Chapter 6

Analysis of performance on empirical data using a combination of Genetic Algorithm, Fuzzy C- Means and Random Forest

6.1 Introduction

In the previous chapter, we analyzed the characteristics of software project process data and proposed a novel method of hybrid of RF and FCM for the software project success prediction. The experimental result shows that the combination of RF and modified FCM method is appropriate for the prediction of software project success. This combination not only has good classification accuracy but also results in relatively simple and interpretable model.

In the previous chapter, we have also empirically evaluated and compared the performance of modified fuzzy c-means clustering technique and adaptive neuro-fuzzy c-means clustering for software defect prediction. The proposed adaptive neuro-fuzzy c-means clustering based prediction technique is expected to result in good accuracy and achieve better performance. It is also observed that the adaptive neuro-fuzzy c-means clustering can satisfactorily find/defect-free projects. Following are some of the observations we have made.

- Most of the studies have defined their own metric suite which has been used for carrying out the analysis.
- Diverse categories of methods are available to predict the most accurate model such as machine learning methods, statistical methods etc., and the trend is however shifting from the traditional statistical methods to the machine learning methods.
- Machine learning is widely used in new bodies of research to predict defect prone classes. Results of various studies also show that better results are obtained with machine learning when compared to statistical methods.
- Researchers have used different types of datasets notable among them being public datasets, commercial datasets, open source or students/university datasets, PROMISE and NASA repositories.

Empirical datasets are found to be rarely used in the studies. Single clustering is used on empirical datasets in chapter 3. In chapter 4, cascading of clustering and classification is applied on empirical datasets. In chapter 5, modified FCM has been used in combination with other techniques. In this chapter modified RF is used which is tested on empirical dataset and
validated using NASA PC1 dataset. We will now analyse the performance on empirical data and NASA data sets using a combination of Genetic Algorithm (GA), Fuzzy C- Means (FCM) and Random Forest (RF).

### 6.2 Proposed Classification Procedure

In the earlier chapters, we have discussed about the collection of empirical data set, its pre-processing etc. In this section, we will present a methodology for predicting defect prone modules using genetic algorithm, modified fuzzy c-means clustering and random forests algorithm on NASA PC1 dataset as shown in figure 6.1.

As the tree keeps on growing its accuracy is also improving by voting on decision classification. To apply the methodology to NASA public domain defect data sets (i.e. PC1). Though these data sets might vary in size, also contains less number of defects in the learning data set. The goal is maximization of overall accuracy, then learning from such data will usually result in a biased classifier which means if more number of data samples fall into the defect free class.

The limitations identified in Random Forest classifier (RF) are:

1. Learning from imbalanced data can cause the classifier to be biased
2. Splitting criteria of a single classification Solution to these limitations is to integrate random forest (RF) classifier with genetic algorithm (GA) and fuzzy c-means clustering (FCM).

6.3 Modified Random Forest Classifier

A classification tree is constructing by the process of recursive partition. Partitioning is by consecutively split data sets into subsets. Each of which is equal to the terminal node. The root node contains whole data sets and remaining node data are evaluated individually and select the suitable split. The correctness of the split is calculated by impurity functions. These impurity functions measure the separations of the classes.

![Figure 6.2 Construction of a Random Forest](image)

The prediction models are faced with class imbalance problem. It occurs when the classes in C have a dramatically different number of representatives in the training dataset and/or have very different statistical distributions.

Learning from imbalanced data can cause the classifier to be biased. Such an imbalance results over-represented in training than the other classes. Classes containing relatively few cases can be largely ignored by the learning algorithms. For instance, a binary classification problem such as the identification of defect prone modules may be represented by 1,000 cases 950 of which are negative cases (majority class, defect free modules) and 50 are positive cases (minority class, defect modules). Even if the model classified all the cases as negative and misclassified all the positive cases, the overall accuracy becomes 95%. In many
situations, the minority class is the subject of one’s major interest. In practice, it is required to achieve a lower minority classification error even at the cost of a higher majority class error rate. For example, suppose the goal of identifying defective modules early in software development is exposing them to a more rigorous set of verification and validation activities. The imperative is to identify as many potentially defect modules as possible. If non defective modules are misclassified as defective ones, the verification process will increase its cost as some modules are unnecessarily analysed. But the consequence of misclassifying defective software as non-defective may lead to highly undesirable outcomes.

6.4 Implementing the integrated approach of Genetic algorithm based modified FCM and modified Random forest classifier

The proposed methodology is implemented using MATLAB. Implementing the model and testing the performance of the model are carried out using the following criteria:

- Perform the training of the dataset.
- Test it on the basis of error values MAE (Mean Absolute Error) and RMSE (Root Mean Square Error) and efficiency parameters like accuracy and net reliability in percentage.

Mean Absolute Error (MAE):

The average difference between predicted and actual values is called mean absolute error (MAE). The formula for calculating MAE is given below:

$$\frac{|a_1 - c_1| + |a_2 - c_2| + \cdots + |a_n - c_n|}{n}$$  \hspace{1cm} (6.1)

Root Mean Squared Error (RMSE):

RMSE is a frequently used measure of difference between values predicted by an approach and actually the values observed. It is shown in equation 6.2.

$$\sqrt{\frac{(a_1 - c_1)^2 + (a_2 - c_2)^2 + \cdots + (a_n - c_n)^2}{n}}$$  \hspace{1cm} (6.2)

After computing MAE and RMSE values, the comparisons are made on measure the small values of MAE and RMSE error values. Mean absolute error is chosen as the standard error. Choosing lower value of mean absolute error is considered to be the best defect prediction technique. Sections 6.4.1 and 6.4.2 explain the steps involved while collecting and selecting
the suitable attributes. Section 6.4.3 describes the combination of Genetic Algorithm (GA), Fuzzy C- Means (FCM) and Random Forest (RF).

6.4.1 Find the structural code and requirement code attributes

The first step is to find the structural code and requirement attributes of software systems. Data sets are taken from NASA’s MDP (Metric Data Program) data repository, [online] Available: http://mdp.ivv.nasa.gov.in named as PC1 dataset. This is collected from flight software from an earth orbiting satellite coded in C programming language, having 1107 modules and requirements specified in 109. The data set PC1 has 320 requirements available and all of them are associated with program modules. PC1 data set have modules with least percentage of defect modules.

6.4.2 Representation of suitable metric values

Suitable metric values used are defective (value 0) and non defective (value 1). The metrics in these datasets (NASA MDP dataset) describe projects which vary in size, complexity, programming languages and development processes etc.

When reporting a defect prediction modelling experiment, it is important to describe the characteristics of the datasets. Each data set contains twenty-one software metrics which describe product’s size, complexity and some structural properties. For the purpose of our work, only defect and defect free attributes are used to classify the selected NASA MDP PC1 dataset. The product metrics and product module metrics which are available in the dataset and which can be used are as follows.

Product requirement metrics:

- Module
- Action
- Conditional
- Continuance
- Imperative
- Option
- Risk_Level
- Source
- Weak_Phrase
Product module metrics:

- Module
- Loc_Blank
- Branch_Count
- LOC_Code_and_Comment
- LOC_Comments
- Condition_Count
- Cyclomatic_complexity
- Cyclomatic_Density
- Decision_Count
- Edge_Count
- Essential_Complexity
- Essential_Density
- LOC_Executeable
- Parameter_Count
- Global_Data_Complexity
- Global_Data_Density
- Halstead_Length
- Halstead_Prog_Time
- Halstead_Volume
- Normalized_Cyclomatic_Complexity
- Num_Operands
- Num_Operators
- Num.Unique_Operands
- Num.Unique_Operators
- Number_Of_Lines
- Pathological_Complexity
- LOC_Total
6.4.3 Combination of Genetic Algorithm, Fuzzy C- Means and Random Forest (GA_FCM_RF)

![Cluster separations in silhouette plot for clustering](image)

Figure 6.3 Cluster separations in silhouette plot for clustering

From the silhouette plot, one can see that most of the points having more value in the third cluster show partition from neighbouring cluster.

![Accuracy rate of classification for a dataset in semilogy plot](image)

Figure 6.4 Accuracy rate of classification for a dataset in semilogy plot

The accuracy of a measurement is the degree of closeness of the measurement of a quantity to that actual (true) value. Semilogy plots data with logarithmic scale for the y-axis as shown in figure 6.4. The NASA MDP dataset named PC1 is used in this. Subsequently, Fuzzy C-means
Clustering based classification approach is applied on the same project data set and the finally classified it as defective or defect free. Classification is based on the values of accuracy which are MAE and RMSE. Genetic algorithm considers fuzzy c-means clustering as a fitness function which needs to be minimized. Accordingly, we give a range (upper bound and lower bound) to genetic algorithm in order to select the number of clusters which will result in optimum classification. Upper bound is considered to be 8 and lower bound is 2 in our work.

![Data Input (PC1 database with attributes)](image_url)

Figure 6.5 Input NASA pc1 dataset with attributes

Figure 6.5 shows the plotting of data set with 100 modules. Though the input data is having 1107 modules, in order to reduce complexity in calculations, we have narrowed down to take only 100 modules for training purpose. Original dataset contains 22 columns out of which columns 1 to 21 show the data numerical values and the last column shows the categorical value which indicates whether the data is having defect or not.
Figure 6.6 NASA pc1 dataset attributes without defect

After loading the data set, the data without defects are classified as shown in figure 6.6. Figure 6.7 shows the plotting of defective ones. X axis shows the data modules with defect and Y axis shows data values stored in 21 columns of input dataset.

Figure 6.7 NASA pc1 dataset attributes with defect

Figure 6.8 shows the variation in objective function with respect to iterations count in random forest. X and Y are the coordinates in the unit square to present the nodes of the tree so as to make a quality representation.
6.5 Results and Discussions

In this section, we have discussed Genetic Algorithm based modified fuzzy c-means along with modified random forest (GA_FCM_RF) approach. Data validation is also discussed. We have implemented genetic algorithm based fuzzy c-means clustering along with random forest algorithm for predicting defect prone modules (GA_FCM_RF) and have applied it to NASA public domain defect data set PC1. The performance measurements namely accuracy
and reliability are calculated using the equations 4.1, 4.2 and 4.3 are shown in chapter 4. Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) are calculated from the equations 6.1 and 6.2 as mentioned in section 6.4.

Figures 6.10 to 6.13 show the defect prediction results of FCM, GA_FCM, GA_FCM_RF and adaptive neuro fuzzy classifiers respectively. The figures also show the values of the four performance measurements namely accuracy, reliability, MAE and RMSE.

Figure 6.10 Defect predictions by FCM
Figure 6.11 Defect prediction by GA_FCM

Figure 6.12 Defect prediction by GA_FCM_RF
Figure 6.13 Defect prediction by Adaptive Neuro Fuzzy classifier

Table 6.1 shows the results of prediction accuracy validation with NASA PC1 data set. Table 6.2 shows the Prediction accuracy for empirical data set with GA, modified FCM and Modified RF.

Table 6.1 Prediction accuracy comparisons of NASA PC1 data set with GA_modified FCM_Modified RF

<table>
<thead>
<tr>
<th>Methods</th>
<th>Performance</th>
<th>Accuracy %</th>
<th>MAE</th>
<th>RMSE</th>
<th>Reliability %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuzzy c-means Clustering</td>
<td></td>
<td>76.200</td>
<td>0.2800</td>
<td>0.1088</td>
<td>56.000</td>
</tr>
<tr>
<td>Adaptive Neuro Fuzzy based classifier</td>
<td></td>
<td>98.126</td>
<td>0.0210</td>
<td>0.01563</td>
<td>48.750</td>
</tr>
<tr>
<td>GA based FCM</td>
<td></td>
<td>90.126</td>
<td>0.0200</td>
<td>0.04950</td>
<td>43.760</td>
</tr>
<tr>
<td>GA-FCM based Random Forest classifier</td>
<td></td>
<td>98.237</td>
<td>0.03000</td>
<td>0.00092</td>
<td>92.625</td>
</tr>
</tbody>
</table>
Figure 6.14 Reliability and accuracy of NASA PC1 data set.

Figure 14 shows the validation results of fuzzy C-means clustering, GA based FCM, GA-FCM based random forest classifier and adaptive neuro fuzzy classifier. GA-FCM based random forest classifier which shows an accuracy of 98.237% performs better when compared to the rest.

Table 6.2 Prediction accuracy comparisons of empirical data set with GA_modified FCM_Modified RF

<table>
<thead>
<tr>
<th>Methods</th>
<th>Performance</th>
<th>Accuracy %</th>
<th>MAE</th>
<th>RMSE</th>
<th>Reliability %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuzzy C-means clustering</td>
<td></td>
<td>71.6561</td>
<td>0.0352</td>
<td>0.1200</td>
<td>75.214</td>
</tr>
<tr>
<td>Adaptive Neuro Fuzzy based classifier</td>
<td></td>
<td>93.125</td>
<td>0.1200</td>
<td>0.01636</td>
<td>77.55</td>
</tr>
<tr>
<td>GA based FCM</td>
<td></td>
<td>90.1253</td>
<td>0.200</td>
<td>0.050</td>
<td>79.855</td>
</tr>
</tbody>
</table>
GA-FCM based Random forest classifier 95.237 0.0300 0.00098 90.625

Figure 6.15  Reliability and accuracy of emperical data set

Figure 15 shows the experimental results on empirical datasets. Also reliability and accuracy result comparisons of fuzzy C-means clustering, GA based FCM, GA-FCM based random forest classifier and adaptive neuro fuzzy classifier is given. GA-FCM based random forest classifier shows an accuracy of 96.938% which appears to be comparatively better when validated with NASA PC1 dataset as shown in figure 14. Though the adaptive neuro fuzzy classifier is having more than 90% accuracy in tables 6.1 and 6.2, its reliability is less when compared to GA-FCM_RF.

As far as the overall performance of the system is concerned, following observations from above experiments can be made:

- Random forest algorithm is always one of the best classifiers if the user-specific voting thresholds approximate the proportion of defect-prone modules in the project’s training set.

- The second observation relates to the classification performance of two variants of the random forest algorithm. Balanced random forests provide a moderate performance increase over the traditional random forest algorithm. Since the improvement is only
moderate, software quality engineers will have to decide whether it is worth additional effort in the design of experiments.

### 6.6 Summary

The possibility of early estimation of potential faults of software can help in the planning, controlling and executing software development activities. As the complexity and the constraints under which the software is being developed are ever increasing, it is difficult to produce software without defects. Such defective software classes might increase developmental and maintenance costs due to software failures and decreased customer satisfaction.

Most of the modules in the training set of software metrics normally contain zero or very small defect densities. Defect density values can thus be classified into two clusters namely defective and non-defective. Clustering as defective or defect free depending on a threshold value and evaluation in the classification process. This threshold value can be set within the learning process. It is a stability point where the maximum learning performance is found.

Unlike traditional random forests and all other classification algorithms reported in this work, balanced random forests require extra effort as their implementation is not immediately available from off-the-shelf software tools. If the software project requires that as many modules as possible should be inspected due to a dire consequence of software failures, this additional effort may be necessary. Prediction of defect-prone modules provides one way to support quality of software by better scheduling and project control. Accordingly, there is a need to develop a real-time assessment technique that classifies these dynamically generated systems as being defect / defect-free. A variety of software fault prediction techniques have been proposed but none has proved to be consistently accurate.

Integrated defect prediction model i.e. GA_FCM_RF based approach has been implemented for predicting software reliability. In order to achieve software quality, it is desirable that defects analysis for project before development so that more emphasis can be made on defect-prone areas.

By analyzing the results of the combined FCM, GA and RF, it is clear that modified fuzzy c-means clustering, genetic algorithm and enhanced random forest classifier based
approach give better results with an accuracy of more than 90%. The ability of the above methods to explicitly account for the unique characteristics of software products will enable researchers to make significant inroads on the hard challenges in the domain. This will result in the creation of robust software products which will have a major impact on the economic growth and technological advancement of modern society.