IJCSC International Journal of Computer Science and Communication Vol. 3, No. 1, January-June 2012, pp. 227-231

CLASSIFICATION OF NEURAL NETWORK STRUCTURES FOR BREAST CANCER DIAGNOSIS

S. Swathi[1], S. Rizwana[2], G. Anjan Babu[3], P. Santhosh Kumar[4] and P.V.G.K. Sarma[5]

[1,2,3]Department of Computer Science, SV University, Tirupati, [1]*E-mail: swathikarthikeya89@gmail.com*

[4,5]Department of Biotechnology, SVIMS, Tirupati

ABSTRACT

Breast cancer diagnosis has been approached by various machine learning techniques for many years. Here we can study the performance of different Neural Network structures: Radial Basis Function(RBF), General Regression Neural Network(GRNN), Probabilistic Neural Network (PNN), Multi layer Perceptron model and Back propagation Neural Network(BPNN), are examined on the Wisconsin Breast Cancer Data (WBCD). This paper present the results of comparison among these networks and the classification results have indicated that the Back Propagation Neural Network gave good diagnostic performance of 99.28%.

Keywords: Breast cancer, Radial basis function, General Regression Neural Network, Probabilistic Neural Network, Wisconsin breast cancer data, Back Propagation Neural Network.

1. INTRODUCTION

Neural Networks are currently a 'hot' research area in medicine, particularly in the fields of radiology, urology, cardiology, oncology and etc. Keeping in view of the significant characteristics of Neural Network (NN) and its advantages for implementation of the classification problem, Neural Network technique is highly used in classification of data related to medical field. Owing to their wide range of applicability and their ability to learn complex and non linear relationships including noisy or less precise information, Neural Networks (NN) technique is used to solve problems in biomedical engineering. By their nature, Neural Networks are capable of high-speed parallel signal processing in real time. They have an advantage over conventional technologies because they can solve problems that are too complex that do not have any algorithmic solution or for which an algorithmic solution is too complex. Neural Networks are trained by examples instead of rules that are automated. This is one of the major advantages of Neural networks over traditional expert systems [1, 2].

2. NEURAL NETWORK IN MEDICAL FIELD

When Neural Network (NN) is used in medical diagnosis they are not affected by factors such as human fatigue, emotional states and habituation. They are capable of rapid identification, analyses of conditions, and diagnosis in real time. With the spread of Neural Networks in almost all fields of science and engineering, it has found extensive application in biomedical engineering field also. The applications of neural networks in biomedical computing are numerous. Various applications of ANN techniques in medical field like medical expert system, cardiology, neurology, rheumatology, mammography and pulmonology were studied [3,4]. In this study medical data related to Breast Cancer is considered for classification purpose to identify the disease. As the Neural Networks are inherently parallel in nature, this technique is considered in this study to implement parallelism for calculating the output at each node in different layers of the network. Basic unit of modularity in a network is neuron. Every neuron operates independently, processing the input receives, adjusting weights, and propagating its computed output thus a neuron is a natural level of parallelization for neural networks. Every neuron is treated as a parallel process. For example a layer other than the input layer consists of '*m*' neurons and assume that processing time '*t*' units to calculate the output at each neuron is similar. If the parallel concept is not adopted in neural network '*mt* units' of time is needed to calculate the output. The needed time can be reduced by m times, if parallel concept is implemented at neuron level. If the network consists of many hidden layers, the processing time can be reduced at each layer in the network and thus the overall training time of the network can be reduced drastically.

3. ABOUT DATA SET

This breast cancer database is downloaded from the UCI machine-learning repository [5], which was collected by Dr. William H. Wolberg from the University of Wisconsin Hospitals, Madison [6]. The dataset is comprised of elements that consist of various scalar observations. The total number of the original samples is 699 with 16 samples contain missing values. The dataset contains two classes referring to benign and malignant samples. There are 458 samples in the dataset that are assigned to benign and the other 241 samples are malignant. The original dataset contains 11 attributes including both sample id number and class label, which are removed in the actual dataset that are used in our experiments. The remaining 9 attributes represent 9 cytological characteristics of breast fine-needle aspirates (FNAs), as shown in Table 1. The cytological characteristics of breast FNAs were valued on a scale of one to ten, with one being the closest to benign and ten the most malignant.

Table 1
Attribute Information

No	*Attribute*	*Domain*
1	Clump thickness	1-10
2	Uniformity of cell size	1-10
3	Uniformity of cell shape	1-10
4	Marginal Adhesion	1-10
5	Single Epithelial Cell Size	1-10
6	Bare Nuclei	1-10
7	Bland Chromatin	1-10
8	Normal Nucleoli	1-10
9	Mitoses	1-10
10	Class	2 for benign and 4 for malignant

These attributes measure the external appearance and internal chromosome changes in nine different scales. There are two values in the class variable of breast cancer:

Benign (non-cancerous)

and

Malignant (cancerous),

Which is represented numerically by 2 and 4 respectively.

Descriptions of Database:

- Number of instances 699
- Number of attributes: 10 plus the class attribute
- Attributes 2 through 10 will be used to represent instances
- Each instance has one of 2 possible classes: benign or malignant
- Class distribution: Benign: 458 (65.5%)

  Malignant: 241 (34.5%)

The data set was partitioned into two sets: training and testing set. The testing set was not seen by any neural network during the training phase. It is only used for testing the generalization of neural network ensembles after they are trained. We used the 80% examples for the training set, and the rest 20% examples for the testing set. At the start of training, all connection weights in network are set to random values. All input vectors are normalized so that the minimum and maximum are 0 and 1 respectively. All the computations are implemented using MATLAB version 7.0.

4. APPLIED NEURAL NETWORK STRUCTURE

4.1 Radial Basis Functions (RBF) [7]

RBF is a different approach by viewing the design of a neural network as a curve-fitting problem in a high-dimensional space. According to this viewpoint, learning is equivalent to finding a surface in a multidimensional space that provides a best fit to the training data, with the criterion for "best fit" being measured in some statistical sense. The construction of a radial-basis function network in its most basic form involves three entirely different layers. The input layer is made up of source nodes. The second layer is a hidden layer of high enough dimension, which serves a different purpose from that in a multilayer perceptron. The output layer supplies the response of the network to the activation patterns applied to the input layer. The transformation from the input space to the hidden-unit space is nonlinear where as the transformation from the hidden-unit space to the output space is linear.

4.2 Probabilistic Neural Networks (PNN) [8]

The Probabilistic Neural Network introduced by Specht is essentially based on the well-known Bayesian classifier technique commonly used in many classical pattern-recognition problems. Consider a pattern vector x with m dimensions that belongs to one of two categories $K1$ and $K2$. Let $F1(x)$ and $F2(x)$ be the probability density functions (pdf) for the classification categories $K1$ and $K2$, respectively. From Bayes' discriminant decision rule, x belongs to $K1$ if

$$\frac{F1(x)}{F2(x)} > \frac{L1}{L2}\frac{p2}{p1} \tag{1}$$

Conversely, x belongs to $K2$ if

$$\frac{F1(x)}{F2(x)} > \frac{L1}{L2}\frac{p2}{p1} \tag{2}$$

where $L1$ is the loss or cost function associated with misclassifying the vector as belonging to category $K1$ while it belongs to category $K2$, $L2$ is the loss function associated with misclassifying the vector as belonging to category $K2$ while it belongs to category $K1$, $P1$ is the prior probability of occurrence of category $K1$, and $P2$ is the prior probability of occurrence of category $K2$. In many situations, the loss functions and the prior probabilities can be considered equal. Hence the key to using the decision rules given by equations (1) and (2) is to estimate the probability density functions from the training patterns.

In the PNN, a nonparametric estimation technique known as Parzen windows is used to construct the class-dependent probability density functions (pdf) for each classification category required by Bayes' theory. This allows determination of the chance a given vector pattern lies within a given category. Combining this with the relative frequency of each category, the PNN selects the most likely category for the given pattern vector. Both Bayes' theory and Parzen windows are theoretically well established, have been in use for decades in many engineering applications, and are treated at length in a variety of statistical textbooks. If the jth training pattern for category $K1$ is x_j, then the Parzen estimate of the pdf for category $K1$ is

$$F_1(x) = \frac{1}{2(\pi)^{m/2}\sigma^m n}\sum \exp\left[-\frac{(x-xj)^T(x-xj)}{2\sigma^2}\right] \quad (3)$$

where n is the number of training patterns, m is the input space dimension, j is the pattern number, and σ is an adjustable smoothing parameter. However, the choice of σ in general has been found to be not too sensitive to variations in its value.

4.3 Generalized Regression Neural Networks (GRNN) [9]

The generalized regression neural networks (GRNNs) are the paradigms of radial basis function (RBF) networks, often used for function approximations. It's another term for Nadaraya- Watson kernel regression, and has the following form for the function mapping.

$$y(x) = \frac{\Sigma_k t_k \exp\left\{-\|x-x_k\|^2/2h^2\right\}}{\Sigma_k \exp\left\{-\|x-x_k\|^2/2h^2\right\}} \quad (4)$$

GRNNs share a special property, namely that they do not require iterative training; the hiddento-output weights are just the target values *tk*, so the output $y(x)$, is simply a weighted average of the target values *tk* of training cases *xk* close to the given input case x. It can be viewed as a normalized RBF network in which there is a hidden unit centered at every training case. These RBF units are called "kernels" and are usually probability density functions such as the Gaussians considered in (4). The only weights that need to be learned are the widths of the RBF units h. These widths (often a single width is used) are called "smoothing parameters" or "bandwidths" and are usually chosen by cross validation. GRNN is a universal approximator for smooth functions, so it should be able to solve any smooth function approximation problem given enough data. The main drawback of GRNNs is that, like kernel methods in general, they suffer seriously from the curse of dimensionality. GRNNs cannot ignore irrelevant inputs without major modifications to the basic algorithm.

4.4 Multi Layer Perceptron Model

If a Multilayer Perceptron has a linear activation function in all neurons that is , a simple on-off mechanism to determine whether or not a neuron fires, then it is easily proved with linear algebra that any number of layers can be reduced to the standard two – layer input – output model. The two main activation functions used in current applications are both sigmoid and are described by

$$\phi(y_i) = \tanh(v_i) \; and \; \phi(y_i) = (1+e^{-v_i})^{-1} \quad (5)$$

In which the former function is hyperbolic tangent which ranges from –1 to 1 and the latter the logistic function is similar in shape but ranges from 0 to 1. Here Y_i is the output of the ith node (neuron) and V_i is the weighted sum of the input synapses.

The multilayer perceptron consist of three or more layers (an input and an output layer with one or more hidden layers) of non linearly-activating nodes. Each nodes in one layer connects with a certain weight W_{ij} to every node in the following layer.

5. BACK PROPAGATION ALGORITHM

In ANN, the assigned weights for each connector of node resemble the long term memory. They contain information of the input's importance and ANN learns by repeated adjustments of these weights. The weight adjustments are carried out according to the mathematical functions known as learning or activation function, which will be compared to the threshold value of the network. A feed forward back propagation artificial can learn a function of mapping inputs to outputs by being trained with cases of input-output pairs. Back propagation neural network (BPNN) is actually a descending slope method to minimize the total square of the output, calculated by the network [11]. There are three phases in the training process: first is to send the signal pattern forward, second is to calculate the propagated error and the last is to update all weights in the network. In addition BPNN also has the advantages of faster learning in multilayer Neural Network, especially sigmoidal activation function is represented by hyperbolic tangent. The neurons in feed forward networks can be any transfer function of the designer wishes to use. The usually used transfer function is the sigmoid function with threshold defined as in equation.

$$f\left(\sum_{i=1}^{n} wixi - \theta\right) = 1/\left(1+\exp\left(-\left(\sum_{i=1}^{n} wixi - \theta\right)\right)\right) \quad (6)$$

where xi is the input to the node and wi is the corresponding input weight, θ is a value which is usually called the threshold, n is the number of inputs to the node. The network performance and convergence depends on many parameters like initial weights, learning rate and momentum used during the training process.

6. SIMULATION RESULTS

The simulations were realized by using MATLAB 6.0 Neural Network Toolbox. Four different neural network structure, multi layer perceptron, radial basis function, probabilistic neural network, generalized regression and a feed forward back propagation neural network were applied to WBCD database to show the performance of statistical neural networks on breast cancer data. The spread value of RBF, PNN, BPNN and GRNN was chosen 4.4, 1 and 3, respectively. In MLP, learning rate was 0.6.

6.1 Training Data Simulation

Half of the database was used for training. 222 samples of the training data belong to benign class and 120 samples belong to malignant class. The classification results of the training set by RBF, PNN, GRNN, BPNN and MLP were given in the Tables.

Table 2
Classification of Training Data by RBF

Class	*Benign*	*Malignant*
True	222	120
False	0	0

Table 3
Classification of Training Data by PNN

Class	*Benign*	*Malignant*
True	222	120
False	0	0

Table 4
Classification of Training Data by GRNN

Class	*Benign*	*Malignant*
True	217	113
False	5	7

Table 5
Classification of Training Data by MLP

Class	*Benign*	*Malignant*
True	217	118
False	5	2

Table 6
Classification of Training Data by BPNN (Average)

Class	*Benign*	*Malignant*
True	220	118
False	2	2

Table 7
Performance for Training Data Classification

Type	*Performance*
RBF	100%
PNN	100%
GRNN	96.4%
MLP	98.04%
BPNN	98.78%

RBF and PNN give the best classification accuracy with 342 correct classifications while GRNN has the lowest accuracy with 330 correct classifications for the training set. MLP has 335 correct classifications, and BPNN has 338 correct classifications in the training set.

6.2 Test Data Simulation

Table 8
Classification of Test Data by RBF

Class	*Benign*	*Malignant*
True	215	113
False	7	6

Table 9
Classification of Test Data by PNN

Class	*Benign*	*Malignant*
True	219	112
False	3	7

Table 10
Classification of Test Data by GRNN

Class	*Benign*	*Malignant*
True	221	116
False	1	3

Table 11
Classification of Test Data by MLP

Class	*Benign*	*Malignant*
True	212	114
False	10	5

Table 12
Classification of Test Data by BPNN (Average)

Class	*Benign*	*Malignant*
True	221	117
False	1	2

Table 13
Performance for Test Data Classification

Type	*Performance*
RBF	96.18%
PNN	97.0%
GRNN	98.18%
MLP	95.74%
BPNN	99.28%

A total of 341 samples were applied to the networks as test data; that is, 50% percent of the database was used for testing. 222 samples, which belong to benign class data, and 119 samples, which belong to malignant class, were chosen for the test. The results for RBF, PNN, GRNN, MLP, and BPNN are shown in the Table 8, 9,10,11,12. For the test set BPNN gives the best classification accuracy with 338 Classifications, GRNN was the second best network with 337 classifications. While MLP has the lowest accuracy with 326 correct classifications. RBF classified 328 samples and PNN with 331 classifications. Overall classification performances were 96.18% for RBF, 97.0% for PNN, 98.8% for GRNN ,95.74% for MLP and 99.28% for BPNN.

7. CONCLUSION

In this research, the performance of neural network structures is classified on a real life problem, the Wisconsin breast cancer diagnosis problem. Here RBF and PNN are the best classifiers in training set, however the important result must be considered with test data since it shows the further performance of the network. BPNN gives the best classification accuracy when the set is considered. According to overall results, it is seen that the most suitable neural network model for classifying WBCD data is BPNN. This network has given the good diagnostic performance of 99.28%. Further work is needed to increase the accuracy of classification of breast cancer diagnosis.

REFERENCES

[1] K. Anil Jain, Jianchang Mao and K.M. Mohiuddin. "Artificial Neural Networks: A Tutorial". *IEEE Computers*, 1996, pp. 31-44.

[2] George Cybenko. "Neural Networks in Computational Science and Engineering". *IEEE Computational Science and Engineering*, 1996, pp. 36-42.

[3] Dr. A. Kandaswamy, "Applications of Artificial Neural Networks in Bio Medical Engineering". *The Institute of Electronics and Telecommunicatio Engineers, Proceedings of the Zonal Seminar on Neural Networks*, Nov 20-21, 1997.

[4] A. Kusiak, K.H. Kernstine, J.A. Kern, K>A. McLaughlin and T.L. Tseng, "Data Mining: Medical and Engineering Case Studies", *Proceedings of the Industrial Engineering Research 2000 Conference*, Cleveland, Ohio, May 21-23, pp. 1-7, 2000.

[5] UCI Machine Learning Repository. [http://archive.ics.uci.edu/ml/]. Irvine, CA: University of California, Center for Machine Learning and Intelligent Systems.

[6] W.H. Wolberg and O.L. Mangasarian, —Multisurface Method of Pattern Separation for Medical Diagnosis Applied to Breast Cytology", *in Proceedings of the National Academy of Sciences*, **87**, pp. 9193-9196, U.S.A., December 1990.

[7] Newton Cheung, "Machine Learning Techniques for Medical Analysis", *Thesis for the Degree of Bachelor of Engineering*, 2001.

[8] S. Haykin, "Neural Networks: A Comprehensive Foundation", *Mac Millan College Publishing Company*, 1994.

[9] Anthony T.C. Goh, "Probabilistic Neural Network for Seismic Liquefaction Potential", *NRC Research Press Web Site*, 2002.

[10] C. Lu, J. De Brabanter, S. Van Huffel, I. Vergote, D. Timmerman, "Using Artificial Neural Networks to Predict Malignancy of Ovarian Tumors", *23rd Annual Int. Conf. Of the IEEE Engineering in Medicine and Biology Society*.

[11] D.W. Ruck,, S.K. Rogers., M. Kabrisky., P.S. Meibeck., and M.E. Oxley., "Comparatives Analysis of Back Propagation and the Extended Kalman Filter for Training Multilayer perceptrons", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, June 1992, **14(6)**, pp. 686-691.

[12] American Cancer Society, Cancer Facts and Figures 2007,2007th ed. American Cancer Society, 2007.

[13] Tuba Kiyan and Tulay Yildirim, "Breast Cancer Diagnosis Using Statistical Neural Networks", *Istanbul University, Journal of Electrical and Electronics Engineering*, Year 2004, **4(2)**, 1149-1153.