Please reference this paper as follows:

Pham Nguyen, A.H., and E. Triantaphyllou, (2005), **"Predicting Protein Folding Structues by Means of a New Classification Approach,"** *Proceedings of the ICDM 2005 Workshop on: Temporal Data Mining: Algorithms, Theory and Applications.* Hold in conjunction with the *Fifth IEEE Inter'l Conference on Data Mining (ICDM'05).*

PREDICTING PROTEIN FOLDING STRUCTURES BY MEANS OF A NEW CLASSIFICATION APPROACH

By H.N.A. Pham and E. Triantaphyllou
Department of Computer Science, 164E Coates Hall, Louisiana State University, Baton Rouge, LA, 70803, U.S.A. Email: hpham15@lsu.edu, and trianta@lsu.edu

Abstract

The structure prediction problem for proteins plays an important role in the protein process. This is a notoriously hard problem and been able to achieve good prediction performance with new methods will certainly have an impact in both the computational arena but also in the Bioinformatics field. This paper proposes a novel classification approach using a binary expansion method based on the density concept for homogenous clauses to predict protein folding structures. The successes of this approach are demonstrated on several protein data sets whose structure is partially known.

Keywords: protein folding, homogenous clause, binary expansion.

1. Introduction

Proteins naturally fold into complex 3D globules from their amino acid sequence. There are 20 different types of amino acids labeled with their initials as: A, C, G, T, ... By this presentation, a protein can be thought as a sequence of ACTG… The protein process has at least two distinct problems [6].

- Structure Prediction Problem: the problem determines the 3D structure of a protein from its amino acid sequence.
- Pathway Prediction Problem: the problem determines the time-ordered sequence of folding events from a given protein amino acid sequence and its 3D structure.

Both problems have received attentions from many researchers. The ability to predict protein folding structure, however, can greatly enhance structure prediction methods. The structure prediction problem or Protein folding problem can offer significant clues about the function of a protein which cannot be found via experimental methods quickly or easily. Once a protein is identified, it can be applied for increasing important players in human disease, limiting the ability to effectively design new proteins, and other applications.

In finding the 3D structure, a protein classified into four structural classes: all-α, all-α, α/β, and α+β introduced by Levitt and Chothia [10] according to their secondary structure composition. Hence this problem can be stated as a four-class classification problem. Once the structural class of a protein is known, it can be used to reduce the search space of the structure prediction problem that most of the structure alternatives will be eliminated, and the structure prediction task will become easier and faster.

There have been many theoretical and practical developments in the last ten years in this problem. Many studies have resulted in classification and prediction systems that are highly accurate or they

are not so accurate. Chou [1] assigned a protein into one of the four structural classes by using Amino Acid Composition (AAC) of a protein and Mahalanobis distance. Wang et al. [2] tried to improve Chou's work using the same data set, without success. Ding and Dubchak [3] used Neural Networks (NNs) and Support Vector Machines (SVMs) on classifying proteins into one of 27 fold classes, which are subclasses of the structural classes. Tan and coworkers [4] also worked on the fold classification problem (for 27 fold classes), using a new ensemble learning method. More recently, Zerrin Isik et al [5] used SVMs for Amino Acid Composition (AAC) of the protein as the base for classification.

A growing belief is that the root of not so accuracy is the overfitting and overgeneralization behavior of such systems. Roughly speaking, overfitting means that the extracted model describes the behavior of known the training data set very well but does poorly on new data points. Overgeneralization occurs when the system uses the available data and then attempts to analyze vast amounts of data that has not seen yet. Both problems may cause poor performance. This is a situation studied in statistics and, to some extend, with some of the data mining methods such as decision trees, NNs, and SVMs.

This paper aims at presenting a useful approach of overfitting and overgeneralization for the purpose of controlling these two key properties. By doing so, it is hoped that the classification / prediction accuracy of the extracted system will be very high or at least as high as it can be achieved with the available training data. In particular, the approach uses the density concept of a homogenous clause described in Section 2.1, and a binary expansion approach in Section 3 to classify the structure of proteins. In section 4, the successes of this approach are demonstrated and assessed on several protein data sets whose structure is partially known in Ding and Dubchak [3], and Zerrin Isik et al [5]. Basically, all of classification assessments in the paper use the average accuracy introduced by Rost & Scander, 1993 [12], Baldi et al, 2000 [11].

2. Preliminaries

2.1 Multi-class prediction method

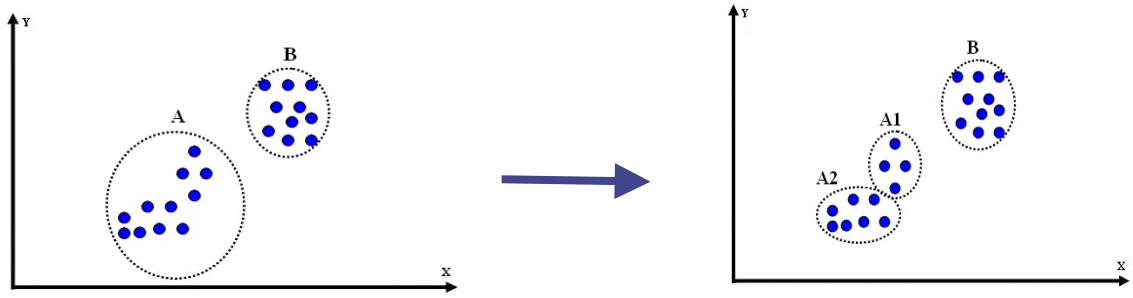
Most of classification methods dealing with two-class problems are often accurate and efficient, for example SVMs or NNs. When dealing with more classes, they, however, usually reduce accuracy and efficiency. This section presents a method, One-vs-Others, that utilizes two-class classification methods as the basic building block for larger number of classes. This is a simple and effective method introduced by Dubchak et al 1999 [8], Brown et al 2000, [9].

In this process, suppose there are K classes in the problem. K classes, firstly, are partitioned into a two-class problem: one class consists of proteins in one "true" class, and the "others" includes all other classes. A two-class classification method then is used to train for this two-class problem. The process then partitions the K classes into another two-class problem: one "true" class consists of another original class and the "others" class is the rest. Another two-class problem is trained. This process is repeated for each of the K classes, and this leads to K two-way trained classifiers.

In the testing process, the system uses testing queries for each of the K two-way classifiers and determines the maximum of K scores from the K classifiers. The maximum score is considered as a classifier for the two-class problem: one class consists of proteins in one "true" class, and the

"others" includes all other classes. All of steps of this process are repeated for the K -1 remaining classes.

2.2 Homogenous clause (HC) and its density



Homogeneous is an adjective that has several meanings. In biology homogeneous has a meaning similar to its meaning in mathematics. In physical chemistry, homogeneous describes a single-phase system as opposed to heterogeneous where more than one thermodynamically distinct phase co-exists. Homogenous (without the second e) has a similar meaning of being the same throughout, and is perhaps more common in everyday speech.

In this paper, homogenous has a meaning similar to the physical chemistry field. It describes a steadiness for distinct phases co-exist. A homogenous clause covers a set of examples of a given class (i.e., positive or negative) and unclassified examples uniformly. That is, within the clause there are no subdivisions with unequal concentrations of classified (i.e., either positive or negative) and unclassified examples.

For instance, Figure 1 depicts a situation defined on two continuous variables X and Y. In the same figure clause A is a non-homogenous clause while clause B is a more homogenous one. Please note that in these two clauses only the classified data are shown as small circles. The unclassified data are the rest of the points of the X-Y plane.

Clause A, however, is replaced by two more homogenous clauses denoted as A_1 and A_2. Then the areas covered by the two new Clauses A1 and A2 are more homogenous than the area covered by the original Clause A.

From the above example, a judgment can be applied for studying a new classification approach. When a classification algorithm that infers a set of classification rules from training examples is applied, these rules may or may not be affected by homogenous clauses and their density. In turn, this may affect the accuracy in correctly classifying new data points. For instance, the clause labeled as "Clause A" in Figure 1 is not as homogenous as Clause B in the same figure. Thus, it is possible that unseen examples covered by clause A are erroneously assumed to be in the same class as the "solid" examples covered by the same clause. In particular, this is most likely to occur in regions of Clause A that are not populated by "solid" points. Such a region, for instance, exists in the upper left corner of Clause A (see also Figure 1). Another similar region is the lower part of the same clause.

On the other hand, Clause B is more homogenous than Clause A. Thus, it is more likely that the

unclassified examples covered by Clause B are more accurately assumed to be in the same class as the "solid" examples covered by the same clause. The above simple observations lead one to surmise that the accuracy of the classification rules can be increased if the derived clauses are, somehow, more compact and homogenous.

Intuitionally, another factor also affects the accuracy of the classification rules. That is the density of a homogenous clause. For example, the unclassified examples covered by Clause B in Figure 1 are more assumed to be in the same class as the "solid" examples covered by B than Clause A1 or A2. Particularly, Clause B may be expanded wider than Clause A1 and A2 since B's density is more concentrated than other clauses. This section ends with a simple definition for the density of a homogenous clause that can be the number of examples of a given class in a unit volume. This factor decides how much that homogenous clause can be expanded.

2.3 Accuracy measure

In two-class problems, assessing the accuracy involves calculating true positive rates and false positive rates. In multi-class problems, particularly converted through the One-vs-Others method, this assessment, however, has to be extended suitable to adapt for more than two classes. A simple standard assessment, Q, was introduced by Rost & Scander, 1993 [12], and Baldi et al, 2000 [11]. Suppose there are $N = n_1 + n_2 + \ldots + n_k$ test proteins where n_i are number of examples in class i^{th}. Let $C = c_1 + c_2 + .. + c_k$ be the total of proteins that are correctly recognized, where c_i is the number of examples that are correctly recognized in class i^{th}. Therefore the accuracy for class i^{th} is $Q_i=c_i/n_i$ and the overall accuracy is Q=C/N.

An individual class contributes to the overall accuracy in proportion to the number proteins in its class. Hence each of Q_i relates to the overall Q by a weight $w_i=n_i/N$. The overall accuracy is:

$$Q = \sum_{i=1}^{k} w_i Q_i$$

If a protein is sequentially tested for all four classes and one of them is correct then c = ¼. Therefore in general, c_i can be a real number.

3 Binary Expansion Approach (BEA)

Input: positive and negative examples

Output: a suitable classification

Step 1: Find positive and negative clauses by using k-means clustering-based with the Euclidean distance

Step 2: Find positive and negative homogenous clauses from positive and negative clauses respectively

Step 3: Sort positive and negative homogenous clauses on densities

Step 4: FOR each homogenous clause C DO

+ Find C's density, say D

+ Expand C by using D

Figure 2: The Binary Expansion Algorithm

This section outlines the binary expansion approach to predict the folding structures of a protein using the idea of "expanding homogenous clauses". Essentially, this approach is a two-class classification method, and the protein folding problem then uses the approach through One-vs-Others method. Suppose each of proteins in data sets is considered as a vector in n dimensions. The Euclidean distance is used for computing distances between proteins. In the training phase, the intuition behind of this approach is to find positive and negative homogenous clauses, and then to expand each of homogenous clauses, considered as spheres, until the area of this homogenous clause overcomes a threshold based on that homogenous clause's density. Then the testing

phase uses expanded positive and negative homogenous clauses to test structures for new proteins. A detailed description of this approach is in Figure 2.

At step 1, K-Means Training starts with generating the k clause centers randomly and goes ahead by fitting the data points in those clauses with the Euclidean distance. This process is repeated until all points are identified in clauses. If the specified clauses of a given class are close together, then they can be joined in a unique clause. Remaining points that do not belong to any clause are created in new clauses with unique point.

The k-means clustering-based method is also used for finding positive and negative homogenous clauses from positive and negative clauses. Only two differences are while fitting the data points in those clauses, the process is stopped when it hits into the border of the positive or negative homogenous clause. And the distance used in the process is the minimum distance between any two points of a given class in the training set. The sorting for positive and negative homogenous clauses decides the order that homogenous clauses are expanded.

Step 4 is the main part of this algorithm. Suppose homogenous clauses are sorted on densities. The expanding process starts with a homogenous clause that has the highest density and so on. For the current homogenous clause considered as a sphere, a new homogenous clause is expanded by:

Where R: Expanded HC's radius

R1: HC's radius

R2: Envelope's radius

Envelope's radius is a double radius of the current homogenous clause. This formula quotes that the density of a homogenous clause decides how

$$R = R_1 + \frac{R_2 - R_1}{2} \times \frac{1}{D}$$

much that clause is expanded. The expanding process stops whether any point differing the class name occurs in the expanded region, the area of the expanded region is greater than a multiple of D, or the current homogenous clause's radius is greater than envelope's radius. The overall approach in 2D is presented in Figure 3

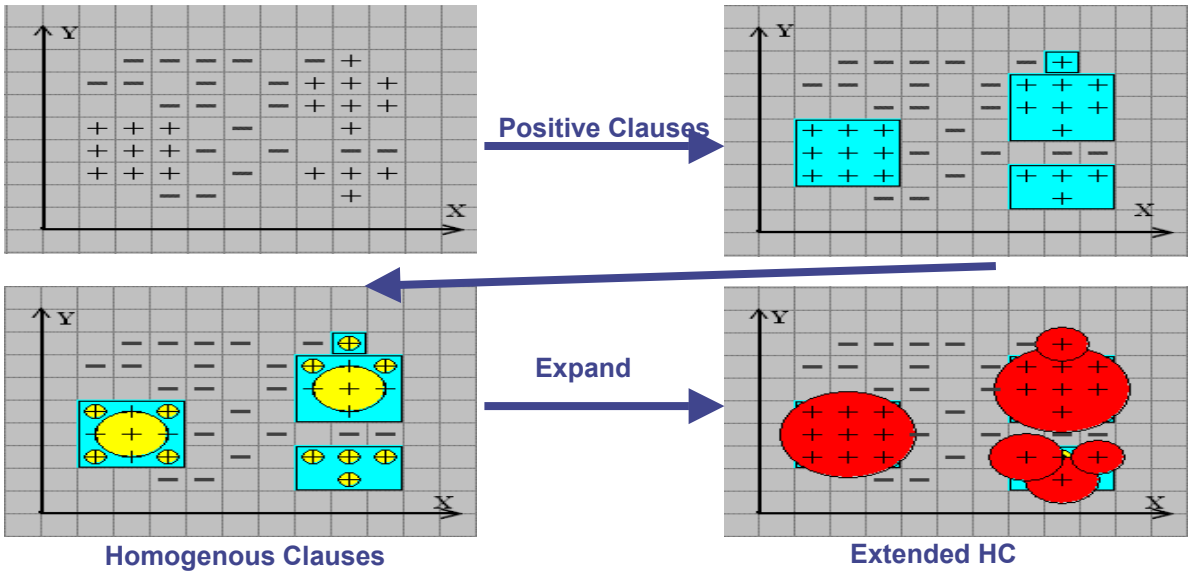


Figure 3: The overall approach for positive examples in 2D

4 Results

This section presents test bed applications for our method with the independent testing method, and assessments based on the standard accuracy measure introduced in Section 2.3.We firstly have applied the approach to data sets studied by Chih. C.Chang and Chih.J.Lin at www.csie.ntu.edu.tw/~cjlin/papers/guide/data/. This data set consists of three small data sets whose features are described in Table 1. Another data set from Chih. C.Chang and Chih.J.Lin at www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary has assessed as in Table 3 and 4. Results obtained from C.J.Lin's experiments [13] and this approach are in Table 1 and 2 respectively.

Training set	Testing set	#atts*	C.J.Lin's SVMs
Train_1 (3089 examples)	Test_1 (4000 examples)	4	96.9%
Train_2 (391 examples)	Train_2 (391 examples) Cross validation	20	85.2%
Train_3 (1243 examples)	Test_3 (41 examples)	21	87.8%

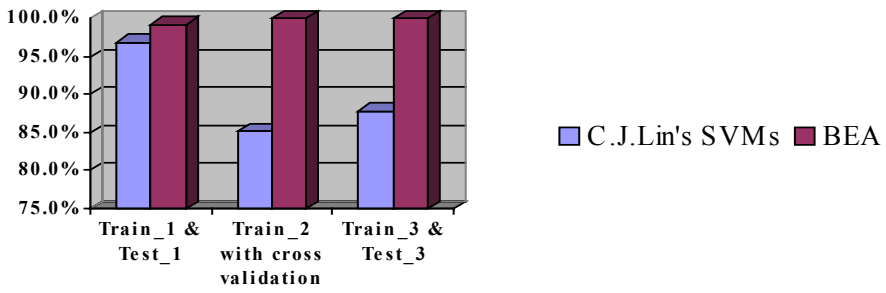
Table 1: results of C.J.Lin's SVM
Source of the data set www.csie.ntu.edu.tw/~cjlin/papers/guide/data/, *Atts: Attributes

Training set	Testing set	#Fail Positive	#Fail Negative	Q
Train_1	Test_1	9 (2000 positive examples)	22 (2000 negative examples)	99.25%
Train_2	Train_2	0	0	100%
Train_3	Test_3	0 (41 positive examples)	0 (0 negative examples)	100%

Table 2: results of BEA
Source of the data set www.csie.ntu.edu.tw/~cjlin/papers/guide/data/

The comparison in Figure 4 shows that BEA provides around 15.5% improvement in classification accuracy as the SVMs method. We can explain this improvement throughout the essential of the SVMs method. Since the SVMs method uses hyperplans to classify training points, it creates a wide undecided region around seen points, and this leads overgeneralization. In contract, BEA starts with the homogeneity of points and the density of homogenous clauses for expanding seen regions. It may create expanded regions that can satisfy both of fitting and generalizing properties. So, this approach provides more classification accuracy than other methods. Table 3 and 4 are other test beds for the approach. These tables show that, BEA obtains better classification rates using more training data, which is as expected

.



	Data	#Atts	#Exps*	Q
Train	W1a	300	2477	
Test	W2a	300	3470	85,97 %
	w3a	300	4912	85,40
	w4a	300	7366	85,08
	w5a	300	9888	84,64
	w6a	300	17188	84,18

	Data	#Exps	Q
Train	w4a	7366	
Test	w1a	2477	85,79%
	w2a	3470	86,57
	w3a	4912	86,16
	w5a	9888	85,41
	w6a	17188	84,83

Table 3: results of BEA
Source of the data set www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary, * Exps: Examples

	Data	#Atts	#Exps	Q
Train	a3a	122	3185	
Test	a4a	122	4781	90,17%
	a5a	122	6414	86,47
	a6a	122	11220	82,17
	a7a	122	16100	79,99

	Data	#Exps	Q
Train	a7a	16100	
Test	a3a	3185	94,98%
	a4a	4781	94,92
	a5a	6414	94,92
	a6a	11220	96,95

Table 4: results of BEA
Source of the data set www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary

Data types	Symbol	#Atts	# Training exps	# Testing exps
A.A.Composition	C	21	605	385
Secondary struc.	S	22	605	385
Polarity	P	22	605	385
Polarizability	Po	22	605	385
Hydrophobicity	H	22	605	385
Volume	V	22	605	385

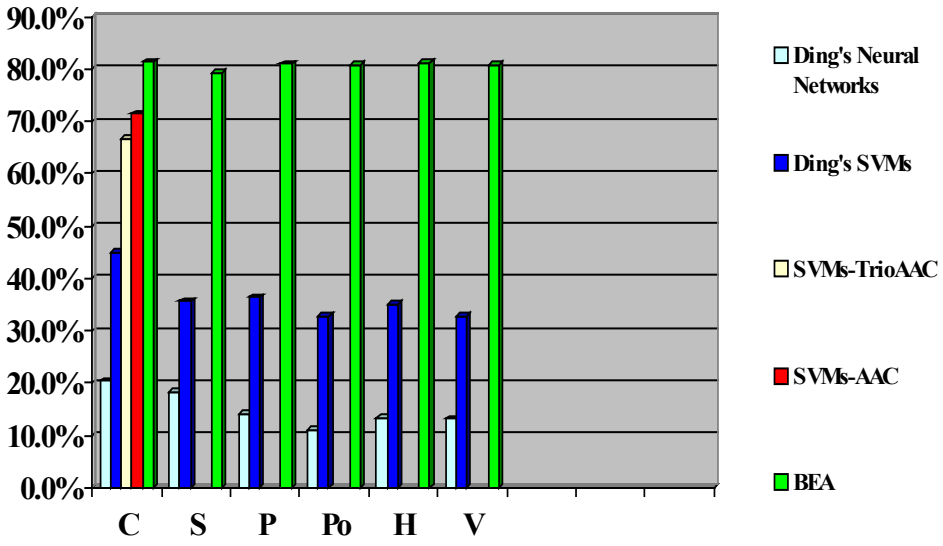
Table 5: Six parameter datasets extracted from protein sequence
Source of the data set http://www.nersc.gov/~cding/protein/

For the protein folding problem, the data set we used for training and testing was selected from the database built by Ding and Dubchak [3]. This database has seven or more proteins and presents all major structure classes: all-α, all-α, α/β, and α+β with 27 most populated folds [7]. Table 5 is a description of this database. Results Obtained Results from Ding and Dubchak's method using SVMs and NNs, and Zerrin's method using $SVMs^{AAC}$ and $SVMs^{trioAAC}$ for the same dataset are in Table 6 respectively. Please note that Zerrin's paper only assessed for the data type of Amino Acid Compostion.

Data types	Q1	Q2	Q3	Q4
Composition	44.9%	20.5%	71.44%	66.66%
Secondary struc.	35.6	18.3		
Hydrophobicity	36.5	14.2		
Polarity	32.9	11.1		
Volume	35.0	13.4		
Polarizability	32.9	13.2		

Table 6: Results of and Dubchak's paper [3], and Zerrin's paper [5]
Q1: Accuracy of the SVMs Independent Test method in Ding's assessment
Q2: Accuracy of the Neural Networks Independent Test method in Ding's assessment
Q3: Accuracy of the $SVMs^{AAC}$ method in Zerrin's assessment
Q4: Accuracy of the $SVMs^{trio\ AAC}$ method in Zerrin's assessment

Data types	all-α	all-β	α/β	α/β	Q5
A.A.Composition	87.27%	74.81%	71.43%	91.95%	81.37%
Secondary struc.	87.23	72.21	66.75	91.17	79.34
Hydrophobicity	86.75	74.55	71.17	91.69	81.04
Polarity	87.27	73.51	70.13	91.95	80.72
Volume	87.01	74.29	71.43	91.95	81.17
Polarizability	86.75	74.29	70.13	91.95	80.78
Table 7: results of BEA					



Obtained results from BEA for the same dataset are in Table 7. The comparison of the Qs in Figure 5 shows that at the data type of Amino Acid Composition, BEA provides around 10% improvement in classification accuracy as the $SVMs^{AAC}$ method, 43% improvement as Ding's SVM at the data type of Secondary Structure, 44% improvement as Ding's SVM at the data type of Hydrophobicity, 48% improvement as Ding's SVM at the data type of Polarity, 46% improvement as Ding's SVM at the data type of Volume, and 47% improvement as Ding's SVM at the data type of Polarizability.

5 Conclusion

This paper has described the intensive novel machine learning method to a notoriously hard problem, structure prediction problem for proteins. The comparison of experiments shows that BEA provides 10-48% improvement in classification accuracy. We have also obtained better classification rates using more training data, which is as expected.

References

[1]. Chou, K.C.: A novel approach to predicting protein structural classes in a (20-1)-d amino acid composition space. Proteins 21 (1995) 319–344

[2]. Wang, Z.X., Yuan, Z.: How good is prediction of protein structural class by the component-coupled method. Proteins 38 (2000) 165–175

[3]. Ding, C.H., Dubchak, I.: Multi-class protein fold recognition using support vector machines and neural networks. Bioinformatics 17 (2001) 349–358

[4]. Tan, A.C., Gilbert, D., Deville, Y.: Multi-class protein fold classification using a new ensemble machine learning approach. Genome Informatics 14 (2003) 206–217

[5]. Zerrin Isik et al: Protein Structural Class Determination Using Support Vector Machines. Lecture Notes in Computer Science-ISCIS (2004), vol: 3280, pp. 82.

[6]. Jason T.L. Wang et al: Data Mining in Bioinformatics (2005), Chapter 7, Predicting Protein Folding Pathway, 127-141.

[7]. Hobohm, U., Scharf et al: Selection of a representative set of structures from the Brookhaven Protein Bank. Protein Sci., 1, (1992), 409-417.

[8]. Dubchack et al: Recognition of a protein fold in the context of Structure Classification of Proteins (SCOP) classification. Protein 35 (1999), 401-7.

[9]. Brown et al: Knowledge-based Analysis of Microarray Gene Expression Data. Support Vector Machines Proc. Natl Acad Sci., 97 (2000), 262-267.

[10]. Levitt, M., Chothia, C.: Structural patterns in globular proteins. Nature 261 (1976) 552–558

[11]. Baldi,P. et al: Assessing the accuracy of prediction algorithms for classification: an overview. Bioinformatics, 16 (2000), 412-424.

[12]. Rost, B. and Sander, C.: Prediction of protein secondary structure at better than 70% accuracy. J.Mol. Bio., 232 (1993), 584-599.

[13]. C.-W. Hsu, C.-C. Chang, C.-J. Lin: A practical guide to support vector classification (July 2003).