```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_absolute_error, mean_squared_error,root_mean_squared_error, r2_score
```

```
X = np.array([[1],[2],[3],[4],[5]])
y = np.array([20, 40, 50, 65, 80])
```

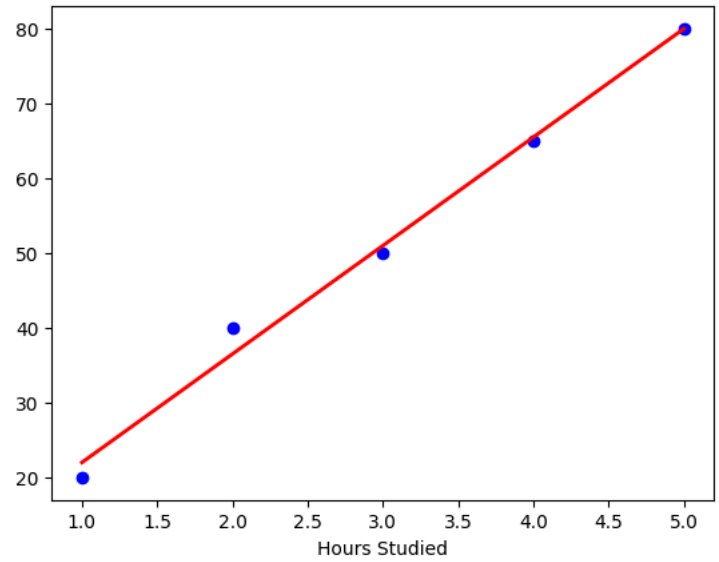
```
model = LinearRegression()
model.fit(X,y)
```

```
▾ LinearRegression
LinearRegression()
```

```
y_pred = model.predict(X)
```

```
plt.scatter(X, y, color='blue',label="Actual Data")
plt.plot(X,y_pred,color='red',linewidth=2,label="Regression Line")
plt.xlabel("Hours Studied")
```

```
Text(0.5, 0, 'Hours Studied')
```




```
mae = mean_absolute_error(y, y_pred)
mse = mean_squared_error(y, y_pred)
rmse = np.sqrt(mse)
```

```
print("Mean Absolute Error (MAE):",mae)
print("Mean Squared Error (MSE):",mse)
print("Root Mean Squared Error (MAE):",rmse)
```

```
Mean Absolute Error (MAE): 1.3999999999999992
Mean Squared Error (MSE): 3.5000000000000013
Root Mean Squared Error (MAE): 1.870828693386971
```

```
r2 = r2_score(y,y_pred)
print(f"R^2 Score: {r2:.2f}")
```

```
R^2 Score: 0.99
```

```
mae = mean_absolute_error(y,y_pred)
print(f"Mean Abolute Error (MAE): {mae:.2f}")
```

```
Mean Abolute Error (MAE): 1.40
```

```
mse = mean_squared_error(y,y_pred)
print(f"Mean Squared Error (MSE): {mse:.2f}")
```