Predicting Students Performance in Exams using Machine Learning Techniques

Khanyisile Sixhaxa
School of Computer Science and Applied Mathematics
The University of the Witwatersrand
Johannesburg, South Africa
1590202@students.wits.ac.za

Ashwini Jadhav
Science Teaching and Learning Unit
Faculty of Science
The University of the Witwatersrand
Johannesburg, South Africa
ashwini.jadhav@wits.ac.za

Ritesh Ajoodha
School of Computer Science and Applied Mathematics
The University of the Witwatersrand
Johannesburg, South Africa
ritesh.ajoodha@wits.ac.za

***Abstract*—Predicting students success has been a very popular study across different fields and with this study we will be focusing on how Machine Learning can aid us in giving us insight in how students will perform in their exams. In this paper, we propose a model that takes in Demographic, Academic and Behavioural attributes and investigate the how these attributes contribute a student's performance and also predict at risk students. For the analysis of the features we will use the Mutual Information algorithm, alongside five machine learning models that are a mixture of classification and regression classifiers. We will also use these models to predict our students performance. The five classifiers used in this study are as follows: Gaussian Naive Bayes, Support Vector Machine, Random Forest, K-Nearest Neighbour and Logistic Regression and we achieved prediction accuracy of 50.83%, 81.67%, 78.33%, 75.00% and 74.17% respectively. The results yielded revealed that there is a strong correlation between a student's behavioural characteristics and their academic performance.**

Keywords*— *Machine Learning, Student Prediction, Learning Management System, Classification, Behavioural Attributes

I. Introduction

Measuring of a students academic performance has been a topic of interest across various fields of study. This study proves itself to be a challenge due to the fact that students academic performance is reliant on various factors like behavioural, academic, demographics and other environmental factors. The purpose of this paper is to predict the students performance in exams and the factors that influence their performance.

Machine learning is a field in which there has been a success in being able to predict students performance, this is due to the fact that there are algorithms that have been developed to help achieve better accuracy on predictions and also being able to realize important attributes that play a critical role.

There has been many attempts to predict student performance using machine learning [1], [2]. We see that in [2], they investigate how behavioural attributes contribute to a student's performance in comparison to the demographic and academic attributes and in [1], we see that the authors attempt to use these behavioural attributes to identify at risk students. In this study, we will attempt to look deeper into which of the specific features within the demographic, academic and behavioural feature sets can help us predict a student's performance in exams.

The use of a definitive and dependable dataset is crucial for the prediction of students performance in exams. The ground breaking work of [2] resulted in the x-API Edu dataset, which is the dataset that will be utilized in this study. The xAPI Edu dataset was created by data mining information from the Kalboard 360 Learning Management System (LMS) [2]. Due the high dimensionality nature of our data set, it is inevitable that we will get low accuracy from our prediction models. To better our prediction accuracy, we will have to pre-process our data to reduce the bias and variance we get from our models. The contributions of this research are as follows; a model that takes in academic, demographic and behavioural attributes as parameters and predicts a students performance in exams and to provide an list of the top features that play a more significant role in a students performance in exams.

In the next section, we will provide a review of past literature that have contributed to the prediction of students performance in the recent years.

II. Related Work

To predict students performance in exams, we have to be able to identify which features play a significant role in a student's performance. In [2], it has been proven that using datasets with behavioural attributes improved prediction accuracy by 29% when compared to datasets that didn't have these attributes. In contrast to [2], the authors in [3] investigated the importance of demographic features on a students academic performance, thus they limited their dataset to demographic attributes and doesn't account for academic and behavioural attributes. In this study, we seek to find the top attributes that play a significant role across the demographic, academic and behavioural feature sets. We will make use of mutual information gain as in [3], to rank our features and extract the best feature set.

Previous studies have shown the reliability of machine learning classifiers for prediction models. In [2] we see the use of Decision Trees (DT), Artificial Neural Networks (ANN) and Naive Bayesian (NB) classifiers to investigate whether adding behavioural attributes to a dataset can improve the prediction of at risk students. The prediction accuracies obtained from these classifiers using data that has behavioural attributes are as follows: 61.3%, 73.8% and 72.5% for DT, ANN and NB respectively. Whereas, without the behavioural attributes the authors in [2] achieved prediction accuracies of 55.6%, 45.8% and 50.4% for DT, ANN and NB respectively, and we see that using the classifiers on the dataset that has behavioural attributes increased prediction accuracy. From the results obtained, we see that behavioural attributes play a significant role in a students academic performance.

The authors in [1] made use Linear Discriminant Analysis (LDA), Logistic Regression (LR), Classification and Regression Trees (CART), K-Nearest Neighbour (KNN), Naive Bayesian (NB) and Support Vector Machines (SVM) as their classifiers for their model to predict at risk students. Using xAPI dataset, which is the same dataset utilized in [2], we see that the the prediction accuracy achieved in this study is on par with that obtained in [4] as they used the similar classifiers on the same dataset. The prediction accuracies obtained in [1] are as follows: 84%, 86%, 86%, 81%, 82% and 72% for LDA, LR, CART, KNN, NB and SVM respectively. From the results obtained, it was concluded that students behavioural attributes are useful predictors of their academic success.

In [4], the authors made use of DT, KNN and SVM ensemble classifiers in the model for enhancing existing student prediction models. Using the same dataset as in [1], [2], the authors in [4] managed to improve prediction accuracies of existing student performance prediction models and achieved high accuracies of 91%, 85% and 88% for DT, KNN and SVM respectively, which is a significant improvement in comparison to the prediction accuracy achieved in [1], [2]. These results also suggest that a students behavioural attitude towards their academics has a direct correlation to their academic success.

Our literature review has shown us that the xAPI Edu dataset has been used extensively in the prediction of a students academic performance and that behavioural attributes play a significant role in how a student performs. We can also see that machine learning classifiers are a reliable and efficient way of predicting students academic success. In the next section, we will look into the methodology employed for our student performance prediction model.

III. Methodology

In this section we will introduce a student's performance prediction model through the use of machine learning classifiers. The main steps in our methodology are as follows: we start by obtaining the xAPI data from Kaggle. After, we look into data pre-processing which includes: removing missing values, encoding categorical data, feature scaling and feature selection. Once we have pre-processed our data, we will move onto training our models using the train data and predict students performance in exams using the test data. We will also provide an analysis of our models using a variety of performance metrics.

A. Data Source

In this study we used the Students' Academic Performance data set (xAPI-Edu-Data) obtained from Kaggle. This is an educational dataset collected from the Kalboard 360 which is an online Learning Management System (LMS) used across the world. This system serves as an online educational platform that enables educators to share resources, make announcements, hand out assignments, etc. and for students to access the given resources, view the announcements and assignments, etc. We are able to obtain information about students using a monitoring component of the LMS called the experience API (xAPI). This API tracks a students behaviour and actions, from viewing announcements to reading of available resources.

In this xAPI-Edu-Data, we have 480 data instances with 16 features. The features are partitioned into three categories: Demographic, Academic and Behavioural feature sets. Our target variable is the Class feature, where the students marks can be classified into three categories:

- **Low:** students who achieve 0-69
- **Medium:** students who achieve 70-89
- **High:** students who achieve 90-100

It is worthy to note that our data only has 4 features which are numerical and the rest are categorical. Let us have a look into the given features according to their categories in Table I.

B. Data Pre-processing

In order for us to use the data, we need the remove the noise in the data so that our models can work efficiently. The term "noise" implies the instances in our data that could lead to distorted results and leaves room for increased error margins. To stabilize our data, we will pre-process our data, this is an extremely crucial step for Machine Learning due to the fact that real-world data consists of incomplete, inaccurate and

TABLE I
Initial feature set

No.	**Demographic Features**
1	Gender
2	Nationality
3	Place of Birth
	Academic Features
4	Educational Stages
5	Grade Levels
6	Section ID
7	Topic
8	Semester
9	Parent responsible for student
	Behaivoural Features
10	Raised Hands
11	Visited Resources
12	Viewing Announcements
13	Discussion Groups
14	Parent Answering Survey
15	Parent School Satisfaction
16	Student Absence Days

inconsistent data instances. In this study, to process our data we perform the following steps:

1) Missing values: It is imperative that we identify and handle missing values in our data, failure to do so can lead to inaccurate results and and we might reach faulty conclusions which could be detrimental to our study. In the case of our data, we found no missing values. Hypothetically, in the event that we find missing values in our data, we would have either delete the particular row or calculate the mean of the missing feature and assign it to the missing value. It is worthy to note that the latter method only applies to numerical data.

2) Encoding the categorical data: Categorical data is a collection of data that is divided into groups, i.e. Gender is represented by Female or Male. In our data set, we have 12 features which are categorical and we need to assign numerical values to these features due to the fact that most Machine Learning algorithms are based on mathematical equations, making them better suited for numerical values.

In this study, we will be encoding our categorical features using One-Hot-Encoding. With One-Hot-Encoding, we remove the categorical variable and assign a new binary value to the unique categorical value, i.e. for Gender, we split it into two features, Gender_0 and Gender_1, where 1 is assigned for every instance where Female appears for Gender_0 and 0 is assigned to all instances that are Male, this also also applies to Gender_1.

We used One-Hot-Encoding over Label Encoder due to the fact that Machine Learning algorithms treat the order of numbers as an attribute of significance, meaning that higher numbers hold more value over lower numbers, thus increasing the bias of the algorithm.

Once we have encoded our categorical features in our data set, we increased our features from 16 to 78.

3) Feature scaling: Some Machine Learning algorithms are sensitive to varying degrees of magnitude, units and range. This is where feature scaling is introduced. Feature scaling is a method used to normalize the varying independent features of a data set and it is very crucial that we perform this step as it is helps us reduce the over-fitting and under-fitting of our Machine Learning algorithms.

You can perform feature scaling in two ways:

- **Normalization:** Normalization is a scaling technique where the data points are shifted and re-scaled in a way that they are bounded in the [0,1] range. The formula for normalization is as follows:

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}}$$

where X_{max} and X_{min} represent the maximum and minimum values of the feature respectively.

- **Standardization:** Standardization is a scaling technique where the data points are centered around the mean with a standard deviation of 1. The formula for standardization is as follows:

$$X' = \frac{X - \mu}{\sigma}$$

where the mean and standard deviation of the feature values is represented by μ and σ respectively.

In this study, we will use Standardization for scaling our features.

4) Splitting the dataset: For every Machine Learning algorithm, we need to split the data into a training and testing data.The training data set refers to the subset of the original data that is used to train the Machine Learning model and the testing data set is the subset of the original data that is used to predict the outcomes. Due to our data set being so small, we used a 75:25 training and testing split.

C. Models

In this section we will provide a high level overview of the Machine Learning (ML) models used in this study. Due to the small size of our data set, we used 10-fold Cross-Validation when training models. The procedure uses a parameter called k that refers to the number of groups that the training data sample is to be split into, and for this study we chose k=10.

1) Gaussian Naive Bayes: The Gaussian Naive Bayes classifier is a supervised ML algorithm, that is derived from the Naive Bayes that follows the Gaussian normal distribution for continuous data variables. The Naive Bayes classifiers are based on the Bayesian Theorem [5], which is used to calculate conditional probability, and the formula for the Bayesian Theorem is given as follows:

$$P(A|B) = \frac{P(A) * P(B|A)}{P(B)}$$

where:

- P(A) = Probability of A occurring
- P(B) = Probability of B occurring
- P(B|A) = Probability of B occurring given A
- P(A|B) = Probability of A occurring given B

For continuous data, Gaussian Naive Bayes makes an assumption that the continuous variable from each feature is distributed along the Gaussian Distribution. Gaussian Naive Bayes uses the following equation to calculate the likelihood of a data point belonging to a certain feature [1]:

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} e^{(-\frac{(x_i-\mu_y)^2}{2\sigma_y^2})}$$

2) Support Vector Machine: Support Vector Machine (SVM) is a Supervised ML algorithm that is mostly used for classification and regression problems. This algorithm plots all data points in the n-dimensional space (n being the number of features we have in our data set). Then it finds a n-dimensional hyper-plane that separates all the features into distinct classes. It does this using the kernel trick, which basically transforms low dimensional data points to higher dimensional spaces. To determine the decision boundary and hyper-plane shape, we used the Radial Basis Function (RBF) kernel, given by the equation:

$$K(x, x') = exp(-\frac{(||x - x'||^2)}{2\gamma^2})$$

where x,x' are two points and γ is the variance and hyper-parameter.

3) Random Forest: Random Forest is a Supervised ML algorithm that is mostly used for classification and regression problems. This algorithm creates different training subsets from the given training data (this is known as Bagging), then it takes the decision tree with the majority vote for classification problems.
What distinguishes Random Forest from most ML algorithms is the fact that it can handle continuous and discrete data sets, for regression and categorical problems, although it performs better for the latter.

4) Logistic Regression: Logistic regression is a classification model, derived by transforming the linear regression cost function using the sigmoid cost function ([6]). The most common logistic regression models are binary, multi-nomial and ordinal logistic regression. In this study we will focus on multi-nomial logistic regression,which is used when there are more than two possible discrete outcomes.
In machine learning, to predict which class a data point belongs to, we set a decision boundary (threshold) and obtain the probability of the the data set falling into the said class using the sigmoid cost function, given by the equation:

$$h\theta(x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x)}}$$

which returns a probability score in the range of [0,1]. If $h\theta$ is greater or equal to the threshold, we can deduce that the data point belongs to the feature.
The cost function of a linear regression model is given as:

$$J(\theta) = \frac{1}{2}\sum_{i=1}^{m}(h_\theta(x^i - y^i)^2)$$

to get an equation for Logistic Regression, we make use of the sigmoid cost function to transform the linear regression cost function, and we get:

$$J(\theta) = \frac{1}{m}\sum[y^i log(h\theta(x(i))) + (1 - y^i)log(1 - h\theta(x(i)))]$$

Then we reduce our cost function using the gradient descent.

5) K-Nearest Neighbour: K-Nearest Neighbor is a supervised ML algorithm used for classification problems. This classifier is commonly based on the Euclidean or Manhattan distance between two points, x = (a,b) which is taken from the test sample and y = (c,d) which is taken from the training sample. For this study we utilized the Manhattan distance, which is given by the equation:

$$distance(x, y) = |a - c| + |b - d|$$

Using this distance metric the, KNN algorithm calculates the distance between all other points in the data sets against the one we are trying to classify (e.g point x) and assumes that x belongs to the class of the other point that has the smallest distance.

D. Evaluation Metrics

In order to measure the efficiency of the classifiers used in this study, we will use a variety of performance metrics such as accuracy, confusion matrix, recall and precision after performing K-fold cross-validation.

- **Confusion Matrix:** The summary of all prediction results obtained from a classification problem
- **Accuracy:** The ratio of the total number of correct predictions and the total number of predictions
- **Recall:** The measure of our model correctly identifying true positives
- **Precision:** The ratio between the true positives and all the positives
- **F1 Score:** Measures the balance between Precision and Recall, and also if there is class imbalance

E. Feature Selection

When trying to identify trends, each feature in our data set represents a pattern [7]. In this section, our aim is to choose features that will allows us to deduce patterns in our data and we do this by removing features that have low variance.To obtain our optimal feature set, we used a dimensionality reduction technique called feature selection. Feature selection methods can be categorised by filter, wrapper and embedded methods [8]. In this study we will focus on filter methods, particularly mutual information [3], [9].

1) Mutual Information: Mutual Information is a filter method that calculates the reduction in entropy by calculating the information gain of each independent feature against the dependent feature and selecting the feature that has the maximum information gain. The mutual information between features X and Y is calculated using the following equation:

$$I(X, Y) = H(X) - H(X|Y)$$

Where I(X,Y) is the mutual information for X and Y, H(X) is the entropy for X and H(X|Y) is conditional entropy for X given Y [10], [11].

Mutual information is always within the [0,1] range, and the larger the value the stronger correlation between the two features. If the calculated result is given as zero, then X is independent of Y [11], [12]. After performing information gain on our encoded feature set, we kept 23 features and removed 49 features. The results of feature selection using information gain ranking are shown in TABLE II. Figure 1 shows the accuracy of our information gain filtering method, which achieved 82% accuracy.

IV. RESULTS AND DISCUSSION

In this section we will present the results we got from training our machine learning models after applying mutual information to our feature set. Thereafter, we will discuss the performance of each model by using accuracy as our numeric metrics and confusion matrix as our visual metric.

Figures 2 - 5, TABLE III highlights the performance of the used classifier and TABLE IV shows the parameters used for each classifier. Figure 6 demonstrates the confusion matrix

TABLE II
LIST OF FEATURES AFTER FEATURE SELECTION

No.	Feature	Entropy
1	Visited Resource	0.3675
2	Raised Hands	0.3218
3	Student Absent Days: Under 7	0.3003
4	Student Absent Days: Over 7	0.2906
5	Grade ID: G-09	0.1964
6	Grade ID: G-07	0.1522
7	Place of Birth: Lybia	0.1319
8	Grade ID: G-04	0.1255
9	Nationality: Lybia	0.1204
10	Student Absence Days	0.1107
11	Place of Birth: Iraq	0.1025
12	Grade ID: G-07	0.0952
13	Discussion	0.0792
14	Place of Birth: Syria	0.0697
15	Grade ID: G-08	0.0688
16	Relation: Mum	0.0676
17	Parent Answering Survey: Yes	0.0644
18	Section ID: A	0.0581
19	Nationality: Jordan	0.0564
20	Topic: Arabic	0.0562
21	Nationality: Syria	0.0473
22	Place of Birth: Tunis	0.0436
23	Place of Birth: Jordan	0.0434

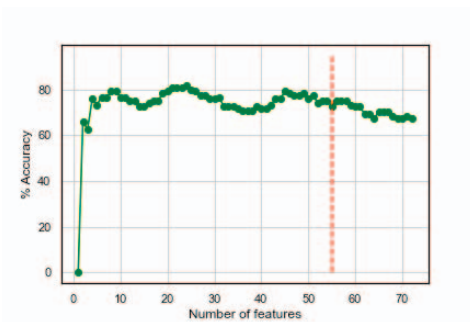

Fig. 1. Accuracy of Mutual Information for the Feature Selection Process

for the Gaussian Naive Bayes model which achieved 50.83% prediction accuracy using 10-fold cross-validation , which is the worst classification accuracy achieved compared to the other four models used in this paper. With the exclusion of KNN, this model has the slowest build time compared to SVM, Logistic Regression, Gaussian Naive Bayes and Random Forest.

Figure 3 demonstrates the confusion-matrix for the Logistic Regression model which achieved 74.17% prediction accuracy using 10-fold cross-validation. With the exclusion of the KNN, SVM and Gaussian Naive Bayes, this model has the slowest build time in comparison to Random Forest.

Figure 4 demonstrates the confusion-matrix for the KNN model, which achieved 75% prediction accuracy using 10-fold cross-validation. Furthermore, compared to the other four models used in this paper this model had the fastest build time.

Figure 2 demonstrates the confusion-matrix for the Random Forest classification model, which achieved 78.33% prediction accuracy using 10-fold cross-validation. This model has the slowest build time in comparison to all the other classification models used in this paper.

TABLE III
EVALUATION OF CLASSIFIERS

Classifier	Accuracy	Recall	Precision
Random Forest	78.33%	75.33%	77.94%
K-Nearest Neighbour	75.00%	75.78%	77.63%
Support Vector Machine	81.67%	81.43%	83.72%
Logistic Regression	74.17%	74.56%	75.42%
Gaussian Naive Bayes	50.83%	60.56%	59.10%

TABLE IV
PARAMETERS USED FOR OUR DIFFERENT CLASSIFIERS

Classifier	Parameters
Random Forest	*criterion:* gini,*n estimators:* 80
K-Nearest Neighbour	*n neighbour:* 5, *weight:* distance, *metric:* manhattan
Support Vector Machine	*kernel:* rbf, *gamma:* Scale, *C:* 10, *Class Weight:* Balanced
Logistic Regression	*penalty:* L2, *class weight:* None
Gaussian Naive Bayes	*var smoothing:* $1e^{-9}$

Figure 5 demonstrates the confusion-matrix for the Support Vector Machine model,which achieved 81.67% prediction accuracy and is the best performing model compared to the other four models used in this paper. Furthermore, compared to the other four models used in this paper this model had the slowest build time.

Given that Support Vector Machine is our best performing, we used it in our feature selection process. TABLE II shows the set of features with variance in descending order [1]. We see that the top 3 features fall under the behavioural features [2], [1]. The top three features are:

- Visited Resources: How frequently did the student utilize the provided resources
- Raised Hands: The amount of times the student has raised their hands in class to ask questions or interact with their educator
- Student Absent Days: The number of days the student has been absent from school

This shows that a student's behavioural attitude towards their academics has a high impact on their performance, which proves that there is a strong correlation between a student's academic performance and their behaviour.

V. CONCLUSION

In this paper, prediction of a students performance was performed by firstly pre-processing our data and extracting features that have more variance in comparison to the other features. Thereafter, we performed dimensionality reduction using information gain to extract features that have the most variance to better our models training and testing accuracy. From the five off the shelf classifiers used, Support Vector Machine was the best performing classifier, achieving accuracy of 81.67%. The results achieved in this paper are on an equal footing with the those obtained in [1], [3].

Actual \ Predicted	High	Low	Medium
High	17	0	11
Low	0	30	4
Medium	4	4	50

Fig. 2. Confusion Matrix for Random Forest Model.

Actual \ Predicted	High	Low	Medium
High	19	0	9
Low	0	30	4
Medium	10	5	43

Fig. 3. Confusion Matrix for Logistic Regression Model.

Actual \ Predicted	High	Low	Medium
High	13	0	15
Low	0	31	3
Medium	7	8	43

Fig. 4. Confusion Matrix for K-Nearest Neighbour Model.

Actual \ Predicted	High	Low	Medium
High	20	0	8
Low	0	31	3
Medium	7	4	47

Fig. 5. Confusion Matrix for Support Vector Machine.

Evaluation of our optimal feature set has shown us that top three features that are useful predictors are: Visited Resources, Raised Hands and Student Absent Days. The analysis on the top features offers a conclusion that there is a high correlation between students academic performance and their behavioural attitude, thus proving of

One of the limiting factors of this study is the sample size of the our dataset, the nature of this study needs a larger dataset due to the fact that a small dataset makes it harder for us to identify significant relationships within our data. A larger dataset could have generated greater prediction accuracy, as we would have a larger sample space to train and test our models.

The other limiting factor was that there is an imbalance in our target variable "Class", we found 43.95% of the data set is comprised of students that fall under the "Medium" class, 29.58% of students that fall under the "High" class and 26.45% for students that fall under the "Low" class. This explains why we have more misclassified students in relation to correctly classified students in our confusion matrices, this is because most standard classifiers assume that the classes of our target variable are balanced and in turn this compromises the models quality of the fit over the tradeoff for accuracy. In future, we need to make sure that there is minimal class imbalance in our dataset.

Acknowldgment

This work is based on the research supported in part by the National Research Foundation of South Africa (Grant numbers: 121835).

References

[1] E. Buraimoh, R. Ajoodha, and K. Padayache, "Prediction of student success using student engagement with learning management system," IRTM, 2021.

[2] E. A. Amrieh, T. Hamtini, and I. Aljarah, "Mining eductional data to predict student's academic performance using ensemble methods," *International Journal of Database Theory and Application*, vol. 9, no. 8, pp. 119–136, 2016.

[3] B. Trstenjak and D. Donko, "Determining the impact of demographic features in predicting student success in croatia," IEEE, 2014.

[4] S.-S. M. Ajibade, N. B. B. Ahmad, and S. M. Shamsuddin, "Educational data mining: Enhancement of student performance model using ensemble methods," IOP Publishing, 2019.

[5] S. Yildirim, "11 most common machine learning algorithms explained in a nitshell," July 2020.

[6] R. Buyya, R. N. Calheiros, and A. V. Dastjerdi, *Big Data Principles and Paradigms*. Morgan Kaufmann, 2016.

[7] P. A. Estevez, M. Tesmer, C. A. Perez, and J. M. Zurada, "Normalized mutual information feature selection," *IEEE Transactions On Neural Networks*, vol. 20, no. 2, 2009.

[8] N. Ibrahim, S. Fong, and S. A. Rahman, "Featuree selection methods: Case of filter and wrapper approaches for maximising classification accuracy," *Pertanika Journal of Science and Technology*, vol. 26, no. 1, 2018.

[9] R. Ajoodha, A. Jadhav, and S. Dukhan, "Forecasting learner attrition for students' success at a south african university," SAICSIT, 2020.

[10] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, 4 ed., 2016.

[11] J. Brownlee, "Information gain and mutual information for machine learning," October 2019.

[12] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

Actual \ Predicted	High	Low	Medium
High	25	0	3
Low	1	33	0
Medium	38	17	3

Fig. 6. Confusion Matrix for Gaussian Naive Bayes.