ISSN 2278-3091
Volume 9, No.3, May - June 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse145932020.pdf
https://doi.org/10.30534/ijatcse/2020/145932020

# Ensemble Methods to Improve Accuracy of a Classifier

**Dr.Dhimant Ganatra[1], Dr.Dinesh Nilkant[2]**
[1]CMS Business School, JAIN (Deemed-to-be University), Bangalore-560009, India, dr.dhimantganatra@cms.ac.in
[2]CMS Business School, JAIN (Deemed-to-be University), Bangalore-560009, India, dineshnilkant@cms.ac.in

**ABSTRACT**
A decision tree algorithm is developed for a B-School which can be used as a strategy to identify potential placeable candidates at the time of admission itself based on their past academic performance. Building the classification model is the easier part. How to get the best performance from the model is the challenging part. While the classification tree is simple and easy to interpret, it is not competitive as compared to other supervised learning approaches mostly in terms of prediction accuracy. The disadvantage of a single tree can be overcome if there is an approach which can produce multiple trees and combine them to yield a better prediction. Ensemble methods are supervised learning algorithms for combining multiple learners, trees in this case, to produce a strong learner. There will be some loss of interpretability but the improvement in accuracy will outweigh this. Baggingand random forests approaches are used to improve the accuracy of the positive class.

**Keywords:** Classification Tree, Ensemble Method, Machine Learning, Model Accuracy.

## 1. INTRODUCTION

### 1.1 The ClassificationTask

The 2021 and 2022 placement season will be an uphill task for India's tier II and tier III B-Schools on account of the current slowdown in the Indian economy [1] and the onslaught of Covid-19. There are 4,000 B-Schools from which around 360,000 management graduates vie for placement every season. Add to this figures the many autonomous institutes and the number of graduates is humongous. Only 60% of them land jobs as per [2]. Since the current economic slowdown is here to stay [3], the B-School under study was interested in knowing whether it is possible to differentiate the pool of applicants based on placeability. A B-School which has a lower acceptance rate into its program looks at many competencies when offering admission. Academic ability among others stands at the top.

Using the classification tree algorithm, a business rule was developed [4], which was able to classify applicants into two classes – Placeable and Not Placeable with 83.33% accuracy. The tree found acceptance with the admission team as it was easily interpretable and closely mirrored the human decision-making process in general. Unfortunately, the admission team was not happy with the model's overall accuracy and more specifically sensitivity which is 64.71% in the base model. The objective of this research paper is to build further on the model as proposed in [4], to improve its accuracy by avoiding misclassification of positives which in this case means wrongly identifying a non-placeable applicant as placeable. The accuracy of a tree can be improved using bagging, random forests, and boosting ensemble methods[5].

### 1.2 Machine Learning Algorithms

Machine Learning concerns the application of artificial intelligence that provides a system the ability to learn and improve from the experience. Machine learning is concerned with improving the prediction accuracy as opposed to statistical learning which is focussed on statistical inference. Machine learning algorithms are largely classified into supervised, unsupervised, reinforcement and evolutionary algorithms.When a training dataset has both predictor and outcome variables, we use supervised learning algorithms. Classification tree is a popular technique under the supervised algorithm category which is used to stratify the predictor space into a few simple regions. A classification tree is used to predict the class of a qualitative response variable based on the values of predictors using different splitting, merging, and stopping criteria. Reference [6] carried out an in-detail evaluation of more than 175 classifiers spread across 17 families over the complete UC Irvine (UCI) machine learning classification database. In the classification tree, the value or class of the outcome variable is predicted based on a single model. A classification tree is considered weak when used alone which is a reflection, in part, of its variance. The tree is known to show a large variability between different samples from the same dataset. In short, a classification tree is high on variance [7]. This is where comes in the concept of an ensemble learning model. The variance of any statistical learning method can be reduced using this general procedure. Ensembles are machine learning methods for combining predictions from multiple separate models into one that is usually more accurate than the individual components [8].Combining outputs of several classifiers reduces the risk of selecting a poorly performing classifier.The two most common methods for ensembling are Bagging and Boosting.

## 2. LITERATURE REVIEW

There have been many studies which have applied decision tree algorithm in an educational setup, but none have connected admission to placement for a B-school. Same is the case for ensemble methods. Classifier models have been developed to understand student success in exam and course completion using academic features. There are numerous research papers on predicting student placement but most of them use complex algorithms and are built on data post enrolment into a course. Reference [9] show how the use of ensemble methods provided better results in an e-learning setup. An ensemble tree based model classifier technique for predicting the student performance was used by [10]. The proposed model essentially combined two consistent machine learning techniques into a voting bagging technique to achieve higherperformance. Ensemble

techniques based on four representative learning algorithms were used by [11] to construct and combine different number of ensembles to predict whether a student will be able to successfully complete his degree. The performance of Adaboost and Bagging ensembles were better than Random Forest. In [12], Bagging and Boosting ensembles were evaluated on 23 datasets with decision tree as the classifier algorithm. Findings suggest that most of the gain in an ensemble's performance comes in the first few classifiers combined. Reference [13] concluded that building a random forest of trees improves the classifier. 10 years of data of a University were used to predict whether a student will complete the degree based on their performance in courses of first two semester. A classification model based on gradient boosting using decision tree as the base classifier was created by [14] to predict academic outcome of student performance at the end of the school year. Ensemble model developed in [15] provided increased accuracy in identifying students who are likely to fail or may drop out. In [16], to enhance the placement probability of students, optimal academic characteristics were selected and modelled using supervised learning techniques.The results suggest that greater accuracy in predicting students' placement can be achieved using the proposed hybrid CT–ANN modelthan other conventional supervised learning models.

## 3. ENSEMBLE METHOD

The ensemble method is a machine learning algorithm that creates several classifiers and then combines them to improve performance. By combining individual models, an ensemble model tends to be less biased (more flexible) with less variance (less data-sensitive). This superior performance comes at the cost of interpretability. Ensemble methods can be used in both classification and regression setup. When used in classification, the multiple classifiers that are developed are likely to classify a new observation in different categories. Then a strategy of majority voting is used to decide the final class of the new observation. Such majority voting could be based on simply counting the vote from each class or could be weighted based on accuracy. In case of regression problems, the prediction of a new observation is simple average or weighted average of all the predictions from the set of the developed regression models [17].

A trio of robust ensemble methods which use trees as building blocks are Bagging, Random Forests and Boosting. All these methods differ on how the data is selected from a weak dataset, how the weak models are generated, and also on how the outputs are combined so as to form a stronger classification model.

### 3.1 Bagging

A common way to increase the prediction accuracy and reduce the variance of a learning method involves taking multiple training sets from a population, building the classification model using each of these sets and then averaging the resultant predictions [7]. However, in practice we do not have the gift of several training sets. Instead, we can generate multiple random samples with replacement from the single training set. By doing so, we have generated n different bootstrapped training sets which are then fed to each classifier. We thenaggregate the class predicted by each of the n trees for a certain test observation and take a majority vote. The most frequently occurring class among the n predictions is the overall prediction for the test observation. This is called bootstrap aggregating akabagging. Bagging is popular not only because it helps to prevent overfitting, but also that, it can be parallelised for application involving large datasets.

The biggest advantage of bagging is to do away with the requirement to perform cross-validation or using the test set approach to check for model accuracy. Each bagged tree makes use of only two-thirds of the records from the single training set. The balance one-third of the records which are not part of training data of a tree are termed as the out-of-bag (OOB) records and can be used as validation data. A classification error can be computed based on OOB prediction for each of the record. This error is usable as an estimate of the test error as the response for each record is being predicted using only those trees that were not fit using the concernedrecord.

A problem with bagging is that, in case, there is a strong predictor present in the data set, then most or allof the trees are likely to use this predictor in the top split. The predictions from such bagged trees will be then highly correlated and averaging will not lead to a large reduction in variance. This problem is overcome by random forests.

### 3.2 Random Forests

Random forests forces each split to consider only a random subset from among the predictors, thereby overcoming the problem of tree correlation. This small tweak provides an improvement over bagging. Each time a spilt is considered, a random sample of only m out of available p predictors is considered. The split is then based on one of those m predictors thereby decorrelating the trees. A new list of m predictors is taken at each split and the value of m is approximately the square root of p. Where m equals p, then this amounts to bagging.On account of the limited number of predictors selected at each of the iteration, the generation of models is faster than bagging. In general, random forest approach is expected to provide much higher accuracy compared to a single tree [18].

### 3.3 Boosting

In bagging a tree is built on a bootstrapped dataset which is independent of other trees. In boosting, the trees are grown sequentially meaning that each tree is grown such that it uses information from trees grown beforehand. A training model concentrates on the misclassified records from previous models.That is, each tree in boosting is fit on a modified version of the original data setand then combining the classifiers via a weighted majority vote.Boosting frequently yields better models than bagging [7]. Two most widely used boosting algorithms are AdaBoost and Gradient Boosting. While AdaBoost focusses on the misclassified records in subsequent classifiers, Gradient Boosting focusses on residuals from previous classifiers and fits a model to the residuals.It learns from the mistake directly –the residual error, rather than updating the weights of data points.

## 4. METHODOLOGY

The objective here is to improve the accuracy especially of the positive class of the classifier model developed by [4]. The classifier model was built following the need of the B-School to differentiate among prospective students into placeableand non-placeable categories. Since the cost of

misclassifying a non-placeable category is high, ensemble methods are brought in to improve the existing model's sensitivity as well as overall accuracy. R language and environment (version 4.0.0) has been used for statistical computing and graphics [19].

215 students who completed their MBA from a Bangalore based B-School have been selected for the study as per [4].The dependent variable is a two-class categorical variable – Placement with labels as Placed and Not Placed. There are 10 predictor variables. The 75–25 technique is used [20] to split the data set. The crosstab in respect of the response variable is as shown in Table 1.

**Table 1:** Count of Train and Test set

| | Placed | Not Placed | Total |
|---|---|---|---|
| **Train set** | 111 | 50 | 161 |
| **Test set** | 37 | 17 | 54 |
| **Total** | 148 | 67 | 215 |

## 5. CLASSIFIER MODEL

### 5.1 Decision Tree Classifier – Base classifier

Recursive Partitioning And Regression Trees (rpart) library [21] provides the algorithm to create the decision tree.

Figure1 shows the classification tree for the train dataset. Nodes that split to the left are the ones which meet the criteria while nodes to the right do not. Each node is labelled by the predicted class, either Placed or as Not Placed. The percentage value is to be read from left to right, with the probability of Not Placed being on the left.

From Figure1, at node 7 of the tree we understand that if a student has scored more than 56% in SSC and more than 66% in Degree, then there is more than 94% chance that the student is likely to be Placed and the support is 49%.

### 5.2 Evaluating the performance of the Base classifier

A. Classification Table

Table 2 shows the confusion matrix for the tree in Figure1 applied to the test set.

The accuracy of classifying Placed (negative) is 91.89%, whereas the accuracy of classifying Not Placed (positive) is 64.71%. The overall accuracy is 83.33%. The positive and negative predicted values are 0.7857 and 0.8500 respectively. Given the context, here, we need a higher accuracy in predicting positive classes (Not Placed, $Y_i = 1$) rather than negative classes (Placed, $Y_i = 0$).

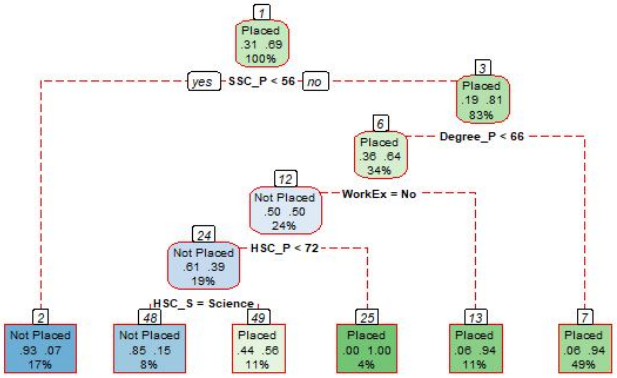


**Figure 1:** Classification tree for train dataset

**Table 2:** Confusion Matrix based on Gini impurity

| Predicted | Actual | | Overall % |
|---|---|---|---|
| | Not Placed | Placed | |
| **Not Placed** | 11 | 3 | |
| **Placed** | 6 | 34 | |
| **% Correct** | 64.71 | 91.89 | 83.33 |

The classification accuracy for the model using the training setis 88.82%.Since the classification accuracy of the test set is within 10% of the training set, this provides evidence of the utility of the model [22].

B. Sensitivity, Specificity, and Precision

In this case, sensitivity also known as Recall, measures how many of the actual Not Placed students are correctly predicted as Not Placed. From Table 2, sensitivity is 64.71%, meaning about 65% of the Not Placed students in the test dataset were correctly predicted as Not Placed. Specificity is the ability of the model to correctly classify the negatives, that is, when the actual value is negative, how often is the prediction correct. From Table 2, specificity is 91.89%, meaning less than 9% of all Placed students are predicted incorrectly as Not Placed. Precision measures how good the model is at assigning positives to the positive class. For our classifier model, the precision is 78.57%, meaning, almost 79% of the students predicted as Not Placed actually belonged to the Not Placed class.

C. F-Score

The F-Measure combines precision and recall and is their harmonic mean [23]. In cases where the cost of false negatives and false positives are very different, the F-score is superior as compared to the overall accuracy. In our case the F-Score is 0.7097,values closer to 1.0 are the best.

D. Receiver Operating Characteristic curve(ROC curve)

ROC curve is used in order to understand the overall worth of a classification tree [24]. The ROC curve for the placement test dataset is shown in Figure2.

The area under the ROC curve (AUC) is 0.8959, indicating the proportion of concordance pairs in the data. Models with higher AUC are preferred.

### 5.3 Bagging Ensemble

Figure3 shows the results from bagging. The plot shows the error and the number of trees. It is observed that as the number of trees increase, the error reduces. The black curve is the OOB error rate, red curve is the error for Not Placed class and the green line is the error for Placed class.Using large values for number of trees will not lead to overfitting, as it is not a critical parameter with bagging [7].

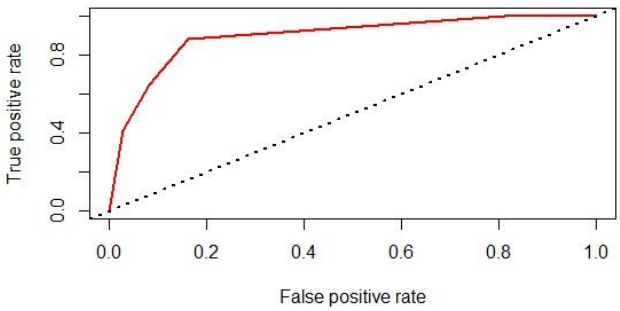


**Figure 2:** ROC Curve for test dataset

**Table 3:** Model Parameters

| Technique | Hyperparameter | | Train set | | | Test set | |
|---|---|---|---|---|---|---|---|
| | ntree | mtry | OOB Error | Sensitivity | Accuracy | Sensitivity | Accuracy |
| Classification Tree | -- | -- | -- | 0.7200 | 0.8882 | 0.6471 | 0.8333 |
| Bagging | 500 | 10 | 16.77% | 0.6200 | 0.8323 | 0.7647 | 0.8889 |
| Random Forest (Default) | 500 | 3 | 16.15% | 0.6200 | 0.8385 | 0.8235 | 0.9259 |
| Random Forest (Tuned) | 500 | 4 | 16.77% | 0.6200 | 0.8323 | 0.8235 | 0.9259 |
| Random Forest (Tuned) | 500 | 6 | 16.77% | 0.6200 | 0.8323 | 0.7647 | 0.9074 |
| Random Forest (Tuned) | 200 | 7 | 14.91% | 0.6800 | 0.8509 | 0.7647 | 0.9074 |
| Random Forest (Optimal m) | 500 | 9 | 16.15% | 0.6400 | 0.8385 | 0.7647 | 0.8889 |
| Random Forest (Ranger) | 500 | 3 | 17.39% | 0.6000 | 0.8261 | 0.8235 | 0.9259 |
| Random Forest (Ranger-best) | 180 | 4 | 11.80% | 0.7000 | 0.8820 | 0.8235 | 0.9259 |

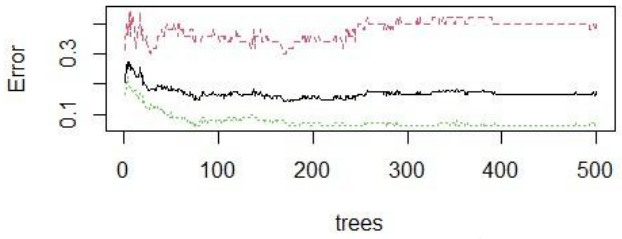


**Figure 3:**Class-Level Error and Number of Trees

Table 3 lists out the results of the bagging model. We see that the sensitivity and accuracy parameter of the test set are better compared to a single tree. Table 4 shows the confusion matrix for the model applied to the test dataset. Usually bagging results in anbetter accuracy as compared to a single tree.Percentage in SSC and Percentage in HSC are the variables with the largest decrease in Gini index. The variable importance indicates the total decrease in node impurity averaged over all trees. The AUC of 0.9634is higher than the AUC of single tree.

### 5.4 Random Forest Ensemble

We first build a random forest ensemble with default hyperparameters – 500 trees and 3 predictors ($\approx \sqrt{p}$). This model has led to a reduction in test error as well as the OOB error over bagging. Table 3 lists outthe model parameters. The AUC is 0.9745 and can be considered good since it is very close the maximum of one.

It is possible to seek improvement to the model by tuning the hyperparameters. The most commonly tuned hyperparameters include – number of variables randomly sampled as predictors at each split (mtry), the number of trees to grow (ntree), and the size of the sample to draw for training. Table 3 shows the output for three such tuned models.

It is possible to perform a larger grid search across many values of the hyperparameters by creating a grid and a loop through each possible combination. It is important that we choose an optimal set of hyperparameters to tune the model so as to better fit the data. We have evaluated 3276 different models by varying the above three hyperparameters.The top 10 performing models have an OOB error between 11.80% and 13.04% which is lower than the default or the three manually tuned models.

**Table 4:** Confusion Matrix based on Bagging

| Predicted | Actual | | Overall % |
|---|---|---|---|
| | Not Placed | Placed | |
| **Not Placed** | 13 | 2 | |
| **Placed** | 4 | 35 | |
| **% Correct** | 76.47 | 94.60 | 88.89 |

The best random forest model we found based on the grid search uses 180 trees, 4 variables, and a sample size of 63.2%. Table 5 shows the confusion matrix for the model applied to the test dataset.The AUC is an impressive 0.9571.

**Table 5:** Confusion Matrix based on Random Forest

| Predicted | Actual | | Overall % |
|---|---|---|---|
| | Not Placed | Placed | |
| **Not Placed** | 14 | 1 | |
| **Placed** | 3 | 36 | |
| **% Correct** | 82.35 | 97.30 | 92.59 |

## 6. CONCLUSION

With increased power of computing infrastructure, the necessary simplifying assumptions of linearity and normality are staring to give way to nonparametric techniques. While trees are easy to interpret and fit the data nicely,they suffer from high variance.With asmall change in the training data,the results that we get could be significantly different in the model [25].Though pruning a tree should help in reducing this variance, alternative methods which exploit the variability of a single tree so as to improve performance are available. One such approachis the Bootstrap Aggregating[26]. The bagging ensemble for the test set gives us an overall accuracy of 88.89% as compared to 83.33% accuracy obtained by a single tree.Bagging typically suffers from tree correlation, which in turn reduces the overall performance of the model. A modification of bagging is random forestswhich build a large collection of de-correlated trees [26]. Random forest is a very popular learning algorithm which enjoys good predictive performance. We have used the randomForest package to implement both bagging and random forest algorithm. The default random forest model based on this package gives a further improved accuracy of 92.59% on the test set. It is possible to tune several hyperparameters of the randomForest function to check on the possibility of finding more superior parameters as compared to the default model. We have

shown the model parameters under three manually tuned random forest model. One such variant with 200 trees and 7 randomly sampled predictors has an OOB error estimate of 14.91% which is lower than the OOBs obtained under the bagging and default random forest algorithms. We have used the ranger package which provides a fast implementation to random forests [5] to perform a large grid search across the three important hyperparameters. A total of 3276 models are evaluated to identify the best combination of the three hyperparameters. The model developed using this best combination has the lowest OOB error of 11.80% among all the other model variants.

Though random forests lack some interpretability, they make up for in prediction power. Gini index can be used to obtain an overall summary of the importance of each predictor. Table 6 indicates the variable importance relevant to the model built using the best combination of the hyperparameters. The scale is irrelevant, only the relative values matter. Percentage in SSC is over one-and-half times more important than Percentage in Degree.

**Table 6:**Variable Importance

| Variable | Importance | Variable | Importance |
|---|---|---|---|
| SSC_P | 13.9708 | Gender | 1.5861 |
| Degree_P | 8.9143 | HSC_S | 1.1463 |
| HSC_P | 8.5988 | Degree_T | 1.2123 |
| Etest_P | 3.9128 | SSC_B | 0.9735 |
| WorkEx | 1.8623 | HSC_B | 0.7431 |

**Future work:** An alternative approach for enhancing the predictions of a classification tree is Boosting. While random forest builds an ensemble of deep independent trees, boosting involves building an ensemble of shallow and weak successive trees.Each successive tree learns and improves on the previous one.There is scope to evaluate the increase in model accuracy using this method [27]. An attempt can also be made to subset the data and include variables based on their importance and use that with another model.

## REFERENCES


1. Economic Times (2019, November 02). **Slowdown has tier II, III B-schools on tenterhooks**. https://economictimes.indiatimes.com/jobs/slowdown-has-tier-ii-iii-b-schools-on-tenterhooks/articleshow/71860868.cms?from=mdr
2. India Today (2019, October 26). **Step by step to the top**. https://www.indiatoday.in/magazine/education/story/20191104-step-by-step-to-the-top-1612697-2019-10-26
3. G. Guru. **Deepening Economic Slowdown**. *Economic and Political Weekly*, vol 54, no. 48, p. 8, Dec 2019.
4. D. Ganatra, and D. Nilkant.**A Business Rule for a B-School using Machine Learning**. *International Journal of Advanced Trends Computer Science and Applications*, vol 8, no. 6, pp. 3621-3627, Dec 2019. https://doi.org/10.30534/ijatcse/2019/145862019
5. L. Breiman. **Random Forests**. *Machine Learning*, vol. 45, no. 1, Springer, pp. 5–32, 2001. https://doi.org/10.1023/A:1010933404324
6. M. Fernández-Delgado, E. Cernadas, S. Barro, and D. Amorim, **Do we Need Hundreds of Classifiers to Solve Real World Classification Problems?**. *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3133-3181, 2014.
7. G. James, D. Witten, T. Hastie and R. Tibshirani. **An Introduction to Statistical Learning with Applications in R**. Springer, 2017, ch.8.
8. G. Seni, andJ. F.Elder.**Ensemble Methods in Data Mining: Improving Accuracy Through Combining Predictions**. Morgan & Claypool Publishers, 2010.
9. P. Kumari, P. K. Jain, and R. Pamula. **An efficient use of ensemble methods to predict students academic performance**.*4th International Conference on Recent Advances in Information Technology (RAIT)*, Dhanbad. IEEE, pp. 1-6, 2018.doi: 10.1109/RAIT.2018.8389056
10. A.Almasri, E.Celebi, and R. S. Alkhawaldeh.**EMT: Ensemble Meta-Based Tree Model for Predicting Student Performance**. *Scientific Programming*, vol. 2019, Article ID 3610248, 13 pages, 2019.
11. M. Pandey, and S. Taruna.**A Comparative Study of Ensemble Methods for Students' Performance Modeling**. *International Journal of Computer Applications*,vol. 103,no.8, pp.26-32, Oct 2014. https://doi.org/10.5120/18095-9151
12. D. Opitz, and R. Maclin.**Popular Ensemble Methods: An Empirical Study**. *Journal of Artificial Intelligence Research*, vol. 11,pp. 169-198, 1999.
13. C. Beaulac, and J. S. Rosenthal.**Predicting University Students' Academic Success and Major Using Random Forests**. *Research in Higher Education*, vol. 60, no. 7, pp. 1048-1064, Nov 2019. https://doi.org/10.1007/s11162-019-09546-y
14. E. Fernandes et al. **Educational data mining: Predictive analysis of academic performance of public school students in the capital of Brazil**. *Journal of Business Research*, vol. 94, pp. 335-343, Jan 2019.
15. P. Kamal, and S. Ahuja. **An ensemble-based model for prediction of academic performance of students in undergrad professional course**. *Journal of Engineering, Design and Technology*, vol. 17 no. 4, pp. 769-781, 2019.
16. T. Chakraborty, S. Chattopadhyay, and A. K. Chakraborty.**A novel hybridization of classification trees and artificial neural networks for selection of students in a business school**. *OPSEARCH*, Springer, vol. 55(2), pp. 434-446, 2018.
17. M. Pradhan, U. D. Kumar, **Machine Learning using Python**. New Delhi: Wiley India, 2019, pp. 225-239.
18. U. D. Kumar, **Business Analytics- The Science of Data Driven Decision Making**. New Delhi: Wiley India, 2017, pp. 403-405.
19. **The R project for statistical computing**. https://www.r-project.org/
20. J.E. Holden, W. H. Finch, andK. Kelley. **A Comparison of Two-Group Classification Methods, Educational and Psychological Measurement**. 71(5). Sage, 2011, pp. 870–901. https://doi.org/10.1177/0013164411398357
21. **Package rpart**. https://cran.r-project.org/web/packages/rpart/rpart.pdf
22. G. Forman, andM. Scholz.**Apples to apples in cross-validation studies: Pitfalls in classifier performance measurement**. *ACM SIGKDD Explorations*, vol.12(1), pp. 49–57, 2010.

23. M. Sokolova, N. Japkowicz, and S. Szpakowicz. **Beyond accuracy, f-score and roc: a family of discriminant measures for performance evaluation**. Australasian Joint Conference on Artificial Intelligence, Springer, 2006, pp. 1015-1021.
24. J. A. Hanley,and B. J. McNeil.**The meaning and use of the area under a receiver operating characteristic (ROC) curve**. *Radiology*, vol. 143(1), pp. 29□36, 1982. https://pubs.rsna.org/doi/pdf/10.1148/radiology.143.1.7063747 https://doi.org/10.1148/radiology.143.1.7063747
25. J. P. Lander. **R for everyone – Advanced Analytics and Graphics**. Pearson Education, Inc, 2016, pp. 310-312.
26. L Breiman, JH Friedman, RA Olshen, and CJ Stone. **Classification and Regression Trees**. Chapman and Hall, 1984.
27. S. I. Manzoor, and J. Singla. **A Comparative Analysis of Machine Learning Techniques for Spam Detection**. *International Journal of Advanced Trends Computer Science and Applications*, vol 8, no. 3, pp. 810-814, Jun 2019. https://doi.org/10.30534/ijatcse/2019/73832019