Contents

[Abstract 2](#_Toc375234359)

[1. Introduction 2](#_Toc375234360)

[2. Biological Background 2](#_Toc375234361)

[2.1 Nucleotide 3](#_Toc375234362)

[2.2 Deoxyribonucleic acid (DNA) 3](#_Toc375234363)

[2.3 Amino Acids 4](#_Toc375234364)

[2.4 Proteins 6](#_Toc375234365)

[2.5 Bioinformatics tools and databases 7](#_Toc375234366)

[2.5.1 Protein Data Bank 8](#_Toc375234367)

[2.5.2 The HSSP Database 9](#_Toc375234368)

[2.5.3 The BLAST Database 9](#_Toc375234369)

[3. Information Theory Backgrounds 10](#_Toc375234370)

[3.1 Shannon Entropy 11](#_Toc375234371)

[3.2 Joint Entropy 11](#_Toc375234372)

[3.3 Conditional Entropy 11](#_Toc375234373)

[3.4 Mutual Information 12](#_Toc375234374)

[3.5 Normalized Mutual Information 12](#_Toc375234375)

[3.6 Kullback Leibler Divergence 12](#_Toc375234376)

[3.7 Jensen Shannon Divergence 12](#_Toc375234377)

[3.8 Position Specific-Scoring Matrix (PSSM) 13](#_Toc375234378)

[4. Machine Learning Methods 14](#_Toc375234379)

[4.1 Support Vector Machine (SVM) 14](#_Toc375234380)

[4.2 Random Forests (RF) 14](#_Toc375234381)

[4.2.1 Classification and Regression Tree (CART) 15](#_Toc375234382)

[4.2.2 Definition of Random Forests 17](#_Toc375234383)

[5. Methods 18](#_Toc375234384)

[Notation 18](#_Toc375234385)

[5.1 Existing Methods for DNA-binding Protein sites Prediction 18](#_Toc375234386)

[5.2 Doubly Stochastic Matrix (DSM) 19](#_Toc375234387)

[5.2.1 Substitution Matrix 19](#_Toc375234388)

[5.2.2 Calculate differential matrix 20](#_Toc375234389)

[5.2.3 Integration of BLOSUM62 21](#_Toc375234390)

[5.2.4 Calculate logarithm marginal distribution 21](#_Toc375234391)

[5.2.5 Calculate DSM 22](#_Toc375234392)

[5.3 Significant Residues/Sites Finder (SRF) 22](#_Toc375234393)

[5.3.1 Normalized Mutual Information Score (NMI Score) 23](#_Toc375234394)

[5.3.2 Connectivity Degree 24](#_Toc375234395)

[5.4 Feature Extraction 25](#_Toc375234396)

[5.4.1 PSSM as Feature Vector 25](#_Toc375234397)

[5.4.2 Boolean Significant Residues as Feature Vector 26](#_Toc375234398)

[5.5 Discussion 27](#_Toc375234399)

[6. Results and Evaluations 27](#_Toc375234400)

[6.1 Data Preparation 27](#_Toc375234401)

[6.2 Results & Evaluations 27](#_Toc375234402)

[7. Conclusions 27](#_Toc375234403)

[8. Discussions 27](#_Toc375234404)

[Bibliography 27](#_Toc375234405)

# Abstract

# Introduction

# Biological Background

In this section, we briefly presents the introductions of some biological term and phrases related to this thesis. After that, we mention a short overview of biological mechanisms which motivate us conduct this project. And last but not least, some useful tools, databases as well as web services relating to bioinformatics will be explained in detail.

## 2.1 Nucleotide

Nucleotides are biological molecules which are the building blocks of DNA or RNA. A typical nucleotide consists of a sugar (deoxyribose), a phosphate and a base (Figure 2.1). There are four types of nucleotides in DNA which are different due to their bases including Adenine(A), Guanine(G), Cytosine(C), and Thymine(T) (for RNA, Uracil(U) replaces Thymine(T)).

Table 2.1: Full name and abbreviation of nucleotides

|  |  |
| --- | --- |
| Full name of base/nucleotide | Abbreviation |
| Adenine | A |
| Guanine | G |
| Cytosine | C |
| Thymine | T |
| Uracil | U |

Sugar

Base

Figure 2.1: Structure of nucleotides

Further, Cytosine and Thymine (or Uracil in RNA) belongs to pyrimidine bases while Adenine and Guanine are in purine bases. Adenine always pairs with Thymine by 2 hydrogen bonds, while Guanine pairs with Cytosine through 3 hydrogen bonds, each due to their unique structures.

## 2.2 Deoxyribonucleic acid (DNA)

Deoxyribonucleic acid (DNA for abbreviation) is the macromolecule encoding genetic information. It is used in all biological mechanisms of all known living forms and virus. DNA is firstly isolated and identified by Friedrich Miescher in late 1890s and its double helix structure is discovered by James Watson and Francis Crick in 1950s. Roughly speaking, DNA comprises of two complement sequences of nucleotides in which pairs A, T and pairs G, C binding together (Figure 2.2[[1]](#footnote-1)).

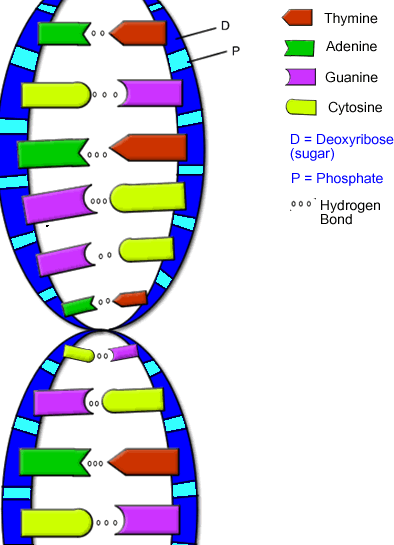


Figure .2: Structure of DNA

## 2.3 Amino Acids

Amino acids are the production of RNA translation which is a part of gene expression. In RNA-translation procedure, a triple nucleotides will be translated into one amino acid. An amino acid consists of central carbon atom (), a hydrogen atom (), free carboxyl group (), free amino group (), and side chain group (). The R group creates the distinction among amino acids due to its chemical structure.

H

COOH

R

Figure 2.3: General structure of amino acid

Overall, there are 20 primary amino acids, and they are classified into two main groups due to their physical and chemical properties.

* Hydrophobic groups:
  + Nonpolar-aliphatic R groups: Glycine, Alanine, Proline, Valine, Leucine, Isoleucine, Methionine.
  + Aromatic R groups: Phenylalanine, Tyrosine, Tryptophan.
* Hydrophilic groups:
  + Polar-charged R groups: Serine, Threonine, Cysteine, Asparagine, Glutamine.
  + Positively charged R groups: Lysine, Arginine, Histidine.
  + Negatively charged R groups: Aspartate, Glutamate.

Table .2: Names and abbreviations of 20 types of amino acids

|  |  |  |  |
| --- | --- | --- | --- |
| Number | Full name | Three letters abbreviation | One letter abbreviation |
| 1 | Alanine | Ala | A |
| 2 | Arginine | Arg | R |
| 3 | Asparagine | Asn | N |
| 4 | Aspartic | Asp | D |
| 5 | Cysteine | Cys | C |
| 6 | Glutamic | Glu | E |
| 7 | Glutamine | Gln | Q |
| 8 | Glycine | Gly | G |
| 9 | Histidine | His | H |
| 10 | Isoleucine | Ile | I |
| 11 | Leucine | Leu | L |
| 12 | Lysine | Lys | K |
| 13 | Methionine | Met | M |
| 14 | Phenylalanine | Phe | F |
| 15 | Proline | Pro | P |
| 16 | Serine | Ser | S |
| 17 | Threonine | Thr | T |
| 18 | Tryptophan | Trp | W |
| 19 | Tyrosine | Tyr | Y |
| 20 | Valine | Val | V |

Moreover, all over 20 amino acids are necessary for protein synthesis to maintain the body growth and function. Among 20 amino acids, there are eleven ones (Alanine, Arginine, Asparagine, Aspartic, Cysteine, Glutamic, Glycine, Proline, Serine, Tyrosine) called non-essential amino acids because they can be synthesized within body. And the rest of nine amino acids are essential ones and only could be taken from additional nutrition [[1](#BHa04)].

## 2.4 Proteins

Protein is a large biological molecule consisting of several chains of amino acids. An individual chain of these amino acids has a specific spatial arrangement driven by non-covalent interactions such as hydrogen bonding, van der Waals forces, and hydrophobic packing. These spatial arrangements play the key role to define structure and functions of protein.

In general, a certain protein is described via four structural levels including primary, secondary, tertiary, and quaternary structure (figure 2.4[[2]](#footnote-2)).

* Primary structure: the linear sequences of amino acids in polypeptide chain.
* Secondary structure: is highly regular local structures consisting of α-helices and β-strand (pleated sheet) driven by hydrogen bonds.
* Tertiary structure: a three dimensional structure of protein chain, including secondary structure.
* Quaternary structure: a complex consisting of tertiary structure of several chains to describe a functional protein.

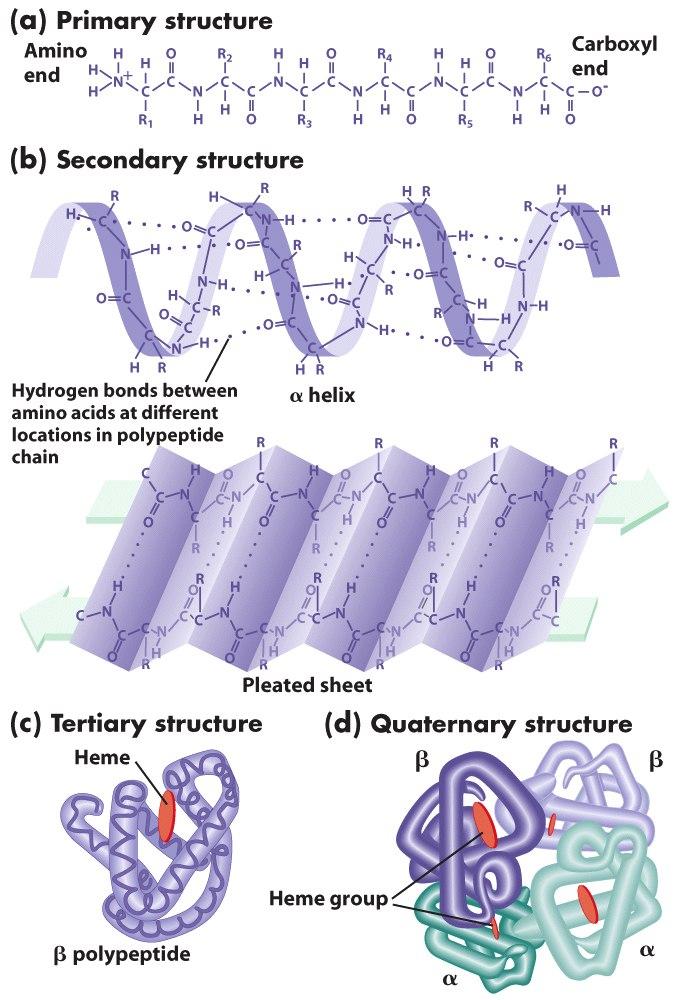


Figure 2.4: a) primary structure: sequence of amino acids in chain. b) secondary structure: α-helix and β-strand (pleated sheet). c) tertiary structure: 3D structure of single chain of amino acids. d) quaternary structure: a complex of several protein chains.

## 2.5 Bioinformatics tools and databases

This section would briefly present some common and useful databases in which all properties of proteins such as three dimensional structure and binding indexes can be found there. In addition, some web services used to detect protein DNA binding residual will be introduced as well.

### 2.5.1 Protein Data Bank[[3]](#footnote-3)

Protein data bank (PDB for abbreviation) is a worldwide collection of three dimensional structures of large biological molecules such as proteins and nucleic acids. This repository includes more than 86000 proteins, 2500 nucleic acids, and 4000 complexes. In particular, all identified proteins are indicated via four characters (A to Z or 0 to 9) such as “1V4S” for human glucokinase protein. In addition, in order to present the structural information of proteins, Proteins Data Bank has a conventional format called “pdb” format which allows researchers retrieve information of atomic coordinates, literature references and experimental details as well. Moreover, primary and secondary structure information such as disulfide bonds, helices, strand, binding sites, active sites are also presented in this format.

In sort, every protein available in PDB is formatted under extension “pdb” in which primary, secondary and tertiary structure information is recorded. Moreover, there are quite a bunch of records in PDB files which are explained in detail in [[2](#PDB)]. In the scope of this dissertation, we will briefly mention some important records which are used in feature extraction process.

#### HEADER and TITLE records

|  |
| --- |
| HEADER TRANSCRIPTION/DNA 27-NOV-97 1A0A  TITLE PHOSPHATE SYSTEM POSITIVE REGULATORY PROTEIN PHO4/DNA |

The HEADER tag consists of classification for the entry, the date when the coordinates were composited to PDB archive, and a unique identified PDB entry. And the TITLE record describes the name of experiment or analysis represented in the entry.

#### HELIX and SHEET records

|  |
| --- |
| HELIX 36 36 SER H 88 LEU H 98 1 11  HELIX 37 37 GLY H 101 THR H 119 1 19  SHEET 1 A 2 ARG A 83 PHE A 84 0  SHEET 2 A 2 THR B 80 VAL B 81 1 N VAL B 81 O ARG A 83 |

HELIX and SHEET record the secondary structure of the protein. Firstly, in HELIX records, the beginning and ending positions are located as well as their chain identity and types. The total length of helix is also mentioned at the end of record. Secondly, SHEET records describe another type of secondary structural information of protein. It records the begin and end position of β-strand as well as the length of this strand.

#### ATOM records

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Pos. | 1-6 | 7-11 | 13-16 | 17 | 18-20 | 22 | 23-26 | 27 | 31-38 | 39-46 | 47-54 | 55-60 | 61-66 | 77-78 | 79-80 |
| Eg. | ATOM | 1 | N | A | ARG | A | 14 |  | 11.281 | 86.699 | 94.383 | 0.50 | 35.88 | N |  |
| Desc. |  | | | | | | | | | | | | | | charge |
|  | | | | | | | | | | | | | Element symbol | |
|  | | | | | | | | | | | | Temperature factor | | |
|  | | | | | | | | | | | Occupancy | | | |
|  | | | | | | | | | | Z coordinate | | | | |
|  | | | | | | | | | Y coordinate | | | | | |
|  | | | | | | | | X coordinate | | | | | | |
|  | | | | | | | Code for insertion of residues | | | | | | | |
|  | | | | | | Residue sequence number | | | | | | | | |
|  | | | | | Chain identifier | | | | | | | | | |
|  | | | | Residue name | | | | | | | | | | |
|  | | | Alternate location indicator | | | | | | | | | | | |
|  | | Atom name | | | | | | | | | | | | |
|  | Atom serial number | | | | | | | | | | | | | |
| Record name | | | | | | | | | | | | | | |

The ATOM records consist of three dimension coordinate of atom in nucleic acid or amino acid. From this coordinate information, we can compute distance between any pair of atom, and easily detect binding residuals.

### 2.5.2 The HSSP Database

The HSSP (homology-derived structure of proteins) [[3](#HSSP)] is modified-database combining primary and tertiary structural information. Proteins of multiple sequence alignment in HSSP database not only share primary information, but they are also very likely to have closed tertiary structure.

### 2.5.3 The BLAST Database

BLAST[[4]](#footnote-4) which stands for Basic Local Alignment Search Tool is a program to retrieve protein sequences similar to a given sequence and then makes an alignment of them. Researchers are able to compare a given query sequence (amino acids or nucleonic) with a library of sequences, and identify the analogous sequences whose scores are above certain threshold. There are several modified version of BLAST according to sequence types, for example:

* Nucleotide Blast: Search a nucleotide database using a nucleotide query.
* Protein Blast: Search protein database using a protein query.
* Blastx: Search protein database using a translated nucleotide query.
* TBlastn: Search translated nucleotide database using a protein query.
* TBlastx: Search translated nucleotide database using a translated nucleotide query.

In this project, because our queries belong to category of protein sequence, Protein Blast is a method of choice. In addition, there are a few library or dataset of sequence corresponding to Protein Blast:

* Non-redundant protein sequences: all protein sequences from GenBank, PDB, SwissProt, PIR, PRF in which all redundant sequences will be counted only one time.
* Reference Proteins: NCBI Protein Reference Sequences.
* UniProtKB/Swiss-Prot: Non-redundant UniProtKB/SwissProt sequences.
* Patented Protein Sequences: Protein sequences derived from the Patent division of GenBank.
* Protein-Data-Bank proteins: This database consists of sequences from the Protein Data Bank (PDB), which contains information about experimentally-determined structures of proteins, nucleic acids, and complex assemblies.
* Metagenomic proteins: Proteins from WGS metagenomic projects.
* Transcriptome Shotgen Assembly proteins[[5]](#footnote-5): The Transcriptome Shotgun Assembly proteins are produced from CDS features on mRNA sequences in the Transcriptome Shotgun Assembly sequences.

Non-redundant protein sequences library is chosen in this project due to several folds. First, though this is the largest dataset and most time consuming one among the others, it is still affordable for modern personal computer. Second, this dataset allow us to get as many truly analogous sequences as we want to create reliable and informative PSSM.

# 3. Information Theory Backgrounds

In this section, we recapitulate some fundamental concepts in classic information theory. Firstly, we will provide some basic definitions such as Shannon entropy, joint entropy and mutual information. Then the next subsections discuss some divergent measurements which allow us to measure how much difference between two distributions.

## 3.1 Shannon Entropy

Given a random variable , Shannon Entropy is defined by:

Where is all possible outcomes, and p(x) = P(X=x) with . The entropy is maximal if is under uniform randomly distribution.

Moreover, there are several interpretations of Shannon entropy which belong to two main categories. The first one interprets Shannon entropy as a capacity in information channel as well as limitation of compression ratio. Another point of view is to consider Shannon entropy is the minimal number of bits (or yes/no questions) required to eliminate the uncertainty.

## 3.2 Joint Entropy

Given a pair discrete random variables () with a joint distribution , then joint entropy is defined:

Where χ, γ are outcome space of random variable and .

## 3.3 Conditional Entropy

Given a pair discrete random variables () with a joint distribution and conditional distribution , then the conditional entropy is defined as:

So, the conditional entropy is:

Where χ, γ are outcome space of random variable and .

When and are completely independent, the .

## 3.4 Mutual Information

Suppose we have two random variable and , and we would like to have a mean to measure the amount of mutual dependency between and . Thus, mutual information is a facility to measure how much and is different from each other.

So, the definition of mutual information is following:

There are two notices from above formulas. Firstly, mutual information will approach zero if and totally independent. Secondly, if and are completely correlated, we have .

## 3.5 Normalized Mutual Information

The capacity of source is sometimes a problem. For instance,

## 3.6 Kullback Leibler Divergence

Kullback Leibler divergence is a non-symmetric measure of the difference between two probability distributions  and . Its formula is given by (assume and are concrete probability distribution):

The value of Kullback Leibler divergence could diverse from zero (when P ≈ Q) to infinitive. Moreover, this divergence could be interpreted as the information lost when is used to approximate .

## 3.7 Jensen Shannon Divergence

Jensen-Shannon divergence is yet another popular method to measure the different between two different probability distributions. In contrast to Kullback-Leibler divergence whose value could go to infinitive, Jensen-Shannon ensures the range of score lie between zero and one; thus it enables us apply some threshold detection method. Overall, the formula of this divergence is defined by:

Where is Kullback-Leibler divergence and .

This divergence is symmetric and approach zero when P and Q are equivalent while get value of one when P and Q completely independent.

## 3.8 Position Specific-Scoring Matrix (PSSM)

PSSM is a table containing likelihood information of nucleotide residues at each position of a set of sequences. In the other word, this table captures the positional information of the set of nucleotide residues sequences. In such table, the row represents residue position and column represents the probability of residue in this position. The value in the table tells us how likely the residue occurs in the set of sequences.

Let us look at the example [[4](#Jin06)], we have a set of sequences which have length of six.

First of all, we have in hand a set of sequences with length 6.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Position | 1 | 2 | 3 | 4 | 5 | 6 |
| Sequence 1 | A | T | G | T | C | G |
| Sequence 2 | A | A | G | A | C | T |
| Sequence 3 | T | A | C | T | C | A |
| Sequence 4 | C | G | G | A | G | G |
| Sequence 5 | A | A | C | C | T | G |

We then store the positional information of these sequences by matrix 4 x 6 (dark blue area). Each entry in this matrix is calculated as the frequency of the letter in a certain position. In addition, the overall frequency tells us the total frequency of letter (or nucleotide residue) in whole dataset. We also call this matrix as a positional frequency matrix because we will utilize it to create randomly typical sequence mentioned in the overflow section, and this will be discussed in detail later.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Position | 1 | 2 | 3 | 4 | 5 | 6 | Overall freq. |
| A | 0.6 | 0.6 | -- | 0.4 | -- | 0.2 | 0.3 |
| T | 0.2 | 0.2 | -- | 0.4 | 0.2 | 0.2 | 0.2 |
| G | -- | 0.2 | 0.6 | -- | 0.2 | 0.6 | 0.27 |
| C | 0.2 | -- | 0.4 | 0.2 | 0.6 | -- | 0.23 |

After that, the matrix is normalized by dividing each entry by its own overall frequency. The ratio entry represents how much the nucleotide residue fitting to the matrix. For example, residue ‘A’ in position 1 has ratio of 2. It means that the probability of occurrence of residue ‘A’ in sequence is twice than random chance.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Position | 1 | 2 | 3 | 4 | 5 | 6 |
| A | 2 | 2 | -- | 4/3 | -- | 2/3 |
| T | 1 | 1 | -- | 2 | 1 | 1 |
| G | -- | 0.74 | 2.22 | -- | 0.74 | 2.22 |
| C | 0.87 | -- | 1.74 | 0.87 | 2.61 | -- |

Finally, we take logarithm (usually base 2) to the matrix in order to have a better interpretation.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Position | 1 | 2 | 3 | 4 | 5 | 6 |
| A | 1 | 1 | -- | 0.41 | -- | -0.58 |
| T | 0 | 0 | -- | 1 | 0 | 0 |
| G | -- | -0.43 | 1.15 | -- | -0.43 | 1.15 |
| C | -0.2 | -- | 0.8 | -0.2 | 1.38 | -- |

The last table allows us to measure the match score of new sequence with respect to the PSSM. For example, let us take the sequence AACTCG, and the corresponding match score is 1+1+0.8+1+1.38+1.15 = 6.33. We could interpret this as following: according to the dataset, the probability to pick up sequence AACTCG is 26.33≈80 times more than by random chance.

# 4. Machine Learning Methods

There are several learning methods such as neutron network, support vector machine, random forest … (viet lai)

## 4.1 Support Vector Machine (SVM)

## 4.2 Random Forests (RF)

Random Forests [[5](#RandomForest)] is very popular statistical learning method developed Breiman. In this method, many classification trees which are substantial modified are ensemble to conduct a final classifier. Assembling or bagging strategy has advantage of reducing the variance of individual tree. In the following sub-section, we will present the concept of tree-based learning called classification and regression tree. Then the formal definition as well as algorithm of Random Forest will be described.

### 4.2.1 Classification and Regression Tree (CART)

“*Our philosophy in data analysis is to look at the data from a number of different viewpoints. Tree structured regression offers an interesting alternative for looking at regression type problems. It has sometimes given clues to data structure not apparent from a linear regression analysis. Like any tool, its greatest benefit lies in its intelligent and sensible application*”. Breiman, Friedman, Olshen, Stone.

The above insight from Breiman et al is the motivation for tree-based learning methods. Among of them, CART is a very popular one and the modified version of it is used for random forests.

Table 4.1: CART algorithm

|  |
| --- |
| **Input**: , where is number of training set. (real value vector of length ). And .  **Convention**: a pair () where is variable having domain from 1 to ( is index of feature dimension), and is real value. |
| **CART algorithm**:   * Choose a pair ( such that we create “greatest separate” in the target. (The term “greatest separate” will be explained later). Alternatively, this pair is also called splitting point. * For i=1 to do:   If < then send to the left node  Else send to the right node.  End for loop.   * Recursively do the above step to these new nodes until a node gets homogeneity or the number of data point is below a certain criteria. * If a node gets homogeneity (the same value of ), the this node will vote for this homogeneous value. * If a node is not homogeneous but its number of data points is below criteria, then voting according to the majority of |
| **Output**: a tree with leaves are decision making of value {0,1}. |

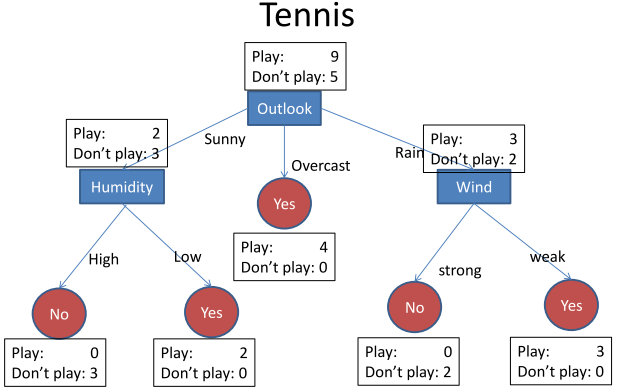


Figure : example of tennis decision making using CART tree. In the observation, there are totally 9 persons would like to play tennis and 5 persons does not.

As mentioned above, choosing a pair to have the greatest separate of data point plays a key role in CART tree performance. There are several methods for conducting splitting point which we will discuss later. In general, the ultimate goal of splitting parent node to two child nodes (left and right) is to maximize the homogeneity of these child nodes.

The homogeneity of a node is defined by the impurity function, called , where is a node.

In another point of view, maximizing the homogeneity of a node is equivalent to maximize the change of impurity between parent node and its child nodes.

So, the constraint for splitting node is defined as

#### There are a lot of definitions for impurity function, but the following definitions are the most popular ones.

#### Entropy based rule

In entropy point of view, the impurity of a node is measured via its entropy. For example, if node is homogenous or impurity of node is zero, it implies that the entropy of node is zero as well.

#### Gini based rule

Where are from 1 to K (K is the number of class. In this scenario, K=2 :{0,1}), and is probability of class given node .

### 4.2.2 Definition of Random Forests

Table : Random Forest algorithm

|  |
| --- |
| **Input**: , where is number of training set. (real value vector of length ). And . |
| **Random Forests algorithm**:   * For b=1 to B do: // B is number of tree in forests   a) Draw a random bootstrap sample of size M from whole training data.  b) Grow a bth tree Tb from this bootstrapped data recursively by following steps:  i) Select m variables at random from the p variables  ii) Pick the best variable and split-point among the m  iii) Split the node into two child nodes |
| Output: . And classification is on majority vote. |

General speaking, Random Forests is a classifier consisting of a collection of tree-constructed classifiers. These trees are building up in independent and identical distribution. Random Forests has overcome the disadvantages of tree based classifier which are sometimes very low performance due to its over-fitting by randomly constructing many trees. As the result, the ensemble of these trees could decrease the variance while maintaining the performance [[5](#RandomForest)].

# 5. Methods

## Notation

Throughout this thesis, we agree on some following terminologies and notations.

|  |  |
| --- | --- |
|  | is the set of protein sequences in training or testing datasets |
|  | is the ith sequence of , i=1..n, where n is the number of protein sequences in dataset. |
|  | is the notation of amino acid of in j position, j=1..α, where α is the length of this protein sequence. |
|  |  |

This chapter is organized as following. Section 5.1 will recapitulate some highlight existing method in detecting DNA-Protein binding residuals. Then section 5.2 will present a method which is able to increase compensatory mutation signal associating with DNA-Protein binding sites as well as to decrease that signal from non-binding residuals. Finally, the whole framework of extracting feature to random forests will be discussed in section 5.3

## 5.1 Existing Methods for DNA-binding Protein sites Prediction

Prediction of DNA-binding Protein residuals has drawn researchers’ attention because it plays some important role in biological processes such as gene regulation, DNA replication and repair as well as in cellular development [???]. Many approaches have been proposed to detect DNA Protein binding including structural information and electrostatic.

Structural information derived from protein sequence has been proved as an informative source for feature extraction of detecting DNA-Protein binding sites. For instance, Tjong et al. [[6](#Tjong07)] have deployed the protein surfaces in 3 dimensional structure to predict the binding sites. In addition, Jones S. et al. [[7](#Jones03)] and Tsuchiya [[8](#Tsuchiya04)] use protein surfaces as well as electrostatic for the prediction. Though utilizing 3D-structural information often gives good results, its limitations are quite significant. Firstly, the number of known 3D structure proteins is rather small, just occupying around 5 percent of all discovered proteins. Secondly, the structures of a certain protein before and after binding to DNA may considerably differ. Thus, using unbound form of protein to predict binding residuals which occur in bound form seems not very reasonable and robust. Last but not least, extracting information from 3D structure consumes a lot of computational power.

Though three-dimensional structure is very informative for DNA-binding proteins residuals detection, its drawbacks are substantial. As consequences, secondary and primary structures seem to be the alternatives. For example, Sun X. et al. in [[9](#Sun09)] [[10](#Sun12)] utilize β-helices and strand in secondary structure of proteins to create hybrid feature. Ahmad S. [[11](#Ahmad05)] and Clote P. [[12](#Clote03)] suggest to use Position Specific Scoring Matrix (PSSM) as a feature vector for machine learning model. Although primary structure is simple, it has a lot of advantages to be considered as a method of choice. First of all, primary structure or sequence based methods do not require any other information, and thus these methods are very flexible to apply to a vast amount of proteins. Secondly, this kind of method requires only evolutionary information derived from PSSM which could be easily created via BLAST program for example.

In the following sections, we present the method to create the new hybrid feature consisting of traditional PSSM and a novel Boolean Significant Identifier which utilize normalized mutual information to detect significant sites. But first of all, we describe how to use framework from [[13](#Gultas12)] to

## 5.2 Doubly Stochastic Matrix (DSM)

Doubly Stochastic Matrix is a square matrix in which the sum of every row or column is equal to one. From majorization theorem of Hardy-Littlewood-Polya [[14](#Hardy52)] stating that the entropy of probability mass function will be increased when transferring through doubly stochastic matrix. Consequently, by transferring empirical amino acids pair of pair distribution of two columns in MSA via DSM, we could penalize those amino acid pairs which do not occur in the original distribution. In the following sub-sections, we use the framework from [[15](#Gultas13)] to create our own specific doubly stochastic matrix.

### 5.2.1 Substitution Matrix

Significant and zero substitution matrices are two components to calculate DSM. From now we will call these matrices as significant and zero matrix for short. Both of them have 400 x 400 dimensions which is sufficient to store occurrences of all amino acids pair of pair. While significant matrix takes into account of significant column pairs, the zero matrix calculate from insignificant column pairs. For more specific, we define a column pair in MSA is significant if and only if both of two columns are away from a certain binding site column at most 5, while insignificant column pair consist of two column sites away each other at most 10 and neither of them is close to binding column site at most 5.

Table : A fragment MSA of 1A0A\_A

|  |
| --- |
| 1 5 10 15 20 25 30 35 40 45 50 |
| MKRESHKHAEQARRNRLAVALHELASLIPAEWKQQNVSAAPSKATTVEAACR  -KRESHKHAEQARRNRLAVALHELSSLIPAEWKQQNVSAAPSKATTVEAACR  -KRESHKHAEQARRNRLAVALHELASLIPAEWKQQNVSAAPSKATTVEAACR  -RRAGHIHAEQKRRYNIKNGFDTLHALIPQ--LQQNPNAKLSKAAMLQKGAD |

For example, from the above MSA of 1A0A\_A, the binding column sites are colored with red and numbered as 15 and 27. Thus, according to the above definition of significant and insignificant, any pair whose index from {10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20} is considered significant. Insignificant pairs are for example (5,6), (1,9), (40, 45), (40,50). Any pair whose column indexes are far away more than 10 such as (39,51) is neither significant nor insignificant.

Let be multiple sequence alignment with rows and and are two columns of . To calculate pair to pair substitution value of these columns to DSM, we deploy the classical concept of computational biology in [[16](#EEyal07)] [[17](#Asper11)]. Let and be two different position in column of and . And in order to avoid reading direction and the sequence ordering in MSA, a pair of pair (where is amino acid at column and row ) is counted four times, as below:

In this substitution calculation, any pair of pair with a gap “-“ will be ignored because it has no meaning in term of compensatory mutation. A 400 x 400 matrix counts pair of pair four times and similarly apply for other column pairs. We denote , be substitution matrix for significant column pair and for insignificant ones respectively.

### 5.2.2 Calculate differential matrix

It is sometimes useful to observe and from another point of view. is indeed a matrix storing amino acids site pairs which occur around binding sites. As we know that the binding sites and surround ones are important for some biological process, thus the matrix stores all the information which could be useful. In the other hand, is a matrix calculating any random nearby column pairs. Our hypothesis is that if a value of entity in is greater than the one in , the corresponding pair of pair with this entity is important. Otherwise it is not important because it just happens by chance.

So, the new matrix is calculated by:

Where iff (same amino acids pair) or

### 5.2.3 Integration of BLOSUM62

BLOSUM62[[6]](#footnote-6) is a 20 x 20 matrix tracking the similarity or dissimilarity among 20 standard amino acids. Two amino acids are considered as similar if its corresponding in BLOSUM62 is non-negative, and dissimilar otherwise.

In next step, we integrate BLOSUM62 matrix to by setting zero value to any similar amino acids pair outside diagonal.

We take into account the amino acids dissimilarity because it play an important role in evolutionary process.

### 5.2.4 Calculate logarithm marginal distribution

Then, in order to neutralize the effect of absolute value in , we calculate a new matrix which is:

Then, we take the logarithm of :

### 5.2.5 Calculate DSM

In the final step, all negative entries in are set to zero, and then we apply iterated column-row normalization [[18](#Cappellini09)] to get the final doubly stochastic matrix.

## 5.3 Significant Residues/Sites Finder (SRF)

We propose a new method to identify the significant or important residues/sites. The residues in protein sequence is supposed to be important if they are DNA-binding sites or nearby to DNA-binding. Overall, we utilize the information theory concept to calculate normalized mutual information between two residues and then using connectivity degree to cut off the threshold.

The workflow of SRF is as following: (1) MSA of protein sequence is created by BLAST program. In this MSA, the first row is the protein sequence that we would like to examine, and the rest of rows derived from BLAST are aligned to the first row. (2) All the distinguished column pairs are retrieved from MSA. We have columns in MSA, thus there are column pairs in total. (3) Normalized mutual information (NMI) of each column pair is calculated. This score ranges from -1 to 1 and enables us know how correlation between these two columns. (4) Among these column pairs, we choose the ones whose NMI belong to top 20 percent highest score. (5) From the high score column pairs, connectivity degrees of these columns/residues are counted. (6) Top ten percent of residues having highest connectivity degree are chosen out and be decided as important sites.

### 5.3.1 Normalized Mutual Information Score (NMI Score)

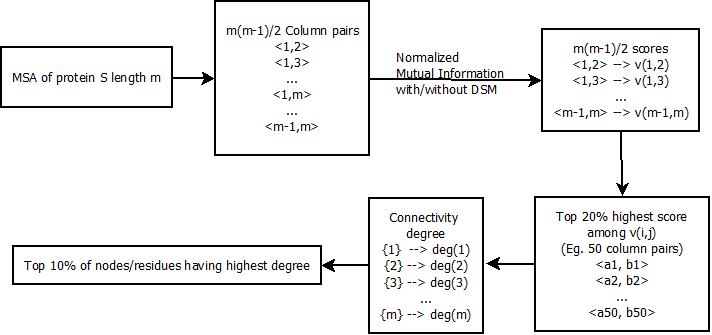


Figure 5.1: Workflow of Significant Residues Finder

In step (3) of the above workflow, one normalized mutual information score is calculated corresponding to one column pair derived from MSA. In this subsection, we will discuss in detail how to calculate that.

Suppose there is a column pair, denoted by where ( is length of sequence as well as number of column in MSA). Let and be two sequences taken from column and and under probability distribution and respectively. Thus, according to formula in Section 3.5, the normalized mutual information score of two columns is defined by:

Where is entropy function mentioned in section 3.

In the above formula, probability distribution and is the marginal distributions of which is fully described by matrix 20 x 20 . Matrix is created through counting the occurrences of amino acids pair in column and , or from the sequence and .

Where , is ith amino acid in 20 standard ones, is from 1 to (length of and ), , is the amino acid at position of and , and is the counting function.

And then the entropy of and could be calculated by:

And then NMI could be derived from H(P) and H(Q) as above formula.

One important thing is that DSM could be integrated into the calculation of NMI. [[15](#Gultas13)] has proven that applying a proper DSM could enhance or reduce NMI score due to its amino acids pair pattern. In order to integrate DSM to NMI, we would integrate it to create new version of matrix . So, let be a fixed DSM, a new matrix is calculated by:

Where is a vector created by concatenating all rows of .

### 5.3.2 Connectivity Degree

The core issue in step (5) of the SRF workflow is how to calculate the connectivity degree of residues. General speaking, the connectivity of a certain residue measures how often this residue appears in the network. Let us take an example, suppose there are four nodes and their connections are (1,2), (1,3), (1,4), (2,3) as figure 5.2.

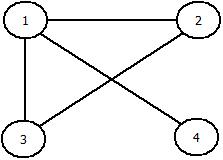


Figure 5.2: Network connection of 4 nodes

By the definition, connectivity of the first, second, third and fourth nodes are 3, 2, 2, 1 respectively.

In this context, after taking those top MNI score column pairs, we could imagine a single column or a residue is a node, and the column pair is the connection of these two nodes.

## 5.4 Feature Extraction

This main goal of this project is to detect the DNA-binding residues from the known DNA-binding protein. In order to extract feature to apply to machine learning framework, we have to extract the feature vector containing informative pattern. We utilize a method called sliding window in which a fixed size window slides on the protein sequence. A window is defined as positive when its middle residue is DNA-binding, or negative otherwise.

W1

MKRESHKHAEQARRNRLAVALHELASLIPAEWKQQNVSAAPSKATTVEAACR

W2

For example, the middle residue of window W1 is E which is non-binding site, so W1 is negative sample, while W2 is positive sample because its middle residue N is binding site.

Our method of choice is hybrid feature which is the concatenation of PSSM and Boolean Significant Residues which will be explained later.

### 5.4.1 PSSM as Feature Vector

Let us denote is target protein sequence with length . The PSSM associated with is a matrix called . We could calculate matrix PSSM through algorithm mentioned in section 3.8, or we could directly retrieve it via BLAST program.

Let denote a certain window by where is a protein sequence of length , and and are the begin and end index in respectively (). Thus, the window’s length is and it is fixed and odd number. Consequently, the matrix corresponding to is the sub-matrix of PSSM defined by:

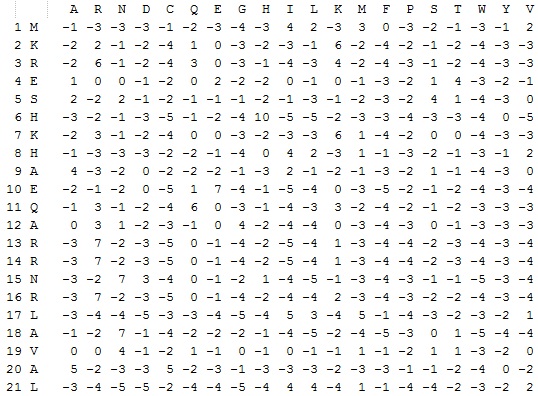


Figure : a sample of PSSM matrix retrieved from BLAST program

Where is a sub-matrix of from row to row and from first column to twentieth column. Then the feature vector associated with this sliding window is the concatenation of all rows in .

### 5.4.2 Boolean Significant Residues as Feature Vector

We utilize the same notation of previous section, the residues in sliding window are . Then we use Significant Residues Finder method discussed in section 5.3 to decide residues are significant or not. Finally, the Boolean feature vector comes along with this window is defined by:

Where is vector length and is the entity in .

Finally, we concatenate feature vector from PSSM and the above vector to create the hybrid feature. This hybrid feature is not only contains the evolutionary information of single residues, but it also has some information of the connectivity degree of residues.

## 5.5 Discussion

# 6. Results and Evaluations

## 6.1 Data Preparation

We utilize two datasets named TR299 and TS75 [[9](#Sun09)] for training and testing respectively. In both datasets, there are protein names as well as their chains in PDB notation standard such as 1A0A\_A where 1A0A is protein name in PDB, associating with chain A. There are 299 and 75 protein sequences in TR299 and TS75 respectively, and they are mutual exclusive.

By using sliding window of length 11 and scan through all 299 protein sequences in training dataset, we retrieve 6291 positive windows corresponding to 6291 DNA-binding residues and 46373 negative windows as well. One sliding window in both positive and negative sets is corresponding to one feature vector of length 231 (11 x 20 from PSSM and 11 from Boolean significant residues finder).

## 6.2 Results & Evaluations

We use TR299 and TS75 as training and testing dataset respectively. In the following subsections, the performance of Significant Residues Finder as well as the prediction of DNA-binding sites will be presented and evaluated.

The performance of predictors will be assessed through some indicators such as sensitivity, specificity and Matthews correlation coefficient (MCC). For example, the confusion matrix is:

|  |  |  |  |
| --- | --- | --- | --- |
|  | | Condition | |
| Condition Positive | Condition Negative |
| Test Outcome | Test Outcome Positive | True Positive (TP) | False Positive (FP) |
| Test Outcome Negative | False Negative (FN) | True Negative (TN) |

First, sensitivity measures the proportion of actual positives which are correctly identified as such.

Second, Specificity measures the proportion of negatives which are correctly identified as such.

One notable thing is that sensitivity and specificity probably are affected a lot by the imbalance between positive and negative in the testing or training set. To compromise the impact from true positive and true negative, Matthews correlation coefficient is considered as the method of choice when the number of positive and negative sample is very imbalance.

### 6.2.1 Performance of Significant Residues Finder

SRF is a method to making a decision of whether a certain residue is important or not in protein chain. A residue is considered important one if it is DNA-binding or nearby. In this context, the nearby distance to binding site is chosen as 5 residues in primary chain. We will show that our approach performs better than happen by chance. However, before evaluating the result from SRF, we would like to recapitulate a little bit of this method’s workflow. As mentioned above, we have to decide the threshold of normalized mutual information score to cut off, and then the cut off value of connectivity degree. These two parameters may heavily impact to the performance of SRF.

Let be the cut off value of mutual information score and connectivity degree respectively. Both of them range from zero to one.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Sensitivity | Specificity | MCC |
|  | 0.434051498 | 0.767475036 | 0.214314492 |
|  | 0.319495533 | **0.840703757** | 0.188568989 |
|  | **0.508145034** | 0.712315739 | **0.225467422** |
|  | 0.456647399 | 0.757489301 | 0.225070184 |

With we seem to get the best performance.

### 6.2.2 Performance of DNA-binding Residues Predictor

By the evaluation from [[9](#Sun09)], we choose the sliding window of length eleven. There is some trade off by choosing the length of window. If we utilize too short window, we may lose some informative context from the neighbor residues. But if we take too wide window, there would probably contain a lot of noisy or irrelative information.

This predictor deploys random forests as classifier due to its suitability for heterogeneously hybrid feature.

# 7. Conclusions

# 8. Discussions

# Bibliography

x

|  |  |
| --- | --- |
| [1] | B. Hansen, L. B. Jorde, "Lecture Notes Biochemistry," *KAPLAN medical*, 2004. |
| [2] | *Protein Data Bank Contents Guide: Atomic Coordinate Entry Format Description*, 320th ed.: wwPDB, 2012. |
| [3] | R. Schneider, C. Sander, "The hssp database of protein structure sequence alignment," *Nucleic Acids Research*, vol. 24, pp. 2019-2022, 1996. |
| [4] | Jin Xiong,.: Cambrige University Press, 2006, ch. 6, pp. 75-76. |
| [5] | Leo Breiman, "Random forrests," *Machine Learning*, vol. 45, no. 1, pp. 5-32, Oct. 2001. |
| [6] | Tjong H., Zhou H. X., "DISPLAR: an accurate method for predicting DNA-binding sites on protein surfaces," *Nucleic Acids Res*, no. 15, pp. 1465-1477, 2007. |
| [7] | Jones S., "Using electrostatic poentials to predict DNA-binding sites on DNA-binding proteins," *Nucleic Acids Res.*, no. 31, pp. 7189-7198, 2003. |
| [8] | Tsuchiya Y., "Structure-based prediction of DNA-binding sites on proteins using the empirical preference of electrostatic potential and the shape of molecular surfaces," *Proteins*, no. 55, pp. 885-894, 2004. |
| [9] | Sun X. et al., "Prediction of DNA-binding residues in proteins from amino acid sequences using a random forests model with hybrid feature," *Bioinformatics*, vol. 1, no. 25, pp. 30-35, January 2009. |
| [10] | et al. Sun X., "Sequence-based Prediction of DNA-binding residues in proteins with conservation and correlation information," *Transactions on computational biology and bioinformatics*, vol. 9, no. 6, pp. 766-775, November 2012. |
| [11] | Sarai A. Ahmad S., "PSSM-based prediction of DNA binding sites in proteins," *BMC Bioinformatics* , vol. 6, no. 33, 2005. |
| [12] | Clote P., "Performance comparision of generalized PSSM in in signal peptide cleavage site and disulfide bond recognition," *Bioinformatics and BioEngineering*, 2003. |
| [13] | et al. Gultas M., "Coupled mutation finder: A new entropy-based method quantifying phylogenetic noise for the detection of compensatory mutations," *BMC Bioinformatics*, vol. 13, no. 25, 2012. |
| [14] | J. Littlewood, G. Polya G. Hardy, "Inequalities," *Oxford University Press*, 1952. |
| [15] | M. Gultas, Ph.D Dissertation: Applying Classical and Quantum Information Theory for Compensatory Mutation Analysis, 2013. |
| [16] | M. Frenkel-Morgenstern, V. Sobolev, S. Pietrokovki E. Eyal, "A pair-to-pair amino acids substitution matrix and its applications for protein structure prediction.," *Proteins: Structure, Function, and Bioinformatics*, vol. 67, no. 1, pp. 142-153, 2007. |
| [17] | R. Y. Asper, "Classifier for Discrimination of Significant Proteins Residues and Protein-Protein Interaction Using Concept of Information Theory and Machine Learning," *Dissertation, Goerg-August-University of Goettingen*, October 2011. |
| [18] | et al. V. Cappellini, "Random bistochastic matrix," *J.Phys.A: Math. Theor*, vol. 42, p. 23, 2009. |

x

1. <http://jameschung3b.edublogs.org/2011/11/28/dna-structure-notes/> [↑](#footnote-ref-1)
2. <http://ameliapsingh.wordpress.com/2013/04/14/duudddeee-its-proteins/> [↑](#footnote-ref-2)
3. <http://www.rcsb.org/pdb/home/home.do> [↑](#footnote-ref-3)
4. <http://blast.ncbi.nlm.nih.gov/Blast.cgi> [↑](#footnote-ref-4)
5. http://www.ncbi.nlm.nih.gov/genbank/TSA.html [↑](#footnote-ref-5)
6. <http://www.ncbi.nlm.nih.gov/Class/FieldGuide/BLOSUM62.txt> [↑](#footnote-ref-6)