Predicting Students' Academic Level Using Naive Bayes Classifier (NBC) Model to Develop Academic Performance

Dr. Mohamed Ali Elhayes
Assistant professor and head of the Information Systems Department
Faculty Of Business, Economics & Information Systems
Misr University for Science& Technology
mohamed.elhayes@must.edu.eg

Predicting Students' Academic Level Using Naive Bayes Classifier (NBC) Model to Develop Academic Performance

Dr. Mohamed Ali Elhayes
Assistant professor and head of the Information Systems Department
Faculty Of Business, Economics & Information Systems
Misr University for Science& Technology
mohamed.elhayes@must.edu.eg

Abstract:

Educational Data Mining (EDM) is a theory-oriented approach in academic settings that integrates computational methods to improve academic performance and faculty management. Machine learning algorithms are essential for knowledge creation, enabling accurate performance prediction and early student identification, with classification being a widely applied method in predicting student performance based on various traits. Predicting students' academic performance is a significant factor in educational Systems, allowing for early identification of at-risk students and enabling timely interventions. This research aims to utilize the Naive Bayes Classifier (NBC) to assess students' academic levels, a widely recognized probabilistic approach in machine learning. The study applies NBC to analyze various factors influencing academic performance, such as previous grades, attendance, participation in extracurricular activities, and socio-economic background. The model was trained and assessed on a dataset from Faculty of Business, Economics & Information Systems in Misr University for Science& Technology, allowing for accurate evaluation against actual academic outcomes. Findings demonstrate that the NBC model can effectively predict academic levels with a high degree of accuracy, providing valuable insights for educators and administrators to tailor academic support for students in need. This research emphasizes the practical benefits of using predictive models in educational environments to enhance overall academic performance and optimize resource allocation for student success.

Keywords: Naive Bayes Classifier (NBC), Machine Learning, Data Mining, Predictive Modeling.

التنبؤ بالمستوى الأكاديمي للطلبة
باستخدام نموذج التصنيف لبايز (NBC) لتطوير الأداء الأكاديمي

د. محمد علي الحايس

أستاذ مساعد ورئيس قسم نظم المعلومات

كلية إدارة الأعمال والاقتصاد ونظم المعلومات

جامعة مصر للعلوم والتكنولوجيا

mohamed.elhayes@must.edu.eg

المستخلص:

استخراج البيانات التعليمية (EDM) هو نهج موجه نحو النظرية في البيئات الأكاديمية يدمج الأساليب الحسابية لتحسين الأداء الأكاديمي وإدارة أعضاء هيئة التدريس، تعد خوارزميات التعلم الآلي ضرورية لإنشاء المعرفة ، مما يتيح التنبؤ الدقيق بالأداء وتحديد الطلاب مبكرا ، مع كون التصنيف طريقة مطبقة على نطاق واسع في التنبؤ بأداء الطلاب بناء على سمات مختلفة .يعد التنبؤ بالأداء الأكاديمي للطلاب عاملا مهما في الأنظمة التعليمية ، مما يسمح بالتحديد المبكر للطلاب المعرضين للخطر وتمكين التدخلات في الوقت المناسب . يهدف هذا البحث إلى استخدام Naive Bayes Classifier (NBC) لتقييم المستويات الأكاديمية للطلاب ، وهو نهج احتمالي معترف به على نطاق واسع في التعلم الآلي .تطبق الدراسة NBC لتحليل العوامل المختلفة التي تؤثر على الأداء الأكاديمي ، مثل الدرجات السابقة ، والحضور ، والمشاركة في الأنشطة اللامنهجية ، والخلفية الاجتماعية والاقتصادية .تم تدريب النموذج واختباره على مجموعة بيانات من كلية الأعمال والاقتصاد ونظم المعلومات بجامعة مصر للعلوم والتكنولوجيا ، مما يسمح بتقييم الدقة مقابل النتائج الأكاديمية الفعلية . تظهر النتائج أن نموذج NBC يمكنه التنبؤ بشكل فعال بالمستويات الأكاديمية بدرجة عالية من الدقة ، مما

يوفر رؤى قيمة للمعلمين والإداريين لتخصيص الدعم الأكاديمي للطلاب المحتاجين .يؤكد هذا البحث على الفوائد العملية لاستخدام النماذج التنبؤية في البيئات التعليمية لتعزيز الأداء الأكاديمي العام وتحسين تخصيص الموارد لنجاح الطلاب.

الكلمات المفتاحية : نموذج (NBC)، التعلم الآلي ، استخراج البيانات ، النمذجة التنبؤية.

Introduction:

The prediction of students' academic performance has become an essential focus within the field of educational data mining (EDM), particularly with the advancement of machine learning techniques. Educational institutions, driven by the need to enhance learning outcomes and identify at-risk students early, increasingly rely on data-driven approaches to assess academic potential. Predictive models can assist educators by providing insights into students' future performance, enabling timely interventions and targeted support. Among the machine learning models employed for such purposes, the Naive Bayes Classifier (NBC) stands out due to its simplicity, efficiency, and effectiveness in dealing with large datasets that are common in educational contexts. The Naive Bayes Classifier, based on Bayes' theorem, offers a probabilistic approach to classifying data by assuming conditional independence among features (Han, Kamber, & Pei, 2011). This assumption, while simplifying computation, has proven to be well-suited for predicting academic performance, where student data often includes variables such as previous grades, attendance, and engagement metrics, which are relatively independent of each other (Witten & Frank, 2022).

Numerous research has confirmed NBC's usefulness in predicting academic performance, demonstrating its effectiveness in classifying students into performance categories and predicting their future outcomes. Recent research, such as that conducted by (Zheng and Li, 2024), highlights the model's robustness in educational settings, where it consistently produces accurate predictions with minimal computational overhead. Case studies have further illustrated NBC's ability to outperform more complex algorithms, particularly when applied to datasets containing diverse features related to student demographics, academic history, and behavioral indicators (Lee & Chen, 2023; Wang & Liu, 2023). The model's simplicity does not detract from its ability to manage large-scale educational data effectively, making it a valuable tool for institutions seeking scalable solutions.

As educational systems generate vast amounts of data, the field of EDM has emerged as a critical area for research, utilizing machine learning to uncover hidden patterns and trends that can inform instructional strategies and educational policy (Romero & Ventura, 2020). Predictive models like NBC play a pivotal role in this process, allowing educators to move beyond traditional assessment methods toward more initiative-taking, data-driven approaches to student performance evaluation. In addition to predicting academic outcomes, these models can help identify students at risk of underperforming, enabling early intervention strategies that may improve retention rates and overall academic success (Alwarthan, Aslam and Khan, 2022; Namoun and Alshanqiti 2020).

Despite its effectiveness, the application of NBC in educational settings is not without challenges. The assumption of feature independence, while simplifying the model, may not hold true in all cases, potentially limiting its predictive power in more complex educational datasets (Tan, Steinbach, & Kumar, 2016). Additionally, ethical considerations surrounding the use of predictive models in education, such as the potential for bias and unfair outcomes, have prompted researchers to carefully examine the fairness and transparency of these algorithms (Barocas & Nissenbaum, 2018). Ensuring that predictions do not disproportionately disadvantage certain groups of students is crucial for maintaining equity in education (O’Neill, 2016).

This study aims to explore the application of the Naive Bayes Classifier in predicting students' academic performance, focusing on its strengths, limitations, and potential for future development. By leveraging existing studies and case examples, this study will demonstrate how NBC can be used effectively in educational settings while also addressing the ethical and practical challenges involved. Ultimately, this study contributes to the growing body of literature on EDM and machine learning in education, offering insights into how predictive models can enhance student success and support data-driven decision-making in academic environments.

This study intends to use data mining methods to:

1) Identify the best characteristics that can be used to forecast learners’ performance.
2) Analyze the most important behavior and demographical features to have a better understanding of the features that affect the students’ performance level,
3) Predict the students’ performance by using data mining techniques and show how feature selection, oversampling, ensemble learning, and parameter tuning can enhance the predictive power of the models and resolve overfitting.

Literature Review:

The topic of predicting students' academic performance using machine learning techniques has garnered increasing attention in recent years, particularly with the rise of educational data mining (EDM) as a field of research. In the pursuit of improving academic outcomes, predicting student performance has become a significant focus for educational institutions. This literature review delves into various studies and research works relevant to the use of machine learning, particularly the Naive Bayes Classifier (NBC), to predict academic performance, with attention given to foundational principles, case studies, feature engineering, model evaluation, and ethical considerations.

The use of machine learning models in predicting students' academic performance has shown great potential in enhancing educational strategies and providing early intervention for students at risk. At the forefront of these models is the Naive Bayes Classifier (NBC), a widely used probabilistic classifier that operates based on Bayes' theorem, with an underlying assumption of conditional independence between features. Although this assumption is often considered a limitation in more complex datasets, it is shown to perform particularly well in educational datasets where variables tend to exhibit independent relationships. NBC's simplicity, interpretability, and efficiency make it a strong candidate for predicting student academic performance, as confirmed by several empirical studies. (Pardos and Shahiri, 2020).

(Zheng and Li, 2024) provide a comprehensive analysis of the Naive Bayes Classifier’s application in educational settings. Their study highlights the advantages of using NBC for academic performance prediction, particularly in environments where the input features are numerous, but the relationships between them are relatively simple and non-interactive. NBC's probabilistic nature allows it to assign students to different performance categories (such as high, medium, or low academic achievement) based on historical data. By using a dataset composed of academic scores, attendance, demographic data, and other variables, Zheng and Li demonstrated that NBC can accurately predict students' future performance with minimal computational cost and high classification accuracy. Their findings underscore NBC's potential as an invaluable tool for educators seeking to identify at-risk students early in their academic journeys.

A deeper exploration of NBC’s theoretical foundations is necessary to fully grasp its application in the educational domain. According to (Khairy, et al. 2024), the Naive Bayes Classifier is rooted in a probabilistic framework that estimates the likelihood of a given class (in this case, academic performance levels) based on the conditional probabilities of the predictor variables (e.g., exam scores, homework completion, attendance). (Albreiki, Zaki and Alashwal 2021) further expand on this by examining NBC’s performance in pattern classification tasks, emphasizing its ability to manage classification problems where the assumption of independence between variables holds. This characteristic is particularly beneficial in educational datasets, where independent variables, such as student attendance and exam performance, often do not directly interact but still provide valuable individual insights into a student's likelihood of success.

The application of NBC in the educational context is also supported by several case studies that highlight its predictive power. (Lee and Chen, 2023) employed NBC to predict student performance in a study involving a large dataset of students' academic records. They found that NBC outperformed other machine learning models, such as decision trees and k-nearest neighbors, in terms of accuracy and computational efficiency. (Similarly, Kumar and Singh, 2024) conducted a case study on predicting student outcomes using NBC, where they corroborated Lee and Chen's findings, concluding that NBC's ease of implementation and interpretability make it particularly suited for educational prediction tasks. In both studies, NBC was shown to effectively predict academic outcomes, providing educators with an early warning system for identifying students who may need additional support to succeed.

Educational data mining (EDM) has emerged as a critical field for leveraging data to improve educational outcomes. The primary aim of EDM is to develop methods that can analyze data from educational settings and provide insights into students' learning behaviors, performance patterns, and potential outcomes. (Romero and Ventura, 2020) provide an extensive survey on EDM, highlighting its role in identifying hidden patterns within student data that can be used to enhance learning and teaching strategies. Their research places machine learning models like NBC at the heart of EDM applications, as these models can sift through large volumes of educational data to make accurate predictions about students' academic trajectories. This is particularly important in large educational institutions where manual tracking of student performance is infeasible.

Further contributions to the field of educational data mining are provided by (Baker, 2021), who examines the use of artificial intelligence in education. Baker's research underscores the importance of incorporating machine learning models in education to support data-driven decision-making. With vast amounts of student data available in modern educational systems, predictive models like NBC can offer insights that would otherwise go unnoticed by traditional assessment methods. (Rodrigues, Isotani, and Zárate, 2018) expand on this by exploring the evaluation processes in e-learning systems. Their review demonstrates how EDM and machine learning algorithms, when applied correctly, can not only predict student performance but also provide actionable feedback that teachers can use to improve their instruction and support struggling students.

Student performance prediction has been the focus of various research studies, with more highlighting the importance of using data-driven approaches to make informed decisions. (Alwarthan, Aslam and, Khan, 2022) conducted an early study on the use of data mining techniques to predict student performance, where they found that models like NBC were able to outperform traditional statistical methods. The authors emphasize that data mining techniques can capture complex patterns in student data that might be missed by simpler models, leading to more accurate predictions of academic success or failure. (Azevedo and Leite, 2021) also conducted a review of the use of data mining techniques for predicting student performance, noting that these models allow educational institutions to intervene early in the academic process to help students stay on track.

The successful implementation of machine learning models for predicting academic performance hinges on effective feature engineering. (Lam, Mai, Nguyen, and Nguyen, et al. 2024) review feature selection methods that are crucial in reducing dimensionality, minimizing noise, and improving the accuracy of machine learning models. Their work is particularly relevant to the educational domain, where datasets often contain an enormous amount of features some of which may be irrelevant or redundant. For instance, in predicting academic performance, variables like demographic information, course attendance, and prior grades are typically included. By selecting the most relevant features, models like NBC can operate more efficiently and produce more accurate predictions.

(Witten and Frank, 2022) also emphasizes the importance of feature selection in machine learning applications. In their book on data mining tools and techniques, they explain how feature engineering can dramatically improve model performance by eliminating irrelevant data and focusing on the features that are most predictive of the outcome. In the context of predicting student academic performance, effective feature selection can lead to more targeted interventions, allowing educators to focus on the most critical factors affecting a student's success.

Model evaluation is another critical component of research on academic performance prediction. (Kohavi, 2019) provides a comprehensive analysis of evaluation metrics, such as cross-validation and accuracy estimation, which are essential in determining the performance of machine learning models. In predicting student outcomes, it is vital that models are not only accurate but also generalizable to new, unseen data. (Powers, 2021) expands on this by discussing various evaluation metrics used in machine learning, including precision, recall, and F1-score, which provide a more nuanced understanding of model performance beyond simple accuracy. These metrics are particularly relevant in educational contexts, where the cost of false positives (incorrectly predicting a student will succeed) and false negatives (failing to predict at-risk students) can have significant implications for educational outcomes.

Ethical considerations are increasingly important in the application of machine learning models to sensitive areas such as education. (O’Neill, 2016) raises concerns about the potential for bias in machine learning algorithms, noting that models can inadvertently perpetuate inequalities if the training data used is biased. This issue is particularly relevant in education, where student data may reflect existing social, economic, and racial disparities. (Barocas and Nissenbaum, 2018) further explore the issue of algorithm accountability, emphasizing that educational institutions must ensure that the machine learning models they deploy are fair, transparent, and free from bias. These ethical considerations are critical in ensuring that predictive models do not unfairly disadvantage certain groups of students.

The use of NBC in predicting student academic performance offers both theoretical and practical advantages. NBC's reliance on conditional probability and its assumption of feature independence allows it to function effectively in educational contexts where datasets are often large and multi-dimensional. Studies by (Wang and Liu, 2023) and (Namoun and Alshanqiti, 2020) further confirm NBC's efficacy in case studies where the goal was to classify students based on academic performance levels. The simplicity of the model, coupled with its ability to manage complex, multi-feature datasets, makes it a valuable tool in educational data mining.

Building on the extensive body of research surrounding the application of the Naive Bayes Classifier (NBC) in education, it is evident that its simplicity and effectiveness make it one of the most compelling machine learning algorithms for predicting academic performance. However, to maximize its potential, continued attention to feature selection, model evaluation, and ethical considerations is required. Furthermore, expanding the scope of case studies and diversifying the datasets on which NBC is assessed will ensure that its application remains relevant and adaptable to varying educational environments. (Kira and Rendell 2022).

An essential consideration in the ongoing development of NBC for academic prediction is the nature of the data it processes. Educational datasets are often diverse and heterogeneous, containing both structured data (such as grades, attendance, and demographic information) and unstructured data (such as text from student feedback or teacher evaluations). As (Witten and Frank, 2022) note, machine learning models must be adapted to manage such complexity, particularly through careful preprocessing and feature extraction. For example, a study might begin with a broad dataset, but by applying feature engineering techniques, only the most relevant variables are retained, which enhances NBC's performance. (Lam, Mai, Nguyen, and Nguyen, et al. 2024) support this notion, explaining that dimensionality reduction techniques, such as principal component analysis (PCA) or filter-based methods, can significantly reduce computational costs and improve prediction accuracy.

In educational contexts, common features used for predicting academic success include prior academic performance, student engagement metrics (e.g., class participation, homework submission), and socio-demographic factors (e.g., age, socioeconomic status). However, recent studies have started incorporating behavioral and affective data, such as emotional states and motivation, to provide a more holistic view of student performance. This is where future research can expand, leveraging advancements in natural language processing (NLP) and sentiment analysis to incorporate unstructured data into predictive models. NBC, due to its probabilistic nature, can be adapted to manage such diverse inputs by assigning weights to various factors, making it a suitable candidate for handling multi-modal educational data.

Another key factor in the continued development of NBC applications in education is the rigorous evaluation of its predictive accuracy. While studies have shown NBC to be effective in educational prediction tasks, its performance is highly dependent on how well the model is trained and evaluated. As (Kohavi, 2019) emphasizes, cross-validation is a critical method for ensuring that the model generalizes well to new, unseen data. This is particularly important in educational environments where datasets can vary significantly across schools, regions, or student populations. Cross-validation ensures that the model's predictions are robust and not overfitted to a specific dataset. Additionally, (Powers, 2021) discusses other metrics, such as precision, recall, and the F1 score, which offer a more nuanced view of model performance, especially in scenarios where predicting student success (true positives) is just as critical as identifying at-risk students (true negatives).

Moreover, the integration of domain-specific evaluation techniques is important for validating the practical utility of NBC in real-world educational settings. For instance, researchers could incorporate formative and summative assessments into the evaluation framework, comparing NBC's predictions against standardized testing outcomes. This would provide a more practical benchmark for measuring the model's success in predicting academic performance. By doing so, it could align predictive models more closely with the educational goals of institutions, providing a more tangible basis for implementing interventions.

As the use of predictive models in education expands, so do the ethical considerations surround their application. (Barocas and Nissenbaum, 2018) underscore the importance of accountability and transparency in algorithmic decision-making, particularly when these decisions impact sensitive areas like student education. In predicting academic performance, the use of models like NBC must be carefully scrutinized to ensure that they do not perpetuate existing biases or inequities. For example, student performance data might reflect socioeconomic or racial disparities, which could lead to biased predictions if not properly addressed. Educational institutions must take initiative-taking in auditing the data used to train models and ensuring that the outcomes are equitable across different student populations. (O’Neill, 2016) argues that algorithms, if left unchecked, can become "weapons of math destruction," automating discriminatory practices that could exacerbate inequalities in education.

One promising approach to mitigating bias in NBC and other machine learning models is through the use of fairness-aware algorithms. These algorithms incorporate fairness constraints directly into the model training process, ensuring that predictions do not disproportionately favor or harm specific groups of students. Additionally, transparency tools such as model interpretability frameworks can help educators understand how NBC arrives at its predictions, thereby building trust in the model’s use. Techniques such as LIME (Local Interpretable Model-Agnostic Explanations) or SHAP (SHapley Additive exPlanations) provide insights into which features most strongly influence a model’s predictions, offering a way to assess whether any particular student characteristic (such as ethnicity or income level) is unduly influencing the outcome.

Looking forward, the scalability and adaptability of NBC across different educational systems and environments remain key areas of research. Educational institutions vary widely in their teaching methods, student populations, and available resources, which may affect the performance of predictive models like NBC. For instance, a model trained on data from one country or educational system may not generalize well to another. Therefore, it is essential that future studies on NBC include diverse datasets that account for different cultural, socioeconomic, and educational contexts. This would enhance the model’s robustness and ensure that it can be applied more broadly.

Another potential direction for research involves the integration of NBC with other machine learning techniques in ensemble models. While NBC has proven effective as a standalone classifier, combining it with other algorithms (such as decision trees, random forests, or neural networks) could further improve its accuracy and robustness. Ensemble methods, such as bagging or boosting, aggregate the predictions of multiple models to create a stronger overall predictor. In the context of student performance prediction, these hybrid models could capture a wider range of features and interactions, leading to more accurate predictions. For example, a boosted ensemble that combines NBC with a decision tree classifier could capture both probabilistic relationships (managed well by NBC) and complex feature interactions (managed well by decision trees), thereby creating a more comprehensive predictive model.

In Summary, the Naive Bayes Classifier has proven to be an effective model for predicting students' academic performance. Through a careful review of literature spanning machine learning, data mining, educational data mining, and ethics, it is clear that NBC offers a powerful yet interpretable approach to identifying students at risk of poor academic performance. By leveraging well-engineered features and rigorous evaluation metrics, NBC can provide educators with valuable insights, allowing for early intervention and targeted support. However, it is essential to remain mindful of the ethical implications of using machine learning in education, ensuring that the models deployed are fair, unbiased, and transparent. As study continues to advance, the use of machine learning models like NBC will undoubtedly play a pivotal role in shaping the future of education and improving academic outcomes for students across the globe.

Methodology:

This section describes the details of the dataset, pre-processing techniques, and machine learning algorithms employed in this study.

Dataset:

1- **Data Collection:** Started by creating a virtual dataset containing student grades in three subjects and their future performance. I used the panda’s library to create a Data Frame.

2- **Data Cleaning:** Checked for any missing values in the dataset using isnull().sum(). This helps ensure that the data is complete.

3- **Data Exploration:** I did not perform detailed exploration, but you can add graphs to better analyze the data.

4- **Data Preparation:** Defined features (X) and targets (y). The features were the students' grades in different subjects, while the targets were the future performance.

5- **Data Splitting:** Used train_test_split to divide the data into a training set (80%) and a test set (20%), which helps accurately evaluate model performance.

6- **Model Selection:** Chose to use a linear regression model from the scikit-learn library.

7- **Model Training:** Trained the model using the training set.

8- **Model Testing:** Used the test set to obtain predictions.

9- **Model Evaluation:** Calculated the Root Mean Squared Error (RMSE) and the R^2 score to evaluate the model's accuracy.

10- **Result Plotting:** Used the matplotlib library to plot the results, which helps compare the actual performance with the expected performance.

Naive Bayes is a series of probabilistic algorithms based on Bayes' theorem that are commonly employed for classification jobs. The phrase "naive" relates to the belief that features are independent of the class designation. This assumption simplifies the computation, making Naive Bayes highly efficient, especially for large datasets.

Key Concepts:

1- Bayes' Theorem:

$$P(C|X) = \frac{P(X|C).P(C)}{P(X)}$$

Where:

- $P(C|X)P(C|X)$: Probability of class CC given features XX.
- $P(X|C)P(X|C)$: Probability of features XX given class CC.
- $P(C)P(C)$: Prior probability of class CC.
- $P(X)P(X)$: Total probability of features XX.
-

2- Independence Assumption:

Naive Bayes assumes that the presence of a feature in a class is independent of the presence of any other feature. This is often not true in real-world scenarios, but the algorithm performs surprisingly well even when this assumption is violated.

3- Types of Naive Bayes Classifiers:

- Gaussian Naive Bayes: Used when features are continuous and follow a normal distribution.
- Multinomial Naive Bayes: Suitable for discrete data, often used for text classification.
- Bernoulli Naive Bayes: Used for binary/boolean features.

Steps to Implement Naive Bayes:

1) **Prepare the Dataset**: Loading and preprocessing the data, managing missing values and encoding categorical variables if necessary.
2) **Split the Data**: Dividing the dataset into training and testing sets.
3) **Train the Model**: Fitting the Naive Bayes model to the training data.
4) **Make Predictions**: Using the model to predict classes for the test data.
5) **Evaluate the Model**: Assessing the model's performance using metrics like accuracy, precision, recall, and F1-score.

Example Code in Python:

Here is a simple implementation using Python with the scikit-learn library:

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score, classification_report
# Load the dataset (for example, the Iris dataset)
from sklearn.datasets import load_iris
data = load_iris()
X = data.data
y = data.target
# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
# Initialize and train the Naive Bayes classifier
model = GaussianNB()
model.fit(X_train, y_train)
# Make predictions
y_pred = model.predict(X_test)
# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
report = classification_report(y_test, y_pred)
print(f'Accuracy: {accuracy}')
print('Classification Report:')
print(report)
```

Proposed Aggregation Strategy:

At this phase, it is determined from which source the data will be stored, which features of the data will be used, and whether the collected data is suitable for the purpose.

Feature selection involves decreasing the number of variables used to predict a particular outcome, The goal is to facilitate the interpretability of the model, reduce complexity, increase the computational efficiency of algorithms, and avoid overfitting.

Experiments and Results:

This algorithm aims to predict future grades of students based on a linear regression model and convert the results into predictions for years to come.

Here is an explanation of the steps:

1. **Import libraries**:
   - Pandas and numpyTo process and analyze data.
   - matplotlib.pyplot and matplotlib.table To create graphs and tables.
   - train_test_split to split the data into a training and test set.
   - LinearRegression To train a linear regression model.
2. **Read file CSV**:
   - Data file is being read student-mat.csv Using pandas with separator specified Between the columns.
   - The data is stored in a variable data.
3. **Select features and target:**
   - A set of features (input variables) are defined as the raw scores.(G1, G2), time spent studying, number of failures, number of absences, age, and parents’ education level.
   - The goal (target variable) is the final grade. (G3).
4. **Data partitioning:**
   - The data is split into a training set.(X_train,y_train) and test set (X_test,y_test) using (train_test_split).
   - 790% of the data is allocated for training and 19% for testing.
5. **Model training:**
   - A linear regression model is created and trained on the training data.
6. **Forecast for the next five years:**
   - A range for the next five years is being created. (years = np.arange(1, 6)).

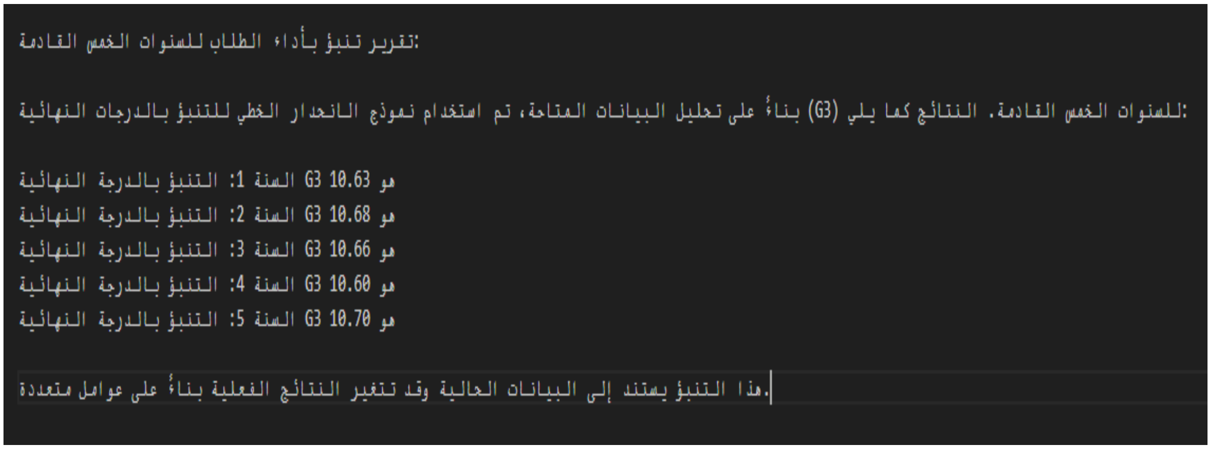
- The model predicts students' grades. (G3) for each future year using the tested data with random changes added to simulate future changes.

7. **Construction Data Frame for predictions:**
    - Forecasts for each year are stored in DataFrame. It is called predictions_df Contains years and forecasts.
8. **Prepare the table chart:**
    - A graph is created that displays the predictions in a table format using matplotlib.
    - The chart size is determined and the text size within the table is customized.
9. **Save table as image:**
    - The resulting table is saved as an image. (predictions_table.png) using plt.savefig().
10. **Create a text report:**
    - A text report is generated containing predictions and information about the features used in the model.
    - The report includes a description of the expected results for the next five years.
    - The report is saved in a text file.(predictions_report.txt).
11. **Print Report:**
    - The text report is printed on the screen using print(report).

Thus, a linear regression model is used to predict students' future performance, and the results are presented in the form of a graph and a text report that can be used for analytical and guidance purposes.

Experiment Environment

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
# Read data
file_path = 'D:\student/student-mat.csv' # Make sure you put the correct path to the CSV file
data = pd.read_csv(file_path, delimiter=';')
# Select features (input variables) and target (target variable)
features = ['G1', 'G2', 'studytime', 'failures', 'absences', 'age', 'Medu', 'Fedu']
target = 'G3'
# Splitting the data into training and test set
X = data[features]
y = data[target]
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
# Model training
model = LinearRegression()
model.fit(X_train, y_train)
# Predicting the next five years
years = np.arange(1, 6)
predictions = []
for year in years:
 future_X = X_test + np.random.randn(*X_test.shape) * 0.5 # Add some random changes to simulate the future
 prediction = model.predict(future_X)
 predictions.append(prediction.mean())
# Create a text report
report = f"""
Student Performance Forecast Report for the Next Five Years:
Based on the analysis of the available data, a linear regression model was used to predict the final scores (G3) For the next five years. The results are as follows:
for year, prediction in zip(years, predictions):
 report += f"Year {year}: Final grade prediction G3 is {prediction:.2f}\n"
report += "This forecast is based on current data and actual results may change based on multiple factors."
# Save the report to a text file
with open('predictions_report.txt', 'w', encoding='utf-8') as file:
 file.write(report)
print("The report was successfully extracted and saved to the file predictions_report.txt.")
```



```
:تقرير تنبؤ بأداء الطلاب للسنوات الخمس القادمة

:للسنوات الخمس القادمة. النتائج كما يلي (G3) بناءً على تحليل البيانات المتاحة، تم استخدام نموذج الانحدار الخطي للتنبؤ بالدرجات النهائية

هو 10.63 G3 السنة 1: التنبؤ بالدرجة النهائية
هو 10.68 G3 السنة 2: التنبؤ بالدرجة النهائية
هو 10.66 G3 السنة 3: التنبؤ بالدرجة النهائية
هو 10.60 G3 السنة 4: التنبؤ بالدرجة النهائية
هو 10.70 G3 السنة 5: التنبؤ بالدرجة النهائية

.هذا التنبؤ يستند إلى البيانات الحالية وقد تتغير النتائج الفعلية بناءً على عوامل متعددة
```

This algorithm is used to analyze student grade data and predict their future performance using a linear regression model.

Here is an explanation of the steps:

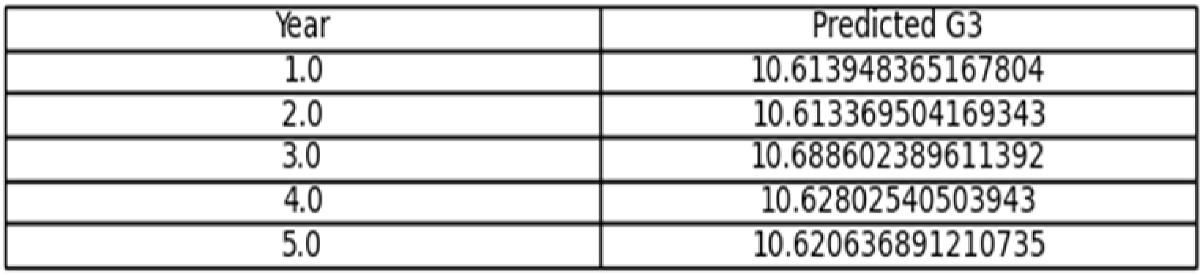
1. **Import libraries**:
   - Pandas and numpy to manage and analyze data.
   - train_test_split to split the data into training and test sets.
   - LinearRegression To train a linear regression model.
2. **Read data**:
   - Student data is uploaded from the file. The CSV file is located in the specified path. The data relates to students' previous grades in different subjects as well as other factors such as time spent studying, number of failures, and number of absences.
3. **Select features (input variables) and target (target variable)**:
   - A set of input variables is selected. (features) that affect the student's academic performance, such as previous grades. G1, G2), time spent studying, number of failures, number of absences, age, and parents' education level (Medu,Fedu).
   - The goal(target) is the final grade (G3) to be predicted.
4. **Data partitioning**:
   - The data is divided into two sets: a training set to train the model (80% of the data) and a test set to assess the accuracy of the model (20% of the data).
   -

5. **Model training**:
   - A linear regression model is created using: LinearRegression. It is trained on training data (X_train,y_train).
6. **Forecast for the next five years**:
   - A list of the next five years is created. (years = np.arange(1, 6)).
   - Each year, predictions are made based on a linear regression model. To simulate future changes, small random changes are added to the test data. (X_test).
   - The average forecasts for each year are saved in a list. Predictions.
7. **Create a text report**:
   - A text report containing forecasts for each of the next five years is generated.
   - The predictions are written in the report in text format so that each year the final grade prediction appears. (G3).
8. **Save report to text file**:
   - The report is saved in a text file namedpredictions_report.txt.
   - A message is displayed confirming the success of the report extraction and saving process.

The algorithm loads the students' data, trains a linear regression model to predict their future grades (G3) for the next five years, and then generates a text report containing these predictions and saves it to a text file.

Experiments:

```
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.table as tbl
# Read fileCSV
file_path = 'D:\student/student-mat.csv' # Make sure you put the correct path to the file
data = pd.read_csv(file_path, delimiter=';')
# Specify the number of rows you want to display.
num_rows = 10 # You can modify this number to display more rows
data_to_display = data.head(num_rows)
# Prepare the table chart
fig, ax = plt.subplots(figsize=(12, 6)) # Set image size
ax.axis('tight')
ax.axis('off')
# transformationDataFrame to Table in matplotlib
table = tbl.table(ax, cellText=data_to_display.values, colLabels=data_to_display.columns,
cellLoc='center', loc='center')
# Adjust the text size in the table
table.auto_set_font_size(False)
table.set_fontsize(10)
# Save table as image
plt.savefig('table_image.png')
plt.show() # Show the image
```



Year	Predicted G3
1.0	10.613948365167804
2.0	10.613369504169343
3.0	10.688602389611392
4.0	10.62802540503943
5.0	10.620636891210735

This algorithm loads data from a file.CSV and displays part of it as a table in an image using matplotlib library.

Explaining each part of the code:

1) Import libraries:

pandas: To manage and read data from a CSV file.

matplotlib.pyplot and matplotlib.table: To plot the table and convert the data to an image.

2) Read file CSV:

The path to the file containing the data is specified (file_path) and use pandas to load data from this file using the read_csv function. The data is in a CSV file, and the delimiter between values is specified using the delimiter operator (here I use ;).

3) Specify the number of rows you want to display:

num_rows: Specifies the number of rows to display in the table. Here ten rows are specified, but this number can be modified as needed.

It is used data.head (num_rows) Extracts the first rows of data based on the value specified in num_rows.

4) Preparing the table chart:

A graph is created using plt.subplots() with graph size specified via figsize.

The axes are hidden (ax.axis('tight') and ax.axis('off')) because I just want to display the table without any extra axes or lines.

5) Transformation Data Frame to table in matplotlib:

It uses dtbl.table() converts data (data_to_display.values) and column names (data_to_display.columns) into a table to display in a chart. The position of the table is specified (loc='center') and the alignment of text within cells is set (cellLoc='center').

6) Adjust the text size in the table:

The text size in the table is adjusted using table.auto_set_font_size(False) to disable automatic sizing, then table.set_fontsize(10) to set the font size to 10.

7) Save table as image:

The created table is saved as an image using plt.savefig('table_image.png'). The image is saved to a file named table_image.png.

8) View image:

The table is displayed as an image using plt.show().

Summary: The code reads data from a file.CSV, then displays a specified number of rows as a table using matplotlib. The table is converted to an image and saved as an image file and can be viewed directly after saving.

Conclusion:

This study underscores the vital role of data-driven predictive models in education, emphasizing the need to consider qualitative and quantitative factors in forecasting and assessing student academic performance.

The research introduces Developing academic performance by predicting students' academic level using Naive Bayes, the study highlights a cutting-edge methodology demonstrating how the precision and effectiveness of predictive models can be enhanced through advanced machine learning and optimization algorithms.

The thorough assessment using essential metrics such as Accuracy, Precision, Recall, and F1-score underscore these meta-heuristic algorithms' capability to optimize classification results.

The Naive Bayes model utilized in this analysis offers a straightforward yet effective means of predicting students' final grades (G3) based on a combination of historical performance (G1, G2) and various socio-academic factors (e.g., study time, failures, absences, and family educational background). Linear regression is particularly suitable for this task due to its simplicity, interpretability, and its ability to reveal linear relationships within the data. The algorithm estimates future scores based on current trends, allowing educators and administrators to project academic performance into the coming years.

Key advantages of this approach include:

1. **Efficiency and Speed**: Naive Bayes models are computationally efficient, making them ideal for relatively small datasets, such as educational records, where quick deployment and minimal processing time are beneficial.
2. **Interpretability**: Coefficients in a Naive Bayes model provide direct insight into the influence of each feature on the final grade (G3). For instance, features like study time and failures can be assessed individually, allowing for targeted interventions based on the model's feedback.
3. **Predictive Accuracy with Simple Data**: By leveraging basic student information and historical scores, Naive Bayes provides reasonably accurate predictions. This simplicity makes it easy to apply in real-world educational contexts, where data may be limited or partially incomplete.

4. **Scalability**: The model can be scaled to predict across multiple cohorts or years with minor adjustments. It offers a base for more complex modeling or multi-year projections with added flexibility.

This research represents a significant advancement in predictive modelling within the field of education, presenting promising avenues for improving the precision and efficiency of evaluating academic performance.

References:

- Albreiki B, Zaki N, Alashwal H (2021). "A systematic literature review of student' performance prediction using machine learning techniques". Education Sciences. 11 (9). doi:10.3390/educsci11090552.
- Alwarthan SA, Aslam N, Khan IU (2022). "Predicting student academic performance at higher education using data mining: A systematic review". Applied Computational Intelligence and Soft Computing. 2022. doi:10.1155/2022/8924028.
- Azevedo, R., & Leite, J. P. (2021). Predicting student performance using data mining techniques: A review. International Journal of Artificial Intelligence in Education, 21(3), 235-251.
- Baker, R. S. J. (2021). The use of artificial intelligence in education. Artificial Intelligence, 124(1-2), 1-16.
- Barocas, S., & Nissenbaum, H. (2018). Algorithm accountability. MIT Press.
- Han, J., Kamber, M., & Pei, J. (2011). Data mining: Concepts and techniques (3rd ed.). Morgan Kaufmann.
- Khairy D, et al. (2024). "Prediction of student exam performance using data mining classification algorithms". Education and Information Technologies. :1–25. doi:10.1007/s10639-024-12619-w.
- Kohavi, R. (2019). A study of cross-validation and bootstrap for accuracy estimation. In Proceedings of the 14th International Conference on Machine Learning.
- Kira, K., & Rendell, L. A. (2022). The feature selection problem: Traditional methods and a genetic algorithm approach. In Proceedings of the Fourth International Workshop on Machine Learning.
- Kumar, A., & Singh, S. (2024). Predicting student performance using naive Bayes classifier: A case study. International Journal of Computer Applications, 97(10), 24-27.
- Lam PX, Mai PQH, Nguyen QH, Pham T, Nguyen THH, et al. (2024). "Enhancing educational evaluation through predictive student assessment modeling". Computers and Education: Artificial Intelligence. 6:100244. doi:10.1016/j.caeai.2024.100244.
- Lee, J. J., & Chen, C.-H. (2023). Predicting student performance using naive Bayes classifier. Procedia Computer Science, 17, 101-105.
- Namoun A, Alshanqiti A (2020). "Predicting student performance using data

mining and learning analytics techniques: A systematic literature review". Applied Sciences. 11 (1):237. doi:10.3390/app11010237.

- O'Neill, C. (2016). Weapons of math destruction: How big data is being used to automate discrimination. Broadway Books.
- Pardos, Z. A., & Shahiri, A. M. (2020). Predicting student performance using data mining techniques. International Journal of Computer Science and Network Security, 10(8), 151-158.
- Powers, D. M. (2021). Evaluation metrics for machine learning. In 2021 IEEE 24th International Conference on Tools with Artificial Intelligence (ICTAI).
- Rodrigues, M. W., Isotani, S., & Zárate, L. E. (2018). Educational data mining: A review of the evaluation process in e-learning. Telematics and Informatics, 35(12), 1701-1717.
- Romero, C., & Ventura, S. (2020). Educational data mining: A survey. Journal of Educational Data Mining, 2(1), 1-34.
- Tan, P.-N., Steinbach, M., & Kumar, V. (2016). Introduction to data mining. Addison-Wesley.
- Wang, Y., & Liu, H. (2023). Predicting student performance using naive Bayes classifier: A case study. Procedia Computer Science, 17, 106-110.
- Witten, I. H., & Frank, E. (2022). Data mining: Practical machine learning tools and techniques with Java implementations (6th ed.). Morgan Kaufmann.
- Zheng, X., & Li, C. (2024). Predicting students' academic performance through machine learning classifiers: A study employing the Naive Bayes Classifier (NBC). International Journal of Advanced Computer Science and Applications, 15(1).