are also considered, these hydrophobic-hydrophobic and electrostatic interactions are represented by inter-molecular interaction of carbon alpha ($C_{\alpha H}$ and $C_{\alpha A}$ respectively).

Finally, each element in the feature vector of a protein-protein complex is represented by the value of inter-molecular interaction based on the combination of the atomic number referencing the atom types:

$$\begin{aligned}\vec{x} = {} & x_{6,6}, x_{6,7}, x_{6,8}, x_{6,16}, x_{7,6}, x_{7,7}, \\ & ....x_{16,16}, x_{C_{\alpha H},C_{\alpha H}}, x_{C_{\alpha A},C_{\alpha A}} \in \mathbb{N}^{18}\end{aligned} \tag{2}$$

Meanwhile, the value of binding affinity K ($K_i \cup K_d$) is transformed by logarithm function to decrease the range of the data [2]. The set of protein-protein complexes after the process of feature extraction is represented by:

$$D = \{(y^{(n)}, \vec{x}^{(n)})\}_{n=1}^{N} \; y \equiv -\log_{10} K \tag{3}$$

C. Random Forest Regression

Random Forest Algorithm (RF) is employed as the regression model. RF is a type of supervised-machine learning algorithm. RF is an ensemble of decision trees. Each trees employs CART algorithm for training [5]. Regression tree is often pruned by the stopping criteria, such as: the number of samples within each leaf is less than threshold, the residual error decrease when splitting is under threshold, or maximum tree depth has been reached. Given a set of data $D = (\vec{x}_1, y_1), ..., (\vec{x}_n, y_n)$, where $\vec{x}_i \in \mathbb{R}^{18}$ and $y_i \in \mathbb{R}$. A binding affinity prediction of over $\vec{x}$ of a complex, with the number of feature m_{try}, on a RF with P number of regression trees is as follows [2]:

$$RF(\vec{x}^{(n)}; m_{try}) \equiv \frac{1}{P}\sum_{p=1}^{P} T_p(\vec{x}^{(n)}; m_{try}) \; T_p : \mathbb{N}^n \rightarrow \mathbb{R}^+ \forall p \tag{4}$$

RF algorithm framework includes feature importance calculation during the model-building phase. The feature matrix is evaluated and ranked based on the contribution on the prediction model. Feature importance calculation within RF utilizes the averaged error decrease (mean impurity decrease) over all trees when splitting a node based on a feature. RF's process on feature importance calculation implies that the feature ranking occurs during training phase. A more important feature implies that the error decrease is higher when the node is split based on that particular feature. Another method to rank the feature importance is by permuting the feature on the set (referred as Permutation Feature Importance) and observing the prediction error difference between default feature matrix and permuted feature matrix.

D. Parallelization

Parallelization serves to increase the efficiency of the computation processes. Algorithm run in parallel will almost always shorten the total time required to be finished except on some cases where splitting the data and computation blocks become bottleneck instead. Parallelization make use and take advantage of the multiple cores/threads handling capability of current generation of computer CPUs. The main concept of parallelization is as follows:

- Splitting the data and/or algorithm, such as arrays, matrices, or loops.
- Distributing the data and/or algorithm into multiple cores/threads.
- Joining/collecting all of the distributed processes across cores/threads into single process.

The most important computation block that needs to be parallelized is the feature extraction due to the high time complexity (polynomial complexity) of approximately around $O(n^4)$.

III. Computational Experiments

A. Parameters

Two types of model were deployed: the alpha-type and beta-type. Alpha type model is the first prototype of the model, it includes $|\vec{x}^{(n)}| = 16$ features. The next iteration results in beta-type model, it consists of $|\vec{x}^{(n)}| = 18$ total features, which includes the hydrophobic-hydrophobic and electrostatic interactions. Another parameter is the cut-off distance d_{cutoff}, the values are $\{4, 8, 12, 16\}$Å.

For each different set of parameters, the feature needs to be re-extracted. The time consumed for one feature-extraction process of 2416 protein-protein complexes is around 1 week on a 6-core 12-threads Ryzen 5 2600x 3.9Ghz CPU (OC) with 16 GB of RAM, using Python 3.7 and by parallelizing the process, the source code is available at https://github.com/berylgithub/ppbap. Without parallelization the feature extraction calculation might take more than a month to finish.

B. Alpha-type and beta-type model benchmark

There are two types of model employed. The first prototype-the alpha type model results in $R = 0.98$ and $RMSE = 0.66$ on training set as shown in figure 1. Although high predictive capability on training set does not always imply high predictive capability on test set, in most cases of machine learning problems, this fact indicates model-overfitting. As shown in figure 2, alpha-type model has lower correlation and higher error, the model overfits on protein-protein complex test-set. This occurs most likely due to the high variance of the protein-protein complex data, as protein-protein complexes can contain more than two chains, which produces much wider range of feature values compared to protein-ligand data. The feature importance data is utilized on interpreting the model's decisions during training process. Alpha-type model feature importance data at figure 3 shows that the importance across all common heavy atom features are almost uniform, in other words no significant feature detected, this might hint that the model is unable to make decision clearly based on the present features.

The beta-type models improve the predictive capability of alpha-type model by including hydrophobic ($x_{C_\alpha H, C_\alpha H}$) and electrostatic ($x_{C_\alpha A, C_\alpha A}$) terms as the feature. By setting the cut-off $d_{cutoff} = 12$ as controlled variable, beta-type model improves the performance of the previous alpha model. Adding the inter-molecular interactions of $x_{C_\alpha H, C_\alpha H}$ and $x_{C_\alpha A, C_\alpha A}$ increases the prediction correlation on experimental data as demonstrated at figure 4. By examining the feature importance at figure 5, the model recognizes the significance of $x_{C_\alpha H, C_\alpha H}$ and $x_{C_\alpha A, C_\alpha A}$ features, both of these features are ranked higher compared to common heavy atoms combination features. Although technically $x_{C_\alpha H, C_\alpha H}$ and $x_{C_\alpha A, C_\alpha A}$ are the subset of $x_{6,6}$ (the carbon-alphas interactions are already included in the calculation of carbon-carbon interactions), indicating the explicit correlation between the carbon variables. As shown in figure 6 the permutation feature importance of the test set agrees with the feature importance of RF on training data regarding the $x_{C_\alpha H, C_\alpha H}$ and $x_{C_\alpha A, C_\alpha A}$ features. Finally, all other cut-off distances features are employed and tested by using the beta-type model, as shown in table I.

IV. Conclusion

A machine learning (RF) prediction model alongside the feature analysis is presented within this research. By using RF, the pattern of structural data from protein-protein complexes are analyzed which in turn transformed into a prediction

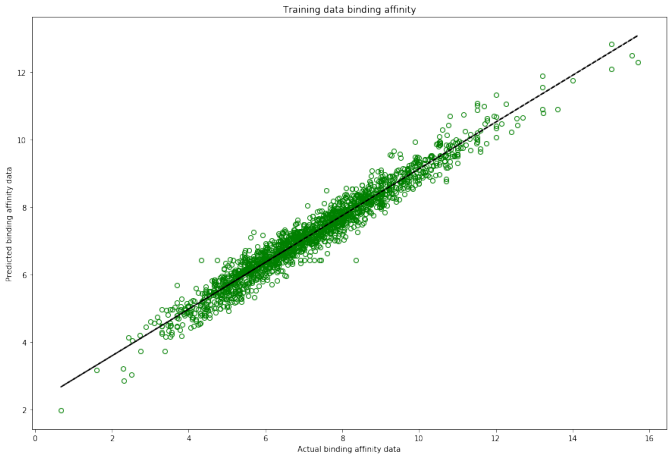

Fig. 1. Correlation between actual and predicted binding affinity of training data on alpha-type model with 12Å cut-off, with $R = 0.98$ and $RMSE = 0.66$.

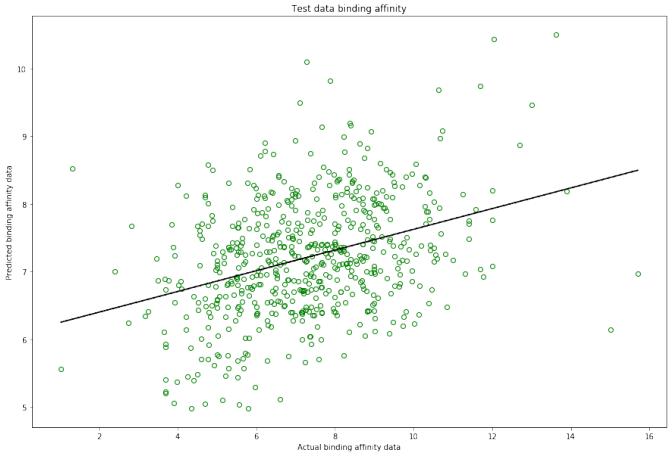

Fig. 2. Correlation between actual and predicted binding affinity of test set on alpha-type model with 12Å cut-off, with $R = 0.35$ and $RMSE = 1.87$.

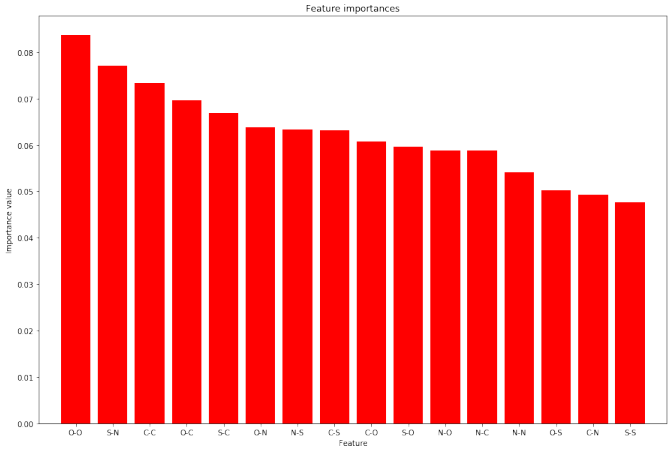

Fig. 3. Feature importance of alpha-type model with 12Å cut-off.

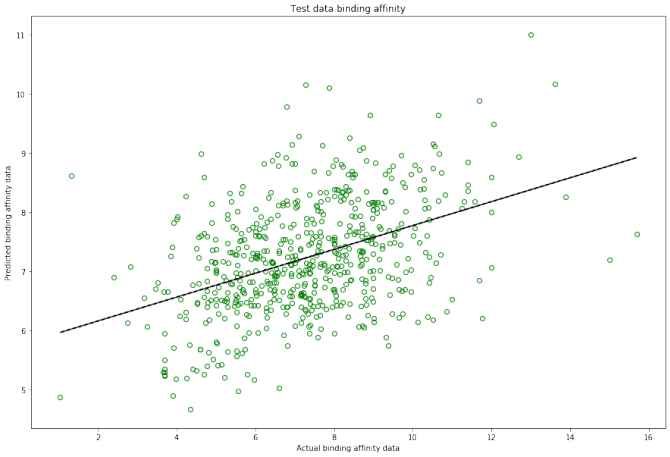

Fig. 4. Correlation between actual and predicted binding affinity of test set on beta-type model with 12Å cut-off, with $R = 0.43$ and $RMSE = 1.8$.

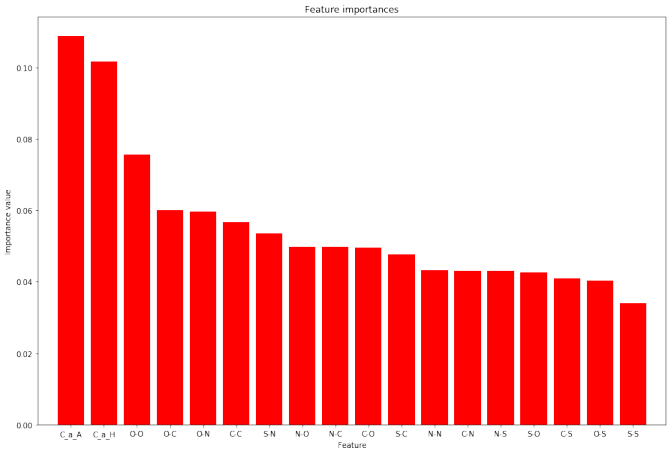

Fig. 5. Feature importance of beta-type model with 12Å cut-off.

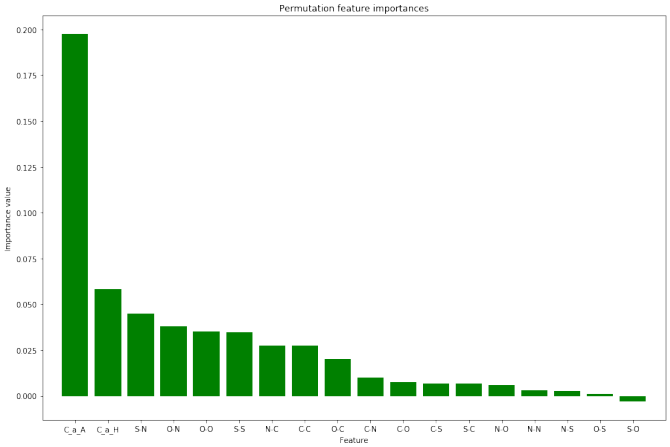

Fig. 6. Permutation feature importance on test-set of beta-type model with 12Å cut-off.

TABLE I
R-VALUE AND $RMSE$ OF BETA-TYPE MODELS ON DIFFERENT CUT-OFF PARAMETERS.

cut-off (Å)	R-value	RMSE
4	0.35	1.87
8	0.42	1.81
12	0.43	1.8
16	0.46	1.76

model, the aforementioned features are the inter-molecular interaction of various atom-types. Feature importance coupled with permutation feature importance calculations are employed to analyze the feature significance on predictive capability. In general it is discovered that hydrophobic and electrostatic interactions play an important role in the prediction of binding affinity values of protein-protein complexes. Overall the prediction model framework still requires many improvements, as protein-protein complexes data mostly contain high variability which imply high difficulty on prediction process.

REFERENCES

[1] Kitchen, D. B., Decornez, H., Furr, J. R., & Bajorath, J. (2004). Docking and scoring in virtual screening for drug discovery: methods and applications. Nature Reviews Drug Discovery, 3(11), 935–949. doi: 10.1038/nrd1549.

[2] Ballester, P. J., & Mitchell, J. B. O. (2010). A machine learning approach to predicting protein–ligand binding affinity with applications to molecular docking. Bioinformatics, 26(9), 1169–1175. doi: 10.1093/bioinformatics/btq112.

[3] Pierce, B. G., Wiehe, K., Hwang, H., Kim, B. H., Vreven, T.,& Weng, Z. (2014). ZDOCK server: interactive docking prediction of protein-protein complexes and symmetric multimers. Bioinformatics, 30(12), 1771–1773. doi: 10.1093/bioinformatics/btu097.

[4] Liu, Z., Li, Y., Han, L., Li, J., Liu, J., Zhao, Z., … Wang, R. (2014). PDB-wide collection of binding data: current status of the PDBbind database. Bioinformatics, 31(3), 405–412. doi: 10.1093/bioinformatics/btu626.

[5] Gordon, A. D., Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). Classification and Regression Trees. Biometrics, 40(3), 874. doi: 10.2307/2530946.