Digital Object Identifier

Machine Learning Approaches to Ethical Analysis of Statistics

MATTHEW KENELY, KARL CHETCUTI, JAN FORMOSA, and DAVID CACHIA ENRIQUEZ

I. INTRODUCTION

The dataset selected for this study is the Student Performance[1] dataset from Kaggle. This dataset provides comprehensive statistical information on a range of academic, behavioural, and demographic variables influencing students' performance.

The data was collected from two Portuguese schools and includes variables such as age, gender, family size, and grades in mathematics and Portuguese, among others. It is particularly well-suited for analysing relationships between educational and demographic factors to determine potential correlations that contribute to students' academic performance.

To analyse the dataset, four machine learning (ML) techniques are applied to classify and predict students' levels of academic performance. The techniques employed are:

- Linear Regression
- Ensemble Learning
- K-means Clustering
- Principal Component Analysis

The GitHub repository for this study is available at https://github.com/matthewkenely/ics5110

II. BACKGROUND

A. MACHINE LEARNING TECHNIQUES

1) Linear Regression

Linear Regression is a supervised machine learning algorithm used to predict a continuous numerical value. For each input, the algorithm estimates a corresponding output value that lies on a continuous scale [1].

Linear Regression models the relationship between inputs and outputs in a linear manner, where the predicted output is expressed as a linear combination of the input features and their corresponding weights, plus a bias term. There are two primary types of linear regression.

Simple Linear Regression (SLR) involves a single independent variable and a single dependent variable [1]. The equation for SLR is:

$$y = \beta_0 + \beta_1 X \quad (1)$$

where:

- y is the dependent variable,
- X is the independent variable,
- β_0 is the intercept,
- β_1 is the slope.

Multiple Linear Regression (MLR) involves multiple independent variables and one dependent variable [2]. The equation for MLR is:

$$y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n \quad (2)$$

where:

- y is the dependent variable,
- $X_1, X_2, \ldots, X_n$ are the independent variables,
- β_0 is the intercept,
- $\beta_1, \beta_2, \ldots, \beta_n$ are the slopes.

The goal of linear regression is to determine the best-fit line for a model. This line is drawn to best represent the data points, serving as a predictor for the output based on a given input. To minimize the differences between the predicted and actual values, the best-fit line is positioned as close as possible to all the data points.

The cost function is used to assess how well the model's predictions match the actual data. Rather than simply indicating whether a prediction was correct or incorrect, the cost function calculates the magnitude of the error. If the prediction is close to the actual value, the cost is small; if the prediction deviates, the cost is larger. The objective of linear regression is to adjust the weights of each feature to minimise the total cost across all predictions, thereby improving the model's accuracy.

2) Ensemble Learning

Ensemble Learning is a machine learning technique that employs a multi-model approach to generate predictions. Ensemble learning algorithms train multiple smaller models, each making its own prediction. These predictions are then aggregated to produce the final output. The rationale behind this approach is that combining the results from multiple learners can help overcome the errors that may be present in individual models, thereby leading to more accurate and reliable predictions [3].

There are various approaches to Ensemble Learning, all of which involve the use of multiple learners but differ in their training processes and decision-making mechanisms. These approaches include:

- **Bagging (Bootstrap Aggregating)**: Trains multiple learners in parallel on different subsets of the input data.

[1] https://www.kaggle.com/datasets/larsen0966/student-performance-data-set

These learners operate independently, without sharing information. At the end of the training process, their predictions are consolidated. Depending on the task, this consolidation is typically done through majority voting (for classification) or averaging (for regression) [4].

- **Boosting**: Trains multiple learners sequentially, with each learner using the results of the previous one. The output of each learner is passed on to the next, which attempts to correct the errors made by its predecessor. This iterative process continues for a set number of iterations. The final prediction is a weighted average of all individual learner predictions [5].
- **Stacking**: Involves training multiple base models in parallel. These models can vary in type and are typically chosen for their complementary strengths. The final prediction is made by a meta-model, which is trained to predict based on the outputs of all the base models [3].
- **Blending**: Similar to stacking, blending also uses multiple base models and a final meta-model to make predictions. The key distinction is that the meta-model in blending is trained on a separate *holdout* dataset or aggregates predictions using a weighted average [6].
- **Voting**: Uses multiple learners to generate predictions but differs in how the results are combined. This aggregation can take two forms:
    - **Hard Voting**: The final prediction is made by taking the majority vote from the learners.
    - **Soft Voting**: The prediction probabilities are averaged, and the class with the highest probability is selected.

For this study, we employ three Ensemble Learning techniques: Bagging (using Random Forest), Boosting (using Gradient Boosting), and Stacking (combining Random Forest and Gradient Boosting as base models, with Logistic Regression as the meta-model).

3) K-means Clustering

K-Means clustering is an unsupervised machine learning algorithm designed to partition data into a specified number, k, of clusters based on feature similarity. It is commonly used for segmentation tasks and works by minimizing the distance between data points and their assigned centroids, which serve as the central points of the clusters.

The algorithm begins by randomly initializing centroids within the feature space of the dataset. The distance between each data point and the centroids is then calculated using the Euclidean distance, defined by the following equation:

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(y_i - x_i)^2}$$

Based on this measurement, each data point is assigned to the cluster of its nearest centroid.

After all data points are assigned to clusters, the centroids are recalculated as the average position of all the data points within their respective clusters. This process is repeated iteratively until the centroids stabilize, or a predefined stopping condition is met.

4) Principal Component Analysis

Principal Component Analysis (PCA) is a dimensionality reduction technique that converts data into a new coordinate system while maintaining important patterns and trends.

PCA can be broken down into five main steps:

1) Data Standardisation
2) Covariance Matrix Computation
3) Eigendecomposition
4) Component Selection
5) Data Transformation

First, the data is standardised so that the variables have a mean of 0 and a standard deviation of 1. This critical step ensures that there is no large difference in scale between the initial features, as such differences would bias the analysis. The standardisation process can be represented as:

$$z = \frac{x - \mu}{\sigma}$$

where z is the standardised value, x is the original value, μ is the mean, and σ is the standard deviation.

Secondly, the covariance matrix of the standardised data is computed. The covariance matrix highlights the relationships between pairs of variables in terms of their variance. It is calculated as:

$$\mathrm{Cov}(X) = \frac{1}{n}X^T X$$

The third step is Eigendecomposition, which is performed on the covariance matrix to extract the eigenvectors and eigenvalues. Eigenvectors define the directions of principal components, while eigenvalues indicate the amount of variance captured by each principal component.

After Eigendecomposition, the eigenvectors are sorted based on their eigenvalues in descending order. This ensures that the first eigenvector corresponds to the direction of maximum variance in the data. k principal components are then selected depending on how much of the total variance should be retained. The explained variance ratio guides this selection:

$$\text{explained variance} = \frac{\sum_{i=1}^{k} \text{eigenvalue}_i}{\sum_{i=1}^{p} \text{eigenvalue}_i}$$

Finally, the selected principal components are used to create a projection matrix. The original data is then multiplied by this projection matrix to transform the data:

$$Z = X \cdot P$$

where Z is the transformed data, X is the original data, and P is the projection matrix.

B. RESCALING AND NORMALISATION

Before processing data, it is essential to standardise the distribution and range of different variables, particularly when they have different units or scales. Rescaling is the process used to achieve this standardization, which involves adjusting the data to a specific range, such as from 1 to 100. This helps ensure that the model interprets the variables on a comparable scale, improving its performance [7].

Normalisation, on the other hand, involves transforming the data to follow a standard distribution, such as a Gaussian distribution with a mean of 0 and a standard deviation of 1 [8]. This transformation is crucial for models like Support Vector Machines (SVM) and Logistic Regression, as it improves both the accuracy and convergence speed during training.

C. CROSS-VALIDATION

Cross-validation is a technique used to evaluate a model's generalisability – its ability to perform well on unseen data, i.e., data not included in the training set. By splitting the dataset into multiple subsets and using different combinations of these subsets for training and testing, cross-validation provides an estimate of the model's performance on new, unseen data [9]. This technique helps prevent overfitting, ensuring that the model's performance is robust and reliable across various data scenarios.

D. DIMENSIONALITY REDUCTION AND FEATURE SELECTION

Dimensionality Reduction and Feature Selection are techniques used to simplify datasets by removing irrelevant or redundant features. By focusing on the most relevant features, these methods can enhance a model's performance.

Dimensionality Reduction techniques reduce the number of features (dimensions) in a dataset while retaining as much important information as possible. These techniques help transform the data into a lower-dimensional space without losing key information, which can reduce model complexity and potentially improve generalization performance [10]. Principal Component Analysis (PCA), used in this study, is one of the most common dimensionality reduction methods.

Feature Selection, by contrast, does not reduce the number of dimensions but rather removes irrelevant or redundant features from the dataset. This process allows the model to focus only on the most significant variables, improving efficiency and potentially boosting model accuracy.

E. QUANTITATIVE MEASURES

1) Accuracy

Accuracy is one of the simplest and most intuitive performance measures for evaluating machine learning models. It quantifies the proportion of correctly predicted instances relative to the total number of instances. It can be expressed as:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} \tag{3}$$

2) PCA Visualisation

Principal Component Analysis (PCA) is commonly used for dimensionality reduction, transforming high-dimensional data into a lower-dimensional space while preserving its essential patterns. PCA visualisation refers to the process of projecting the reduced data onto a 2D or 3D space to facilitate easier interpretation and exploration of the underlying structure of the data.

By plotting the first two or three principal components, which capture the most variance in the data, PCA visualisation allows for the identification of clusters, outliers, and patterns that may not be apparent in higher dimensions. This technique is particularly useful for understanding the distribution of data points and the relationships between variables, making it a valuable tool in exploratory data analysis.

III. DATA PREPARATION

For this study, we selected the Student Performance dataset for our experiments. This dataset contains information collected from two Portuguese schools, focusing on student performance in two subjects: Mathematics and Portuguese. As a result, the dataset is divided into two sections, each corresponding to one of the subjects.

For the purposes of this study, we chose to work with only the Portuguese section of the dataset, as it contained a larger volume of data, which better suited the requirements of our analysis.

A. DATA CLEANING AND TRANSFORMATION

Data cleaning and transformation are crucial steps in data preprocessing, aimed at improving the quality, usability, and efficiency of the data for analysis or application.

Data cleaning involves identifying and correcting errors, inconsistencies, missing values, or irrelevant data within a dataset. This may include tasks such as removing duplicates, standardizing formats, or imputing missing values using techniques like interpolation.

Data transformation, on the other hand, focuses on converting the data into a more suitable or meaningful format for analysis. Common operations include scaling numerical values, encoding categorical variables, aggregating data, or restructuring the dataset to meet specific algorithmic or workflow requirements.

In this study, no single cleaning or transformation method was applied universally across the dataset. Instead, these processes were tailored individually to each experiment, as the handling of the data varied depending on the specific needs of the experiment. Examples of the techniques applied include:

- **Standardization**: A feature-scaling technique that transforms all values in the dataset to a common range, typically between 0 and 1.
- **Feature Selection**: The process of determining which features to include or exclude based on their relevance to the analysis. The approach and rationale for feature selection varied across experiments.

- **Label Encoding**: A method for converting categorical data into numerical format by assigning a unique numerical label to each category.

These techniques will be discussed in greater detail in the following sections, where each experiment is explored comprehensively.

B. DISTRIBUTION ANALYSIS

Distribution analysis involves examining how the data points in a dataset are spread. This analysis is crucial for identifying patterns and trends, as well as detecting outliers that may affect the results of experiments. Visualizations such as histograms or box plots are commonly used to facilitate a better understanding of the data distribution. By conducting distribution analysis, we can derive insights that guide the direction of an experiment and ensure that each experiment is sufficiently distinct from others.

C. FEATURE ANALYSIS

The initial examination of the dataset involved performing feature analysis on each feature, which was visualized using histograms. This analysis revealed the distribution of each feature and helped determine whether it would be worthwhile to train a machine learning model to predict that specific feature.

One of the ideas discussed early on was to train a model to predict the number of past failures without using any grade-related features. Although this idea seemed promising, the feature analysis showed a significant imbalance in the 'failures' feature. The experiment confirmed this, as the model learned to predict '0' for nearly all data points due to the fact that over 80

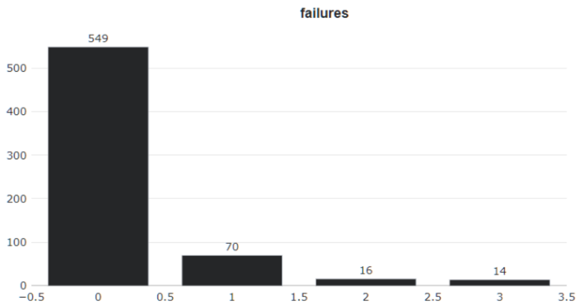

FIGURE 1. Feature distribution of the "failures" feature, demonstrating data imbalance.

This issue of imbalance was not limited to the 'failures' feature. Other features, such as "Pstatus", "schoolsup", "paid", and "Dalc", also exhibited imbalances. Despite these imbalances, most of the other features had distributions that were much more suitable for training machine learning models.

D. CORRELATION MATRIX

A second important analysis involved creating a correlation matrix to examine the relationships between the features in the dataset. The correlation matrix allows for a clear visualization of pairwise correlations, helping identify relationships and patterns between different features. This tool provided valuable insights that guided several experimental decisions.

One key finding from the correlation matrix was the high correlation between the three grade features: G1, G2, and G3. These features had an average correlation score of 87%, suggesting that they were strongly related. This observation aligns with logical reasoning: if a student performs well in one or two of the grades, it is likely that the third grade will also reflect similar performance. The same applies to poor grades, where all three features tend to show similar patterns. Consequently, these features can act as proxies for each other, meaning that when predicting one of them, the other two will have a significant impact on the model's decision.

Further instances of how correlations influenced the experimental setup will be discussed in later sections, providing additional context for these relationships.

IV. EXPERIMENTS

For this study, four distinct experiments were conducted on the dataset, each aimed at exploring different aspects of the data through various machine learning techniques.

An overview of the experiments is as follows:

1) **General Final Grade Prediction**: investigates training a model to predict a student's final grade using an optimal number of features, constrained by specific correlation bounds.
2) **Tailored Final Grade Prediction**: examines the use of different feature subgroups to predict the final grade.
3) **Dimensionality Reduction**: explores how reducing the number of features affects the accuracy of the model predicting the final grade.
4) **Student Segmentation**: explores how students can be clustered into different segments based on their characteristics.

A. GENERAL FINAL GRADE PREDICTION

1) Overview

The objective of this experiment was to develop a model capable of predicting a student's final grade (denoted as G3) using the most relevant features, determined by specific correlation thresholds.

To understand the relationship between G3 and the other features, we referred to the previously generated correlation matrix, which highlighted two features—G1 and G2—as having extremely high correlations with G3. These features, which represent the grades for the first and second terms, showed a correlation score exceeding 80%, much higher than the correlations with other features.

As discussed earlier, this high correlation implies that any model trained to predict G3 using G1 and G2 would almost entirely rely on these two features, making the model less informative. Therefore, for this experiment, G1 and G2 were excluded in order to train a more balanced model. To achieve this, an "upper-bound" threshold was set to exclude features with extremely high correlation values, either positive or negative. The optimal upper-bound was set to 0.8, meaning any feature with an absolute correlation above 80

Additionally, a "lower-bound" threshold was applied to remove features with extremely low correlation values (close to 0%) that were identified as noise and impeded the model's learning. The optimal lower-bound value was set to 0.065, meaning features with an absolute correlation below 6.5% were excluded.

Thus, four different configurations were tested for this experiment, based on whether the bounded features were included or excluded:

1) Including both upper-bound'' and lower-bound'' features.
2) Including upper-bound'' but excluding lower-bound'' features.
3) Excluding upper-bound'' but including lower-bound'' features.
4) Excluding both upper-bound'' and lower-bound'' features.

2) Machine Learning Techniques

For this experiment, Ensemble Learning was chosen as the primary machine learning technique, utilizing a method called Stacking. Stacking involves training multiple base models on the data, and then using a final meta-model to combine their predictions, giving different weights to each base model depending on the context.

The base models used were:

1) **Random Forest**: an ensemble method based on the bagging approach. Random Forest constructs multiple decision trees using random subsets of the data and combines their results. This technique is particularly effective in dealing with noisy data, as the ensemble of trees helps to reduce noise by averaging out the predictions.
2) **Gradient Boosting**: an ensemble method based on the boosting approach. This technique builds decision trees sequentially, with each tree learning from the errors of its predecessor and correcting them. Gradient Boosting is particularly useful for capturing complex patterns in data but requires careful tuning to optimize performance.

For the meta-model, a simple Logistic Regression model was selected. Logistic Regression was chosen for its efficiency and ability to provide probabilistic predictions, making it a suitable choice for aggregating the predictions from the base models.

3) Results

As previously mentioned, four tests were conducted for this experiment, each with different configurations of the upper and lower bounds.

The best performing model was, as expected, Case 2. This model utilized both G1 and G2, which naturally led to highly accurate predictions, while excluding features that acted as noise. The final accuracy of the model was slightly above 48%. In Figure 2, the predicted values of G3 are plotted against the actual values, with most points lying close to the dotted red line, indicating correct predictions.

The second-best performing model was Case 1, which also included G1 and G2 but did not exclude the lower-bound features. This model achieved an accuracy slightly above 46%. Cases 3 and 4 performed significantly worse, with accuracies just above 21% and just below 24%, respectively.

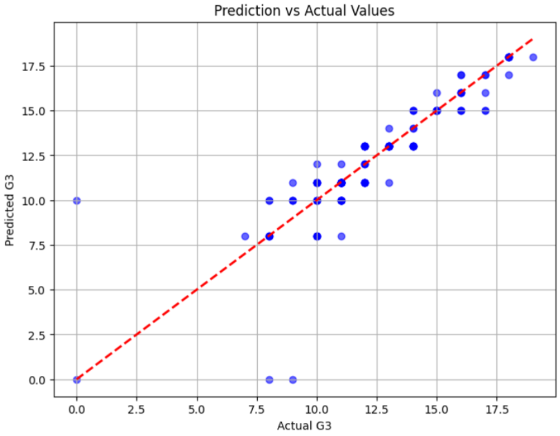

FIGURE 2. Model predictions vs ground truth for best performing model.

The results further emphasize the significant influence of G1 and G2 on predicting G3. The test cases incorporating these features achieved approximately twice the accuracy of the best-performing model that excluded them. However, it is important to highlight that this over-reliance on G1 and G2 may lead the model to underweight other features, potentially hindering its ability to learn diverse patterns in the data. This could negatively impact the model's generalization capability, particularly when handling unseen data.

B. TAILORED FINAL GRADE PREDICTION

1) Overview

The goal of this experiment was to train a model to predict a student's final grade, but with the added complexity of using predefined feature "sub-groups" for training. This approach complements the previously discussed General Final Grade Prediction experiment by examining how well the model performs when trained on different feature subsets.

The three chosen feature sub-groups were as follows:

- **Group 1**: All features.
- **Group 2**: Only the grade features, G1 and G2.
- **Group 3**: All features excluding the grade features.

These groupings were selected based on the findings from the General Final Grade Prediction experiment, where G1 and G2 were shown to have high correlation with G3. By including or excluding these features, the experiment explores whether the data relies too heavily on these two features.

2) Machine Learning Techniques

For this experiment, a Linear Regression model was chosen. Linear Regression is one of the simplest and most widely used regression techniques. It models the relationship between a dependent variable and one or more independent variables by fitting a linear equation to the data. While Linear Regression is highly interpretable and works well with datasets exhibiting linear relationships, it assumes that the input variables are independent and that multicollinearity is minimal.

It is important to note that the model's hyperparameters were left at their default settings throughout the experiment, with no adjustments made between different test cases.

3) Results

The results clearly demonstrated a distinct performance difference between the models trained with the grade features and those that did not include them. The model trained without the grade features achieved an accuracy of approximately 34%. In contrast, the model trained exclusively on the two grade features (G1 and G2) reached an accuracy of nearly 85%, more than double that of the previous model. The third test, which used all features, resulted in a slight increase in accuracy, with the model achieving just under 86

Although these findings were not entirely unexpected, they reinforce the conclusions of the General Final Grade Prediction experiment, further highlighting the significant dependence of final grade prediction on the G1 and G2 features.

C. DIMENSIONALITY REDUCTION

1) Overview

The goal of this experiment was to reduce the number of features in the dataset and investigate how this reduction would impact the model's accuracy when predicting the final grade (represented by the feature G3).

The experiment was conducted using three versions of the dataset, each with different dimensions. One version used the full dataset, while the other two employed datasets with 95% and 90% explained variance, respectively. Figure 3 illustrates the explained variance as a function of the number of components, which helps to visualize the dimensionality reduction.

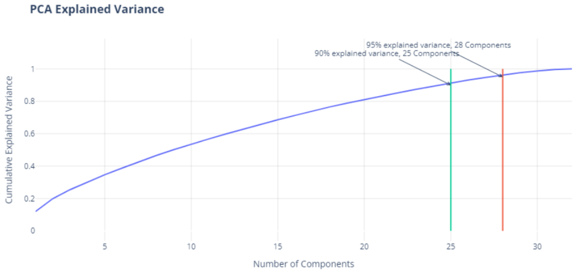

FIGURE 3. Graph illustrating the reduction in cumulative explained variance as the number of components decreases.

2) Machine Learning Techniques

Principal Component Analysis (PCA) was used for dimensionality reduction. PCA combines the original features into new variables, called principal components, which capture the most significant patterns in the data. This technique reduces the number of dimensions while retaining essential information, making it valuable for simplifying data, reducing noise, and preparing for further analysis or visualization.

Following dimensionality reduction, Linear Regression was employed to predict G3. Linear Regression models the relationship between a dependent variable and one or more independent variables by fitting a linear equation to the data. The model minimizes the sum of squared differences between observed and predicted values, making it suitable for predicting outcomes and analysing variable relationships.

3) Results

The reduction in model size was notable: from 1.25 KB for the original dataset to 0.85 KB and 0.80 KB for the 95% and 90% explained variance versions, respectively. The dimensionality reduction removed only up to six features from the original dataset.

In terms of accuracy, the Logistic Regression model exhibited the following results:

The original dataset achieved an accuracy of 31The 95% explained variance dataset resulted in an accuracy of 28%. The 90% explained variance dataset yielded an accuracy of 29%. The worst-performing model was the 95% explained variance dataset, which was nearly 10% less accurate than the original. Although this decrease may seem relatively small, it is worth noting that the accuracy was already low, and thus, the change should be interpreted with caution.

D. STUDENT SEGMENTATION

1) Overview

The goal of this experiment was to segment the students into distinct clusters to uncover specific characteristics.

First, the dataset was analyzed for its suitability for K-Means Clustering. Since this algorithm is not ideal for categorical data, a subset of numerical features was selected for clustering. These features were then standardized to ensure that variables with larger scales did not dominate the clustering process.

Next, the number of clusters, k, was determined. The elbow method, a widely-used technique, was employed to identify the optimal number of clusters. This method involves plotting the Within-Cluster Sum of Squares (WCSS) against various values of k. By visualizing the curve, the optimal k is determined at the "elbow point," where the rate of decrease in WCSS significantly slows. As shown in Figure 4, the elbow point occurs at $k = 2$, making it the optimal choice for this dataset. Given the clarity of the elbow point, other methods, such as the silhouette coefficient or gap statistic, were deemed unnecessary.

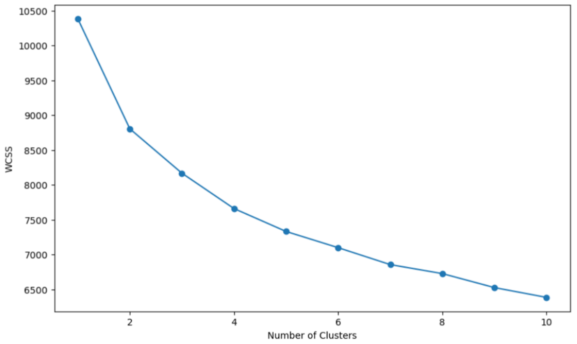

FIGURE 4. Graph showing the relationship between the number of clusters (*k*) and the Within-Cluster Sum of Squares (WCSS) for the dataset. WCSS represents the total variance within each cluster, and it decreases as the number of clusters increases.

2) Machine Learning Techniques

K-Means Clustering was used for this experiment. K-Means is an unsupervised machine learning algorithm that partitions data into a specified number of clusters based on feature similarity. It is commonly used for segmentation tasks and operates by minimizing the distance between data points and their respective centroids, the centre points of each cluster, effectively grouping similar data points together.

3) Results

Upon analysing the centroids for each cluster, it was found that Cluster 1 (C1) contained slightly older students than Cluster 0 (C0). Students in C0 generally exhibited higher academic performance, as reflected by their "failures" and "G3" scores. This difference may be attributed to several factors, such as higher levels of absenteeism, more frequent social activities, greater alcohol consumption, and less time spent studying in C1 compared to C0. Additionally, the parents of C1 students typically had lower educational levels than those of students in C0, and the students in C1 appeared to have somewhat poorer relationships with their families.

Since the dataset is high-dimensional, with more than two features, Principal Component Analysis (PCA) was used to reduce the dimensionality for visualization purposes. The reduced data was then plotted in a scatter plot, as shown in Figure 5.

V. ETHICAL REVIEW

Ethical considerations must underpin the use of datasets in machine learning to avoid bias and ensure fairness, transparency, and respect for privacy, especially within the domain of education. This analysis examines the ethical implications brought by the "Student Performance" dataset and the results generated from applying machine learning techniques to it.

Foremost, it is important to note that the dataset was taken from Kaggle, which is known for its reputable datasets, and compiles student data gathered from a limited number of classes across two Portuguese school involving the subjects of mathematics and Portuguese. C. Paulo [11] claims that the data was collected through a combination of school reports and questionnaires. Therefore, it should be assumed that the data may not accurately reflect the entire student body of these schools, or any school in general for that matter, and may also be subject to outliers.

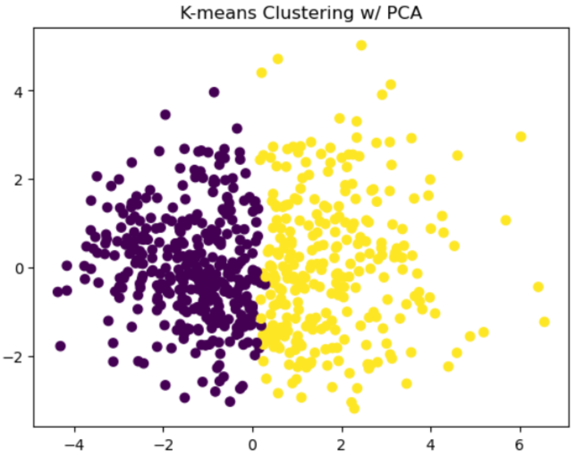

FIGURE 5. Scatter plot visualizing clusters identified by K-Means Clustering after applying dimensionality reduction with Principal Component Analysis (PCA). Each point represents a student, and the two clusters are differentiated by color.

Furthermore, while the dataset includes information on a total of 649 students, it raises bias concerns due to its sparse geographic representation which might indirectly favour certain groups of people when extending it beyond the scope of Portuguese schools [12], [13]. Additionally, variables in the dataset such as "Medu" and "Fedu" which represent parental education, among other family-related data, pose extensive socioeconomic and cultural concerns that falsely indicate a correlation between familial statuses and student performance. In the context of this dataset, it serves to provide better context towards the specific students in the regional area of these two schools, however, if it were to be used as training data for a machine learning model and extended beyond this region, it might show signs of bias towards those students who come from larger families or whose parents have higher levels of education than others.

As discussed previously, four machine learning experiments were conducted on this dataset. The "General Final Grade Prediction" and "Tailored Final Grade Prediction" experiments both work synonymously to predict what the final grade of a student is. While they make use of the variables "G1" and "G2", which represent a student"s first and second period grades, respectively, to predict the final grade, they also investigate the use of other features within the dataset to accomplish this task. It should be noted that the best accuracies were mostly correlated to the two grade features, signifying that the models performed better when using statistical performance-related data rather than demographic or familial data. This shows that the models do not exhibit any inherent signs of bias based on a student"s background. On the other hand, the "Dimensionality Reduction" experiment may pose ethical issues since it is not transparent with the features it is utilising. The accuracies of this model being low further prove that it is not suitable for such a task since it is most likely making use of demographic data to predict grades.

The "Student Segmentation" experiment involved segmenting students into different clusters to identify specific characteristics. In this case, the model detected certain biases existing within the dataset. For instance, a correlation between age and student performance was identified, with younger students generally performing better in school. Analysing the resulting clusters further revealed an additional correlation between parental education and student performance, as mentioned previously. This is indicative that models trained on this data to predict student grades could inaccurately favour younger students whose parents have higher levels of education, which is reflected in the three previously discussed experiments that performed poorly when taking into consideration such attributes.

Addressing such biases is an important step when applying machine learning models to datasets as it could lead to discriminatory outcomes, such as penalizing students from disadvantaged family backgrounds. The societal impact of deploying such models could unintentionally harm vulnerable groups and enhance existing stigmatization. However, if the biases are rooted from the model, the result might aid struggling students by identifying those who are most likely to fare poorly in school through carefully selected, unbiased features. These issues heavily affect the long-term societal implications of such a model. At a large scale, it could influence educational policies, teacher evaluations, or student tracking systems. With biased data, this would risk institutionalising inequities and perpetuating harmful stereotypes. Furthermore, if the model were to continuously improve itself based upon the gathering of more student data, the chance of there being biased data might increase. Therefore, continuous monitoring would be necessary to ensure that the long-term impact remains positive and aligned with societal values.

VI. WEB PORTAL USAGE GUIDE

Web portal is available at https://mkenely.com/ics5110/.

- **Home Page**
  - Compilation of links to all relevant subpages
- **Data Visualisation**
  - **Feature Reference**
    - ∗ Table of all features in the dataset we used.
    - ∗ Feature: string name of the feature
    - ∗ Type: object = categorical feature, int64 = numerical feature
    - ∗ Description: textual description of the feature and its categorical labels (if any)
    - ∗ Encoding Mappings: labels and their numerical encodings for categorical features
  - **Feature Distributions**
    - ∗ Distribution plots for each feature in the database
  - **Feature Correlation Matrix**
    - ∗ Shows the correlation between all features. Larger data points indicate greater absolute correlation. Blue points indicate positive correlation, and red points indicate inverse correlation.
  - **Feature vs G3 Scatter Plots**
    - ∗ Scatter plots between all features and G3 which we are predicting. The features with the greatest absolute correlation with G3 are shown first. Data points are jittered to get a rough idea of the quantity of individuals at each point due to the large amount of overlap due to G3 being a discrete numerical variable.
- **ML Techniques**
  - Links to gradio implementations of our 4 ML models. In each implementation, the user is prompted to input their own student features and will be presented with predictions made by the models.
    - ∗ **PCA**
      - · Makes predictions for G3 using 3 different Linear Regression models, each trained on datasets with different numbers of components according to retained variance compared to the original dataset (original, 95% variance, 90% variance). Also shows the runtime and size of each model.
    - ∗ **Ensemble Model**
      - · Makes predictions for G3 using the stacking ensemble model we trained. Also shows the runtime.
    - ∗ **K-Means Clustering**
      - · Predicts the cluster to which the inputted student belongs (0 or 1). Also shows the runtime and the indices of the top 5 performing students in the same cluster as the inputted student.
    - ∗ **Linear Regression**
      - · Makes predictions for G3 using 3 different Linear Regression models, each trained on datasets with different combinations of features: all features, all features except grades (G1, G2), and only grades (G1, G2).

REFERENCES

[1] D. Maulud and A. Abdulazeez, "A review on linear regression comprehensive in machine learning," *Journal of Applied Science and Technology Trends*, vol. 1, pp. 140–147, 12 2020.

[2] G. Kaya Uyanık and N. Güler, "A study on multiple linear regression analysis," *Procedia - Social and Behavioral Sciences*, vol. 106, p. 234–240, 12 2013.

[3] S.-A. Alexandropoulos, C. Aridas, S. Kotsiantis, and M. Vrahatis, *Stacking Strong Ensembles of Classifiers*, 05 2019, pp. 545–556.

[4] G. Ngo, R. Beard, and R. Chandra, "Evolutionary bagging for ensemble learning," *Neurocomputing*, vol. 510, p. 1–14, Oct. 2022. [Online]. Available: http://dx.doi.org/10.1016/j.neucom.2022.08.055

[5] H. Drucker, C. Cortes, L. Jackel, Y. Lecun, and V. Vapnik, "Boosting and other ensemble methods," *Neural Computation*, vol. 6, pp. 1289–1301, 01 1994.

[6] M. Hasan, M. Abedin, P. Hájek, K. Coussement, N. Sultan, and B. Lucey, "A blending ensemble learning model for crude oil price forecasting," *Annals of Operations Research*, 01 2024.

[7] V. Sharma, "A study on data scaling methods for machine learning," *International Journal for Global Academic Scientific Research*, vol. 1, 02 2022.

[8] K. Sankpal, "A review on data normalization techniques," *International Journal of Engineering Research and*, vol. V9, 07 2020.

[9] D. Berrar, *Cross-Validation*, 01 2018.

[10] S. Mishra, U. Sarkar, S. Taraphder, S. Datta, D. Swain, R. Saikhom, S. Panda, and M. Laishram, "Principal component analysis," *International Journal of Livestock Research*, p. 1, 01 2017.

[11] P. Cortez, "Student performance," 2008.

[12] J. Chakraborty, S. Majumder, and T. Menzies, "Bias in machine learning software: why? how? what to do?"   Association for Computing Machinery, 2021. [Online]. Available: https://doi.org/10.1145/3468264.3468537%7D,doi={10.1145/3468264.3468537},pages={429âĂŞ440}

[13] "Ethical and bias considerations in artificial intelligence (ai)/machine learning," *Modern Pathology*, p. 100686, 2024.

• • •