Protein Fold Classi cation using Kohonen's Self-Organizing Map

Ozlem Ozbudak and Zümray Dokur

Istanbul Technical University, Electronics and Communication Engineering Deptartment, 34469 Maslak, Istanbul, Turkey
{ozbudak,dokur}@itu.edu.tr

Abstract. Protein fold classification is an important problem in bioinformatics and a challenging task for machine-learning algorithms. In this paper we present a solution which classifies protein folds using Kohonen's Self-Organizing Map (SOM) and a comparison between few approaches for protein fold classification. We use SOM, Fisher Linear Discriminant Analysis (FLD), K-Nearest Neighbour (KNN), Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP) methods to classify three SCOP folds with six features (amino acid composition, predicted secondary structure, hydrophobicity, normalized van der Waals volume, polarity and polarizability). This paper has a novelty in the way of applying SOM to these six features, and also portrays the capabilities of SOM among the other methods in protein fold classification. The methods are tested on 120 proteins by applying 10-fold cross-validation technique and 93.33% classification performance is obtained with SOM.

Keywords: protein fold classification, protein fold recognition, self organizing map, neural networks, k-fold cross-validation

1 Introduction

Proteins are large biological macromolecules which organize essential parts of living organisms to control all of their vital functionalities. Protein functions are related to protein chemical reactions with their surrounding and other proteins. Also protein functions depend on its shape and three-dimensional (3D) structure [1]. There are currently 97,362 (at 01/02/2014) experimentally determined 3D structures of protein deposited in the Protein Data Bank (PDB) [2] with an increment of about 700 new molecules for month. However, there are a lot of similar structures (not identical) in this protein set. So protein structure comparison, fold recognition and fold classification came into question in computational biology. In the literature there are di erent types of works about proteins. The basic works are about prediction of protein secondary structures [3, 4], prediction of protein structural classes [5, 6] and classification of protein folds [7 11]. Protein structure predictions represent a key step in studying and understanding protein functions. The fact that protein function does not only depend on protein sequence but also the shape and structure induces the importance of

protein structure identification. Given a protein sequence, the secondary structure prediction problem is to predict whether each amino acid is in alpha-helix, beta-strand or coil [3]. According to convention a protein could be classified into one of four structural classes based on its secondary structure components, all-α, all-β, α/β and $\alpha+\beta$ [12]. Prediction of protein structural classes is to decide that the new query protein belongs to which of four structural classes (all-α, all-β, α/β or $\alpha+\beta$). Structural Classification of Proteins (SCOP) [13] provides a detailed and comprehensive description of the structural and evolutionary relationships among all proteins whose structures are known [7]. According to the SCOP four structural classes are divided into folds. Protein fold classification problem is to determine that the query protein belongs to which fold. [7 10] deal with the classification of 27 folds belonging to four structural classes and [11] tries to classify only three of 27 folds belonging to α/β structural class. [11] uses Self-Organizing Map for Structured Data (SOM-SD) as classifier and uses directions of secondary structures in proteins as features. As a result [11] classifies the three folds with an 86.42% accuracy rate. In this paper we use Kohonen's SOM as classifier and amino acid composition, predicted secondary structure, hydrophobicity, normalized van der Waals volume, polarity and polarizability as features. We classify the same three folds as in [11] with an 93.33% accuracy rate.

The remainder of the paper is as follows. In Section 2 the dataset and features are expressed. In Section 3 the classifiers are told and the algorithm related to SOM is presented. In Section 4 experiments and performances are shown and Section 5 concludes the paper.

2 Dataset and Features

The dataset used in this paper was taken from [8]. The original training dataset and test dataset contain 313 and 385 proteins, respectively and there are 27 folds in this dataset. In this paper we consider only three folds used also in [11] for classification. These folds are namely Flavodoxin-like , RibonucleaseH-likemotif and TIMbeta/alpha-barrel belonging to alpha and beta proteins class (α/β). These folds contain 74, 24 and 22 proteins respectively. Hence, in this paper we use totally 120 proteins for classification and we try to solve three-class protein fold classification problem.

To deal with the problem, Ding *et al.* [8] extracted the following six features from protein sequences: amino acid composition, predicted secondary structure, hydrophobicity, normalized van der Waals volume, polarity and polarizability. Of the above six features, only the amino acid composition contains 20 components, with each representing the occurrence frequency of one of the 20 native amino acids in a given protein. For the remaining five features, each contains 3+3+5×3=21 components [9].

The composition vector is computed directly from amino acid sequence. Given that the 20 amino acids which are ordered alphabetically (A,C,D,E,F,G,H,I K,L,M,N,P,Q,R,S,T,V,W,Y) are represented as AA_1, AA_2,...,AA_{19} and AA_{20},

and the number of occurrences of AA_i in the entire sequence is denoted as n_i, the composition vector is defined as:

$$\frac{n_1}{L}, \frac{n_2}{L}, ..., \frac{n_{19}}{L}, \frac{n_{20}}{L} \tag{1}$$

where L is the length of the sequence [10].

As mentioned above the predicted secondary structure is divided into three classes which are helix, strand and coil and also for the other four attributes, those of hydrophobicity, normalized van der Waals volume, polarity and polarizability, the 20 amino acids are divided into three groups according to the magnitudes of their numerical values. These three groups are shown in Table 1. Three descriptors, composition (C), transition (T) and distribution (D) are calculated for a given attribute to describe the global percent composition of each of the three groups in a protein, the percent frequencies with which the attribute change its index along the entire length of the protein, and the distribution pattern of the attribute along the sequence, respectively. The complete parameter vector for these five attributes contains 3(C)+3(T)+5×3(D)=21 scalar components. Consequently the feature vector for a protein includes 20+21×5=125 components [7].

Table 1. Amino acid attributes and the division of the amino acids into three groups for each attribute [7].

Property	Group 1	Group 2	Group 3
Hydrophobicity	Polar R,K,E,D,Q,N	Neutral G,A,S,T,P,H,Y	Hydrophobic C,V,L,I,M,F,W
Norm. van der Waals vol.	0-2.78 G,A,S,C,T,P,D	2.95-4.0 N,V,E,Q,I,L	4.43-8.08 M,H,K,F,R,Y,W
Polarity	4.9-6.2 L,I,F,W,C,M,V,Y	8.0-9.2 P,A,T,G,S	10.4-13.0 H,Q,R,K,N,E,D
Polarizability	0-0.108 G,A,S,D,T	0.128-0.186 C,P,N,V,E,Q,I,L	0.219-0.409 K,M,H,F,R,Y,W

In this paper the dataset is tested for the protein fold classification problem by using five different methods. Among these methods SOM has a good performance in terms of accuracy rate and computation time.

3 Classifiers

3.1 Self-Organizing Map

Kohonen's Self-Organizing Map, which is developed by Tuevo Kohonen in 1982 [14], is a type of neural network which uses unsupervised learning method. In the

Kohonen's network, with the training algorithm the class distribution in the n-dimensional space is carried to two-dimensional space. There are two layers in the Kohonen's network (see Fig. 1). The distance between the jth node ($\boldsymbol{w}_j$) in the output layer and the input vector $\boldsymbol{x}$ is calculated as follows:

$$D_j = \sum_{i=1}^{n} (x_i - w_{ji}(k))^2 \ . \tag{2}$$

Training of Kohonen's SOM network is as follows:
Step 1. Before starting the learning; number of output nodes, number of iterations and neighbourhood function are determined. Initial weights of the output nodes are set to random values within [0-1] range.
Step 2. A vector randomly chosen from the training set is given to the network as input.
Step 3. The distances between the input vector and network nodes are calculated using (2). Here, x_i and w_{ji} represent the ith element of the input vector and ith weight of the jth output node, respectively ($i = 1, 2, ...n$).
Step 4. jth output node having the minimum distance is found.
Step 5. The weights of the jth output node and its neighbours are updated using the expression below.

$$w_{ji}(k+1) = w_{ji}(k) + \eta(k).(x_i - w_{ji}(k)) \ . \tag{3}$$

$\eta(k)$ is learning rate and k is iteration number.
Step 6. Number of iterations is reduced. If the number of iterations is not equal to 0, Step 2 and other steps are repeated. If the number of iterations is 0, the learning algorithm is terminated.

After completing the training, class labels are assigned to the output nodes. To accomplish the labeling, each vector in the training set is fed to the trained network and the winner node at the output layer whose weight vector lies closest to the input vector is determined. The output nodes are associated with training data classes according to majority voting, i.e, the training data class that is assigned most frequently to an output node becomes its label.

In this study the SOM was trained on 7×7, 8×8 and 9×9 neurons with a neighbourhood spread $\sigma = 1$, considering learning rate $\eta = 0.5$ and di erent iterations (500, 1000 and 1500). 10-fold cross-validation was applied to the dataset. The performance was given as the average of 10 set's performance.

3.2 Fisher Linear Discriminant Analysis

There are many techniques for classification of data and FLD is one of them. FLD is also used for dimension reduction. The aim of the FLD is to find the optimal linear projection for classification. Thus, FLD is the projection on a line in the direction which maximizes the ratio of between-class scatter to within-class scatter. FLD uses supervised learning so the class labels of samples are

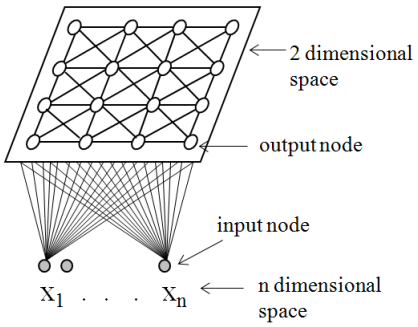

Fig. 1. Input and output space related to Kohonen's SOM network.

important. It encodes discriminating information in a linearly separable space using bases that are not necessarily orthogonal [15]. Applying FLD to multi-class problem, in the case of c classes the dimension of data is reduced to $c-1$. In this paper we try to solve three-class problem, so 125 dimensional data are reduced to 2 dimensional data and then FLD classifier is processed.

3.3 K-Nearest Neighbour Classifier

This method generally is used for classification problems and it is easy to apply. In this method, examples are classified based on the class of their nearest neighbours. It is often useful to take more than one neighbour into account so the technique is more commonly referred to as K-Nearest Neighbour Classification where K nearest neighbours are used in determining the class. In this method there are some important parameters. These are K, distance function and cross-validation method. In this study using 10-fold cross-validation 10 datasets are formed and tested with KNN classifier. As the distance function Euclidean distance is used. For this method different numbers of nearest neighbours were tested, but the best performance was obtained for $K = 1$.

3.4 Support Vector Machine

SVM is a machine learning technique that is based on the statistical learning theory developed by Vapnik [16]. It is a group of supervised learning methods that can be applied to classification or regression. The SVM is a binary classification algorithm and is designed to maximize the margin to separate two classes. The main task of SVMs is in fact to find a hyperplane and with this attribute, it is suitable for the task of classifying protein folds. SVM is widely used in bioinformatics and is able to classify data in the field of protein fold classification.

In this paper applying SVM, multi-class problem is reduced to two-class problem. For testing one-against-all approach is used. According to this three binary classifiers are constructed and each of them separates one class from all the rest.

3.5 Multi-Layer Perceptron

The Multi-Layer Perceptron is an example of an artificial neural network that is used extensively for the solution of a number of different problems, including pattern recognition and interpolation. It is a development of the Perceptron neural network model, that was originally developed in the early 1960s but found to have serious limitations [17]. In that network there are input layer, hidden layer(s) and output layer. In this study two hidden layers are added to the network and the classification is performed with 10-fold cross-validation technique.

4 Experiments and Performances

In this work 125 dimensional 120 proteins belonging to three different folds were used as the dataset. 10-fold cross-validation is implemented for all used methods to classify protein folds in SCOP. By using 10-fold cross-validation, the datasets are partitioned into 10 sets having 12 samples in each. Among the 10 sets, one of them is assigned as testing data to validate the data and the rest are used as training data. The process of cross-validation is repeated 10 times, where each of the 10 sets is used once as the validation model. So, each time 108 protein and 12 protein were used for training data and test data, respectively. After the training and test processes the accuracy rate is calculated as follow:

$$Accuracy\ Rate = 100 \times \frac{True\ positives}{All\ samples} \tag{4}$$

Performance is calculated as the average of 10 set's performance and reported in terms of accuracy rate and computation time (the given computation times are related to a desktop computer with a processor Intel Core 2 Duo E7500, 2.94 GHz, 2 GB RAM).

Firstly, SOM is used to classify the three folds in SCOP. Here, in order to see the efficiency of the SOM, the data is tested for different number of nodes and different number of iterations. Test results in terms of accuracy rate and computation time are shown in Table 2

Table 2. Performance of SOM for different values of iterations and nodes.

	Test Set								
Number of iterations	500			1000			1500		
Number of nodes	7×7	8×8	9×9	7×7	8×8	9×9	7×7	8×8	9×9
Accuracy Rate	86.67	90.00	**93.33**	90.83	90.83	91.67	87.50	90.00	91.67
Computation Time (msec)	77.5	84.4	94.5	133.2	147.9	163.8	196.7	210.5	231.9

As seen from Table 2 the best classification performance is obtained by the 9×9 node topology and with 500 iterations. Also in this configuration computation time is very low. So, this configuration will be used in comparison with other classifiers.

Secondly, other four classification methods (FLD, KNN, SVM, MLP) were used to classify the protein folds and their results are compared to the above SOM's result. The comparisons in terms of classifier performance and computation time are shown in Table 3.

Table 3. Performances related to five methods in terms of accuracy rate and computation time.

Classifiers	Accuracy Rate (%)	Computation Time (sec)
SOM	**93.33**	0.0945
FLD	73.33	0.2623
KNN	79.16	0.0032
SVM	77.50	0.6624
MLP	88.33	3.5833

As seen from Table 3 SOM has the best classification performance among the other classifiers and the computation time is very low.

Lastly, we compare our results with the results in [11]. Both works basically use SOM but apply it in di erent ways. [11] uses SOM-SD but we use classical SOM. The di erence between these methods are used features' types. We use six features and the dimension of the data is fixed (125 dimensional data) but [11] uses a new data type called Protein Gaussian Image (PGI) which includes variable number of feature dimensions. In that work features are directions of the secondary structures in the protein so the dimension of the data in [11] is variable, because the number of secondary structures in the proteins are variable. So these two methods are very di erent from each other. Comparison results related to these methods are shown in Table 4

Table 4. Comparison results in terms of number of nodes, number of used proteins and accuracy rate for the proposed SOM and SOM-SD classifiers.

Methods	Number of nodes	Number of used proteins in tests	Accuracy Rate (%)
SOM	9×9	120	**93.33**
SOM-SD [11]	200×200	45	86.42

According to Table 4 SOM has better classification performance with 9×9 neuron topology.

5 Conclusion

Proteins are very important macromolecules because they form much of the functional and structural machinery in every cell in all organisms. Function of the protein is determined by its spatial structure so it is important to learn protein folds using protein fold classification methods.

This paper proposes a new solution to three-class protein fold classification problem using SOM. In this work 120 proteins belonging to Flavodoxin-like , RibonucleaseH-likemotif and TIMbeta/alpha-barrel folds from the α/β structural class are used as the dataset. Each protein in the dataset is represented by 125 dimensional vector formed by six features: amino acid composition, predicted secondary structures, hydrophobicity, normalized van der Waals volume, polarity and polarizability. Five di erent methods are used to classify the protein folds. To calculate the performances of the classifiers 10-fold cross-validation and relative error rate are used. Test results show that SOM has a good performance in terms of accuracy rate and computation time for protein fold classification.

In future works 27-class protein fold classification problem with SOM can be pursued. Moreover, the dimension of the dataset, in this paper 125, can be reduced and the classifier can be tested with low-dimensional data for high performance and low computation time. For better classification performances, new network structures can be searched.

References

1. Hashemi, H.B., Shakery, A., Naeini, M.P.: Protein Fold Pattern Recognition Using Bayesian Ensemble of RBF Neural Networks. International Conference of Soft Computing and Pattern Recognition, 2009. SOCPAR'09, pp. 436 441 (2009)
2. Protein Data Bank, `http://www.rcsb.org/`
3. Fai, C.Y., Hassan, R., Mohamed, S.: Protein Secondary Structure Prediction using Optimal Local Protein Structure and Support Vector Machine. International Journal of Bio-Science and Bio-Technology, vol.4, no.2, pp. 35-43 (2012)
4. Liang, L.: Predicting the Secondary Structure of Proteins using New Ways of Classification. 4th International Conference on Intelligent Human-Machine Systems and Cybernatics, pp. 212-215 (2012)
5. Sun, X.D., Huang, R.B.: Prediction of Protein Structural Classes using Support Vector Machine. Amino Acids, vol.30 pp. 469-475 (2006)
6. Zhang, C.T., Chou, K.C.: An Optimization Approach to Predicting Protein Structural Class from Amino Acid Composition. Protein Science, Cambridge University Press, pp. 401-408 (1992)
7. Dubchak, I., Muchnik, I., Dralyuk, I., Kim, S.H.: Recognition of a Protein Fold in the Context of the Scop Classification. Proteins: Structure, Function and Bioinformatics, vol.35, no.4, pp. 401-407 (1999)
8. Ding, C.H.Q., Dubchak, I.: Multi-class Protein Fold Recognition using Support Vector Machines and Neural Networks. Bioinformatics, vol.17, no.4, pp. 349-358 (2001)
9. Shen, H.B., Chou, K.C.: Ensemble Classifier for Protein Fold Pattern Recognition. Bioinformatics, vol.22, no.14 pp. 1717-1722 (2006)

10. Sun, X.D., Huang, R.B.: PFRES: Protein Fold Classification by using Evolutionary Information and Predicted Secondary Structure. Bioinformatics, vol.23, no.21, pp. 2843-1850 (2007)
11. Cantoni, V., Ferone, A., Ozbudak, O., Petrosino, A.: Protein Structural Block Representation and Search through Unsupervised NN. 22nd International Conference on Artificial Neural Networks, ICANN'12, Lausanne, SWITZERLAND, vol.7553, pp. 515-522, (2012), Springer, ISBN: 9783642332661.
12. Levitt, M., Chothia, C.: Structural Patterns in Globular Proteins. Nature, vol.27, pp. 254-256 (1976)
13. Murzin, A.G., Brenner, S.E., Hubbard, T., Chothia, C.: SCOP: A Structural Classification of Proteins Database for the Investigation of Sequences and Structures. J. Mol. Biol., vol.247, pp. 536-540 (1995)
14. Kohonen, T.: Self-Organized Formation of Topologically Correct Feature Maps. Biological Cybernetics, vol.43 no.1, pp. 5969 (1982)
15. Pearson, K.: On Lines and Planes of Closet Fit to Systems of Points in Space. Philosophical Magazine, vol.2, no.6, pp. 559-572 (1901)
16. Vapnik, V.: The Nature of Statistical Learning Theory. Springer-Verlag, New York (1995)
17. Noriega, L.: Multilayer Perceptron Tutorial. School of Computing Sta ordshire University, pp. 1-12 (2005)