DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

LEARNING-BASED APPROACHES IN PROTEIN MOTIF EXTRACTION

by
Çağla ÇINAR

July, 2019
İZMİR

LEARNING-BASED APPROACHES IN PROTEIN MOTIF EXTRACTION

A Thesis Submitted to the
Graduate School of Natural And Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Master of Science in Computer Science, Computer Science Program

by
Çağla ÇINAR

July, 2019
İZMİR

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled "**LEARNING-BASED APPROACHES IN PROTEIN MOTIF EXTRACTION**" completed by **ÇAĞLA ÇINAR** under supervision of **ASSOC. PROF. DR. ÇAĞIN KANDEMİR ÇAVAŞ** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

.....

Assoc. Prof. Dr. Çağın KANDEMİR ÇAVAŞ

Supervisor

Doç. Dr. Emel KURUOĞLU KANDEMİR

Jury Member

Prof. Dr. Urfat NURİYEV

Jury Member

Prof. Dr. Kadriye ERTEKİN

Director

Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENTS

Firstly, I would like to express my sincere gratitude to my supervisor Assoc. Prof. Dr. Çağın KANDEMİR ÇAVAŞ for the valuable guidance she has shown throughout this study, her encouraging attitudes and belief in me have led to the completion of this study.

I would like to thank all my friends who were with me. Special thankfullness to Tolga YAMUT, Gülşah AYDIN and Yusuf Can SEVİL for their motivational support, understanding in this process and their wonderful friendship.

Lastly, I am very grateful to my mother and father for their trust and endless love, which is the greatest support in my life.

Çağla ÇINAR

LEARNING-BASED APPROACHES IN PROTEIN MOTIF EXTRACTION

ABSTRACT

Motif extraction is a challenging problem in bioinformatics. The reason for this arises from the importance in identifiying and predicting the structural and functional regions in the biological sequences. Various statistical methods and machine learning-based approaches could be used for motif extraction.

Neural fuzzy systems are hybrid structures where fuzzy systems have the learning ability of artificial neural networks. Adaptive neural fuzzy inference systems (ANFIS) are integrated systems in which fuzzy concepts are applied in adaptive neural networks.

In this thesis, learning-based approaches are on the motif extraction are investigated. A method has been developed to extract motifs from the Manganese and Iron SuperOxide Dismutase enzyme family. Method has the following stages; selection of the high-frequency short patterns as features, then the generation of motif candidates from the selected features and finally the motif extraction. The train and test data set of the extracted motif was generated by using sum square error. Then the classification performance was tested with ANFIS which is a kind of the neural-fuzzy system.

Keywords: Anfis, bioinformatics, motif extraction, learning-based approaches

PROTEİN MOTİF ÇIKARIMINDA ÖĞRENME-TABANLI YAKLAŞIMLAR

ÖZ

Biyoenformatikte motif çıkarımı zor bir problemdir. Bunun nedeni biyolojik sekanslarda belirli bölgelerin yapısal ve işlevsel olarak tanımlanmasının ve tahminlenmesinin büyük önem taşımasından kaynaklanmaktadır. Motif çıkarımı için çeşitli istatiksel yöntemler ve makine öğrenimi tabanlı yaklaşımlar kullanılabilir.

Sinirsel bulanık sistemler bulanık sistemlerin yapay sinir ağlarının öğrenme yeteneğine sahip olduğu karma yapılardır. Uyarlanabilir sinirsel bulanık çıkarım sistemleri (USBÇS), uyarlanabilir sinirsel ağlarda bulanık konseptlerin uygulandığı entegre sistemlerdir.

Bu tez çalışmasında motif çıkarımında öğrenme-tabanlı yaklaşımlar incelenmiştir. Manganez ve Demir SüperOksit Dismutaz enzim ailesinden motif çıkarmak için bir yöntem geliştirilmiştir. Yöntemin aşamaları şöyledir; frekansı yüksek kısa örüntülerin özellik olarak seçilmesi, daha sonra seçilen özelliklerden motif adaylarının üretimi ve en son motif çıkarımıdır. Çıkarılan motife ait eğitim ve test veri seti toplam karesel hata kullanılarak oluşturulmuştur. Daha sonra, sınıflandırma performansı bir tür sinirsel-bulanık sistem çeşidi olan USBÇS ile test edilmiştir.

Anahtar kelimeler: Usbçs, biyoenformatik, motif çıkarımı, öğrenme-tabanlı yaklaşımlar

CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

CHAPTER ONE
INTRODUCTION

Bioinformatics is the integration of two sciences ”biology” and ”informatics”. It is an interdisciplinary branch of science that consists of computer science, statistics and biology. Such discipline is based on the processing and compiling of the data which is collected and arranged from molecular biology.

Bioinformatic provides many innovations in medicine, pharmacology, agriculture, chemistry and so on. For example, it offers new possibilities for diagnosis and treatment of hereditary diseases in the medical field. For drug discovery, it provides an effective and new methods (Lesk, 2019).

Pauling and Corey’s approach to predict secondary structures of proteins in 1951 may be considered as ground zero for bioinformatics. Along with that, in 1966, the first article of the drawing of moleculer graphs with computers is published in American Scientific Magazine. The term bioinformatics began to be used frequently in the late 1980s and has been carried until today with the development of human genome project is launched in 1987 (Polat & Karahan, 2009).

The first aim of bioinformatics is to arrange the data which is stored in databanks. The second one is developing tools and resources which makes data meaningful and open to interpretation. The last one is analyzing the processes with the produced tools and transferring these processes to information (Ogbe et al., 2016).

Data mining, statistical methods and machine learning techniques have been used for the data analysis. It is very important to determine which method will be more suitable for the interested study.

Various studies in the literature shows that, bioinformatics and machine learning studies become quite widespread recently. Identification of protein family is an one of the important subject of bioinformatic and machine learning studies.

Hong et al. (2009) studied the prediction of protein families in amino acid sequences with support vector machines (SVM). Their results revealed that this process could achieve a high prediction accuracy rate on 20 protein families. Kharsikar et al. (2007) have studied about the similar sequences' gene ontology comments to predict protein function. This study remarked that the weighted K nearest neighbour could produce better results than K nearest neighbour in predicting protein functions.

In bioinformatics, motifs (also known as sequence motifs) are described as the highly conserved regions that are existed in protein or DNA sequences. In other words, motif is a pattern in sequences that are in similar functions with these similar sequences. In bioinformatics, motif extraction has a great importance of identifiying and predicting the structure of a biosequence.

According to the literature, for the protein motif extraction there are several supervised and unsupervised methods. It has been seen that these methods encompass Fuzzy Logic Systems, Support Vector Machine (SVM), Hidden Markov Model (HMM) and other varying clustring algorithms could be performed alone or by combined with each other.

Chang & Halgamuge (2002) have developed a novel approach to extract protein motifs from the same family sequences. A new algorithm is developed to optimize the accuracy of high frequency patterns of C2H2 Zinc Finger and EFG (epidermal growth factor) protein families. The extracted motifs are classified with neuro-fuzzy training. In this thesis, their study is applied for the motif extraction of enzyme family.

There are five chapters in the thesis. In the first chapter, general information and definitions on bioinformatics are given and the literature is reviewed comprehensively. In the second chapter, general terms used in bioinformatics and information about biological databases are given. The third chapter consists of the theoretical background of learning approaches in detail which will be used for motif extraction. Since a neuro-fuzzy method is utilized in this application, artificial neural networks and also fuzzy logic are described in this section. In the fourth chapter,

application section, the steps of extracting motifs from the Manganese and Iron SuperOxide Dismutase enzyme family is referred. The training and test data are obtained as a result of these steps. Then, they are used as ANFIS data set. The testing error is obtained for the determined motif. Finally, the fifth chapter gives the obtained conclusions.

CHAPTER TWO
BIOINFORMATIC THEMES

2.1 Data Types

In order to carry out bioinformatic studies, computer scientists must know the basic terms of molecular biology and also biologists must know the basic terms used by computer scientists in the field of bioinformatics. This is necessary for both groups of researchers to use and analyze biological data and databases. This chapter includes brief information about biological data and databases that are used in bioinformatic studies.

The origin of modern biology is the work of Gregor Mendel in 1865 which is based on the basic rules of heredity. Molecular biology is based on the study of molecules that carry out the vital functions in the cell. In this manner, the cell is the fundamental unit of life in all living organisms (Verma & Agarwal, 2007).

2.1.1 Cell

The cells are two types; as eukaryotic which contain the nucleus and the prokaryotic cells does not have one. The bacteria E. coli is an example of a prokaryote. Eukaryotic cells have more complicated forms, for example, yeasts, plants, and animals. The genetic material is the present form called chromatin or chromosomes in cells. The genome informatics is data-driven like bioinformatics. There are many computational tools are in use. However, they become obsolute while new technologies are improved. (Jiang et al., 2013)

2.1.2 Nucleic Acids

The nucleic acids are complicated biological macromolecules. These molecules control biosynthetic activities in the cell and provide the transferring of hereditary information to the next generations (Garrett & Grisham, 2010).

Living organisms have in two types of nucleic acids which are known as deoxyribonucleic acid (DNA) and ribonucleic acid (RNA). Both kinds are the nucleotides' polymers. Nucleoside and phosphoric acid form a nucleotide. The pentose sugars and nitrogen bases compose the nucleoside. Pentose sugars are Ribose or Deoxyribose and nitrogen bases are purines or pyrimidines. The purines group consists of Adenine and guanine, while the pyrimidines group involves cytosine, thymine and uracil (Verma & Agarwal, 2007).

DNA is formed by a serial of nucleotides. A nucleotide consists of a phosphate group, a ribose (or a deox.. sugar), a single nitrogen-containing base. As aformentioned, nitrogenous bases have two groups where the purines have two fused rings as adenine (A) and guanine (G). Whereas DNA contains A, G, C, and T, whereas RNA contains A, G, C, and U (Pray, 2008). Nitrogenous bases DNA is shown in Figure 2.1.

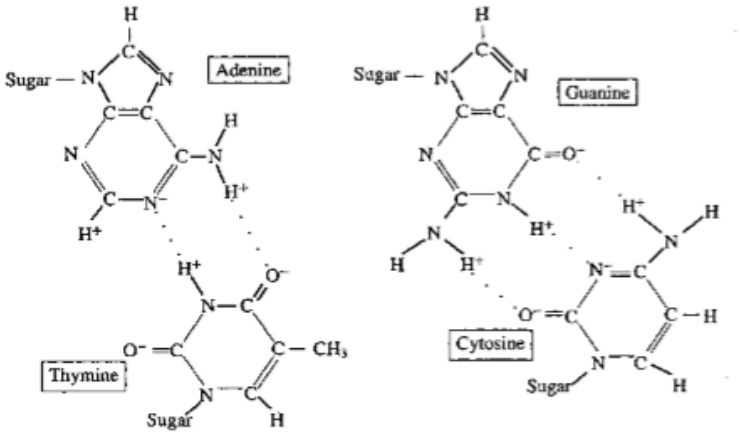

Figure 2.1 DNA nitrogenous bases (Setubal & Meidanis, 1997)

The structure of DNA is a double helix. And it is formed by two long strands. It has

two sites called 3'and 5' (Setubal & Meidanis, 1997).

The phosphate-sugar bonds provide the backbone of strands. Here, the base A always corresponds to the base T; the base G corresponds to the base C. This situation is named base pairing (Jiang et al., 2013). The double helix structure of DNA and an example of the DNA sequence with the letters string are illustrated in Figure 2.2 and Figure 2.3, respectively;

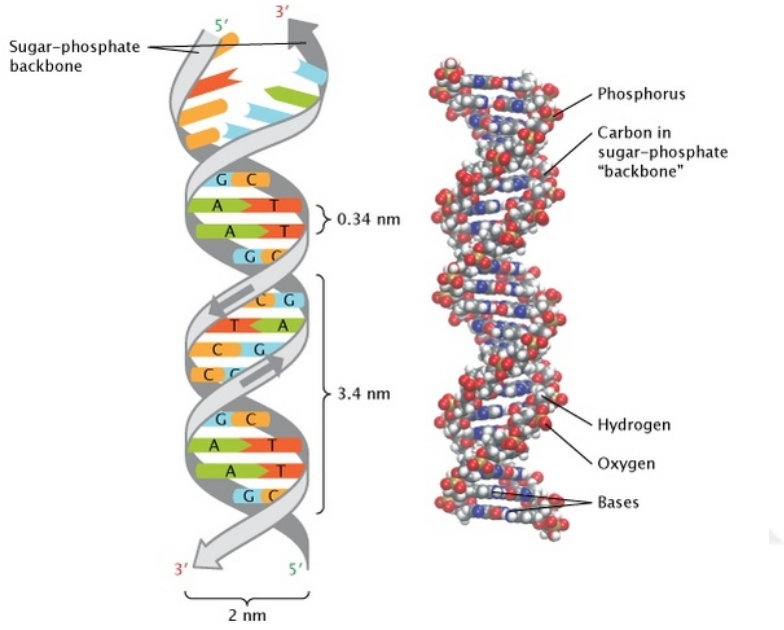

Figure 2.2 Double helix structure of DNA and complementary bases (Pray, 2008)

5'-A T T A C G G T A C C G T- 3'

3'-T A A T G C C A T G G C A-5'

Figure 2.3 A DNA sequence with the letters

After discovering the DNA structure, scientists began to investigate the structure of ribonucleic acid (RNA) for understanding the molecular basis of life. One of the differences among these structures is the type of sugar they involve (DNA has deoxyribose and RNA has ribose). In RNA instead of the base thymine (T), it contains base uracil (U) which makes a bond to the adenine (A). Moreover, RNA has

a variety of forms known as the messenger RNA (mRNA), transfer RNA (tRNA), and ribosamal RNA (rRNA) in the cell. However, altough RNA has different forms its basic structure remains the same across different forms. These different forms are important to serve different tasks in the cell; messenger RNA (mRNA), for instance, plays a role as a pattern in protein production. (Clancy, 2008).

In genetics, the central dogma is that the information in DNA makes its task. Firstly, coded information in DNA sequences passes to the messenger RNA (mRNA). This step is called transcription and the rule of complementary base pairing is valid. Namely, base A in the DNA is transcribed to base U in the RNA. Also from DNA to the RNA, it is transcribed T to A and G to C and C to G (Jiang et al., 2013). The RNA polymerase II enzyme serves in this step and catalyzes the creation of a pre-mRNA molecule. (Clancy & Brown, 2008). Then, the information carried in mRNA is converted to proteins, which is called translation process (Jiang et al., 2013).

Translation occurs on ribosomes. Ribosomes are formed by proteins and rRNA. In translation process, tRNA and rRNA of RNA have a significant task. Transfer RNA (tRNA) molecules serve as molecular adaptors. One side of tRNA binds to mRNA and its other side binds any amino acid. A codon is composed by three bases in mRNA. A codon is called triplet code and define a particular amino acid. The anticodon is the complementary three nucleotides on the tRNA. (Clancy, 2008). Thus using mRNA, sequences of amino acids form a protein.

The initial amino acid of a new composed protein is Methionine (Met corresponds to AUG codon). Methionine (Met) is starting amino acid. Translation ends when all the codons in the mRNA are read by tRNA molecules and the respective amino acids are orderly formed with the polypeptide chain. After that, the translation must be completed. There are three termination codons (UAA, UAG, and UGA) in mRNA for the stopping creation of protein. These codons are not recognized by tRNA. This is the termination of translation (Clancy & Brown, 2008). The initial translation complex is shown in Figure 2.4.

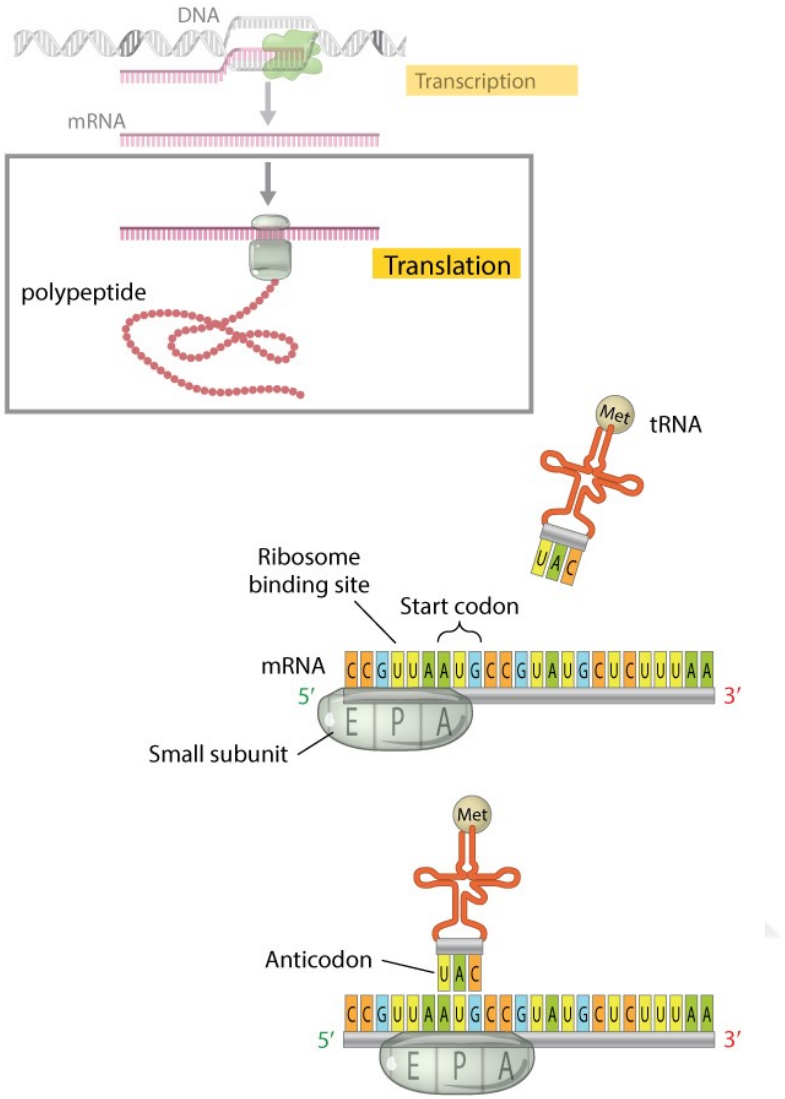

Figure 2.4 The translation initiation complex, (Clancy & Brown, 2008)

2.1.3 Protein

Amino acids are simple molecules. They include a central carbon atom (Ca) which is also known as an alpha carbon. This alpha carbon is attached to a hydrogen atom, an amino group (NH2), a carboxy group (COOH), and a side chain. The difference between the different amino acids is the side chain (Setubal & Meidanis, 1997). The examples of the amino acid is shown in Figure 2.5.

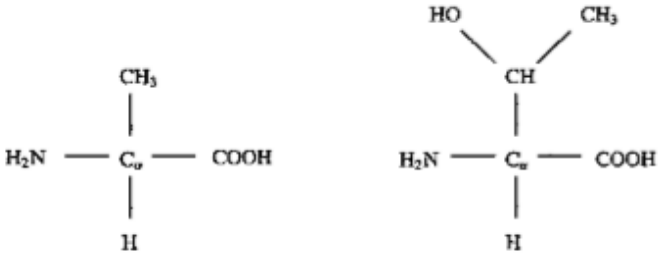

Figure 2.5 Amino acid examples as alanine and threonine (Setubal & Meidanis, 1997)

20 types of amino acids exist in proteins. The 20 different amino acids are given in Table 2.2 with their abbreviations as chemical alphabets.

Table 2.2 Amino Acids (Setubal & Meidanis, 1997)

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

Amino acids are encoded by four bases combination (triplet code). So, 64 ($4^3 = 64$) possible combinations (Smith, 2008).

Because of the amino acids are tied with peptid bonds which are given in table, proteins are polypeptid chains. The starting of the polypeptides is the amino group (N-terminal) and the ending is the carboxy (C-terminal). But proteins are not only

linear sequences. As a result of the combination of amino acids, the primary structure formes, but after passing through multiple stages a three-dimensional structure is revealed (Setubal & Meidanis, 1997).

The four structral types of protein as primary, secondary, tertiary and quaternary is shown in Figure 2.6 (Rashid et al., 2015).

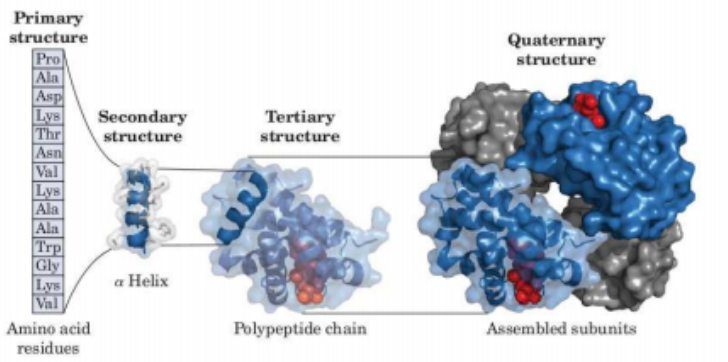

Figure 2.6 Primary, secondary, tertiary and quanternary stuctures of proteins

Proteins in our bodies are different types. Tissues are made by structural proteins, whereas chemical reactions are realized with protein known as enzymes. Enzymes are acting as catalysts of chemical reactions, they speeds up to them. Each biochemical reaction has spesific enzyme. It must be a lot of number reactions to continue our lives so it is clear that we need many enzymes (Setubal & Meidanis, 1997).

2.2 Motif Discovery

Motif means short conserved patterns in DNA or protein sequences which are associated with the different functions of sequences.

Identification of motifs and domains is important because of the characterization of an unknown protein. If the database sequence with a newly obtained sequence does not have significant similarity, the scientists can understand the function of the new sequence based on the identification of the consensus sequences. These consensus sequences are named as motifs and domains. Domain is a conserved pattern which

is an independent functional and structural unit. Motif and domain identification is important to classification of sequences and functional commentary (Xiong, 2006).

Motif discovery can be defined as, finding at different points on the sequences a previously unknown pattern. Therefore, several methods have been developed and the most used approaches for unaligned sequences are;

- Expectation Maximization
- Gibbs Motif Sampling

The consensus sequence information of motifs and domains can be stored in a database for later searches of the existence of similar sequence patterns from unknown sequences. In a query sequence, motifs and domains can be scanned and functions may be discovered but it is not possible in the primary databases by full-length sequences matched. In multiple sequence alignment for motif identification, there are two types of databases which are using regular expressions and statistical models (Xiong, 2006).

2.3 Biological Databases

It is very useful to retrieve biological information from biological databases. Examples of databases; European Molecular Biology Laboratory (EMBL), for Protein Families (Pfam) and National Center for Biotechnology Information (NCBI) (Khandelwal et al., 2017).

Some of the commonly used biological databases are as follows;

- **National Center for Biotechnology Information (NCBI)**: NCBI is founded in 1988. The main tasks are; to create a system that can analyze and store data on molecular biology, genetics, and biochemistry, to facilitate the use of useful

analytical software and databases for scientific communities, to make a worldwide effort to collect biological data and to examine the structure and function relations of key biological molecules (Kaya et al., 2012).

GenBank: GenBank contains the collection of nucleic acid sequence data which contains genomic DNA, mRNA, cDNA, ESTs, high throughput raw sequence data, and sequence polymorphisms. On the other hand, there is GenPept database for the protein sequences. One of the two ways for searching sequences is using Blast and the other one is text-based keywords searching (Xiong, 2006).

European Molecuular Biology Laboratory (EMBL): DNA and protein sequences are provided by The European Molecular Biology Laboratory (EMBL). It contains the literature references of each entered sequences, the information about the function of sequence, the location of the important mutations, mRNA and encoded regions. The EMBL sequence format is similar to GenBank (Kaya et al., 2012).

The Universal Protein Resource (UniProt): The UniProt consortium incorporates of The European Bioinformatics Institute (EBI), the Swiss Institute of Bioinformatics (SIB) and the Protein Information Resource (PIR). For different utilisations, UniProt has four databases: UniProt Archive (UniParc), UniProt Knowledgebase (UniProtKB), UniProt Reference Clusters (UniRef) and the UniProt Metagenomic and Environmental Sequence Database (UniMes) (UniProt Consortium., 2009).

PROSITE: The PROSITE database comprises of a huge set of motifs. These motifs are are biologically meaningful to describe the protein family, domain and to identify the functional site. For description patterns or profiles are used. The conserved regions of protein domains and patterns identified by converted to a consensus expression called regular expression by multiple alignments. The generalized profiles are identified by weight matrices (Sigrist et al., 2012).

2.3.1 Information Retrieval

For biological data, there are different sequence formats for different biological databases usage and programs. The FASTA sequence format, which is one of the most commonly used formats, is described below.

2.3.1.1 Fasta Sequence Format

FASTA format is a combination of the words “fast” and “alignment”. It is used for quick similarity search. This format contains only the sequence, so many programs use FASTA format to read sequences. It does not contain any numbers or signs, this makes it easier to copy the sequence (Kaya et al., 2012).

FASTA format contains a beginning line which contains sequence definitions. All sequences must begin with greater than “>” symbol. The rest part of the line is optional. A “|” symbol between sequence name and number or comments can be available optionally. A bottom line continues through by itself of sequence. An example of FASTA format for two different sequences is shown below;

```
>sp|Q60769|TNAP3_MOUSE Tumor necrosis factor alpha-induced protein 3 OS=Mus musculus OX=10090 GN=Tnfaip3 PE=1 SV=2
MAEQLLPQALYLSNMRKAVKIRERTPEDIFKPTNGIIYHFKTMHRYTLEMFRTCQFCPQF
REIIHKALIDRSVQASLESQKKLNWCREVRKLVALKTNGDGNCLMHAACQYMWGVQDTDL
VLRKALCSTLKETDTRNFKFRWQLESLKSQEFVETGLCYDTRNWNDEWDNLVKMASADTP
AARSGLQYNSLEEIHIFVLSNILRRPIIVISDKMLRSLESGSNFAPLKVGGIYLPLHWPA
QECYRYPIVLGYDSQHFVPLVTLKDSGPELRAVPLVNRDRGRFEDLKVHFLTDPENEMKE
KLLKEYLIVMEIPVQGWDHGTTHLINAAKLDEANLPKEINLVDDYFELVQHEYKKWQENS
DQARRAAHAQNPLEPSTPQLSLMDIKCETPNCPFFMSVNTQPLCHECSERRQKNQSKLPK
LNSKLGPEGLPGVGLGSSNWSPEETAGGPHSAPPTAPSLFLFSETTAMKCRSPGCPFTLN
VQHNGFCERCHARQINASHTADPGKCQACLQDVTRTFNGICSTCFKRTTAEPSSSLTSSI
PASCHQRSKSDPSQLIQSLTPHSCHRTGNVSPSGCLSQAARTPGDRAGTSKCRKAGCMYF
GTPENKGFCTLCFIEYRENKQSVTASEKAGSPAPRFQNNVPCLGRECGTLGSTMFEGYCQ
KCFIEAQNQRFHEARRTEEQLRSSQHRDMPRTTQVASRLKCARASCKNILACRSEELCME
CQHLSQRVGSVAHRGEPTPEEPPKQRCRAPACDHFGNAKCNGYCNECYQFKQMYG
>sp|Q6GQQ9|OTU7B_HUMAN OTU domain-containing protein 7B OS=Homo sapiens OX=9606 GN=OTUD7B PE=1 SV=1
MTLDMDAVLSDFVRSTGAEPGLARDLLEGKNWDVNAALSDFEQLRQVHAGNLPPSFSEGS
GGSRTPEKGFSDREPTRPPRPILQRQDDIVQEKRLSRGISHASSSIVSLARSHVSSNGGG
GGSNEHPLEMPICAFQLPDLTVYNEDFRSFIERDLIEQSMLVALEQAGRLNWWVSVDPTS
QRLLPLATTGDGNCLLHAASLGMWGFHDRDLMLRKALYALMEKGVEKEALKRRWRWQQTQ
QNKESGLVYTEDEWQKEWNELIKLASSEPRMHLGTNGANCGGVESSEEPVYESLEEFHVF
VLAHVLRRPIVVVADTMLRDSGGEAFAPIPFGGIYLPLEVPASQCHRSPLVLAYDQAHFS
ALVSMEQKENTKEQAVIPLTDSEYKLLPLHFAVDPGKGWEWGKDDSDNVRLASVILSLEV
KLHLLHSYMNVKWIPLSSDAQAPLAQPESPTASAGDEPRSTPESGDSDKESVGSSSTSNE
GGRRKEKSKRDREKDKKRADSVANKLGSFGKTLGSKLKKNMGGLMHSKGSKPGGVGTGLG
GSSGTETLEKKKKNSLKSWKGGKEEAAGDGPVSEKPPAESVGNGGSKYSQEVMQSLSILR
TAMQGEGKFIFVGTLKMGHRHQYQEEMIQRYLSDAEERFLAEQKQKEAERKIMNGGIGGG
PPPAKKPEPDAREEQPTGPPAESRAMAFSTGYPGDFTIPRPSGGGVHCQEPRRQLAGGPC
VGGLPPYATFPRQCPPGRPVPHQDSIPSLEPGSHSKDGLHRGALLPPPYRVADSYSNGYR
EPPEPDGWAGGLRGLPPTQTKCKQPNCSFYGHPETNNFCSCCYREELRRRREREPDGELLV
HRF
```

Figure 2.7 Example of FASTA format for 2 sequences

CHAPTER THREE
METHODS

Machine learning is a technology based on developing algorithms that would take decisions or develop solutions for future events. And it uses knowledge or experiences about a past event as the input. A model is developed by using available data sets and the problem specific algorithms. Meanwhile the development, the highest performance is aimed (Kantardzic, 2011).

In machine learning three different models are used as supervised learning, unsupervised learning and reinforcement learning. The first two models are based on whether the data is labeled or unlabeled. There are also systems in which two different models are used together.

Labeled data set is used to define data sets which were previously observed and whose results are known. In supervised learning it is aimed to create a model that includes labeled data and results. Whereas, unsupervised learning is based on uncovering similar structures between data and assumes there is no prior knowledge for the unlabeled data (Han et al., 2011).

Supervised learning is a learning model which provides prediction in classification and regression problems. By using the relationship between inputs and outputs in the training set, future outputs that correspond to future inputs are predicted (Kantardzic, 2011).

As a classification problem that is examined by Jiang et al. (2013), if it is assumed that label Y is the dependent variable and x_i values are expression of genes, the relationship Y and x_i could be learned by training data and then by using this information it can be predicted Y for a novel data for the known x_i's. In this example, to determine and estimate different tumor types or long-lived against short-lived patients by using gene expression data. The supervised learning model as a classification problem, it is assumed that there are two classes labeled as normal and

disease. x_i values are expression of genes and Y label is the dependent variable. The relationship Y and x_i could be learn by training data. By using this information it can be predicted Y for a novel data if x_i's are known.

As it mentioned before supervised learning is used for both classification and regression problems. The regression is used in problems where numeric estimates are made from the data, namely the output is numeric. When category estimation is made, classification is used. Some of the supervised learning algorithms as follows;

- Support Vector Machines (SVM)
- K-Nearest Neighbor (KNN)
- Naive Bayes
- Linear Regression
- Logistic Regression
- Artificial Neural Networks (ANNs)
- Decision Trees

The main goal of the unsupervised learning is to model a fundamental structure. Identifying the hidden patterns is beneficial to extent the understanding about the data. The techniques of unsupervised learning are clustering, association mining, anomaly detection and latent variable.

The steps of the clustering analysis start with the definition of distance measure (similarity) of the objects according to the observed features and follow with the applying selected clustering algorithm. There are two clustering methods; first is based on the criteria or model as K-means and hierarchical clustering, second is based on the model as a statistical mixture (Jiang et al., 2013).

The reinforcement learning model is based on recurring. Trials that produce the best

possible results are rewarded. If the target is not reached, it will be penalized, at this point, it does not repeat the same experiment.

3.1 Artificial Neural Networks

Artificial neural networks (ANN) is an artificial intelligence technology that imitates the human brain.

ANNs are computer systems those are capable of performing the abilities of human brain as creating, developing and discovering a new information. Due to the complexity of the human brain, traditional programming methods found to be not proper or useful for this purpose. Thus, scientists have been developing artificial neural network systems. And it can be applied for following topics; learning, correlation, classification, generalization and feature specification (Öztemel, 2003).

The structure of ANN models the biological neural structure. Before describing the structure of the neural networks, it is useful to describe the structure of the biological neural networks which is shown in Figure 3.1.

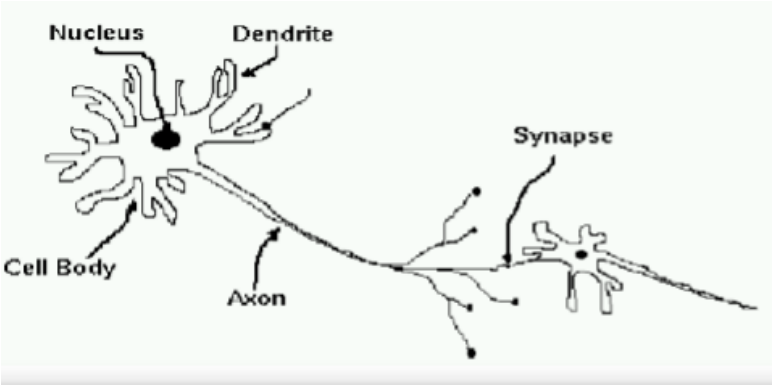

Figure 3.1 A biological neuron (Engelbrecht, 2007)

The basic building unit of the biological nervous system is neurons. Dendrites, axons, nucleus and connection vesicles form the structure of neurons. The nucleus is located in the cell body. Dendrites collect data from other nerve cells and receptors. This data is processed in the cell body. The processed information is transferred to the

dendrites of the other neurons by the connection vesicles in the axon. Synapse is the connection region between neurons.

Artificial nerve cells are similar to biological nerve cells. They are composed of inputs, weights, summation function, activation function and output components. An artificial neuron is shown in Figure 3.2

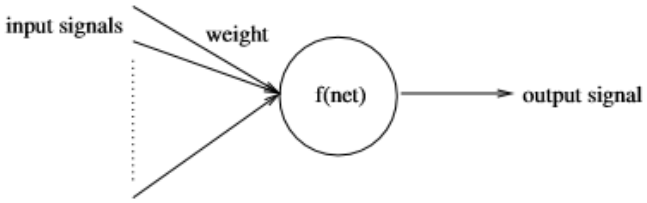

Figure 3.2 An artificial neuron (Engelbrecht, 2007)

Inputs are data which come to neurons from an other cell or outside. These data is sent to nucleus. Weights connect the information from the external environment to the neuron and they determine the importance of related inputs. In the summation function, the weights are multiplying by the inputs. The result is named as net input. Although different functions can be used for calculating net input, generally weighted summation function is used. The summation is determined by the formula as shown in 3.1.

$$N = \sum_{i=1}^{n} x_i w_i \tag{3.1}$$

In activation function the net input is processed and the correspond output is calculated during the process. The activation function can be determined as the most appropriate function as linear, sigmoid and tangent hyperbolic function. The formula of the most commonly used function is sigmoid which is shown below;

$$F(N) = \frac{1}{1 + e^{-N}} \tag{3.2}$$

The output value is determined by the activation function. A nerve cell can have more than one input, but only one output.

In an ANN, the model of the network is determined by topology, summation and activation functions of the process elements, learning strategies, and learning rules.

The most widely used artificial neural networks models are; single layer perceptron, multilayer perceptron, vector quantization models, self-organizing models, adaptive resonance theory models, Hopfield networks, counterpropagation networks, neocognitron networks, Boltzmann machine, Probabilistic networks, Elman network and radial based networks (Öztemel, 2003).

3.1.1 Artificial Neural Networks By Connection Type

3.1.1.1 Single Layer Artificial Neural Networks

As shown in Figure 3.3 the single layer neural networks have the input and output layers.

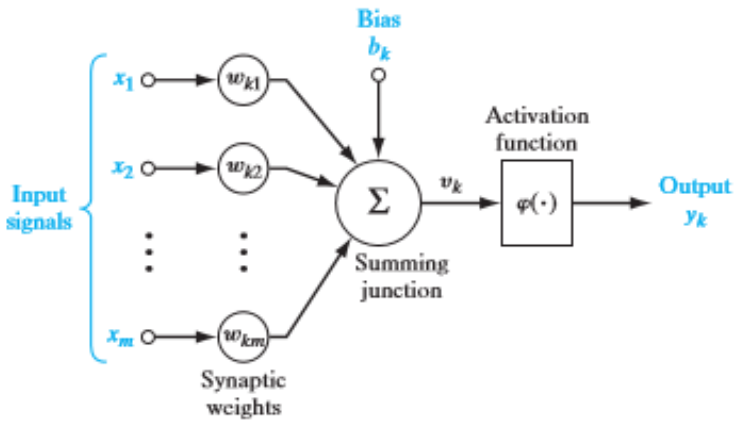

Figure 3.3 A single layer artificial neural network (Haykin, 2009)

The output units are connected to all input units where each connection has a weight. There is a bias value which prevents the 0 value of outputs and units. The summation of threshold value and weighted inputs which passes through weighted activation function gives the output value. The output takes the -1 or +1 value. These values express the classes (Nabiyev, 2012). The output calculation is shown in 3.3.

$$O = f\left(\sum_{i=1}^{m} w_i x_i + \Phi\right) \tag{3.3}$$

3.1.1.2 Multi Layer Artificial Neural Network

Unlike a single layer artificial neural network, it is found the hidden layer in a multi layer artificial neural network model. The number of hidden layers may be one or more, it depends on the problem. In this layer, information from the input layer is processed. A multi layer artificial neural network is shown in Figure 3.4

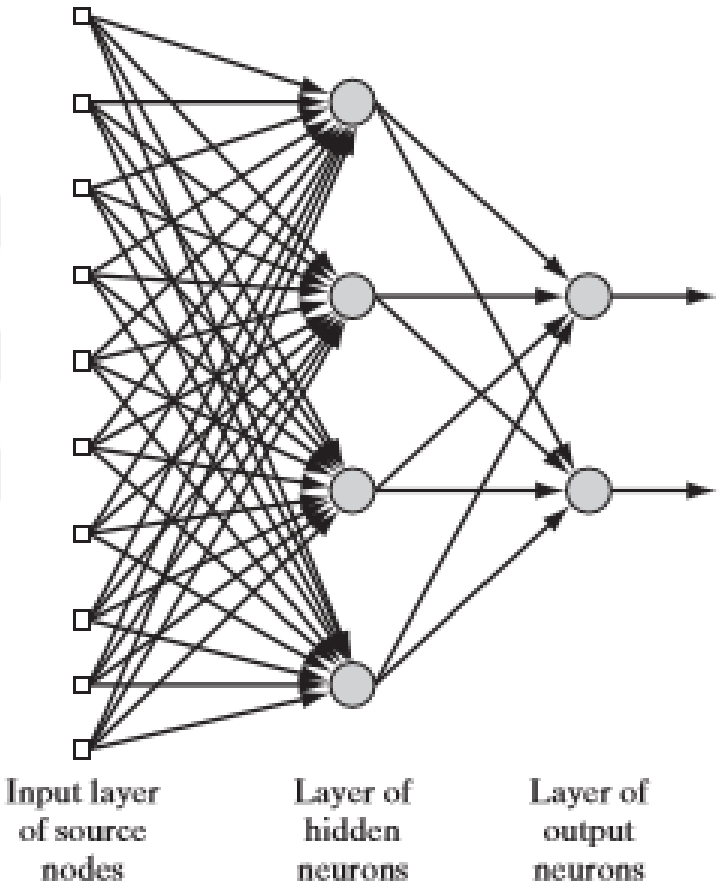

Figure 3.4 The multi layer structure with feed forward (Haykin, 2009)

The multi layer perceptron is the standart type of ANN. It is used in many applications for the analysis of non-linear events. The task of the network is producing the outputs that corresponds to each input. They work according to the supervised learning approaches. The learning rule is a delta rule which has two stages as feedforward and backpropagation (Öztemel, 2003).

3.1.1.3 FeedForward

Each unit in the hidden layer receives the information from the input layer by weighting. The net input is calculated as in the following formula.

$$N_j^a = \sum_{i=1}^{n} x_k^i w_{jk} \tag{3.4}$$

According to the 3.4 the weight of the connection that connects k^{th} input unit to j hidden element is indicated by x, while the output of the process is indicated by x_k^i.

When the sigmoid function is used as the activation function of the j hidden layer, the output is calculated as follows;

$$x_j^a = \frac{1}{(1 + e^{-(N_j^a + \beta_j^a)})} \tag{3.5}$$

β_j is the weight of the bias which depends on j^{th} element in the hidden layer. The value of β_j is determined by the network during the training.

When all output values are found, the calculation of the network is completed correctly.

3.1.1.4 Backpropagation

Back-propagation is adjusted according to the error correction rule. The difference between the desired response and the actual response is considered as error. This error is propagated backward to the weights of the network. The weights are adjusted so that the actual response of the network becomes close to the desired response (Kantardzic, 2011).

The following equation shows the error value in iteration n for the j^{th}unit.

$$e_j n = d_j(n) - y_j(n) \tag{3.6}$$

In 3.6 it is seen that the error value is represented by e, estimated value is by d and actual value is by y. This is the error value for a sinle unit. The equation that computes the error value of whole network is as follows;

$$E(n) = 1/2 \sum_{j=C} e_j^2(n) \tag{3.7}$$

The aim of the back-propagation is the minimize the $E(n)$ value by changing the weights.

$$net_j(n) = \sum_{i=1} w_{ji}(n) x_i(n) \tag{3.8}$$

In 3.8, m is the number of the input for j^{th} neuron. The output of the j^{th} neuron is;

$$y_j = f_j(net) \tag{3.9}$$

After this point, $w_{ji}(n)$ applies adjustments to $\Delta w_{ji}(n)$ by using the chain rule for derivation.

$$\frac{\partial E(n)}{\partial w_{ji}(n)} = \frac{\partial E(n)}{\partial e_j(n)} \cdot \frac{\partial e_j(n)}{\partial y_j(n)} \cdot \frac{\partial y_j(n)}{\partial net_j(n)} \cdot \frac{\partial net_j(n)}{\partial w_{ji}(n)} \tag{3.10}$$

After the derivative calculation in the 3.10 equation, the newly formed equation is;

$$\frac{\partial E(n)}{\partial w_{ji}(n)} = -e_j(n) f_j^{'}(net_j(n)) x_i(n) \tag{3.11}$$

According to weight adjustment by delta rule, the resulting equation is as follows;

$$\Delta w_{ji}(n) = \eta \cdot \delta(n) \cdot x_i(n) \tag{3.12}$$

The η represents the learning rate. It determines how much $E(n)$ value may change by the weight. $\delta(n)$ is defined as local gradient and shows the amount of change in weights.

3.2 Fuzzy Logic

In the classical logic, an entity is either the element of a set or not. This state is expressed by 1 or 0 mathematically. In fuzzy logic, belongingness to a set is not determined by such crisp boundaries. Unlike classical logic, each entity is the element with a certain degree of membership in (0,1) range of a set in fuzzy logic.

The notion of the fuzzy logic is entered the science world by Zadeh in 1965. According to Zadeh, the problem that is the relationship between the elements of a set can not define precisely is, can be eliminated by defining the intermediate values of exist – non exist pairs. Zadeh in his study of 1978, suggests an idea that fuzzy set theory is capable of modeling man's decision-making ability, which makes it easier to understand rather than a mathematical definition. The central concept of the fuzzy theory is fuzzy sets (Civalek, 2003).

3.2.1 Fuzzy Sets

In fuzzy sets belongingness of the elements to a set is determined by the membership degrees. Membership degrees are expressed in real numbers in the range of [0-1]. In this way, the elements fit into smaller or larger values in the cluster indicated by the membership degrees. The membership function of the fuzzy set S with x elements is $\mu_S(x)$ is shown in 3.13 (Ćurić et al., 2011).

$$S = \{(x, \mu_S(x))|x \in X\} \tag{3.13}$$

Some of the methods that form the membership functions as triangular, trapozeidal

and gausian can be seen with the 3.14, 3.15 and 3.16 respectively;

$$triangular(x; a, b, c) = \begin{cases} 0, & x \leq a \\ (x-a)/(b-a), & a \leq x \leq b \\ (c-x)/(c-b), & b \leq x \leq c \end{cases} \tag{3.14}$$

$$trapezodial(x; a, b, c, d) = \begin{cases} (x-a)/(b-a), & a \leq x \leq b \\ 1, & b \leq x \leq c \\ (d-x)/(d-c), & c \leq x \leq d \end{cases} \tag{3.15}$$

$$gaussian(x; c, \sigma) = e^{\frac{-(x-c)^2}{2\sigma}} \tag{3.16}$$

3.2.2 Linguistic Variables

Zadeh (1975) emphasizes the linguistic variables as the words or sentences which take their values from natural language. Examines as “old” is a linguistic term, its value can be “very young”, “young”, “middle age”, “old” “very old”, not numerical values as “15”, “20”, “35”, “40”.

A linguistic variable is denoted as a quintuple $(x, T(x), X, G, M)$. The name of the variable is x, $T(x)$ is the linguistic values of x , X is the universe of discourse, the syntactic rule that generates the terms in $T(x))$ is G. M is a semantic rule. That relates to each linguistic value of L. $M(L)$ is a fuzzy set in X (Jang et al., 1997). Figure 3.5 shows the membership functions of T(age).

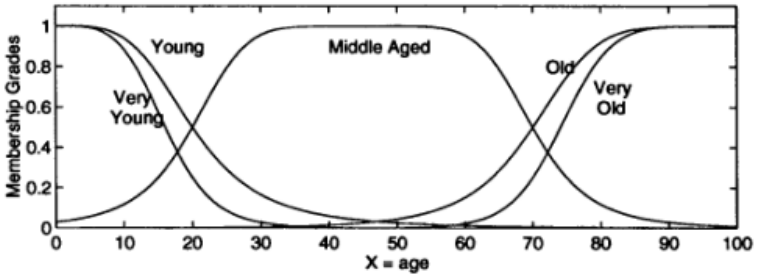

Figure 3.5 Membership functions of T(age) example

3.2.3 If-Then Rules

The fuzzy rules named as fuzzy if then rules or conditional statements and expressed as follows;

IF x is A THEN y is B

According to the expression, the linguistic variables x and y defines the linguistic values of the fuzzy sets A and B. The premise or antecedent is “x is A” and “y is B” is the consequence. The fuzzy if then rules can be used with NOT, AND, OR statements (Czogala & Leski, 2012).

3.2.4 Fuzzy Reasoning

The steps of the approximate or fuzzy reasoning are as follows;

- “Degrees of compatibility: Compare the known facts with the antecedents of fuzzy rules to find the degrees of compatibility with respect to each antecedent MF.
- Firing strength: Combine degrees of compatibility with respect to antecedent MFs in a rule using fuzzy AND or OR operators to form a firing strength that indicates the degree to which the antecedent part of the rule is satisfied.

- Qualified (induced) consequent MFs: Apply the firing strength to the consequent MF of a rule to generate a qualified consequent MF. (The qualified consequent MFs represent how the firing strength gets propagated and used in a fuzzy implication statement.)
- Overall output MF: Aggregate all the qualified consequent MFs to obtain an overall output MF ”(Jang et al., 1997, p. 69).

3.2.5 Fuzzy Systems

Fuzzy systems are named as rule-based or knowledge based systems which has knowledge base and reasoning mechanism. Fuzzy IF-THEN rules are collected in the rule based part. The fuzzy inference system (FIS) combines the rules from the inputs of the system into its outputs by using fuzzy reasoning mechanism (Czogala & Leski, 2012). A fuzzy inference system is shown in Figure 3.6.

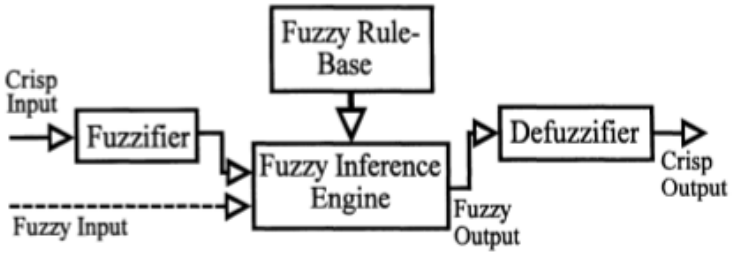

Figure 3.6 Fuzzy inference system (Czogala & Leski, 2012)

The fuzzifier interface takes inputs as crisp values and converts them into appropriate fuzzy sets. The IF-THEN rules are in the rule base. To obtain fuzzy outputs the inference engine is used, and a decision making based on the fuzzy rules is performed. In the defuzzification interface, fuzzy values are converted to crisp values.

A nonlinear mapping is implemented from crisp inputs to outputs in a fuzzy inference system. This mapping is performed by the if-then rules that each of them characterizes the local demeanor of this mapping. A fuzzy region from the input

space is defined by the antecedent of the rule. Consequent of the rule determines the output in the fuzzy region (Jang et al., 1997).

The methods used for defuzzifying are;

- Max-membership method: In this method the highest membership degree is considered as the output value. Max-membership principle for defuzzifying is illustrated by mathematically as below;

$$\mu_{\underset{\sim}{c}}(z^*) \geq \mu_{\underset{\sim}{c}}(z) \text{ for all } z \in Z \tag{3.17}$$

- Weighted avarage method: Weighted Average is based on the weighting each membership function with the largest membership value and shown mathematically as below;

$$z^* = \frac{\sum \mu_{\underset{\sim}{c}}(\bar{z})\bar{z}}{\sum \mu_{\underset{\sim}{c}}(\bar{z})} \tag{3.18}$$

- Centroid method : The center of the gravity under the membership function area is calculated. The expression is shown in below;

$$z^* = \int \frac{\mu_{\underset{\sim}{c}}(z)zdz}{\mu_{\underset{\sim}{c}}(z)dz} \tag{3.19}$$

- Centre of sums: Although this method is similar to the weighted average method, it contains the algebraic sum of the outputs of two fuzzy sets instead of the join. Intersect areas are added twice.

$$z^* = \frac{\int_2 z \sum_{k=1}^{n} \mu_{\underset{\sim}{c}}(z)dz}{2} \tag{3.20}$$

- Mean-max membership method: It is used when the maximum membership is a range, not a single point. the method, in the values with the largest membership, a as the smallest value and the maximum value as b is shown below;

$$z^* = \frac{a+b}{2} \tag{3.21}$$

- Centre of largest area method: For calculating the defuzzification value, with the largest area of the weight of the convex subdivision can be used if fuzzy set has two subdivisions as shown in below equation;

$$z^* = \int \frac{\mu_{\underset{\sim}{cm}}(z) z dz}{\mu_{\underset{\sim}{cm}}(z) dz} \tag{3.22}$$

- First of maxima or last or maxima method: If the maximum value of the membership function value in the fuzzy inference set is not only one value, then the first maximum value is considered as defuzzified value. According to the latest value, the latest value is considered as defuzzified value. (Sivanandam et al., 2007)

The examples of the fuzzy inference systems (FIS) are Mamdani and Sugeno inference systems. In Mamdani inference system, it is anticipated the output functions to be fuzzy sets. The fuzzy sets which are from the consequents of each rule are combined with aggregation processing. The resulting fuzzy set is defuzzified for getting output of the system. In Sugeno, the linear combination of inputs is the consequent of each rule where the output is a linear combination of the consequents.

3.2.5.1 Mamdani Inference System

A fuzzy model of Mamdani type is generated in the following steps;

1- A set of fuzzy rules are specified.

2- The inputs are fuzzified by using the membership functions of them.

3- To set up a rule strong, the fuzzified inputs are combined according to the fuzzy rules

4- The output membership function and the rule strength is combined and the consequence of the rule is determined.

5- For obtaining an output distribution the consequents are unified.

6- This step is applied exclusively when a crisp output is required. The output distribution is defuzzified. (Sivanandam et al., 2007)

A Mamdani fuzzy inference system with two rules and two inputs as x and y, the resulting output Z is shown in Figure 3.7

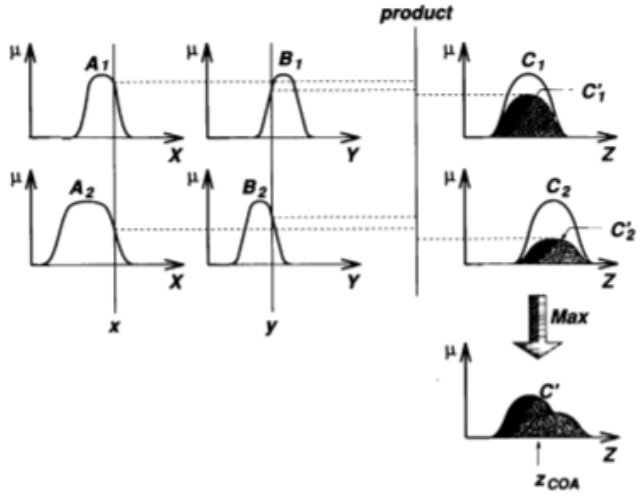

Figure 3.7 Mamdani Inference System (Jang et al., 1997)

3.2.5.2 Takagi-Sugeno Inference System

A Sugeno fuzzy model can be described as shown in below;

$$\text{IF } x \text{ is A and } y \text{ is B THEN } z = f(x, y)$$

According to the model, A and B are the fuzzy sets in the antecedent and defined for x and y membership functions, z = $f(x, y)$ is a crisp function in the consequent. Generally, $f(x, y)$ is a polynomial in the input variables x and y. However, it can be any other function which defines the output of the system appropriately in the fuzzy region. If $f(x, y)$ is a first-order polynomial, it means the model is a first-order Sugeno

model. In the zero-order Sugeno fuzzy model, f is a constant and it is a special case of the Mamdani fuzzy inference model or Tsukamoto's fuzzy model. In the case of, Mamdani inference model, a fuzzy singleton determines the consequent of each rule. According to the second case, a membership function of a step function centered at the constant determines each rule's consequent (Sivanandam et al., 2007).

A Takagi-Sugeno fuzzy inference system with single input is shown in Figure 3.8

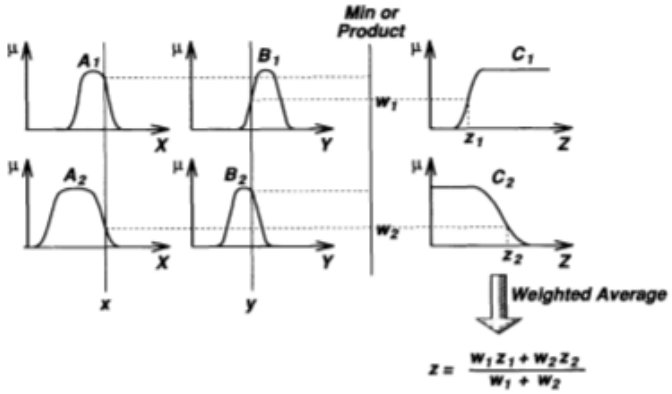

Figure 3.8 Takagi-Sugeno Inference System

3.3 Neuro-Fuzzy Networks

Neuro-fuzzy systems are the systems which developed with neural networks' learning ability and yield this ability to fuzzy systems. Neuro-fuzzy systems combine the advantages of both systems together. The comprehensible outputs are obtained by using the linguistic terms and If-Then rules of fuzzy logic.

In neuro fuzzy systems, the most commonly used neural network type is radial basis neural network, where each node has Gaussian or Ellipsodial functions.

The nodes have the same functionality in standard neural networks. However, nodes have different functionalities in neural fuzzy systems. Some nodes refer to the linguistic terms of input variables ,some nodes refers to the output variables. In neural networks, the nodes in the adjacent layers are completely interconnected. Some nodes

and connections in neural fuzzy networks are used to represent fuzzy rules. A neural fuzzy system should be able to learn or optimize the linguistic rules and membership functions. There may be three cases as follows; (Baykal & Beyan, 2004)

1- The system starts without a rule and the new rule generation is triggered by the current rule base, until the problem is solved.

2- The system starts with rules and evaluates their performance and deletes the ones that are insufficient from the rule base.

3- Rules are replaced by an optimization process during learning.

ANFIS, FALCON, RuleNet, NEFCLASS, NEFCON, NEFPROX, FINEST, GARIC, FuNe are among the most known neural fuzzy systems developed in recent years. The ANFIS model includes the Sugeno type fuzzy system and also uses back propagation. GARIC, NEFCON, NEFCLASS and NEFPROX models use Mamdani type fuzzy inference systems (Elmas, 2007).

3.3.1 ANFIS

ANFIS is a type of neuro-fuzzy network. It is based on the Takagi-Sugeno fuzzy inference model.

3.3.1.1 ANFIS Architecture

Basically Takagi-Sugeno contains at least one rule. The general set of rules for Takagi-Sugeno, which has two fuzzy If-Then rules expressed as the form of;

Rule 1: If x is A_1 and y is B_1, then $p_1x + q_1y + r_1$,

Rule 2: If x is A_2 and y is B_2, then $p_2x + q_2y + r_2$.

A two-input with two rules Sugeno model is shown in Figure 3.9 and its corresponding ANFIS structure is shown in Figure 3.10

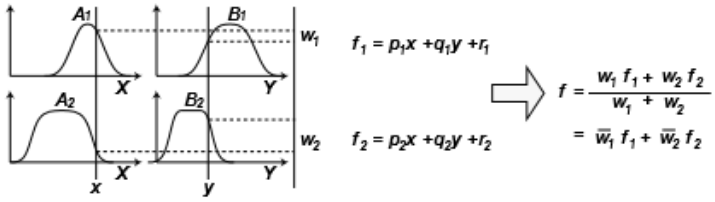

Figure 3.9 Sugeno fuzzy model with two input (Jang et al., 1997)

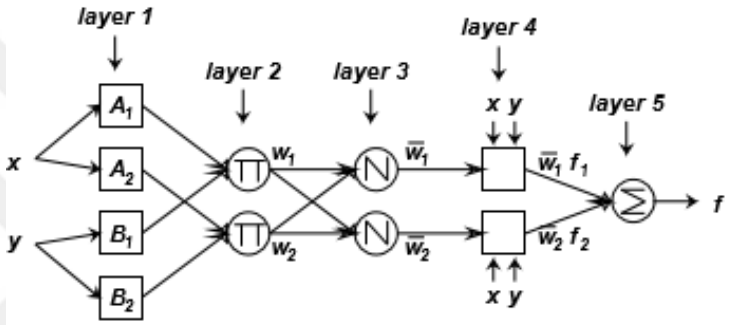

Figure 3.10 ANFIS architecture (Jang et al., 1997)

Operationally, the first step is to find the membership degree of the "If" part. Then the antecedent rules in "Then" part are connected with "AND" processor. Firing strength of each rule is found by using the multiplication. The total output is also obtained by weighted average. In ANFIS architecture, square-shaped nodes are adaptive nodes, and round nodes have fixed or non-fixed parameters. Unlike regular neural networks, there is no weight function in the connections (Baykal & Beyan, 2004). ANFIS structure consists of 5 layers as shown in Figure 3.10. The nodes of the layers functions and the operation of layers are respectively (Jang, 1993).

- Layer 1: The layer of the antecedent parameters called fuzzification layer. The output is defined as follows;

$$O_{1,i} = \mu A_i(x), for\ i = 1, 2\ or$$
$$O_{1,i} = \mu B_{i-2}(y), for\ i = 3, 4 \tag{3.23}$$

In this layer , the adaptive nodes A_i and B_{i-2} have linguistic labels. The output for each i node is the degree of membership function that determines the value of the inputs.

- Layer 2: The second layer is the fixed nodes indicated by Π. The layer is called rule layer. Each node refers to the rules and the number that are generated by the fuzzy logic system. The output of each node is referred as the product of membership degree. The output as follows;

$$O_{2,i} = w_i = \mu A_i(x)\mu B_i(y), i = 1, \tag{3.24}$$

- Layer 3: It is the normalization layer where each node has an N label. Each node accepts all nodes from the rule layer as input values and calculates the normalized firing level of each rule. The outputs of this layer are called as normalized firing strengths. The calculation of normalized is;

$$O_{3,i} = \overline{w_i} = \frac{w_i}{w_1 + w_2}, i = 1, 2 \tag{3.25}$$

- Layer 4: It is the the defuzzification layer which contain the adaptive nodes. The parameters are defined as consequent parameters. Each node is the normalized firing strength of w from layer 3. The output of this layer's th node is calculated as follows;

$$O_{4,i} = \overline{w_i} f_i = \overline{w_i}(p_i x + q_i y + r_i) \tag{3.26}$$

  p_i, q_i and r_i are the parameter set of this node.

- Layer 5: The singular fixed node is the total layer and is denoted by $\sum$. The actual value of the ANFIS system is obtained by summing the output value of each node in layer 4. Output value of the system;

$$O_{5,i} = \sum_i \overline{w_i} f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \tag{3.27}$$

An ANFIS structure with 2 input and 9 rules is shown in Figure 3.11;

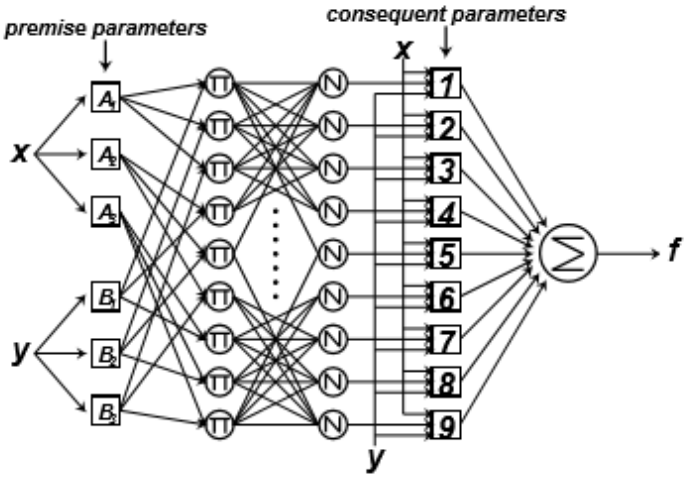

Figure 3.11 ANFIS structure with 2 input and 9 rules (Jang, 1993)

In Figure 3.11 each input is connected with membership functions. Fuzzy if-then rules manage the input part which is subdivided into 9 fuzzy sub-regions. Fuzzy part is represented by the premise part, the output is determined by the consequent part.

3.3.1.2 ANFIS Learning

According to the Figure 3.11 the output value of the system is expressed as follows:

$$\begin{aligned} f &= \frac{w_1}{w_1 + w_2} f_2 + \frac{w_2}{w_1 + w_2} f_2 \\ &= \overline{w_1}(p_1 x + q_1 y + r_1) + \overline{w_2}(p_2 x + q_2 y + r_2) \\ &= (\overline{w_1} x) p_1 + (\overline{w_1} x) r_1 + (\overline{w_2} x) p_2 + (\overline{w_2} y) q_2 + (\overline{w_2} x) r_2 \end{aligned} \tag{3.28}$$

In 3.28 p_1, q_1, r_1, p_2, q_2 and r_2 are the linear consequent parameters. The methods to update these parameters are in the following ways (Jang, 1993);

1) Backpropagation: All parameters are updated with the gradient descent backpropagation.

2) Backpropagation and least-squares method: In order to the consequent parameter to take the initial value, the least-squares method is applied once at the beginning,

and all parameters are then updated with the backpropagation method.

CHAPTER FOUR
APPLICATION

In this chapter, the steps for extracting motifs from sequences belonging to the Manganese and Iron SuperOxide Dismutase enzyme family is described. Prediction performance of the extracted motif is tested by using the ANFIS which is one of the neuro-fuzzy methods.

In the thesis the classification error of a consensus pattern extracted from a protein family is calculated. Chang & Halgamuge (2002) use the technique in order to extract the motif of Zinc Finger protein family.

MATLAB (R2013b version) programming language is used in this study for implementation. The study consists of the following 5 stages;

- ◇ Data Retrieving Step
- ◇ Preprocessing Step
- ◇ Generation of Motifs Step
- ◇ Determining Motif Step
- ◇ Anfis Step

4.1 Data Retrieving Step

In the first phase of this study, the first 300 entries of Manganese and Iron SuperOxide Dismutase (SOD) is chosen from PROSITE as protein family. From the UniProt database the first 301 sequences of Manganese and Iron SOD is selected. Only sequence 301^{th} is used instead of sequence 281^{th}. The 300 entries which contain amino acid letters are downloaded in FASTA format and read in MATLAB R2013b.

4.2 Preprocessing Step

The general representation of a sequence before the preprocesing step, a sequence consists of amino acid characters and gaps. The representation of a sequence is shown below;

$$C_1 - G_{1,2} - C_2 - G_{2,3} - ...G_{N-1,N} - C_N$$

In the above sequence representation, C_1 means the first amino acid character, while the $G_{1,2}$ means the number of gap between the first and the second characters. The gap means the characters that can be any amino acids. A pattern can be rigid or flexible. An example of a rigid pattern is A$-x(3)-$H. The number of $x(3)$ refers to the gap number. In other words, the number of gaps between A and H amino acids is determined by definite limit as 3. However, the flexible pattern gap number is shown in the indicated lower and upper limits. As an example, considering the ATGHHHCLM and ATGAAAACLM patterns, both provide the ATG$-x(1,4)-$CLM condition. It means the number of amino acids may be from 1 to 4 between ATG and CLM patterns. Another representation of this pattern is A$-x(O)-$T$-x(O)-$G$-x(1,4)-$C$-x(O)-$L$-x(0)-$M.

The preprocessing step aims to select the features of the existing sequences, so to filter out low-frequency features. For this purpose,initially an array is created and the features from the array is selected.

4.2.1 Array Creation

The purpose of the array creation which is the first stage of preprocessing step, is to determine whether an array element exists within the sequences. As an example, for two different sequences ATGCLM and ATGCM, G$-x(1)-$M array element exists in the second sequence but not in the first sequence.

As Chang & Halgamuge (2002) emphasized, since there are 20 different amino acid

letters, there can be one of the 20 amino acid letters for the first and second amino acids location. The array creation is developed to have a maximum of 20 gaps between two possible amino acids that may appear side by side. Because of the minimum number of spaces that can be created in the array is considered to be 1, the size of the vector is $20 \times 20 \times 20$, that is, 8000.

In the implementation, it is examined that whether each element of the created array is in a sequence or not. This is done for 300 sequences. The existing array element is determined as 1, and the non existing array element is determined as 0.

4.2.2 Feature Selection

For the feature selection, the threshold value is set at 90%. Depending on the previous step, the frequency is calculated for the each of the elements in the created array. In the implementation, a total of 159 elements which has a higher frequency than the threshold value is selected as features.

4.3 Generation of Motifs Step

The features selected in the previous step are considered while generating the motif. The method of generation is performed by linking these selected features with each other. As an example of this method is, A$-x(1)-$G and G$-x(2)-$ S are two features that start and end with the same amino acid characters. A$-x(1)-$G is connected with G$-x(2)-$S, and the result gets A$-x(4)-$S. At this point, it is examined whether A$-x(4)-$S is in the selected features. If A$-x(4)-$S is one of the features, A$-x(1)-$G$-x(2)-$S becomes one of the generated motifs. Afterward, it is examined the features that the same starting amino acid letter with the last amino acid letter of the generated motif. For instance, if S$-x(2)-$ T is a feature in the selected features, it is connected with the previous generated motif A$-x(1)-$G$-x(2)-$S. At this point, unlike the previous situation, the A$-x(1)-$G$-x(2)-$S$-x(2)-$T motif is

not generatable even if A-$x(7)$−T is in features. The condition required for generating the A−$x(1)$−G−$x(2)$−S−$x(2)$−T motif is both A−$x(7)$−T and G−$x(7)$−T features have to be within the selected features.

By the method described above with the example, in the implementation the number of the motifs generated is 247.

4.4 Determining Motif Step

The aim of this step is to extract a determined motif as a primer pattern. For this purpose, the frequencies of the generated motifs in the previous step are calculated in 300 sequences.

Frequency values are found by matching. The Knuth-Morris Pratt which is a kind of string matching algorithm has been used for sequence matching (Cinar & Kandemir-Cavas, 2016). The threshold value is determined as 80%. The longest of the motifs whose frequency value is higher than the threshold value is determined as the primer pattern.

Short motifs are considered to be more common than long motifs in sequences. The idea behind the selection of the longest pattern is, compensating the accuracy of the classification. The first 25 determined motifs with frequency values that are greater than thereshold values are given in Table 4.2 (the '*'character represents a gap).

Table 4.2 The first 25 determined motifs with frequencies

	Primer Pattern Candidates	Frequency
1	A**P**********H**Y	0.81
2	A*Y****N	0.923333333
3	A**********H*******Y	0.85
4	A**P**********H**Y	0.81
5	A*Y****N	0.923333333
6	A**********H*******Y	0.85
7	A**********H***H***Y	0.846666667
8	A**********H****H**Y	0.823333333
9	L*****E********N	0.813333333
10	A***************H***Y	0.846666667
11	A****************H**Y	0.83
12	G****************G*G	0.92333333
13	G****************G**W	0.94
14	G**W**L	0.936666667
15	G****************S*W	0.92
16	L*****A**P*********H	0.833333333
17	L*****A**********H	0.85
18	L********A**P	0.83
19	L**L*****A**P	0.806666667
20	L**L********P	0.863333333
21	L******L*P*********H	0.87
22	L******L**********H	0.873333333
23	L*********L*P	0.87
24	L**********K****Y	0.803333333
25	L*****E***Y****N	0.806666667

The extracted primer pattern is G$-x(17)-$G$-x(2)-$W with the 0.94 frequency.

4.5 ANFIS Step

In this part of the study, the data set is created for training and testing. Afterwards, the testing performance is calculated and optimized with ANFIS.

4.5.1 Training and Testing Data

According to primer pattern, all patterns as possible patterns are extracted from all sequences. The first step to extract possible pattern is locating the first character of primer pattern in the sequence, then the other characters are checked respectively. The parts up to the point where each character matches are assumed gap. In other words, the letter characters of the primer pattern are matched to the same letter characters in the sequence part, excluded parts are assumed to be gap. This part is extracted from the string and is named as possible pattern. This process continues with the ending of the first character of the primer pattern in all the sequence.

Example 4.5.1. *As an example if it is assumed that sequence 1 is CGLAGHKHGLCGAT and primer pattern is* $G{-}x(2){-}G{-}x(1){-}G$*, in this case the possible patterns of the sequence for the primary patterns are;*

1) $G{-}x(2){-}G{-}x(3){-}G$ *as GLAGHKHG*

2) $G{-}x(3){-}G{-}x(2){-}G$ *as GHKHGLCG*

In application, after finding all possible patterns the sum square error is calculated. The possible pattern is chosen as the almost similar pattern which has the smallest sum square error. The square of the number of gaps between the two amino acid characters in the possible match and the number of gaps between the two amino acid characters in possible match difference is taken as a basis for calculating the sum square error.

Example 4.5.2. *According to the Example* $4.5.1$ *, the sum square errors of the possible patterns of sequence 1 is;*

1) $(2-2)^2 + (3-1)^2 = 4$

2) $(3-2)^2 + (2-1)^2 = 2.$

Possible pattern 2 is determined as the most likely pattern because it has the smallest sum square error. The gap numbers and the sum square error of the most likely pattern

are used to generating the data set. According to the above example 4.5.2, the gap numbers (3,2) of the most likely pattern are selected as input values in the data set. In the data set, the inverse of sum square error is determined as output values. If sum square is equal to 0 then the output values are determined as 1. Since there are 300 sequences in the protein family, a 300×3 size data set is created. 80% of the data set is used as training data and 20% as test data.

4.5.2 Classification

In this section, MATLAB Anfis Editor is used. As it is mentioned before, 60 data are separated from dataset for testing. 240 of 300 data is uploaded to the Anfis Editor to train.

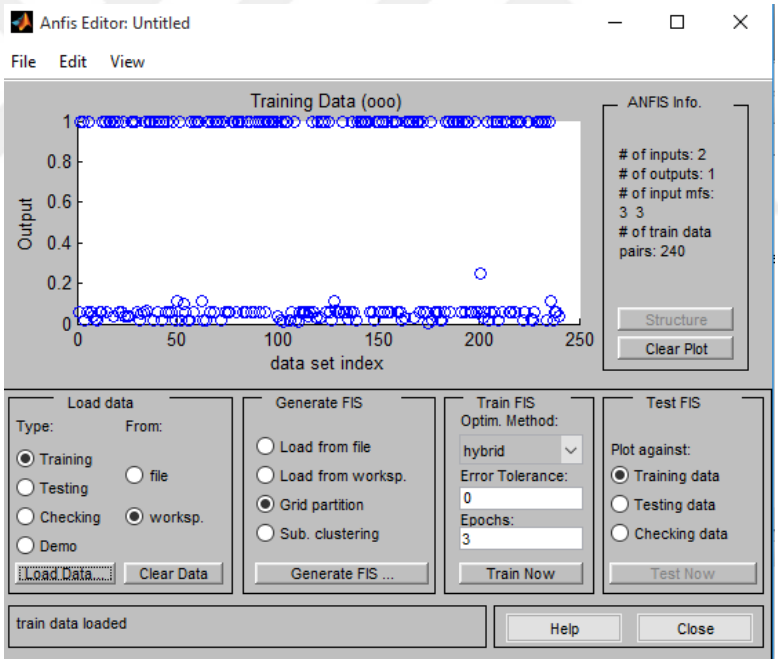

Figure 4.1 Loading train data to ANFIS

While fuzzy inference system (FIS) is being generated, how many inputs will be and the membership functions of these inputs, also the rules will be formed by these membership functions are determined according to the primer pattern. If the primer pattern has three amino acid characters, then a model with only 2 inputs is used for $G_{1,2}$

and $G_{2,3}$. The rules are created based on these gap numbers. The output is determined by "IF G_1 is μ_1 and G_2 is μ_1 THEN output is μ_c" rule.

The FIS is generated with 2 inputs for the resulting primer pattern G$-x(17)-$G$-x(2)-$W. As an example for designing a two-input fuzzy inference system with only one membership function is shown in Figure 4.2 and Figure 4.3 which shows the membership functions created for the gap numbers of the primer pattern respectively.

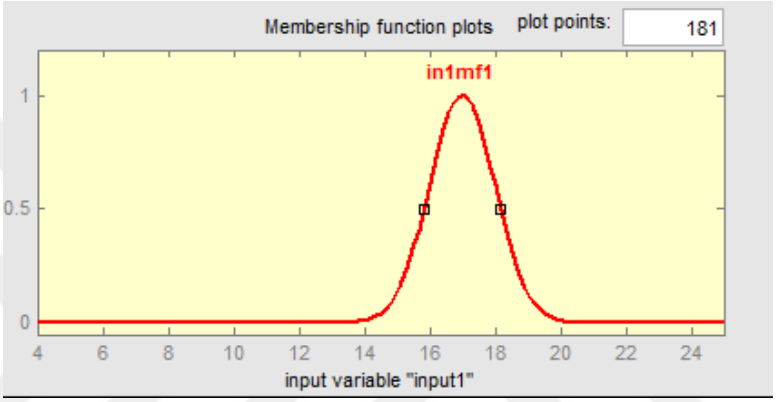

Figure 4.2 The membership function of the primer pattern according to G_1

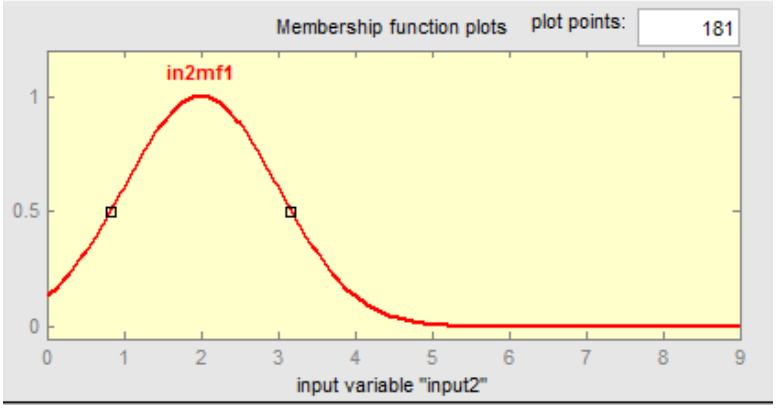

Figure 4.3 The membership function of the primer pattern according to G_2

It is observed that the testing error decreases as the number of rules increases. And 5 membership functions is assigned for each input. The membership functions generated for two inputs are shown respectively in Figure 4.4 and Figure 4.5.

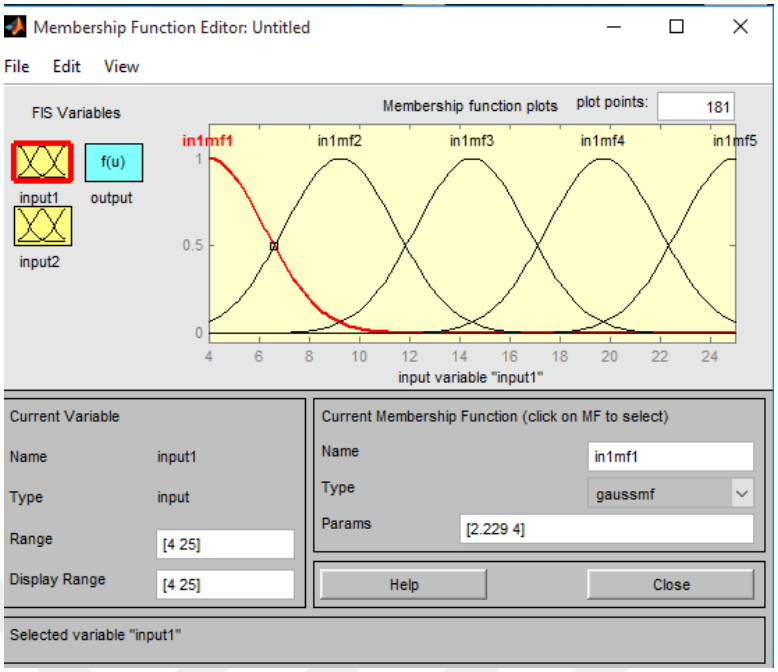

Figure 4.4 The membership functions for the first input

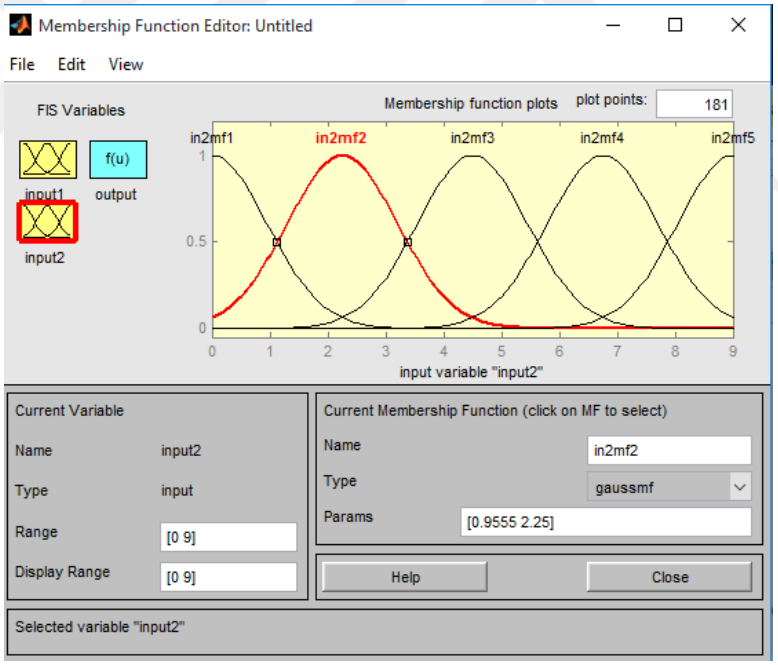

Figure 4.5 The membership functions for the second input

The rule editor with 25 rules according to the 5 membership functions and the Anfis model structure is shown in Figure 4.6 and Figure 4.7.

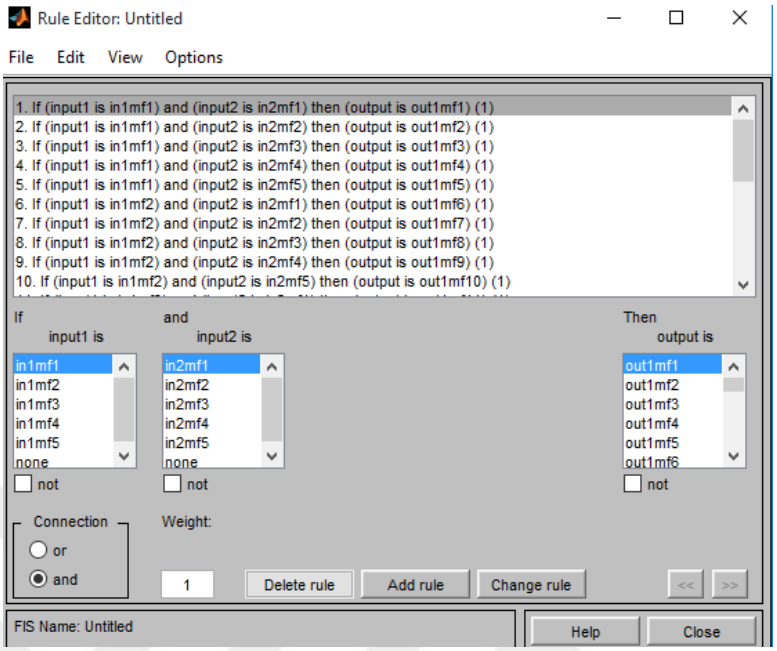

Figure 4.6 The rules of ANFIS

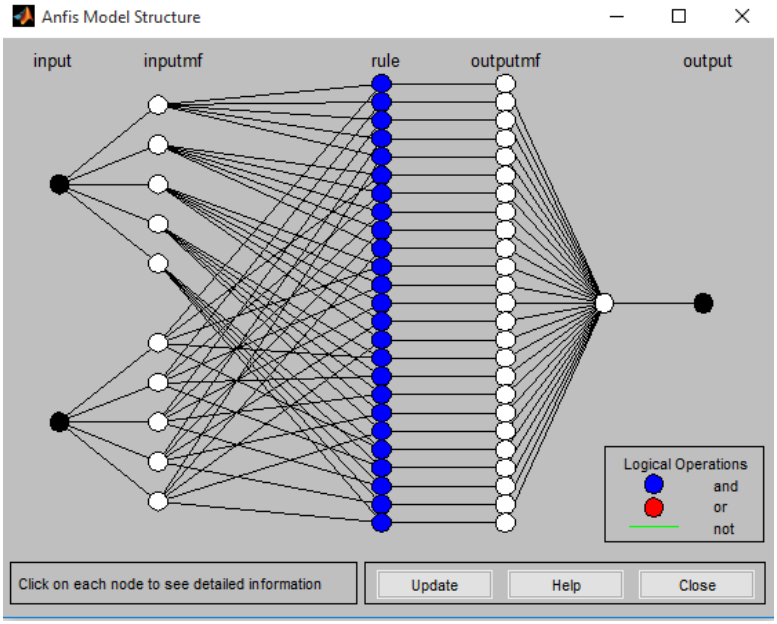

Figure 4.7 Anfis model structure

The epoch number is determined as 40 by the end of some trials. After FIS is trained by hybrid method, the training error is shown in Figure 4.8.

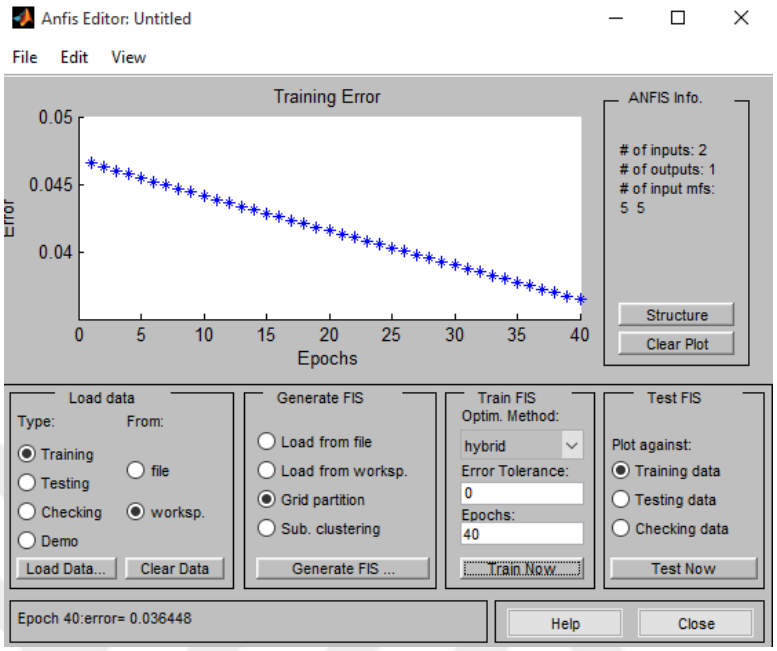

Figure 4.8 Training error

The 60 test data which are separated at initially are upload to Anfis Editor and the average testing error is calculated as 0.023034 and shown in Figure 4.9

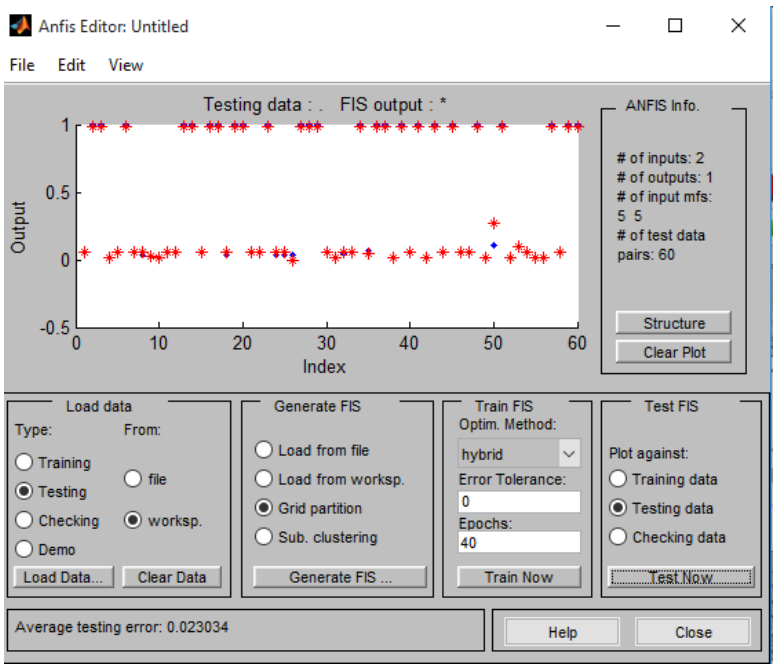

Figure 4.9 Testing error

As it is mentioned before, according to some trials it is observed that if number of the rules are increases the error is decreases. This was observed by trying out the number of various membership functions and epoch numbers. Table 4.4 shows the error values corresponding to the number of various membership functions (Mfs).

Table 4.4 Error values corresponds to alternative Mfs and epoch numbers

Mfs Num-Type of Input1	Mfs Num-Type of Input2	Epoch Number	Testing Error
3 (Gaussian)	3 (Gaussian)	30	0.11001
3 (Gaussian)	3 (Gaussian)	40	0.10499
4 (Gaussian)	4 (Gaussian)	40	0.10101
5 (Gaussian)	5 (Gaussian)	30	0.025299
5 (Gaussian)	3 (Gaussian)	40	0.023034

CHAPTER FIVE
CONCLUSION

Motifs are conserved regions which can be found in different locations in biosequences. This gives us functional and structural knowledge about the protein or DNA/RNA family. At the same time it provides the classification of these biosequences. Motif extraction problem is the emerging process of the special patterns. A variety of statistical methods and learning-based approaches are available for motif extraction.

The concepts of this thesis include learning-based approaches as neuro-fuzzy which combines artificial neural networks and fuzzy logic. Initially, the preprocessing step was performed in the Manganese and Iron SOD enzyme family and the features were extracted. The possible motifs were determined by adding the extracted features according to the similarity of the first and last amino acid. The frequency of occurrence of these possible motifs in all sequences within the enzyme family was calculated. String matching algorithms are used for frequency calculation (Cinar & Kandemir-Cavas, 2016). A threshold value was determined and a certain portion of the selected features were filtered according to this threshold value. Values with a frequency above the threshold are determined as primer motif candidates. Among the primer motifs, the longest motif whose frequency value is above the threshold value was extracted as the primer motif. Similar patterns to the primer motif were extracted from each sequence and sum square error was calculated with the difference of gap numbers of the primer motif with these similar patterns. Among the similar patterns, the pattern with the least sum square error was determined as the most likely pattern. The data set to be tested and trained with ANFIS was generated using the number of gaps between the amino acid characters of the most likely motifs.

In the ANFIS part of the study, the model is generated with 2 inputs and 1 output type. The number of memberships was tested with different numbers as 3, 4, 5. The type of all membership functions for both inputs determined as Gaussian type, and the output as constant. It was observed that as the number of membership functions

increases the accuracy rate has increased, that is, the error rate has decreased. The test error value was found as 0.023034 in the model with two inputs with 5 membership functions.

The other factors that determine the performance of the model are epoch number, input membership function types, output membership types as constant/linear and membership function parameters.

In bioinformatics researches, the experimental solutions are more time-consuming and costly than algorithms. Algorithmic approaches provide faster and more reliable results. In this thesis, it is seen that neuro-fuzzy method used in motif extraction problem is very successful and fast in terms of performance in optimizing classification accuracy.

REFERENCES

Baykal, N., & Beyan, T. (2004). *Bulanık mantık: uzman sistemler ve denetleyiciler*. Ankara: Bıçaklar Kitabevi.

Chang, B. C., & Halgamuge, S. K. (2002). Protein motif extraction with neuro-fuzzy optimization. *Bioinformatics*, *18*(8), 1084–1090.

Cinar, C., & Kandemir-Cavas, C. (2016). On the string matching algoritms in sequence analysis. In *1st International Mediterranean Science and Engineering Congress (IMSEC 2016),October 26-28*, (5698).

Civalek, Ö. (2003). Yapay zeka-Ömer Civalek'le söyleşi. *TMH-Türkiye Mühendislik Haberleri*, *423*(1), 40–50.

Clancy, S. (2008). Chemical structure of RNA. *Nature Education*, *1*(1), 223.

Clancy, S., & Brown, W. (2008). Translation: DNA to mRNA to protein. *Nature Education*, *1*(1), 101.

Ćurić, V., Lindblad, J., & Sladoje, N. (2011). Distance measures between digital fuzzy objects and their applicability in image processing. In *International Workshop on Combinatorial Image Analysis*, (385–397). Springer.

Czogala, E., & Leski, J. (2012). *Fuzzy and neuro-fuzzy intelligent systems, 47*. Heidelberg; New York: Physica-Verlag.

Elmas, Ç. (2007). *Yapay zeka uygulamaları:(yapay sinir ağı, bulanık mantık, genetik algoritma)*. Ankara: Seçkin Yayıncılık.

Engelbrecht, A. P. (2007). *Computational intelligence: an introduction*. West Sussex, England: John Wiley & Sons.

Garrett, R. H., & Grisham, C. M. (2010). *Biochemistry, Fourth Edition*. Boston,USA: Brooks/Cole, Cengage Learning.

Han, J., Kamber, M., & Pei, J. (2011). *Data mining: concepts and techniques*. USA: Elsevier.

Haykin, S. (2009). *Neural networks and learning machines*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc.

Hong, H., Hong, Q., Perkins, R., Shi, L., Fang, H., Su, Z., Dragan, Y., Fuscoe, J. C., & Tong, W. (2009). The accurate prediction of protein family from amino acid sequence by measuring features of sequence fragments. *Journal of Computational Biology*, *16*(12), 1671–1688.

Jang, J. S. (1993). Anfis: adaptive-network-based fuzzy inference system. *IEEE transactions on systems, man, and cybernetics*, *23*(3), 665–685.

Jang, J. S. R., Sun, C. T., & Mizutani, E. (1997). *Neuro-fuzzy and soft computing - a computational approach to learning and machine intelligence*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc.

Jiang, R., Zhang, X., & Zhang, M. Q. (2013). *Basics of Bioinformatics: Lecture Notes of the Graduate Summer School on Bioinformatics of China*. Berlin Heidelberg: Springer-Verlag.

Kantardzic, M. (2011). *Data mining: concepts, models, methods, and algorithms*. Hoboken, New Jersey: John Wiley & Sons.

Kaya, H., Soya, S., Akkale, C., & Tanyolac, B. (2012). *Biyoteknoloji ve Biyoinformatik*, (35).

Khandelwal, I., Sharma, A., Agrawal, P. K., & Shrivastava, R. (2017). Bioinformatics database resources. In *Library and Information Services for Bioinformatics Education and Research* (45–90). IGI Global.

Kharsikar, S., Mugler, D., Sheffer, D., Moore, F., & Duan, Z.-H. (2007). A weighted k-nearest neighbor method for gene ontology based protein function prediction. In *Second International Multi-Symposiums on Computer and Computational Sciences (IMSCCS 2007)*, (25–31). IEEE.

Lesk, A. (2019). *Introduction to bioinformatics*. New York, United States of America: Oxford University Press.

Nabiyev, V. V. (2012). *Yapay zeka: insan-bilgisayar etkileşimi*. Ankara: Seçkin Yayıncılık.

Ogbe, R. J., Ochalefu, D. O., & Olaniru, O. B. (2016). Bioinformatics advances in genomics-a review. *International Journal of Current Research and Review*, *8*(10), 5.

Öztemel, E. (2003). *Yapay sinir ağlari*. İstanbul: Papatya Yayincilik.

Polat, M., & Karahan, A. (2009). Multidisipliner yeni bir bilim dalı: biyoinformatik ve tıpta uygulamaları. *SDÜ Tıp Fakültesi Dergisi*, *16*(3), 41–50.

Pray, L. (2008). Discovery of DNA structure and function: Watson and Crick. *Nature Education*, *1*(1), 100.

Rashid, M. A., Khatib, F., & Sattar, A. (2015). Protein preliminaries and structure prediction fundamentals for computer scientists. *arXiv preprint arXiv:1510.02775*.

Setubal, J., & Meidanis, J. (1997). *Introduction to computational molecular biology*. Boston: PWS Publishing Company.

Sigrist, C. J., De Castro, E., Cerutti, L., Cuche, B. A., Hulo, N., Bridge, A., Bougueleret, L., & Xenarios, I. (2012). New and continuing developments at PROSITE. *Nucleic acids research*, *41*(D1), D344–D347.

Sivanandam, S., Sumathi, S., Deepa, S. et al. (2007). *Introduction to fuzzy logic using MATLAB*. Berlin: Springer.

Smith, A. (2008). Nucleic acids to amino acids: DNA specifies protein. *Nature Education*, *1*(1), 126.

UniProt Consortium. (2009). The universal protein resource (uniprot) in 2010. *Nucleic acids research*, *38*(suppl_1), D142–D148.

Verma, P., & Agarwal, V. (2007). *Cell biology, genetics, molecular biology, evolution and ecology*. New Delhi: S. Chand & Company Limited.

Xiong, J. (2006). *Essential bioinformatics*. New York: Cambridge University Press.