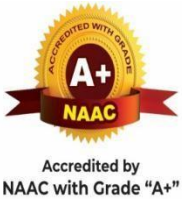


Department of Computer Application

STUDENT PERFORMANCE PREDICTION

A PROJECT REPORT
FOR
INTRODUCTION TO AI(AI101B)
SESSION(2024-25)

SUBMITTED BY
SHANTANU YADAV(202410116100193)
SHIVAM CHATURVEDI(202410116100198)
SONAM DOBRIYAL(202410116100210)
VINEET KUMAR(202410116100246)

Submitted to
KOMAL SALGOTRA
ASS. PROFESSOR

Department of Computer Application

INTRODUCTION

In today's fast-evolving educational landscape, predicting student performance using artificial intelligence (AI) has become an essential tool for educators and institutions. This project focuses on developing a machine learning model that analyzes various academic and non-academic factors to predict student outcomes accurately. Key parameters such as attendance, past academic records, assignment scores, participation in extracurricular activities, study habits, and socio-economic background play a significant role in determining a student's academic success. By leveraging AI-driven predictive analytics, this project aims to identify students who may be at risk of underperformance, allowing educators to take proactive measures such as personalized mentoring, targeted interventions, and curriculum adjustments.

The project follows a systematic approach, starting with data collection and preprocessing to ensure the accuracy and reliability of the dataset. Feature selection techniques will be applied to identify the most influential factors affecting student performance. Machine learning algorithms, including regression models, decision trees, and neural networks, will be trained and evaluated to determine the most effective model for prediction. Performance metrics such as accuracy, precision, recall, and F1-score will be used to assess the model's effectiveness.

By implementing AI in student performance prediction, this project aims to enhance the educational system by providing valuable insights to teachers, parents, and administrators. The ability to predict academic outcomes can lead to improved student support, better resource allocation, and more informed decision-making. Ultimately, this project contributes to the broader goal of using technology to foster academic excellence and ensure that every student receives the guidance they need to succeed.

Department of Computer Application

METHODOLOGY

1. **Data Collection**: Gather student data, including academic records, attendance, assignments, extracurricular activities, and socio-economic factors.
2. **Data Preprocessing**: Clean the dataset by handling missing values, normalizing numerical data, and encoding categorical variables.
3. **Feature Selection & Engineering**: Identify key factors influencing performance using statistical techniques and create meaningful features for better prediction.
4. **Model Selection & Training:** Train various machine learning models such as Linear Regression, Decision Trees, Random Forests, and Neural Networks using an 80-20 train-test split.
5. **Model Evaluation**: Assess model performance using accuracy, precision, recall, F1-score, and cross-validation techniques.
6. **Optimization & Tuning**: Improve model accuracy by tuning hyperparameters using Grid Search and applying regularization techniques.
7. **Continuous Improvement**: Regularly update the model with new data and feedback to enhance accuracy and effectiveness over time.

This structured approach ensures accurate student performance predictions, enabling educators to take proactive measures for better academic outcomes.

CODE

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
df= pd.read_csv('/content/student_exam_data.csv')
df.head()



df.dtypes
```

	0
Study Hours	float64
Previous Exam Score	float64
Pass/Fail	int64

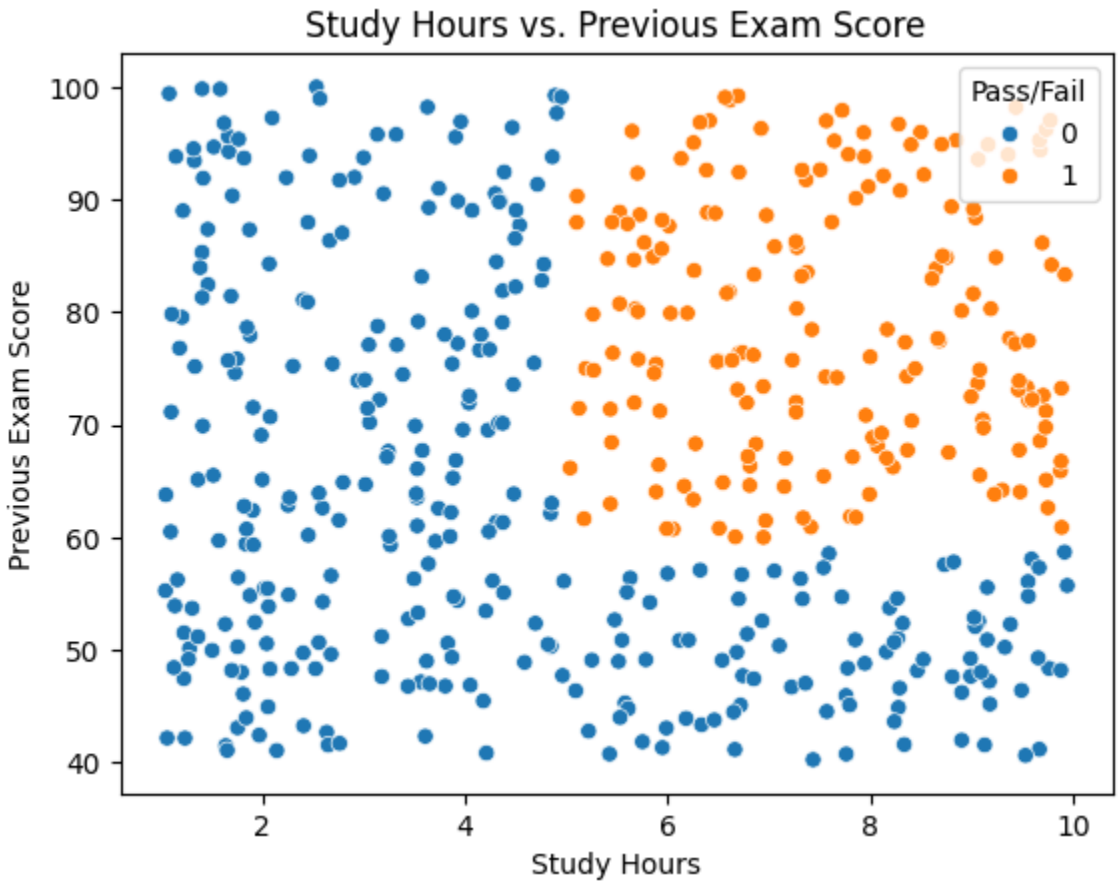
	Study Hours	Previous Exam Score	Pass/Fail
0	4.370861	81.889703	0
1	9.556429	72.165782	1
2	7.587945	58.571657	0
3	6.387926	88.827701	1
4	2.404168	81.083870	0

dtype: object

```
print("Sum of null values: ")
print(df.isnull().sum())
print("Percentage of null values: ")
print(df.isnull().sum() * 100 / len(df))
```

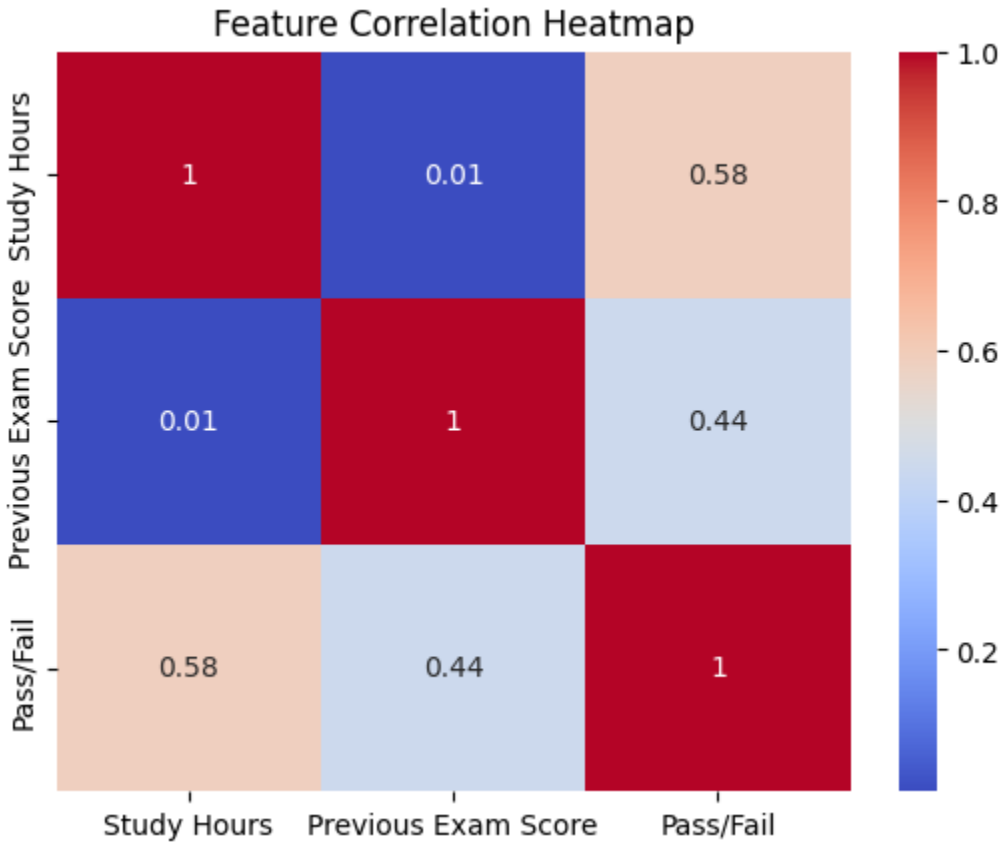
Sum of null values: Study Hours 0 Previous Exam Score 0 Pass/Fail 0 dtype: int64 Percentage of null values: Study Hours 0.0 Previous Exam Score 0.0 Pass/Fail 0.0 dtype: float64

```
sns.scatterplot(x='Study Hours', y='Previous Exam Score',
hue='Pass/Fail', data=df)
plt.title('Study Hours vs. Previous Exam Score')
plt.xlabel('Study Hours')
plt.ylabel('Previous Exam Score')
plt.show()
```

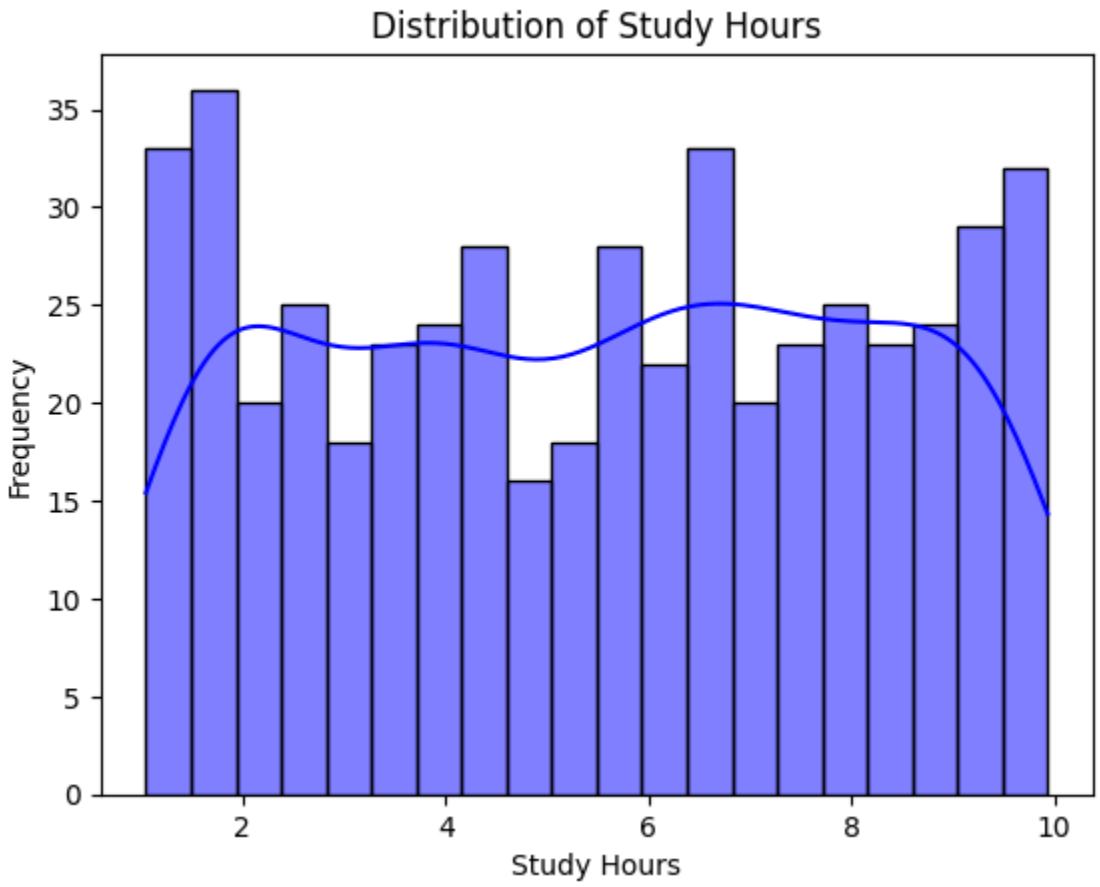



```
import seaborn as sns
import matplotlib.pyplot as plt
correlation_matrix = df.corr()
```

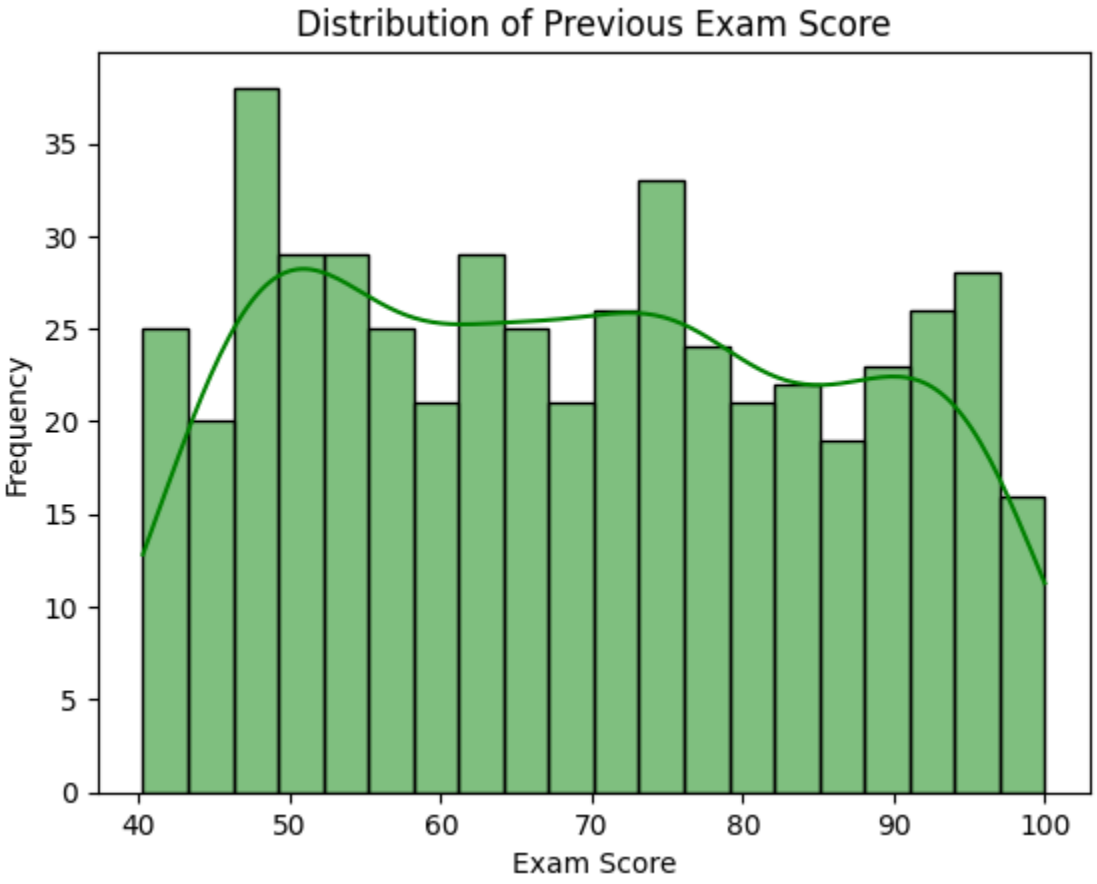
```
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm')
plt.title("Feature Correlation Heatmap")
plt.show()
```



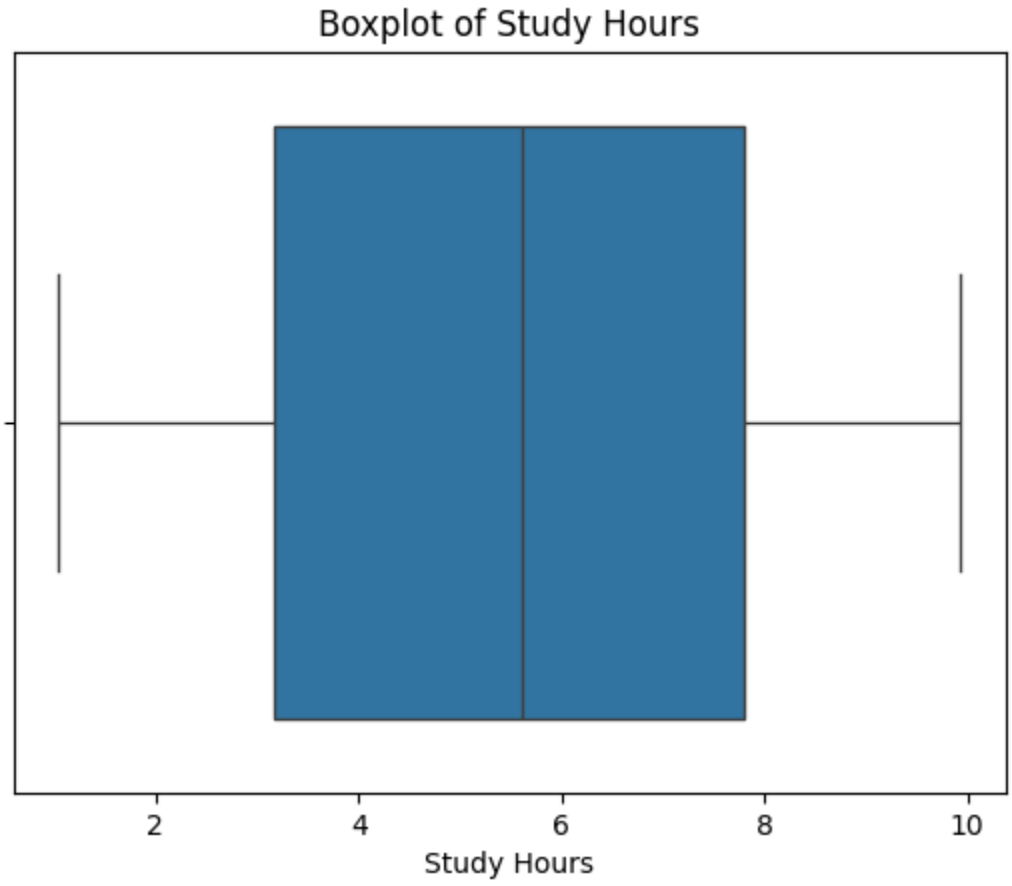

```
sns.histplot(df['Study Hours'], bins=20, kde=True,
color='blue')
plt.title('Distribution of Study Hours')
plt.xlabel('Study Hours')
plt.ylabel('Frequency')
plt.show
```



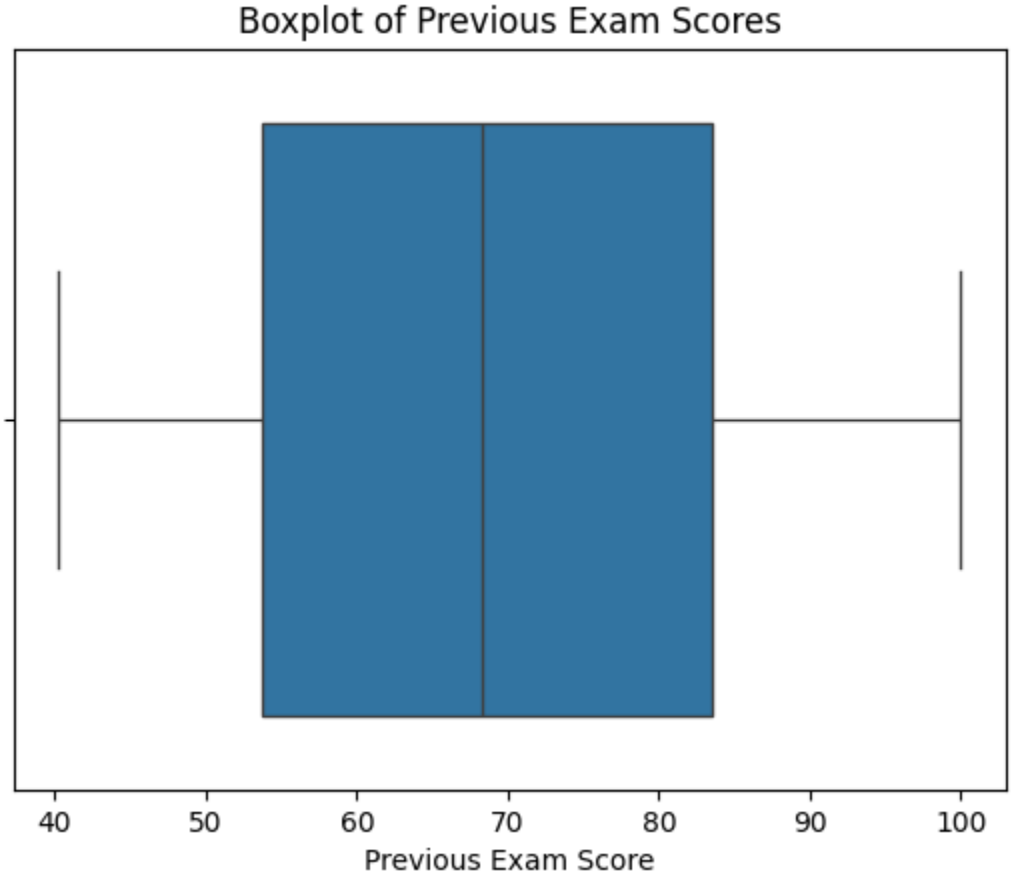

```
sns.histplot(df['Previous Exam Score'], bins=20, kde=True,
color='green')
plt.title('Distribution of Previous Exam Score')
plt.xlabel('Exam Score')
plt.ylabel('Frequency')
plt.show()
```




```
sns.boxplot(x=df['Study Hours'])
plt.title('Boxplot of Study Hours')
plt.show()
```




```
sns.boxplot(x=df['Previous Exam Score'])
plt.title('Boxplot of Previous Exam Scores')
plt.show()
```




```
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

X = df[['Study Hours', 'Previous Exam Score']]
y = df['Pass/Fail']
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)
```

```
rf_model = RandomForestClassifier(n_estimators=100,
random_state=42)
rf_model.fit(X_train, y_train)
```

```
y_pred = rf_model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
print(f"Model Accuracy: {accuracy * 100:.2f}%")
```

```
study_hours = float(input("Enter study hours: "))
previous_exam_score = float(input("Enter previous exam score:
"))
prediction = rf_model.predict([[study_hours,
previous_exam_score]])
result = "Pass" if prediction[0] == 1 else "Fail"
print(f"Predicted Result: {result}")
```

Model Accuracy: 99.00% Enter study hours: 20 Enter previous exam score: 99 Predicted Result: Pass
/usr/local/lib/python3.11/dist-packages/sklearn/utils/validation.py:2739: UserWarning: X does not have valid feature names, but RandomForestClassifier was fitted with feature names warnings.warn()

NIRF-2024
Engineering Rank Band (151-200)
Pharmacy Rank - 77
Innovation Rank Band (11-50)

www.kiet.edu
Delhi-NCR, Ghaziabad

KIET GROUP OF INSTITUTIONS
Connecting Life with Learning

Accredited by
NAAC with Grade "A+"

Department of Computer Application