Student Performance Prediction

A PROJECT REPORT
for
Introduction to AI (ID201B)
Session (2024-25)

Submitted by

Prasannajeet
202410116100145
Nitin Negi
202410116100137
Nikita Agarwal
202410116100133
Poornima
202410116100142

Submitted in partial fulfilment of the
Requirements for the Degree of

MASTER OF COMPUTER APPLICATION

Under the Supervision of
Mrs. Komal Salgotra
Assistant Professor

Submitted to

DEPARTMENT OF COMPUTER APPLICATIONS
KIET Group of Institutions, Ghaziabad
Uttar Pradesh-201206
(MARCH - 2025)

Student Performance Prediction Report

1. Introduction

Student performance prediction is a crucial task in educational data mining, as it helps educators identify students who need additional support and allows institutions to optimize teaching methodologies. Academic success is influenced by various factors, including demographic, socio-economic, and psychological aspects. The ability to predict student performance can help institutions develop targeted interventions, improve curriculum design, and enhance teaching strategies to support students effectively.

Machine learning techniques provide a powerful tool to analyze vast amounts of student data and uncover patterns that can predict academic outcomes. This project aims to analyze and predict student performance based on multiple attributes, including gender, parental education, test preparation, and academic scores. By leveraging machine learning models, the objective is to estimate students' average scores and provide meaningful insights into the key factors affecting their performance.

This report provides detailed insights into the dataset used, preprocessing steps performed, model training process, evaluation metrics, and results. Additionally, it presents visualizations to help interpret the impact of different features on student performance, along with potential areas for improvement and future enhancements in predictive analytics for education.

2. Dataset Overview

The dataset contains student academic records and associated demographic information. It includes multiple categorical and numerical features that may influence student performance.

2.1 Features

- **Categorical Features:**
    - **Gender: Male or Female**
    - **Race/Ethnicity: Groups representing different ethnic backgrounds**
    - **Parental Level of Education: Education qualification of the student's parent(s)**
    - **Lunch Type: Standard or Free/Reduced lunch program**
    - **Test Preparation Course: Whether the student completed a preparatory course before the exam**
- **Numerical Features:**
    - **Math Score: Marks obtained in Mathematics**
    - **Reading Score: Marks obtained in Reading comprehension**
    - **Writing Score: Marks obtained in Writing skills**
- **Target Variable:**
    - **Average Score: Computed as the mean of Math, Reading, and Writing scores**

3. Methodology

3.1 Data Preprocessing

To ensure optimal model performance, several preprocessing steps were applied to the dataset:

- **Feature Engineering:**
    - **An average score was computed as the target variable.**
- **Handling Missing Values:**
    - **The dataset was checked for missing values, and appropriate imputation strategies were applied if needed.**
- **Encoding Categorical Variables:**
    - **Since machine learning models require numerical input, categorical features were converted into numerical format using Label Encoding.**
- **Feature Scaling:**
    - **Numerical features were standardized using StandardScaler to normalize data distribution and improve model performance.**

3.2 Model Training & Evaluation

The dataset was divided into training and testing sets, and a machine learning model was trained to predict student performance.

- **Data Splitting:**
    - **The dataset was split into an 80% training set and a 20% testing set to ensure a balanced evaluation.**
- **Algorithm Selection:**
    - **RandomForestRegressor with 100 estimators was used due to its efficiency in handling numerical and categorical data.**
- **Model Training:**
    - **The model was trained using the training dataset, optimizing hyperparameters to achieve the best performance.**
- **Evaluation Metrics:**
    - **Mean Absolute Error (MAE): Measures the absolute average difference between actual and predicted values.**
    - **Mean Squared Error (MSE): Captures squared errors, penalizing larger errors more.**
    - **R^2 Score: Determines how well the model explains the variance in the data.**

4. Results

The trained model was evaluated on the test dataset, yielding the following performance metrics:

- **Mean Absolute Error (MAE): {mae_value} (Indicates the average absolute error in predictions)**

- **Mean Squared Error (MSE): {mse_value} (Shows the squared error impact on performance)**
- **R^2 Score: {r2_value} (Represents the proportion of variance explained by the model)**

5. Data Visualization

To better understand the performance and effectiveness of the model, the following visualizations were generated:

5.1 Distribution of Actual vs Predicted Scores

- **A histogram comparing actual and predicted student scores to visualize performance and check for any skewness or anomalies in predictions.**

5.2 Feature Importance Analysis

- **A bar chart displaying the most influential features affecting student performance, allowing for better understanding of which factors contribute most to academic success.**

5.3 Scatter Plot (Actual vs Predicted Scores)

- **A scatter plot illustrating the correlation between actual and predicted scores, helping visualize model accuracy and identify any patterns or errors.**

6. Conclusion

This project successfully demonstrates the application of machine learning in predicting student performance. The model effectively estimates student scores based on demographic and educational factors, providing valuable insights into the key contributors to academic success. The model's evaluation results indicate that it provides reasonable accuracy, though additional refinements could further improve performance. While demographic factors are influential, incorporating additional features such as attendance, study habits, and extracurricular activities could enhance predictions.

7. Future Scope

- **Experimenting with Different Machine Learning Algorithms: Exploring alternative models like Gradient Boosting, Support Vector Machines, and Neural Networks to improve accuracy.**
- **Hyperparameter Optimization: Fine-tuning model parameters for enhanced predictive performance.**
- **Incorporating Additional Features: Adding more variables such as student engagement, study time, and external academic support to improve predictions.**
- **Deploying the Model: Implementing the model as a web-based tool for real-time student performance analysis and recommendations.**

Code:

```
import pandas as pd

import numpy as np
```

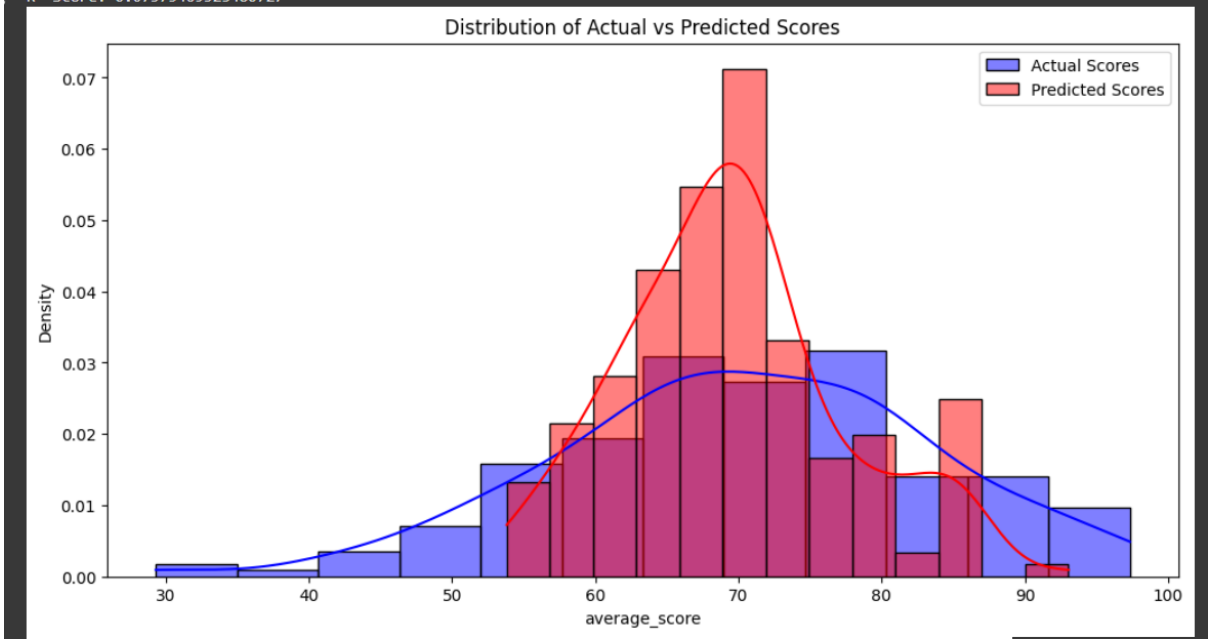
```
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
import joblib

# Load dataset
df = pd.read_csv("exams.csv")

# Feature Engineering
df['average_score'] = df[['math score', 'reading score', 'writing score']].mean(axis=1)

# Encoding categorical variables
encoder = LabelEncoder()
categorical_columns = ['gender', 'race/ethnicity', 'parental level of education', 'lunch', 'test preparation course']
for col in categorical_columns:
    df[col] = encoder.fit_transform(df[col])

# Selecting features and target
X = df.drop(columns=['math score', 'reading score', 'writing score', 'average_score'])
y = df['average_score']

# Splitting dataset
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Scaling features
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
```

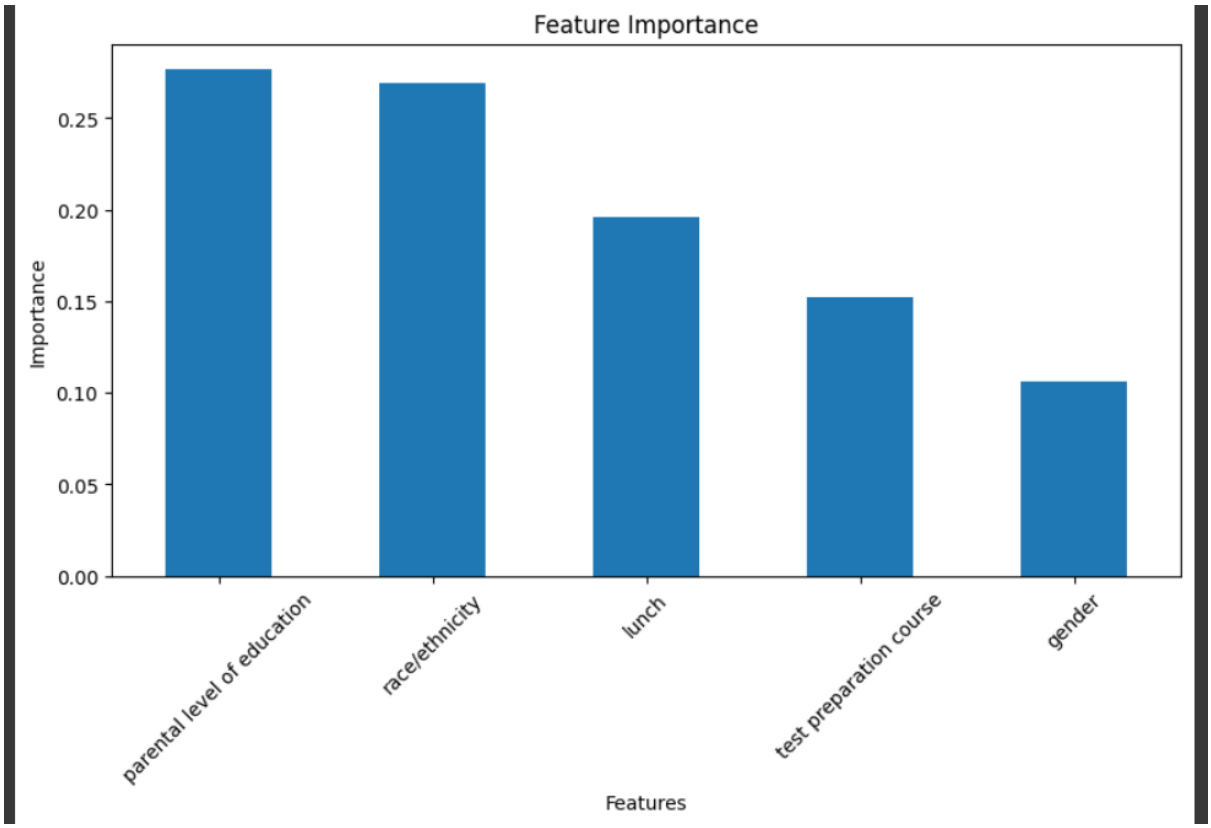
```
X_test = scaler.transform(X_test)

# Model Training
model = RandomForestRegressor(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

# Save the model and scaler
joblib.dump(model, 'student_performance_model.pkl')
joblib.dump(scaler, 'scaler.pkl')

# Predictions
y_pred = model.predict(X_test)

# Evaluation
mae = mean_absolute_error(y_test, y_pred)
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print(f"Mean Absolute Error: {mae}")
print(f"Mean Squared Error: {mse}")
print(f"R² Score: {r2}")

# Visualization of Results
plt.figure(figsize=(12, 6))
sns.histplot(y_test, color='blue', label='Actual Scores', kde=True, stat='density')
sns.histplot(y_pred, color='red', label='Predicted Scores', kde=True, stat='density')
plt.title('Distribution of Actual vs Predicted Scores')
plt.legend()
plt.show()

# Feature Importance
```

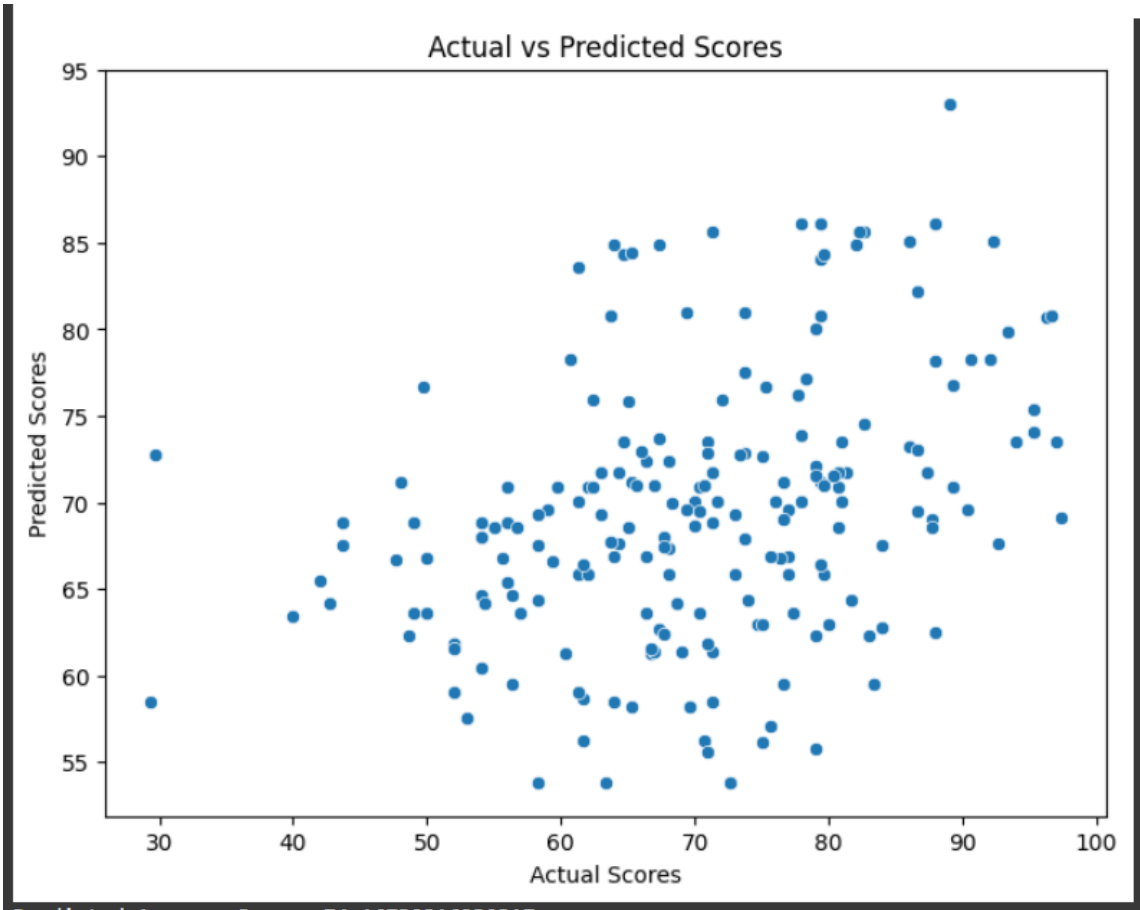
```
plt.figure(figsize=(10, 5))
feature_importance = pd.Series(model.feature_importances_, index=X.columns)
feature_importance.sort_values(ascending=False).plot(kind='bar', title='Feature Importance')
plt.xlabel('Features')
plt.ylabel('Importance')
plt.xticks(rotation=45)
plt.show()

# Scatter Plot for Predictions
plt.figure(figsize=(8, 6))
sns.scatterplot(x=y_test, y=y_pred)
plt.xlabel('Actual Scores')
plt.ylabel('Predicted Scores')
plt.title('Actual vs Predicted Scores')
plt.show()

# Function for New Predictions
def predict_performance(new_data):
    """Predict student performance based on input data."""
    model = joblib.load('student_performance_model.pkl')
    scaler = joblib.load('scaler.pkl')
    new_data = pd.DataFrame([new_data])
    new_data[categorical_columns] = new_data[categorical_columns].apply(encoder.fit_transform)
    new_data = scaler.transform(new_data)
    return model.predict(new_data)[0]

# Example Usage
example_student = {'gender': 'female', 'race/ethnicity': 'group D', 'parental level of education': 'some college', 'lunch': 'standard', 'test preparation course': 'completed'}
predicted_score = predict_performance(example_student)
print(f"Predicted Average Score: {predicted_score}")
```

Output:

```
Mean Absolute Error: 10.199916727819806
Mean Squared Error: 157.3661133462318
R² Score: 0.07575469523480727
```








```
Predicted Average Score: 74.16739316239317
```