

# Supervised Machine Learning Model-Based Approach for Performance Prediction of Students

<sup>1</sup>Abdul Razaque and <sup>2</sup>Abrar M. Alajlan

<sup>1</sup>Department of Computer Engineering and Information Security, International IT University, Almaty Kazakhstan

<sup>2</sup>Self-Development Skills Department, Common First Year Deanship, King Saud University, KSA

## Article history

Received: 07-06-2020

Revised: 07-08-2020

Accepted: 25-08-2020

## Corresponding Authors:

Abrar M. Alajlan

Self-Development Skills

Department,

Common First Year Deanship,

King Saud University, KSA

Email: aalajlan1@ksu.edu.sa

**Abstract:** Predicting students' performance is one of the crucial issue for learning contexts, since it helps to develop alternative recommendation systems for academically weak students. Thus, several methods and practices have been applied for educational improvement. However, most of the existing methods do not determine the performance of the students. In this study, we have studied the execution of six machine learning models (Decision tree, Random Forest, Support Vector Machine, Logistic Regression, Ada Boost, Stochastic Gradient Descent) to analyze and evaluate the students' achievements. The performance is evaluated in term of accuracy, precision, sensitivity and f-measure. Among the selected models, the results validate that the efficiency of Stochastic Gradient Descent is better in training small dataset. In addition, it also produces the higher accuracy as compared with other models. This contribution aims to develop the best model which may derive the conclusion on students' academic achievement.

**Keywords:** Machine Learning, Student Performance, Decision Tree, Random Forest, Support Vector Machine, Logistic Regression, Stochastic Gradient Descent

## Introduction

There has been a revival of interest in recent years for the universities to adopt and innovate in global educational standards and requirements, where they faced stiff competition to attract talented students and to retain them (Razaque *et al.*, 2019) Thus, making timely decisions for the transformation in the field of education and for modulating the educational systems are highly demanded (Varvarigos, 2020).

The current evaluation system of student's knowledge is barely showing the preparation degree, especially for those who can potentially fail the final examination. Due to the large flow of students and time constraints; the professors are not always able to indicate precisely a "risky" student. In addition to this, universities are increasingly introducing e-learning systems (Barteit *et al.*, 2020). Creating an automated student performance prediction method will allow teachers to track down weak students and thereby help them complete the current training course (Ivanović *et al.*, 2018)

The prediction of student performance is a fundamental task that is being researched by using Educational Data Mining (EDM) (Injadat *et al.*, 2020).

This process predicts the value of an unidentified variable that describes the students' outcomes (Pass/Fail), grades, marks, etc. The study in the field of EDM was presented in (Peña-Ayala, 2014). Later, a more detailed theoretical study of the EDM was conducted in (Angeli *et al.*, 2017).

The educational data mining technique was introduced to improve graduate students' performance in research (Ashraf *et al.*, 2020). The purpose of their work was to solve the problem of students' low grades (Sati, 2018). The problem of predicting student performance was also addressed in (Amrieh *et al.*, 2016). The results revealed a relationship between students' behavior and academic performance. Tomasevic *et al.* (2020) considered the use of the decision tree classifier as the main method for EDM research.

Asif *et al.* (2017) aimed to justify the capabilities of data mining techniques in the context of higher education. The classification methods of data mining were applied. Yahya and Osman (2019) analyzed data mining algorithms (a rule learner, the nearest neighbor classifier and neural network) with application in graduate students' performance. An empirical analysis of data mining algorithms for educational purposes was discussed in (Zhang *et al.*, 2018).

In this study, we have studied the execution of six machine learning algorithms: (Decision tree, Random Forest, Support Vector Machine, Logistic Regression, Ada Boost, Stochastic Gradient Descent). Evaluation measures such as sensitivity, precision, accuracy and f-measure have been used as performance evaluation metrics. The purpose of the prediction is to estimate the unknown value - the final student rating, where a model for each algorithm is formed and the data trained into one dataset to identify the most effective algorithm. The key points of this study are as follows:

- Classification of students' performance based on several attributes
- Preprocessing techniques to convert raw data into a suitable form which can be used by mining algorithms
- Comparative analysis of the performance of learning models based on sensitivity, precision, accuracy and f-measure

The remainder of the paper is organized as follows: Section 2 discusses the problem and significance. Section 3 presents the related work. Section 4 demonstrates the materials and background of the supervised models used in this study. Section 5 presents the proposed methodology. Section 6. Discusses the results. Section 7 is the conclusion and future work.

## Problem Identification and Significance

The main asset of higher educational institutions are students. Thus, the performance of students reflects the quality of teaching and, thus, affects the ranking of the university. Universities should deeply analyze their activities in the educational market to identify their uniqueness and build on their further development. In this regard, the university management should focus more on educational performance of learners. Also, they should analyze the characteristics of students at the admission stage to select the most promising ones. Moreover, academic performance plays an important role in the career development of students after their graduation, since this is one of the factors that employers pay attention to the labor market.

However, the current knowledge control system in the form of tests regulates the students' performance post factum, not allowing to identify potentially weak students during the whole educational process. As a result, students of educational institutions lose attractiveness for potential employers and universities go down in the ranking.

## Related Work

In this section, the salient features of existing approaches are extensively discussed. Even though the use of mining techniques in predicting the academic

performance of students is a relatively young field, many methods have been introduced on this subject. The reason for this is the huge potential for educational institutions to improve their performance.

There are various aspects that may affect the performance of students during their studies. These factors could be social, demographic and academic interests. Academically speaking, grades in previous classes are of great importance because they help to measure students' abilities and play a very significant role in predicting their final grades (Roy and Garg, 2017).

The authors (Hussain *et al.*, 2019) performed a study that used machine learning techniques to predict the difficulties that students will encounter in a subsequent digital design course session. This study analyzed the data logged by a technology- enhanced learning system called Electronics Education and Design Suite (DEEDS) using machine learning algorithms, where students are allowed to solve digital design exercises with different levels of difficulty while logging input data. The researchers observed that ANNs and SVMs can integrate into the TEL system easily and perform higher accuracy than do other algorithms.

In research, (Kumar and Singh, 2019) combined K-Means clustering algorithm with Artificial Neural Network and Support Vector Machine classification algorithm to assess student performance and reduce human effort. The dataset used in the study was taken from ED Facts about the results of state assessment in reading/language, art and mathematics of session 2010-11. Using Bayesian classification method on 34 different attributes, they determined that some school attributes and some subject marks related attributes were highly correlated with the student academic performance.

Hlosta *et al.* (2017) introduced the self-learning system using the machine learning algorithm to find students at risk in a new course without any previous history data. Their approach uses information about already submitted assessments that presents the problem of unbalanced data for training and testing models. The study validated that XGBoost achieved the best performance.

The research carried out by (Czibula *et al.*, 2019) proposed a new classification model Students Performance prediction using Relational Association Rules (S PRAR) for predicting the students' results based on the different academic discipline applying the relational association rules. RARs extend the classic correlation rules to express the different relationships between data attributes.

Tran *et al.* (2017) predict students' performance in academic environments by analyzing and comparing a regression-based model and recommendation-based model. In the regression-based approach, several skills

related to the course have been added with the aim of enhancing the overall predictive performance.

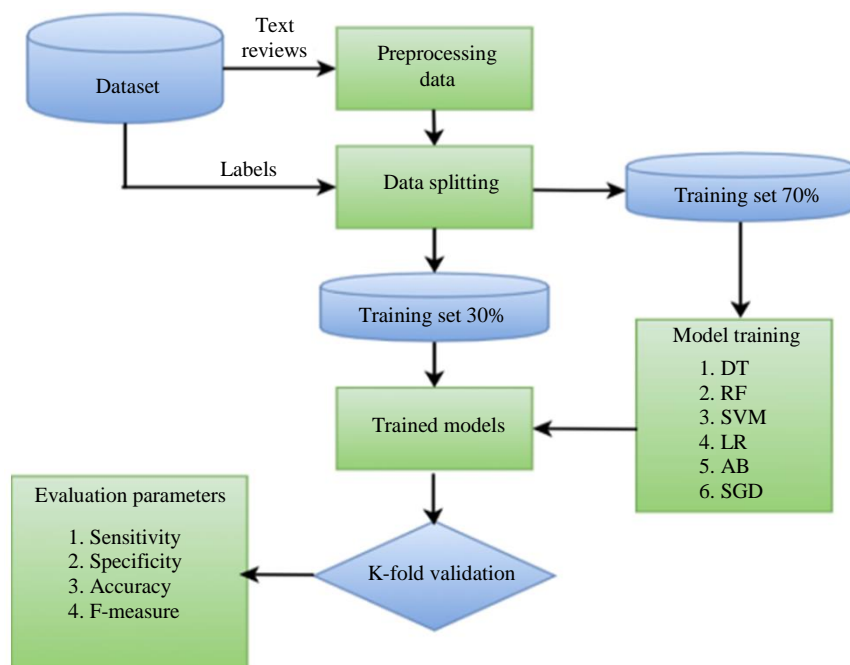
Ahmad *et al.* (2020) highlighted the importance of Educational Data Mining. The study conducted to find out the performance-based prediction of students by investigating their demographic and academic factors using machine learning methods. The research was recommended to predict the academic performance of the students by taking other circumstances of students into account such as personality, cognitive, psychological and economic domain.

As all the existing approaches focused either on determining the outliers or improving the educational performance. Our study particularly focuses on accuracy, sensitivity, precision and f-measure using state-of-the-art algorithms. Based on the results, the

best algorithm is recommended for improving the performance of educational institutions.

## Materials and Background

This work uses several machine learning algorithms to solve classification problems. Figure 1 illustrates the steps for systems architecture. First, the data goes through preprocessing phase to convert raw data into a suitable form which can be used by mining algorithms. When the review texts are cleaned, we split the datasets for training and testing in ratio of 70 to 30%. The training set is partitioned into smaller splits and the model is trained using k-folds cross validation, while the testing set is used to validate the trained model and measure the overall performance.



**Fig. 1:** System architecture conducting the experiments

**Table 1:** Student’s attributes and their description

Attribute	Description	Data Type
Gender	student's sex	Binary: 'F' - female or 'M' - male
Age	student's age	Numeric: 18 to 22
Reason	Reason to choose this school	Nominal: close to 'home', school 'reputation', 'course' preference or 'other'
Famsize	family size	Binary: 'LE3' - less or equal to 3 or 'GT3' – greater than 3
Studytime	weekly study time	Numeric: 1 - <2 h, 2-2 to 5 h, 3-5 to 10 h, or 4 - >10 h
Failures	number of past class failures	Numeric: n if 1 ≤ n < 3, else 4
Internet	Internet access at home	Binary: yes, or no
Freetime	free time after school	Numeric: from 1 - very low to 5 - very high
Health	current health status	Numeric: from 1 - very bad to 5 - very good
Traveltime	home to school travel time	Numeric: 1 - <15 min., 2-15 to 30 min., 3-30 min. to 1 h, or 4 - >1 h
G1	First period grade	Numeric: 0 to 20
G2	Second period grade	Numeric: 0 to 20
G3	Final grade	Numeric: 0 to 20

### A. Data Preparation

The dataset used in this study was taken from the University of California Irvine (UCI) machine learning repository. The name of the dataset is “Student Performance Data Set”. Data was collected based on student performance from Portugal which is built from the data of 395 students (Cortez (n.d.)).

### B. Data Selection and Transformation

The data were collected on grounds of students and surveys database. Predictors used to form the proposed model which include: Estimates, demographic data, social status and educational related features is demonstrated in Table 1.

### C. Supervised Machine Learning Methods

In this section, we discuss the machine learning algorithms; we describe the mechanism and hyper-parameters of each. All the supervised machine learning algorithms mentioned below were installed in Python using the scikit-learn module. This work used six different machine-learning algorithms to solve the classification problem.

#### Decision tree

Decision tree method is a way of representing rules in a hierarchical, sequential structure, where each object corresponds to a single node that provides a solution as illustrated in algorithm 1. The field of application of decision trees is currently wide, but all tasks solved by this method can be combined into the following three classes:

- **Data description.** Decision trees allow to store information about data in a compact form
- **Classification.** Decision trees do an excellent job with classification tasks, i.e., assigning objects to one of the previously known classes. The target variable must have discrete values
- **Regression.** If the dependent (target) variable has continuous values, decision trees allow to set the dependence of the dependent variable on independent (input) variables. For example, the tasks of numerical prediction (prediction of values of the target variable) belong to this class

---

#### Algorithm 1. Decision tree algorithm

---

**Input:**  $TR$ : Training dataset,  $D$ : Default,  $Attr$ : Set of attributes.

**Output:** Decision tree.

```

begin
    if  $TR$  is empty then return  $D$ 
    else if all  $TR$  have same classification then return
    the classification
    else if  $Attr$  is empty then return  $MODE (TR)$ 
    
```

---



---

```

else
    
```

```

    Let  $Tg = CHOOSE-ATTRIBUTE (Attr, TR)$ 
    
```

```

    Let  $NT =$  new decision tree with root test ( $Tg$ )
    
```

```

    for each value  $vi$  of  $Tg$  do
    
```

```

         $TRi =$  elements of  $TR$  with  $Tg = vi$ 
    
```

```

         $SubT =$  call  $DTL (TR, Attr-Tg, MODE(TR))$ 
    
```

```

        add a branch to  $NT$  with label  $vi$  and subtree  $SubT$ 
    
```

```

    return  $NT$ 
    
```

---

#### Random Forest

Random Forest is a composition (ensemble) of many decision trees, which helps to reduce the problem of retraining and improve accuracy in comparison with a single tree. The forecast is obtained by aggregating the responses of multiple trees. Training trees occurs independently of each other (on different subsets), which not only solves the problem of building identical trees on the same dataset, but also makes it very convenient for use in distributed computing systems. Random forest algorithm is illustrated below.

---

#### Algorithm 2. Random Forest algorithm

---

**Input:**  $T$ : Training dataset,  $N$ : node.

**Output:** Random forest.

```

begin
    
```

```

    for  $t = 1$  to  $T$  do
    
```

```

        Randomly sample  $T_i$  with replacement from  $T$ ;
    
```

```

        Create  $N_i$ , containing  $T_i$ 
    
```

```

        Call  $BuildTree (N_i)$ 
    
```

```

    end for
    
```

```

BuildTree (N):
    
```

```

    if  $N$  contain instances of only one class then
    
```

```

        return
    
```

```

    else
    
```

```

        Randomly select  $d\%$  of the possible splitting
        features in  $N$ 
    
```

```

        Select the feature  $F$  with highest information
        gain to split on
    
```

```

        Create  $f$  child nodes of  $N, N_1, \dots, N_f$ , where  $F$ 
        has  $f$  possible values ( $F_1, \dots, F_f$ )
    
```

```

        for  $t = 1$  to  $f$  do
    
```

```

            Set the contents of  $N_t$  to  $T_t$ , where  $T_t$  is all
            instances in  $N$  that match  $F_t$ 
    
```

```

            Call  $BuildTree (N_t)$ 
    
```

```

        end for
    
```

```

    end if
    
```

---

#### Support Vector Machine

This method initially refers to binary classifiers, although there are ways to make it works for multi classification tasks. The fundamental function of the algorithm is to find the most correct line, or hyperplane, which divides the data into two classes as illustrated in algorithm 3.

---

**Algorithm 3.** Support Vector Machine

---

1. Create H, where  $H_{ij} = y_i y_j x_i x_j$
  2. Choose how significantly misclassifications should be treated, by selecting a suitable value for the parameter C.
  3. Find  $\alpha$  so that  $\sum_{i=1}^L \alpha_i - \frac{1}{2} \alpha^T H \alpha$  is maximized, subject constraints  $0 \leq \alpha_i \leq C \forall_i$  and  $\sum_{i=1}^L \alpha_i y_i = 0$
- This is done using a QP solver
4. Calculate  $w = \sum_{i=1}^L \alpha_i y_i x_i$
  5. Determine the set of Support Vectors  $S$  by finding the indices such that  $0 \leq \alpha_i \leq C$
  6. Calculate  $b = \frac{1}{N_s} \sum_{s \in S} (y_s - \sum_{m \in S} \alpha_m y_m x_m x_s)$
  7. Each new point  $x'$  is classified by evaluating  $y' = \text{sgn}(w * x' + b)$
- 

**Logistic Regression**

Logistic regression is a form of multiple regression, the general purpose of which is to analyze the relationship between several independent variables (also called regressors or predictors) and the dependent variable. Binary logistic regression, as the name implies, applies when the dependent variable is binary (that is, it can take on only two values). In other words, with the help of logistic regression, it is possible to estimate the probability that an event will occur for a particular test subject (sick/healthy, loan repayment/default, etc.). To solve the problem, the regression problem can be formulated differently: Instead of predicting a binary variable, we predict a continuous variable with values on the interval [0,1] for any values of independent variables. This is achieved by applying the following regression equation (logit transform):

$$p = \frac{1}{1 + e^{-y}} \tag{1}$$

where,  $P$  is the probability that an event of interest will occur;  $e$  - base of natural logarithms 2.71 ...;  $y$  is the standard regression equation.

---

**Algorithm 4.** Logistic Regression

---

- Assume that the probability of occurrence of the event  $y = 1$  is equal to  $\Pr\{y = 1|x\} = \frac{1}{1 + e^{-y}}$ . The probability of the even  $y = 0$  is equal to  $\Pr\{y = 0|x\} = 1 - \frac{1}{1 + e^{-y}}$
2. Selection of parameters
  3. Regularization
  4. Simplified cost function
  5. Calculating Gradients
- 

**AdaBoost**

AdaBoost (short for Adaptive Boosting) is a machine learning algorithm proposed by Yoav Freund and Robert Schapier. This algorithm can be used in combination with several classification algorithms to improve their efficiency. The algorithm strengthens the classifiers, combining them into a “committee”. AdaBoost is adaptive in the sense that each next classifier committee is built on objects that have been incorrectly classified by previous committees. AdaBoost is sensitive to data noise and outliers. However, it is less prone to retraining compared to other machine learning algorithms. AdaBoost algorithm is shown in algorithm 5.

---

**Algorithm 5.** AdaBoost algorithm

---

- Input:** Training set  $S = ((x_1, y_1), \dots, (x_n, y_n))$   
**Initialization:**  $T =$  maximum number of iterations;  
 Initialize the weight distribution  $\forall_i \in \{1, \dots, n\}, D^{(1)}(i) = 1/n$
- for**  $t = 1, \dots, T$  **do**
- Find the classifier  $h_t: X \rightarrow \{-1, +1\}$  which minimizes the weighted classification error:  $h_t = \arg \min_{h_t \in \mathcal{H}} \varepsilon_t$
- $$\varepsilon_t = \sum_{i=1}^m D_t(i) [y_i \neq h_t(x_i)].$$
- if** the value  $\varepsilon_t > 0.5$  **then stop**.
- Select  $\alpha_t \in R$ , usually  $\alpha_t = \frac{1}{2} \ln \frac{1 - \varepsilon_t}{\varepsilon_t}$  where  $\varepsilon_t$  weighted classifier error  $h_t$ .
- Update:  $D_{t+1}(i) = \frac{D_t(i) e^{-\alpha_t y_i h_t(x_i)}}{z_t}$  where  $Z_t$  is the normalizing parameter
- Build the resulting classifier:  $H(x) = \text{sign}(\sum_{t=1}^T \alpha_t h_t(x))$
- end for**
- 

**Stochastic Gradient Descent**

Stochastic gradient descent is a very efficient model for linear classifiers and regressors. It is simply substitute the actual gradient that calculated from the whole dataset with an approximation of thereof which calculated from a randomly selected subset. With a stochastic (or “operational”) gradient descent, the value of the gradient is approximated by a gradient of the cost function calculated on only one learning element. Then the parameters are changed in proportion to the approximate gradient. Thus, the model parameters changed after each learning object. For large data sets, stochastic gradient descent can provide a significant advantage in speed compared to standard gradient descent. The algorithm of SGD is presented in algorithm 6.

---

**Algorithm 6.** Stochastic Gradient Descent (SGD)

---

**Input:** T: Training data,  $\lambda$ : Regularizing parameters,  $\eta$ : Learning rate,  $\sigma$ : Initialization

**Output:** Model parameters  $\Theta = (w_0, w, V)$  where  $w_0 = 0$ ;  $w = (0, \dots, 0)$ ;  $v \sim N(0, \sigma)$

**repeat**

**for**  $(x, y) \in T$  **do**

$$w_0 = w_0 - \eta \frac{\alpha}{aw_0} l(\hat{y}(x|\Theta), y) + 2\lambda^0 w_0;$$

**for**  $i \in \{1, \dots, p\} \wedge x_i \neq 0$  **do**

$$w_i = w_i - \eta \frac{\alpha}{aw_i} l(\hat{y}(x|\Theta), y) + 2\lambda_{\pi(i)}^w w_i;$$

**for**  $i \in \{1, \dots, k\}$  **do**

$$v_{i,f} = v_{i,f} - \eta \frac{\alpha}{av_{i,f}} l(\hat{y}(x|\Theta), y) + 2\lambda_{f,\pi(i)}^v w_i;$$

**end for**

**end for**

**end for**

**until** *stopping criterion is met*;

---

## Proposed Methodology

This section presents our dataset that are implemented in this study, the preprocessing of our data and several classification models leveraged for the prediction students' academic performance.

### A. Data Collection

It is very important to collect data properly, since the quality and quantity of data are affecting how good our model is and how accurate the desired result will be. Irrelevant data can reduce accuracy and therefore need to be cleared. Non-relevant data includes missing values, extraneous data and other data that are out of the common set and here there are two options: Either this data is removed from the sample, or replaced with dummy values. For example, missing values can be replaced with zeros. For the algorithm, obtaining approximate values is more proper than the missing ones and so the prioritized factors should be taken into consideration for analyzing dataset: The accuracy of the forecast or the time spent on preparing the data. The data in this study have predictors such as student scores, demographic and social data where the data was collected by questioning.

Students were classified into three categories as "good", "fair" and "poor" based on their results. Also, for the analysis of what influences student performance, such factors were added as whether the student in a relationship, use of alcohol, educational level of their parents, etc.

### B. Data Reparation

After collecting data, the rows sequence is changed randomly which might affect the learning process. In

addition, the data was divided into two parts: One for training and the other for evaluating the model, where the optimal ratio is 70/30.

### C. Modeling Method

There are some parameters considered when selecting our model as follows:

- **Accuracy:** It is not always necessary to get results with maximum accuracy. In some cases, it is more proper to use approximate values, which can significantly reduce the processing time
- **Studying time:** The amount of time required to train a model depends on the algorithm we choose. As a rule, the lower the accuracy, the shorter the training time. Also, some algorithms are more dependent on the amount of data than others and this can significantly affect the choice of the algorithm for the model, especially when time is limited
- **Linearity:** Most of the algorithms in machine learning are linear, that is, data classes can be separated by a straight line. For some problems, linear algorithms work fine, but in some cases they can significantly reduce the accuracy
- **Number of parameters:** The total number of parameters is directly affect the algorithm behavior such as the number of iterations or sensitivity to errors. Usually, the greater the number of parameters, the greater the number of attempts and errors on the way to find the best combination. On the other hand, a large number of parameters would result in a greater flexibility and higher accuracy

Based on the characteristics listed above, six different models were assessed, which respectively are DT, RF, SVM, AB, LR and SGD. The goal of this study is to choose the model with the highest prediction performance as the primary model.

### D. Evaluation

While training a model is a key step, how the model generalizes on unseen data is an equally important aspect that should be considered in every machine learning pipeline. Model evaluation aims to estimate the generalization performance of a model on future (unseen/out-of-sample) data. Indeed, it's important to use new data when evaluating the model to prevent the likelihood of overfitting to the training set.

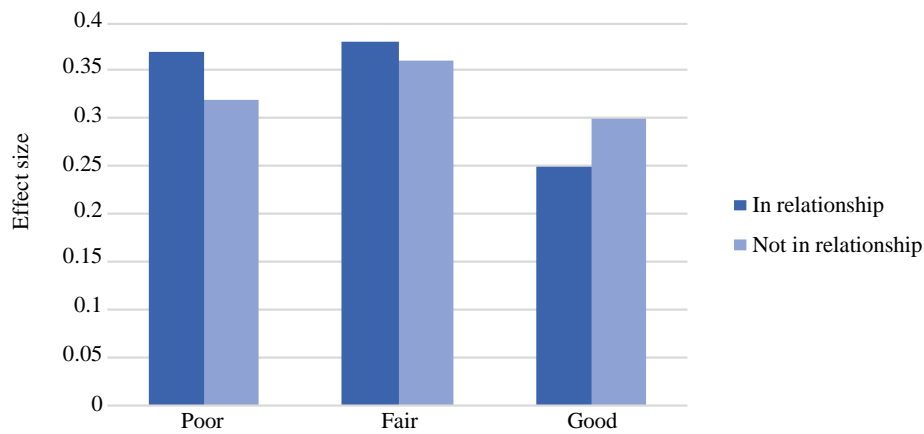
## Results and Discussion

### A. Research Model and Hypothesis

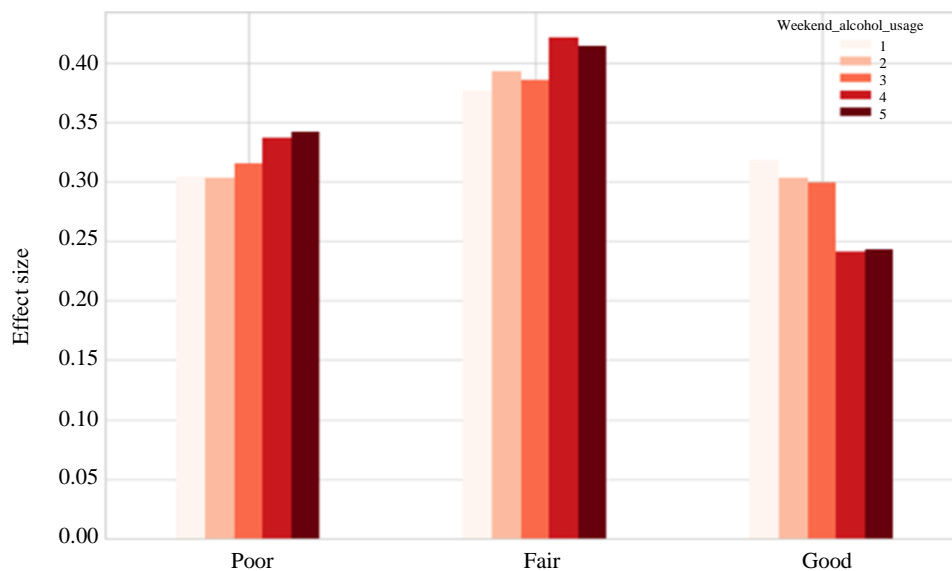
From the dataset described earlier, we can interpret data by using hypothesis testing to answer the following questions:

- Do relationships affect student performance?
- Does drinking alcohol affect student performance?
- Does parental education affect the academic performance of their child?
- How does the frequency of hanging out with friends on academic performance?
- Does a student's residence affect his academic performance?

Based on the analysis, relationships can affect students when it comes to their performance in school and the kind of grades they earn. Relationships play an essential role in understanding student achievement. The relational process is regarded as an inherent aspect of educational life and the foundation for encouraging performance as shown in Fig. 2.



**Fig. 2:** The impact of relationships on students' performance



**Fig. 3:** The effect of alcohol usage on academic achievement

Figure 3 shows the impact of drinking alcohol on students' performance, where they rate how often they drink alcohol from level 1 to 5. Data shows that increases in alcohol consumption has a significant correlation with final grade that result in lower academic performance.

Figure 4 reveals the association between parent's qualification and the academic performance of their children. Perhaps the most noticeable explanation of the relationship between parents' education and the academic achievement of their children is based on the assumption that parents learn something during schooling that effects the ways in which they interact with their children around learning activities in the home. Parents with higher education ensure that their children are exposed to many educational opportunities in their communities.

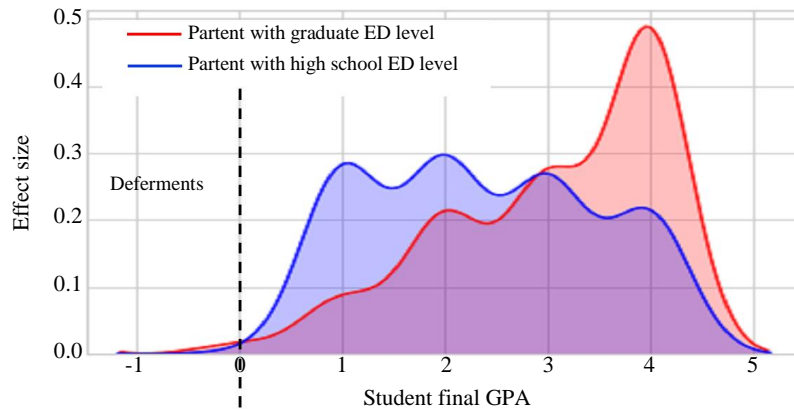


Fig. 4: The relationship between parental educational level and academic performance of children

---

**Algorithm 7: K-Fold Algorithm for The Proposed Prediction Model**

---

**Input:** *n\_split*: the number of splits required, *shuffle*: Boolean shuffle the data point or not

**Output:** *accuracy\_model*

**Begin**

    kf = KFold(*n\_split*, *shuffle*) // Kfold function from sklearn.model\_selection package

    kf.split(x) // The split function splits the index of the X and returns an iterator object

    // The iterator object is iterated using for loop and the integer index of //train and test is printed

        print(" Train Data: ", " Test Data: ")  
         for train\_index, test\_index in kf.split(x)  
             print( train\_index, test\_index)

    // To determine the accuracy for each model, fit method is called to //train the model and accuracy\_score to return the accuracy of the //model.

        // initialize blank list where accuracy of each model is added

        accuracy\_model = []

        // iterate over each

        for train\_index, test\_index in kf.split(x) :

            \\ split train test

            x\_train, x\_test = x.iloc[train\_index], x.iloc[test\_index]

            \\ split train test

            y\_train, y\_test = y[train\_index], y[test\_index]

            // train the model

            model = clf.fit(x\_train, y\_train)

    // append to the list accuracy\_model the accuracy of the model

        accuracy\_model.append(accuracy\_score(y\_test, model.predict(x\_test)))

        print (accuracy\_model) // print the accuracy

---



---

**end for**  
**end for**

---

We also estimate the influence of spending time with friends on educational attainment. More specifically, we rate the frequency from rarely to almost always as illustrated in Fig. 5. We find that spending more time with friends has a significant correlation with final grade.

We have examined rural and urban differences in students' achievement. Many of the students that lives in rural areas has low academic achievement when we compared it with the students that lives in urban areas, as depicted in Fig. 6. The excellent performance of urban students is because of their better quality in their education, availability of the information that they get from various sources.

**B. Model Validation**

Validation is the process of deciding whether the numerical results quantifying hypothesized relationships between variables, are acceptable as descriptions of the data. K-Fold Cross Validation is one of the common methods used for evaluating ML models, where the data is divided into *k* subsets (known as folds). A model is trained using *k*-1 of the folds as training data and the resulting model is validated on the remaining data used as a test set to estimate a performance measure such as accuracy, sensitivity, precision and *f*-measure.

The performance measure reported by *k*-fold cross-validation is the average of the values computed in the loop.

Our model is passed through 5-fold cross validation (*K* = 5) where the dataset is split into 5 folds. In the first iteration, the first fold is used to test the model and the rest are used to train the model. In the second iteration, 2<sup>nd</sup> fold is used as the testing set while the rest serve as the training set. This process is repeated until each fold of the 5 folds have been used as the testing set. The K- fold algorithm used in our prediction model is shown in Algorithm 7.



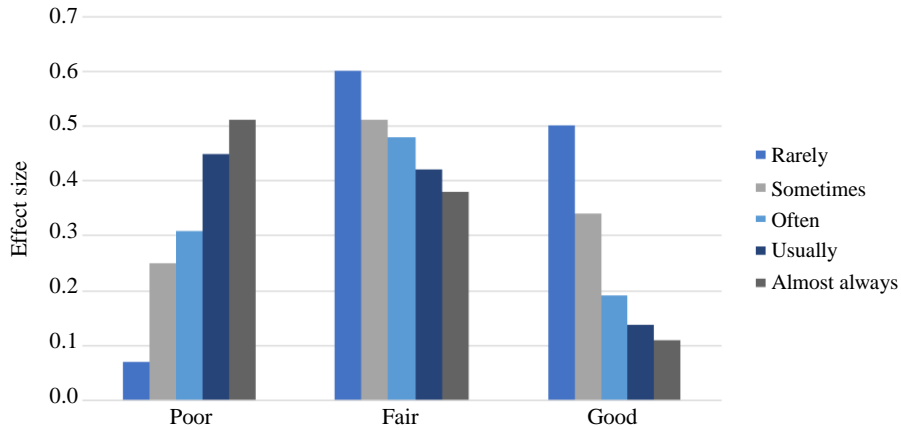


Fig. 5: Final grade by frequency of going out with friends

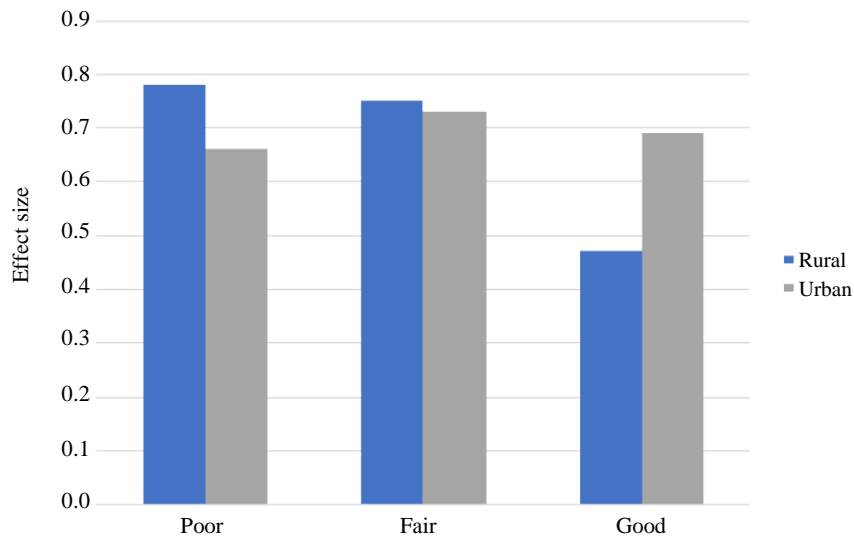


Fig. 6: Comparison of the academic performance of rural and urban students

Table 2: Confusion matrix

		Predicted	
		Positive	Negative
Actual	Positive	$T_p$	$F_n$
	Negative	$F_p$	$T_n$

Table 3: Classifier evaluation measures

Tools	Statistic
Sensitivity	$Sens = T_p / (T_p + F_n)$
Precision	$Prec = T_p / (T_p + F_p)$
Accuracy	$Accu = (T_p + T_n) / (T_p + T_n + F_p + F_n)$
F-measure	$F = (2 * Sens * Prec) / (Sens + Prec)$

### C. Performance Evaluation

All the supervised machine learning algorithms mentioned in this study were deployed in Python (Appendix 1).

The overall performance metric is the average of all five-separate evaluations. The confusion matrix in Table 2 is used to check the result of the testing calculation and provide the number of True Positive (TP), False Positives (FP), True Negatives (TN) and False Negatives (FN). The classifier evaluation measures presented in this section are summarized in Table 3.

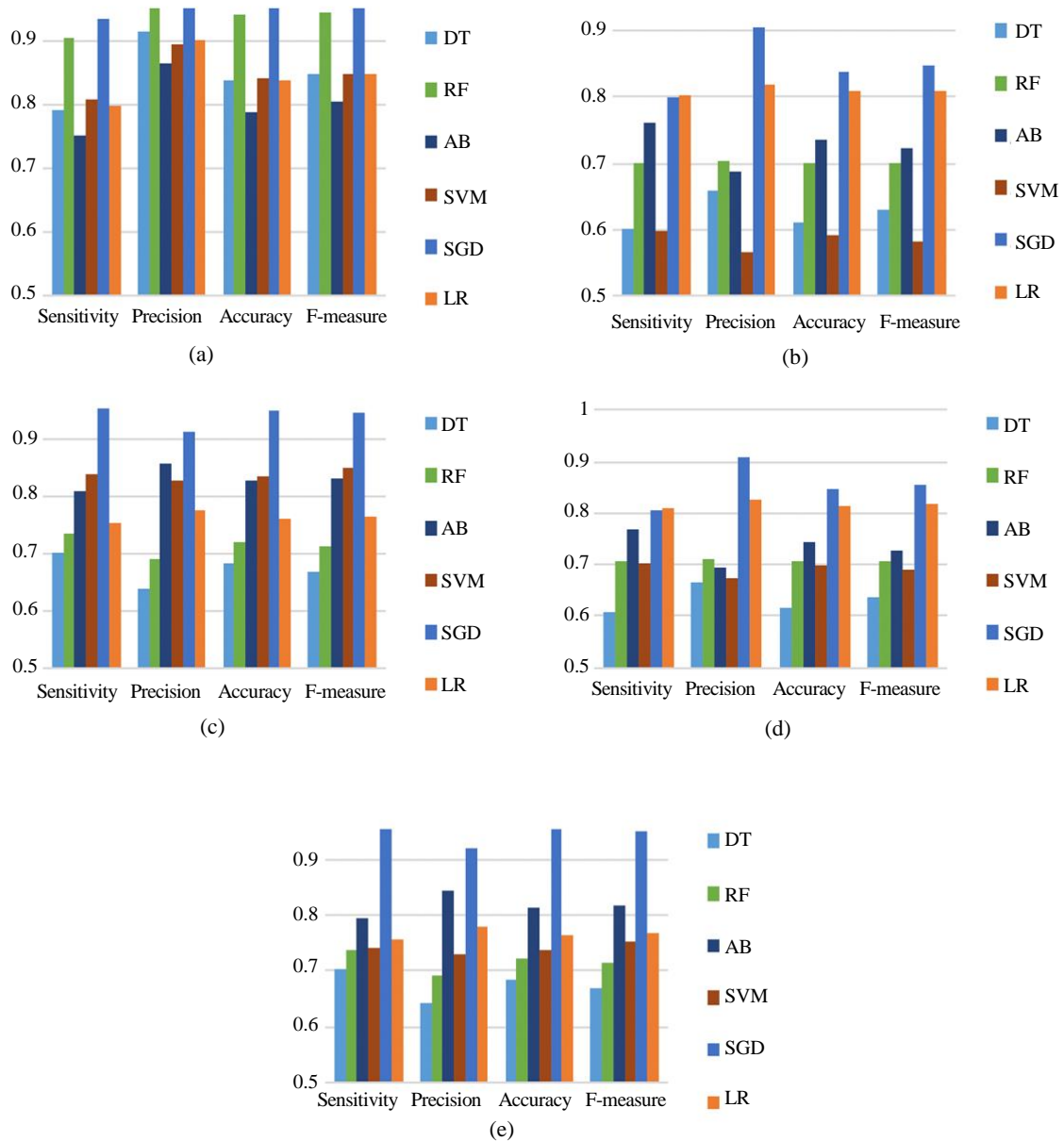
Table 4. shows the classification accuracy and other performance measures results from conducting 5 processes of applying supervised ML algorithms and the corresponding charts are shown in Fig. 7.

We compare the results of six supervised machine learning algorithms, DT, RF, AB, LR, SVM and SGD. In the light of results mentioned in Table 4 and graphical representation of Fig. 7, it has been observed that Stochastic Gradient Descent classifier

has the highest accuracy result where it could classify above 99% all the cases correctly and so it performs the best in our dataset. On the other hand, AdaBoost model has the lowest accuracy as it has 79% correct classification from the entire dataset. In Fig. 7.b, the whole dataset is partitioned into two splits; 70% for training the model and 30% for testing the model. The highest accuracy is achieved by Stochastic Gradient

Descent while the lowest achieved Support Vector Machine.

The overall performance of process three, four and five are demonstrated in Fig. 7c to 7e respectively; The highest accuracy score is obtained by SGD classifier as it classifies most cases in our dataset, while the Decision tree classifier perform the worst and only classifies the half of our dataset.



**Fig. 7:** Performance evaluation of all models based on classifier evaluation measure (a) Comparison between sensitivity, precision, accuracy and f-measure of all learning models in process 1; (b) Comparison between sensitivity, precision, accuracy and f-measure of all learning models in process 2; (c) Comparison between sensitivity, precision, accuracy and f-measure of all learning models in process 3; (d) Comparison between sensitivity, precision, accuracy and f-measure of all learning models in process 4; (e) Comparison between sensitivity, precision, accuracy and f-measure of all learning models in process 5

**Table 4:** Performance evaluation of all models based on classifier evaluation measure

	$T_p$	$F_p$	$T_n$	$F_n$	$Senc$	$Prec$	$Accu$	$F-Score$
<i>Process 1</i>								
DT	0.916	0.0857	0.7624	0.2393	0.7941	0.91609	0.83926	0.8507
RF	0.9896	0.0121	0.8985	0.1032	0.907	0.98963	0.94407	0.9465
AB	0.8662	0.1355	0.714	0.2877	0.7519	0.86622	0.79013	0.805
SVM	0.8975	0.1042	0.7875	0.2143	0.8086	0.89753	0.84251	0.8507
SGD	0.9996	0.0021	0.9334	0.0684	0.9375	0.9996	0.9665	0.9676
LR	0.9037	0.098	0.7727	0.229	0.7991	0.90377	0.83827	0.8482
<i>Process 2</i>								
DT	0.6582	0.3436	0.564	0.4378	0.6016	0.6582	0.6111	0.6286
RF	0.7027	0.2991	0.6992	0.3026	0.7002	0.7027	0.7009	0.7014
AB	0.6878	0.314	0.7847	0.2171	0.7615	0.6878	0.7362	0.7227
SVM	0.5662	0.4356	0.6193	0.3825	0.5979	0.5662	0.5927	0.5816
SGD	0.9037	0.098	0.7727	0.229	0.7991	0.90377	0.83827	0.8482
LR	0.8177	0.1841	0.7987	0.2031	0.8024	0.8177	0.8082	0.8103
<i>Process 3</i>								
DT	0.6384	0.3634	0.7265	0.2753	0.6999	0.6384	0.6824	0.6677
RF	0.6898	0.312	0.7498	0.252	0.7337	0.6898	0.7198	0.7111
AB	0.8582	0.1436	0.7965	0.2053	0.8083	0.8582	0.8273	0.8325
SVM	0.8263	0.1755	0.8406	0.1612	0.8382	0.8263	0.8334	0.8507
SGD	0.9146	0.0872	0.9844	0.0174	0.9831	0.9146	0.9495	0.9476
LR	0.7765	0.2253	0.7475	0.2543	0.7546	0.7765	0.7620	0.7654
<i>Process 4</i>								
DT	0.6653	0.3507	0.5711	0.4449	0.6087	0.6653	0.6182	0.6357
RF	0.7098	0.3062	0.7063	0.3097	0.7073	0.7098	0.708	0.7085
AB	0.6949	0.3211	0.7918	0.2242	0.7686	0.6949	0.7433	0.7298
SVM	0.6733	0.5427	0.7264	0.4896	0.705	0.6733	0.6998	0.6887
SGD	0.9108	0.1051	0.7798	0.2361	0.8062	0.91087	0.84537	0.8553
LR	0.8248	0.1912	0.8058	0.2102	0.8095	0.8248	0.8153	0.8171
<i>Process 5</i>								
DT	0.6415	0.3665	0.7296	0.2784	0.703	0.6415	0.6855	0.6708
RF	0.6929	0.3151	0.7529	0.2551	0.7368	0.6929	0.7229	0.7142
AB	0.8443	0.1297	0.7826	0.1914	0.7944	0.8443	0.8134	0.8186
SVM	0.7294	0.0786	0.7437	0.0643	0.7413	0.7294	0.7365	0.7538
SGD	0.9177	0.0903	0.9875	0.0205	0.9862	0.9177	0.9526	0.9507
LR	0.7796	0.2284	0.7506	0.2574	0.7577	0.7796	0.7651	0.7685

## Conclusion and Future Work

In this study, we have studied the execution of six machine learning algorithms, namely, Decision tree, Random Forest, Support Vector Machine, Logistic Regression, Ada Boost, Stochastic Gradient Descent for student academic performance predictions. The hypothesis testing results shows that student academic performance may be affected by relationships, alcohol use, parental educational level and their residential area. The evaluation measures of the methods are further determined in terms of sensitivity, precision, accuracy and F-measure achieved from five processes. The overall results of the observed dataset showed that Stochastic Gradient Descent classification algorithm is considered as the most accurate classifier and outperformed many other supervised machine learning classification algorithms in various cases. The future work involves presenting intelligent machine learning algorithms useful to a huge collection of real dataset and higher accuracy

percentage. More attributes can enhance the performance and give more precise prediction.

## Author's Contributions

**Abdul Razaque:** Conceived of the presented idea, verified the analytical methods, supervised the findings of this work and discussed the results and contributed to the final manuscript.

**Abrar M. Alajlan:** Developed the theory and performed the computations, discussed the results and contributed to the final manuscript and wrote the manuscript with support from Dr. Abdul Razaque.

## Ethics

This manuscript is original and has not been published anywhere. The confirmation from the corresponding author is that the other author has read and approved the manuscript and has no ethical issue involved.

## References

- Ahmad, S. A., Hussain, I., Ahmad, R., & Din, M. N. U. (2020). Performance Based Prediction of the Students in the Physics Subject using Traditional and Machine Learning Approach at Higher Education Level. *International Journal of Innovation in Teaching and Learning (IJITL)*, 6(1), 174-190.
- Amrieh, E. A., Hamtini, T., & Aljarah, I. (2016). Mining educational data to predict student's academic performance using ensemble methods. *International Journal of Database Theory and Application*, 9(8), 119-136.
- An introduction to seaborn. (n.d.). <https://seaborn.pydata.org/introduction.html>
- Angeli, C., Howard, S. K., Ma, J., Yang, J., & Kirschner, P. A. (2017). Data mining in educational technology classroom research: Can it make a contribution?. *Computers & Education*, 113, 226-242.
- Ashraf, M., Zaman, M., & Ahmed, M. (2020). An Intelligent Prediction System for Educational Data Mining Based on Ensemble and Filtering approaches. *Procedia Computer Science*, 167, 1471-1483.
- Asif, R., Merceron, A., Ali, S. A., & Haider, N. G. (2017). Analyzing undergraduate students' performance using educational data mining. *Computers & Education*, 113, 177-194.
- Barteit, S., Guzek, D., Jahn, A., Bärnighausen, T., Jorge, M. M., & Neuhann, F. (2020). Evaluation of e-learning for medical education in low-and middle-income countries: A systematic review. *Computers & Education*, 145, 103726.
- Cortez, P. (n.d.). Student Performance Data Set, UCI Machine Learning Repository. <http://www3.dsi.uminho.pt/pcortez>
- Czibula, G., Mihai, A., & Crivei, L. M. (2019). S PRAR: A novel relational association rule mining classification model applied for academic performance prediction. *Procedia Computer Science*, 159, 20-29.
- ECOSYSTEM. (n.d.). NumPy. <https://www.numpy.org/>
- Hlosta, M., Zdrahal, Z., & Zendulka, J. (2017, March). Ouroboros: early identification of at-risk students without models based on legacy data. In *Proceedings of the seventh international learning analytics & knowledge conference* (pp. 6-15).
- Hussain, M., Zhu, W., Zhang, W., Abidi, S. M. R., & Ali, S. (2019). Using machine learning to predict student difficulties from learning session data. *Artificial Intelligence Review*, 52(1), 381-407.
- Injadat, M., Moubayed, A., Nassif, A. B., & Shami, A. (2020). Systematic ensemble model selection approach for educational data mining. *Knowledge-Based Systems*, 105992.
- Ivanović, M., Milićević, A. K., Aleksić, V., Bratić, B., & Mandić, M. (2018). Experiences and perspectives of Technology-enhanced learning and teaching in higher education–Serbian case. *Procedia Computer Science*, 126, 1351-1359.
- Kumar, M., & Singh, A. J. (2019). Performance Analysis of Students Using Machine Learning & Data Mining Approach. *Int. J. Eng. Adv. Technol.*, 8(3), 75-79.
- Pandas, Python Data Analysis Library. (n.d.). <https://pandas.pydata.org/>
- Peña-Ayala, A. (2014). Educational data mining: A survey and a data mining-based analysis of recent works. *Expert systems with applications*, 41(4), 1432-1462.
- Razaque, A., Rizvi, S., Almiani, M., & Al Rahayfeh, A. (2019). State-of-art review of information diffusion models and their impact on social network vulnerabilities. *Journal of King Saud University-Computer and Information Sciences*.
- Real Python. (2020, May 29). Jupyter Notebook: An Introduction. <https://realpython.com/jupyter-notebook-introduction/>
- Roy, S., & Garg, A. (2017, October). Predicting academic performance of student using classification techniques. In *2017 4th IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics (UPCON)* (pp. 568-572). IEEE.
- Sati, N. U. (2018). Semi-Supervised Classification in Educational Data Mining: Students' Performance Case Study. *International Journal of Computer Applications*, 179(26), 13-17.
- Scikit-Learn. (n.d.). Scikit-learn/scikit-learn. <https://github.com/scikit-learn/scikit-learn>
- Tomasevic, N., Gvozdenovic, N., & Vranes, S. (2020). An overview and comparison of supervised data mining techniques for student exam performance prediction. *Computers & Education*, 143, 103676.
- Tran, T. O., Dang, H. T., Dinh, V. T., & Phan, X. H. (2017). Performance prediction for students: a multi-strategy approach. *Cybernetics and Information Technologies*, 17(2), 164-182.
- Varvarigos, D. (2020). Cultural transmission, education-promoting attitudes and economic development. *Review of Economic Dynamics*.
- Visualization with Python, Matplotlib. (n.d.). <https://matplotlib.org/>
- Welcome to Statsmodels's Documentation. (n.d.). <https://www.statsmodels.org/stable/index.html>
- What is Python? Executive Summary. (n.d.). <https://www.python.org/doc/essays/blurb/>

Yahya, A. A., & Osman, A. (2019). Using Data Mining Techniques to Guide Academic Programs Design and Assessment. *Procedia Computer Science*, 163, 472-481.

Zhang, C., Almpandis, G., Wang, W., & Liu, C. (2018). An empirical evaluation of high utility itemset mining algorithms. *Expert Systems with applications*, 101, 91-115.

## Appendix:

### Appendix 1: Software and tools used for conducting the experiment

---

Tools	Description
Python	Python is a high-level, interpreted, object-oriented programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic binding and dynamic typing, make it very good for Rapid Application Development (What is Python? Executive Summary. (n.d.)).
Jupyter notebook	The Jupyter Notebook is an open-source web application that allows you to share and create documents that have live equations, code, narrative text and visualizations (Real Python, 2020).
numpy	NumPy is the fundamental package for scientific computing with Python (ECOSYSTEM. (n.d.)).
pandas	Pandas is an open source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language (Pandas, Python Data Analysis Library. (n.d.)).
matplotlib	Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and IPython shells, the Jupyter notebook, web application servers and four graphical user interface toolkits (Visualization with Python, Matplotlib. (n.d.)).
seaborn	Seaborn is a library for making statistical graphics in Python. It is built on top of matplotlib and closely integrated with pandas data structures (An introduction to seaborn. (n.d.)).
Statsmodels.api	statsmodels is a Python module that provides classes and functions for the estimation of many different statistical models, as well as for conducting statistical tests and statistical data exploration (Welcome to Statsmodels's Documentation. (n.d.)).
scikit-learn	scikit-learn is a Python module for machine learning built on top of SciPy and distributed under the 3-Clause BSD license (Scikit-Learn (n.d.)).

---