15rdrdcrs

July 28, 2023

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

[2]:
```
df=pd.read_csv("5_Instagram data.csv")
df
```

[2]:
```
     Impressions  From Home  From Hashtags  From Explore  From Other  Saves  \
0           3920       2586           1028           619          56     98
1           5394       2727           1838          1174          78    194
2           4021       2085           1188             0         533     41
3           4528       2700            621           932          73    172
4           2518       1704            255           279          37     96
..             …          …              …             …           …      …
114        13700       5185           3041          5352          77    573
115         5731       1923           1368          2266          65    135
116         4139       1133           1538          1367          33     36
117        32695      11815           3147         17414         170   1095
118        36919      13473           4176         16444        2547    653

     Comments  Shares  Likes  Profile Visits  Follows  \
0           9       5    162              35        2
1           7      14    224              48       10
2          11       1    131              62       12
3          10       7    213              23        8
4           5       4    123               8        0
..          …       …      …               …        …
114         2      38    373              73       80
115         4       1    148              20       18
116         0       1     92              34       10
117         2      75    549             148      214
118         5      26    443             611      228

                                               Caption  \
0    Here are some of the most important data visua…
1    Here are some of the best data science project…
```

```
2    Learn how to train a machine learning model an…
3    Here s how you can write a Python program to d…
4    Plotting annotations while visualizing your da…
..                                                …
114  Here are some of the best data science certifi…
115  Clustering is a machine learning technique use…
116  Clustering music genres is a task of grouping …
117  Here are some of the best data science certifi…
118  175 Python Projects with Source Code solved an…

                                              Hashtags
0    #finance #money #business #investing #investme…
1    #healthcare #health #covid #data #datascience …
2    #data #datascience #dataanalysis #dataanalytic…
3    #python #pythonprogramming #pythonprojects #py…
4    #datavisualization #datascience #data #dataana…
..                                                …
114  #datascience #datasciencejobs #datasciencetrai…
115  #machinelearning #machinelearningalgorithms #d…
116  #machinelearning #machinelearningalgorithms #d…
117  #datascience #datasciencejobs #datasciencetrai…
118  #python #pythonprogramming #pythonprojects #py…

[119 rows x 13 columns]
```

[3]:
```
df.head()
```

[3]:
```
   Impressions  From Home  From Hashtags  From Explore  From Other  Saves  \
0         3920       2586           1028           619          56     98
1         5394       2727           1838          1174          78    194
2         4021       2085           1188             0         533     41
3         4528       2700            621           932          73    172
4         2518       1704            255           279          37     96

   Comments  Shares  Likes  Profile Visits  Follows  \
0         9       5    162              35        2
1         7      14    224              48       10
2        11       1    131              62       12
3        10       7    213              23        8
4         5       4    123               8        0

                                             Caption  \
0  Here are some of the most important data visua…
1  Here are some of the best data science project…
2  Learn how to train a machine learning model an…
3  Here s how you can write a Python program to d…
4  Plotting annotations while visualizing your da…
```

```
                                            Hashtags
0  #finance #money #business #investing #investme…
1  #healthcare #health #covid #data #datascience …
2  #data #datascience #dataanalysis #dataanalytic…
3  #python #pythonprogramming #pythonprojects #py…
4  #datavisualization #datascience #data #dataana…
```

1 DATA CLEANING AND DATA PREPROCESSING

[4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 119 entries, 0 to 118
Data columns (total 13 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   Impressions     119 non-null    int64
 1   From Home       119 non-null    int64
 2   From Hashtags   119 non-null    int64
 3   From Explore    119 non-null    int64
 4   From Other      119 non-null    int64
 5   Saves           119 non-null    int64
 6   Comments        119 non-null    int64
 7   Shares          119 non-null    int64
 8   Likes           119 non-null    int64
 9   Profile Visits  119 non-null    int64
 10  Follows         119 non-null    int64
 11  Caption         119 non-null    object
 12  Hashtags        119 non-null    object
dtypes: int64(11), object(2)
memory usage: 12.2+ KB
```

[5]:
```
df.describe()
```

[5]:
```
        Impressions     From Home  From Hashtags  From Explore   From Other  \
count    119.000000    119.000000     119.000000    119.000000   119.000000
mean    5703.991597   2475.789916    1887.512605   1078.100840   171.092437
std     4843.780105   1489.386348    1884.361443   2613.026132   289.431031
min     1941.000000   1133.000000     116.000000      0.000000     9.000000
25%     3467.000000   1945.000000     726.000000    157.500000    38.000000
50%     4289.000000   2207.000000    1278.000000    326.000000    74.000000
75%     6138.000000   2602.500000    2363.500000    689.500000   196.000000
max    36919.000000  13473.000000   11817.000000  17414.000000  2547.000000

             Saves    Comments      Shares        Likes  Profile Visits  \
count    119.000000  119.000000  119.000000  119.000000      119.000000
mean     153.310924    6.663866    9.361345  173.781513       50.621849
std      156.317731    3.544576   10.089205   82.378947       87.088402
min       22.000000    0.000000    0.000000   72.000000        4.000000
25%       65.000000    4.000000    3.000000  121.500000       15.000000
50%      109.000000    6.000000    6.000000  151.000000       23.000000
75%      169.000000    8.000000   13.500000  204.000000       42.000000
max     1095.000000   19.000000   75.000000  549.000000      611.000000

          Follows
count  119.000000
mean    20.756303
std     40.921580
min      0.000000
25%      4.000000
50%      8.000000
75%     18.000000
max    260.000000
```

[6]:
```
df.columns
```

[6]:
```
Index(['Impressions', 'From Home', 'From Hashtags', 'From Explore',
       'From Other', 'Saves', 'Comments', 'Shares', 'Likes', 'Profile Visits',
       'Follows', 'Caption', 'Hashtags'],
      dtype='object')
```

[7]:
```
df1=df.dropna(axis=1)
df1
```

[7]:
```
     Impressions  From Home  From Hashtags  From Explore  From Other  Saves  \
0           3920       2586           1028           619          56     98
1           5394       2727           1838          1174          78    194
2           4021       2085           1188             0         533     41
3           4528       2700            621           932          73    172
4           2518       1704            255           279          37     96
..            …          …              …             …          …     …
114        13700       5185           3041          5352          77    573
115         5731       1923           1368          2266          65    135
116         4139       1133           1538          1367          33     36
117        32695      11815           3147         17414         170   1095
118        36919      13473           4176         16444        2547    653

     Comments  Shares  Likes  Profile Visits  Follows  \
0           9       5    162              35        2
1           7      14    224              48       10
2          11       1    131              62       12
3          10       7    213              23        8
```

```
4           5       4    123               8        0
..        …     …    …             …      …
114         2      38    373              73       80
115         4       1    148              20       18
116         0       1     92              34       10
117         2      75    549             148      214
118         5      26    443             611      228

                                               Caption  \
0    Here are some of the most important data visua…
1    Here are some of the best data science project…
2    Learn how to train a machine learning model an…
3    Here s how you can write a Python program to d…
4    Plotting annotations while visualizing your da…
..                                                 …
114  Here are some of the best data science certifi…
115  Clustering is a machine learning technique use…
116  Clustering music genres is a task of grouping …
117  Here are some of the best data science certifi…
118  175 Python Projects with Source Code solved an…

                                              Hashtags
0    #finance #money #business #investing #investme…
1    #healthcare #health #covid #data #datascience …
2    #data #datascience #dataanalysis #dataanalytic…
3    #python #pythonprogramming #pythonprojects #py…
4    #datavisualization #datascience #data #dataana…
..                                                 …
114  #datascience #datasciencejobs #datasciencetrai…
115  #machinelearning #machinelearningalgorithms #d…
116  #machinelearning #machinelearningalgorithms #d…
117  #datascience #datasciencejobs #datasciencetrai…
118  #python #pythonprogramming #pythonprojects #py…

[119 rows x 13 columns]
```

[8]:
```
df1.columns
```

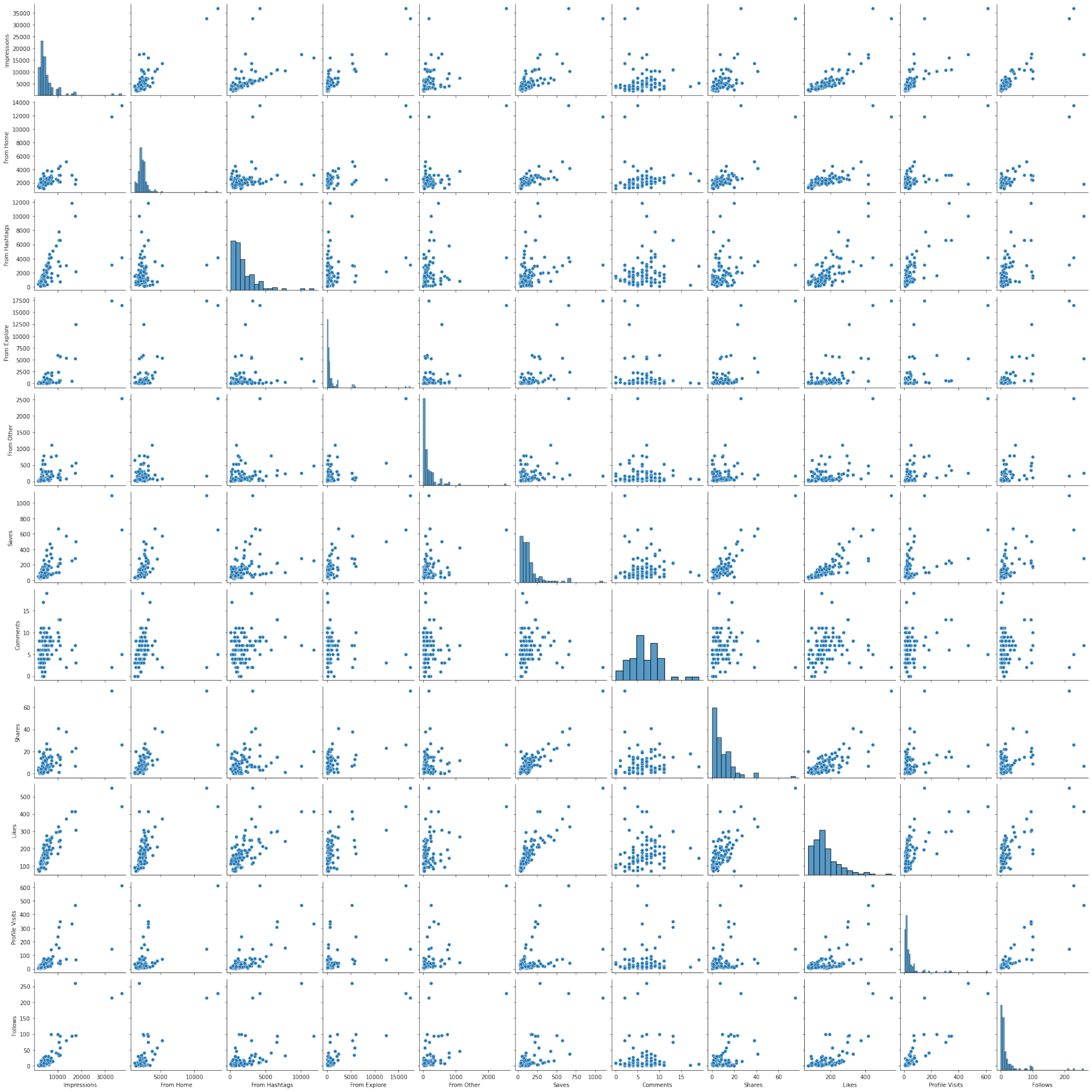
[8]:
```
Index(['Impressions', 'From Home', 'From Hashtags', 'From Explore',
       'From Other', 'Saves', 'Comments', 'Shares', 'Likes', 'Profile Visits',
       'Follows', 'Caption', 'Hashtags'],
      dtype='object')
```

[9]:
```
df1=df1[['Impressions', 'From Home', 'From Hashtags', 'From Explore',
       'From Other', 'Saves', 'Comments', 'Shares', 'Likes', 'Profile Visits',
       'Follows']]
```

2 EDA AND VISUALIZATION

[10]:
```
sns.pairplot(df1)
```

[10]:
```
<seaborn.axisgrid.PairGrid at 0x208b87275b0>
```

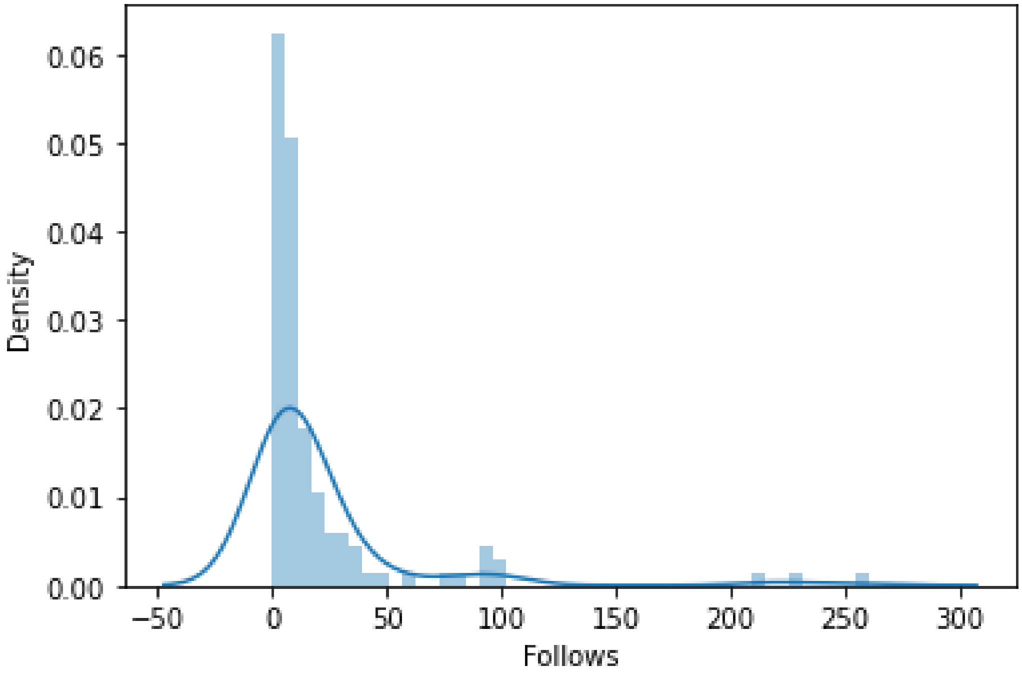


[11]:
```
sns.distplot(df1['Follows'])
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2557:
FutureWarning: `distplot` is a deprecated function and will be removed in a
future version. Please adapt your code to use either `displot` (a figure-level
function with similar flexibility) or `histplot` (an axes-level function for
histograms).
```

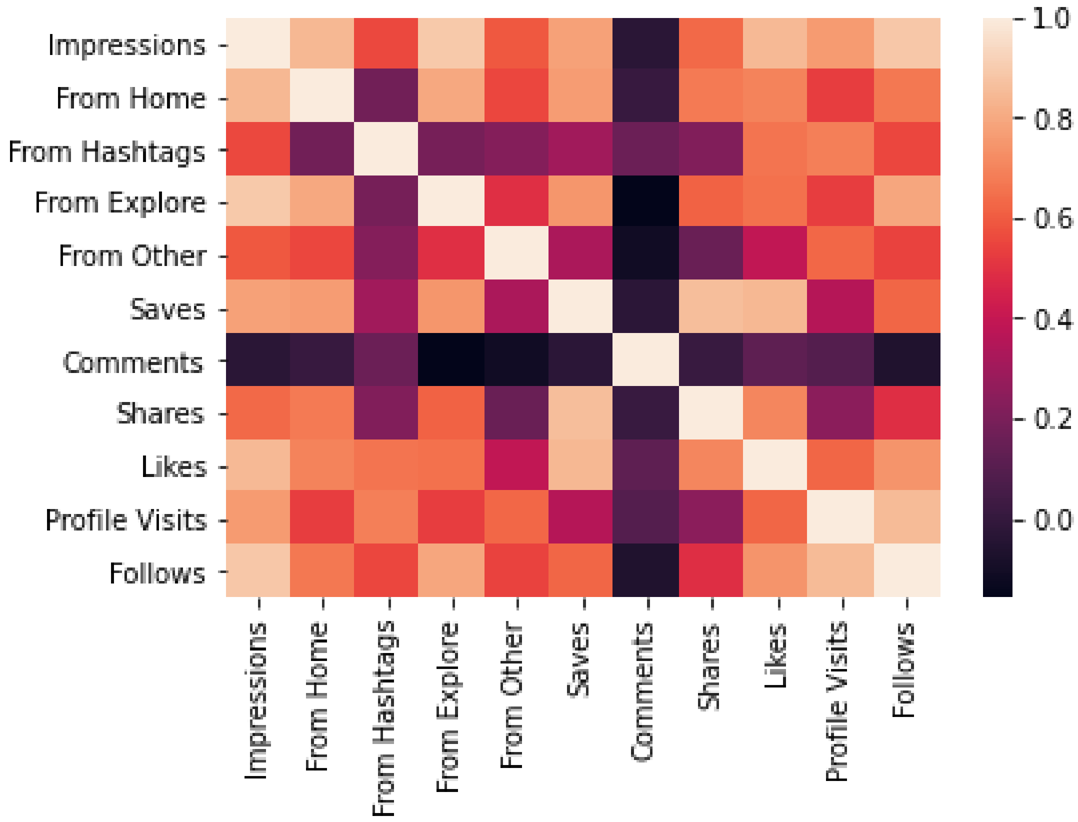
```
    warnings.warn(msg, FutureWarning)
```

[11]:
```
<AxesSubplot:xlabel='Follows', ylabel='Density'>
```



[12]:
```
sns.heatmap(df1.corr())
```

[12]:
```
<AxesSubplot:>
```



3 TO TRAIN THE MODEL AND MODEL BULDING

[13]:
```
x=df[['Impressions', 'From Home', 'From Hashtags', 'From Explore',
       'From Other', 'Saves', 'Comments', 'Shares', 'Likes', 'Profile Visits']]
y=df['Follows']
```

[14]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3)
```

[15]:
```
from sklearn.linear_model import LinearRegression
lr=LinearRegression()
lr.fit(x_train,y_train)
```

[15]:
```
LinearRegression()
```

[16]:
```
lr.intercept_
```

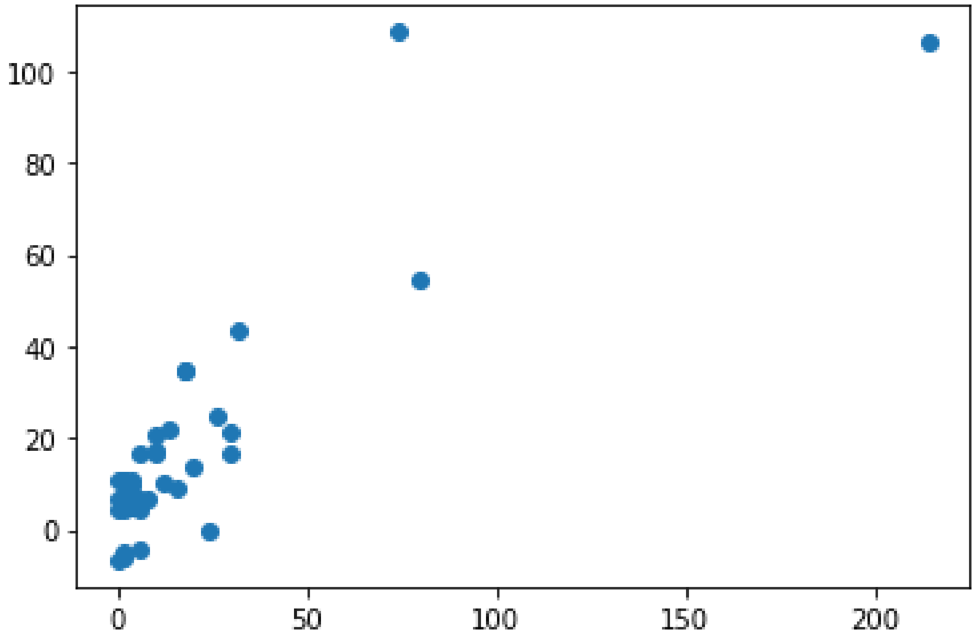
[16]:
```
5.867015580134375
```

[17]:
```
coeff=pd.DataFrame(lr.coef_,x.columns,columns=['Co-efficient'])
coeff
```

[17]:
```
                Co-efficient
Impressions         0.007473
From Home          -0.016841
From Hashtags      -0.012272
From Explore       -0.003465
From Other         -0.016010
Saves               0.058606
Comments           -0.535775
Shares             -0.464667
Likes               0.122242
Profile Visits      0.413406
```

[18]:
```
prediction =lr.predict(x_test)
plt.scatter(y_test,prediction)
```

[18]:
```
<matplotlib.collections.PathCollection at 0x208befe3a00>
```



4 ACCURACY

[19]:
```
lr.score(x_test,y_test)
```

[19]:
```
0.6781144571217874
```

[20]:

```
lr.score(x_train,y_train)
```

[20]:

```
0.938302169652006
```

[21]:

```
from sklearn.linear_model import Ridge,Lasso
rr=Ridge(alpha=10)
rr.fit(x_train,y_train)
```

[21]:

```
Ridge(alpha=10)
```

[22]:

```
rr.score(x_train,y_train)
```

[22]:

```
0.9383018791055572
```

[23]:

```
rr.score(x_test,y_test)
```

[23]:

```
0.6780827579277924
```

[24]:

```
la=Lasso(alpha=10)
la.fit(x_train,y_train)
```

```
C:\ProgramData\Anaconda3\lib\site-
packages\sklearn\linear_model\_coordinate_descent.py:530: ConvergenceWarning:
Objective did not converge. You might want to increase the number of iterations.
Duality gap: 4020.5990347859397, tolerance: 14.70488674698795
  model = cd_fast.enet_coordinate_descent(
```

[24]:

```
Lasso(alpha=10)
```

[25]:

```
la.score(x_test,y_test)
```

[25]:

```
0.6999650982152639
```

[26]:

```
la.score(x_train,y_train)
```

[26]:

```
0.9333092713010613
```