Multilinear Regression

Importing the libraries

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

Importing the dataset

In [32]:
```
dataset = pd.read_csv('50_Startups.csv')
```

Lets look at the dataset

In [33]:
```
dataset.head(7)
```

Out[33]:

	R&D Spend	Administration	Marketing Spend	State	Profit
0	165349.20	136897.80	471784.10	New York	192261.83
1	162597.70	151377.59	443898.53	California	191792.06
2	153441.51	101145.55	407934.54	Florida	191050.39
3	144372.41	118671.85	383199.62	New York	182901.99
4	142107.34	91391.77	366168.42	Florida	166187.94
5	131876.90	99814.71	362861.36	New York	156991.12
6	134615.46	147198.87	127716.82	California	156122.51

Encoding Categorical Data

The `State` column contains categorical features. This needs to be converted into Dummy Variables

In [34]:
```
from sklearn.preprocessing import LabelEncoder, OneHotEncoder
labelencoder = LabelEncoder()

# Taking the categorical column and label encoding it
state = dataset.State
dataset['State_Encoded'] = labelencoder.fit_transform(state.values)

# Perform OneHotEncoding on the Label Encoded Column
onehotencoder = OneHotEncoder(categorical_features = 'all')
ohe = onehotencoder.fit_transform(dataset['State_Encoded'].values.reshape(-1, 1)).toarray

# Adding the dummy variables to the dataset
new_columns = list(state.sort_values().unique())
for index, column in enumerate(new_columns):
    dataset[column] = ohe[:,index]

# Re-arranging the required columns
dataset = dataset.iloc[:, [0,1,2,6,7,8,4]]

# Removing the intermediate variables (Optional)
del ohe, state, column, index, new_columns
```

After pre-processing, the dataset looks like this:

In [35]:
```
dataset.head(7)
```

Out[35]:

	R&D Spend	Administration	Marketing Spend	California	Florida	New York	Profit
0	165349.20	136897.80	471784.10	0.0	0.0	1.0	192261.83
1	162597.70	151377.59	443898.53	1.0	0.0	0.0	191792.06
2	153441.51	101145.55	407934.54	0.0	1.0	0.0	191050.39
3	144372.41	118671.85	383199.62	0.0	0.0	1.0	182901.99
4	142107.34	91391.77	366168.42	0.0	1.0	0.0	166187.94
5	131876.90	99814.71	362861.36	0.0	0.0	1.0	156991.12
6	134615.46	147198.87	127716.82	1.0	0.0	0.0	156122.51

Splitting the Independent and Dependent Variables

In [36]:
```
X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, -1].values
```

Avoiding the Dummy Variable Trap (Optional, the library already does this)

In [15]:
```
X = X[:, :-1]
```

Splitting the dataset into the Training set and Test set

In [16]:
```
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 =
```

Fitting Multiple Linear Regression to the Training set

The library is same as it was for Simple Linear Regression

In [17]:
```
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(X_train, y_train)
```

Out[17]:
```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=1, normalize=False)
```

Predicting the Test set results

In [18]:
```
y_pred = regressor.predict(X_test)
```

R-Squared value to evaluate Model performance

In [20]:
```
print("R-Sq Value = {}".format(regressor.score(X_test, y_test)))
```

```
R-Sq Value = 0.9347068473294998
```

Backward Elimination of unnecessary columns

Adding an intercept at the beginning

In [21]:
```
X = np.append(arr = np.ones((50, 1)).astype(int), values = X, axis = 1)
```

Building the optimal model using Backward Elimination

In [22]:
```
import statsmodels.formula.api as sm
X_opt = X[:, [0, 1, 2, 3, 4, 5]]
regressor_OLS = sm.OLS(endog = y, exog = X_opt).fit()
print(regressor_OLS.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.951
Model:                            OLS   Adj. R-squared:                  0.945
Method:                 Least Squares   F-statistic:                     169.9
Date:                Sat, 14 Jul 2018   Prob (F-statistic):           1.34e-27
Time:                        04:44:01   Log-Likelihood:                -525.38
No. Observations:                  50   AIC:                             1063.
Df Residuals:                      44   BIC:                             1074.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const       5.008e+04   6952.587      7.204      0.000    3.61e+04    6.41e+04
x1             0.8060      0.046     17.369      0.000       0.712       0.900
x2            -0.0270      0.052     -0.517      0.608      -0.132       0.078
x3             0.0270      0.017      1.574      0.123      -0.008       0.062
x4            41.8870   3256.039      0.013      0.990   -6520.229    6604.003
x5           240.6758   3338.857      0.072      0.943   -6488.349    6969.701
==============================================================================
Omnibus:                       14.782   Durbin-Watson:                   1.283
Prob(Omnibus):                  0.001   Jarque-Bera (JB):               21.266
Skew:                          -0.948   Prob(JB):                     2.41e-05
Kurtosis:                       5.572   Cond. No.                     1.47e+06
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specif
ied.
[2] The condition number is large, 1.47e+06. This might indicate that there are
strong multicollinearity or other numerical problems.
```

As we can see in the above summary, the biggest p-value is for column with index number 4 . We will remove this column and run the model again

In [23]:
```python
# Removing the column with index 4
X_opt = X[:, [0, 1, 2, 3, 5]]
regressor_OLS = sm.OLS(endog = y, exog = X_opt).fit()
print(regressor_OLS.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.951
Model:                            OLS   Adj. R-squared:                  0.946
Method:                 Least Squares   F-statistic:                     217.2
Date:                Sat, 14 Jul 2018   Prob (F-statistic):           8.49e-29
Time:                        04:45:55   Log-Likelihood:                -525.38
No. Observations:                  50   AIC:                             1061.
Df Residuals:                      45   BIC:                             1070.
Df Model:                           4
Covariance Type:            nonrobust
==============================================================================
                 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             0.8060      0.046     17.606      0.000       0.714       0.898
x2            -0.0270      0.052     -0.523      0.604      -0.131       0.077
x3             0.0270      0.017      1.592      0.118      -0.007       0.061
x4           220.1585   2900.536      0.076      0.940   -5621.821    6062.138
==============================================================================
Omnibus:                       14.758   Durbin-Watson:                   1.282
Prob(Omnibus):                  0.001   Jarque-Bera (JB):               21.172
Skew:                          -0.948   Prob(JB):                     2.53e-05
Kurtosis:                       5.563   Cond. No.                     1.40e+06
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specif
ied.
[2] The condition number is large, 1.4e+06. This might indicate that there are
strong multicollinearity or other numerical problems.
```

As we can see in the above summary, the biggest p-value is for column with index number 4 . We will remove this column and run the model again

In [25]:

```python
# Removing the column with index 4
X_opt = X[:, [0, 1, 2, 3]]
regressor_OLS = sm.OLS(endog = y, exog = X_opt).fit()
print(regressor_OLS.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.951
Model:                            OLS   Adj. R-squared:                  0.948
Method:                 Least Squares   F-statistic:                     296.0
Date:                Sat, 14 Jul 2018   Prob (F-statistic):           4.53e-30
Time:                        04:48:05   Log-Likelihood:                -525.39
No. Observations:                  50   AIC:                             1059.
Df Residuals:                      46   BIC:                             1066.
Df Model:                           3
Covariance Type:            nonrobust
==============================================================================
                 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
==============================================================================
Omnibus:                       14.838   Durbin-Watson:                   1.282
Prob(Omnibus):                  0.001   Jarque-Bera (JB):               21.442
Skew:                          -0.949   Prob(JB):                     2.21e-05
Kurtosis:                       5.586   Cond. No.                     1.40e+06
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specif
ied.
[2] The condition number is large, 1.4e+06. This might indicate that there are
strong multicollinearity or other numerical problems.
```

As we can see in the above summary, the biggest p-value is for column with index number 2 . We will remove this column and run the model again

In [29]:
```python
X_opt = X[:, [0, 1, 3]]
regressor_OLS = sm.OLS(endog = y, exog = X_opt).fit()
print(regressor_OLS.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.950
Model:                            OLS   Adj. R-squared:                  0.948
Method:                 Least Squares   F-statistic:                     450.8
Date:                Sat, 14 Jul 2018   Prob (F-statistic):           2.16e-31
Time:                        04:52:22   Log-Likelihood:                -525.54
No. Observations:                  50   AIC:                             1057.
Df Residuals:                      47   BIC:                             1063.
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 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
==============================================================================
Omnibus:                       14.677   Durbin-Watson:                   1.257
Prob(Omnibus):                  0.001   Jarque-Bera (JB):               21.161
Skew:                          -0.939   Prob(JB):                     2.54e-05
Kurtosis:                       5.575   Cond. No.                     5.32e+05
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specif
ied.
[2] The condition number is large, 5.32e+05. This might indicate that there are
strong multicollinearity or other numerical problems.
```

As we can see in the above summary, the biggest p-value is for column with index number 2 . This is still above 5% significance level. Hence we need to remove this column too.

In [30]:
```python
X_opt = X[:, [0, 1]]
regressor_OLS = sm.OLS(endog = y, exog = X_opt).fit()
print(regressor_OLS.summary())
```

```
                            OLS Regression Results                            
==============================================================================
Dep. Variable:                      y   R-squared:                       0.947
Model:                            OLS   Adj. R-squared:                  0.945
Method:                 Least Squares   F-statistic:                     849.8
Date:                Sat, 14 Jul 2018   Prob (F-statistic):           3.50e-32
Time:                        04:52:44   Log-Likelihood:                -527.44
No. Observations:                  50   AIC:                             1059.
Df Residuals:                      48   BIC:                             1063.
Df Model:                           1                                         
Covariance Type:            nonrobust                                         
==============================================================================
                 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
==============================================================================
Omnibus:                       13.727   Durbin-Watson:                   1.116
Prob(Omnibus):                  0.001   Jarque-Bera (JB):               18.536
Skew:                          -0.911   Prob(JB):                     9.44e-05
Kurtosis:                       5.361   Cond. No.                     1.65e+05
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specif
ied.
[2] The condition number is large, 1.65e+05. This might indicate that there are
strong multicollinearity or other numerical problems.
```

All the columns in X_opt seem to now have p-values less than 5%. Hence we will consider only the one column to be actually helpful in making the model.

Hence, only the R&D Spends column is actually of use.