I.J. Modern Education and Computer Science, **2022, 3, 1-25**
Published Online June 2022 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijmecs.2022.03.01

A Comprehensive Study to Investigate Student Performance in Online Education during Covid-19

Musaddiq Al Karim
American International University-Bangladesh, Dhaka, 1229, Bangladesh
E-mail: musaddiqmk19@gmail.com

Md. Mahadi Masnad
American International University-Bangladesh, Dhaka, 1229, Bangladesh
E-mail: atime691@gmail.com

Mst. Yeasmin Ara
American International University-Bangladesh, Dhaka, 1229, Bangladesh
E-mail: yeasminararenu007@gmail.com

Mostafa Rasel
American International University-Bangladesh, Dhaka, 1229, Bangladesh
E-mail: mostafarasel9@gmail.com

Dip Nandi
American International University-Bangladesh, Dhaka, 1229, Bangladesh
E-mail: dip.nandi@aiub.edu

Received: 12 July 2021; Revised: 26 August 2021; Accepted: 16 October 2021; Published: 08 June 2022

Abstract: During the recent Covid-19 pandemic, there has been a tremendous increase in online-based learning (e-learning) activities as nearly every educational institution has transferred its programs to digital platforms. This makes it crucial to investigate student performance under this new mode of delivery. This research conducts a comparison among the traditional educational data mining techniques to detect the best performing classifier for analyzing as well as predicting students' performance in online learning platforms during the pandemic. It is achieved through extracting four datasets from X-University student information system and learning platform, followed by the application of 6 classifiers to the extracted datasets. Random Forest Classifier has demonstrated the highest accuracy in the first two out of the four datasets, while Simple Cart and Naïve Bayes Classifiers presented the same for the remainder two. All the classifiers have demonstrated medium to high TP rates, class precision and recall, ranging from 60% to 100% for almost all of the classes. This study emphasized the attributes that have a direct impact on students' performance. The outcomes of this study will assist the instructors and educational institutions to identify important factors in the analysis and prediction of student performance for online program delivery.

Index Terms: E-learning, Student performance analysis, data mining, educational data mining, Covid-19, pandemic.

1. Introduction

E-Learning refers to the process of delivering courses/ trainings via electronic media and/ or online platforms. It can be accessed by any electrical interface that can connect to the internet, such as computers and smartphones. Advances in computer-mediated communication technologies have driven and continue to aid the growth of online classes, degree programs, and educational institutions. While E-learning has been a useful option for learners throughout the world for quite some time, it saw an enormous surge during the recent Covid-19 Pandemic. During this global lockdown, e-learning has increased [23]. It is considered as the second wave of learning whereas traditional learning is considered as the first.

According to World Health Organization (WHO), coronavirus is an infectious condition that is transmitted from person to person through the droplets of saliva or nose release when an infected person coughs or sneezes. People with no significant pre-existing medical condition are subject to a mild to severe form of repository disease, which usually recovers without assisted medical care. However, there is an increased risk of developing severe disorders, especially among the elderly people, presenting pre-existing medical conditions such as coronary illness, asthma, chronic respiratory disease or cancer. Due to the less deadly and more infectious nature of the virus, the maintenance of social distancing, observing personal hygiene, and wearing face covering in public presence count for the most effective ways to reduce its spread. On March 11, 2020 WHO declared COVID-19 as a global pandemic [25]. This dreadful pandemic has also brought a significant global change in the way education is delivered. Schools and colleges all over the world had to shut down for significant periods of time in order to prevent the spread of the virus, leaving more than 1.2 billion students staying at home. In a situation where schools and colleges were shut down at a global scale, delivering face-to-face education became impossible. Therefore, schools and colleges had to quickly shift all of their academic operations to online platforms. However, educators and students have been facing difficulties in adjusting to the sudden, radical change. Moreover, in an offline/ campus-based environment, the method of analyzing student performance has been a straightforward process unlike in an online setting, where the educators cannot monitor each and every student's performance individually both in the classes and in exams. In overcoming this problem, data mining techniques have come to use in terms of informing the best method of analyzing student performance on online platforms. Numerous models have appeared to analyze educational data with different prediction algorithms.

Data mining techniques are proven to predict the most effective ways for improving student performance. Educational data mining (EDM) techniques can be utilized in analyzing student performance more efficiently, especially on online platforms. Through utilizing these techniques, it is possible to extract significant information that might assist the educational institutions to improve the quality of their delivery while also predicting students‘shortcomings. Many data mining prediction algorithms have been used in predicting student performances [26], while not all of them are proven to provide the best results.

Based on the above discussion we needed to predict how to analyze student performance in the online education environment. The main focus was on the approach of analyzing student performance using various data mining algorithms and determining the most effective algorithms in predicting and analyzing student performance in a fully online environment during this pandemic.

Hence, this research focuses on conducting a comparison among selected classifiers that are commonly used to predict and analyze students' performance based on offline environment data and determining the most effective algorithm for analyzing student performance on online education platforms. We approached this by extracting and preparing several datasets and running them through a set of selected algorithms that are known to be the best-performing EDM classifiers for student performance analysis and prediction. One of the main limitations of this research is the number of responses we got due to the Covid-19 pandemic situation. We would have got more responses in the conventional situation. We only collected data from few elective courses from one semester only. Data collected from seven courses: Algorithms, Artificial Intelligence and Expert System, Introduction to programming (Lab), Introduction to Programming (Theory), Object-Oriented Programming 1 (Java), Object-Oriented Programming 2 (C#), and Web Technologies. In future work, the research can be extended by using comprehensive data from different courses and departments.

This study is comprised of five chapters. In Section 2, we have pointed out the findings from relevant literature. Section 3 presents the data collection methods and the pre-processing and analysis of the collected data. A discussion of the key findings of this research appears in Section 4, while Chapter 5 presents the concluding remarks.

2. Literature Review

2.1. Background Study

Attribute selection is an important factor for predicting students' performance. Among all the reviewed papers the Gender attribute has been used repeatedly in almost every paper [1,3,4,5,7,9,10,13,14,15,16,17] because the learning process of male and female students are different from each other [1]. From the academic attribute, CGPA was considered as the key attribute by most of the investigators [6,7,9,12]. Moreover, previous semesters' grades, mid-exam, final exam marks, and end semester grades are also considered crucial academic attributes for anticipating students' future results [8,12,19,20]. Assignment submission, quiz marks, and lab work are also found convenient by some of the researchers in their studies [8,19,20,21]. Grades were converted to nominal values for better identifying previous performance of students [1,5,8,13,20]. Furthermore, attendance percentages [1,4,8,13,19,20] were also taken as it has a strong relation to student's success rate [19]. In a study [11], several significant criteria emerged, as well as a graphic representation of completely online students' preferences. According to the findings of the data analysis, there are eight factors that educators must consider. Structure, course management, commonly visited and participated discussion sections, chosen mode of interaction, type of preferred book, community development, and task allocation are amidst them.

The initial approach of most of the researchers was started from collecting data from surveys or student databases of different schools, colleges and universities and then data was pre-processed with various pre-processing techniques. The approach was finished with running several data mining classification, regression and clustering algorithms. These three phases are called data understanding phase, data preprocessing phase and data modeling phase [9].

Table 1. Common attributes used in the reviewed papers.

Common attributes used in the reviewed papers		
Factor	**Attribute**	**Reference**
Demographic	Gender	[1,4,5,7,9,10,13,14,15,16,17,20]
	Age	[1,9,10,14,15,16,17]
	Fathers' Occupation	[4,5,13,14,15,16]
	Mothers' occupation	[4,5,13,15,16]
	Fathers' Education Background	[4,5,13,15,16,20]
	Monthly/ annual income	[4,5,13,20]
	Mothers' Educational Background	[4,5,13,15,16,20]
	Number of siblings	[4,5,7,13,15,16,20]
Academic	CGPA	[6,7,9,12]
	Course grade	[1,10]
	Previous semester grades	[1,8,9,13,20]
	Pre-requisite course grades	[1,8,9,13]
	10th and 12th grades result	[3,4,5,10,13,20]
	Class timing	[1]
	Section size nominal	[1]
	Counseling with course instructor	[2]
	Medium of teaching	[4,5,13]
	Attendance percentage	[1,4,8,13,19,20]
	Number of absences	[1,15,16]
	Scholarship status	[1,6,7,13,15]
	High school name	[1,6,7,9,10,14,16]
	Course Load per semester	[19]
	Admission test marks	[1,7,9,10,17]
	Class test	[8,20]
	Assignment	[8,19,20,21]
	Lab evaluation	[8,19]
	Mid exam	[19]
	Final Exam/End semester	[8,12,19,20]
Psychological and socio-economic	Extra-curricular activities	[4,6,12,15,16]
	Health status	[4,10,16]
	Time spent on social media	[5,6,7]

Table 2. Common algorithms used in reviewed papers

Common algorithms used in reviewed papers		
Algorithm Category	**Algorithm Name**	**Reference**
Decision Tree	Default	[1,3,6,7,8,12,15,16,17,21]
	ID3	[8,13]
	Simple Cart	[13]
	C4.5	[8,13,19]
	CHAID classification tree	[4,13]
	J48	[7,9,12,15,16,17,20]
	Random Forest	[1,3,12,15,16,21]
	Gradient Boosted Trees	[1]
Artificial Neural Network	Deep Learning	[1,10]
	Multilayer Perceptron	[7,10,17]
	Neural Networks	[3,6,21]
K-Nearest Neighbor	K-Nearest Neighbor	[3,6,9,14,17,20]
Naïve Bayes	Naïve Bayes	[1,3,5,6,7,9,10,12,14,15,16,20,21]
CRISP-DM	CRISP-DM	[9]
Random Tree	Random Tree	[9]
REP Tree	REP Tree	[16]
Regression	Logistic Regression	[1,16]
	Generalized Linear Model	[1]
Rule Learner	JRip	[9,16,20]
	OneR	[9,16,17,20]
ZeroR	ZeroR	[16]

From different data mining techniques, decision trees, naïve Bayes, neural networks, and random forest classification algorithms were widely used by most of the researchers for prediction purposes. A greater number of researchers used WEKA, an open-source data mining tool among all other data mining tools [5,7,9,12,13,15,16,17,20]. Major number of researchers rely on decision tree algorithms to get a closer prediction result [1,2,6,7,8,9,12,13,15,16,17,19,20,21]. In a study of 5 years of student data of 231,782 records and four types of 34 attributes for predicting students' performance, the default decision tree algorithm of rapid miner gave 68.49% accuracy where the random forest algorithm gave an accuracy of 75.52% a better result [1]. In another research, after performing 10-fold cross-validation the C4.5 decision tree was given an accuracy of 80.5% with the best-selected attributes. In the study [15], researchers got 91.87% accuracy from the J48 decision tree algorithm with binary grading which is the best among all our reviewed papers. Unexpectedly, [12] got 100% accuracy applying Random Forest algorithm where students' academic information and students' activity-related data respectively from Students Information Center (SIS) and Moodle (VLE). Around 10 researchers used the Naïve Bayes algorithm for predicting and analyzing student data. Most of the papers got results from 60 to 90%. Two of the papers got more than 90% accuracy and another two got below 40% [1,2,9,10,12,13,14,15,16,21]. From all the papers that used Naïve Bayes, the highest accuracy was found 93.17% with a dataset of 500 instances pre-processed 500 records from 2000 records. Almost all the attributes were demographic attributes except an attribute named student status. The data was split into 70% training data and 30% test data and applied on RapidMiner IDE [12]. A number of studies also applied neural network [1,6,9,14,17,21] and K Nearest Neighbor (KNN) techniques for examining students‘ records [7,17]. All the accuracy levels lie between 75% to 80% for applying neural network algorithms [1,6,9,17,14,17,21]. A study containing 20 parameters from 10330 students from 2007-2009 had been got 60% accuracy implementing the K-Nearest-Neighbor algorithm [9] where the parameter value k was 100 and 250. Another study [17] got a better accuracy of 73.59% where the value of k was 50.

There are also a few other data mining techniques like deep learning, logistic regression, generalized linear model, clustering, rule learner, specific efficient optimization(SMO) that were used by several researchers for forecasting students datasets [1,2,3,9,12,16,17]. Researchers [9] used two rule learners OneR and JRip in this study where JRip performed slightly better than OneR. In [17], the researcher also used JRip and OneRip got an accuracy of 74.11% and 76.73% respectively. A study [1] used Logistic Regression and Generalized Linear Model (GMO) for exploring student performance. In this regression algorithm category logistic regression got a good result of 72.88%. On the other hand, slightly fall behind with an accuracy of 58.7%. Unexpectedly, researchers [12] found 100% accuracy and kappa value 1 with the Specific Efficient Optimization (SMO) algorithm. But it is a matter of concern that they used a dataset of only 22 students. Most of the researchers moved towards similar goal applying different techniques and resources. Some researchers used classification algorithm and some used logistic regression to find out the best result. Here Table 1 and Table 2 shows the algorithms and attributes with high impact in the reviewed papers.

2.2. Classifier selection

Classifier Selection According to the findings of our discussion mentioned above, best educational data mining algorithms to analyze and predict students‘ performance are Naïve Bayes, Decision Trees, Random Forest algorithm, Random Tree algorithm, Artificial Neural networks, Cart, ID3,C4,5, Regression method etc. However, our research focuses on comparison of educational data mining algorithms on different datasets with changes in attributes and number of instances. Here, in Table: 3, we have mentioned our selected algorithm for this study.

Table 3. Selected data mining algorithms

Algorithm Selection	
Algorithm Category	**Algorithm Name**
Naïve Bayes	Naïve Bayes
Decision Tree	Decision Tree (J48)
Decision Tree	Random Forest
Decision Tree	Random Tree
Decision Tree	Simple Cart

3. Data Collection, Preprocessing and Analysis

3.1. Data collection

In the previous section, we have mentioned some potential attributes from different articles, however due to pandemic and shifting to online platforms, not all of those attributes were available. We extracted 589 instances of seven courses: Algorithms, Artificial Intelligence and Expert System, Introduction to programming (Lab), Introduction to Programming (Theory), Object Oriented Programming 1 (Java), Object Oriented Programming 2 (C#) and Web Technologies from the central server of ―X University. But all of these courses did not have the same attributes and evaluation criteria, so we created four different datasets based on their common attributes and our research will emphasize on these.

3.2. Preprocessing and analysis

In data mining research, data preprocessing is a critical procedure. By adapting this data mining technique, we transformed raw data to understandable and similar format. Total marks were omitted to reduce biasing in the dataset and Class attribute was categorized into predefined measures based on total marks to make the model more efficient. Below Table 4 presents the preprocessed data and interpretation of the attributes.

Table 4. Preprocessed data and interpretation of the attributes

Attributes	Datasets	Preprocess and Interpretation
Gender	1,2,3,4	It was generated based on their names.
Attendance	1,2,4	It was collected from Microsoft Teams Attendance generator. Some of the courses did not have the same number of classes throughout the semester. So they were scaled to 10. We divided the total attendance of an individual student with total classes throughout the semester and then multiplied by 10. After multiplication the ceiling values (0.05) were taken.
Absence	1,2,4	Absences were calculated by subtracting attendance value from 10.
Mid-term Attendance	3	It was collected from Microsoft Teams Attendance generator. Some of the courses did not have the same number of classes throughout Mid-term. So they were scaled to 10. We divided the total attendance of an individual student with total classes throughout the semester and then multiplied by 10. After multiplication the ceiling values (0.05) were taken.
Mid-term Absence	3	Mid-term Absences were calculated by subtracting Mid-term attendance value from 10.
Final-term Attendance	3	Following the same procedure of Mid-term attendance, data for this attribute were calculated after mid-term exam to before final term examination.
Final-term absence	3	By subtracting Final term attendance value from 10.
Quiz 1	2,4	Most of the courses had 4 quizzes (2 in Mid-term and 2 in final term). So for Quiz 1 we took the average value for mid-term quizzes.
Quiz 2	2,4	The average values of Final term quizzes are considered as Quiz 2 in this dataset.
Mid-term Quiz 1	3	This attribute represents data of first quiz marks of Mid-term.
Mid-term Quiz 2	3	This attribute represents data from the second quiz of Mid-term.
Final-term Quiz 1	3	This attribute represents data of first quiz marks of the final term.
Final-term Quiz 2	3	This attribute represents data from the second quiz of the final term.
CGPA	4	CGPA until current semester (While extracting data).
Lab performance	4	Lab performance has a 20% impact on Midterm and Final term results. It is collected based on lab quiz, lab exam and lab report marks.
Mid-term	1,2,3,4	Midterm grade is based on mid-term attendance, mid quiz marks, mid assignments, mid-term lab exams and Mid-term exam.
Final term	1,2,3,4	Final term grade is based on final term attendance, final term quiz marks, final term assignments, final term lab exams and final term exam.
Class	1,2,3,4	Students were classified based on their total marks, which represented 40% of Mid-term and 60% of final term grades, where: High Performer: Total marks between 85 and 100. Medium Performer: Total Marks less than 85 but greater or equal to 70. Low performer: Total marks between 50 and 69. Failure: Total marks less than 50. Dropped: Students' who dropped the course were marked as -2.

Dataset Version 1:

Attributes for our first dataset are mentioned in Table 5. There are 6 attributes and 589 instances of seven courses in this dataset: Algorithms, Artificial Intelligence and Expert System, Introduction to Programming (Lab), Introduction to Programming (Theory), Object Oriented Programming 1 (Java), Object Oriented Programming 2 (C#), and Web Technologies. Table 5 presents the performance summary of dataset version 1 on 6 classifiers. Table 6 manifests the performance summary of 6 classifiers on this dataset.

Table 5. Dataset version 1

Dataset Version 1		
Attribute name	**Type**	**Summary**
Gender	Nominal	F = 133 M = 456
Attendance(10)	Numeric	Min = 0 Maximum = 10 Mean = 8.918 StdDev = 1.892
Absence(10)	Numeric	Min = 0 Maximum = 10 Mean = 1.082 StdDev = 1.892

Mid-term(100)	Numeric	Min = -2 Maximum = 99 Mean = 74.803 StdDev = 16.776
Final-Term(100)	Numeric	Min= -2 Maximum = 100 Mean = 73.302 StdDev = 20.847
Class	Nominal	High Performer (Total marks>85) = 203 Medium Performer (70<Total marks<85) = 262 Low Performer(50<Total marks<69) = 93 Failure(0<Total marks<50) = 8 Dropped(Marked as -2) = 23

Table 6. Performance summary of all algorithms on Dataset Version 1

Performance Summary (dataset version 1)			
Naïve Bayes	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	510	86.5874 %
	Incorrectly classified instances	79	13.4126 %
Decision tree (J48)	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	510	89.8132 %
	Incorrectly classified instances	79	10.1868 %
Random Forest	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	531	90.1528 %
	Incorrectly classified instances	58	9.8472 %
Random Tree	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	507	86.0781 %
	Incorrectly classified instances	82	13.9219 %
Simple Cart	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	508	86.2479 %
	Incorrectly classified instances	81	13.7521 %
Multilayer Perceptron	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	516	87.6061 %
	Incorrectly classified instances	73	12.3939 %

Dataset Version 2:

This dataset is based on 5 courses which are Algorithms, Introduction to Programming (Theory), Object Oriented Programming 1 (Java), Object Oriented Programming 2 (C#) and Web Technologies. In this dataset we have added 2 new attributes with our previous version. Now in this version of the dataset, there are 8 attributes with 330 instances. We had to drop 259 instances from our previous dataset, as two attributes of this dataset were not available in the previous version. In this dataset we have added Quiz 1 and Quiz 2 attributes, as they have a 40% impact on analyzing students' performance. Attributes of our second dataset is described in the table (Table 7.) below, also Table 8 shows the performance summary of the classifiers on this dataset.

Table 7. Dataset version 2

Dataset Version 2		
Attribute name	**Type**	**Summary**
Gender	Nominal	F = 72 M = 258
Attendance(10)	Numeric	Min = 0 Maximum = 10 Mean = 8.788 StdDev = 2.005
Absence(10)	Numeric	Min = 0 Maximum = 10 Mean = 1.212 StdDev = 2.005
Quiz 1	Numeric	Min = 0 Maximum = 20 Mean = 9.623 StdDev = 3.699
Quiz 2	Numeric	Min = 0 Maximum = 18 Mean = 6.337 StdDev = 2.423

Mid-term(100)	Numeric	Min = -2 Maximum = 97.5 Mean = 72.045 StdDev = 18.266
Final-Term(100)	Numeric	Min= -2 Maximum = 100 Mean = 70.87 StdDev = 24.342
Class	Nominal	High Performer (Total marks>85) = 114 Medium Performer (70<Total marks<85) = 125 Low Performer(50<Total marks<69) = 65 Failure(0<Total marks<50) = 5 Dropped(Marked as -2) = 21

Table 8. Performance summary of all algorithms on Dataset Version 2

Performance Summary (dataset version 2)				
Naïve Bayes	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	284	86.0606 %
	Incorrectly classified instances	46	13.9394 %
Decision tree (J48)	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	305	92.4242 %
	Incorrectly classified instances	25	7.5758 %
Random Forest	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	311	94.2424 %
	Incorrectly classified instances	19	5.7576 %
Random Tree	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	278	84.2424 %
	Incorrectly classified instances	52	15.7576 %
Simple Cart	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	304	92.1212 %
	Incorrectly classified instances	26	7.8788 %
Multilayer Perceptron	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	289	87.5758 %
	Incorrectly classified instances	41	12.4242 %

Dataset Version 3:

This dataset is developed based on 4 courses: Algorithms, Introduction to Programming (Theory), Object Oriented Programming 1 (Java) and Object Oriented Programming 2 (C#). It has 12 attributes with 280 instances. Attendance, absence, and quiz marks are divided into Midterm and final term category. Other attributes are the same as previous datasets. Attributes of our third dataset are described in the table (Table 9) below and performance summary of dataset version 3 on our selected classifiers are presented in Table 10.

Table 9. Dataset version 3

Dataset Version 3		
Attribute name	**Type**	**Summary**
Gender	Nominal	F = 61 M = 219
Mid-term Attendance(10)	Numeric	Min = 0 Maximum = 10 Mean = 9.011 StdDev = 1.893
Mid-term Absence(10)	Numeric	Min = 0 Maximum = 10 Mean = 0.989 StdDev = 1.893
Mid-term Quiz 1	Numeric	Min = 0 Maximum = 20 Mean = 10.782 StdDev = 4.33
Mid-term Quiz 2	Numeric	Min = 0 Maximum = 19 Mean = 10.143 StdDev = 3.982
Mid-term(100)	Numeric	Min = -2 Maximum = 97.5 Mean = 73.227 StdDev = 17.049

Final-term Attendance(10)	Numeric	Min = 0 Maximum = 10 Mean = 8.796 StdDev = 2.388
Final-term Absence(10)	Numeric	Min = 0 Maximum = 10 Mean = 1.505 StdDev = 2.736
Final-term Quiz 1	Numeric	Min = 0 Maximum = 20 Mean = 12.314 StdDev = 6.026
Final-term Quiz 2	Numeric	Min = 0 Maximum = 20 Mean = 13.411 StdDev = 6.161
Final-Term(100)	Numeric	Min= -2 Maximum = 100 Mean = 71.611 StdDev = 24.253
Class	Nominal	High Performer (Total marks>85) = 101 Medium Performer (70<Total marks<85) = 103 Low Performer(50<Total marks<69) = 54 Failure(0<Total marks<50) = 5 Dropped(Marked as -2) = 17

Table 10. Performance summary of all algorithms on Dataset Version 3

Performance Summary (dataset version 3)			
Naïve Bayes	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	234	83.5714 %
	Incorrectly classified instances	46	16.4286 %
Decision tree (J48)	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	261	93.2143 %
	Incorrectly classified instances	19	6.7857 %
Random Forest	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	262	93.5714 %
	Incorrectly classified instances	18	6.4286 %
Random Tree	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	236	84.2857 %
	Incorrectly classified instances	44	15.7143 %
Simple Cart	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	263	93.9286 %
	Incorrectly classified instances	17	6.0714 %
Multilayer Perceptron	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	249	88.9286 %
	Incorrectly classified instances	31	11.0714 %

Dataset Version 4:

In this dataset, we have considered 10 attributes of 2 courses: Algorithms and Web technologies. In this dataset there are two new attributes which are CGPA and Lab performance. CGPA is one of the most potential attributes to analyze and predict students' performance. There are 91 instances only. Due to lack of data, we couldn't consider other courses. Attributes of our fourth dataset are described in the table (Table 11) below and Table 12 shows the performance summary of the classifiers on dataset version 4.

Table 11. Dataset version 4

Dataset Version 4		
Attribute name	**Type**	**Summary**
CGPA	Numeric	Min = 2.51 Maximum = 3.98 Mean = 3.235 StdDev = 0.357
Gender	Nominal	F = 22 M = 69
Attendance(10)	Numeric	Min = 0 Maximum = 10 Mean = 7.583 StdDev = 2.394

Absence(10)	Numeric	Min = 0 Maximum = 10 Mean = 2.417 StdDev = 2.394
Quiz 1	Numeric	Min = 0 Maximum = 20 Mean = 7.126 StdDev = 4.084
Quiz 2	Numeric	Min = 0 Maximum = 18 Mean = 8.368 StdDev = 3.322
Lab Performance	Numeric	Min = 0 Maximum = 20 Mean = 12.242 StdDev = 4.507
Mid-term(100)	Numeric	Min = -2 Maximum = 90 Mean = 66.927 StdDev = 22.473
Final-Term(100)	Numeric	Min= -2 Maximum = 100 Mean = 65.099 StdDev = 26.389
Class	Nominal	High Performer (Total marks>85) = 19 Medium Performer (70<Total marks<85) = 41 Low Performer(50<Total marks<69) = 21 Failure(0<Total marks<50) = 0 Dropped(Marked as -2) = 10

Table 12. Performance summary of all algorithms on Dataset Version 4

Performance Summary (dataset version 4)			
Naïve Bayes	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	84	92.3077 %
	Incorrectly classified instances	7	7.6923 %
Decision tree (J48)	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	75	82.4176 %
	Incorrectly classified instances	16	17.5824 %
Random Forest	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	74	81.3187 %
	Incorrectly classified instances	17	18.6813 %
Random Tree	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	68	74.7253 %
	Incorrectly classified instances	23	25.2747 %
Simple Cart	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	75	82.4176 %
	Incorrectly classified instances	16	17.5824 %
Multilayer Perceptron	**Evaluation Criteria**	**Instance count**	**Percentage**
	Correctly classified instances	78	85.7143 %
	Incorrectly classified instances	13	14.2857 %

4. Discussion

This section presents the key findings of our analysis and summarizes the performance of the algorithms on our datasets. It is comprised of two sections: (1) summarizing students' performance based on accuracy, kappa statistical value and confusion matrices, (2) summarizing classifier performance based on TP rate, FP rate and precision.

4.1. Summarizing students' performance prediction based on accuracy, kappa statistic value and confusion matrices

In this study, four different datasets were evaluated on the selected algorithms. Table 13 presents the accuracy with kappa statistical value of six classifiers for the four datasets. We used 10 folds cross validation method for all classifiers. As can be seen from Table 1, for the dataset version 1, the Random Forest algorithm had demonstrated the highest accuracy (90.1528%) with 0.85 kappa statistical value, whereas the other algorithms had performed within a range of 86.0781% to 89.8132%. This means that, overall; all classifiers had correctly predicted 86 to 90% classes for this dataset. The Kappa statistical values for other algorithms were between 0.7881 and 0.8132.

For the dataset version 2, Random Forest algorithm had continued to demonstrate the best performance with an accuracy of 94.2424% for a Kappa statistical value of 0.9171, which stands higher than the previous instance. The accuracy of the other algorithms came close to the Random forest algorithm, within a range of 84.2424% to 92.4242% and the Kappa values of 0.7736 to 0.8915. However, for the dataset version 3, the Simple Cart classifier had demonstrated the best performance with an accuracy of 93.9286%, while the Random forest and decision tree (J48) classifiers performed the closest with a difference of less than 1%. The other three algorithms (Naïve Bayes, Random tree and Multilayer perceptron) had performed reasonably well, with accuracies of 83.5714%, 84.2857% and 88.9286%, respectively. The Kappa values for all algorithms stood within a range of 0.7736 to 0.9069. Naïve Bayes algorithm had, on the other hand, demonstrated the best performance for the dataset version 4, with an accuracy of 92.3077% and a Kappa value of 0.7444. The other classifiers had performed reasonably well for this dataset, within an accuracy range of 74.7253% to 85.7143%. These variations in performance may have occurred due to the variations that are present in attributes and instance counts. The attributes available in Dataset version 1 are the common attributes for all four datasets. Moreover, it contains the highest number of instances. Comparing to dataset version 1, our second dataset (Dataset version 2) have two new attributes and 259 fewer data. According to the performance summary table (Tale 13.) below, the changes in dataset version 2 result in a slight change in accuracy for Naïve Bayes and Multilayer perceptron, an increase in accuracy for Decision tree (j48), Random forest, and Simple cart classifier, but a drop in accuracy for Random tree. Also, the kappa statistical value increased for almost all the classes except for the Random tree classifier. While four additional attributes were introduced in Dataset version 3 over Dataset version 2, 50 instances had to be discarded. Meanwhile, the accuracy of the Decision tree (j48), Simple cart, Multilayer perceptron, and a small modification in the Random tree improved while it decreased for Naïve Bayes and Random forest classifiers. The Random tree's Kappa statistical value remained constant, while the value for Random forest decreased and the other four classifiers improved. We eliminated four attributes from our previous dataset (Dataset version 3) and added two new attributes in dataset version 4, which has just 91 instances. Except for Nave Bayes, nearly all classifiers' accuracy decreased for this dataset. Also, there is a significant change in kappa statistical value comparing to previous datasets.

Table 13. Performance summary of all algorithms on Dataset version (1-4)

Algorithms	**Evaluation Criteria**	**Dataset Version 1**	**Dataset Version 2**	**Dataset Version 3**	**Dataset Version 4**
Naïve Bayes	Accuracy	86.5874 %	86.0606 %	83.5714 %	92.3077 %
	Kappa Statistic	0.8132	0.8915	0.9024	0.7444
Decision Tree (J48)	Accuracy	89.8132 %	92.4242 %	93.2143 %	82.4176 %
	Kappa Statistic	0.8132	0.8915	0.9024	0.7444
Random Forest	Accuracy	90.1528 %	94.2424 %	93.5714 %	81.3187 %
	Kappa Statistic	0.85	0.9171	0.9069	0.7274
Random Tree	Accuracy	86.0781 %	84.2424 %	84.2857 %	74.7253 %
	Kappa Statistic	0.7881	0.7736	0.7736	0.6359
Simple Cart	Accuracy	86.2479 %	92.1212 %	93.9286 %	82.4176 %
	Kappa Statistic	0.7916	0.8869	0.9125	0.7463
Multilayer Perceptron	Accuracy	87.6061 %	87.5758 %	88.9286 %	85.7143 %
	Kappa Statistic	0.8114	0.8213	0.8402	0.7931

The resultant matrices are shown in Tables 14 – 17. For a comprehensive understanding, the first confusion matrix i.e., the Naïve Bayes classifier on dataset 1 is described here, where all accurately predicted classes are located diagonally in each matrix. In the first column, out of a total count of 265 medium performer (MP) students' class, 228 instances were predicted accurately to be medium performers, resulting in a recall count of (228/265 =) 0.87; whereas in the first row, 262 instances were predicted to be medium performers, resulting in a class precision of (228/262 =) 0.87. In the second column, 77 instances out of a total count of 98 (i.e., 77+17+4) were predicted accurately to be low performers, with a class recall and class precision of 0.828 and 0.786, respectively. Naïve Bayes classifier demonstrated 0.913 class precision and recall for the 'dropped' class. In a similar way, 178 instances out of a total count of 195 in the 'High Performer' class were predicted accurately, with a class recall and class precision of 0.877 and 0.913, respectively. Finally, out of a total count of 8 instances in the 'Failure' class, 6 instances were predicted accurately, whereas the remainder two were misclassified as 'Dropped', resulting in a class recall and class precision of 0.75 and 0.75, respectively. All the remainder confusion matrices for the 4 datasets and the 6 classifiers were calculated following the same procedure. As there was no 'Failure' class in dataset 4, in Table 39, this class was dropped. As evident throughout the Tables 36 – 39, almost all of the classes had demonstrated a reasonable class recall and class precision performance.

Table 14. Resultant matrices of all algorithms on Dataset version 1

Confusion Matrices for dataset version 1							
Naïve Bayes		Actual					Class Precision
		MP	LP	D	HP	F	
Predicted	MP	**228**	17	0	17	0	0.86
	LP	16	**77**	0	0	0	0.786
	D	0	0	**21**	0	2	0.913
	HP	21	4	0	**178**	0	0.913
	F	0	0	2	0	**6**	0.75

Class Recall		0.87	0.828	0.913	0.877	0.75	
Decision tree (J48)		Actual					Class Precision
		MP	LP	D	HP	F	
Predicted	MP	**227**	12	0	23	0	0.904
	LP	10	**83**	0	0	0	0.874
	D	0	0	**23**	0	0	0.958
	HP	14	0	0	**189**	0	0.892
	F	0	0	1	0	**7**	1
Class Recall		0.866	0.892	1	0.931	0.875	
Random Forest		Actual					Class Precision
		MP	LP	D	HP	F	
Predicted	MP	232	9	0	21	0	0.899
	LP	10	83	0	0	0	0.902
	D	0	0	22	0	1	0.957
	HP	15	0	0	188	0	0.9
	F	1	0	1	0	6	0.857
Class Recall		0.885	0.892	0.957	0.926	0.75	
Random Tree		Actual					Class Precision
		MP	LP	D	HP	F	
Predicted	MP	221	15	0	26	0	0.86
	LP	14	79	0	0	0	0.832
	D	1	1	19	0	2	0.905
	HP	21	0	0	182	0	0.875
	F	0	0	2	0	6	0.75
Class Recall		0.844	0.849	0.826	0.897	0.75	
Simple Cart		Actual					Class Precision
		MP	LP	D	HP	F	
Predicted	MP	217	15	0	30	0	0.861
	LP	10	83	0	0	0	0.847
	D	0	0	23	0	0	0.958
	HP	25	0	0	178	0	0.856
	F	0	0	1	0	7	1
Class Recall		0.828	0.892	1	0.877	0.875	
Multilayer Perceptron		Actual					Class Precision
		MP	LP	D	HP	F	
Predicted	MP	239	13	0	10	0	0.866
	LP	6	86	1	0	0	0.86
	D	0	0	17	0	6	0.739
	HP	31	1	0	171	0	0.945
	F	0	0	5	0	3	0.333
Class Recall		0.912	0.925	0.739	0.842	0.375	

Table 15. Resultant matrices of all algorithms on Dataset version 2

Confusion Matrices for dataset version 2							
Naïve Bayes		Actual					Class Precision
		MP	LP	D	HP	F	
Predicted	MP	98	9	0	18	0	0.852
	LP	8	57	0	0	0	0.851
	D	0	0	21	0	0	0.955
	HP	9	1	0	104	0	0.852
	F	0	0	1	0	4	1
Class Recall		0.784	0.877	1	0.912	0.8	
Decision tree (J48)		Actual					Class Precision
		MP	LP	D	HP	F	

Predicted	MP	109	10	0	6	0	0.924
	LP	2	63	0	0	0	0.863
	D	0	0	21	0	0	1
	HP	7	0	0	107	0	0.947
	F	0	0	0	0	5	1
Class Recall		0.872	0.969	1	0.939	1	
Random Forest		Actual					Class Precision
		MP	LP	D	HP	F	
Predicted	MP	232	9	0	21	0	0.934
	LP	10	83	0	0	0	0.912
	D	0	0	22	0	1	1
	HP	15	0	0	188	0	0.957
	F	1	0	1	0	6	1
Class Recall		0.912	0.954	1	0.965	0.8	
Random Tree		Actual					Class Precision
		MP	LP	D	HP	F	
Predicted	MP	102	11	1	11	0	0.823
	LP	12	51	2	0	0	0.81
	D	0	1	19	0	1	0.792
	HP	10	0	0	103	1	0.904
	F	0	0	2	0	3	0.6
Class Recall		0.816	0.785	0.905	0.904	0.6	
Simple Cart		Actual					Class Precision
		MP	LP	D	HP	F	
Predicted	MP	111	11	0	3	0	0.902
	LP	4	61	0	0	0	0.847
	D	0	0	21	0	0	1
	HP	7	0	0	107	0	0.973
	F	1	0	0	0	4	1
Class Recall		0.888	0.938	**1**	0.939	0.8	
Multilayer Perceptron		Actual					Class Precision
		MP	LP	D	HP	F	
Predicted	MP	239	13	0	10	0	0.899
	LP	6	86	1	0	0	0.908
	D	0	0	17	0	6	0.739
	HP	31	1	0	171	0	0.891
	F	0	0	5	0	3	0
Class Recall		0.856	0.908	0.81	0.93	0	

Table 16. Resultant matrices of all algorithms on Dataset version 3

Confusion Matrices for dataset version 3							
Naïve Bayes		Actual					Class Precision
		MP	LP	D	HP	F	
Predicted	MP	76	10	0	17	0	0.817
	LP	8	46	0	0	0	0.807
	D	0	0	17	0	0	0.944
	HP	9	1	0	91	0	0.843
	F	0	0	1	0	4	1
Class Recall		0.738	0.852	1	0.901	0.8	
Decision tree (J48)		Actual					Class Precision
		MP	LP	D	HP	F	
Predicted	MP	95	4	0	4	0	0.896
	LP	1	53	0	0	0	0.93
	D	0	0	17	0	0	1
	HP	10	0	0	91	0	0.958
	F	0	0	0	0	5	1

Class Recall		0.922	0.981	1	0.901	1	
Random Forest		Actual					Class Precision
		MP	LP	D	HP	F	
Predicted	MP	95	2	0	6	0	0.922
	LP	4	50	0	0	0	0.943
	D	0	1	16	0	0	0.941
	HP	3	0	0	98	0	0.942
	F	1	0	1	0	3	1
Class Recall		0.922	0.926	0.941	0.97	0.6	
Random Tree		Actual					Class Precision
		MP	LP	D	HP	F	
Predicted	MP	82	4	1	15	1	0.854
	LP	4	47	3	0	0	0.855
	D	2	3	12	0	0	0.706
	HP	7	1	0	92	1	0.86
	F	1	0	1	0	3	0.6
Class Recall		0.796	0.87	0.706	0.911	0.6	
Simple Cart		Actual					Class Precision
		MP	LP	D	HP	F	
Predicted	MP	91	5	0	7	0	0.948
	LP	0	54	0	0	0	0.915
	D	1	0	16	0	0	1
	HP	3	0	0	98	0	0.933
	F	1	0	0	0	4	1
Class Recall		0.883	1	0.941	0.97	0.8	
Multilayer Perceptron		Actual					Class Precision
		MP	LP	D	HP	F	
Predicted	MP	90	5	0	8	0	0.865
	LP	7	47	0	0	0	0.887
	D	0	0	15	0	2	0.938
	HP	7	1	0	93	0	0.921
	F	0	0	1	0	4	0.667
Class Recall		0.874	0.87	0.882	0.921	0.8	

Table 17. Resultant matrices of all algorithms on Dataset version 4

Confusion Matrices for dataset version 4						
Naïve Bayes		Actual				Class Precision
		MP	LP	D	HP	
Predicted	MP	39	1	0	1	0.886
	LP	2	19	0	0	0.95
	D	0	0	10	0	1
	HP	3	0	0	16	0.941
Class Recall		0.951	0.905	1	0.842	
Decision tree (J48)		Actual				Class Precision
		MP	LP	D	HP	
Predicted	MP	33	5	0	3	0.805
	LP	4	17	0	0	0.773
	D	0	0	10	0	1
	HP	40	0	0	15	0.833
Class Recall		0.805	0.81	1	0.789	
Random Forest		Actual				Class Precision
		MP	LP	D	HP	
Predicted	MP	33	6	0	2	0.786
	LP	5	16	0	0	0.727
	D	0	0	10	0	1
	HP	4	0	0	15	0.882
Class Recall		0.805	0.762	1	0.789	
Random Tree		Actual				Class Precision
		MP	LP	D	HP	

Predicted	MP	30	7	1	3	0.789
	LP	6	14	1	0	0.583
	D	0	3	7	0	0.778
	HP	2	0	0	17	0.85
Class Recall		0.732	0.667	0.7	0.895	
Simple Cart		Actual				Class Precision
		MP	LP	D	HP	
Predicted	MP	32	6	0	3	0.821
	LP	4	17	0	0	0.739
	D	0	0	10	0	1
	HP	3	0	0	16	0.842
Class Recall		0.78	0.81	1	0.842	
Multilayer Perceptron		Actual				Class Precision
		MP	LP	D	HP	
Predicted	MP	34	5	0	2	0.85
	LP	4	17	0	0	0.773
	D	0	0	10	0	1
	HP	2	0	0	17	0.895
Class Recall		0.829	0.81	1	0.895	

4.2 Summarizing classifier performance based on TP rate, FP rate and precision

Naïve Bayes: After applying Naïve Bayes algorithm in Weka, the highest accuracy (92.31%) was found on dataset-4, having 91 instances with 10 attributes. On the other datasets, it fell between 83 to 87%, denoting the classifier correctly predicting 83 to 87 instances out of 100. Figure 1 shows the performance summary for Naïve Bayes classifier on all datasets.

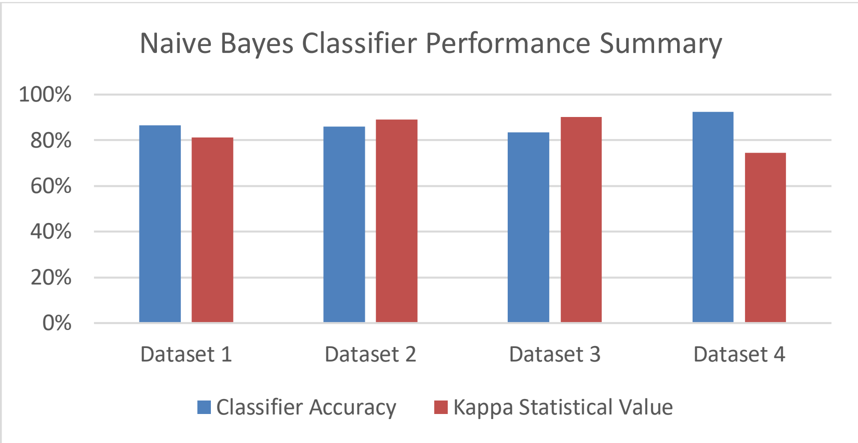

Fig.1. Naïve Bayes classifier performance summary

On dataset-1 (Table 18), the ‘Dropped’ class has exhibited the highest TP rate with Naïve Bayes technique, while the ‘Dropped’ and ‘High Performer’ classes produced the same precision, standing to be the highest for this dataset. The ‘Failure’ class had the lowest false positive rate compared to the other classes. Overall, the FP rates — falling between 0.3% and 11.3% — demonstrated a reasonably well performance.

Table 18. Performance summary (TP rate, FP rate and Precision) of Naïve Bayes classifier on Dataset version 1

Class	Naïve Bayes on Dataset 1		
	TP Rate	FP Rate	Precision
Medium Performer	0.870	0.113	0.860
Low Performer	0.828	0.042	0.786
Dropped	0.913	0.004	0.913
High Performer	0.877	0.044	0.913
Failure	0.750	0.003	0.750
Weighted Average	0.866	0.072	0.867

On dataset-2 (Table 19), the ‘Dropped’ class demonstrated the highest possible TP rate (100%), while the ‘High Performer’ class had crossed 90% TP rate, which indicated that the Naïve Bayes classifier had accurately classified all of the ‘Dropped’ class instances and more than 90% of the ‘High Performer’ instances. In terms of precision, the ‘Failure’ class exhibited the highest possible precision, while the ‘Dropped’ class revealed a high precision (95.5%). Moreover, the precision values for the ‘Medium Performer’ (85.2%), ‘Low Performer’ (85.1%), and ‘High Performer’

(85.2%) classes stood fairly well. In terms of the returned result dataset, the Naïve Bayes classifier performed with more than 85% accuracy. The FP rates were also excellent for this dataset, ranging between 0% to 8.3%.

Table 19. Performance summary (TP rate, FP rate and Precision) of Naïve Bayes classifier on Dataset version 2

Class	Naïve Bayes on Dataset 2		
	TP Rate	FP Rate	Precision
Medium Performer	0.784	0.083	0.852
Low Performer	0.877	0.038	0.851
Dropped	1.000	0.003	0.955
High Performer	0.912	0.083	0.852
Failure	0.800	0.000	1.000
Weighted Average	0.861	0.068	0.861

On dataset-3 (Table 20), the 'Dropped' class had demonstrated the highest possible TP rate (100%), while the 'Failure' class demonstrated the highest possible precision (100%). The 'High Performer' class exhibited a better TP rate (90%) than the remaining classes. While the 'Failure' class had produced the lowest FP rate, the other classes also demonstrated excellent FP rates (less than 10%).

Table 20. Performance summary (TP rate, FP rate and Precision) of Naïve Bayes classifier on Dataset version 3

Class	Naïve Bayes on Dataset 3		
	TP Rate	FP Rate	Precision
Medium Performer	0.738	0.096	0.817
Low Performer	0.852	0.049	0.807
Dropped	1.000	0.004	0.944
High Performer	0.901	0.095	0.843
Failure	0.800	0.000	1.000
Weighted Average	0.836	0.079	0.835

The 'Dropped' class had exhibited the highest possible TP rate in dataset-4 (Table 21), similar to the second and third datasets. This class correctly classified all the returned result dataset; obtained 100% precision, while the 'High Performer' and the 'Low Performer' classes exhibited more than 90% precision.

Table 21. Performance summary (TP rate, FP rate and Precision) of Naïve Bayes classifier on Dataset version 4

Class	Naïve Bayes on Dataset 4		
	TP Rate	FP Rate	Precision
Medium Performer	0.951	0.100	0.886
Low Performer	0.905	0.014	0.950
Dropped	1.000	0.000	1.000
High Performer	0.842	0.014	0.941
Weighted Average	0.923	0.051	0.925

Decision Tree Classifier: We used the Weka Decision tree (J48) classifier with 10-fold cross validity in this analysis. For dataset-3, the maximum accuracy for the decision tree (J48) classifier was 93.21 percent, with a Kappa statistic of 0.9024. In the other datasets, the percentages ranged from 82 to 93%, indicating that the classifiers had accurately estimated 82 to 93 out of every 100 cases. Figure 63 shows the performance summary of decision tree (J48) classifier for all datasets.

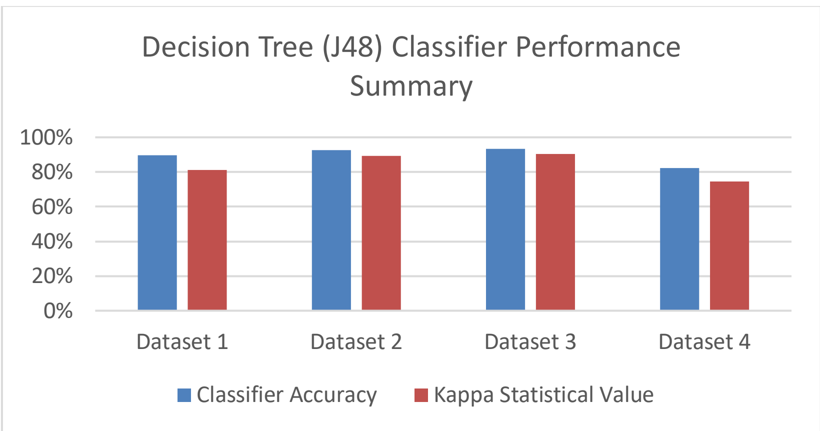

Fig.2. Decision Tree (J48) classifier performance summary

On dataset-1(Table 22), this classifier had accurately classified 89.8132% instances out of a total count of 589 instances, with a 0.8132 Kappa value. Table 44 presents the TP rate, FP rate and precision of decision tree (j48) classifier on dataset-1. The results show that the 'Dropped' class had exhibited the highest true positive value (100%). The 'Medium Performer' (86.6%),'Low Performer' (89.2%), 'High Performer' (93.1%) and 'Failure' (87.5%) classes performed with high TP rate as well, while the precision for all the classes was high. The 'Failure' class had performed with the highest possible precision, followed by the 'Dropped' (95.8%), 'Medium Performer' (90.4%), and 'High Performer' (89.2%) and 'Low Performer' (87.4%) classes. The FP rates had ranged between 0 to 7.3%.

Table 22. Performance summary (TP rate, FP rate and Precision) of Decision tree (J48) classifier on Dataset version 1

Class	Decision Tree (J48) on dataset-1		
	TP Rate	FP Rate	Precision
Medium Performer	0.866	0.073	0.904
Low Performer	0.892	0.024	0.874
Dropped	1.000	0.002	0.958
High Performer	0.931	0.060	0.892
Failure	0.875	0.000	1.000
Weighted Average	0.898	0.057	0.899

On dataset-2 (Table 23), decision tree (j48) classifier had performed better — with an accuracy of 92.4242% and a Kappa value of 0.8915 — than dataset-1. The results from Table 45 show that the 'Dropped' and 'Failure' classes had continued to exhibit the highest true positive values (100%). The other three classes had also produced high TP rates for decision tree classifier ('Medium Performer' (87.2%), 'Low Performer' (96.9%) and 'High Performer' (93.9%)). The 'Failure' and 'Dropped' classes had continued to exhibit the highest precision value (100%), with the other classes also returning good precision rates ('Medium Performer' (92.4%), 'Low Performer' (86.3%) and 'High Performer' (94.7%)). Both of the 'Dropped' and 'Failure' classes had demonstrated 0% FP rate, meaning, no instances within these classes were falsely classified. The other classes had also exhibited excellent FP rates (less than 5%).

Table 23. Performance summary (TP rate, FP rate and Precision) of Decision tree (J48) classifier on Dataset version 2

Class	Decision Tree (J48) on dataset-2		
	TP Rate	FP Rate	Precision
Medium Performer	0.872	0.044	0.924
Low Performer	0.969	0.038	0.863
Dropped	1.000	0.000	1.000
High Performer	0.939	0.028	0.947
Failure	1.000	0.000	1.000
Weighted Average	0.924	0.034	0.926

On dataset-3 (Table 24), decision tree (j48) classifier had performed better (with an accuracy of 93.2143% and a 0.9024 Kappa value) than datasets 1 and 2. The results from Table 46 show that the 'Dropped' and 'Failure' classes have continued to produce the highest true positive values (100%). The other three classes had also produced high TP rates ('Medium Performer' (92.2%), 'Low Performer' (98.1%) and 'High Performer' (90.1%)). The 'Failure' and 'Dropped' classes had consistently produced the highest precision values (100%), with the other classes also returning good precision rates ('Medium Performer' (89.6%), 'Low Performer' (93%) and 'High Performer' (95.8%)). Both of the 'Dropped' and 'Failure' classes had demonstrated 0% FP rate, while the 'Medium Performer' class had produced the highest FP rate (6.2%).

Table 24. Performance summary (TP rate, FP rate and Precision) of Decision tree (J48) classifier on Dataset version 3

Class	Decision Tree (J48) on dataset-3		
	TP Rate	FP Rate	Precision
Medium Performer	0.922	0.062	0.896
Low Performer	0.981	0.018	0.930
Dropped	1.000	0.000	1.000
High Performer	0.901	0.022	0.958
Failure	1.000	0.000	1.000
Weighted Average	0.932	0.034	0.933

On dataset-4 (Table 25), decision tree (j48) classifier had produced low accuracy (82.4176%, with a 0.7444 Kappa value) than the previous datasets. This dataset had only 4 classes, excluding the 'Failure' class. The results from Table 47 show that the 'Dropped' class had continued to produce the highest true positive value (100%), while the 'High Performer' had produced a low TP rate (70.9%). The 'Dropped' class had produced the highest precision value (100%), while the 'High Performer' have achieved very low precision rate (3.3%). No instances for the 'Dropped' class were false positive.

Table 25. Performance summary (TP rate, FP rate and Precision) of Decision tree (J48) classifier on Dataset version 4

Class	Decision Tree (J48) on dataset-4		
	TP Rate	FP Rate	Precision
Medium Performer	0.805	0.160	0.805
Low Performer	0.810	0.071	0.773
Dropped	1.000	0.000	1.000
High Performer	0.709	0.042	0.033
Weighted Average	0.824	0.097	0.825

Random Forest classifier: The Random Forest classifier with 10 folds cross validation using Weka was run on all four datasets. The dataset-2 had produced the highest accuracy (94.24%, with a Kappa value of 0.9171). The other datasets had performed admirably well (81% -94%).

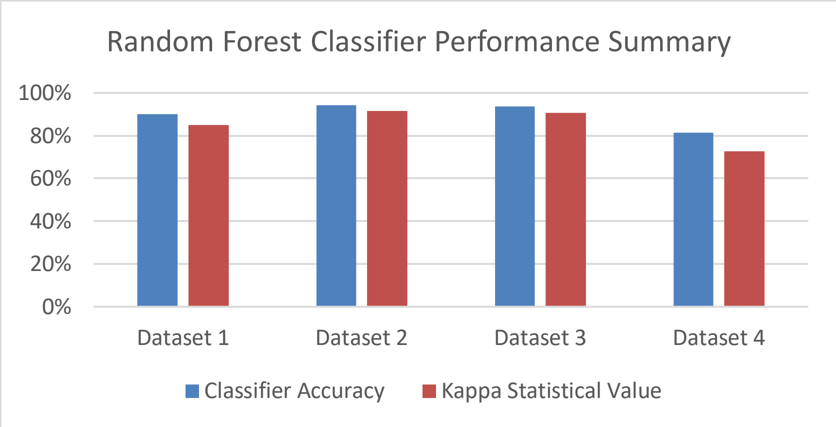

Fig.3. Random Forest classifier performance summary

On dataset-1 (Table 26), the Random Forest classifier had returned the highest accuracy of 90.1528% (with 0.85 Kappa value). Table 48 presents the TP rate, FP rate and precision for Random Forest classifier on dataset-1. The results show that the ‘Dropped’ class had produced the highest true positive value (96%). The other classes had also performed well (‘Medium Performer’ (88.5%), ‘Low Performer’ (89.2%), ‘High Performer’ (92.6%) and ‘Failure’ (75%)). The precision values were also high for all these classes.

Table 26. Performance summary (TP rate, FP rate and Precision) of Random Forest classifier on Dataset version 1

Class	Random Forest on dataset-1		
	TP Rate	FP Rate	Precision
Medium Performer	0.885	0.080	0.899
Low Performer	0.892	0.018	0.902
Dropped	0.957	0.002	0.957
High Performer	0.926	0.054	0.900
Failure	0.750	0.002	0.857
Weighted Average	0.902	0.057	0.901

On dataset-2 (Table 27), Random Forest classifier performed better — with an accuracy of 94.2424% and Kappa value of 0.9171 — than dataset-1. The results from Table 49 show that the ‘Dropped’ class had continued to produce the highest true positive value (100%), with three out of the remaining four classes performing excellent. The ‘Failure’ and ‘Dropped’ classes had demonstrated the highest precision value (100%), with the ‘Medium Performer’, ‘Low Performer’ and ‘High Performer’ classes returning very good precision rates. Both the ‘Dropped’ and ‘Failure’ classes had exhibited 0% FP rate, while the ‘Medium performer’ class returning the highest FP rate (less than 4%).

Table 27. Performance summary (TP rate, FP rate and Precision) of Random Forest classifier on Dataset version 2

Class	Random Forest on dataset-2		
	TP Rate	FP Rate	Precision
Medium Performer	0.912	0.039	0.934
Low Performer	0.954	0.023	0.912
Dropped	1.000	0.000	1.000
High Performer	0.965	0.023	0.957
Failure	0.800	0.000	1.000
Weighted Average	0.942	0.027	0.943

On dataset-3 (Table 28), decision tree (j48) classifier performed close to the datasets 1 and 2 (with an accuracy of 93.5714% and 0.9069 Kappa statistical). The results from Table 50 show that for this dataset, the ‘High Performer’ class had produced the highest true positive value (97%), while the ‘Medium Performer’ (92.2%), ‘Low performer’ (92.6%) and ‘Dropped’ (94.1%) classes had also returned high TP rates. Only the ‘Failure’ class had turned a moderately low TP rate (60%), although achieving the highest precision value (100%). The other classes had also returned very high precision rates. Both the ‘Dropped’ and ‘Failure’ classes had produced low FP rates for this dataset.

Table 28. Performance summary (TP rate, FP rate and Precision) of Random Forest classifier on Dataset version 3

Class	Random Forest on dataset-3		
	TP Rate	FP Rate	Precision
Medium Performer	0.922	0.045	0.922
Low Performer	0.926	0.013	0.943
Dropped	0.941	0.004	0.941
High Performer	0.970	0.034	0.942
Failure	0.600	0.000	1.000
Weighted Average	0.936	0.032	0.936

On dataset-4 (Table 29), Random Forest classifier had produced low accuracy (81.3187%, with 0.7274 Kappa value), compared to the previous datasets. This dataset does not have any ‘Failure’ class. The results from Table 51 show that the ‘Dropped’ class had demonstrated the highest true positive value (100%). The ‘Medium Performer’, ‘Low Performer’ and ‘High Performer’ classes had also produced good TP rates. The ‘Dropped’ class had demonstrated the highest precision value (100%), followed by the ‘High Performer’ class (88.2%). The ‘Medium Performer’ (78.6%) and ’Low Performer’ (72.7%) classes had also performed moderately well in terms of precision rate. The ‘Medium Performer’ class had 18% FP rate, standing higher than all other classes.

Table 29. Performance summary (TP rate, FP rate and Precision) of Random Forest classifier on Dataset version 4

Class	Random Forest on dataset-4		
	TP Rate	FP Rate	Precision
Medium Performer	0.805	0.180	0.786
Low Performer	0.762	0.086	0.727
Dropped	1.000	0.000	1.000
High Performer	0.789	0.028	0.882
Weighted Average	0.813	0.107	0.816

Random Tree: On our four datasets, we ran Random Tree classifier with 10-fold cross validation. It performed the best on dataset-1 out of the four datasets, returning an accuracy of 86.0781% and a Kappa statistical value of 0.7881. The outputs of the other classifiers were within an acceptable range (74 % -86 %). The Random Tree classifier performance summary is shown in Figure 4.

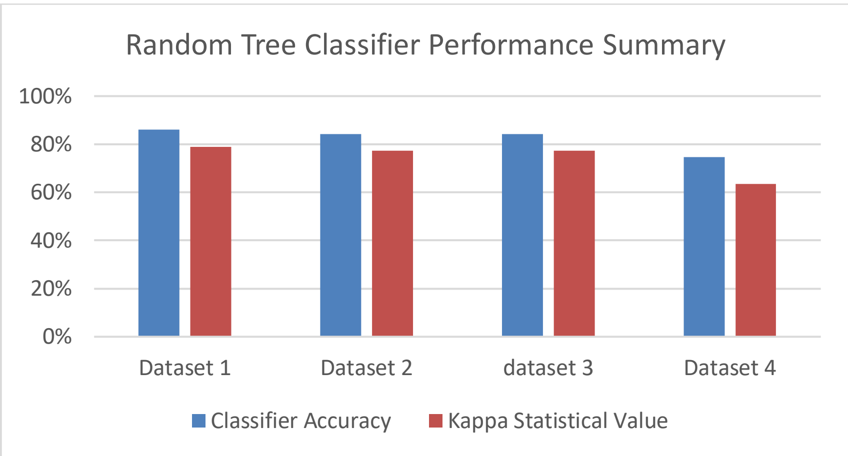

Fig.4. Random Tree classifier performance summary

On dataset-1 (Table 30), Random Tree classifier had produced 86.0781% accuracy with 0.7881 Kappa value. Table 52 presents the TP rate, FP rate and precision of Random forest classifier on dataset-1. The results show that the ‘High Performer’ class had yielded the highest true positive value (nearly 89.7%). The ‘Medium Performer’ (84.4%), ‘Low Performer’ (84.9%) and ‘Dropped’ (82.6%) classes returned good TP rates, while the ‘Failure’ (75%) class had performed reasonably well. The precision was high for all the classes. The ‘Dropped’ class had performed with the highest precision (90.5%), while the 'Failure' class had produced the lowest precision rate (75%). All classes had produced low FP rates, ranging between 0.3 to 11%.

Table 30. Performance summary (TP rate, FP rate and Precision) of Random Tree classifier on Dataset version 1

Class	Random Tree on dataset-1		
	TP Rate	FP Rate	Precision
Medium Performer	0.844	0.110	0.860
Low Performer	0.849	0.032	0.832
Dropped	0.826	0.004	0.905
High Performer	0.897	0.067	0.875
Failure	0.750	0.003	0.750
Weighted Average	0.861	0.077	0.861

On dataset-2 (Table 31), Random Tree classifier had produced less accuracy (84.2424%, with the Kappa value of 0.7736), compared to dataset-1. The results from Table 53 show that the 'Dropped' and 'High Performer' classes had produced the highest true positive values (90.5% and 90.4%, respectively). The 'Medium Performer' (81.6%) and 'Low Performer' (78.5%) classes had performed moderately well, while the 'Failure' class had returned a low TP rate (60%). In terms of precision, the 'High Performer' class had produced the highest precision value (90.4%), while the 'Failure' class yielding a low precision rate (60%). The FP rates of the classes for this dataset are found similar to dataset-1 (0.6% to 10.7%).

Table 31. Performance summary (TP rate, FP rate and Precision) of Random Tree classifier on Dataset version 2

Class	Random Tree on dataset-2		
	TP Rate	FP Rate	Precision
Medium Performer	0.816	0.107	0.823
Low Performer	0.785	0.045	0.810
Dropped	0.905	0.016	0.792
High Performer	0.904	0.051	0.904
Failure	0.600	0.006	0.600
Weighted Average	0.842	0.068	0.843

On dataset-3 (Table 32), Random Tree classifier had performed in a similar way that it did on dataset-2 (84.2857% accuracy, with 0.7736 Kappa value). The results from Table 54 show that for this dataset, the 'High Performer' had exhibited the highest true positive value (91%). While the 'Medium Performer' (79.6%) and 'Low Performer' (87%) had produced high TP rates, the 'Dropped' (70.6%) and 'Failure' (60%) classes yielded moderately low TP rates. In terms of precision, the 'High Performer' class had produced the highest value (86%), while the 'Dropped' and 'Failure' classes produced low precision rates (70.6% and 60%, respectively). The FP rates for all the classes had ranged between 0.7 to 8.4%.

Table 32. Performance summary (TP rate, FP rate and Precision) of Random Tree classifier on Dataset version 3

Class	Random Tree on dataset-3		
	TP Rate	FP Rate	Precision
Medium Performer	0.796	0.079	0.854
Low Performer	0.870	0.035	0.855
Dropped	0.706	0.019	0.706
High Performer	0.911	0.084	0.860
Failure	0.600	0.007	0.600
Weighted Average	0.843	0.067	0.843

On dataset-4 (Table 33), Random Tree classifier had produced 74.7253% accuracy, with a Kappa value of 0.6359. The 'High Performer' class had produced the highest TP rate (89.5%) and a precision rate of 85%. The 'Failure' was absent in this dataset. The 'Medium Performer' (73.2%), 'Low Performer' (66.7%) and 'Dropped' (70%) classes had exhibited moderately well TP rates. In terms of precision, 'Medium Performer' (78.9%) and 'Dropped' (77.8%) classes had produced good precision values, while the 'Low Performer' class had produced the lowest (58.3%). The 'Medium Performer' and 'Low Performer' classes had demonstrated higher FP rates than the other classes.

Table 33. Performance summary (TP rate, FP rate and Precision) of Random Tree classifier on Dataset version 4

Class	Random Tree on dataset-4		
	TP Rate	FP Rate	Precision
Medium Performer	0.732	0.160	0.789
Low Performer	0.667	0.143	0.583
Dropped	0.700	0.025	0.778
High Performer	0.895	0.042	0.850
Weighted Average	0.747	0.116	0.753

Simple Cart: Out of all algorithms, the Simple Cart algorithm had yielded the best accuracy (nearly 94%, with 0.9125 Kappa value) on dataset-3, while yielding a very good accuracy rate (nearly 92%) on dataset-2. It had performed well on the two other datasets as well (86.25% and 82.42%). While we applied 10-fold cross validation technique on all datasets for this algorithm, Figure 5 shows its performance summary.

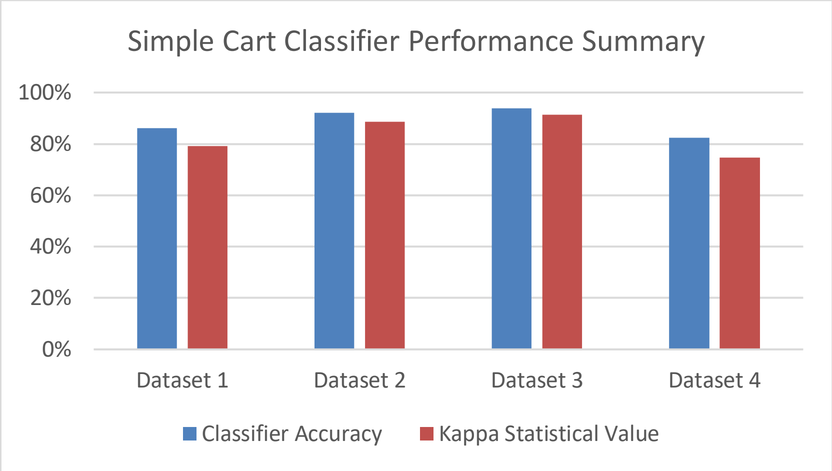

Fig.5. Simple Cart classifier performance summary

On dataset-1 (Table 34), the ‘Dropped’ class had produced the highest achievable TP rate (100%), while the ‘Failure’ class had yielded the most achievable Precision rate (100%). The other classes had also produced high TP rates. The ‘Dropped’ class had demonstrated more than 95% precision rate, while the ‘Failure’ class had demonstrated the lowest FP rate (0%).

Table 34. Performance summary (TP rate, FP rate and Precision) of Simple Cart classifier on Dataset version 1

Class	Simple Cart on dataset-1		
	TP Rate	FP Rate	Precision
Medium Performer	0.828	0.107	0.861
Low Performer	0.892	0.030	0.847
Dropped	1.000	0.002	0.958
High Performer	0.877	0.078	0.856
Failure	0.875	0.000	1.000
Weighted Average	0.862	0.079	0.863

On dataset-2 (Table 35), the ‘Dropped’ class had, instead, produced the highest possible TP and precision rates (100%), while the ‘Failure’ class also producing a 100% precision rate, followed by the ‘High Performer’ class (97.3%). The other classes had also produced high TP rates. Both of the ‘Dropped’ and ‘Failure’ classes had yielded the lowest FP rate (0%).

Table 35. Performance summary (TP rate, FP rate and Precision) of Simple Cart classifier on Dataset version 2

Class	Simple Cart on dataset-2		
	TP Rate	FP Rate	Precision
Medium Performer	0.888	0.059	0.902
Low Performer	0.938	0.042	0.847
Dropped	1.000	0.000	1.000
High Performer	0.939	0.014	0.973
Failure	0.800	0.000	1.000
Weighted Average	0.920	0.035	0.924

On dataset-3 (Table 36), the ‘Low Performer’ class has produced 100% TP rate, followed by the ‘High Performer’ class. In terms of precision rates, the ‘Dropped’ and ‘Failure’ classes had produced a 100% value, similar to its performance on dataset-2. The FP rates for all classes had remained low for this dataset (0% to 3.9%).

Table 36. Performance summary (TP rate, FP rate and Precision) of Simple Cart classifier on Dataset version 3

Class	Simple Cart on dataset-3		
	TP Rate	FP Rate	Precision
Medium Performer	0.883	0.028	0.948
Low Performer	1.000	0.022	0.915
Dropped	0.941	0.000	1.000
High Performer	0.970	0.039	0.933
Failure	0.800	0.000	1.000
Weighted Average	0.939	0.029	0.940

On dataset-4 (Table 37), the 'Dropped' class had produced the highest possible TP and precision rates (100%), similar to its performance on dataset-2. The 'Low Performer' and 'High Performer' classes had produced good TP rates as well, while the 'Medium Performer' and 'High Performer' classes had produced high precision values (82.1% and 84.2%, respectively). The 'Failure' class had yielded a 0% FP rate.

Table 37. Performance summary (TP rate, FP rate and Precision) of Simple Cart classifier on Dataset version 4

Class	Simple Cart on dataset-4		
	TP Rate	FP Rate	Precision
Medium Performer	0.780	0.140	0.821
Low Performer	0.810	0.086	0.739
Dropped	1.000	0.000	1.000
High Performer	0.842	0.042	0.842
Weighted Average	0.824	0.092	0.826

Multilayer Perceptron: Multilayer Perceptron algorithm with 10-fold cross validation had produced a decent, yet similar accuracy across the four datasets (the highest on dataset-3 at 89%). Figure 6 shows the performance summary for Multilayer Perceptron classifier.

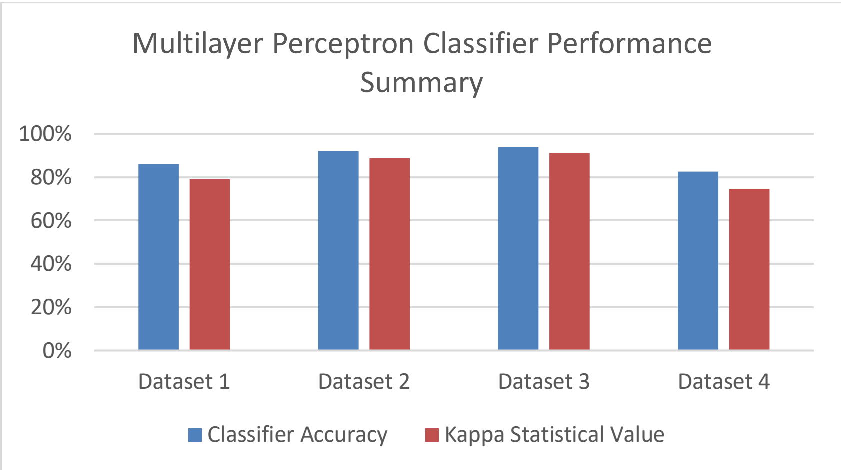

Fig.6. Multilayer Perceptron classifier performance summary

On executing Multilayer Perceptron in Weka on dataset-1 (Table 38), the 'Low Performer' and 'High Performer' classes had produced satisfactory TP rates (92.5% and 91.2%, respectively). Besides, the 'High Performer' class had produced a high precision rate (94.5%), while the 'Failure' class producing the worst TP and precision rates among all the classes. The FP rates had ranged between 1 to 11.3%.

Table 38. Performance summary (TP rate, FP rate and Precision) of Multilayer Perception classifier on Dataset version 1

Class	Multilayer Perceptron on dataset-1		
	TP Rate	FP Rate	Precision
Medium Performer	0.912	0.113	0.866
Low Performer	0.925	0.028	0.860
Dropped	0.739	0.011	0.739
High Performer	0.842	0.026	0.945
Failure	0.375	0.010	0.333
Weighted Average	0.876	0.064	0.880

On dataset-2 (Table 39), the 'High Performer' class had exhibited the highest true positive values (93%). The 'Medium Performer', 'Dropped' and 'High Performer' classes had produced more than 80% TP rates, while the 'Medium Performer', 'Low Performer' and 'High Performer' classes had yielded high precision values (at around 90%). All the classes had yielded excellent FP rates (1 to 6%).

Table 39. Performance summary (TP rate, FP rate and Precision) of Multilayer Perception classifier on Dataset version 2

Class	Multilayer Perceptron on dataset-2		
	TP Rate	FP Rate	Precision
Medium Performer	0.856	0.059	0.899
Low Performer	0.908	0.023	0.908
Dropped	0.810	0.019	0.739
High Performer	0.930	0.060	0.891
Failure	0.000	0.012	0.000
Weighted Average	0.876	0.049	0.874

On dataset-3 (Table 40), the 'High Performer' class had continued to produce the highest true positive value (92%). The 'Medium Performer' (87.4%), 'Low Performer' (87%) and 'Dropped' (88.2%) classes had produced high TP rates, while the 'Dropped' and 'High Performer' classes had produced high precision rates (more than 92%). All the classes had yielded excellent FP rates (0.4 to 7.9%).

Table 40. Performance summary (TP rate, FP rate and Precision) of Multilayer Perception classifier on Dataset version 3

Class	Multilayer Perceptron on dataset-3		
	TP Rate	FP Rate	Precision
Medium Performer	0.874	0.079	0.865
Low Performer	0.870	0.027	0.887
Dropped	0.882	0.004	0.938
High Performer	0.921	0.045	0.921
Failure	0.800	0.007	0.667
Weighted Average	0.889	0.051	0.890

On dataset-4 (Table 41), the 'Dropped' class had demonstrated the highest possible TP and precision rates (100%). The 'Failure' class was absent from this dataset. Out of the remaining classes, the 'High Performer' class had produced the highest TP and precision rates. The 'Dropped' class had produced the lowest FP rate.

Table 41. Performance summary (TP rate, FP rate and Precision) of Multilayer Perception classifier on Dataset version 4

Class	Multilayer Perceptron on dataset-4		
	TP Rate	FP Rate	Precision
Medium Performer	0.829	0.120	0.850
Low Performer	0.810	0.071	0.773
Dropped	1.000	0.000	1.000
High Performer	0.895	0.028	0.895
Weighted Average	0.857	0.076	0.858

5. Conclusion

The main objective of this study was to compare the EDM classifiers for the determination of the best performing algorithm for analyzing and predicting student performance on online academic platforms during the pandemic. We extracted data on seven computing courses from student information systems and Microsoft teams, followed by creating four datasets with a difference in attributes and instances, and then ran these datasets through six EDM techniques that are frequently used for student performance analysis and prediction.

We utilized 10 folds cross-validation technique for all the classifiers. Out of the four datasets, Random Forest Classifier has demonstrated the highest accuracy in the first two datasets (90.1528% and 94.2424%, respectively), while Simple Cart and Naïve Bayes have produced the same for the remainder two datasets, with 93.93% and 92.31% accuracies, respectively. In our datasets, Students were classified into five classes based on their total grade obtained at the end of the semester. All these classifiers have demonstrated medium to high TP rates, class precision and recall, ranging from 60 to 100% for almost all of the classes. Only in a few instances, classes had demonstrated a low class-recall (0-5%), which could result from imbalances existing in the datasets. The performance variations of the algorithms on the datasets could be due to the presence of differences in attributes and instances across the datasets. Overall, the performance of all the classifiers was satisfactory. The result indicates that only academic information and gender attribute can successfully predict students' performance. In our study, we have emphasized the attributes that have a direct effect on students' performance. The findings of this study will help educational institutions, instructors, and students to detect the inadequacies and variables influencing students' performance, as well as serve as an early warning system for anticipating students' failures and poor academic performance.

However, there are a few limitations to this study. First, we couldn't collect complete datasets on the students due to the effects of the pandemic. In future research, a more comprehensive dataset and relevant attributes can be included. Second, the data collection of this research was limited to a few courses of one department in a university. In further research, more courses and a diverse list of departments can be included. Lastly, we have analyzed the data of only one semester of academic activities. Further studies are recommended to incorporate longer periods of data for a more comprehensive student performance analysis and prediction on online academic setting.

References

[1] Saa, A. A., Al-Emran, M., & Shaalan, K. (2020). Mining Student Information System Records to Predict Students' Academic Performance. In *Advances in Intelligent Systems and Computing* (Vol. 921). Springer International Publishing. https://doi.org/10.1007/978-3-030-14118-9_23.

[2] De Morais, A. M., Araújo, J. M. F. R., & Costa, E. B. (2015). Monitoring student performance using data clustering and predictive modelling. *Proceedings - Frontiers in Education Conference, FIE, 2015-February*(February). https://doi.org/10.1109/FIE.2014.7044401.

[3] Asif, R., Merceron, A., Ali, S. A., & Haider, N. G. (2017). Analyzing undergraduate students' performance using educational data mining. *Computers and Education*, *113*, 177–194. https://doi.org/10.1016/j.compedu.2017.05.007

[4] Ramaswami, M., & Bhaskaran, R. (2010). *A CHAID Based Performance Prediction Model in Educational Data Mining*. *7*(1), 10–18. http://arxiv.org/abs/1002.1144.

[5] Devasia, T., Vinushree, T. P., & Hegde, V. (2016). Prediction of students performance using Educational Data Mining. *Proceedings of 2016 International Conference on Data Mining and Advanced Computing, SAPIENCE 2016*, 91–95. https://doi.org/10.1109/SAPIENCE.2016.7684167

[6] Shahiri, A. M., Husain, W., & Rashid, N. A. (2015). A Review on Predicting Student's Performance Using Data Mining Techniques. *Procedia Computer Science*, *72*, 414–422. https://doi.org/10.1016/j.procs.2015.12.157

[7] Suljic, M. (2012). Data Mining Approach for Predicting Student Performance. *Economic Review : Journal of Economics and Business*, *X*(1), 3–12.

[8] Pisani, S., Fioriti, D., Conte, M. P., Chiarini, F., Seganti, L., & Degener, A. M. (2002). Involvement of herpes simplex virus type 2 in modulation of gene expression of human papillomavirus type 18. *International Journal of Immunopathology and Pharmacology*, *15*(1), 59–63. https://doi.org/10.1177/039463200201500108

[9] Kabakchieva, D. (2013). Predicting student performance by using data mining methods for classification. *Cybernetics and Information Technologies*, *13*(1), 61–72. https://doi.org/10.2478/cait-2013-0006

[10] Guo, B., Zhang, R., Xu, G., Shi, C., & Yang, L. (2016). Predicting Students Performance in Educational Data Mining. *Proceedings - 2015 International Symposium on Educational Technology, ISET 2015*, 125–128. https://doi.org/10.1109/ISET.2015.33

[11] Nandi, D., Hamilton, M., & Harland, J. (2015). What Factors Impact Student – Content Interaction in Fully Online Courses. *International Journal of Modern Education and Computer Science*, *7*(7), 28–35.

[12] Hasan, R., Palaniappan, S., Raziff, A. R. A., Mahmood, S., & Sarker, K. U. (2018). Student Academic Performance Prediction by using Decision Tree Algorithm. *2018 4th International Conference on Computer and Information Sciences: Revolutionising Digital Landscape for Sustainable Smart Society, ICCOINS 2018 - Proceedings*, 1–5. https://doi.org/10.1109/ICCOINS.2018.8510600

[13] Abu, A. (2016). Educational Data Mining & Students' Performance Prediction. *International Journal of Advanced Computer Science and Applications*, *7*(5), 212–220. https://doi.org/10.14569/ijacsa.2016.070531

[14] Nurhuda, A., Rosita, D., Series, I. O. P. C., Science, M., Amra, I. A. A., Maghari, A. Y. A., Yulianto, L. D., Triayudi, A., Sholihati, I. D., Riyanto, V., Hamid, A., Ridwansyah, R., Ahmad, I., Sulistiyani, H., Saputra, H., Informasi, S., Teknik, F., Ilmu, D., Indonesia, U. T., … Handhayani, T. (2018). Prediksi Kelulusan Mahasiswa Tepat Waktu Menggunakan Metode Desicion Tree Dan Artificial Neural Network. *Proceeding - 2017 3rd International Conference on Science in Information Technology: Theory and Application of IT for Education, Industry and Society in Big Data Era, ICSITech 2017*, *4*(1), 21–30.

[15] Rodríguez, E., Arqués, J. L., Rodríguez, R., Nuñez, M., Medina, M., Talarico, T. L., Casas, I. A., Chung, T. C., Dobrogosz, W. J., Axelsson, L., Lindgren, S. E., Dobrogosz, W. J., Kerkeni, L., Ruano, P., Delgado, L. L., Picco, S., Villegas, L., Tonelli, F., Merlo, M., … Masuelli, M. (1989). We are IntechOpen , the world ' s leading publisher of Open Access books Built by scientists , for scientists TOP 1 %. *Intech*, *32*(tourism), 137–144. https://www.intechopen.com/books/advanced-biometric-technologies/liveness-detection-in-biometrics

[16] Salal, Y. K., Abdullaev, S. M., & Kumar, M. (2019). Educational data mining: Student performance prediction in academic. *International Journal of Engineering and Advanced Technology*, *8*(4C), 54–59.

[17] Kabakchieva, D. (2012). Student performance prediction by using data mining classification algorithms. *International Journal of Computer Science and Management Research*, *1*(4), 686–690.

[18] Bhargava, N., Dayma, S., Kumar, A., & Singh, P. (2017). An approach for classification using simple CART algorithm in WEKA. *Proceedings of 2017 11th International Conference on Intelligent Systems and Control, ISCO 2017*, 212–216. https://doi.org/10.1109/ISCO.2017.7855983

[19] Yassein, N. A., M Helali, R. G., & Mohomad, S. B. (2017). Predicting Student Academic Performance in KSA using Data Mining Techniques. *Journal of Information Technology & Software Engineering*, *07*(05). https://doi.org/10.4172/2165-7866.1000213

[20] Patil, R., & Tamane, S. (2018). A comparative analysis on the evaluation of classification algorithms in the prediction of diabetes. *International Journal of Electrical and Computer Engineering*, *8*(5), 3966–3975. https://doi.org/10.11591/ijece.v8i5.pp3966-3975
[21] Hooshyar, D., Pedaste, M., & Yang, Y. (2020). Mining educational data to predict students' performance through procrastination behavior. *Entropy*, *22*(1), 12. https://doi.org/10.3390/e22010012
[22] Teräs, M., Suoranta, J., Teräs, H., & Curcher, M. (2020). Post-Covid-19 Education and Education Technology 'Solutionism': a Seller's Market. *Postdigital Science and Education*, *2*(3), 863–878. https://doi.org/10.1007/s42438-020-00164-x
[23] Li,Cathy ; Lalani, Farah. "The COVID-19 pandemic has changed education forever. This is how", WORLD ECONOMIC FORUM, 29 Apr 2020
[24] Abdalla, Jihan. "US universities switch to online courses due to coronavirus", AL JAZEERA, 10 Mar 2020
[25] Coronavirus disease (COVID-19) advice for the public 2021, viewed 11 April 2021.
[26] Shahiri, A. M., Husain, W., & Rashid, N. A. (2015). A Review on Predicting Student's Performance Using Data Mining Techniques. *Procedia Computer Science*, *72*, 414–422. https://doi.org/10.1016/j.procs.2015.12.157
[27] Mussa S. Abubakaria, Fatchul Arifin, Gilbert G. Hungilo. "Predicting Students' Academic Performance in Educational Data Mining Based on Deep Learning Using TensorFlow", International Journal of Education and Management Engineering, Vol.10, No.6, pp.27-33, 2020.
[28] Mohammed Abdullah Al-Hagery, Maryam Abdullah Alzaid, Tahani Soud Alharbi, Moody Abdulrahman Alhanaya, "Data Mining Methods for Detecting the Most Significant Factors Affecting Students' Performance", International Journal of Information Technology and Computer Science, Vol.12, No.5, pp.1-13, 2020.
[29] V. Vijayalakshmi, K. Venkatachalapathy, "Comparison of Predicting Student's Performance using Machine Learning Algorithms", International Journal of Intelligent Systems and Applications, Vol.11, No.12, pp.34-45, 2019.
[30] Siddu P. Algur, Prashant Bhat, Narasimha H Ayachit,"Educational Data Mining: RT and RF Classification Models for Higher Education Professional Courses", International Journal of Information Engineering and Electronic Business, Vol.8, No.2, pp.59-65, 2016.

Authors' Profiles

Musaddiq Al Karim received his Bachelor of Science degree in Computer Science and Engineering from American International University-Bangladesh. He worked as an Undergraduate Teaching Assistant at American International University-Bangladesh and Software Quality Assurance Engineer intern at Olivine Limited, Dhaka, Bangladesh. His research interest includes Artificial Intelligence, Data mining, Machine learning, Algorithms, IoT and Human-computer Interaction, etc. He can be contacted at musaddiqmk19@gmail.com.

Md. Mahadi Masnad has received his BSc in Computer Science and engineering from American International University-Bangladesh. He also completed his internship as a Teaching Assistant at American International University-Bangladesh. His areas of interest in research include artificial intelligence, machine learning, cloud computing, human computer interaction and also in different algorithms. He can be contacted at atime691@gmail.com.

Mst. Yeasmin Ara has completed her Bachelor of Science (BSc) in Computer Science & Engineering (CSE) at American International University -Bangladesh (AIUB). She has completed her internship as a Business Analyst at Akij Group department of Information Technology. And currently she is planning for pursuing MSc degree. She is highly interested in IoT, Data Mining as research area and Software Quality and Assurance. She can be contacted at yeasminararenu007@gmail.com.

Mostafa Rasel completed his BSc in Computer Science & Engineering (CSE) at American International University -Bangladesh (AIUB). He has held an internship at Banglalink and another at Global Islami Bank Ltd. He is highly interested in IoT and networking as a research area. Currently, he is planning for his MSc degree. He can be contacted at mostafarasel9@gmail.com.

Dr. Dip Nandi is currently working as a Professor and Director of Faculty of Science and Technology in American International University- Bangladesh. His research interest includes Software Engineering, Management Information Systems, E-learning etc. He can be contacted at dip.nandi@aiub.edu.

How to cite this paper: Musaddiq Al Karim, Md. Mahadi Masnad, Mst. Yeasmin Ara, Mostafa Rasel, Dip Nandi, "A Comprehensive Study to Investigate Student Performance in Online Education during Covid-19", International Journal of Modern Education and Computer Science(IJMECS), Vol.14, No.3, pp. 1-25, 2022.DOI: 10.5815/ijmecs.2022.03.01