



Simple Linear Regression

MACHINE LEARNING USING PYTHON

CONTENTS

INTRODUCTION
REGRESSION ANALYSIS
DEPENDENT AND INDEPENDENT VARIABLES
LINES OF REGRESSION
EXAMPLE

INTRODUCTION

WHAT IS REGRESSION

Regression is a supervised learning technique that supports finding the correlation among variables. A regression problem is when the output variable is a real or continuous value.

WHAT IS SIMPLE LINEAR REGRESSION

In Machine Learning **Simple Linear Regression** is a type of Regression algorithms that models the relationship between a dependent variable and a single independent variable. The relationship shown by a **Simple Linear Regression** model is linear or a sloped straight line, hence it is called **Simple Linear Regression**.

REGRESSION ANALYSIS

- In statistics, regression analysis includes many technologies for modelling and analyzing several variables, when the focus is on relationship between one dependent variable and one or more independent variables.
- More specifically, regression analysis helps one understand how the typical value of the dependent variable changes when any one of the independent variables is varied, while the other independent variables are held fixed . Regression analysis is widely used for prediction and forecasting. Regression analysis is also used to understand which among the independent variables are related to the dependent variable, and to explore the forms of these relationships.
- In restricted circumstances, regression analysis can be used to infer causal relationships between the independent and dependent variables. However this can lead to illusions or false relationships, so caution is advisable

DEPENDENT AND INDEPENDENT VARIABLES

Independent variables are regarded as inputs to a system and may take different values freely in a system.

Independent variables are also called as predictor or explanatory variables and are denoted by X .

Dependent variables are those values that change as a consequence of change of other values in a system.

Dependent variables are also called as response variables and are denoted by Y .

LINES OF REGRESSION

A line that can be taken as representative of the ideal variation is called as the line of best fit .

It is a line such that the sum of the distances of the points from the line is minimum .It's called as "THE LINE OF REGRESSION" .

The distance is not measured by dropping a perpendicular from appoint to the line. We measure the deviation

(1) vertically and

(2) horizontally, and get one line where distance is minimized vertically and on other horizontally respectively .Thus we have two lines of regression

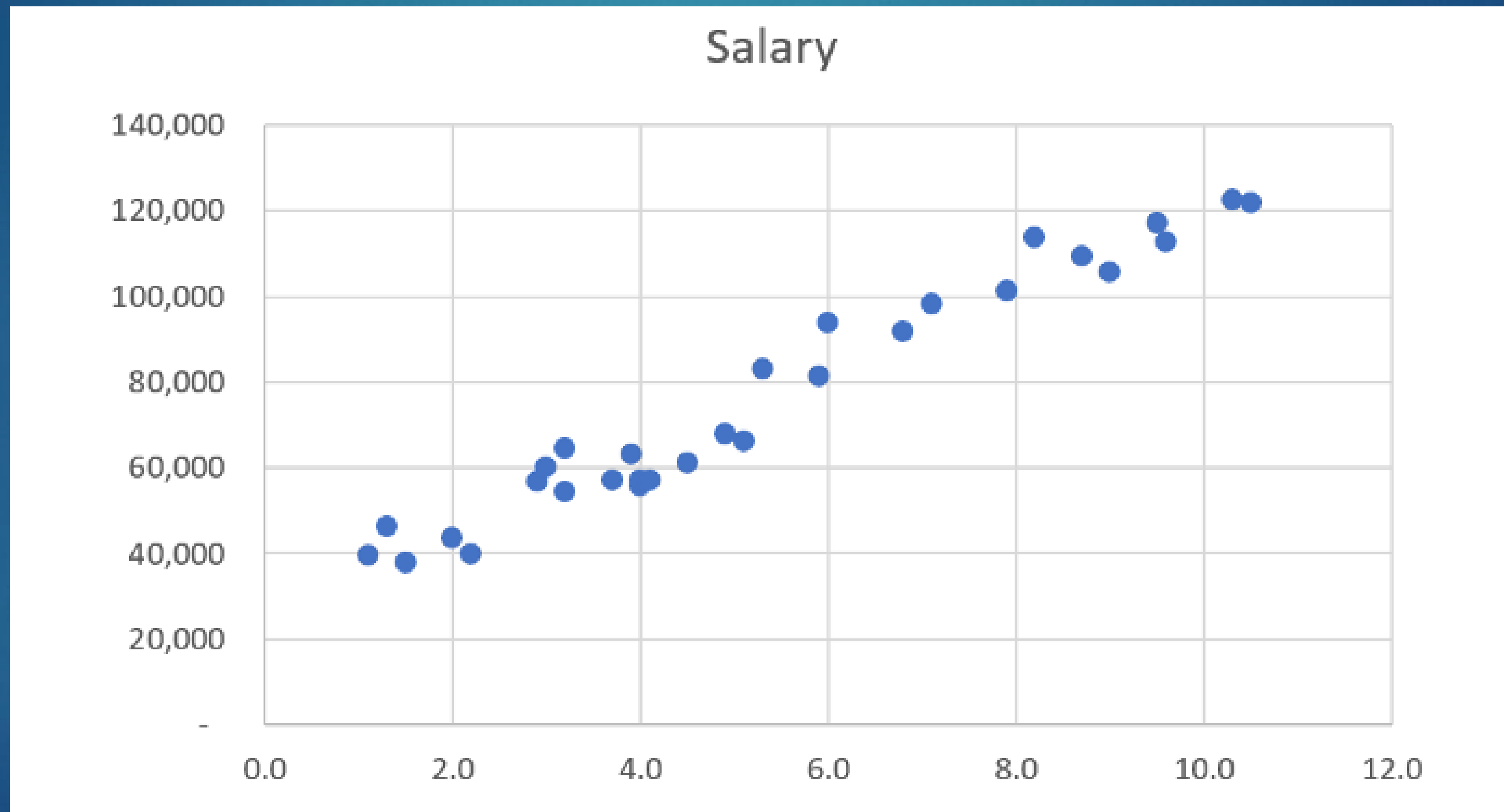
EXAMPLE

Let's take an example, in AB Company, there is a salary distribution table based on Year of Experience as per below:

YearsExperience	Salary
1.1	39,343
1.3	46,205
1.5	37,731
2.0	43,525
2.2	39,891
2.9	56,642
3.0	60,150
3.2	54,445
3.2	64,445
3.7	57,189
3.9	63,218
4.0	55,794
4.0	56,957
4.1	57,081
4.5	61,111
4.9	67,938
5.1	66,029
5.3	83,088
5.9	81,363
6.0	93,940
6.8	91,738
7.1	98,273
7.9	101,302
8.2	113,812
8.7	109,431
9.0	105,582
9.5	116,969
9.6	112,635
10.3	122,391
10.5	121,872

EXAMPLE

The scenario is you are a HR officer, you got a candidate with 5 years of experience. Then what is the best salary you should offer to him?"



EXAMPLE

Here all the observations are not in a line. It means we cannot find out the equation to calculate the (y) value.

All the points is not in a line BUT they are in a line-shape! **It's linear!**



Linear Regression with Python

Before moving on, we summarize 2 basic steps of Machine Learning as per below:

Training

Predict

Okay, we will use 4 libraries such as numpy and pandas to work with data set, sklearn to implement machine learning functions, and matplotlib to visualize our plots for viewing:

```
1  import numpy as np
2  import matplotlib.pyplot as plt
3  import pandas as pd
4
5  # Importing the dataset
6  dataset = pd.read_csv('salary_data.csv')
7  X = dataset.iloc[:, :-1].values #get a copy of dataset exclude last column
8  y = dataset.iloc[:, 1].values #get array of dataset in column 1st
```


Linear Regression with Python

Next, we have to split our dataset (total 30 observations) into 2 sets: training set which used for training and test set which used for testing:

```
1 # Splitting the dataset into the Training set and Test set
2 from sklearn.model_selection import train_test_split
3 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=1/3, random_state=0)
```

We already have the train set and test set, now we have to build the Regression Model:

```
1 # Fitting Simple Linear Regression to the Training set
2 from sklearn.linear_model import LinearRegression
3 regressor = LinearRegression()
4 regressor.fit(X_train, y_train)
```


Linear Regression with Python

Let's visualize our training model and testing model:

```
1  # Visualizing the Training set results
2  viz_train = plt
3  viz_train.scatter(X_train, y_train, color='red')
4  viz_train.plot(X_train, regressor.predict(X_train), color='blue')
5  viz_train.title('Salary VS Experience (Training set)')
6  viz_train.xlabel('Year of Experience')
7  viz_train.ylabel('Salary')
8  viz_train.show()
9
10 # Visualizing the Test set results
11 viz_test = plt
12 viz_test.scatter(X_test, y_test, color='red')
13 viz_test.plot(X_train, regressor.predict(X_train), color='blue')
14 viz_test.title('Salary VS Experience (Test set)')
15 viz_test.xlabel('Year of Experience')
16 viz_test.ylabel('Salary')
17 viz_test.show()
```

Linear Regression with Python

After running above code, you will see 2 plots in the console window



Linear Regression with Python

We already have the model, now we can use it to calculate (predict) *any values of X depends on y* or *any values of y depends on X*. This is how we do it:

```
1 # Predicting the result of 5 Years Experience
2 y_pred = regressor.predict(5)
```

```
In [2]: y_pred = regressor.predict(5)
```

```
In [3]: print(y_pred)
[73545.90445964]
```

```
In [4]:
```

Predict `y_pred` using single value of `X=5`

Linear Regression with Python

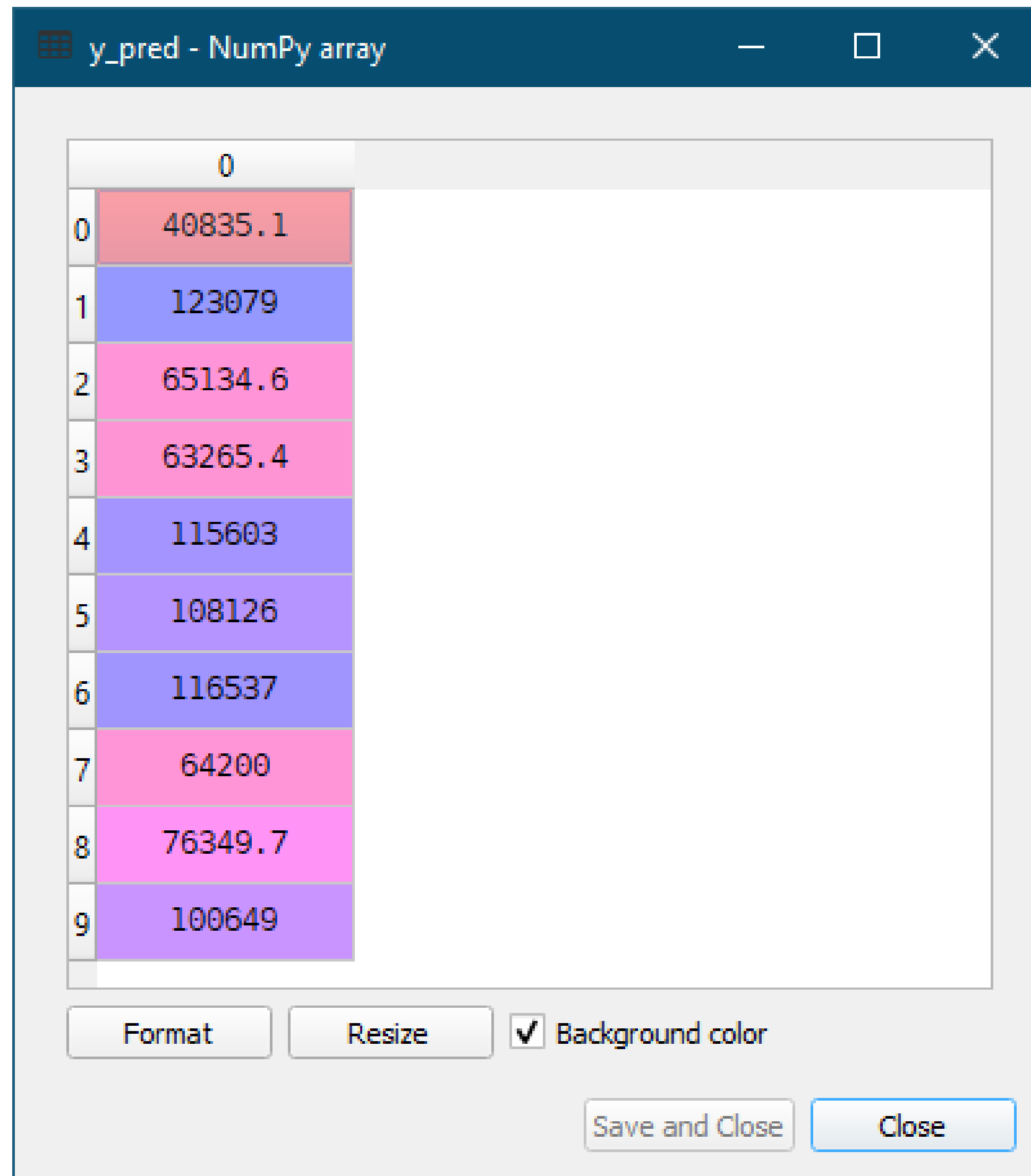
The value of y_{pred} with $X = 5$ (5 Years Experience) is 73545.90

You can offer to your candidate the salary of \$73,545.90 and this is the best salary for him!

We can also pass an **array of X** instead of **single value of X**:

```
1 # Predicting the Test set results
2 y_pred = regressor.predict(X_test)
```

Linear Regression with Python



y_pred - NumPy array

	0
0	40835.1
1	123079
2	65134.6
3	63265.4
4	115603
5	108126
6	116537
7	64200
8	76349.7
9	100649

Format Resize ☒ Background color Save and Close Close

Predict y_pred using array of X_test

In conclusion, with Simple Linear Regression, we have to do 5 steps as per below:



Importing the dataset.

Splitting dataset into training set and testing set (2 dimensions of X and y per each set)
Normally, the testing set should be 5% to 30% of dataset.

Visualize the training set and testing set to double check (you can bypass this step if you want).

Initializing the regression model and fitting it using training set (both X and y).
Let's predict!!



THANKS