

PERFORMANCE ANALYSIS AND VALIDATION OF CLUSTERING ALGORITHMS USING SOFT COMPUTING TECHNIQUES

Bondu Venkateswarlu

Research Scholar, Department of CS&SE,
Andhra University College of Engineering, A.U,
Visakhapatnam, India.

Dr.GSV Prasad Raju

Professor in School of Distance Education,
Andhra University College of Engineering, A.U,
Visakhapatnam, India.

Abstract—The research explores the use of the Artificial Neural Networks models to predict the availability of the blood donors basing on blood group, age and location code. The training simulation results indicate the feed forward back propagation model two input neurons and two hidden units and one output unit produce best predictive results for neural network. By varying the simulation is train the neural network in the three performance functions namely mse, msereg and sse and the transfer functions TRANsig, logsig and purelin to validate the results. The results obtained after training the data under different scenarios mse gives better validation results. The errors obtained the training errors for mse is 0.90091, testing error is 0.037872 and validation is 0.65643.

Keywords—mse, msereg, sse, TRANsig, logsig and purelin

1. INTRODUCTION

Neural Network is the biological structure inspired by the working of human nervous system. Neural network is very popular due to its capability of generalization and resistance to the noisy and erroneous data. Neural Network is being applied widely on different application areas due to its learning ability i.e. capability to extract rules and learn from the data and create a network model which can be used for classification, pattern recognition and forecasting on the data. Most promising feature of the Neural network that other classification techniques do not possess is that it helps to simulate the network and create model which can be used further and applied on the new data which was not previously exposed to the network. A research is undergoing to improve the efficiency and accuracy of the modeling and training of the neural network architecture. The data set size and architecture of the network and the best suitable algorithm for the NN modeling because selection of these parameters depends on the nature of the data. In this work we find the values of different parameters suitable for this problem such that we obtain better and accurate results.

Neural Networks are connected with one another by groups of selected neurons. Neural Networks are functional circuits in the brain that process information and sending the outputs of the important activities to the body. Artificial Neural Networks (ANN) is a mapping network it is able to

compute functional relationship between its input and outputs. The neurons are composed of billions of interconnected processing elements; it is called the biological brain which transmits the information and strengthens the brain. Neural Networks are allowed to learn from example, and identify the patterns in noisy data and learn from mistakes and operate with incomplete data. Based on the evaluating capabilities of the human brain, Neural Networks attempt to overcome the limitations of traditional computers.

An Artificial Neural Networks is a model composed with several highly interconnected units called nodes or neurons. Each neuron performs the simple operation on an input produce an output that output is forwarded to the next node or neuron in the sequence. Artificial Neural Networks are used various engineering applications, science and the properties of the Artificial Neural Networks is approximately complex and non-linear equations makes it a useful tools in electronic analysis. Artificial Neural Networks are suitable for pattern recognition and classification takes due to their non-linear, non-parametric adaptive learning properties.

To recognize the specific blood group the system is develop using MAT Lab software using Neural Networks Tool. The multilayer feed forward neural network is the work horse of the Neural Networks software. This Neural Networks tool is used for both function fitting and pattern recognition problems, it can also use for prediction problems by using the addition of a tapped delay line.

2. LITERATURE REVIEW

The Neural Network architecture is most commonly used with back propagation multilayer feed forward network [2]. Back propagation neural network is best suited for various applications. Assemble and create the network and find out the number of hidden layers and train the network. Neural Network is able to determine state of the brain according MR image [1] Mat lab was included. Back propagation neural Network was designed and trained using NN tool in Mat lab, one hidden layer between input and output layers. The type of activation function was sigmoid, and then the activation function was replaced by

tan sigmoid function and then pure linear function. The Artificial Neural Network were formed by Mat lab software [6] so that the minimum error in information training and maximum correlation coefficient in data. The feed forward back propagation with TrainBR training function, LEARNGD adaptive learning function and SSE performance function were the last networks tried. Predicting the properties concrete is by using ANN, this process uses NN tool in Mat lab software to train, verify and test the network module.

Back propagation is most widely used Neural Network projects because it is easy to train and for its accuracy [5]. It consists of two facts, the first one generate the input pattern of the network and the another one to adjust the output through altering the weights of the network. Three layers model has been used for training and studying different attributes of the hidden neurons in the network. Different networks give different results with same training functions and adaptive learning functions having the same number of neurons. To increase the number of neurons of the network shows decrease mse. The adaptive learning function LEARNGDM results with minimum mse. TRAINLM shows best results in training, testing and validation data. If the data is increased the performance of the network also increases. Neural Network tool is best for training the data using other algorithms and functions.

The glass data classification and finds the impact of different Activation functions on the error obtained while training and testing of the neural network model created by the NN Tool provided by the MATLAB Toolbox [3]. Experiment was conducted on the Matlab NN tool with glass data it has been observed that LOGSIG function gives the minimum MSE and gives more accurate results in comparison to the other activation functions provided by MATLAB (NN Tool). The relation of the nature of the dataset and the activation functions on the error obtained from the training of neural network model. By observing the limitations and effect of the different parameters such as number of hidden layers, activation functions, nature of the data, adjustments of weights, size of data and many more on the network modeling we will be able to understand and develop an improved algorithm and data mining tool for neural network classification technique with more accurate results.

3. Steps for the neural network design process

A). Collect data B). Create the network C). Configure the network D). Initialize the weights and biases E). Train the network F). Validate the network G). Use the network

A. Collect data

Data is collected from various data sources, such as blood bank repositories and social network sites, blood donation sites, and various blood donation camps conducted by engineering colleges. We collect 5310 sample instances. The dataset is created in .csv file. The dataset is having 6 attributes, such as name of the donor, blood group, age, gender, location and district.

B. Create the network

To create the Neural Network, the easiest way is to create network functions and then how it is done is investigate by using to create a simple two-layer feed forward network. To create two-layer feed forward network by using the command feed forwardnet.

C. Configure the network

After Neural Network has been created, it can be configured. The configuration of the Neural Network consists of input and target data examining, setting the network's input and output sizes to match the data, and choosing settings for processing inputs and outputs that will enable best network performance. When the training function is called the network configuration is automatically done. However it can be done manually by using the configuration function to configure the network, issue the following commands.

```
p = -2:1:2;  
t = sin (pi*p/2);  
net1 = configure (net, p, t);
```

With this information, the configure function can set the network input and output sizes to match the data. The resulting network can then be configured with the configure command.

D. Initialize the weights and biases

After the Neural Network has been configured initialize the weights and biases of network parameters are turned then the network performance is optimized. When reinitialize the configuration of the network by using the command init. This function takes the network object as input and returns the object with all weights and biases are initialized. To initialize or reinitialize the network issue the following command.

```
net = init (net);
```

E. Train the network

After the network parameters are initialized then it is used for training. The multilayer feed forward network can be trained for non-linear regression or pattern reorganization. It requires a set of examples of proper network behavior-network inputs p and target outputs t. to optimize the network performance by define the network performance function is net.performFcn. The default performance function for feed forward networks is mean square error mse—the average squared error between the network outputs and the target outputs. Neural Networks has two different training concepts.

F. Validate the Network

After the training is complete check the performance and determine if any changes need to be made to the training process, network architecture on the datasets. First check the training record. The functions tr.trainInd, tr.valInd and tr.testInd contain the indices of the data points that were used in the training, validation and test sets,

respectively. The structure also keeps track of several variables during the course of training, such as the value of the performance function, the magnitude of the gradient, etc. You can use the training record to plot the performance progress by using the plotperf command.

G. Use the Network

After the network is trained and validated, the network object is used to calculate the network response to any input. If the output is different, depending on the state of random number generator when the network was initialized, each time Neural Network is trained, the result can be different due to different network parametric values and different divisions of data into training, validation and test sets. For the Same problem different networks gives different outputs for the same input. To ensure that a Neural Network of good accuracy has been found, retain several times.

4. EASE OF USE

A. Back propagation Network

Back propagation was created by the Window-Hoff learning rule to multiple-layer networks and nonlinear differentiable transfer functions. Inputs and the corresponding target outputs are used to train a network until it can approximate a function, associate input vectors with specific output vectors, or classify input vectors in an appropriate way. Networks with biases, a sigmoid layer, and a linear output layer are capable of approximating any function with a finite number of discontinuities.

The back propagation networks tend to give reasonable answers when presented with inputs that they have never seen. A new input leads to an output similar to the correct output for input vectors used in training that are similar to the new input being presented. This generalization property makes it possible to train a network on a representative set of input/target pairs and get good results without training the network on all possible input/output pairs. The multilayer feed forward network is most commonly used with the back propagation algorithm.

The neural network model developed to identify the number of neurons in the first layer is 'n' which is equal to dimensionality of the input pattern vectors. The network recognizes the pattern vector p as belonging class Oⁱ if the ith output of the network is high while all other outputs are low. Due to the choice of the Back propagation neural network, the input data is normalized before being presented to the Artificial Neural Network. The number of nodes in the hidden layer is calculated by using the following formula.

$$n = \frac{i+O}{2} + y^{0.5}$$

Where n= number of nodes in the hidden layer

I= number of input features

O=number of output features

Y=number of input patterns in the training set

Graphical description of training multi-layer neural network using BP algorithm

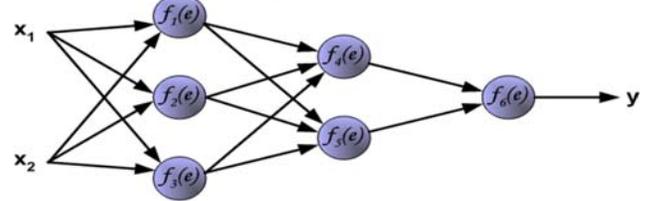


Figure 1: Graphical description of training multi-layer nn

The training data set consists of input signals (x_1 and x_2) assigned with corresponding target (desired output) z . It is an iterative process. In each iteration weights coefficients of nodes are modified using new data from training data set. After that determine output signals values for each neuron in each network layer.

5. Setting up the Experiment

Matlab is a data mining tool which provides Neural Network Toolbox for Neural Network modeling. Neural Network Toolbox consist of NN tool which helps to create a neural network model to train and test the data for classification of the data, and find out the number of hidden units, clustering and future forecasting. There are several other neural network tools but MATLAB NN Tool is popular because of large number of functionality support.

Here we present the results of the feed forward Back propagation technique in Neural Networks Tool in MAT lab with the synthetic dataset, in which the data is clustered according to blood group and location. Regression is another data mining technique which is based on supervised learning and is used to predict a continuous and numerical target. Based on training process, it estimates the value by comparing already known and predicted values.

The experiment was performed on the Neural Network Tool box provided by MAT Lab with blood donation dataset. We have created a neural network model by the help of the train dataset and varied the activation functions associated with the layers of the network and analyzed its effect on the type of blood group and performance functions mse, msereg and ssewith training functions transig, logsig and purelin obtained in the results. The feed forward back propagation algorithm is used in network type and LEARNGDM adaptive learning function with 2 layers and 10 neurons.

Fig 2: shows the neural network model created for blood donor data classification. This figure describes the architecture of the model created and how input parameters such as location code and age are connected to hidden layers and further hidden layers are connected to output layer such as blood group. In this work we create the network model with two input parameters are connected to two hidden layers and then the hidden layers are connected to one input layer. The input parameters are age and location code, the output parameter is blood group.

Simulate the network by using one input parameter and train the network by using two input parameters and one output parameter.

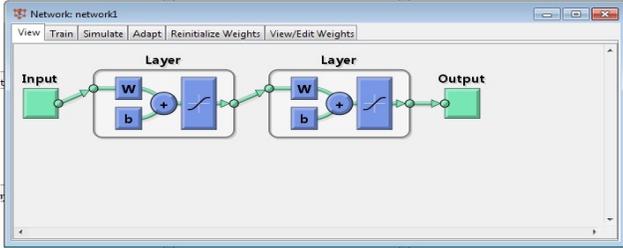


Figure 2: Neural Network Model created by the Matlab

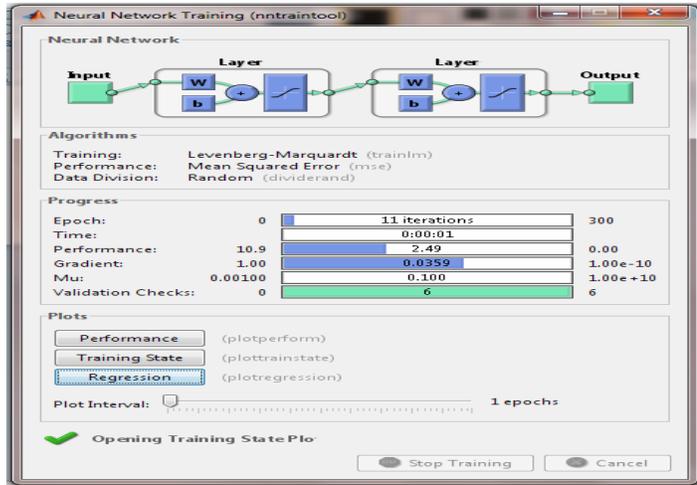


Figure 3: Neural Network model training by the Matlab

Fig 3: shows the detail of the type of algorithm applied, type of error selected for training the network i.e. MSE (Mean Square Error) or msereg or sse and we can also see the performance chart, training chart and the regression chart from it. This figure also gives us value of the error obtained while training the network. In this implementation feed forward back propagation algorithm is applied, training the network by using different performance functions such as mse, msereg and sse and to apply different transfer functions such as tansig, logsig and purelin. After that it gives the no of iterations to train the network and how many validation checks required completing the training and the plots of performance, training state and regression charts are available

6. RESULTS ANALYSIS

Table1: Results for various Transfer functions

Performance function	Transfer function		
	Logsig	Purelin	Tansig
Mse	0.60071	0.19282	0.66857
Msereg	0.59596	0.19283	0.57181
Sse	0.60071	0.19282	0.57199

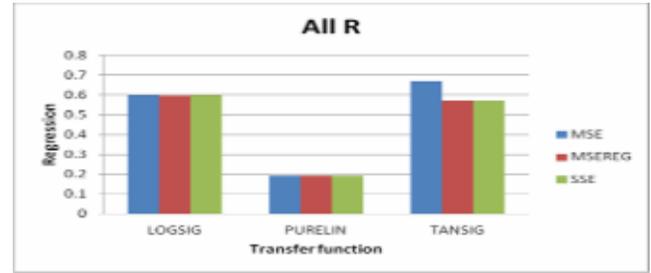


Figure 4: Results for various Transfer functions

Table2: Results for various Transfer functions for training R

Performance function	Transfer function		
	Logsig	Purelin	Tansig
Mse	0.89855	0.45461	0.90091
Msereg	0.88749	0.45461	0.83291
Sse	0.89855	0.45461	0.81125

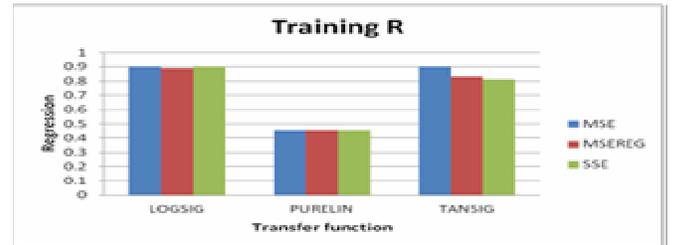


Figure 5: Results for various Transfer functions for training

Table3: Results for various Transfer functions for validation R

Performance function	Transfer function		
	Logsig	Purelin	Tansig
Mse	0.6552	0.51689	0.65643
Msereg	0.6755	0.51688	0.36574
Sse	0.6552	0.51689	0.040082

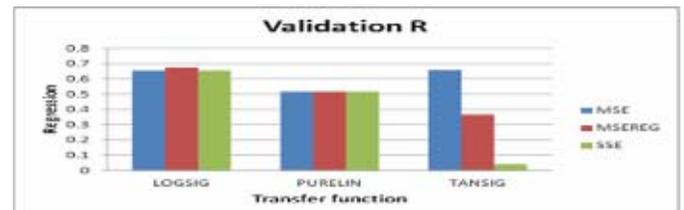


Figure 6: Results for various Transfer functions for validation R

Table4: Results for various Transfer functions test R

Performance function	Transfer function		
	Logsig	Purelin	Tansig
Mse	0.26608	0.23251	0.038782
Mserreg	0.26608	0.23251	0.12578
Sse	0.26608	0.23251	0.17851

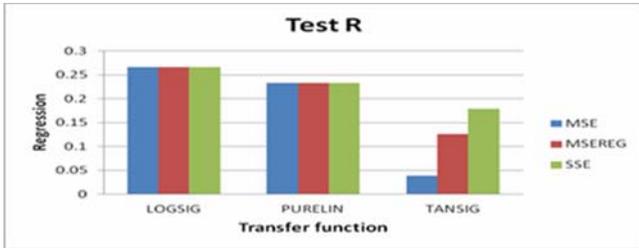


Figure 7: Results for various Transfer functions test R

The experiment Results are shown in table 1, table 2, table 3, and table 4. From table 1, it has been observed that the impact of different activation functions of transfer functions on the performance functions while training the neural network model. By varying the simulation is train the Neural Network Tool in the three performance functions namely mse, mserreg, and sse, and the transfer functions are Transig, Logsig, and Purelin to validate the results.

The Table 1 is All R(All Regression) results obtained under different scenarios of activation functions. For mse performance function with transig transfer function gives better validation results. That is 0.66857. For mserreg performance function with logsig transfer function gives better validation results. That is 0.59596. For sse performance function with logsig transfer function gives better validation results. That is 0.60071. Comparatively mse performance function with transig transfer function gives better validation All R results, i.e. 0.66857

The Table 2 is Training R (Training Regression) results obtained under different scenarios of activation functions. For mse performance function with transig transfer function gives better validation results. That is 0.90091. For mserreg performance function with logsig transfer function gives better validation results. That is 0.88749. For sse performance function with logsig transfer function gives better validation results. That is 0.89855. Comparatively mse performance function with transig transfer function gives better Training results, i.e. 0.90091

The Table 3 is Validation R (Validation Regression) results obtained under different scenarios of activation functions. For mse performance function with transig transfer function gives better validation results. That is 0.65643. For mserreg

performance function with logsig transfer function gives better validation results. That is 0.6755. For sse performance function with logsig transfer function gives better validation results. That is 0.6552. Comparatively mserreg performance function with logsig transfer function gives better validation results, i.e. 0.6755

The Table 4 is Test R (Test Regression) results obtained under different scenarios of activation functions. For mse performance function with logsig transfer function gives better validation results. That is 0.26608. For mserreg performance function with logsig transfer function gives better validation results. That is 0.26608. For sse performance function with logsig transfer function gives better validation results. That is 0.26608. Comparatively mse performance function with transig transfer function gives better Test R results, i.e. 0.26608. After considering the all the table and graphs of the results obtained after training the data different scenarios mse gives better validation results. The errors obtained the training errors for mse is 0.90091, testing error is 0.037872 and validation is 0.65643.

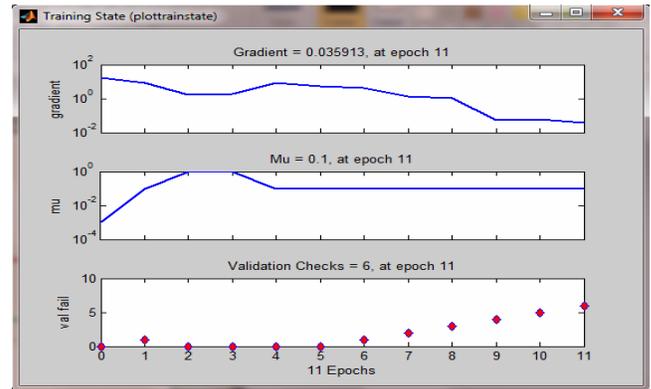


Figure 8: Training state chart

From the three activation functions such as logsig, transig and purelin with performance functions such as mse, mserreg, and sse provided by NN tool in Matlab produced different graphs such as training state, performance and regression. Fig 8 shows the training state graph with mse performance function and transig activation function the Gradient=0.035913 at epoch 11, mu=0.1 at epoch 11 and the validation checks =6 at epoch 11

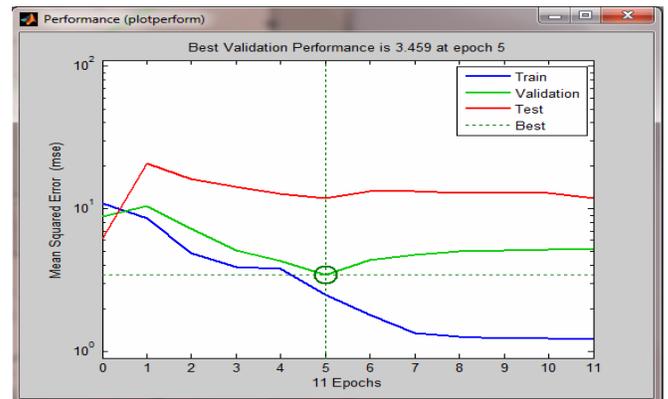


Figure 9: Performance chart

Fig 9 shows the performance validation graph with transig transfer function and mse error performance function. The best validation performance is 3.459 at epoch 5. The train graph and test graph having more difference in performance. The validation is almost similar to train graph. It is best fitted at epoch 5.

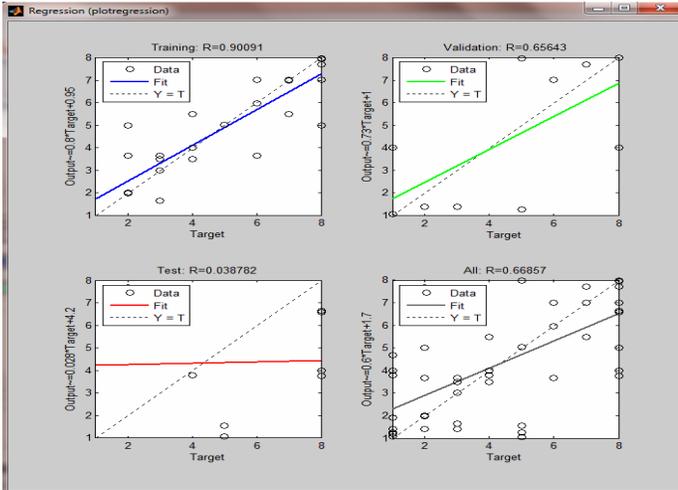


Figure 10: Regression chart for mse with TRANsig activation function

Fig 10 shows the regression analysis chart for mse error performance function with logsig activation function. Within this graph the training R=0.90091 and the test R=0.038782 and the validation is 0.66857. Finally the all R =0.66857.

In this research by varying the simulation is trained the NN tool in three performance functions and the three activation functions. The results obtained after training the data under different scenarios mse gives better validation results. The errors obtained the training for mse is 0.90091, testing error is 0.037872 and validation is 0.65643.

7. CONCLUSION AND FUTURE SCOPE

By applying the Artificial Neural Networks models to predict the availability of the blood donors basing on blood group, age and location. The training simulation results indicate the feedforward backpropagation model produce a best predictive results for neural network. By varying the simulation is train the NN in the three performance functions and the transfer functions to validate the predictive results. The results obtained after training the data under different scenarios mse gives better validation results. For the real time application, we will apply to validate the results and by applying ANFIS model and IOT etc.

References

- [1]. “Brain Tumor Detection Using Artificial Neural Networks” Eltaher Mohamed Hussein, Daliem Mahmoud Adam Mahmoud December 2012 Journal of Science and Technology Vol.13, No.2.
- [2]. “Crop Species Identification Using NN Tool” Anandkumar Patil, Lalitha Y S IJRET: International Journal of Research in Engineering and Technology ISSN: 2319-1163 | pISSN: 2321-7308 Volume: 04 Special Issue: 05 NCATECS 2015 | May 2015
- [3]. “Case Study On Classification Of Glass Using Network Tool In Matlab” Devika Chhachhiya, Amita Sharma, Manish Gupta , International Journal of Computer Applications (IJCA) (0975 – 8887) International Conference on Advances in Computer Engineering & Applications (ICACEA-2014) at IMSEC, GZB 11
- [4]. “Classification Of Pomegranate Diseases Based On Back Propagation Neural Network “ S. S. Sannakki, V. S. Rajpurohit International Journal of Advance Foundation and Research in Computer (IJAFRC) Volume 2, Special Issue (NCRTIT 2015), January 2015 ISSN 2348 – 4853 309 | © 2015, IJAFRC and NCRTIT – 2015
- [5]. “Rainfall Prediction Using Back -Propagation Feed Forward Network” Ankit Chaturvedi, International Journal of Computer Applications (0975 –8887) Volume 119 –No.4, June 2015
- [6]. “Using Feed-Forward Back Propagation (FFBP) Neural Networks For Compressive Strength Prediction Of Lightweight Concrete Made With Different Percentage Of Scoria Instead Of Sand “ Razavi S. V. , Jumaat M. Z. and Ahmed H. El-Shafie International Journal of the Physical Sciences Vol. 6(6), pp. 1325-1331, 18 March, 2011
- [7]. “an analysis of performance of feed forward neural network: using back propagation learning algorithm “ Sonam Tripathi, Rakesh Kumar Sharma International Journal of Science and Engineering Volume 1, Number 2 – 2013
- [8]. “A Review On Offline Leaf Recognition Using Neural Network”, Miss Needa Samreen I.Khan, Prof. Rajesh, B.Pandhare International Journal of Computer Science and Mobile Computing, Vol.4 Issue.1 January -2015, pg. 478-482
- [9]. “Data Mining to Improve Safety of Blood Donation Process “, Madhav Erraguntla , Peter Tomasulo, Kevin Land, Hany Kamel, Marjorie Bravo, Barbee Whitaker, Richard Mayer, Sarita Khaire 978-1-4799-2504-9/14 2014 IEEE
- [10]. “Data Mining Techniques and Their Implementation in Blood Bank Sector –A Review”, Ankit Bhardwaj, Arvind Sharma, V.K.

Shrivastava / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 Vol. 2, Issue4, July-August 2012, pp.1303-1309 1303.

- [11]. “An application to find spatial distribution of blood donors from blood bank information system”, B. G. Premasudha, Dr. Shivakumar Swamy & Dr. B. Suryanarayana Adiga, International Journal of Information Technology and Knowledge Management July-December 2009, Volume 2, No. 2, pp. 401-403
- [12]. “Prediction of Blood Donors Population using Data Mining Classification Technique”, Ritika1, Aman Paul2 Volume 4, Issue 6, June 2014 ISSN: 2277 128X International Journal of Advanced Research in Computer Science and Software Engineering

AUTHORS PROFILE

Bondu Venkateswarlu received the Master’s Degree in Computer Science and Systems Engineering from Andhra University College of Engineering, pursuing Ph.D in Department of Computer Science and Systems Engineering Andhra University College of Engineering, Andhra University, Visakhapatnam. He is life member of ISTE. His research area is Data Mining and Data Warehousing.

Prof G.S.V.Prasad Raju Professor in Computer Science, in School of Distance Education, Andhra University, Visakhapatnam. His area of specialization is E-Commerce, Internet Technologies.