→ MACHINE LEARNING

▼ 1. Simple Linear Regression

▼ Step-1 Import Libraries

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import sklearn
```



```
df=pd.read_csv('salary_data.csv')
df.head()
```

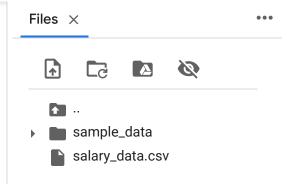
| | YearsExperience | Salary |
|---|-----------------|--------|
| 0 | 1.1 | 39343 |
| 1 | 1.3 | 46205 |
| 2 | 1.5 | 37731 |
| 3 | 2.0 | 43525 |
| 4 | 2.2 | 39891 |

▼ Step-3 Selecting Input And Output Variables

```
X=df[["YearsExperience"]]
y=df["Salary"]
```

▼ Step-4 Making Linear Regression Model

from sklearn.linear_model import LinearRegression
model=LinearRegression()



▼ Step-5 Fitting The Model

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

v LinearRegression
LinearRegression()
```

▼ Step-6 Predicting The Model

```
model.predict([[10]])

/usr/local/lib/python3.10/dist-packages/sklearn/base
    warnings.warn(
    array([120291.82341322])
```

▼ Step-7 Evaluating The Model

```
import numpy as np
from sklearn.model selection import train test split
from sklearn.linear model import LinearRegression
from sklearn.metrics import mean_squared_error
# Assuming you have your data stored in X and y arrays
# Splitting the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y,
# Training the linear regression model
regressor = LinearRegression()
regressor.fit(X train, y train)
# Predicting on the testing set
y_pred = regressor.predict(X_test)
# Calculating the mean squared error (MSE)
mse = mean_squared_error(y_test, y_pred)
# Calculating the accuracy (R-squared score)
accuracy = regressor.score(X_test, y_test)
print("Mean Squared Error: ", mse)
print("Accuracy (R-squared): ", accuracy)
```

Mean Squared Error: 12823412.298126549 Accuracy (R-squared): 0.988169515729126

Disk 84.44 GB available

✓ 0s completed at 7:22 PM

×