Project Report: Student Performance Prediction

NAME: SIDH AGARWAL

BRANCH:CSE-AI

SEC: D

UNIV. ROLL NO: 202401100300247

DATE: 11 MARCH 2025

2. Data Collection

2.1 Dataset Description

The dataset used for this project is a CSV file
named student_data.csv, which contains the following columns:

- study_hours: Number of hours a student studies per week.
- attendance: Percentage of classes attended.
- previous_scores: Average scores from previous exams.
- target: Final exam score (or pass/fail status).

2.2 Data Source

The dataset was collected from student_data.csv. It contains records of 20 students.

LIBRARY USED:

PANDAS:

Pandas overview is an open-source Python data analysis and manipulation tool. It offers tools and data structures required for flawless working with structured data. Pandas mostly uses Series (1-dimensional) and DataFrame (2-dimensional) data structures that let one easily handle tabular data.

NUMpy:

A basic tool for scientific computing in Python is NumPy, sometimes known as Numerical Python. Along with a set of mathematical operations to run on big, multi-dimensional arrays and matrices, it supports these structures.

4. Methodology

4.1 Model Selectivity

This project used a Random Forest Regressor because of its resilience and capacity to manage non-linear interactions**.**

4.2 Model Learning

The training dataset helped the model to be developed. Although hyperparameters were left at default values, additional adjustment is possible for best performance.

4.3 Model Assessment

Performance of the model was assessed in respect to the following criteria:

R^2 Score; Mean Squared Error, or MSE

CODE:

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

# Load dataset
file_path = "/mnt/data/student_data.csv"
df = pd.read_csv('student_data.csv')

# Display basic info about dataset
print(df.head())

# Feature selection (excluding StudentID)
features = ['StudyHours', 'PreviousScores']
X = df[features]
y = df['FinalExamScore']

# Split 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)
```

```
# Train the model
model = RandomForestRegressor(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

# Predict on test set
y_pred = model.predict(X_test)

# Evaluate the model
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}")
```

OUTPUT:

```
   StudentID  StudyHours  PreviousScores  FinalExamScore
0          1    8.777482              75              64
1          2    9.161915              55              82
2          3    3.278010              77              70
3          4    4.500247              60              60
4          5    2.264931              72              60
Mean Absolute Error: 20.134999999999998
Mean Squared Error: 574.4306499999999
R² Score: -1.1026974147792266
```

REFERENCE:

1. student_data.csv file for dataset

2. https://pandas.pydata.org/pandas-docs/stable/

3.https://numpy.org/doc/stable/