CH5019: Prediction of student performance in secondary education.

Ahmed Shmels Muhe
M.Tech In Data Science
Indian Institute of Technology Madras, IIT Madras
600036, Chennai, Tamil Nadu, India
ge22m009@smail.iitm.ac.in

***Abstract*—In this study, we investigated the effect of various factors on students' academic success using real-world data, including grades, demographics, social factors, and school-related features. We applied three machine learning algorithms - logistic regression, K-nearest neighbours (KNN), and support vector machines (SVMs) - to assess the performance of students in mathematics and Portuguese, the two core subjects. We evaluated the performance of the models using the f1 score, ROC curve, and confusion matrix and found that SVM had the highest accuracy at 84%. Our descriptive analysis revealed that several factors had a positive impact on student's academic success, including the father's education level, guardian involvement, desire to pursue higher education, and study time. Conversely, factors such as the number of absences, parents' employment and education, socializing with friends, health, and alcohol consumption had a negative impact on academic performance.**

***Index Terms*—logistic regression, SVM, KNN, F1 score, Confusion matrix, descriptive analysis.**

I. INTRODUCTION AND BACKGROUND OF THE STUDY

The present study analyzed data collected during the 2005-2006 school year from two public schools in the Alentejo region of Portugal [1]. Education is a crucial aspect of economic development in Portugal, but the country has struggled with high rates of student failure and dropout. In 2006, for instance, the early school leaving rate in Portugal was 40% for 18-24 year-olds, compared to a European Union average of 15% [1] [2].

Different research in the literature analyses and verifies the development and evaluation of ML tools to classify data and learn from patterns to know what's going on under the hood.

In the previous study by [3] created a Naive Bayes classification model to accurately identify factors affecting student performance, the authors suggest their prediction model can then be used to implement early intervention to give Timely feedback for educators to be able to provide early intervention strategies. a study by [4] gave a case study that uses students' data to analyze their learning behaviour to predict the results and to warn students at risk before their final exams.

Bhardwaj and Pal [5] conducted a study in Faizabad, India, to identify the factors that had the greatest impact on student performance. For their research, they used Bayesian classification. Their research found that student's academic performance is not always dependent on their own efforts, but rather on their living location, medium of instruction, mother's qualification, students' other habits, family annual income, and students' family status, all of which are high potential variables that affect students' performance.

Erkan Er's study proposes a model for predicting student performance levels using ML algorithms: an instance-based learning classifier, a decision tree, and a naïve Bayes classifier [6]. These algorithms were used with three decision schemes to see if better classification performance could be achieved. The study found that the instance-based algorithm K-star had the best performance compared to the other algorithms [7]. The authors also concluded that using initial attendance and homework grades as predictors produced a better prediction rate at earlier stages than using demographic characteristics of students [6]. The authors suggest that a potential future study using only instance-based learning algorithms and combining the results of these algorithms again using decision schemes may result in more accurate results [2].

In particular, failure in the core classes of mathematics and Portuguese (the native language) is extremely serious since they provide the fundamental knowledge for success in the remaining school subjects (e.g., physics, astronomy and computer) [1]. What types of systems can integrate into these courses to attract more students? What are the main reasons for student failure in this subject and What are the factors that affect student achievement? It is possible to predict student performance.

Modelling student performance is an important tool for educators and students since it can help better understand this problem of student failure and ultimately improve it. For instance, school professionals could perform corrective measures for weak students (e.g., remedial and tutorial classes) [8]. This article will focus on how to figure out the reason for student failure in the two core classes (i.e., mathematics and Portuguese), and the two core classes will be modelled under three decision-making goals as follows:

1) regression, with a numeric output that ranges between zero (0%) and twenty (100%).
2) classification with five levels (from I very good or excellent to V - insufficient); and
3) binary classification (pass/fail).

For each of these approaches, three input setups (e.g. with and without the school period grades) and two classification-making algorithms (e.g.logistic regression, Random Forest)

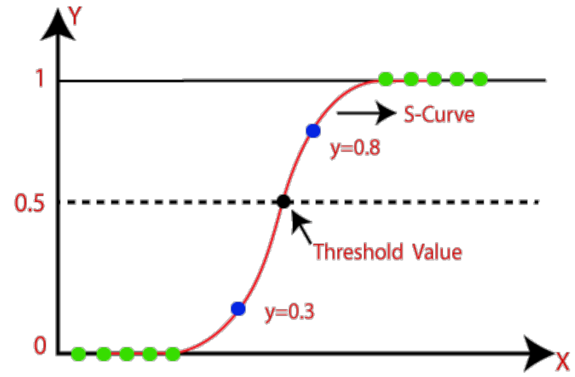

Fig. 1. example of logistic regression

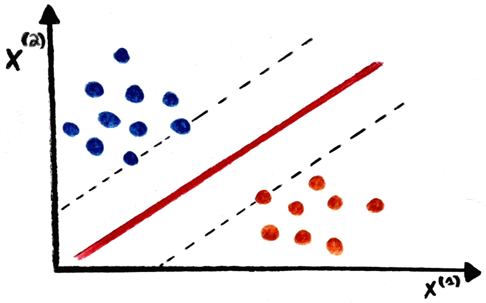

Fig. 2. Example of Supported vector machine

will be tested. Moreover, an explanatory analysis will be performed over the best models, in order to identify the most relevant features that correlate to student failure.

II. Methodology and Algorithms Used

A. Logistic regression

Logistic regression is an extension of linear regression, and by making two adjustments, we may expand linear regression to the (binary) classification scenario. First, we substitute the Gaussian distribution for y with a Bernoulli distribution, which is better suited for binary responses, $y \epsilon 0, 1$. That is, we employ [9].

Logistic regression is a statistical analysis method to predict a binary outcome, such as yes or no, based on prior observations of a data set. A logistic regression model predicts a dependent data variable by analyzing the relationship between one or more existing independent variables.

$$p(y|x,w) = Ber(y|\mu(x)) \tag{1}$$

where $\mu(x) = E[y|x] = p(y = 1|x)$. Second, we compute a linear combination of the inputs, as before, but then we pass this through a function that ensures $0 \leq \mu(x) \leq 1$ by defining

$$\mu(x) = \sigma w^T(x) \tag{2}$$

where $\sigma(\eta)$ refers to the sigmoid function, also known as the logistic function. This is defined as

$$\sigma(\eta) = \frac{exp(\eta)}{exp(\eta)+1} \tag{3}$$

Putting these two steps together we get

$$p(y|x,w) = Ber(y|sigm(w^T x)) \tag{4}$$

This is called logistic regression due to its similarity to linear regression (although it is a form of classification).

A simple example of logistic regression is shown above a figure, where we plot

$$p(y_i = 1|x_i, w) = \sigma(w_0 + w_1 x_i) \tag{5}$$

If we threshold the output probability at 0.5, we can induce a decision rule of the form

$$y(x) = 1 \iff p(y = 1|x) > 0.5 \tag{6}$$

Logistic regression may be readily expanded to handle higher-dimensional inputs. If we set the threshold for these probabilities to 0.5, we have a linear decision boundary with the normal (perpendicular) provided by w. In the next part, we shall derive the loss function associated with logistic regression [9].

1) Maximum likelihood estimation: The negative log-likelihood for logistic regression is given by:

$$NLL(w) = -\sum_{i=0}^{N} \log[\mu_i^{(I(y_i=1)} \times (1-\mu_i)^{(y_i=0)}] =$$
$$-\sum_{i=0}^{N} [y_i \log \mu_i + (1-y_i)\log(1-\mu_i] \tag{7}$$

This is also called the cross-entropy error function. Unlike linear regression, we can no longer write down the Maximum Likelihood Estimation in closed form. Instead, we need to use an optimization algorithm to compute it. For this, we need to derive the gradient and Hessian. In the case of logistic regression, the gradient and Hessian are given by the following

$$g = \frac{d}{dw} f(w) = \sum_i (\mu_i - y_i) x_i = X^T(\mu, y) \tag{8}$$

$$h = \frac{d}{dw} g(w)^T = \sum_i (\Delta_w \mu_i) x_i^T = S^T S X \tag{9}$$

where $S = diag(\mu_i(1-\mu_i))$. Here the NLL is convex and has a unique global minimum.

B. Supported vector machine (SVM)

A support vector machine constructs a hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be used for classification, regression, or other tasks like outliers detection. We are given a training dataset of n points of the form

$$(x1, y1), ..., (x_n, y_n)$$

where the y_i is either 1 or -1, each indicating the class to which the point x_i belongs. Each x_i is a p dimensional real vector. We want to find the ”maximum-margin hyperplane” that divides the group of points x_i for which $y_i = 1$ from the group of points for which $y_i = -1$, which is defined so that the distance between the hyperplane and the nearest point x_i from either group is maximized. Any hyperplane can be written as the set of points x satisfying

$$w^T x - b = 0$$

Geometrically, the distance between these two hyperplanes is $\frac{2}{||\mathbf{w}||}^{[17]}$ so to maximize the distance between the planes we want to minimize $||\mathbf{w}||$. The distance is computed using the distance from a point to a plane equation [**?**]. We also have to prevent data points from falling into the margin, we add the following constraint: for each i either

$$w^T x_i - b \geq 1, if y_i = 1$$

or

$$w^T x_i - b \leq -1, if y_i = -1$$

These constraints state that each data point must lie on the correct side of the margin. This can be rewritten as

$$y_i(w^T x_i - b \geq 1, forall 1 \leq_i \leq n$$

We can put this together to get the optimization problem: "Minimize $|\mathbf{w}|$ subject to y_i $(\mathbf{w}^T \mathbf{x}_i - b) \leq 1$ for $i = 1, ..., n$." The w and b that solve this problem determine our classifier, $\mathbf{x} \mapsto sgn(\mathbf{w}^T \mathbf{x} - b$ where $sgn(.)$ is the sign function. An important consequence of this geometric description is that the max-margin hyperplane is completely determined by those x_i that lie nearest to it. These x_i are called support vectors. To extend SVM to cases in which the data are not linearly separable, the hinge loss function is helpful

$$max(0, 1 - y_i(\mathbf{w}^T \mathbf{x}_i - b))$$

Note that y_i is the $i-th$ target (i.e., in this case, 1 or -1), and $\mathbf{w}^T \mathbf{x} - b$ is the i^{th} output. This function is zero if the constraint in (1) is satisfied, in other words, if x_i lies on the correct side of the margin. For data on the wrong side of the margin, the function's value is proportional to the distance from the margin. The goal of the optimization then is to minimize

$$\lambda ||\mathbf{w}||^2 + [\frac{1}{n} \sum_{i=n}^{m} max(0, 1 - y_i(\mathbf{w}^T \mathbf{x}_i - b))]$$

where the parameter $\lambda > 0$ determines the trade-off between increasing the margin size and ensuring that the x_i lie on the correct side of the margin. Thus, for sufficiently small values of λ, it will behave similarly to the hard-margin SVM, if the input data are linearly classifiable, but will still learn if a classification rule is viable or not. (This parameter λ is also called C, e.g. in LIBSVM.)

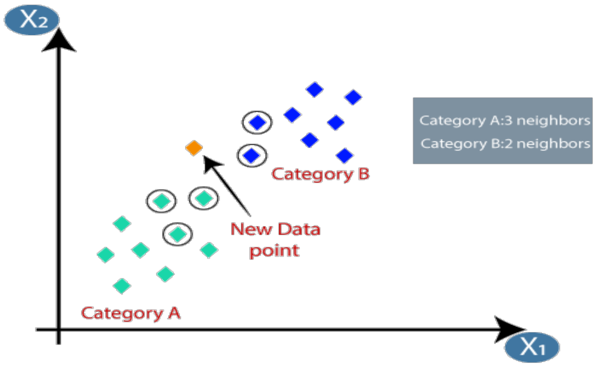

Fig. 3. Example of KNN

C. K-nearest neighbours (KNN)

K-nearest neighbours (KNN) algorithm is a type of supervised ML algorithm which can be used for both classifications as well as regression predictive problems. However, it is mainly used for the classification of predictive problems in the industry. The following two properties would define KNN well.

KNN is a lazy learning algorithm because it does not have a specialized training phase and uses all the data for training while classification.

The K-NN working can be explained on the basis of the below Algorithm:

1) Select the number K of the neighbours.
2) Calculate the Euclidean(or any other type of distance) distance of K number of neighbours.
3) Among the k nearest neighbours, count the number of the data points in each category.
4) Assign the new data points to that category for which the number of neighbours is maximum.

lets take a point $\mathbf{x}$ and define the set of the k nearest neighbours of $\mathbf{x}$ as $S_{\mathbf{x}}$. Formally S_x $\mathbf{D} s.t||S_{\mathbf{X}} = k$ & $\forall(\mathbf{x}^{'}, y^{'})$.

$$\text{dist}(\mathbf{x}, \mathbf{x}') \geq \max_{(\mathbf{x}'', y'') \in S_{\mathbf{x}}} \text{dist}(\mathbf{x}, \mathbf{x}''),$$

(i.e. every point in D but not in S_x is at least as far away from s as the furthest point in S_x). We can then define the classifier $h()$ as a function returning the most common label in S_x:

$$h(\mathbf{x}) = \text{mode}(\{y'' : (\mathbf{x}'', y'') \in S_{\mathbf{x}}\}),$$

where $mode(.)$ means to select the label of the highest occurrence. The k-nearest neighbour classifier fundamentally relies on a distance metric. The better that metric reflects label similarity, the better the classification will be. The most common choice is the Minkowski distance:

$$\text{dist}(\mathbf{x}, \mathbf{z}) = \left(\sum_{r=1}^{d} |x_r - z_r|^p \right)^{1/p}.$$

III. Exploratory Data Analysis

In summary, this dataset includes data on 395 students' academic performance in secondary education from two Portuguese schools, including demographic, social, and school-related features. The data was collected using school reports and questionnaires and includes a column indicating whether or not the student passed their final exam. There are no missing values in the data let's see some of the features

- **famsize** : family size (binary: "LE3" - less or equal to 3 or "GT3" - greater than 3)
- **status**: parent's cohabitation status (binary: "T" - living together or "A" - apart)
- **Medu**: mother's education (numeric: 0 - none, 1 - primary education (4th grade), 2 - 5th to 9th grade, 3 - secondary education or 4 - higher education)
- **Fedu**: father's education (numeric: 0 - none, 1 - primary education (4th grade), 2 - 5th to 9th grade, 3 - secondary education or 4 - higher education)
- **Mjob**: mother's job (nominal: "teacher", "health" care related, civil "services" (e.g. administrative or police), at home or other)
- **guardian**: student's guardian (nominal: "mother", "father" or "other")
- **studytime**: weekly study time (numeric: 1 - <2 hours, 2 - 2 to 5 hours, 3 - 5 to 10 hours, or 4 - >10 hours)

A. Feature Engineering

1) Sometimes, datasets came up with non-numerical values and it is impossible to give them to any classifier. So our job is to convert non-numeric values to numerical ones. And we will do that by calling: $defnumerical - data()$ - This function will map each string to an appropriate integer.

B. Feature scaling

1) Feature scaling is a method used to normalize the range of independent variables or features of data. Data processing is also known as data normalization and is generally performed during the data preprocessing step. This will help our learning algorithms to converge quickly.

C. Data Normalization

1) All values are scaled in a specified range between 0 and 1 via normalisation (or min-max normalisation). This modification has no influence on the feature's distribution. Just by calling: $deffeature - scaling(df)$, This function takes the dataset as an argument and replaces each column, let's say 'col', to :

$$\frac{col - mean(col)}{max(col)},$$

, where $mean$: is the mean or the average.

D. Data Standardization/z-score normalisation

it is the process of scaling values while accounting for standard deviation. To arrive at a distribution with a 0 mean and 1 variance, all the data points are subtracted by their mean and the result is divided by the distribution's variance.

E. Data visualisation

Firstly we are going to look deeper into each feature by using multiple methods of visualisation such as distribution plot, Density... After the visualisation, we are going to understand which features are most impactful for students' performances .

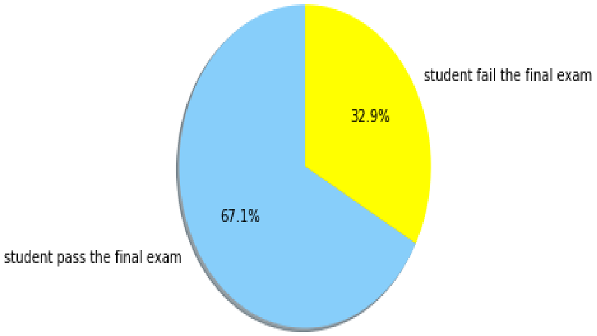

Fig. 4. Student pass the final exam vs student fails the final exam

F. Correlation of the data set

Based on this heatmap we can do a quick conclusion about the most impactful features on student status:

1) The three most impactful (positive) features are mother and father with height education had a positive impact on student performance and the students who want to take higher education having also had good grades.
2) The three most impactful (negative) features are going out with friends for many hours can impact badly and age and failures are other features that also impact negatively student performances.
   In the next steps we will confirm this conclusion by using a distribution plot, and density graphs, let's look deeper into each feature and make a final summary of the best social, demographic and school conditions.

G. Distribution plot and Descriptive analysis of the data

1) Student status By goout: it seems that most of the people who passed the exam had fewer hours of going out, as a conclusion we should limit the hour of going out with friends

2) Student status by romantic relation: Most of the people who passed the exam had no romantic relation, so no relation could be a good choice for better performance.

3) Student status by mother job : It seems that students whose mother work as doctors reach good status

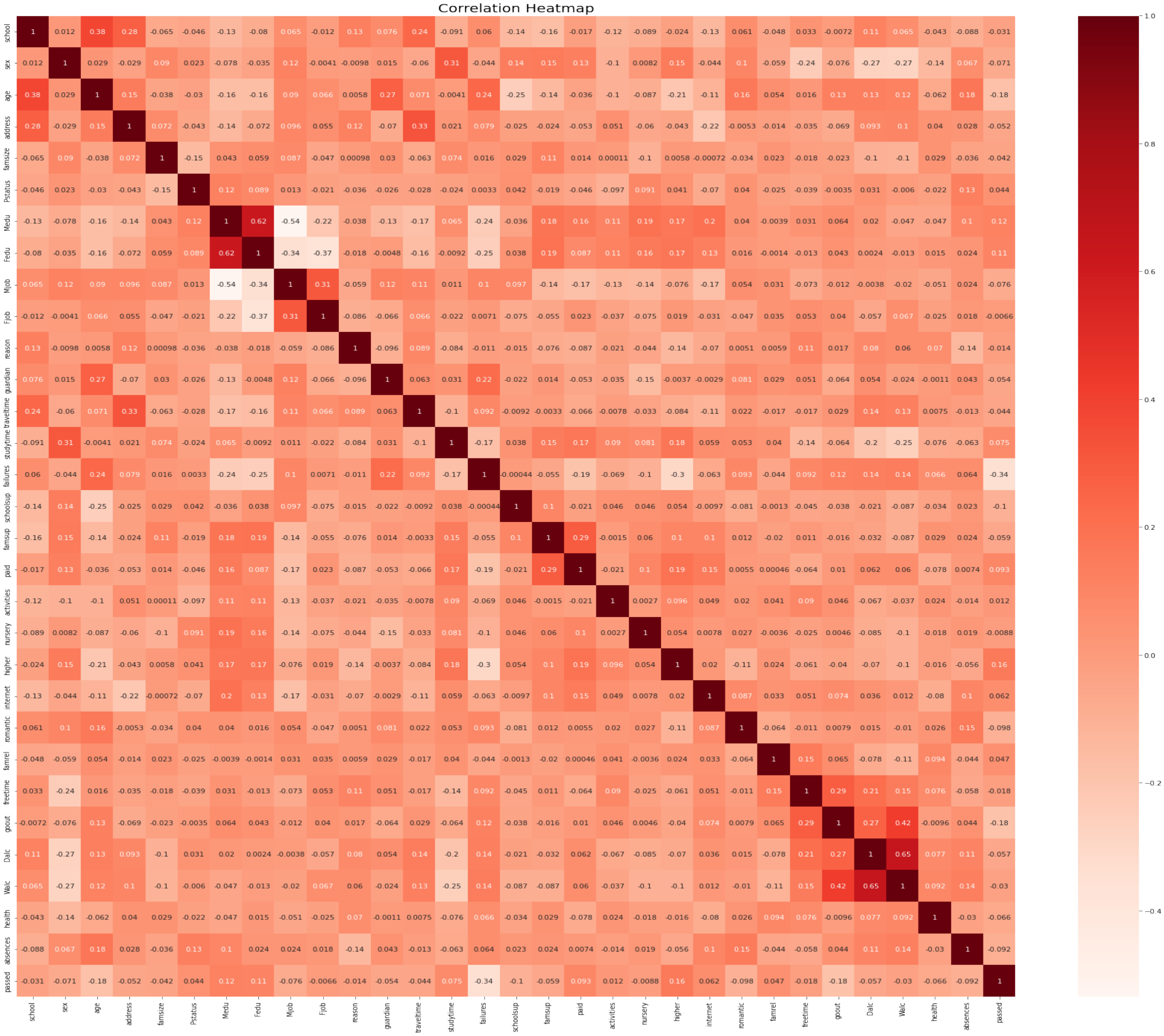

Fig. 5. correlation between variables through a correlation heatmap

4) Student status by mother job and mother education: Mother height education had a good impact on student status. If we look into the second heatmap previously it seems that Medu is more impactful than Fedu.

5) Student status by a desire to take higher education: Most of the people who passed the exam want to take higher education sow it could be a good idea to encourage your kids or students to take higher education.

6) Student status by age: Age also plays an important role in student success, most of the people who passed the exam had an early age of 15, and most people who failed the exam had an age of 22 . In a conclusion, it could be better to go to school at an early age.

7) Student status by alcohol consumption: Weekly alcohol consumption, doesn't have a strong impact on student performance. Even people with low consumption had low grad.

8) Student status by weekly Study time: Most of the people who passed the exam study 5-10 hours weekly.

Generally based on the analysis of the data, After dealing with the most relevant features, the valedictorian of excellent conditions for high academic potential is likely to have this profile:

1) Does not go out with friends frequently
2) Is not in romantic relation
3) Parents receive higher education, especially woman
4) Have a strong desire to receive higher education
5) Mother is a health care professional

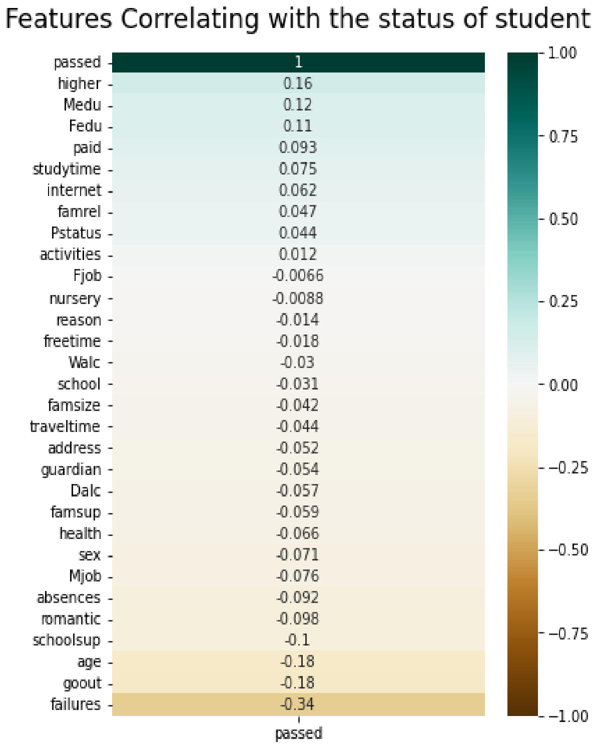

Fig. 6. correlation between student status and other features

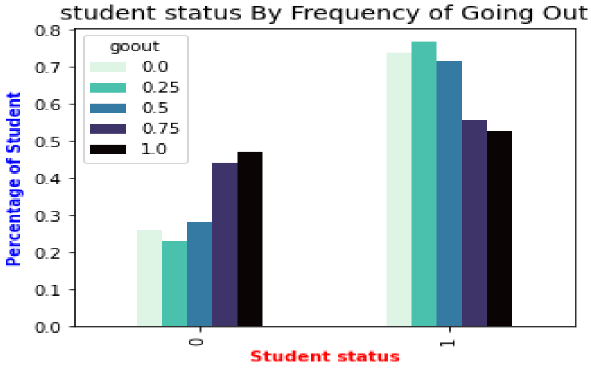

Fig. 7. student status By goout

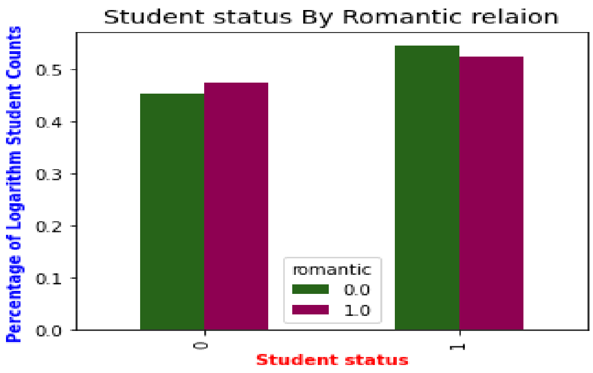

Fig. 8. Student status by romantic relation

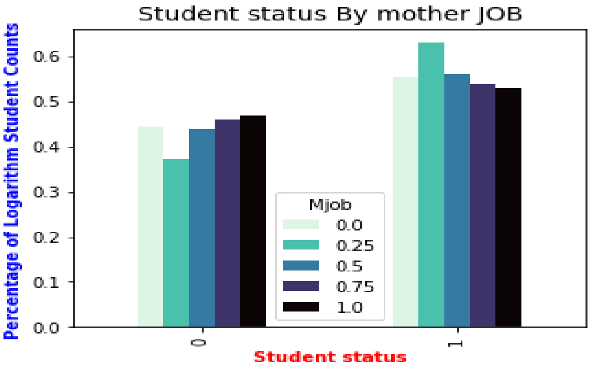

Fig. 9. Student status by mother job and mother education

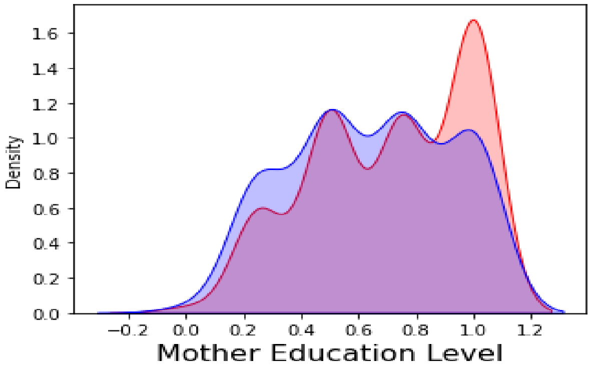

Fig. 10. Student status by mother job and mother education

6) father is a teacher
7) No absences to classes
8) has access to the internet
9) study more than 10 hours a week
10) Is healthy

IV. Model Evaluation and Comparison

In summary, you are trying to predict whether a student will pass their final exam based on specific information about them. You will use various classifiers, such as KNN and SVM,

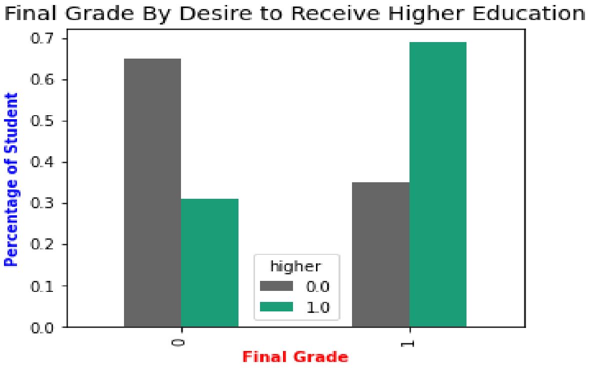

Fig. 11. Student status by the desire to take higher education

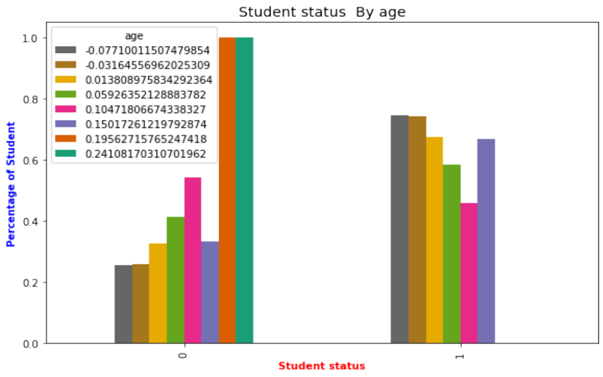

Fig. 12. Student status by age

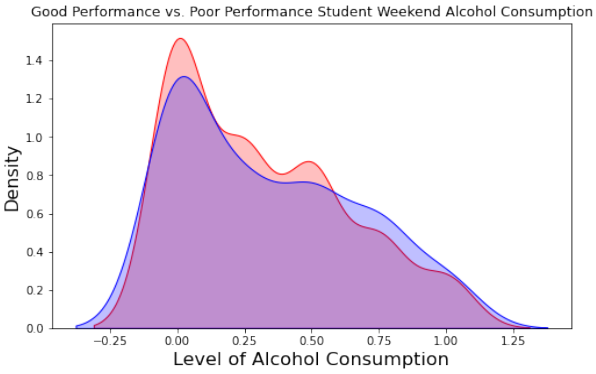

Fig. 13. Student status by alcohol consumption

to make this prediction and compare the performance of these classifiers to determine the most accurate. You will also use techniques to prevent overfitting and underfitting and consider the factors that most impact student performance.

A. Result of the Logistic Regression model

In order to evaluate our model, we will first calculate the accuracy of the model, visualize the confusion matrix, and then plot the ROC curve.

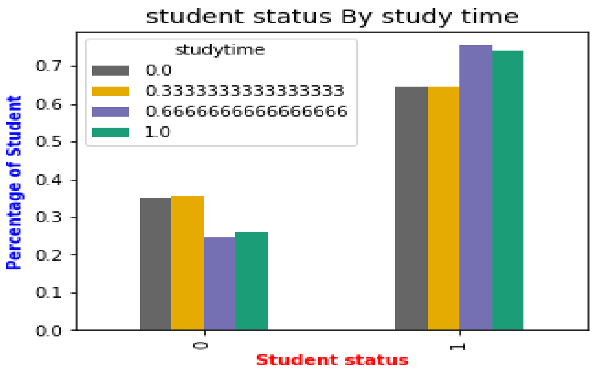

Fig. 14. Student status by weekly Study time

We got two values of accuracy, one obtained with the training set and the other with the test set test of Accuracy of 0.64 and train accuracy of 0.75, It might be a good idea to compare the two, in a situation where the training set accuracy is much higher might indicate overfitting. The test set accuracy is more relevant for evaluating the performance on unseen data since it's not biased.

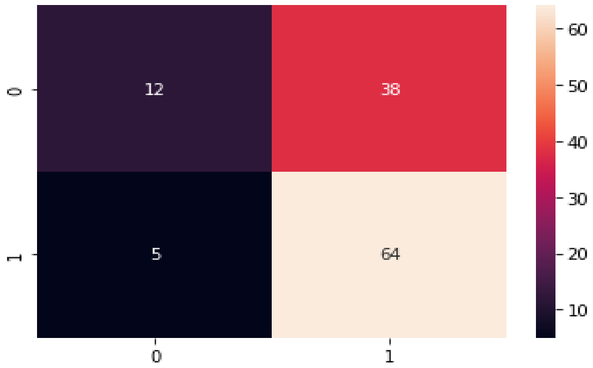

Fig. 15. visualize the confusion matrix: Accuracy of the Logistic Regression model

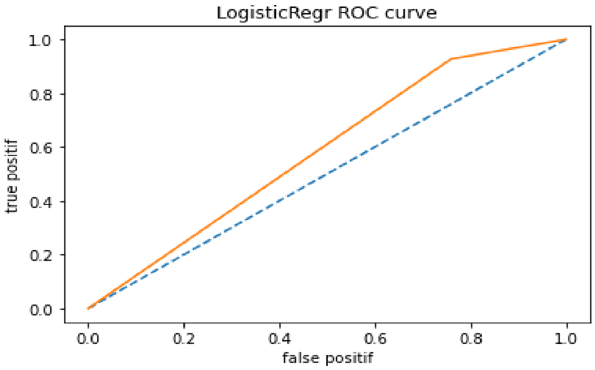

Fig. 16. visualize the ROC curve: Accuracy of the Logistic Regression model

improving model accuracy, The algorithm gives different accuracy each time we change the data split. And we know that if we built a good model, then the accuracy should not vary too much depending on the random state. ...But still, we can train the model for some iterations and instead of using the values 0 and 1 for the random state, we will choose the value $optimal_state$ that maximizes the accuracy and the F1 score for the iterations given. after improving the model we notice that we went from an accuracy of 64% to 80.67% and we got a higher value for $F1$ score as well; from 0.55 to 0.74.

B. Result of the k-nearest neighbors model

we evaluate this model and see the impact of knn parameters by using a roc curv and a confusion matrix and an f1 score. Using this model I got an accuracy of 78% which is a good accuracy and the f1 score is 70%.

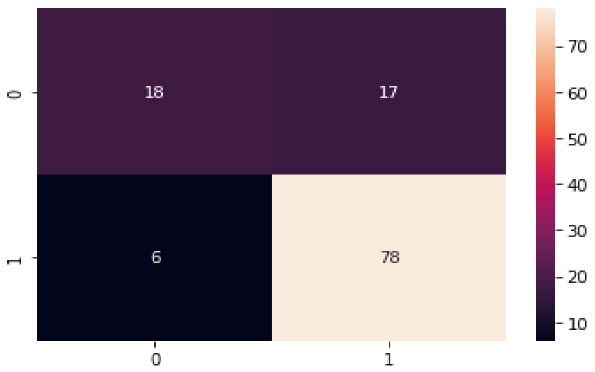

Fig. 17. improving model accuracy: visualize the confusion matrix of logistic Regression model

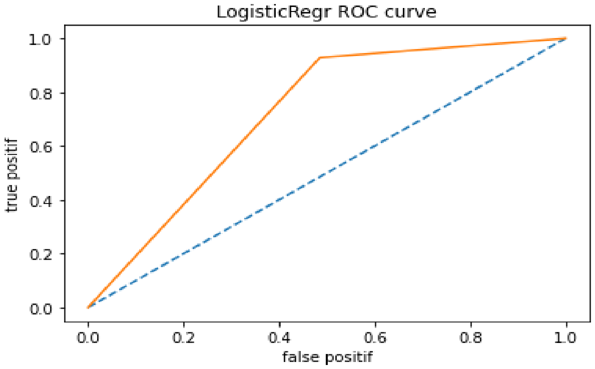

Fig. 18. improving model accuracy, the ROC curve of Logistic Regression model

By tuning parameters related to knn algorithm (choice of k_value,metric=distance) by using these 3 methods we can achieve better accuracy.

1) by visualisation: In this method, we are going to choose best K base on visualisation, In our case study we had a binary classification sow it could be better to choose an odd value of K. By looking into the curv We can observe above that we get maximum testing accuracy for k=5 .In next step let's confirm if 5 was a good choice by using gridsearchCV.
2) using GridSearchCV, In this step we will confirm if our tuning is correct for k value by using gridsearch cross-validation. "Hyperparameter tuning is used in order to determine the optimal values for a given model. As mentioned above, the performance of a model significantly depends on the importance of hyperparameters. Note that there is no way to know the best values for hyperparameters in advance, so ideally, we need to try all possible values to know the optimal ones. Doing this manually could take a considerable amount of time and resources and thus we use GridSearchCV to automate the tuning of hyperparameters.
3) In this method we are going to commbine knn hyperparameters tuning (second step) and the optimal random state(first step) to get high accuracy

The main goal of this part was to understand the impact of knn hyper parameters tuning. As a first step, we implement a model that tunes the optimal $randomstate$ without specifying knn parameters, after we evaluate the model and we get an accuracy of 78%,next we search to increase this accuracy by fitting the model with the best parameters for that we search at first for the best k using the gridsearchCV and after we fix the best value of k and search for The best metrics. Finally, we got the best model with k=17 and metrics=chebychev and a better accuracy of 79% as we can see the impact of hyper parameters wasn't so strong the important parameter is the randome state.

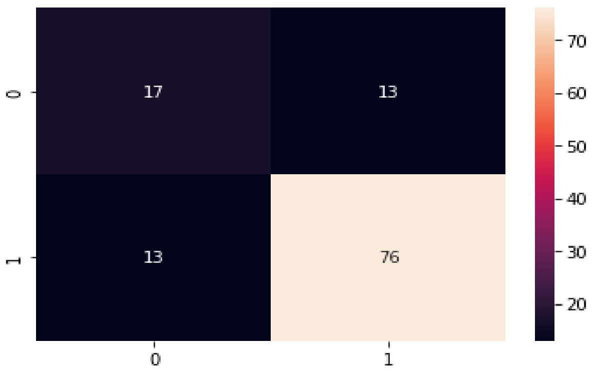

Fig. 19. visualize the confusion matrix KNN

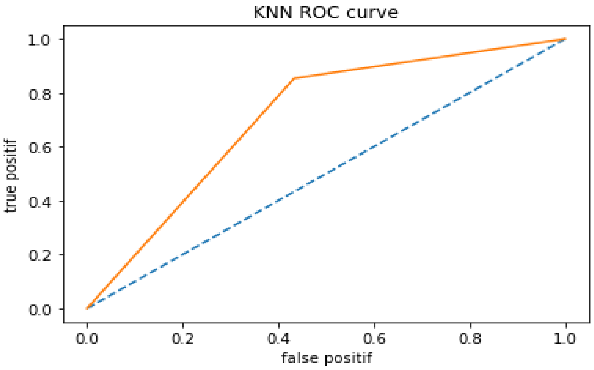

Fig. 20. visualize the ROC curve KNN

C. Result of the Support vector machine model

In machine learning, support-vector machines are supervised learning models with associated learning algorithms that analyze data for classification and regression analysis.

It uses a kernel trick technique to transform your data and then finds an optimal boundary between the possible outputs based on these transformations. We will use three kernels: Linear, polynomial and Gaussian kernel.

1) 1) Linear kernel: Linear Kernel is used when the data is Linearly separable, that is, it can be separated using a single Line. It is one of the most common kernels to

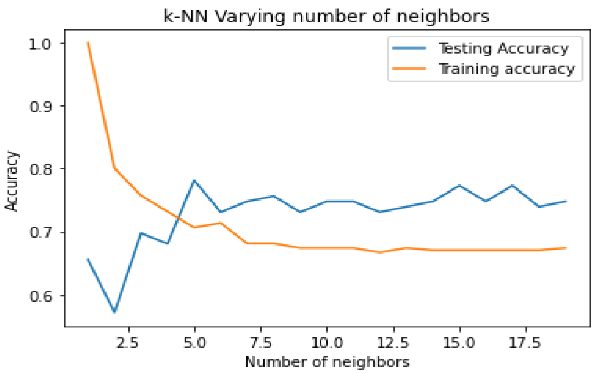

Fig. 21. improving model accuracy KNN

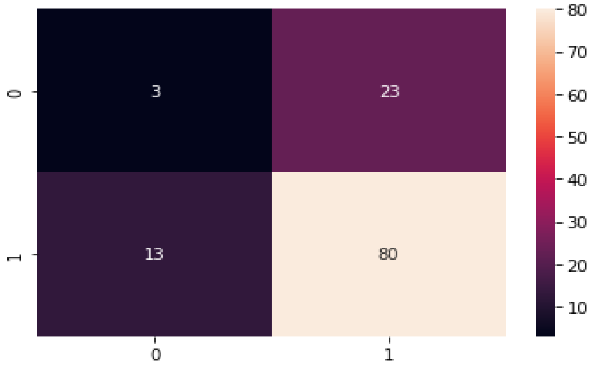

Fig. 22. improving model accuracy, the ROC curve KNN

be used. It is mostly used when there are many features in a particular Data Set.

2) the polynomial kernel is a kernel function commonly used with support vector machines and other kernelized models, that represents the similarity of vectors in a feature space over polynomials of the original variables, allowing learning of non-linear models.
3) Gaussian RBF(Radial Basis Function) is another popular Kernel method used in SVM models for more. RBF kernel is a function whose value depends on the distance from the origin or from some point.

Comparison of the three algorithms, The metrics that we will be used to compare those three algorithms are :

1) F1 score: say + 0.75 is good, we got 0.82 and it's a good value.
2) Accuracy score: well, it depends on your application, sometimes if you are working on a critical problem, say malignant tumour prediction, then 90 % isn't good enough, but for our problem +80 % is much acceptable.
3) Confusion matrix: The confusion matrix should be nearly diagonal which indicates the classifier does not miss the test examples($Y_{predicted} \approx Y_{test}$)
4) ROC curve: The curve should be way above the blue dashed line.
5) ROC score: Calculate the value of the surface under the ROC curve, an optimal value is 1, say + 0.75 is acceptable for our problem.

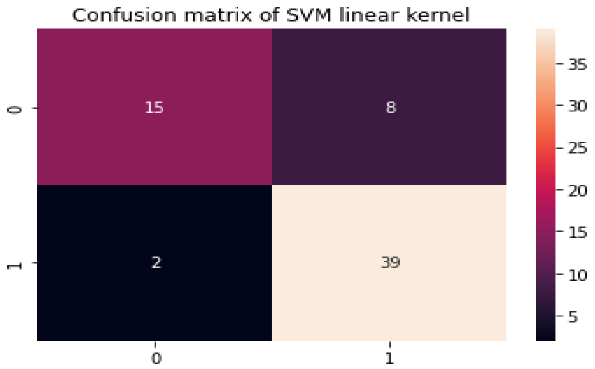

Fig. 23. visualize the confusion matrix of SVM

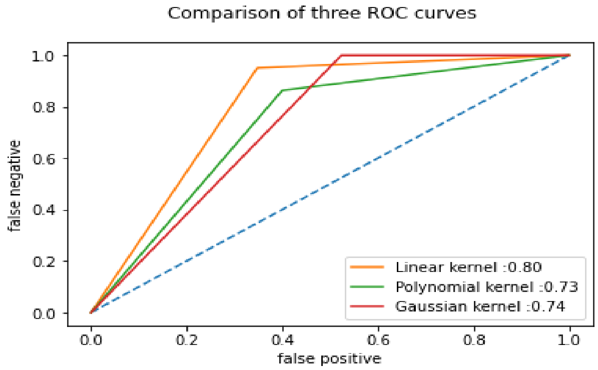

Fig. 24. Comparison of Linear kernel, polynomial kernel and gaussian kernel

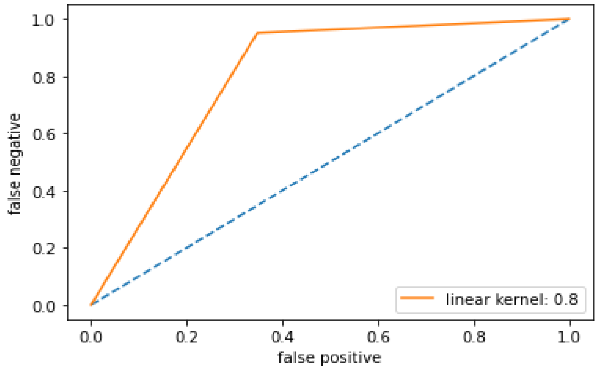

Fig. 25. The most accurate SVM kernel is the linear kernel

As you can see the classifier with high metrics is the support vector machine classifier with high accuracy of 84%, the confusion matrix is roughly diagonal which indicates that this classifier is able to label data correctly. If we see other metrics such as the f1 score, we managed to have a good value of 0.82 which means that we have low false positives and low false negatives.

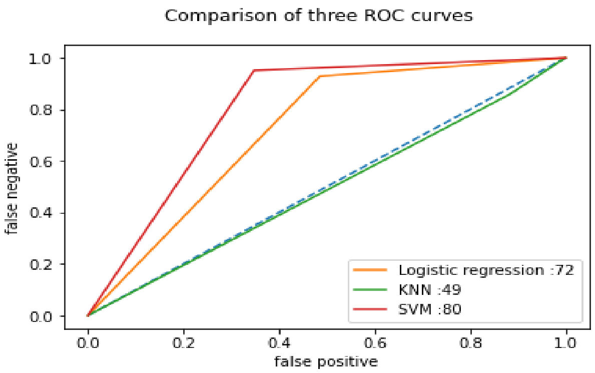

Fig. 26. Comparison of the three algorithms

V. CONCLUSIONS

In summary, the goal of this project was to build a model that could predict student academic status based on various features. To achieve this, we followed several steps including data processing, data visualization, and implementation of multiple classification algorithms. The team faced challenges in defining the best classification algorithm and identifying the most impactful factors for student academic status, but ultimately was able to build a model using the SVM algorithm that had an accuracy of 84%. and also identified a list of features that had a positive and negative impact on student success.

Following features have a positive impact on the student's success:

1) **Father's education** : if the father has a higher education, he will help his children in their studies so that they will not struggle for a long time with their homework.
2) **Guardian**: This instance takes three values as our convention: 0,1 and 2, 2 refers to 'other', we conclude that if the guardian is neither mother nor father then the student has a big chance to succeed, but this just the result of our classifier, it is difficult to judge that.
3) **Wants to take higher education**: Students who are looking forward to taking higher education seem to be motivated and have goals to achieve.
4) **Study time** : This is an important thing to keep in mind, students need to spend many hours studying, do not imagine a student succeeding in his exams and yet do not spend one hour at his desk, but it depends on many things such as subject, timetable.
5) **Father's job**: If the father has a promising career, then, of course, he will fulfil the needs of their children in terms of paying additive classes and internet.

The following are negative factors that affect student's success:

1) **Age**: It is difficult to judge that age is a negative factor, we do not have a big dataset to generalize, but we will assume it's a negative factor for the two chosen Portuguese schools, so students should go to high school early.
2) **Going out with friends**: going out with friends helps relieve stress, but sometimes if the students spend a lot of time outside the home this will definitely affect their studies.
3) **Absences**: Students who missed classes will find it difficult to take the exams, sometimes you find students read a one-month course in just one day, imagine how those students prepare for exams. Now, we did not talk about the students having problems letting them do that, we will later talk about that.
4) **Health**: This can not be taken into consideration, we cannot say that students having good health fail the exams, but we will assume again it's a negative factor for the two chosen Portuguese schools as our classifier told us.
5) **Failures**: Having a lot of failures is an indication of a lack of good exam preparations.

REFERENCES

[1] P. Cortez and A. M. G. Silva, "Using data mining to predict secondary school student performance," 2008.

[2] A. Acharya and D. Sinha, "Early prediction of students performance using machine learning techniques," *International Journal of Computer Applications*, vol. 107, no. 1, 2014.

[3] E. Er, "Identifying at-risk students using machine learning techniques: A case study with is 100," *International Journal of Machine Learning and Computing*, vol. 2, no. 4, p. 476, 2012.

[4] R. R. Halde, A. Deshpande, and A. Mahajan, "Psychology assisted prediction of academic performance using machine learning," in *2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, pp. 431–435, IEEE, 2016.

[5] M.-L. Zhang and Z.-H. Zhou, "Ml-knn: A lazy learning approach to multi-label learning," *Pattern recognition*, vol. 40, no. 7, pp. 2038–2048, 2007.

[6] E. Galy, C. Downey, and J. Johnson, "The effect of using e-learning tools in online and campus-based classrooms on student performance," *Journal of Information Technology Education: Research*, vol. 10, no. 1, pp. 209–230, 2011.

[7] H. Waheed, S.-U. Hassan, N. R. Aljohani, J. Hardman, S. Alelyani, and R. Nawaz, "Predicting academic performance of students from vle big data using deep learning models," *Computers in Human behavior*, vol. 104, p. 106189, 2020.

[8] S. Kotsiantis, C. Pierrakeas, and P. Pintelas, "Predicting students'performance in distance learning using machine learning techniques," *Applied Artificial Intelligence*, vol. 18, no. 5, pp. 411–426, 2004.

[9] R. E. Wright, "Logistic regression.," 1995.