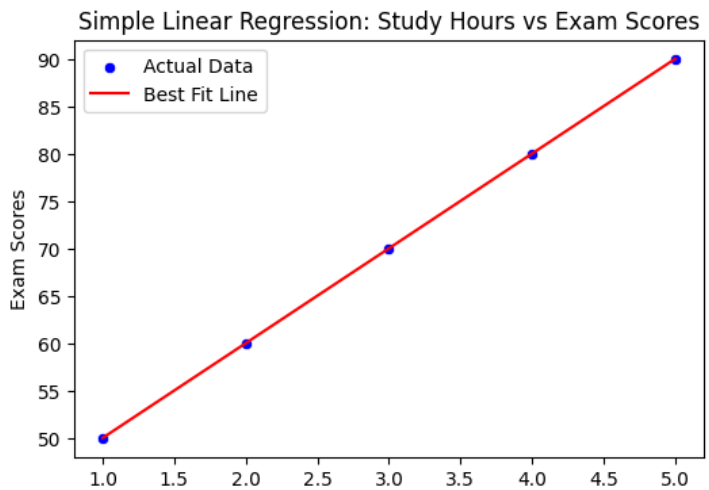
```
"""
Python Tutorial: Regression Analysis

Topics Covered:
1. Simple Linear Regression Model
2. Multiple Linear Regression Model
3. Best Fitting Line
4. Slope of the Regression Line
5. Regression Coefficients
6. Error Estimation
"""


import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score


# Section 1: Simple Linear Regression
print("Section 1: Simple Linear Regression")
study_hours = np.array([1, 2, 3, 4, 5]).reshape(-1, 1)
exam_scores = np.array([50, 60, 70, 80, 90])

model = LinearRegression()
model.fit(study_hours, exam_scores)

predictions = model.predict(study_hours)
print(f"Slope: {model.coef_[0]:.2f}")
print(f"Intercept: {model.intercept_:.2f}\n")
```

```
Section 1: Simple Linear Regression
Slope: 10.00
Intercept: 40.00
```

```
# Plot Best Fitting Line
plt.figure(figsize=(6, 4))
sns.scatterplot(x=study_hours.flatten(), y=exam_scores, color='blue', label='Actual Data')
plt.plot(study_hours, predictions, color='red', label='Best Fit Line')
plt.xlabel("Study Hours")
plt.ylabel("Exam Scores")
plt.title("Simple Linear Regression: Study Hours vs Exam Scores")
plt.legend()
plt.show()
```




```
# Section 2: Multiple Linear Regression
print("Section 2: Multiple Linear Regression")
data = pd.DataFrame({
    'Study Hours': [1, 2, 3, 4, 5],
    'Practice Tests': [1, 1, 2, 2, 3],
```

```
    'Exam Scores': [50, 60, 70, 80, 90]
})
X = data[['Study Hours', 'Practice Tests']]
y = data['Exam Scores']

multi_model = LinearRegression()
multi_model.fit(X, y)
pred_multi = multi_model.predict(X)

print(f"Coefficients: {multi_model.coef_}")
print(f"Intercept: {multi_model.intercept_}\n")
```

```
Section 2: Multiple Linear Regression
Coefficients: [1.00000000e+01 4.99424994e-15]
Intercept: 39.999999999999986
```

```
# Section 3: Error Estimation
print("Section 3: Error Estimation")
mse = mean_squared_error(y, pred_multi)
r2 = r2_score(y, pred_multi)
print(f"Mean Squared Error: {mse:.2f}")
print(f"R-Squared: {r2:.2f}\n")
```

```
Section 3: Error Estimation
Mean Squared Error: 0.00
R-Squared: 1.00
```