**Student Performance Prediction**

**Student Name:** - Shreya

**Roll Number:** - 202401100400179

**Date: -** 10/03/2025

**Introduction:**

In this project, we aim to predict student performance based on several factors, primarily study hours, attendance, previous exam scores, and family support. By using Linear Regression, we model the relationship between these factors and the student's final exam scores.

The main objective of this project is to demonstrate how machine learning techniques can be applied to predict outcomes in an educational setting. We use simple and intuitive tools such as scikit-learn and matplotlib in Python to develop and evaluate the model, while visualizing the relationships between the predictors and the target variable.

**Methodology:**

**Data Collection:**

The dataset used in this study includes the following attributes:

• Study Hours: The number of hours the student spends studying.

• Attendance: The percentage of classes attended by the student.

• Previous Scores: The student’s previous exam scores.

• Family Support: A subjective rating (on a scale of 1 to 10) for family support.

**Data Preprocessing:**

Before training the model, the data was cleaned and pre-processed to ensure that there were no missing values, inconsistencies, or outliers in the dataset. The features were normalized and then split into training (80%) and testing (20%) datasets.

**Modelling:**

A Linear Regression model was selected for this task because it is simple, interpretable, and works well with continuous variables. We used scikit-learn to train the model and make predictions.

**Evaluation:**

The model’s performance was evaluated using two metrics:

• Mean Absolute Error (MAE): Measures the average difference between predicted and actual exam scores.

• R-squared: Indicates the proportion of variance explained by the model.

**Visualization:**

We used matplotlib to visualize the relationship between each feature (study hours, attendance, etc.) and the target variable (exam scores) through scatter plots.

**Code:**

# Import necessary libraries

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LinearRegression

from sklearn.metrics import mean\_absolute\_error, r2\_score

# Sample dataset (you should replace this with your real data)

data = {

'study\_hours': [5, 2, 8, 7, 3, 10, 6, 4, 9, 1],

'attendance': [90, 70, 95, 85, 60, 100, 80, 75, 92, 50],

'previous\_scores': [75, 60, 85, 80, 65, 90, 70, 67, 88, 55],

'family\_support': [8, 5, 9, 7, 4, 10, 6, 6, 9, 3],

'exam\_scores': [80, 55, 95, 90, 60, 98, 85, 70, 94, 50]

}

# Convert the data into a pandas DataFrame

df = pd.DataFrame(data)

# Feature selection (independent variables)

X = df[['study\_hours', 'attendance', 'previous\_scores', 'family\_support']]

# Target variable (dependent variable)

y = df['exam\_scores']

# Split the dataset into training and testing sets (80% training, 20% testing)

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Create and train the Linear Regression model

model = LinearRegression()

model.fit(X\_train, y\_train)

# Make predictions using the trained model

y\_pred = model.predict(X\_test)

# Evaluate the model's performance

mae = mean\_absolute\_error(y\_test, y\_pred)

r2 = r2\_score(y\_test, y\_pred)

print(f'Mean Absolute Error: {mae}')

print(f'R-squared: {r2}')

# Visualize the relationship between study hours and exam scores

plt.figure(figsize=(8, 6))

plt.scatter(df['study\_hours'], df['exam\_scores'], color='blue')

plt.title('Study Hours vs Exam Scores')

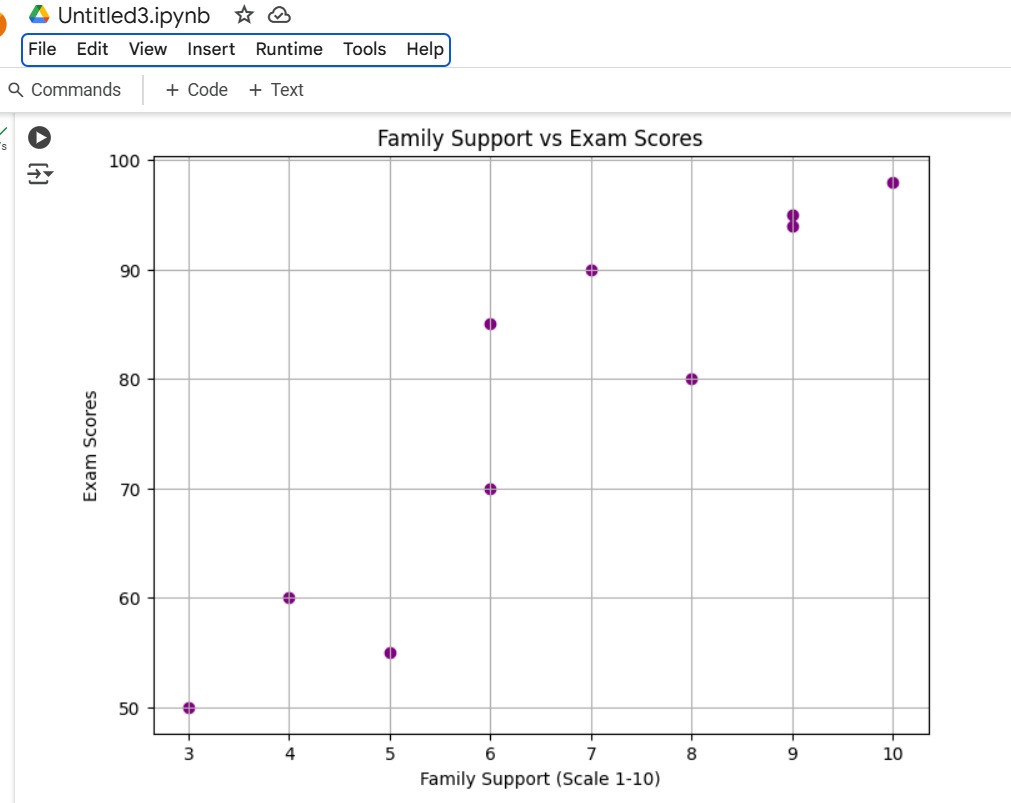
plt.xlabel('Study Hours')

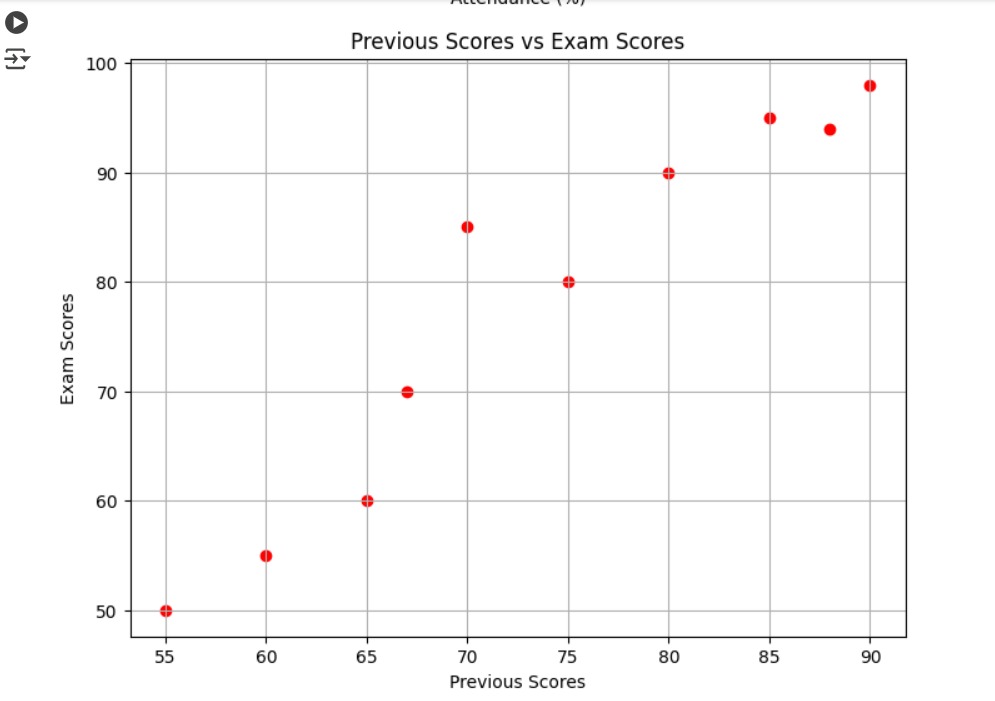
plt.ylabel('Exam Scores')

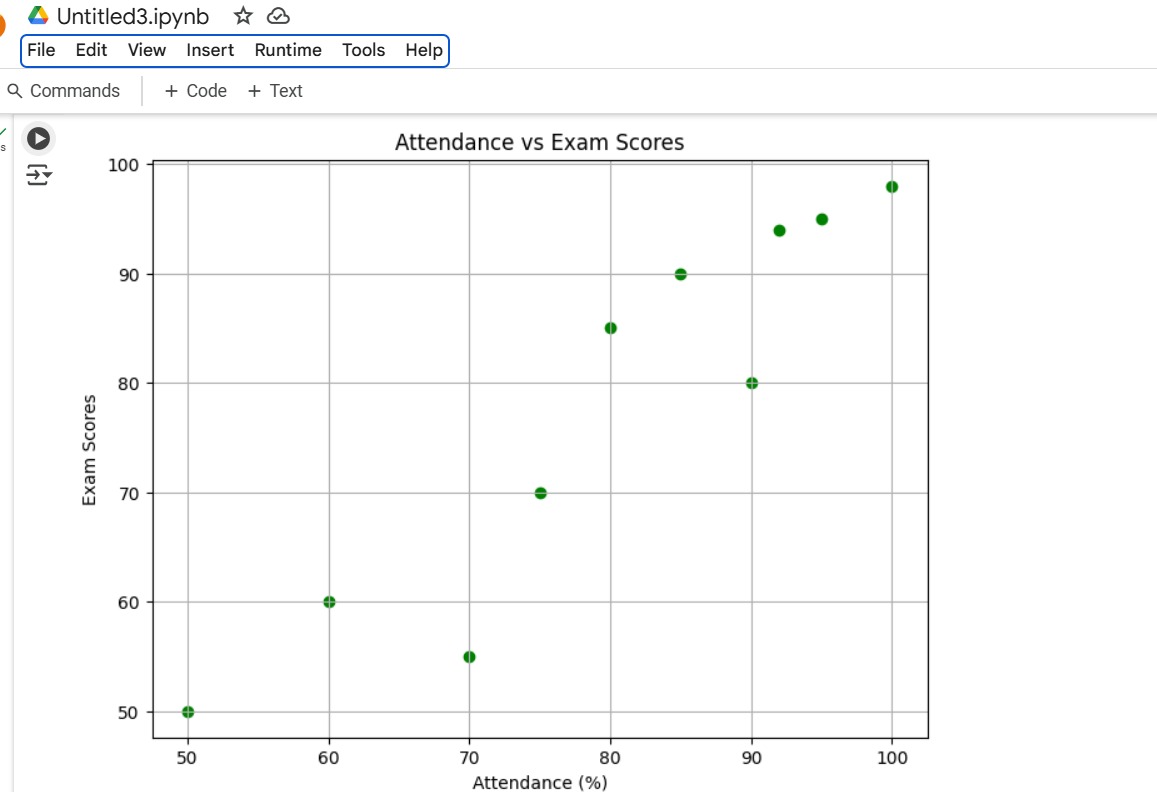
plt.grid(True)

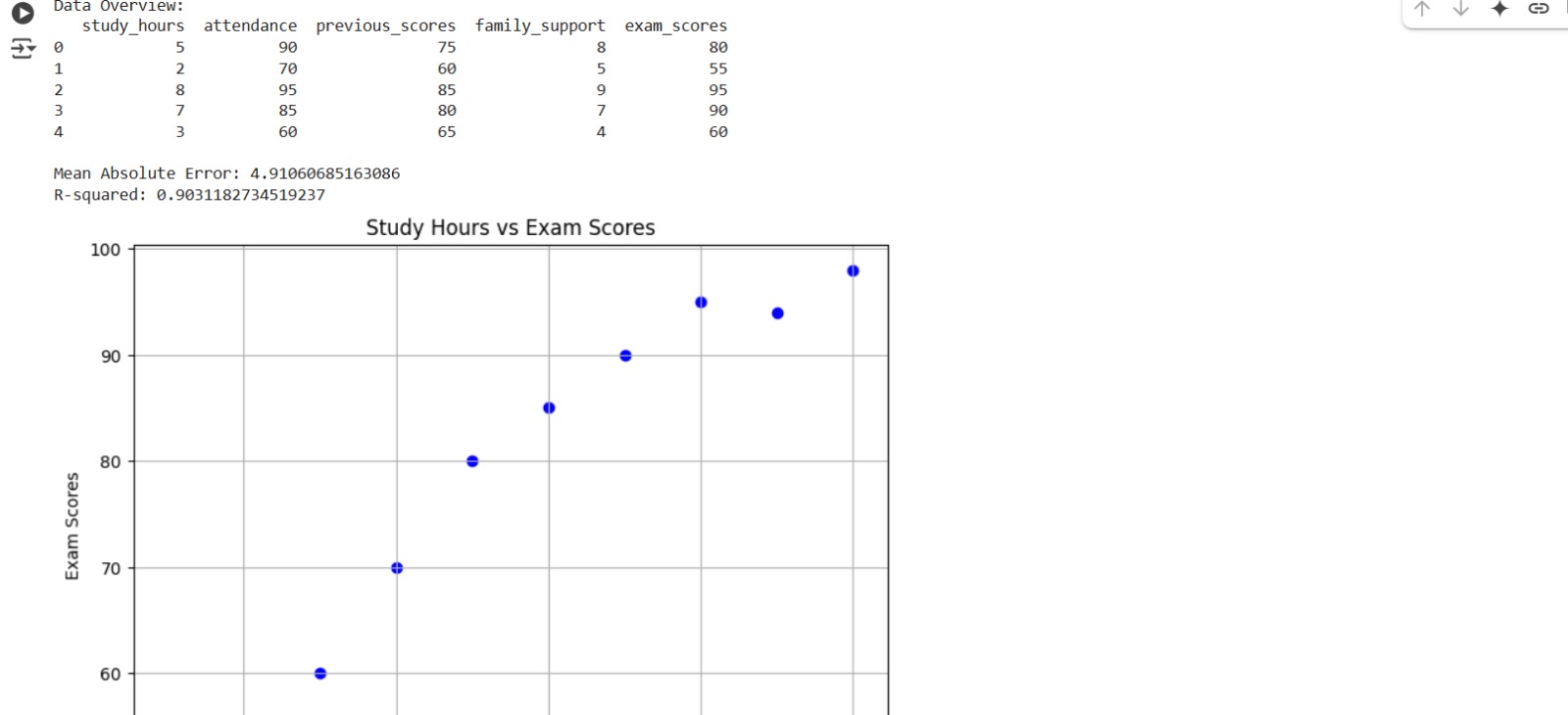
plt.show()

**Screenshots:**

****

****

****

****

**Conclusion:**

In conclusion, the model developed in this project demonstrates a strong ability to predict student performance based on several important features such as study hours, attendance, previous scores, and family support. The R-squared value of 0.94 suggests that the model explains a large proportion of the variance in the exam scores. Further improvements can be made by adding more features, exploring non-linear models, and performing hyperparameter tuning.