International Journal of Multidisciplinary and Current Research

Research Article

ISSN: 2321-3124
Available at: http://ijmcr.com

Performance Analysis of Radial Basis Function Neural Network for Pattern Classification

Holkar S.R.[†] and Manu Pratap Singh[‡]

[†]Shri Venketeshwara University, Gajraula, Amroha, Uttar Pradesh, India
[‡]Department of Computer Science, Dr. B.R.Ambedkar University, Agra -282002 Uttar Pradesh, India

Accepted 05 June 2015, Available online 11 June 2015, **Vol.3** (**May/June 2015 issue)**

Abstract

Feed forward neural networks with Backpropagation learning rule has been used widely for the generalized pattern classification but the ill posing and unknown local error minimum problem limits the performance of Backpropagation learning rule for the problems of large feature vectors. Another type of feed forward neural network architecture i.e. Radial Basis exhibits more efficient and general approximation with respect to Backpropagation network. The purpose of this study is to analyze the performance of Radial Basis function type feed forward neural networks for the pattern classification. Therefore to perform this analysis the task of pattern classification for hand written English vowels using radial basis function neural network is used. This Implementation has been done with the training of five different samples of hand written English vowels. Adjusting the connection strength and network parameters perform the training process in the neural network. By using a simulator program, both the algorithms i.e. BP and RBF are compared with five data sets of handwritten English language vowels. The simulated results indicate the fast & good convergence and high classification rate for the RBF network.

Keywords: *Pattern Classification, Radial Basis Function Neural Network, Feed Forward neural networks, handwritten pattern Recognition*

1. Introduction

The increasing popularity of the neural networks is partly due to their ability to learn and generalization [1- 6]. The feed forward neural network makes no prior assumption about the statistics of input data and can construct complex decision boundaries [7]. This property makes neural networks, an attractive tool to many pattern classification problems such as hand written curve scripts [8-10]. The neural network consists of an input layer of nodes, one or more hidden layers, and an output layer [11]. Each node in the layer has one corresponding node in the next layer, thus creating the stacking effect.

In neural networks, the choice of learning algorithm, network topology, weight and bias initialization and input pattern representation are important factors for the network performance in order to accomplish the learning. In particular, the choice of learning algorithm determines the rate of convergence, computational cost and the optimality of the solution. The multilayer feed forward is one of the most widely used neural network architecture. The learning process for the feed forward network can consider as the minimization of the specified error (E) that depends on all the free parameters of the network. Among the various learning algorithms, the back propagation algorithm [12] is one of the most important and widely used algorithms and has been successfully applied in many fields. It is based on the steepest descent gradient and has the advantage of being less computationally expensive. However, the conventional back propagation learning algorithm suffers from short coming, such as slow convergence rate and fixed learning rate. Furthermore it can be stuck to a local minimum of the error.

There are numerous algorithms have been proposed to improve the back propagation learning algorithm [13-15]. Several other variations of back propagation algorithms based on second order methods have been proposed [16-21]. This method generally converges to minima more rapidly than the method based solely on gradient decent method. However, they require an additional storage and the inversion of the second-order derivatives of the error function with respect to the weights. The storage requirement and computational cost, increases with the square of the number of weights
In this paper the objective is to analyze the performance of Radial Basis function type feed forward neural networks for the pattern classification. Therefore, we consider the neural networks architecture which is trained with the gradient descent generalize delta

learning rule for Radial basis function [22] in the single hidden layer for the handwritten English vowels. The rate of convergence and the number of epochs for each pattern are important observation of this study. The simulated results are determined from the number of trails with five sets of handwritten characters of English vowels.

The next section presents the implementation of the neural network architecture with Radial basis function. The experimental results and discussion are presented in section 3. Section 5 contents the conclusion of this paper.

2. Implementation of the Radial basis function

The architecture and training methods of the RBF network are well known [23-25] and well established. An RBFN is a three layer feed forward network that consists of one input layer, one hidden layer and one output layer as shown in figure (1), each input neuron corresponds to a component of an input vector x. The hidden layer consists of K neurons and one bias neuron. Each node in the hidden layer uses an RBF denoted with $\phi(r)$, as its non-linear activation function.

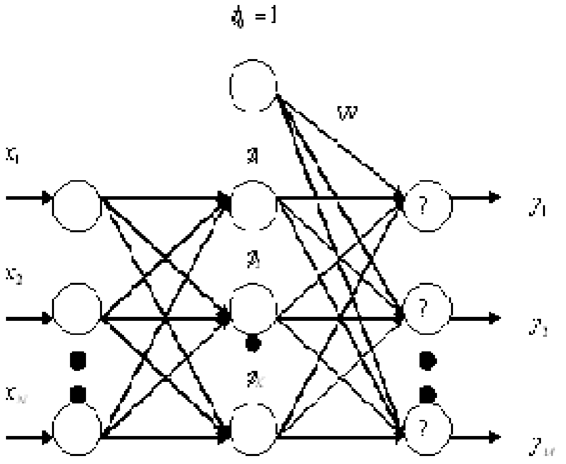

Fig. 1: Architecture of the RBFN.

The input layer has N nodes; the hidden and the output layer have K and M neurons, respectively. $\phi_0(x)=1$, corresponds to the bias

The RBFN can achieve a global optimal solution to the adjustable weights in the minimum MSE range by using the linear optimization method. Thus, for an input pattern x, the output of the jth node of the output layer can define as:

$$y_j(x)=\sum_{k=1}^{K} w_{kj}\phi_k(\|x_i-\mu_k\|)+w_{0j} \qquad (2.1)$$

for $j=(1,2,....,M)$ where $y_j(x)$ is the output of the j^{th} processing element of the output layer for the RBFN, w_{kj} is the connection weight from the kth hidden unit to the jth output unit, μ_k is the prototype or centre of the kth hidden unit. The Radial Basis Function $\phi(.)$ is typically selected as the Gaussian function that can be represented as:

$$\phi_k(x_l)=\exp\left(-\frac{\|x_l-\mu_k\|^2}{2\sigma_k^2}\right) \text{ for } k=(1,2,....,K) \qquad (2.2)$$

Where x is the N- dimensional input vector, μ_k is the vector determining the centre of the basis function ϕ_k and σ_k represents the width of the neuron. The weight vector between the input layer and the kth hidden layer neuron can consider as the centre μ_k for the feed forward RBF neural network.

Hence, for a set of L pattern pairs $\{(x_l, y_l)\}$, (2.1) can be expressed in the matrix form as

$$Y=w^T\phi \qquad (2.3)$$

where $W=[w_1,..............w_m]$ is a KxM weight matrix, $w_j=(w_{0j},...........w_{kj})^T$, $\phi=[\phi_0,..........\phi_k]$ is a K x L matrix, $\phi_{l,k}=[\phi_{l,1},..........\phi_{l,k}]^T$ is the output of the hidden layer for the lth sample, $\phi_{l,k}=\phi(\|x_l-c_k\|)$, $Y=[y_1, y_{2,}......y_m]$ is a M x L matrix and $y_{lj}=(y_{l1,}.....y_{lm})^T$

The Radial basis function uses the weight and parameter updating in two different ways. The first is in between single hidden layer and output layer and second one is the adjustment of radial distance parameter between input layer and hidden layer. It is found by treating the basis function centers and widths along with the second layer weights, as adaptive parameters to be determined by minimization of an error function as the least mean square error (LMS). This error will minimize along the decent gradient of error surface in the weight space between hidden layer and the output layer. The same error will minimize with respect to the Gaussian basis function's parameter as defined in equation (2.2). Thus, we obtain the expressions for the derivatives of the error function with respect to the weights and basis function parameters for the set of L pattern pairs (x^l, y^l) as; where $l=1$ to L.

$$\Delta w_{jk}=-\eta_1\frac{\partial E^l}{\partial w_{jk}} \qquad (2.4)$$

$$\Delta\mu_k=-\eta_2\frac{\partial E^l}{\partial\mu_k} \qquad (2.5)$$

and $\Delta\sigma_k=-\eta_3\dfrac{\partial E^l}{\partial\sigma_k}$ (2.6) here, $E^l = \frac{1}{2}\sum_{j=1}^{M}(d_j^l - y_j^l)^2$

and $y_j^l = \sum_{k=1}^{K} w_{jk}\phi_k(\|x^l - \mu_k^l\|)$ (2.7)

and $\phi_k(\|x^l - \mu_k^l\|) = \exp(-\frac{\|x^l - \mu_k^l\|^2}{2\sigma_k^2})$

Hence, from the equation (2.4) we have,

$$\Delta w_{jk} = -\eta_1 \frac{\partial E^l}{\partial w_{jk}} = -\eta_1 \frac{\partial E^l}{\partial y_j^l}.\frac{\partial y_j^l}{\partial w_{jk}} = -\eta_1 \frac{\partial E^l}{\partial y_j^l}.\phi_k\|x^l - \mu_k^l\|$$

So, that $\Delta w_{jk} = \eta_1 \sum_{j=1}^{M}\sum_{k=1}^{K}(d_j^l - y_j^l)\dot{s}_j^l(y_j^l)\exp(-\frac{(\|x^l - \mu_k^l\|)^2}{2\sigma_k^2})$ (2.8)

In the same way we can update the basis function parameter as:

$$\Delta\mu_{ki} = -\eta_2 \frac{\partial E^l}{\partial \mu_{ki}} = -\eta_2 \frac{\partial E^l}{\partial y_j^l}.\frac{\partial y_j^l}{\partial \mu_{ki}}$$

or $\Delta\mu_{ki} = \eta_2 \sum_{j=1}^{M}\sum_{k=1}^{K}(d_j^l - y_j^l)\dot{s}_j^l(y_j^l)w_{jk}.\exp(-\frac{(\|x_i^l - \mu_{ki}^l\|)^2}{2\sigma_k^2})\ (\frac{x_i^l - \mu_{ki}^l}{\sigma_k^2})$

(2.9)

Now, from the equation (2.6) we have

$$\Delta\sigma_k = -\eta_3 \frac{\partial E^l}{\partial \sigma_k} = -\eta_3 \frac{\partial E^l}{\partial y_j^l}.\frac{\partial y_j^l}{\partial \sigma_k}$$

$$\Delta\sigma_k = \eta_3 \sum_{j=1}^{M}\sum_{k=1}^{K}(d_j^l - y_j^l)\dot{s}_j^l(y_j^l).w_{jk}.\exp(-\frac{(\|x_i^l - \mu_{ki}^l\|)^2}{2\sigma_k^2})\frac{\|x_i^l - \mu_{ki}^l\|^2}{\sigma_k^3} \quad (2.10)$$

So that, we have from equations (2.8), (2.9) & (2.10) the expressions for change in weight vector & basis function parameters to accomplish the learning in supervised way. Thus, for reasonable well-localized RBF, an input will generate a significant activation in a small region and the opportunity of getting stuck at a local minimum is small. Hence, the training of the network for L pattern pair i.e. (x^l, y^l) will accomplish in iterative manner with the modification of weight vector and basis function parameters corresponding to each presented pattern vector. The parameters of the network at the mth step of iteration can express as;

$$w_{jk}(m) = w_{jk}(m-1) + \eta_1 \sum_{j=1}^{M}\sum_{k=1}^{K}(d_j^l - y_j^l)\dot{s}_j^l(y_j^l).\exp(-\frac{(\|x_i^l - \mu_{ki}^l\|)^2}{2\sigma_k^2}) \quad (2.11)$$

$$\mu_{ki}(m) = \mu_{ki}(m-1) + \eta_2 \sum_{j=1}^{M}\sum_{k=1}^{K}(d_j^l - y_j^l)\dot{s}_j^l(y_j^l).w_{jk}.\phi_k(x_i^l).(\frac{x_i^l - \mu_{ki}^l}{\sigma_k^2}) \quad (2.12)$$

$$\sigma_k(m) = \sigma_k(m-1) + \eta_3 \sum_{j=1}^{M}\sum_{k=1}^{K}(d_j^l - y_j^l)\dot{s}_j^l(y_j^l).w_{jk}.\phi_k(x_i^l).\frac{\|x_i^l - \mu_{ki}^l\|^2}{\sigma_k^3} \quad (2.13)$$

where η_1, η_2 & η_3 are the coefficient of learning rate.

Hence, among the neural network models, RBF network seems to be quit effective for pattern recognition task such as handwritten character recognition. Since it is extremely flexible to accommodate various and minute variations in data. Now, in the following subsection we are presenting the simulation designed and implementation details of redial basis function worked as a classifier for the handwritten English vowels recognition problem.

3. Simulation Design and Implementation Details

The experiment described in this segment is designed to implement the algorithm for RBF network with decent gradient method and simple BP algorithm. The task associated to the neural networks in both experiments was to accomplish the training of the handwritten English language vowels in order to generate the appropriate classification. For this, first we obtained the scanned image of five different types of samples of handwritten English language vowels as shown in figure (2). After collecting these samples, we partitioned an English vowel image in to four equal parts and calculated the density of the pixels, which belong to the central of gravities of these partitioned images of an English vowel. Like this, we will get 4 densities from an image of handwritten English language vowel, which we use to provide the input to the feed forward neural network. We use this procedure of generating input for a feed forward neural network with each sample of English vowel scanned images.

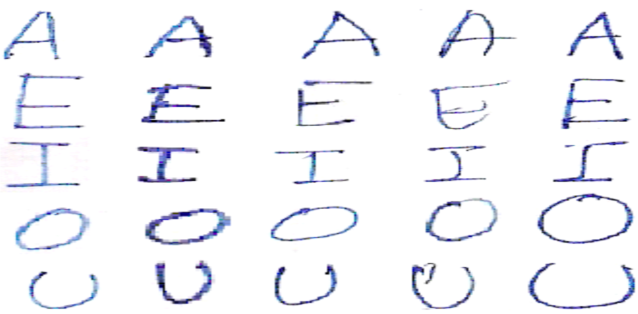

Fig.2: Scanned images of five different samples of handwritten English language vowels.

3. Results

The results presented in this section are demonstrating the implementation of Back Propagation learning algorithm and gradient descent learning for RBF network for handwritten English language vowels classification problem. Tables 2 and 3 are representing the results for handwritten English language vowels classification problem performed up to the maximum limit of 50000 iterations. Both results contain five different types of handwritten samples for each English vowel character.

The training has been performed in such a way that repetition of same input sample for a character cannot be happen simultaneously, i.e. if we have trained our network with a input sample of a character then next training cannot be happen with the other input sample of the same character. This input sample will appear for training after other samples of other characters training. It can observe from the results of tables that the RBFN has converged for 75 percent cases. The tables are showing some real numbers. These entries represents the error exit in the network after executing the simulation program up to 50000 iterations i.e. up to 50000 iterations the algorithm could not converge for a sample of a hand written English language vowels into the feed forward neural network.

Table 1: Results for Classification of Handwritten English Language Vowels using BP algorithm

S. No.	Characters	BP Epochs					
		Sample 1	Sample 1	Sample 2	Sample 3	Sample 4	Sample 5
1	**A**	550	72	33	0.2	605	0.1
2	**E**	0.2	115	0.4	89	867	1525
3	**I**	0.4	5594	867	9232	0.4	0.3
4	**O**	0.4	89	0.1	0.4	1759	3179
5	**U**	0.4	674	933	97	29	8398

Table 2: The Results for Classification of Handwritten English Language Vowels decent gradient withRBF network

S. No.	Characters	DG-RBF Epochs				
		Sample 1	Sample 2	Sample 3	Sample 4	Sample 5
1	**A**	4672	5134	0.2	0.1	0.3
2	**E**	97	0.4	3583	0.4	148
3	**I**	532	232	0.3	0.2	0.2
4	**O**	17248	0.4	0.1	2975	8282
5	**U**	5928	0.2	0.5	0.2	0.3

Conclusions

The results described in this paper exhibit the performance analysis of RGF network and BP algorithm. The implementation details for descent gradient learning for radial basis function neural network has been explored for the classification of the handwritten English language vowels classification problem. The comparative results for BP algorithm and RBF are represented. It has been also observed that the RBF network has also stuck in local minima of error for some of the cases. The reason for this observation is quite obvious, because there is no guarantee that RBFNN remains localized after the supervised learning and the adjustment of the basis function parameters with the supervised learning represents a non-linear optimization, which may lead to the local minimum of the error function. But the considered RBF neural network is well localized and it provides that an input is generating a significant activation in a small region. So that, the opportunity is getting stuck at local minima is small with respect to BP algorithm. Thus the number of cases for descent gradient RBF network to trap in local minimum is very low in respect of BP algorithm.

References

[1]. K. Fukushima and N. Wake, "*Handwritten alphanumeric character recognition by the neocognitron*," IEEE Trans. on Neural Networks, 2(3) 355-365 (1991).

[2]. Manish Mangal and Manu Pratap Singh, "*Analysis of Classification for the Multidimensional Parity-Bit-Checking Problem with Hybrid Evolutionary Feed-forward Neural Network*." Neurocomputing, Elsevier Science, 70 1511-1524 (2007).

[3]. D. Aha and R. Bankert, "*Cloud classification using error-correcting output codes*", Artificial Intelligence Applications: Natural Resources, Agriculture, and Environmental Science, 11(1) 13-28 (1997).

[4]. Y.L. Murphey, Y. Luo, "*Feature extraction for a multiple pattern classification neural network system*", IEEE International Conference on Pattern Recognition, (2002).

[5]. L. Bruzzone, D. F. Prieto and S. B. Serpico, "*A neural-statistical approach to multitemporal and multisource remote-sensing image classification*", IEEE Trans. Geosci. Remote Sensing, 37 1350-1359 (1999).

[6]. J.N. Hwang, S.Y. Kung, M. Niranjan, and J.C. Principe, "*The past, present, and future of neural networks for signal processing*'', IEEE Signal Processing Magazine, 14(6) 28-48 (1997).

[7]. C. Lee and D. A. Landgrebe, "*Decision boundary feature extraction for neural networks*," IEEE Trans. Neural Networks, 8(1) 75–83 (1997).

[8]. C. Apte, et al., "*Automated Learning of Decision Rules for Text Categorization*", ACM Transactions for Information Systems, 12 233-251 (1994).

[9]. Manish Mangal and Manu Pratap Singh, " *Handwritten English Vowels Recognition using Hybrid evolutionary Feed-Forward Neural Network*", Malaysian Journal of Computer Science, 19(2) 169 –187 (2006).

[10]. Y. Even-Zohar and D. Roth, "A sequential model for multi class classification", In EMNLP-2001, the SIGDAT. Conference on Empirical Methods in Natural Language Processing, 10-19 (2001).

[11]. Martin M. Anthony, Peter Bartlett, "*Learning in Neural Networks: Theoretical Foundations*", Cambridge University Press, New York, NY, (1999).

[12]. S. Haykin, "*Neural Networks",* A Comprehensive Foundation, Second Edition, Prentice-Hall, Inc., New Jersey, (1999).

[13]. T. P. Vogl, J. K. Mangis, W. T. Zink, A. K. Rigler, D. L. Alkon "Accelerating the Convergence of the Back Propagation Method", Biol. Cybernetics, 59 257-263 (1988)

[14]. R. A. Jacobs, "*Increased Rates of Convergence through Learning Rate Adaptation*", Neural Networks, 1 295-307 (1988).

[15]. X.H. Yu, G.A. Chen, and S.X. Cheng, "*Dynamic learning rate optimization of the backpropagation algorithm"*, IEEE Trans. Neural Network, 6 669 - 677 (1995).

[16]. Stanislaw Osowski, PiotrBojarczak, MaciejStodolski, "*Fast Second Order Learning Algorithm for Feedforward Multilayer Neural Networks and its Applications*", Neural Networks, 9(9) 1583-1596 (1996).

[17]. R. Battiti, "*First- and Second-Order Methods for Learning: Between Steepest Descent and Newton's Method*". Computing: archives for informatics and numerical computation, 4(2) 141-166 (1992).

[18]. S. Becker, & Y. Le Cun, "*Improving the convergence of the back-propagation learning with second order methods*", In D. S. Touretzky, G. E. Hinton, & T. J. Sejnowski (Eds.), Proceedings of the Connectionist Models Summer School. San Mateo, CA: Morgan Kaufmann, 29-37(1988).

[19]. M. F. Moller, "A *Scaled Conjugate Gradient Algorithm for Fast Supervised learning*", Neural Networks, 6 525-533(1993).

[20]. M.T. Hagan, and M. Menhaj, "*Training feed-forward networks with the Marquardt algorithm*", IEEE Transactions on Neural Networks, 5(6) 989-993(1999).

[21]. P. Christenson, A.Maurer&G.E.Miner, "handwritten recognition by neural network", http://csci.mrs.umn.edu/UMMCSciWiki/pub/CSci4555s04/InsertTeamNameHere/ handwritten.pdf, (2005).

[22]. M. J. D. Powell, "*Radial basis functions for multivariable interpolation: A review*", in *Algorithms for Approximation of Functions and Data*, J. C. Mason, M. G. Cox, Eds. Oxford, U.K.: Oxford Univ. Press, 143-167 (1987).

[23]. T. Poggio, and F. Girosi, "*Regularization algorithms for learning that are equivalent to multilayer networks*", *Science*, 247 978-982 (1990b).

[24]. M. T. Musavi , W. Ahmed , K. H. Chan , K. B. Faris , *D. M. Hummels, "On the training of radial basis function classifiers*", Neural Networks, 5(4) 595-603 (1992).

[25]. W. P. Vogt, "*Dictionary of Statistics and Methodology: A Nontechnical Guide for the Social Sciences",* Thousand Oaks: Sage (1993).