exp-4

April 26, 2024

```
[ ]: #exp_4
     #Name:Tushar Holkar
     #Roll No: A-36
```

```
[4]: %matplotlib inline
     import numpy as np
     import pandas as pd
     import matplotlib.pyplot as plt
     plt.rcParams['figure.figsize'] = (20.0, 10.0)

     data = pd.read_csv('/home/kj-comp/Tushar Holkar/GCR/DB/HousingData(1).csv')
     print(data.shape) data.head()
```

```
(506, 14)
```

```
[4]:        CRIM    ZN  INDUS  CHAS    NOX     RM   AGE     DIS  RAD  TAX  PTRATIO  \
     0  0.00632  18.0   2.31   0.0  0.538  6.575  65.2  4.0900    1  296     15.3
     1  0.02731   0.0   7.07   0.0  0.469  6.421  78.9  4.9671    2  242     17.8
     2  0.02729   0.0   7.07   0.0  0.469  7.185  61.1  4.9671    2  242     17.8
     3  0.03237   0.0   2.18   0.0  0.458  6.998  45.8  6.0622    3  222     18.7
     4  0.06905   0.0   2.18   0.0  0.458  7.147  54.2  6.0622    3  222     18.7

             B  LSTAT  MEDV
     0  396.90   4.98  24.0
     1  396.90   9.14  21.6
     2  392.83   4.03  34.7
     3  394.63   2.94  33.4
     4  396.90    NaN  36.2
```

```
[3]: X = data['DIS'].values
     Y = data['MEDV'].values
```

```
[4]: Y
```

```
[4]: array([24. , 21.6, 34.7, 33.4, 36.2, 28.7, 22.9, 27.1, 16.5, 18.9, 15. ,
            18.9, 21.7, 20.4, 18.2, 19.9, 23.1, 17.5, 20.2, 18.2, 13.6, 19.6,
            15.2, 14.5, 15.6, 13.9, 16.6, 14.8, 18.4, 21. , 12.7, 14.5, 13.2,
```

```
         13.1, 13.5, 18.9, 20. , 21. , 24.7, 30.8, 34.9, 26.6, 25.3, 24.7,
         21.2, 19.3, 20. , 16.6, 14.4, 19.4, 19.7, 20.5, 25. , 23.4, 18.9,
         35.4, 24.7, 31.6, 23.3, 19.6, 18.7, 16. , 22.2, 25. , 33. , 23.5,
         19.4, 22. , 17.4, 20.9, 24.2, 21.7, 22.8, 23.4, 24.1, 21.4, 20. ,
         20.8, 21.2, 20.3, 28. , 23.9, 24.8, 22.9, 23.9, 26.6, 22.5, 22.2,
         23.6, 28.7, 22.6, 22. , 22.9, 25. , 20.6, 28.4, 21.4, 38.7, 43.8,
         33.2, 27.5, 26.5, 18.6, 19.3, 20.1, 19.5, 19.5, 20.4, 19.8, 19.4,
         21.7, 22.8, 18.8, 18.7, 18.5, 18.3, 21.2, 19.2, 20.4, 19.3, 22. ,
         20.3, 20.5, 17.3, 18.8, 21.4, 15.7, 16.2, 18. , 14.3, 19.2, 19.6,
         23. , 18.4, 15.6, 18.1, 17.4, 17.1, 13.3, 17.8, 14. , 14.4, 13.4,
         15.6, 11.8, 13.8, 15.6, 14.6, 17.8, 15.4, 21.5, 19.6, 15.3, 19.4,
         17. , 15.6, 13.1, 41.3, 24.3, 23.3, 27. , 50. , 50. , 50. , 22.7,
         25. , 50. , 23.8, 23.8, 22.3, 17.4, 19.1, 23.1, 23.6, 22.6, 29.4,
         23.2, 24.6, 29.9, 37.2, 39.8, 36.2, 37.9, 32.5, 26.4, 29.6, 50. ,
         32. , 29.8, 34.9, 37. , 30.5, 36.4, 31.1, 29.1, 50. , 33.3, 30.3,
         34.6, 34.9, 32.9, 24.1, 42.3, 48.5, 50. , 22.6, 24.4, 22.5, 24.4,
         20. , 21.7, 19.3, 22.4, 28.1, 23.7, 25. , 23.3, 28.7, 21.5, 23. ,
         26.7, 21.7, 27.5, 30.1, 44.8, 50. , 37.6, 31.6, 46.7, 31.5, 24.3,
         31.7, 41.7, 48.3, 29. , 24. , 25.1, 31.5, 23.7, 23.3, 22. , 20.1,
         22.2, 23.7, 17.6, 18.5, 24.3, 20.5, 24.5, 26.2, 24.4, 24.8, 29.6,
         42.8, 21.9, 20.9, 44. , 50. , 36. , 30.1, 33.8, 43.1, 48.8, 31. ,
         36.5, 22.8, 30.7, 50. , 43.5, 20.7, 21.1, 25.2, 24.4, 35.2, 32.4,
         32. , 33.2, 33.1, 29.1, 35.1, 45.4, 35.4, 46. , 50. , 32.2, 22. ,
         20.1, 23.2, 22.3, 24.8, 28.5, 37.3, 27.9, 23.9, 21.7, 28.6, 27.1,
         20.3, 22.5, 29. , 24.8, 22. , 26.4, 33.1, 36.1, 28.4, 33.4, 28.2,
         22.8, 20.3, 16.1, 22.1, 19.4, 21.6, 23.8, 16.2, 17.8, 19.8, 23.1,
         21. , 23.8, 23.1, 20.4, 18.5, 25. , 24.6, 23. , 22.2, 19.3, 22.6,
         19.8, 17.1, 19.4, 22.2, 20.7, 21.1, 19.5, 18.5, 20.6, 19. , 18.7,
         32.7, 16.5, 23.9, 31.2, 17.5, 17.2, 23.1, 24.5, 26.6, 22.9, 24.1,
         18.6, 30.1, 18.2, 20.6, 17.8, 21.7, 22.7, 22.6, 25. , 19.9, 20.8,
         16.8, 21.9, 27.5, 21.9, 23.1, 50. , 50. , 50. , 50. , 50. , 13.8,
         13.8, 15. , 13.9, 13.3, 13.1, 10.2, 10.4, 10.9, 11.3, 12.3,  8.8,
          7.2, 10.5,  7.4, 10.2, 11.5, 15.1, 23.2,  9.7, 13.8, 12.7, 13.1,
         12.5,  8.5,  5. ,  6.3,  5.6,  7.2, 12.1,  8.3,  8.5,  5. , 11.9,
         27.9, 17.2, 27.5, 15. , 17.2, 17.9, 16.3,  7. ,  7.2,  7.5, 10.4,
          8.8,  8.4, 16.7, 14.2, 20.8, 13.4, 11.7,  8.3, 10.2, 10.9, 11. ,
          9.5, 14.5, 14.1, 16.1, 14.3, 11.7, 13.4,  9.6,  8.7,  8.4, 12.8,
         10.5, 17.1, 18.4, 15.4, 10.8, 11.8, 14.9, 12.6, 14.1, 13. , 13.4,
         15.2, 16.1, 17.8, 14.9, 14.1, 12.7, 13.5, 14.9, 20. , 16.4, 17.7,
         19.5, 20.2, 21.4, 19.9, 19. , 19.1, 19.1, 20.1, 19.9, 19.6, 23.2,
         29.8, 13.8, 13.3, 16.7, 12. , 14.6, 21.4, 23. , 23.7, 25. , 21.8,
         20.6, 21.2, 19.1, 20.6, 15.2,  7. ,  8.1, 13.6, 20.1, 21.8, 24.5,
         23.1, 19.7, 18.3, 21.2, 17.5, 16.8, 22.4, 20.6, 23.9, 22. , 11.9])
```

```
[5]: mean_x = np.mean(X)
     mean_y = np.mean(Y)
     # Total number of values
```

```
n = len(X)
```

[6]:
```
n
```

[6]: 506

[10]:
```
# Using the formula to calculate b1 and b2
numer = 0
denom = 0
for i in range(n):
  numer += (X[i] - mean_x) * (Y[i] - mean_y)
  denom += (X[i] - mean_x) ** 2
b1 = numer / denom
b0 = mean_y - (b1 * mean_x)
#m(b1) and c(bo)
# Printing coefficients
print("Coefficients")
print("m=",b1)
print("c=",b0)
```

```
Coefficients
m= 1.0916130158411097
c= 18.390088330493384
```

[11]:
```
# Plotting Values and Regression Line
max_x = np.max(X)
min_x = np.min(X)
# Calculating line values x and y
x = np.linspace(min_x, max_x, 1000)
y = b0 + b1 * x
# Ploting Line
#plt.plot(x, y, color='#58b970', label='Regression Line')
plt.plot(x, y, color='green', label='Regression Line')
# Ploting Scatter Points
#plt.scatter(X, Y, c='#ef5423', label='Scatter Plot')
plt.scatter(X, Y, c='red', label='Scatter Plot')
plt.xlabel('Head Size in cm3')
plt.ylabel('Brain Weight in grams')
plt.legend()
plt.show()
```
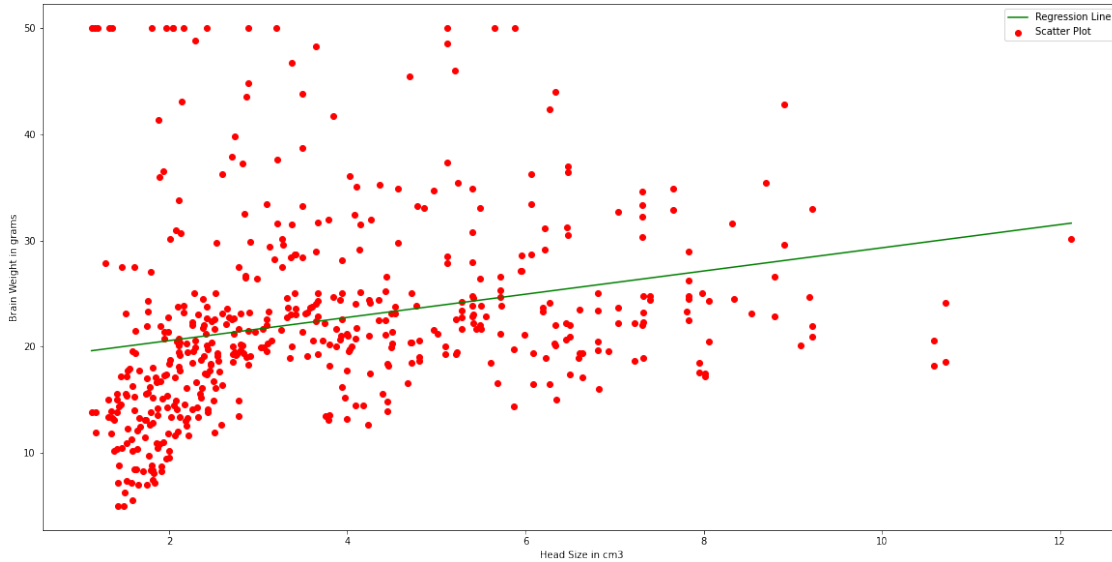
```python
[13]: ss_tot = 0
      ss_res = 0
      for i in range(n):
       y_pred = b0 + b1 * X[i]
       ss_tot += (Y[i] - mean_y) ** 2
       ss_res += (Y[i] - y_pred) ** 2
      r2 = 1 - (ss_res/ss_tot)
      print("R2 Score")
      print(r2)
```

```
R2 Score
0.06246437212178291
```

```python
[14]: import numpy as np
      import matplotlib.pyplot as plt
      import pandas as pd
      from sklearn.linear_model import LinearRegression
```

```python
[16]: data=pd.read_csv('/home/kj-comp/HousingData.csv')
      X = data.iloc[:,7].values.reshape(-1,1) #converts it into numpy array
      Y = data.iloc[:,13].values.reshape(-1,1)
      linear_regressor=LinearRegression() # create obect for class
      linear_regressor.fit(X,Y) # perform linear regression
      y_pred=linear_regressor.predict(X) # make prediction
```

```python
[17]: plt.scatter(X,Y)
      plt.plot(X,y_pred, color='red')
```

[17]: [<matplotlib.lines.Line2D at 0x7faecffa9160>]
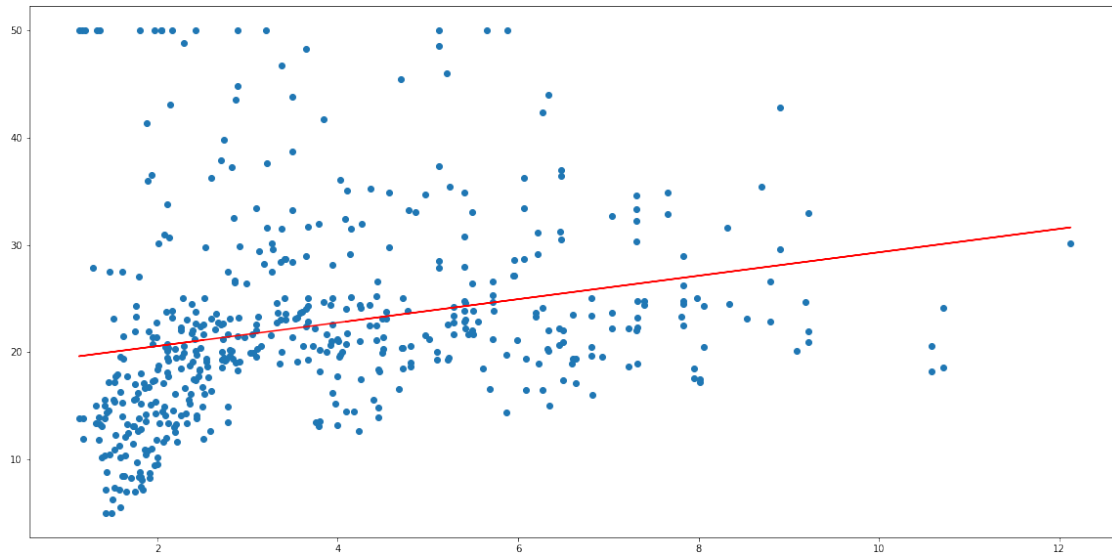
[18]:
```python
print("Coefficients: \n", linear_regressor.coef_)
```

Coefficients:
 [[1.09161302]]

[19]:
```python
from sklearn.metrics import mean_squared_error, r2_score
print("Coefficient of determination: %.2f" % r2_score(Y, y_pred))
```

Coefficient of determination: 0.06

[ ]: