# 1r3xrg4gk

August 14, 2024

## 0.1 Case: Build machine learning models for prediction Regression Task

## 0.2 Dataset: House_Price_prediction.csv

## 0.3 Problem Statement: Analyse the dataset and perform the steps below to build linear regression machine

learning model

## 0.4 Import Libraries

[1]:
```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import seaborn as sns
%matplotlib inline
```

[2]:
```
#Import Dataset
```

[3]:
```
m1=pd.read_csv("House_Price_prediction.csv")
print(m1)
```

```
     Unnamed: 0   price   lotsize  bedrooms  bathrms  stories  garagepl  \
0             0   42000  8.674197         3        1        2         1
1             1   38500  8.294050         2        1        1         0
2             2   49500  8.026170         3        1        1         0
3             3   60500  8.802372         3        1        2         0
4             4   61000  8.757784         3        1        1         0
..          …       …       …       …       …       …        …
546         546  107500  8.699515         3        2        4         1
547         547  108000  8.699515         3        2        3         0
548         548  113750  8.699515         3        1        4         2
549         549  120000  8.853665         3        1        4         2
550         550   70000  9.464983         3        1        1         2

     driveway_yes  recroom_yes  fullbase_yes  gashw_yes  airco_yes  \
0               1            0             1          0          0
1               1            0             0          0          0
2               1            0             0          0          0
```

```
3                 1            1            0          0          0
4                 1            0            0          0          0
..              ...          ...          ...        ...        ...
546               1            0            0          0          1
547               1            0            0          0          1
548               1            1            0          0          1
549               1            0            0          0          1
550               1            0            0          0          0

     prefarea_yes
0               0
1               0
2               0
3               0
4               0
..            ...
546             0
547             0
548             0
549             0
550             0

[551 rows x 13 columns]
```

[4]:
```
#Exploratory Data Analysis(EDA)
```

[5]:
```
m1.head()
```

[5]:
```
   Unnamed: 0  price  lotsize  bedrooms  bathrms  stories  garagepl  \
0           0  42000  8.674197         3        1        2         1
1           1  38500  8.294050         2        1        1         0
2           2  49500  8.026170         3        1        1         0
3           3  60500  8.802372         3        1        2         0
4           4  61000  8.757784         3        1        1         0

   driveway_yes  recroom_yes  fullbase_yes  gashw_yes  airco_yes  prefarea_yes
0             1            0             1          0          0             0
1             1            0             0          0          0             0
2             1            0             0          0          0             0
3             1            1             0          0          0             0
4             1            0             0          0          0             0
```

[6]:
```
m1.describe()
```

[6]:
```
       Unnamed: 0         price    lotsize    bedrooms     bathrms  \
count  551.000000    551.000000  551.000000  551.000000  551.000000
mean   275.000000  68445.811252    8.470413    2.967332    1.286751
```

```
std    159.204271   26848.486040    0.399086    0.732880    0.502165
min      0.000000   25000.000000    7.408531    1.000000    1.000000
25%    137.500000   49500.000000    8.188689    3.000000    1.000000
50%    275.000000   62500.000000    8.433812    3.000000    1.000000
75%    412.500000   82950.000000    8.757784    3.000000    2.000000
max    550.000000  190000.000000    9.692767    6.000000    4.000000

          stories    garagepl  driveway_yes  recroom_yes  fullbase_yes  \
count  551.000000  551.000000    551.000000   551.000000    551.000000
mean     1.820327    0.698730      0.860254     0.177858      0.346642
std      0.881334    0.863386      0.347038     0.382741      0.476333
min      1.000000    0.000000      0.000000     0.000000      0.000000
25%      1.000000    0.000000      1.000000     0.000000      0.000000
50%      2.000000    0.000000      1.000000     0.000000      0.000000
75%      2.000000    1.000000      1.000000     0.000000      1.000000
max      4.000000    3.000000      1.000000     1.000000      1.000000

        gashw_yes   airco_yes  prefarea_yes
count  551.000000  551.000000    551.000000
mean     0.045372    0.321234      0.232305
std      0.208308    0.467375      0.422686
min      0.000000    0.000000      0.000000
25%      0.000000    0.000000      0.000000
50%      0.000000    0.000000      0.000000
75%      0.000000    1.000000      0.000000
max      1.000000    1.000000      1.000000
```

[7]:
```
m1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 551 entries, 0 to 550
Data columns (total 13 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Unnamed: 0    551 non-null    int64
 1   price         551 non-null    int64
 2   lotsize       551 non-null    float64
 3   bedrooms      551 non-null    int64
 4   bathrms       551 non-null    int64
 5   stories       551 non-null    int64
 6   garagepl      551 non-null    int64
 7   driveway_yes  551 non-null    int64
 8   recroom_yes   551 non-null    int64
 9   fullbase_yes  551 non-null    int64
 10  gashw_yes     551 non-null    int64
 11  airco_yes     551 non-null    int64
 12  prefarea_yes  551 non-null    int64
```

```
dtypes: float64(1), int64(12)
memory usage: 56.1 KB
```

[8]:
```python
print(m1.shape)
```

```
(551, 13)
```

[9]:
```python
m1.corr()
```

[9]:
```
              Unnamed: 0     price  lotsize  bedrooms   bathrms   stories  \
Unnamed: 0      1.000000  0.387924  0.386297  0.109361  0.109914  0.248420
price           0.387924  1.000000  0.560017  0.363247  0.513014  0.435332
lotsize         0.386297  0.560017  1.000000  0.151814  0.198791  0.112181
bedrooms        0.109361  0.363247  0.151814  1.000000  0.371325  0.399058
bathrms         0.109914  0.513014  0.198791  0.371325  1.000000  0.322034
stories         0.248420  0.435332  0.112181  0.399058  0.322034  1.000000
garagepl        0.135595  0.385734  0.365816  0.136709  0.170263  0.052983
driveway_yes    0.315854  0.298859  0.332750 -0.010833  0.042566  0.125817
recroom_yes     0.095841  0.253611  0.176168  0.079088  0.122017  0.046397
fullbase_yes   -0.013450  0.175110  0.035777  0.094997  0.100567 -0.180526
gashw_yes      -0.036458  0.089257 -0.015737  0.045456  0.066592  0.014774
airco_yes       0.169776  0.462458  0.262216  0.158087  0.187823  0.312520
prefarea_yes    0.503575  0.318696  0.212355  0.077366  0.062495  0.034156

              garagepl  driveway_yes  recroom_yes  fullbase_yes  gashw_yes  \
Unnamed: 0    0.135595      0.315854     0.095841     -0.013450  -0.036458
price         0.385734      0.298859     0.253611      0.175110   0.089257
lotsize       0.365816      0.332750     0.176168      0.035777  -0.015737
bedrooms      0.136709     -0.010833     0.079088      0.094997   0.045456
bathrms       0.170263      0.042566     0.122017      0.100567   0.066592
stories       0.052983      0.125817     0.046397     -0.180526   0.014774
garagepl      1.000000      0.205116     0.041400      0.046609   0.066032
driveway_yes  0.205116      1.000000     0.091646      0.040602  -0.012735
recroom_yes   0.041400      0.091646     1.000000      0.369287  -0.010181
fullbase_yes  0.046609      0.040602     0.369287      1.000000   0.006119
gashw_yes     0.066032     -0.012735    -0.010181      0.006119   1.000000
airco_yes     0.159165      0.109127     0.137408      0.037930  -0.131303
prefarea_yes  0.087499      0.196923     0.159972      0.231448  -0.057977

              airco_yes  prefarea_yes
Unnamed: 0     0.169776      0.503575
price          0.462458      0.318696
lotsize        0.262216      0.212355
bedrooms       0.158087      0.077366
bathrms        0.187823      0.062495
stories        0.312520      0.034156
garagepl       0.159165      0.087499
```

```
driveway_yes   0.109127    0.196923
recroom_yes    0.137408    0.159972
fullbase_yes   0.037930    0.231448
gashw_yes     -0.131303   -0.057977
airco_yes      1.000000    0.109357
prefarea_yes   0.109357    1.000000
```

#Splitting Data into Training and Testing Dataset

[10]:
```
x=m1[['price','lotsize','bedrooms','bathrms','stories','garagepl','driveway_yes','recroom_yes'
↪variables
```

[11]:
```
y=m1.prefarea_yes #dependant variable
```

[12]:
```
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=1␣
↪)
```

[13]:
```
print(x_train.shape)
print(x_test.shape)
print(y_train.shape)
print(y_test.shape)
```

```
(440, 11)
(111, 11)
(440,)
(111,)
```

[14]:
```
print(x_train)
print(y_train)
```

```
      price   lotsize  bedrooms  bathrms  stories  garagepl  driveway_yes  \
92   163000  8.911934         4        1        2         2             1
66    60000  8.525161         3        1        2         0             1
201   53900  7.832014         5        2        1         1             0
397   80750  8.803875         4        2        2         1             1
521  105000  8.699515         4        2        4         1             1
..      …         …       …       …        …         …
129  127000  8.433812         3        2        2         2             1
144   57250  8.411833         3        1        2         0             0
72    32500  7.515345         2        1        1         0             0
235   42500  8.378391         4        1        2         1             0
37    67000  8.550628         3        1        4         0             1

     recroom_yes  fullbase_yes  gashw_yes  airco_yes
92             1             1          0          1
66             0             1          0          1
201            0             1          0          1
```

```
397            1            1          0          0
521            1            0          0          1
..           …            …          …          …
129            1            0          0          1
144            0            1          0          1
72             0            1          0          0
235            0            0          0          0
37             0            0          0          1

[440 rows x 11 columns]
92     0
66     0
201    0
397    1
521    0
      ..
129    0
144    0
72     0
235    0
37     0
Name: prefarea_yes, Length: 440, dtype: int64
```

[15]:
```
# Model buiding | model training
```

[16]:
```
from sklearn.linear_model import LinearRegression
linreg=LinearRegression()
linreg.fit(x_train,y_train) # Model buiding | model training

#X_TRAIN =440 rows x 11 columns
#Y_TRAIN= 440 RECORDS, 1 COLUMNS
```

[16]:
```
LinearRegression()
```

[17]:
```
print(linreg.intercept_)
```

```
-0.545539630293612
```

[18]:
```
print(linreg.coef_) # Next we try to obtain a1,a2,.....a11 and also b
```

```
[ 4.86545317e-06  5.36246111e-02  1.57066101e-03 -8.32654349e-02
 -2.47227400e-02 -3.36147082e-02  9.81946453e-02  5.34496848e-02
  1.53422486e-01 -1.91696418e-01  2.89096643e-02]
```

[19]:
```
print("x_test:(price,␣
↪lotsize','bedrooms','bathrms','stories','garagepl','driveway_yes','recroom_yes','fullbase_y
print(x_test)
```

```
#Sales=0.0468431 X TV + 0.17854434  X Radio + 0.00258619 X Newspaper + 2.
↪9079470208164313
print("y_test: (prefarea_yes-->actual values)")
print(y_test)
```

```
x_test:(price, lotsize','bedrooms','bathrms','stories','garagepl','driveway_yes'
,'recroom_yes','fullbase_yes','gashw_yes','airco_yes)
     price   lotsize  bedrooms  bathrms  stories  garagepl  driveway_yes  \
160  63900  8.058960         3        1        2         1             1
306  67000  9.176473         4        2        2         1             1
65   60000  8.612503         3        1        2         0             1
423  62900  7.965546         3        1        2         0             1
135  90000  8.699515         4        2        4         1             1
..     …        …       …      …       …       …         …
408  89000  8.794825         3        2        1         0             1
17   40750  8.556414         4        1        3         0             1
247  42000  8.797095         3        1        2         0             1
124  70000  8.323608         2        1        1         1             1
257  75500  8.433812         2        2        1         2             1

     recroom_yes  fullbase_yes  gashw_yes  airco_yes
160            0             0          0          1
306            1             1          0          0
65             0             0          0          1
423            0             0          0          0
135            0             0          0          0
..           …            …         …         …
408            0             1          0          1
17             0             0          0          0
247            0             0          0          0
124            0             1          0          0
257            0             0          0          1

[111 rows x 11 columns]
y_test: (prefarea_yes-->actual values)
160    0
306    0
65     0
423    1
135    0
      ..
408    1
17     0
247    0
124    0
257    0
Name: prefarea_yes, Length: 111, dtype: int64
```

[20]:
```python
# Making predictions using the built model on test dataset
```

[21]:
```python
y_pred=linreg.predict(x_test)  #x_test-->- 111 records ,(actual values)␣
 ↪y_test-->sales
```

[22]:
```python
print(y_pred)
```

```
[ 0.16301209  0.33428895  0.20733508  0.15784237  0.1643      0.38864294
  0.3769083   0.0515332   0.22644814  0.34188307  0.0496893   0.18451213
 -0.06550692  0.2897895   0.16202616  0.04709468  0.27923778  0.22507599
  0.12653864  0.38901906  0.21396333  0.2833614   0.22425638  0.26118611
  0.48653114  0.22415522  0.42025623  0.66789153  0.32314802  0.23312456
 -0.09947451  0.44921925  0.24414098  0.2421242   0.14290019  0.15230468
 -0.15590387  0.02693835  0.05212203  0.39888901  0.25175938  0.17481732
  0.3946586  -0.039609   -0.03340065 -0.010992    0.06631354  0.22592543
  0.47371444  0.35196897  0.03637799  0.35767401  0.37024095  0.18951081
  0.1341406   0.21810362  0.07814129  0.36179345  0.53737528  0.17067184
  0.49532839  0.20492066  0.11088533  0.42204509 -0.02635494  0.05905336
  0.22814137  0.12661147  0.134337    0.05050866 -0.04168664 -0.0258426
  0.32222521  0.18876281  0.41483327  0.33644101  0.13880515 -0.00972749
  0.17393439  0.24448947  0.0267609   0.27678444  0.07057038  0.21580421
  0.36978681  0.29810874  0.42789253  0.29007743  0.10312811 -0.02761722
  0.11159785  0.30699882  0.197092   -0.04589009  0.15079872  0.29827833
  0.15449345  0.02489536  0.30511509  0.24430335  0.19575923  0.47936997
  0.45314895  0.12874258  0.32123281  0.14540722  0.45308994  0.05860559
  0.10074592  0.35454793  0.14582456]
```

[23]:
```python
print(y_test)
```

```
160    0
306    0
65     0
423    1
135    0
      ..
408    1
17     0
247    0
124    0
257    0
Name: prefarea_yes, Length: 111, dtype: int64
```

[24]:
```python
from sklearn.linear_model import LinearRegression
lr=LinearRegression()  # This is an object of the LinearRegression Class
lr.fit(x_train,y_train) # This trains our model with the 440 training records
```

[24]:
```
LinearRegression()
```

[25]:
```
print(lr.intercept_)
print(lr.coef_)
```

```
-0.545539630293612
[ 4.86545317e-06  5.36246111e-02  1.57066101e-03 -8.32654349e-02
 -2.47227400e-02 -3.36147082e-02  9.81946453e-02  5.34496848e-02
  1.53422486e-01 -1.91696418e-01  2.89096643e-02]
```

[26]:
```
y_pred=lr.predict(x_test)
```

[27]:
```
print(y_pred)
```

```
[ 0.16301209  0.33428895  0.20733508  0.15784237  0.1643      0.38864294
  0.3769083   0.0515332   0.22644814  0.34188307  0.0496893   0.18451213
 -0.06550692  0.2897895   0.16202616  0.04709468  0.27923778  0.22507599
  0.12653864  0.38901906  0.21396333  0.2833614   0.22425638  0.26118611
  0.48653114  0.22415522  0.42025623  0.66789153  0.32314802  0.23312456
 -0.09947451  0.44921925  0.24414098  0.2421242   0.14290019  0.15230468
 -0.15590387  0.02693835  0.05212203  0.39888901  0.25175938  0.17481732
  0.3946586  -0.039609   -0.03340065 -0.010992    0.06631354  0.22592543
  0.47371444  0.35196897  0.03637799  0.35767401  0.37024095  0.18951081
  0.1341406   0.21810362  0.07814129  0.36179345  0.53737528  0.17067184
  0.49532839  0.20492066  0.11088533  0.42204509 -0.02635494  0.05905336
  0.22814137  0.12661147  0.134337    0.05050866 -0.04168664 -0.0258426
  0.32222521  0.18876281  0.41483327  0.33644101  0.13880515 -0.00972749
  0.17393439  0.24448947  0.0267609   0.27678444  0.07057038  0.21580421
  0.36978681  0.29810874  0.42789253  0.29007743  0.10312811 -0.02761722
  0.11159785  0.30699882  0.197092   -0.04589009  0.15079872  0.29827833
  0.15449345  0.02489536  0.30511509  0.24430335  0.19575923  0.47936997
  0.45314895  0.12874258  0.32123281  0.14540722  0.45308994  0.05860559
  0.10074592  0.35454793  0.14582456]
```

[28]:
```
# Next we compare our results to see how our model has perfomed
df=pd.DataFrame({'Actual':y_test,'Predicted':y_pred,'Difference +/-':
 ↪y_test-y_pred})
print(df)
```

```
     Actual  Predicted  Difference +/-
160       0   0.163012       -0.163012
306       0   0.334289       -0.334289
65        0   0.207335       -0.207335
423       1   0.157842        0.842158
135       0   0.164300       -0.164300
..      …         …             …
408       1   0.453090        0.546910
17        0   0.058606       -0.058606
247       0   0.100746       -0.100746
124       0   0.354548       -0.354548
```

```
257        0   0.145825       -0.145825

[111 rows x 3 columns]
```

[29]:
```
y_train_pred=lr.predict(x_train)
print(y_train_pred)
```

```
[ 8.65747042e-01  3.56073886e-01  1.02014147e-01  3.81208476e-01
  3.19641148e-01  3.42863991e-01  1.15236254e-01  9.82600418e-02
  5.79238925e-02  8.65685027e-02  2.61279992e-01  2.71129466e-01
  1.23105851e-01  1.65350483e-01 -7.56530984e-02  6.97223258e-02
  2.76784436e-01  7.73194520e-02  5.57535647e-01  2.99946982e-01
  3.76226573e-01  2.27994357e-01  9.02836738e-02  4.78182688e-01
  5.96232768e-01  1.51108519e-01  7.25765979e-02  1.79387983e-02
  1.91356728e-01  1.49657903e-01  4.40436942e-01  3.17879233e-01
  1.28373746e-01  8.34867444e-02  1.91835450e-01 -7.26518143e-02
  2.87217730e-01  3.37554610e-01  9.68046337e-02  4.72558285e-02
  2.67143520e-01  2.15906004e-01 -5.29296831e-02  3.63109600e-01
  3.33272127e-01  2.30367818e-01  2.33856988e-01  3.56236912e-01
  5.50516843e-01  1.61235181e-01  1.02015921e-01  5.45812354e-01
  1.95100129e-01  7.29711075e-01  3.86136476e-01  3.62594997e-01
  4.10293929e-01  3.28149015e-01  1.44728240e-01 -2.08542858e-02
  2.55592351e-02  4.51038619e-01  3.61589328e-01  2.55175628e-01
 -2.67952925e-02  1.42359723e-01  5.59405683e-01  2.82080438e-01
  3.62734142e-01  1.21783230e-01  6.49256415e-01 -4.94684300e-02
  3.52719763e-01  1.54493744e-01  7.51749029e-02  5.29385446e-02
  2.08379398e-01  3.55698526e-01  1.38226621e-02 -1.03126546e-01
  6.46230633e-02  4.22301695e-01  2.25828752e-01  1.10780756e-01
  7.77060872e-02  3.07768627e-01  1.10663357e-01  7.65683820e-02
 -2.29884789e-02 -4.65102244e-02  4.83935070e-01  1.35330504e-01
  2.58802340e-01  2.56345424e-01  1.16892466e-01  2.44178513e-01
  4.55480208e-02  2.80787823e-01  2.25865438e-01  1.09153325e-01
  3.55580419e-01  4.03796546e-01  3.35418331e-01  2.03543451e-01
  2.21022892e-01  6.95945774e-02  2.19962434e-01  3.86134815e-01
  1.07783673e-01  7.98932956e-02  2.95118617e-01  3.00520702e-01
  2.99683698e-01  5.82349548e-02  9.10397021e-03 -2.52535536e-01
  4.86079106e-01  1.80567283e-01 -2.78579958e-01  6.11436821e-01
  9.06012486e-02  3.48848877e-01  2.09528017e-01  1.12377545e-01
  1.64466896e-01  4.87119493e-01  1.43614898e-01  2.68685781e-01
  6.16251284e-02  2.38529604e-01  2.84541340e-01  5.76228355e-01
 -7.07510133e-02  2.99011954e-01 -1.20079607e-02  1.63294982e-01
 -4.50962572e-02  4.54283527e-01  2.58016446e-01  3.66388200e-01
  3.14956358e-01  2.30287139e-01  3.87996285e-01 -3.86922813e-02
  8.98413609e-02  4.36674824e-01  1.09865147e-01  2.09216294e-01
  1.39577719e-01  6.69451992e-02  3.24009944e-01  5.35649986e-01
  3.45012310e-01  8.38843380e-02  2.51329239e-01  4.86528081e-01
  1.71714408e-01  1.35522968e-01 -1.25907382e-01  2.29597473e-01
 -4.84238160e-03  3.57918612e-01  1.61479213e-01  1.50496284e-01
```

| | | | |
|---|---|---|---|
| 1.41334012e-01 | 2.54600879e-01 | 1.09282515e-01 | 2.44715526e-01 |
| 1.13580856e-01 | 4.64207744e-01 | 4.71043253e-01 | 3.45809755e-01 |
| 9.52542713e-02 | 1.72154435e-01 | 1.19315047e-01 | 1.30501617e-01 |
| -1.02433169e-01 | 5.99046749e-02 | 6.36965900e-02 | 3.40584776e-02 |
| 7.23847497e-02 | -1.37064508e-01 | 2.38356247e-01 | 1.76541031e-01 |
| -9.53457944e-02 | 4.65417306e-01 | 3.78040265e-01 | 1.66543053e-01 |
| 4.78027791e-01 | 2.75997465e-01 | 2.78595978e-01 | 5.14138024e-01 |
| -1.98211339e-01 | 3.80066837e-01 | 2.35511936e-01 | 9.83869638e-02 |
| 3.81961855e-01 | 1.08280008e-01 | 2.79336675e-01 | 3.73527241e-01 |
| 1.53940240e-01 | 7.90312300e-02 | 7.67428838e-02 | 2.60284263e-01 |
| 2.00090227e-01 | 3.95519597e-01 | 3.95519597e-01 | 1.24224909e-01 |
| 2.08878949e-01 | 2.58089658e-01 | 1.89669130e-01 | 3.38679268e-01 |
| 1.55680811e-02 | 2.74035592e-01 | 4.21610067e-01 | 4.56572810e-01 |
| -2.23490195e-02 | 3.32779247e-01 | 3.18731649e-01 | 1.44522364e-01 |
| 3.77060804e-01 | 7.19877149e-02 | 3.41721867e-01 | 5.18518078e-01 |
| 1.30980554e-01 | 8.33274325e-02 | 1.47729147e-01 | 5.43906805e-01 |
| 6.13930812e-01 | 1.79491182e-01 | 2.59448554e-02 | 1.98271103e-01 |
| 2.21358522e-01 | 8.85685114e-03 | 3.27407001e-01 | 5.82468387e-01 |
| 5.10342329e-01 | 8.91564344e-02 | -6.84843832e-03 | 1.43144187e-01 |
| 7.62098177e-02 | 1.49460256e-01 | 1.57362309e-01 | 3.94516946e-01 |
| 1.42155635e-01 | 2.53674615e-01 | 2.81649770e-01 | 7.72724015e-01 |
| 2.75800835e-02 | 3.37554610e-01 | 1.58854338e-01 | 1.05686307e-01 |
| 9.27482251e-02 | 2.34523906e-01 | 1.51512685e-01 | 8.85151430e-02 |
| 1.23072932e-01 | 2.68217448e-01 | 1.01925446e-01 | 2.33696224e-01 |
| 4.59319154e-01 | 2.17403972e-01 | 2.69748659e-01 | 9.79843598e-02 |
| 1.08457646e-01 | -9.77998645e-02 | -1.85409593e-02 | 1.39851005e-01 |
| 1.50975949e-01 | 5.96645359e-02 | 7.92781527e-03 | 4.10293929e-01 |
| 2.62545847e-01 | 3.67515967e-01 | 6.61350964e-02 | -3.72348282e-02 |
| -1.08615096e-01 | 2.84541340e-01 | 1.39458120e-01 | 4.42668777e-01 |
| 1.84730082e-01 | 1.69314467e-01 | 1.12559699e-01 | 2.76982882e-01 |
| 1.09714657e-01 | 1.46213155e-01 | 4.28012542e-01 | 2.24586677e-01 |
| 2.96227229e-01 | 5.18295149e-01 | 4.59048525e-01 | -1.62172237e-01 |
| 2.27666645e-01 | 2.59460366e-01 | 4.08311115e-02 | 1.96755367e-02 |
| 5.06437137e-01 | 2.45888186e-01 | 1.80714989e-01 | 2.30287139e-01 |
| 2.17553866e-01 | 1.63879911e-01 | -1.82861735e-02 | 4.83634122e-02 |
| 4.79099809e-01 | 3.97331717e-01 | 7.63409147e-01 | 1.15091182e-01 |
| 1.02977440e-01 | 7.34698462e-01 | 2.31582623e-01 | 1.26459808e-01 |
| 1.86908124e-01 | 2.71190156e-02 | 6.06638129e-02 | 7.87648017e-02 |
| 6.82253933e-01 | 2.44352879e-01 | 6.30570010e-01 | 3.34085877e-01 |
| 3.98190892e-01 | 3.61777217e-01 | 1.94829761e-01 | 1.27982385e-01 |
| -6.44670056e-02 | -6.62242028e-02 | 2.21336849e-01 | 1.65992966e-01 |
| 2.09928963e-01 | 2.95498699e-01 | 3.46621585e-01 | 1.59586013e-01 |
| 3.17171958e-01 | 3.15774831e-01 | 1.34856831e-01 | -4.11965133e-02 |
| 1.12748403e-01 | 2.69698188e-01 | 1.31027591e-01 | 4.54569550e-02 |
| 1.21759269e-01 | 9.61088867e-02 | 1.20619010e-01 | 2.58449820e-01 |
| 3.82992780e-01 | 1.23490454e-01 | 8.08726608e-02 | 4.00957894e-01 |
| 1.13799837e-01 | 1.58123675e-01 | 3.04071731e-01 | 2.99168224e-01 |
| 1.73316153e-01 | -7.14766273e-02 | 3.47042479e-01 | 5.84499878e-01 |

```
 1.19398494e-01  1.22972319e-01  1.81335946e-01  1.37674483e-01
 1.90156353e-02  4.04837137e-01  3.08339581e-01  4.13355271e-01
 1.52868647e-03  4.88328845e-02  1.56505554e-01  6.82398590e-02
 7.51944994e-01 -4.42739712e-03  9.79762123e-02  1.49189570e-01
 1.06506207e-04  3.27753404e-01  5.07844449e-01 -1.24701108e-01
 3.29192512e-01  6.25562988e-01  3.50681219e-01  4.66790098e-01
 1.11361026e-01 -4.33807920e-02  1.35654087e-01  8.27316090e-02
 1.30028203e-02 -6.36889674e-02  1.62328260e-01  4.71133429e-01
 3.22550306e-01  2.77616190e-01  3.04481731e-01  5.82324385e-02
 8.82231910e-02  3.67406436e-01  9.36657455e-02  8.93683539e-02
 6.57555762e-02  5.15410756e-01  4.04940942e-01  5.95676618e-01
 4.88861856e-01  1.01469979e-01  1.27347059e-01  1.52944620e-01
-4.71428307e-02  2.22814251e-01  4.34176867e-01  4.52972436e-01
 2.58171381e-01  5.11240945e-01  4.07840506e-01  9.71739254e-02
 4.29852552e-01  2.21178030e-01  3.56036793e-01  3.46287006e-01
 2.91509989e-01  1.82517994e-01  2.52111856e-01  1.61768060e-01
 3.44528544e-01  4.39528496e-01  2.62621519e-01  8.68520319e-02
 4.13692326e-01  2.00390087e-01  3.09007367e-02  1.56253231e-01
 2.77931888e-01  1.72405819e-01  2.20765242e-01  6.91004879e-02
 1.98149655e-01  4.50203232e-01  2.69280258e-01  4.26693000e-01
 2.38422038e-01  6.41706604e-02 -4.95129018e-02  1.88629729e-01]
```

[30]:
```
# Now we evaluate our model
import sklearn.metrics as metrics
print("For Testing dataset:")
print("RMSE:",np.sqrt(metrics.mean_squared_error(y_test,y_pred)))
print('R SCORE:',metrics.r2_score(y_test,y_pred))
```

```
For Testing dataset:
RMSE: 0.4368626501051918
R SCORE: 0.051834588033688456
```

[31]:
```
print("For Training dataset:")
print("RMSE:",np.sqrt(metrics.mean_squared_error(y_train,y_train_pred)))
print('R SCORE:',metrics.r2_score(y_train,y_train_pred))
```

```
For Training dataset:
RMSE: 0.3731035920024518
R SCORE: 0.189975119031903
```

[32]:
```
# Model Bias is =0.4368626501051918- 0.3731035920024518 = 0.0637
#  Model Variance is = 0.189975119031903-0.051834588033688456  = 0.13807
# 10% of STD = 0.1*26848.49 = 2684.85
# This Shows that our model has performed very well due to it's low bias and␣
 ↪low varience.
```

### 0.5 .2 GRIDSEARCHCV FOR LINEAR REGRESSION

[33]:
```
from sklearn.model_selection import GridSearchCV
```

[34]:
```
from sklearn.preprocessing import StandardScaler
st_x= StandardScaler()
x = st_x.fit_transform(x)
```

[35]:
```
parameters = {'fit_intercept':[True,False],'positive':[True,False]}
gscv = LinearRegression()
grid = GridSearchCV(gscv, parameters, cv=5)
grid.fit(x,y)
print(grid.best_params_)
print(grid.best_estimator_)
print(grid.best_score_)
```

```
{'fit_intercept': True, 'positive': True}
LinearRegression(positive=True)
-0.463653852128387
```

#### 0.5.1 3. BUILDING A MACHINE LEARNING MODEL FOR PREDICTING CANCER IN PATIENTS DATA SET

[36]:
```
# Importing The Cancer Dataset
d2=pd.read_csv('cancer-data-2.csv')
print(d2)
```

```
     diagnosis  radius_mean  texture_mean  perimeter_mean  area_mean  \
0            1        17.99         10.38          122.80     1001.0
1            1        20.57         17.77          132.90     1326.0
2            1        19.69         21.25          130.00     1203.0
3            1        11.42         20.38           77.58      386.1
4            1        20.29         14.34          135.10     1297.0
..         …           …            …              …          …
564          1        21.56         22.39          142.00     1479.0
565          1        20.13         28.25          131.20     1261.0
566          1        16.60         28.08          108.30      858.1
567          1        20.60         29.33          140.10     1265.0
568          0         7.76         24.54           47.92      181.0

     smoothness_mean  compactness_mean  concavity_mean  concave points_mean  \
0            0.11840           0.27760         0.30010              0.14710
1            0.08474           0.07864         0.08690              0.07017
2            0.10960           0.15990         0.19740              0.12790
3            0.14250           0.28390         0.24140              0.10520
4            0.10030           0.13280         0.19800              0.10430
..               …                 …               …                    …
564          0.11100           0.11590         0.24390              0.13890
```

```
565          0.09780           0.10340        0.14400               0.09791
566          0.08455           0.10230        0.09251               0.05302
567          0.11780           0.27700        0.35140               0.15200
568          0.05263           0.04362        0.00000               0.00000

     symmetry_mean  …  radius_worst  texture_worst  perimeter_worst  \
0           0.2419  …        25.380          17.33           184.60
1           0.1812  …        24.990          23.41           158.80
2           0.2069  …        23.570          25.53           152.50
3           0.2597  …        14.910          26.50            98.87
4           0.1809  …        22.540          16.67           152.20
..             …  …           …              …                …
564         0.1726  …        25.450          26.40           166.10
565         0.1752  …        23.690          38.25           155.00
566         0.1590  …        18.980          34.12           126.70
567         0.2397  …        25.740          39.42           184.60
568         0.1587  …         9.456          30.37            59.16

     area_worst  smoothness_worst  compactness_worst  concavity_worst  \
0        2019.0           0.16220            0.66560           0.7119
1        1956.0           0.12380            0.18660           0.2416
2        1709.0           0.14440            0.42450           0.4504
3         567.7           0.20980            0.86630           0.6869
4        1575.0           0.13740            0.20500           0.4000
..          …               …                  …                …
564      2027.0           0.14100            0.21130           0.4107
565      1731.0           0.11660            0.19220           0.3215
566      1124.0           0.11390            0.30940           0.3403
567      1821.0           0.16500            0.86810           0.9387
568       268.6           0.08996            0.06444           0.0000

     concave points_worst  symmetry_worst  fractional_dimension_worst
0                  0.2654          0.4601                  0.11890
1                  0.1860          0.2750                  0.08902
2                  0.2430          0.3613                  0.08758
3                  0.2575          0.6638                  0.17300
4                  0.1625          0.2364                  0.07678
..                    …               …                      …
564                0.2216          0.2060                  0.07115
565                0.1628          0.2572                  0.06637
566                0.1418          0.2218                  0.07820
567                0.2650          0.4087                  0.12400
568                0.0000          0.2871                  0.07039

[569 rows x 31 columns]
```

[37]:
```
d2.describe()
```

[37]:
```
        diagnosis  radius_mean  texture_mean  perimeter_mean    area_mean  \
count  569.000000   569.000000    569.000000      569.000000   569.000000
mean     0.372583    14.127292     19.289649       91.969033   654.889104
std      0.483918     3.524049      4.301036       24.298981   351.914129
min      0.000000     6.981000      9.710000       43.790000   143.500000
25%      0.000000    11.700000     16.170000       75.170000   420.300000
50%      0.000000    13.370000     18.840000       86.240000   551.100000
75%      1.000000    15.780000     21.800000      104.100000   782.700000
max      1.000000    28.110000     39.280000      188.500000  2501.000000

       smoothness_mean  compactness_mean  concavity_mean  concave points_mean  \
count       569.000000        569.000000      569.000000           569.000000
mean          0.096360          0.104341        0.088799             0.048919
std           0.014064          0.052813        0.079720             0.038803
min           0.052630          0.019380        0.000000             0.000000
25%           0.086370          0.064920        0.029560             0.020310
50%           0.095870          0.092630        0.061540             0.033500
75%           0.105300          0.130400        0.130700             0.074000
max           0.163400          0.345400        0.426800             0.201200

       symmetry_mean  …  radius_worst  texture_worst  perimeter_worst  \
count     569.000000  …    569.000000     569.000000       569.000000
mean        0.181162  …     16.269190      25.677223       107.261213
std         0.027414  …      4.833242       6.146258        33.602542
min         0.106000  …      7.930000      12.020000        50.410000
25%         0.161900  …     13.010000      21.080000        84.110000
50%         0.179200  …     14.970000      25.410000        97.660000
75%         0.195700  …     18.790000      29.720000       125.400000
max         0.304000  …     36.040000      49.540000       251.200000

        area_worst  smoothness_worst  compactness_worst  concavity_worst  \
count   569.000000        569.000000         569.000000       569.000000
mean    880.583128          0.132369           0.254265         0.272188
std     569.356993          0.022832           0.157336         0.208624
min     185.200000          0.071170           0.027290         0.000000
25%     515.300000          0.116600           0.147200         0.114500
50%     686.500000          0.131300           0.211900         0.226700
75%    1084.000000          0.146000           0.339100         0.382900
max    4254.000000          0.222600           1.058000         1.252000

       concave points_worst  symmetry_worst  fractal_dimension_worst
count            569.000000      569.000000               569.000000
mean               0.114606        0.290076                 0.083946
std                0.065732        0.061867                 0.018061
min                0.000000        0.156500                 0.055040
25%                0.064930        0.250400                 0.071460
50%                0.099930        0.282200                 0.080040
```

```
75%                0.161400         0.317900                 0.092080
max                0.291000         0.663800                 0.207500

[8 rows x 31 columns]
```

[38]:

```
d2.corr()
```

[38]:

```
                         diagnosis  radius_mean  texture_mean  perimeter_mean  \
diagnosis                 1.000000     0.730029      0.415185        0.742636
radius_mean               0.730029     1.000000      0.323782        0.997855
texture_mean              0.415185     0.323782      1.000000        0.329533
perimeter_mean            0.742636     0.997855      0.329533        1.000000
area_mean                 0.708984     0.987357      0.321086        0.986507
smoothness_mean           0.358560     0.170581     -0.023389        0.207278
compactness_mean          0.596534     0.506124      0.236702        0.556936
concavity_mean            0.696360     0.676764      0.302418        0.716136
concave points_mean       0.776614     0.822529      0.293464        0.850977
symmetry_mean             0.330499     0.147741      0.071401        0.183027
fractal_dimension_mean   -0.012838    -0.311631     -0.076437       -0.261477
radius_se                 0.567134     0.679090      0.275869        0.691765
texture_se               -0.008303    -0.097317      0.386358       -0.086761
perimeter_se              0.556141     0.674172      0.281673        0.693135
area_se                   0.548236     0.735864      0.259845        0.744983
smoothness_se            -0.067016    -0.222600      0.006614       -0.202694
compactness_se            0.292999     0.206000      0.191975        0.250744
concavity_se              0.253730     0.194204      0.143293        0.228082
concave points_se         0.408042     0.376169      0.163851        0.407217
symmetry_se              -0.006522    -0.104321      0.009127       -0.081629
fractal_dimension_se      0.077972    -0.042641      0.054458       -0.005523
radius_worst              0.776454     0.969539      0.352573        0.969476
texture_worst             0.456903     0.297008      0.912045        0.303038
perimeter_worst           0.782914     0.965137      0.358040        0.970387
area_worst                0.733825     0.941082      0.343546        0.941550
smoothness_worst          0.421465     0.119616      0.077503        0.150549
compactness_worst         0.590998     0.413463      0.277830        0.455774
concavity_worst           0.659610     0.526911      0.301025        0.563879
concave points_worst      0.793566     0.744214      0.295316        0.771241
symmetry_worst            0.416294     0.163953      0.105008        0.189115
fractal_dimension_worst   0.323872     0.007066      0.119205        0.051019

                         area_mean  smoothness_mean  compactness_mean  \
diagnosis                 0.708984         0.358560          0.596534
radius_mean               0.987357         0.170581          0.506124
texture_mean              0.321086        -0.023389          0.236702
perimeter_mean            0.986507         0.207278          0.556936
area_mean                 1.000000         0.177028          0.498502
smoothness_mean           0.177028         1.000000          0.659123
```

| | | | |
|---|---|---|---|
| compactness_mean | 0.498502 | 0.659123 | 1.000000 |
| concavity_mean | 0.685983 | 0.521984 | 0.883121 |
| concave points_mean | 0.823269 | 0.553695 | 0.831135 |
| symmetry_mean | 0.151293 | 0.557775 | 0.602641 |
| fractal_dimension_mean | -0.283110 | 0.584792 | 0.565369 |
| radius_se | 0.732562 | 0.301467 | 0.497473 |
| texture_se | -0.066280 | 0.068406 | 0.046205 |
| perimeter_se | 0.726628 | 0.296092 | 0.548905 |
| area_se | 0.800086 | 0.246552 | 0.455653 |
| smoothness_se | -0.166777 | 0.332375 | 0.135299 |
| compactness_se | 0.212583 | 0.318943 | 0.738722 |
| concavity_se | 0.207660 | 0.248396 | 0.570517 |
| concave points_se | 0.372320 | 0.380676 | 0.642262 |
| symmetry_se | -0.072497 | 0.200774 | 0.229977 |
| fractal_dimension_se | -0.019887 | 0.283607 | 0.507318 |
| radius_worst | 0.962746 | 0.213120 | 0.535315 |
| texture_worst | 0.287489 | 0.036072 | 0.248133 |
| perimeter_worst | 0.959120 | 0.238853 | 0.590210 |
| area_worst | 0.959213 | 0.206718 | 0.509604 |
| smoothness_worst | 0.123523 | 0.805324 | 0.565541 |
| compactness_worst | 0.390410 | 0.472468 | 0.865809 |
| concavity_worst | 0.512606 | 0.434926 | 0.816275 |
| concave points_worst | 0.722017 | 0.503053 | 0.815573 |
| symmetry_worst | 0.143570 | 0.394309 | 0.510223 |
| fractal_dimension_worst | 0.003738 | 0.499316 | 0.687382 |

| | concavity_mean | concave points_mean | symmetry_mean \ |
|---|---|---|---|
| diagnosis | 0.696360 | 0.776614 | 0.330499 |
| radius_mean | 0.676764 | 0.822529 | 0.147741 |
| texture_mean | 0.302418 | 0.293464 | 0.071401 |
| perimeter_mean | 0.716136 | 0.850977 | 0.183027 |
| area_mean | 0.685983 | 0.823269 | 0.151293 |
| smoothness_mean | 0.521984 | 0.553695 | 0.557775 |
| compactness_mean | 0.883121 | 0.831135 | 0.602641 |
| concavity_mean | 1.000000 | 0.921391 | 0.500667 |
| concave points_mean | 0.921391 | 1.000000 | 0.462497 |
| symmetry_mean | 0.500667 | 0.462497 | 1.000000 |
| fractal_dimension_mean | 0.336783 | 0.166917 | 0.479921 |
| radius_se | 0.631925 | 0.698050 | 0.303379 |
| texture_se | 0.076218 | 0.021480 | 0.128053 |
| perimeter_se | 0.660391 | 0.710650 | 0.313893 |
| area_se | 0.617427 | 0.690299 | 0.223970 |
| smoothness_se | 0.098564 | 0.027653 | 0.187321 |
| compactness_se | 0.670279 | 0.490424 | 0.421659 |
| concavity_se | 0.691270 | 0.439167 | 0.342627 |
| concave points_se | 0.683260 | 0.615634 | 0.393298 |
| symmetry_se | 0.178009 | 0.095351 | 0.449137 |

| | | | |
|---|---|---|---|
| fractal_dimension_se | 0.449301 | 0.257584 | 0.331786 |
| radius_worst | 0.688236 | 0.830318 | 0.185728 |
| texture_worst | 0.299879 | 0.292752 | 0.090651 |
| perimeter_worst | 0.729565 | 0.855923 | 0.219169 |
| area_worst | 0.675987 | 0.809630 | 0.177193 |
| smoothness_worst | 0.448822 | 0.452753 | 0.426675 |
| compactness_worst | 0.754968 | 0.667454 | 0.473200 |
| concavity_worst | 0.884103 | 0.752399 | 0.433721 |
| concave points_worst | 0.861323 | 0.910155 | 0.430297 |
| symmetry_worst | 0.409464 | 0.375744 | 0.699826 |
| fractal_dimension_worst | 0.514930 | 0.368661 | 0.438413 |

| | … | radius_worst | texture_worst | perimeter_worst \ |
|---|---|---|---|---|
| diagnosis | … | 0.776454 | 0.456903 | 0.782914 |
| radius_mean | … | 0.969539 | 0.297008 | 0.965137 |
| texture_mean | … | 0.352573 | 0.912045 | 0.358040 |
| perimeter_mean | … | 0.969476 | 0.303038 | 0.970387 |
| area_mean | … | 0.962746 | 0.287489 | 0.959120 |
| smoothness_mean | … | 0.213120 | 0.036072 | 0.238853 |
| compactness_mean | … | 0.535315 | 0.248133 | 0.590210 |
| concavity_mean | … | 0.688236 | 0.299879 | 0.729565 |
| concave points_mean | … | 0.830318 | 0.292752 | 0.855923 |
| symmetry_mean | … | 0.185728 | 0.090651 | 0.219169 |
| fractal_dimension_mean | … | -0.253691 | -0.051269 | -0.205151 |
| radius_se | … | 0.715065 | 0.194799 | 0.719684 |
| texture_se | … | -0.111690 | 0.409003 | -0.102242 |
| perimeter_se | … | 0.697201 | 0.200371 | 0.721031 |
| area_se | … | 0.757373 | 0.196497 | 0.761213 |
| smoothness_se | … | -0.230691 | -0.074743 | -0.217304 |
| compactness_se | … | 0.204607 | 0.143003 | 0.260516 |
| concavity_se | … | 0.186904 | 0.100241 | 0.226680 |
| concave points_se | … | 0.358127 | 0.086741 | 0.394999 |
| symmetry_se | … | -0.128121 | -0.077473 | -0.103753 |
| fractal_dimension_se | … | -0.037488 | -0.003195 | -0.001000 |
| radius_worst | … | 1.000000 | 0.359921 | 0.993708 |
| texture_worst | … | 0.359921 | 1.000000 | 0.365098 |
| perimeter_worst | … | 0.993708 | 0.365098 | 1.000000 |
| area_worst | … | 0.984015 | 0.345842 | 0.977578 |
| smoothness_worst | … | 0.216574 | 0.225429 | 0.236775 |
| compactness_worst | … | 0.475820 | 0.360832 | 0.529408 |
| concavity_worst | … | 0.573975 | 0.368366 | 0.618344 |
| concave points_worst | … | 0.787424 | 0.359755 | 0.816322 |
| symmetry_worst | … | 0.243529 | 0.233027 | 0.269493 |
| fractal_dimension_worst | … | 0.093492 | 0.219122 | 0.138957 |

| | area_worst | smoothness_worst | compactness_worst \ |
|---|---|---|---|
| diagnosis | 0.733825 | 0.421465 | 0.590998 |

| | | | |
|---|---|---|---|
| radius_mean | 0.941082 | 0.119616 | 0.413463 |
| texture_mean | 0.343546 | 0.077503 | 0.277830 |
| perimeter_mean | 0.941550 | 0.150549 | 0.455774 |
| area_mean | 0.959213 | 0.123523 | 0.390410 |
| smoothness_mean | 0.206718 | 0.805324 | 0.472468 |
| compactness_mean | 0.509604 | 0.565541 | 0.865809 |
| concavity_mean | 0.675987 | 0.448822 | 0.754968 |
| concave points_mean | 0.809630 | 0.452753 | 0.667454 |
| symmetry_mean | 0.177193 | 0.426675 | 0.473200 |
| fractal_dimension_mean | -0.231854 | 0.504942 | 0.458798 |
| radius_se | 0.751548 | 0.141919 | 0.287103 |
| texture_se | -0.083195 | -0.073658 | -0.092439 |
| perimeter_se | 0.730713 | 0.130054 | 0.341919 |
| area_se | 0.811408 | 0.125389 | 0.283257 |
| smoothness_se | -0.182195 | 0.314457 | -0.055558 |
| compactness_se | 0.199371 | 0.227394 | 0.678780 |
| concavity_se | 0.188353 | 0.168481 | 0.484858 |
| concave points_se | 0.342271 | 0.215351 | 0.452888 |
| symmetry_se | -0.110343 | -0.012662 | 0.060255 |
| fractal_dimension_se | -0.022736 | 0.170568 | 0.390159 |
| radius_worst | 0.984015 | 0.216574 | 0.475820 |
| texture_worst | 0.345842 | 0.225429 | 0.360832 |
| perimeter_worst | 0.977578 | 0.236775 | 0.529408 |
| area_worst | 1.000000 | 0.209145 | 0.438296 |
| smoothness_worst | 0.209145 | 1.000000 | 0.568187 |
| compactness_worst | 0.438296 | 0.568187 | 1.000000 |
| concavity_worst | 0.543331 | 0.518523 | 0.892261 |
| concave points_worst | 0.747419 | 0.547691 | 0.801080 |
| symmetry_worst | 0.209146 | 0.493838 | 0.614441 |
| fractal_dimension_worst | 0.079647 | 0.617624 | 0.810455 |

| | concavity_worst | concave points_worst | \ |
|---|---|---|---|
| diagnosis | 0.659610 | 0.793566 | |
| radius_mean | 0.526911 | 0.744214 | |
| texture_mean | 0.301025 | 0.295316 | |
| perimeter_mean | 0.563879 | 0.771241 | |
| area_mean | 0.512606 | 0.722017 | |
| smoothness_mean | 0.434926 | 0.503053 | |
| compactness_mean | 0.816275 | 0.815573 | |
| concavity_mean | 0.884103 | 0.861323 | |
| concave points_mean | 0.752399 | 0.910155 | |
| symmetry_mean | 0.433721 | 0.430297 | |
| fractal_dimension_mean | 0.346234 | 0.175325 | |
| radius_se | 0.380585 | 0.531062 | |
| texture_se | -0.068956 | -0.119638 | |
| perimeter_se | 0.418899 | 0.554897 | |
| area_se | 0.385100 | 0.538166 | |

| | | |
|---|---|---|
| smoothness_se | -0.058298 | -0.102007 |
| compactness_se | 0.639147 | 0.483208 |
| concavity_se | 0.662564 | 0.440472 |
| concave points_se | 0.549592 | 0.602450 |
| symmetry_se | 0.037119 | -0.030413 |
| fractal_dimension_se | 0.379975 | 0.215204 |
| radius_worst | 0.573975 | 0.787424 |
| texture_worst | 0.368366 | 0.359755 |
| perimeter_worst | 0.618344 | 0.816322 |
| area_worst | 0.543331 | 0.747419 |
| smoothness_worst | 0.518523 | 0.547691 |
| compactness_worst | 0.892261 | 0.801080 |
| concavity_worst | 1.000000 | 0.855434 |
| concave points_worst | 0.855434 | 1.000000 |
| symmetry_worst | 0.532520 | 0.502528 |
| fractal_dimension_worst | 0.686511 | 0.511114 |

| | symmetry_worst | fractal_dimension_worst |
|---|---|---|
| diagnosis | 0.416294 | 0.323872 |
| radius_mean | 0.163953 | 0.007066 |
| texture_mean | 0.105008 | 0.119205 |
| perimeter_mean | 0.189115 | 0.051019 |
| area_mean | 0.143570 | 0.003738 |
| smoothness_mean | 0.394309 | 0.499316 |
| compactness_mean | 0.510223 | 0.687382 |
| concavity_mean | 0.409464 | 0.514930 |
| concave points_mean | 0.375744 | 0.368661 |
| symmetry_mean | 0.699826 | 0.438413 |
| fractal_dimension_mean | 0.334019 | 0.767297 |
| radius_se | 0.094543 | 0.049559 |
| texture_se | -0.128215 | -0.045655 |
| perimeter_se | 0.109930 | 0.085433 |
| area_se | 0.074126 | 0.017539 |
| smoothness_se | -0.107342 | 0.101480 |
| compactness_se | 0.277878 | 0.590973 |
| concavity_se | 0.197788 | 0.439329 |
| concave points_se | 0.143116 | 0.310655 |
| symmetry_se | 0.389402 | 0.078079 |
| fractal_dimension_se | 0.111094 | 0.591328 |
| radius_worst | 0.243529 | 0.093492 |
| texture_worst | 0.233027 | 0.219122 |
| perimeter_worst | 0.269493 | 0.138957 |
| area_worst | 0.209146 | 0.079647 |
| smoothness_worst | 0.493838 | 0.617624 |
| compactness_worst | 0.614441 | 0.810455 |
| concavity_worst | 0.532520 | 0.686511 |
| concave points_worst | 0.502528 | 0.511114 |

```
symmetry_worst                  1.000000                 0.537848
fractal_dimension_worst         0.537848                 1.000000

[31 rows x 31 columns]
```

[39]:

```python
# Indicate the dependent(x) and independent(y) variables
x=d2.iloc[:,1:]
y=d2.iloc[:,0]
```

[40]:

```python
# Verify x and y
print(x)
print(y)
```

```
     radius_mean  texture_mean  perimeter_mean  area_mean  smoothness_mean  \
0          17.99         10.38          122.80     1001.0          0.11840
1          20.57         17.77          132.90     1326.0          0.08474
2          19.69         21.25          130.00     1203.0          0.10960
3          11.42         20.38           77.58      386.1          0.14250
4          20.29         14.34          135.10     1297.0          0.10030
..           …            …              …          …               …
564        21.56         22.39          142.00     1479.0          0.11100
565        20.13         28.25          131.20     1261.0          0.09780
566        16.60         28.08          108.30      858.1          0.08455
567        20.60         29.33          140.10     1265.0          0.11780
568         7.76         24.54           47.92      181.0          0.05263

     compactness_mean  concavity_mean  concave points_mean  symmetry_mean  \
0             0.27760         0.30010              0.14710         0.2419
1             0.07864         0.08690              0.07017         0.1812
2             0.15990         0.19740              0.12790         0.2069
3             0.28390         0.24140              0.10520         0.2597
4             0.13280         0.19800              0.10430         0.1809
..                …               …                    …            …
564           0.11590         0.24390              0.13890         0.1726
565           0.10340         0.14400              0.09791         0.1752
566           0.10230         0.09251              0.05302         0.1590
567           0.27700         0.35140              0.15200         0.2397
568           0.04362         0.00000              0.00000         0.1587

     fractal_dimension_mean  …  radius_worst  texture_worst  \
0                   0.07871  …        25.380          17.33
1                   0.05667  …        24.990          23.41
2                   0.05999  …        23.570          25.53
3                   0.09744  …        14.910          26.50
4                   0.05883  …        22.540          16.67
..                      …  …           …             …
564                 0.05623  …        25.450          26.40
```

```
565                0.05533  ...       23.690          38.25
566                0.05648  ...       18.980          34.12
567                0.07016  ...       25.740          39.42
568                0.05884  ...        9.456          30.37

     perimeter_worst  area_worst  smoothness_worst  compactness_worst  \
0             184.60      2019.0           0.16220            0.66560
1             158.80      1956.0           0.12380            0.18660
2             152.50      1709.0           0.14440            0.42450
3              98.87       567.7           0.20980            0.86630
4             152.20      1575.0           0.13740            0.20500
..               ...         ...               ...                ...
564           166.10      2027.0           0.14100            0.21130
565           155.00      1731.0           0.11660            0.19220
566           126.70      1124.0           0.11390            0.30940
567           184.60      1821.0           0.16500            0.86810
568            59.16       268.6           0.08996            0.06444

     concavity_worst  concave points_worst  symmetry_worst  \
0             0.7119                0.2654          0.4601
1             0.2416                0.1860          0.2750
2             0.4504                0.2430          0.3613
3             0.6869                0.2575          0.6638
4             0.4000                0.1625          0.2364
..               ...                   ...             ...
564           0.4107                0.2216          0.2060
565           0.3215                0.1628          0.2572
566           0.3403                0.1418          0.2218
567           0.9387                0.2650          0.4087
568           0.0000                0.0000          0.2871

     fractal_dimension_worst
0                    0.11890
1                    0.08902
2                    0.08758
3                    0.17300
4                    0.07678
..                       ...
564                  0.07115
565                  0.06637
566                  0.07820
567                  0.12400
568                  0.07039

[569 rows x 30 columns]
0      1
1      1
2      1
```

```
3      1
4      1
      ..
564    1
565    1
566    1
567    1
568    0
Name: diagnosis, Length: 569, dtype: int64
```

[41]:
```python
#split the dataset into training and testing datasets
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.1,random_state=2)
```

[42]:
```python
# displaying the training datasets
print('X TRAIN:\n',x_train)
print('\nY TRAIN:\n',y_train)
```

```
X TRAIN:
      radius_mean  texture_mean  perimeter_mean  area_mean  smoothness_mean  \
60         10.17         14.88           64.55      311.9          0.11340
3          11.42         20.38           77.58      386.1          0.14250
426        10.48         14.98           67.49      333.6          0.09816
204        12.47         18.60           81.09      481.9          0.09965
430        14.90         22.53          102.10      685.0          0.09947
..           …           …               …         …               …
299        10.51         23.09           66.85      334.2          0.10150
534        10.96         17.62           70.79      365.6          0.09687
493        12.46         12.83           78.83      477.3          0.07372
527        12.34         12.27           78.94      468.5          0.09003
168        17.47         24.68          116.10      984.6          0.10490

     compactness_mean  concavity_mean  concave points_mean  symmetry_mean  \
60            0.08061        0.010840              0.01290         0.2743
3             0.28390        0.241400              0.10520         0.2597
426           0.10130        0.063350              0.02218         0.1925
204           0.10580        0.080050              0.03821         0.1925
430           0.22250        0.273300              0.09711         0.2041
..               …              …                    …              …
299           0.06797        0.024950              0.01875         0.1695
534           0.09752        0.052630              0.02788         0.1619
493           0.04043        0.007173              0.01149         0.1613
527           0.06307        0.029580              0.02647         0.1689
168           0.16030        0.215900              0.10430         0.1538

     fractal_dimension_mean  …  radius_worst  texture_worst  \
60                  0.06960  …         11.02          17.45
3                   0.09744  …         14.91          26.50
```

```
426                0.06915  …          12.13           21.57
204                0.06373  …          14.97           24.64
430                0.06898  …          16.35           27.57
..                    …   …            …             …
299                0.06556  …          10.93           24.22
534                0.06408  …          11.62           26.51
493                0.06013  …          13.19           16.36
527                0.05808  …          13.61           19.27
168                0.06365  …          23.14           32.33

     perimeter_worst  area_worst  smoothness_worst  compactness_worst  \
60             69.86       368.6           0.12750            0.09866
3              98.87       567.7           0.20980            0.86630
426            81.41       440.4           0.13270            0.29960
204            96.05       677.9           0.14260            0.23780
430           125.40       832.7           0.14190            0.70900
..               …           …                …                  …
299            70.10       362.7           0.11430            0.08614
534            76.43       407.5           0.14280            0.25100
493            83.24       534.0           0.09439            0.06477
527            87.22       564.9           0.12920            0.20740
168           155.30      1660.0           0.13760            0.38300

     concavity_worst  concave points_worst  symmetry_worst  \
60           0.02168               0.02579          0.3557
3            0.68690               0.25750          0.6638
426          0.29390               0.09310          0.3020
204          0.26710               0.10150          0.3014
430          0.90190               0.24750          0.2866
..               …                     …               …
299          0.04158               0.03125          0.2227
534          0.21230               0.09861          0.2289
493          0.01674               0.02680          0.2280
527          0.17910               0.10700          0.3110
168          0.48900               0.17210          0.2160

     fractal_dimension_worst
60                   0.08020
3                    0.17300
426                  0.09646
204                  0.08750
430                  0.11550
..                       …
299                  0.06777
534                  0.08278
493                  0.07028
527                  0.07592
168                  0.09300
```

```
[512 rows x 30 columns]

Y TRAIN:
 60     0
3      1
426    0
204    0
430    1
      ..
299    0
534    0
493    0
527    0
168    1
Name: diagnosis, Length: 512, dtype: int64
```

[43]:
```
#datasets shape
print('X:\n',x_train.shape)
print('Y:\n',y_train.shape)
```

```
X:
 (512, 30)
Y:
 (512,)
```

[44]:
```
# Display testing datasets
print('X:\n',x_test)
print('Y:\n',y_test)
```

```
X:
     radius_mean  texture_mean  perimeter_mean  area_mean  smoothness_mean  \
528       13.940         13.17           90.31      594.2          0.12480
291       14.960         19.10           97.03      687.3          0.08992
467        9.668         18.10           61.06      286.3          0.08311
108       22.270         19.67          152.80     1509.0          0.13260
340       14.420         16.54           94.15      641.2          0.09751
256       19.550         28.77          133.60     1207.0          0.09260
160       11.750         20.18           76.10      419.8          0.10890
306       13.200         15.82           84.07      537.3          0.08511
155       12.250         17.94           78.27      460.3          0.08654
511       14.810         14.70           94.66      680.7          0.08472
171       13.430         19.63           85.84      565.4          0.09048
109       11.340         21.26           72.48      396.5          0.08759
275       11.890         17.36           76.20      435.6          0.12250
200       12.230         19.56           78.54      461.0          0.09586
55        11.520         18.75           73.34      409.0          0.09524
161       19.190         15.94          126.30     1157.0          0.08694
```

| | | | | | |
|---|---|---|---|---|---|
| 67 | 11.310 | 19.04 | 71.80 | 394.1 | 0.08139 |
| 540 | 11.540 | 14.44 | 74.65 | 402.9 | 0.09984 |
| 281 | 11.740 | 14.02 | 74.24 | 427.3 | 0.07813 |
| 72 | 17.200 | 24.52 | 114.20 | 929.4 | 0.10710 |
| 152 | 9.731 | 15.34 | 63.78 | 300.2 | 0.10720 |
| 304 | 11.460 | 18.16 | 73.59 | 403.1 | 0.08853 |
| 246 | 13.200 | 17.43 | 84.13 | 541.6 | 0.07215 |
| 294 | 12.720 | 13.78 | 81.78 | 492.1 | 0.09667 |
| 453 | 14.530 | 13.98 | 93.86 | 644.2 | 0.10990 |
| 517 | 19.890 | 20.26 | 130.50 | 1214.0 | 0.10370 |
| 544 | 13.870 | 20.70 | 89.77 | 584.8 | 0.09578 |
| 500 | 15.040 | 16.74 | 98.73 | 689.4 | 0.09883 |
| 1 | 20.570 | 17.77 | 132.90 | 1326.0 | 0.08474 |
| 365 | 20.440 | 21.78 | 133.80 | 1293.0 | 0.09150 |
| 209 | 15.270 | 12.91 | 98.17 | 725.5 | 0.08182 |
| 84 | 12.000 | 15.65 | 76.95 | 443.3 | 0.09723 |
| 312 | 12.760 | 13.37 | 82.29 | 504.1 | 0.08794 |
| 265 | 20.730 | 31.12 | 135.70 | 1419.0 | 0.09469 |
| 129 | 19.790 | 25.12 | 130.40 | 1192.0 | 0.10150 |
| 355 | 12.560 | 19.07 | 81.92 | 485.8 | 0.08760 |
| 178 | 13.010 | 22.22 | 82.01 | 526.4 | 0.06251 |
| 40 | 13.440 | 21.58 | 86.18 | 563.0 | 0.08162 |
| 303 | 10.490 | 18.61 | 66.86 | 334.3 | 0.10680 |
| 103 | 9.876 | 19.40 | 63.95 | 298.3 | 0.10050 |
| 567 | 20.600 | 29.33 | 140.10 | 1265.0 | 0.11780 |
| 141 | 16.110 | 18.05 | 105.10 | 813.0 | 0.09721 |
| 561 | 11.200 | 29.37 | 70.67 | 386.0 | 0.07449 |
| 320 | 10.250 | 16.18 | 66.52 | 324.2 | 0.10610 |
| 512 | 13.400 | 20.52 | 88.64 | 556.7 | 0.11060 |
| 463 | 11.600 | 18.36 | 73.88 | 412.7 | 0.08508 |
| 525 | 8.571 | 13.10 | 54.53 | 221.3 | 0.10360 |
| 332 | 11.220 | 19.86 | 71.94 | 387.3 | 0.10540 |
| 368 | 21.710 | 17.25 | 140.90 | 1546.0 | 0.09384 |
| 516 | 18.310 | 20.58 | 120.80 | 1052.0 | 0.10680 |
| 371 | 15.190 | 13.21 | 97.65 | 711.8 | 0.07963 |
| 444 | 18.030 | 16.85 | 117.50 | 990.0 | 0.08947 |
| 363 | 16.500 | 18.29 | 106.60 | 838.1 | 0.09686 |
| 251 | 11.500 | 18.45 | 73.28 | 407.4 | 0.09345 |
| 222 | 10.180 | 17.53 | 65.12 | 313.1 | 0.10610 |
| 205 | 15.120 | 16.68 | 98.78 | 716.6 | 0.08876 |
| 136 | 11.710 | 16.67 | 74.72 | 423.6 | 0.10510 |

| | compactness_mean | concavity_mean | concave points_mean | symmetry_mean | \ |
|---|---|---|---|---|---|
| 528 | 0.09755 | 0.101000 | 0.066150 | 0.1976 | |
| 291 | 0.09823 | 0.059400 | 0.048190 | 0.1879 | |
| 467 | 0.05428 | 0.014790 | 0.005769 | 0.1680 | |
| 108 | 0.27680 | 0.426400 | 0.182300 | 0.2556 | |
| 340 | 0.11390 | 0.080070 | 0.042230 | 0.1912 | |

| | | | | |
|---|---|---|---|---|
| 256 | 0.20630 | 0.178400 | 0.114400 | 0.1893 |
| 160 | 0.11410 | 0.068430 | 0.037380 | 0.1993 |
| 306 | 0.05251 | 0.001461 | 0.003261 | 0.1632 |
| 155 | 0.06679 | 0.038850 | 0.023310 | 0.1970 |
| 511 | 0.05016 | 0.034160 | 0.025410 | 0.1659 |
| 171 | 0.06288 | 0.058580 | 0.034380 | 0.1598 |
| 109 | 0.06575 | 0.051330 | 0.018990 | 0.1487 |
| 275 | 0.07210 | 0.059290 | 0.074040 | 0.2015 |
| 200 | 0.08087 | 0.041870 | 0.041070 | 0.1979 |
| 55 | 0.05473 | 0.030360 | 0.022780 | 0.1920 |
| 161 | 0.11850 | 0.119300 | 0.096670 | 0.1741 |
| 67 | 0.04701 | 0.037090 | 0.022300 | 0.1516 |
| 540 | 0.11200 | 0.067370 | 0.025940 | 0.1818 |
| 281 | 0.04340 | 0.022450 | 0.027630 | 0.2101 |
| 72 | 0.18300 | 0.169200 | 0.079440 | 0.1927 |
| 152 | 0.15990 | 0.410800 | 0.078570 | 0.2548 |
| 304 | 0.07694 | 0.033440 | 0.015020 | 0.1411 |
| 246 | 0.04524 | 0.043360 | 0.011050 | 0.1487 |
| 294 | 0.08393 | 0.012880 | 0.019240 | 0.1638 |
| 453 | 0.09242 | 0.068950 | 0.064950 | 0.1650 |
| 517 | 0.13100 | 0.141100 | 0.094310 | 0.1802 |
| 544 | 0.10180 | 0.036880 | 0.023690 | 0.1620 |
| 500 | 0.13640 | 0.077210 | 0.061420 | 0.1668 |
| 1 | 0.07864 | 0.086900 | 0.070170 | 0.1812 |
| 365 | 0.11310 | 0.097990 | 0.077850 | 0.1618 |
| 209 | 0.06230 | 0.058920 | 0.031570 | 0.1359 |
| 84 | 0.07165 | 0.041510 | 0.018630 | 0.2079 |
| 312 | 0.07948 | 0.040520 | 0.025480 | 0.1601 |
| 265 | 0.11430 | 0.136700 | 0.086460 | 0.1769 |
| 129 | 0.15890 | 0.254500 | 0.114900 | 0.2202 |
| 355 | 0.10380 | 0.103000 | 0.043910 | 0.1533 |
| 178 | 0.01938 | 0.001595 | 0.001852 | 0.1395 |
| 40 | 0.06031 | 0.031100 | 0.020310 | 0.1784 |
| 303 | 0.06678 | 0.022970 | 0.017800 | 0.1482 |
| 103 | 0.09697 | 0.061540 | 0.030290 | 0.1945 |
| 567 | 0.27700 | 0.351400 | 0.152000 | 0.2397 |
| 141 | 0.11370 | 0.094470 | 0.059430 | 0.1861 |
| 561 | 0.03558 | 0.000000 | 0.000000 | 0.1060 |
| 320 | 0.11110 | 0.067260 | 0.039650 | 0.1743 |
| 512 | 0.14690 | 0.144500 | 0.081720 | 0.2116 |
| 463 | 0.05855 | 0.033670 | 0.017770 | 0.1516 |
| 525 | 0.07632 | 0.025650 | 0.015100 | 0.1678 |
| 332 | 0.06779 | 0.005006 | 0.007583 | 0.1940 |
| 368 | 0.08562 | 0.116800 | 0.084650 | 0.1717 |
| 516 | 0.12480 | 0.156900 | 0.094510 | 0.1860 |
| 371 | 0.06934 | 0.033930 | 0.026570 | 0.1721 |
| 444 | 0.12320 | 0.109000 | 0.062540 | 0.1720 |
| 363 | 0.08468 | 0.058620 | 0.048350 | 0.1495 |

| | | | | |
|---|---|---|---|---|
| 251 | 0.05991 | 0.026380 | 0.020690 | 0.1834 |
| 222 | 0.08502 | 0.017680 | 0.019150 | 0.1910 |
| 205 | 0.09588 | 0.075500 | 0.040790 | 0.1594 |
| 136 | 0.06095 | 0.035920 | 0.026000 | 0.1339 |

| | fractal_dimension_mean | … | radius_worst | texture_worst | \ |
|---|---|---|---|---|---|
| 528 | 0.06457 | … | 14.620 | 15.38 | |
| 291 | 0.05852 | … | 16.250 | 26.19 | |
| 467 | 0.06412 | … | 11.150 | 24.62 | |
| 108 | 0.07039 | … | 28.400 | 28.01 | |
| 340 | 0.06412 | … | 16.670 | 21.51 | |
| 256 | 0.06232 | … | 25.050 | 36.27 | |
| 160 | 0.06453 | … | 13.320 | 26.21 | |
| 306 | 0.05894 | … | 14.410 | 20.45 | |
| 155 | 0.06228 | … | 13.590 | 25.22 | |
| 511 | 0.05348 | … | 15.610 | 17.58 | |
| 171 | 0.05671 | … | 17.980 | 29.87 | |
| 109 | 0.06529 | … | 13.010 | 29.15 | |
| 275 | 0.05875 | … | 12.400 | 18.99 | |
| 200 | 0.06013 | … | 14.440 | 28.36 | |
| 55 | 0.05907 | … | 12.840 | 22.47 | |
| 161 | 0.05176 | … | 22.030 | 17.81 | |
| 67 | 0.05667 | … | 12.330 | 23.84 | |
| 540 | 0.06782 | … | 12.260 | 19.68 | |
| 281 | 0.06113 | … | 13.310 | 18.26 | |
| 72 | 0.06487 | … | 23.320 | 33.82 | |
| 152 | 0.09296 | … | 11.020 | 19.49 | |
| 304 | 0.06243 | … | 12.680 | 21.61 | |
| 246 | 0.05635 | … | 13.940 | 27.82 | |
| 294 | 0.06100 | … | 13.500 | 17.48 | |
| 453 | 0.06121 | … | 15.800 | 16.93 | |
| 517 | 0.06188 | … | 23.730 | 25.23 | |
| 544 | 0.06688 | … | 15.050 | 24.75 | |
| 500 | 0.06869 | … | 16.760 | 20.43 | |
| 1 | 0.05667 | … | 24.990 | 23.41 | |
| 365 | 0.05557 | … | 24.310 | 26.37 | |
| 209 | 0.05526 | … | 17.380 | 15.92 | |
| 84 | 0.05968 | … | 13.670 | 24.90 | |
| 312 | 0.06140 | … | 14.190 | 16.40 | |
| 265 | 0.05674 | … | 32.490 | 47.16 | |
| 129 | 0.06113 | … | 22.630 | 33.58 | |
| 355 | 0.06184 | … | 13.370 | 22.43 | |
| 178 | 0.05234 | … | 14.000 | 29.02 | |
| 40 | 0.05587 | … | 15.930 | 30.25 | |
| 303 | 0.06600 | … | 11.060 | 24.54 | |
| 103 | 0.06322 | … | 10.760 | 26.83 | |
| 567 | 0.07016 | … | 25.740 | 39.42 | |
| 141 | 0.06248 | … | 19.920 | 25.27 | |

| | | | | |
|---|---|---|---|---|
| 561 | 0.05502 | … | 11.920 | 38.30 |
| 320 | 0.07279 | … | 11.280 | 20.61 |
| 512 | 0.07325 | … | 16.410 | 29.66 |
| 463 | 0.05859 | … | 12.770 | 24.02 |
| 525 | 0.07126 | … | 9.473 | 18.45 |
| 332 | 0.06028 | … | 11.980 | 25.78 |
| 368 | 0.05054 | … | 30.750 | 26.44 |
| 516 | 0.05941 | … | 21.860 | 26.20 |
| 371 | 0.05544 | … | 16.200 | 15.73 |
| 444 | 0.05780 | … | 20.380 | 22.02 |
| 363 | 0.05593 | … | 18.130 | 25.45 |
| 251 | 0.05934 | … | 12.970 | 22.46 |
| 222 | 0.06908 | … | 11.170 | 22.84 |
| 205 | 0.05986 | … | 17.770 | 20.24 |
| 136 | 0.05945 | … | 13.330 | 25.48 |

| | perimeter_worst | area_worst | smoothness_worst | compactness_worst \ |
|---|---|---|---|---|
| 528 | 94.52 | 653.3 | 0.13940 | 0.13640 |
| 291 | 109.10 | 809.8 | 0.13130 | 0.30300 |
| 467 | 71.11 | 380.2 | 0.13880 | 0.12550 |
| 108 | 206.80 | 2360.0 | 0.17010 | 0.69970 |
| 340 | 111.40 | 862.1 | 0.12940 | 0.33710 |
| 256 | 178.60 | 1926.0 | 0.12810 | 0.53290 |
| 160 | 88.91 | 543.9 | 0.13580 | 0.18920 |
| 306 | 92.00 | 636.9 | 0.11280 | 0.13460 |
| 155 | 86.60 | 564.2 | 0.12170 | 0.17880 |
| 511 | 101.70 | 760.2 | 0.11390 | 0.10110 |
| 171 | 116.60 | 993.6 | 0.14010 | 0.15460 |
| 109 | 83.99 | 518.1 | 0.16990 | 0.21960 |
| 275 | 79.46 | 472.4 | 0.13590 | 0.08368 |
| 200 | 92.15 | 638.4 | 0.14290 | 0.20420 |
| 55 | 81.81 | 506.2 | 0.12490 | 0.08720 |
| 161 | 146.60 | 1495.0 | 0.11240 | 0.20160 |
| 67 | 78.00 | 466.7 | 0.12900 | 0.09148 |
| 540 | 78.78 | 457.8 | 0.13450 | 0.21180 |
| 281 | 84.70 | 533.7 | 0.10360 | 0.08500 |
| 72 | 151.60 | 1681.0 | 0.15850 | 0.73940 |
| 152 | 71.04 | 380.5 | 0.12920 | 0.27720 |
| 304 | 82.69 | 489.8 | 0.11440 | 0.17890 |
| 246 | 88.28 | 602.0 | 0.11010 | 0.15080 |
| 294 | 88.54 | 553.7 | 0.12980 | 0.14720 |
| 453 | 103.10 | 749.9 | 0.13470 | 0.14780 |
| 517 | 160.50 | 1646.0 | 0.14170 | 0.33090 |
| 544 | 99.17 | 688.6 | 0.12640 | 0.20370 |
| 500 | 109.70 | 856.9 | 0.11350 | 0.21760 |
| 1 | 158.80 | 1956.0 | 0.12380 | 0.18660 |
| 365 | 161.20 | 1780.0 | 0.13270 | 0.23760 |
| 209 | 113.70 | 932.7 | 0.12220 | 0.21860 |

| | | | | |
|---|---|---|---|---|
| 84 | 87.78 | 567.9 | 0.13770 | 0.20030 |
| 312 | 92.04 | 618.8 | 0.11940 | 0.22080 |
| 265 | 214.00 | 3432.0 | 0.14010 | 0.26440 |
| 129 | 148.70 | 1589.0 | 0.12750 | 0.38610 |
| 355 | 89.02 | 547.4 | 0.10960 | 0.20020 |
| 178 | 88.18 | 608.8 | 0.08125 | 0.03432 |
| 40 | 102.50 | 787.9 | 0.10940 | 0.20430 |
| 303 | 70.76 | 375.4 | 0.14130 | 0.10440 |
| 103 | 72.22 | 361.2 | 0.15590 | 0.23020 |
| 567 | 184.60 | 1821.0 | 0.16500 | 0.86810 |
| 141 | 129.00 | 1233.0 | 0.13140 | 0.22360 |
| 561 | 75.19 | 439.6 | 0.09267 | 0.05494 |
| 320 | 71.53 | 390.4 | 0.14020 | 0.23600 |
| 512 | 113.30 | 844.4 | 0.15740 | 0.38560 |
| 463 | 82.68 | 495.1 | 0.13420 | 0.18080 |
| 525 | 63.30 | 275.6 | 0.16410 | 0.22350 |
| 332 | 76.91 | 436.1 | 0.14240 | 0.09669 |
| 368 | 199.50 | 3143.0 | 0.13630 | 0.16280 |
| 516 | 142.20 | 1493.0 | 0.14920 | 0.25360 |
| 371 | 104.50 | 819.1 | 0.11260 | 0.17370 |
| 444 | 133.30 | 1292.0 | 0.12630 | 0.26660 |
| 363 | 117.20 | 1009.0 | 0.13380 | 0.16790 |
| 251 | 83.12 | 508.9 | 0.11830 | 0.10490 |
| 222 | 71.94 | 375.6 | 0.14060 | 0.14400 |
| 205 | 117.70 | 989.5 | 0.14910 | 0.33310 |
| 136 | 86.16 | 546.7 | 0.12710 | 0.10280 |

| | concavity_worst | concave points_worst | symmetry_worst | \ |
|---|---|---|---|---|
| 528 | 0.155900 | 0.101500 | 0.2160 | |
| 291 | 0.180400 | 0.148900 | 0.2962 | |
| 467 | 0.064090 | 0.025000 | 0.3057 | |
| 108 | 0.960800 | 0.291000 | 0.4055 | |
| 340 | 0.375500 | 0.141400 | 0.3053 | |
| 256 | 0.425100 | 0.194100 | 0.2818 | |
| 160 | 0.195600 | 0.079090 | 0.3168 | |
| 306 | 0.011200 | 0.025000 | 0.2651 | |
| 155 | 0.194300 | 0.082110 | 0.3113 | |
| 511 | 0.110100 | 0.079550 | 0.2334 | |
| 171 | 0.264400 | 0.116000 | 0.2884 | |
| 109 | 0.312000 | 0.082780 | 0.2829 | |
| 275 | 0.071530 | 0.089460 | 0.2220 | |
| 200 | 0.137700 | 0.108000 | 0.2668 | |
| 55 | 0.090760 | 0.063160 | 0.3306 | |
| 161 | 0.226400 | 0.177700 | 0.2443 | |
| 67 | 0.144400 | 0.069610 | 0.2400 | |
| 540 | 0.179700 | 0.069180 | 0.2329 | |
| 281 | 0.067350 | 0.082900 | 0.3101 | |
| 72 | 0.656600 | 0.189900 | 0.3313 | |

| | | | |
|---|---|---|---|
| 152 | 0.821600 | 0.157100 | 0.3108 |
| 304 | 0.122600 | 0.055090 | 0.2208 |
| 246 | 0.229800 | 0.049700 | 0.2767 |
| 294 | 0.052330 | 0.063430 | 0.2369 |
| 453 | 0.137300 | 0.106900 | 0.2606 |
| 517 | 0.418500 | 0.161300 | 0.2549 |
| 544 | 0.137700 | 0.068450 | 0.2249 |
| 500 | 0.185600 | 0.101800 | 0.2177 |
| 1 | 0.241600 | 0.186000 | 0.2750 |
| 365 | 0.270200 | 0.176500 | 0.2609 |
| 209 | 0.296200 | 0.103500 | 0.2320 |
| 84 | 0.226700 | 0.076320 | 0.3379 |
| 312 | 0.176900 | 0.084110 | 0.2564 |
| 265 | 0.344200 | 0.165900 | 0.2868 |
| 129 | 0.567300 | 0.173200 | 0.3305 |
| 355 | 0.238800 | 0.092650 | 0.2121 |
| 178 | 0.007977 | 0.009259 | 0.2295 |
| 40 | 0.208500 | 0.111200 | 0.2994 |
| 303 | 0.084230 | 0.065280 | 0.2213 |
| 103 | 0.264400 | 0.097490 | 0.2622 |
| 567 | 0.938700 | 0.265000 | 0.4087 |
| 141 | 0.280200 | 0.121600 | 0.2792 |
| 561 | 0.000000 | 0.000000 | 0.1566 |
| 320 | 0.189800 | 0.097440 | 0.2608 |
| 512 | 0.510600 | 0.205100 | 0.3585 |
| 463 | 0.186000 | 0.082880 | 0.3210 |
| 525 | 0.175400 | 0.085120 | 0.2983 |
| 332 | 0.013350 | 0.020220 | 0.3292 |
| 368 | 0.286100 | 0.182000 | 0.2510 |
| 516 | 0.375900 | 0.151000 | 0.3074 |
| 371 | 0.136200 | 0.081780 | 0.2487 |
| 444 | 0.429000 | 0.153500 | 0.2842 |
| 363 | 0.166300 | 0.091230 | 0.2394 |
| 251 | 0.081050 | 0.065440 | 0.2740 |
| 222 | 0.065720 | 0.055750 | 0.3055 |
| 205 | 0.332700 | 0.125200 | 0.3415 |
| 136 | 0.104600 | 0.069680 | 0.1712 |

| | fractal_dimension_worst |
|---|---|
| 528 | 0.07253 |
| 291 | 0.08472 |
| 467 | 0.07875 |
| 108 | 0.09789 |
| 340 | 0.08764 |
| 256 | 0.10050 |
| 160 | 0.07987 |
| 306 | 0.08385 |
| 155 | 0.08132 |

| | |
|---|---|
| 511 | 0.06142 |
| 171 | 0.07371 |
| 109 | 0.08832 |
| 275 | 0.06033 |
| 200 | 0.08174 |
| 55 | 0.07036 |
| 161 | 0.06251 |
| 67 | 0.06641 |
| 540 | 0.08134 |
| 281 | 0.06688 |
| 72 | 0.13390 |
| 152 | 0.12590 |
| 304 | 0.07638 |
| 246 | 0.07198 |
| 294 | 0.06922 |
| 453 | 0.07810 |
| 517 | 0.09136 |
| 544 | 0.08492 |
| 500 | 0.08549 |
| 1 | 0.08902 |
| 365 | 0.06735 |
| 209 | 0.07474 |
| 84 | 0.07924 |
| 312 | 0.08253 |
| 265 | 0.08218 |
| 129 | 0.08465 |
| 355 | 0.07188 |
| 178 | 0.05843 |
| 40 | 0.07146 |
| 303 | 0.07842 |
| 103 | 0.08490 |
| 567 | 0.12400 |
| 141 | 0.08158 |
| 561 | 0.05905 |
| 320 | 0.09702 |
| 512 | 0.11090 |
| 463 | 0.07863 |
| 525 | 0.10490 |
| 332 | 0.06522 |
| 368 | 0.06494 |
| 516 | 0.07863 |
| 371 | 0.06766 |
| 444 | 0.08225 |
| 363 | 0.06469 |
| 251 | 0.06487 |
| 222 | 0.08797 |
| 205 | 0.09740 |
| 136 | 0.07343 |

```
[57 rows x 30 columns]
Y:
 528    0
291    0
467    0
108    1
340    0
256    1
160    0
306    0
155    0
511    0
171    1
109    0
275    0
200    0
55     0
161    1
67     0
540    0
281    0
72     1
152    0
304    0
246    0
294    0
453    0
517    1
544    0
500    0
1      1
365    1
209    0
84     0
312    0
265    1
129    1
355    0
178    0
40     1
303    0
103    0
567    1
141    1
561    0
320    0
512    1
```

```
463    0
525    0
332    0
368    1
516    1
371    0
444    1
363    0
251    0
222    0
205    1
136    0
Name: diagnosis, dtype: int64
```

[45]:
```
#scaing our data
from sklearn.preprocessing import StandardScaler
scaler=StandardScaler()  # Object of StandardScaler Class
x_train_scaled=scaler.fit_transform(x_train)
x_test_scaled=scaler.transform(x_test)
```

[46]:
```
# Now we train our model to learn the datasets
from sklearn.linear_model import LogisticRegression
lgr=LogisticRegression()
lgr.fit(x_train_scaled,y_train) # Model is trained with the 512 training records
```

[46]:
```
LogisticRegression()
```

[47]:
```
# Then we test our model
y_pred=lgr.predict(x_test_scaled)
print( y_pred)
```

```
[0 0 0 1 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 1 0 0 0 0 0 1 0 0 1 1 0 0 0 1 1 0 0
 0 0 0 1 1 0 0 1 0 0 0 1 1 0 1 1 0 0 1 0]
```

[48]:
```
# We compare the Actual value to the predicted values
dfm=pd.DataFrame({'Actual':y_test,'Predicted':y_pred, 'Difference':
 ↪y_test-y_pred})
print(dfm)
```

```
     Actual  Predicted  Difference
528       0          0           0
291       0          0           0
467       0          0           0
108       1          1           0
340       0          0           0
256       1          1           0
160       0          0           0
306       0          0           0
```

| | | | |
|---|---|---|---|
| 155 | 0 | 0 | 0 |
| 511 | 0 | 0 | 0 |
| 171 | 1 | 1 | 0 |
| 109 | 0 | 0 | 0 |
| 275 | 0 | 0 | 0 |
| 200 | 0 | 0 | 0 |
| 55 | 0 | 0 | 0 |
| 161 | 1 | 1 | 0 |
| 67 | 0 | 0 | 0 |
| 540 | 0 | 0 | 0 |
| 281 | 0 | 0 | 0 |
| 72 | 1 | 1 | 0 |
| 152 | 0 | 0 | 0 |
| 304 | 0 | 0 | 0 |
| 246 | 0 | 0 | 0 |
| 294 | 0 | 0 | 0 |
| 453 | 0 | 0 | 0 |
| 517 | 1 | 1 | 0 |
| 544 | 0 | 0 | 0 |
| 500 | 0 | 0 | 0 |
| 1 | 1 | 1 | 0 |
| 365 | 1 | 1 | 0 |
| 209 | 0 | 0 | 0 |
| 84 | 0 | 0 | 0 |
| 312 | 0 | 0 | 0 |
| 265 | 1 | 1 | 0 |
| 129 | 1 | 1 | 0 |
| 355 | 0 | 0 | 0 |
| 178 | 0 | 0 | 0 |
| 40 | 1 | 0 | 1 |
| 303 | 0 | 0 | 0 |
| 103 | 0 | 0 | 0 |
| 567 | 1 | 1 | 0 |
| 141 | 1 | 1 | 0 |
| 561 | 0 | 0 | 0 |
| 320 | 0 | 0 | 0 |
| 512 | 1 | 1 | 0 |
| 463 | 0 | 0 | 0 |
| 525 | 0 | 0 | 0 |
| 332 | 0 | 0 | 0 |
| 368 | 1 | 1 | 0 |
| 516 | 1 | 1 | 0 |
| 371 | 0 | 0 | 0 |
| 444 | 1 | 1 | 0 |
| 363 | 0 | 1 | −1 |
| 251 | 0 | 0 | 0 |
| 222 | 0 | 0 | 0 |
| 205 | 1 | 1 | 0 |

```
136         0           0            0
```

[49]:
```
# Observating the differences where the data is not zero, the prediction was␣
↪WRONG
# Therefore our prediction looks near accurate
```

[50]:
```
# our regression equation we looks like
print(lgr.intercept_)  # Gives the value of 'b'
print(lgr.coef_)  # Gives the 30 coefficients of the regression equation
```

```
[-0.15791469]
[[ 0.51563131  0.35322049  0.48347474  0.56198034  0.2312156  -0.59345537
   0.79362221  1.04598705 -0.15160525 -0.28608271  1.22193988 -0.05599549
   0.68455271  0.94413944  0.21225854 -0.7550556  -0.09215395  0.40112978
  -0.18033158 -0.67414476  0.98243049  1.2028827   0.83067229  0.97306258
   0.86102185  0.00832564  0.81321907  0.79232381  0.81335579  0.53795661]]
```

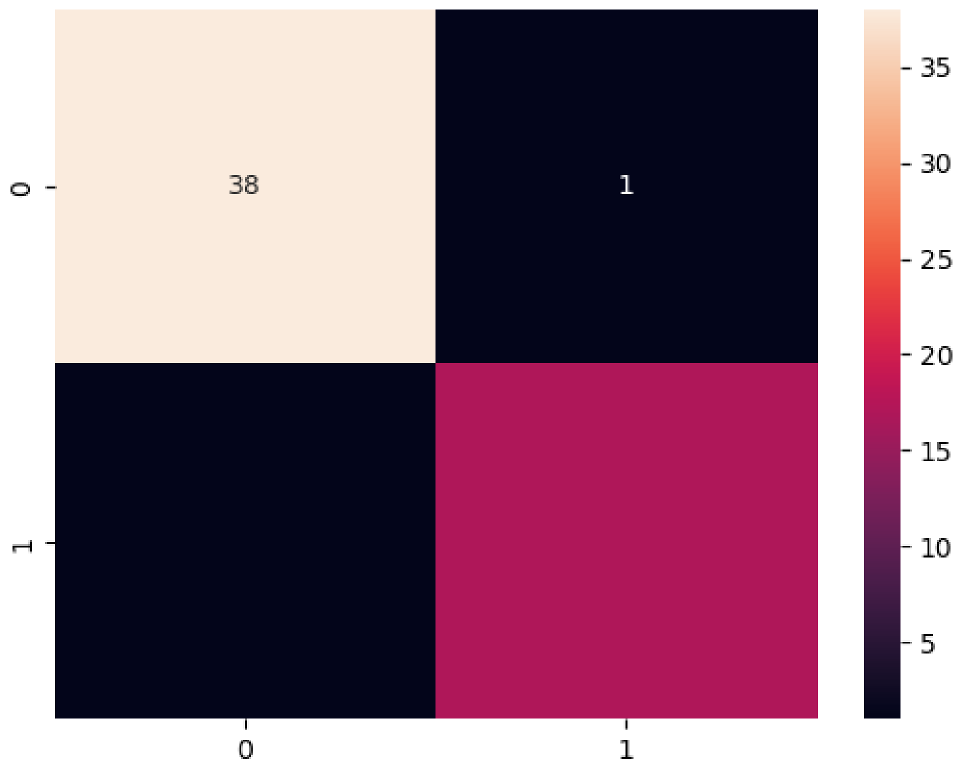
[51]:
```
# Next we evaluate our model
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix
cf=confusion_matrix(y_test,y_pred)
print(cf)
print("Classification Report For Testing Dataset:")
print(classification_report(y_test,y_pred))
```

```
[[38  1]
 [ 1 17]]
Classification Report For Testing Dataset:
              precision    recall  f1-score   support

           0       0.97      0.97      0.97        39
           1       0.94      0.94      0.94        18

    accuracy                           0.96        57
   macro avg       0.96      0.96      0.96        57
weighted avg       0.96      0.96      0.96        57
```

[52]:
```
# the Model Accuracyis = 96%
# our model has learn the dataset (precision, recall and f1 score are constant␣
↪for both +ve and -ve predictions)
```

[53]:
```
# Heatmap
cf=confusion_matrix(y_test,y_pred)
sns.heatmap(cf,annot=True)
```

[53]: <Axes: >




### 0.6 4. USING THE K-NN ON CLASSIFICATION MODEL

[54]:
```
# Scaling The Dataset
from sklearn.preprocessing import StandardScaler
st_x= StandardScaler()
x= st_x.fit_transform(x)
```

[55]:
```
#importing liabery
from sklearn.neighbors import KNeighborsRegressor
```

[56]:
```
#Tuning The Parameters
parameters = {'n_neighbors': range(30),
              'metric':['manhattan','euclidean']}
c = KNeighborsRegressor()
grid = GridSearchCV(c, parameters, cv=5)
grid.fit(x,y)
print("Best Parameters:",grid.best_params_)
print("Best Estimators:",grid.best_estimator_)
print("Best Score:",grid.best_score_)
```

```
Best Parameters: {'metric': 'manhattan', 'n_neighbors': 2}
Best Estimators: KNeighborsRegressor(metric='manhattan', n_neighbors=2)
```

```
Best Score: 0.8688563481494486

C:\Users\PC\anaconda3\Lib\site-
packages\sklearn\model_selection\_validation.py:378: FitFailedWarning:
10 fits failed out of a total of 300.
The score on these train-test partitions for these parameters will be set to
nan.
If these failures are not expected, you can try to debug them by setting
error_score='raise'.

Below are more details about the failures:
--------------------------------------------------------------------------------
10 fits failed with the following error:
Traceback (most recent call last):
  File "C:\Users\PC\anaconda3\Lib\site-
packages\sklearn\model_selection\_validation.py", line 686, in _fit_and_score
    estimator.fit(X_train, y_train, **fit_params)
  File "C:\Users\PC\anaconda3\Lib\site-
packages\sklearn\neighbors\_regression.py", line 215, in fit
    self._validate_params()
  File "C:\Users\PC\anaconda3\Lib\site-packages\sklearn\base.py", line 600, in
_validate_params
    validate_parameter_constraints(
  File "C:\Users\PC\anaconda3\Lib\site-
packages\sklearn\utils\_param_validation.py", line 97, in
validate_parameter_constraints
    raise InvalidParameterError(
sklearn.utils._param_validation.InvalidParameterError: The 'n_neighbors'
parameter of KNeighborsRegressor must be an int in the range [1, inf) or None.
Got 0 instead.

  warnings.warn(some_fits_failed_message, FitFailedWarning)
C:\Users\PC\anaconda3\Lib\site-packages\sklearn\model_selection\_search.py:952:
UserWarning: One or more of the test scores are non-finite: [       nan
0.77474909 0.86885635 0.86803264 0.8579427  0.8601354
 0.86278342 0.86005462 0.85770716 0.85638616 0.85895758 0.85783803
 0.85087744 0.84624967 0.84450975 0.84081186 0.84011566 0.84033493
 0.83749609 0.83508343 0.83336695 0.83088478 0.82958427 0.82858995
 0.82593815 0.82496799 0.82279226 0.82224485 0.82259019 0.82008483
        nan 0.80155327 0.82798541 0.85156749 0.85649723 0.8598864
 0.85453138 0.85033685 0.85028413 0.84813779 0.84795082 0.8446083
 0.84186243 0.84597571 0.84221027 0.83891439 0.83474065 0.8337275
 0.83258391 0.82925887 0.82954448 0.82847209 0.82732878 0.82646872
 0.82522312 0.82591248 0.82418473 0.82014577 0.82088392 0.81866076]
  warnings.warn(
```

[ ]:
```
# the Best Estimators was accurate in this case too.
```