APPLICATION OF TOPOLOGICAL DATA ANALYSIS AND MACHINE LEARNING FOR MUTATION INDUCED PROTEIN PROPERTY CHANGE PREDICTION

By

Menglun Wang

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Applied Mathematics – Doctor of Philosophy

2021

ABSTRACT

APPLICATION OF TOPOLOGICAL DATA ANALYSIS AND MACHINE LEARNING FOR MUTATION INDUCED PROTEIN PROPERTY CHANGE PREDICTION

By

Menglun Wang

Mutagenesis is a process by which the genetic information of an organism is changed, resulting in a mutation. A lot of diseases are caused by mutation of protein, including Cystic fibrosis, Alzheimer's Disease, and most cancer. To get a better understanding of mutation induced protein properties change, accurate and efficient computational models are urgently needed.

Algebraic topology, a champion in recent worldwide competitions for protein-ligand binding affinity predictions, is a promising approach for simplifying the complexity of biological structures. In this thesis, we introduce element-specific and site-specific persistent homology, a new branch of algebraic topology, to simplify the structural complexity of protein-protein complexes and embed crucial biological information into topological invariants. Additionally, we propose a new deep learning algorithm called NetTree, to take advantage of convolutional neural networks and gradient boosting trees. A topology-based network tree (TopNetTree) is constructed by integrating the topological representation and NetTree for predicting protein-protein interaction $\Delta\Delta G$. Tests on major benchmark datasets indicate that the proposed TopNetTree significantly improves the current state-of-art in $\Delta\Delta G$ prediction.

For mutation induced protein folding energy change, we proposed a local topological predictor (LTP) based machine learning model. To characterize the molecular structure, Hessian matrix of the local surface is generated from the Exponential and Lorentz density kernel. Eigenvalues of Hessian matrix are calculated as the local topological predictor, which is then fed into the gradient boost machine learning model as features. Our LTP model obtained state-of-art results for various benchmark data sets of mutation induced protein folding energy change.

This thesis is dedicated to my parents, for their love.

ACKNOWLEDGEMENTS

First and foremost I am extremely grateful to thank my advisor, Dr. Guo-Wei Wei for his invaluable advice, continuous support, and patience during my PhD study. His passion in research, visionary insights and positive attitude in life has guided me to be a better researcher and a better man.

I would like to thank Dr. Moxun Tang, Dr. Ming Yan and Dr. Yiying Tong, for serving on my thesis committee and providing me with extensive guidance and helpful comments. I also want to thank Dr. Irfan and Dr. Saleh for offering me the opportunity of collaborating at John D. Dingell VA Medical Center. In addition, I want to thank Dr. Hao Zhu and Dr. Ruihao Huang for offering me the research fellowship at U.S. Food and Drug Administration.

I want to thank my fellow group members, Zixuan Cang, Yin Cao, Duc Nguyen, Kelin Xia, Kedi Wu, Rui Wang, Jiahui Chen, Yuchi Qiu for their kindness help in my research projects. I also want to thank fellow graduate students at Math Department, Anqi Chen, Wenzhao Chen, Weicong Zhou, just to name a few, for their companion in this long journey.

Last but not the least, I would like to thank my family. I am so grateful to my grandmother who raised me up as a little kid. I want to express thanks to my parents Zhenfu Wang and Zhifeng Xu. Their unconditional support and love give me the energy to pursue higher goals in life. Thank you.

TABLE OF CONTENTS

LIST OF TABLES ix

LIST OF FIGURES xi

LIST OF ALGORITHMS xiv

KEY TO ABBREVIATIONS xv

CHAPTER 1 BIOCHEMICAL BACKGROUND 1
1.1 An overview of biomolecular modeling 1
1.2 An overview of binding 2
1.3 An overview of protein folding 3
1.4 An overview of mutation 4
1.5 Motivation 6

CHAPTER 2 MACHINE LEARNING METHODS AND ALGORITHMS 7
2.1 An Overview of machine learning algorithm 7
2.2 Ensemble methods 8
2.2.1 Decision Tree 8
2.2.2 Random Forest 10
2.2.3 Gradient Boost Decision Tree 10
2.2.4 Feature importance 11
2.3 Neural Network 12
2.3.1 Layers and process of Neural Network 12
2.3.2 Convolutional Neural Network (CNN) 15
2.3.3 Other Neural Network 16

CHAPTER 3 TOPNETTREE: A TOPOLOGY-BASED NETWORK TREE FOR THE PREDICTION OF PROTEIN-PROTEIN BINDING AFFINITY CHANGES UPON MUTATION 18
3.1 Introduction 18
3.2 Dataset 21
3.2.1 AB-Bind dataset 21
3.2.2 SKEMPI and SKEMPI 2.0 dataset 22
3.2.3 Preprocessing of dataset 22
3.3 Topological Modelling 23
3.3.1 Persistent homology 23
3.3.1.1 Simplicial complex and filtration 23
3.3.1.2 Homology and persistence 24
3.3.2 Topological representation of protein-protein interactions 26
3.4 Auxiliary features 29
3.4.1 Atom-level features 29

3.4.2 Residue-level features . . . 30
3.5 Machine learning architecture . . . 31
3.5.1 TopGBT: Topology based gradient boosting tree model . . . 32
3.5.2 TopCNN: Topology based convolutional neural network model . . . 32
3.5.3 TopNetTree: Topology based network tree model . . . 32
3.5.4 Model parametrization and software used . . . 34

CHAPTER 4 LTP MODEL: APPLICATION OF LOCAL TOPOLOGICAL CHARACTERISTICS IN PROTEIN FOLDING ENERGY CHANGE UPON MUTATION . . . 37
4.1 Introduction . . . 37
4.2 Dataset . . . 38
4.2.1 ProTherm protein mutation database S2648 and S350 . . . 38
4.2.2 Preprocessing of dataset . . . 39
4.3 Local topological characterization of molecules . . . 39
4.3.1 Molecular surface representations and molecular density function . . . 39
4.3.2 Evaluation of curvature . . . 42
4.3.3 Critical points and Poincare Hopf index theorem . . . 44
4.3.4 Mutation site based element specific density field generation . . . 48
4.3.5 Explicit expression of Hessian matrix . . . 49
4.4 Prediction model design . . . 52
4.4.1 Eigenvalue learning feature generation . . . 53
4.4.2 Auxiliary features . . . 53
4.4.3 Machine learning model . . . 54

CHAPTER 5 RESULT . . . 56
5.1 Evaluation criteria . . . 56
5.2 Model performance of TopNetTree on PPI binding free energy change . . . 56
5.2.1 Prediction result on AB-Bind dataset . . . 56
5.2.1.1 Overall result . . . 56
5.2.1.2 Protein level non overlapping test on the AB-bind 645 dataset . . . 59
5.2.1.3 Protein level leave-one-out validation test . . . 60
5.2.2 Prediction result on SKEMPI dataset . . . 61
5.2.3 Prediction result on SKEMPI 2.0 dataset . . . 62
5.3 Performance of LTP model on ProTherm protein mutation database S2648 and S350 64
5.3.1 General performance . . . 64
5.3.2 Inter/Intra-protein-level crossvalidation . . . 65
5.3.3 Prediction result for different density kernel . . . 67

CHAPTER 6 DISCUSSION . . . 68
6.1 Prediction result analysis for different mutation type . . . 68
6.1.1 Analysis of TopNetTree prediction result on S645 . . . 68
6.1.2 Analysis of LTP prediction result on S2648 . . . 70
6.2 Prediction result analysis for different mutation regions . . . 71
6.2.1 Definition of mutation region . . . 71

6.2.2 Analysis of TopNetTree prediction result on S645 71
6.2.3 Analysis of LTP prediction result on S2648 73
6.3 Alanine scanning test of 1AK4 74

CHAPTER 7 THESIS CONTRIBUTION 76
7.1 Protein-Protein interactions energy change upon mutation 76
7.2 Protein folding energy change upon mutation 76

APPENDICES 78
APPENDIX A SUPPLEMENTARY MATERIALS FOR TOPNETTREE MODEL . 79
APPENDIX B SUPPLEMENTARY MATERIALS FOR LTP MODEL 105

BIBLIOGRAPHY 108

LIST OF TABLES

Table 3.1: Summary of topological descriptors. The barcodes are generated upon mutant and wild type complexes. 27

Table 4.1: Properties of critical points 48

Table 5.1: Comparison of the Pearson correlation coefficients of various methods for the AB-bind S645 set. Except for those from present TopNetTree and TopGBT, the other results are adopted from Ref. [68]. 57

Table 5.2: Result of non-overlapping protein level test on AB-bind dataset, including Pearson correlation coefficient and RMSE in kcal/mol. 59

Table 5.3: Result of protein-level leave-one-out-validation test on AB-bind dataset, including Pearson correlation coefficient and RMSE in kcal/mol. 60

Table 5.4: Comparison of the Pearson correlation coefficients of various methods for the single point mutation in SKEMPI dataset of 1131 mutations. Except for those from TopNetTree and SAAMBE, the other results are adopted from Ref. [91]. . . 61

Table 5.5: Prediction results of S350 and 5-fold cross validation results of S2648. All the result of other methods listed in the table are from the paper cited. 65

Table 5.6: Intra-protein-level cross-validation result of S2648. Mutations in each protein are 5-fold cross-validated inside protein. 66

Table 6.1: Criteria of residue regions [55] 71

Table 6.2: Alanine mutation test on 1AK4 chain A. 75

Table A.1: TopNetTree crossvalidation result on S645 80

Table B.1: Comparison of Pearson correlation (R_p) and RMSEs (kcal/mol) of various methods on prediction of mutation induced protein stability changes of the S350 set and 5-fold cross validation of mutation induced protein stability changes of the S2648. *n* represents number of samples successfully processed. LTP1 is our topological based mutation predictor that solely utilizes structural information. LTP2 is our model that complements LTP1 with auxiliary features. The results reported in the publications are listed in the table [70].50 repeated runs are conducted and median values of metrics are picked for LTP2 and LTP1 methods. 105

Table B.2: 5-fold cross validation results of Q3421 with respect to R_p and RMSE. 50 repeated runs were conducted and median values of metrics were picked for LTP2 and LTP1 methods. Comparison of Pearson correlation (R_p) and RMSEs (kcal/mol) of various methods on 5-fold cross validation of mutation induced protein stability changes of the Q3421. *n* represents number of samples successfully processed. In LTP1 method, two-scales models are considered by coupling two sets of Hessian eigenvalue features. Moreover, R_p/RMSE (kcal/-mol) of 0.79/1.2 was reported for STRUM method [70]. 50 repeated runs are conducted and median values of metrics are picked for LTP2 and LTP1 methods. 106

LIST OF FIGURES

Figure 1.1: Example of single site mutation from DNA genome to amino acids 5

Figure 2.1: An example of fully connected neural network 13

Figure 2.2: An example of convolution operation on a 4×4 matrix by 2×2 kernel 16

Figure 3.1: Counts of mutation type in AB-bind dataset 21

Figure 3.2: Illustration of filtration and persistence diagram of a set of points on a plane. . . 25

Figure 3.3: Topological barcode change associated with a mutation. Residue Leucine in the wild type is mutated into Alannine. Barcodes are generated for carbon atoms within a cutoff of 12Å of the mutant residue. 26

Figure 3.4: Illustration of the point cloud generation of antibody-antigen complex 1DQJ . . 28

Figure 3.5: Illustration of the proposed TopNetTree model. 33

Figure 3.6: Illustration of CNN parameters. 35

Figure 4.1: Illustration of van der Waals (vdWS) surface (yellow region), Solvent accessible surface (SAS) (red dotted margin) and solvent excluded surface (SES) (blue dotted margin 40

Figure 4.2: An example of 2-D exponential density function generated surface. 41

Figure 4.3: Example of different critical points in a standard cube. Nucleic critical point (NCP) in red, Bond critical point (BCP) in yellow, Ring critical point (RCP) in cyan and Cage critical point (CCP) in green. 45

Figure 4.4: Density field of naphthalne 46

Figure 4.5: Gradient field of naphthalne 47

Figure 4.6: An example of two eigenvalue maps of naphthalne using two different density fields 51

Figure 4.7: an example of three eigenvalue isosurface maps of naphthalne using the exponential density field 52

Figure 4.8: Flowchart of eigenvalue learning model of protein folding energy change upon mutation with example of 1A3J A 12 F A. First, mutation site is selected for both wild and mutant structure. Corresponding exponential/Lorentz density fields are generated. Then Hessian matrix and its eigenvalues are evaluated at the near mutation region with respect to different element groups. Finally, eigenvalue features and auxiliary features are fed into the GBDT model to get the prediction value of energy change. 55

Figure 5.1: (a)Scatter plot of TopNetTree prediction crossvalidation result on S645 (b)Scatter plot of TopNetTree prediction crossvalidation result on S645 exclude 27 non-binders 58

Figure 5.2: Scatter plot of TopNetTree blind test prediction on homology models 58

Figure 5.3: Performance evaluation on the 10-fold cross-validation on set S1131. 62

Figure 5.4: (a)Scatter plot of TopNetTree prediction crossvalidation result on S4947 (b)Scatter plot of TopNetTree prediction crossvalidation result on S4169 63

Figure 5.5: Scatter plot of TopNetTree prediction crossvalidation result on S8338 63

Figure 5.6: (a)Scatter plot of prediction result on S350 (b) Scatter plot of 5-fold crossvalidation result on S2648 64

Figure 5.7: Parameter selection heatmap in prediction model of S350. 67

Figure 6.1: Comparison of average experimental and prediction binding affinity changes upon mutation associated with different amino acid types for the AB-Bind dataset. The x-axis labels the residue type of the original, while the y-axis labels the residue type of the mutant. For a reverse mutation, its $\Delta\Delta G$ is taken the same magnitude as the original value with an opposite sign. **a** Average binding affinity changes upon mutation (kcal/mol) **b** Variance of binding affinity changes upon mutation (kcal/mol) 69

Figure 6.2: Comparison of average experimental (a) and prediction binding (b) affinity changes upon mutation associated with different amino acid types for S2648 dataset. 70

Figure 6.3: Prediction results for different residue region types in S645 dataset 72

Figure 6.4: Scatter plot of 5-fold crossvalidation result for S2648 set in (a) Buried mutation region, with $Rp = 0.79$ and $RMSE = 0.94$,(b) Exposed mutation region, with $Rp = 0.78$ and $RMSE = 0.92$ (c) Intermediate mutation region, with $Rp = 0.78$ and $RMSE = 0.92$ 73

Figure 6.5: Structure of protein complex 1AK4, chain A in blue and chain D in red 74

Figure B.1: Predictive behaviors of LTP1 model on protein folding energy change upon mutation. 107

LIST OF ALGORITHMS

Algorithm 1: Hunt's algorithm 9

Algorithm 2: Gradient Boosting 11

KEY TO ABBREVIATIONS

ML Machine learning

DL Deep learning

PH Persistent homology

ESPH Element-specific persistent homology

RF Random Forest

GBDT Gradient boosting decision tree

NN Neural network

CNN Convolutional neural network

GD Gradient descent

SGD Stochastic gradient descent

PPI Protein-protein interaction

RMSE Root mean square error

MIBPB Matched interface and boundary Poisson Boltzmann

SAS Solvent accessible surface

SES Solvent excluded surface

ESES Eulerian solvent excluded surface

CP Critical point

NCP Nucleic critical point

BCP Bond critical point

CCP Cage critical point

RCP Ring critical point

PDB Protein data bank

SKEMPI Structural database of Kinetics and Energetics of Mutant Protein Interactions

AB Antibody

AG Antigen

WT Wild type

MT Mutant

SNPs Single nucleotide polymorphisms

CHAPTER 1

BIOCHEMICAL BACKGROUND

1.1 An overview of biomolecular modeling

Biomoleculars, such as DNA, RNA and proteins are the fundamental and essential to human body. Understanding structure-function relationships is a major challenge in the molecular level. Research on the relationship between structures of biomolecules and their functions is one of the hot topic in drug design and pharmaceutical industry. Among all the properties of a biomolecular system, thermodynamic properties are crucial to the functionality of the system. Those thermodynamic properties include the binding affinity of protein-ligand complexes, the stability changes induced by amino acid mutations in protein, and the flexibility of protein residues. To get a better understanding of the properties mentioned above, effective prediction model is urgently needed.

There are three major types of biomolecular prediction model. Physics-based methods build models according to physical laws and are indispensable for molecular modeling which provide predictions and reveal underlying mechanisms. Examples of such kind include quantum mechanics calculation, molecular dynamics simulation, and Monte Carlo sampling. There are also more efficient approximations to the atomic systems by using a continuum for part of the systems. For example, the Poisson-Boltzmann model delivers efficient description of the electrostatics in the solvation processes of molecules by using a continuum solvent.

Empirical model is the type of model using weights determined by the experimental data. A widely used setup for empirical models is to combine molecular mechanics energies with polar part of solvation process modeled by Poisson-Boltzmann model or Generalized Born model and nonpolar part of solvation process reflected by surface areas which are usually called MM/PBSA or MM/GBSA models[29, 30].

Machine learning based model is the type of model using machine learning algorithm to learn from existing training data. With machine learning, the prediction model can handle more

detailed descriptions of the systems and descriptors of various types. Also, machine learning based algorithm is more adaptive to large dataset.

In this work, we are interested in designing machine learning based prediction model on biomolecular properties, including protein-protein binding energy change upon mutation and protein folding energy change upon mutation. Detailed features and model architecture will be discussed in the later chapters of the thesis.

1.2 An overview of binding

In physics and chemistry, binding energy is the smallest amount of energy required to remove a particle from a system of particles or to disassemble a system of particles into individual parts. In this section, we will introduce the basic conceptions and evaluation s of binding energy.

$\mathbf{K}_D$: In chemistry, biochemistry, and pharmacology, a dissociation constant (K_D) is a specific type of equilibrium constant that measures the propensity of a larger object to separate (dissociate) reversibly into smaller components, as when a complex falls apart into its component molecules, or when a salt splits up into its component ions.

For general reaction:

$$A_xB_y \rightleftharpoons xA + yB \tag{1.1}$$

in which a complex A_xB_y breaks down in to xA subunits and yB subunits, the dissociation constant is defined as

$$K_D = \frac{[A]^x[B]^y}{[A_xB_y]} \tag{1.2}$$

In the specific case of antibodies (Ab) binding to antigen (Ag), usually the term affinity constant refers to the association constant.

$$K_A = \frac{[AbAg]}{[Ab][Ag]} = \frac{1}{K_D} \tag{1.3}$$

$\mathbf{K}_i$: The inhibitory constant (K_i) also represents a dissociation constant, but more specific for the binding of an inhibitor (I) to an enzyme (E).

$$E + I \rightleftharpoons EI \tag{1.4}$$

IC_{50}: The half-maximal inhibitory concentration (IC_{50}) is a measure of the potency of a substance in inhibiting a specific biological or biochemical function. More specifically, IC_{50} is the concentration of inhibitor required to reduce the biological activity of interest to half of the uninhibited value.

Although IC_{50} is not a direct indicator of affinity, for enzymatic reactions, binding affinity of the inhibitor (K_i) could be solved by the Cheng-Prusoff equation.

$$K_i = \frac{IC_{50}}{1 + \frac{[S]}{K_m}} \tag{1.5}$$

Here in the equation, [S] is fixed substrate concentration and K_m is the Michaelis constant.

EC_{50}: The half-maximal effective concentration (EC_{50}) refers to the concentration of a drug, antibody or toxicant which induces a response halfway between the baseline and maximum after a specified exposure time.

For inhibition constants at celluar receptors, one can get binding affinity from the following equation:

$$K_i = \frac{IC_{50}}{1 + \frac{[A]}{EC_{50}}} \tag{1.6}$$

where [A] is the fixed concentration of agonist.

ΔG: For the binding of receptor and ligand molecules in solution, the molar Gibbs free energy ΔG, or the binding affinity is related to the dissociation constant K_D via

$$\Delta G = RT \ln \frac{K_D}{c^\theta} \tag{1.7}$$

in which R is the ideal gas constant, T is temperature and the standard reference concentration c^θ = 1 mol/L.

1.3 An overview of protein folding

Protein folding is the process by which a protein chain is translated to a folded conformation. Folding process is mainly guided by hydrophobic interactions, formation of intramolecular hydrogen bonds, van der Waals forces, and it is opposed by conformational entropy[2].

Process of protein folding can be divided into the following steps

- **Primary structure** The primary structure of a protein is the linear amino-acid sequence. This sequence is the essential start to folding process, which specifies both the native structure and the pathway to attain the final state.

- **Secondary structure** The secondary structure is the first step in the folding process that a protein takes to assume its native structure. There are two major types of secondary structure, namely alpha helices and beta sheets, both of them are stabilized by intramolecular hydrogen bonds.

- **Tertiary structure** Secondary structure hierarchically gives way to tertiary structure formation of a protein. The major force to generate tertiary structure is hydrophobic interaction. Once the protein's tertiary structure is formed and stabilized by the hydrophobic interactions, there may also be covalent bonding in the form of disulfide bridges formed between two cysteine residues.

Protein misfolding can lead to aggregate protein. Aggregated proteins are associated with prion-related illnesses such as Creutzfeldt–Jakob disease, bovine spongiform encephalopathy (mad cow disease), amyloid-related illnesses such as Alzheimer's disease and familial amyloid cardiomyopathy or polyneuropathy[42] as well as intracellular aggregation diseases such as Huntington's and Parkinson's disease.[80, 17] Mutation of protein sequence is one of the reason to cause protein misfolding, and protein folding energy change is an essential thermodynamic problem. In chapter 5 of this thesis, we will set up a model to predict the protein folding energy change upon mutation.

1.4 An overview of mutation

In biology, a mutation is an alteration in the nucleotide sequence of the genome of an organism, virus, or extrachromosomal DNA. Mutations play a part in both normal and abnormal biological processes including: evolution, cancer, and the development of the immune system, including junctional diversity. Mutation is the ultimate source of all genetic variation, providing the raw material on which evolutionary forces such as natural selection can act.

There are four major classes of mutation, including:

- spontaneous mutation (molecular decay)
- mutations due to error-prone replication bypass of naturally occurring DNA damage
- errors introduced during DNA repair
- induced mutations caused by mutagens

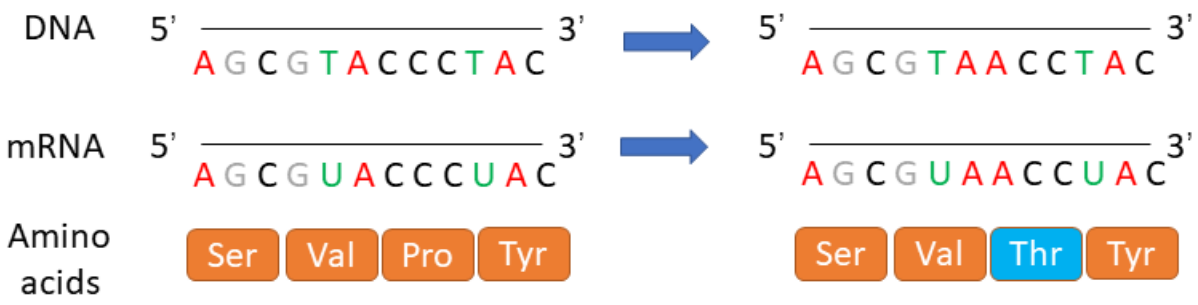


Figure 1.1: Example of single site mutation from DNA genome to amino acids

Mutation of gene could impact human health and functioning in several ways. In this work, we mainly focus on mutation impact on protein sequence. More specifically, we focus on the mutations occur in coding regions of the genome, which are more likely to alter the protein product. To better understand the mechanism and impact on mutation, site-directed mutagenesis is being applied to the research. Site-directed mutagenesis is an invaluable tool to modify genes and study the structural and functional properties of a protein, based on the structure, function, catalytic mechanism, and catalytic residues of enzymes. Site-directed mutagenesis includes two classes: single site mutation and combinational mutations.

Since mutation could alter the protein sequence and structure, in the binding process, such as antibody neutralization and protein-ligand binding, binding affinity of wild type could vary from mutant type in a large scale. According to recent research[16], binding free energy change of SARS-CoV-2 mutations is closely related to the infectivity of virus. So finding computational

estimation of mutation-induced binding free energy changes is crucial to uncover the mysterious of a lot of biological challenge. In this thesis, we are going to combine topological tools and machine learning techniques to build models for mutation induced binding energy prediction.

1.5 Motivation

Biological data is growing at a fascinating speed nowadays, for example, Protein Data Bank (PDB) has accumulated near 130,000 tertiary structures. Thus, machine learning model becomes more and more powerful in structure based biomolecular properties prediction. The evolution of machine learning itself also boost the ability for biomolecular properties prediction. Algorithms such as gradient boosting trees and convolutional neural network allow researchers to build more accurate model on specific types of tasks.

On the other hand, topology, a branch of mathematics, is proved useful in characterizing molecular structure. With the powerful topological tools such as persistent homology, one can reduce the dimensionality of biomolecular structural data and get more learnable input features.

In this thesis, we incorporated topological based features with advanced machine learning algorithms on protein properties prediction, including protein-protein binding energy change upon mutation and protein folding energy change upon mutation. Our models proved accurate and efficient on the given dataset, including AB-Bind, SKEMPI and ProTherm.

CHAPTER 2

MACHINE LEARNING METHODS AND ALGORITHMS

In this chapter, we will give an overview of the major machine learning algorithms and architectures used in the paper. For each algorithms, we will introduce the process, application and pros/cons.

2.1 An Overview of machine learning algorithm

Machine learning algorithms build a model based on sample data, known as "training data", in order to make predictions or decisions without being explicitly programmed to do so. Depend on the label information of training data, machine learning approaches can be divided into the following categories:

- Supervised learning

  In supervised learning, the target value (label) of training set is explicitly given. In mathematical model, supervised learning algorithm maps a vector of features to a target value. Depend on the target predicting value, supervised learning can be divide into two types: classification and regression. For classification, the target value must be categorical and for regression, the target value can be any continuous numerical value. A lot of machine learning algorithm, for example support vector machine (SVM), random forest, gradient boosting tree, are supervised learning algorithm.

- Unsupervised learning

  On the contrary side of supervised learning, unsupervised learning algorithms does not have label information in the training set. Different from supervised learning, the main purpose of unsupervised learning is to find structure character and commonalities of data. . A central application of unsupervised learning is in the field of density estimation in statistics, such as finding the probability density function.

- Semi-supervised learning

Semi-supervised learning is the problems between supervised learning and unsupervised learning. Typically, the training labels are noisy, limited or imprecise. The main purpose of semi-supervised learning algorithm is to improve the learning accuracy from the limited training labels.

- Reinforcement learning

  Reinforcement learning is an area of machine learning concerned with how software agents ought to take actions in an environment so as to maximize some notion of cumulative reward. Different from supervised or unsupervised learning, reinforcement learning is mainly used to find strategy instead of finding certain result. A lot of other disciplines of studies are closely related to reinforcement learning, including game theory, control theory, operations research, information theory, simulation-based optimization, multi-agent systems, swarm intelligence, statistics and genetic algorithms. There are several famous and successful applications of reinforcement learning, such as alpha-go (AI based go program) and autonomous driving software.

2.2 Ensemble methods

Ensemble methods is the general categories of machine learning algorithm based on decision tree algorithms. In the following section we will briefly introduce the algorithms including decision tree, random forest and gradient boost decision tree.

2.2.1 Decision Tree

Decision Trees are a non-parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. There are three basic nodes of a decision tree, including decision nodes, chance nodes and end nodes.

Hunt's algorithm Hunt's algorithm is considered one of the most used algorithm for building decision tree. The algorithm can recursively partition the training dataset into successively purer subsets. The following algorithm shows the procedure of Hunt's algorithm

Algorithm 1 Hunt's algorithm

while not all the records in the subset belong to the same class **do**
  if D_t contains records that belong the same class y_t **then**
    t is labeled as y_t
  else if D_t not empty but Att_{list} is empty **then**
    t is labeled as majority records in the dataset
  else if D_t belong to more than one class and Att_{list} is not empty **then**
    Use attribute selection methods to choose next best attribute from the Att_{list} and remove that list from Att_{list} use the attribute and its condition as next test condition
  end if
end while

Here in the algorithm, D is the training dataset with a number of attributes, Att_{list} is the subset and its testing criterion and attribute selection method is the procedure to determine the best splitting. For different target values, there are different measures to determine the best way to split the records. The principle of the splitting algorithm is to get more purity. Following measurements are the most commonly used to define a split's purity:

Entropy:

$$E(x) = \sum_{i=1}^{n} p_i log_2(p_i) \tag{2.1}$$

Gini index:

$$GINI(x) = 1 - \sum_{i=1}^{n} p_i^2 \tag{2.2}$$

Classification Error:

$$Classificationerror(x) = 1 - max_i p_i \tag{2.3}$$

With the measurements mentioned above, one can get the best split based on the impurity.

2.2.2 Random Forest

Just like its name, random forest is an ensemble method by constructing a multitude of decision trees. Random decision forests correct for decision trees' habit of overfitting to their training set.[33]

Bootstrap aggregating, or bagging is the general technique to generate a random forest model. Suppose we have a training set X, using bootstrapping by selecting a random sample with replacement, we generate m decision tree $f_1, ..., f_m$. Then random forest prediction for unseen sample x' is defined by averaging the predictions from all the individual regression trees on

$$f = \frac{1}{m} \sum_{n=1}^{m} f_n(x') \tag{2.4}$$

For classification problem, the predicted class f is the class that the majority of trees vote for. To further avoid overfitting, random forest also include another type of bagging scheme: feature bagging. During the training process, for each decision tree, only a random subset of the features are used.

2.2.3 Gradient Boost Decision Tree

Gradient boosting decision tree is another popular ensemble method for regression and classification tasks. Gradient boosting algorithm was first observed by Leo Breiman[9], and was subsequently developed by Friedman[32]. The algorithm iteratively generate weak learners by minimizing loss function using gradient decent method. For a n samples training set with features and labels $\{(x_i, y_i)\}_{i=1}^{n}$, a differentiable loss function is defined as $L(y, F(x))$. With presetted stop iteration M, the gradient boosted tree algorithm is shown as below:

In the algorithm, there are several important hyperparameters which will heavily impact the model performance. The most important hyperparameters are number of estimators and learning rate. Learning rate controls the speed of optimization of loss function in gradient decent and number of estimators controls the number of boosting stages to perform.

Algorithm 2 Gradient Boosting

Initialize $F_0(x)$ with constant value

$$F_0(x) = \underset{\gamma}{\operatorname{argmin}} \sum_{i=1}^{n} L(y_i, \gamma)$$

while $m < M$ **do**

$$r_{im} = -[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}]_{F(x)=F_{m-1}(x)}, i = 1, ..., n$$

,

$$h_m(x) = fittree\{(x_i, r_{im})\}_{i=1}^{n}$$

$$\gamma_m = \underset{\gamma}{\operatorname{argmin}} \sum_{i=1}^{n} L(y_i, F_{m-1}(x_i) + \gamma h_m(x_i))$$

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x_i)$$

end while
Output $F_M(x)$

2.2.4 Feature importance

Generally, feature importance provides a score that indicates how useful or valuable each feature was in the construction of the boosted decision trees within the model. Feature importance is calculated as the decrease in node impurity weighted by the probability of reaching that node. The node probability can be calculated by the number of samples that reach the node, divided by the total number of samples. The higher the value the more important the feature.

We use Scikit-learn's built in function to get the feature importance, the implementation in Scikit-learn is shown below:

For each decision tree, Gini importance is calculated as

$$ni_j = w_j C_j - w_{left(j)} C_{left(j)} - w_{right(j)} C_{right(j)} \tag{2.5}$$

Here i represent feature and j represent node. n is feature importance and C is impurity value of

node. The importance for feature i on a single decision tree is calculated as

$$fi_i = \frac{\sum_{\text{j: node j splits on feature i}} ni_j}{\sum_{k \in \text{all nodes}} ni_k} \tag{2.6}$$

$$\hat{fi}_i = \frac{fi_i}{\sum_{j \in \text{all features}} fi_j} \tag{2.7}$$

The overall feature importance is the average over all decision trees, which is calculated as :

$$FI_i = \frac{\sum_{j \in \text{all trees}} \hat{fi}_{ij}}{T} \tag{2.8}$$

2.3 Neural Network

Neural network (NNs) is one of the most popular machine learning algorithm which is inspired by the neural networks that consist human brains. The history of neural network can trace back to 1943, Warren McCulloch and Walter Pitts [49] opened the subject by creating a computational model for neural networks. In 1958, psychologist Frank Rosenblatt invented the perceptron[74, 73, 87, 43], the first artificial neural network. The first functional networks with many layers were published by Ivakhnenko and Lapa [78, 45, 46]in 1965, as the Group Method of Data Handling.

As the implication of its name, neural network is based on a collection of connected units or nodes called artificial neurons. The connections between the neurons are called edges and each edge are assigned with specific weights. According to the connection relations, neurons can be grouped into layers. Neurons of one layer connect only to neurons of the immediately preceding and immediately following layers. In the following section we will briefly introduce the basic process and layers used in the work.

2.3.1 Layers and process of Neural Network

In this section, we will introduce the component of a basic neural network, including different types of layers and process.

Input and output layers

Input and output layers are the basic component of neural network. Input layer is determined by feature size of the training sample, can be either one dimension or high dimension with multiple

channels. Output layer is determined by the target value type, either a value for regression problem or a class for classification problem.

Activation layers

Activation layers are the layers which apply activation function to the given input layers. There are several activation functions, which is listed below:

- Rectified Linear Activation (ReLU): $f(x) = max(0, x)$
- Logistic (Sigmoid): $f(x) = \frac{1}{1+e^{-x}}$
- Hyperbolic Tangent (tanh): $f(x) = tanh(x)$

In general, the activation functions are non linear. With those activation layers, the network will gain more nonlinearities and a better overall performance.

Dense layers

Dense layers, also called as fully connected layers, are the layers connect all neurons of its previous layer to the current layer. Mathematically, dense layer performs a matrix multiplication and can be used to change the dimension of the vector or matrix. In general, dense layer has a large weight matrix and is relatively computational expensive.

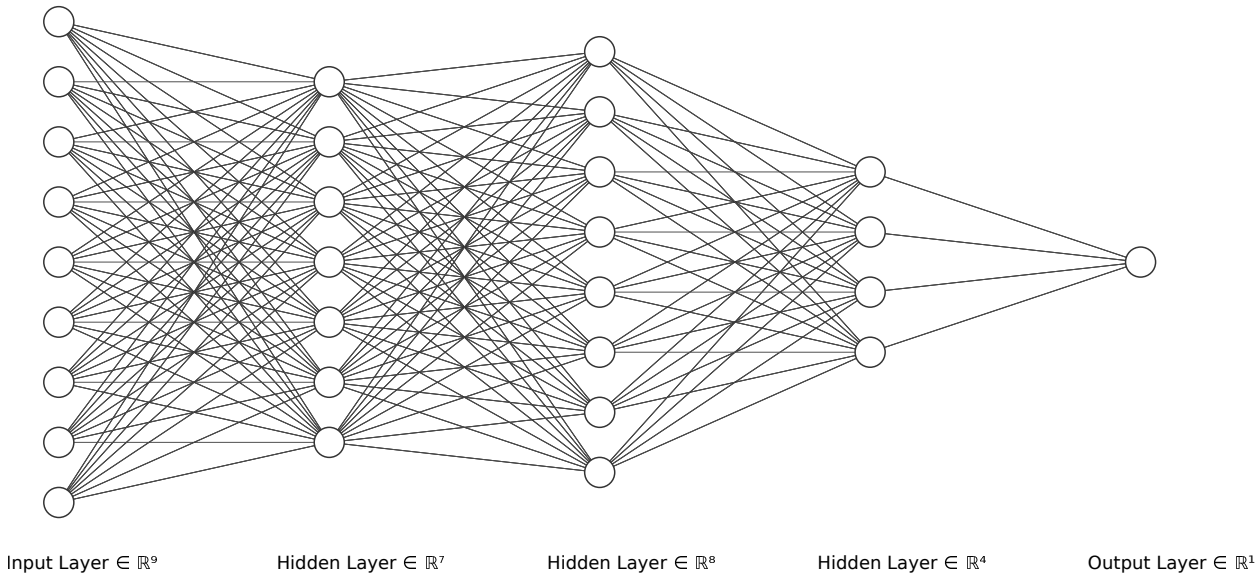


Figure 2.1: An example of fully connected neural network

Pooling layers

Pooling layers are the layers used to reduce the dimensions of the feature maps. There are two typical pooling methods: max pooling and average pooling. The pooling layers apply sliding window to the previous layer and returns either max value or average value of the sliding window. Pooling layers are usually used in image processing neural networks.

Dropout layers

Dropout layers are the layers which randomly set input neurons to 0. The rate of setting as 0 is called dropout rate. Inputs not set to 0 are scaled up by $\frac{1}{1-d_{rate}}$, which make the sum over all inputs unchanged. This dropout process can reduce the overfitting of the neural network.

Optimization and backpropagation

Backpropagation is the process which calculates the gradient of the loss function with respect to the neural network's weights backwards through the network. Backpropagation was invented in the 1970s[78] as a general optimization method for performing automatic differentiation of complex nested functions[26, 60]. However, it wasn't until 1986, with the publishing of a paper by Rumelhart, Hinton, and Williams[76].

For optimization algorithm of neural network, most of algorithms are variant of gradient decent. We listed several commonly used algorithm below:

- Gradient decent (GD)

$$\theta = \theta - a\nabla J(\theta) \tag{2.9}$$

- Stochastic Gradient Descent (SGD)

$$\theta = \theta - a\nabla J(\theta; x(i); y(i), \text{where}\{x(i), y(i)\}\text{are training sample} \tag{2.10}$$

  Stochastic Gradient Decent is a variant of Gradient Descent, instead of update the parameters based on whole training set, SGD update model parameter based on randomly chosen training sample. In general, SGD has a faster converge speed than GD.

- Adaptive Moment Estimation (Adam) To reduce the high variance in SGD and accelerates the convergence, momentum terms are introduced in Adam:

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \tag{2.11}$$

$$\hat{v_t} = \frac{v_t}{1 - \beta_2^t} \tag{2.12}$$

The model is updated as follows:

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v_t}} + \epsilon}\hat{m}_t \tag{2.13}$$

In the following part of this thesis, we use Adam algorithm as the optimizer of all the neural network.

Hyperparameters

Hyperparameters are the parameters of neural network other than the weights determined by the training set. Common hyperparameters include number of layers, type of layers, learning rate of optimizer, dimension of channel etc. How to tune hyperparameters to get a better performance is always a headache for researchers. For some hyperparameters, one can do Bayesian optimization tuning. For hyperparameters like number of layers, the only way is grid search.

2.3.2 Convolutional Neural Network (CNN)

Convolutional neural network (CNN) is a class of artificial neural network. The neocognitron was introduced by Kunihiko Fukushima in 1980s.[36]. In this paper, convolutional layers and downsampling layers were first introduced. The first modern application of convolutional neural networks was implemented in the 90s by Yann LeCun etc.[53]

The core part of CNN is convolution operation and its related layer. Convolution operation applies tensor production by sliding window over input data. This operation can efficiently capture local patterns of data and avoid the curse of dimensionality of fully connected layer. An example of 2D convolution is shown in figure2.2

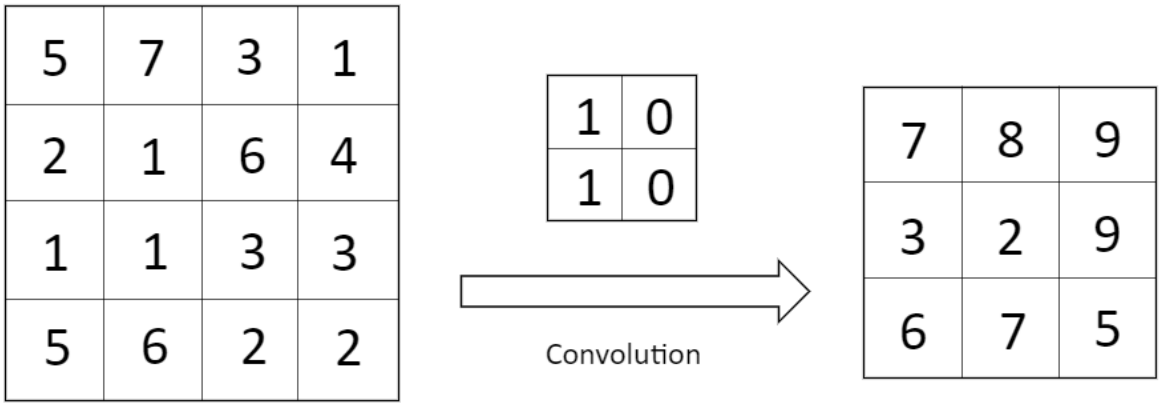


Figure 2.2: An example of convolution operation on a 4×4 matrix by 2×2 kernel

There are several conceptions and parameters related to convolution operation, we listed below

- **Kernel size**: the size of sliding window, for example 2×2
- **Padding**: Padding is the operation related to the margin of input data. If no padding to input data, the output dimension after convolution will decrease. To keep same dimensionality, adding 0 valued on the borders of input data is needed, which is called zero padding. Padding is typically set to the kernel dimension -1.
- **Stride**: Stride is the step size of sliding window moves on each iteration.

Beside convolution layer, a complete convolutional neural network also contains the layers we mentioned in previous section, including dense layers, pooling layers etc. We will show the CNN structure of TopNetTree model in the later section of the work.

2.3.3 Other Neural Network

Other than CNN, there are a lot of artificial neural network structures, each structure are adapted to certain types of tasks. We listed a few below:

- **Recurrent neural network**: Recurrent neural network (RNN) is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence.

The RNN networks were based on David Rumelhart's work in 1986[75]. Long short-term memory (LSTM) network is one of the most commonly used RNN architecture. In general, Recurrent neural networks have outstanding performance in sequence and time related data

- **Graph neural network**: Graph neural network (GNN) is a type of neural network for processing graph data structures. It has been mathematically proven that GNNs are a weak form of the Weisfeiler–Lehman graph isomorphism test[77], so any GNN model is at least as powerful as this test.

- **Residual neural network**: Residual neural network is a type of neural network which envolves shipping process or shortcut between layers. It was first proposed by Kaiming He etc in 2015[44].

CHAPTER 3

TOPNETTREE: A TOPOLOGY-BASED NETWORK TREE FOR THE PREDICTION OF PROTEIN-PROTEIN BINDING AFFINITY CHANGES UPON MUTATION

3.1 Introduction

Protein-protein interactions (PPIs) are crucial to a wide range of biological activities and functions in the human body, including cell metabolism, signal transduction, muscle contraction, and immune systems. Antibody-antigen is one of the most essential systems among all PPIs and plays a unique role in studying PPIs. Antibodies (Abs) are large proteins serving important roles in the immune system by counteracting antigens which are chemicals recognized as alien by the human body. On the tip of an antibody, there is an antigen-binding fragment (Fab) that contains a paratope for recognizing a unique antigen via its epitope. More specifically, a paratope consists of a set of complementarity-determining regions (CDRs) which have the highest conformational flexibility among sites on an antibody [19]. The high selectivity of antibody-antigen recognition mechanism and the flexibility of antibodies as large proteins make antibodies a suitable platform for designing counteractants of target molecules. Antibodies have been widely used as therapeutic agents to treat human diseases. Antibody therapy has several advantages over traditional therapy including longer serum half-life, higher avidity and selectivity, and the ability to invoke desired immune responses [15, 23, 81]. Also, antibody therapy brings hope to several previously incurable diseases and there are ongoing efforts in the direction of HIV vaccine development [3] and cancer therapeutic antibodies [41, 6].

Three-dimensional (3D) structural information and thermodynamic measurements are two essential components for understanding the molecular mechanism of PPIs. Many experimental methods have been developed to determine the structure of protein-protein complexes. Among them, X-ray crystallography, nuclear magnetic resonance (NMR) and cryo-electron microscopy (cryo-EM) are the main workhorses [38]. Protein Data Bank (PDB) [7], one of the largest pro-

tein structure databases, includes tons of thousands of protein-protein complex structures and is expanding at an unprecedented rate.

Site-directed mutation is a key technology for probing PPI thermodynamic properties, including binding affinities of antibody-antigen interactions. Sirin *et al.* [83] collected an AB-Bind database of mutation-induced antibody-antigen complex binding free energy changes. This database contains 1101 mutation data entries, including 645 single-point mutations on 32 different antibody-antigen complexes. SKEMPI is a more general database for protein-protein binding affinity changes upon mutation ($\Delta\Delta G$)[61]. It contains 3047 mutation data entries for protein-protein heterodimeric complexes with experimentally determined structures.

The aforementioned databases have been widely used as benchmark tests for evaluating the predictive power of computational methods, which are indispensable for the investigation of PPIs, especially for the systematic screening of mutations [64] [24]. There are many reliable computational methods that can predict mutant structures upon the wild type, including Rosetta [50] and Jackal [90]. Computational methods for generating protein structures from sequences (e.g., MODELLER [85]) and predicting docking poses for protein-protein complexes (e.g., BioLuminate [98]) are also available.

For the thermodynamic properties of PPIs, the information is usually interpreted as the binding affinity or binding free energy, ΔG. Given their importance, a variety of computational methods has been developed for the prediction of antibody-antigen binding affinities based on structures. DFIRE [94] relies on an all-atom, distance scaled, pairwise potential derived using a database of high-quality diverse protein structures. STATIUM uses a pairwise statistical potential that scores how well a protein complex can accommodate different pairs of residues in the parent complex geometry. Also, force fields for proteins can be used to compute the binding free energy, representing van der Waals interactions, hydrophobic packing, electrostatics, and solvation effects. These approaches include FoldX (FOLDEF) [79], Discovery Studio (CHARMMPLR) [8], and Rosetta [50]. Typically, physics-based methods provide mechanistic interpretations but are not designed for handling large and diverse datasets.

Pires *et al.* optimized their graph-based CSM method for predicting antibody-antigen affinity changes upon mutation given in the AB-Bind database [68]. This method, called mCSM-AB, was shown to outperform the aforementioned physical methods but only achieving a Pearson's correlation coefficient (R_p) of 0.53 with 10-fold cross-validation on a set of 645 single-point mutations. Therefore, the limited performance of current methods highlights a pressing need for a new generation of $\Delta\Delta$G predictors that are constructed with entirely new design principles and/or innovative machine learning algorithms. While the physics-based methods assume potential functions of certain forms and the graph-based method only considers pairwise interactions, we seek an approach that makes fewer assumptions and allows a systemic description of the protein-protein interaction.

Persistent homology [35, 27, 99, 100], a new branch of algebraic topology, is able to bridge geometry and topology, leading to a new efficient approach for the simplification of biological structural complexity [88, 37, 89, 10, 93, 51]. However, it neglects critical chemical/biological information when it is directly applied to complex biomolecular structures. Element-specific persistent homology can retain critical biological information during the topological abstraction. Paired with advanced machine learning, such as a convolutional neural network (CNN), this new topological method gives rise to some of the best predictions for protein-ligand binding affinities [14], protein folding free energy changes upon mutations [11, 13] and drug virtual screening [12]. This approach has won many contests in D3R Grand Challenges, a worldwide competition series in computer-aided drug design [63]. However, the techniques designed for protein-ligand binding analysis can not be directly applied to PPI due to biological differences and the different characteristics of available datasets.

In this work, we introduce site-specific persistent homology tailored for PPI analysis. We hypothesize that a topological approach that generates intrinsically low-dimensional representations of PPIs could dramatically reduce the dimensionality of antibody-antigen complexes, leading to a reliable high-throughput screening in searching for valuable mutants in protein design.

3.2 Dataset

3.2.1 AB-Bind dataset

The AB-bind dataset includes 1101 mutational data points with experimentally determined binding affinities [83]. We follow Pires *et al.* [68] to consider only 645 single mutations across 29 antibody-antigen complexes. Among them, 87 mutations are on 5 complexes with homology structures. This dataset, called the AB-bind S645 set, consists of about 20% stabilizing mutations and 80% destabilizing ones. Besides, in the whole dataset, there are 27 non-binders, which are variants determined not to bind within the sensitivity of the assay. The binding affinity changes upon mutation of these non-binders were set to -8 kcal/mol. These non-binders could be regarded as outliers in the database and have a strongly negative impact on the prediction model accuracy. Following figure shows the counts of mutation type in AB-bind database

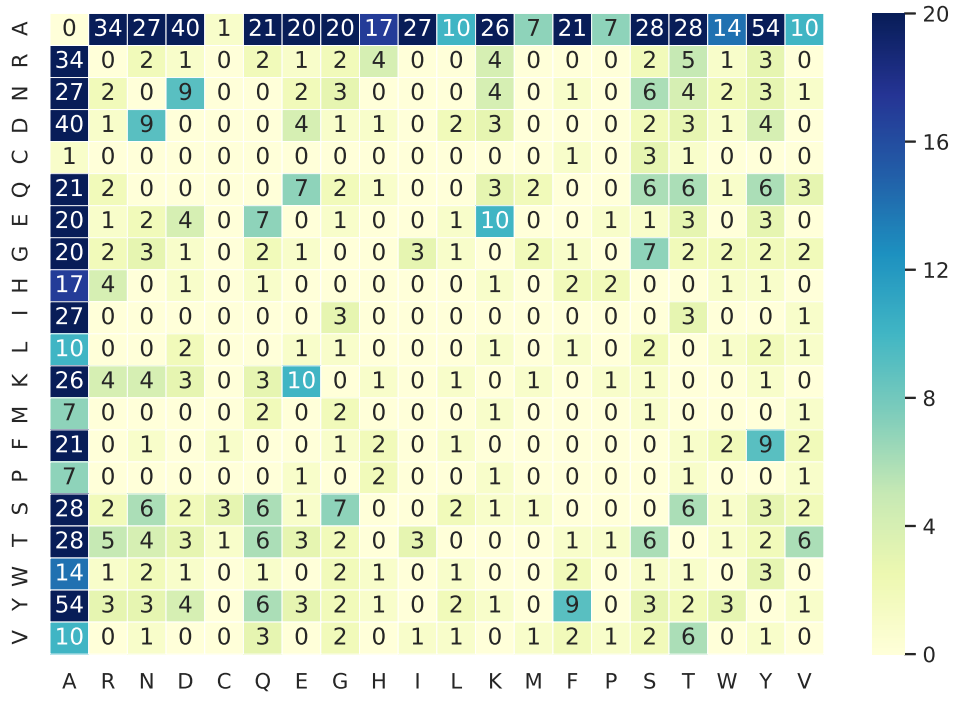


Figure 3.1: Counts of mutation type in AB-bind dataset

3.2.2 SKEMPI and SKEMPI 2.0 dataset

The SKEMPI dataset [61] contains 3047 binding free energy changes upon mutation assembled from the scientific literature, for protein-protein heterodimeric complexes with experimentally determined structures. It includes single-point mutations and multi-point mutations. Among the whole database, there are 2317 single point mutation data entries, called the SKEMPI S2317 set.

Recently, Xiong *et al.* selected a subset of 1131 non-redundant interface single-point mutations, denoted set S1131, from the SKEMPI set S2317 [91]. The same authors applied several methods to the SKEMPI S1131 set [91], including BindProfX [91], Profile-score [54, 84] FoldX [79] BeAtMuSiC [22],SAMMBE [66] and Dcomplex [57].

The SKEMPI 2.0 [47] database is the updated version of the SKEMPI database and contains new mutations collected after the first version was released. There are 7085 mutations in the SKEMPI 2.0 dataset. We choose only single-point mutations with full energy change information, called set S4947. Since binding energy changes upon mutation ($\Delta\Delta G$) are not directly given in the SKEMPI 2.0 database, the following formula is used to obtain the $\Delta\Delta G$ value for each mutation with a given kd value:

$$\Delta G = \frac{8.314}{4184} \times (273.15 + 25) \times log(kd)$$
$$\Delta\Delta G = \Delta G_{MT} - \Delta G_{WT}.$$

Set S4169 is directly adopted from mCSM-PPI2 [72] paper, which is also derived from the SKEMPI 2.0 dataset. Set S8338 is derived from the S4169 set by setting the reverse mutation energy change with a negative sign [72]

3.2.3 Preprocessing of dataset

For the aforementioned databases, crystal structures of the wild type, mutation type, and binding affinity change are given for each data entry. To calculate our structure-based topological feature, the structures of mutant type are also needed. Scap utility in the Jackal package [90] is used to generate mutant structures. This utility predicts side-chain conformations on a given backbone. To

fix the missing atoms and residues, the profix utility in the Jackal package [90] is applied to all raw pdb files.

3.3 Topological Modelling

3.3.1 Persistent homology

In algebraic topology, atomic coordinates of protein structures are organized into simplicial complexes, which are the basic elements of chains and homology groups, enabling the topological description of biomolecular datasets. Persistent homology further introduces a filtration parameter to examine biomolecular datasets at a variety of spatial scales. Element-specific persistent homology embeds chemical and biological information in topological representations by controlling certain atomic types in each simplicial complex.

3.3.1.1 Simplicial complex and filtration

A (geometric) simplicial complex is a finite collection of sets of affinely independent points (i.e., atomic positions) $K = \{\sigma_i\}_i$, where the elements in σ_i are called vertices and σ_i is called a k-simplex if it has $k + 1$ distinct vertices. If $\tau \subseteq \sigma_i$, τ is called a face of σ_i. A simplicial complex K is valid if $\tau \subseteq \sigma_i$ for $\sigma_i \in K$ indicates $\tau \in K$, and that the intersection of two simplices is either a simplex in K or empty.

In practice, it is favorable to characterize points clouds or atomic positions in various spatial scales rather than in a fixed scaled simplicial complex representation. To construct a scale-changing simplicial complex, consider a function $f : K \to \mathbb{R}$ satisfying $f(\tau) \leq f(\sigma)$ whenever $\tau \subseteq \sigma$. Given a real value x, f induces a subcomplex of K by constructing a sub-level set, $K(x) = \{\sigma \in K \mid f(\sigma) \leq x\}$. Since K is finite, the range of f is also finite and the induced subcomplexes, when ordered, form a filtration of K

$$\emptyset \subset K(x_1) \subset K(x_2) \subset \cdots \subset K(x_\ell) = K. \tag{3.1}$$

There are many constructions of f and a widely used one for point clouds is the Vietoris-Rips

complex. Given K as the collection of all possible simplices from a set of atomic coordinates up to a fixed dimension, the filtration function is defined as $f_{\text{rips}}(\sigma) = \max\{d(v_i, v_j) \mid v_i, v_j \in \sigma\}$ for $\sigma \in K$ where d is a predefined distance function between vertices, such as the Euclidean distance. In practice, an upper bound of the filtration value is set to avoid excessively large simplicial complex. Another efficient construction called alpha complex is often used to characterize geometry and we denote the filtration function by $f_\alpha : \text{DT}(X) \to \mathbb{R}$ where $\text{DT}(X)$ is the simplicial complex induced by the Delaunay triangulation of the set of atomic coordinates X. The filtration function is defined as $f_\alpha(\sigma) = \max\{\frac{1}{2}D_e(v_i, v_j) | v_i, v_j \in \sigma\}$ for $\sigma \in \text{DT}(X)$ where D_e is the Euclidean distance. Back to molecular structures, the filtration of simplicial complexes describes the topological characteristics of interaction hypergraphs under various interaction range assumptions.

3.3.1.2 Homology and persistence

Homology group (in singular homology) of a simplicial complex topologically depicts hole-like structures of different dimensions. Given a simplicial complex K, a k-chain is a finite formal sum of k-simplices in K, $\sum_i a_i \sigma_i$. There are many choices for the coefficients a_i and we choose $a_i \in \mathbb{Z}_2$ for simplicity. The kth chain group denoted $C_k(K)$ consists of all the k-chains under the addition induced by the addition of coefficients. A boundary operator $\partial_k : C_k(K) \to C_{k-1}(K)$ connects chain groups of different dimensions by mapping a chain to the alternating sum of codim-1 faces. It suffices to give the definition of the boundary operator on simplices,

$$\partial_k(\{v_0, \cdots, v_k\}) = \sum_{i=0}^{k} (-1)^i \{v_0, \cdots, \hat{v}_i, \cdots, v_k\}, \tag{3.2}$$

where $\hat{v}_i$ means the absence of vertex v_i. The kth cycle group denoted $Z_k(K)$ is defined to be the kernel of ∂_k whose members are called k-cycles. The kth boundary group is the image of ∂_{k+1} and is denoted $B_k(K)$. It follows that $B_k(K)$ is a subgroup of $Z_k(K)$ based on the property of boundary maps, $\partial_k \circ \partial_{k+1} = 0$. The kth homology group $H_k(K)$ is defined to be quotient group $Z_k(K)/B_k(K)$. The equivalent class in $H_k(K)$ corresponds to k-dimensional holes in K that can not be deformed to each other by adding/subtracting the boundary of a subcomplex.

Given a filtration as in Eq. (3.1), in addition to characterizing the homology group at each frame $H_k(K(x_i))$, we also want to track how topological features persist along the sequence. Viewing $H_k(K(x_i))$ as vector spaces together with inclusion map induced linear transformations gives a persistent module,

$$H_k(K(x_1)) \to H_k(K(x_2)) \to \cdots \to H_k(K(x_\ell)). \tag{3.3}$$

An interval module with respect to $[b, d)$ denoted $\mathbb{I}_{[b,d)}$ is defined as a collection of vector spaces $\{V_i\}$ connected by linear maps $f_i : V_i \to V_{i+1}$, where $V_i = \mathbb{Z}_2$ for $i \in [b, d)$ and $V_i = 0$ elsewhere and f_i is identity map when possible and 0 otherwise. The persistence module in Eq. (3.3) can be decomposed as a direct sum of interval modules $\oplus_{[b,d)\in B}\mathbb{I}_{[b,d)}$. Each $\mathbb{I}_{[b,d)}$ corresponds to a homology class that appears at filtration value b and disappears at filtration value d. The values b and d are usually called the birth and death values. The collection of these pairs B encodes the evolution of k-dimensional holes when varying the filtration parameter and thus records the topological configuration of the input point cloud under different interaction ranges if a distance based filtration is used. Fig. 3.2 illustrates filtration and persistence.

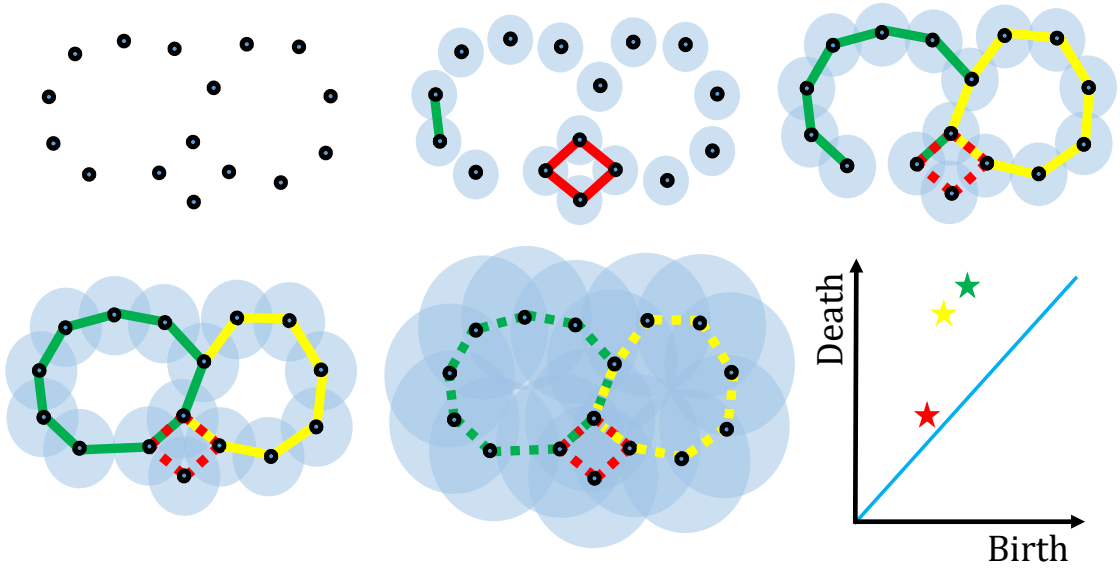


Figure 3.2: Illustration of filtration and persistence diagram of a set of points on a plane.

3.3.2 Topological representation of protein-protein interactions

The pairwise interactions between atoms are characterized by the 0th homology group, H_0 (also known as the size function [34]). The higher dimensional homology groups encode higher-order patterns in PPI complexes. The 1st homology group (H_1) generated with the Euclidean distance-based filtration characterizes loop or tunnel-like structures as shown in Fig.3.3, whereas the H_2 homology group describes cavity structures in PPI complexes. Combining various dimensions, we obtain a comprehensive topological description of PPIs.

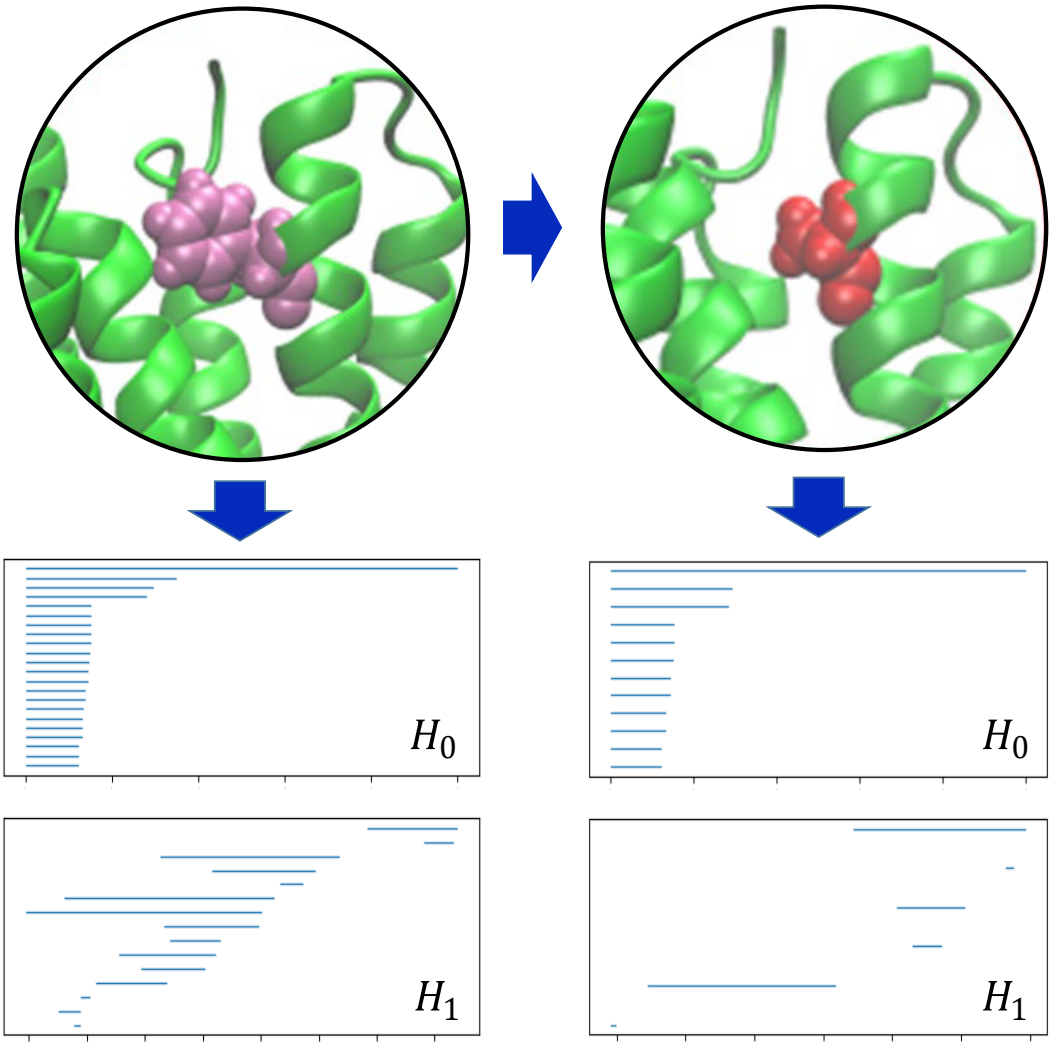


Figure 3.3: Topological barcode change associated with a mutation. Residue Leucine in the wild type is mutated into Alannine. Barcodes are generated for carbon atoms within a cutoff of 12Å of the mutant residue.

Given a PPI system represented by a set of atomic coordinates (i.e., a point cloud), a topological representation should be able to extract patterns of different biological or chemical aspects, such as hydrogen bonds between oxygen and nitrogen atoms, hydrophobicity, polarizability, etc. To achieve this goal, we construct simplicial complexes using selected subsets of atomic coordinates and modified distance matrices.

For constructing element-specific and site-specific persistent homology, we classify the atoms in a PPI complex into various subsets: (1) $\mathcal{A}_m$: atoms of the mutation site. (2) $\mathcal{A}_{mn}(r)$: atoms in the neighborhood of the mutation site within a cutoff distance r. (3) $\mathcal{A}_{ab}(r)$: antibody atoms within a distance r of the binding site. (4) $\mathcal{A}_{ag}(r)$: antigen atoms within a distance r of the binding site. Finally, (5) $\mathcal{A}_e$(E): atoms in the system that has atoms of element type E. When characterizing interactions between atoms a_i and a_j in set $\mathcal{A}$ and/or set $\mathcal{B}$, we use a modified distance matrix to exclude the interactions between the atoms from the same set,

$$D_m(a_i, a_j) = \begin{cases} \infty & \text{if } a_i, a_j \in \mathcal{A}, \text{ or } a_i, a_j \in \mathcal{B}, \\ D_e(a_i, a_j) & \text{if } a_i \in \mathcal{A}, \text{ and } a_j \in \mathcal{B}, \end{cases} \tag{3.4}$$

where D_e is the Euclidean distance. Specific designations for sets $\mathcal{A}$ and $\mathcal{B}$ are given in Table. 3.1, which summarizes various topological barcodes.

Table 3.1: Summary of topological descriptors. The barcodes are generated upon mutant and wild type complexes.

$\mathcal{A}$	$\mathcal{B}$	Dis.	Complex	Dim.
$\mathcal{A}_m \cap \mathcal{A}_{ele}(E_1)$	$\mathcal{A}_{mn}(r) \cap \mathcal{A}_{ele}(E_2)$	D_{mod}	Rips	H_0
$\mathcal{A}_m \cap \mathcal{A}_{ele}(E_1)$	$\mathcal{A}_{mn}(r) \cap \mathcal{A}_{ele}(E_2)$	D_e	alpha	H_1, H_2
$\mathcal{A}_{Ab}(r) \cap \mathcal{A}_{ele}(E_1)$	$\mathcal{A}_{Ag}(r) \cap \mathcal{A}_{ele}(E_2)$	D_{mod}	Rips	H_0
$\mathcal{A}_{Ab}(r) \cap \mathcal{A}_{ele}(E_1)$	$\mathcal{A}_{Ag}(r) \cap \mathcal{A}_{ele}(E_2)$	D_e	alpha	H_1, H_2

Following Fig.3.4 shows the point cloud generation of antibody-antigen complex 1DQJ

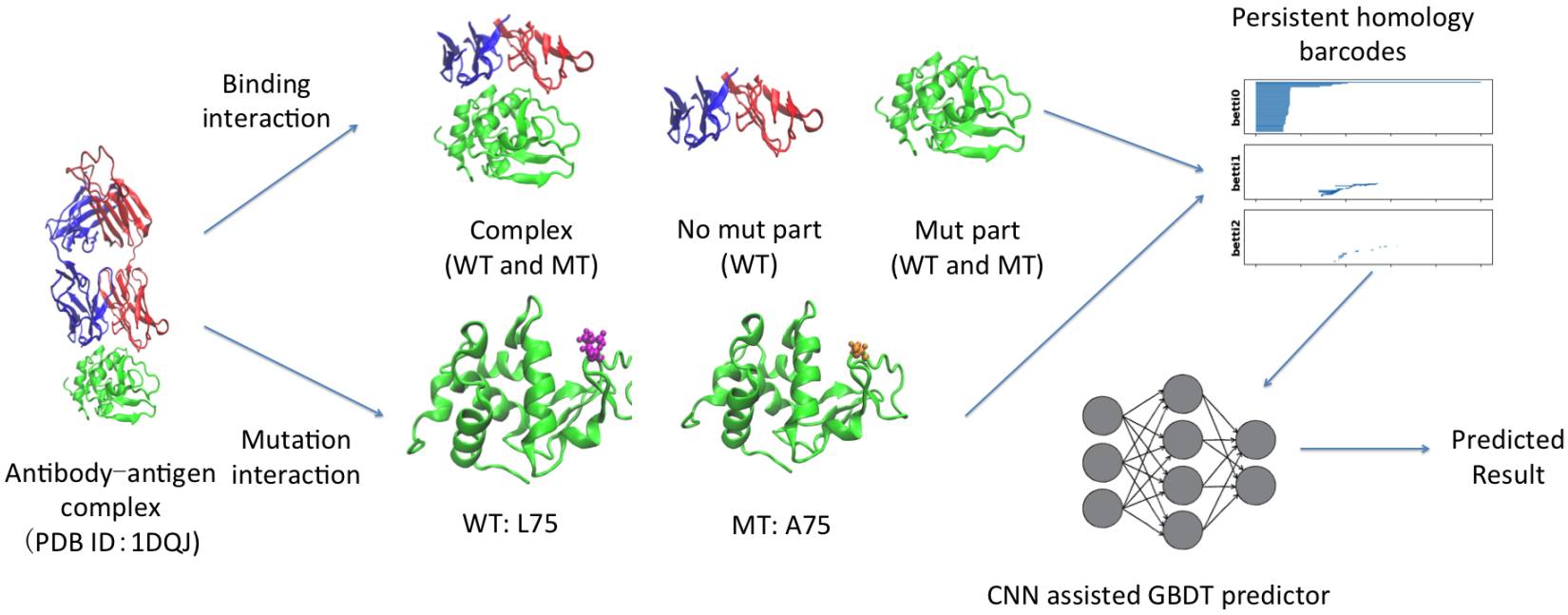


Figure 3.4: Illustration of the point cloud generation of antibody-antigen complex 1DQJ

3.4 Auxiliary features

As we mentioned in the previous section, element-specific and site-specific persistent homology is able to embed chemical information into topological representations. However, there are other important chemical and physical information that has not been incorporated into persistent homology but could improve the predictive power of the present topological model. In this work, all none topological features are named as auxiliary features. These features are appended into the machine learning model at the last step of GBT training or the dense layer of a neural network. In general, auxiliary features are categorized into atom-level features and residue-level ones.

3.4.1 Atom-level features

According to different criteria, atoms can be categorized into different groups for feature generation. First, with respect to atom types, we divide atoms into 7 groups, i.e., C, N, O, S, H, all heavy atoms, and all atoms. Additionally, with respect to distance to mutation site, atoms are grouped into 3 groups, namely, mutation site atoms, near mutation site atoms (within 10Å of mutation site), and all atoms. Finally, similar to the treatment in topological feature generation, 3 cases, i.e., wild type, mutant type, and their difference are considered, respectively.

- **Surface areas** Atom-level solvent excluded surface areas are computed through our in-house software ESES [56]. All atom areas within the same group are summed as one feature. In this manner, a total of 7*3*3 = 63 features is generated.

- **Partial charges** Partial charge of each atom is generated from pdb2pqr software [25] using the amber force field. After the procedure, the radius and the partial charge of each atom are calculated. The sum of the partial charges and the sum of absolute values of partial charges for each atomic group are counted as partial charge features. In this way, a total of 7*3*3*2 =126 features is generated.

- **Coulomb interactions** Coulomb energy of the ith single atom is calculated as the sum of

pairwise coulomb energy with every other atom.

$$C_i = \sum_{j,j\neq i} k_e \frac{q_i q_j}{r_{ij}}. \tag{3.5}$$

Here, k_e is the Coulomb's constant. Since multiplying the constant coefficient has no effect on machine learning result, we use $k_e = 1$ in our calculation.

In coulomb interaction feature generation, only 5 groups (C, N, O, S, and all heavy atoms) are counted. Both coulomb interaction energy and absolute value are counted. In this manner, a total of 5*3*3*2 = 90 features is generated.

- **van der Waals interaction** The van der Waals energy of the ith atom is modeled as the sum of pairwise Lennard-Jones potentials with every other atom. Only 5 groups (C, N, O, S, and all heavy atoms) are counted.

$$V_i = \sum_{j,j\neq i} \epsilon \left[\left(\frac{r_i + r_j}{r_{ij}} \right)^{12} - 2 \left(\frac{r_i + r_j}{r_{ij}} \right)^{6} \right]. \tag{3.6}$$

Here, ϵ is the depth of the potential well. Since multiplying the constant coefficient has no effect on machine learning result, we use $\epsilon = 1$ in our calculation. In this manner, a total of 5*3*3 = 45 features is generated.

- **Electrostatic solvation free energy** Electrostatic solvation free energy of each atom is calculated using Poisson-Boltzmann model through our in-house software MIBPB [97, 96, 39]. By summing up all the solvation free energies in same atom groups, 7*3*3 = 63 features are generated.

3.4.2 Residue-level features

- **Mutation site neighborhood amino acid composition** The residues within 10 Å of the mutation site are regarded as neighbor residues. Distances between residues are calculated using their alpha carbon atoms. Amino acid residues are divided into 5 groups as hydrophobic, polar, positively charged, negatively charged and special cases. The count and percentage of

the 5 groups of amino acids in neighbor site are regarding as the environment composition features of the mutation site, which leads to 5*2 = 10 features. Also, the sum, average and variance of residue volumes, surface areas, weights and hydropathy scores are generated as the environment chemical and physical features of a mutation site, which leads to 3*4 = 12 features. In this manner, 10+12 = 22 features are generated.

- **pK_a shifts** The pK_a values of 7 ionizable amino acids, namely, ASP, GLU, ARG, LYS, HIS, CYS, and TYR, are calculated using the PROPKA software [4]. The difference of pK_a values between a wild type and its mutant type are calculated as pK_a shifts. The maximum, minimum, sum, the sum of absolute values, the minimum of absolute value of total pK_a shifts are calculated, which leads to 5 features. Also, besides the shifts of all groups, the sum and the sum of absolute value of pK_a shifts based on the 7 ionizable amino acid groups are calculated, which leads to 2*7=14 features. In this manner, 5+14 = 19 features are generated.

- **Secondary structures** Using SPIDER2 [92] software, the probability score of mutation site residues to be coil, helix or strand are calculated as well as torsion angles. The wild type, the mutant type and their difference are calculated as secondary structure features. In this manner, 4*3 =12 features are generated.

3.5 Machine learning architecture

A major challenge in the prediction of binding affinity changes upon mutation for PPIs is that the data is highly complex due to 3D structures while the datasets are relatively small. To overcome this difficulty, we designed a hybrid machine learning algorithm combining CNN and GBT. The topologically simplified description of the 3D structures are further converted into concise features by the CNN module. The GBT module then builds robust predictors with effective control of overfitting.

3.5.1 TopGBT: Topology based gradient boosting tree model

Ensemble method is a class of machine learning algorithms that build a powerful model from weak learners. It improves the performance upon the weak learners with the assumption that the individual learners are likely to make *different* mistakes and thus summing up the weak learners will reduce the overall error. In this work, we use GBTs which add a tree to the ensemble according to the current prediction error on the training data. This method performs well when there is a moderate number of features and is relatively robust against hyperparameter tuning and overfitting. The implementation provided by the scikit-learn package (version 0.18.1) [65] is used.

3.5.2 TopCNN: Topology based convolutional neural network model

CNN is one of the most successful deep learning architectures. A regular CNN is a special case of a multilayer artificial neural network where only local connections are allowed between convolution layers and the weights are shared across different locations. We use topology-based CNN (TopCNN) as an intermediate model. Specifically, we feed vectorized 0th-dimensional topological barcode (H_0) features into CNNs to extract higher-level features for the downstream model.

3.5.3 TopNetTree: Topology based network tree model

CNN can automatically extract high-level features from the 0th-dimensional topological barcodes (H_0). These CNN extracted features are combined with features constructed from high-dimensional topological barcodes, H_1 and H_2, as the inputs of GBTs. Specifically, we build a supervised CNN model with the PPI $\Delta\Delta$G as labels. After the model is trained, we feed the flatten layer neural outputs into a GBT model to rank their importance. Based on the importance, a subset of CNN features is combined with other features, such as the statistics of H_1 and H_2 barcodes, for the final GBT model as shown in Fig.3.5. The GBT is used for its robustness against overfitting, good performance for moderately small data sizes and its model interpretability.

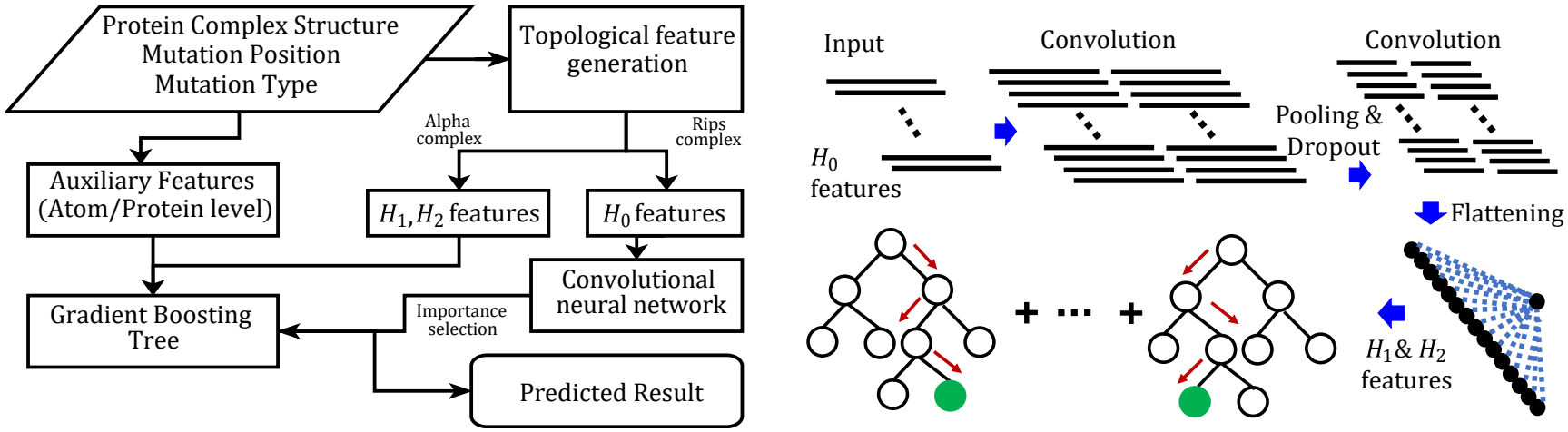


Figure 3.5: Illustration of the proposed TopNetTree model.

3.5.4 Model parametrization and software used

The details of model parameters and software packages are given below.

TopGBT: Topology based GBT model

- H_1 and H_2 features. Element-specific persistent homology H_1 and H_2 barcodes are constructed as described in Table 3.1 with cutoff value $r = 12$Å. We consider a wide type and mutant complexes. For each barcode, we extract birth death and persistence information. Statistical values, namely sum, min, max, mean, and standard deviation are computed from these barcodes to generate H_1 and H_2 features, giving rise to a total of 540 features.

TopCNN: Topology based CNN model

- H_0 feature. The same as what described above, except for a finer bin size of 0.25 Å, which leads to a total of 1296 features for CNN.

- Four 1D convolutional layers and one dropout layer have been used in the CNN model.

TopNetTree: Topology based network tree model

- H_0 features. Top 300 high-level CNN features are selected according to their feature importance.

- H_1 and H_2 auxiliary features are the same as those in the TopGBT model.

Model parameters

- CNN network structure and parameters are shown in Fig.3.6. This CNN network structure contains two 1D convolutional layer of 64 channels and two 1D convolutional layer of 128 channels and 1 flatten layers. On the convolutional dimension, 12 Åcut off and 0.25 Åbin size was chosen, so 48 bins are the size for that dimension. Other parameters for the CNN are listed as follow: $kernal\_initializer = lecun\_uniform$, $optimizer = adam$ and $epochs = 2000$

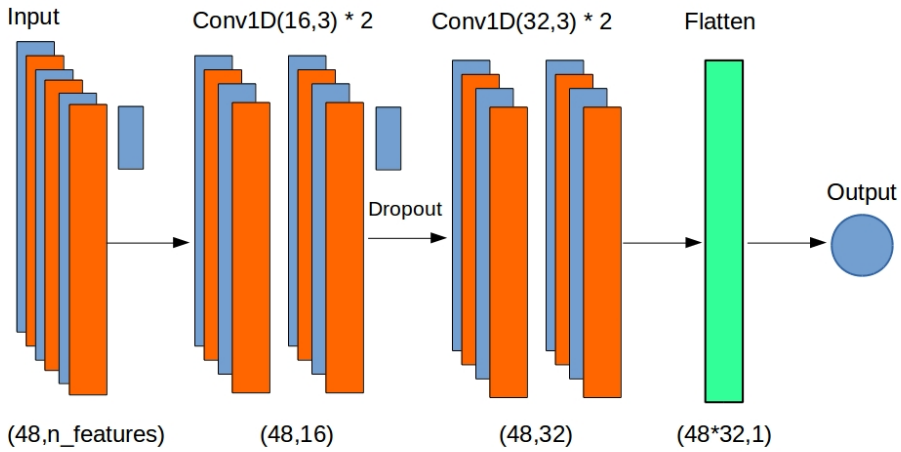


Figure 3.6: Illustration of CNN parameters.

- GBT parameters: $n\_estimators = 20000, max\_depth = 6, min_samples\_split = 3$, and $learning\_rate = 0.001$.

Software used

- GBT. The scikit-learn (version 0.18.1)[65] is used for the gradient boost regressor function.
- CNN. The Keras (version 2.0.2)[18] package is used for convolutional neural network model.
- Persistent homology feature: Javaplex [1] is used to generate H_0 barcodes and TDA package in R [28] is used to generate H_1 and H_2 barcodes.

Time and memory cost

- All the models are generated and tested on computer facilities at Michigan State University's High performance computing center (HPCC). 8 GB of memory and 5 cpu cores are requested for each feature generation job.
- Average running time for generating topological features for one sample is 1.01 min (time for generating mutant structure is included).

- Average running time for generating auxiliary features is 9.21 min .

CHAPTER 4

LTP MODEL: APPLICATION OF LOCAL TOPOLOGICAL CHARACTERISTICS IN PROTEIN FOLDING ENERGY CHANGE UPON MUTATION

4.1 Introduction

Mutagenesis is a process by which the genetic information of an organism is changed, resulting in a mutation. It may occur due to exposure to natural mutagens such as ultraviolet (UV) light, to industrial or environmental mutagens such as benzene or asbestos, or by deliberate mutagenesis for purposes of genetic research. Single Nucleotide Polymorphisms (SNPs) is one of the most important kind mutation in genetic research. For each human individual, around 10000-20000 non synonymous single-nucleotide polymorphisms (nsSNPs) appears in the genome[71]. Among those nsSNPs, some of them have no harm to protein function, but the rest loss-of-function nsSNPs are regarded as the most common cause of human heritable diseases[95, 52, 69]. Since the close relation between mutagenesis and certain type of disease, research of mutagenesis has always be at the cutting edge of molecular biology.

In thermodynamics, mutation could be accessed by the stability change $\Delta\Delta G$. ΔG is the energy change from unfolding state to folding state, and stability change.

$$\Delta\Delta G = \Delta G_w - \Delta G_m$$

While existing experimental methods for determine the $\Delta\Delta G$ value of mutation are very expensive and time consuming, fast computation methods to calculate mutation stability change is essentially needed. Current computational approaches could be categorized into three types: physical based methods, empirical methods and machine learning methods. For physical based methods, molecular mechanics (MM) is typically used to model the mutation, for example knowledge-modified MM/PBSA approach [40], EASE-MM [31]. For empirical models, empirical functions and potential terms are used to model the stability change and their weights are determined by the experimental data, for example Rosetta (high) protocols [48]. The last category of method is the

knowledge based machine learning method, for example STRUM [70]. In this type of method, various features which cannot easily modeled by physical term and potential term are regarded as input of machine learning protocol.

In the following chapter, we applied local topological descriptor to characterize the protein structure and generated a machine learning prediction model on protein folding energy change upon mutation. Our model get a performance of $R_p = 0.78$ on S2648 dataset, which exceed most of the state of art models.

4.2 Dataset

In this section we introduced ProTherm protein mutation database. Selection criteria and pre-processing steps were shown in the following parts.

4.2.1 ProTherm protein mutation database S2648 and S350

ProTherm[5] is a protein mutation database. Among the database 2648 different point mutations (S2648) in 131 proteins are chosen following the criteria below:

- Only mutations in globular proteins were considered
- Only mutant proteins whose experimental structure is available were taken
- Only single-site mutations were considered
- Mutations in heme-proteins are considered only if the stability measurements were performed on the apo form of the protein, and the structure of this apo form is available. Indeed, the interactions between residues and the heme are not taken into account
- Mutations that destabilize the structure by more than 5 kcal/mol and mutations involving a proline were not considered

A subset of 350 mutants corresponding to 67 different proteins was randomly selected as the evaluation set, namely S350 set [21]

4.2.2 Preprocessing of dataset

For the aforementioned databases, crystal structures of the wild type, mutation type, and binding affinity change are given for each data entry. To calculate our structure-based topological feature, the structures of mutant type are also needed. Scap utility in the Jackal package [90] is used to generate mutant structures. This utility predicts side-chain conformations on a given backbone. To fix the missing atoms and residues, the profix utility in the Jackal package [90] is applied to all raw pdb files.

4.3 Local topological characterization of molecules

Geometric modeling is one of the crucial part in molecular property prediction. With the power of differential geometry tools and Poincare Hopf index theorem, molecules can be interperated as a set of topological variables and thus can be used as the input features for machine learning model. In this section, we will introduce and discuss the topological tools for molecule characterization.

4.3.1 Molecular surface representations and molecular density function

With the assist of computer meshing, molecular area, volume and more detailed structural information could be available through molecular surface modeling. Moreover, with the generated molecular surface, a lot of molecular interaction properties, including protein-ligand binding energy, protein B-factor, protein folding energy upon mutation can be predicted with a higher accuracy.

There are several molecular surface models has been proposed. The van der Waals (vdWS) surface of a molecule is a representation of surface which based on the hard cutoffs of van der Waals radii for individual atoms. According to the definition, vdWS is not smooth at the intersection area of two or more atoms. Solvent accessible surface (SAS) is defined as the tracing of the center of a probe sphere rolling over the van der Waals surface of a molecule. Similar to the generation of SAS, solvent excluded surface (SES) is by tracing the inward union of areas by the spherical probe rolling over the vdWS of the molecule. According to the definition of the previous surface models, vdWS is inside SES and SES is smaller than SAS, and SES are relatively more smooth than the

other surfaces, although it still has singularities at the intersecting region. An example of vdWS, SAS and SES is shown in Fig.4.1

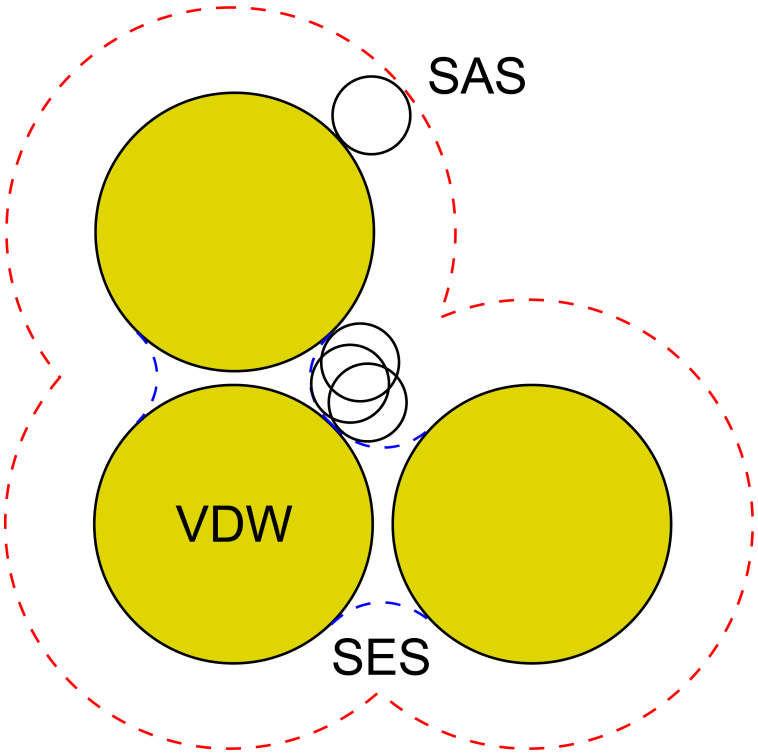


Figure 4.1: Illustration of van der Waals (vdWS) surface (yellow region), Solvent accessible surface (SAS) (red dotted margin) and solvent excluded surface (SES) (blue dotted margin

Instead of using the hard cutoff according to the atom radius, smooth biomolecular surfaces can be generated using smooth density functions. The rigid index is a smooth function which describe the interaction of two particles with decay to the distance using generalized exponential functions

$$\Phi(r_{ij};\eta_{ij}) = e^{-(r_{ij}/\eta_{ij})^{\kappa}}, \kappa > 0 \tag{4.1}$$

or generalized Lorentz function

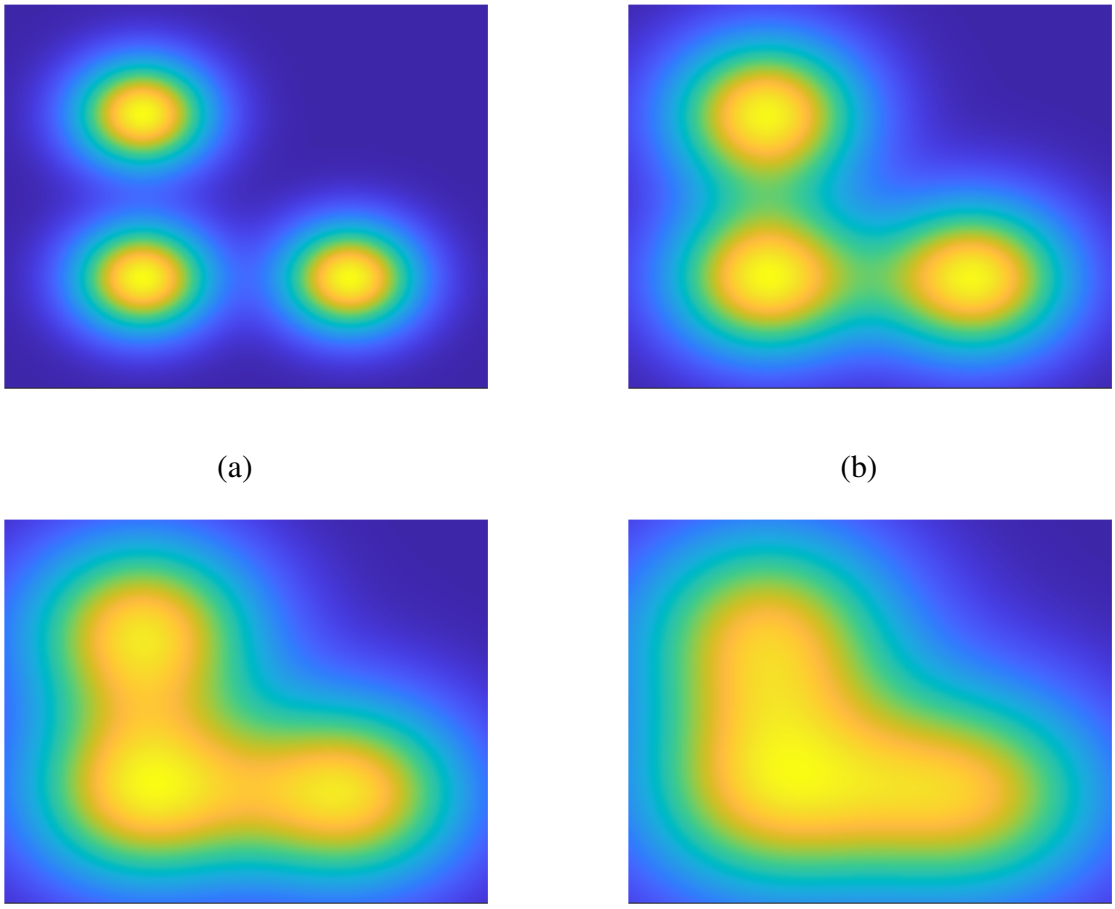
$$\Phi(r_{ij};\eta_{ij}) = \frac{1}{1+(r_{ij}/\eta_{ij})^{v}}, v > 0 \tag{4.2}$$

Here η_{ij} is a constant of characteristic distance between particles.

By extending and sum all the pairwise rigid index, one can get a continuous rigidity density

$$\mu(\mathbf{r}) = \sum_{j=1}^{N} w_j \Phi(r_{ij};\eta_{ij}) \tag{4.3}$$

Example of 2-D exponential density function using different η is shown in Fig 4.2. Subfigure a,b,c,d are generated from density function with $\eta = 1, 2, 3, 4$, respectively. Centers of three atoms are (0,0),(3,0),(0,3) respectively. With larger characteristic distance constant η , atom radius probes trend to merge into one connected surface.



(a) (b)

(c) (d)

Figure 4.2: An example of 2-D exponential density function generated surface.

With the smooth and density function, topological characters can be explicitly calculated in the following section.

4.3.2 Evaluation of curvature

The curvature of a surface is defined by the relationship between small positional changes on the surface, and the resulting changes in the surface normal.

Differential geometry tools have been introduced to evaluate the property of curvature. In differential geometry, the first fundamental form is the inner product on the tangent space of a surface in three-dimensional Euclidean space, noted by Roman numeral I. For a parametric surface $f(u,v) = (x(u,v), y(u,v), z(u,v))$ in $\mathbb{R}^3$, the inner product of two tangent vectors is

$$\begin{aligned} &\mathrm{I}(af_u + bf_v, cf_u + df_v) \\ =&ac < f_u, f_v > +(ad+bc) < f_u + f_v > +bd < f_v, f_v > \\ =&Eac + F(ad+bc) + Gbd \end{aligned} \tag{4.4}$$

The coefficient of first fundamental form is often written as a metric tensor of g_{ij}

$$(g_{ij}) = \begin{pmatrix} E & F \\ F & G \end{pmatrix}, g_{ij} = f_i \cdot f_j \tag{4.5}$$

Second fundamental form of a parametric surface is defined as: $\vec{r} = \vec{r}(u,v)$ be a regular parametrization of a surface in $\mathbb{R}^3$, which r_u and $\vec{r}_v$ are linearly independent for any (u,v) in the domain of $\vec{r}$. The unit normal vector can be thus calculated as

$$\vec{n} = \frac{\vec{r}_u \times \vec{r}_v}{|\vec{r}_u \times \vec{r}_v|} \tag{4.6}$$

Then the second fundamental form can be written as

$$\mathrm{II} = Ldu^2 + 2Mdudv + Ndv^2 \tag{4.7}$$

And the coefficient matrix in basis $\{\vec{r}_u, \vec{r}_v\}$ of the tangent plane is

$$(g_{ij}) = \begin{pmatrix} L & M \\ M & N \end{pmatrix} \times \vec{n} \tag{4.8}$$

$$L = \vec{r}_{uu} \times \vec{n}, M = \vec{r}_{uv} \times \vec{n}, N = \vec{r}_{vv} \tag{4.9}$$

With the definition of first and second fundamental form, one can calculate the Gaussian curvature as the fraction of the determinant of two fundamental forms as

$$K = \frac{det\text{II}}{det\text{I}} = \frac{LN - M^2}{EG - F^2} \tag{4.10}$$

Mean curvature can be calculated as the trace of $(\text{II})(\text{I}^{-1})$

$$H = \frac{1}{2} Trace((\text{II})(\text{I}^{-1})) \tag{4.11}$$

For a parametrization of surface $S = (x, y, f(x, y))$, according to the definition of K and H, one can get the expression of K and H in terms of f as following:

$$K = \frac{f_{xx} \cdot f_{yy} - f_{xy}^2}{(1 + f_x^2 + f_y^2)^2} \tag{4.12}$$

$$H = \frac{1}{2} \frac{(1 + f_x^2) f_{yy} - 2 f_x f_y f_{xy} + (1 + f_y^2) f_{xx}}{(1 + f_x^2 + f_y^2)^{3/2}} \tag{4.13}$$

For a given density function $\Phi(x, y, z)$, one can get the level set surface by letting $\Phi(x, y, z) = S_0$. Then implicit function theorem states that locally, there exists a function $z = f(x, y)$ which parametrizes the surface as $(x, y, f(x, y))$. By differentiating $\Phi(x, y, f(x, y)) = S_0$ with respect to x and y, one can get

$$\begin{aligned} \Phi_x + \Phi_z f_x = 0, f_x = -\frac{\Phi_x}{\Phi_z} \\ \Phi_y + \Phi_z f_y = 0, f_y = -\frac{\Phi_y}{\Phi_z} \end{aligned} \tag{4.14}$$

By further differentiation with respect to x and y, we can express f_{xx},f_{xy} and f_{yy} using the density function Φ by the following terms:

$$\begin{aligned} f_{xx} &= -\frac{\Phi_{xx} + \Phi_{xz} \cdot f_x - (\Phi_{xz} + \Phi_{zz} \cdot f_x)}{\Phi_z^2} = -\frac{\Phi_{xx} + \Phi_{xz} \cdot -\frac{\Phi_x}{\Phi_z} - (\Phi_{xz} + \Phi_{zz} \cdot -\frac{\Phi_x}{\Phi_z})}{\Phi_z^2} \\ f_{yy} &= -\frac{\Phi_{yy} + \Phi_{yz} \cdot f_y - (\Phi_{yz} + \Phi_{zz} \cdot f_y)}{\Phi_z^2} = -\frac{\Phi_{yy} + \Phi_{yz} \cdot -\frac{\Phi_y}{\Phi_z} - (\Phi_{yz} + \Phi_{zz} \cdot -\frac{\Phi_y}{\Phi_z})}{\Phi_z^2} \\ f_{xy} &= -\frac{\Phi_{xy} + \Phi_{xz} \cdot f_y - (\Phi_{yz} + \Phi_{zz} \cdot f_y)}{\Phi_z^2} = -\frac{\Phi_{xy} + \Phi_{xz} \cdot -\frac{\Phi_y}{\Phi_z} - (\Phi_{yz} + \Phi_{zz} \cdot -\frac{\Phi_y}{\Phi_z})}{\Phi_z^2} \end{aligned} \tag{4.15}$$

Plug in the terms of Equation.4.15 into Equation.4.14 we could get the explicit formula of H and K with the terms of derivative of density function. As we can see from the formula, H and K are only depend on the gradient $\nabla\Phi = (\frac{\partial\Phi}{\partial x}, \frac{\partial\Phi}{\partial y}, \frac{\partial\Phi}{\partial z})$ and the Hessian $Hess(\Phi)$ and the adjoint of the hessian $Hess^*(\Phi)$,

$$Hess(\Phi) = \begin{pmatrix} \frac{\partial^2\Phi}{\partial^2 x} & \frac{\partial^2\Phi}{\partial x\partial y} & \frac{\partial^2\Phi}{\partial x\partial z} \\ \frac{\partial^2\Phi}{\partial x\partial y} & \frac{\partial^2\Phi}{\partial^2 y} & \frac{\partial^2\Phi}{\partial y\partial z} \\ \frac{\partial^2\Phi}{\partial x\partial z} & \frac{\partial^2\Phi}{\partial y\partial z} & \frac{\partial^2\Phi}{\partial^2 z} \end{pmatrix} \tag{4.16}$$

$$Hess^*(\Phi) = \begin{pmatrix} \Phi_{yy}\Phi_{zz} - \Phi_{yz}\Phi_{zy} & \Phi_{yz}\Phi_{zx} - \Phi_{yx}\Phi_{zz} & \Phi_{yx}\Phi_{zy} - \Phi_{yy}\Phi_{zx} \\ \Phi_{xz}\Phi_{zy} - \Phi_{xy}\Phi_{zz} & \Phi_{xx}\Phi_{zz} - \Phi_{xz}\Phi_{zx} & \Phi_{xy}\Phi_{zx} - \Phi_{xx}\Phi_{zy} \\ \Phi_{xy}\Phi_{yz} - \Phi_{xz}\Phi_{yy} & \Phi_{yx}\Phi_{xz} - \Phi_{xx}\Phi_{yz} & \Phi_{xx}\Phi_{yy} - \Phi_{xy}\Phi_{yx} \end{pmatrix} \tag{4.17}$$

Expression of H and K by $\nabla\Phi$, $Hess^*(\Phi)$ and $Hess^*(\Phi)$ can be written as

$$K = \frac{\nabla\Phi \cdot Hess^*(\Phi)\nabla\Phi^T}{|\nabla\Phi|^4} \tag{4.18}$$

$$H = \frac{\nabla\Phi \cdot Hess(\Phi)\nabla\Phi^T - |\nabla\Phi|^2 Trace(Hess(\Phi))}{2|\nabla\Phi|^3} \tag{4.19}$$

4.3.3 Critical points and Poincare Hopf index theorem

A critical point (CP) of a field ρ is a point where $\nabla\rho = 0$. Gradient paths always originate at a critical point and terminate at another critical point.

The properties of critical can be determined by the Hessian matrix. The sum of eigenvalues of Hessian matrix equals to the Laplacian of density ρ

$$\nabla^2\rho = \gamma_1 + \gamma_2 + \gamma_3 = \frac{\partial^2\rho}{\partial^2 x} + \frac{\partial^2\rho}{\partial^2 y} + \frac{\partial^2\rho}{\partial^2 z} \tag{4.20}$$

For a convex function, the Hessian matrix is positive semi-definite, therefore all the eigenvalues are real. Based on the positive and negative sign of the three eigenvalues of Hessian matrix, one could determine if a critical point is a local maximum, local minimum or a saddle point as follows:

- if the Hessian is positive definite at point x (with three positive eigenvalues), then it's an isolated local minimum critical point
- if the Hessian is negative definite at point x (with three negative eigenvalues), then it's an isolated local maximum critical point
- if the Hessian has both positive and negative eigenvalues at point x, then x is a saddle point

There is another case that the determinant of the Hessian at x is zero, then x is called a degenerate critical point.

For simplicity, the rank of a critical point is defined as the number of non-zero eigenvalues of Hessian, and the signature of critical point is defined as the algebraic sum of the signs (+1 or -1) of the eigenvalues. For a non-degenerate critical point, its signature can be 3,1,-1 or -3.

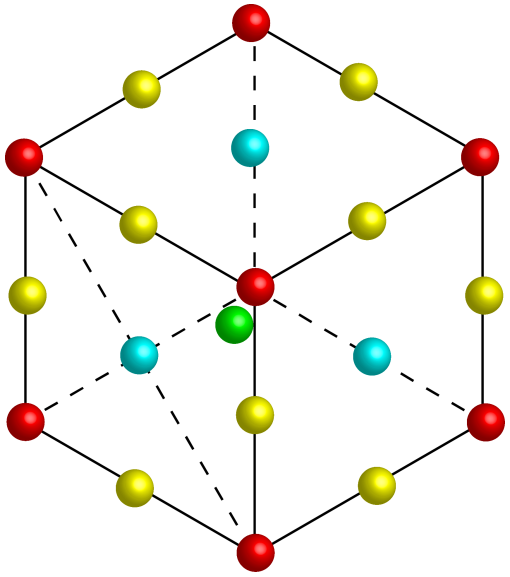


Figure 4.3: Example of different critical points in a standard cube. Nucleic critical point (NCP) in red, Bond critical point (BCP) in yellow, Ring critical point (RCP) in cyan and Cage critical point (CCP) in green.

Apply the previous definition into molecular density field, we could define the following molecular critical points:

- Nucleic critical point (NCP), a nucleic center of an atom, which is the local maximum of the field
- Bond critical point (BCP), a saddle point on a bond center between two atoms
- Cage critical point (CCP), a saddle point at the center of a cage
- Ring critical point (RCP), a center of a ring structure, which is the local minimum of the field

Examples of different types of critical point are shown in the Fig.4.4 and Fig.4.5 with the structure of naphthalne ($C_{10}H_8$). NCP in red, BCP in yellow and RCP in cyan.

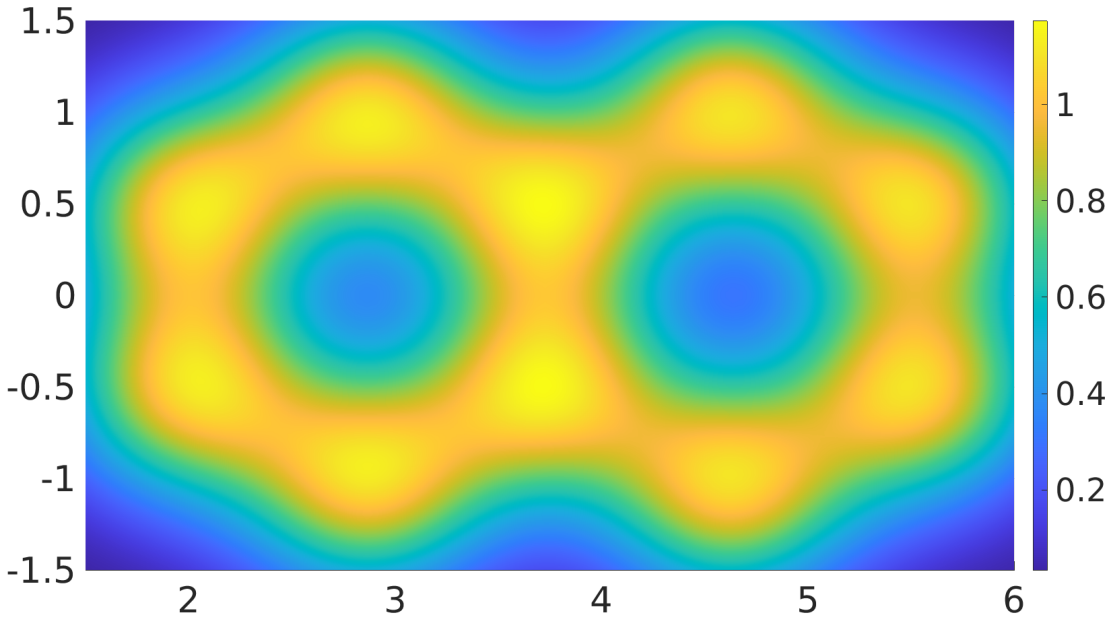


Figure 4.4: Density field of naphthalne

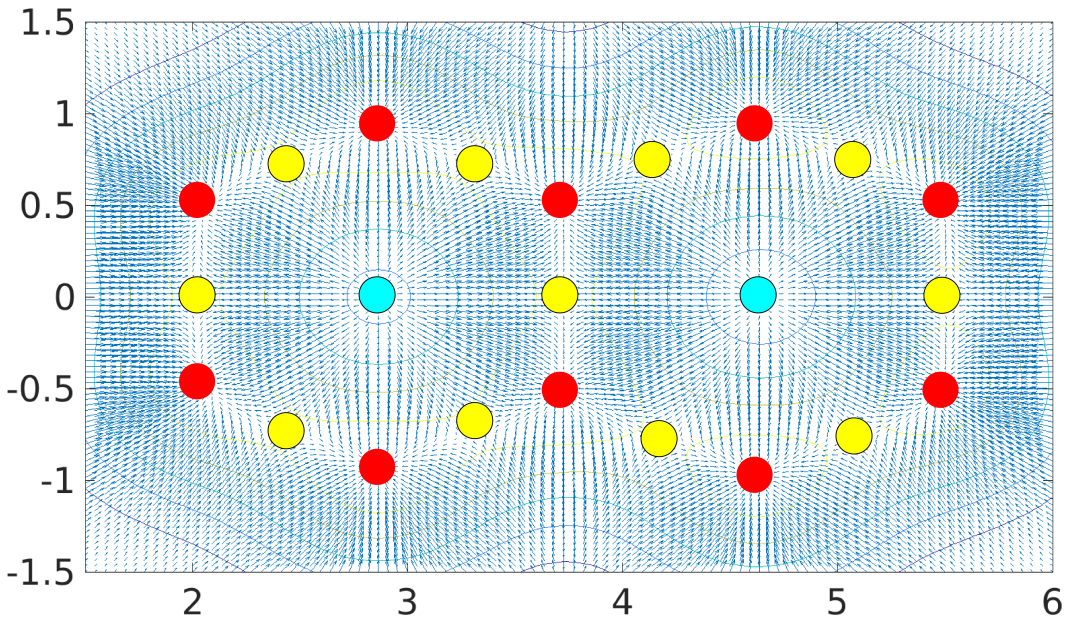


Figure 4.5: Gradient field of naphthalne

The properties of critical points are also close related to the Euler characteristic using the Poincare-Hopf theorem:

Theorem 1 (Poincare-Hopf) *Let v be a vector field on M, where M is without boundary, which also only has isolated zeros. Then the sum of the indices of the zeros of the vector field is equal to the Euler characteristic of the manifold.*

According to the theorem, the Poincare indices for NCP, BCP, RCP and NCP are 1,-1,1,and -1, respectively, so the Euler characteristic $\chi(\phi)$ could be calculated using the number of critical points as following:

$$N_n - N_b + N_r - N_c = \chi(\phi) \tag{4.21}$$

Combine all the properties of critical points, we can get the following table states the molecular critical points of density field:

Table 4.1: Properties of critical points

Critical point type	rank	signature	Poincare index	max/min/saddle
Nucleic	3	1	1	local maxima
Bond	3	-1	-1	saddle
Cage	3	1	1	saddle
Ring	3	-1	-1	local minima

4.3.4 Mutation site based element specific density field generation

In protein folding energy change upon mutation prediction, we want to focus on the mutation interaction and get rid of other unrelated interaction, mutation site based density field generation has been applied.

Two sets of atoms are used in the mutation site based density field, named density generation set and eigenvalue evaluation set. We choose all the atom center in mutation residue as the density field generation set D. The corresponding density field is then defined as:

$$\mu(\mathbf{r}) = \sum_{j \in D} \Phi(r_{ij}; \eta_{ij}) \tag{4.22}$$

After the generation of density field, we want to evaluate the eigenvalues of Hessian matrix of the density field at certain points, namely eigenvalue evaluation set. In this problem, we set all the atom center near mutation site within a cutoff (exclude the mutation residue atoms) as the evaluation set E.

$$< \gamma_{i1}, \gamma_{i2}, \gamma_{i3} >= eig(Hess(\mu(r_i))), r_i \in E \tag{4.23}$$

To incorporating chemical information into the density field, we further categorized density field generation set and eigenvalue evaluation set with respect to atom types of $\{C, N, O, S\}$. Raw feature set of atom type a_1 of D and atom type a_2 of E can be expressed as following:

$$F_{a_1,a_2} = \{eig(Hess(\mu(r_i)))\}_{i \in E, A_i = a_2} = \{eig(Hess(\sum_{j \in D, A_j = a_1} \Phi(r_{ij}; \eta_{ij}))\} \tag{4.24}$$

4.3.5 Explicit expression of Hessian matrix

Hessian matrix for a density function Φ is depend on the second derivative terms of Φ. To get the eigenvalue of Hessian, we calculate the explicit form of Hessian matrix with respect to density function.

For exponential kernel

$$\Phi(r_{ij};\eta_{ij}) = e^{-(r_{ij}/\eta_{ij})^{\kappa}}, \kappa > 0 \tag{4.25}$$

In terms of 3-D coordinate (x, y, z) with density generation set D,

$$\Phi(x, y, z; \eta) = \sum_{i \in D} e^{-((x-x_i)^2+(y-y_i)^2+(z-z_i)^2)^{1/2}/\eta)^{\kappa}}, \kappa > 0 \tag{4.26}$$

We can get the explicit expression of terms in Hessian as follows

$$\begin{aligned} c_1 &= e^{-((x-x_i)^2+(y-y_i)^2+(z-z_i)^2)^{1/2}/\eta)^{\kappa}} = \Phi \\ c_2 &= (x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2 = r^2 \end{aligned} \tag{4.27}$$

For $\frac{\partial^2\Phi}{\partial^2 x}, \frac{\partial^2\Phi}{\partial^2 y}, \frac{\partial^2\Phi}{\partial^2 z}$

$$\begin{aligned} \frac{\partial^2\Phi}{\partial^2 x} &= \frac{-\kappa[-\kappa \cdot c_1 \cdot c_2^{\kappa-2}(x-x_i)^2 + 2c_1 \cdot (0.5\kappa-1)c_2^{0.5\kappa-2}(x-x_i)^2 \cdot \eta^{\kappa} + c_1 \cdot c_2^{0.5\kappa-1} \cdot \eta^{\kappa}]}{\eta^{2\kappa}} \\ \frac{\partial^2\Phi}{\partial^2 y} &= \frac{-\kappa[-\kappa \cdot c_1 \cdot c_2^{\kappa-2}(y-y_i)^2 + 2c_1 \cdot (0.5\kappa-1)c_2^{0.5\kappa-2}(y-y_i)^2 \cdot \eta^{\kappa} + c_1 \cdot c_2^{0.5\kappa-1} \cdot \eta^{\kappa}]}{\eta^{2\kappa}} \\ \frac{\partial^2\Phi}{\partial^2 z} &= \frac{-\kappa[-\kappa \cdot c_1 \cdot c_2^{\kappa-2}(z-z_i)^2 + 2c_1 \cdot (0.5\kappa-1)c_2^{0.5\kappa-2}(z-z_i)^2 \cdot \eta^{\kappa} + c_1 \cdot c_2^{0.5\kappa-1} \cdot \eta^{\kappa}]}{\eta^{2\kappa}} \end{aligned} \tag{4.28}$$

For $\frac{\partial^2\Phi}{\partial x \partial y}, \frac{\partial^2\Phi}{\partial x \partial z}, \frac{\partial^2\Phi}{\partial y \partial z}$

$$\begin{aligned} \frac{\partial^2\Phi}{\partial x \partial y} &= \frac{-\kappa \cdot (x-x_i) \cdot (y-y_i)[-\kappa \cdot c_1 \cdot c_2^{\kappa-2} + c_1 \cdot c_2^{0.5\kappa-2}(\kappa-2) \cdot \eta^{\kappa}]}{\eta^{2\kappa}} \\ \frac{\partial^2\Phi}{\partial x \partial z} &= \frac{-\kappa \cdot (x-x_i) \cdot (z-z_i)[-\kappa \cdot c_1 \cdot c_2^{\kappa-2} + c_1 \cdot c_2^{0.5\kappa-2}(\kappa-2) \cdot \eta^{\kappa}]}{\eta^{2\kappa}} \\ \frac{\partial^2\Phi}{\partial y \partial z} &= \frac{-\kappa \cdot (y-y_i) \cdot (z-z_i)[-\kappa \cdot c_1 \cdot c_2^{\kappa-2} + c_1 \cdot c_2^{0.5\kappa-2}(\kappa-2) \cdot \eta^{\kappa}]}{\eta^{2\kappa}} \end{aligned} \tag{4.29}$$

Similarly, for Lorenz kernel

$$\Phi(r_{ij};\eta_{ij}) = \frac{1}{1+(r_{ij}/\eta_{ij})^{v}}, v > 0 \tag{4.30}$$

In terms of 3-D coordinate (x, y, z) with density generation set D,

$$\Phi(x, y, z; \eta) = \sum_{i \in D} \frac{\eta^{v}}{\eta^{v} + [(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2]^{0.5v}} \tag{4.31}$$

We can get the explicit expression of terms in Hessian as follows

$$\begin{aligned} c_1 &= \eta^{v} + [(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2]^{0.5v} = \frac{\eta^{v}}{\Phi} \\ c_2 &= (x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2 = r^2 \end{aligned} \tag{4.32}$$

For $\frac{\partial^2\Phi}{\partial^2 x}, \frac{\partial^2\Phi}{\partial^2 y}, \frac{\partial^2\Phi}{\partial^2 z}$

$$\begin{aligned} \frac{\partial^2\Phi}{\partial^2 x} &= \frac{-\eta^{v} \cdot v[c_1 \cdot c_2^{0.5v-1} + (v-2) \cdot c_1 \cdot c_2^{0.5v-2}(x-x_i)^2 - 2v \cdot c_2^{v-2}(x-x_i)^2]}{c_1^3} \\ \frac{\partial^2\Phi}{\partial^2 y} &= \frac{-\eta^{v} \cdot v[c_1 \cdot c_2^{0.5v-1} + (v-2) \cdot c_1 \cdot c_2^{0.5v-2}(y-y_i)^2 - 2v \cdot c_2^{v-2}(y-y_i)^2]}{c_1^3} \\ \frac{\partial^2\Phi}{\partial^2 z} &= \frac{-\eta^{v} \cdot v[c_1 \cdot c_2^{0.5v-1} + (v-2) \cdot c_1 \cdot c_2^{0.5v-2}(z-z_i)^2 - 2v \cdot c_2^{v-2}(z-z_i)^2]}{c_1^3} \end{aligned} \tag{4.33}$$

For $\frac{\partial^2\Phi}{\partial x\partial y}, \frac{\partial^2\Phi}{\partial x\partial z}, \frac{\partial^2\Phi}{\partial y\partial z}$

$$\begin{aligned} \frac{\partial^2\Phi}{\partial x\partial y} &= \frac{-\eta^{v} \cdot v \cdot (x-x_i) \cdot (y-y_i)[(v-2) \cdot c_1 \cdot c_2^{0.5v-2} - 2v \cdot c_2^{v-2}]}{c_1^3} \\ \frac{\partial^2\Phi}{\partial x\partial z} &= \frac{-\eta^{v} \cdot v \cdot (x-x_i) \cdot (z-z_i)[(v-2) \cdot c_1 \cdot c_2^{0.5v-2} - 2v \cdot c_2^{v-2}]}{c_1^3} \\ \frac{\partial^2\Phi}{\partial y\partial z} &= \frac{-\eta^{v} \cdot v \cdot (y-y_i) \cdot (z-z_i)[(v-2) \cdot c_1 \cdot c_2^{0.5v-2} - 2v \cdot c_2^{v-2}]}{c_1^3} \end{aligned} \tag{4.34}$$

With the explict Hessian matrix, we could calculate the eigenvalues easily with the matrix diagnalization.

Fig.4.6 is an example of two eigenvalue maps of naphthalne ($C_{10}H_8$) using the density field of (a) $\eta = 0.6$ and (b) $\eta = 1.0$, respectively. Compare (a1) with (a2), the small eigenvalue map captures

more bond information while the big eigenvalue map captures more nucleic center information. Compare (a) with (b), we get that with a smaller η value, eigenvalue maps trend to capture more detailed topological information of molecule. By control the value of η we can get the topological information at a scale suitable for the specific biomolecular property.

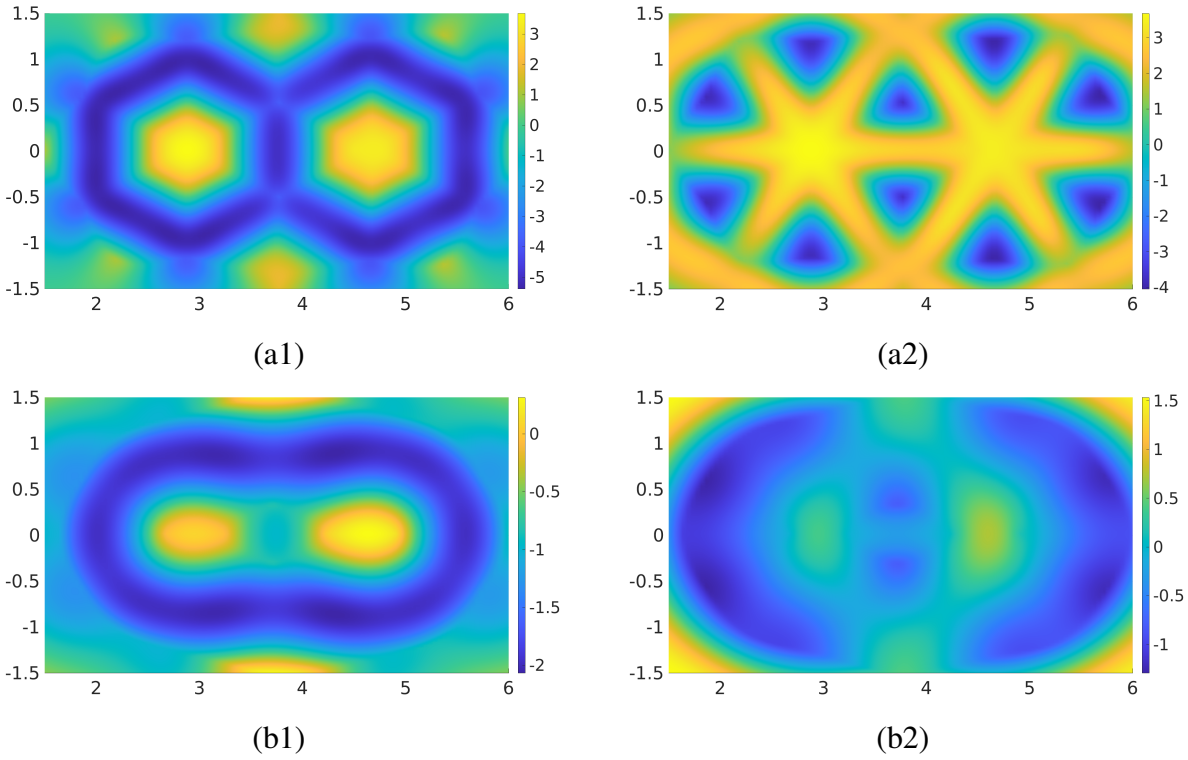
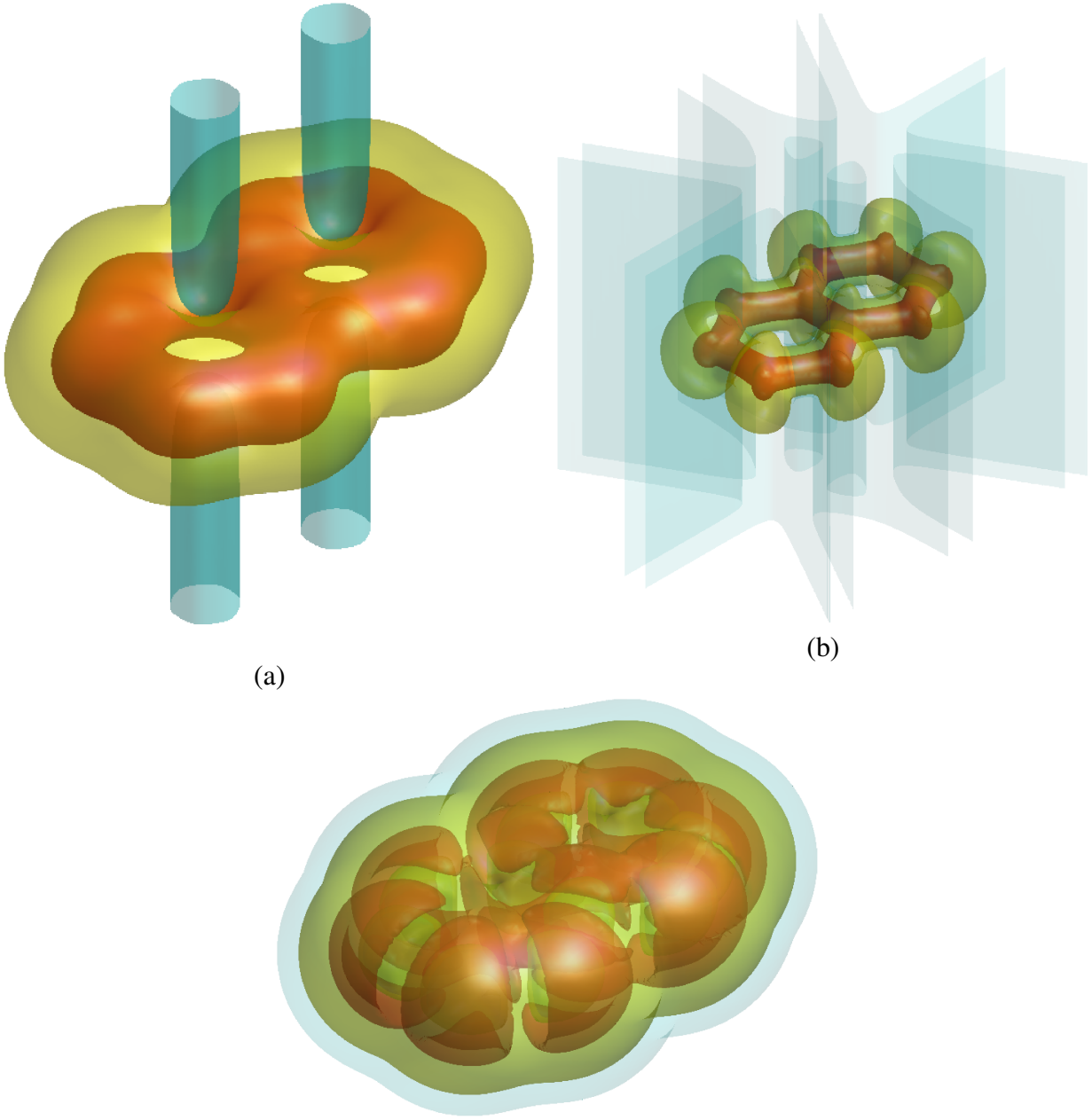


Figure 4.6: An example of two eigenvalue maps of naphthalne using two different density fields

Fig.4.7 shows an example of three eigenvalue isosurface maps of naphthalne ($C_{10}H_8$) using the exponential density field of $\eta = 0.6$. For the smallest eigenvalue map (a), the isovalues are -3,-0.5,0 for red, green and cyan respectively. For the second smallest eigenvalue map (b), the isovalues are -3,-0.3,0 for red, green and cyan respectively. For the largest eigenvalue map (c), the isovalues are 2.5,1,0.3 for red, green and cyan respectively.



(a) (b)

(c)

Figure 4.7: an example of three eigenvalue isosurface maps of naphthalne using the exponential density field

4.4 Prediction model design

Machine learning is a power tool in molecular property prediction. It can incorporate different types of features and learn the internal relations between features and properties. In the previous

section, we know that a lot of the topological characters can be derived from Hessian matrix. Instead of directly using the topological characters as the features of machine model, we decide to use the eigenvalues of Hessian matrix as the initial features of our model. We believe with the power of machine learning model, using eigenvalues as feature could capture more insight relation unlimited to the known topological characters.

4.4.1 Eigenvalue learning feature generation

Since machine learning requires the features to be scalable between samples, which means the number of features must be the same between samples. It is obvious that even use a same cutoff value near the mutation site, the number of atoms to evaluate eigenvalues of density could vary a lot, which lead to different scale of raw feature. To make it suitable for machine learning input features, we calculate the following five statistic values for each raw feature set F_{a_1,a_2}, including sum, max, min, average and standard deviation.

So for the final feature set, we have 3 eigenvalues, 5 stat values for each set, 4*4 =16 combination of atom type, in total 240 features for one structure. And we consider wild type features, mutant type features and their difference, which makes the feature set size to 240*3 = 720.

4.4.2 Auxiliary features

Although our eigenvalue features capture a lot of geometric information of the molecule, some crucial chemical physical properties can not be represented with only geometric features. To further improve the quality of our prediction models, several auxiliary features have been added to the model, including charge information, electrostatic solvation free energy, secondary structure information .etc. Detailed description of auxiliary features could be found at Chapter 3 of the thesis.

4.4.3 Machine learning model

Gradient boosting is a machine learning technique for regression problems, which produces a prediction model in the form of an ensemble of weak prediction models. We used Gradient boosting decision tree (GBDT) as the machine learning algorithm and feed both eigenvalue features and auxiliary features into the model.

The scikit-learn (version 0.18.1)[65] is used for the gradient boost regressor function with following parameters: $n\_estimators = 20000, max\_depth = 6, min\_samples\_split = 3$, and $learning\_rate = 0.001$. Pipeline of our model is shown as follows.

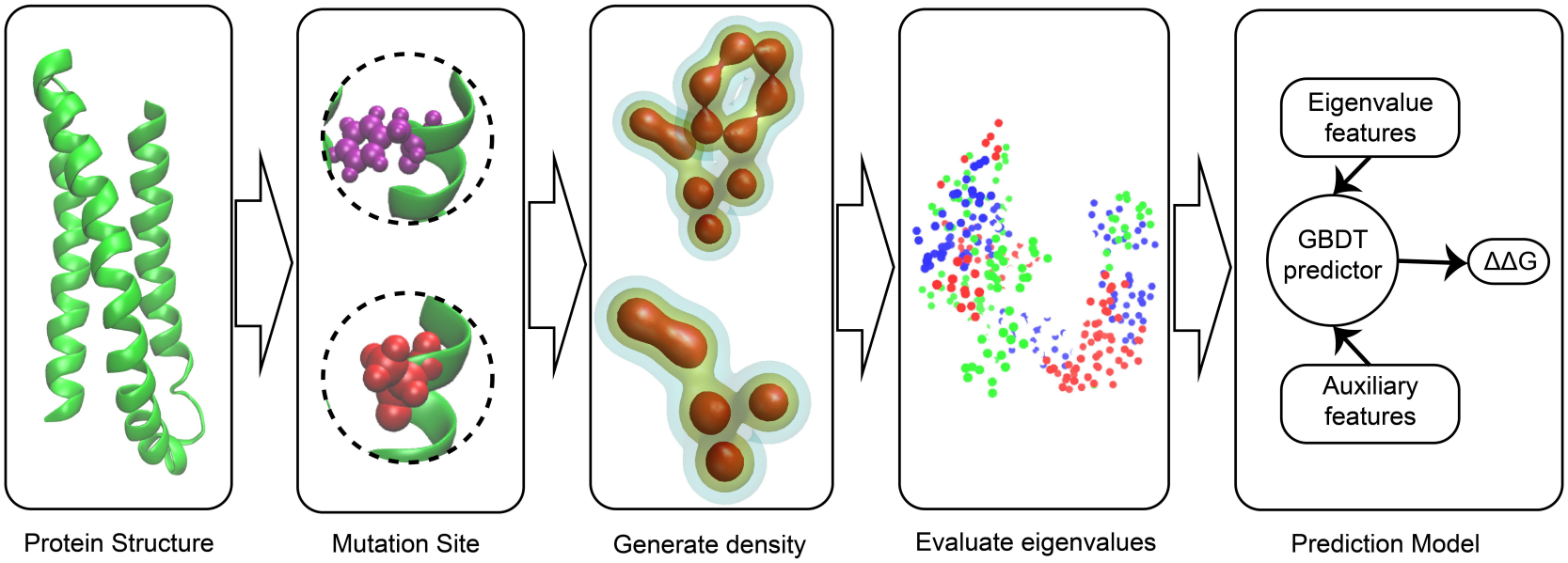


Figure 4.8: Flowchart of eigenvalue learning model of protein folding energy change upon mutation with example of 1A3J A 12 F A. First, mutation site is selected for both wild and mutant structure. Corresponding exponential/Lorentz density fields are generated. Then Hessian matrix and its eigenvalues are evaluated at the near mutation region with respect to different element groups. Finally, eigenvalue features and auxiliary features are fed into the GBDT model to get the prediction value of energy change.

CHAPTER 5

RESULT

5.1 Evaluation criteria

Two evaluation metrics, Pearson's correlation coefficient (R_p) and root-mean-squared error (RMSE), are used to assess the quality of predictions. Let x and y be the vector of predicted values and the ground truth of the n samples, respectively. The definition of R_p is given by

$$R_p = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}, \tag{5.1}$$

where $\bar{x}$ and $\bar{y}$, the means of x and y, respectively. RMSE is computed as

$$\text{RMSE} = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2/n} \tag{5.2}$$

For cross validation, the R_p and RMSE of all folds are averaged.

5.2 Model performance of TopNetTree on PPI binding free energy change

In this section, model performance of TopNetTree on PPI binding free energy change is evaluated on AB-Bind and SKEMPI dataset.

5.2.1 Prediction result on AB-Bind dataset

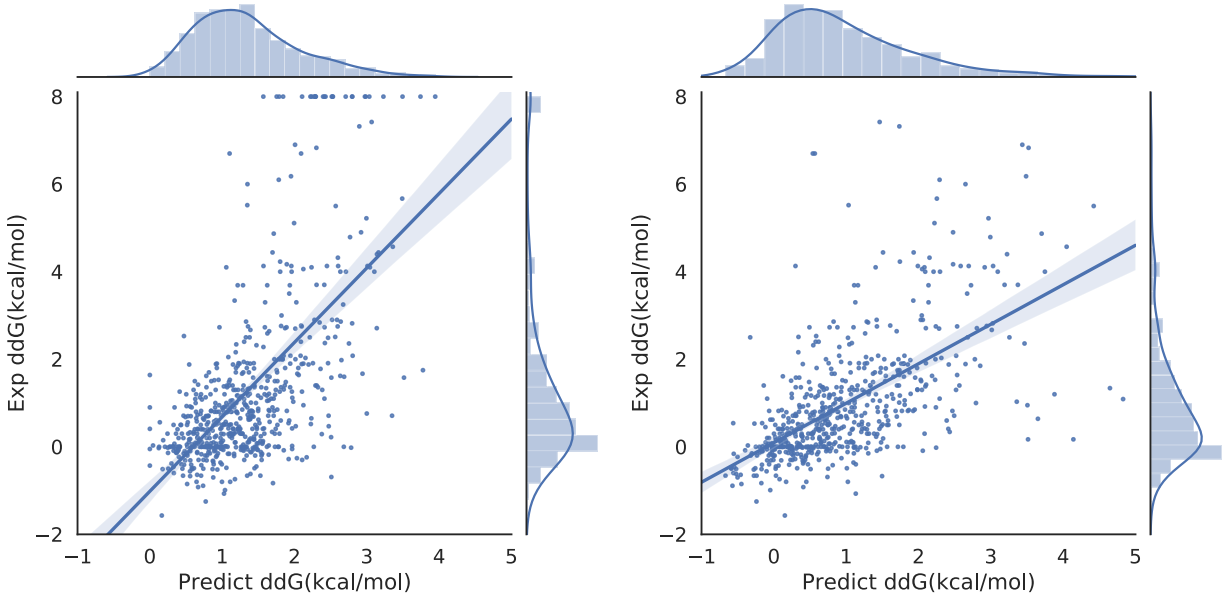
5.2.1.1 Overall result

Our model achieved a Pearson's correlation coefficient (R_p) of 0.65 on the AB-Bind S645 dataset, which is significantly better than those of other existing methods as shown in Table 5.1. Comparing to non-machine learning methods such as Rosetta and bASA, our method is over 100% more accurate in terms of R_p, indicating our topology-based machine learning methods have a better predictive power for PPI systems. Comparing to the best existing score of $R_p = 0.53$ given by mCSM-AB, our method is about 22% more accurate, indicating the power of our TopNetTree.

Both GBT and neural network are quite sensitive to system errors since the training of a model is based on optimizing the mean square error of the loss function. The ΔΔG of 27 non-binders (-8 kcal/mol) did not follow the distribution of the whole dataset. Pires *et al.* [68] found that excluding non-binders from the dataset would significantly increase the performance of a prediction model. In our case, the R_p increased from $R_p = 0.65$ in Fig.5.1a to $R_p = 0.68$ for the same treatment as shown in Fig.5.1b. We also applied a blind test on homology structures using the rest of the samples as the training set, achieving $R_p = 0.55$ as shown in Fig. 5.2

Table 5.1: Comparison of the Pearson correlation coefficients of various methods for the AB-bind S645 set. Except for those from present TopNetTree and TopGBT, the other results are adopted from Ref. [68].

Method	R_p
TopNetTree	0.65/0.68*
TopGBT	0.56
mCSM-AB	0.53/0.56*
TopCNN	0.53
Discovery Studio	0.45
mCSM-PPI	0.31
FoldX	0.34
STATIUM	0.32
DFIRE	0.31
bASA	0.22
dDFIRE	0.19
Rosetta	0.16



(a) (b)

Figure 5.1: (a)Scatter plot of TopNetTree prediction crossvalidation result on S645 (b)Scatter plot of TopNetTree prediction crossvalidation result on S645 exclude 27 non-binders

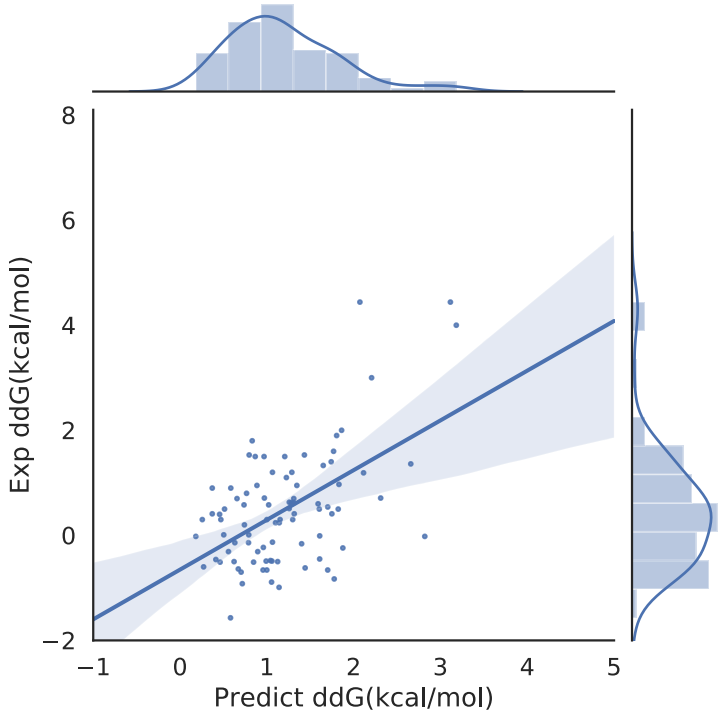


Figure 5.2: Scatter plot of TopNetTree blind test prediction on homology models

5.2.1.2 Protein level non overlapping test on the AB-bind 645 dataset

To further test the predicting power of our model, we applied protein level non overlapping test on the AB-bind dataset.

Table 5.2: Result of non-overlapping protein level test on AB-bind dataset, including Pearson correlation coefficient and RMSE in kcal/mol.

Name	Counts	R_p	RMSE(kcal/mol)
1AK4	16	0.528	0.837
1BJ1	19	0.103	1.502
1CZ8	19	0.506	1.077
1DQJ	21	0.568	1.877
1DVF	26	0.553	1.163
1FFW	9	-0.043	1.052
1JRH	2	1	0.812
1JTG	5	0.757	0.549
1KTZ/HM_1KTZ	44	0.866	0.496
1MHP	68	0.505	3.216
1MLC	11	0.397	1.508
1N8Z	34	0.626	2.273
1VFB	41	0.689	1.653
1YY9/HM_1YY9	21	-0.068	1.531
2JEL	43	0.818	0.954
2NYY/HM_2NYY	53	0.589	1.238
2NZ9/HM_2NZ9	35	0.665	1.367
3BDY	34	0.615	0.692
3BE1	34	0.474	0.941
3BN9/HM_3BN9	43	0.368	1.743
3HFM	22	0.262	2.69
3K2M	7	0.705	1.416
3NGB	11	0.459	1.147
3NPS	27	0.242	0.953
Total	645		
Average	27	0.508	1.362
Median	24	0.541	1.201

645 mutations in the dataset could be separated into 24 different protein-protein complexes (we merged the complex and its homology model as one category since they are very similar). To do the non overlapping test, all the mutations in one specific protein complex are split as the test set

and all other mutations are split as the training set. The result of non overlapping test of 24 protein complexes is shown in Table 5.2

5.2.1.3 Protein level leave-one-out validation test

To further test the predicting power of our model, we applied protein-level leave-one-out cross-validation test on the AB-bind dataset.

Table 5.3: Result of protein-level leave-one-out-validation test on AB-bind dataset, including Pearson correlation coefficient and RMSE in kcal/mol.

Name	Counts	R_p	RMSE(kcal/mol)
1AK4	16	0.139	0.977
1BJ1	19	0.392	1.350
1CZ8	19	0.671	0.921
1DQJ	21	0.318	1.885
1DVF	26	-0.103	1.345
1FFW	9	0.149	0.512
1JRH	2	-1	0.13
1JTG	5	-0.998	0.689
1KTZ/HM_1KTZ	44	0.973	0.222
1MHP	68	0.145	3.432
1MLC	11	0.606	0.418
1N8Z	34	0.029	3.023
1VFB	41	0.533	1.755
1YY9/HM_1YY9	21	0.547	0.225
2JEL	43	0.707	0.968
2NYY/HM_2NYY	53	0.514	1.175
2NZ9/HM_2NZ9	35	0.121	1.732
3BDY	34	0.604	0.548
3BE1	34	0.054	1.077
3BN9/HM_3BN9	43	0.281	1.938
3HFM	22	-0.121	2.726
3K2M	7	-0.993	1.234
3NGB	11	0.411	1.186
3NPS	27	0.099	0.766
Total	645		
Average	27	0.170	1.218
Median	24	0.215	1.027

645 mutations in the dataset could be separated into 24 different protein-protein complexes (we merged the complex and its homology model as one category since they are very similar). Then each protein complex is treated individually as a set to do the leave one out cross-validation test. The result of test of 24 protein complexes is shown in Table 5.3

For this test, our model reached an average/median $R_p's$ of 0.170/0.215, which are significantly lower than the 10-fold cross-validation result over the entire dataset. One possible reason for this behavior is that the training set for each complex is too small with only an average of 27 samples per complex. Also, this result implies that our model needs a diversity of training samples to achieve stable and consistent prediction quality.

5.2.2 Prediction result on SKEMPI dataset

Table 5.4 shows the Pearson correlation coefficients on 10-fold cross-validations. It is found that the proposed TopNetTree is about 15% more accurate than the best existing method.

Table 5.4: Comparison of the Pearson correlation coefficients of various methods for the single point mutation in SKEMPI dataset of 1131 mutations. Except for those from TopNetTree and SAAMBE, the other results are adopted from Ref. [91].

Method	R_p
TopNetTree	0.850
BindProfX	0.738
Profile-score+FoldX	0.738
Profile-score	0.675
SAAMBE [66]	0.624
FoldX	0.457
BeAtMuSic	0.272
Dcomplex	0.056

Scatter plot of the S1131 crossvalidation result is shown in Fig.5.3. TopNetTree model was able to achieve the R_p of 0.85 and RMSE of 1.55 kcal/mol.

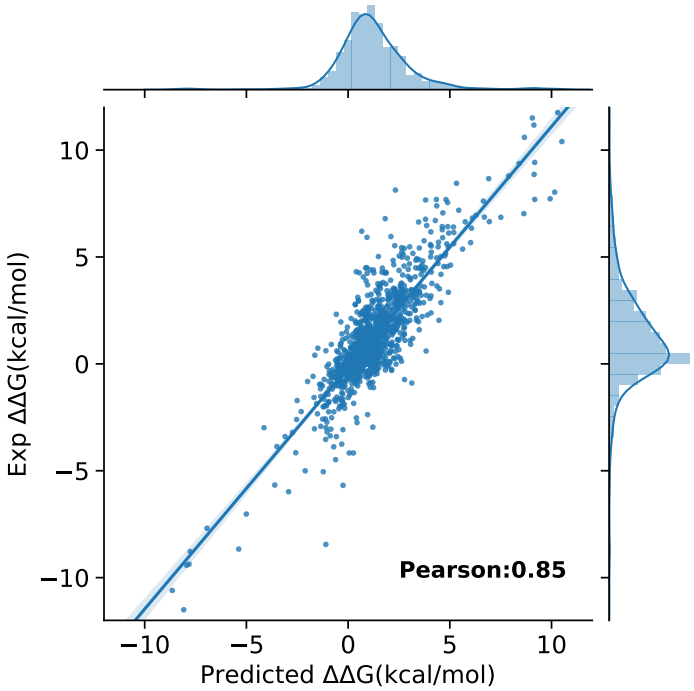
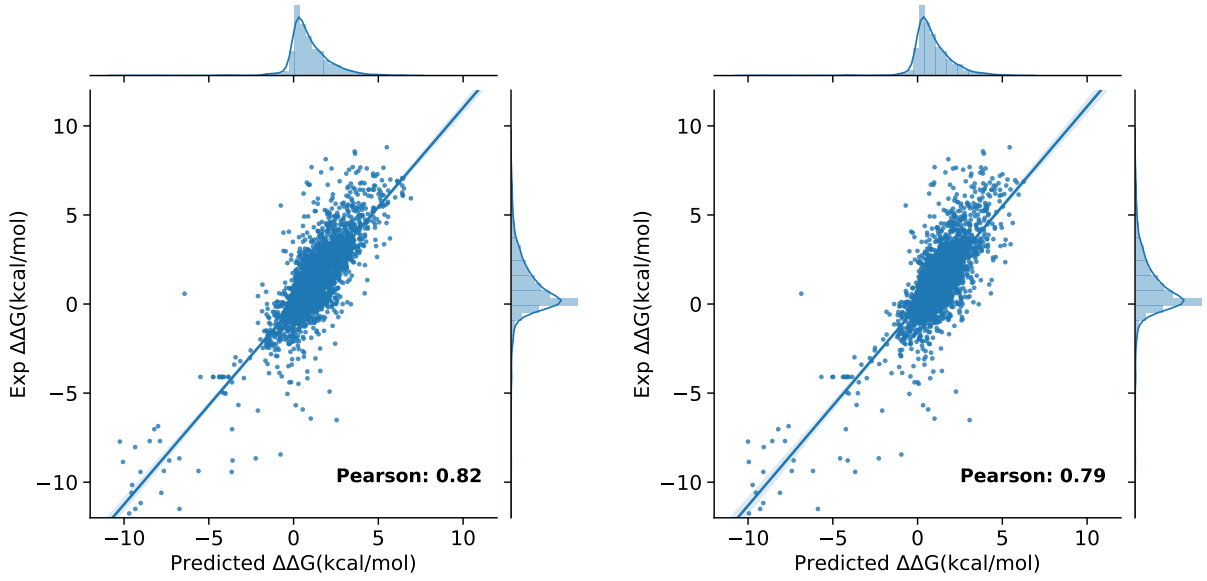


Figure 5.3: Performance evaluation on the 10-fold cross-validation on set S1131.

5.2.3 Prediction result on SKEMPI 2.0 dataset

For SKEMPI 2.0 dataset, we tested our model on S4947 , S4169 and S8338 datasets. For set S4947, we carry out the regular 10-fold cross-validation 10 times. For S4169 and S8338 sets, we follow the 10-fold stratified cross-validation used in mCSM-PPI2 paper. [72].

TopNetTree model achieved following crossvailidation results: Fig.5.4 shows the crossvalidation result of et S4947 with the R_p of 0.82 and RMSE of 1.11 kcal/mol and Set S4168 with the R_p of 0.78 and RMSE of 1.13 kcal/mol. Fig.5.5 shows the crossvalidation result of Set S8338 with R_p of 0.85 and RMSE of 1.09 kcal/mol.



(a) (b)

Figure 5.4: (a)Scatter plot of TopNetTree prediction crossvalidation result on S4947 (b)Scatter plot of TopNetTree prediction crossvalidation result on S4169

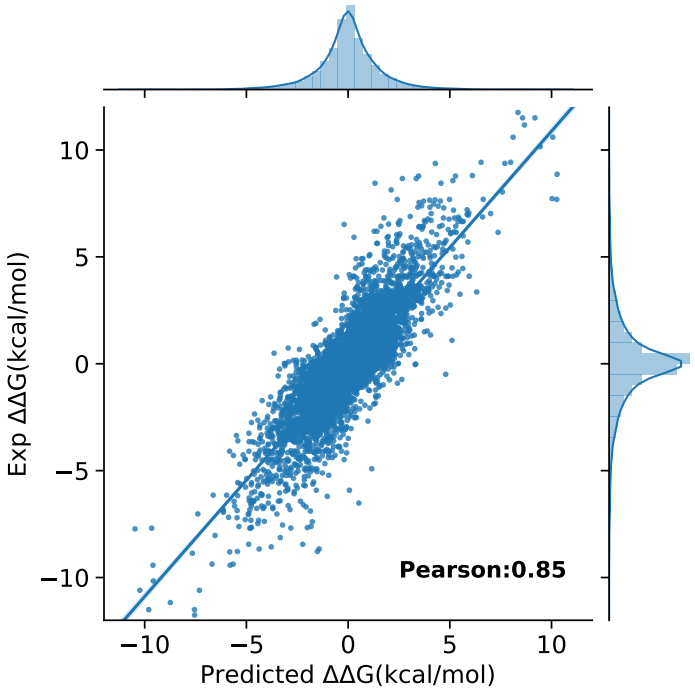
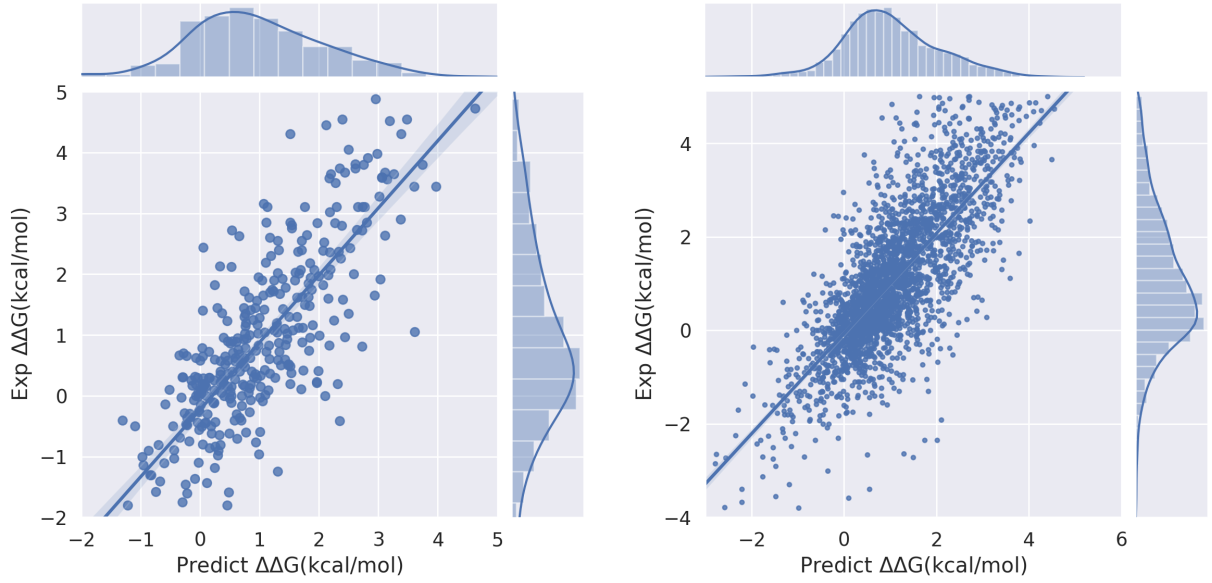


Figure 5.5: Scatter plot of TopNetTree prediction crossvalidation result on S8338

5.3 Performance of LTP model on ProTherm protein mutation database S2648 and S350

5.3.1 General performance

Following Fig5.6 shows the prediction result for S350 and 5-fold crossvalidation result for S2648. Our LTP model had prediction result for S350, with $R_p = 0.79$ and $RMSE = 0.94$ and (b) crossvalidation result of S2648, with $R_p = 0.78$ and $RMSE = 0.92$



(a) (b)

Figure 5.6: (a)Scatter plot of prediction result on S350 (b) Scatter plot of 5-fold crossvalidation result on S2648

We also compared our result with other methods in table 5.5. Our method reached R_p of 0.80 and RMSE of 0.94 kcal/mol in S350 set and R_p of 0.78 and RMSE of 0.92 kcal/mol in S2648 set, which is among the top of the state-of-art methods.

Table 5.5: Prediction results of S350 and 5-fold cross validation results of S2648. All the result of other methods listed in the table are from the paper cited.

Method	S350			s2648		
	n	R_p	RMSE	n	R_p	RMSE
LTP	350	0.80	0.94	2648	0.78	0.92
STRUM	350	0.79	0.98	2647	0.77	0.94
mCSM	350	0.73	1.08	2643	0.69	1.07
INPS	350	0.68	1.25	2648	0.56	1.26
PoPMuSiC 2.0	350	0.67	1.16	2647	0.61	1.17
PoPMuSic 1.0	350	0.62	1.23	-	-	-
I-Mutant 3.0	338	0.53	1.35	2636	0.60	1.19
Dmutant	350	0.48	1.28	-	-	-
Automute	315	0.46	1.42	-	-	-
CUPSAT	346	0.37	1.46	-	-	-
Eris	324	0.35	1.49	-	-	-
I-Mutant 2.0	346	0.29	1.50	-	-	-

5.3.2 Inter/Intra-protein-level crossvalidation

To validate our model with respect to types of protein, we did inter and intra protein level cross validation on S2648 dataset. To perform inter-protein-level cross-validation for each protein, the samples in one protein are taken as the test set while the rest of the dataset is used as the training set. For inter protein level cross -validation, all the mutations within one protein are selected as the test set while the whole set exclude test set are chosen as training set. For intra protein level cross-validation, all the proteins with more than 20 mutations are indivisually selected and leave-one-out cross validated. Considering too small sample size will lead to extreme overfitting, those proteins with less than 20 mutations are not included in the intra protein level cross-validation test.

For inter-protein-level cross-validation, we get the result of $R_p = 0.55$, $RMSE = 1.23$ kcal/mol (R_p and RMSE are evaluated for combining all inter-protein-level test result as the test result for whole set). For intra-protein-level cross-validation, the results are shown as table 5.6

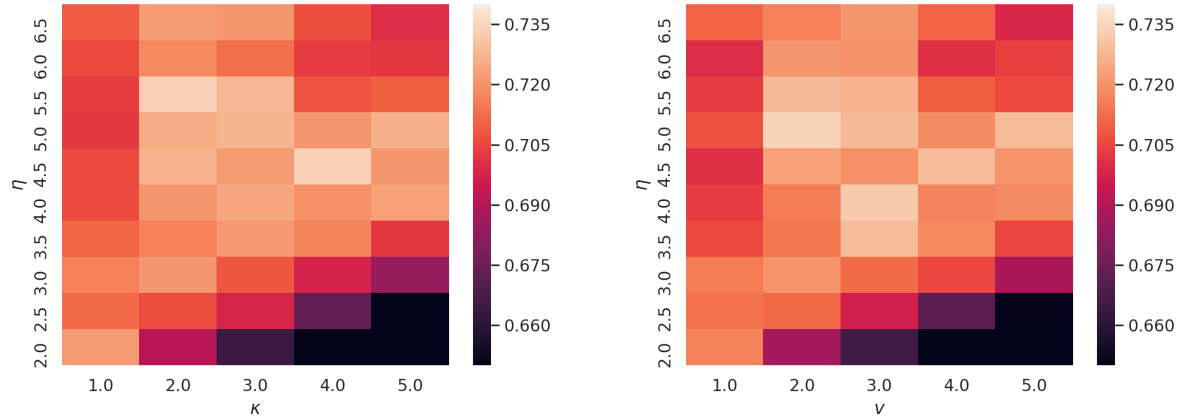
Table 5.6: Intra-protein-level cross-validation result of S2648. Mutations in each protein are 5-fold cross-validated inside protein.

Pdbid	Sample size	R_p	RMSE	Pdbid	Sample size	R_p	RMSE
1aj3	63	0.781	0.752	1aps	21	-0.154	1.095
1bni	153	0.645	0.962	1bvc	41	0.512	0.662
1c9o	25	0.03	0.581	1csp	21	0.724	0.893
1cun	26	0.726	0.851	1e65	21	0.558	1.073
1ey0	482	0.728	0.937	1fkj	36	0.898	0.639
1fna	34	0.067	1.413	1ftg	27	0.764	0.867
1h7m	26	-0.132	0.62	1hfz	22	0.08	1.465
1hmk	25	0.789	0.993	1lni	43	0.795	1.088
1lz1	110	0.659	0.954	1qlp	43	0.386	1.125
1rn1	31	0.867	0.983	1rop	20	0.562	0.931
1rtb	37	0.673	1.124	1sak	28	0.126	0.81
1shf	36	0.168	1.074	1ten	28	0.5	1.284
1uzc	45	0.492	0.779	1vqb	90	0.693	0.805
1wq5	44	0.703	1.329	1yyj	37	0.321	1.295
2abd	29	0.231	1.264	2ci2	75	0.707	0.744
2lzm	98	0.673	1.178	2rn2	70	0.62	1.101
4lyz	57	0.393	1.359	5dfr	102	0.709	0.687
5pti	21	0.364	1.307				
Sum	2067	N/a	N/a	Mean	59	0.505	1.001
Median	36	0.62	0.983				

Comparing with the 5-fold crossvalidation result of S2648 of $R_p = 0.78$ (each fold is randomly selected), R_p of inter-protein-level cross-validation is much lower. The reason is machine learning method is highly depend on relations between training and testing data. For inter-protein-level cross-validation, testing set protein is not included in training set thus it's hard to predict the energy change. For intra-protein-level cross-validation set, proteins with sample size larger than 100 (1ey0, $R_p = 0.728$; 1bni, $R_p = 0.645$; 1lz1, $R_p = 0.659$; 5dfr, $R_p = 0.687$) get higher R_p than median and mean value for all proteins. For some proteins, R_p is very low but RMSE result is very good, for example, 1c9o and 1h7m. The reason is that for those proteins, all energy change upon mutations are very close to 0 kcal/mol, our model is able to predict the energy change value but hard to track the ralative small perterbations of energy change between similar mutations.

5.3.3 Prediction result for different density kernel

According to the property of exponential kernel and Lorentz kernel, with a large η and κ/v, the decay rate will be higher for a single atom probe. Since we only include $\{C, N, O, S\}$ in our calculation, and those four atom types have very close VdW radius, same characteristic distance η is used in our model. In Fig.5.7, we used grid search on η and κ/v to get the best performance on prediction model. In Fig.5.7, (a) Density kernel $\Phi(r_{ij};\eta) = e^{-(r_{ij}/\eta)^{\kappa}}, \kappa > 0$, at $\eta = 5.5, \kappa = 2.0$, the model reaches its largest $R_p = 0.731$ (b) Density kernel $\Phi(r_{ij};\eta) = \frac{1}{1+(r_{ij}/\eta)^{v}}, v > 0$,at $\eta = 5.5, v = 2.0$,the model reaches its best $R_p = 0.728$



(a) (b)

Figure 5.7: Parameter selection heatmap in prediction model of S350.

As we can see from the graph, the two kernels have very similar pattern for the parameter choosing. In general, model has a relatively high performance at η between 4.0 Å and 5.5 Å, consider the VdW radius of C is 1.7 Å, the best choice for characteristic distance is around 2.4 to 3.2 times of the VdW radius.

CHAPTER 6

DISCUSSION

6.1 Prediction result analysis for different mutation type

The pattern of PPI binding affinity changes and protein folding energy change over different mutation types is important for protein design. In this section, We test how well can the model prediction resemble the distribution in experimental data. A reverse mutation from "B" to "A" is considered as the same mutation type as from "A" to "B" and the associated energy change admits an opposite sign.

6.1.1 Analysis of TopNetTree prediction result on S645

Overall, our predicted patterns are remarkably similar to those of experimental data in terms of both average binding energy changes and variance of binding energy changes as shown in Fig. 6.1. It is interesting to note that all the mutations to Alanine have positive energy change. A possible reason is that mutations from a large residue to a small one could lead to a stabilizing effect to the whole system. Besides the size of amino acids, we also categorized amino acids into charged, polar, hydrophobic and special case groups. In terms of binding affinity changes, we find that most mutations from polar to hydrophobic residues have a positive free energy change (for example, S to M), which means mutations from polar residues to hydrophobic residues would make the whole PPI system more stable. We also observed that a mutation from charged residues to uncharged polar residues could lead to a negative energy change, for example, Lysine to Serine (K to S), which means such mutations might have broken some charge-charge interaction pairs.

Although our model shares a similar pattern in the variance of energy changes with experimental data, the variance of the model predictions is generally lower than the experimental data as shown in Fig. 6.1. It remains a challenging task to come up with predictions with a diversity level the same as that of experimental data.

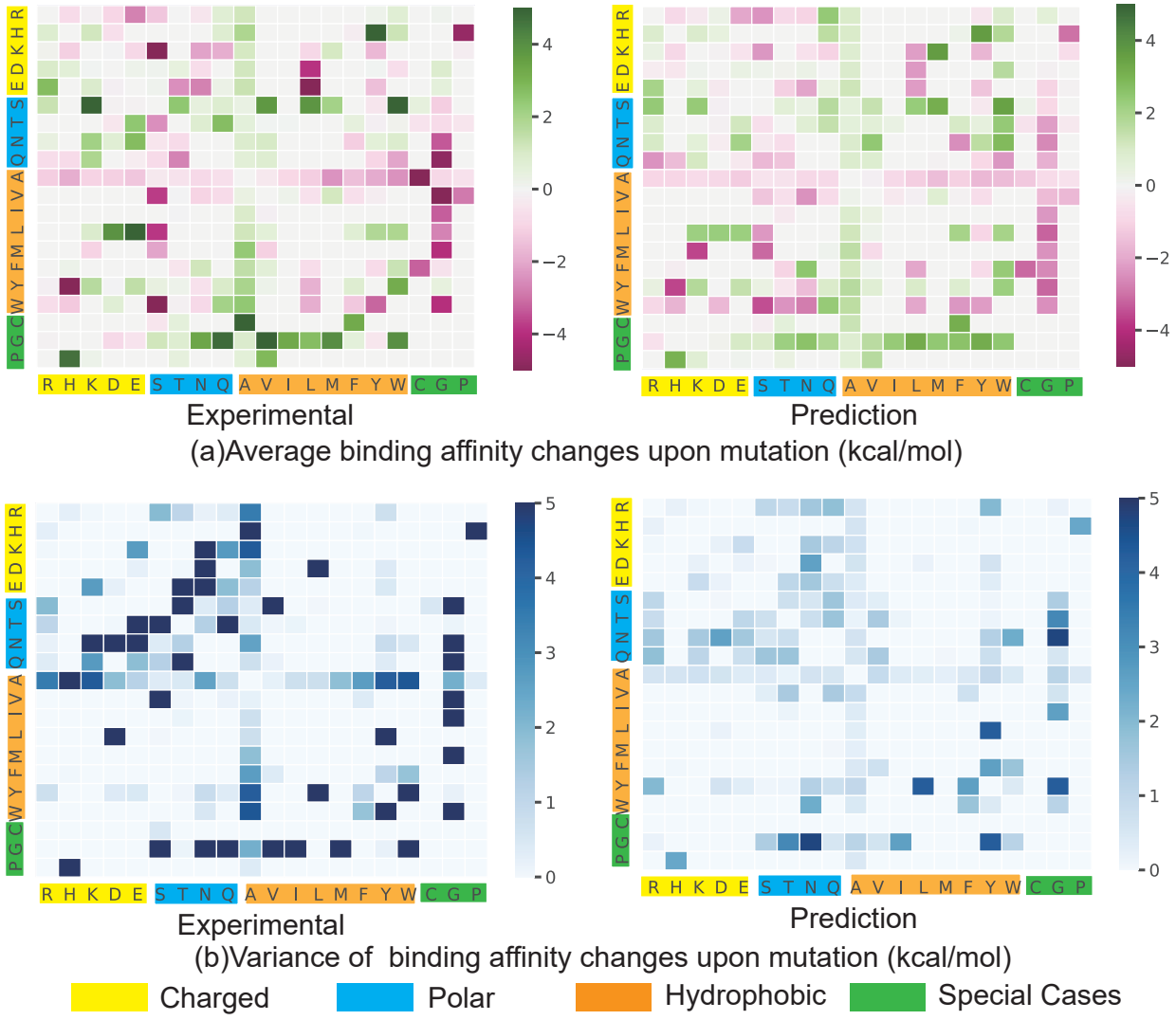


Figure 6.1: Comparison of average experimental and prediction binding affinity changes upon mutation associated with different amino acid types for the AB-Bind dataset. The x-axis labels the residue type of the original, while the y-axis labels the residue type of the mutant. For a reverse mutation, its $\Delta\Delta G$ is taken the same magnitude as the original value with an opposite sign. **a** Average binding affinity changes upon mutation (kcal/mol) **b** Variance of binding affinity changes upon mutation (kcal/mol)

6.1.2 Analysis of LTP prediction result on S2648

The pattern of protein folding energy changes over different mutation types is important for protein design.

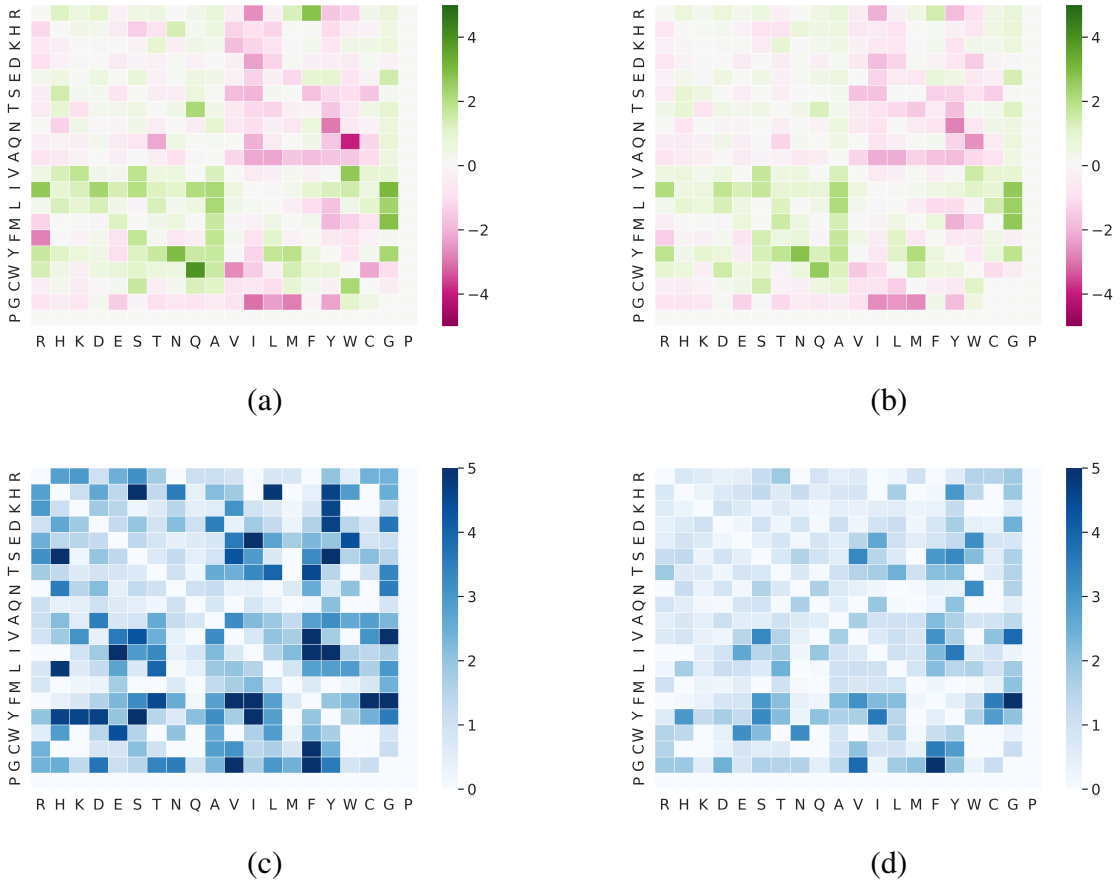


Figure 6.2: Comparison of average experimental (a) and prediction binding (b) affinity changes upon mutation associated with different amino acid types for S2648 dataset.

Overall, our predicted patterns are remarkably similar to those of experimental data in terms of both average binding energy changes and variance of binding energy changes as shown in Fig. 6.2. It is interesting to note that all the mutations to Alanine have positive energy change. A possible reason is that mutations from a large residue to a small one could lead to a stabilizing effect to the whole system. Besides the size of amino acids, we also categorized amino acids into charged, polar, hydrophobic and special case groups. In terms of binding affinity changes, we find that most mutations from polar to hydrophobic residues have a positive free energy change (for example, S

to M), which means mutations from polar residues to hydrophobic residues would make the whole PPI system more stable. We also observed that a mutation from charged residues to uncharged polar residues could lead to a negative energy change, for example, Lysine to Serine (K to S), which means such mutations might have broken some charge-charge interaction pairs.

6.2 Prediction result analysis for different mutation regions

6.2.1 Definition of mutation region

Mutant residue locations were classified into interface and non-interface regions. Interface residues were further classified as the rim, support and core and non-interface residues were also further classified as surface and interior, based on the classification approach by Levy [55].

Residue classification is mainly based on the change of relative residue accessible surface area (rASA) between protein-protein complex ($rASA_c$) and individual protein components of complex ($rASA_m$), as shown in Table 6.1. ASA was calculated with AREAIMOL from the CCP4 suite [20] and relative solvent accessibility was obtained by normalizing the absolute value with that of the same amino acid in a G-X-G peptide. [59].Here, $\Delta rASA = rASA_m - rASA_c$.

Table 6.1: Criteria of residue regions [55]

Region	Δ rASA	$rASA_c$	$rASA_m$
Interior	0	$< 25\%$	
Surface	0	$> 25\%$	
Rim	> 0	> 25 %	
Support	> 0		$< 25\%$
Core	> 0	$< 25\%$	$> 25\%$

6.2.2 Analysis of TopNetTree prediction result on S645

The locations of the site mutations could be categorized into 5 different regions as interior, surface, rim, support, and core. The detailed definition can be found in method mutation region section. In experimental data, mutations at the core or support region have a higher average energy change

around 1.8 kcal/mol (1.72 kcal/mol and 1.91 kcal/mol respectively), while mutations at the rim or interior region have an average energy change around 0.8 kcal/mol (0.82 kcal/mol and 0.83 kcal/mol respectively) as shown in Fig. 6.3. On the other hand, the surface mutations have an average energy change less than 0.2 kcal/mol. Similar patterns regarding mutation sites and energy changes were reported in the literature [67]. A possible reason for these patterns is that different mutation regions vary in their accessibility to the water. In general, surface, interior, and rim regions are much more accessible to the water than the core and support regions.

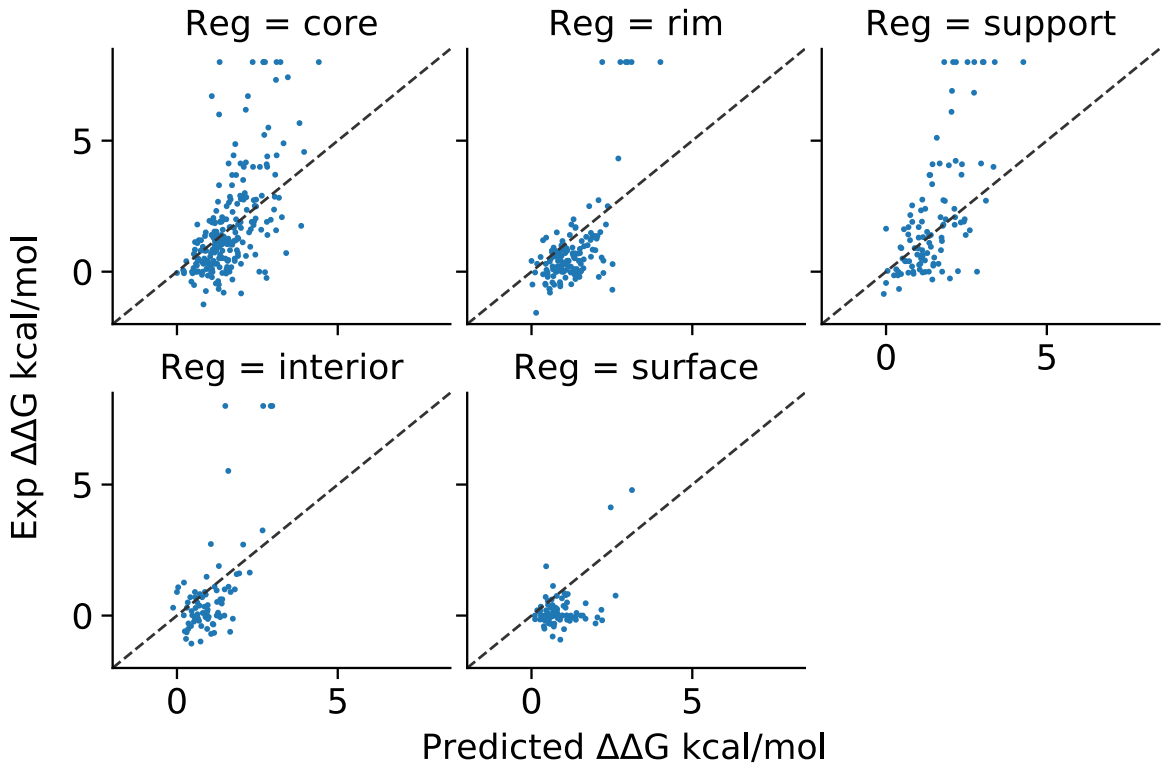


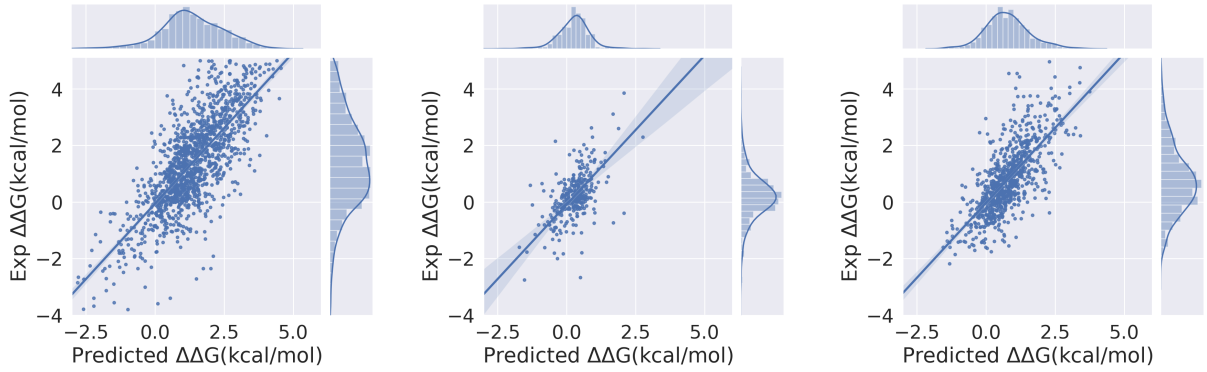
Figure 6.3: Prediction results for different residue region types in S645 dataset

Fig. 6.3 shows our predictions concerning different mutation regions. Average $R'_p s$ of 0.60, 0.66, 0.66, 0.65, and 0.48 were achieved for the core, rim, support, interior, and surface regions, respectively. This result shows that the performance is consistent among different mutation regions except for the surface region. We believe that the relative inferior performance for surface mutations is due to its small data size and that the energy disturbance caused by surface mutations is small on

average.

6.2.3 Analysis of LTP prediction result on S2648

Fig.6.4 shows the 5-fold crossvalidation result for S2648 set for different mutation region.



(a) (b) (c)

Figure 6.4: Scatter plot of 5-fold crossvalidation result for S2648 set in (a) Buried mutation region, with $Rp = 0.79$ and $RMSE = 0.94$,(b) Exposed mutation region, with $Rp = 0.78$ and $RMSE = 0.92$ (c) Intermediate mutation region, with $Rp = 0.78$ and $RMSE = 0.92$

From the scatter plot, we can see that for buried region, it is more likely to have larger folding energy change upon mutation, while exposed region has much smaller energy perterbation upon mutation. The reason is that exposed region residues are more accessible to the water. Desipite the distribution of energy change, our model worked well on all three regions with simialr R_p value, which means our model is stable with respect to the residue regions and our model is able to predict energy change value for all three regions.

6.3 Alanine scanning test of 1AK4

In molecular biology, alanine scanning is a site-directed mutagenesis technique used to determine the contribution of a specific residue to the stability or function of a given protein[62, 86, 82]. Alanine is used because of its non-bulky, chemically inert, methyl functional group that nevertheless mimics the secondary structure preferences that many of the other amino acids possess.

Alanine scanning test on 1AK4 chain A, using TopNetTree model with Ab-Bind training data. All the $\Delta\Delta G$ values are in kcal/mol. In total there are 165 residues in the chain A of 1AK4. Results of the alanine mutation are also separated into 5 region groups, interior, surface, rim, support, and core, respectively.

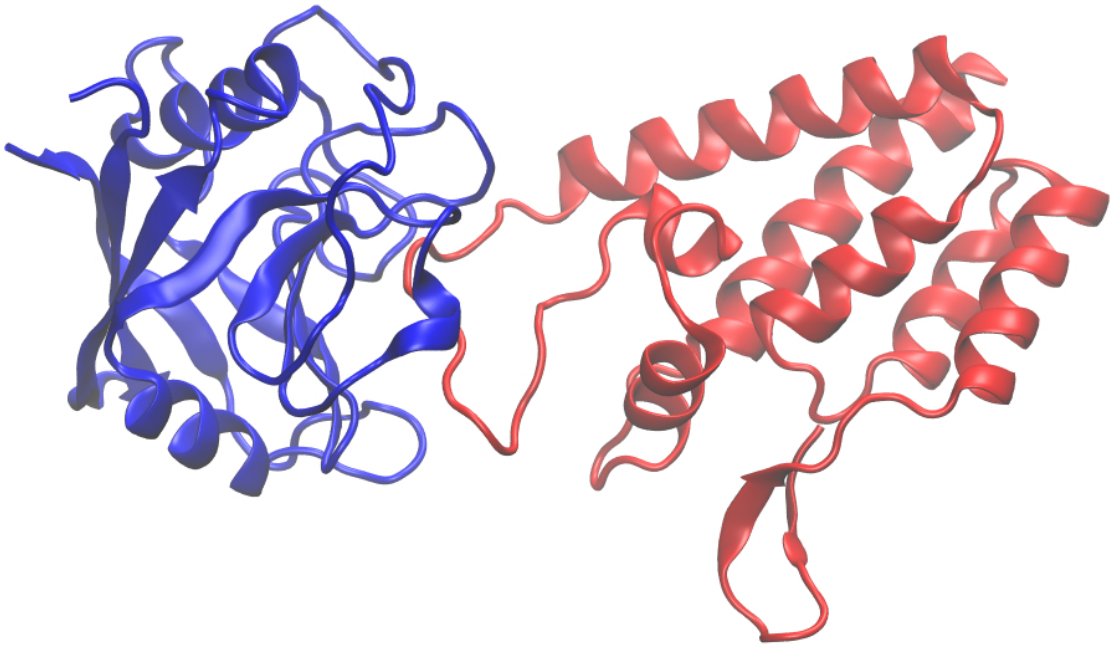


Figure 6.5: Structure of protein complex 1AK4, chain A in blue and chain D in red

Table 6.2: Alanine mutation test on 1AK4 chain A.

	Interior		Surface		Rim		Support		Core		All	
	Avg	Var	Avg	Var	Avg	Var	Avg	Var	Avg	Var	Avg	Var
Arg	0.8435	0	1.7463	0.5962	1.6302	0	–	–	1.5676	0	1.5466	0.4017
Asn	1.0064	0.1031	1.3790	1.2943	1.4870	0	1.6727	0	–	–	1.2581	0.5352
Asp	1.0726	0.1059	0.8942	0.0837	–	–	–	–	–	–	0.9707	0.1010
Cys	0.8236	0.0172	0.6246	0	–	–	–	–	–	–	0.7739	0.0203
Gln	–	–	0.9425	0	–	–	2.8500	0.0041	–	–	2.2142	0.8114
Glu	0.8466	0.0174	1.2533	0.3848	–	–	–	–	–	–	1.1794	0.3426
Gly	1.0956	0.4595	0.6091	0.1135	–	–	–	–	1.3921	0	0.9322	0.3761
His	1.2941	0	1.1008	0	–	–	1.7063	0.0410	–	–	1.4519	0.0899
Ile	1.0031	0.1280	0.1595	0	–	–	0.8719	0	–	–	0.9056	0.1658
Leu	1.2601	0.1447	1.8473	0.8087	–	–	1.5133	0	–	–	1.4641	0.3798
Lys	–	–	0.6113	0.1443	–	–	–	–	–	–	0.6113	0.1443
Met	2.2145	0.4070	0.9892	0	–	–	1.9440	0.2721	–	–	1.9153	0.4696
Phe	2.1307	0.9728	1.0778	0.0046	–	–	1.7958	0.2721	–	–	1.9457	0.8788
Pro	0.8306	0	0.7735	0.3486	–	–	–	–	–	–	0.7831	0.2909
Ser	1.0374	0.1150	0.3301	0.0010	–	–	–	–	–	–	0.8606	0.1803
Thr	1.1284	0.1641	0.8129	0.0432	1.1802	0	–	–	–	–	0.9898	0.1205
Trp	–	–	–	–	1.3124	0	–	–	–	–	1.3124	0
Tyr	2.5878	0	1.0924	0	–	–	–	–	–	–	1.8401	0.5590
Val	1.0002	0.0229	0.4613	0	–	–	–	–	–	–	0.9403	0.0490

CHAPTER 7

THESIS CONTRIBUTION

In this chapter, thesis contribution is highlighted for our prediction model on mutation induced affinity change of protein-protein interactions and protein folding, respectively.

7.1 Protein-Protein interactions energy change upon mutation

The importance of protein-protein interactions (PPIs) is evident from the intensive efforts to study them from many perspectives, including quantum mechanics, molecular mechanics, biochemistry, biophysics, and molecular biology. For example, the Pearson's correlation coefficients (R_p) between predicted $\Delta\Delta$G values and experimental data in cross-validations of a commonly used PPI database, AB-Bind [83], is only 0.53.

Recently, topology has been shown to be surprisingly effective in simplifying biomolecular structural complexity [88, 37, 10]. It has been devised to win worldwide competitions in computer-aided drug design [63]. Therefore, it is of enormous importance to exploit topology for understanding PPIs. In this work, we propose topology-based network trees (TopNetTrees) for $\Delta\Delta$G predictions. Specifically, element-specific and site-specific persistent homology is introduced to characterize PPIs. Additionally, we propose machine learning algorithm, convolutional neural networks (CNNs) assisted gradient boosting trees (GBTs), to pair with the topological method for the prediction of PPI $\Delta\Delta$G. We demonstrate that the proposed TopNetTree achieves R_p of 0.65, which is about 22% better than the previous best result for the AB-Bind dataset. For another benchmark PPI dataset, SKEMPI, the present method significantly outperforms the state-of-the-art in the literature.

7.2 Protein folding energy change upon mutation

Protein folding is the process by which a polypeptide chain folds to become a biologically active protein in its native 3D structure. It is the fundamental basis for living organisms. Mutation

in protein folding process are related to several diseases[52]. Also, mutation can lead to drug resistance[58]. To support and assist the timely and costly mutagenesis experiment, computational model for protein folding energy change is strongly needed.

In this work, we proposed LTP prediction model for protein folding energy change upon mutation. To characterize the structure information, we used density function to generate local surface. Through calculation of eigenvalues of Hessian matrix, we got the local topological descriptor of the surface. Clustering the eigenvalues by element type and integrated with Gradient boosting decision model, we introduced our LTP model.We demonstrate that the proposed LTP model achieves R_p of 0.78 on S2648 dataset and R_p of 0.80 on S350 dataset, which outperforms the state-of the-art in the literature.

APPENDICES

APPENDIX A

SUPPLEMENTARY MATERIALS FOR TOPNETTREE MODEL

A.1 Code and Data availability

Code availability:

All source codes and models are publicly available through a Code Ocean compute capsule (https://doi.org/10.24433/CO.0537487.v1)

Data availability:

Copyright and credits of the databases used in the thesis all belong to their original authors. The databases are publicly available at the listed web pages

- **AB-Bind database**

  https://github.com/sarahsirin/AB-Bind-Database

- **SKEMPI database**

  https://life.bsc.es/pid/skempi2

- **Pro-Therm database**

  https://web.iitm.ac.in/bioinfo2/prothermdb/index.html

A.2 Cross validation result on AB-Bind S645 dataset

Table A.1: TopNetTree crossvalidation result on S645

PDBID	Mutation Site	Experiment value	Predicted value
1AK4	D:A488G	2.49	1.772247901
1AK4	D:A488V	0.6	1.70987115
1AK4	D:A492G	0.41	0.895272322
1AK4	D:A492V	0.19	1.097418158
1AK4	D:G489A	1.91	1.173092962
1AK4	D:G489V	2.86	1.587869628
1AK4	D:H487A	0.84	1.460316679
1AK4	D:H487Q	0.8	1.007496241
1AK4	D:H487R	1.48	1.388147881
1AK4	D:I491A	0.07	0.841468807
1AK4	D:I491V	-0.17	0.434347712
1AK4	D:P485A	0.92	0.513298461
1AK4	D:P490A	2	1.625125531
1AK4	D:P490V	2.86	2.903634847
1AK4	D:P493A	0.51	0.659453086
1AK4	D:V486A	0.82	0.817046
1BJ1	V:F17A	0	0.667371629
1BJ1	V:Y21A	0	0.386553596
1BJ1	W:E93A	0.82	2.052724853
1BJ1	W:G88A	2.76	1.824412366
1BJ1	W:G92A	3.69	2.265639133
1BJ1	W:H86A	0	0.495278556
1BJ1	W:H90A	0	0.55636088

Table A.1 (cont'd)

1BJ1	W:I80A	0.82	1.006377789
1BJ1	W:I83A	3.69	0.919007453
1BJ1	W:I91A	0.41	1.255807724
1BJ1	W:K48A	0.41	1.475578215
1BJ1	W:K84A	0.65	0.617726683
1BJ1	W:M81A	3.69	1.373023595
1BJ1	W:M94A	1.42	0.826574734
1BJ1	W:Q79A	0	1.782212133
1BJ1	W:Q87A	0	2.410079235
1BJ1	W:Q89A	1.75	4.259019508
1BJ1	W:R82A	3.69	1.520556303
1BJ1	W:Y45A	0.82	1.980918971
1CZ8	V:F17A	0	2.712193372
1CZ8	V:Y21A	0	0.593485188
1CZ8	W:E93A	1.15	1.266828729
1CZ8	W:G88A	2.67	1.354456702
1CZ8	W:G92A	4.1	2.891246561
1CZ8	W:H86A	0	0.877283637
1CZ8	W:H90A	0	0.702263438
1CZ8	W:I80A	0.95	1.187477651
1CZ8	W:I83A	1.3	1.14279147
1CZ8	W:I91A	1.06	1.525802887
1CZ8	W:K48A	0	0.733548424
1CZ8	W:K84A	1.36	1.004540774
1CZ8	W:M81A	4.1	1.518174577
1CZ8	W:M94A	1.91	0.969634812

Table A.1 (cont'd)

1CZ8	W:Q79A	0.65	1.184333944
1CZ8	W:Q87A	0	0.52369491
1CZ8	W:Q89A	1.06	2.106217314
1CZ8	W:R82A	0.82	1.989554698
1CZ8	W:Y45A	1.94	1.539212479
1DQJ	C:D101A	1.3	1.785713715
1DQJ	C:K96A	6.1	1.753633758
1DQJ	C:K97A	3.5	1.981746346
1DQJ	C:L75A	1.5	1.763301137
1DQJ	C:N93A	0.6	1.90652708
1DQJ	C:R21A	1.3	1.182521681
1DQJ	C:S100A	0.8	0.969499244
1DQJ	C:T89A	0.8	1.435309483
1DQJ	C:W62A	0.8	0.784356207
1DQJ	C:W63A	1.3	1.304424197
1DQJ	C:Y20A	3.3	1.677039544
1DQJ	H:D32A	2	1.583783786
1DQJ	H:W98A	4.9	3.350083739
1DQJ	H:Y33A	5.5	2.707997194
1DQJ	H:Y50A	6.9	2.536556738
1DQJ	H:Y53A	1.2	0.759781316
1DQJ	L:N31A	2	1.84136089
1DQJ	L:N32A	4.1	2.614977004
1DQJ	L:S91A	1.4	1.216659126
1DQJ	L:Y50A	2.7	1.43641669
1DQJ	L:Y96A	1.1	1.552117066

Table A.1 (cont'd)

1DVF	A:H30A	1.67	0.971417375
1DVF	A:S93A	1.17	1.761136416
1DVF	A:W92A	0.34	0.918307718
1DVF	A:Y32A	2.05	1.281131406
1DVF	A:Y49A	1.64	0.113538779
1DVF	A:Y49A	1.75	2.067569689
1DVF	A:Y50A	0.69	1.475205049
1DVF	B:D100A	2.82	3.20042784
1DVF	B:D54A	4.32	2.256333782
1DVF	B:D58A	1.62	0.512496378
1DVF	B:E98A	4.23	1.961528555
1DVF	B:N56A	1.17	1.553101451
1DVF	B:R99A	1.89	1.888709858
1DVF	B:T30A	0.92	1.292450289
1DVF	B:W52A	4.17	2.08684492
1DVF	B:Y101F	2.03	2.358112289
1DVF	B:Y32A	1.85	0.992016256
1DVF	C:Y49A	1.88	2.38382718
1DVF	D:D52A	1.7	1.241237441
1DVF	D:H33A	1.88	0.8326038
1DVF	D:I101A	2.71	2.05527356
1DVF	D:K30A	1.01	1.649048839
1DVF	D:N55A	1.88	0.450320114
1DVF	D:Q104A	1.64	2.146662618
1DVF	D:R106A	4.13	2.304954282
1DVF	D:Y102A	4.79	3.255549884

Table A.1 (cont'd)

1FFW	A:A90V	0.09	0.704165614
1FFW	A:D13K	0.05	0.575279005
1FFW	A:E117K	0.71	0.42333309
1FFW	A:E93K	0.82	1.249733698
1FFW	A:F111V	1.26	0.342520128
1FFW	A:T112I	0.56	0.350163226
1FFW	A:T87I	-0.32	0.906796345
1FFW	A:V108M	1.13	0.522969592
1FFW	A:Y106W	0.71	3.141303285
1JRH	I:E45Q	0.11	1.209955109
1JRH	I:T14V	-0.02	0.734714999
1JTG	A:K234A	0.82	0.76336782
1JTG	A:R243A	0.56	1.28977497
1JTG	A:S130A	0.5	1.100434164
1JTG	A:S235A	1.32	0.684225643
1JTG	B:D49A	1.98	3.260976281
1KTZ	B:D118A	0.9	1.574396523
1KTZ	B:D32A	1.4	1.183164012
1KTZ	B:D32N	1.7	1.537020647
1KTZ	B:E119A	1.4	0.923715298
1KTZ	B:E119Q	1.7	1.819594258
1KTZ	B:E55A	1.3	0.674391378
1KTZ	B:E75A	1	1.275212918
1KTZ	B:F110A	0.9	0.381708447
1KTZ	B:F30A	2.9	2.890359842
1KTZ	B:H79A	0.4	0.438193601

Table A.1 (cont'd)

1KTZ	B:I125A	0.5	0.805530864
1KTZ	B:I50A	1.9	2.507838141
1KTZ	B:I53A	1.2	1.10341562
1KTZ	B:L27A	1.9	1.452828559
1KTZ	B:M112A	0.8	0.771273735
1KTZ	B:N47A	0.3	0.408989806
1KTZ	B:S49A	0.5	0.488502427
1KTZ	B:S52A	0.4	0.400807647
1KTZ	B:S52L	3.7	2.998590038
1KTZ	B:T51A	1.3	1.32474016
1KTZ	B:V62A	0.7	0.862856594
1KTZ	B:V77A	0.5	0.729256061
1MHP	H:F99W	0	1.511132256
1MHP	H:F99Y	0.76	2.600869319
1MHP	H:G100F	0.57	2.375511513
1MHP	H:G100I	8	3.232489423
1MHP	H:G100L	2.73	2.803968286
1MHP	H:G100M	0.1	1.978972993
1MHP	H:G100S	0.56	0.277025289
1MHP	H:G100V	8	2.588756997
1MHP	H:G102S	8	3.075302826
1MHP	H:G53A	0.51	1.085337422
1MHP	H:G53N	1.26	2.193903725
1MHP	H:G53Q	8	2.859680867
1MHP	H:G53S	0.68	0.29963622
1MHP	H:G53W	8	2.610651222

Table A.1 (cont'd)

1MHP	H:G54I	1.36	2.053849691
1MHP	H:G54M	8	3.263506111
1MHP	H:G54N	8	3.956349362
1MHP	H:G54T	1.17	1.083141006
1MHP	H:G54Y	0	1.361053347
1MHP	H:H56Y	8	2.793931876
1MHP	H:K64D	-0.12	0.758579301
1MHP	H:K64E	-0.16	2.852913642
1MHP	H:K64N	-0.07	1.004445379
1MHP	H:K64Q	-0.16	0.752612944
1MHP	H:L60D	-0.12	1.762583453
1MHP	H:R31Q	1.31	1.856987057
1MHP	H:S35A	1.58	1.399227777
1MHP	H:S35Q	2.73	1.844218833
1MHP	H:S35V	8	0.902972107
1MHP	H:S52M	2.08	1.507612619
1MHP	H:S52T	8	2.160307508
1MHP	H:T33N	2.73	1.164517369
1MHP	H:T33Q	8	2.011373376
1MHP	H:T33V	0.43	1.269510301
1MHP	H:T50E	8	2.721531952
1MHP	H:T50Q	8	2.060513216
1MHP	H:T50V	-0.26	3.066232312
1MHP	H:Y58E	2.08	1.589362497
1MHP	H:Y58Q	2.08	3.235696527
1MHP	H:Y58W	0.82	1.681574377

Table A.1 (cont'd)

1MHP	H:Y59E	0.54	3.81896283
1MHP	L:G92Q	1.58	2.356168725
1MHP	L:G92S	0.51	1.226035214
1MHP	L:H31K	1.17	1.317764394
1MHP	L:H31R	1.12	1.492543745
1MHP	L:H31W	1.26	1.481678655
1MHP	L:L49F	0.71	1.007443222
1MHP	L:L49K	0.93	0.461418046
1MHP	L:L49W	1.91	1.939848971
1MHP	L:L49Y	8	4.188789486
1MHP	L:N30K	8	1.347084757
1MHP	L:N30V	1.05	1.760356356
1MHP	L:N30W	1.58	4.39788735
1MHP	L:N30Y	1.91	2.446100825
1MHP	L:N52D	0.5	1.226667242
1MHP	L:N52E	-0.18	1.516983952
1MHP	L:N52K	0.66	1.186057489
1MHP	L:N52R	0.85	1.011625931
1MHP	L:N52Y	-0.1	0.685355368
1MHP	L:N93D	8	2.950172584
1MHP	L:S24R	-0.07	0.482326814
1MHP	L:S28Q	-0.67	0.514440248
1MHP	L:S91K	8	3.037964039
1MHP	L:S91Q	1.78	2.340801141
1MHP	L:S91R	2.73	2.673891363
1MHP	L:S91T	2.32	1.408651946

Table A.1 (cont'd)

1MHP	L:S91W	8	2.142880094
1MHP	L:W90Q	2.08	1.973060762
1MLC	H:K65D	0.02	0.772076764
1MLC	H:S57A	-0.38	0.238518223
1MLC	H:S57V	-0.49	1.67920641
1MLC	H:T28D	-0.15	0.279988554
1MLC	H:T31A	0.45	1.389369554
1MLC	H:T31V	0.53	1.239429551
1MLC	H:T31W	0.13	1.895184435
1MLC	H:T58D	-0.56	0.666008019
1MLC	L:N32G	-0.85	-0.175578736
1MLC	L:N32Y	0	1.126775376
1MLC	L:N92A	-1.25	0.686745352
1N8Z	H:D31A	0.25	0.695942219
1N8Z	H:D98A	1.21	1.821430834
1N8Z	H:D98W	-0.69	2.672934996
1N8Z	H:F100A	1.17	1.554348682
1N8Z	H:K30A	0.61	0.499821357
1N8Z	H:N54A	-0.15	0.715626098
1N8Z	H:R50A	8	3.0454875
1N8Z	H:T32A	0.38	0.411223308
1N8Z	H:T53A	0.78	1.166937169
1N8Z	H:W95A	8	2.781804442
1N8Z	H:Y100aA	8	2.794566125
1N8Z	H:Y100aF	-0.05	0.54905435
1N8Z	H:Y100aF	0.82	1.266112137

Table A.1 (cont'd)

1N8Z	H:Y102V	0.22	0.776690057
1N8Z	H:Y33A	-0.09	0.453388764
1N8Z	H:Y52A	0.23	2.384646317
1N8Z	H:Y56A	0.87	1.728654937
1N8Z	L:D28N	-0.28	0.949429435
1N8Z	L:F53N	1.21	0.99378604
1N8Z	L:H91A	8	3.035039866
1N8Z	L:H91F	-0.43	0.113538779
1N8Z	L:H91F	0.03	0.422061501
1N8Z	L:N30A	1.12	1.712643518
1N8Z	L:N30S	0.06	0.31163856
1N8Z	L:R66G	0.22	0.787922453
1N8Z	L:S50A	-0.07	1.307035447
1N8Z	L:S52A	-0.31	0.711932707
1N8Z	L:T31A	0.8	0.624779485
1N8Z	L:T93A	0.82	0.832968299
1N8Z	L:T94A	-0.12	0.913693287
1N8Z	L:T94S	0.31	0.741801789
1N8Z	L:Y49A	1.05	1.83223489
1N8Z	L:Y92A	1.36	0.985871557
1N8Z	L:Y92F	-0.21	1.032859711
1VFB	C:D119A	1	0.960852994
1VFB	C:D18A	0.3	0.738494684
1VFB	C:I124A	1.2	0.874839065
1VFB	C:K116A	0.7	1.295609012
1VFB	C:L129A	0.2	0.553255228

Table A.1 (cont'd)

1VFB	C:N19A	0.3	0.665018923
1VFB	C:Q121A	2.9	2.145117407
1VFB	C:R125A	1.8	2.107791787
1VFB	C:S24A	0.8	0.888296621
1VFB	C:T118A	0.8	0.602773202
1VFB	C:V120A	0.9	0.806104381
1VFB	C:Y23A	0.4	0.589905404
1VFB	H:D100A	2.9	2.118798539
1VFB	H:D54A	1	0.767356465
1VFB	H:D58E	0.08	0.965146096
1VFB	H:D58N	-0.13	0.528588246
1VFB	H:G31A	0.3	0.733750556
1VFB	H:G31E	-0.51	0.717865289
1VFB	H:G31W	0.01	0.481503215
1VFB	H:R99W	0.71	0.85370306
1VFB	H:R99Y	1.26	1.889616403
1VFB	H:S28D	0	0.151328361
1VFB	H:S28E	-0.1	0.756153385
1VFB	H:S28N	0.15	0.169814022
1VFB	H:S28Q	0.08	0.806274008
1VFB	H:W52A	0.9	1.635290525
1VFB	H:Y101A	4	2.32718193
1VFB	H:Y101F	1.6	2.709148226
1VFB	H:Y32A	1.1	1.395961726
1VFB	H:Y32E	1.91	1.696019146
1VFB	L:L46D	8	3.927971666

Table A.1 (cont'd)

1VFB	L:L46E	8	2.726775795
1VFB	L:N31W	0.17	0.330030168
1VFB	L:T52F	0.47	1.46582676
1VFB	L:T53R	1.67	1.930301498
1VFB	L:W92A	3.3	1.422575552
1VFB	L:Y32A	1.7	1.893113252
1VFB	L:Y32W	8	3.95789319
1VFB	L:Y50A	0.5	1.482213498
1VFB	L:Y50K	1.67	1.316597932
1VFB	L:Y50R	0.85	1.083387413
1YY9	H:N56A	-0.06	0.44529548
1YY9	H:T61E	-0.06	1.004986158
1YY9	L:N93A	-0.74	0.852318085
1YY9	L:S26D	-0.2	0.416451333
1YY9	L:T31E	-0.52	1.000581844
2JEL	P:A82S	0	0.62358458
2JEL	P:D69E	0.96	0.758253135
2JEL	P:E5D	0.41	0.793075738
2JEL	P:E5Q	0.72	0.568361032
2JEL	P:E66K	4.13	1.826121282
2JEL	P:E68A	0.41	1.095711209
2JEL	P:E70A	2.75	1.324838712
2JEL	P:E70K	4.13	3.142100614
2JEL	P:E75R	2.75	2.484719967
2JEL	P:E83A	0	0.015917697
2JEL	P:E85A	0	0.755415193

Table A.1 (cont'd)

2JEL	P:E85D	0	0.138530725
2JEL	P:E85K	0	0.664799549
2JEL	P:E85Q	0	0.477949894
2JEL	P:F2W	2.64	1.677778688
2JEL	P:F2Y	0	0.690321795
2JEL	P:H76A	-0.41	0.800214403
2JEL	P:H76D	-0.66	0.162703811
2JEL	P:K24E	0	0.076843167
2JEL	P:K27E	0	1.075453115
2JEL	P:K72E	0.41	1.015558596
2JEL	P:K72R	0	1.344040071
2JEL	P:K79E	0.41	0.941212343
2JEL	P:N12D	0	0.53221519
2JEL	P:N38T	0	0.650443812
2JEL	P:P11E	0	0.438658069
2JEL	P:Q3K	4.13	1.899204206
2JEL	P:Q4K	1.38	0.967431396
2JEL	P:Q57E	-0.41	0.586016362
2JEL	P:Q71E	2.75	2.228121794
2JEL	P:R17G	0	0.506077878
2JEL	P:R17K	0	-0.218487614
2JEL	P:S41C	1.51	1.753646505
2JEL	P:S43C	0	0.251503369
2JEL	P:S46C	0	0.262453297
2JEL	P:S64T	4.13	0.786654792
2JEL	P:T34Q	0	0.595213209

Table A.1 (cont'd)

2JEL	P:T36Q	0.41	0.81999459
2JEL	P:T62A	0	0.702519448
2JEL	P:T62N	0	0.738162379
2JEL	P:T7N	0.41	0.654010822
2JEL	P:T7S	0	0.372232377
2JEL	P:V6F	0	0.445051654
2NYY	A:D1058A	0.01	0.525443185
2NYY	A:D1062A	2.53	0.532715101
2NYY	A:E920A	2.84	2.000718398
2NYY	A:F917A	0.42	0.794587442
2NYY	A:F953A	4.06	1.684166884
2NYY	A:H1064A	7.32	3.824604807
2NYY	A:I956A	-0.01	0.870105035
2NYY	A:K1056A	-0.03	0.515474786
2NYY	A:K903A	0.29	0.74346297
2NYY	A:K923A	-0.14	0.076819464
2NYY	A:L919A	2.59	2.208152106
2NYY	A:N918A	0.89	1.056838524
2NYY	A:N954A	0.1	0.914624936
2NYY	A:Q915A	0.52	0.992770347
2NYY	A:R1061A	0.82	0.917340792
2NYY	A:R1294A	0.3	1.082284119
2NYY	A:S902A	-0.23	0.987631227
2NYY	A:S955A	0	0.653515584
2NYY	A:T1063A	1.62	2.436781712
2NYY	H:K30R	0.07	0.690428166

Table A.1 (cont'd)

2NYY	H:M34Q	-0.06	1.095716335
2NYY	H:Y31D	0.07	1.015319328
2NYY	H:Y31Q	0.13	1.439321457
2NYY	H:Y57Q	0.52	1.562704507
2NYY	L:D30Y	0.65	2.386377244
2NYY	L:H34R	0.1	1.081628988
2NYY	L:S28Q	0.42	1.037599326
2NYY	L:S31N	-0.36	1.198890994
2NZ9	A:D1058A	-0.03	0.018030161
2NZ9	A:D1062A	2.34	0.972425747
2NZ9	A:E920A	2.77	1.907371345
2NZ9	A:F917A	-0.05	1.499934326
2NZ9	A:F953A	3.34	1.477590993
2NZ9	A:H1064A	7.42	3.686355671
2NZ9	A:I956A	0.07	0.521508086
2NZ9	A:K1056A	-0.01	0.761719976
2NZ9	A:K903A	0.54	1.079718724
2NZ9	A:K923A	-0.3	1.195504875
2NZ9	A:L919A	2.28	1.756742478
2NZ9	A:N918A	2.16	0.707199903
2NZ9	A:N954A	-0.15	0.982384186
2NZ9	A:Q915A	0.1	0.957360586
2NZ9	A:R1061A	0.29	2.460852085
2NZ9	A:R1294A	0.39	0.270213282
2NZ9	A:S902A	-0.12	1.063609589
2NZ9	A:S955A	-0.08	1.241966479

Table A.1 (cont'd)

2NZ9	A:T1063A	2.37	2.895512163
3BDY	H:D31A	0.2	0.462999503
3BDY	H:D98A	0	0.816723653
3BDY	H:F100A	0.7	0.39798716
3BDY	H:G96A	0.2	0.038339515
3BDY	H:G97A	-0.1	0.21307577
3BDY	H:G99A	0.8	0.83948409
3BDY	H:K30A	0.2	0.589573872
3BDY	H:N54A	-0.2	0.955567274
3BDY	H:R50A	-0.3	0.932303598
3BDY	H:R58A	-0.2	0.390652519
3BDY	H:T32A	-0.4	0.222194885
3BDY	H:T53A	-0.5	0.304702079
3BDY	H:W95A	2	1.765017977
3BDY	H:Y100aA	0.5	0.41832798
3BDY	H:Y33A	0.3	1.363225032
3BDY	H:Y52A	0.4	0.066635255
3BDY	H:Y56A	0.2	0.693914709
3BDY	L:D27aA	0	0.942937671
3BDY	L:G31A	1.2	1.386320861
3BDY	L:G51A	1	1.093660591
3BDY	L:H91A	2	2.241317608
3BDY	L:I27bA	1.1	1.339230158
3BDY	L:I29A	0.8	0.509603382
3BDY	L:P27cA	0.2	0.688282228
3BDY	L:R27dA	-0.2	1.436376744

Table A.1 (cont'd)

3BDY	L:S28A	1	1.189582364
3BDY	L:S30A	1.3	1.257045566
3BDY	L:S52A	0.1	0.864860272
3BDY	L:T93A	-0.3	1.074481741
3BDY	L:T94A	0	1.795663852
3BDY	L:W50A	1.3	0.81352744
3BDY	L:Y32A	1.7	0.575994608
3BDY	L:Y53A	0.7	1.429535516
3BDY	L:Y92A	1.4	1.305772173
3BE1	H:D31A	0.4	0.780318088
3BE1	H:D98A	-0.1	0.052540047
3BE1	H:F100A	1.9	0.187278535
3BE1	H:G96A	0.1	-0.262728687
3BE1	H:G97A	0.3	1.12837483
3BE1	H:G99A	1.8	1.868551839
3BE1	H:K30A	-0.3	0.817743524
3BE1	H:N54A	-0.8	0.176916878
3BE1	H:R50A	2.4	2.146816267
3BE1	H:R58A	2.5	1.486981348
3BE1	H:T32A	-0.4	0.815500345
3BE1	H:T53A	0.1	1.232608627
3BE1	H:W95A	1.8	1.947953267
3BE1	H:Y100aA	1.2	0.830610968
3BE1	H:Y33A	2.4	1.186536328
3BE1	H:Y52A	-0.8	0.257574045
3BE1	H:Y56A	1.8	0.867268608

Table A.1 (cont'd)

3BE1	L:D27aA	0	1.018711282
3BE1	L:G31A	0.3	1.95817431
3BE1	L:G51A	-0.3	0.513404332
3BE1	L:H91A	0.9	0.482463821
3BE1	L:I27bA	-0.2	0.955205659
3BE1	L:I29A	0.9	1.265787158
3BE1	L:P27cA	0.1	0.00789498
3BE1	L:R27dA	-0.2	0.446215309
3BE1	L:S28A	0.3	0.287493294
3BE1	L:S30A	-0.2	0.236613237
3BE1	L:S52A	0.3	0.25247753
3BE1	L:T93A	-0.4	0.644443182
3BE1	L:T94A	1.4	1.194623934
3BE1	L:W50A	1.4	0.743452596
3BE1	L:Y32A	-0.8	1.369839294
3BE1	L:Y53A	0.9	1.107454542
3BE1	L:Y92A	0.5	0.415530772
3BN9	A:D217A	0.57	1.344612036
3BN9	A:D60aA	0.42	-0.013424513
3BN9	A:D60bA	0.31	0.696714558
3BN9	A:D96A	6.7	0.839577235
3BN9	A:E169A	0.37	0.927464902
3BN9	A:F60eA	-0.05	0.696510849
3BN9	A:F94A	0.64	0.796281704
3BN9	A:F97A	6.7	1.803046157
3BN9	A:H143A	0.09	1.326110308

Table A.1 (cont'd)

3BN9	A:I41A	0	0.503108154
3BN9	A:I60A	0.84	1.004853709
3BN9	A:K224A	0.79	0.452139457
3BN9	A:L153A	0.34	0.489134099
3BN9	A:N95A	0.77	0.466347123
3BN9	A:Q145A	0.13	0.342717139
3BN9	A:Q174A	-0.03	0.532123895
3BN9	A:Q175A	2.51	1.808248335
3BN9	A:Q221aA	0.71	1.03577532
3BN9	A:Q38A	-0.42	0.609958937
3BN9	A:R222A	-0.09	0.610519354
3BN9	A:R60cA	-0.05	0.220826913
3BN9	A:R60fA	-0.07	0.889075772
3BN9	A:R87A	-0.16	0.278420036
3BN9	A:T150A	0.29	0.335359473
3BN9	A:T98A	1.13	0.431271266
3BN9	A:Y146A	1.09	0.631743368
3BN9	A:Y60gA	0.02	0.183751899
3BN9	H:P100H	8	2.90160336
3BN9	H:Q100aV	1.36	2.01919693
3BN9	H:S30G	1.36	1.092683778
3BN9	H:S30N	1.36	1.176220347
3BN9	H:T28R	0.3	1.094147801
3BN9	H:T98Q	0	1.203092267
3BN9	H:T98R	-0.06	1.729186005
3BN9	H:Y99S	0.41	1.466513323

Table A.1 (cont'd)

3HFM	H:C95A	5.52	1.594987571
3HFM	H:C95F	3.25	3.115214161
3HFM	H:D32A	1.9	2.232648204
3HFM	H:D32N	0.17	1.615671996
3HFM	H:S31A	0.17	1.149026922
3HFM	H:Y33A	6	1.579874167
3HFM	H:Y50A	8	2.442873652
3HFM	L:N31A	5.22	2.510529204
3HFM	L:N31D	1.34	1.764424941
3HFM	L:N31E	5.67	3.673517204
3HFM	L:N32A	5.11	1.946142038
3HFM	L:Q53A	0.95	2.273443746
3HFM	L:Y50A	4.57	4.287597028
3HFM	L:Y50F	2.36	1.943871252
3HFM	L:Y50L	4.4	2.647111021
3HFM	L:Y96A	2.71	2.960946398
3HFM	L:Y96F	1.4	2.952138972
3HFM	Y:K96A	6.83	2.747755689
3HFM	Y:K97A	6.18	2.104647427
3HFM	Y:K97M	1.09	2.476272635
3HFM	Y:R21A	1.03	1.71014554
3HFM	Y:Y20A	4.87	2.155665707
3K2M	D:E52A	2.5	2.400196772
3K2M	D:M88A	4	2.177564766
3K2M	D:R38A	2.5	1.580952505
3K2M	D:W80A	4	2.293547317

Table A.1 (cont'd)

3K2M	D:Y35A	1	1.146032541
3K2M	D:Y36A	3.7	2.745885379
3K2M	D:Y87A	4	3.222661154
3NGB	H:A56G	-0.06	0.14552782
3NGB	H:G54S	0.73	1.536982886
3NGB	H:I30T	0.08	1.484114774
3NGB	H:K52N	0.18	1.200330023
3NGB	H:P62K	0.23	1.115225538
3NGB	H:R53N	0.05	1.148569033
3NGB	H:R61Q	0.1	0.591696346
3NGB	H:T33Y	0.08	0.832934832
3NGB	H:V57T	0.14	0.378418438
3NGB	H:V73T	0.16	0.850283401
3NGB	H:Y74S	0.33	0.54945593
3NPS	A:D214A	1.48	0.735888768
3NPS	A:D46A	0.34	0.751372318
3NPS	A:D47A	1.08	-0.005291115
3NPS	A:D91A	1.52	1.360870246
3NPS	A:E163A	0.62	0.466130221
3NPS	A:F50A	0.22	1.403497591
3NPS	A:F89A	1.61	1.66551116
3NPS	A:F92A	0.47	1.437897506
3NPS	A:H138A	1.89	0.907842784
3NPS	A:I26A	0.65	0.982534852
3NPS	A:I45A	-0.34	1.051965331
3NPS	A:K221A	-0.1	0.56344418

Table A.1 (cont'd)

3NPS	A:L147A	0.3	0.595928346
3NPS	A:N90A	0.26	0.804218843
3NPS	A:Q140A	0.3	0.05200386
3NPS	A:Q168A	-0.06	0.97202142
3NPS	A:Q169A	0.75	1.077402593
3NPS	A:Q218A	-0.04	0.783809807
3NPS	A:Q23A	0.03	1.317106432
3NPS	A:R219A	-0.08	0.631615871
3NPS	A:R48A	-1.07	0.423264568
3NPS	A:R51A	0.14	0.853977131
3NPS	A:R82A	-0.15	0.372059036
3NPS	A:T144A	0.18	0.238918702
3NPS	A:T93A	0.73	0.500676638
3NPS	A:Y141A	1.79	1.823759807
3NPS	A:Y52A	0.46	1.409161225
HM_1KTZ	B:D118A	0.8	0.857318869
HM_1KTZ	B:D32A	1.5	0.672650201
HM_1KTZ	B:D32N	2	1.365241973
HM_1KTZ	B:E119A	1.5	2.103855065
HM_1KTZ	B:E119Q	1.6	1.723233644
HM_1KTZ	B:E55A	1.2	0.545773096
HM_1KTZ	B:E75A	1.1	1.240564568
HM_1KTZ	B:F110A	0.9	0.113538779
HM_1KTZ	B:F30A	3	2.395796743
HM_1KTZ	B:H79A	0.3	-0.146093759
HM_1KTZ	B:I125A	0.5	0.596465756

Table A.1 (cont'd)

HM_1KTZ	B:I50A	1.9	2.295236759
HM_1KTZ	B:I53A	1.4	2.866933899
HM_1KTZ	B:L27A	1.8	0.888414405
HM_1KTZ	B:M112A	0.9	0.676475044
HM_1KTZ	B:N47A	0.3	1.074625229
HM_1KTZ	B:S49A	0.3	0.270350336
HM_1KTZ	B:S52A	0.2	0.912370354
HM_1KTZ	B:S52L	4	3.351510141
HM_1KTZ	B:T51A	1.5	1.216283828
HM_1KTZ	B:V62A	0.7	0.515725847
HM_1KTZ	B:V77A	0.4	0.407517511
HM_1YY9	H:A98W	-0.23	0.418109098
HM_1YY9	H:G33D	-0.89	0.05958884
HM_1YY9	H:I51G	-0.51	0.044470256
HM_1YY9	H:R97D	-0.99	0.442808427
HM_1YY9	H:T100Y	-0.45	0.975050876
HM_1YY9	H:T57G	-0.46	0.074634377
HM_1YY9	H:T57P	-0.48	0.667425841
HM_1YY9	H:V50L	-0.6	-0.330566734
HM_1YY9	H:V50Q	-0.64	0.330782419
HM_1YY9	H:Y32R	-0.13	0.968267084
HM_1YY9	L:A25V	-0.62	1.606320727
HM_1YY9	L:G30Y	-0.92	0.771856501
HM_1YY9	L:Q27Y	-0.83	1.455638579
HM_1YY9	L:T97C	-0.51	0.320064
HM_1YY9	L:T97D	-0.66	0.640174187

Table A.1 (cont'd)

HM_1YY9	L:T97S	-0.7	0.865009006
HM_2NYY	A:D1062A	-0.14	0.351028066
HM_2NYY	A:D902A	-0.49	0.575859888
HM_2NYY	A:E920A	0.5	2.064501081
HM_2NYY	A:F953A	1.53	1.209780382
HM_2NYY	A:I917A	0.95	0.788091299
HM_2NYY	A:K915A	-0.31	0.450918691
HM_2NYY	A:K955A	-0.5	0.152298061
HM_2NYY	A:L919A	0.41	1.443340344
HM_2NYY	A:N918A	0.3	0.816280603
HM_2NYY	A:N957A	0.01	0.287045341
HM_2NYY	A:P1063A	0.24	0.619803278
HM_2NYY	A:R1061A	0.58	0.541139471
HM_2NYY	A:R1064A	4.44	3.124877033
HM_2NYY	A:R903A	-0.02	0.354346073
HM_2NYY	A:S954A	-0.66	0.027106388
HM_2NYY	A:T923A	0.62	1.054733231
HM_2NYY	H:K30R	0.51	0.565871848
HM_2NYY	H:M34Q	-0.01	1.20044627
HM_2NYY	H:Y31D	0.97	1.443423688
HM_2NYY	H:Y31Q	1.33	1.995269269
HM_2NYY	H:Y57Q	1.19	1.562775376
HM_2NYY	L:D30Y	0.6	0.980379677
HM_2NYY	L:H34R	0.63	1.12922711
HM_2NYY	L:S28Q	0.7	1.168373331
HM_2NYY	L:S31N	1.2	1.293459172

Table A.1 (cont'd)

HM_2NZ9	A:D1062A	-0.14	0.11737132
HM_2NZ9	A:D902A	-0.49	-0.241554685
HM_2NZ9	A:E920A	0.5	1.035459785
HM_2NZ9	A:F953A	1.53	1.031164807
HM_2NZ9	A:I917A	0.95	1.372849249
HM_2NZ9	A:K915A	-0.31	0.162128377
HM_2NZ9	A:K955A	-0.5	0.619860501
HM_2NZ9	A:L919A	0.41	0.577141695
HM_2NZ9	A:N918A	0.3	1.316139699
HM_2NZ9	A:N957A	0.01	0.006647221
HM_2NZ9	A:P1063A	0.24	0.187907852
HM_2NZ9	A:R1061A	0.58	0.695574023
HM_2NZ9	A:R1064A	4.44	1.932082274
HM_2NZ9	A:R903A	-0.02	2.923683823
HM_2NZ9	A:S954A	-0.66	0.852797077
HM_2NZ9	A:T923A	0.62	1.028792168
HM_3BN9	H:P100H	1.36	2.322189072
HM_3BN9	H:Q100aV	0.41	0.43859132
HM_3BN9	H:S30G	0.71	0.850440861
HM_3BN9	H:S30N	0.71	2.00668105
HM_3BN9	H:T28R	-0.24	2.270706962
HM_3BN9	H:T98Q	-0.16	1.495593848
HM_3BN9	H:T98R	-1.57	0.343438481
HM_3BN9	H:Y99S	0.54	1.790362979

APPENDIX B

SUPPLEMENTARY MATERIALS FOR LTP MODEL

B.1 Supplementary result on S2648 and S350 dataset

For quantitative prediction of protein folding energy change upon mutation, we first apply our LTP1 and LTP2 methods to S2648 and S350 datasets. S350 set is tested with models trained on S2648 excluding S350 set and 5-fold cross validation is conducted for S2648.

Table B.1: Comparison of Pearson correlation (R_p) and RMSEs (kcal/mol) of various methods on prediction of mutation induced protein stability changes of the S350 set and 5-fold cross validation of mutation induced protein stability changes of the S2648. n represents number of samples successfully processed. LTP1 is our topological based mutation predictor that solely utilizes structural information. LTP2 is our model that complements LTP1 with auxiliary features. The results reported in the publications are listed in the table [70].50 repeated runs are conducted and median values of metrics are picked for LTP2 and LTP1 methods.

Method	S350			s2648		
	n	R_p	RMSE	n	R_p	RMSE
LTP2 ($EIG^{E}_{11.0,0.5}$)	350	0.81	0.93	2648	0.78	0.92
LTP2 ($EIG^{L}_{12.0,2.0}$)	350	0.80	0.96	2648	0.78	0.92
STRUM	350	0.79	0.98	2647	0.77	0.94
LTP1 ($EIG^{E}_{4.0,1.3}$)	350	0.76	1.05	2648	0.72	1.03
LTP1 ($EIG^{L}_{6.0,1.3}$)	350	0.75	1.07	2648	0.72	1.02
mCSM	350	0.73	1.08	2643	0.69	1.07
INPS	350	0.68	1.25	2648	0.56	1.26
PoPMuSiC 2.0	350	0.67	1.16	2647	0.61	1.17
PoPMuSic 1.0	350	0.62	1.23	-	-	-
I-Mutant 3.0	338	0.53	1.35	2636	0.60	1.19
Dmutant	350	0.48	1.28	-	-	-
Automute	315	0.46	1.42	-	-	-
CUPSAT	346	0.37	1.46	-	-	-
Eris	324	0.35	1.49	-	-	-
I-Mutant 2.0	346	0.29	1.50	-	-	-

B.2 Supplementary result on Q3421 dataset

In 5-fold cross validation of Q3421 set, R_p/RMSE (kcal/mol) of 0.79/1.2 was reported for STRUM method [70].

Table B.2: 5-fold cross validation results of Q3421 with respect to R_p and RMSE. 50 repeated runs were conducted and median values of metrics were picked for LTP2 and LTP1 methods. Comparison of Pearson correlation (R_p) and RMSEs (kcal/mol) of various methods on 5-fold cross validation of mutation induced protein stability changes of the Q3421. n represents number of samples successfully processed. In LTP1 method, two-scales models are considered by coupling two sets of Hessian eigenvalue features. Moreover, R_p/RMSE (kcal/mol) of 0.79/1.2 was reported for STRUM method [70]. 50 repeated runs are conducted and median values of metrics are picked for LTP2 and LTP1 methods.

Method	Features	Q3421	
		R_p	RMSE
LTP2	$EIG^{E}_{14.0,0.6}$	0.79	1.22
	$EIG^{L}_{10.0,0.6}$	0.79	1.22
LTP1	$EIG^{E}_{2.0,1.5}$	0.68	1.44
	$EIG^{L}_{3.0,1.3}$	0.68	1.44
	$EIG^{E}_{2.0,1.5}$; $EIG^{E}_{5.0,0.7}$	0.70	1.42
	$EIG^{E}_{2.0,1.5}$; $EIG^{L}_{22.0,0.9}$	0.70	1.41

B.3 Supplementary result on optimal parameter selection

We searched for optimal parameters (i.e. α, β and τ) for Hessian eigenvalues features $EIG^{\alpha}_{\beta,\tau}$ on both datasets and methods.

Predictive behaviors of LTP1 model on protein folding energy change upon mutation. Median values of Pearson correlation (R_p) are plotted against β and τ. Predictions on S350 with (A) Exponential kernel and (B) Lorentz kernel. 5-fold cross validation on S2648 with (C) Exponential kernel and (D) Lorentz kernel. In (A-B), Pearson correlations below 0.7 take the same color with 0.7; in (C-D), Pearson correlations below 0.68 take the same color with 0.68. Parameters with highest Pearson correlation: (A) $EIG^{E}_{4.0,1.4}$ with R_p/RMSE of 0.76/1.05; (B) and (D) $EIG^{L}_{6.0,1.3}$ with R_p/RMSE of 0.75/1.07; (C) $EIG^{E}_{4.0,1.4}$ with R_p/RMSE of 0.72/1.03; (D) $EIG^{L}_{6.0,1.3}$ with R_p/RMSE of 0.72/1.02.

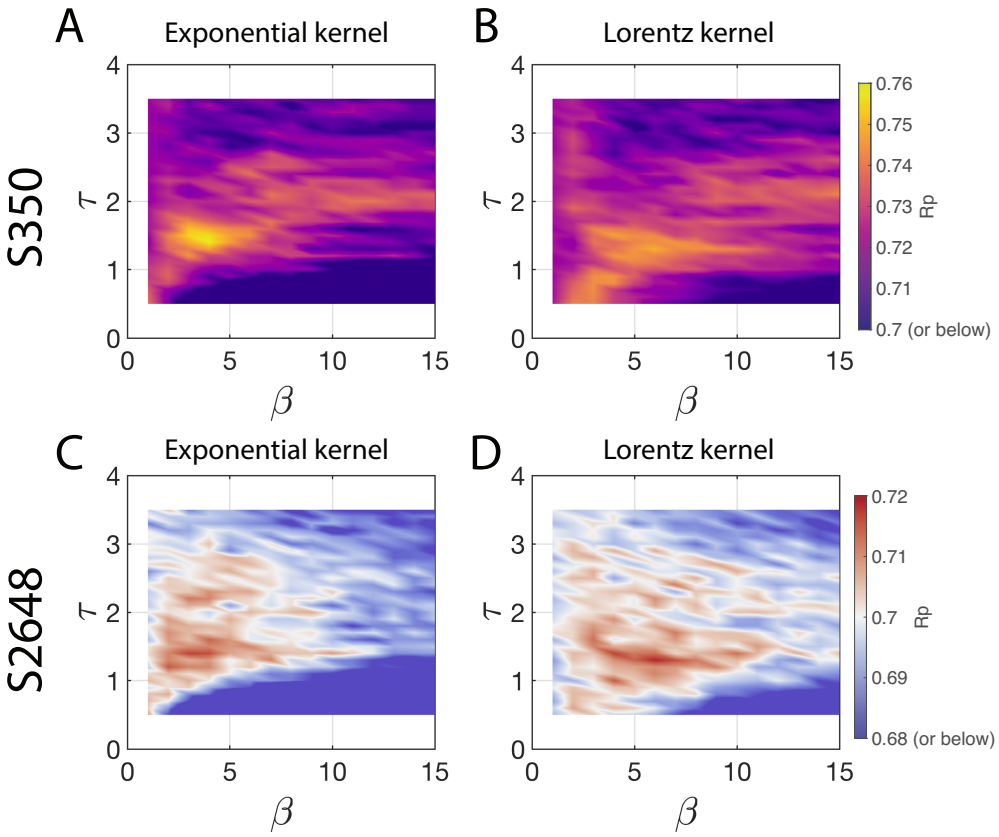


Figure B.1: Predictive behaviors of LTP1 model on protein folding energy change upon mutation.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Adams, Henry, Andrew Tausz & Mikael Vejdemo-Johansson. 2014. Javaplex: A research software package for persistent (co) homology. In *International congress on mathematical software*, 129–136. Springer.

[2] Anfinsen, Christian B. 1973. Principles that govern the folding of protein chains. *Science* 181(4096). 223–230.

[3] Barouch, Dan H, James B Whitney, Brian Moldt, Florian Klein, Thiago Y Oliveira, Jinyan Liu, Kathryn E Stephenson, Hui-Wen Chang, Karthik Shekhar, Sanjana Gupta et al. 2013. Therapeutic efficacy of potent neutralizing hiv-1-specific monoclonal antibodies in shiv-infected rhesus monkeys. *Nature* 503(7475). 224.

[4] Bas, Delphine C, David M Rogers & Jan H Jensen. 2008. Very fast prediction and rationalization of pka values for protein–ligand complexes. *Proteins: Structure, Function, and Bioinformatics* 73(3). 765–783.

[5] Bava, K Abdulla, M Michael Gromiha, Hatsuho Uedaira, Koji Kitajima & Akinori Sarai. 2004. Protherm, version 4.0: thermodynamic database for proteins and mutants. *Nucleic acids research* 32(suppl 1). D120–D121.

[6] Ben-Kasus, Tsipi, Bilha Schechter, Michael Sela & Yosef Yarden. 2007. Cancer therapeutic antibodies come of age: targeting minimal residual disease. *Molecular oncology* 1(1). 42–54.

[7] Berman, Helen M, John Westbrook, Zukang Feng, Gary Gilliland, Talapady N Bhat, Helge Weissig, Ilya N Shindyalov & Philip E Bourne. 2000. The protein data bank. *Nucleic acids research* 28(1). 235–242.

[8] Biovia, Dassault Systemes. 2017. Discovery studio modeling environment.

[9] Breiman, Leo. 1997. Arcing the edge. Tech. rep. Technical Report 486, Statistics Department, University of California at . . . .

[10] Cang, Z. X., Lin Mu, Kedi Wu, Kris Opron, Kelin Xia & Guo-Wei Wei. 2015. A topological approach to protein classification. *Molecular based Mathematical Biology* 3. 140–162.

[11] Cang, Z. X. & G. W. Wei. 2017. Analysis and prediction of protein folding energy changes upon mutation by element specific persistent homology. *Bioinformatics* 33. 3549–3557.

[12] Cang, Zixuan, Lin Mu & Guo-Wei Wei. 2018. Representability of algebraic topology for biomolecules in machine learning based scoring and virtual screening. *PLoS computational biology* 14(1). e1005929.

[13] Cang, Zixuan & Guo-Wei Wei. 2017. Topologynet: Topology based deep convolutional and multi-task neural networks for biomolecular property predictions. *PLoS computational biology* 13(7). e1005690.

[14] Cang, Zixuan & Guo-Wei Wei. 2018. Integration of element specific persistent homology and machine learning for protein-ligand binding affinity prediction. *International journal for numerical methods in biomedical engineering* 34(2). e2914.

[15] Carter, Paul J. 2006. Potent antibody therapeutics by design. *Nature reviews immunology* 6(5). 343.

[16] Chen, Jiahui, Rui Wang, Menglun Wang & Guo-Wei Wei. 2020. Mutations strengthened sars-cov-2 infectivity. *Journal of Molecular Biology* 432(19). 5212–5226. doi:https://doi.org/10.1016/j.jmb.2020.07.009. `https://www.sciencedirect.com/science/article/pii/S0022283620304563`.

[17] Chiti, Fabrizio & Christopher M Dobson. 2006. Protein misfolding, functional amyloid, and human disease. *Annu. Rev. Biochem.* 75. 333–366.

[18] Chollet, François. 2015. Keras. `https://github.com/fchollet/keras`.

[19] Chothia, Cyrus, Arthur M Lesk, Anna Tramontano, Michael Levitt, Sandra J Smith-Gill, Gillian Air, Steven Sheriff, Eduardo A Padlan, David Davies, William R Tulip et al. 1989. Conformations of immunoglobulin hypervariable regions. *Nature* 342(6252). 877.

[20] Collaborative, Computational Project et al. 1994. The ccp4 suite: programs for protein crystallography. *Acta crystallographica. Section D, Biological crystallography* 50(Pt 5). 760.

[21] Dehouck, Yves, Aline Grosfils, Benjamin Folch, Dimitri Gilis, Philippe Bogaerts & Marianne Rooman. 2009. Fast and accurate predictions of protein stability changes upon mutations using statistical potentials and neural networks: Popmusic-2.0. *Bioinformatics* 25. 2537–2543.

[22] Dehouck, Yves, Jean Marc Kwasigroch, Marianne Rooman & Dimitri Gilis. 2013. Beatmusic: prediction of changes in protein–protein binding affinity on mutations. *Nucleic acids research* 41(W1). W333–W339.

[23] Demarest, Stephen J & Scott M Glaser. 2008. Antibody therapeutics, antibody engineering, and the merits of protein stability. *Current opinion in drug discovery & development* 11(5). 675–687.

[24] Demerdash, O. N. A., M. D. Daily & J. C. Mitchell. 2009. Structure-based predictive models for allosteric hot spots. *PLOS Computational Biology* 5. e1000531.

[25] Dolinsky, Todd J, Jens E Nielsen, J Andrew McCammon & Nathan A Baker. 2004. Pdb2pqr: an automated pipeline for the setup of poisson–boltzmann electrostatics calculations. *Nucleic acids research* 32(suppl_2). W665–W667.

[26] Dreyfus, Stuart E. 1990. Artificial neural networks, back propagation, and the kelley-bryson gradient procedure. *Journal of guidance, control, and dynamics* 13(5). 926–928.

[27] Edelsbrunner, H., D. Letscher & A. Zomorodian. 2002. Topological persistence and simplification. *Discrete Comput. Geom.* 28. 511–533.

[28] Fasy, Brittany Terese, Jisu Kim, Fabrizio Lecci & Clément Maria. 2014. Introduction to the r package tda. *arXiv preprint arXiv:1411.1830* .

[29] Fogolari, F., A. Brigo & H. Molinari. 2002. The Poisson-Boltzmann equation for biomolecular electrostatics: a tool for structural biology. *Journal of Molecular Recognition* 15(6). 377–92. http://dx.doi.org/10.1002/jmr.577.

[30] Fogolari, F., A. Brigo & H. Molinari. 2003. Protocol for mm/pbsa molecular dynamics simulations of proteins. *Biophysical Journal* 85(1). 159–166. http://www.biophysj.org/cgi/content/full/85/1/159.

[31] Folkman, L., B. Stantic, A. Sattar & Y. Zhou. 2016. EASEMM: sequence-based prediction of mutation-induced stability changes with feature-based multiple models. *J. Mol. Biol.* 428. 1394–1405.

[32] Friedman, Jerome H. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics* 1189–1232.

[33] Friedman, Jerome H. 2017. *The elements of statistical learning: Data mining, inference, and prediction*. springer open.

[34] Frosini, Patrizio. 1990. A distance for similarity classes of submanifolds of a euclidean space. *Bulletin of the Australian Mathematical Society* 42(3). 407–415.

[35] Frosini, Patrizio & Claudia Landi. 1999. Size theory as a topological tool for computer vision. *Pattern Recognition and Image Analysis* 9(4). 596–603.

[36] Fukushima, Kunihiko & Sei Miyake. 1982. Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In *Competition and cooperation in neural nets*, 267–285. Springer.

[37] Gameiro, M., Y. Hiraoka, S. Izumi, M. Kramar, K. Mischaikow & V. Nanda. 2014. Topological measurement of protein compressibility via persistence diagrams. *Japan Journal of Industrial and Applied Mathematics* 32. 1–17.

[38] Geng, Cunliang, Li C Xue, Jorge Roel-Touris & Alexandre MJJ Bonvin. 2019. Finding the $\delta\delta$g spot: Are predictors of binding affinity changes upon mutations in protein–protein interactions ready for it? *Wiley Interdisciplinary Reviews: Computational Molecular Science* e1410.

[39] Geng, Weihua, Sining Yu & G. W. Wei. 2007. Treatment of charge singularities in implicit solvent models. *Journal of Chemical Physics* 127. 114106.

[40] Getov, Ivan, Marharyta Petukh & Emil Alexov. 2016. Saafec: Predicting the effect of single point mutations on protein folding free energy using a knowledge-modified mm/pbsa approach. *International Journal of Molecular Sciences* 17. 512.

[41] Glennie, Martin J & Jan GJ van de Winkel. 2003. Renaissance of cancer therapeutic antibodies. *Drug discovery today* 8(11). 503–510.

[42] Hammarström, Per, R Luke Wiseman, Evan T Powers & Jeffery W Kelly. 2003. Prevention of transthyretin amyloid disease by changing protein misfolding energetics. *Science* 299(5607). 713–716.

[43] Haykin, Simon. 2010. *Neural networks and learning machines, 3/e*. Pearson Education India.

[44] He, Kaiming, Xiangyu Zhang, Shaoqing Ren & Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 770–778.

[45] Ivakhnenko, Alekseĭ. ????. Cybernetic predicting devices. Tech. rep.

[46] Ivakhnenko, Alekseĭ. ????. *Cybernetics and forecasting techniques*.

[47] Jankauskaitė, Justina, Brian Jiménez-García, Justas Dapkūnas, Juan Fernández-Recio & Iain H Moal. 2018. Skempi 2.0: an updated benchmark of changes in protein–protein binding energy, kinetics and thermodynamics upon mutation. *Bioinformatics* 35(3). 462–469.

[48] Kellogg, E. H., A. Leaver-Fay & D. Baker. 2011. Role of conformational sampling in computing mutation-induced changes in protein structure and stability. *Proteins: Struct., Funct., Genet.* 79. 830–838.

[49] Kleene, Stephen Cole. 2016. *Representation of events in nerve nets and finite automata*. Princeton University Press.

[50] Kortemme, Tanja, Alexandre V Morozov & David Baker. 2003. An orientation-dependent hydrogen bonding potential improves prediction of specificity and structure for proteins and protein–protein complexes. *Journal of molecular biology* 326(4). 1239–1259.

[51] Kovacev-Nikolic, Violeta, Peter Bubenik, Dragan Nikolić & Giseon Heo. 2016. Using persistent homology and dynamical distances to analyze protein binding. *Stat. Appl. Genet. Mol. Biol.* 15(1). 19–38.

[52] Kucukkal, Tugba G, Marharyta Petukh, Lin Li & Emil Alexov. 2015. Structural and physico-chemical effects of disease and non-disease nssnps on proteins. *Current opinion in structural biology* 32. 18–24.

[53] LeCun, Yann, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard & Lawrence D Jackel. 1989. Backpropagation applied to handwritten zip code recognition. *Neural computation* 1(4). 541–551.

[54] Lensink, Marc F & Shoshana J Wodak. 2013. Docking, scoring, and affinity prediction in capri. *Proteins: Structure, Function, and Bioinformatics* 81(12). 2082–2095.

[55] Levy, Emmanuel D. 2010. A simple definition of structural regions in proteins and its use in analyzing interface evolution. *Journal of molecular biology* 403(4). 660–670.

[56] Liu, Beibei, Bao Wang, Rundong Zhao, Yiying Tong & Guo Wei Wei. 2017. ESES: software for Eulerian solvent excluded surface. *Journal of Computational Chemistry* 38. 446–466.

[57] Liu, Song, Chi Zhang, Hongyi Zhou & Yaoqi Zhou. 2004. A physical reference state unifies the structure-derived potential of mean force for protein folding and binding. *Proteins: Structure, Function, and Bioinformatics* 56(1). 93–101.

[58] Martinez, JL & F Baquero. 2000. Mutation frequencies and antibiotic resistance. *Antimicrobial agents and chemotherapy* 44(7). 1771–1777.

[59] Miller, Susan, Joel Janin, Arthur M Lesk & Cyrus Chothia. 1987. Interior and surface of monomeric proteins. *Journal of molecular biology* 196(3). 641–656.

[60] Mizutani, Eiji, Stuart E Dreyfus & Kenichi Nishio. 2000. On derivation of mlp backpropagation from the kelley-bryson optimal-control gradient formula and its application. In *Proceedings of the ieee-inns-enns international joint conference on neural networks. ijcnn 2000. neural computing: New challenges and perspectives for the new millennium*, vol. 2, 167–172. IEEE.

[61] Moal, Iain H & Juan Fernández-Recio. 2012. Skempi: a structural kinetic and energetic database of mutant protein interactions and its use in empirical models. *Bioinformatics* 28(20). 2600–2607.

[62] Morrison, Kim L & Gregory A Weiss. 2001. Combinatorial alanine-scanning. *Current opinion in chemical biology* 5(3). 302–307.

[63] Nguyen, Duc Duy, Zixuan Cang, Kedi Wu, Menglun Wang, Yin Cao & Guo-Wei Wei. 2018. Mathematical deep learning for pose and binding affinity prediction and ranking in d3r grand challenges. *Journal of Computer Aided Molecular Design* in press. https://doi.org/10.1007/s10822–018–0146–6.

[64] Patil, Sachin P, Pedro J Ballester & Cassidy R Kerezsi. 2014. Prospective virtual screening for novel p53–mdm2 inhibitors using ultrafast shape recognition. *Journal of computer-aided molecular design* 28(2). 89–97.

[65] Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg et al. 2011. Scikit-learn: Machine learning in python. *Journal of machine learning research* 12(Oct). 2825–2830.

[66] Petukh, Marharyta, Luogeng Dai & Emil Alexov. 2016. Saambe: webserver to predict the charge of binding free energy caused by amino acids mutations. *International journal of molecular sciences* 17(4). 547.

[67] Petukh, Marharyta, Minghui Li & Emil Alexov. 2015. Predicting binding free energy change caused by point mutations with knowledge-modified mm/pbsa method. *PLoS computational biology* 11(7). e1004276.

[68] Pires, Douglas EV & David B Ascher. 2016. mcsm-ab: a web server for predicting antibody–antigen affinity changes upon mutation with graph-based signatures. *Nucleic acids research* 44(W1). W469–W473.

[69] Puente, Xose S, Luis M Sánchez, Christopher M Overall & Carlos López-Otín. 2003. Human and mouse proteases: a comparative genomic approach. *Nature Reviews Genetics* 4(7). 544–558.

[70] Quan, Lijun, Qiang Lv & Yang Zhang. 2016. Strum: structure-based prediction of protein stability changes upon single-point mutation. *Bioinformatics* 32(19). 2936–2946.

[71] Ramensky, Vasily, Peer Bork & Shamil Sunyaev. 2002. Human non-synonymous snps: server and survey. *Nucleic acids research* 30(17). 3894–3900.

[72] Rodrigues, Carlos H M, Yoochan Myung, Douglas E V Pires & David B Ascher. 2019. mcsm-ppi2: predicting the effects of mutations on protein–protein interactions. *Nucleic acids research* .

[73] Rosenblatt, Frank. 1957. *The perceptron, a perceiving and recognizing automaton project para*. Cornell Aeronautical Laboratory.

[74] Rosenblatt, Frank. 1958. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review* 65(6). 386.

[75] Rumelhart, David E, Geoffrey E Hinton & Ronald J Williams. 1985. Learning internal representations by error propagation. Tech. rep. California Univ San Diego La Jolla Inst for Cognitive Science.

[76] Rumelhart, David E, Geoffrey E Hinton & Ronald J Williams. 1986. Learning representations by back-propagating errors. *nature* 323(6088). 533–536.

[77] Scarselli, Franco, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner & Gabriele Monfardini. 2008. The graph neural network model. *IEEE transactions on neural networks* 20(1). 61–80.

[78] Schmidhuber, Jürgen. 2015. Deep learning in neural networks: An overview. *Neural Networks* 61. 85–117.

[79] Schymkowitz, Joost, Jesper Borg, Francois Stricher, Robby Nys, Frederic Rousseau & Luis Serrano. 2005. The foldx web server: an online force field. *Nucleic acids research* 33(suppl_2). W382–W388.

[80] Selkoe, Dennis J. 2003. Folding proteins in fatal ways. *Nature* 426(6968). 900–904.

[81] Shire, Steven J, Zahra Shahrokh & Jun Liu. 2004. Challenges in the development of high protein concentration formulations. *Journal of pharmaceutical sciences* 93(6). 1390–1402.

[82] Simonsen, Shane M, Lillian Sando, K Johan Rosengren, Conan K Wang, Michelle L Colgrave, Norelle L Daly & David J Craik. 2008. Alanine scanning mutagenesis of the prototypic cyclotide reveals a cluster of residues essential for bioactivity. *Journal of biological chemistry* 283(15). 9805–9813.

[83] Sirin, Sarah, James R Apgar, Eric M Bennett & Amy E Keating. 2016. Ab-bind: Antibody binding mutational database for computational affinity predictions. *Protein Science* 25(2). 393–409.

[84] Szilagyi, Andras & Yang Zhang. 2014. Template-based structure modeling of protein–protein interactions. *Current opinion in structural biology* 24. 10–23.

[85] Webb, Benjamin & Andrej Sali. 2014. Comparative protein structure modeling using modeller. *Current protocols in bioinformatics* 47(1). 5–6.

[86] Weiss, Gregory A, Colin K Watanabe, Alan Zhong, Audrey Goddard & Sachdev S Sidhu. 2000. Rapid mapping of protein functional epitopes by combinatorial alanine scanning. *Proceedings of the National Academy of Sciences* 97(16). 8950–8954.

[87] Werbos, Paul. 1974. Beyond regression:" new tools for prediction and analysis in the behavioral sciences. *Ph. D. dissertation, Harvard University* .

[88] Xia, K. L. & G. W. Wei. 2014. Persistent homology analysis of protein structure, flexibility and folding. *International Journal for Numerical Methods in Biomedical Engineering* 30. 814–844.

[89] Xia, K. L. & G. W. Wei. 2015. Persistent topology for cryo-EM data analysis. *International Journal for Numerical Methods in Biomedical Engineering* 31. e02719.

[90] Xiang, Jason Z & B Honig. 2002. Jackal: A protein structure modeling package. *Columbia University and Howard Hughes Medical Institute, New York* .

[91] Xiong, Peng, Chengxin Zhang, Wei Zheng & Yang Zhang. 2017. Bindprofx: assessing mutation-induced binding affinity change by protein interface profiles with pseudo-counts. *Journal of molecular biology* 429(3). 426–434.

[92] Yang, Yuedong, Rhys Heffernan, Kuldip Paliwal, James Lyons, Abdollah Dehzangi, Alok Sharma, Jihua Wang, Abdul Sattar & Yaoqi Zhou. 2017. Spider2: A package to predict secondary structure, accessible surface area, and main-chain torsional angles by deep neural networks. In *Prediction of protein secondary structure*, 55–63. Springer.

[93] Yao, Yuan, Jian Sun, Xuhui Huang, Gregory R Bowman, Gurjeet Singh, Michael Lesnick, Leonidas J Guibas, Vijay S Pande & Gunnar Carlsson. 2009. Topological methods for exploring low-density states in biomolecular folding pathways. *The Journal of chemical physics* 130(14). 04B614.

[94] Zhang, CHI, Song Liu & Yaoqi Zhou. 2004. Accurate and efficient loop selections by the dfire-based all-atom statistical potential. *Protein science* 13(2). 391–399.

[95] Zhang, Zhe, Maria A Miteva, Lin Wang & Emil Alexov. 2012. Analyzing effects of naturally occurring missense mutations. *Computational and mathematical methods in medicine* 2012.

[96] Zhou, Y. C., M. Feig & G. W. Wei. 2008. Highly accurate biomolecular electrostatics in continuum dielectric environments. *Journal of Computational Chemistry* 29. 87–97.

[97] Zhou, Y. C., Shan Zhao, Michael Feig & G. W. Wei. 2006. High order matched interface and boundary method for elliptic equations with discontinuous coefficients and singular sources. *J. Comput. Phys.* 213(1). 1–30.

[98] Zhu, Kai, Tyler Day, Dora Warshaviak, Colleen Murrett, Richard Friesner & David Pearlman. 2014. Antibody structure determination using a combination of homology modeling, energy-based refinement, and loop prediction. *Proteins: Structure, Function, and Bioinformatics* 82(8). 1646–1655.

[99] Zomorodian, A. & G. Carlsson. 2005. Computing persistent homology. *Discrete Comput. Geom.* 33. 249–274.

[100] Zomorodian, Afra & Gunnar Carlsson. 2008. Localized homology. *Computational Geometry - Theory and Applications* 41(3). 126–148.