

Simple Linear Regression

Simple Linear regression algorithm has mainly two objectives:

1. Model the relationship between the two variables.
2. Forecasting new observations.

In []:

Importing the libraries

```
In [1]: import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

Importing the dataset

```
In [2]: dataset = pd.read_csv('Salary_Data.csv')
```

data_set - DataFrame

Index	YearsExperience	Salary
0	1	32383
1	1.1	45207
2	1.3	39751
3	2	43525
4	2.2	39891
5	2.7	56642
6	3	60150
7	3.2	54445
8	3.2	64445
9	3.7	57189
10	3.9	63218
11	4	55794
12	4	56957
13	4.1	57081

Format Resize ☒ Background color ☒ Column min/max Save and Close Close

```
In [3]: x = dataset.iloc[:, :-1].values  
        y = dataset.iloc[:, -1].values
```

x - NumPy array

	0
0	1
1	1.1
2	1.3
3	2
4	2.2
5	2.7
6	3
7	3.2
8	3.2
9	3.7
10	3.9
11	4

Format Resize ☒ Background color

Save and Close Close

y - NumPy array

	0
0	32383
1	45207
2	39751
3	43525
4	39891
5	56642
6	60150
7	54445
8	64445
9	57189
10	63218
11	55794

Format Resize ☒ Background color

Save and Close Close

Splitting the dataset into the Training set and Test set

In [4]: `from sklearn.model_selection import train_test_split`
`X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 1/3, random`

Test dataset

x_test - NumPy array

	0
0	1.3
1	10.3
2	4.1
3	3.9
4	9.5
5	8.7
6	9.6
7	4
8	5.3
9	7.9

Format Resize ☒ Background color

Save and Close Close

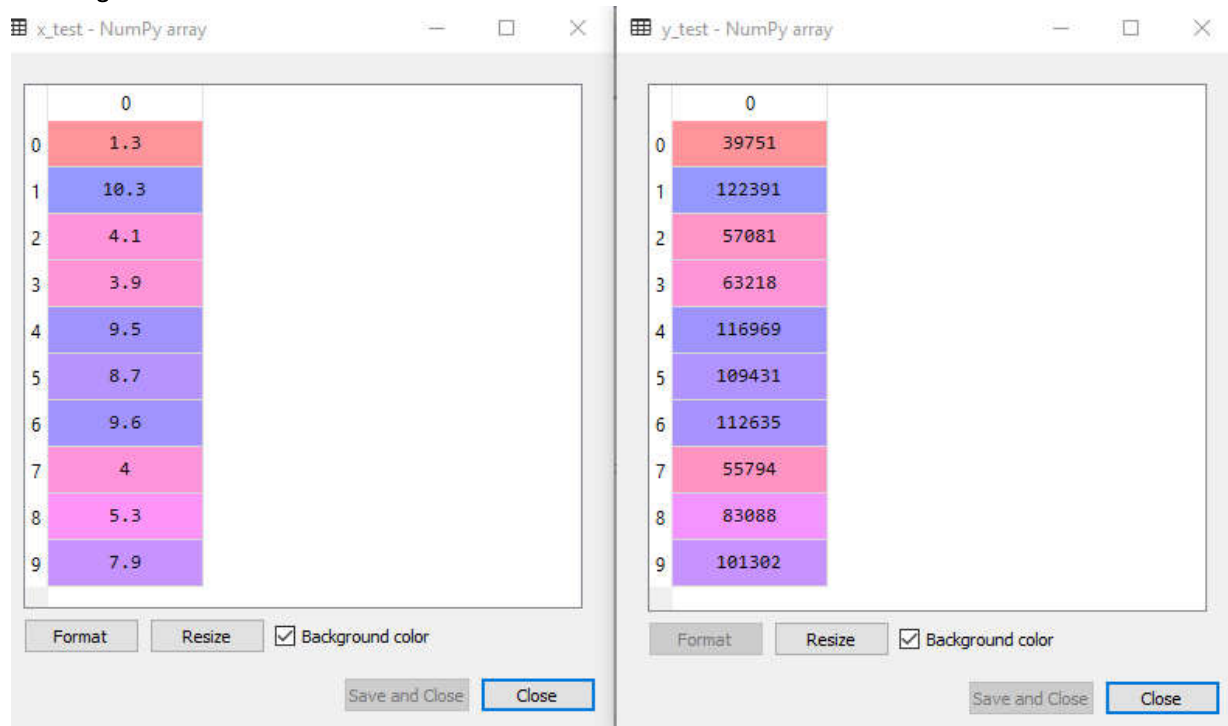
y_test - NumPy array

	0
0	39751
1	122391
2	57081
3	63218
4	116969
5	109431
6	112635
7	55794
8	83088
9	101302

Format Resize ☒ Background color

Save and Close Close

Training dataset



Training the Simple Linear Regression model on the Training set

```
In [5]: from sklearn.linear_model import LinearRegression  
regressor = LinearRegression()  
regressor.fit(X_train, y_train)
```

```
Out[5]: LinearRegression()
```

Predicting the Test set results

```
In [6]: y_pred = regressor.predict(X_test)
```

Visualising the Training set results

```
In [7]: plt.scatter(X_train, y_train, color = 'red')
plt.plot(X_train, regressor.predict(X_train), color = 'blue')
plt.title('Salary vs Experience (Training set)')
plt.xlabel('Years of Experience')
plt.ylabel('Salary')
plt.show()
```



Visualising the Test set results

```
In [8]: plt.scatter(X_test, y_test, color = 'red')
plt.plot(X_train, regressor.predict(X_train), color = 'blue')
plt.title('Salary vs Experience (Test set)')
plt.xlabel('Years of Experience')
plt.ylabel('Salary')
plt.show()
```



In []:

In []: