Software Fault Prediction using Hybrid K-Means-Feed Forward Neural Network

Shyna Kakkar, Amanpreet Singh Dhanoa

Abstract- Software fault prediction is essential to increase the software reliabilty.More reliabilty provide better software quality.Faults are defects,bugs that results in failure and provide abnormal output.Software fault prediction technique can be used to detect faults.Repair of only faulty modules reduce overall development time significantly.Early fault prediction also reduces effort and development cost.Faults in software systems are major problems that need to be resolved.In this paper we have proposed hybrid approch K-Means-Feed Forward Neural Network.Proposed technique is compared against fuzzycmeans-feed forward neural network.Results of proposed technique are in terms of Accuracy,MAE,RMSE.Dataset used for experiment is NASA MDP PC1.

Keywords---Fuzzycmeans,K-Means,Feed forward neural network,NASA MDP.

I. INTRODUCTION

SOFTWARE quality and reliability are main concerns in modern era.It is widely accepted that software with defects lacks quality.Real time software application and complex software systems demands high quality.A software system contain many modules and any of these can contain faults.A fault prone software module is the one containing more number of expected faults.A fault is a defect, an error in source code that causes failures when executed.Early prediction of software fault at coding phase can result in decrease cost and effort for software development.So,it is better to categorize the software module in faulty / non - faulty module just after completing the coding phase.Software fault prediction uses historical and development data to identify fault in software.Various techniques have been applied for software fault prediction like partional clustering,hierarchical clustering,neural network,naive bayes,support vector machine,random forest tree,quad tree based kmeans,expectation maximization,genetic algorithm and many more.The disadvantage of the methods based on genetic algorithms is that genetically optimized fuzzy clustering method have high computational complexity and large time. The choice of application of a particular method generally depends on the type of output desired, the known performance of the method with particular type of data, available hardware and software facilities and size of the dataset.The main objective is described as follows:

Shyna Kakkar is a student of M.Tech. at Rayat&Bahra Engineering College,Institute of Engineering and Bio-Technology, Punjab, INDIA.

Amanpreet Singh Dhanoa is Assistant Professor at CSE Deptt,Rayat&Bahra Institute of Engineering and Bio-Technology, Punjab, INDIA.

• Find the best algorithms that can be used to predict faults in the software system

This paper is organized as follows:Section two describes the related work in software fault prediction. Section three describes clustering algorithms.Section four describes the steps to train neural network.Section five has proposed technique which shows the steps used in order to reach the objective and carry out the results. In the section six experimental setup in matlab is discussed.In Section seven results of the implementation are discussed.In the last section, on the basis of the results Conclusion is drawn.

II. RELATED WORK

Many techniques have been proposed for faults detection and classification.

In[1]Software fault prediction was done by fuzzy-genetic hybrid approach and compared with fuzzy c means clustering.Data clustered by fuzzycmeans is fed to genetic algorithm. For fault prediction.Accuracy obtained for hybrid approach is 99.14% on MDP PC1 dataset.

In [2]Inference system is fed to feed forward neural network with training data. In training stage neural network modifies the structure of fuzzy inference system to obtain high level of accuracy. It was found that adaptive fuzzy c means detects fault s better than fcm.

In[3]Comparitive study is conducted between quad tree based k means and quad tree based EM algorithm.It is givem that EM algorithm is more accurate than K-Means owing to lower error rates.when compared to K-Means.Using EM along with Quad Tree makes the classification process faster. With K-means, convergence is not guaranteed but EM guarantees elegant convergence.The proposed approach starts with a huge set (the popular Iris dataset).

In[4] In this paper fuzzycmeans output is fed to feed forward neural network.Researcher found that fuzzy-neural feed forward better perform than hierarchal clustering and neural network.

In[6]Researchers has conducted comparitive study on

fault prediction of fuzzy c means and KNN on NASA MDP PC1.It was noted that performance of KNN is better than fuzzy c means with accuracy of 88.3 1%.

In[8] C.catal has compared x means,fuzzy c means and kmeans on Turkish white-goods manufacturer datasets, AR3, AR4, and AR5.Performance of k-means and fuzzy c means was same and didnot improve but x-means shown good perfomance.

In [10]Software Fault Prediction Model is proposed using Clustering Algorithms xmeans compared against EM.Experiment conducted on AR3,AR4,AR5 dataset.x means

is better than k means as user has not to specify number of clusters and is faster than k means.

In[12]Comparitive study has been conducted on k means with different measures like Euclidean distance and Manhattan distance,Chebyshev distance function is used for analyzing the result of number of iterations, Overall Accuracy, Mean absolute error.

III. Clustering Algorithms

A.K-Means:It is also referred to as Lloyd's algorithm, particularly in the computer science community. k-means clustering is popular for cluster analysis in data mining.K-Means algorithm organizes objects into k clusters.The main idea is to define k centers, one for each cluster. The better choice is to place them as much as possible far away from each other. The next step is to take each point of given data set and associate it to the nearest cluster. When no point is left, the first step is completed. At this point we need to re-calculate k new centroids. After we have these k new centroids, a allocation of points has to be done between the same data set points and the nearest new center.k centers change their location step by step until no more changes are done or in other words centers do not move any more.Most of time,K-Means computationally faster than hierarchical clustering and k-Means produce tighter clusters than hierarchical clustering, especially if the clusters are globular.K-Means is applied in data compression,data modelling,expression analysis and other fields

Algorithmic steps for K-Means clustering

1) Set K – choose number of clusters, K.

2) Initialization – To choose k starting points which are used as initial cluster centroids.

3) Classification – To assign each datapoint to the cluster where euclidean distance small between cluster centroid and point

4) Centroid calculation – When each point in the data set is assigned to a cluster, it is needed to recalculate the new k centroids.

5) Convergence criteria – The steps of (iii) and (iv) require to be repeated until the centroids no longer move

Finally, this algorithm aims at minimizing an objective function, in this case a squared error function. The objective function

$$j = \sum_{j=1}^{k} \sum_{i=1}^{n} \| x_i^j - c_j \|^2$$

Where $\| x_i^j - c_j \|^2$ is a chosen distance measure between a data point and the cluster centre , is an indicator of the distance of the n data points from their respective cluster centers.Different distance functions used are:

-Euclidean distance

-Manhattan distance

-City block distance

B.Fuzzycmeans:In fuzzy cmeans clustering, every point has a degree of belonging to clusters, as in fuzzy logic, rather than belonging completely to just one cluster.Thus, points on the edge of a cluster, may be in the cluster to a lesser degree than points in the center of cluster. Any point x has a set of coefficients giving the degree of being in the kth cluster wk(x). With fuzzy c-means, the centroid of a cluster is the mean of all points, weighted by their degree of belonging to the cluster. Fuzzy c-means has been a very important tool for image processing in clustering objects in an image. The existence of a point in more than one cluster depends on the fuzzification value defined by the user in the range of [0, 1] which determines the degree of fuzziness in the cluster Steps in fuzzycmeans are:

1.Initialize $U=[u_{ij}]$ matrix, $U^{(0)}$

2.At k-step: calculate the centers vectors $C^{(k)}=[c_j]$ with $U^{(k)}$

$$c_j = \frac{\sum_{i=1}^{N} \mu_{ij}^{m} * x_i}{\sum_{i=1}^{N} \mu_{ij}^{m}}$$

3.Update $U^{(k)}$, $U^{(k+1)}$

$$\mu_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{\| x_i - c_i \|}{\| x_i - c_i \|} \right)^{\frac{2}{m-1}}}$$

4.If $\| U^{(k+1)} - U^{(k)} \| < f$ then STOP; otherwise return to step

*C.Feed Forward Neural Network:*The feedforward neural network is the simplest type of artificial neural network.In this network, the information moves in only one direction, forward, from the input nodes, through the hidden nodes to the output nodes. There are no cycles or loops in the network.Feed-forward networks have been applied successfully in such diverse fields as speech recognition, financial prediction, image compression, medical diagnosis.Each layer has number of neurons called as nodes.Each node is connected to one or more other nodes by real-valued weights, but not to nodes in the same layer.Feed-forward nets are generally implemented with an additional node - called the bias unit - in all layers except the output layer.The output y of each (non-input) node in the network (for a given pattern p) is simply the weighted sum of its inputs.

$$a_{i,p} = \sum_{p} w_{ij} y_{i,p}$$

$$y_{i,p} = f(a_{i,p})$$

Activation function f(x), which is required to be both monotonic and differentiable, is typically the sigmoid or logistic function, given by

$$f(x) = \frac{1}{1+e^{-x}}$$

Description of each layer of feed forward neural network is given as under:

1) *Input layer*: It consists of a set of neurons equal to number of input variables that receive inputs. In this layer, there is no activation function,no processing of input variables.

2) *Hidden Layer* :There can be a number of hidden layers between the input and output layers. The number of neurons in a hidden layer is chosen carefully to limit the complexity and to achieve better accuracy.

*3)Output Layer:*The output layer provides neural network output.The number of neurons in the output layer is equal to the number of output variables. The neurons in this layer receive their input from the preceding layer or last hidden layer in the network.

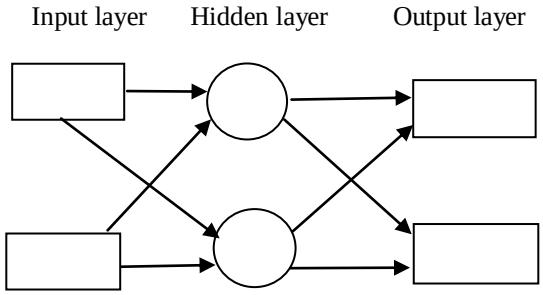

Fig. 1 Feed-forward neural network

IV. Training Feed Forward Neural Network

The following steps will be followed to train a Neural Network:

- Load the data
- Divide data into Training, Validation and Test data
- Set number of hidden neurons
- Training is accomplished by providing neural input and comparing the results with a set of target outputs.
- If there is a difference between the actual and target outputs, the weights are adjusted to produce a set of outputs closer to the target values.
- Network weights are determined by adding an error correction value to the old weight.
- The amount of correction is determined
- This Training procedure is repeated until the network performance no longer improves.
- If the network is successfully trained, it can then be given new sets of input.

V. Proposed Methodology

A.Find the structural code and requirement attributes

The first step is to find the structural code and requirement attributes of software systems i.e. software metrics.Data set NASA's MDP (Metric Data Program) data repository, PC1 dataset is available on http://promise.site.uottawa.ca/SERepository containing 1109 modules.Data is from flight software for earth orbiting satellite.

Class Distribution: the class value (defects) is discrete

false: 77 = 6.94%

true: 1032 = 93.05

B.Select the suitable metric values as representation of statement

The Suitable metric values used are fault and without fault attributes, we set these values in database create in MATLAB R2010 A as 0 and 1. Means 0 for data with fault and 1 for data without fault. Each data set contains twenty-one software metrics, which describe product's size, complexity and some structural properties.

Number of attributes: 22 (5 different lines of code measure, 3 McCabe metrics, 4 base Halstead measures, 8 derived Halstead measures, a branch-count, and 1 goal field)

TABLE I
ATTRIBUTES OF DATASET

loc	McCabe's line count of code
V(g)	McCabe "cyclomatic complexity"
Ev(g)	McCabe "essential complexity"
Iv(g)	McCabe "design complexity"
n	Halstead total operators + operands
v	Halstead "volume"
l	Halstead "program length"
d	Halstead "difficulty"
i	Halstead "intelligence"
e	Halstead "effort"
b	Halstead
t	Halstead's time estimator
lOCode	Halstead's line count
lOComment	Halstead's count of lines of comments
lOBlank	Halstead's count of blank lines
lOCodeAndComment	numeric
uniq_Op	unique operators
uniq_Opnd	unique operands
total_Op	total operators
total_Opnd	total operands
branchCount	numeric % of the flow graph
defects	{True,false}

C.Hybrid K-Means-Feed Forward Neural Network Approach:

In our proposed hybrid approach,firstly the dataset is normalised.The normalized value of e_i for variable E in the i^{th} row is calculated as:

$$Normalised(e_i) = \frac{e_i - E_{\min}}{E_{\max} - E_{\min}} \qquad (1)$$

E_{min} = the minimum value for variable E

E_{max} = the maximum value for variable E

Normalised dataset is then passed through k-means clustering algorithm block based on attributes(true/false attributes).Output obtained from k-means will serve as target for feed forward neural network.K-Means output will act as supervised classifier for feed forward neural network.Feed forward neural network after training classify faults as non-faulty modules and faulty modules.Here,Feed forward neural network uses back propagation algorithm for classification.Backpropagation, an abbreviation for "backward propagation of errors", is a common method of training artificial neural networks used in conjunction with an optimization method.

Output is obtained from output layer.Error in feed forward neural network is difference between actual output and predicted output.

Identified faulty modules can be treated to remove faults and reduce cost and effort in development phase.

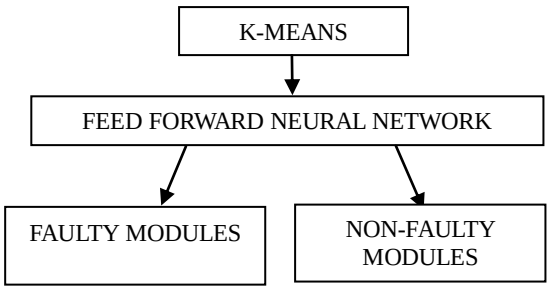

Fig.2 K-Means-Feed Forward Neural Network

D.Comparision between K-Means and Fuzzycmeans with Feed Forward Neural Network:

The time complexity of the K-Means algorithm is O(ncdi) and the time complexity of FCM algorithm is O(ndc2i).Because k-means just needs to do a distance calculation, whereas fuzzy c means calculate inverse-distance weighting.K-Means is an exclusive clustering algorithm, Fuzzy C-means is an overlapping clustering algorithm.In kmeans,the data are grouped in an exclusive way, so that if a certain data belongs to a definite cluster then it could not be included in another cluster.On the contrary fuzzycmeans, the overlapping clustering, uses fuzzy sets to cluster data, so that each point may belong to two or more clusters with different degrees of membership.

K-Means algorithm is better than FCM algorithm.Fuzzycmeans requires more computation time than K-Means because of the fuzzy measures calculations involvement in the algorithm.K-Means also produces better accurate results in comparision to Fuzzycmeans.

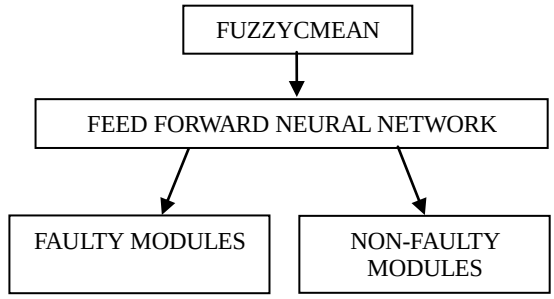

Fig.3 Fuzzycmean-Feed Forward Neural Network

VI. Experimental Setup

TABLE II

Number of hidden neurons	20
Epochs	100
Training algorithm	Levenberg Marquardt
Training goal	1e-5
Neural network	Feed-Forward Neural Network

VII. Results

In this study, training and testing methodology is being used, The NASA MDP dataset named PC1 is used in experiment. Performance of hybrid approach is measured in terms of:

-Accuracy

-Mean Absolute Error(MAE)

-Root Mean Square Error(RMSE)

$$MAE = \frac{1}{n}\sum_{j=1}^{n} | y_j - f_j | \qquad (2)$$

$$RMSE = \sqrt{\frac{\sum_{j=1}^{n}(y_j - f_j)^2}{n}} \qquad (3)$$

Where n is total instances, y_j is actual value and f_j is predicted value

Mean absolute error, MAE is the average of the difference between predicted and actual value in all test cases. The root mean-squared error i.e. RMSE is simply the square root of the mean-squared-error. The root mean-squared error gives the error value as the same dimensionality as the actual and predicted values.

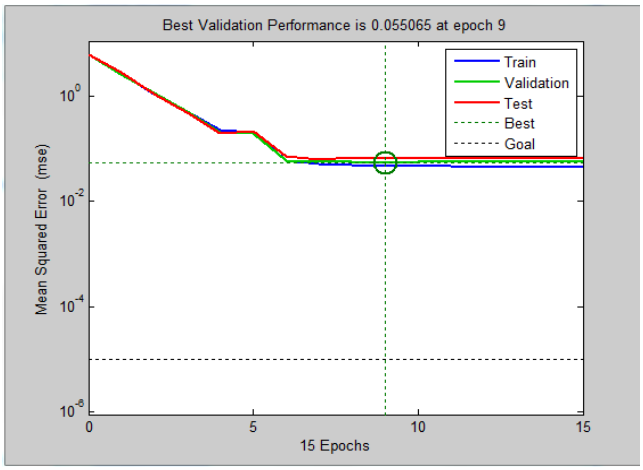

Fig.3 Performence Plot

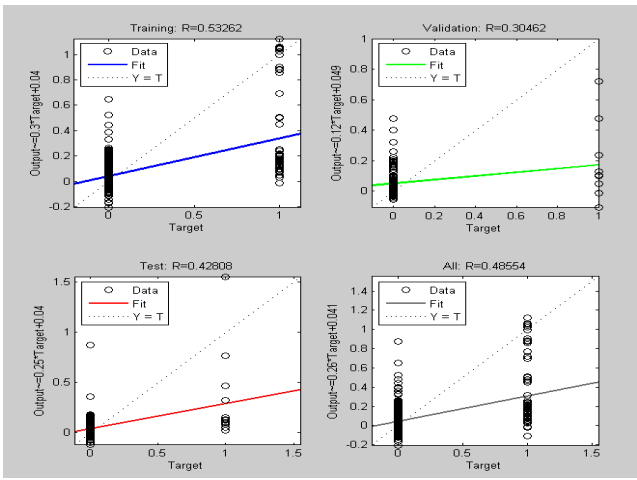

Fig. 4 Training Plot

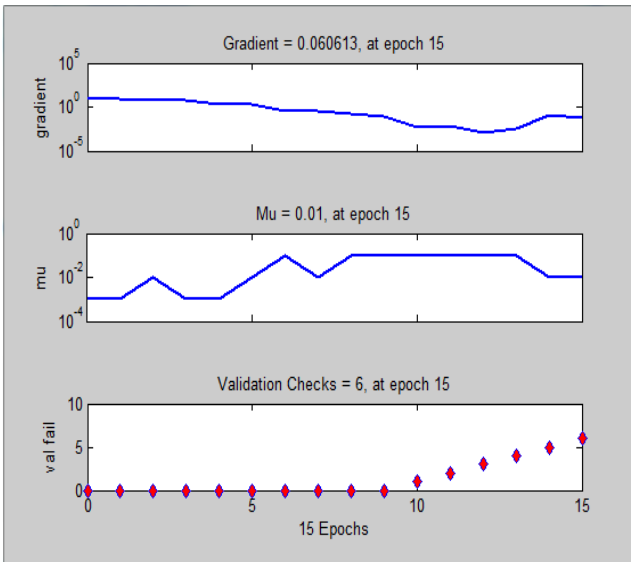

Fig. 5 Regression Plot

TABLE III
PERFORMANCE RESULTS OF DIFFERENT K-MEANS-FEED FORWARD NEURAL NETWORK AND FUZZYCMEANS-FEED FORWARD NEURAL NETWORK ALGORITHMS

TECHNIQUE	ACCURACY %	MAE	RMSE
K-Means-Feed Forward Neural Network	94	0.056	0.24
Fuzzycmeans-Feed Forward Neural Network	87	0.13	0.0194

VIII. Conclusion

In this paper, we have proposed hybrid approach to predict faults in software systems.K-Means-feedforward neural network has better accuracy than fuzzycmeans-feed forward neural network.Proposed approach provide less mean absolute error.The fault prediction in software is significant because it can help in directing test effort, reducing cost, and increasing quality of software and its reliability.

REFERENCES

[1] Saurabh Bhattacharya ,Dr.Sourabh Rungta, Naresh Kar, "*Software Fault Prediction using Fuzzy Clustering & Genetic Algorithm*"Volume 2, Issue 5, December 2013,IJDACR

[2] Pushpavathi T.P, Suma V, Ramaswamy V "*Analysis of Software Fault and Defect Prediction by Fuzzy C-Means Clustering and Adaptive Neuro Fuzzy C-Means Clustering*"International Journal of Scientific & Engineering Research, Volume 5, Issue 9, September-2014

[3] Swapna,M.Patil,R.V.Argiddi "*Comparision between Quad tree based K-Means and EM AlgorithmforFault Prediction*"(IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5(6) , 2014, 7984-7988

[4] Kriti Purswani,Pankaj Dalal,Dr. Avinash Panwar,Kushagra Dashora "*Software Fault Prediction Using Fuzzy C-Means Clustering and Feed Forward Neural Network*"Volume 2, Issue 1, July 2013,IJDACR

[5] Karwan Qader, and Mo Adda,"*A Survey of Network Faults Classification Using Clustering Techniques*",International JournalofAdvanced Research in Computer and Communication Engineering Vol. 2, Issue 10, October 2013

[6] **A**nil Kumar Singh,Rajkumar Goel, Pankaj Kumar "*Comparative Analysis of Accuracy Prediction using Fuzzy C-Means and KNN Clasiffier*",Volume 2, Issue 7, February 2014 ,IJDACR

[7] **A**nil Kumar Singh,Rajkumar Goel, Pankaj Kumar "*Fault Prediction using Hybrid Fuzzy C-Means with Genetic Algorithm and KNN Classifier*",Volume 2, Issue 8, March 2014,IJDACR

[8] C. Catal, U. Sevim and B. Diri, "*Software fault prediction of unlabeled program modules*" Proc. of WCE,(2009).

[9] P. S. Bishnu and V. Bhattacherjee, "*Software fault prediction using quad tree-based k-means clustering algorithm*", IEEE Trans. Knowledge and Data Eng., vol. 24, no. 6, (2012), pp. 1146-1150.

[10] Mikyeong Park and Euyseok Hong "*Software Fault Prediction Model using Clustering Algorithms Determining the Number of Clusters Automatically*",International Journal of Software Engineering and Its Applications Vol.8, No.7 (2014), pp.199-204

[11] Gayathri M, A. Sudha "*Software Defect Prediction System using Multilayer Perceptron Neural Network with Data Mining*"Volume-3,Issue-2,May2014,IJRE

[12] KahkashanKouser1, Sunita "*A comparative study of K Means Algorithmby Different* Distance Measure"Vol. 1, Issue 9,November 2013

[13] Aditi Sanyal, Balraj Singh, "*A Systematic Literature Survey on Various Techniques for Software Fault Prediction*",Volume 4, Issue 1, January

2014

[14] Jyoti nagpal,Dr.Ajay Khuteta "*Software Fault Estimation using Fuzzy C-Means and Neuro-Fuzzy Classification*"Volume 2, Issue 10, May 2014,IJDACR

[15] Kritika Gupta,Sandeep Kang,Dr. Parvinder S. Sandhu "*Comparison of Resilient Backpropagation & Fuzzy Clustering Based Approach for Prediction of Level of Severity of Faults in Software Systems*"

[16] http://promise.site.uottawa.ca/SERepository

[17] Tejwant Singh, Mr. Manish Mahajan "*Performance Comparison of Fuzzy C Means with Respect to Other Clustering Algorithm"*Volume 4, Issue 5, May 2014,IJARCSSE