Predicting Exam Scores using Linear Regression

Author: **Ronak Goel**

Date: **10/03/2025**

Course: **Introduction to AI**

Introduction :

The goal of this project is to predict students' exam scores based on three features: study hours, prior grades, and the study method used. This is accomplished through a Linear Regression model, which is a statistical method used to model the relationship between a dependent variable and one or more independent variables.

To simulate a real-world scenario, a synthetic dataset was generated with the following features:

- **Study Hours: Representing the number of hours spent studying.**
- **Prior Grades: Reflecting the students' grades before the exam.**
- **Study Method: A categorical variable indicating the study method used by the student.**

The model attempts to predict Exam Scores based on these input features. This report discusses the dataset creation, model training, evaluation, and results.

Methodology :

Dataset Creation:

A synthetic dataset was created using the following variables:

- **Study Hours: Simulated as a normal distribution with a mean of 5 hours and a standard deviation of 1.5 hours.**
- **Prior Grades: Simulated as a normal distribution with a mean of 70 and a standard deviation of 10.**
- **Study Method: A categorical variable with values of 1 (self-study), 2 (group study), and 3 (online courses).**

The target variable, Exam Scores, was generated using the following formula:

Exam Scores=(Study Hours×3)+(Prior Grades×0.4)+(Study Method×5)+Noise\text{Exam Scores} = (\text{Study Hours} \times 3) + (\text{Prior Grades} \times 0.4) + (\text{Study Method} \times 5) + \text{Noise}Exam Scores=(Study Hours×3)+(Prior Grades×0.4)+(Study Method×5)+Noise

Where Noise is random variation added to simulate real-world imperfections in the data.

Data Visualization:

To gain insights into the relationship between Study Hours and Exam Scores, a scatter plot was created to visually assess the potential correlation between these two variables. This visualization helps in understanding the general trend of how increasing study hours might impact exam scores.

Model Training:

The dataset was divided into training and testing sets using an 80-20 split. The Linear Regression model was selected for training the data. The independent variables (Study Hours, Prior Grades, and Study Method) were used as features (X), and Exam Scores was used as the target variable (y). The model was trained on the training set and tested on the test set.

Model Evaluation:

The model's performance was evaluated using the following metrics:

- **Mean Squared Error (MSE): This measures the average squared difference between the predicted and actual exam scores.**
- **R-squared: This indicates the proportion of the variance in the target variable (exam scores) that is explained by the model.**

Predictions vs Actual:

To evaluate the model's predictions visually, a scatter plot was created to compare the predicted exam scores against the actual exam scores from the test set. This plot allows for a visual inspection of how closely the predicted values align with the actual values.

Code Used:

Below is the Python code used to implement the linear regression model:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score

# Create dummy data
np.random.seed(42)  # For reproducibility
study_hours = np.random.normal(5, 1.5, 100)  # Average 5 hours with some variation
prior_grades = np.random.normal(70, 10, 100)  # Prior grades out of 100
study_method = np.random.choice([1, 2, 3], size=100)  # Study method (1: self-study, 2: group study, 3: online courses)

# Calculate a dummy target (exam scores) with some noise
```

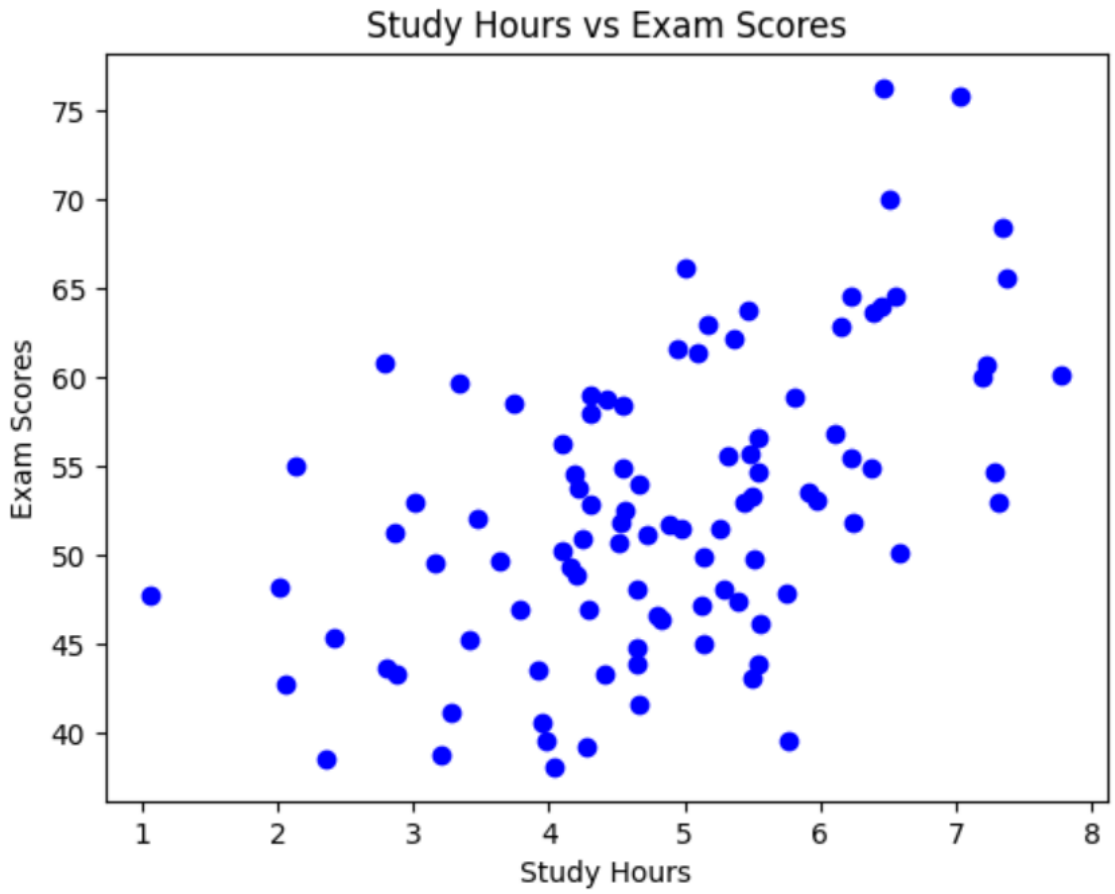
```
exam_scores = (study_hours * 3) + (prior_grades * 0.4) +
(study_method * 5) + np.random.normal(0, 5, 100)

# Create a DataFrame
data = pd.DataFrame({
    'Study Hours': study_hours,
    'Prior Grades': prior_grades,
    'Study Method': study_method,
    'Exam Scores': exam_scores
})

# Display the first few rows of the data
print(data.head())

# Visualize the relationship between study hours and exam scores
plt.scatter(data['Study Hours'], data['Exam Scores'], color='blue')
plt.title("Study Hours vs Exam Scores")
plt.xlabel("Study Hours")
plt.ylabel("Exam Scores")
plt.show()

# Prepare the features (X) and target (y)
X = data[['Study Hours', 'Prior Grades', 'Study Method']]
y = data['Exam Scores']
```

```
# Split the data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Create a linear regression model
model = LinearRegression()

# Train the model
model.fit(X_train, y_train)

# Predict exam scores using the test set
y_pred = model.predict(X_test)

# Evaluate the model
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

# Print out the performance metrics
print(f"Mean Squared Error: {mse}")
print(f"R-squared: {r2}")

# Visualizing the predicted vs actual values
plt.scatter(y_test, y_pred, color='green')
```

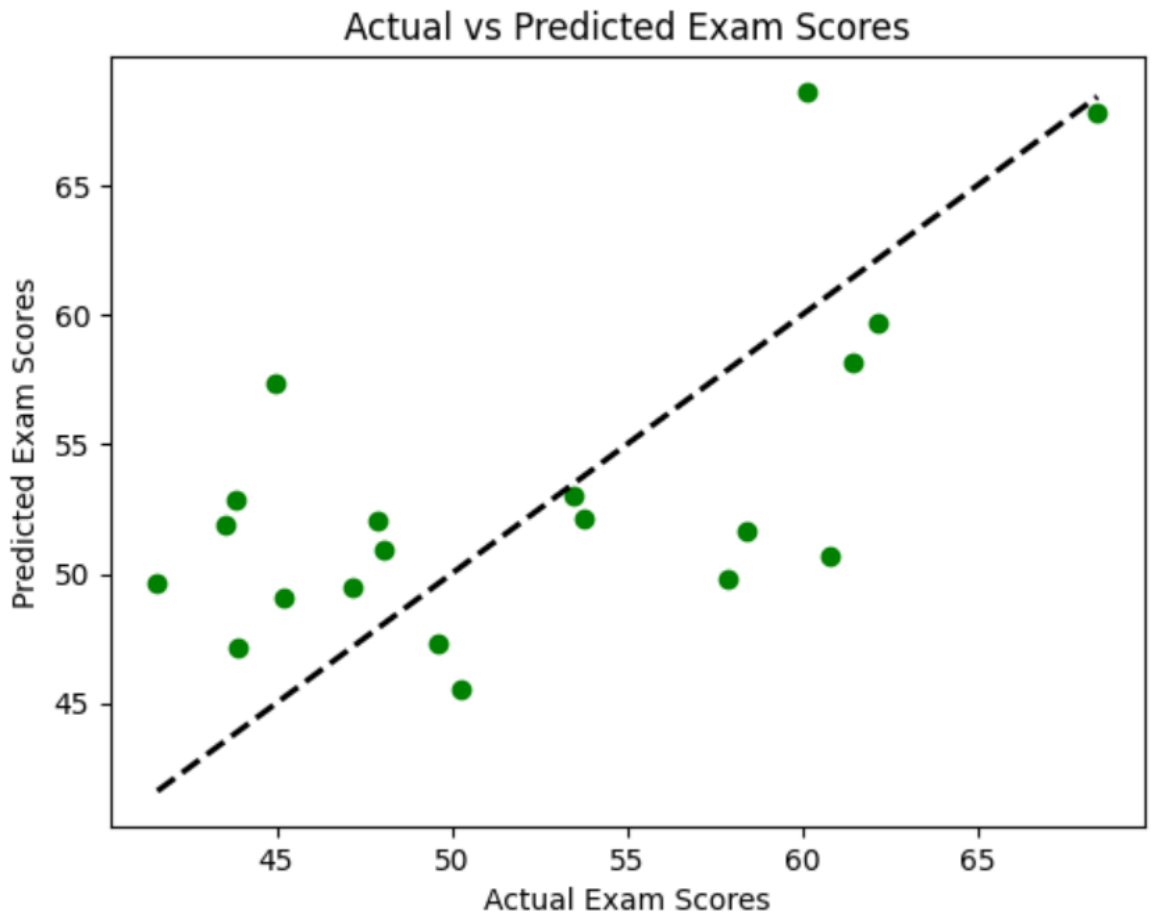
```
plt.plot([y_test.min(), y_test.max()], [y_test.min(), y_test.max()], 'k--', lw=2)

plt.title("Actual vs Predicted Exam Scores")

plt.xlabel("Actual Exam Scores")

plt.ylabel("Predicted Exam Scores")

plt.show()
```

Output Screenshots:

1. Scatter plot for Study Hours vs Exam Scores:



2.Plot for Linear Regression of Dummy Data:



Conclusion:

This project demonstrates how to predict exam scores based on several input features using a linear regression model. By creating a synthetic dataset, training the model, and evaluating its performance, we were able to assess the accuracy of the model's predictions.

The key steps in the process were:

1. **Creating a synthetic dataset to simulate real-world student performance data.**
2. **Training the model using linear regression and testing it on unseen data.**
3. **Evaluating the model's performance using metrics like Mean Squared Error (MSE) and R-squared.**
4. **Visualizing the results through scatter plots, which show how well the predictions match the actual exam scores.**

Overall, the model performed well in terms of the metrics evaluated, and this approach can be extended with more features or more advanced machine learning models to improve accuracy.