

```

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_squared_error, r2_score

df = pd.read_csv("/content/Salary_dataset.csv")
df

{
  "summary": {
    "name": "df",
    "rows": 30,
    "fields": [
      {
        "column": "Unnamed: 0",
        "properties": {
          "dtype": "number",
          "std": 8,
          "min": 0,
          "max": 29,
          "num_unique_values": 30,
          "samples": [27, 15, 23]
        }
      },
      {
        "column": "YearsExperience",
        "properties": {
          "dtype": "number",
          "std": 2.8378881576627184,
          "min": 1.2000000000000002,
          "max": 10.6,
          "num_unique_values": 28,
          "samples": [4.0, 9.7, 3.8]
        }
      },
      {
        "column": "Salary",
        "properties": {
          "dtype": "number",
          "std": 27414.4297845823,
          "min": 37732.0,
          "max": 122392.0,
          "num_unique_values": 30,
          "samples": [112636.0, 67939.0, 113813.0]
        }
      }
    ],
    "type": "dataframe",
    "variable_name": "df"
  }
}

X = df[['YearsExperience']]
y = df['Salary']

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)

from sklearn.linear_model import LinearRegression

model = LinearRegression()
model.fit(X_train, y_train)
LinearRegression()

y_pred = model.predict(X_test)

from sklearn.metrics import mean_squared_error

```

```

mse = mean_squared_error(y_test, y_pred)
print("Mean Squared Error:", mse)

Mean Squared Error: 49830096.855908394

from sklearn.metrics import r2_score

r2 = r2_score(y_test, y_pred)
print("R-squared value:", r2)

R-squared value: 0.9024461774180497

plt.scatter(X, y, color='blue')
plt.plot(X, model.predict(X), color='red')
plt.xlabel("Years of Experience")
plt.ylabel("Salary")
plt.title("Salary vs Experience")
plt.show()

```



```

plt.scatter(y_test, y_pred)
plt.xlabel("Actual Salary")
plt.ylabel("Predicted Salary")

```

```
plt.title("Actual vs Predicted Salary")
plt.show()
```

