JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 37, 993-1009 (2021)
DOI: 10.6688/JISE.202109_37(5).0001

Exploiting Machine Learning and Feature Selection Algorithms to Predict Instructor Performance in Higher Education

RAVINDER AHUJA AND S. C. SHARMA
Electronics and Computer Discipline
Indian Institute of Technology, Roorkee
Saharanpur Campus, Saharanpur, 247001 Inida
E-mail: {ahujaravinder022; scs60fpt}@gmail.com

Machine learning has emerged as the most important and widely used tool in resolving the administrative and other educational related problems. Most of the research in the educational field centers on demonstrating the student's potential rather than focusing on faculty quality. In this paper the performance of the instructor is evaluated through feedback collected from students in the questionnaire form. The unlabelled dataset is taken from UCI machine learning repository consisting of 5820 records with 33 attributes. Firstly, the dataset is labelled(three labels) using agglomerative clustering and the k-means algorithms. Further, five feature selection techniques (Random Forest,Principal Component Analysis, Recursive Feature Selection, Univariate Feature Selection, and Genetic Algorithm) are applied to extract essential features. After feature selection, twelve classification algorithms (K Nearest Neighbor, XGBoost, Multi-Layer Perceptron, AdaBoost, Random Forest, Logistic Regression, Decision Tree, Bagging, LightGBM, Support Vector Machine, Extra Tree and Naïve Bayes) are applied using Python language. Out of all algorithms applied, Support Vector Machine with PCA feature selection technique has given the highest accuracy value 99.66%, recall value 99.66%, precision value 99.67%, and f-score value 99.67%. To prove that results are statistically different, we have applied ANOVA one way test.

Keywords: classification algorithms, decision tree, ensemble learning, faculty performance evaluation, feature selection, support vector machine

1. INTRODUCTION

There has been a significant increase in the number of higher education systems over the last few years , resulting in more students, including both masters and bachelor's, every year [1, 11]. Many higher educational institutions and universities have made changes in their teaching methods or in the way of organizing exams. However, they haven't realized the increase in issues relating to unemployment and dropout students [4]. Understanding the reasons for low performance or massive growth in the dropout rate is a tough task. EDM (Educational Data Mining) [2] uses algorithms to analyse educational statistics to find trends and forecasts in data that explain results. Although the institutions / universities have maintained valuable educational databases, but they are not used in any decision-making process, improve the quality of academic programs and services [3].

Received September 26, 2020; revised November 9, 2020; accepted November 18, 2020.
Communicated by Maria José Sousa.

EDM can be used to gain useful information about all entities playing a role in educational institutions like faculty, student, staff, and management. A lot of methods (use of various algorithms depending upon the type of dataset) are being used in higher educational institutions to solve critical problems related to higher education, which have kept them missing from achieving their quality objectives. The earlier methods were focused on improving the students' performance via its academics rather than considering/evaluating the faculties performance [5, 6, 7, 8]. These techniques lacked the skill to reveal necessary useful information. The most commonly faced problem in evaluating a faculty's performance was, which tool should be used to measure the faculty performance over the courses present? Now a days the most frequently used method to estimate the faculty performance or effectiveness in a particular subject is via surveying the student's feedback for that specific faculty and course utilizing a set of questionnaires. In the 1920s, there was a debate on the validity and reliability of these methods [9]. Most of the EDM applications are on student's performance evaluations, redesigning the curricula, and exploring online learning environments [10]. This work aims to use EDM's power in the faculty's performance evaluation so that teaching quality can be improved. **The paper's major contributions are:** (i) Design of machine learning based system in which missing values are replaced with mean, data is normalized using MIN-MAX scalar, outliers are detected through boxplot and replaced with median values; (ii) Optimizing the machine learning based system by picking best combination of feature selection and classification algorithms among the five feature selection algorithms (Principal Component Analysis (PCA), Random Forest (RF), Univariate Feature Selection, Recursive Feature Elimination (RFE), and Genetic Algorithm) and twelve classification algorithms (NB, ET, MLP, LGBM, SVM, AdaBoost, RF, LR, DT, XGBoost, KNN, and bagging). We found that PCA (Principal Component Analysis) with linear SVM classifier gives a 6% higher performance as compared to existing approaches in the literature; (iii) Demonstration of an increase in performance by 6% compared to the results present in the literature. The statistical method (ANOVA one-way) have been applied to prove that results produced are not fluke. The remaining paper is divided into following parts: part 2 contains literature survey, part 3 contains the dataset description, part 4 contains feature selection methods, part 5 presents the methodology used, Section 6 is about results and discussion and Section 7 finally concludes the paper with future work.

2. LITERATURE SURVEY

EDM is a research field that is related to the applications of data mining/machine learning on the information collected from educational institutions to determine patterns and learning methods. L. Rahman *et al*. [11] have applied two feature selection methods namely information gain and wrapper method. On the selected features three classification algorithms namely Naïve Bayes, Decision Tree, and Artificial Neural Networks have been applied on the 481 instances to predict student's performance. The ANN algorithm has given the best performance with an accuracy of 79.37% with information gain feature selection. B. K. Bhardwaj *et al*. [12] proposed a model to predict the students' results in the end term examination and help identify dropouts and students who need special care. They have applied decision tree, neural networks, and *K*-Nearest Neighbors algorithm.

The results depicted that the decision trees have the best accuracy for classifying. Their study also determines the factors affecting student's performance. M. Zaffar [13] have applied six feature selection algorithms for analysis of student's performance. Feature selection techniques enhances the performance of algorithms by removing the irrelevant data from the educational dataset. They find out that principal component analysis with the random forest is giving the highest performance among all the combinations of feature selection and classification algorithms. A. Ogunde *et al*. [14], proposed an approach to predict student's final marks based on data at entry-level using the ID3 decision tree algorithm. They initially applied the ID3 algorithm to train the data, then extracted the knowledge, and represented it in the form of rules. Then they developed a method using trained data for future prediction of student's grades. P. Guleria *et al*. [15] proposed a method for evaluating an educational institution's performance by taking feedback from its faculty members and students so that they can enhance the productivity and quality of programs that are currently being running in the institutions with the help of data mining methods. They have taken infrastructure, the facilities provided to teachers and students, teaching skills, course content, and its weight as parameters for evaluating the performance of the Institution. I. A. A. Amra *et al*. [16] have proposed a student's performance prediction model with Naïve Bayes and KNN algorithms. They collected an educational dataset of secondary schools from the ministry of education in Gaza for the year 2015. They concluded that Naive Bayes is better than KNN and gives much higher classification accuracy. W. Jie [17] have used data generated by students during the online study and applied machine learning algorithms to find the important parameter which can improve teaching process and make the contents according to students learning pattern. A. M. Ahmed *et al*. [18] have applied Naïve Bayes, Multilayer Perceptron, J48 Decision Tree, and Sequence Minimal Optimization classification algorithms to predict instructors performance. They found that SMO algorithm with attribute evaluation is giving the highest accuracy of 85.8%. M. Agaoglu [19], has predicted the instructor's performance using seven classification algorithms. The data is collected from the 2850 students of Marmara University, Turkey. They found that C 5.0 algorithm has performed better among all the algorithms applied, and instructor performance is based on student's perception of the interest in course. S. Suh [20] applied seven classification algorithms on the Turkiye student evaluation dataset, and one more dataset. They have considered a full dataset and with selected features. They found J4.8 is giving the highest accuracy of 84.35% with the full feature set. T. Selvy *et al*. [21] applied the decision tree algorithm for feature selection and three classification algorithms (C 4.5, Naïve Bayes, and Support Vector Machine) to predict instructor performance. They found that naïve bayes classifier has performed better among all the algorithms. H. Suparwito [22], has applied three classification algorithms (random forest, deep learning, and gradient boosting) to find the course's difficulty level from students, teachers, and infrastructure point of view. They found that gradient boosting algorithm is better in this case. D. Geremew *et al*. [23] have applied J48, decision tree, and naïve Bayes classifier using a weka tool for predicting instructor performance. M. O. Asanbe *et al*. [24] applied C 4.5, Multi-layer perceptron, and ID3 for prediction of instructor performance. They found that C4.5 algorithm has given the highest accuracy of 83.5%. A. Kumar *et al*. [25] have applied four classification algorithms (Naïve Bayes, CART, LDA, and ID3) to predict the instructor's performance. They found that naïve Bayes classifier has given the highest accuracy of 80.35%. S. Mardikyan and

B. Badur [26] have applied decision tree and step wise regression algorithms on the data collected at Bogazici University. They identified the employment status of the teacher's and attendance of the student's as important factors in teacher performance. E. Taherifar *et al*. [27] applied principal component analysis for reducing the features, and two clustering techniques have been applied to make the dataset labeled. Further, the decision tree algorithm is applied to predict the instructor's performance. D. Buenaño-Fernández *et al*. [28] applied the decision tree algorithm to predict student grades based on their previous grades. N. Chauhan *et al*. [29] applied five machine learning algorithms to predict the students' performance so that help can be provided to the students at the appropriate stage. J. Sowmiya *et al*. [30] have applied a feed-forward neural network, linear regression, and association rules to predict the instructor's performance. R. Kh. Hemaid *et al*. [31] have applied four classification algorithms (K-NN, Naïve Bayes, Rule Induction, and decision tree) to predict instructor performance. They found KNN is the best performing algorithm among all, giving an accuracy of 79.92%.

In the literature, various machine learning algorithms have been applied to educational datasets (mostly to predict student's performance). But, the work which thoroughly explores the predictive power of machine learning algorithms and feature selection algorithms in predicting instructor performance is not presented yet. To address the gap, this paper presents an exploratory analysis of five feature selection techniques (Principal Component Analysis, Random Forest, Recursive Feature Elimination, Univariate Feature Selection, and Genetic Algorithm) and twelve classification algorithms (Random Forest, Extra Tree, Decision Tree, Logistic Regression, Support Vector Machine, Adaboost, Naïve Bayes, XGBoost, LightGBM, *K*-Nearest Neighbors, Bagging, and Multi-Layer Perceptron) to predict instructor performance.

3. VARIABLES AND DATASET COLLECTION

The dataset is taken from the UCI machine learning repository [32]. There are 5820 records with 33 attributes over three instructors, as given in Table 1. The attributes are: instr-identifier for instructor having values 1 to 3, class-course code having values from 1 to 13 (Fig. 1 shows the feedback count corresponding to all the thirteen courses), nb.repeat- number of times student is taking the course having values 0 on wards, attendance-attendance of the student having values from 1 to 4, difficulty- the difficulty of the course having values 1 to 4, and Q1-Q28 are course-specific attributes having values 1 to 5 collected at Gazi University in Ankara (Turkey). The value 4, 5 is considered as very good, value 3 is considered as good, and value 1, 2 is considered as bad. Fig. 2 shows feedback given by the student in attributes corresponding to Q1 to Q28. Q14, Q15, Q17, Q19, Q20,Q21, Q22, Q25, and Q28 are given good ratings.

Table 1. Description of total records in the data set.

Course Code	Instructor	Number of Students
3,4,5,8,9,12,13	3	3600
1,6,11,13	2	1444
2,7,10	1	776

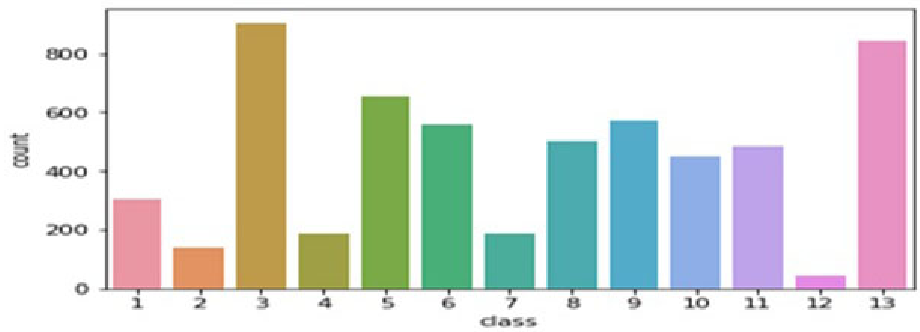

Fig. 1. Feedback count corresponding to different courses.

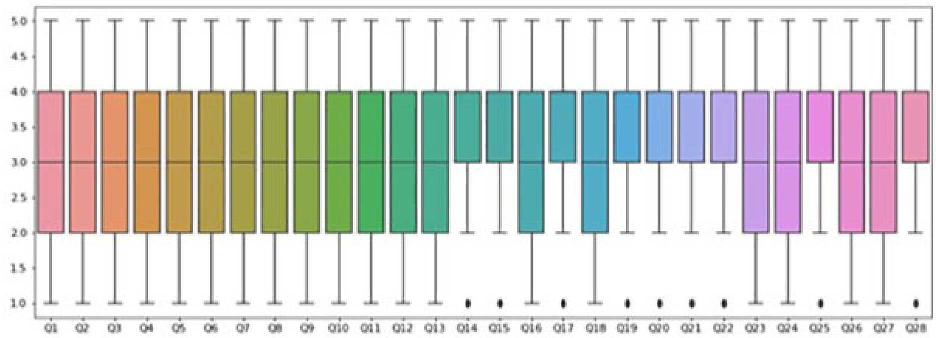

Fig. 2. Ratings for the question1 to question 28.

4. FEATURE SELECTION ALGORITHMS

Feature selection algorithms are applied to reduce the number of input variables so that computation cost is reduced, over fitting is reduced, and performance may increase. In this paper, following feature selection techniques are applied:

4.1 Principal Component Analysis

PCA is an unsupervised technique used to reduce the number of features in a dataset [33]. The new features created are uncorrelated to each other. The components are ranked according to their variance. So, features can be selected by keeping the principal component whose cumulative variance is 90%. The number of components taken in our study is 8.

4.2 Random Forest

Random forest [34] is robust algorithm and consists of many decision trees. Tree-based strategies used by RF are capable of ranking the purity of nodes. Nodes with less impurity are present at the start of the tree, and those with more impurity occur at the end of the tree. Thus, traversing trees below a particular node, a subset of essential features can be created. The threshold value taken is 0.001, that returns ten features.

4.3 Univariate Feature Selection

In this approach, a statistical test is applied to select the best features from the set of all input features [35]. Each feature at a time is considered to find the statistical relationship with the target variable. One feature is considered at a time; that's why it is called univariate feature selection. We have used CHI 2 statistical test to find the best features. We find best 8 features out of 33 input features.

4.4 Recursive Feature Elimination (RFE)

As the name suggests, this technique [36] removes the features using recursion, and it builds a model using the remaining attributes and calculates the accuracy of the model. It can easily find the combination of features that can best predict the target class. The technique firstly ranks the attributes according to their ability to predict the target class then selects the elements with rank one while making a reduced dimensionality dataset.We have used logistic regression model in our study and best 10 features are selected.

4.5 Genetic Algorithm Based Feature Selection

Genetic algorithm is an optimization technique based on the principle of Darwin's theory [37]. Initially, a population is created by randomly selected solutions. We have used binary coding for solution representation. The length of the solution is equal to input features (33 in our case). In the second step, the fitness function is evaluated (SVM classifier with accuracy) for all the solutions. The fittest solution from the current population survives to the next generation. The next stage is a crossover; in this two new solutions are generated based on existing solutions. The last step is mutation; in this genes of solutions are randomly flipped (0 to 1 and 1 to 0). The hyper parameters used in our study are: (i) population size is 100; (ii) the fitness function used is random forest with accuracy performance parameter; (iii) Single point crossover; (iv) Tournament selection. The number of features selected by this approach are eight.

5. METHODOLOGY USED

The methodology used in this paper is shown in Fig. 3. The dataset of 5820 instances with 33 attributes is taken from the UCI machine learning repository, which is collected at Gazi University in Ankara (Turkey). The dataset is preprocessed by handling missing values and outliers in it. The features are scaled using standard scalar as the attributes have values from 1 to 5. In standard scalar, the mean value is subtracted from the feature value and divided by standard deviation, as given in Eq. (1). For labeling the dataset, we have applied K-Means clustering and agglomerative clustering to find the number of clusters in the dataset. We have applied the elbow method to find the optimal number of clusters in the case of k-means clustering. The three clusters are formed in the dataset using both the clustering techniques. Based on this information dataset is labeled with three labels. In the next step, features are selected by applying five techniques (principal component analysis, random forest, recursive feature elimination, univariate feature selection, and genetic algorithm).

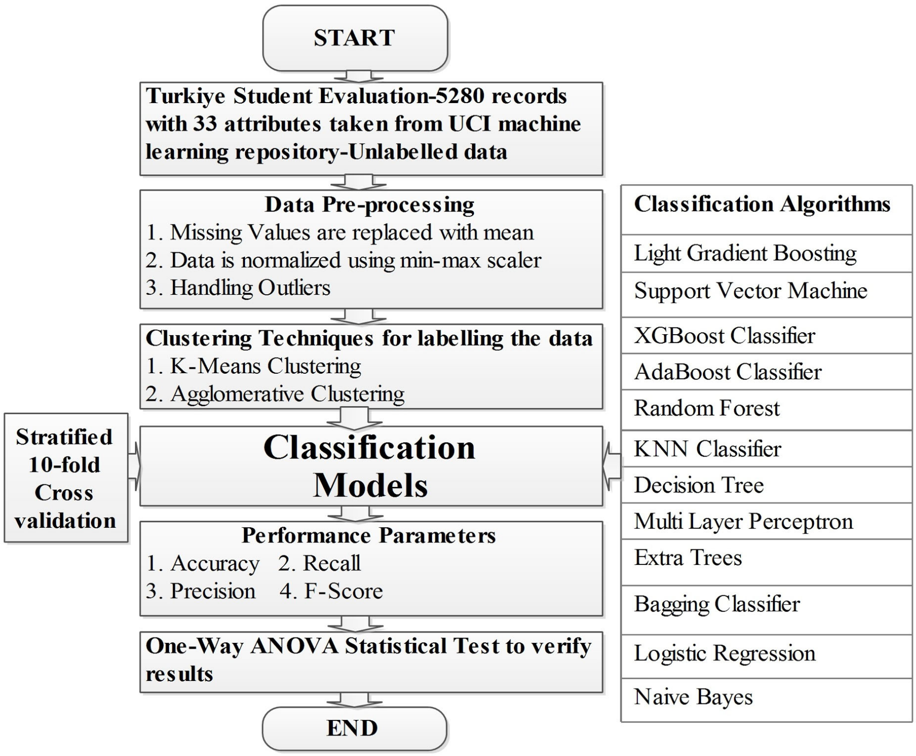

Fig. 3. Flowchart of the methodology used.

$$X\_Scaled = \frac{(x - mean)}{(standard\ deviation)} \tag{1}$$

On the selected features, we have applied twelve classification algorithms (linear support vector machine [39], K-nearest neighbors [40], the hyper parameters used in KNN are n_neighbors=5, metric="euclidean", decision tree [41] with criterion="entropy", logistic regression [42], naïve Bayes,bagging [43] the hyper parameters used are base_estimator=DecisionTreeClassifier(), n_estimators=10, random_state=7, AdaBoost [44] the hyper parameters used are n_estimators=100, random_state=7, XGboost [45] the hyper parameters used are n_estimators=300, learning_rate=,0.05, max_depth=5, extra tree [46] the hyper parameters used are n_estimators=100, max_depth=5, random forest [47] the hyper parameters used are n_estimators=100, criterion="entropy", multi-layer perceptron [51], the hidden layer size used is (100,100,100), and lightgbm [49]) with 10-fold cross-validation.The data is divided into 10 portions. One portion is used for evaluation, while the remaining nine portions are used for training and this process is applied on each portion of the dataset. The average performance of each model in 10-fold is taken into consideration so that biasness can be reduced. We have taken four performance parameters namely accuracy, recall, precision, and f-score to compare the models applied to predict instructor performance.

6. RESULTS AND DISCUSSION

All the experiments are performed on kaggle.com using the GPU processing unit. We have used various libraries of python like scikit learn [50], pandas, NumPy, matplotlib, seaborn, *etc*. Data cleaning is done using panda's library. Machine learning algorithms are implemented using scikit-learn library. All the graphs are plotted using the seaborn library. We have used recall, accuracy, precision, and f-score as performance parameters.

6.1 Performance Parameters

All the specified algorithms are compared over four parameters, including Recall, Precision, F-score, and Accuracy.

Accuracy: This is the ratio of correctly measured samples to the total samples present in the data set as given in Eq. (2).

$$Accuracy = \frac{(TruePositive + TrueNegative)}{(TotalPositive + TotalNegative)} \tag{2}$$

Precision: This is the ratio of true positive samples to the total predicted positive samples, as given in Eq. (3).

$$Precision = \frac{(TruePositive)}{(TruePositive + FalsePositive)} \tag{3}$$

Recall: This is the ratio of true positive samples to the total actual positive samples as given in Eq. (4).

$$Recall = \frac{(TruePositive)}{(TruePositive + FalseNegative)} \tag{4}$$

F-Score: This measure is combination of precision and recall. It is defined as 2 multiplied by the ratio of precision and recall product to the sum of precision and recall as given in Eq. (5). In this measure both precision and recall are weighted equally.

$$FScore = 2 * \frac{(Precision * Recall)}{(Precision + Recall)} \tag{5}$$

Table 2 is showing the results of principal component analysis method and twelve classification algorithms. Through principal component analysis, only eight attributes are selected because they represent 90% of the data. It is observed that the linear support vector algorithm has better among all the classifiers applied. The performance achieved with linear SVM is accuracy 99.66%, f-score 99.67%, recall 99.66%, and precision of 99.67% with 0.25 standard deviation. XGBoost, lightgbm, and decision tree are the algorithms that are competing with the linear SVM. The worst performance is reported by the AdaBoost classifier giving an accuracy of 93.13%, f-score of 90.82, recall of 90.88%, and precision of 96.85%, as shown in Table 2. Table 3 shows the results of all the classification algorithms applied to the features selected by the random feature elimination algorithm. The logistic regression model was used in RFE feature selection, and a total

Table 2. Performance comparison of classification algorithms based on PCA.

Sr. No.	Classification Algorithm	Accuracy	Precision	Recall	F-Score
1.	KNN	98.08	97.90	98.13	98.01
2.	**Linear SVM**	**99.66**	**99.67**	**99.66**	**99.67**
3.	Random Forest	99.50	99.60	99.63	99.61
4.	Bagging	99.48	99.44	99.54	99.49
5.	AdaBoost	93.13	96.85	90.88	90.82
6.	XGBoost	99.64	99.63	99.62	99.62
7.	Multi-Layer Perceptron	99.09	98.97	99.34	99.02
8.	Logistic Regression	98.44	98.10	98.54	98.31
9.	Decision Tree(ID3)	99.52	99.52	99.56	99.44
10.	Light GBM	99.55	99.50	99.54	99.52
11.	Extra Tree Classifier	99.16	99.23	99.04	99.07
12.	Naive Bayes	97.42	97.38	97.66	97.50

of 10 attributes were extracted out of 33 attributes. It is observed that the linear support vector algorithm has performed better among all the classifiers applied. The performance achieved with linear SVM is accuracy-96.99%, f-score 97.10%, recall 97.14%, and precision of 97.08% with approx. 0.68 standard deviation. Naïve Bayes, lightgbm, XGBoost, and decision tree are the algorithms that are competing with the linear SVM. The worst performance is reported by the decision tree classifier giving an accuracy of 94.59%, f-score of 94.77, recall of 94.69%, and precision of 94.70%, as shown in Table 3.

Table 3. Performance comparison of classification algorithms based on recursive feature elimination.

Sr. No.	Classification Algorithm	Accuracy	Precision	Recall	F-Score
1.	KNN	95.89	95.86	96.22	96.01
2.	**Linear SVM**	**96.99**	**97.08**	**97.14**	**97.10**
3.	Random Forest	96.08	96.20	96.12	96.27
4.	Bagging	95.53	95.46	95.77	95.59
5.	AdaBoost	95.43	95.34	95.80	95.50
6.	XGBoost	96.19	96.29	96.28	96.27
7.	Multi-Layer Perceptron	95.95	96.18	96.19	96.04
8.	Logistic Regression	96.17	95.97	96.54	96.22
9.	Decision Tree(ID3)	94.59	94.70	94.69	94.77
10.	Light GBM	96.22	96.33	96.36	96.32
11.	Extra Tree Classifier	96.13	96.28	96.17	96.25
12.	Naive Bayes	96.70	96.99	96.70	96.82

Table 4 shows the results of all the classification algorithms applied to the features selected by the univariate feature selection algorithm. CHI2 statistical method was used in our study, and a total of 10 attributes were extracted out of 33 attributes. It is observed that the linear support vector algorithm performed better among all the classifiers applied. The performance achieved with linear SVM is accuracy 95.50%, f-score 95.54%, recall 95.58%, and precision of 95.56% with 0.99 standard deviation. XGBoost and logistic regression algorithm are competing in terms of performance with the linear SVM. The worst performance is reported by the AdaBoost classifier giving an accuracy of 90.77%, f-score of 90.86, recall of 92.19%, and precision of 90.70%.

Table 4. Performance comparison of classification algorithms based on univariate feature selection.

Sr. No.	Classification Algorithm	Accuracy	Precision	Recall	F-Score
1.	KNN	95.00	95.02	95.18	95.08
2.	**Linear SVM**	**95.50**	**95.56**	**95.58**	**95.54**
3.	Random Forest	94.95	95.04	94.78	94.84
4.	Bagging	94.12	94.20	94.22	94.18
5.	AdaBoost	90.77	90.70	92.19	90.86
6.	XGBoost	95.17	95.26	95.29	95.25
7.	Multi-Layer Perceptron	94.40	94.79	94.81	94.73
8.	Logistic Regression	95.09	94.74	95.48	95.06
9.	Decision Tree(ID3)	93.44	93.16	93.88	93.48
10.	Light GBM	94.97	95.08	95.13	95.07
11.	Extra Tree Classifier	94.67	94.89	95.09	94.82
12.	Naive Bayes	94.09	94.33	94.38	94.30

Table 5 shows the results of all the classification algorithms applied to the features selected by the genetic algorithm. A total of eight attributes were selected out of 33 attributes. It is observed that the linear support vector algorithm is performing better among all the classifiers applied. The performance achieved with linear SVM is accuracy 92.97%, f-score 93.04%, recall 93.09%, and precision of 93.08% with approx.1.96 standard deviation. Naïve Bayes, XGBoost, and logistic regression algorithms are competing in terms of performance with the linear SVM. The worst performance is reported by the decision tree classifier giving an accuracy of 89.04%, f-score of 89.06%, recall of 89.29%, and precision of 89.18%. Table 6 shows the results of all the classification algorithms applied to the features selected by the random forest algorithm. A total of 16 attributes were extracted out of 33 attributes. It is observed that the linear support vector

Table 5. Performance comparison of classification algorithms based on genetic algorithm feature selection.

Sr. No.	Classification Algorithm	Accuracy	Precision	Recall	F-Score
1.	K-Nearest Neighbor	91.43	91.64	91.29	91.41
2.	**Linear SVM**	**92.97**	**93.08**	**93.09**	**93.04**
3.	Random Forest	92.16	92.59	91.89	92.12
4.	Bagging	90.74	91.16	90.83	90.93
5.	AdaBoost	89.81	89.39	90.51	89.55
6.	XGBoost	92.11	92.57	92.00	92.22
7.	Multi-Layer Perceptron	90.55	91.17	90.93	90.98
8.	Logistic Regression	92.20	92.20	92.47	92.26
9.	Decision Tree(ID3)	89.04	89.18	89.29	89.06
10.	Light GBM	92.08	92.44	92.04	92.16
11.	Extra Tree Classifier	91.86	92.03	91.76	91.86
12.	Naive Bayes	92.68	93.11	92.47	92.73

Table 6. Performance comparison of classification algorithms based on random forest feature selection.

Sr. No.	Classification Algorithm	Accuracy	Precision	Recall	F-Score
1.	KNN	96.58	96.44	96.69	96.55
2.	**Linear SVM**	**97.25**	**97.24**	**97.44**	**97.33**
3.	Random Forest	96.41	96.43	96.36	96.42
4.	Bagging	95.45	95.31	95.59	95.43
5.	AdaBoost	94.78	94.80	95.19	94.88
6.	XGBoost	96.53	96.56	96.63	96.58
7.	Multi-Layer Perceptron	96.32	96.39	96.41	96.54
8.	Logistic Regression	96.74	96.47	96.94	96.68
9.	Decision Tree(ID3)	94.33	94.39	94.70	94.70
10.	Light GBM	96.62	96.71	96.72	96.70
11.	Extra Tree Classifier	96.44	96.64	96.52	96.46
12.	Naive Bayes	96.00	96.30	96.15	96.20

algorithm performed better among all the classifiers applied. The performance achieved with linear SVM is accuracy 97.25%, f-score 97.33%, recall 97.44%, and precision of 97.24% with approx. 1.06 standard deviation. Logistic regression, lightgbm, and XGBoost algorithms are competing in terms of performance with the linear SVM. The worst performance is reported by the decision tree classifier giving an accuracy of 94.33%, f-score of 94.70, recall of 94.70%, and precision of 94.39%. Fig. 4 presents a comparative analysis of five feature selection techniques applied. It has been observed that the principal component analysis feature selection approach is performing better among all the five techniques applied in this study. The next to this is a random forest feature selection. The worst performance is achieved with genetic algorithm-based feature selection method. Table 7 shows the execution time taken by SVM classifier(best performer) without feature selection and with five feature selection techniques. The execution time of SVM classifier without feature selection has taken largest execution time (3.5151 seconds), this shows that feature selection algorithms reduced the computational time. It is observed that SVM+PCA algorithm has taken less time (1.7142 seconds) among all the feature selection algorithms applied. Table 8 contains the performance of various approaches presented in the literature for instructor performance prediction with our study. Results presented by E. Taherifar *et al*. used PCA, clustering techniques, and decision tree algorithm for predicting instructor performance and achieved an accuracy of 93.7%,

Table 7. Execution time comparison of feature selection algorithms with SVM classifier.

Sr. No.	Feature Selection Algorithm	Execution Time(seconds)
1	Without Feature Selection	3.5151
2	**Principal Component Analysis**	**1.7142**
3	Random Forest	2.8246
4	Univariate Feature Selection	2.2358
5	Recursive Feature Elimination	2.6821
6	Genetic Algorithm	2.8960

precision: 91.5%, recall: 92.6%, f1-score: 92.0%. This performance is the highest in the literature. The next to this is Agaoglu, Mustafa, who has used a C5.0 classification algorithm and achieved an accuracy of 92.3%, precision: 94.4%, recall: 92.1%, specificity: 92.5%. Our model has used PCA, K-means clustering, and linear support vector machine algorithms and achieved an accuracy of 99.66%, precision: 99.66%, recall: 99.67%, f-score: 99.67%, which is 6% higher than the state of art results presented earlier on the prediction of instructor performance.

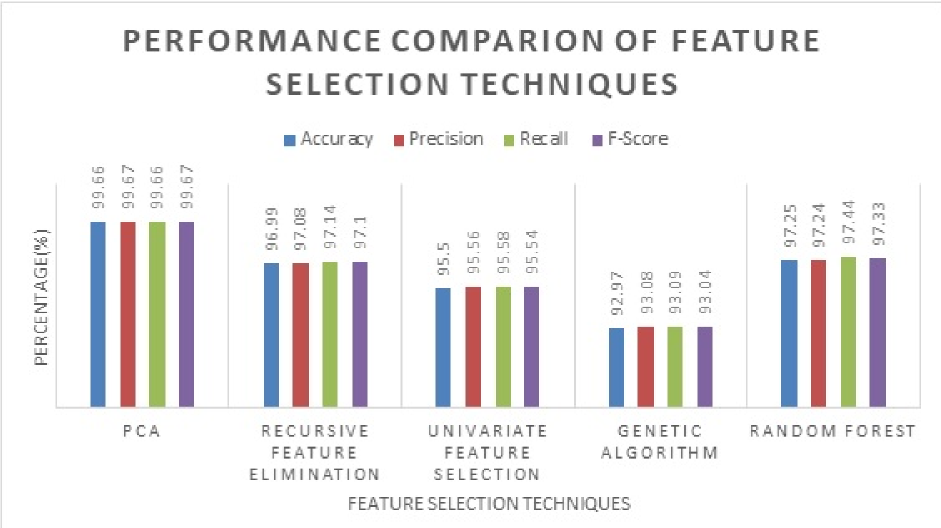

Fig. 4. Performance comparison concerning feature selection techniques.

ANOVA Statistical Test: To assess how the results (presented in Section 6.2) are statistically different, we have applied ANOVA one way test. Table 9 shows the output of the test, considering the accuracy and f-score parameters of all the five feature selection techniques used. The f statistic value 54.28 for accuracy and f statistic value 37.92 for f-score and PR value, which is very less ($PR < 0.001$) shows that feature selection algorithms are performing statistically different. Where F, sum_sq, FS, mean_sq, DF, and PR represents f value, sum of squares, feature selection, mean square, degree of freedom, and probability.

Discussions:

- In this paper, 12 classification and ensemble algorithms with 5 feature selection algorithms have been explored to find a best combination that predicts the instructor performance efficiently. It was found that Support Vector Machine classifier with principal component analysis feature selection algorithm has given the highest performance in terms of accuracy 99.66%, precision 99.67%, recall 99.66%, and f-score 99.67%.
- The results show that machine learning algorithms, along with feature selection techniques, can give better results on educational data.
- The machine learning techniques on educational data can help administrators in

Table 8. Comparison of our model with existing approaches.

Reference Paper	Algorithms Used	Performance Parameters
Ahmed Mohamed *et al*. [18]	SMO	Accuracy: 85.8%
Agaoglu, Mustafa [19]	C 5.0	Accuracy: 92.3%, Precision: 94.4%, Recall: 92.1%, Specificity: 92.5%
Suh, Sangho [20]	J4.8	Accuracy: 84.35%
P.Tamije Selvy *et al*. [21]	NB with SVM Features	Accuracy: 92.5%, Precision: 93.5%, Recall: 92%, Specificity: 91%
Asanbe, M. O. *et al*. [24]	C 4.5	Accuracy: 83.5%, Precision: 83.5%, Recall: 83.5%, F-Score:83.5%
Ajay Kumar *et al*. [25]	Naïve Bayes	Accuracy: 80.35%
Taherifar, E. *et al*. [27]	PCA+Two-Step Clustering+Decision Tree	Accuracy: 93.7%, Precision: 91.5%, Recall: 92.6%, F-Score: 92.0%
RK Hemaid *et al*. [31]	K-Nearest Neighbor	Acc:79.92%
This Work	**PCA+SVM**	**Accuracy: 99.66% Precision: 99.66%, Recall: 99.67%, F-Score: 99.67%**

Table 9. Results of ANOVA test (One way) of feature selection techniques.

Source	DF	sum_sq	mean_sq	F	PR(>F)
Based on Accuracy					
FS	4.0	327.979773	81.994943	54.275151	1.8430e-18
Residual	55.0	83.089992	1.5107		
Based on F1-Score					
FS	4.0	308.188657	77.047164	37.916804	3.2806e-15
Residual	55.0	111.760317	2.032006		

higher education to manage resources (faculty) effectively. Administrators can also find out outstanding faculty and under performing faculty.

- The dataset (5820 instances) considered is very small, that's why we have implemented classification algorithms with stratified K-fold cross validation. In future large data can be collected and processed using deep learning approaches.
- In this paper, we have processed data of the Likert scale (ranges from 0 to 5 normally) form. We can collect a dataset that contains comments by students regarding the teaching process of an instructor, which can give more insights into the teaching methods.

7. CONCLUSION

Machine Learning methods are applied in education sector for understanding administrative problems for improving the managerial process. However, most of the research is focused on predicting students' performance, as given in the literature survey. In this article, machine learning algorithms are applied to predict instructor performance based on the student's feedback. The dataset is preprocessed first and further, five feature selection algorithms are applied to extract important features. In the next step, twelve classification algorithms are applied using stratified 10-fold cross validation technique.The models are compared based on recall, accuracy, precision, and f-score. We found that the linear support vector machine algorithm with principal component analysis gives the highest accuracy 99.66%, precision 99.67%, recall of 99.66%, and f-score 99.67% among all the combination (sixty) of classification algorithms and feature selection technique. The combination of PCA and linear SVM is giving 6% better performance compared to existing work done in the literature. It is observed that the linear SVM classification algorithm performs better in all the feature selection techniques. The results are statistically verified using ANOVA test.

REFERENCES

1. R. Baker and K. Yacef, "The state of educational data mining in 2009: A review and future visions," *Journal of Educational Data Mining*, Vol. 1, 2009, pp. 3-17.
2. V. Kumar and A. Chadha, "An empirical study of the applications of data mining techniques in higher education," *International Journal of Advanced Computer Science and Applications*, Vol. 2, 2011, pp. 80-84.
3. J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, Morgan Kaufman Publishers, MA, Vol. 10, 2012, pp. 978-1.
4. N. Delavari, S. Phon-Amnuaisuk, and M. R. Beikzadeh, "Data mining application in higher learning institutions," *Informatics in Education*, Vol. 7, 2008, pp. 31-54.
5. J. Sojka, A. K. Gupta, and D. R. Deeter-Schmelz, "Student and faculty perceptions of student evaluations of teaching: A study of similarities and differences," *College Teaching*, Vol. 50, 2002, pp. 44-49.
6. V. Sarala and J. Krishnaiah, "Empirical study of data mining techniques in education system," *International Journal of Advances in Computer Science and Technology*, Vol. 4, 2015, pp. 15-21.
7. S. A. Radmacher and D. J. Martin, "Identifying significant predictors of student evaluations of faculty through hierarchical regression analysis," *The Journal of Psychology*, Vol. 135, 2001, pp. 259-268.
8. S. M. Hobson and D. M. Talbot, "Understanding student evaluations: What all faculty should know," *College Teaching*, Vol. 49, 2001, pp. 26-31.
9. M. Ayub, H. Toba, M. C. Wijanto, and S. Yong, "Modelling online assessment in management subjects through educational data mining," in *Proceedings of IEEE International Conference on Data and Software Engineering*, 2017, pp. 1-6.
10. S. Roy and S. N. Singh, "Emerging trends in applications of big data in educational data mining and learning analytics," in *Proceedings of IEEE 7th International Conference on Cloud Computing, Data Science and Engineering-Confluence*, 2017, pp. 193-198.
11. L. Rahman, N. A. Setiawan, and A. E. Permanasari, "Feature selection methods in improving accuracy of classifying students' academic performance," in *Proceedings of IEEE 2nd International Conferences on Information Technology, Information Systems and Electrical Engineering*, 2017, pp. 267-271.

12. B. K. Baradwaj and S. Pal, "Mining educational data to analyze students' performance," *arXiv Preprint*, 2012, arXiv:1201.3417.
13. M. Zaffar, M. A. Hashmani, and K. S. Savita, "Performance analysis of feature selection algorithm for educational data mining," in *Proceedings of IEEE Conference on Big Data and Analytics*, 2017, pp. 7-12.
14. A. O. Ogunde and D. A. Ajibade, "A data mining system for predicting university students' graduation grades using ID3 decision tree algorithm," *Journal of Computer Science and Information Technology*, Vol. 2, 2014, pp. 21-46.
15. P. Guleria, M. Arora, and M. Sood, "Increasing quality of education using educational data mining," in *Proceedings of IEEE 2nd International Conference on Information Management in the Knowledge Economy*, 2013, pp. 118-122.
16. I. A. A. Amra and A. Y. A. Maghari, "Students performance prediction using KNN and Naïve Bayesian," in *Proceedings of IEEE 8th International Conference on Information Technology*, 2017, pp. 909-913.
17. J. Wang, H. Y. Lv, B. Cao, and Y. Zhao, "Application of educational data mining on analysis of students' online learning behavior," *Proceedings of IEEE 2nd International Conference on Image, Vision and Computing*, 2017, pp. 1011-1015.
18. A. M. Ahmed, A. Rizaner, and A. H. Ulusoy, "Using data mining to predict instructor performance," *Procedia Computer Science*, Vol. 102, 2016, pp. 137-142.
19. M. Agaoglu, "Predicting instructor performance using data mining techniques in higher education," *IEEE Access*, Vol. 4, 2016, pp. 2379-2387.
20. S. Suh, "Learning analytics & educational data mining," https://docplayer.net/85403178-Learning-analytics-educational-data-mining-sangho-suh-computer-science-korea-university-seoul-republic-of-korea.html, 2016, pp. 1-21.
21. P. T. Selvy, D. Madhumathi, B. Preetha, and S. Sarika, "Predicting instructor performance in educational institution using data mining techniques," *International Journal of Engineering Science and Computing*, Vol. 7, 2017, pp. 5641-5645.
22. H. Suparwito, "Factors influencing the difficulty level of the subject: Machine learning technique approaches," *International Journal of Applied Sciences and Smart Technologies*, Vol. 1, 2019, pp. 65-82.
23. D. Geremew and R. Gangarde, "Data mining techniques to evaluate and predict teachers' performance in higher education," *International Journal of Innovations & Advancement in Computer Science*, Vol. 6, 2017, pp. 441-445.
24. M. O. Asanbe, A. O. Osofisan, and F. W. William, "Teachers' performance evaluation in higher educational institution using data mining technique," *International Journals of Applied Information System*, Vol. 10, 2016, pp. 10-15.
25. A. K. Pal and S. Pal, "Evaluation of teacher's performance: a data mining approach," *International Journal of Computer Science and Mobile Computing*, Vol. 2, 2013, pp. 359-369.
26. S. Mardikyan and B. Badur, "Analyzing teaching performance of instructors using data mining techniques," *Informatics in Education*, Vol. 10, 2011, pp. 245-257.
27. E. Taherifar and T. Banirostam, "Assessment of student feedback from the training course and instructor' performance through the combination of clustering methods and decision tree algorithms," *International Journal of Advanced Research in Computer Science and Software Engineering*, Vol. 6, 2016, pp. 56-64.

28. D. Buenaño-Fernández, D. Gil, and S. Luján-Mora, "Application of machine learning in predicting performance for computer engineering students: A case study," *Sustainability*, Vol. 11, 2019, p. 2833.
29. N. Chauhan, K. Shah, D. Karn, and J. Dalal, "Prediction of student's performance using machine learning," in *Proceedings of the 2nd International Conference on Advances in Science & Technology*, 2019, pp. 1-5.
30. J. Sowmiya and K. Kalaiselvi, "Instructor performance evaluation through machine learning algorithms," in *Proceedings of International Conference on Computational Vision and Bio Inspired Computing*, 2019, pp. 751-767.
31. R. K. Hemaid and A. M. El-Halees, "Improving teacher performance using data mining," *Improving Teacher Performance using Data Mining*, Vol. 4, 2015, pp. 407-4012.
32. A. Frank, "UCI machine learning repository," School of Information and Computer Science, University of California, Irvine, http://archive.ics.uci.edu/ml, 2010.
33. S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, Vol. 2, 1987, pp. 37-52.
34. I. Guyon, S. Gunn, M. Nikravesh, and L. A. Zadeh, "Feature extraction: foundations and applications, Vol. 207, Springer, 2008.
35. A. Jović, K. Brkić, and N. Bogunović, "A review of feature selection methods with applications," in *Proceedings of IEEE 38th International Convention on Information and Communication Technology, Electronics and Microelectronics*, 2015, pp. 1200-1205.
36. A. Filali, C. Jlassi, and N. Arous, "Recursive feature elimination with ensemble learning using som variants," *International Journal of Computational Intelligence and Applications*, Vol. 16, 2017, p. 1750004.
37. O. H. Babatunde, L. Armstrong, J. Leng, and D. Diepeveen, "A genetic algorithm-based feature selection," *International Journal of Electronics Communication and Computer Engineering*, Vol. 5, 2014, pp. 899-905.
38. I. Rish, *et al*., An empirical study of the naive Bayes classifier," in *Proceedings of IJCAI Workshop on Empirical Methods in Artificial Intelligence*, Vol. 3, 2001, pp. 41-46.
39. A. Holzinger, "Introduction to machine learning & knowledge extraction (MAKE)," *Machine Learning and Knowledge Extraction*, Vol. 1, 2019, pp. 1-20.
40. D. W. Aha, D. Kibler, and M. K. Albert, "Instance-based learning algorithms," *Machine Learning*, Vol. 6, 1991, pp. 37-66.
41. S. R. Safavian and D. Landgrebe, "A survey of decision tree classifier methodology," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 21, 1991, pp. 660-674.
42. T. Hastie, R. Tibshirani, and J. Friedman, "The elements of statistical learning: data mining, inference, and prediction," *Springer Science & Business Media*, 2009, pp. 101-137.
43. L. Breiman, "Bagging predictors," *Machine Learning*, Vol. 24, 1996, pp. 123-140.
44. Y. Freund, R. E. Schapire, *et al*., "Experiments with a new boosting algorithm," in *Proceedings of International Conference on Machine Learning*, Vol. 96, 1996, pp. 148-156.
45. G. Ridgeway, "Generalized boosted models: A guide to the gbm package," *Update*, Vol. 1, 2007, p. 2007.

46. P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Machine Learning*, Vol. 63, 2006, pp. 3-42.
47. A. Liaw, M. Wiener, *et al*., "Classification and regression by random forest," *R News*, Vol. 2, 2002, pp. 18-22.
48. S. Walczak, "Artificial neural networks," *Advanced Methodologies and Technologies in Artificial Intelligence, Computer Simulation, and Human-Computer Interaction*, IGI Global, 2019, pp. 40-53.
49. G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, O. Ye, and T. Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," *Advances in Neural Information Processing Systems*, 2017, pp. 3146-3154.
50. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, *et al*., "Scikit-learn: Machine learning in python," *Journal of Machine Learning Research*, Vol. 12, 2011, pp. 2825-2830.
51. Developers, Scikit-learn, "Scikit-learn documentation," 2017.

Ravinder Ahuja received the B.Tech. in Computer Science and Engineering from Kurukshetra University Kurukshetra Haryana India and M.Tech (CSE) from IIT Roorkee, Uttrakhand India. He is currently pursuing Ph.D. in Data Science from IIT Roorkee, Uttrakhand India.

S. C. Sharma received his Ph.D. degree in Electronics and Computer Science Engineering from Indian Institute of Technology Roorkee in 1991. He is currently working as a Professor at Indian Institute of Technology Roorkee. His research interests include wireless sensor networks, cloud computing, data science, and internet of things (IOT).