ÇUKUROVA UNIVERSITY
INSTITUTE OF NATURAL AND APPLIED SCIENCES

PhD THESIS

İrem ERSÖZ KAYA

COMPUTATIONAL PREDICTION OF DISORDERED REGIONS IN PROTEINS

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

ADANA, 2008

ÇUKUROVA ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

COMPUTATIONAL PREDICTION OF DISORDERED REGIONS IN PROTEINS

İrem ERSÖZ KAYA

DOKTORA TEZİ

ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

Bu tez 09/06/2008 Tarihinde Aşağıdaki Jüri Üyeleri Tarafından Oybirliği/~~Oyçokluğu~~ İle Kabul Edilmiştir.

Yrd. Doç. Dr. Turgay İBRİKÇİ DANIŞMAN	Prof. Dr. Seyhan TÜKEL ÜYE	Prof. Dr. Fikret GÜRGEN ÜYE
Prof. Dr. Hamza EROL ÜYE		Yrd. Doç.Dr. Ulus ÇEVİK ÜYE

Bu tez Enstitümüz Elektrik-Elektronik Mühendisliği Anabilim Dalında hazırlanmıştır.

Kod No

Prof. Dr. Aziz ERTUNÇ
Enstitü Müdürü

Bu Çalışma Çukurova Üniversitesi Bilimsel Araştırma Projeleri Fonu ve kısmen TÜBİTAK-EEEAG Tarafından Desteklenmiştir.

Proje No: MMF2006D12 & TUBITAK – TBAG104T505

ABSTRACT

PhD THESIS

COMPUTATIONAL PREDICTION OF DISORDERED REGIONS IN PROTEINS

İrem ERSÖZ KAYA

ÇUKUROVA UNIVERSITY
DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING
INSTITUTE OF NATURAL AND APPLIED SCIENCES

Advisor	Asst. Prof. Dr. Turgay İBRİKÇİ
Co-advisor	Prof. Dr. Okan K. ERSOY
	Year 2008, 85 pages
Jury	Asst. Prof. Dr. Turgay İBRİKÇİ
	Prof. Dr. Seyhan TÜKEL
	Prof. Dr. Hamza EROL
	Prof. Dr. Fikret GÜRGEN
	Asst. Prof. Dr. Ulus ÇEVİK

Proteins are one of the most important parts of an organism because of their vital tasks. Hence, it is necessary to know the structure of a protein since it is closely related to its biological function. Studies on structural genomics indicate that numerous protein segments remain unfolded in their native states. Currently these proteins are generally referred to as “natively unfolded” or “intrinsically disordered”. Upon noticing that many of these proteins play key role in vital functions and also in some diseases, identification of the disordered regions has became a demanding process for structure prediction and functional characterization of proteins. Therefore, many studies have been motivated on accurate prediction of disorder. Mostly, machine learning techniques have been used for dealing with the prediction problem of disorder due to capability of extracting the complex relationships and correlations hidden in large data sets.

In the study, a novel method, named Border Vector Detection and Extended Adaptation (BVDEA) was developed for predicting disorder as an alternative accurate classifier. For attesting the performance of the method, three computational learning techniques and eleven specific tools were used for comparison. In order to perform the predictions, firstly the collected data was prepared by choosing a variety of structural features for proteins. Then, training was executed based on the data by 5-fold cross validation. When compared with the three learning methods of GRNN, LVQ and BVDA, the proposed method gives the best accuracy on classification. Here, BDVEA provides more fast and robust learning as compared to the others. Furthermore, comparison with the other existing disorder predictors demonstrates that the method achieves rather good prediction performance in finding disordered regions of proteins. As a conclusion, it has been shown that the new method provides a significant contribution on predicting disorder and order regions of proteins.

Keywords: Intrinsically Disordered Proteins, Compositional-Based Features, Property-Based Features, Evaluation-Based Features, Border Vector Detection and Extended Adaptation.

ÖZ

DOKTORA TEZİ

PROTEİNDEKİ DÜZENSİZ BÖLGERLERİN HESAPSAL TAHMİNİ

İrem ERSÖZ KAYA

ÇUKUROVA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ELEKTRİK-ELEKTRONİK ANA BİLİM DALI

Danışman	Yard. Doç. Dr. Turgay İBRİKÇİ
İkinci Danışman	Prof. Dr. Okan K. ERSOY
	Yıl 2008, 85 sayfa
Jüri	Yard. Doç. Dr. Turgay İBRİKÇİ
	Prof. Dr. Seyhan TÜKEL
	Prof. Dr. Hamza EROL
	Prof. Dr. Fikret GÜRGEN
	Yard. Doç. Dr. Ulus ÇEVİK

Proteinler hayati önem taşıyan görevleri nedeni ile canlı organizmaların en önemli moleküllerdir. Bu bakımdan, bir organizmanın yaşam sürecini anlayabilmek için ilk olarak yapı- işlev ilişkisinin bilinmesi gerekmektedir. Yapılan çalışmalar ile birçok proteinin 3B yapısına sahip olmadıkları halde bir veya birden fazla fonksiyona sahip oldukları bulunmuştur. Bu proteinlerin önemli fonksiyonlarda ve bazı hastalıklarda anahtar rol oynadıklarının bulunmasıyla bu proteinlerin tanımlanması, protein yapısının tahmini ve fonksiyonun belirlenmesi için talep gören bir işlem haline gelmiştir. Bu nedenle, düzensizliğin başarılı tahmini üzerine birçok çalışma yapılmıştır. Bu düzensizliğin tahmin edilmesi problemi ile başa çıkabilmek için, büyük veri setlerindeki karmaşık ilişkileri ve korelâsyonu ortaya çıkarmadaki yeteneği nedeniyle çoğunlukla makine öğrenmesi teknikleri kullanılmaktadır.

Bu çalışmada, Sınır Vektör Belirleme ve Genişletilmiş Adaptasyon (SVBGA) metodu düzensiz bölgelerin tahmini için başarılı bir alternatif sınıflandırıcı olarak geliştirilmiştir. Metodun başarısının onaylanması için üç hesapsal öğrenme tekniği ve on bir mevcut düzensiz bölge tahmin aracı karşılaştırmada kullanılmıştır. Tahminlerin gerçekleştirilebilmesi için öncelikle, birçok yapısal özellik kullanılarak yeni bir veri hazırlanmıştır. Daha sonra, 5 katlı çapraz değerlendirme ile eğitimler gerçekleştirilmiştir. Verilen öğrenme yöntemleri ile karşılaştırıldığında, metot sınıflandırmada en başarılı sonucu vermiştir. Burada, SVBGA diğerlerine göre daha başarılı ve daha hızlı öğrenme sağlamıştır. Ayrıca, verilen mevcut tahmin araçları ile yapılan diğer karşılaştırma, metodun oldukça iyi bir tahmin performansı elde ettiğini göstermiştir. Sonuç olarak, yeni metodun düzensiz ve düzenli bölgelerin tahmini üzerine dikkate değer bir katkı sağladığı gösterilmiştir.

Anahtar Kelimeler: Özünde Düzensiz Proteinler, Bileşim-Tabanlı Öznitelikler, Özellik-Tabanlı Öznitelikler, Evrim-Tabanlı Öznitelikler, Sınır Vektör Belirleme ve Genişletilmiş Adaptasyon.

ACKNOWLEDGEMENTS

I would like to express my respects and gratitude to my advisor Asst. Prof. Dr. Turgay İbrikçi and to co-advisor Prof. Dr. Okan K. Ersoy. Many thanks for all supports, guidance, cooperates, suggestions on initiating, improving and completing the study.

I would like to thank Prof. Dr. Süleyman Güngör, the head of the Department of Electrical and Electronics Engineering, and all my friends in the Electrical Electronics department for their moral supports.

I would like to express my appreciation and my respects to Prof. Dr. Y. Gürcan Ültanır, the dean of Tarsus Technical Education Faculty in Mersin University for his help, tolerance and encouragement. He created a convenient environment for me to be able to study my Phd thesis. I also thank all staff and colleagues in the faculty for their supports and motivations.

I also thank my committee members, Prof. Dr. Seyhan Tükel, Prof. Dr. Fikret Gürgen, Prof. Dr. Hamza Erol and Asst. Prof. Dr. Ulus Çevik for their supports and valuable discussions.

For their great supports and considerable assists, I would like to express my appreciation to my best friends, M. Tulin Tokgöz, Gülhan Orekici and Aslı Sezeroğlu Ökten. Besides, I thank all my friends that I couldn't mention their names here for their lovely wishes, beliefs and supports.

I would like to express my best feelings to my lovely and best friend Ayça Çakmak Pehlivanlı for her helpful collaboration and moral support. She is always with me in my trouble times.

I greatly appreciate my father Muhlis, my mother Gönül and my sister Ceren. They always encouraged and supported me with their loves and beliefs. I know they are always there to love and care me. I would like to denote my lovely feelings to them for everything. I also thank my mother-in-law Hamiyet, my father-in-law Avni for their sincere wishes and cares.

Lastly, many special thanks to my husband Cumhur, my love, for his endless patience, care, support, love and for standing with me every time. He made me feel comfortable and calm during the study. He always believes in me and motivates me to complete the thesis. I dedicate this study to him.

CONTENTS **PAGE**

LIST OF ABBREVIATIONS AND NOTATIONS

GRNN	:	General Regression Neural Networks
LVQ	:	Leraning Vector Quantization
BDVA	:	Border Vector Detection and Adaptation
BDVEA	:	Border Vector Detection and Extended Adaptation
LR	:	Logistic REgresion
PCA	:	Principal Component Analysis
PCs	:	Principle Components
Aa	:	Amino acid
3D	:	3 Dimensional
PSSM	:	Position Specific Scoring Matrice
Sens	:	Sensitiviy
Spec	:	Specificity
ProbEx	:	Probability Excess
Mcc	:	Methews' Correlation Coefficient
Acc	:	Accuracy
h	:	Learning Rate
t	:	Decreasing Constant
n_c	:	Number of Codebook Vectors
σ	:	Smoothing Factor, Sigma

LIST OF TABLES **PAGE**

LIST OF FIGURES **PAGE**

1. INTRODUCTION

Proteins are the building blocks of life. They have vital responsibilities in living organisms such as catalyzing and regulating biochemical reactions, transporting molecules and also the chemistry of the photosynthetic conversion of light to growth, defense against alien substances, etc. The proteins accomplish almost all of the biochemical activities of the organisms. To understand the life process of an organism, it is necessary to have the knowledge of the protein's structure at first. Indeed, the three-dimensional structure of a protein is closely related to its biological function.

Routinely used and reliable methods to determine the structure of a protein are laborious techniques such as Nuclear Overhauser Effect, X-Ray crystallography, Nuclear Magnetic Resonance spectroscopy (NMR) and DNA microarray technology. Using direct biological techniques requires skilled experts to operate the equipment and this is quite expensive and time consuming. DNA microarray technology (sequence comparison) is quite expensive and also necessitates proficiency to operate. Although assessing protein function by sequence comparison is rather quick, it is limited to the proteins that have strong sequence similarity with functionally known proteins, and it might cause large experimental errors in measurements of the gene expression level. Therefore, assigning protein functions and determination of protein three-dimensional structures by traditional experimental methods cannot catch up with the speed of new genomic data generation since it is much easier to infer the protein sequence. In recent times, roughly a million protein sequences have been identified but less than 50.000 structures among them have been determined and also insufficient number of the known genes has only been assigned functions. The data explosion in Protein Data Bank is shown in Figure 1.1, retrieved in April, 2008 from PDB [Protein Data Bank].

Since the capacity to generate data has outstripped the ability to analyze the data, in recent years scientists have been concerned with new opportunities for research in statistical modeling and computer intelligence. Computer information

technology has also advanced rapidly and has provided methods for handling this data explosion.

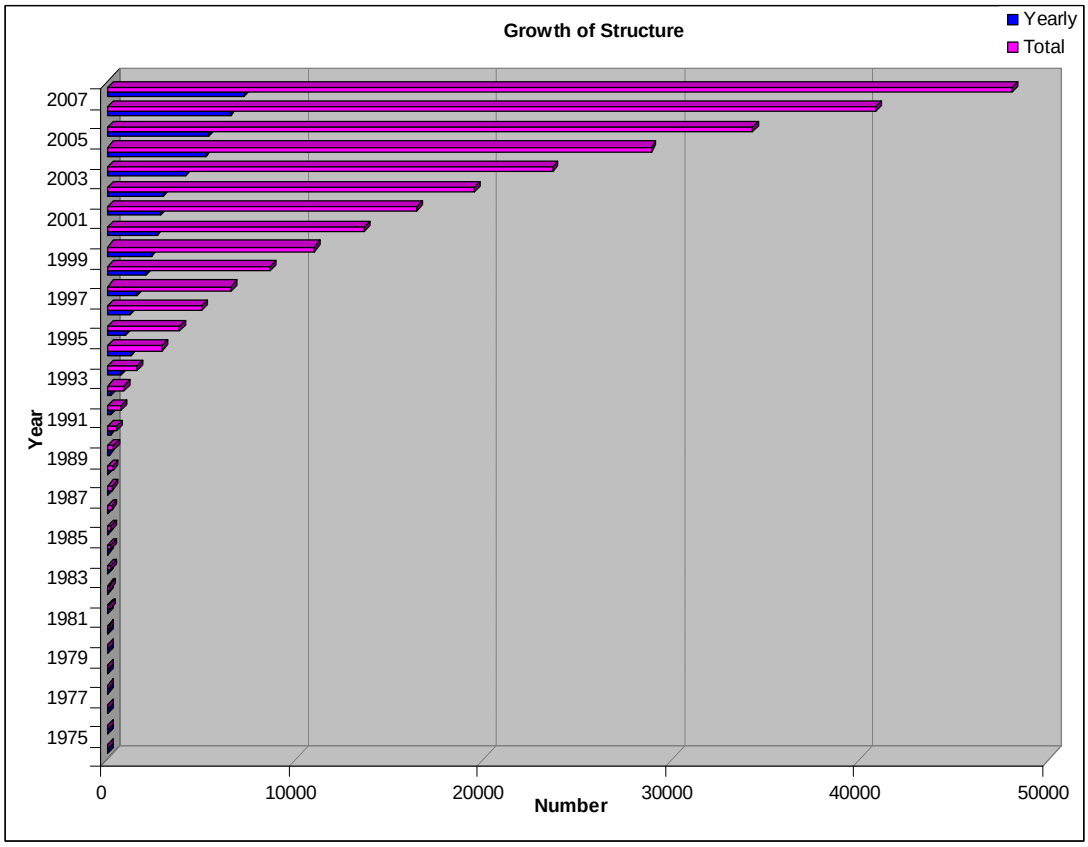


Figure 1.1. The exponential growth of the PDB Database in the period 1974-2007.

The rapid pace of genomic sequencing and thus the abundance of the sequence data have motivated the studies on structure prediction of the protein from its amino acid sequence. These predictions are then used to deduce function. The scheme assumes that tertiary structures are a pre-requisite for function [Fisher, 1894]. However, this view ignores numerous proteins that utilize unfolded or incompletely folded regions for function [Romero et al., 2001]. A protein that lacks three-dimensional (3-D) structure in its intrinsic state has been called natively unfolded or intrinsically disordered [Dunker et al., 2000]. Such disordered regions of amino acid sequences are also involved in a variety of biological functions. The existence of these proteins reveals a “protein non-folding problem” [Li et al., 2000].

Upon noticing that many intrinsically disordered protein regions play key role in vital functions and also in some diseases, identification of disordered regions in proteins has became a consequential process for structure prediction and functional characterization of the protein [Weinreb et al., 1996: from Tompa, 2002]. The aim of this study is to find an efficient computational method that can provide information about the structural class among ordered and disordered proteins concerning different biochemical and physical features of amino acids. For this purpose, a new algorithm was proposed in the study, named Border Vector Detection and Extended Adaptation (BVDEA) to achieve an accurate prediction of disordered data.

1.1. Proteins

A protein consists of a linear sequence of the twenty naturally occurring amino acids. An amino acid contain an alpha carbon (Cα) that is attached to a carboxyl group (-COOH), an amino group (-NH2), a hydrogen atom (-H) and a side chain group (-R) that gives each amino acid its distinguished properties [Solomons, 1995]. Figure 1.2 illustrates the general structure of an amino acid.

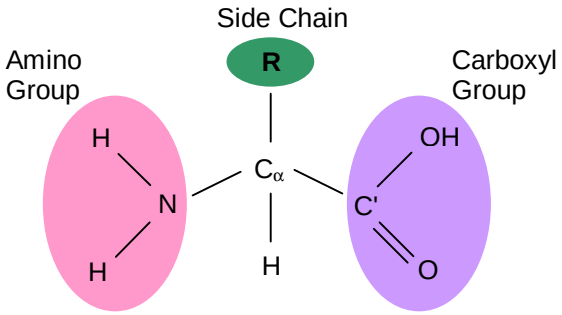


Figure 1.2. The general structure of an amino acid.

When the carboxyl group of one amino acid is joined to the amino group of another amino acid by a peptide bond, this forms a dipeptide (Figure 1.3). The molecule constituted by n-sequence of amino acid residues is termed as a polypeptide. This sequence of amino acids is called as the primary structure.

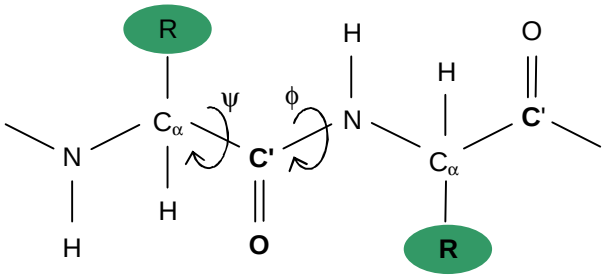


Figure 1.3. The structure of a dipeptide.

The bonds that join Cα to amino group (N-Cα) and to carboxyl group (Cα-C) can rotate in a great flexibility with the angles of PHI (ϕ) and PSI (ψ), respectively. Thus, the protein folds into a three-dimensional configuration according to the association between amino acid residues by stabilizing certain conformations above others to conserve the largest amount of energy with this rotation of its segments along the ϕ and ψ angles. This leads to the local and repetitive spatial arrangements of the sequence, named as the secondary structure [Denniston et al., 2001]. α-helices, β-strands and coils are the most common types of the secondary structure.

A protein can take on its final three-dimensional shape via the arrangement of secondary structure regions. This completion is termed as the tertiary structure. For many proteins, the functional form is composed of several polypeptide chains. The quaternary structure involves differences on how chains combine to compose the functional form (protein complexes) [Solomons, 1995].

Amino acids can be grouped according to their physicochemical properties owing to their different side chain-R groups as mentioned above. The names of the known 20 amino acids, their 1-letter and 3-letter abbreviations and several physicochemical properties are shown in Table 1.1 [Baldi and Brunak, 2001]. There is numerous scale values specified through different studies for amino acid properties. In AAIndex database, more than 500 physicochemical feature scales have been listed [AAIndex] [Kawashima et al, 1999].

Table 1.1. Names, symbols and physicochemical properties of amino acid residues.

Amino Acid	***3-Letter***	***1-Letter***	***Acidity/ Basicity***	***Hydropathy***	***Polarity***
Alanine	Ala	A	Neutral	Hydrophobic	Nonpolar
Cysteine	Cys	C	Neutral	Hydrophilic	Polar
Aspartic Acid	Asp	D	Acidic	Hydrophilic	Polar
Glutamic Acid	Glu	E	Acidic	Hydrophilic	Polar
Phenylalanine	Phe	F	Neutral	Hydrophobic	Nonpolar
Glycine	Gly	G	Neutral	Hydrophobic	Nonpolar
Histidine	His	H	Basic	Hydrophilic	Polar
Isoleucine	Ile	I	Neutral	Hydrophobic	Nonpolar
Lysine	Lys	K	Basic	Hydrophilic	Polar
Leucine	Leu	L	Neutral	Hydrophobic	Nonpolar
Methionine	Met	M	Neutral	Hydrophobic	Nonpolar
Asparagine	Asn	N	Neutral	Hydrophilic	Polar
Proline	Pro	P	Neutral	Hydrophobic	Nonpolar
Glutamine	Gln	Q	Neutral	Hydrophilic	Polar
Arginine	Arg	R	Basic	Hydrophilic	Polar
Serine	Ser	S	Neutral	Hydrophilic	Polar
Threonine	Thr	T	Neutral	Hydrophilic	Polar
Valine	Val	V	Neutral	Hydrophobic	Nonpolar
Tryptophan	Trp	W	Neutral	Hydrophobic	Nonpolar
Tryosine	Tyr	Y	Neutral	Hydrophilic	Polar

1.2. The Protein Folding Problem

Widely believed dogma of structural biology states that the three dimensional (3D) structure of a protein is a prerequisite for its biological function. The structure – function paradigm has been given as [Romero et al., 2000];

$$\text{DNA} \longrightarrow \text{mRNA} \longrightarrow \underset{\text{sequence}}{\text{AA}} \longrightarrow \underset{\text{structure}}{\text{3D}} \longrightarrow \text{Function}$$

The tenet has been arisen more than 100 years ago with Fischer's "lock and key" model or Koshland's "induced fit" theory in which both the enzyme and

its substrate have fit to each other with the complementary 3D shapes like a lock and key in order to fulfil enzymatic behaviour [Fisher, 1894] [Koshland, 1958].

The functional form of a protein is named as “the native state”; on the other hand, the loss of functional activity exhibits “the denatured state” of the protein. Generally the loss of function is associated with the lack of specific 3D structure [Mirsky and Pauling, 1936].

Due to the influence of structure in protein function, it is also a need to study structural prediction. The breakthrough that structure (also referred to as ‘the fold’) is uniquely determined by the sequence has been made by Christian B. Anfinsen in the mid 1950’s and this brought his outstanding success as the 1972 Nobel Prize in chemistry [Anfinsen, 1973]. Afterwards, tremendous efforts have been performed by experimental and computational methods on that challenge, which has come to be widely known as the “The Protein Folding Problem”. The goal of the studies on solving the problem is mostly the determination of protein structures, and so functions, and provides insights for the projects on drug discovery and gene finding.

1.3. The Protein Non-Folding Problem

At the end of 1970s, researches on structural/functional analysis indicated that some protein segments remain unfolded in their native states. Contrary to the structure – function paradigm, several protein regions that fail to fold into a fixed 3D structure under physiological condition, yet exhibit function; have been discovered by the experimental methods such as X-Ray crystallography, Nuclear Magnetic Resonance (NMR) spectroscopy and Circular Dichroism (CD) spectroscopy.

Over the years, many proteins have been found to contain these unstructured regions and to play crucial role in a variety of functions including DNA binding, cell signaling and protein modification [Dyson and Wright, 2005]. It has been shown that the proteins can also be entirely disordered. Currently these proteins are generally referred to as “natively unfolded” or “intrinsically

disordered" [Dunker et al., 2000]. Different combinations of natively/intrinsically with unfolded, unstructured or disordered can also be given as alternative terms of such proteins that have been used in the literature. Figure 1.4 shows the structure of calcineurin protein as an example of disordered proteins.

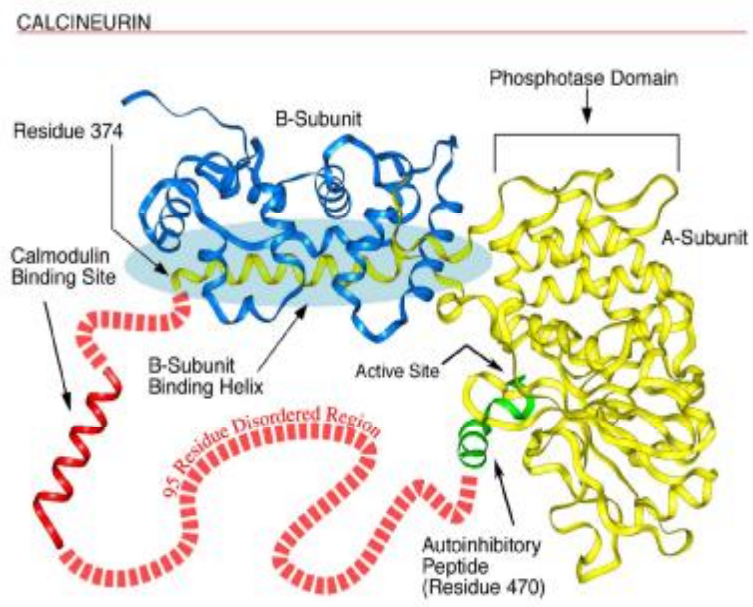


Figure 1.4. The structure of calcineurin [Kissinger et al, 1995: from DisProt].

In the prior work of Dunker and colleagues, they have pointed out the functional importance of the type of proteins [Dunker et. al, 1997]. Moreover, many studies have also emphasized the importance and the necessity of the intrinsically disordered proteins for functionality. For instance, it has been elucidated that their flexible conformation enables them to form complexes with several different targets via disorder-to-order transitions upon binding as in enzyme binding with substrate, and protein binding with RNA/DNA/protein [Romero et al., 1997] [Obradovic et al., 2003]. These findings have prompted researchers to re-assess the protein structure – function paradigm. The requirement was mentioned firstly by Wright and Dyson in 1999 [Wright and Dyson, 1999].

Afterwards, Dunker et al. have proposed the Protein Trinity model that a

particular protein function may arise from any one of three structural states of a protein or transitions between them (Figure 1.5) [Dunker et al., 2001a]. The three state of protein folding have been known as the ordered state, molten globule and random coil where molten globule–like and random coil–like forms have been defined as collapsed–disorder and extended–disorder, respectively. Hence, wholly or locally unstructured proteins can also be in a functional (native) form in contrast to the view of "function requires structure".

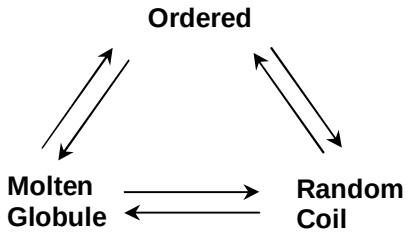


Figure 1.5. The Protein Trinity Model [Dunker, 2001b].

In 2002, the trinity model was widened according to the acceptance of "pre-molten globule" as a distinct conformational state in protein folding [Uversky, 2002]. A protein in the state is segregated from the others by several characteristic properties such as secondary structure content, all-or-none transition conduct and size of compactness.

The alternative proposal was introduced by Uversky as "The Protein Quartet Model" in which the biological activity, henceforth, includes the pre-molten globule state and also transitions from one to any other (Figure 1.6).

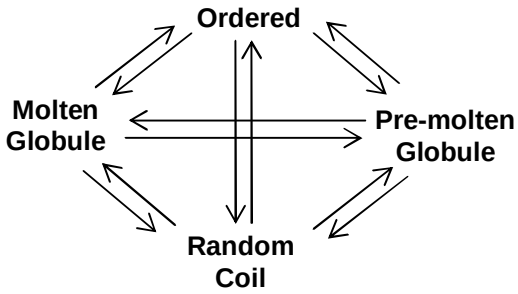


Figure 1.6. The Protein Quartet Model [Uversky, 2002].

Recently, it has been confirmed that some disordered proteins can also involve in several diseases such as Alzheimer disease, Parkinson disease and certain types of cancer [Williams et al., 2001] [Galzitskaya et al., 2006]. Therefore, studies on structural identification of intrinsically disordered proteins have been thought to be an aid in drug design, protein expression and functional recognition. Due to their significance, dealing with structural identification of the proteins that was named as "The Protein Non-Folding Problem", have gained growing interest in structural bioinformatics [Li et al., 2000].

In accordance with the studies on protein folding problem, disordered regions have also been identified by using experimental methods. In X-ray diffraction, missing electron density indicates disorder. In NMR, disorder is revealed by sharp peaks or the negative values of N-H heteronuclear Nuclear Overhauser Effect (NOE) measurements. CD utilizes UV spectra in order to identify disorder by low intensity wavelength. The method does not give residue-by-residue basis information but provide secondary structure estimation [Tompa, 2002]. Each method has its own pros and cons in characterization of disorder due to the quality of being able to examine different aspects of protein structure. For instance, because of its lack of regular secondary structure, a loopy protein can be identified as a disordered protein by CD spectroscopy analysis. A wobbly ordered region could be misclassified in X-ray crystallography. NMR has difficulties in discerning the molten globules from the random coils [Romero et al., 2001].

Because of the constraints, it is desirable to confirm or to reinforce the experimental results by multiple methods. Besides, experimental methods are also quite expensive and time-consuming as mentioned before. Thus, alternatively, several computational methods have been suggested for disorder prediction based on the amino acid sequence [Romero et al., 1997] [Garner et al., 1998] [Obradovic et al., 2003] [Weathers et al., 2004] [Coeytaux and Poupon, 2005]. It has been shown that the amino acid properties can be used as a discriminator for characterizing disorder. For example, Uversky and co-workers have proposed that a low mean hydrophobicity combined with relatively high mean net charge is a significant indicator of disorder [Uversky et al., 2000].

1.4. Machine Learning in Bioinformatics

Machine learning is a field of research concerning the design and development of computational algorithms and techniques that have the learning ability to solve complicated problems. Machine learning techniques are ideally used for extracting the complex relationships and correlations hidden in large data sets. Therefore, the methods are well suited for typically multidimensional and noisy, and complex data of computational biology.

Machine learning approaches can be applied in two distinct manners: supervised learning and unsupervised learning.

In supervised learning, training requires the corresponding labels of given input objects to achieve classification or regression of the unseen instances. The task of the machine is to find the correct class label (in classification) or real number output (in regression) of the query via the generalization of the training data. The supervised information enables the learning machine to detect misclassifications to use it as a feedback. Artificial neural networks like Generalized Regression Neural Network (GRNN) or statistical methods like Logistic Regression (LR) can be cited as examples for supervised learning, i.e. classification, techniques. Classification is generally referred to as a supervised learning approach.

On the contrary, unsupervised learning approaches do not utilize a supervisor. Training is executed based on a heuristic search for interesting and reasonable features of unlabeled data. Clustering and dimensionality reduction are the main targets of the analysis.

Clustering is a process by which samples that are logically similar in characteristics are grouped together and by which dissimilar samples are separated from each other, whereas classification process associate each sample with an appropriate response (Figure 1.7).

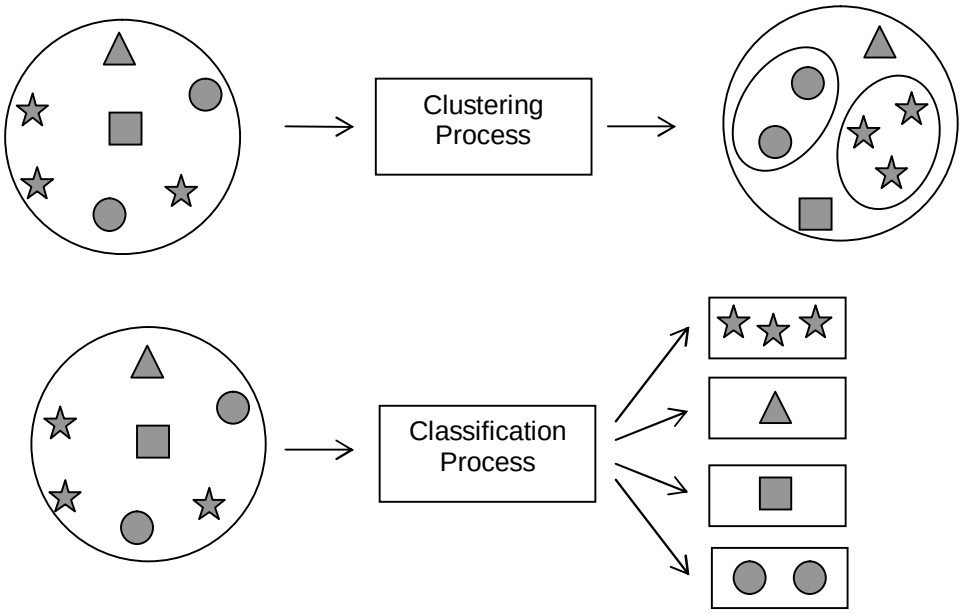


Figure 1.7. Clustering process versus classification process.

There are two basic types of clustering: partitioning and hierarchical. In partitioning, the data is partitioned into a pre-defined number of k mutually exclusive and collectively exhaustive groups. The allocation of the samples to appropriate groups iteratively continues until some convergence criterion is provided, e.g. no more reassignments occur. For instance, K-means and Self Organizing Maps (SOM) clusters the data via partitioning by using unsupervised learning. In hierarchical methods like Hierarchical Clustering, a tree (called a dendrogram) is produced with individual elements at one end and a single cluster containing every element at the other. A clustering at a selected precision is provided via cutting the tree at a given depth.

In machine learning applications, the *curse of dimensionality* is a common problem that might be a factor of degradation in the performance of a given learning algorithm as the number of features increases [Bishop, 1995]. Consequently, it might be necessary to use models that perform dimensionality reduction for high dimensional data sets. The reduction techniques are often used as a data pre-processing step to simplify the data so that a suitable low-dimensional representation of the original high-dimensional data set can be identified. Dimensionality reduction can be performed by selecting a relevant

subset of the existing features, known as feature selection or transforming to a new reduced set of features, known as feature extraction. The choice of strategies differs respective of whether the learning process is supervised or unsupervised. The most popular feature extraction method is the Principal Components Analysis (PCA) which is an unsupervised feature transformation technique. On the other hand, a supervised feature selection can be performed by the Stepwise Logistic Regression Analysis.

Artificial Neural Networks (ANNs) are computational paradigms that have an amazing capacity to actually learn from input data, inspired by the way the biological nervous system in the brain processes information. A neural network is composed of highly interconnected processing units or artificial neurons. These units are joined together with weighted connections that are similar to synapses of the brain. The neural network architecture consists of three or more layers of neurons that fully connect to consecutive layer neurons in which each connection is associated with numerical weights [Sarle, 1994]. The input exemplars from previous neurons are processed through the weighted sum and then transmitted to another neuron via a transfer or activation function as an input [Jordan and Bishop, 1996]. This process is repeated after the modification of the weights until the errors between the predicted and expected output values are minimized [Lippmann, 1987]. This iterative modification known as training, characterize the learning ability of ANNs. The goal of the training is to reach an optimal solution based on a performance measure.

2. RELATED WORKS

Since experimental analyses are quite expensive, complex and time-consuming, the statistical modeling, data mining and neural network research on genome informatics (bioinformatics) have advanced very rapidly in recent years. Owing to the benefits of recognition, generalization and classification, machine learning techniques, especially artificial neural networks, are well suited for structural genomic studies.

Specifically, the rapid pace of genomic sequencing and thus the abundance of the sequence data have motivated the studies on structure prediction of the proteins from their amino acid sequences. Although the ultimate goal is to predict 3-D structure, 1-D predictions such as secondary structure or solvent accessibility predictions have a great interest for researchers because of being an intermediate step on the way to predicting 3-D structure. Moreover, the approaches for prediction of 3-D from sequence are frustrated by high complexity of protein structure formation [Rost, 1998]. The prediction of disordered proteins (per protein) or disordered regions (per-residue) is also a 1-D prediction problem and an important tread for structural analysis and functional characterization of the protein [Uversky, 2005].

As the first application of the computational and statistical studies on structural predictions, Krigbaum and Kuntton used the multiple linear regression analysis for prediction of secondary structure [Krigbaum and Kuntton, 1973: from Cai et al., 2003]. This attempt continued with Chou and Fasman. They achieved a three-state (Q3) accuracy of 52% with their well-known statistical algorithm that is based on the frequencies of the secondary structure types [Chou and Fasman, 1978]. Neural network applications on structural studies were started with Stormo et al. via using perceptron-learning algorithm developed by Rosenblatt in the late 1950's to distinguish ribosomal binding sites from non-binding sites [Stormo et al., 1982]. Early genome informatics applications mostly involved the use of perceptron or backpropagation networks [Wu and McLarty, 2000].

In 1988, Qian and Sejnowski attempted to predict protein secondary

structure conformation (α-helix, β-sheet and coil) of non-homologous data set by using neural networks models [Qian and Sejnowski, 1988]. They applied the sliding window approach to pre-process the non-homologous protein sequence data with neural network application and achieved 64.3% success rate on prediction.

Up to the last decade of the 20th century, the accuracy of secondary structure prediction did not exceed the level of about 65%. A significant increase in performance was obtained by Rost and Sander surpassing 70% via the use of evolutionary information of proteins [Rost and Sander, 1993]. They verified the prediction performance of two-layered feed forward neural network by 7-fold cross validation, concluding with 70.8% accuracy. In 1994, Rost and Sander used a similar method to predict the relative solvent accessibility in either of two states, buried or exposed. Solvent accessibility prediction has been thought as another intermediate step on the way to predicting 3D structure as well as secondary structure prediction. Afterwards, Rost proposed a method, called PHD (Profile network from HeiDelberg) that consisted of two level structure-to-structure network, both designed according to three-layer back propagation procedure [Rost, 1996]. The network input was comprised of the amino acid residue frequencies and 13 amino acid length sliding window of the HSSP alignments. Recently, the methods of Rost et al. have been known as PHDsec and PHDacc.

As the field continues to develop, researchers broaden the choices of neural network architectures and other statistical methods. Krogh and Riis made another study on protein structure prediction. They applied structured neural networks and multiple sequence alignment with an accuracy of 80% [Krogh and Riis, 1996]. In 1999, Jones et al. applied a two stage neural network based on the position specific scoring matrices (PSSMs) to predict the secondary structure of a protein [Jones, 1999]. Position specific scoring matrix profiles obtained from Psi-Blast execution involves the evolutionary information with the level of position conservation. The method, named PSIPRED, achieved an average Q3 score between 76.5% and 78.3% which is the highest score within the published results

of any method to date. Ward et al. used PSSM profiles to train Support Vector Machines (SVMs) for secondary structure prediction [Ward et al., 2003]. The average three-state prediction accuracy per protein (Q_3) was estimated by cross-validation to be above %77.

Since amino acid sequence attributes have been widely used for determining protein structure, it has been theorized that the sequence would also determine disorder, as well. Thus, alternative to experimental methods, several computational methods have been suggested for disorder prediction based on the amino acid sequence. Mostly preferred machine learning techniques can be given as neural networks [Romero et al., 1997] [Romero et al., 1998] [Garner et al., 1998] [Li et al., 1999] [Li et al., 2000] [Dunker et al., 2000] [Romero et al., 2000] [Romero et al., 2001] [Vucetic et al., 2001] [Obradovic et al., 2003] [Linding et al., 2003a] [Vucetic et al., 2003] [Jones et al., 2003] [Radivojac et al., 2004] [Yang et al., 2005] [Cheng et al., 2005] [Peng et al., 2005] [Peng et al., 2006] and support vector machines [Shimzu et al., 2004] [Weathers et al., 2004] [Peng et al., 2006] [Shimzu et al., 2007]. In majority of the studies, the input patterns have been mostly derived from a variety of sequence properties such as flexibility, amino acid frequency, complexity, charge, and secondary structure that characterize disorder.

To improve the quality of prediction, it has been recently made efforts to find more useful features and to develop more robust predictors [Garbuzynskiy et al, 2004] [Shimzu et al., 2004] [Dosztányi et al., 2005] [Shimzu et al., 2007]. For instance, Jones et al. demonstrated in their study that using the position specific scoring matrices (PSSMs) within a defined length of window can improve the accuracy of predicting its disorder attribute [Jones and Ward, 2003].

Eventually, specific disorder prediction tools have been developed such as PONDRs, DisEMBL, GlobPlot, DISOPRED2, FoldIndex, RONN, DisPRO, PreLink, DisPSSMP.

The first tool designed specifically for prediction of protein disorder was PONDR (Predictor of Naturally Disordered Regions) called XL1 [Romero, 1997]. The trained feedforward neural network model achieved 58% prediction accuracy

by using only 7 X-ray characterized partially disordered proteins. Prediction was based solely on the frequencies of 8 amino acid sequences (His, Glu, Lys, Ser, Asp, Cys, Trp and Tyr) and two average attributes. Subsequently, a series of PONDRs were constructed, and overall prediction accuracy ultimately exceeded the level of 80%. PONDR VLXT reached 70% prediction accuracy by integrating their three initial neural network predictors, VL1, XN and XC. VL1 was trained on 25 variously characterized disordered proteins and gave an accuracy of 64% [Romero et al., 2001]. XN and XC were proposed as N- and C- terminal predictors [Li et al., 1999]. The overall accuracy of the most recent PONDR predictors (VL3 series) ranges from 79% to 83% with a Win size of 41 and Wout sizes ranging from 1 to 121 [Peng et al., 2005]. They attained this significant improvement by adding the feature of evolutionary knowledge.

DisEMBL provides a method based on artificial networks trained for predicting three different definitions of disorder. The method was used with all three types of disorder which are hot loops, coils and REMARK465, named as EMBL(hot), EMBL(coil) and EMBL(465) [Linding et al., 2003a]. REMARK465 contains the regions that lack electron density in crystal structure. Hot loops indicate highly mobile loops where coils are the regions that have no regular secondary structures. The most accurate result for the prediction of hot loops was correct prediction of 64%. The method is publicly accessible as a web server at http://dis.embl.de.

Linding et al. designed a model, called GlobPlot based on amino-acid propensities for disorder/globularity [Linding et al., 2003b]. They proposed a new scale for propensities, named Russell/Linding to be either in regular secondary structures (a-helices or b-strands) or outside of them (‘random coil’, loops, turns etc.). GlobPlot is a web service at http://globplot.embl.de that allows the user to plot the disorder/globularity tendency of a query protein. At a specificity of 88% we obtained a sensitivity of 28%.

Linear support vector machines based on a local amino acid sequence was trained in the DISOPRED2 method [Ward et al., 2004a]. Missing residues from the electron density map were defined as disordered regions to form the data set.

A sequence profile derived from a PSI-BLAST search was used to construct the input vector for each residue within the window length of 15. Testing results was given with a ~93% accuracy. DISOPRED2 can be downloaded from the web site at http://bioinf.cs.ucl.ac.uk/disopred/ [Ward et al., 2004b].

Uversky and coworkers proposed that a low mean hydrophobicity combined with relatively high mean net charge is a significant indicator of disorder [Uversky et al., 2000]. The web-based implementation of the algorithm, named as FoldIndex is served at http://bip.weizmann.ac.il/fldbin/findex by Prilusky et al [Prilusky et al., 2005]. In the per-sequence prediction method, 30 proteins from among the 39 intrinsically disordered proteins were correctly assigned with the label of “disordered” while 15 thru 151 ordered proteins were correctly classified as “ordered”. The overall accuracy was reported as 83%.

Yang et al. proposed a novel model, Regional Order Neural Network (RONN), for prediction of disordered region based on the alignment scores of the windowed sequences against a series of sequences of known folding state [Yang et al., 2005]. In RONN, the determined distances from a subset of well-characterized prototype sequences are used to train a neural network for classifying the query sequence as ordered or disordered. They created a data set through the proteins from BDP by applying some filtering applications. 80 proteins of the data set were reserved for main testing; the remaining sequences were used for training and validation. Another test set was derived from 79 completely disordered proteins and 80 completely ordered proteins. The method achieved the testing accuracies of 0.849% and of 0.789% for the main testing set and for the other, respectively. RONN is available at http://www.strubi.ox.ac.uk/RONN.

DISpro is a 1D-recursive neural network method that involves the use of evolutionary information incorporating the predicted secondary structure and relative solvent accessibility [Cheng et al., 2005]. The proteins used for training and testing were extracted from the PDB. They used only the proteins that were solved by X-ray crystallography with a resolution better than 2.5 Å. All proteins were required to be at least 30 residues long. Disordered regions were determined

according to missing residues of ATOM records in PDB. Sensitivity result was given as 75.4% and the specificity was 38.8%.

PreLink is a method that relies on compositional bias and low hydrophobic cluster content and is accessible at http://genomics.eu.org/spip/PreLink. In this method, a linker set was created by aligning the BDP proteins with its corresponding Swiss-Prot record and extracting non-aligned regions. The resulting segments were classified by their position as N-terminal, C-terminal or inner fragments. The prediction involved using three values obtained from the calculation of the amino acid distributions in structured and unstructured regions, of the probability that a given sequence fragment be part of either a structured or an unstructured region, and of the distance to the nearest hydrophobic cluster of each amino acid [Coeytaux and Poupon, 2005]. PreLink correctly predicted 59.9% of the N-terminal fragments, 70.5% of the C-terminal fragments and 61.1% of the inner fragments.

Su et al. used condensed position specific scoring matrix profiles by merging associated columns of the matrix concerning the several physicochemical properties of amino acids, called PSSMP [Su et al., 2006]. The training data set that contained totally 336880 residues was collected from the Protein Data Bank and DisProt Database. They achieved rather successful predictions for two distinct test sets via training a Radial Basis Function (RBF) neural network based on their proposed PSSMP patterns. In the study, the same testing sets were used as with the study of Yang et al. The sensitivity of 0.767 and the specificity of 0.848 were given for the prediction results of the main test set while the obtained scores on testing the other set were 0.825 and 0.765.

3. MATERIALS AND METHODS

In the problem of predicting disordered regions, the first step is to prepare a data set for training machine learning techniques and for measuring their prediction performances. In the study, two data sets were constructed. The one has been used as the main data set and the other has been used as blind testing.

To evaluate the proposed method, several applications were performed. Firstly, a series of evaluations and validations have been performed for finding the setting values of the method. Afterwards, to compare the performances of the method, Border Vector Detection and Adaptation (BVDA), Learning Vector Quantization (LVQ) and Generalized Regression Neural Networks (GRNN) were utilized as machine learning techniques. For the second comparison, the existing tools of disorder predictor in literature were used. PONDRs, DisEMBL, GlobPlot, DISOPRED2, FoldIndex, RONN, DisPRO, PreLink and DisPSSMP were chosen for this aim. These tools have been mentioned in literature review. Finally, the effect of dimension reduction to performance was investigated with Principal Component Analyse (PCA) and Logistic Regression (LR)

3.1. Data Sets

Although the regions of the proteins whose structure is specified by X-ray crystallography in Protein Data Bank /PDB with missing electron density can be classified as possible disorder regions, it has no certainty. The unobserved residues in the structure obtained at the end of X-ray diffraction, may arise from the technical malfunctions or measurement errors during the experiment (Protein Data Bank). Hence, it can be insufficient to use only this criterion in selecting the disordered proteins for the training set in terms of reliability. Due to this reasoning, proteins that were used and verified in previous studies were preferred in this study for composing the data sets. In this study, 3 protein sets from different studies were used for training and validation processes.

1. Yang et al. organized a data set including 80 proteins to test the prediction accuracy of Regional Order Neural Network / RONN that they developed. [Yang et al., 2005]. This data set was constituted through the analysis of all entries obtained by NMR and X-ray crystallography in the Molecular Structure Database/MSD [Boutselakis et al, 2003]. For each MSD entry; the residue of the sequence observed in ATOM coordinate records in PDB is aligned with both the SEQRES records providing protein sequences and corresponding UniProt sequences. In this way, the unseen regions could be detected. 7327 entries were obtained considering unobserved regions containing at least 5 consecutive residues. These entries were divided into two as the ones having more than 20 consecutive unobserved residues (long set, 1573) and the others (short set, 5754). The long set received 872 entries after the filtering applications like the extraction of multi component complexes, the preference of only the highest numbered sequence within the same 3-letter code entries, the selection of the residues which have not been observed in other chains for entries containing multiple chains. Then, by using CD-Hit program, the ones with more than 70% sequence identity among the remaining proteins were eliminated. [Li et al, 2001]. Consequently Yang et al. used randomly selected 80 proteins from these entries as the basic test set. The composition of the data set called D80 in our study is shown in Table 3.1.

Table 3.1. The composition of D80 data set.

	D80
Number of Chains	80
Number of Ordered Regions	151
Number of Disordered Regions	183
Number of Ordered Residues	29909
Number of Disordered Residues	3649
Total Residues	33558

2. Data set with 80 completely ordered proteins were obtained from PONDR® website by Yang et al (retrieved in February 2003). Romero et

al. developed this data set in 2001 for the PONDR studies from their non-redundant database, named as O_PDB_Select_25 that was created by using PDB_Select_25 database [Romero et al., 2001]. PDB_Select_25 is a database that contains one protein sequence representing each protein group in PDB [Hobohm and Sander, 1994]. By extracting only the ordered sequences from this database, O_PDB _Select_25 database was created. The composition of the completely ordered data set, called CO80 in this thesis is shown in Table 3.2.

Table 3.2. The composition of CO80 data set.

	CO80
Number of Chains	80
Number of Ordered Regions	80
Number of Disordered Regions	0
Number of Ordered Residues	16568
Number of Disordered Residues	0
Total Residues	16568

3. In the study conducted by Uversky et al. in 2000, they reported 91 completely disordered proteins defined through spectroscopic methods in the literature [Uversky et al, 2000]. These proteins ranging in size between 50 and 1827 were stated in the literature as having the chemical shifts of a random-coil in nuclear magnetic resonance and/or lacking of regular secondary structure detected by circular dichroism/CD or fourier transform infrared spectroscopy/FTIS and/or showing close hydrodynamic dimensions to a polypeptide chain under physiological conditions. In the studies, Uversky et al. pointed out that the proteins were not in ordered structure via low mean hydrophobicity and relatively high net charge relation. In this thesis, 79 out of 91 proteins in certainly disordered structure were used [Yang et al, 2006]. The composition of the completely disordered data set called CD79 in this study is shown in Table 3.3.

Table 3.3. The composition of CD79 data set.

	CD79
Number of Chains	79
Number of Ordered Regions	0
Number of Disordered Regions	79
Number of Ordered Residues	0
Number of Disordered Residues	14462
Total Residues	14462

From the 3 chosen protein sets, 2 data sets with different characteristics were constructed. The first data set was created by using D80 protein set which contains disordered regions and thus having an unbalanced distribution. This set was used as blind test set for comparing the success of the proposed method with existing predictors. The second set was developed by balancing equal amounts of completely ordered (CO) and completely disordered (CD) protein data sets, named as COD159. The data set was used for training and testing of the methods to measure the performances.

3.2. Data Representation

In order to develop successful models in estimating the protein structure, it is beneficial to consider amino acid neighboring in knowledge representation [Dunker et al., 2001]. For this reason, the information about an amino acid in *i*th position of a protein is used together with the amino acids surrounding it. According to this, all amino acids' knowledge in a *k* residue length window is included to represent the central amino acid of the window. Through sliding the window, all residues in the protein are scanned in order to extract information for each sequence position [Qian and Sejnowski, 1988]. However, inclusion of each amino acid individually in the window causes a *curse of dimensionality* [Bishop, 1995]. Besides, using 20bits of binary representation for these amino acids enlarges this problem. In order to avoid this dimensionality problem, it has been preferred to use the real value representations of amino acids and the average information of all the amino acids within the window [Dunker et al., 2001] [Peng

et al., 2005]. In this way, each feature is represented by only one attribute in a pattern. This also removes the dependency on the window size. For *n* attributes, an *n*-dimensional pattern is created for a given position of protein and labeled with the class of the corresponding amino acid at that position. Here, class label can take the value of ordered or disordered.

The estimation of disordered regions is considered as a binary classification problem. The per-residue prediction values obtained by the computational learning methods, take the value of "1" for ordered or "0" for disordered. In this study, new patterns of amino acids in the data sets were constructed in 3 different manners as training inputs for machine learning methods.

The first step for obtaining network input is to create initial profiles for each amino acid of all proteins in the data set by means of the sliding window technique. The centre of a window, with a size of an odd number, is placed targeting an amino acid and starting from the first residue; this is repeated by sliding one residue to the right until the end of sequence, thus all residues of a sequence are treated. In this study, the size of the sliding window was determined as 21 by considering the references [Linding et al., 2003a] [Dosztányi et al., 2005]. N and C terminals of proteins can cause bias as they have tendency to be disordered [Li et al, 1999]. For this reason, the regions of the protein remaining outside the window with a number of $(w-1)/2$ amino acids from the beginning and from the end were excluded.

Previous studies have shown that the several physicochemical properties of amino acids can be used for distinguishing disorder. Therefore, in the next step, by using property, composition and evolution values of the amino acids in the window, input profiles were constructed for each residue within a protein sequence. Then, these profiles were combined to derive an input pattern. The pattern attributes are named as $X(i)$ where i is the attributes number.

Finally, for each attribute, the values of the input patterns were re-scaled between 0-1 with min-max normalization technique (Equation 3.1).

$$v' = \frac{v - \min_A}{\max_A - \min_A}(new\_\max_A - new\_\min_A) + new\_\min_A \quad (3.1)$$

where v' is the new scale value for an attribute. $new\_\min_A$ and $new\_\max_A$ are given as the expected minimum and maximum values respectively, here 0 and 1. $\min_A$ and $\max_A$ get the minimum and maximum values of an attribute in the data set.

3.2.1. Property-Based Profiles

In the thesis, 49 property scales of amino acids from AAIndex and from different references were used for constituting property-based profiles. The 6 known scale values of physicochemical properties of amino acids obtained from AAIndex are given in Table 3.4 [Lise and Jones, 2005] [Xie et al, 1998] [Williams et al., 2001] [Radijovac, 2003]. In the table, the scales have been presented as; Hydro: Hydrophobicity [Kyte and Doolittle, 1982], Flexi: Flexibility [Vihinen et al.,1994], Bulki: Bulkiness [Zimmerman et al, 1968] Pol: Polarity [Grantham, 1974] , Alpha: Alpha Helix Propensity [Koehl-Levitt, 1999] and Beta: Beta Helix Propensity [Koehl-Levitt, 1999]. The contents and the references of all properties and their scale values are given in Appendix A.

The calculation of X_{iprop} attributes for *i*th position residue of a sequence with length N within the window of length w is given in Equation 3.2.

$$X_{iprop} = \frac{1}{w}\sum_{j=w_s}^{w_f} prop_j \quad (3.2)$$

where $w_s = i - (w-1)/2$, $w_f = i + (w-1)/2$, $prop_j$ is the scale value of a given property for the amino acid at position *j*.

Table 3.4. The scale values of 6 different physicochemical properties of amino acids.

AA	*Hydro*	*Flexi*	*Bulki*	*Pol*	*Alpha*	*Beta*
A	1.8	0.36	11.5	8.1	-0.040	-0.120
C	2.5	0.35	13.46	5.5	0.570	-0.630
D	-3.5	0.51	11.68	13.0	0.270	1.120
E	-3.5	0.5	13.57	12.3	-0.330	0.910
F	2.8	0.31	19.8	5.2	0.010	-0.670
G	-0.4	0.54	3.4	9.0	1.240	0.760
H	-3.2	0.32	13.69	10.4	-0.110	1.340
I	4.5	0.46	21.4	5.2	-0.260	-0.770
K	-3.9	0.47	15.7	11.3	-0.180	0.290
L	3.8	0.37	21.4	4.9	-0.380	0.150
M	1.9	0.3	16.2	5.7	-0.090	-0.710
N	-3.5	0.46	12.82	11.6	0.250	1.050
P	-1.6	0.51	17.4	8.0	0.000	0.000
Q	-3.5	0.49	14.45	10.5	-0.020	1.670
R	-4.5	0.53	14.28	10.5	-0.300	0.340
S	-0.8	0.51	9.47	9.2	0.150	1.450
T	-0.7	0.44	15.77	8.6	0.390	-0.700
V	4.2	0.39	21.57	5.9	-0.060	-0.700
W	-0.9	0.31	21.67	5.4	0.210	-0.140
Y	-1.3	0.42	18.03	6.2	0.050	-0.490

As the final property-based attribute, the measure of sequence complexity called *K2* entropy was used in the thesis [Wootton and Federhen, 1996]. Complexity is a measure which shows how many different ways a sequence can be rearranged [Weathers, 2004]. It has been demonstrated that low complexity regions are more likely to be disordered than ordered [Peng et al., 2005] [Romero et al., 2001]. Shannon's *K2* entropy, X_{iK2} value over a window was calculated from the 20 amino acid frequencies, X_{ia} within the window (Equation 3.3).

$$X_{iK2} = -\sum_{a=1}^{20} X_{ia} \log_2 X_{ia} \tag{3.3}$$

Consequently, 50 attributes of an input pattern were derived from the property-based profile.

3.2.2. Composition-Based Profiles

In the construction of composition-based profiles, first order statistics of 20 known amino acids and compositions for 30 different property groups of amino acids within the window were examined.

The attendance of each known 20 amino acids within the window, $a \in$ {A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y}, constitutes the first 20 attributes of the profile. The next 30 attribute values of the profile are derived from the frequency of the specified amino acids belong to the same property groups in a given window (Appendix A). A total of 50 attributes constitutes a composition-based profile.

For the *i*th position residue of a sequence with length *N*, the calculation of X_{ia} attribute value corresponding to an amino acid within the window of length w is given in Equation 3.4.

$$X_{ia} = \frac{1}{w} \sum_{j=w_s}^{w_f} P_{ja} \tag{3.4}$$

where $w_s = i - (w-1)/2$ and $w_f = i + (w-1)/2$, $P_{ja} = 1$ if amino acid a is at position j, otherwise $P_{ja} = 0$.

Similarly, an attribute value of a group composition is calculated by averaging the numbers of amino acids which pertain to a given property group within the window. For instance, according to Table 1.1, the sum of X_{ia} values of amino acids which have hydrophobicity feature, $a \in$ {A, I, L, M, F, P, W, V, G},

provides the X_i(property) attribute value of the property group (Equation 3.5).

$$X_i(\text{hydrophobicity}) = X_{iA} + X_{iI} + X_{iI} + X_{iM} + X_{iF} + X_{iP} + X_{iW} + X_{iV} + X_{Ig} \quad (3.5)$$

Here, net charge value for a residue should be calculated considering the sign of that amino acid. Based on this, the number of negative charged amino acids, $a \in \{D, E\}$, is subtracted from the number of positive charged amino acids, $a \in \{K, R\}$, (Equation 3.6).

$$X_i(\text{net charge}) = X_{iK} + X_{iR} - X_{iD} - X_{iE} \quad (3.6)$$

3.2.3. Evolution-Based Profiles

It was shown that the use of evolutionary knowledge improves the accuracy of disorder prediction. [Jones and Ward, 2003] [Ward et al., 2004a] [Su et al., 2006]. Therefore, position-specific scoring matrices (PSSM) generated by PSI-BLAST search were used to construct evolution-based profiles in the study.

PSI-BLAST provides a measure of residue conservation in a given position [Altschul et al., 1997], and returns 20 dimensional vector representing probabilities of conservation against mutations to 20 different amino acids. Thus, for a sequence of length *N*, an *N*x20 matrix called PSSM is formed. PSI-BLAST search was performed against a filtered non-redundant sequence database, Uniref100 with 3 iterations by using “blastpgp” program.

The moving-average approach was used for obtaining evolution-based profiles as in construction of the other profiles to avoid the curse of dimensionality. The 20 columns of the matrix were averaged over the input window, resulting in the second set of 20 features.

For the *i*th position residue of a sequence, X_{ia} attribute is calculated as given in Equation 3.7.

$$X_{ia} = \frac{1}{w} \sum_{j=w_s}^{w_f} M_{ja} \tag{3.7}$$

where M_{ja} is the PSSM element at residue (row) j for the column of a amino acid. The procedure is given in Figure 3.1.

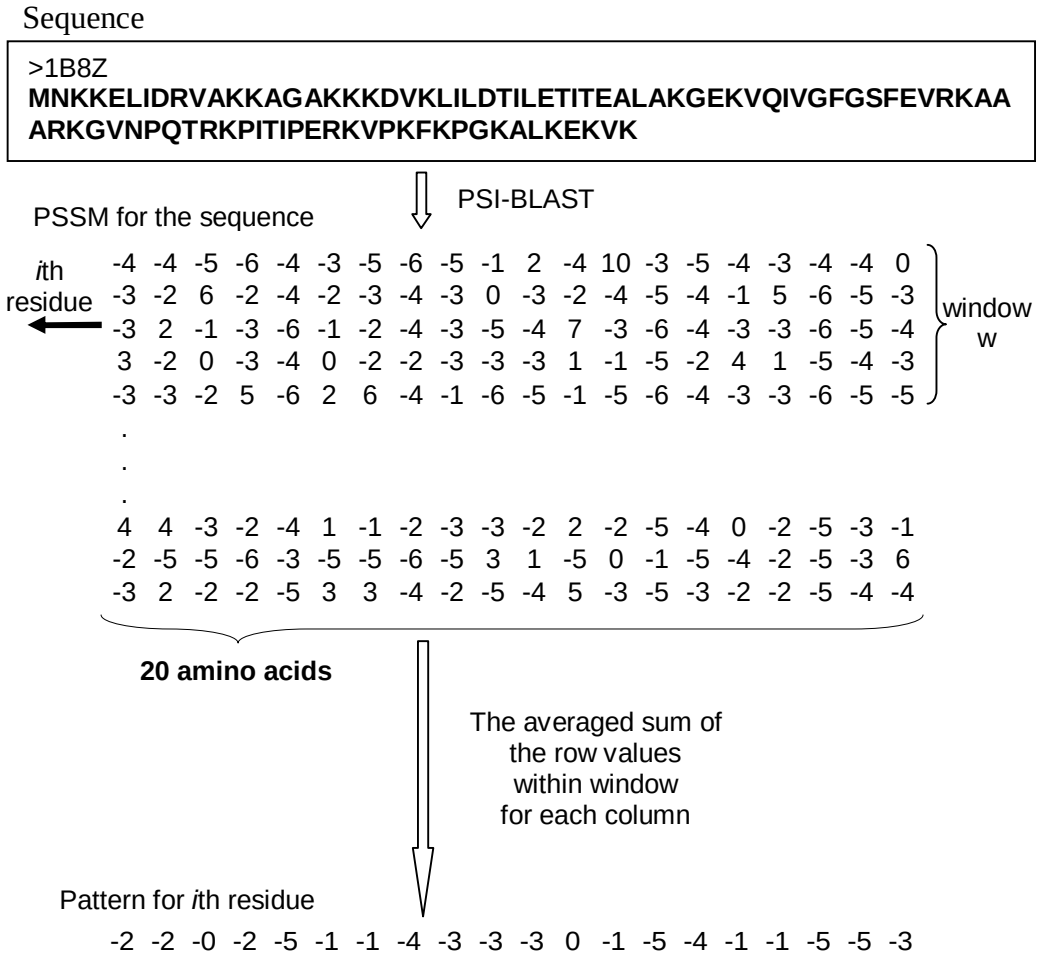


Figure 3.1. The constructing of PSMM profile of a protein within a given window.

3.3. Performance Assessment

There are many measures for quantifying the prediction accuracy in a binary classification problem, herein order and disorder. Four widely used indices

can be given as *Sensitivity*, *Specificity*, *Accuracy* and *Matthew' Correlation Coefficient* [Melamud and Moult, 2003] [Yang et al., 2005 /RONN] [Su et al., 2006]. The definitions of the measures are given by the following equations (Equation 3.8 – Equation 3.11);

$$\text{Accuracy } (Acc) = \frac{(\text{TP}+\text{TN})}{(\text{TP}+\text{FP}+\text{TN}+\text{FN})} \tag{3.8}$$

$$Sensitivity\ (Sens) = \frac{TP}{TP + FN} \tag{3.9}$$

$$Specificity\ (Spec) = \frac{TN}{TN + FP} \tag{3.10}$$

$$Matthews'\ Correlation\ Coefficient\ (Mcc) = \frac{(TP \times TN - FP \times FN)}{\sqrt{(TP + FP) \times (FP + TN) \times (TN + FN) \times (FN + TP)}} \tag{3.11}$$

where ***TP*** (true positive) is the number of correctly classified disordered residues, ***TN*** (true negative) is the number of correctly classified residues which have the ordered structure, ***FP*** (false positive) is the number of ordered residues incorrectly classified as disordered and ***FN*** (false negative) is the number of disordered residues incorrectly classified as ordered. Therefore ***sensitivity*** (***Sens***) and ***specificity*** (***Spec***) represent the fraction of correctly identified residues as disorder and order, respectively [Wu and McLarty, 2000]. The correlation coefficient, introduced by Matthews, gives 1, 0 and -1 for perfect, random and completely wrong predictions, respectively.

Although the ***matthews' correlation coefficient*** provides a much more balanced evaluation of the prediction than the percentages, all three measures are critically affected by the relative frequency of the target and they are not suitable for the isolated evaluation. For example, in a situation in which all residues are estimated to be disordered, taking a value of ***sensitivity*** equal to 1 yields a ***specificity*** equal to 0. Then, accuracy is rather affected by the relative frequencies of the two classes. A ***sensitivity*** of less than 50% but a ***specificity*** of more than 80% demonstrates an ***under-prediction*** of a predictor which has the tendency of predicting order more than disorder.

Therefore, ***probability excess*** has been recommended as an unbiased

measure for evaluating the performance of prediction by Yang et al. [Yang et al., 2005]. *Probability excess* is independent of the relative class frequencies by means of the evaluation of sensitivity and specificity values in cooperatively with *sensitivity* + *specificity* − 1, that can be graphed by a plot of *sensitivity* versus *specificity*. It is defined by Equation 3.12;

$$Probability\ Excess\ (ProbEx) = \frac{TPxTN - FPxFN}{(TP + FN)x(TN + FP)} \tag{3.12}$$

The values of greater than 0.5 reveal an acceptable prediction performance in *probability excess* criteria. Here the value of 1 is also an indicator of a perfect predictor. When the properties of all aforementioned evaluation measurements are considered, it can be said that the probability excess can be assumed as a more effective evaluation criteria; and accordingly it was the preferred measure to evaluate the success rate of the methods in our study.

3.4. Pattern Dimensionality Reduction

It is not convenient for multicollinearity or correlation between variables to exist at a significant level in estimation problems. This case may cause an overestimate as in all classification models. To deal with the problem, it is necessary to purify the data set from multicollinearity and to define the data set with fewer variables. The purpose of pattern dimensionality reduction process is to obtain the most appropriate purified data set for estimation.

In the investigation of the relationship between two or more variables, it is impotant to find which factors have an impact on the result and which variables have more effect among the variables. In the model, the affected variable is named as dependent variable and the variables which affect are named as independent variable.

Recently, being parallel to the changes in computer technology, there are many techniques that provide opportunity to assess many variables together and

to choose independent variables which explains the dependent variable best. In the literature, there are many statistical techniques that serve this aim, such as Classification and Regression Trees (CART), Multivariate Adaptive Regression Splines (MARS), Logistic Regression (LR), Principal Component Analysis (PCA).

In the study, Logistic Regression technique has been used for attribute selection, and Principal Component Analysis has been used for attribute extraction. The logic of attribute selection is to decrease data dimensionality by eliminating some of the pattern attributes without causing any loss of information in prediction. In the attribute extraction, the n-dimensional pattern attributes are recombined by passing through a transformation process to constitute lower dimensional patterns.

3.4.1. Principal Components Analysis (PCA)

Principal Components Analysis is a method that projects the multivariate data onto the subspace spanned by a set of orthogonal vectors in order to perform a dimensionality reduction while retaining as much of the available information (variance) as possible [Haykin, 1994]. The analysis is mathematically defined as the transformation of correlated variables of the data to a new set of uncorrelated principal components (PCs). The first principal component encapsulates the greatest variance among the other PCs. The second principal component is then lying in the subspace orthogonal to the first so as to maximize the variation of the projected data (Figure 3.2). The remaining components (if any) can be construed in a similar manner i.e. a PC is taken to be along the maximum variance direction in the subspace orthogonal and therefore uncorrelated to the remaining components. Hence most important aspects of the data are captured in the first few PCs. Here, dimensionality reduction can be performed by ignoring the remaining components of lesser significance.

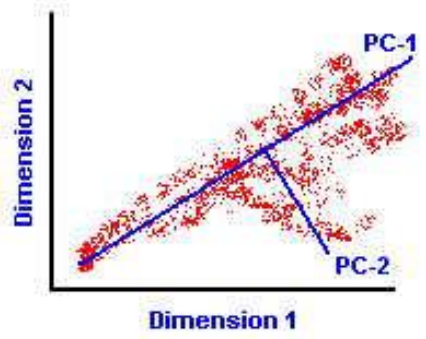


Figure 3.2. Two consecutive principle components for two dimensional data.

The starting point for PCA is to compute the mean for a given n independent m dimensional sample vectors, $\boldsymbol{X}_j = [x_1\ x_2\ \ldots x_m]^{\mathrm{T}}$ where $j = (1,\ 2,\ \ldots,\ \mathrm{n})$ and then subtracting the mean from each of the measurements $\boldsymbol{X}_j$ by $\boldsymbol{Y}_j = \boldsymbol{X}_j - m_x$. Hence, zero mean i.e. $E\{\boldsymbol{X}\} = 0$ is achieved.

For a data matrix Y (m x n) where each row represents the measurement of a particular probe while each column corresponds to a set of measurements from one repetition of the experiment, PCA linearly transforms each vector $\boldsymbol{Y}_j$ into a new vector $\boldsymbol{T}_j$ by an orthogonal matrix $\boldsymbol{P}$ with $\boldsymbol{T}_j = \boldsymbol{P}\boldsymbol{Y}_j$.

Here the covariance matrix of T ($\boldsymbol{C}_{\boldsymbol{T}} = \boldsymbol{T}\boldsymbol{T}^{\mathrm{T}} = \boldsymbol{P}\boldsymbol{C}_{\boldsymbol{Y}}\boldsymbol{P}^{\mathrm{T}}$) is diagonalized by P where the rows are the eigenvectors of covariance matrix of Y ($\boldsymbol{C}_{\boldsymbol{Y}} = \boldsymbol{Y}\boldsymbol{Y}^{\mathrm{T}}$). The diagonal values of $\boldsymbol{C}^{\mathrm{T}}$ denote the variances of $\boldsymbol{Y}$ along their associated principal components. The principal components of $\boldsymbol{Y}$ are the eigenvectors of $\boldsymbol{Y}\boldsymbol{Y}^{\mathrm{T}}$ that are sorted in the order of descending eigenvalues (largest first).

In order to perform the transformation, it is necessary to find the eigenvectors, $\boldsymbol{U}_k$ and the corresponding eigenvalues, l_k where $k = (1,\ 2,\ \ldots,\ \mathrm{m})$ such that $\boldsymbol{U} = \boldsymbol{P}^{\mathrm{T}}$. The solution of the following equation by means of identity transformation ($\boldsymbol{I}\boldsymbol{U}_k = \boldsymbol{U}_k$) gives the eigenvalues (Equation 3.13).

$$\boldsymbol{C}_{\boldsymbol{Y}}\boldsymbol{U}_k = l_k\boldsymbol{U}_k \tag{3.13}$$

To find the components of the eigenvectors, it is necessary to solve a system of linear equations given in a matrix form that is obtained by substituting the found eigenvalues in the above equation. The best low-dimensional space can efficiently be determined by a few of the eigenvectors corresponding to the largest eigenvalues.

3.4.2. Logistic Regression (LR)

Logistic regression provides a method that can describe a relationship only between a *dichotomous* response variable (*Y*) which takes a value of binary and a set of explanatory variables (***x***) in any types [Tabachnick et al., 2001]. That is, logistic regression makes no assumption about the distribution of the independent variables. In case of binary definition, the dependent variable can take only two possible outcomes; 1, with a probability of success *P* and 0, with a probability of failure 1 – *P*.

In logistic regression, the probability of $y_i = 1$ is given via a logistic function (Equation 3.14).

$$P(y_i = 1) = \frac{1}{1 + e^{-(b_0 + b_1 x_{1i} + b_2 x_{2i} + \ldots + b_m x_{mi})}} \tag{3.14}$$

where x_{ij} are the independent variables of *i*th dependent variable y_i (j=1, 2, …, m) and b_0, b_1, b_2, ..., b_j are known as the regression coefficients which have to be estimated from the data. The logistic regression model can be identified by the logit (the logarithm of the odds) transformation (Equation 3.15).

$$\text{logit}(P) = \ln\left[\frac{P}{1-P}\right] = b_0 + b_1 x_{1i} + b_2 x_{2i} + \ldots + b_m x_{mi} \tag{3.15}$$

The unknown parameters β_j are usually estimated by using the method of

maximum likelihood. A maximum likelihood estimation procedure maximizes the probability of getting the observed results given the fitted regression coefficients. To determine whether a variable should be included in the model or eliminated, logistic regression iteratively test the fit of the model after each coefficient is added or deleted, called stepwise logistic regression. If no significant effect is observed, then the resulting model excludes the variable. When no more variables can be included to, or removed from, the model, the analysis has been completed.

In the study, during the stepwise selection, the Chi-square test was performed to find the significance of the attribute's effect on the model [Li et al., 1999]. The significance level of 0.05 was used for choosing the attribute. The threshold level for order or disorder predictions was set to be 0.5 i.e. if $P \geq 0.5$, then the residue was labeled as disordered; otherwise, ordered.

3.5. Computational Methods

Border Vector Detection and Extended Adaptation (BVDEA) method was proposed in the study. In addition the three methods, Generalized Regression Neural Network (GRNN), Learning Vector Quantization (LVQ), Border Vector Detection and Adaptation (BVDA) were also carried out for benchmarking.

3.5.1. Generalized Regression Neural Networks (GRNN)

GRNN introduced by Donald Specht in 1990 is a memory-based feed forward neural network based on the estimation of Probability Density Function (pdf) from observed samples using Parzen-window estimation [Specht, 1991]. It approximates any arbitrary function (either linear or non-linear relationships between input and output vectors), drawing the appropriate function estimate directly from the training data. This approach removes the necessity to specify a functional form of the estimation.

The method is a kind of regression network that utilizes a probabilistic model between an independent vector random variable $\boldsymbol{X}$ (i.e. input) and a

dependent scalar random variable Y (i.e. output). It will be assumed that $\boldsymbol{x}$ and y are the particular measured values of $\boldsymbol{X}$ and Y random variables, respectively, and g ($\boldsymbol{X}$, Y) is the joint continuous probability density function of these random variables.

A good choice for non-parametric estimate of the probability density function g (as noted in Specht) is using the Parzen window estimator that is proposed by Parzen and performed for multidimensional cases by Cacoullos. Given a sample of n real d dimensional $\boldsymbol{x}_i$ values for a scalar y_i values, the estimate of joint probability density is (Equation3.16):

$$\hat{g}(x,y)=\frac{1}{(2p)^{(d+1)/2}s^{(d+1)}}\frac{1}{n}\sum_{i=1}^{n}\left[\exp\left(-\frac{(x-x_i)^T(x-x_i)}{2s^2}\right)\exp\left(-\frac{(y-y_i)^2}{2s^2}\right)\right] \quad (3.16)$$

where σ is the window width of a sample probability and called smoothing factor of the kernel [Yeung and Chow, 2002].

The expected value of Y given $\boldsymbol{x}$ (the regression of Y on $\boldsymbol{x}$) can be given as (Equation3.17):

$$E[Y/x]=\frac{\int_{-\infty}^{+\infty}Y.g(x,Y)dY}{\int_{-\infty}^{+\infty}g(x,Y)dY} \quad (3.17)$$

Substituting the joint probability estimate $\hat{g}(\boldsymbol{x}, y)$ into the definition of the conditional mean, $E[Y/x]$ yields (Equation3.18):

$$\hat{y}(x)=E[Y/x]=\frac{\sum_{i=1}^{n}\left[y_i\exp(d_i)\right]}{\sum_{i=1}^{n}\exp(d_i)} \quad (3.18)$$

where d_i is the distance function between the input vectors and the centers

recorded in the pattern nodes, which is given by Equation 3.19.

$$d_i = -\frac{(x - x_i)^T (x - x_i)}{2s^2} \tag{3.19}$$

The estimate $\hat{y}$ (x) is thus a weighted average of all the observed y_i valves where each weight is exponentially proportional to its Euclidean distance from $\boldsymbol{x}$.

3.5.2. Learning Vector Quantization (LVQ)

Learning Vector Quantization is a supervised classification algorithm based on a comparison with a number of so-called prototype vectors [Kohonen, 1984]. The learning is executed by adjusting the vector positions in order to describe optimal class boundaries.

LVQ applies a "winner takes all" approach and a nearest neighbor search (in the Euclidean distance sense) during the competitive learning procedure. For each training pattern, $\boldsymbol{x}_k$, the nearest prototype, $\boldsymbol{b}_w$ called "winner" is identified by Equation 3.20.

$$\begin{aligned} &\boldsymbol{D}_j(\boldsymbol{x}_k, \boldsymbol{b}_j) = \left\| \boldsymbol{x}_k - \boldsymbol{b}_j \right\|, \ \ j = 1, ..., m \\ &w = \arg\min\left\{ \boldsymbol{D}_j \right\} \end{aligned} \tag{3.20}$$

where m is the number of prototype vectors.

In classification, the data sample, $\boldsymbol{x}_k$ is assigned to the class according to the class label of the closest prototype vector. If a data sample is correctly classified (the labels of the winner vector and the data sample are the same, $y_w \neq y_k$), the winner vector is attracted towards the sample. If the vector is incorrectly classified, the data sample has a repulsive effect on the vector.

The adaptation procedure of the winner vector $\boldsymbol{b}_w$ is defined by the nearest-neighbor rule as given in Equation 3.21.

$$\boldsymbol{b}_w(t+1) = \boldsymbol{b}_w(t) \pm h(\boldsymbol{x}_k - \boldsymbol{b}_w(t)) \tag{3.21}$$

where h is the learning rate parameter and the sign depends on whether the data sample is correctly classified (+) or misclassified (−). The training algorithm involves an iterative adaptation and terminates if the prototypes stop to change significantly.

3.5.3. Border Vector Detection and Adaptation (BVDA)

The algorithm of the border vector detection and adaptation (BVDA) is based on the approach which is defined as partitioning of the feature space by using reference vectors selected from the training set [Kasapoğlu et al., 2007]. The processing which is named as border vector detection forms the initial step. In the next step, the selected vectors' adaptation is made in a way that is similar to the LVQ (Learning Vector Quantization) method [Kohonen, 1989].

In this method, class centers are used for the initial border vectors. The samples which are the nearest to the class mean are determined as class centers. The training set $\{(\boldsymbol{x}_1, y_1),\ (\boldsymbol{x}_2, y_2),\ \ldots,\ (\boldsymbol{x}_n, y_n)\}$, has *n* number of samples and *m* number of class labels, $\boldsymbol{x}_i \in \mathbf{R}^N$, $y_i \in \{1, 2, ..., m\}$.

Class means $\boldsymbol{o}_i$, are presented as an arithmetic mean of the training samples belonging to the related class. By using the obtained means, class centers are determined (Equation 3.22).

$$\boldsymbol{c}_i = \boldsymbol{x}_k, \left\{ \begin{array}{l} k = \arg\min\{\boldsymbol{D}_j\} \\ \boldsymbol{D}_j(\boldsymbol{o}_i, \boldsymbol{x}_j) = \sum_{d=1}^{N} \sqrt{(o_i(d) - x_j(d))^2},\ \{x_j \,|\, y_j = i\} \end{array} \right\}, (1 \le i \le m),\ (1 \le j \le n) \tag{3.22}$$

The attained class centers are the initial border vector set for all classes. Initial border vector set, $\mathbf{B_0}$ is given as $\mathbf{B_0} = \{(\boldsymbol{c}_1, y_1),\ (\boldsymbol{c}_2, y_2),\ \ldots,\ (\boldsymbol{c}_m, y_m)\}$. In

addition, $m_0 = m$ is the number of the members of the initial border vector set.

Initial border vector set constitutes the initial reference vectors for all classes. In the beginning, reference vector set of class i is defined as $\mathbf{R}_i(0) = \mathbf{B}_0$, $i = 1, 2, ..., m$. During the detection process, new border vectors are added to these reference vectors. At the end of the process, the reference vector set obtained for class i becomes $\mathbf{R}_i = \mathbf{B}_0 \cup \mathbf{B}_i$. The number of members of class i reference vector is determined as $(m_0 = m) + m_i$. $\mathbf{B}_i$ is the added border vector set and m_i is the number of its members.

In the detection procedure, all the training samples of the class $i=q$ are processed iteratively. Any of the input samples are compared with the given reference vectors of its class. If the input training sample $(\boldsymbol{x}_k, y_k)$ is not in the same class with $\boldsymbol{b}_w$ which is in the nearest Euclidean distance of it, then this sample is added to the reference vectors, and the process is continued with the next sample. The detection process of the nearest reference vector is given as follows (Equation 3.23);

$$\begin{aligned} &\boldsymbol{D}_j(\boldsymbol{x}_k, \boldsymbol{b}_j) = \left\| \boldsymbol{x}_k - \boldsymbol{b}_j \right\|, \ j = 1, ..., (m + m_t) \\ &w = \arg\min\left\{ D_j \right\} \end{aligned} \tag{3.23}$$

where m_t represents the number of the border vectors that are added until time t. If the class label of $\boldsymbol{b}_w$ detected as the nearest reference vector is $y_w \neq y_k = q$ then the sample $(\boldsymbol{x}_k, y_k)$ is added to the reference vector set $\mathbf{R}_{i=q}(t) = \mathbf{R}_{i=q}(t-1) \cup \{(\boldsymbol{x}_k, y_k = q)\}$. This procedure is repeated for all classes. The detected border vectors are combined before the adaptation process. At the time, $t = 0$, of the adaptation process the border vector set $\mathbf{B}^0$ is the combination of the border vectors and center vectors of all classes (Equation3.24). The attained vector set $\mathbf{B}^0$ is used for the partitioning of the feature space.

$$\mathbf{B}^0 = \bigcup_{0 \le i \le m} \mathbf{B}_i \tag{3.24}$$

In the beginning of the adaptation, the number of the total vectors, *nb* is given by $nb = \sum_{i=0}^{m} \mathrm{m}_i = m + \sum_{i=1}^{m} \mathrm{m}_i$. For the adaptation stage, first, the mean vectors are determined by computation of the arithmetic mean of the border vectors belonging to each class. These vectors are used in the adaptation process but not in the test stage. Mean vector set is defined as $\mathbf{M}^0 = \{(\boldsymbol{m}_1, y_1),\ (\boldsymbol{m}_2, y_2),\ \ldots,\ (\boldsymbol{m}_m, y_m)\}$.

Adaptation process is done by comparing $\mathbf{B}^0$ vectors with the training samples at each time t. The vector $\boldsymbol{b}_w$ and mean vector belonging to its class $\boldsymbol{m}_{y_{b_w}}$ are adapted together at time t. If the winning border vector is in the same class with the sample then it is brought closer. Otherwise, it is pushed away. At each time t, the vectors which are detected and adapted in the previous step are used. If the class of $\boldsymbol{b}_w$ border vector that is nearest to input training sample $\boldsymbol{x}_j$ is different from the class of sample then adaptation process occurs as given in Equation 3.25.

$$y_j \neq y_{b_w} \Rightarrow \begin{cases} \boldsymbol{b}_w(\mathrm{t}+1) = \boldsymbol{b}_w(\mathrm{t}) - h(\mathrm{t}).(\boldsymbol{x}_j - \boldsymbol{b}_w(\mathrm{t})) \\ \boldsymbol{m}_{y_{b_w}}(\mathrm{t}+1) = (m_{y_{b_w}}.\boldsymbol{m}_{y_{b_w}}(\mathrm{t}) - h(\mathrm{t}).(\boldsymbol{x}_j - \boldsymbol{b}_w(\mathrm{t})))/m_{y_{b_w}} \end{cases} \tag{3.25}$$

On the other hand, the nearest vector, $\boldsymbol{b}_l$ which has the same label with $\boldsymbol{x}_j$, is adapted as (Equation 3.26).

$$y_j = y_{b_l} \Rightarrow \begin{cases} \boldsymbol{b}_l(\mathrm{t}+1) = \boldsymbol{b}_l(\mathrm{t}) + h(\mathrm{t}).(\boldsymbol{x}_j - \boldsymbol{b}_l(\mathrm{t})) \\ \boldsymbol{m}_{y_{b_l}}(\mathrm{t}+1) = (m_{y_{b_l}}.\boldsymbol{m}_{y_{b_l}}(\mathrm{t}) + h(\mathrm{t}).(\boldsymbol{x}_j - \boldsymbol{b}_l(\mathrm{t})))/m_{y_{b_l}} \end{cases} \tag{3.26}$$

where learning rate $h(\mathrm{t})$ is given as a function decreasing in time. In the thesis the

most appropriate value was obtained by attempting different learning rate coefficients.

During training, a reference set is formed by combining the vector sets. $\mathbf{B}^t$ and $\mathbf{M}^t$ at the end of the predefined time t′. Then, input samples are compared with these reference vectors. If the winning vector is a mean vector $\boldsymbol{m}_w(t > t')$ and its class y_{m_w} is different from the sample class then that sample is added to border vectors set, $y_j \neq y_{m_w} \Rightarrow \mathbf{B}^{t+1} = \mathbf{B}^t \cup \{(\boldsymbol{x}_j, y_j)\},\ (t > t')$. Henceforth, the related mean vectors are adapted (Equation 3.27).

$$\boldsymbol{m}_{y_j}(t+1) = (m_{y_j}.\boldsymbol{m}_{y_j}(t) + \boldsymbol{x}_j)/(m_{y_j}(t)+1) \tag{3.27}$$

where $m_{y_j}(t)$ is defined as the number of the border vectors belonging to class y_j at the end of the iteration t.

3.5.4. Border Vector Detection and Extended Adaptation (BVDEA)

In the BVDA method, if the winner vector is not in the same class with given sample, it will be adapted. In this approach, the vector closest to the sample is sent away if it is not in the same class with the sample while the closest vector in the same class with the sample is brought closer to the sample. In this way, learning is realized in some of repetitions. In the proposed method, Border Vector Detection and Extended Adaptation (BVDEA), a rewarding is definitely provided in some iterations, namely learning is realized in each step.

Determination of vectors is applied in the way offered by the BVDA method. For a given training set, $(\boldsymbol{x}_i, y_i)$ which is composed of n number of samples and m class labels, $i \in \{1,2,...,n\}$, $y_i \in \{1,2,...,m\}$, first the closest samples to class means are determined (Equation 3.22). These vectors are assigned as reference vectors. Then, training samples of each class are respectively compared

with the reference vectors and given sample is added in reference vectors if it is not in the same class with the closest reference vector (Equation 3.23). Next, the following sample is compared with the new reference vector set.

Reference vectors existing for all classes are combined, and then border vectors are obtained, $\mathbf{B}=\{(\boldsymbol{b}_1, y_1), (\boldsymbol{b}_2, y_2), \ldots, (\boldsymbol{b}_{nb}, y_{nb})\}$ where *nb* is the total number of found vectors.

In the adaptation stage, all samples are compared with existing border vectors and the winner vector is determined according to the nearest neighbor rule. If the winner vector has not the same label with the sample, it is pushed away. On the other hand, the nearest vector which is in the class of input sample is brought closer to input vector, $\boldsymbol{x}$ (Equation 3.28).

$$y_j \neq y_{b_w} \Rightarrow \begin{cases} \boldsymbol{b}_w(t+1) = \boldsymbol{b}_w(t) - h(t).(\boldsymbol{x}_j - \boldsymbol{b}_w(t)) \\ \boldsymbol{b}_{y_j}(t+1) = \boldsymbol{b}_{y_j}(t) + h(t).(\boldsymbol{x}_j - \boldsymbol{b}_{y_j}(t)) \end{cases} \tag{3.28}$$

If the winner vector has the same label with the given sample, it is awarded by being brought closer (Equation 3.29).

$$y_j = y_{b_w} \Rightarrow \{\boldsymbol{b}_w(t+1) = \boldsymbol{b}_w(t) + h(t).(\boldsymbol{x}_j - \boldsymbol{b}_w(t)) \tag{3.29}$$

In this way, at least one border vector is ensured to be adapted in each repetition. This approach which shortens the learning period provides an improvement in generalization capability of the algorithm.

Vector adding approach used in BVDA has been extracted from the method BVDEA because supplementing the number of vectors during training may cause over-fitting and increase training time.

4. RESULTS AND DISCUSSION

In the problem of predicting disordered regions, the first step for applying the machine learning techniques is to prepare the dataset. For this purpose, a balanced data set was constituted from completely ordered and completely disordered proteins in this work. Then, the input pattern for each residue of the dataset was obtained by using the sliding windows with length equal to 21. Each pattern contains 120 attributes comprised of 50 composition-based attributes, 50 property-based attributes and 20 evolution-based attributes. Patterns are labeled as 0 or 1 if the class label of the corresponding residue is either order or disorder.

The aim of the study is to test the success of the prediction of the proposed method together with chosen optimum parameters and to compare the results with a few known methods. For this purpose, the proteins of the dataset were divided into 6 different subsets whose sequence lengths are approximately equal. Each subset includes the balanced number of order and disorder residues. One of these subsets was allocated as validation set for choosing the optimum parameters. Remaining 5 subsets were used for training the methods via 5 fold cross validation. Therefore, at each time, the different combinations of 4 subsets were used for training while the remaining subset was used for testing. The average of the success rates comes from 5 repetitions of training/testing. Thus, any dependency on some initializations such as detection of border vectors and order of training samples on determining the prediction success has been prevented by means of k-fold cross validation technique.

4.1. Evaluation Applications

In this work, initially, the obtained dataset COD159 was randomly partitioned into two parts as training and evaluation set to pre-evaluate the optimum learning rate (h) parameter of the proposed method. Then, BVDEA was trained for all h parameter values between 0.1 and 0.9 interval with 0.1 increments and also for the smallest values 0.01, 0.001, 0.0001. During training,

the prediction results both on training and evaluating sets were recorded for one of every 50 adaptations. For each candidate parameter, the maximum *probability excess* (*probEx*) values of the evaluating and training sets were found from among 12000 records. Besides, the *probEx* value of the evaluation set at the time of the maximum success on training self-consistency was assigned as the testing result of training with the corresponding *h* parameter. The maximum and the testing *probEx* values of the evaluation set are given in Table 4.1.

Table 4.1. The maximum and the testing *probability excess*es of the evaluation set.

h	***probEx* (max)**	***probEx* (test)**
0.0001	0.3709	0.3688
0.001	0.6374	0.6281
0.01	0.6612	0.5620
0.1	0.6674	0.5320
0.2	0.7562	0.6529
0.3	0.7831	0.6322
0.4	0.8006	0.7324
0.5	0.8016	0.7366
0.6	0.8071	0.6915
0.7	0.7758	0.6353
0.8	0.7872	0.6405
0.9	0.8192	0.6901

According to the testing results of the evaluation set, the learning rate parameters of 0.4, 0.5, 0.6 and 0.9 were determined as the optimum parameter selection of the method. Here, since 0.4, 0.5 and 0.6 values of rate are rather comparable, the only value of 0.5 is used for further investigation.

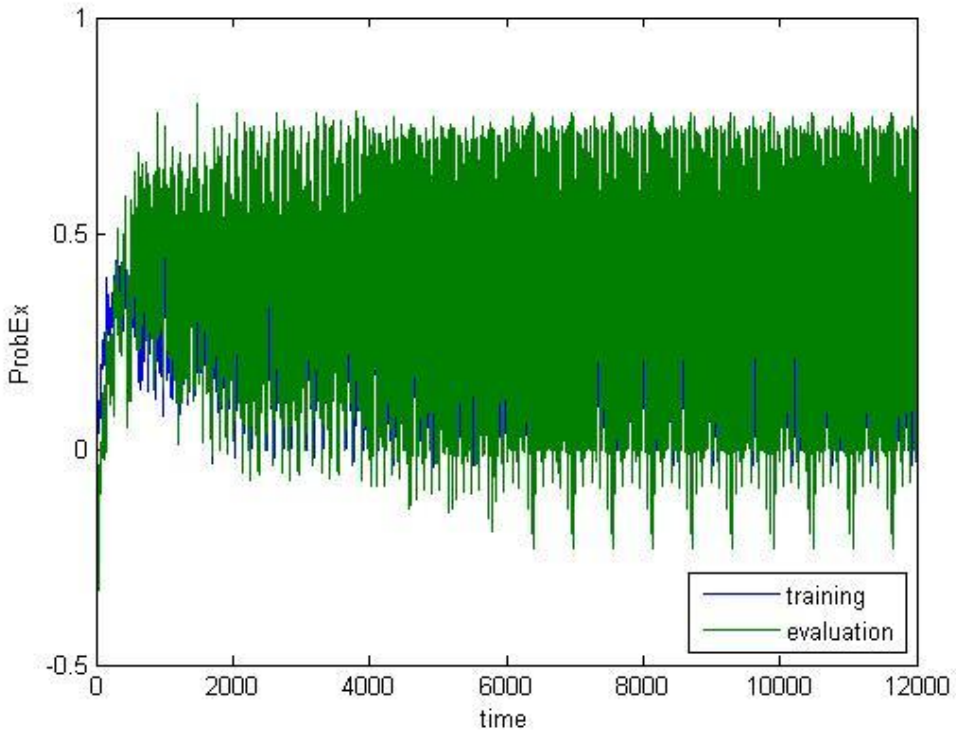


Figure 4.1. Prediction results of both evaluation and training sets at $h = 0.5$.

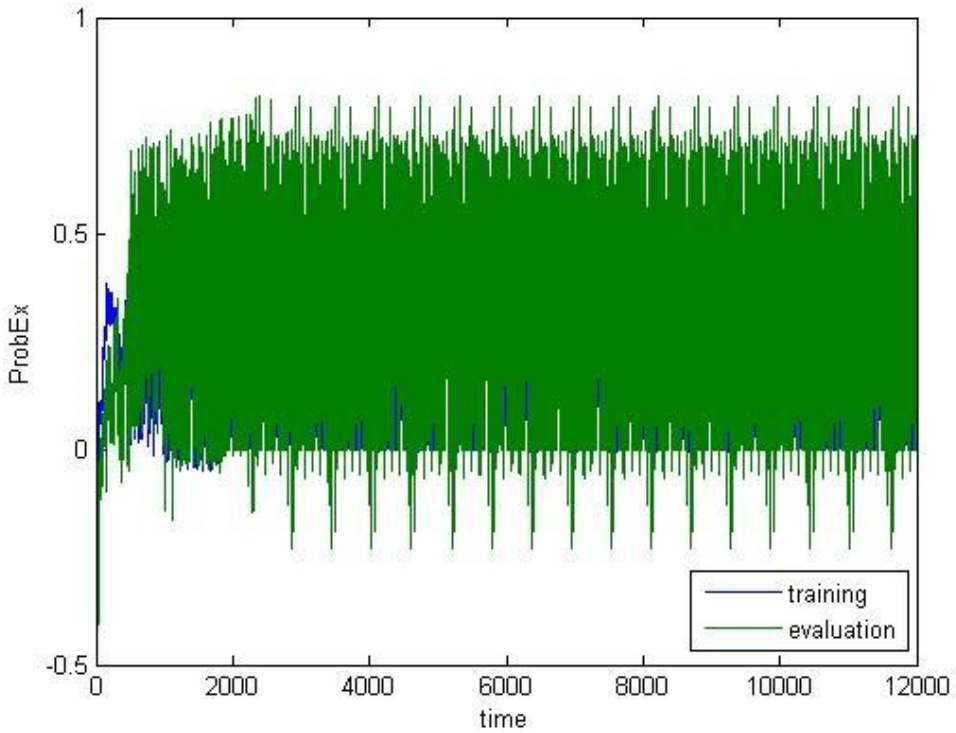


Figure 4.2. Prediction results of both evaluation and training sets at $h = 0.9$.

However, using a constant rate during training causes oscillation in success values, especially at high rates (Figure 4.1, Figure 4.2). The plots in Figure 4.1 and Figure 4.2 show the prediction results of both evaluation and training sets for the chosen *h* parameters, 0.5 and 0.9 respectively. The oscillation can be seen in these figures.

The continuing oscillation during training makes difficult to determine an optimum stopping criterion. Hence, it has been necessary to use an exponentially descending function of time for learning rate (Equation 4.1).

$$h(t) = h_0 e^{-t/t} \tag{4.1}$$

where, τ parameter is added to the system. Kasapoğlu et al. determined the optimum values for the pair of *h*-*t* parameter as 01-1000. For complex problems, the pair of 02-6750 was suggested. Nevertheless, since the appropriate values of *h* was determined as 0.5 and 0.9 for BVDEA, several combinations of these two values (*h*-*t*) were evaluated in this study. The evaluated values are given as $h \in \{0.5, 0.9\}$ and $t \in \{6750, 9500, 11500, 13500, 15000\}$.

The evaluations were performed after 5 cross training and testing applications for the reorganized data that was mentioned above. For choosing candidate parameter pairs, the success values of all testing subsets have been taken into consideration, concurrently, to make a general evaluation. The maximum success rates in *probability excess* for all testing sets were obtained at each parameter pair. The values are given in the following tables for each subset (Table 4.2-Table 4.6). It is noted that these maximum values are not accepted as prediction success of the method. The test successes of the method are determined after evaluation and validation applications.

By examining the results given in Table 4.2-Table 4.6, the 05-9500, 05-13500, 09-13500 and 09-15000 *h*-*t* pairs were determined as candidate parameters for the method.

Table 4.2. The maximum probability excesses of testing subset 1 at given *h*-*t* pairs.

h-*t*	***probEx* (max)**	*h*-*t*	***probEx* (max)**
0.5-6750	0.6426	0.9-6750	0.6333
0.5-9500	0.6767	0.9-9500	0.6467
0.5-11500	0.6798	0.9-11500	0.6653
0.5-13500	0.6746	0.9-13500	0.7180
0.5-15000	0.6808	0.9-15000	0.6777

Table 4.3. The maximum probability excesses of testing subset 2 at given *h*-*t* pairs.

h-*t*	***probEx* (max)**	*h*-*t*	***probEx* (max)**
0.5-6750	0.4317	0.9-6750	0.3309
0.5-9500	0.4338	0.9-9500	0.4275
0.5-11500	0.3971	0.9-11500	0.4191
0.5-13500	0.4884	0.9-13500	0.4160
0.5-15000	0.4160	0.9-15000	0.3981

Table 4.4. The maximum probability excesses of testing subset 3 at given *h*-*t* pairs.

h-*t*	***probEx* (max)**	*h*-*t*	***probEx* (max)**
0.5-6750	0.5685	0.9-6750	0.6107
0.5-9500	0.6148	0.9-9500	0.6004
0.5-11500	0.6097	0.9-11500	0.5788
0.5-13500	0.5984	0.9-13500	0.5644
0.5-15000	0.5911	0.9-15000	0.5994

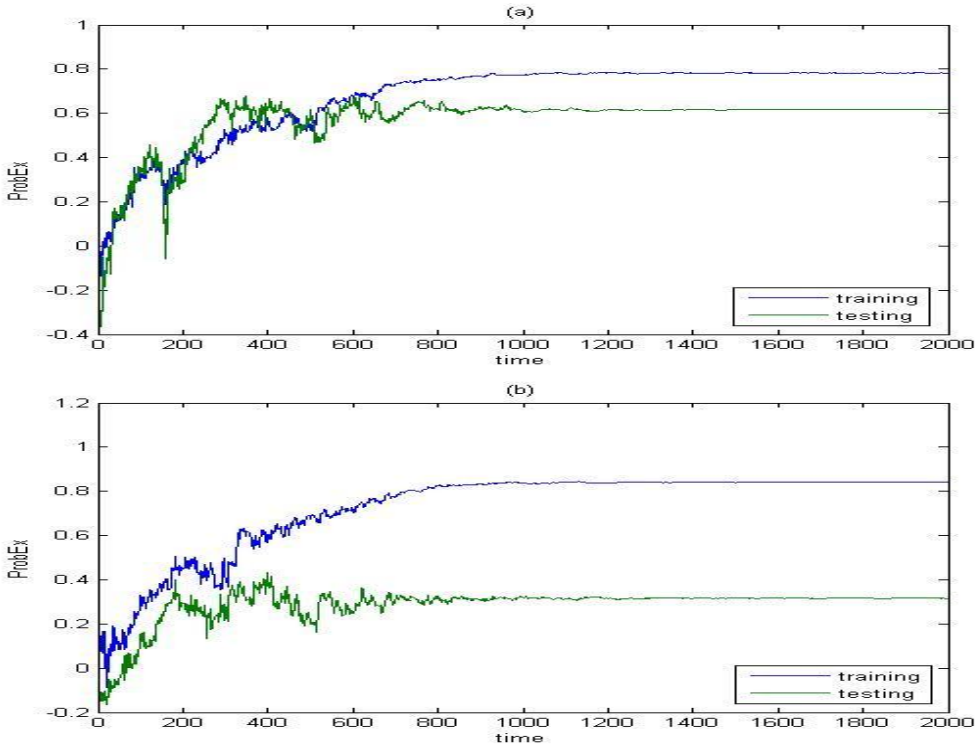
Table 4.5. The maximum probability excesses of testing subset 4 at given *h*-*t* pairs.

h-*t*	***probEx* (max)**	*h*-*t*	***probEx* (max)**
0.5-6750	0.4801	0.9-6750	0.4619
0.5-9500	0.5295	0.9-9500	0.4662
0.5-11500	0.4404	0.9-11500	0.5274
0.5-13500	0.4415	0.9-13500	0.4984
0.5-15000	0.4415	0.9-15000	0.5134

Table 4.6. The maximum probability excesses of testing subset 5 at given *h-t* pairs.

h-t	***probEx* (max)**	*h-t*	***probEx* (max)**
0.5-6750	0.6320	0.9-6750	0.5876
0.5-9500	0.6237	0.9-9500	0.6144
0.5-11500	0.6062	0.9-11500	0.6155
0.5-13500	0.6072	0.9-13500	0.6144
0.5-15000	0.6340	0.9-15000	0.6515

Any other general evaluation was verified by investigating the training behavior for chosen parameter pairs. For this purpose, for each candidate pair, the prediction results of both testing and training sets of each subset for the chosen *h-t* parameters, 05-9500, 05-13500, 09-13500 and 09-15000 were graphed regarding the *probability excess* values of 2000 records (Figure 4.3-Figure 4.6). Figure 4.3, Figure 4.4, Figure 4.5 and Figure 4.6 show the plots of training and testing for the pairs 0.5-9500, 0.5-13500, 0.9-13500 and 0.9-15000, respectively. Each figure contains 5 plots for 5 subsets.



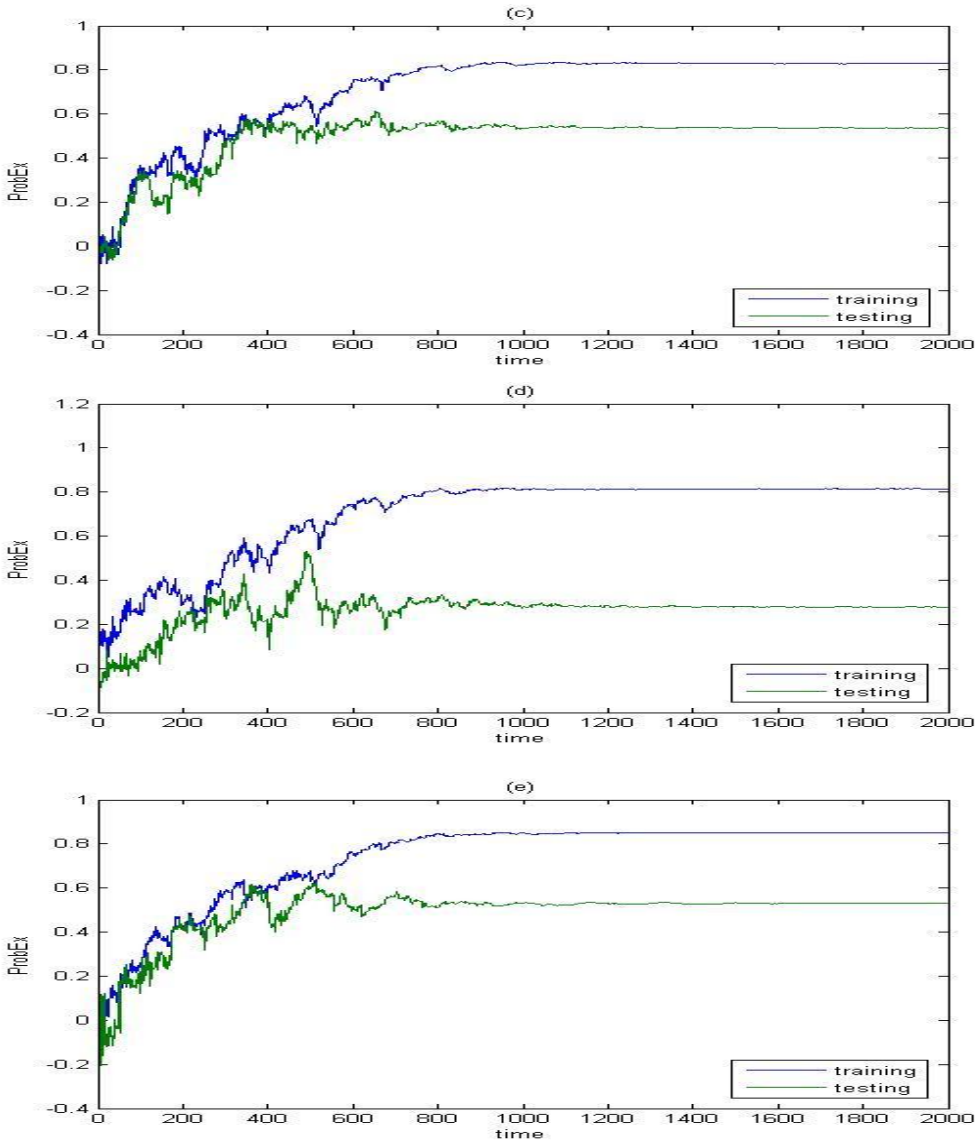
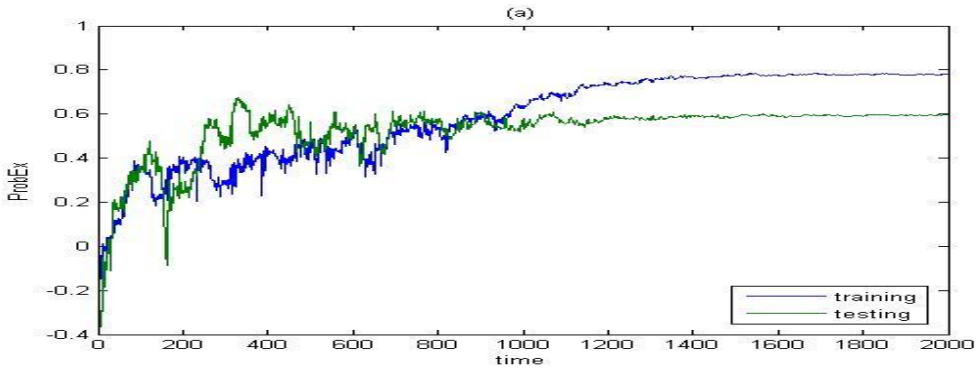


Figure 4.3. Prediction results of both testing and training sets at $h-t=0.5-9500$ (a) for subset 1 (b) for subset 2 (c) for subset 3 (d) for subset 4 (e) for subset 5.



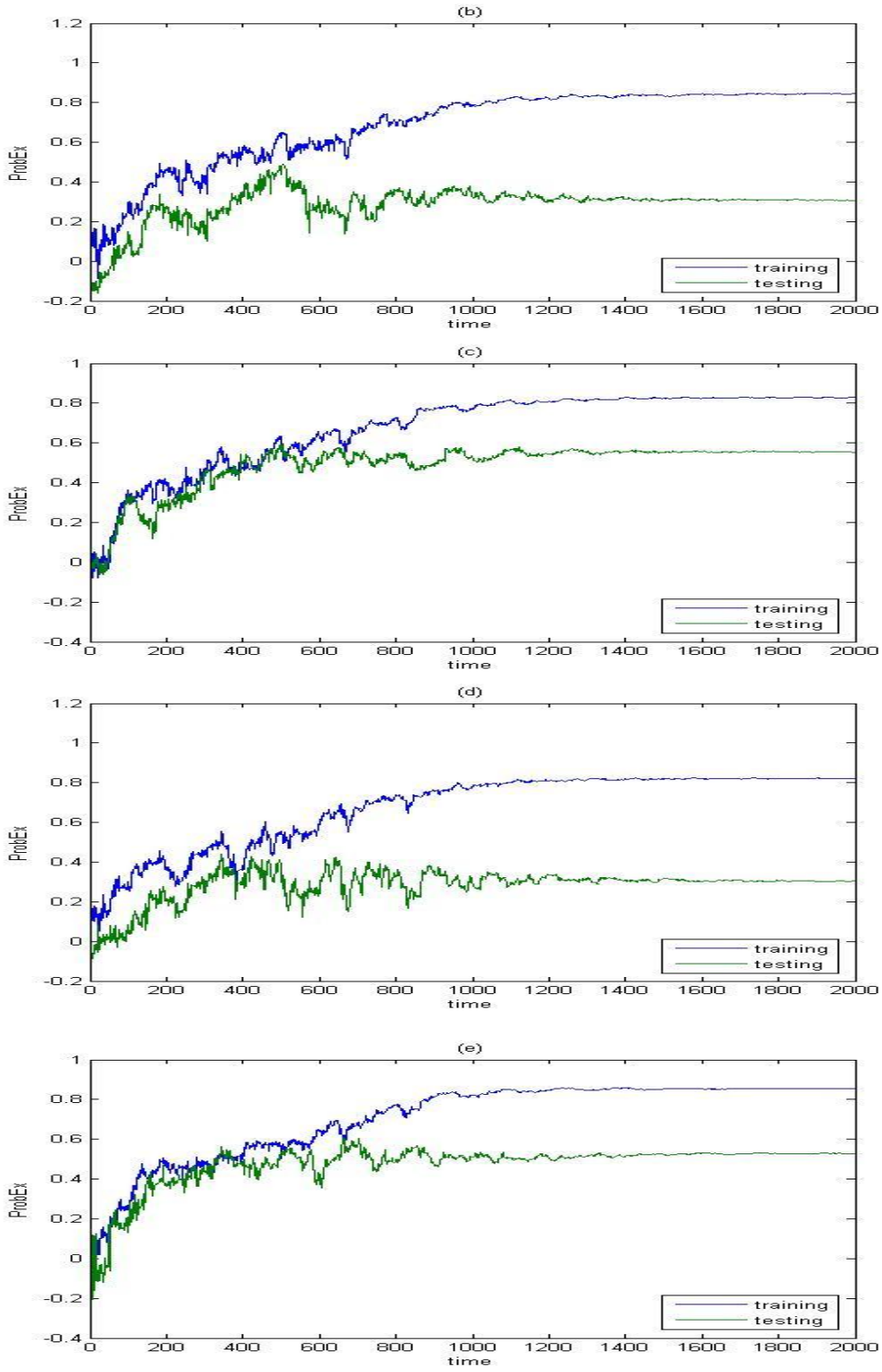
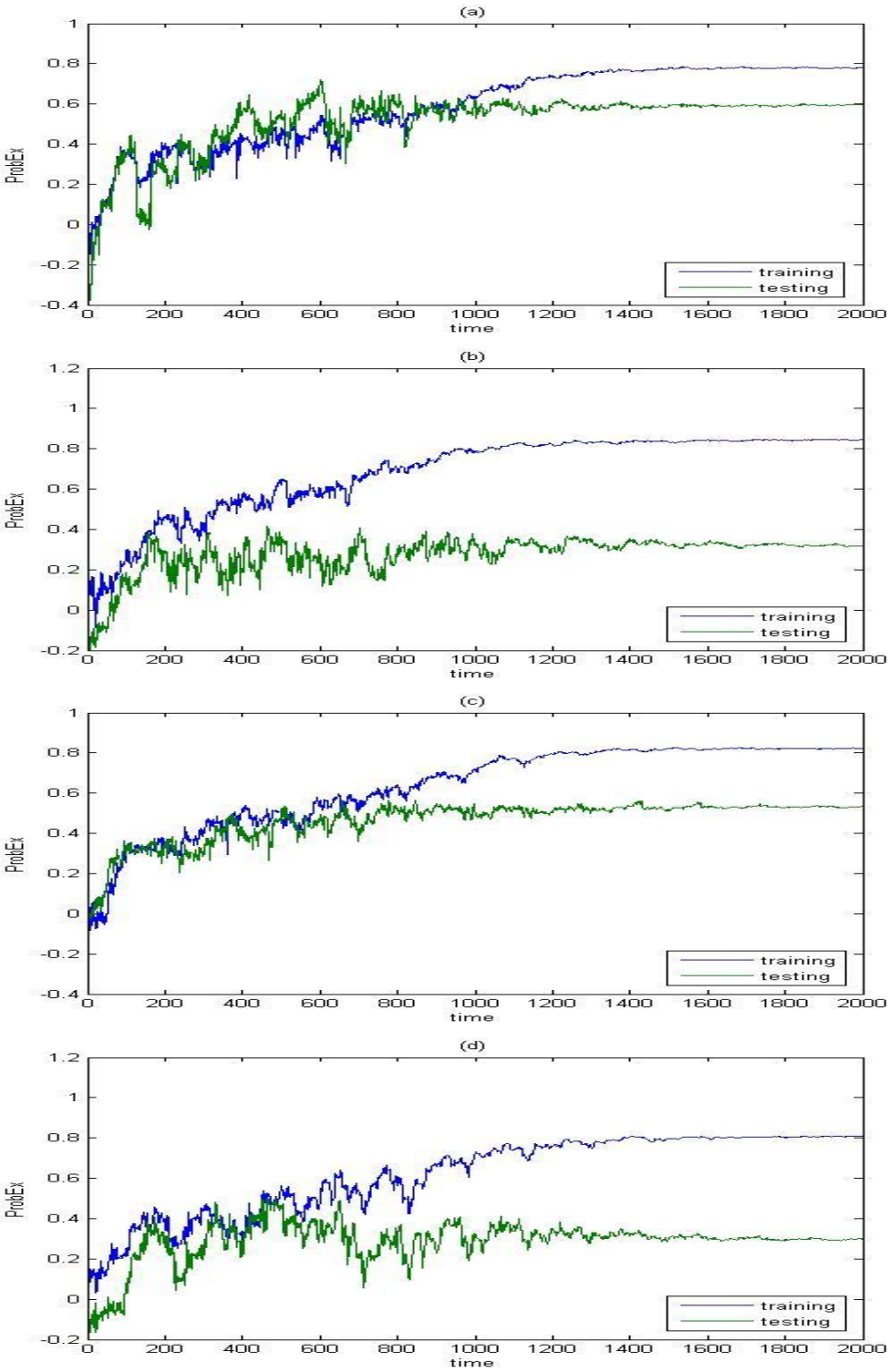


Figure 4.4. Prediction results of both testing and training sets at $h-t=0.5-13500$ (a) for subset 1 (b) for subset 2 (c) for subset 3 (d) for subset 4 (e) for subset 5.


(a)
ProbEx
time
training
testing
(b)
ProbEx
time
training
testing
(c)
ProbEx
time
training
testing
(d)
ProbEx
time
training
testing

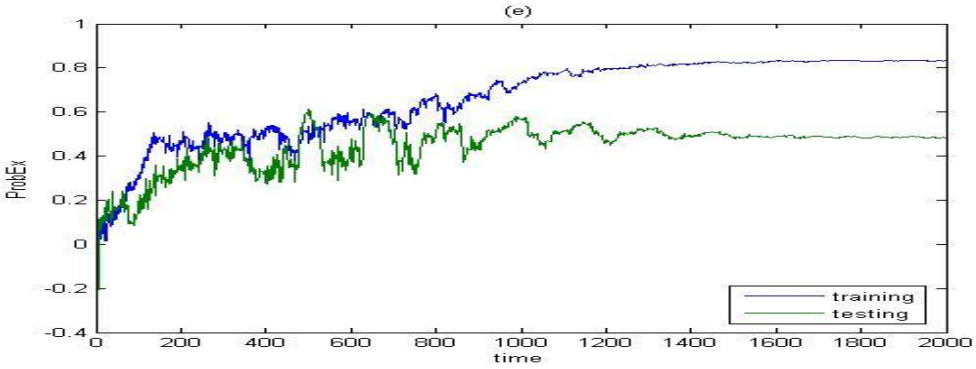
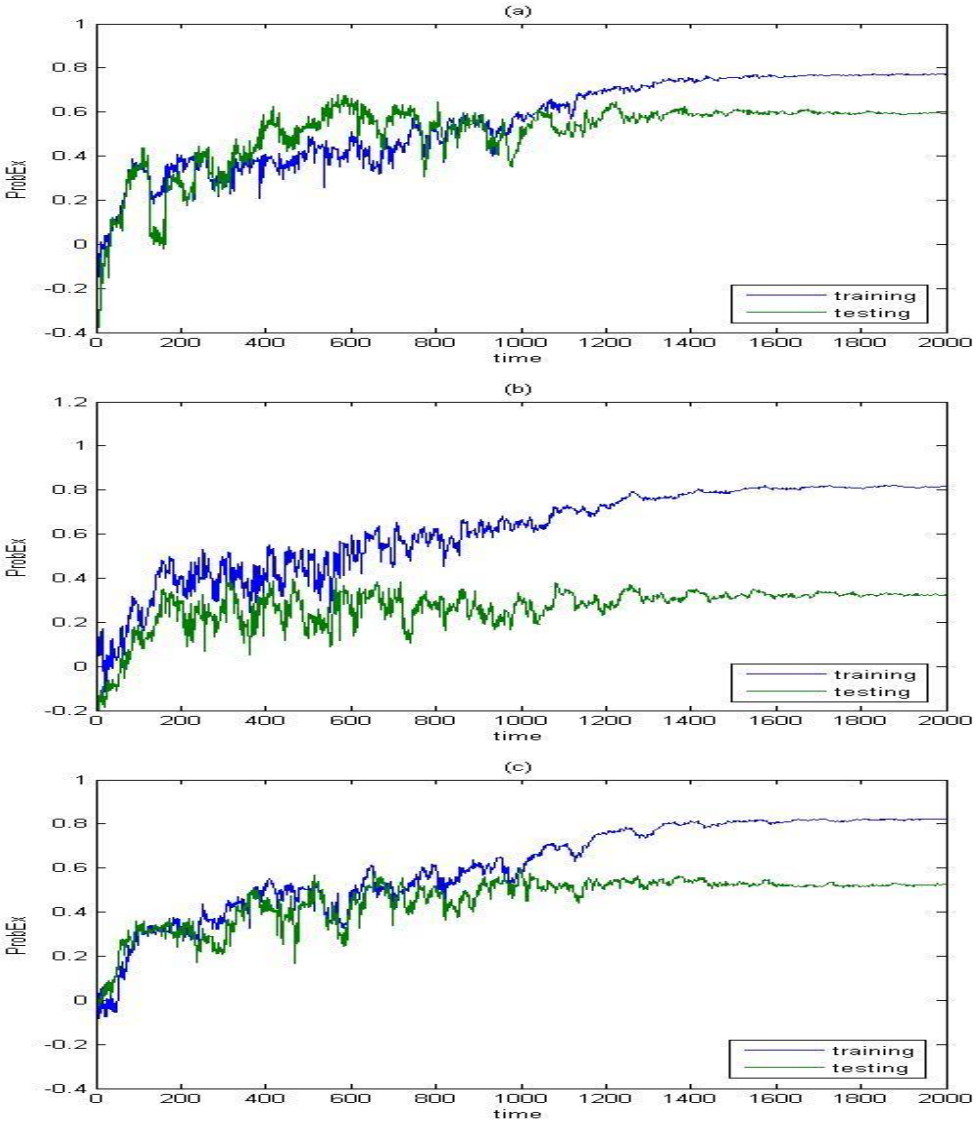


Figure 4.5. Prediction results of both testing and training sets at $h-t=0.9-13500$ (a) for subset 1 (b) for subset 2 (c) for subset 3 (d) for subset 4 (e) for subset 5.



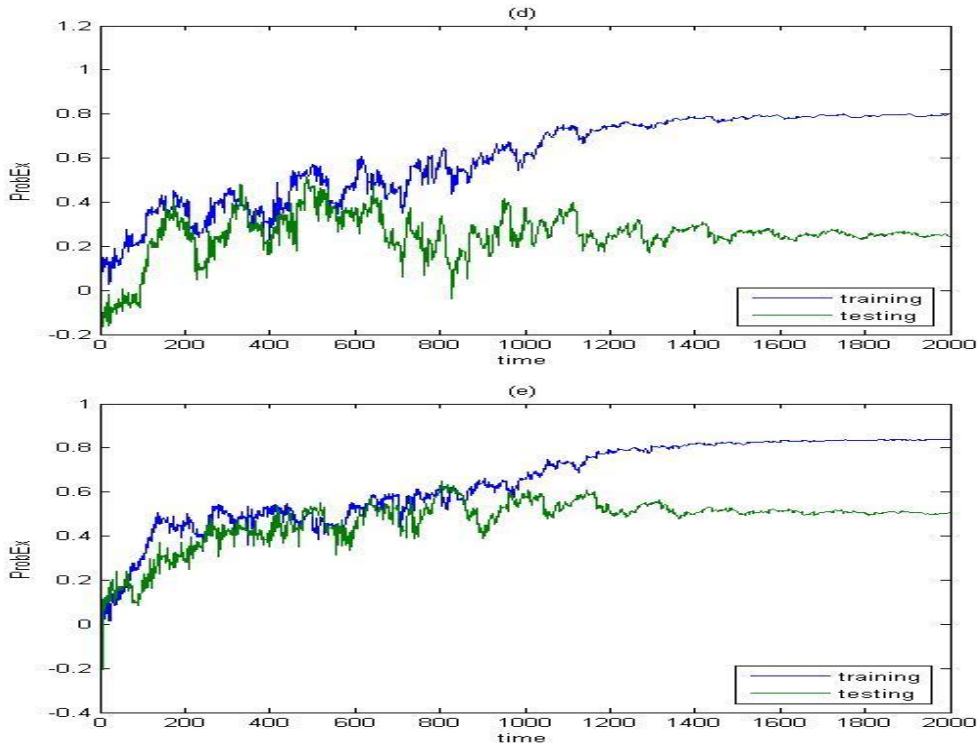


Figure 4.6. Prediction results of both testing and training sets at $h-t = 0.9-15000$ (a) for subset 1 (b) for subset 2 (c) for subset 3 (d) for subset 4 (e) for subset 5.

By examining the graphs presented in Figure 4.3, it is observed that the curves of the testing success for the h-t pair 0.5-9500 changes their direction at 450-500 levels of the records i.e. while the training success goes on increasing, the testing success begins to decrease at these levels, and the two curves start to separate from each other. Correspondingly, it has been decided that the training for this h-t pair had to go on up to level ~500 at most. For the other rate pairs, the corresponding graphs were also examined and this limitation was determined as ~550 for 0.5-13500 and ~700 for 0.9-13500, heuristically. On the other hand, in the graphs for the parameter pair of 0.9-15000, the regions of separation for the curves exhibit significant diversity. For example, while the level is approximately 500 in the curve of subset 4, it is almost 1000 for subset 5 as seen in Figure 4.6. This precludes performing a generalization, so the pair 0.9-15000 was eliminated from the evaluations.

After the general evaluations, validation applications were performed to determine the parameter pairs belonging to the related subset. In order to make a decision, the training of a subset with all candidate parameter pairs is continued until predefined times (record level) and then, up to this time, it is investigated at which time the training set has the maximum success value i.e. the maximum self-consistency result of training in *probEx*.

The results attained from testing of validation set were obtained according to the decision regions described by the border vectors at stopping points of trainings. The parameter pair that gives the most successful validation result for the corresponding subset is then assigned to the associated subset. The validation results of training BVDEA with 3 candidate parameter pairs, obtained for each subset, are given in Appendix B.1. The optimum parameter pairs selected for subsets are presented in Table 4.7. The stopping times of training with related parameter pairs determined by the self-consistency results are given for each subset in Table 4.8.

Table 4.7. The optimum η-τ pairs of training for each subset.

Subset 1	**Subset 2**	**Subset 3**	**Subset 4**	**Subset 5**
0.9-13500	0.5-13500	0.5-13500	0.5-9500	0.5-9500

Table 4.8. The stopping times of training for each subset in terms of record number.

Subset 1	**Subset 2**	**Subset 3**	**Subset 4**	**Subset 5**
604	503	498	494	494

For attaining prediction results of BVDEA, the classification of testing subsets was accomplished according to the border vectors of the training which was performed by using the optimum *h*-*t* pairs (Table 4.7) and was stopped at the given time in Table 4.8. It has to be emphasized that the given stopping times are the times of the maximum training performance up to given limiting time. The rates of testing successes for the testing sets with the given settings are presented in Table 4.9.

Table 4.9. The testing probability excesses of the subsets for BVDEA.

Subset 1	Subset 2	Subset 3	Subset 4	Subset 5
0.7128	0.4811	0.5901	0.5177	0.5948

The results given in Table 4.9 show the prediction performance of BVDEA with subsets of data generated by aforementioned training applications. All were used for comparing the performance with the other computational methods and the other tools of disorder predictors, as presented later.

4.2. Comparison with Other Methods

To evaluate the performance of BVDEA on prediction disordered regions of a protein, it was compared with several methods. LVQ and BVDA have been preferred due to their similar learning approach with BVDEA. Besides, the method of GRNN was also included.

Before determining the success values for the chosen methods, some evaluation and/or validation applications were executed for choosing associated parameters.

A series of run which is similar to BVDEA were generated for the method BVDA. At first, in all subsets, trainings were realized for different h-t pairs. On this occasion, the candidate parameter pairs defined in BVDEA was compared with suggested values by Kasapoglu and Ersoy. The evaluated values were given as h-$t \in$ {0.1-1000, 0.2-6750, 0.5-9500, 0.5-13500, 0.9-13500}. The maximum success rates in terms of *probability excess* for testing the subsets at each parameter pair are given in Appendix B.2. It is observed that the optimum parameter values given in BVDEA causes more successful results than the suggested values in Kasapoğlu et al.'s article [Kasapoğlu et al., 2007]. Afterwards, the prediction results of both testing and training sets of subset 1, subset 3 and subset 5 with associated candidate h-t parameters, 09-13500, 05-13500 and 05-9500, respectively, were graphed regarding the *probability excess* values of 2000 records as examples for evaluation (Appendix C). It was

concluded that the limiting levels for the training parameter pairs are similar to the found levels in BVDEA.

However, some differences stand out in results. For example, in BVDA, adding new vectors during the training causes a continual increase in training success and so the success approaches 1. Furthermore, testing success usually gets its maximum value at earlier times of training. But, due to the oscillations, the success rates fluctuate between very high and very low values at around these regions. This makes it impossible to stop the training around these regions. As a result, because of the similarity in general evaluation results for both methods, the optimum parameters found in BVDEA for the subsets have been preferred. Anyway, using any other parameter values did not change the results significantly due to the nearness of the success values among parameter values (Appendix B.2)

Eventually, for attaining prediction results of BVDA, the prediction was performed according to the training settings which are the optimum h-t pairs given in Table 4.7 and the stopping times of trainings with related parameter pairs determined by the self-consistency results given in Appendix B.3. The testing successes for the testing sets are given in Table 4.10.

Table 4.10. The testing probability excesses of the subsets for BVDA.

Subset 1	Subset 2	Subset 3	Subset 4	Subset 5
0.6198	0.3372	0.5427	0.3974	0.5103

In the LVQ method, the number of codebook vectors (n_c) was considered as parameter. In this study, two values of n_c were validated for this number. One of them was determined as 50 because BVDEA was performed with approximately 50 vectors; but for the other, the smaller value, 10 was preferred to compete against the value of 50. The training in LVQ was carried on until 100 epochs given as a default value in MATLAB application of “learnlv1”.

According to the validation success results given in Appendix B.4, it was recognized that the most successful learning was achieved with the number of 10 for all subsets. Thus, 10 codebook vectors were used for training in LVQ.

The default value of learning rate, 0.01 was used for learning. It was demonstrated that the success was badly affected for greater values of rate. Testing results of validation set for rate 0.05 with 10 vectors are given in Appendix B.4 as an example.

The prediction results of LVQ for each subset were obtained by using the optimum number of vectors for subset trainings. The performances in terms of *probability excess* are provided in Table 4.11.

Table 4.11. The testing probability excesses of the subsets for LVQ.

Subset 1	Subset 2	Subset 3	Subset 4	Subset 5
0.7820	0.4412	0.5417	0.5627	0.4969

Similarly, GRNN was also trained with two different sigma parameters (σ) and validated to find the optimums for the subsets. Validation applications were verified with $\sigma \in \{0.3, 0.7\}$. The value, 0.3 was suggested in previous studies [Ersöz et al., 2004]. As a higher value, 0.7 was chosen to compete against the other. The values obtained for all subsets are listed in Appendix B.5. The value of 0.7 was found to give maximum success for all subsets. The classification of testing subsets was achieved by using the optimum σ values in training (Table 4.12).

Table 4.12. The testing probability excesses of the subsets for GRNN.

Subset 1	Subset 2	Subset 3	Subset 4	Subset 5
0.7180	0.4034	0.6220	0.5134	0.5474

Consequently, the overall performances for all methods were calculated by averaging the results of 5 subsets. The success values in terms of all measures for all methods are listed in Table 4.13. The given list is ordered concerning the *probability excess* measures.

Table 4.13. Comparing the performance of four prediction methods.

Methods	***Sens***	***Spec***	***Acc***	***Mcc***	***ProbEx***
BVDEA	0.7964	0.7850	0.7907	0.5858	0.5814
LVQ	0.6981	0.8707	0.7844	0.5818	0.5688
GRNN	0.7263	0.8345	0.7804	0.5714	0.5608
BVDA	0.7309	0.7506	0.7408	0.4878	0.4815

Results on testing data indicate that the proposed classifier, BVDEA achieves the best performance in prediction with the *ProbEx* of 0.5814. The nearest performance belongs to the method of LVQ. However it provides an unbalanced prediction between order and disorder residues with high *specificity* but relatively low *sensitivity*. Overall, the results support BVDEA as a successful classifier for predicting disorder and order.

4.3. Comparison with Existing Disorder Prediction Tools

For the second comparison, the performance of specific disorder prediction tools, PONDRs, DisEMBL, GlobPlot, DISOPRED2, FoldIndex, RONN, DisPRO, PreLink and DisPSSMP were investigated. These methods have been presented in literature as publicly available web servers or packages. The server for DisEMBL provides three choices for prediction as mentioned before. The three applications have been given as DisEMBL (hot), DisEMBL (465) and DisEMBL (coils). The results for the methods were collected from the work of Yang et al. and the work of Su et al [Yang et al., 2005] [Su et al., 2006].

The performance of BVDEA on COD159 dataset was compared with the 11 methods in Table 4.14. Similarly, the performance of the methods on blind testing set, D80, were also examined (Table 4.14). In Table 4.14 and Table 4.15, the results were compiled in all mentioned measure types, *sensitivity*, *specificity*, *accuracy*, *matthews' correlation coefficient* and *probability excess*. The success order of the algorithms was given with regard to the favoured measure, probability excess.

Table 4.14. Comparing the performance of BVDEA on testing data COD159 with eleven existing tools.

Methods	***Sens***	***Spec***	***Acc***	***Mcc***	***ProbEx***
DisPSSMP	0.825	0.765	0.795	0.589	0.590
BVDEA	0.796	0.785	0.791	0.586	0.581
RONN	0.675	0.888	0.782	0.580	0.563
FoldIndex	0.722	0.815	0.769	0.540	0.536
DISOPRED2	0.469	0.981	0.725	0.543	0.449
PONDR	0.632	0.782	0.707	0.420	0.414
DisPro	0.383	0.982	0.683	0.467	0.365
DisEMBL(465)	0.348	0.978	0.663	0.430	0.327
PreLink	0.319	0.991	0.655	0.430	0.310
DisEMBL(hot)	0.502	0.749	0.626	0.260	0.251
DisEMBL(coil)	0.719	0.446	0.583	0.170	0.165
GlobPlot	0.308	0.821	0.565	0.151	0.129

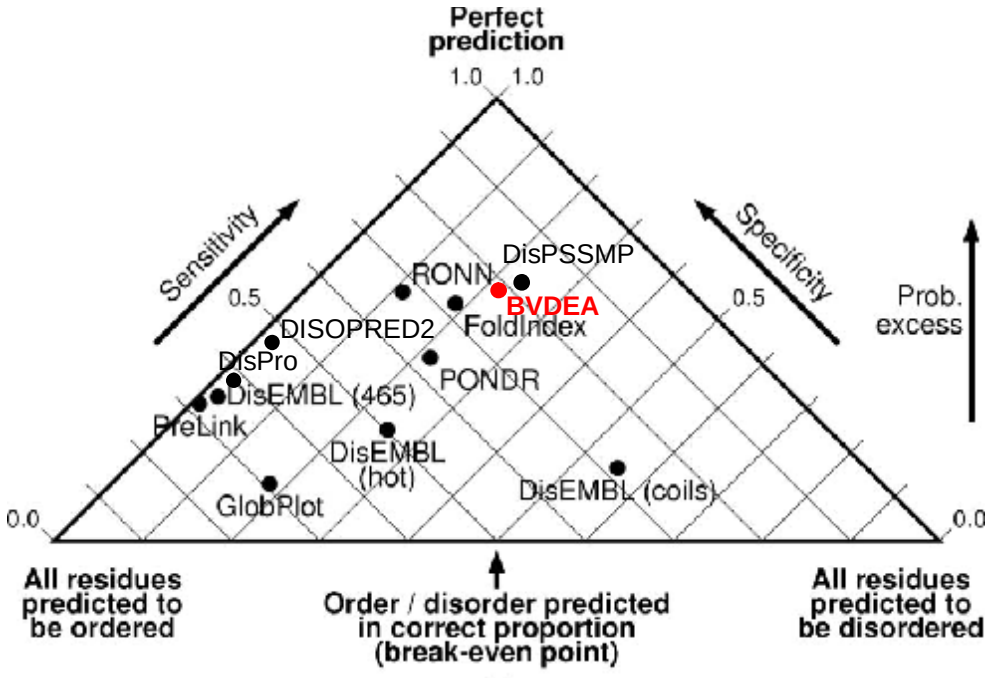


Figure 4.7. The performances of twelve predictors on testing data, COD159.

The prediction performances on testing the balanced set of COD159 were also compared with the given methods via the graph of probability excess given in Figure 4.7. Inspection of the results exhibits that while DisPSSMP performs slightly better than BVDEA, BVDEA gives slightly more balanced prediction compared with DisPSSMP. The four methods, BVDEA, DisPSSMP, RONN and FoldIndex have significantly performed better than from the remaining methods. Most of them are found in the left side of the trapezoid graph. This show the tendency of predicting order i.e. *under-prediction* of disorder.

For obtaining prediction results of BVDEA for blind testing set of D80, the classification was performed by testing the set with the found border vectors under the given conditions of training (Table 4.7, Table 4.8).

Table 4.15. The performance for each subset of BVDEA on blind testing.

Subset 1	**Subset 2**	**Subset 3**	**Subset 4**	**Subset 5**
0.5323	0.5583	0.5718	0.5205	0.5416

Table 4.16. Comparing the performance of BVDEA on testing data D80 with eleven existing tools.

Methods	***Sens***	***Spec***	***Acc***	***Mcc***	***ProbEx***
DisPSSMP	0.767	0.848	0.808	0.463	0.615
BVDEA	0.817	0.728	0.773	0.451	0.545
RONN	0.603	0.878	0.741	0.395	0.481
DisPro	0.418	0.993	0.706	0.578	0.411
DISOPRED2	0.405	0.972	0.689	0.470	0.377
PONDR	0.557	0.816	0.687	0.278	0.373
DisEMBL(hot)	0.492	0.840	0.666	0.260	0.332
DisEMBL(465)	0.334	0.981	0.658	0.437	0.315
FoldIndex	0.488	0.811	0.650	0.224	0.299
PreLink	0.237	0.947	0.592	0.219	0.183
GlobPlot	0.372	0.811	0.592	0.140	0.183
DisEMBL(coil)	0.740	0.424	0.582	0.104	0.165

The performances in terms of probability excess are provided in Table 4.15. The testing performances for each training application were averaged and this gives the overall success results for D80 blind set (Table 4.16). All performances are presented in Figure 4.8. The reproduced graphs of Figure 4.7 and Figure 4.8 were given from Yang et al. in original and used by both Esnouf et al. and Su et al., later [Yang et al., 2005] [Esnouf et al., 2006 / honing] [Su et al., 2006].

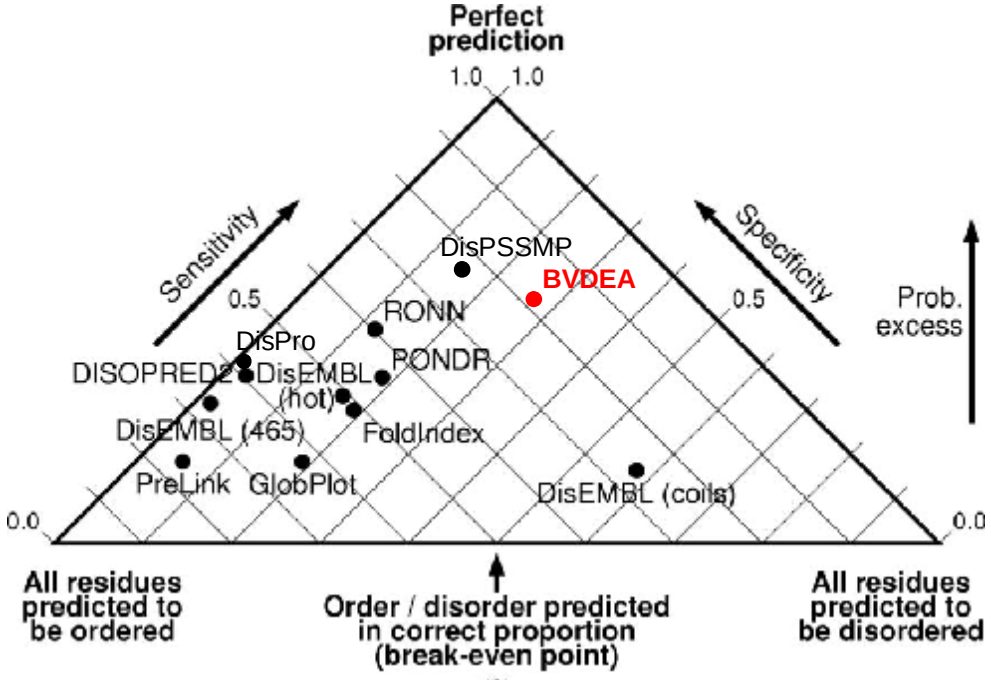


Figure 4.8. The performances of twelve predictors on testing data, D80.

As seen from the results, BVDEA gives rather high accuracy even with the dominance of ordered residues in the blind test set. Indeed, for prediction of blind set, D80, DisPSSMP outperforms all methods. But, this can be caused by the training set of DisPSSMP. It is composed of the disordered proteins which were collected from PDB and involves the disordered regions, as like D80. Nevertheless, BVDEA has the closest results to DisPSSMP among the others. Both are the only methods that have a probability excess value over 0.5. On the other side, BVDEA perform best in predicting disorder with the highest value of *specificity* at 82%. DisEMBL(coils) has a good performance of *sensitivity* but

reveals a significant *over-prediction*. For the other methods, they yield very high scores of *specificity* but they attain this at the expense of missing estimate with a significant number of disordered residues. This means that *under-prediction* occurs.

4.4. Reduction of Dimension

In this stage of the work, the contribution of data reduction to the training success was investigated. Accordingly, the data, COD159 was reduced by the methods Principal Component Analyze (PCA) and Logistic Regression (LR). Concerning the LR results, only 14 attributes are selected. The list of determined attributes by LR is given in Table 4.17. On the other hand, PCA reduced the data to 21 dimensions keeping the 95% variance of the original data.

Table 4.17. List of selected attributes by LR.

Attributes						
X50	X101	X102	X104	X105	X106	X107
X109	X112	X115	X116	X117	X118	X120

During the LR-based (BVDEA_LR) and PCA-based (BVDEA_PCA) applications of BVDEA, predefined parameters and criterions were used for the best comparison. The performances of using PCA-based and LR-based data have been given in Table 4.17 and Table 4.18, respectively.

Table 4.18. The testing probability excesses of the data reduced with PCA.

Subset 1	**Subset 2**	**Subset 3**	**Subset 4**	**Subset 5**
0.3337	0.1691	0.2945	0.1955	0.4351

Table 4.19. The testing probability excesses of the data reduced with LR.

Subset 1	**Subset 2**	**Subset 3**	**Subset 4**	**Subset 5**
0.6694	0.2731	0.3203	0.1106	0.5918

The results given in Table 4.18 and Table 4.19 show that dimension reduction does not provide any improvement in performance for predicting disorder. On the contrary, it causes a significant decrease in prediction success. For this reason the overall comparison was not executed here. Briefly, the reduction does not enhance the learning capability of the proposed method. Here, it can be said that all used features are needed for prediction in spite of the multicollinearity between them.

5. CONCLUSIONS

Studies on structural genomics indicate that numerous protein segments remain unfolded in their native states. Contrary to the structure – function paradigm, these regions that fail to fold into a fixed 3D structure under physiological condition yet exhibit function. Currently, these proteins are generally referred to as “natively unfolded” or “intrinsically disordered”. Upon noticing that many of these proteins play key role in vital functions and also in some diseases, identification of the disordered regions has became a demanding process for structure prediction and functional characterization of proteins. Therefore, many studies have been motivated on accurate prediction of disorder.

Since amino acid sequence attributes have been widely used for determining protein structure, it has been theorized that the sequence would also determine disorder, as well. Thus, alternative to experimental methods, several computational methods have been suggested for disorder prediction based on the amino acid sequence. Recently, several specific disorder prediction tools have been developed include PONDRs, DisEMBL, GlobPlot, DISOPRED2, FoldIndex, RONN, DisPRO, PreLink, DisPSSMP.

In the study, a novel method was developed for predicting disorder as an alternative accurate classifier. For attesting the performance of the method, three computational learning techniques and eleven specific tools were used for comparison. In order to perform the predictions, first the collected data was prepared by utilizing a variety of structural features for proteins. Then, training was executed based on the data by 5-fold cross validation.

When compared with the three learning methods of GRNN, LVQ and BVDA, the proposed method BDVEA gives the best accuracy on classification. Here, BDVEA provides more fast and robust learning as compared to the others. The training time is almost the same with LVQ (~3 minutes), the half of BVDA and the quarter of GRNN. Despite, the training in GRNN is only the loading of the train data.

Furthermore, comparison with the other existing disorder predictors

demonstrates that the method achieves quite good prediction performance in finding disordered regions of proteins. Except for DisPSSMP, BVDEA outperforms all other ten methods, significantly, without either under-predicting or over-predicting the disordered regions. This is confirmed with both main and blind testing successes. Nevertheless, for testing the main set, BVDEA yields roughly equal performance with DisPSSMP and more balanced prediction accuracy. Besides, it reveals a highest score of sensitivity on prediction of blind test set within all methods, including DisPSSMP. As evident from the results, BVDEA can be suggested as an influential method to achieve accurate predictions of disordered regions.

As a conclusion, it has been shown that the new method provides a significant contribution on predicting disorder and order regions of proteins. Thus, accurate prediction of disorder that is a demanding problem due to its importance on structural and functional identification of protein was significantly achieved. This also provides insights for projects on drug discovery and gene finding.

REFERENCES

AAINDEX. Laboratory of Genome Database, Japan, http://www.genome.jp/aaindex/

ALTSCHUL, S.F., MADDEN, T.L., SCHIFFER, A.A., ZHANG, J., ZHANG, Z., MILLER, W., LIPMAN, D.J., 1997. Gapped BLAST and PSI - BLAST: a new generation of protein database search programs. Nucleic Acids Res., 25, 3389-3402.

ANFINSEN, C. B., 1973. Principles that Govern the Folding of Protein Chains. Science, 181, 223–230.

BALDI, P., BRUNAK, S., 2001. Bioinformatics: The Machine Learning Approach, Second Edition. The MIT Press, Cambridge, London, England, 114-118.

BISHOP, C.M., 1995. Neural Networks for Pattern Recognition. Oxford University Press Inc., New York, 482.

BOUTSELAKIS, H., DIMITROPOULOS, D., FILLON, J., GOLOVIN A., HENRICK, K., HUSSAIN, A., IONIDES, J., JOHN, M., KELLER, P. A., KRISSINEL, E., MCNEIL, P., NAIM, A., NEWMAN, R., OLDFIELD, T., PINEDA, J., RACHEDI, A., COPELAND, J., SITNOV, A., SOBHANY, S., SUAREZ-URUENA, A., SWAMINATHAN, J., TAGARI, M., TATE, J., TROMM, S., VELANKAR, S., VRANKEN, W., 2003. E-Msd: The European Bioinformatics Institute Macromolecular Structure Database. Nucleic Acids Res., 31, 458-462.

CAI, Y., LIU, X., XU, X., CHOU, K., 2002. Artificial Neural Network for Predicting Protein Secondary Structure Content. Computers and Chemistry, 26, 347-350.

CHENG, J., SWEREDOSKI, M.J, BALDI, P., 2005. Accurate Prediction of Protein Disordered Regions By Mining Protein Structure Data. Data Min. Knowl. Dis., 11, 213–222.

COEYTAUX, K. and POUPON, A., 2005. Prediction of Unfolded Segments in a Protein Sequence Based On Amino Acid Composition. Bioinformatics, 21(9), 1891–1900.

CHOU, P.Y. and FASMAN, G.D., 1978. Empirical Predictions of Protein Conformation. Annual Review Biochemistry, 7, 251-276.

DENNISTON, K.J., TOPPING, J.J., CARET, R.L., 2001. General, Organic and Biochemistry. Third International Edition, McGraw-Hill Companies Inc., New York.

DOSZTÁNYI, Z., CSIZMÓK, V., TOMPA, P., SIMON, I., 2005. The Pairwise Energy Content Estimated from Amino Acid Composition Discriminates Between Folded and Intrinsically Unstructured Proteins. J. Mol. Biol., 347, 827–839.

DUNKER, A.K., OBRADOVIC, Z., ROMERO, P., KISSINGER, C., VILLAFRANCA, E., 1997. On the Importance of Being Disordered. PDB Newsletter, 81, 3-5.

DUNKER, A.K., ROMERO, P., OBRADOVIC, Z., GARNER, E.C., BROWN, C.J, 2000. Intrinsic Protein Disorder in Complete Genomes. Genome Informatics, 11, 161-171.

DUNKER, A. K., LAWSON, J.D., BROWN, C. J., WILLIAMS, R. M., ROMERO, P., OH, J. S., OLDFIELD, C. S., CAMPEN, A. M., RATLIFF, C. M., HIPPS, K. W., AUSIO, J., NISSEN, M. S., REEVES, R., KANG, C., KISSINGER, C. R., BAILEY, R. W., GRISWOLD, M. D., CHIU, W., GARNER, E. C., OBRADOVIC, Z., 2001a. Intrinsically Disordered Protein. Journal of Molecular Graphics and Modeling, 19, 26-59.

DUNKER, A.K., and OBRADOVIC, Z., 2001b. The Protein Trinity – Linking Function and Disorder. Nature Biotechnology, 19, 805-806.

ERSÖZ, İ., İBRIKÇI, T., ÇAKMAK, A., ERSOY, O. K., 2004. Secondary Structure Prediction of Hemoglobin by Neural Networks. ANNIE 2004, St. Louis, Missouri, USA, 2004.

FISCHER, E., 1894. Einfluss Der Configuration Auf Die Wirkung Der Enzyme. Ber. Dt. Chem. Ges. 27, 2985–2993.

GALZITSKAYA, O.V, GARBUZYNSKIY, S.O., LOBANOV, M.Y., 2006. Prediction of Amyloidogenic and Disordered Regions in Protein Chains. PLoS Bioinf., 2(12), 1639-1648.

GARBUZYNSKIY, S.O., LOBANOV, M.Y., GALZITSKAYA, O.V., 2004. To Be Folded or To Be Unfolded? Protein Sci., 13, 2871-2877.

GARNER, E., CANNON, P., ROMERO, P., OBRADOVIC, Z., DUNKER, A.K., 1998. Predicting Disordered Regions From Amino Acid Sequence: Common Themes Despite Differing Structural Characterization. Genome Informatics, 9, 201–213.

GRANTHAM, R., 1974. Polarity Scale. Science, 185, 862-864.

HAYKIN, S., 1994. Neural Networks. Prentice-Hall Inc., New Jersey, 363-369.

HU, Y., ZHANG, H., WIELICKI, B., STACKHOUSE, P., 2002. A Neural Network MODIS-CERES Narrowband to Broadband Conversion, IGARSS 2002 Conference Proceedings.

HOBOHM, U., AND SANDER, C., 1994. Enlarged representative set of protein structures. Protein Sci., 3, 522-524.

JONES, D.T., 1999. Protein Secondary Structure Prediction Based on Position-specific Scoring Matrices. J. Mol. Biol., 292, 195-202.

JONES, D. T. and WARD, J. J., 2003. Prediction of Disordered Regions in Proteins from Position Specific Scoring Matrices. Proteins, 53, 573-578.

JORDAN, M.I. and BISHOP, C.M., 1996. Neural Networks. CRC Handbook of Computer Science.

KASAPOĞLU, N.G., ERSOY, O.K., YAZGAN, B., 2006. Border Feature Detection and Adaptation for Classification of Remote Sensing Images. IEEE Geoscience and Remote Sensing Symposium, Denver, USA.

KAWASHIMA, S., OGATA, H., KANEHISA, M., 1999. AAIndex: Amino Acid Index Database. Nucleic Acids Res., 27, 368-369.

KOEHL, P. and LEVITT, M., 1999. Structure-based conformational preferences of amino acids. J Proc Natl Acad Sci U S A, 96, 12524-12529.

KOHONEN, T., 1989. Self-Organization and Associative Memory. 3rd Ed., Springer-Verlag, Berlin.

KOSHLAND, D. E., 1958. Application of a Theory of Enzyme Specificity to Protein Synthesis. Proc. Natl. Acad. Sci. U.S.A. 44, 98–104.

KYTE, J., and DOOLITTLE, R., 1982. A Simple Method for Displaying The Hydropathic Character of A Protein. J. Mol. Biol., 157, 105-132.

LI, X., ROMERO, P., RANI, M., DUNKER, A.K., OBRADOVIC, Z., 1999.

Predicting Protein Disorder for N-, C- and Internal Regions. Genome Inform Ser. Workshop Genome Infor., 10, 30-40.

LI, X., BROWN, C.J, OBRADOVIC, Z., GARNER, E.C., DUNKER, A.K., 2000. Comparing Predictors of Disordered Protein. Genome Informatics, 11, 172-184.

LINDING, R., JENSEN, L.J., DIELLA, F., BORK, P., GIBSON, T.J., RUSSELL, R.B., 2003a. Protein Disorder Prediction: Implications for Structural Proteomics. Structure, 11(11), 1316-1317.

LINDING, R., RUSSELL, R.B., NEDUVA, V., GIBSON, T.J., 2003b. Globplot: Exploring Protein Sequences for Globularity And Disorder. Nucleic Acids Research, 31(13), 3701–3708.

LIPPMANN, R.P., 1987. An Introduction to Computing with Neural Nets. Institute of Electrical and Electronics Engineers ASSP Magazine, 4-22.

LISE, S. and JONES, D. T., 2005. Sequence Patterns Associated with Disordered Regions in Proteins. Proteins, 58, 144-150.

MELAMUD, E., MOULT, J., 2003. Evaluation of Disorder Predictions in CASP5. Proteins, 53, 561-565.

MIRSKY A.E. and PAULING L., 1936. On the Structure of Native, Denatured, and Coagulated Proteins. Proc. Natl. Acad. Sci. U.S.A. 22(7), 439–447.

OBRADOVIC, Z., PENG, K., VUCETIC, S., RADIVOJAC, P., BROWN, C.J, DUNKER, A.K., 2003. Predicting Intrinsic Disorder from Amino Acid Sequence. Proteins: Structure, Function, and Genetics, 53, 566-572.

PENG, K., VUCETIC, S., RADIVOJAC, P., BROWN, C. J., DUNKER, A. K., OBRADOVIC, Z., 2005. Optimizing Long Intrinsic Disorder Predictors with Protein Evolutionary Information. Journal of Bioinformatics and Computational Biology, 3(1), 35-60.

PENG, K., RADIVOJAC, P., VUCETIC, S., DUNKER A.K, OBRADOVIC, Z., 2006. Length-Dependent Prediction of Protein Intrinsic Disorder. BMC Bioinformatics, 7 (1), 208.

PRILUSKY, J., FELDER, C.E., ZEEV-BEN-MORDEHAI, T., RYDBERG, E.H., MAN , O., BECKMANN, J.S., SILMAN, I., SUSSMAN, J.L., 2005.

FoldIndex: a simple tool to predict whether a given protein sequence is intrinsically unfolded. Bioinformatics, 21(16), 3435-3438.

PROTEIN DATA BANK, Chemistry Department, Brookhaven National Laboratory. Upton, NY 11973 USA, www.rcsb.org/pdb/

QIAN, N., SEJNOWSKI, T.J., 1988. Predicting the Secondary Structure of Globular Proteins Using Neural Network Models. J. Mol. Biol., 202, 865-884.

RADIVOJAC, P., OBRADOVIC, Z., SMITH, D. K., ZHU, G., VUCETIC, S., BROWN C. J., LAWSON, J. D., DUNKER, A. K., 2004. Protein Flexibility and Intrinsic Disorder. Protein Science, 13(1), 71-80.

RIIS, S.K. and KROGH, A., 1996. Improving Prediction of Protein Secondary Structure Using Structured Neural Networks and Multiple Sequence Alignments. J. Comp. Biol., 1(3), 163-183.

ROMERO, P., OBRADOVIC, Z., KISSINGER, C., VILLAFRANCA, E., DUNKER, A.K., 1997. Identifying Disordered Regions in Proteins from Amino Acid Sequence. In Proc. IEEE Int. Conf. On Neural Networks, 90-95.

ROMERO, P., OBRADOVIC, Z., KISSINGER, C.R., VILLAFRANCA, J.E., GUILLIOT, S., GARNER, E., DUNKER, A.K., 1998. Thousands of Proteins Likely To Have Long Disordered Regions. Pacific Symp. Biocomputing, 1998, 3, 437-448.

ROMERO, P., OBRADOVIC, Z., DUNKER, A.K., 2000. Intelligent Data Analysis for Protein Disorder Prediction. Artificial Intelligence Review, 14, 447-484.

ROMERO, P., OBRADOVIC, Z., LI, X., GARNER, E.C., BROWN, C.J, DUNKER, A.K., 2001. Sequence Complexity of Disordered Protein. Proteins: Structure, Function and Genetics, 42, 38-48.

ROST, B., 1996. PHD: Predicting One-Dimensional Protein Structure by Profile-Based Neural Networks. Methods Enzymol., 266, 525-539.

ROST, B., 1998. Protein Structure Prediction in 1D, 2D, 3D. Chichester: Wiley, 2242- 2255.

ROST, B., SANDER, C., 1993. Prediction of Protein Secondary Structure at Better Than 70% Accuracy. J. Mol. Biol., 232, 584-599.

SARLE, W.S., 1994. Neural Networks and Statistical Models. Proceedings of the

Nineteenth Annual SAS Users Group International Conference.

SCHLESSINGER, A. and ROST, B., 2005. Protein Flexibility and Rigidity Predicted From Sequence. Proteins: Structure, Function, and Bioinformatics, 61, 115–126.

SHIMIZU, K., HIROSE, S., NOGUCHI, T., MURAOKA, Y., 2004. Predicting The Protein Disordered Region Using Modified Position Specific Scoring Matrix. The 15th International Conference on Genome Informatics, 150.

SHIMIZU, K., MURAOKA, Y., HIROSE, S., TOMII, K., NOGUCHI, T., 2007. Predicting Mostly Disordered Proteins by Using Structure-Unknown Protein Data. BMC Bioinformatics, 8, 78.

SMITH, D.K., RADIVOJAC, P., OBRADOVIC, Z., DUNKER, A.K., ZHU, G., 2003. Improved Amino Acid Flexibility Parameters. Protein Sci., 12, 1060-1072.

SOLOMONS, T.W.G., 1995. Organic Chemistry. Fifth Edition. John Wiley & Sons Inc, New York.

SPECHT, D., 1991. A General Regression Neural Network. IEEE Trans. Neural Networks, 2(6), 568-576.

STORMO, G. D., SCHNEIDER, T. D., GOLD, L., EHRENFEUCHT, A., 1982. Use of the `Perceptron' Algorithm to Distinguish Translational Initiation Sites In E. Coli. Nucl. Acids Res., 10, 2997-3011.

SU, C., CHEN, C., OU, Y., 2006. Protein Disorder Prediction by Condensed PSSM Considering Propensity for Order or Disorder. BMC Bioinformatics, 7 ,319.

TABACHNICK, G.B., 2001. Using Multivariate Statistics (Fourth Edition). Allyn and Bacon, USA, 517.

TOMPA, P., 2002. Intrinsically Unstructured Proteins. Trends in Biochemical Sciences, 27(10), 527-533.

UVERSKY V. N., GILLESPIE J. R., FINK A. L., 2000. Why are “natively unfolded” proteins unstructured under physiologic conditions? Proteins, 41, 415-427.

UVERSKY V.N., OLDFIELD, C.J., DUNKER, A.K., 2005. Showing Your ID: Intrinsic Disorder as an ID for Recognition, Regulation and Cell Signaling.

Review, Journal of Molecular Recognit., 18, 343-384.

UVERSKY V.N., 2002. Natively Unfolded Proteins: A Point Where Biology Waits For Physics. Review. Protein Sci., 11 (4), 739-756.

WARD, J.J., MCGUFFIN, L.J., BUXTON, B.F., JONES, D.T., 2003. Secondary structure prediction with support vector machines. Bioinformatics, 19(13), 1650-1655.

WARD, J. J., SODHI, J. S., MCGUFFIN, L. J., BUXTON, B. F., JONES, D. T., 2004a. Prediction and Functional Analysis of Native Disorder in Proteins from the Three Kingdoms of Life. J. Mol. Biol., 337, 635–645.

WARD, J.J., MCGUFFIN, L. J., BRYSON, K., BUXTON, B. F., JONES, D. T., 2004b. The DISOPRED Server for the Prediction of Protein Disorder. Bioinformatics, 20, 2138–2139.

WEATHERS E.A., PAULAITIS, M.E., WOOLF, T.B., HOH, J.H, 2004. Reduced Amino Acid Alphabet Is Sufficient To Accurately Recognize Intrinsically Disordered Protein. FEBS Letters, 576, 348–352.

WILLIAMS, R. M., OBRADOVIC, Z., MATHURA, V., BRAUN, W., GARNER, E. C., YOUNG, J., TAKAYAMA, S., BROWN, C. J., DUNKER, A. K., 2001. The Protein Non-Folding Problem: Amino Acid Determinants of Intrinsic Order and Disorder. Pacific Symposium on Biocomputing, 89-100.

WOOTTON, J.C. and FEDERHEN, S., 1993. Statistic of local complexity in amino acid sequences and sequence databases. Comput. Chem., 17, 149-163.

WRIGHT, P.E. and DYSON, H.J., 1999. Intrinsically Unstructured Proteins: Re-assessing the Protein Structure-Function Paradigm. J. Mol. Biol., 293, 321-331.

WU, C.H. and McLARTY, J.W., 2000. Neural Networks and Genome Informatics. Elseiver, USA, 8-11, 69-76, 97-99, 116-119.

VIHINEN, M., TORKKILA, E., RIIKONEN, P., 1994. The Pedant Genome Database. Proteins, 19, 141-149.

VUCETIC, S., RADIVOJAC, P., DUNKER, A.K., BROWN, C.J. OBRADOVIC, Z., 2001. Methods for Improving Protein Disorder Prediction. IEEE/INNS International Joint Conference on Neural Networks, 4, 2718-2723.

VUCETIC, S., BROWN, C., DUNKER, A.K, OBRADOVIC, Z., 2003. Flavors of Protein Disorder. Proteins: Structure, Function and Genetics, 52, 573-584.

XIE, Q., ARNOLD, G. E., ROMERO, P., OBRADOVIC, Z., GARNER, E., DUNKER, A. K., 1998. The Sequence Attribute Method For Determining Relationships Between Sequence and Protein Disorder. Genome Informatics, 9, 193-200.

YANG, R.Z., THOMSON, R., MCNEIL, P., ESNOUF, R.M., 2005. RONN: The Bio-basis Function Neural Network Technique Applied to the Detection of Natively Disordered Regions in Proteins. Bioinformatics, 21, 3369-3376.

YEUNG, D., CHOW, C., 2002. Parzen Window Network Intrusion Detectors. Proceedings of the Sixteenth International Conference on Pattern Recognition. Quebec City, Canada, 4, 385-388.

ZIMMERMAN, J.M., ELIEZER, N. AND SIMHA, R., 1968. The characterization of amino acid sequences in proteins by statistical methods. J J. Theor. Biol., 21, 170-201.

BIOGRAPHY

She was born on Nov. 19th. 1975 in Ankara, TÜRKİYE. After completing her primary (1982-1987), secondary (1987-1990) and high school (1990-1993) education at TED Ankara College, she received her B.S. degree from Hacettepe University in the field of Physics Engineering.

For a while, she worked for T.E.D. Ankara College as a Science Education Project Assistant. Afterwards, she was employed as a Preparing and Controlling Data Operator at the Student's Affairs Office of the Çukurova University for three years. Meanwhile, she got her M.S. degree from Electrical and Electronics Engineering Department of Çukurova University (2001-2003). Since 2003, she has been working as a Lecturer in Computer and Electronics Education Department of Tarsus Technical Education Faculty in Mersin University.

APPENDIX A

The List of Used Property-Based and Composition-Based (Groups) Features

X(1)	Bulkiness [Zimmerman et al., 1968: from AAINDEX]
X(2)	Amino acid composition [Dayhoff et al., 1978a: from AAINDEX]
X(3)	Amino acid distribution [Jukes et al., 1975: from AAINDEX]
X(4)	Isoelectric point [Zimmerman et al., 1968: from AAINDEX]
X(5)	Localized electrical effect [Fauchere et al., 1988: from AAINDEX]
X(6)	Solvation free energy [Eisenberg-McLachlan, 1986: from AAINDEX]
X(7)	Hydration free energy [Robson-Osguthorpe, 1979: from AAINDEX]
X(8)	Absolute entropy [Hutchens, 1970: from AAINDEX]
X(9)	Polarizability parameter [Charton-Charton, 1982: from AAINDEX]
X(10)	Polarity [Grantham, 1974: from AAINDEX]
X(11)	Polarity [Zimmerman et al., 1968: from AAINDEX]
X(12)	Steric parameter [Charton, 1981: from AAINDEX]
X(13)	Side chain volume [Krigbaum-Komoriya, 1979: from AAINDEX]
X(14)	Residue volume [Bigelow, 1967: from AAINDEX]
X(15)	Residue volume [Goldsack-Chalifoux, 1973: from AAINDEX]
X(16)	Volume [Grantham, 1974: from AAINDEX]
X(17)	Normalized van der Waals volume [Fauchere et al., 1988: from AAINDEX]
X(18)	Average volume of buried residue [Chothia, 1975: from AAINDEX]
X(19)	Mass [Wu and McLarty, 2000]
X(20)	Average accessible surface area [Janin et al., 1978: from AAINDEX]
X(21)	Surface Area [Wu and McLarty, 2000]
X(22)	Propensity to be buried inside [Wertz-Scheraga, 1978: from AAINDEX]
X(23)	Alpha-helix propensity derived from designed sequences [Koehl-Levitt, 1999: from AAINDEX]
X(24)	Beta-sheet propensity derived from designed sequences [Koehl-Levitt, 1999: from AAINDEX]
X(25)	Alpha Helix Popensity [Wu and McLarty, 2000]
X(26)	Beta Strand Propensity [Wu and McLarty, 2000]
X(27)	Turn Propensity [Wu and McLarty, 2000]
X(28)	Alpha-helix indices [Geisow-Roberts, 1980: from AAINDEX]
X(29)	Beta-strand indices [Geisow-Roberts, 1980: from AAINDEX]
X(30)	Normalized frequency of alpha-helix [Chou-Fasman, 1978b: from AAINDEX]
X(31)	Normalized frequency of beta-sheet [Chou-Fasman, 1978b: from AAINDEX]
X(32)	Normalized frequency of beta-turn [Chou-Fasman, 1978b: from AAINDEX]
X(33)	Hydrophilicity scale [Kuhn et al., 1995: from AAINDEX]
X(34)	Average flexibility indices [Bhaskaran-Ponnuswamy, 1988: from AAINDEX]
X(35)	Normalized flexibility parameters (B-values), average [Vihinen et al., 1994: from AAINDEX]

X(36)	Location Parameters of the fit of the B Factors [Smith et al., 2003]
X(37)	Hydrophobicity index [Engelman et al., 1986: from AAINDEX]
X(38)	Hydrophobicity [Prabhakaran, 1990: from AAINDEX]
X(39)	Hydrophobicity scales [Ponnuswamy, 1993: from AAINDEX]
X(40)	Consensus normalized hydrophobicity scale [Eisenberg, 1984: from AAINDEX]
X(41)	Hydrophobicity [Zimmerman et al., 1968: from AAINDEX]
X(42)	Hydrophobicity [Jones, 1975: from AAINDEX]
X(43)	Hydrophobicity [Prabhakaran, 1990: from AAINDEX]
X(44)	Hydropathy index [Kyte-Doolittle, 1982: from AAINDEX]
X(45)	Electron-ion interaction potential [Veljkovic et al., 1985: from AAINDEX]
X(46)	Refractivity [McMeekin et al., 1964: from AAINDEX]
X(47)	Nunber of Contacts [Garbuzynskiy et al., 2004]
X(48)	Russell/Linding Propensity [Linding et al., 2003]
X(49)	Deleage/Roux Propensity [Linding et al., 2003]
. . .	. . .
X(71)	NetCharge (K+R-D-E) [Xie et al., 1998]
X(72)	Disorder Promoting (R+K+E+P+S) [Romero et al., 2001]
X(73)	Order Promoting (C+W+Y+I+V) [Romero et al., 2001]
X(74)	Flexibility (G+T+R+S+N+Q+D+P+E+K) [Smith et al., 2003]
X(75)	Rigidity (W+Y+F+C+I+V+H+L+M+A) [Smith et al., 2003]
X(76)	Beta Strand Related (E+B) [Schlessinger and Rost, 2005]
X(77)	Alpha Helix Related (H+G+I) [Schlessinger and Rost, 2005]
X(78)	Best Helix Formers (E+M+A+L) [Xie et al., 1998]
X(79)	Best Helix Breakers (Y+N+P+G) [Xie et al., 1998]
X(80)	Best Sheet Formers (V+I+Y+F+W) [Xie et al., 1998]
X(81)	Best Sheet Breakers (S+G+K+P+D+E) [Xie et al., 1998]
X(82)	Ambivalent (A+C+G+P+S+T+W+Y) [Baldi and Brunak, 2001]
X(83)	External (R+N+D+Q+E+H+K) [Baldi and Brunak, 2001]
X(84)	Internal (I+L+M+F+V) [Baldi and Brunak, 2001]
X(85)	Acidic (D+E) [Baldi and Brunak, 2001]
X(86)	Basic (R+H+K) [Baldi and Brunak, 2001]
X(87)	Amide (N+Q) [Baldi and Brunak, 2001]
X(88)	Aliphatic (A+G+I+L+V) [Baldi and Brunak, 2001]
X(89)	Aromatic (F+W+Y) [Baldi and Brunak, 2001]
X(90)	Hydroxyl (S+T) [Baldi and Brunak, 2001]
X(91)	Imino (P) [Baldi and Brunak, 2001]
X(92)	Sulfur (C+M) [Baldi and Brunak, 2001]
X(93)	Small (A+C+D+G+N+P+S+T+V) [Su et al., 2006]

X(94)	Tiny (C+A+G+S) [Su et al., 2006]
X(95)	Most Hydrophilic (D+E+N+Q+R+K) [Wu and McLarty, 2000]
X(96)	Most Hydrophobic (A+M+I+L+V+F+W) [Wu and McLarty, 2000]
X(97)	Hydrophobic (A+I+L+M+F+P+W+V+G) [Baldi and Brunak, 2001]
X(98)	Helix Propensity (A+E+Q+H+K+M+L+R) [Wu and McLarty, 2000]
X(99)	Sheet Propensity (C+T+I+V+F+Y+W) [Wu and McLarty, 2000]
X(100)	Coil Propensity (S+G+P+D+N) [Wu and McLarty, 2000]

APPENDIX B

The success results of BVDEA, BVDA, LVQ and GRNN for each subset

Table B.1. The probability excesses of validation set for each subset at given *h-t* pairs in BDVEA.

h-t	**Subset 1**	**Subset 2**	**Subset 3**	**Subset 4**	**Subset 5**
0.5-9500	0.0671	0.2074	0.3787	0.3385	0.2949
0.5-13500	0.1517	0.3364	0.4747	0.1032	0.2250
0.9-13500	0.4211	0.2477	0.1280	0.0310	0.2745

Table B.2. The maximum probability excesses of testing set for each subset at given *h-t* pairs in BDVA.

h-t	**Subset 1**	**Subset 2**	**Subset 3**	**Subset 4**	**Subset 5**
0.1-1000	0.3812	0.1964	0.2987	0.3222	0.3897
0.2-6750	0.6209	0.4160	0.5170	0.4909	0.4701
0.5-9500	0.7438	0.4118	0.5901	0.5929	0.6155
0.5-13500	0.7459	0.4286	0.5644	0.5811	0.6093
0.9-13500	0.7500	0.4685	0.5901	0.5714	0.5753

Table B.3. The stopping times of training for each subset in terms of record number in BDVA.

Subset 1	**Subset 2**	**Subset 3**	**Subset 4**	**Subset 5**
694	545	469	499	498

Table B.4. The probability excesses of validation set for each subset at given n_c in LVQ.

n_c	**Subset 1**	**Subset 2**	**Subset 3**	**Subset 4**	**Subset 5**
10	0.3529	0.3003	0.3540	0.3416	0.3292
50	0.2508	0.2766	0.3499	0.3199	0.3241
10 ($h = 0.05$)	0.2136	0.1558	0.3230	0.3220	0.3148

Table B.5. The probability excesses of validation set for each subset at given σ in GRNN.

s	**Subset 1**	**Subset 2**	**Subset 3**	**Subset 4**	**Subset 5**
0.3	0.3447	0.3088	0.3787	0.3602	0.3416
0.7	0.4417	0.4037	0.4582	0.4778	0.3963

APPENDIX C

The graphs of subset 1, subset 3 and subset 5 in BVDA

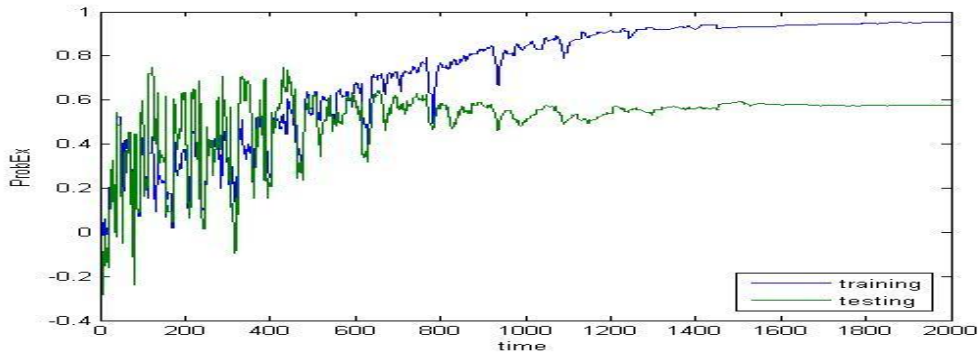


Figure C.1. Prediction results of both testing and training sets at $h-t=0.9-13500$ for subset 1.

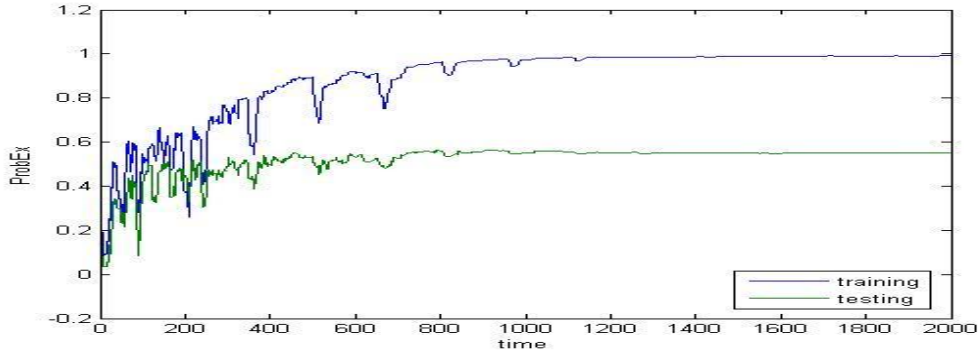


Figure C.2. Prediction results of both testing and training sets at $h-t=0.5-13500$ for subset 3.

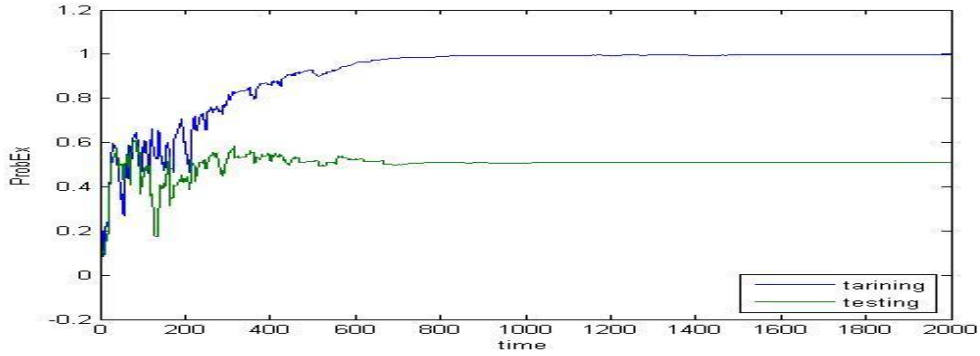


Figure C.3. Prediction results of both testing and training sets at $h-t=0.5-9500$ for subset 5.