

# Ad\_budget

June 7, 2020

```
[3]: #import libraries
import pandas as pd
```

```
[5]: #import dataset
df_adv_dataset = pd.read_csv('Advertising Budget and Sales.csv', index_col = 0)
```

```
[7]: #view first 5 records
df_adv_dataset.head()
```

```
[7]:
```

	TV Ad Budget (\$)	Radio Ad Budget (\$)	Newspaper Ad Budget (\$)	Sales (\$)
1	230.1	37.8	69.2	22.1
2	44.5	39.3	45.1	10.4
3	17.2	45.9	69.3	9.3
4	151.5	41.3	58.5	18.5
5	180.8	10.8	58.4	12.9

```
[9]: #view dataset size
df_adv_dataset.size
```

```
[9]: 800
```

```
[12]: #view shape of the dataset
df_adv_dataset.shape
```

```
[12]: (200, 4)
```

```
[15]: #view columns
df_adv_dataset.columns
```

```
[15]: Index(['TV Ad Budget ($)', 'Radio Ad Budget ($)', 'Newspaper Ad Budget ($)',
        'Sales ($)'],
        dtype='object')
```

```
[20]: #create a feature object from columns
X_feature = df_adv_dataset[['TV Ad Budget ($)', 'Newspaper Ad Budget ($)', 'Radio_
    ↳Ad Budget ($)']]
```

```
[22]: #view feature data
X_feature.head()
```

```
[22]:    TV Ad Budget ($)  Newspaper Ad Budget ($)  Radio Ad Budget ($)
1          230.1          69.2          37.8
2           44.5          45.1          39.3
3           17.2          69.3          45.9
4          151.5          58.5          41.3
5          180.8          58.4          10.8
```

```
[24]: #create target object from sales which is a response in the dataset
Y_target = df_adv_dataset[['Sales ($)']]
```

```
[26]: #view target object
Y_target.head()
```

```
[26]:    Sales ($)
1      22.1
2      10.4
3       9.3
4      18.5
5      12.9
```

```
[30]: #view feature object shape
Y_target.shape
```

```
[30]: (200, 1)
```

```
[37]: #split test and training data
# by default training data is 75% and testing data is 25%
from sklearn.model_selection import train_test_split
X_train,X_test,Y_train,Y_test = train_test_split(X_feature,Y_target,
→random_state = 1)
```

```
[41]: # view shape of training and test for both feature and response
print (X_train.shape)
print (Y_train.shape)
print (X_test.shape)
print (Y_test.shape)
```

```
(150, 3)
(150, 1)
(50, 3)
(50, 1)
```

```
[47]: # liner regression model
from sklearn.linear_model import LinearRegression
```

```
#create instance of the model
linearReg = LinearRegression()
#fit the training dataset
linearReg.fit(X_train,Y_train)
```

[47]: LinearRegression(copy\_X=True, fit\_intercept=True, n\_jobs=None, normalize=False)

```
[55]: # print intercept and coefficient
print (linearReg.intercept_)
print (linearReg.coef_)
```

```
[2.87696662]
[[0.04656457 0.00345046 0.17915812]]
```

```
[59]: #prediction
Y_pred = linearReg.predict(X_test)
Y_pred
```

```
[59]: array([[21.70910292],
             [16.41055243],
             [ 7.60955058],
             [17.80769552],
             [18.6146359 ],
             [23.83573998],
             [16.32488681],
             [13.43225536],
             [ 9.17173403],
             [17.333853  ],
             [14.44479482],
             [ 9.83511973],
             [17.18797614],
             [16.73086831],
             [15.05529391],
             [15.61434433],
             [12.42541574],
             [17.17716376],
             [11.08827566],
             [18.00537501],
             [ 9.28438889],
             [12.98458458],
             [ 8.79950614],
             [10.42382499],
             [11.3846456 ],
             [14.98082512],
             [ 9.78853268],
             [19.39643187],
             [18.18099936],
```

```
[17.12807566],  
[21.54670213],  
[14.69809481],  
[16.24641438],  
[12.32114579],  
[19.92422501],  
[15.32498602],  
[13.88726522],  
[10.03162255],  
[20.93105915],  
[ 7.44936831],  
[ 3.64695761],  
[ 7.22020178],  
[ 5.9962782 ],  
[18.43381853],  
[ 8.39408045],  
[14.08371047],  
[15.02195699],  
[20.35836418],  
[20.57036347],  
[19.60636679]])
```

```
[61]: # import required libraries for calculating the MSE  
from sklearn import metrics  
import numpy as np
```

```
[65]: #calculate MSE  
print (np.sqrt(metrics.mean_squared_error(Y_test,Y_pred)))
```

```
1.4046514230328957
```

```
[ ]:
```