PROTEIN STRUCTURE PREDICTION:

Feature Analysis and Identification of Membrane Proteins

A thesis submitted in partial fulfillment of the requirements

For the award of the degree of

Master of Technology

In

Computational & Systems Biology

Submitted by:

Ashutosh Kumar Pandey

E. No: 11/75/MT/001

Under the supervision of

Dr Lovekesh Vig

Center for Computational Biology & Bioinformatics
School of Computational & Integrative Sciences
Jawaharlal Nehru University
New Delhi - 110067
2013

जवाहरलाल नेहरू विश्वविद्यालय
Jawaharlal Nehru University,
संगणकीय एवं समेकित विज्ञान संस्थान
School of Computational and Integrative Sciences
नई दिल्ली-110067 New Delhi-110067 (INDIA)

Hall No-6, Lecture Hall Complex, JNU
Tel. . (Direct) 26741517 (Off.) : 26704171
Email : dean_scis@mail.jnu.ac.in

Certificate

This is to certify that the thesis entitled "**PROTEIN STRUCTURE PREDICTION: Feature Analysis and Identification of Membrane Proteins**" is an authentic record of the dissertation carried out by **Ashutosh Kumar Pandey** at **School of Computational and Integrative Sciences** under my guidance and supervision. The contents of this project work, in full or parts have not been submitted elsewhere for award of any other degree or diploma. This work fulfills the requirement for the award of the "**Master of Technology (M.Tech) degree in Computational and Systems Biology**"

Dated: 26·7·13

26/7/13

Dr. Lovekesh Vig
Supervisor
Assistant Professor
SCIS, JNU.

Professor Karmeshu
Dean
SCIS, JNU.

Acknowledgement

I take this opportunity to express my gratitude and sincere thanks to my supervisor Dr Lovekesh Vig for his constant support, guidance and encouragement that he provided throughout this project. His most revered help came in form of encouragement and bringing to me newer and newer concepts in the field of Machine Learning apart from guiding towards the completion of the thesis.

My sincere thanks to our Dean Prof Karmeshu for providing research facilities and I would also like to thank Dr Devapriya Choudhary (SBT,JNU) for his critical guidance regarding the database for proteins and the most innovative talks with him in department compound, under the tree!

I gratefully acknowledge Dr N Subbarao for his support and guidance in the analysis of protein binding sites problem and to my teachers Dr A Krishnamachari, Dr Pradipta Bandyopadhaya, Dr Andrew Lyn and Dr Narinder Singh Sahni for their valuable guidance during the course.

I also offer my thanks to DBT, JNU for providing financial assistance and resources for the work and to my parents, classmates and JNU staff for every support they have provided to me.

I offer my respectful obeisance to my counselor HG Sarvapriya Prabhu and unto the Lotus feet of my master Lord Krsna, who is the cause of all causes and without His will no inspiration ever ignites!

Abstract

With increasing database of identified protein structures in **PDB (Protein Data Bank)** the need to classify them into proper class becomes a task of utmost importance. Classification of protein structures has been a primary concern for structural biologist because such classification is important to establish many properties of the protein. It helps to identify class, evolutionary relationship, conserved sites, functional sites and interacting domains (binding domain). If this information is available then it can be used in many important tasks such as creating **protein interaction networks**, understanding drug discovery pathways and structural basis of protein misfolded diseases etc. The primary concern of this work is to provide a platform to classify membrane proteins whose PDB structure is known from any given large database of proteins. Initially we have considered membrane proteins for this task as reliable datasets are available. The classification is then followed by prediction of several other features that are specific to membrane proteins. This work provides possibility of data regeneration and establishing higher relationships among proteins which can be used to develop interaction network. There are many methods to classify the proteins but here we are focusing upon machine learning techniques that specifically exploit the **Neural Network** and **Naïve Bayes classifier** to classify the membrane proteins. In this work we have also considered the fact which is quite common among biological data that the available concerned class data is sometimes small and the random protein structure data is large. So from a pool of large protein database identifying the concerned data is important as such cases have been seen to make classifiers biased. We use the **Semi Supervised Algorithm** to solve this problem.

Keywords: PDB, Protein Interaction Network, Neural Network, Naïve Bayes Classifier, Semi Supervised Algorithm

Table of Contents

1 Contents

List of Tables

List of Figures

Abbreviations

1. ML – Machine Learning
2. ANN – Artificial Neural Network
3. PU – Positive Unlabeled
4. SSL – Semi Supervised Learning
5. RN – Reliable Negative
6. PDB – Protein Data Bank
7. REST – Representational State Transfer
8. SOAP – Simple Object Access Protocol
9. RegEx – Regular Expression
10. ROC –Receiver Operator Characteristics
11. MPTopo – Membrane Protein Topology (Server)
12. TMPred – TransMembrane Prediction
13. SMART- Simple Modular Architecture Research Tool
14. CD-HIT – Cluster Database at high Identity Tolerance
15. GPCR – G-Protein Coupled Receptors

1 INTRODUCTION

1.1 Protein Class Identification: The Idea & Importance

The problem of protein class identification is not new and has been one of the most sought areas of work by many scholars. The recent improvement in structural determination techniques have allowed newer and newer protein structures to be identified with database of known proteins doubling every 2 years [1]. Another important aspect is that, how protein sequences as products of translation are folded into their native structure is still unknown and a mystery yet to be solved. Our point of concern is to classify the proteins once its sequence and PDB structure is known. Such classification and categorization is important in several aspects [2] such as

a) Maintaining a standard database of proteins through the FAMILIES to which it belongs
b) Locating conserved DOMAINS and SEQUENCE FEATURES
c) Establishing evolutionary relationship among proteins
d) Predicting functional and other properties based on its classification results
e) Development of protein interaction networks

In this work we will consider membrane proteins and try to find out a possible method of classification better than the most available ones.

1.1.1 Protein Types and Properties [3]

Proteins based on functional role can be classified into three major classes

a) Fibrous proteins: Polypeptide chains arranged in long strands or sheets. Involved in structural function and have simple tertiary structures.
b) Globular Proteins: Polypeptide chains are folded into globular or spherical shape. Involved in many metabolic functions. Have quite complex structure and so the range of functions they perform
c) Membrane Proteins: A third class can be attributed to them as they don't have either of the above two criteria in complete. These proteins are involved in special tasks across the cell membrane right from transport phenomenon to signal transduction.

1.1.2 Membrane Proteins

Biological membranes are explained with fluid mosaic model. This model describes biological membranes as lipid bilayer and membrane proteins are embedded to it either inside or more outside or spanning across the membranes. These proteins as stated are classified into three major groups namely

a) Integral membrane proteins
b) Transmembrane proteins
c) Peripheral membrane proteins

The fluid mosaic model of biological membranes [3], details and description with the picture are provided.

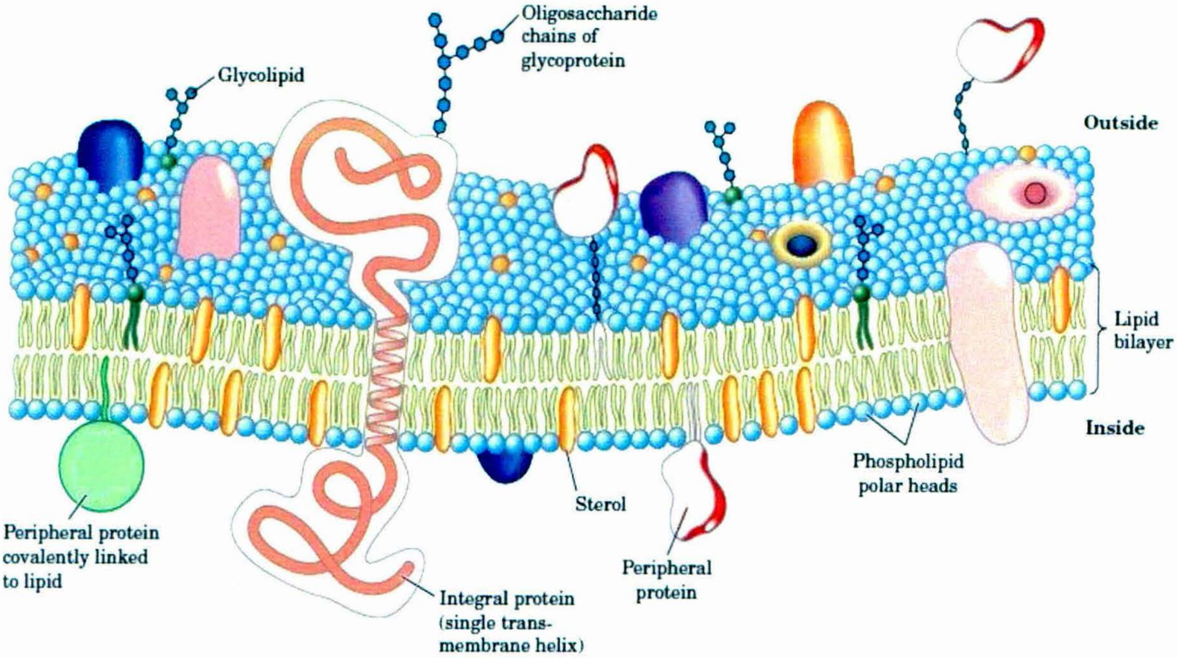

Oligosaccharide chains of glycoprotein
Glycolipid
Outside
Lipid bilayer
Inside
Phospholipid polar heads
Sterol
Peripheral protein covalently linked to lipid
Integral protein (single trans-membrane helix)
Peripheral protein

FIGURE 1 Fluid mosaic model for membrane structure. The fatty acyl chains in the interior of the membrane form a fluid, hydrophobic region. Integral proteins float in this sea of lipid, held by hydrophobic interactions with their nonpolar amino acid side chains. Both proteins and lipids are free to move laterally in the plane of the bilayer, but movement of either from one leaflet of the bilayer to the other is restricted. The carbohydrate moieties attached to some proteins and lipids of the plasma membrane are exposed on the extracellular surface of the membrane.

The image below shows the 3D structure obtained of Aquaporin [4], a class of integral membrane proteins, from PDB

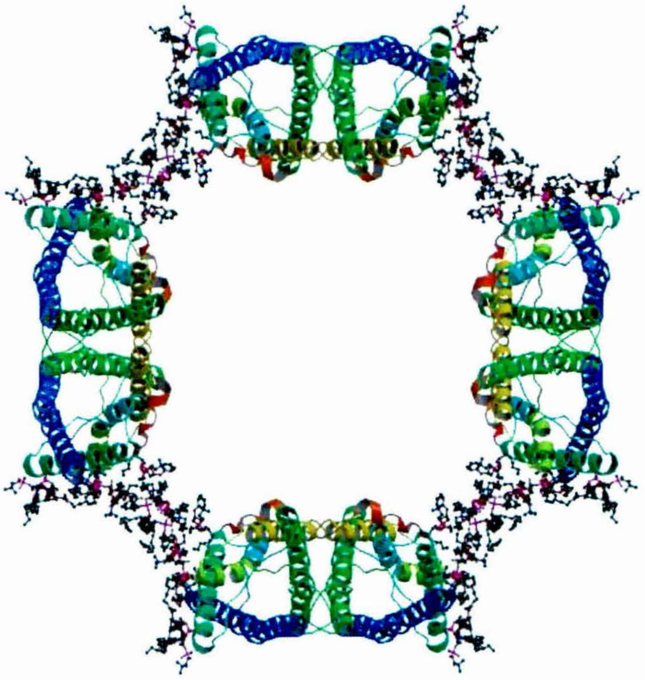


Fig2: electron crystallographic structure of lens Aquaporin-0 (AQP0) (lens MIP) at 1.9A resolution, in a closed pore state

Details of these three classes can be listed as [3]:

I. Integral Proteins:
These proteins are firmly integrated into the lipid bilayer and are not exposed to the outer cytosolic or extracellular environment. These proteins hence are rich in hydrophobic residues.

II. Peripheral Proteins:
Weakly attached to the membrane, these proteins are rich in hydrophilic residues and are exposed towards the outer sides of the membrane. With the membrane they are bounded very weakly through hydrogen bond interactions.

III. Transmembrane proteins:
These proteins are largest among the membrane proteins and have both hydrophobic interiors and hydrophilic exteriors. They cross the whole membrane and are made of integral and peripheral proteins both combined. GPCR or G-Protein Coupled Receptors are special class of transmembrane proteins that are involved in the signal transduction.

Membrane proteins have striking features [3] that distinguishes them from other proteins such as:

a) They have highly distinguishable hydrophobic and hydrophilic regions, identifiable from both sequence as well as structural data (PDB)
b) Hydrophilic regions are exposed to either end of the membranes and generally don't form any secondary structures
c) Hydrophobic regions are buried inside and have higher tendency to form secondary structures
d) Implied by the above facts, these regions are rich in those amino acids that are suitable for that. For example the inner core comprises of hydrophobic amino acid residues and vice versa.
e) Presence of tryptophan residues is observed at the either end of the membranes
f) Charges are more localized rather than distributed evenly over the whole protein. This gives rise to dipole moment and other charge based properties.

Our work considers these features of membrane proteins and tries to quantify them so that each membrane protein could be represented in a vector form. This is explained in more detail in 'METHODOLOGY' section.

1.1.3 Methods of Class Identification

For classification purpose several techniques have been proposed

a) Classification based on sequence similarity: This method used BLAST [4] with BLOSUM [5] or PAM matrices to assign the most similar sequence to the given sequence (query). Then the results are used to identify the class to which the given query may belong to
b) Structural classification CATH [6] & SCOP [7] Databases: proteins have many conserved domains and structural entities that are conserved and common in a family which gives them specificity to a particular function. These databases exploit the same for the classification of protein query
c) Machine Learning: This is another technique in which a classifier is trained using the given samples of a class and then it is used to predict the nature of a given protein sequence or structure.

1.2 Machine Learning Approach

There are several other methods but we mention a few. The last method of machine learning based approach mentioned in previous discussion is quite advantageous for following reasons:

a) Machine learning approach is quite fast. Once a classifier is trained it can give results in almost no time. Given the large database of protein a sequence based query will take a long time
b) Other methods of classification fails to establish complex relationships in the observed features of a particular class of protein, But Machine learning with the concept of hierarchical classifiers can learn many complex patterns that can't be established directly
c) In SYSTEMS BIOLOGY while developing protein interaction networks one needs to look for relationships among the proteins. This is not possible with simple family based classification as it needs to find and establish more complex relationships.

1.2.1 Basic Structure of Machine Learning

The Machine Learning Approach can be explained in simple steps as:

a) Collect the dataset that represents the object of classification, positive sets and the object of non-classification, negative sets.
b) Represent each data in both kind of dataset in the form of features unique to the positive sets but not to the negative sets.
c) Train a classifier considering the fact whether the data is linear or non-linear. In biological world it is mostly non-linear.
d) Use of algorithm is crucial which depends on kind of data, its source and availability in terms of quantity.
e) Test your classifier with previously stored datasets from both classes that were unused for training.
f) Evaluate your classifier based on the predicted results for test sets.

The base of Machine learning Approach is a capable classifier and a suitable algorithm.

1.2.2 Semi Supervised Algorithm [8]

In traditional supervised learning technique two different sets of classes are needed with ample amount of data in both to make the classifier learn correctly. The main limitation of this approach is that examples from both classes are required which is sometimes not available as generally with biological data. Data regarding the membrane proteins belong to this class of problem where the data of membrane proteins is available in small amount as compared to the large database of known proteins. The problem can be addressed by the technique of semi-supervised machine learning also called positive unlabeled (PU) learning that uses a positive set with a large unlabeled set. PU-learning starts by identifying reliable negative (RN) examples iteratively from the unlabeled set until convergence and builds a classifier using the positive and the final RN set.

1.2.3 Artificial Neural Networks

Artificial Neural Network (ANN) is a method of Machine Learning which tries to emulate the functioning of a biological neuron in terms of structure as well as learning an object. ANN is quite vast subject and is used in machine learning for following reasons [9]:

i) Massive parallelism
ii) Distributed representation and computation
iii) Learning ability
iv) Generalization ability
v) Adaptivity
vi) Inherent contextual information processing
vii) Fault tolerance and Low energy consumption

Because of following reasons, especially the (III), (IV) and (VI), we included it in our model of protein classification

1.3 Database and its Importance

Database is important for protein classification as the machine learning approach needs a primer or initial set of data to learn and then only it can do the job of prediction either supervised or unsupervised. Although many databases are available but we considered only those which were more specific for the membrane proteins while for the negative classes we approached general methods so that our classifier is more generalized for the classification task.

1.3.1 Methods to extract data

Once a database has been selected the next task that comes is to obtain the desired datasets from it. This can be done using the following techniques.

1.3.1.1 Advance Query Search using Web Form

Every database provides an option of advanced query search that allows one to select the proteins of a particular class or with a particular property. The IDs associated with the results and other data related to it can be downloaded that depends upon the database. Like Swiss-Prot database allows you to download ID for any amount of results at a time.

1.3.1.2 Web Services (REST or SOAP clients)

Databases sometimes provide options of web services [10], [11] which allows user to connect to the server through a computer code and send requests for different kind of data. We have used such Web Service for PDB using the SOAP Client.

1.3.1.3 Python Based Web Page Parsing

Sometimes webpage don't allow either downloading online or through Web Services. It just displays results for a query. In such situation this method becomes useful. In this method the webpage is saved either through a browser or through a code and then using REGEX search the concerned data is extracted out separately. Python language provides a good number of handy tools for such complicated tasks.

2 LITERATURE SURVEY

As discussed earlier the approach of Machine Learning for protein structure prediction and classification is not new rather old and very popular among scholars. In general it has been observed that most of the articles revolve around the different techniques or algorithms that can be used for this task. The following table summarizes the findings in PubMed [12] search for machine learning techniques for protein classification.

PubMed SEARCH QUERY	Numbers (2008-2013)
Articles on Protein Classification/ Prediction	44957
Protein classification using Machine Learning/ classifier	221
Protein classification Using Semi-Supervised Algorithm	21
Protein classification using neural network	155

Table 1: Results of PubMed search for Research Articles

The results show that the topic of protein classification and prediction is a widely discussed and analyzed one. Machine learning approach is also found in good number of research articles. The semi-supervised algorithm since its inception has quickly found a place in this computational task with 21 articles mentioning them. The success of ANN is quite visible from the results that 155 research articles discuss it for solving the protein classification tasks.

In our work we have focused more on finding a possible set of optimum features that could enhance the performance of classifiers and algorithms. This in turn could help us to develop "**automated protein data generators**".

3 METHODOLOGY

3.1 Outline of Approach

This work discusses about the membrane protein properties and algorithms that can be used to classify them. Use of apt properties (as features in input) has shown to give better results in the classification task.

We calculate (or derive) several properties that are closely related to the membrane properties and generate a feature matrix for the dataset available to us. This feature matrix is then used to train the classifier (or network). Before training a part of all dataset available to us for both categories (labeled and unlabeled) are kept aside, to be used for testing, called test sets. The remaining are called training sets. The trained classifier is then tested with training sets and test sets. Results are recorded in terms of accuracy, specificity and ROC plot. Same set of data is used with different algorithms to generate classifiers and the results are compared on the basis difference in accuracy and specificity.

We also keep some generalized properties like alpha propensity and hydrophobicity plot to understand the nature of membrane proteins better. In nutshell the objective of this work can be outlined as:

- Identify OPTIMAL features from protein sequence and PDB structures to classify the membrane proteins
- Build a classifier based on Semi Supervised Learning using Naïve Bayes Classifier and Neural Network Classifier
- Test classifier with appropriate data and include testing of neural network classifier for number of hidden neurons
- Analysis of properties, other than features used for classification, for post-classification prediction of properties of the membrane proteins.
- Provide outline for use of such classification

3.2 Computational Resources Used

- Below is a table describing various computational resources used for the work

PARAMETER	DESCRIPTION
Processor	Pentium Core 2 Duo
RAM	3GB
Operating System	Linux
Programming Language(s)	Python[13], MATLAB[14], Linux Shell

Table 2: Computational resource statistics

The cost of computational resource was a consideration and hence it was ensured that it is done with minimum possible computational resource keeping in mind that it does not hamper performance either.

3.3 Tools Involved

This section discusses the tools that were used to gather data regarding feature matrix generation that is properties of the membrane proteins.

3.3.1 Weizmann Dipole Server

This server [15] is hosted by the Weizmann Institute of Science, Israel and provides web application that takes the PDB Id as query and calculates the dipole and other charge dependent properties of the protein. We used this server to obtain values of dipole moment, overall charge, charge/nat, dipole/nat and Rm (mean radius) values of the proteins.

3.3.2 Propka 3.0

Propka 3.0 [16], [17], [18], [19] is a program that runs locally to calculate the optimum energy of stability, pH and pI values for the protein. Another reason to use this program was to filter those proteins whose PDB data contained missing residues. Such data is problematic for the calculation of dipole moments and hence needs to be removed to have high curated data.

3.3.3 Fpocket [20]

This program runs locally to identify the pockets in any given protein from its PDB file. We used its results for comparison with the results of the "graph based pocket detection method for proteins".

3.3.4 Prody [21]

It's a python package which specializes in handling the PDB files for various analyses. It was used to download and store the PDB files in large numbers from the initial curated lists of PDB files

3.3.5 PDB (Protein Databank)

Protein Data Bank is an online resource portal from where all information regarding the crystal structures of a known protein can be obtained

3.4 Data Extraction

Protein data (PDB Files) were extracted from various sources to ensure maximum coverage of various membrane and non-membrane proteins. Below we discuss various methods used to extract data from different sources. Species were not considered strictly, although preference was made for Human, Mouse and Yeast genomes.

- Swiss Prot
  With a simple query search for "membrane proteins" returns a list of proteins with Swiss Prot Accession numbers, which is downloadable in various format. These accession numbers were copied and used in PDB Advance Search Page. This feature of PDB Website allows user to provide Swiss Prot [22] accession IDs as input and gives corresponding PDB entries. The tool also allows providing sequence identity cutoff. The results (PDB IDs) can be downloaded in text file and further processed.
- TMBase (from TMPred server page)
  TMBase [23] provides a list of Proteins with Swiss Prot ID which is downloadable. The downloaded file is the used to generate a Swiss Prot ID list and processed further with same technique as mentioned above (using PDB Website Advance Search).
- MPTopo Server:
  MPTopo [24] website provides details of structurally determined membrane proteins. The webpage itself was saved (as they don't provide link to download list) and from the html page the PDB IDs were extracted
- SMART Domain Based Database Search
  Initially we included the proteins searched through this database but there were several anomalies in this process. Firstly that there is no proper method to obtain data from SMART Database [25] using search and secondly it is doubtful that the DOMAIN may be present in some other non-membrane proteins too.

The table below provides statistical details about the Protein data extracted

SOURCE	NUMBER OF ENTRIES PARSED		NUMBER OF ENTRIES DOWNLOADED	
Swiss Prot	5000 unlabeled	1000 labeled	3000 unlabeled	1000 labeled
TMBase	3496 labeled		2000 labeled	
MPTopo	707 labeled		707 labeled	
SMART Domain Search	NA		NA: Not Included in our model	
TOTAL ENTRIES DOWNLOADED			3707 labeled and 3000 unlabeled	

Table 3: Data Extraction Statistics

Since the data source for TMBase is also Swiss Prot, most of them were truncated by the CD-HIT as redundant and similar categories.

3.5 Feature Matrix Generation

Feature matrix is the most important part of machine learning based classification. In simple terms, it is the matrix that defines a set of properties (also called features) that describes best about the object to be classified. For example a flower can be classified on the basis of color, height of plant and number of petals as features. So when many objects are converted or expressed in terms of features describing them, then it forms the matrix. A general convention is to keep the rows and columns of the matrix as objects and its features respectively.

3.5.1 Generating Representative sequence and Information Enhancement

The final number of proteins used for classification was 478 for unlabeled class and 133 for labeled class (membrane proteins). But before this the initial sequences were scanned through CD-HIT program to contain only representative sequences and then through PROPKA 3.0 and Dipole Server to obtain highly curated data for classification studies.

3.5.1.1 CD-HIT sequence similarity cutoff

CD-HIT [23] is a tool that can be run locally to select out those sequences from a large group of sequences that have sequence similarity less than the desired level. For positive groups the sequence identity was kept 40%, because the dataset will be less as compared to unlabeled ones and hence we need more informative data. For unlabeled class it was kept 70% to keep the dataset large by removing only fewer sequences. Following table shows the statistics after CD-HIT use for both classes of sequences.

CLASS	BEFORE CD-HIT RUN	AFTER CD-HIT	SEQUENCE IDENTITY
Positive Labeled	3700	650	40%
Negative Unlabeled	3000	978	70%

Table 4: Generating representative sequences using CD-HIT program (standalone)

3.5.2 Hydrophobicity

Membrane proteins are known to have very distinct property of separate hydrophobic and hydrophilic regions. Based on the hydrophobicity a very good classification feature can be made for membrane proteins. For the training we use 'sum of hydrophobicity' values for individual amino acid residues in the sequence as a feature vector. The hydrophobicity values were taken from **Kyte-Doolitle** scale [31]

After a protein is classified as membrane protein, number of 'Transmembrane' regions could be estimated by hydrophobicity profile. To generate a hydrophobicity profile we take the protein sequence and then calculate the local average for each amino acid in the sequence by averaging within a window. For this we took a window of 7 residues and the average of the hydrophobicity values within the window is the local hydrophobicity value for the central sequence. This procedure of local averaging is continued (scanning through window) residue by residue starting from 4th residue. Following illustration helps to understand how it is done

A	W	R	V	P	G	M	M	L	K
	3 RESIDUES TO LEFT OF 'P'			CENTRAL	3 RESIDUES TO RIGHT OF 'P'				
1.8	**-0.9**	**-4.5**	**4.2**	**-1.6**	**-0.4**	**1.9**	**1.9**	3.8	-3.9
Above values are hydrophobicity value for residue by Kyte-Doolitle scale [28]									

Figure 3: Hydrophobicity window with 7 residues. Highlighted residues are within window

The central residue will have average hydrophobicity = [-0.9-4.5+4.2-1.6-0.4+1.9+1.9] / 7 = 0.08

Likewise it is calculated for all possible residues to give a hydrophobicity profile. It can be plotted to give a hydrophobicity variation along the sequence. More will be discussed and figures will be explained later in this thesis.

3.5.3 Moment Based Properties

3.5.3.1 Dipole Moment

Membrane proteins being rich in hydrophobic residues (which are of course charged) have quite a significant dipole moment. This provides a remarkable feature difference from other proteins and hence a good one to classify the membrane proteins.

3.5.3.2 Quadrapole moment

Quadrapole moment is more localized property for molecules with high charge profile. Membrane proteins having large hydrophobic residues show up large concentrated electric moment at several locations (the part or area where charges are concentrated)

3.5.4 Charge Based Properties

3.5.4.1 Overall Charge

The sum of individual charge of the amino acid residues in the protein makes the overall charge of the protein. The value of overall charge was extracted from Weizmann Dipole Server [15].

3.5.4.2 Charge per Native

It is the charge per unit mass. The values of charge per native were extracted from Weizmann Dipole Server [15].

3.5.5 Energy & Stability Based Properties

3.5.5.1 Optimum pH

Optimum pH is the theoretical values at which the most stable form of any protein will remain in isoelectric point (the state at which the positive and negative charges on protein coexist). The values of optimum pH were calculated by running Propka 3.0 [16] locally with default parameters

3.5.5.2 Optimum Energy of Stability

The energy at which the protein is most stable and it is theoretical calculated value. The values of optimum energy were calculated by running Propka 3.0 [16] locally with default parameters

3.5.5.3 Isoelectric Points: $pI_{(folded)}$ and $pI_{(unfolded)}$

It is the isoelectric point of protein when it will be folded and unfolded. All proteins have definite isoelectric point at which they remain unfolded or folded. This isoelectric point is important for structural modifications in protein. The values of both kinds of pI have been extracted from Weizmann Dipole Server [15]

3.5.6 Other Properties Analysis

3.5.6.1 Transmembrane Regions

We also included two features post identification of membrane proteins. If any protein is detected to be membrane protein then number of transmembrane regions (or hydrophobic regions) could be estimated. This number gives an idea of size of protein. More will be discussed about the propensities and transmembrane region in **Post Classification Analyses** topic

3.5.6.2 Alpha and Beta Propensities

Alpha and Beta propensities are the tendencies of any amino acid to form alpha and beta strands. Membrane proteins have high tendencies to form alpha and beta strand in the hydrophobic core if they are transmembrane segments so as to maintain the stability of the structure. The exact propensity values are determined experimentally in terms of change in free energy. The work by ***Nick Pace et al*** [27], ***L. Regan et al*** [28] and [32] is the source of predefined values for both propensities that we have used in this work.

Amino Acid	Hydrophobicity	Alpha (Helix) Propensity	Beta (sheet) Propensity
Alanine (A)	1.80	1.37	0.72
Arginine (R)	-4.50	1.13	0.82
Asparagine (N)	-3.50	0.77	0.76
Aspartate (D)	-3.50	0.73	0.76
Cytosine (C)	2.50	0.85	1.07
Glutamine (Q)	-3.50	1.21	0.82
Glutamate (E)	-0.40	1.25	0.86
Glycine (G)	-0.40	0.59	0.81
Histidine (H)	-3.20	0.85	0.98
Isoleucine (I)	4.50	1.01	1.39
Leucine (L)	3.80	1.27	0.93
Lysine (K)	-3.90	1.13	0.98
Methionine (M)	1.90	1.29	0.84
Phenylalanine (F)	2.80	0.99	1.10
Proline (P)	-1.60	0.41	0.42
Serine (S)	-0.80	0.80	0.85
Threonine (T)	-0.70	0.84	1.08
Tryptophan (W)	-0.90	1.09	0.91
Tyrosine (Y)	-1.30	0.98	1.12
Valine (V)	4.20	0.89	1.57

Table: 5 Amino acid hydrophobicity and secondary structure propensity values

3.5.6.3 Binding Sites

Membrane proteins having functional role, have one or more binding sites that provides it functionality. The binding sites can be estimated using '**Fpocket**' program [20] which is based on convex hull approach or by "**Graph Based Identification**" [29]. Our target was not just to identify the pockets in a protein, but to analyze the effect of cut off radius and fraction of Van-der Waals surface area exposed to binding. This was done by estimating the binding sites by fpocket program and then by the graph method. The atoms detected by graph method were compared to those of fpocket program for first two best outputs of each. This step was repeated for different values of cut off radius and Van-Der Waals fraction of exposure. The numbers of matches in both results were counted every time and any change in it was also recorded. The results have been tabulated and will be discussed in following section.

3.5.6.4 Aliphaticity Index and Extinction Coefficient

These were calculated based on formula provided by SwissProt protein properties calculation server. Formulae for both are

Aliphatic Index:

The aliphatic index [30] of a protein is defined as the relative volume occupied by aliphatic side chains (alanine, valine, isoleucine, and leucine). It may be regarded as a positive factor for the increase of thermostability of globular proteins. The aliphatic index of a protein is calculated according to the following formula

Aliphatic index = X (Ala) + a * X (Val) + b * (X (Ile) + X (Leu))

Where X (Ala), X (Val), X (Ile), and X (Leu) are mole percent (100 X mole fraction) of alanine, valine, isoleucine, and leucine.

The coefficients 'a' and 'b' are the relative volume of valine side chain (a = 2.9) and of Leu/Ile side chains (b = 3.9) to the side chain of alanine.

Extinction coefficient:

The extinction coefficient [30] indicates how much light a protein absorbs at a certain wavelength. It is useful to have an estimation of this coefficient for following a protein which a spectrophotometer when purifying it.

E (Prot) = Numb (Tyr)*Ext (Tyr) + Numb (Trp)*Ext (Trp) + Numb (Cysteine)*Ext (Cysteine)

Absorb (Prot) = E (Prot) / Molecular weight

During analysis we found that values of these two properties for these two classes do not show considerable differences rather the values for most of the proteins lie within a most probable range and hence this redundancy lead us to conclude that these two parameters are not good for classification and were not included in our final feature matrix.

3.5.7 Statistical Analysis Factsheet

3.5.7.1 Mean, Standard Deviation & Fisher's Score Analysis

The mean and standard deviation for all the features selected were calculated for the class of datasets that we have produced. The negative dataset statistics have large standard deviation because of non-homogenous source of data class. **Fisher's Score** is also calculated for all features of both the classes. Fisher's score gives an idea of how much the selected feature is capable to distinguish the two classes. Greater the score higher is its classification capability. These figures also give an idea of how the values of selected features vary for membrane and non-membrane proteins. Greater the difference of means greater will be distinction ability of a feature. Following table gives the mentioned statistical figures

$$Fisher's\ Score\ (FS_j) = \frac{(\mu_1 - \mu_2)^2}{(\sigma_1^2 + \sigma_2^2)}$$

Where j is the feature number and 1 and 2 subscripts denote the classes under observation

FEATURES	Fisher's Score	MEAN VALUE FOR DATASETS		VARIANCE FOR DATASETS		DIFFERENCE OF MEANS
		POSITIVES	NEGATIVES	POSITIVES	NEGATIVES	
Hydrophobicity sum	1.9802	0.34817	-0.3993	0.348	0.1287	0.522
Rm (mean radius)	$1.08*10^{-6}$	685.248	359.819	284652.899	131622.193	325.428
Charge	0.9104	-0.00035	-0.00866	0.00407	0.0077	0.00831
Dipole	$7*10^{-8}$	1444.188	666.504	2773333.7	784926.839	777.684
Quadrapole	0.0	10274.03	4650.359	249409637.38	114238160.6	5623.670
Crg/Nat	100.03	-0.00016	-0.00067	10^{-5}	$5*10^{-5}$	0.00051
Dip/Nat	0.2474	0.3083	0.3791	0.03616	0.1377	0.0708
Optimum pH	0.0016	5.692	6.487	14.16076	13.571	0.79551
Optimum Energy	$1.01*10^{-5}$	124.392	43.441	23989.809	8225.835	80.9511
pI folded	0.0046	7.551	7.185	3.336	4.201	0.3658
pI unfolded	0.0017	7.408	7.185	3.548	4.080	0.2229

Table 6: Data Statistical Analyses Results for Features Selected

3.5.7.2 PCA Analysis and Biplot of features

After calculating the statistical features we made a PCA Analysis and then generated the Biplot to observe which features were contributing most to the first two principal components. All the features were are not included to reduce the low energy components. The "**Energy based Parameters"** were removed from the plot and only those which contribute significantly were plotted.

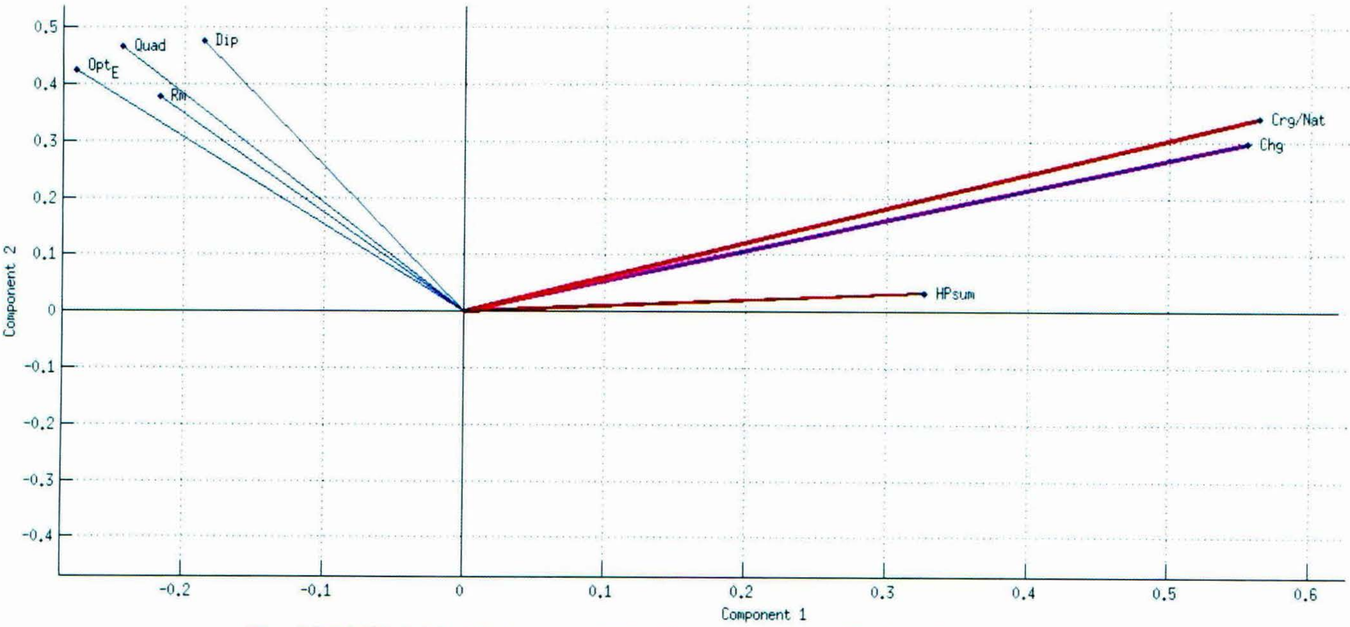


Fig: 4 PCA Biplot for feature vectors selected for classification task

Upon observation it is clear that the features which are based upon charge on the amino acid residues are prominent in defining the membrane proteins. Those which give most weight to the first two components are colored. It is quite evident from the plot that the features on the right **"charge"** and **"charge/nat"** are highly correlated for the whole database and similarly the features on the second half of the plot. It is quite interesting to note that all the features have been categorized into two different sections by the plot. The feature **R$_m$** and **Quadrapole** are most correlated among all giving possibility that Quadrapole varies with mean radius of proteins (or say protein size).

Features	**Contribution to the component 1**	**Contribution to the component 2**
Charge	0.57	0.30
Charge/Nat	0.58	0.35
HP Sum	0.32	0.03
Dipole	-0.19	0.48
Quadrapole	-0.25	0.47
Optimum Energy	-0.29	0.42
R$_m$ value	-0.22	0.38

Table 7: Features contributing most to the principal components in scale of -1 to 1 (through PCA Analysis)

3.5.7.3 Optimal Features for classification

From the statistical analyses and other observations it was found that only few properties in actual can define the membrane proteins. These properties are charge based properties like hydrophobicity, dipole etc. The reason that could be attributed is the presence of high number of charge residues both as overall composition and segregation in form of segments. Other properties although quite relevant to membrane proteins do not make useful because of two reasons

a) The dataset for negative ones used is highly varied and hence includes very large sized to very small sized proteins. It is quite reasonable that a property will show high standard deviation from mean and resemblance to the membrane proteins if its size is too large and vice versa.

b) Membrane proteins included peripheral proteins also that lack several features which are unique to transmembrane proteins, like size and Aliphaticity index.

SO FINAL SELECTED FEATURES FOR CLASSIFICATION ARE

a) OVERALL CHARGE
b) HYDROPHOBICITY SUM
c) DIPOLE MOMENT AND
d) CHARGE/NAT

Below we represent a feature matrix for a membrane protein that will be generated using these features:

PROTEIN	HP_Sum	Avg Charge	Rm	Dipole	Quadrapole	Crg/Nat	Optimum pH	Optimum E	pI (u)	pI(f)
Pro 1	-210.6	-6	1709	14858	-0.0007	0.1865	9.5	543.4	6.94	7.15
Pro 2	-223.9	-13	1279	5060	-0.0026	0.2562	4.1	107.5	6.53	5.53
Pro 3	-130	3	1517	3094	0.0004	0.188	3.7	442.5	8.04	8.02
Pro 4	-76.6	6	1886	1393	0.0024	0.7462	4.3	45.1	9.3	9.25
Pro 5	-13.9	-6	2597	7192	-0.0006	0.2605	9.9	463.8	7.17	7.03
Pro 6	-96.7	14	1168	12604	0.0019	0.1587	4.2	133.6	9.21	9.39
Pro 7	-493.5	-3	1270	12271	-0.0006	0.2672	3.7	159.8	7.08	7.12
Pro 8	-61.6	4	911	944	0.0027	0.6194	5.7	11.7	9.37	9.43
Pro 9	-84.9	35	1489	11048	0.0061	0.2583	10.4	182.5	9.88	10.02
Pro 10	-273.9	-37	1297	9400	-0.0041	0.1446	7	164.1	5.45	5.48

Table 8: Feature Vector Representation of proteins

Now after the optimal features have been selected we move towards the simulation of the same using the datasets mentioned earlier. The simulation procedure runs the code and results including the ROC plot and Confusion Matrix was generated using the MATLAB [14]. More is discussed in the next section.

3.5.8 Semi Supervised Algorithm

The algorithm used for training and classification is Semi Supervised Algorithm [8] which can be used with any classifier and is advantageous in cases such as biological data where generally we have small dataset of concerned class and a large mixed database from where we want to separate out our concerned class of data. The algorithm exactly we used is a special case of semi supervised algorithm, called **spy technique [8]**. In the **spy technique**, "spy" examples from the positive set (called the P set) are sent to the mixed or unlabeled set (called the U set) (Figure 3). This approach randomly selects s% of the examples from the Pset (in our experiment, we use 15%). These examples form the 'spies' set, denoted by S, which is added to the U set. The spies behave identically to the unknown positive examples in U and hence allow us to reliably infer the behavior of the unknown positive examples.

Semi supervised algorithm first selects the spy sets and then adds it to the negative sets. The classifier is eventually trained with positive sets (without the spy sets) and negative sets (including the spy sets). The trained classifier is then used to test with the negative sets to check which among them are classified as positives. The classified false positives are removed which gives real negative sets and again the classifier is trained with positive sets (this time with spy sets) and the real negative sets. Again the classifier is tested with the negative sets and the false positives are identified. The steps mentioned above are repeated, till no samples in subsequent negative sets are classified as false positives. The final real negative sets and original positive sets will be used to train the final classifier and eventually for the classification task.

3.5.9 Performance Criteria

The performance of our classifier with the selected feature sets is evaluated on four parameters, they are:

i) $Accuracy = \dfrac{True\ Positive + True\ Negatives}{Total\ Predictions}$

ii) $Precision = \dfrac{True\ Positive}{True\ Positives + False\ Positives}$

iii) *Sensitivity = True Positive Rate*

iv) *Specificity = True Negative Rate*

These results have been shown as confusion matrices which are designed as a matrix. Confusion matrix is a table layout that allows one to see the performance of an algorithm. Columns represent Target Class and Rows represent Predicted class while the third column and row as shown in **Fig. 6 & 7** represent the overall percentage of correct classification.

A plot is also made between **True Positive Rate** and **False Positive Rate called Receiver Operator characteristics or ROC-plot.**

Predicted Class ↓	Target Class →	Negatives	Positives
Negatives		True Negatives	False Negatives
Positives		False Positives	True Positives

Fig 5 Confusion Matrix

These performance meters help to analyze the classifier. This also helps to know how the classifier behaves accordingly with the changing features for classification.

3.5.10 Simulation and Cross validation Results

After processing through the Propka [16] and Dipole server [15] programs the final curated data obtained was used for simulation. Following table lists some basic information regarding the simulation parameters

Positive Sets	133
Negative Sets	478
Classifier	Bayesian Classifier, Neural Network
Algorithm	Semi Supervised Algorithm

Table 9: Simulation Parameters

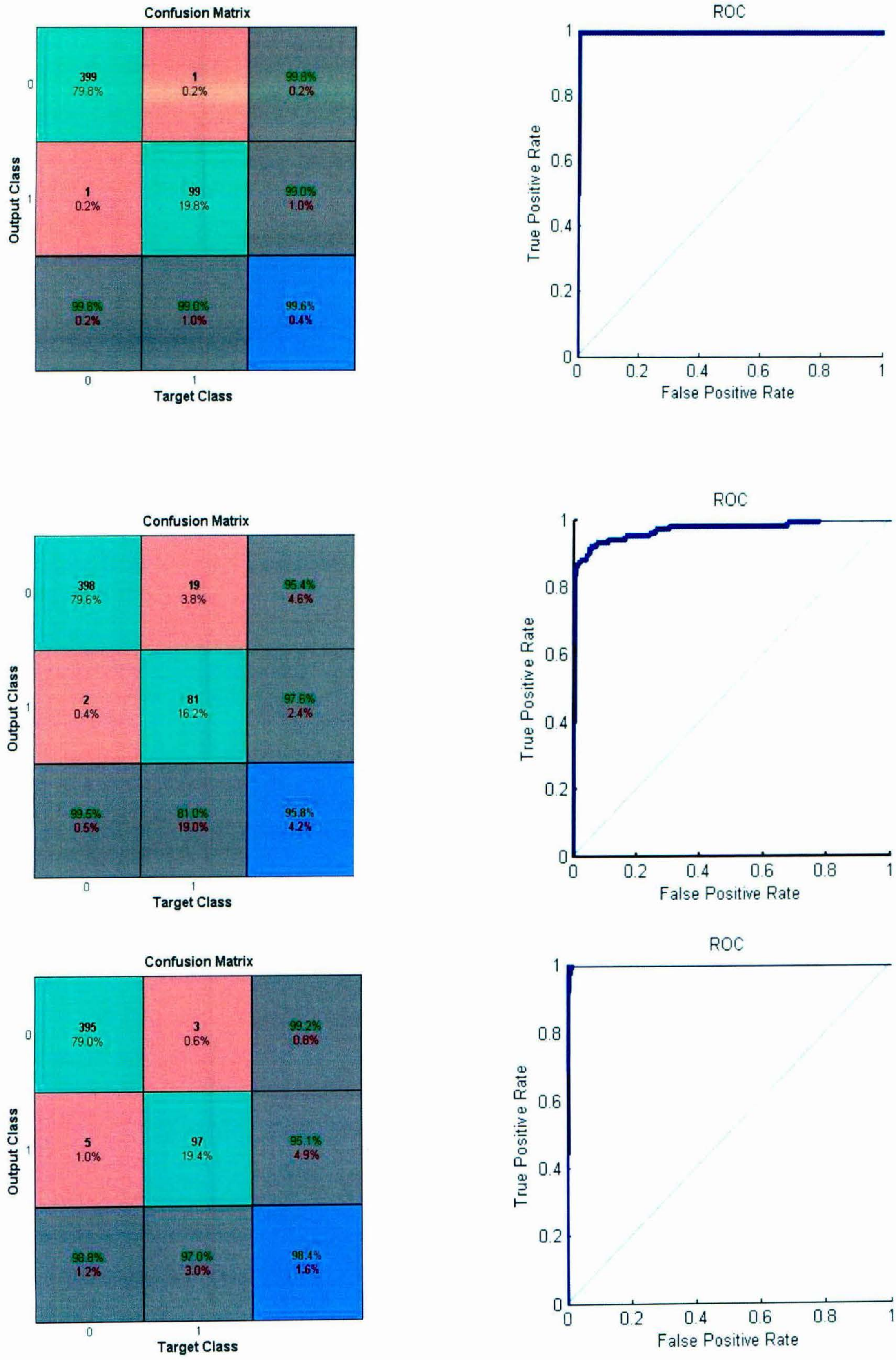
Since the data used for training and testing were representative for the membrane proteins hence smaller number may be misleading but it is equally effective. In fact use of representative highly selected data reduces the computation time without compromising the effectiveness of the classifier learning.

Simulation was carried out with both classifiers, Bayesian and Neural Network and each simulation was carried out with different set of training and validation sets so as to ensure unbiased results. The following table provides summary of results which were averaged for two classifiers (for each parameter) over simulation for 5 times.

Iteration No		% of misclassification	True Positive	True Negative	False positive	False Negative	Accuracy	Sensitivity	Precision	Specificity
1	**Train**	0.60	98	399	2	1	99.4	98	98	99.75
	Validation	0.18	33	76	0	2	98.19	100	100	97.4
2	**Train**	0.4	99	399	1	1	99.6	99	99	99
	Validation	1.8	31	78	0	2	98.2	100	93.9	97.5
3	**Train**	0.4	99	399	1	1	99.89	99.75	99	99
	Validation	1.8	31	78	0	2	98.2	100	93.9	100
4	**Train**	4.2	81	398	2	19	95.8	95.4	81	95.4
	Validation	16.2	18	78	0	15	86.5	81.25	54.5	100
5	**Train**	1.6	97	395	5	3	98.4	95.1	97	95.1
	Validation	2.7	31	77	1	2	97.3	99	93.9	99
6	**Test Set (30 samples)**	3.3	29	NA	NA	1	96.7	NA	NA	NA

Table 10: Cross-validation Result Summary for 5 iterations

Fig 6: Confusion matrix and ROC plot for three iterations with training sets

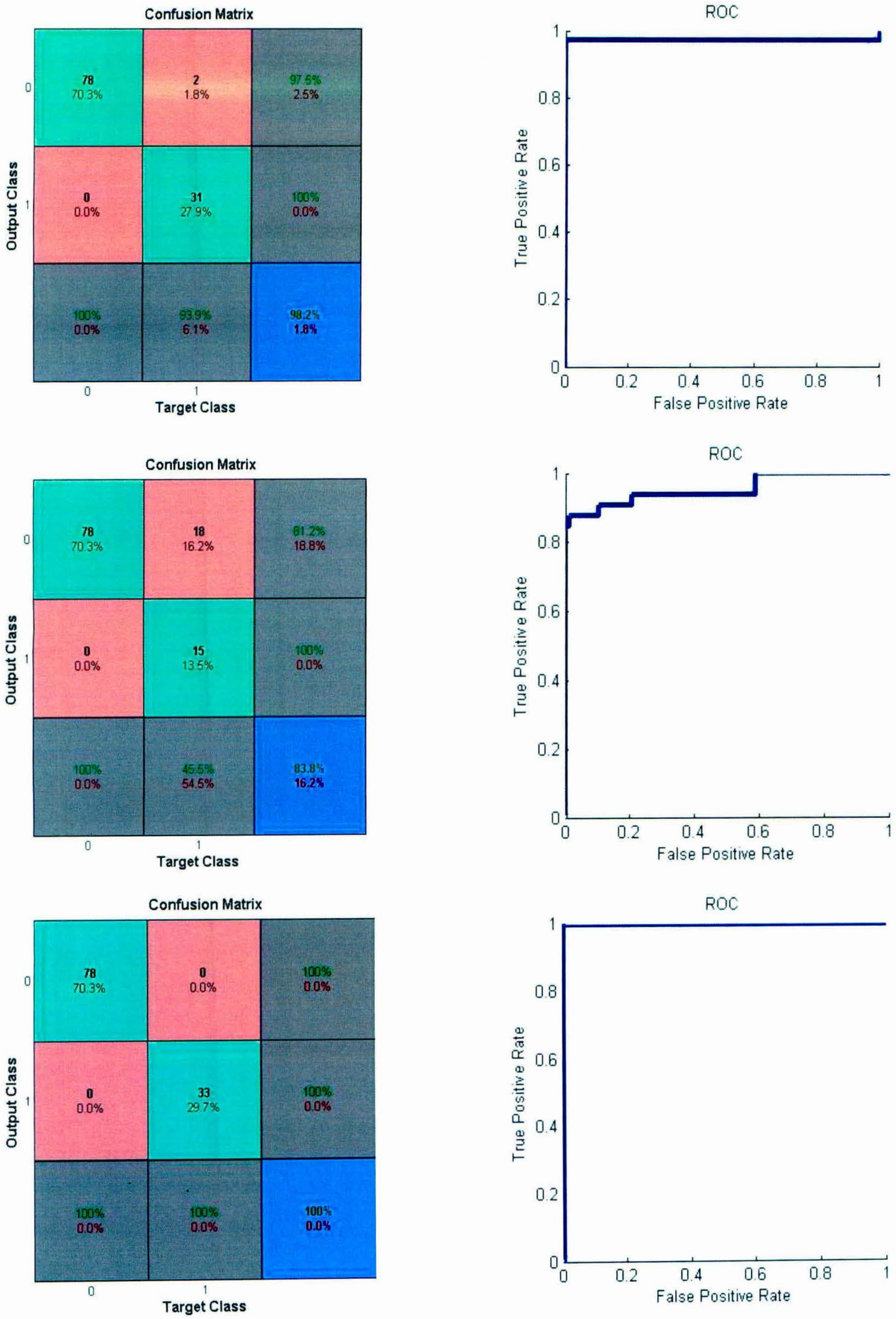


Output Class \ Target Class	0	1	
0	399 79.8%	1 0.2%	99.8% 0.2%
1	1 0.2%	99 19.8%	99.0% 1.0%
	99.8% 0.2%	99.0% 1.0%	99.6% 0.4%

Output Class \ Target Class	0	1	
0	398 79.6%	19 3.8%	95.4% 4.6%
1	2 0.4%	81 16.2%	97.6% 2.4%
	99.5% 0.5%	81.0% 19.0%	95.8% 4.2%

Output Class \ Target Class	0	1	
0	395 79.0%	3 0.6%	99.2% 0.8%
1	5 1.0%	97 19.4%	95.1% 4.9%
	98.8% 1.2%	97.0% 3.0%	98.4% 1.6%

Fig 7: Confusion Matrix and ROC Plot for three iterations with Validation sets



Since we used Neural Network classifier too in our analysis we were interested in knowing how much hidden neurons are good for attaining the extreme accuracy? A simulation using the neural network classifier was done separately **without applying the semi-supervised algorithm** and following results were obtained:

Hidden Neurons	Percentage of misclassification		Sensitivity		Specificity	
	Training Set	Validation Set	Training Set	Validation Set	Training Set	Validation Set
2	0.02	0.03	98.03	97.05	100.0	100.0
10	0.00	0.02	100.0	97.06	100.0	97.06
15	0.02	0.00	98.04	100.0	98.04	100.0
20	0.02	0.00	97.98	100.0	97.98	100.0
25	0.01	0.00	99.00	100.0	99.00	100.0
200	0.00	0.00	100.0	100.0	100.0	100.0

Table 11: Effect of Hidden Neurons on performance of Neural Networks

Following these results a set of plots of the training results and regression analysis for the neural network with 200 hidden neurons was also generated **(Fig9)**. Observation brings several important facts about the classification. Firstly that the algorithm has little role to play if a good dataset and carefully selected feature is available and secondly, number of neurons not necessarily influence learning ability in a neural network rather they increase the complexity. It is expected that hidden neurons may play a good role where features have high degree of relationship among them and when dataset is really large, which is not the case here.

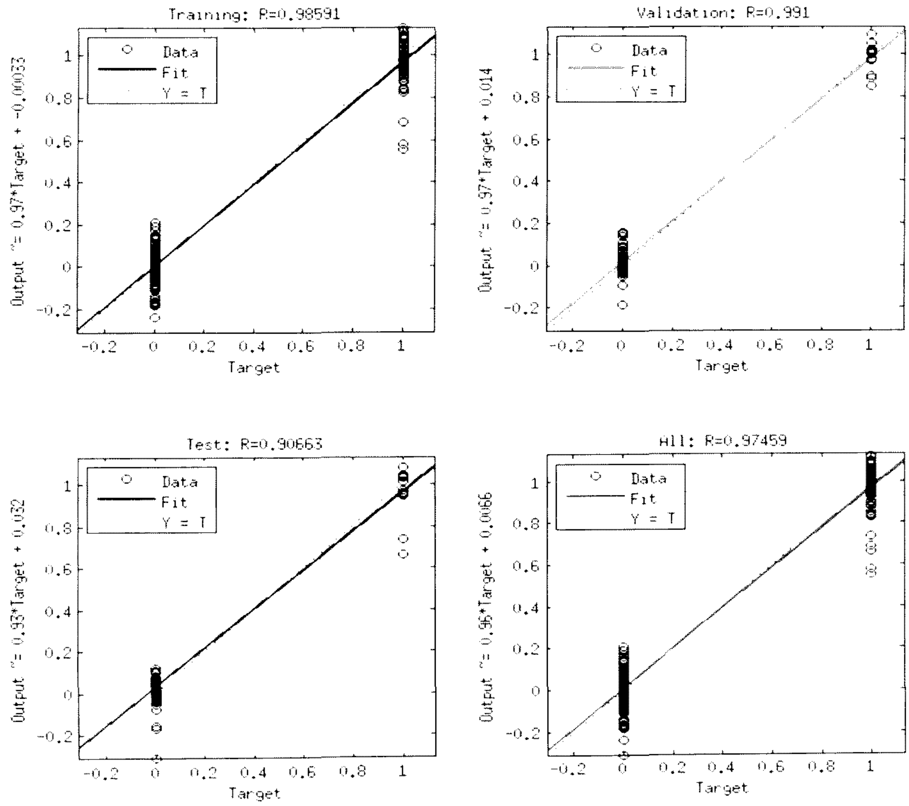


Fig: 8: Regression Analysis Plot for Neural Network Classifier with 15 hidden Neurons

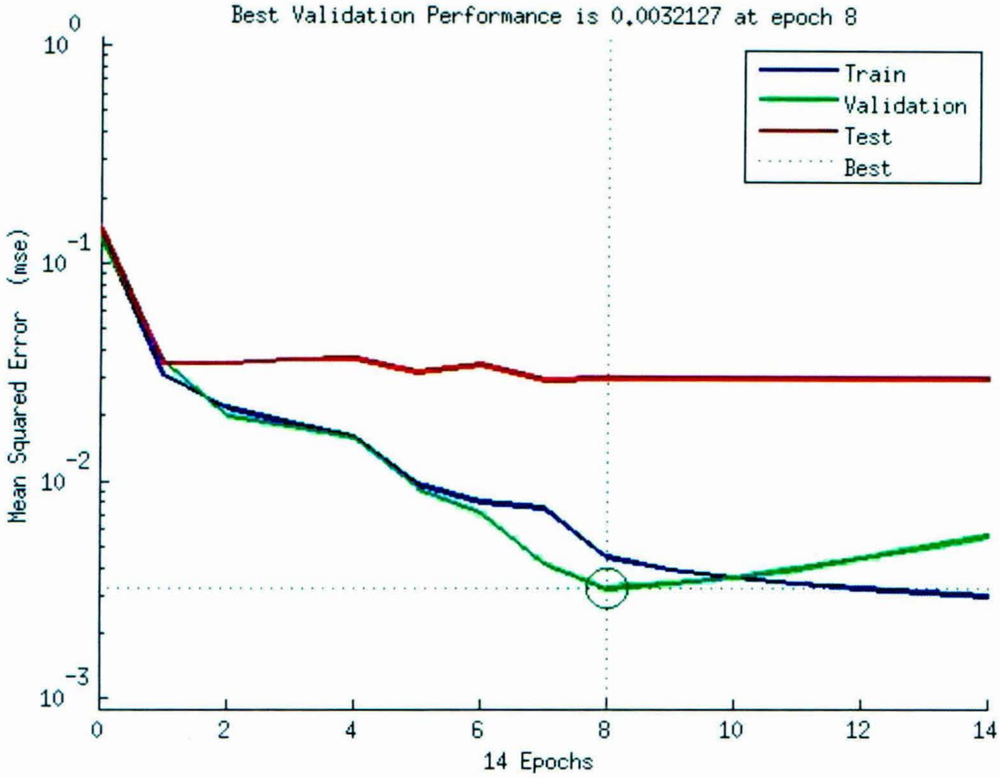


Fig 9: Performance Plot for neural network classifier with 15 hidden neurons

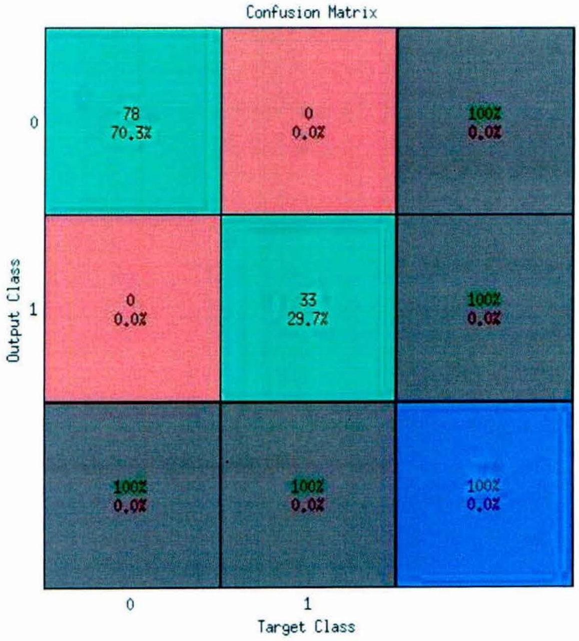


Figure 10: The confusion Matrix for neural network classifier results with 15 hidden neurons

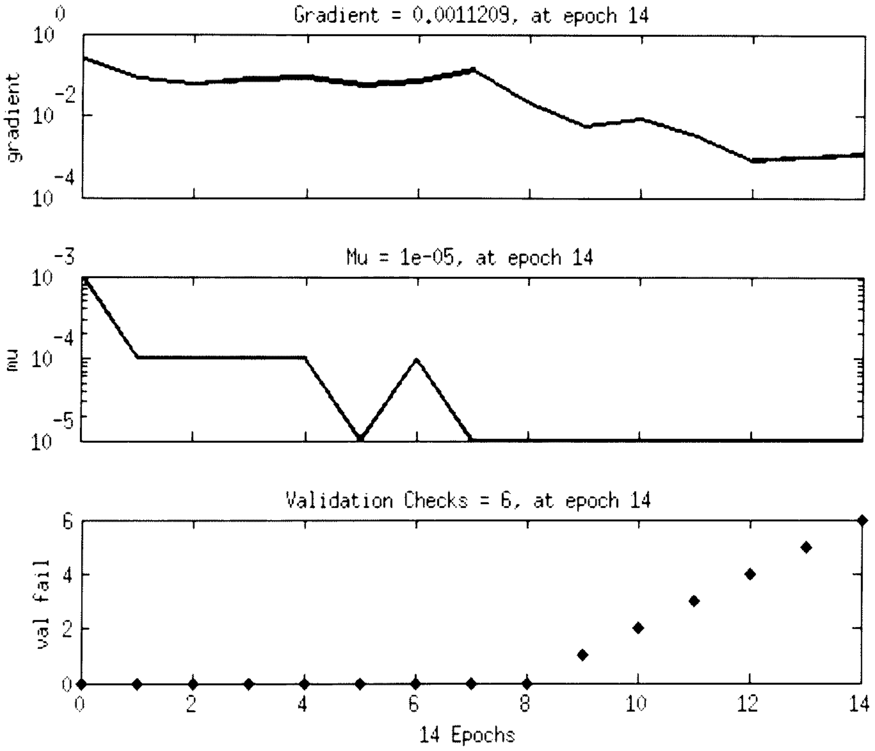


Fig 11: Training State Plot for Neural network classifier with 15 neurons

(a) Gradient Plot
(b) Mu value Plot
(c) Validation fails plot

The Regression Plot is shown in the following **Figure 8**. The three axes represent the training, validation and testing data. The dashed line in each axis represents the perfect

Result – Outputs = Targets

The solid line represents the best fit linear regression line between outputs and targets. The **R value** is an indication of the relationship between the outputs and targets. If R = 1, this indicates that there is an exact linear relationship between outputs and targets. If R is close to zero, then there is no linear relationship between outputs and targets.

The performance plot in **Fig 9** shows the **Mean Squared Error** values for the training, validation and testing data. It is important to know that these training, validation and testing sets are internal partitions of the MATLAB neural network commands and different from one used by us.

The plots in **Fig 10** are used as parameter to stop training process. The magnitude of the gradient and the number of validation checks are used to terminate the training. The gradient in **Fig 10 (a)** will become very small as the training reaches a minimum of the performance. If the magnitude of the gradient is less than 1e-5, the training will stop, which isn't in this case. The number of validation checks in **Fig 10 (b)** represents the number of successive iterations that the validation performance fails to decrease. If this number reaches 6 (the default value), the training will stop, which is the case here at 15 epochs.

MU value controls how much the weights are changed on each iteration and the value to use it depends on the particular problem, being as low as 10-6, or as high as 0.1. If mu value is too small the network will converge very slowly and similarly if it is too large then it will cause the convergence to be erratic. There will be chaotic oscillations around the final solution. So '**mu**' value is also monitored for each epoch.

3.5.11 Post- Classification Analysis

3.5.11.1Predicting regions with possible secondary structures

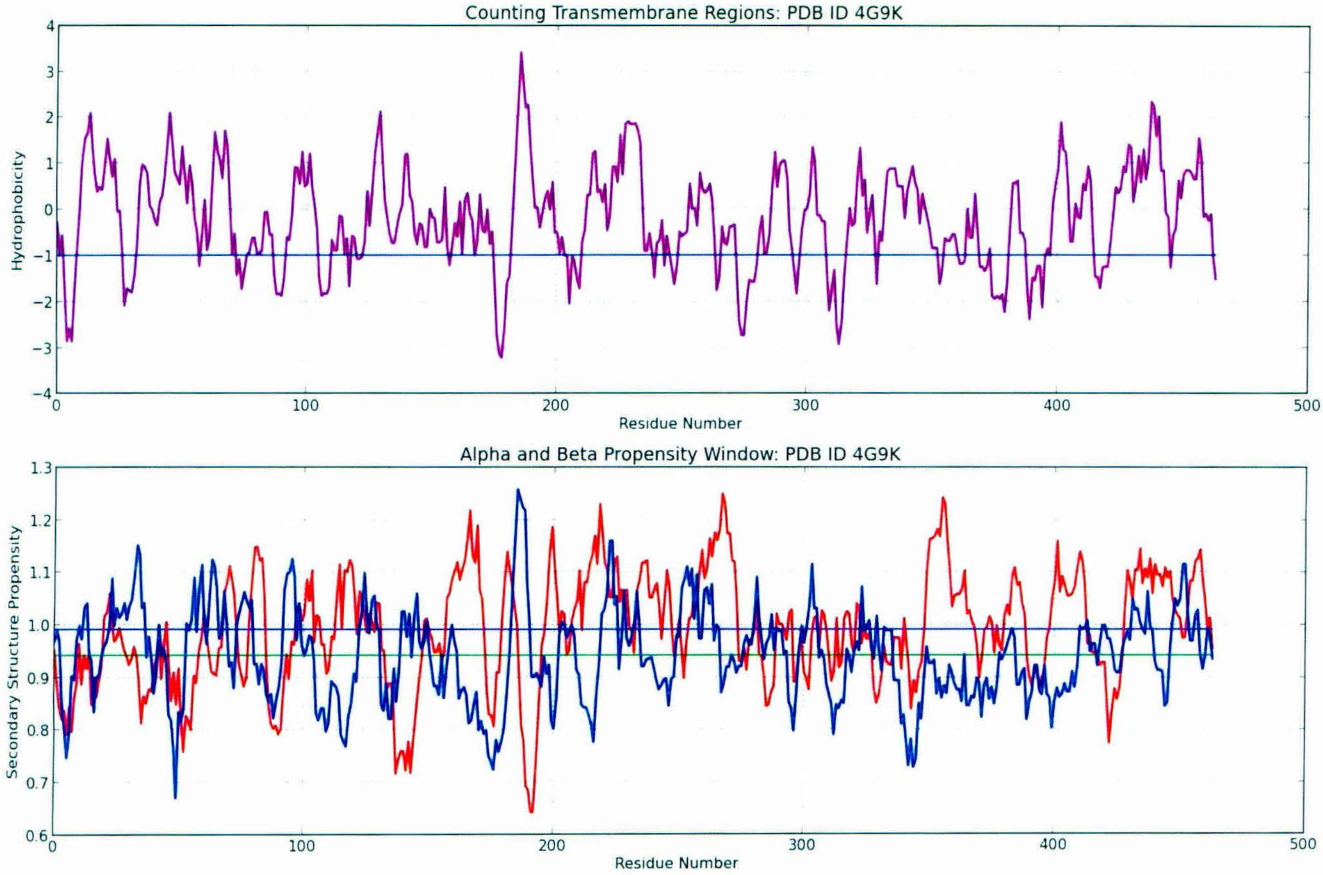


Fig 11: (a) Hydrophobicity 7 residue window plot for PDB ID 4G9K

(b) Alpha and Beta Propensity window plot for PDB ID 4G9K

Upon careful observation it is possible to find the regions with secondary structures in proteins. The regions with possible secondary structures may also represent the region with transmembrane. There are methods to predict secondary structure regions, but here we propose another method that can predict transmembrane regions using only basic information and with lesser complexity. The objective of this method is to provide insight into the secondary structures after the sequence has been classified as membrane proteins.

For three sequences a table has been showed in the next page and its comparison with the secondary structure regions shown by PDB results on the same protein.

Predicted Secondary Structure Sites		PDB Represented Sites of Secondary Structures
Sequence 4G9K	Stretch	Stretch
ISFLK	23-27	44-46
FAL	71-73	55-59
YYE	81-83	63-71
SLSA	100-103	78-83
VSQ	104-106	87-96
KYDY	125-128	104-107
LIS	129-131	111-117
FLK	151-153	122-150
IRRTF	161-165	165-174
LSIV	186-189	188-271
DYV	206-208	283-357
QDL	210-212	378-419
EVQIHLVE	222-229	427-433
ALPIV	230-234	445-449
NMF	236-238	455-505
VHLRTAVAKVEEKQ	257-270	
LIWAT	292-296	
NDF	324-326	
KIDLLF	383-388	
IATI	413-416	
TFYLWRILYLSMI	431-443	
RLKVFF	449-454	
WIKLAF	456-462	

Table 11: Comparison of predicted and PDB represented secondary structure regions in membrane protein PDB ID 4G9K

3.5.12 Summary of Results

The results can be summarized as:

Optimal Features	**Overall charge, HP sum, DIPOLE moment, Charge/Nat**	
Feature with Highest t-score	Charge/Nat	
	TRAIN SETS	**VALIDATION SETS**
Average Accuracy	98.62	95.68
Average Precision	94.8	87.24
Average Sensitivity	97.45	96.05
Average Specificity	97.65	98.78

Table 12: Summary of Results Using Naïve Bayes Classifier

Classification Set Type →	**TRAIN SETS**	**VALIDATION SETS**
Average Sensitivity	98.04	98.04
Average Specificity	100.0	100.0
Hidden Neurons	15 neurons	

Table 13: Summary of Results Using Neural Network Classifier

4 DISCUSSIONS

4.1 What we can conclude?

From the results we can conclude several properties of membrane proteins and its classification with greater confidence

a) Properties that are function of charge on residues in a protein are the best features for classification
b) A sequence based classification is possible because the above mentioned property is reflected in protein sequences too
c) Any size based property will be a bad option for classification if semi supervised learning is being used as an option because the negative dataset is quite varying in its composition and hence will violate size based features.
d) Semi supervised learning algorithm is a good option when a small data of concern is available and most of the data is of negative class.
e) Fisher's is a good estimator for the analyses of features for any kind of protein classification

4.1.1 Effectiveness of data and features

From the analyses of all the features in our work it was observed that features can be classified as two zone groups. The first zone of features is crucial to segregate out the membrane proteins from the pool of large protein data. These features are primarily the charge based features like hydrophobicity and dipole moment. The second zone of features includes those which can be used for further assurance of classified membrane proteins and for inner classification of membrane proteins into transmembrane and peripheral proteins. For example the feature aliphatic index gives high value for transmembrane proteins while lower for peripheral proteins. Hence effectiveness of any feature is global when it can segregate a data from the pool while it will be local if it can classify within the classified community.

4.1.2 Effectiveness of Algorithms

The classifiers tested show that more than the choice of classifiers it is the data and its features that influence the classification. Of course, the classifier must be designed for non-linear data but that is the primary requirement in such classification problems. Algorithms come to play role when the data size becomes larger and larger. Such as in this case the data pool was large but the concerned data (membrane proteins) was small in number. This high degree of variability can induce problems like biasing and over fitting and hence undermine the whole classification task. Semi supervised learning technique comes to rescue in such complex scenarios with making network learn iteratively with new sets of negative data every time it iterates.

4.1.3 A note about binding sites

The comparison of binding sites from the results of **'Fpocket'** [20] and **'GRAPH Based Pocket Identification'** [29] shows that the later approach to identify the binding sites is either very abstract or the binding sites itself is quite exposed in membrane proteins. This is why we obtain NO variation in identified atoms of binding sites (as compared with fpocket) while changing the cut off radius and exposed Van-Der Waals surface area. To obtain a conclusion about which postulate is correct a deeper analyses may be necessary which is beyond the scope of our current work. A summary of work is provided below.

- The program Fpocket [20] was run for 50 membrane proteins and the identified pockets were compared with the results of Graph Based Pocket Identification Program [29]
- The same was done with non-membrane proteins also but results are not recorded as these proteins belong to random classes.
- Parameters altered were the percentage of van-der Waals surface area and cut off radius. The output is provided in a separate file at the end of the report

Van-Der Waals Exposure	20% and 60%
Cut off range	3Å to 9Å and 15Å to 23Å
Number of proteins assessed (membrane)	50

Table 14: Binding Site comparison fact sheet

4.2 Future Possibilities in Classification: Inner Class Specificity

The membrane proteins themselves can be classified depending upon their functionality. The first classification level would be **Integral Proteins, Transmembrane Proteins** and **Peripheral Proteins.** These proteins then can be classified further like **GPCRs** belong to the category of transmembrane proteins with 7 transmembrane helices. Such inner class classification is also possible if their properties are understood and analyzed properly to be used as features specific to those inner classes. This is the future approach of our work where we would be developing a GUI Platform that takes a protein sequence as input and then after verifying that it is a membrane protein also predicts its inner class for more specific classification. While this thesis was being written, the work has already been started on the same.

Before these inclusions the work will approach to achieve following modifications

i) Inclusion of Advanced Query Based Web Applet for classification

ii) Remote processing of charge based properties and eliminating need of secondary web server based property calculation

iii) Integration of information from KEGG database so that the work could be used for further network and interactome analysis.

References

1. Tooze, John & Branden, Carl : **Folding & Flexibility, Introduction to Protein Structure, 1999**
2. **"Why Classify Proteins?", Introduction To Protein Classification**
   European Bioinformatics Institute
3. Nelson and Cox : **"The composition and Architecture of Membranes"**
   Lehninger Principles of Biochemistry, 2008
4. Altschul, S.F., Gish, W., Miller, W., Myers, E.W. & Lipman, D.J. (1990) "Basic local alignment search tool (BLAST)." J. Mol. Biol. 215:403-410
5. Henikoff, S; Henikoff, JG (1992). **"Amino acid substitution matrices from protein blocks".** Proceedings of the National Academy of Sciences of the United States of America 89 (22): 10915-9.doi:10.1073/pnas.89.22.10915
6. Sillitoe I, Cuff AL, Dessailly BH, Dawson NL, Furnham N, Lee D, Lees JG, Lewis TE, Studer RA, Rentzsch R, Yeats C, Thornton JM, Orengo CA : **"New functional families (FunFams) in CATH to improve the mapping of conserved functional sites to 3D structures"**, Nucleic Acids Res. 2013 January

7. Murzin A. G., Brenner S. E., Hubbard T., Chothia C. (1995): **"SCOP: A structural classification of proteins database for the investigation of sequences and structures"**, J. Mol. Biol. 247, 536-540.

8. Bhardwaj Nitin, Gerstein Mark & Lu Hui: **Genome Wide Sequence Based Prediction of Peripheral Proteins using a novel semi-supervised learning technique**
   Eighth Asia Pacific Bioinformatics Conference (APBC 2010)
9. Jain K Anil, Mao Jianchang : **"Artificial Neural Networks"**
   IEEE Computer, March 1996
10. REST: http://wiki.python.org/moin/WebServices
11. SOAP: http://wiki.python.org/moin/WebServices
12. PubMed: Online Literature Search: http://www.ncbi.nlm.nih.gov/pubmed
13. Python Programming Language: http://www.python.org/
14. MATLAB: http://www.mathworks.in/products/matlab/
15. Clifford E. Felder, Jaime Prilusky, Israel Silman, and Joel L. Sussman: " **A server and database for dipole moments of proteins"**, *Nucleic Acids Research 2007,* http://bioinfo.weizmann.ac.il/dipol/index.html
16. Mats H.M. Olsson, Chresten R. Søndergard, Michal Rostkowski, and Jan H. Jensen **"PROPKA3: Consistent Treatment of Internal and Surface Residues in Empirical pKa predictions"** Journal of Chemical Theory and Computation, 2011 7 (2), 525-537
17. Hui Li, Andrew D. Robertson, and Jan H. **Jensen "Very Fast Empirical Prediction and Interpretation of Protein pKa Values"** Proteins, 2005, 61, 704-721.

18. Delphine C. Bas, David M. Rogers, and Jan H. Jensen "**Very Fast Prediction and Rationalization of pKa Values for Protein-Ligand Complexes**" Proteins, 2008, 73, 765-783
19. Chresten R. Søndergaard, Mats H.M. Olsson, Michaz Rostkowski, and Jan H. Jensen "**Improved Treatment of Ligands and Coupling Effects in Empirical Calculation and Rationalization of pKa Values**" Journal of Chemical Theory and Computation, 2011 7 (7), 2284-2295
20. Vincent Le Guilloux, Peter Schmidtke and Pierre Tuffery, "**Fpocket: An open source platform for ligand pocket detection**", BMC Bioinformatics, 2009, 10:168
21. Bakan A, Meireles LM, Bahar I: "**ProDy: Protein Dynamics Inferred from Theory and Experiments**", *Bioinformatics* **2011** 27(11):1575-1577.
22. Swiss Prot: http://web.expasy.org/docs/swiss-prot guideline.html
23. K. Hofmann & W. Stoffel (1993), "**TMBase - A database of membrane spanning proteins segments**", Biol. Chem. Hoppe-Seyler **374**,166
24. Jayasinghe, S., Hristova, K., and White, S. H. (2001): "**MPTopo: A database of membrane protein topology**", Protein Science **10**:455-458
25. Schultz, J., Milpetz, F., Bork, P. & Ponting, C.P.: "**SMART, a simple modular architecture research tool: Identification of signaling domains**", PNAS 1998; **95**: 5857-5864
26. Weizhong Li & Adam Godzik : "**Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences**", Bioinformatics, (2006) 22:1658-9
27. C Nick Pace & J. Martin Schultz, "**A helix propensity scale based on experimental studies of peptides and proteins**", Biophysical Journal, Vol 75, July 1998
28. Smith CK, Withka JM, Regan L: "**A thermodynamic scale for the beta sheet forming tendencies of the amino acids**", Biochemistry 1994, 33:5510-5517
29. Subbarao N, "**Graph Based Theoretical Method for Identification of Binding Sites (Clique Detection)**"
30. Gasteiger E, Hoogland C, Gattiker A: "**Protein Identification and Analysis Tools on ExPASY server**", The Proteomics Protocols Handbook, Humana Press, 2005
31. J. Kyte, R.F. Doolittle,: "**A simple method for displaying the hydropathic character of a protein**", J. Mol. Biol. 157 (1982) 105
32. Kazuo Fujiwara, Hiromi Toda and Masamichi Ikeguchi: "**Dependence of α-helical and β-sheet amino acid propensities on the overall protein folds type.**", BMC Structural Biology 2012, 12:18