Exposys Data Labs

Data science Internship

Report

Name: Mokkala Gunavarshin

ABSTRACT

The project deals with predicting the profit from start-ups dataset with the features available to us. We're using the 50_Startups dataset for this given problem statement. Start-ups are not such economically balanced company that has covered a path from an idea to a product, so for the same reason no established investor will be going to come forward for those companies which don't have their market value hence, start-ups allow early investors to start supporting in the format of seed funding which would help them to make a product out of their idea. In a nutshell, we can see that it's hard to manage and analyse the investments and to make a profit out of them. No company exists that doesn't want profit in minimum cost spend. So, to spend a minimum and get the maximum analysis is necessary regarding the investments and cost required. Based on the R&D Spend, Administration Cost and Marketing Spend data, we will predict the company's maximum profit based on their respective values. And we are trying to get the maximum accuracy when making predictions about the profit of the start-ups.

CONTENTS

1. INTRODUCTION

1.1 Introduction to Prediction Models

We all know that customer satisfaction is key to boosting a company's performance, but organizations still strive to utilize the increasing availability of data to satisfy customers. The report illustrates how machine learning and data science techniques can be employed to assess and evaluate customer satisfaction. It is necessary to present steps to develop customer-driven prediction models, starting from problem framing, to data exploratory analysis, data transformation, ML training, and recommendations.

Predictive analytics involves certain manipulations of data from existing data sets to identify new trends and patterns. These trends and patterns are then used to predict future outcomes and trends. By performing predictive analysis, we can predict future trends and performance.

It is also defined as the prognostic analysis; the word prognostic means prediction. Predictive analytics uses data, statistical algorithms and machine learning techniques to identify the probability of future outcomes based on historical data. In predictive analysis, we use historical data to predict future outcomes. Thus, predictive analysis plays a vital role in various fields. It improves decision-making, helps increase the profit rates of businesses, and reduces risk by identifying them early. Predictive analysis is used in various fields like Online Retail, and Improvised market campaigning.

1.2 Objective of the work

The objective is to analyse our expenditure on the start-ups and then know the profit put them. The r programming language will be quite helpful in such a situation where we need to find a profit based on how much we are spending in the market and for the market. In a nutshell, will help to find out the profit based on the amount we spend from the 50_ Start-up dataset.

2. EXISTING METHOD

Gentle working of Prediction Models:

Step 1: Sample data

Data is information about the problem that you are working on. Imagine we want to identify the species of flower from the measurements of a flower. The data is comprised of four flower measurements in centimetres, these are the columns of the data. Each row of data is one example of a flower that has been measured and its known species. The problem we are solving is to create a model from the sample data that can tell us which species a flower belongs to from its measurements alone.

Step 2: Learn a Model

This problem described above is called supervised learning. The goal of a supervised learning algorithm is to take some data with a known relationship and create a model of those relationships. In this case, the output is a category and we call this type of problem a classification problem. If the output was a numerical value, we would call it a regression problem. The algorithm does the learning. The model contains the learned relationships.

Step 3: Make Predictions

We don't need to keep the training data as the model has summarized the relationships contained within it. The reason we keep the model learned from data is that we want to use it to make predictions. Our model will read the input perform a calculation of some kind with its internal numbers and make a prediction. The prediction may not be perfect, but if you have good sample data and a robust model learned from that data, it will be quite accurate.

3. THE PROPOSED METHOD WITH ARCHITECTURE

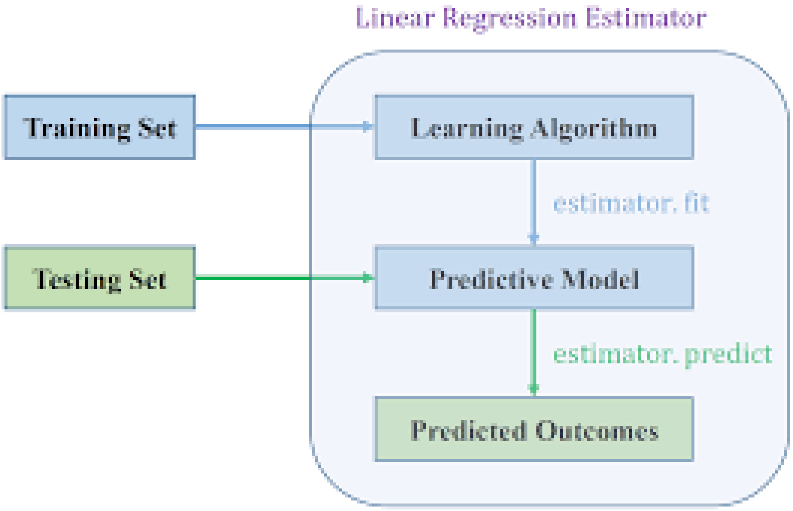
- **Regression Models:**

Regression methods fall within the category of supervised ML. They help to predict or explain a particular numerical value based on a set of prior data, for example predicting the price of a property based on previous pricing data for similar properties.

- **Linear Regression:**

Linear regression is a statistical regression method which is used for predictive analysis. It is one of the very simple and easy algorithms which works on regression and shows the relationship between the continuous variables. It is used for solving the regression problem in machine learning. Linear regression shows the linear relationship between the independent variable (X-axis) and the dependent variable (Y-axis), hence called linear regression.

If there is only one input variable (x), then such linear regression is called simple linear regression. And if there is more than one input variable, then such linear regression is called multiple linear regression.



Linear Regression Model

- **Some popular applications of linear regression are:**

1. Analysing trends and sales estimates
2. Salary forecasting
3. Profit prediction
4. Arriving at ETAs in traffic.

In Regression, we plot a graph between the variables which best fit the given data points, using this plot, the machine learning model can make predictions about the data. In simple words, Regression shows a line or curve that passes through all the data points on the target-predictor graph in such a way that the vertical distance between the data points and the regression line is minimum. The distance between data points and line tells whether a model has captured a strong relationship or not.

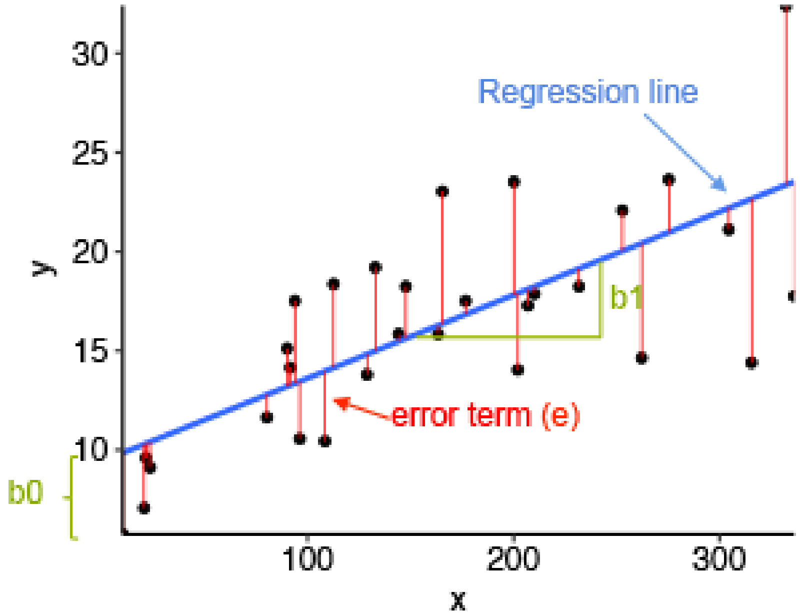


Image Source: Statistical tools for high-throughput data analysis

Representation of Regression Line and its relation with Data points

4. METHODOLOGY

We need a way by which we can analyse our expenditure on the start-ups and then know a profit out of them. The Python programming language will be quite helpful in such a situation where we need to find a profit based on how much we are spending in the market and for the market. In a nutshell, will help to find out the profit based on the amount we spend from the 50_ Start-up dataset.

Data Scientists need to prepare the dataset and need to perform the Data pre-processing and Data cleaning. Data analysts need to analyse the data set and predict the solution.

5. IMPLEMENTATION

Some so many good packages and libraries that can be used for building predictive models. Some of the most common packages and libraries for predictive analytics include:

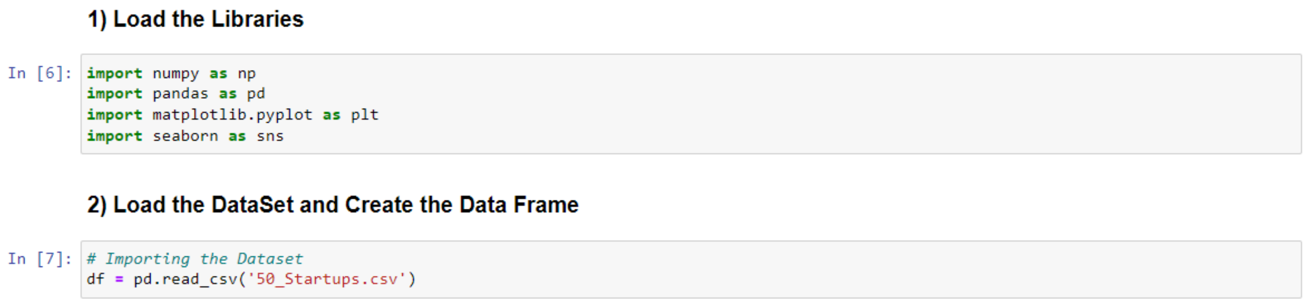
numpy 1.24.2

Panda 1.5.3

Matplotlib 3.7.0

Seaborn 0.11.0

We are working with the 50_Startup.csv dataset file. After importing the libraries, use pd. Read from Pandas to read/import dataset.



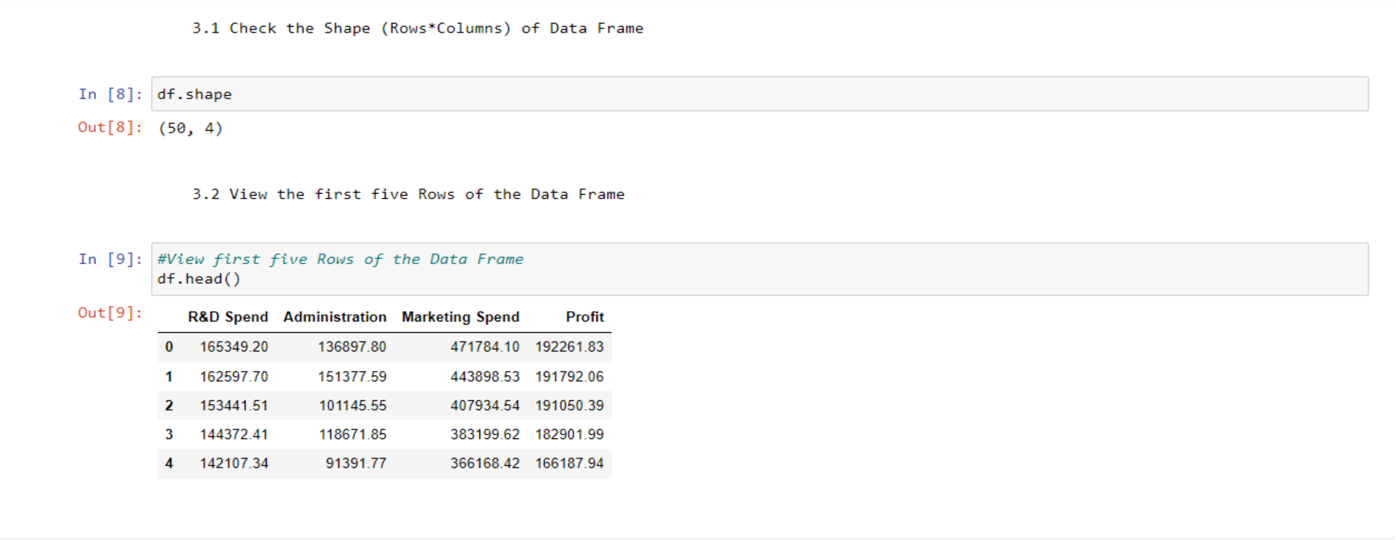
1) Load the Libraries

```
In [6]: import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        import seaborn as sns
```

2) Load the DataSet and Create the Data Frame

```
In [7]: # Importing the Dataset
        df = pd.read_csv('50_Startups.csv')
```

To check whether the dataset is imported correctly or not, we can visualize the data from the dataset or can get the basic information using matplotlib.



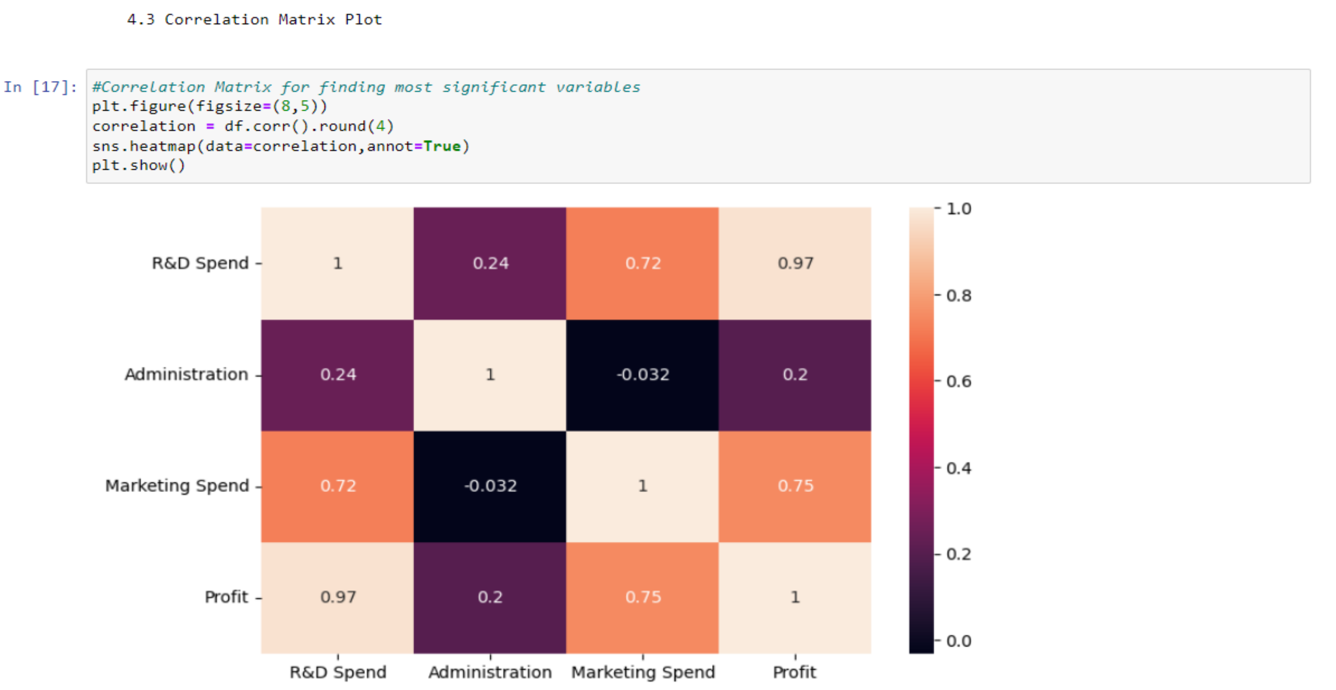
3.1 Check the Shape (Rows*Columns) of Data Frame

```
In [8]: df.shape
Out[8]: (50, 4)
```

3.2 View the first five Rows of the Data Frame

```
In [9]: #View first five Rows of the Data Frame
        df.head()
Out[9]:
```

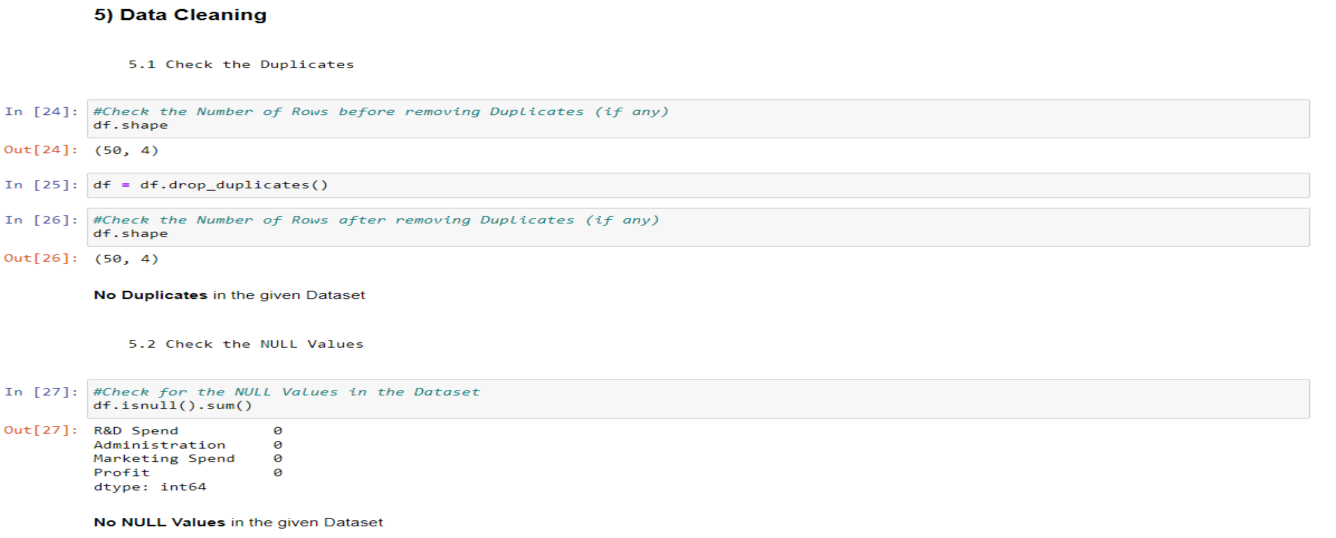
	R&D Spend	Administration	Marketing Spend	Profit
0	165349.20	136897.80	471784.10	192261.83
1	162597.70	151377.59	443898.53	191792.06
2	153441.51	101145.55	407934.54	191050.39
3	144372.41	118671.85	383199.62	182901.99
4	142107.34	91391.77	366168.42	166187.94



4.3 Correlation Matrix Plot

```
In [17]: #Correlation Matrix for finding most significant variables
         plt.figure(figsize=(8,5))
         correlation = df.corr().round(4)
         sns.heatmap(data=correlation,annot=True)
         plt.show()
```

Data Cleaning and Transformation: The second task is to perform data wrangling. Luckily, the dataset does not require any cleaning.



5) Data Cleaning

5.1 Check the Duplicates

```
In [24]: #Check the Number of Rows before removing Duplicates (if any)
         df.shape
Out[24]: (50, 4)

In [25]: df = df.drop_duplicates()

In [26]: #Check the Number of Rows after removing Duplicates (if any)
         df.shape
Out[26]: (50, 4)
```

No Duplicates in the given Dataset

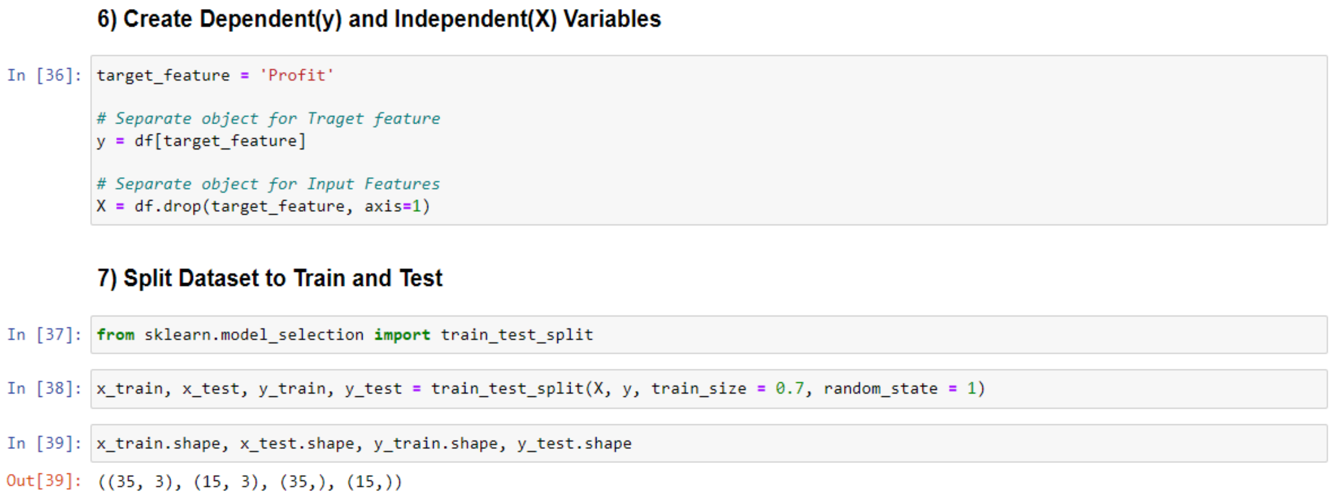
5.2 Check the NULL Values

```
In [27]: #Check for the NULL Values in the Dataset
         df.isnull().sum()
Out[27]: R&D Spend          0
         Administration     0
         Marketing Spend    0
         Profit             0
         dtype: int64
```

No NULL Values in the given Dataset

To use the regression model on the dataset, first, we need to create dependent and independent variables by setting profit as our target feature.

And we also need to split our data into train data and test data to use the data for training and testing respectively.



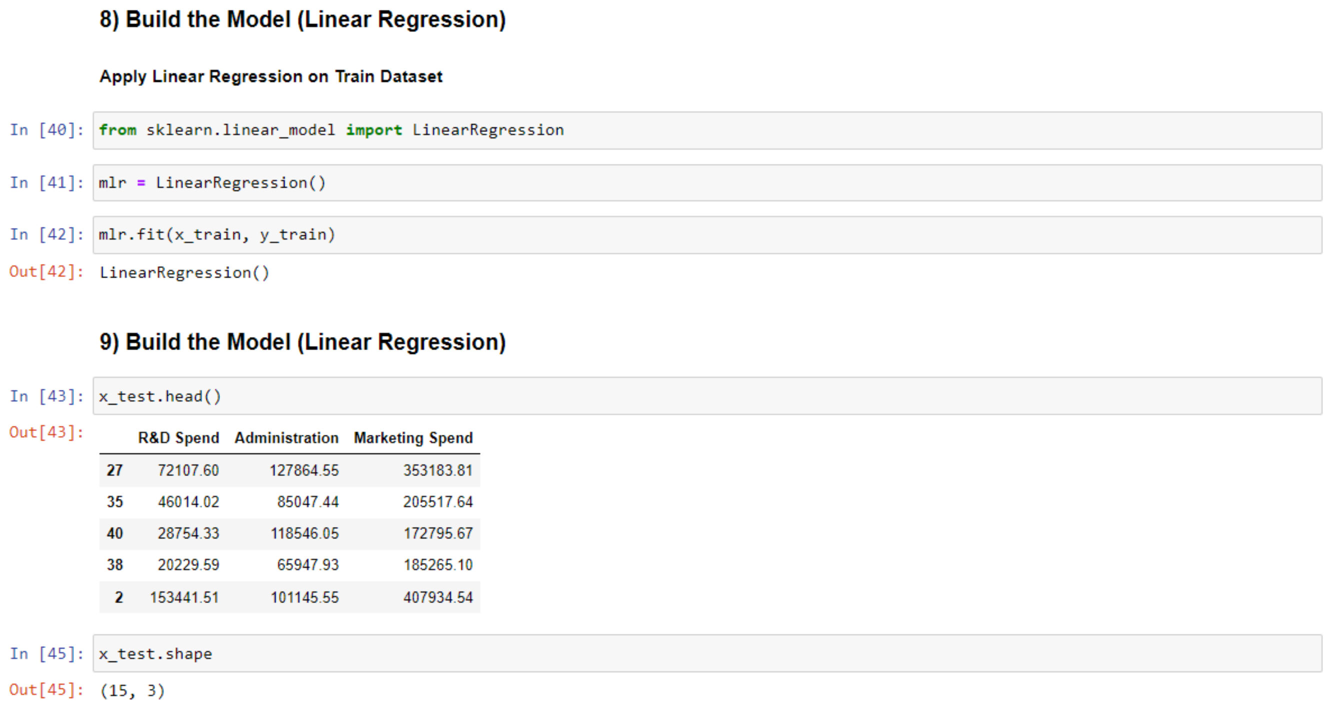
6) Create Dependent(y) and Independent(X) Variables

```
In [36]: target_feature = 'Profit'

         # Separate object for Traget feature
         y = df[target_feature]

         # Separate object for Input Features
         X = df.drop(target_feature, axis=1)
```

7) Split Dataset to Train and Test

```
In [37]: from sklearn.model_selection import train_test_split

In [38]: x_train, x_test, y_train, y_test = train_test_split(X, y, train_size = 0.7, random_state = 1)

In [39]: x_train.shape, x_test.shape, y_train.shape, y_test.shape

Out[39]: ((35, 3), (15, 3), (35,), (15,))
```

After splitting the dataset, we are now able to apply linear regression to the training dataset. And to check if it is successfully applied or not, we can check the output of the step.



8) Build the Model (Linear Regression)

Apply Linear Regression on Train Dataset

```
In [40]: from sklearn.linear_model import LinearRegression

In [41]: mlr = LinearRegression()

In [42]: mlr.fit(x_train, y_train)

Out[42]: LinearRegression()
```

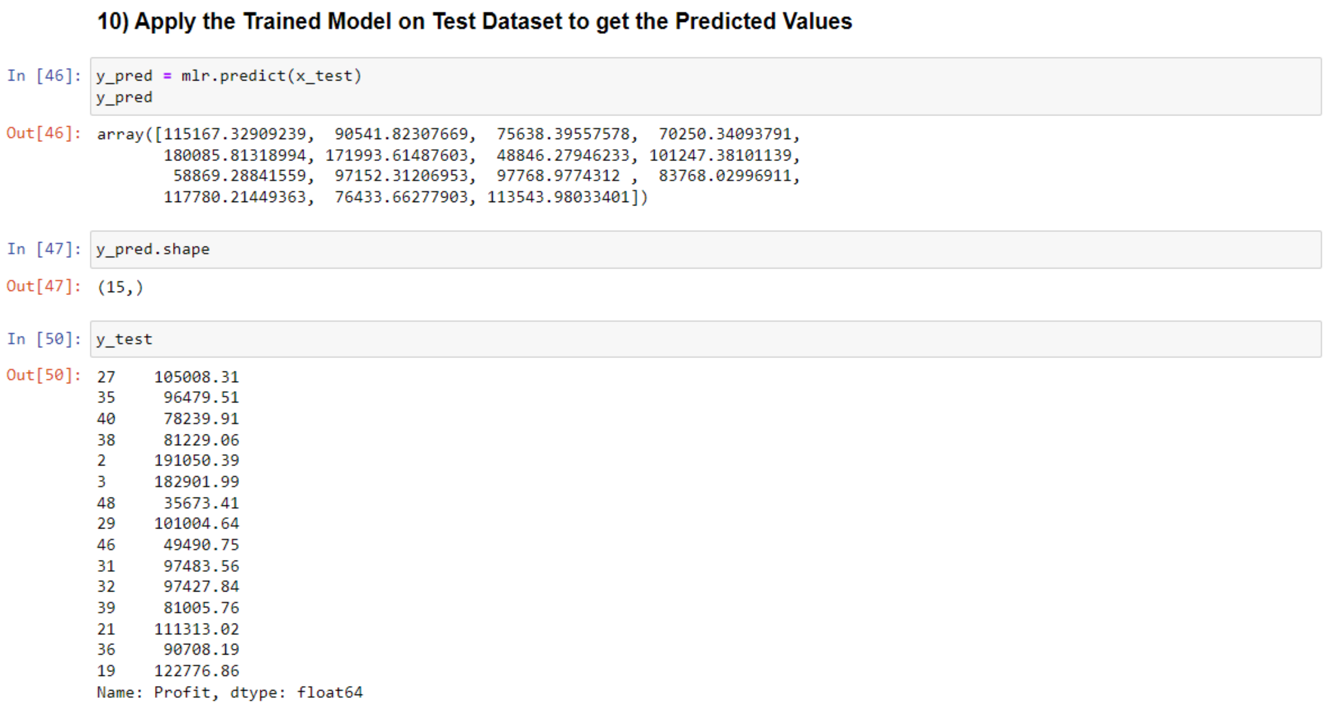
9) Build the Model (Linear Regression)

```
In [43]: x_test.head()

Out[43]:
```

	R&D Spend	Administration	Marketing Spend
27	72107.60	127864.55	353183.81
35	46014.02	85047.44	205517.64
40	28754.33	118546.05	172795.67
38	20229.59	65947.93	185265.10
2	153441.51	101145.55	407934.54

```
In [45]: x_test.shape

Out[45]: (15, 3)
```

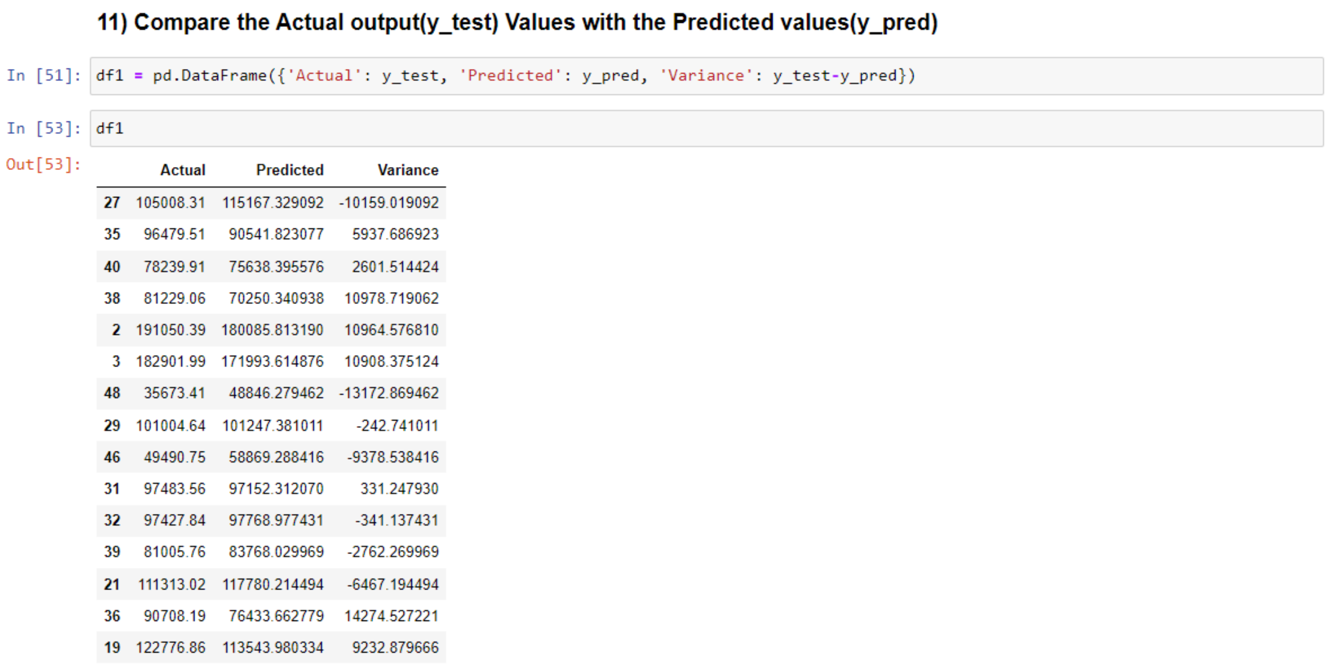
Once the model is successfully trained, we can apply the Trained Model to the Test dataset to get the predicted values.



10) Apply the Trained Model on Test Dataset to get the Predicted Values

```
In [46]:  y_pred = mlr.predict(x_test)
           y_pred

Out[46]:  array([115167.32909239,  90541.82307669,  75638.39557578,  70250.34093791,
                 180085.81318994, 171993.61487603,  48846.27946233, 101247.38101139,
                  58869.28841559,  97152.31206953,  97768.9774312 ,  83768.02996911,
                 117780.21449363,  76433.66277903, 113543.98033401])

In [47]:  y_pred.shape

Out[47]:  (15,)

In [50]:  y_test

Out[50]:  27    105008.31
          35     96479.51
          40     78239.91
          38     81229.06
          2     191050.39
          3     182901.99
          48     35673.41
          29    101004.64
          46     49490.75
          31     97483.56
          32     97427.84
          39     81005.76
          21    111313.02
          36     90708.19
          19    122776.86
          Name: Profit, dtype: float64
```

Now we got our predicted values and we already have our actual values so we can compare them and observe the difference or relations between values.

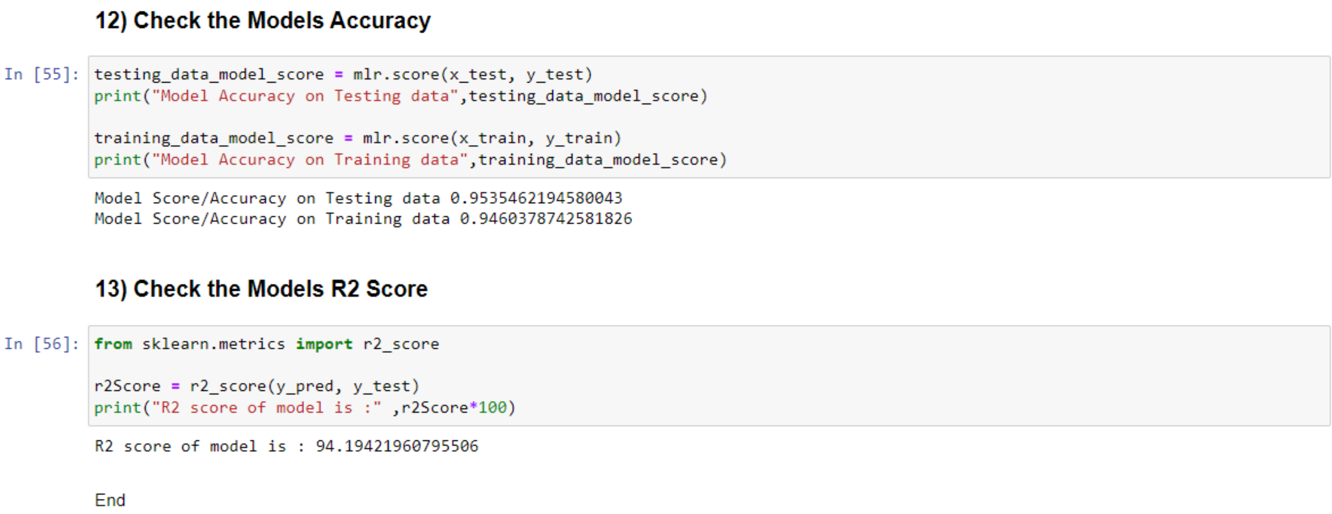


11) Compare the Actual output(y_test) Values with the Predicted values(y_pred)

```
In [51]:  df1 = pd.DataFrame({'Actual': y_test, 'Predicted': y_pred, 'Variance': y_test-y_pred})

In [53]:  df1

Out[53]:
```

	Actual	Predicted	Variance
27	105008.31	115167.329092	-10159.019092
35	96479.51	90541.823077	5937.686923
40	78239.91	75638.395576	2601.514424
38	81229.06	70250.340938	10978.719062
2	191050.39	180085.813190	10964.576810
3	182901.99	171993.614876	10908.375124
48	35673.41	48846.279462	-13172.869462
29	101004.64	101247.381011	-242.741011
46	49490.75	58869.288416	-9378.538416
31	97483.56	97152.312070	331.247930
32	97427.84	97768.977431	-341.137431
39	81005.76	83768.029969	-2762.269969
21	111313.02	117780.214494	-6467.194494
36	90708.19	76433.662779	14274.527221
19	122776.86	113543.980334	9232.879666

After comparing the predicted values and actual values we can finally check the accuracy of the model or score of the model. And we can also check the R2 Score of our model which determines how well a statistical model predicts an outcome.



12) Check the Models Accuracy

```
In [55]: testing_data_model_score = mlr.score(x_test, y_test)
         print("Model Accuracy on Testing data",testing_data_model_score)

         training_data_model_score = mlr.score(x_train, y_train)
         print("Model Accuracy on Training data",training_data_model_score)
```

```
Model Score/Accuracy on Testing data 0.9535462194580043
Model Score/Accuracy on Training data 0.9460378742581826
```

13) Check the Models R2 Score

```
In [56]: from sklearn.metrics import r2_score

         r2Score = r2_score(y_pred, y_test)
         print("R2 score of model is :" ,r2Score*100)
```

```
R2 score of model is : 94.19421960795506
```

End

6. CONCLUSION

Based on the findings, we conclude that our dataset does not require any cleaning and profit is the most important feature of our dataset. In general, our predictive algorithm has a 94% predictive power to predict the profit value of the company based on R&D Spend, Administration cost and Marketing Spend.