**Liner Regression**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

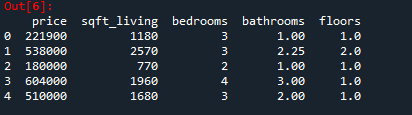
from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LinearRegression

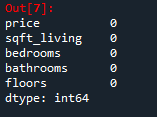
from sklearn.metrics import mean\_squared\_error, r2\_score

**dataset=pd.read\_excel("Linear Regression.xlsx")**

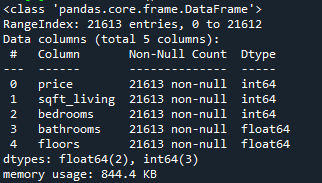
**dataset.head()**

****

**dataset.isna().sum()**

****

**dataset.info()**

****

**dataset.hist(figsize=(20,20))**

Out[10]:

array([[<matplotlib.axes.\_subplots.AxesSubplot object at 0x0000000007AA9208>,

<matplotlib.axes.\_subplots.AxesSubplot object at 0x000000000D27CB48>],

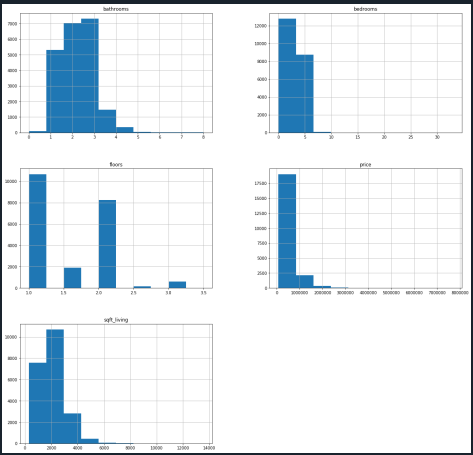
[<matplotlib.axes.\_subplots.AxesSubplot object at 0x000000000D214188>,

<matplotlib.axes.\_subplots.AxesSubplot object at 0x000000000CD94948>],

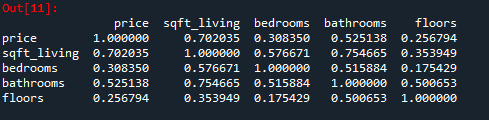
[<matplotlib.axes.\_subplots.AxesSubplot object at 0x000000000CF92E48>,

<matplotlib.axes.\_subplots.AxesSubplot object at 0x000000000CE02748>]],

dtype=object)



**dataset.corr()**

****

**Price vs sqft\_living:**

**y2=dataset.iloc[:,:1]**

y2.head()

Out[14]:

price

0 221900

1 538000

2 180000

3 604000

4 510000

**x3=dataset.iloc[:,1:2]**

x3.head()

Out[16]:

sqft\_living

0 1180

1 2570

2 770

3 1960

4 1680

**plt.scatter(x3,y2),plt.title('Price variation with square feet of area'),plt.ylabel('Price of the House'),plt.xlabel('Area in squarefeet')**

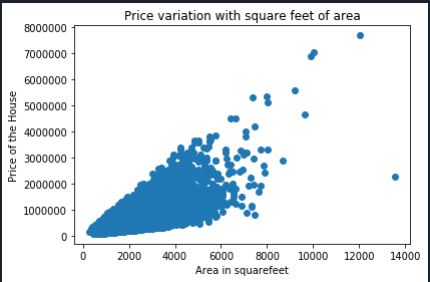
Out[18]:

(<matplotlib.collections.PathCollection at 0xcce6308>,

Text(0.5, 1.0, 'Price variation with square feet of area'),

Text(0, 0.5, 'Price of the House'),

Text(0.5, 0, 'Area in squarefeet'))



**x\_train,x\_test,y\_train,y\_test=train\_test\_split(x3,y2,test\_size=0.2,random\_state=2)**

**lin\_reg=LinearRegression()**

**lin\_reg.fit(x\_train,y\_train)**

Out[25]: LinearRegression(copy\_X=True, fit\_intercept=True, n\_jobs=None, normalize=False)

**lin\_reg.coef\_**

Out[26]: array([[280.67382569]])

**lin\_reg.intercept\_**

Out[27]: array([-42568.70358496])

**plt.scatter(x\_train,y\_train,color='green'),plt.plot(x\_train,lin\_reg.predict(x\_train),color='blue'),plt.title('Price variation with squarefeet of area'),plt.ylabel('Price of the House'),plt.xlabel('Area in squarefeet')**

Out[28]:

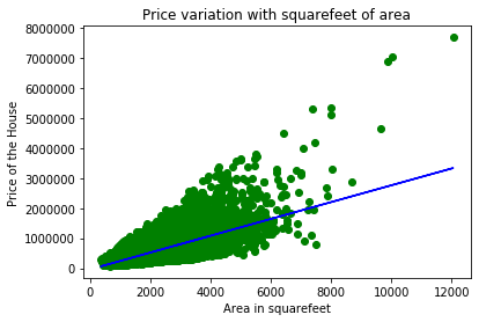
(<matplotlib.collections.PathCollection at 0xcad8088>,

[<matplotlib.lines.Line2D at 0xcdc95c8>],

Text(0.5, 1.0, 'Price variation with squarefeet of area'),

Text(0, 0.5, 'Price of the House'),

Text(0.5, 0, 'Area in squarefeet'))



**plt.scatter(x\_test,y\_test,color='green'),plt.plot(x\_test,lin\_reg.predict(x\_test),color='blue'),plt.title('Price variation with squarefeet of area'),plt.ylabel('Price of the House'),plt.xlabel('Area in squarefeet')**

Out[31]:

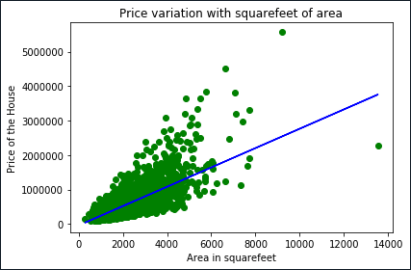
(<matplotlib.collections.PathCollection at 0xc573fc8>,

[<matplotlib.lines.Line2D at 0xcc521c8>],

Text(0.5, 1.0, 'Price variation with squarefeet of area'),

Text(0, 0.5, 'Price of the House'),

Text(0.5, 0, 'Area in squarefeet'))



**RMSE=np.sqrt(mean\_squared\_error(y\_test,ypred))**

**r\_square=r2\_score(y\_test,ypred)**

print("The R Square value is:",r\_square)

**The R Square value is: 0.5031163723285275**

print("The RMSE value is:",RMSE)

**The RMSE value is: 263380.00189817196**

**Inference:**

**R-Square value is 0.50311, which means 50% of the variance in the Price can be explained by the Square feet Area. Hence, it is called a Good Fit.**

**Price and bedroom:**

**y2=dataset.iloc[:,:1]**

**y2.head()**

Out[39]:

price

0 221900

1 538000

2 180000

3 604000

4 510000

**x4=dataset.iloc[:,2:3]**

**x4.head()**

Out[42]:

bedrooms

0 3

1 3

2 2

3 4

4 3

**plt.scatter(x4,y2),plt.title('Price variation with bedrooms'),plt.ylabel('Price of the House'),plt.xlabel('Bedrooms')**

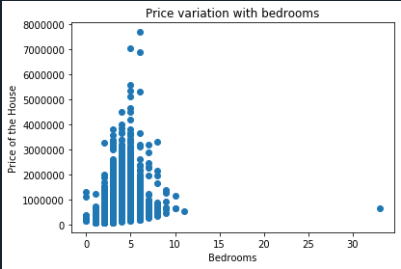
Out[43]:

(<matplotlib.collections.PathCollection at 0xce869c8>,

Text(0.5, 1.0, 'Price variation with bedrooms'),

Text(0, 0.5, 'Price of the House'),

Text(0.5, 0, 'Bedrooms'))



**x\_train,x\_test,y\_train,y\_test=train\_test\_split(x4,y2,test\_size=0.2,random\_state=2)**

**lin\_reg=LinearRegression()**

**lin\_reg.fit(x\_train,y\_train)**

Out[46]: LinearRegression(copy\_X=True, fit\_intercept=True, n\_jobs=None, normalize=False)

**lin\_reg.coef\_**

Out[47]: array([[118660.62797869]])

**lin\_reg.intercept\_**

Out[48]: array([139952.87593386])

**ypred=lin\_reg.predict(x\_test)**

**RMSE=np.sqrt(mean\_squared\_error(y\_test,ypred))**

**r\_square=r2\_score(y\_test,ypred)**

print("The R Square value is:",r\_square)

**The R Square value is: 0.10886345250291596**

print("The RMSE value is:",RMSE)

**The RMSE value is: 352717.96541876445**

**Inference:**

**R- Square value is 0.1, which means 10% of the variance in the price can be explained by the Bedrooms. Hence, it is called a Poor Fit.**

**Price and Bathroom:**

**y2=dataset.iloc[:,:1]**

**x5=dataset.iloc[:,3:4]**

**x5.head()**

Out[56]:

bathrooms

0 1.00

1 2.25

2 1.00

3 3.00

4 2.00

**x\_train,x\_test,y\_train,y\_test=train\_test\_split(x5,y2,test\_size=0.2,random\_state=2)**

**lin\_reg=LinearRegression()**

**lin\_reg.fit(x\_train,y\_train)**

Out[60]: LinearRegression(copy\_X=True, fit\_intercept=True, n\_jobs=None, normalize=False)

**lin\_reg.coef\_**

Out[61]: array([[249143.95803858]])

**lin\_reg.intercept\_**

Out[62]: array([13073.9957529])

**ypred=lin\_reg.predict(x\_test)**

**RMSE=np.sqrt(mean\_squared\_error(y\_test,ypred))**

**r\_square=r2\_score(y\_test,ypred)**

print("The R Square value is:",r\_square)

**The R Square value is: 0.28122887124177365**

print("The RMSE value is:",RMSE)

**The RMSE value is: 316774.90190998075**

**Inference:**

**R -Square value as 0.28, which means 28% of the variance in the Price can be explained by the Bathrooms. Hence,it is called a Poor Fit.**

**Price and floor:**

**y2=dataset.iloc[:,:1]**

**x6=dataset.iloc[:,4:5]**

**x6.head()**

Out[70]:

floors

0 1.0

1 2.0

2 1.0

3 1.0

4 1.0

**x\_train,x\_test,y\_train,y\_test=train\_test\_split(x6,y2,test\_size=0.2,random\_state=2)**

**lin\_reg=LinearRegression()**

**lin\_reg.fit(x\_train,y\_train)**

Out[73]: LinearRegression(copy\_X=True, fit\_intercept=True, n\_jobs=None, normalize=False)

**lin\_reg.coef\_**

Out[75]: array([[171376.44562902]])

**lin\_reg.intercept\_**

Out[76]: array([283309.93245028])

**ypred=lin\_reg.predict(x\_test)**

**RMSE=np.sqrt(mean\_squared\_error(y\_test,ypred))**

**r\_square=r2\_score(y\_test,ypred)**

print("The R Square value is:",r\_square)

**The R Square value is: 0.0733487976687478**

print("The RMSE value is:",RMSE)

**The RMSE value is: 359677.77234107786**

**Inference:**

**R-Square value as 0.07, which means 7% of the variance in the Price can be explained by the Floors. Hence, it is called a Poor Fit.**