Maninder Singh

101703325

Coe 15

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import os
os.chdir("D:/Desktop")
dataset=pd.read_csv("50_Startups.csv")

X=dataset.iloc[:,:-1].values
Y=dataset.iloc[: ,4].values
```

- Dataset :

Index	R&D Spend	Administration	Marketing Spend	State	Profit
0	165349	136898	471784	New York	192262
1	162598	151378	443899	California	191792
2	153442	101146	407935	Florida	191050
3	144372	118672	383200	New York	182902
4	142107	91391.8	366168	Florida	166188
5	131877	99814.7	362861	New York	156991
6	134615	147199	127717	California	156123
7	130298	145530	323877	Florida	155753

- X:

Index	R&D Spend	Administration	Marketing Spend	State
0	165349	136898	471784	New York
1	162598	151378	443899	California
2	153442	101146	407935	Florida
3	144372	118672	383200	New York
4	142107	91391.8	366168	Florida
5	131877	99814.7	362861	New York
6	134615	147199	127717	California

- Y :

	0
0	192262
1	191792
2	191050
3	182902
4	166188
5	156991
6	156123
7	155753

- Third Column in X needs to be pre-processed .

```
from sklearn.preprocessing import LabelEncoder, OneHotEncoder
labelencoder_X_1 = LabelEncoder()
X[: , 3]=labelencoder_X_1.fit_transform(X[:,3])
```

```
onehotencoder = OneHotEncoder(categorical_features =[3])
X=onehotencoder.fit_transform(X).toarray()
```

- To Avoid Dummy Variable Trap

```
X = X[ : ,1:]
```

- To Split dataset into Training and Test sets

```
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 = 0)
```

- Applying Multiple Linear Regression

```
from sklearn.linear_model import LinearRegression
Reg = LinearRegression()

Reg.fit(X_train , Y_train);
Y_predict = Reg.predict(X_test)
```

- Adding a Column of 1's in the beginning of X

```
import statsmodels.formula.api as sm
X=np.append(arr=np.ones((50,1)).astype(int) , values = X , axis = 1);
```

X:

	0	1	2	3	4	5
0	1	0	1	165349	136898	471784
1	1	0	0	162598	151378	443899
2	1	1	0	153442	101146	407935
3	1	0	1	144372	118672	383200
4	1	1	0	142107	91391.8	366168

```
X_opt = X[ : ,[0,1,2,3,4,5]]
regressor_OLS = sm.OLS(endog = Y , exog = X_opt).fit();
```

```
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const       5.013e+04   6884.820      7.281      0.000    3.62e+04     6.4e+04
x1           198.7888   3371.007      0.059      0.953   -6595.030    6992.607
x2           -41.8870   3256.039     -0.013      0.990   -6604.003    6520.229
x3             0.8060      0.046     17.369      0.000       0.712       0.900
x4            -0.0270      0.052     -0.517      0.608      -0.132       0.078
x5             0.0270      0.017      1.574      0.123      -0.008       0.062
==============================================================================
```

- X2 has highest p Value and it is greater than Significant Level (0. 05) , So NULL hypothesis is Accepted and X2 has been eliminated in the next Step .

```
X_opt = X[ : ,[0,1,3,4,5]]
regressor_OLS = sm.OLS(endog = Y , exog = X_opt).fit();
```

	coef	std err	t	P>|t|	[0.025	0.975]
const	5.011e+04	6647.870	7.537	0.000	3.67e+04	6.35e+04
x1	220.1585	2900.536	0.076	0.940	-5621.821	6062.138
x2	0.8060	0.046	17.606	0.000	0.714	0.898
x3	-0.0270	0.052	-0.523	0.604	-0.131	0.077
x4	0.0270	0.017	1.592	0.118	-0.007	0.061

- X2 has highest p Value and it is greater than Significant Level (0. 05) , So NULL hypothesis is Accepted and X2 has been eliminated in the next Step .

```
X_opt = X[ : ,[0,3,4,5]]
regressor_OLS = sm.OLS(endog = Y , exog = X_opt).fit();
```

	coef	std err	t	P>|t|	[0.025	0.975]
const	5.012e+04	6572.353	7.626	0.000	3.69e+04	6.34e+04
x1	0.8057	0.045	17.846	0.000	0.715	0.897
x2	-0.0268	0.051	-0.526	0.602	-0.130	0.076
x3	0.0272	0.016	1.655	0.105	-0.006	0.060

- X2 has highest p Value and it is greater than Significant Level (0. 05) , So NULL hypothesis is Accepted and X2 has been eliminated in the next Step .

```
X_opt = X[ : ,[0,3,5]]
regressor_OLS = sm.OLS(endog = Y , exog = X_opt).fit();
```

```
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const       4.698e+04   2689.933     17.464      0.000    4.16e+04    5.24e+04
x1             0.7966      0.041     19.266      0.000       0.713       0.880
x2             0.0299      0.016      1.927      0.060      -0.001       0.061
==============================================================================
```

- X2 has highest p Value and it is greater than Significant Level (0. 05) , So NULL hypothesis is Accepted and X2 has been eliminated in the next Step .

```
48 X_opt = X[ : ,[0,3]]
49 regressor_OLS = sm.OLS(endog = Y , exog = X_opt).fit();
```

```
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const       4.903e+04   2537.897     19.320      0.000    4.39e+04    5.41e+04
x1             0.8543      0.029     29.151      0.000       0.795       0.913
==============================================================================
```

- Since X1 is the only attribute and it has p Value less than Significant Level(0.05) . So NULL hypothesis is Rejected and only X1 remains .