Radial Basis Function Neural Networks in Protein Sequence Classification

Zarita Zainuddin and Maragatham Kumar
School of Mathematical Sciences,
University Science Malaysia, 11800 USM Pulau Pinang

ABSTRACT

Applications of neural networks in bioinformatics have expanded tremendously in recent years due to the capabilities of neural networks to solve biological problems. Neural networks have been implemented in numerous biological fields. In this paper, standard radial basis function and modular radial basis function neural networks are used to classify protein sequences to multiple classes. *n*-gram method is used to transform protein features to real values. A learning strategy known as the self-organized selection of centers is presented. In this strategy, a training algorithm based on subtractive clustering is used to train the network. The radial basis function created by the newrb function from Matlab uses gradient based iterative method as the learning strategy. The proposed method is implemented in the Matlab which creates a new network that undergo a hybrid learning process. The networks called SC/RBF (Subtractive Clustering–Radial Basis Function) and SC/Modular RBF (Subtractive Clustering-Modular Radial Basis Function) are used to test against the standard Radial Basis Function and modular Radial Basis Function in protein classification. Classification criteria consist of two heuristic rules are implemented to test on the classification performance rate. The real world problem that has been considered is classification of human protein sequences into ten different superfamilies which based on protein function groups. These human protein sequences are downloaded from Protein Information Resource (PIR) database.

INTRODUCTION

Bioinformatics is a combination of information technology and biology. Bioinformatics is the information management for biology involving techniques from the applied mathematics, informatics, statistics and computer science to solve biological problems. Neural networks have been used intensively in bioinformatics. Neural networks are capable of handling large amount of bioinformatics data. Wu *et al.* (1992) used Protein Classification Artificial Neural Network System (ProCans) for rapid superfamily classification of the unknown proteins based on the information content of the neural interconnections. Wang *et al.* (2001) presented a Bayesian Neural Network (BNN) approach to classify protein sequences. Blekas (2005) presented a system for multi-class protein classification based on neural network.

The classification problem studied here as follows: Given an unlabeled protein sequence, *S* and a known superfamily, *T*, *S* needs to be determined whether belongs to *T* or not. *T* is referred as the target class and the set of protein sequences not in *T* is defined as the nontarget class. A protein superfamily consists of proteins which share amino acid sequence homology that are evolutionally related and may be functionally and structurally relevant with each other. If the unlabeled sequence, *S* is classified to superfamily, *T*, the structure and the function of *S* can be determined. This process is important in many aspects of bioinformatics and computational biology. A benefit gained from this category of grouping is that some molecular analysis can be carried out within a particular superfamily instead of an individual protein sequence. For example, if the unknown sequence is determined to be in the hemoglobin superfamily, involvement of this unknown protein sequence in the oxygen transporting activity can be assumed. This unknown protein sequence is also assumed to be closely related to human protein sequence. Moreover, the model organism of the unknown protein sequence can be used for new drug testing purposes. A new drug can be tested to determine whether it is safe to be taken by human because the model organism of the unknown protein sequence is closely related to human.

DATA SET

Protein sequences from ten different superfamilies are downloaded from Protein Information Resource (PIR) database. Ten different superfamilies with specific protein functional group are chosen. Protein sequences belonging to these superfamilies are protein sequences from human or protein sequences which are closely related to human protein sequences. These protein sequences are grouped together in a superfamily because of their protein function similarities. Ten superfamilies that are downloaded to be trained / tested in this study are hemoglobin (vertebrate type), villin (validated), cytochrome-b5 reductase (validated), human casein kinase II beta chain, ubiquinol-cytochrome-c Reductase (cytochrome c1) (complex III), human methionyl amino peptidase, insulin-like growth factor binding protein (validated), serine/threonine-protein kinase, human transcription factor 3 and human cytochrome P450 CYP4B1. If an unknown protein is related to one of known function in superfamily, inferences based on the known function and the degree of the relationship can provide the most related clues to the nature of the unknown protein. With total number of 287 protein sequences, 227 protein sequences are used as training set and 60 sequences are used as testing set. The training set is selected by using every fourth entry from each superfamily class.

METHODOLOGY OF IMPLEMENTATION

The implementation of protein sequence classification into ten superfamilies can be divided into two parts:

Protein Feature Extraction Method: *n*-gram Method

An important problem in applying neural network to classify protein sequences is how to interpret the protein sequences as the input to neural network. Good input representation is an important factor for effective neural network learning. The key element of the sequence encoding scheme presented here is a hashing function called the *n*-gram extraction method that was originally used by Cherkassky and Vassilas (1989) for associative database retrieval. The *n*-gram extraction method extracts various features of *n* consecutive residues from a sequence string and gives the number of occurrences of all possible letter pairs, triplet, etc. Wu *et al.* (1995) used a hashing method that counts occurrences of *n*-gram words. The protein sequences were encoded into neural inputs vectors using *n*-gram method. Sharma *et al.* (2004) used bi-gram measure to assess the global frequency of occurrence of any two amino acids consecutively.

This encoding scheme is divided into three parts:

i) sequence interpretation (each protein sequence string is interpreted into strings of different alphabet sets).
ii) *n*-gram extraction (different features are extracted from protein sequence strings).
iii) feature transformation (*n*-gram features are converted to real-valued input of neural network). The *n*-gram features are transformed to real values as an input vector to neural network. Each of the input vectors represents a *n*-gram feature. The real value is scaled between 0 and 1. Let y_i denote the frequency of occurrence of *i*-th *n*-gram feature. *L* denotes the length of the sequence. *n* is the size of the *n*-gram feature.

The *i*-th *n*-gram features value, Y_i is calculated as:

$$Y_\mathrm{i} = \frac{y_i}{L-n+1}$$

where $1 \leq i \leq k^n$.

Y_{i} is used as an input vector to neural network. The denominator denotes the number of n-gram possible in a sequence of length L.

The method was implemented by using Matlab 7.0. nmers function is called from the bioinformatics toolbox. This function is used to implement this method.

Artificial neural network model

In order to perform protein sequence classification, Radial Basis Function (RBF) network is selected because according to Hwang and Bang (1997), it has advantages in architecture interpretability and learning efficiency. RBF network has a faster learning speed because it has two layers of weight and each layer can be determined sequentially. The Mean Square Error (MSE) is used for error measurement. The performance of the network was accessed by comparing the classification errors with actual data and computation time taken to train the network. The Mean Square Error (MSE) is defined as:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2$$

where x_i is the desired output and y_i is the actual system output. n is the number of the testing samples.

Data clustering is an interesting approach to find similarities in data and placing similar data into groups. Clustering algorithms are developed to organize and categorize data. These algorithms are also useful for data compression and model construction. Besides this, clustering algorithms are used to discover relevance knowledge in data. Subtractive clustering uses position of data points to calculate the density function. It uses data points as the candidates for clustering centers. A density measure at data point x_i is defined as:

$$D_i = \sum_{j=1}^{k} \exp(-\frac{\left\|x_i - x_j\right\|^2}{(\frac{r_b}{2})^2})$$

where r_a is a positive constant presenting a neighbourhood radius. A data point will have a high density value if it has many neighbouring data points. The first cluster centre x_{c_1} is chosen as the point with the largest density value D_{c_1}. The density measure of each data point x_i as follows:

$$D_i = D_i - D_{c_1} \exp(-\frac{\left\| x_i - x_{c_1} \right\|^2}{(\frac{r_b}{2})^2})$$

where r_b is a neighbourhood which has measurable reductions in density measure. Subtractive clustering for training RBF network is proposed, which selects the hidden nodes centres (Sarimveis *et al.*, 2003). This subtractive clustering approach will be used in the training of protein sequence data.

The networks called the SC/RBF (Subtractive Clustering – Radial Basis Function) and SC/ Modular RBF (Subtractive Clustering – Modular Radial Basis Function) are used to test against the standard RBF (standard Radial Basis Function) and modular RBF (modular Radial Basis Function) in protein classification. Standard RBF is created by using newrb function. The modular RBF is a combination of two standard RBF and a linear network by using newrb function and newlind function. There are 56 inputs for modular RBF based on a1 *n*-gram and e2 *n*-gram features. Number of protein sequences in the training set is clustered into several groups by using subtractive clustering method with different neighbourhood radius for each data point. Heuristic classification criteria (Wang *et al.*, 2002) with two different rules are used in SC/RBF, SC/Modular RBF, standard RBF and modular RBF to compare in terms of classification performance rate.

Heuristic classification criteria are:

Rule 1: (*pred(x)* ≥ *delta* AND *diff(x)* ≥ *gamma*),
THEN *x* is classified

Rule 2: (*pred(x)* < *delta* OR *diff(x)* < *gamma*),
THEN *x* is unclassified

where network outputs, *pred(x)* are sorted in decreasing order and *diff(x)* is the representation of the difference between the largest output value and second largest output value.

A mathematical expression to show the relationship between the two parameters, gamma and delta:

$$gamma = \frac{delta}{delta + 1}$$

Where delta value characterizes the confidence of the result and *gamma* value controls the quality of the classification. *gamma* and *delta* control the classification performance.

RESULTS AND DISCUSSION

The classified rate (C rate) and unclassified rate (UN rate) changed accordingly with delta values. Good classification performance rates were obtained when delta values from 0.1 to 0.4. These delta values produced a reliable classification performance rate with higher quality and confidence results. Classified rate (C rate) decreases while unclassified rate (UN rate) increases when delta values increase from 0.5 to 0.9. A poor quality and unconfident classification performance rate produced when delta values from 0.5 to 0.9. Delta values from 0.1 to 0.4 were used to compare the classification performance rate for each case.

Table 1 shows the classification performance rates for delta values at 0.1, 0.2, 0.3 and 0.4. a1 *n*-gram extracts better features compared to e2 *n*-gram and a combination of a1 *n*-gram and e2 *n*-gram. Feature extracted were used as an input to RBF neural network and modular RBF neural network which produced the results in Table 1. Percentages of C rate for standard RBF (a1 *n*-gram feature) outperformed percentages of C rate for standard RBF (e2 *n*-gram feature) and modular RBF. This suggest that the standard RBF (a1 *n*-gram feature) produced a better classification rate compared to standard RBF (e2 *n*-gram feature) and modular RBF. Modular RBF produced the worst classification rate.

TABLE 1 : Performance comparison for standard RBF and a modular RBF classification results for test data set (delta = 0.1, 0.2, 0.3 and 0.4).

Radial Basis Network (RBF)	**MSE**	**CPU Time (s)**	**Delta Value**							
			0.1		**0.2**		**0.3**		**0.4**	
			C rate (%)	**UN rate (%)**	**C rate (%)**	**UN rate (%)**	**C rate (%)**	**UN rate (%)**	**C rate (%)**	**UN rate (%)**
Standard RBF (a1 *n*-gram feature)	0.3964	49.203	98.3	1.7	95.0	5.0	93.3	6.7	93.3	6.7
Standard RBF (e2 *n*-gram feature)	0.1010	40.219	96.7	3.3	95.0	5.0	93.3	6.7	90.0	10.0
Modular RBF	0.0204	112.812	96.7	3.3	91.7	8.3	88.3	11.7	85.0	15.0

TABLE 2 : Performance comparison for SC/RBF network (a1 *n*-gram feature) classification results corresponding to the values of radii used to perform subtractive clustering algorithm for training data set.

Radii	**Number of Sequences in Training Set**	**MSE**	**CPU Time(s)**
0.1	188	0.3841	13.078
0.2	188	0.3841	13.090
0.3	188	0.3841	13.081
0.4	187	0.3680	11.034
0.5	186	0.3736	10.282
0.6	185	0.3813	9.5780
0.7	183	0.3478	9.8910
0.8	180	0.2589	9.9530
0.9	174	0.2222	7.719

TABLE 3 :Performance comparison for SC/RBF network (e2 *n*-gram feature) classification results corresponding to the value of radii used to perform subtractive clustering algorithm for training data set.

Radii	**Number of Sequences in Training Set**	**MSE**	**CPU Time(s)**
0.1	176	0.0876	12.969
0.2	176	0.0876	13.013
0.3	176	0.0876	12.839
0.4	176	0.0876	12.900
0.5	175	0.1028	12.860
0.6	175	0.1028	12.781
0.7	174	0.0983	15.230
0.8	173	0.0968	17.890
0.9	170	0.1086	16.010

Table 2 and Table 3 show the number of sequences in the training set which were reduced using subtractive clustering. Percentages of C rate for SC/RBF (a1 *n*-gram feature) when radii = 0.9 outperformed the percentages of C rate when radii values from 0.1 to 0.8. This shows that the best classification performance rate for SC/RBF (a1 *n*-gram feature) is when radii = 0.9. For SC/RBF (e2 *n*-gram feature), the best classification performance rate is when radii = 0.1, 0.2, 0.3 or 0.4. Radii = 0.1, 0.2, 0.3 or 0.4 give the same classification performance rate because the number of sequences in the training set after clustering are the same.

Each radii value consists of different number of protein sequences in the training set. Radii values for SC/RBF (a1 *n*-gram feature) from Table 2 are combined with radii values for SC/RBF (e2 *n*-gram feature) in Table 3. For example, number of sequences in training set for radii = 0.1 from Table 2 and number of sequences in training set for radii = 0.1 from Table 3 are used as inputs to SC/Modular.

TABLE 4: Performance comparison for SC/RBF and a SC/Modular RBF classification results for test data set (delta = 0.1, 0.2, 0.3 and 0.4).

Radial Basis Network (RBF)	**MSE**	**CPU Time (s)**	**Delta Value**							
			0.1		**0.2**		**0.3**		**0.4**	
			C rate (%)	**UN rate (%)**	**C rate (%)**	**UN rate (%)**	**C rate (%)**	**UN rate (%)**	**C rate (%)**	**UN rate (%)**
SC/RBF (a1 *n*-gam feature) (radii = 0.9)	0.2222	7.719	96.7	3.3	96.7	3.3	95.0	5.0	91.7	8.3
SC/RBF (e2 *n*-gram feature) (radii = 0.1)	0.0876	12.969	91.6	8.3	86.7	13.3	83.3	16.7	81.7	18.3
SC/Modul ar RBF (radii = 0.4)	0.0207	58.000	98.3	1.7	96.7	3.3	90.0	10.0	88.3	11.7

Results in Table 4 were produced by the simulation of the proposed networks which were SC/RBF and SC/Modular RBF networks. Results with different number of sequences in the training set were obtained by using the off-line training mode. Reduced number of sequences in the training set took shorter time compared to the actual number of training set. From Table 4, the computation time was reduced about 85% which was produced by the ninth radii value for SC/RBF (a1 *n*-gram feature). The computation time was reduced about 68% which was produced by the first radii value for SC/RBF (e2 *n*-gram feature) and about 50% for SC/Modular RBF which used radii = 0.4 (a1 *n*-gram feature and e2 *n*-gram feature).As a conclusion based on the results above, smaller training sets used less computation time.

Based on the results in Table 1 and Table 4, there were not many differences in the percentages of C rate for delta values from 0.1 to 0.4 for each case. This suggests that there were no significant results in terms of classification performance rate for delta from 0.1 to 0.4 for the standard RBF and modular RBF. There were also not many differences in the percentages of C rate for delta values from 0.1 to 0.4 for different radii values. This shows that simulation done for different radii values for SC/RBF (a1 *n*-gram and e2 *n*-gram features) and SC/Modular RBF also do not give much significant results in terms of classification performance rate and MSE.

CONCLUSIONS

This paper studies about classifying protein sequences into ten different superfamilies. The radial basis function neural network is applied to this classification problem. The main investigations in this study were as follows: (i) A comparative study is done using standard RBF, modular RBF, SC/RBF and SC/modular RBF. (ii) Classification criteria approach is applied to this study to compare the classification performance rate in the networks. Results from the experiments and case study show that standard RBF(a1 *n*-gram feature) performs better classification rate compared to standard RBF(e2 *n*-gram feature) and modular RBF. The proposed networks produce shorter training time compared to the standard networks. Classification performance rates are compared between the proposed networks and the standard networks. Delta values from 0.1 to 0.4 are considered because the classification performance rates for these delta values produce quality and confidence results. Based on the results, there are not much significant results in terms of the classification rates for the proposed networks and standard networks.

REFERENCES

Blekas, K., Fotiadis, D., Likas, A. 2005. Motif–Based Protein Sequence Classification using Neural Network. *Journal of Computational Biology* **12**(1), 64-82.

Cherkassky, V. and Vassilas, N. 1989. Performance of the Back Propagation Networks for Associative Database Retrieval. *Proceedings of the International Conference on Neural Networks* **1**, 77-83.

Wang, J.T.L., Ma, Q., Shasha, D. and Wu, C.H. 2001. New Techniques for Extracting Features from Protein Sequences. *IBM Systems Journal* **40**(2), 426-441.

Wu, C., Whitson, G., McLarty, J., Ermongkonchai, A. and Chang, T.C. 1992. Protein Classification Artificial Neural System. *Protein Science* **1**(5), 667-677.

Hwang, Y.S. and Bang, Y.S. 1997. An Efficient Method to Construct Neural Network Classifier. *Neural Networks* **10**(8), 1495-1503.

Wang, D., Lee, N. L., Dillon, T. S. and Hoogenraad, N.J. 2002. Protein Sequence Classification using Radial Basis Function (RBF) Neural Networks. *Proceedings of the 9th International Conference on Neural Information Processing* **2**, 764-768.

Wu, C., Berry, M., Shivakumar, S. and McLarty, J. 1995. Neural Networks for Full-Scale Protein Sequence Classification: Sequence Encoding with Singular Value Decomposition. *Machine Learning* **21**, 177-193.

Sarimveis, H., Alexandridis, A., Bafas, G. 2003. A fast training algorithm for RBF networks based on subtractive clustering. *Neurocomputing* **51**, 501-505.

Sharma, S., Kumar, V., Rani, T.S., Bhavani., S.D. and Raju, S.B. 2004. Application of Neural Networks for Protein Sequence Classification. *Proceedings of International Conference on Intelligent Sensing and Information Processing 2004,* 325 – 328.