App score prediction

In this project, we are going to predict the score of an app on the playstore. We have got the data from Google playstore. We are going to use linear regression,decision trees and techniques like that to create a Machine learning algorithm that can predict the score of an app on the playstore. The algorithm will look at variables like number of downloads, reviews etc and calculate a score of the app. We are also going to analyse the data and answer some questions like:

1.Is the rating dependent on the pricing of the app?

2.Do more number of reviews guarantee a higher rating?

3.What is the usual price of apps?

Thus there are many questions like that will be answered further in this analysis.

Step 1: Setting up the work environment

We are going to download the necessary packages for our work. We are going to view the dataset and check the datatypes.

In [1]:
```
pip install notebook-as-pdf
```

```
Requirement already satisfied: notebook-as-pdf in c:\users\sujoydutta\anaconda3\lib\site-packages (0.5.0)
Requirement already satisfied: PyPDF2 in c:\users\sujoydutta\anaconda3\lib\site-packages (from notebook-as-pdf)
(2.11.2)
Requirement already satisfied: nbconvert in c:\users\sujoydutta\anaconda3\lib\site-packages (from notebook-as-p
df) (6.4.4)
Requirement already satisfied: pyppeteer in c:\users\sujoydutta\anaconda3\lib\site-packages (from notebook-as-p
df) (1.0.2)
Requirement already satisfied: bleach in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbconvert->noteb
ook-as-pdf) (4.1.0)
Requirement already satisfied: jupyterlab-pygments in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbc
onvert->notebook-as-pdf) (0.1.2)
Requirement already satisfied: pandocfilters>=1.4.1 in c:\users\sujoydutta\anaconda3\lib\site-packages (from nb
convert->notebook-as-pdf) (1.5.0)
Requirement already satisfied: mistune<2,>=0.8.1 in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbcon
vert->notebook-as-pdf) (0.8.4)
Requirement already satisfied: defusedxml in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbconvert->n
otebook-as-pdf) (0.7.1)
Requirement already satisfied: entrypoints>=0.2.2 in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbco
nvert->notebook-as-pdf) (0.4)
Requirement already satisfied: jinja2>=2.4 in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbconvert->
notebook-as-pdf) (3.0.3)
Requirement already satisfied: jupyter-core in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbconvert-
>notebook-as-pdf) (4.9.2)
Requirement already satisfied: testpath in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbconvert->not
ebook-as-pdf) (0.5.0)
Requirement already satisfied: nbclient<0.6.0,>=0.5.0 in c:\users\sujoydutta\anaconda3\lib\site-packages (from
nbconvert->notebook-as-pdf) (0.5.13)
Requirement already satisfied: traitlets>=5.0 in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbconver
t->notebook-as-pdf) (5.1.1)
Requirement already satisfied: pygments>=2.4.1 in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbconve
rt->notebook-as-pdf) (2.11.2)
Requirement already satisfied: beautifulsoup4 in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbconver
t->notebook-as-pdf) (4.11.1)
Requirement already satisfied: nbformat>=4.4 in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbconvert
->notebook-as-pdf) (5.3.0)
Requirement already satisfied: MarkupSafe>=2.0 in c:\users\sujoydutta\anaconda3\lib\site-packages (from jinja2>
=2.4->nbconvert->notebook-as-pdf) (2.0.1)
Requirement already satisfied: nest-asyncio in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbclient<
0.6.0,>=0.5.0->nbconvert->notebook-as-pdf) (1.5.5)
Requirement already satisfied: jupyter-client>=6.1.5 in c:\users\sujoydutta\anaconda3\lib\site-packages (from n
bclient<0.6.0,>=0.5.0->nbconvert->notebook-as-pdf) (6.1.12)
Requirement already satisfied: python-dateutil>=2.1 in c:\users\sujoydutta\anaconda3\lib\site-packages (from ju
pyter-client>=6.1.5->nbclient<0.6.0,>=0.5.0->nbconvert->notebook-as-pdf) (2.8.2)
Requirement already satisfied: pyzmq>=13 in c:\users\sujoydutta\anaconda3\lib\site-packages (from jupyter-clien
t>=6.1.5->nbclient<0.6.0,>=0.5.0->nbconvert->notebook-as-pdf) (22.3.0)
Requirement already satisfied: tornado>=4.1 in c:\users\sujoydutta\anaconda3\lib\site-packages (from jupyter-cl
ient>=6.1.5->nbclient<0.6.0,>=0.5.0->nbconvert->notebook-as-pdf) (6.1)
Requirement already satisfied: pywin32>=1.0 in c:\users\sujoydutta\anaconda3\lib\site-packages (from jupyter-co
re->nbconvert->notebook-as-pdf) (302)
Requirement already satisfied: jsonschema>=2.6 in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbforma
t>=4.4->nbconvert->notebook-as-pdf) (4.4.0)
Requirement already satisfied: fastjsonschema in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbformat
>=4.4->nbconvert->notebook-as-pdf) (2.15.1)
Requirement already satisfied: attrs>=17.4.0 in c:\users\sujoydutta\anaconda3\lib\site-packages (from jsonschem
a>=2.6->nbformat>=4.4->nbconvert->notebook-as-pdf) (21.4.0)
Requirement already satisfied: pyrsistent!=0.17.0,!=0.17.1,!=0.17.2,>=0.14.0 in c:\users\sujoydutta\anaconda3\l
ib\site-packages (from jsonschema>=2.6->nbformat>=4.4->nbconvert->notebook-as-pdf) (0.18.0)
Requirement already satisfied: six>=1.5 in c:\users\sujoydutta\anaconda3\lib\site-packages (from python-dateuti
l>=2.1->jupyter-client>=6.1.5->nbclient<0.6.0,>=0.5.0->nbconvert->notebook-as-pdf) (1.16.0)
Requirement already satisfied: soupsieve>1.2 in c:\users\sujoydutta\anaconda3\lib\site-packages (from beautiful
soup4->nbconvert->notebook-as-pdf) (2.3.1)
Requirement already satisfied: packaging in c:\users\sujoydutta\anaconda3\lib\site-packages (from bleach->nbcon
vert->notebook-as-pdf) (21.3)
Requirement already satisfied: webencodings in c:\users\sujoydutta\anaconda3\lib\site-packages (from bleach->nb
convert->notebook-as-pdf) (0.5.1)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in c:\users\sujoydutta\anaconda3\lib\site-packages (fro
m packaging->bleach->nbconvert->notebook-as-pdf) (3.0.4)
Requirement already satisfied: typing_extensions>=3.10.0.0 in c:\users\sujoydutta\anaconda3\lib\site-packages
(from PyPDF2->notebook-as-pdf) (4.1.1)
Requirement already satisfied: importlib-metadata>=1.4 in c:\users\sujoydutta\anaconda3\lib\site-packages (from
pyppeteer->notebook-as-pdf) (4.11.3)
Requirement already satisfied: pyee<9.0.0,>=8.1.0 in c:\users\sujoydutta\anaconda3\lib\site-packages (from pypp
eteer->notebook-as-pdf) (8.2.2)
Requirement already satisfied: websockets<11.0,>=10.0 in c:\users\sujoydutta\anaconda3\lib\site-packages (from
pyppeteer->notebook-as-pdf) (10.4)
Requirement already satisfied: appdirs<2.0.0,>=1.4.3 in c:\users\sujoydutta\anaconda3\lib\site-packages (from p
yppeteer->notebook-as-pdf) (1.4.4)
Requirement already satisfied: certifi>=2021 in c:\users\sujoydutta\anaconda3\lib\site-packages (from pyppeteer
->notebook-as-pdf) (2021.10.8)
Requirement already satisfied: tqdm<5.0.0,>=4.42.1 in c:\users\sujoydutta\anaconda3\lib\site-packages (from pyp
peteer->notebook-as-pdf) (4.64.0)
Requirement already satisfied: urllib3<2.0.0,>=1.25.8 in c:\users\sujoydutta\anaconda3\lib\site-packages (from
pyppeteer->notebook-as-pdf) (1.26.9)
Requirement already satisfied: zipp>=0.5 in c:\users\sujoydutta\anaconda3\lib\site-packages (from importlib-met
adata>=1.4->pyppeteer->notebook-as-pdf) (3.7.0)
Requirement already satisfied: colorama in c:\users\sujoydutta\anaconda3\lib\site-packages (from tqdm<5.0.0,>=
4.42.1->pyppeteer->notebook-as-pdf) (0.4.4)
Note: you may need to restart the kernel to use updated packages.
```

In [2]:
```
pip install jinja2==3.0.3
```

```
Requirement already satisfied: jinja2==3.0.3 in c:\users\sujoydutta\anaconda3\lib\site-packages (3.0.3)
Requirement already satisfied: MarkupSafe>=2.0 in c:\users\sujoydutta\anaconda3\lib\site-packages (from jinja2=
=3.0.3) (2.0.1)
Note: you may need to restart the kernel to use updated packages.
```

In [3]:
```
pip install nbconvert
```

```
Requirement already satisfied: nbconvert in c:\users\sujoydutta\anaconda3\lib\site-packages (6.4.4)
Requirement already satisfied: jinja2>=2.4 in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbconvert)
(3.0.3)
Requirement already satisfied: beautifulsoup4 in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbconver
t) (4.11.1)
Requirement already satisfied: entrypoints>=0.2.2 in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbco
nvert) (0.4)
Requirement already satisfied: mistune<2,>=0.8.1 in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbcon
vert) (0.8.4)
Requirement already satisfied: jupyterlab-pygments in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbc
onvert) (0.1.2)
Requirement already satisfied: defusedxml in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbconvert)
(0.7.1)
Requirement already satisfied: testpath in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbconvert) (0.
5.0)
Requirement already satisfied: nbformat>=4.4 in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbconver
t) (5.3.0)
Requirement already satisfied: pandocfilters>=1.4.1 in c:\users\sujoydutta\anaconda3\lib\site-packages (from nb
convert) (1.5.0)
Requirement already satisfied: nbclient<0.6.0,>=0.5.0 in c:\users\sujoydutta\anaconda3\lib\site-packages (from
nbconvert) (0.5.13)
Requirement already satisfied: traitlets>=5.0 in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbconver
t) (5.1.1)
Requirement already satisfied: pygments>=2.4.1 in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbconve
rt) (2.11.2)
Requirement already satisfied: bleach in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbconvert) (4.1.
0)
Requirement already satisfied: jupyter-core in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbconvert)
(4.9.2)
Requirement already satisfied: MarkupSafe>=2.0 in c:\users\sujoydutta\anaconda3\lib\site-packages (from jinja2>
=2.4->nbconvert) (2.0.1)
Requirement already satisfied: jupyter-client>=6.1.5 in c:\users\sujoydutta\anaconda3\lib\site-packages (from n
bclient<0.6.0,>=0.5.0->nbconvert) (6.1.12)
Requirement already satisfied: nest-asyncio in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbclient<
0.6.0,>=0.5.0->nbconvert) (1.5.5)
Requirement already satisfied: tornado>=4.1 in c:\users\sujoydutta\anaconda3\lib\site-packages (from jupyter-cl
ient>=6.1.5->nbclient<0.6.0,>=0.5.0->nbconvert) (6.1)
Requirement already satisfied: pyzmq>=13 in c:\users\sujoydutta\anaconda3\lib\site-packages (from jupyter-clien
t>=6.1.5->nbclient<0.6.0,>=0.5.0->nbconvert) (22.3.0)
Requirement already satisfied: python-dateutil>=2.1 in c:\users\sujoydutta\anaconda3\lib\site-packages (from ju
pyter-client>=6.1.5->nbclient<0.6.0,>=0.5.0->nbconvert) (2.8.2)
Requirement already satisfied: pywin32>=1.0 in c:\users\sujoydutta\anaconda3\lib\site-packages (from jupyter-co
re->nbconvert) (302)
Requirement already satisfied: jsonschema>=2.6 in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbforma
t>=4.4->nbconvert) (4.4.0)
Requirement already satisfied: fastjsonschema in c:\users\sujoydutta\anaconda3\lib\site-packages (from nbformat
>=4.4->nbconvert) (2.15.1)
Requirement already satisfied: attrs>=17.4.0 in c:\users\sujoydutta\anaconda3\lib\site-packages (from jsonschem
a>=2.6->nbformat>=4.4->nbconvert) (21.4.0)
Requirement already satisfied: pyrsistent!=0.17.0,!=0.17.1,!=0.17.2,>=0.14.0 in c:\users\sujoydutta\anaconda3\l
ib\site-packages (from jsonschema>=2.6->nbformat>=4.4->nbconvert) (0.18.0)
Requirement already satisfied: six>=1.5 in c:\users\sujoydutta\anaconda3\lib\site-packages (from python-dateuti
l>=2.1->jupyter-client>=6.1.5->nbclient<0.6.0,>=0.5.0->nbconvert) (1.16.0)
Requirement already satisfied: soupsieve>1.2 in c:\users\sujoydutta\anaconda3\lib\site-packages (from beautiful
soup4->nbconvert) (2.3.1)
Requirement already satisfied: webencodings in c:\users\sujoydutta\anaconda3\lib\site-packages (from bleach->nb
convert) (0.5.1)
Requirement already satisfied: packaging in c:\users\sujoydutta\anaconda3\lib\site-packages (from bleach->nbcon
vert) (21.3)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in c:\users\sujoydutta\anaconda3\lib\site-packages (fro
m packaging->bleach->nbconvert) (3.0.4)
Note: you may need to restart the kernel to use updated packages.
```

In [4]:
```
pip install pandoc
```

```
Requirement already satisfied: pandoc in c:\users\sujoydutta\anaconda3\lib\site-packages (2.3)
Requirement already satisfied: plumbum in c:\users\sujoydutta\anaconda3\lib\site-packages (from pandoc) (1.8.0)
Requirement already satisfied: ply in c:\users\sujoydutta\anaconda3\lib\site-packages (from pandoc) (3.11)
Requirement already satisfied: pywin32 in c:\users\sujoydutta\anaconda3\lib\site-packages (from plumbum->pando
c) (302)
Note: you may need to restart the kernel to use updated packages.
```

In [5]:
```python
#installing packages
import numpy as np
import pandas as pd
import matplotlib as plt
import seaborn as sb
from scipy import stats
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import PolynomialFeatures
from sklearn.metrics import r2_score
from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import cross_val_predict
from sklearn.linear_model import Ridge
from sklearn.model_selection import GridSearchCV
import os
from collections import Counter
from numpy import mean
from numpy import std
from pandas import read_csv
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import RepeatedStratifiedKFold
from sklearn.metrics import precision_recall_curve
from sklearn.metrics import auc
from sklearn.metrics import make_scorer
from sklearn.dummy import DummyClassifier
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsClassifier
from sklearn.pipeline import Pipeline
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import RepeatedStratifiedKFold
from sklearn.metrics import precision_recall_curve
from sklearn.metrics import auc
from sklearn.metrics import make_scorer
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import ExtraTreesClassifier
from sklearn.ensemble import BaggingClassifier
from sklearn.naive_bayes import GaussianNB
from pandas import Series;  from numpy.random import randn
from statsmodels.stats.weightstats import ttest_ind
import scipy.stats as stats
from scipy.stats import ttest_ind
import pandas as pd
import numpy as np
import seaborn as sb
%matplotlib inline
from matplotlib import pyplot as plt
from scipy.stats import shapiro
from scipy import stats
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
from sklearn.metrics import classification_report
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import PolynomialFeatures
from sklearn.metrics import r2_score
from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import cross_val_predict
from sklearn.linear_model import Ridge
from sklearn.model_selection import GridSearchCV
from datetime import datetime
from datetime import date
import statsmodels.api as sm
from statsmodels.formula.api import ols
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA
from sklearn.linear_model import LogisticRegression as lgr

from sklearn import preprocessing as preproc
from sklearn import metrics
from sklearn.datasets import make_classification
from sklearn.model_selection import train_test_split
from sklearn.pipeline import make_pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsRegressor
from sklearn.neural_network import MLPRegressor
```

In [6]:
```python
#downloading the file
appdf= pd.read_csv("C:\\Users\\sujoydutta\\Desktop\\Data analysis\\Datasets\\googleplaystore.csv")
appdf.head()
```

Out[6]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19M	10,000+	Free	0	Everyone	Art & Design	January 7, 2018	1.0.0	4.0.3 and up
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14M	500,000+	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	4.0.3 and up
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7M	5,000,000+	Free	0	Everyone	Art & Design	August 1, 2018	1.2.4	4.0.3 and up
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25M	50,000,000+	Free	0	Teen	Art & Design	June 8, 2018	Varies with device	4.2 and up
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8M	100,000+	Free	0	Everyone	Art & Design;Creativity	June 20, 2018	1.1	4.4 and up

In [7]:
```python
#examining the dataset
appdf.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10841 entries, 0 to 10840
Data columns (total 13 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   App             10841 non-null  object 
 1   Category        10841 non-null  object 
 2   Rating          9367 non-null   float64
 3   Reviews         10841 non-null  object 
 4   Size            10841 non-null  object 
 5   Installs        10841 non-null  object 
 6   Type            10840 non-null  object 
 7   Price           10841 non-null  object 
 8   Content Rating  10840 non-null  object 
 9   Genres          10841 non-null  object 
 10  Last Updated    10841 non-null  object 
 11  Current Ver     10833 non-null  object 
 12  Android Ver     10838 non-null  object 
dtypes: float64(1), object(12)
memory usage: 1.1+ MB
```

In [8]:
```python
#examining the dataset
appdf.shape
```

Out[8]:
```
(10841, 13)
```

Step 2: Data cleaning

In this step, we are going to clean our dataset. We are going to look for null values and replace them with mean and mode. We are going to modify some variables if it is necessary and change datatypes for better analysis. We will also remove outliers from the dataset. Outliers hamper the machine learning algorithms and hence they have to be removed.

In [9]:
```python
#Converting M to 000 as 1MB=1000KB
appdf["Size"]=appdf["Size"].str.replace("M","000")
appdf["Size"]
```

Out[9]:
```
0                     19000
1                     14000
2                    8.7000
3                     25000
4                    2.8000
                ...        
10836                 53000
10837                3.6000
10838                9.5000
10839    Varies with device
10840                 19000
Name: Size, Length: 10841, dtype: object
```

In [10]:
```python
#replacing k with nil
appdf["Size"]=appdf["Size"].str.replace("k","")
appdf["Size"]
```

Out[10]:
```
0                     19000
1                     14000
2                    8.7000
3                     25000
4                    2.8000
                ...        
10836                 53000
10837                3.6000
10838                9.5000
10839    Varies with device
10840                 19000
Name: Size, Length: 10841, dtype: object
```

In [11]:
```python
#finding rows that have this irrelevant value and dropping them
vwd=appdf.loc[appdf["Size"]=="Varies with device"].index
vwd
```

Out[11]:
```
Int64Index([   37,    42,    52,    67,    68,    73,    85,    88,    89,
               92,
            ...
            10647, 10679, 10681, 10707, 10712, 10713, 10725, 10765, 10826,
            10839],
           dtype='int64', length=1695)
```

In [12]:
```python
appdf=appdf.drop(vwd)
appdf
```

Out[12]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Androi Ve
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19000	10,000+	Free	0	Everyone	Art & Design	January 7, 2018	1.0.0	4.0. and u
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14000	500,000+	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	4.0. and u
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7000	5,000,000+	Free	0	Everyone	Art & Design	August 1, 2018	1.2.4	4.0. and u
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25000	50,000,000+	Free	0	Teen	Art & Design	June 8, 2018	Varies with device	4.2 an u
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8000	100,000+	Free	0	Everyone	Art & Design;Creativity	June 20, 2018	1.1	4.4 an u
...	...	...	...	...	...	...	...	...	...	...	...	...	
10835	FR Forms	BUSINESS	NaN	0	9.6000	10+	Free	0	Everyone	Business	September 29, 2016	1.1.5	4.0 an u
10836	Sya9a Maroc - FR	FAMILY	4.5	38	53000	5,000+	Free	0	Everyone	Education	July 25, 2017	1.48	4.1 an u
10837	Fr. Mike Schmitz Audio Teachings	FAMILY	5.0	4	3.6000	100+	Free	0	Everyone	Education	July 6, 2018	1.0	4.1 an u
10838	Parkinson Exercices FR	MEDICAL	NaN	3	9.5000	1,000+	Free	0	Everyone	Medical	January 20, 2017	1.0	2.2 an u
10840	iHoroscope - 2018 Daily Horoscope & Astrology	LIFESTYLE	4.5	398307	19000	10,000,000+	Free	0	Everyone	Lifestyle	July 25, 2018	Varies with device	Varie wit devic

9146 rows × 13 columns

In [13]:
```python
#seeing if modification done correctly
appdf["Size"].value_counts()
```

Out[13]:
```
11000    198
12000    196
14000    194
13000    191
15000    184
        ... 
429        1
200        1
460        1
728        1
619        1
Name: Size, Length: 461, dtype: int64
```

In [14]:
```python
#removing dollar symbol in Price
appdf["Price"]=appdf["Price"].str.replace("$","")
appdf["Price"]
```

```
C:\Users\sujoydutta\AppData\Local\Temp\ipykernel_5680\2377636401.py:2: FutureWarning: The default value of rege
x will change from True to False in a future version. In addition, single character regular expressions will *n
ot* be treated as literal strings when regex=True.
  appdf["Price"]=appdf["Price"].str.replace("$","")
```

Out[14]:
```
0        0
1        0
2        0
3        0
4        0
        ..
10835    0
10836    0
10837    0
10838    0
10840    0
Name: Price, Length: 9146, dtype: object
```

In [15]:
```python
#checking the work
appdf["Price"].value_counts()
```

Out[15]:
```
0        8421
0.99      145
2.99      114
1.99       66
4.99       65
         ... 
19.90       1
1.75        1
14.00       1
4.85        1
1.04        1
Name: Price, Length: 88, dtype: int64
```

In [16]:
```python
#removing from price
ind=appdf[appdf["Price"]=='Everyone'].index
appdf=appdf.drop(ind)
appdf
```

Out[16]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Androi Ve
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19000	10,000+	Free	0	Everyone	Art & Design	January 7, 2018	1.0.0	4.0. and u
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14000	500,000+	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	4.0. and u
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7000	5,000,000+	Free	0	Everyone	Art & Design	August 1, 2018	1.2.4	4.0. and u
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25000	50,000,000+	Free	0	Teen	Art & Design	June 8, 2018	Varies with device	4.2 an u
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8000	100,000+	Free	0	Everyone	Art & Design;Creativity	June 20, 2018	1.1	4.4 an u
...	...	...	...	...	...	...	...	...	...	...	...	...	
10835	FR Forms	BUSINESS	NaN	0	9.6000	10+	Free	0	Everyone	Business	September 29, 2016	1.1.5	4.0 an u
10836	Sya9a Maroc - FR	FAMILY	4.5	38	53000	5,000+	Free	0	Everyone	Education	July 25, 2017	1.48	4.1 an u
10837	Fr. Mike Schmitz Audio Teachings	FAMILY	5.0	4	3.6000	100+	Free	0	Everyone	Education	July 6, 2018	1.0	4.1 an u
10838	Parkinson Exercices FR	MEDICAL	NaN	3	9.5000	1,000+	Free	0	Everyone	Medical	January 20, 2017	1.0	2.2 an u
10840	iHoroscope - 2018 Daily Horoscope & Astrology	LIFESTYLE	4.5	398307	19000	10,000,000+	Free	0	Everyone	Lifestyle	July 25, 2018	Varies with device	Varie wit devic

9145 rows × 13 columns

In [17]:
```python
#modifying number of installs
appdf["Installs"]=appdf["Installs"].str.replace("+","");
appdf["Installs"]=appdf["Installs"].str.replace("Free","0")
appdf["Installs"]=appdf["Installs"].str.replace(",","")
appdf["Installs"]
```

```
C:\Users\sujoydutta\AppData\Local\Temp\ipykernel_5680\1388774615.py:2: FutureWarning: The default value of rege
x will change from True to False in a future version. In addition, single character regular expressions will *n
ot* be treated as literal strings when regex=True.
  appdf["Installs"]=appdf["Installs"].str.replace("+","");
```

Out[17]:
```
0           10000
1          500000
2         5000000
3        50000000
4          100000
           ...   
10835          10
10836        5000
10837         100
10838        1000
10840    10000000
Name: Installs, Length: 9145, dtype: object
```

In [18]:
```python
#checking the changes if done properly
appdf["Installs"].value_counts()
```

Out[18]:
```
1000000       1303
100000        1054
10000         1011
1000           873
10000000       825
100            702
5000000        535
500000         492
5000           461
50000          446
10             376
500            322
100000000      201
50             201
50000000       147
5               78
1               64
500000000       30
0               14
1000000000      10
Name: Installs, dtype: int64
```

In [19]:
```python
#dropping unneccessary variables
appdf=appdf.drop(['Last Updated','Current Ver','Android Ver'],axis=1)
appdf
```

Out[19]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19000	10000	Free	0	Everyone	Art & Design
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14000	500000	Free	0	Everyone	Art & Design;Pretend Play
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7000	5000000	Free	0	Everyone	Art & Design
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25000	50000000	Free	0	Teen	Art & Design
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8000	100000	Free	0	Everyone	Art & Design;Creativity
...	...	...	...	...	...	...	...	...	...	...
10835	FR Forms	BUSINESS	NaN	0	9.6000	10	Free	0	Everyone	Business
10836	Sya9a Maroc - FR	FAMILY	4.5	38	53000	5000	Free	0	Everyone	Education
10837	Fr. Mike Schmitz Audio Teachings	FAMILY	5.0	4	3.6000	100	Free	0	Everyone	Education
10838	Parkinson Exercices FR	MEDICAL	NaN	3	9.5000	1000	Free	0	Everyone	Medical
10840	iHoroscope - 2018 Daily Horoscope & Astrology	LIFESTYLE	4.5	398307	19000	10000000	Free	0	Everyone	Lifestyle

9145 rows × 10 columns

In [20]:
```python
#changing data types
appdf["Price"] = appdf["Price"].astype(float)
appdf["Installs"]=appdf["Installs"].astype(float)
appdf["Reviews"]=appdf["Reviews"].astype(float)
appdf["Size"]=appdf["Size"].astype(float)
```

In [21]:
```python
#Seeing free apps
appdf.loc[appdf['Type']=='Free']
```

Out[21]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159.0	19000.0	10000.0	Free	0.0	Everyone	Art & Design
1	Coloring book moana	ART_AND_DESIGN	3.9	967.0	14000.0	500000.0	Free	0.0	Everyone	Art & Design;Pretend Play
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510.0	8.7	5000000.0	Free	0.0	Everyone	Art & Design
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644.0	25000.0	50000000.0	Free	0.0	Teen	Art & Design
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967.0	2.8	100000.0	Free	0.0	Everyone	Art & Design;Creativity
...	...	...	...	...	...	...	...	...	...	...
10835	FR Forms	BUSINESS	NaN	0.0	9.6	10.0	Free	0.0	Everyone	Business
10836	Sya9a Maroc - FR	FAMILY	4.5	38.0	53000.0	5000.0	Free	0.0	Everyone	Education
10837	Fr. Mike Schmitz Audio Teachings	FAMILY	5.0	4.0	3.6	100.0	Free	0.0	Everyone	Education
10838	Parkinson Exercices FR	MEDICAL	NaN	3.0	9.5	1000.0	Free	0.0	Everyone	Medical
10840	iHoroscope - 2018 Daily Horoscope & Astrology	LIFESTYLE	4.5	398307.0	19000.0	10000000.0	Free	0.0	Everyone	Lifestyle

8421 rows × 10 columns

In [22]:
```python
#seeing paid apps
index=appdf.loc[appdf['Price']>0]
index
```

Out[22]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres
234	TurboScan: scan documents and receipts in PDF	BUSINESS	4.7	11442.0	6.8	100000.0	Paid	4.99	Everyone	Business
235	Tiny Scanner Pro: PDF Doc Scan	BUSINESS	4.8	10295.0	39000.0	100000.0	Paid	4.99	Everyone	Business
290	TurboScan: scan documents and receipts in PDF	BUSINESS	4.7	11442.0	6.8	100000.0	Paid	4.99	Everyone	Business
291	Tiny Scanner Pro: PDF Doc Scan	BUSINESS	4.8	10295.0	39000.0	100000.0	Paid	4.99	Everyone	Business
477	Calculator	DATING	2.6	57.0	6.2	1000.0	Paid	6.99	Everyone	Dating
...	...	...	...	...	...	...	...	...	...	...
10735	FP VoiceBot	FAMILY	NaN	17.0	157.0	100.0	Paid	0.99	Mature 17+	Entertainment
10760	Fast Tract Diet	HEALTH_AND_FITNESS	4.4	35.0	2.4	1000.0	Paid	7.99	Everyone	Health & Fitness
10782	Trine 2: Complete Story	GAME	3.8	252.0	11000.0	10000.0	Paid	16.99	Teen	Action
10785	sugar, sugar	FAMILY	4.2	1405.0	9.5	10000.0	Paid	1.20	Everyone	Puzzle
10798	Word Search Tab 1 FR	FAMILY	NaN	0.0	1020.0	50.0	Paid	1.04	Everyone	Puzzle

724 rows × 10 columns

In [23]:
```python
#seeing if there is any free app which is charging a price
index['Type'].value_counts()
```

Out[23]:
```
Paid    724
Name: Type, dtype: int64
```

In [24]:
```python
#dropping apps which has more reviews than installs
bogus= appdf.loc[appdf['Reviews']>appdf['Installs']].index

print(len(bogus))
bogus
```

```
10
```

Out[24]:
```
Int64Index([2454, 4550, 5812, 5917, 6508, 6700, 7402, 8591, 9096, 10697], dtype='int64')
```

In [25]:
```python
appdf=appdf.drop(bogus)
appdf
```

Out[25]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159.0	19000.0	10000.0	Free	0.0	Everyone	Art & Design
1	Coloring book moana	ART_AND_DESIGN	3.9	967.0	14000.0	500000.0	Free	0.0	Everyone	Art & Design;Pretend Play
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510.0	8.7	5000000.0	Free	0.0	Everyone	Art & Design
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644.0	25000.0	50000000.0	Free	0.0	Teen	Art & Design
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967.0	2.8	100000.0	Free	0.0	Everyone	Art & Design;Creativity
...	...	...	...	...	...	...	...	...	...	...
10835	FR Forms	BUSINESS	NaN	0.0	9.6	10.0	Free	0.0	Everyone	Business
10836	Sya9a Maroc - FR	FAMILY	4.5	38.0	53000.0	5000.0	Free	0.0	Everyone	Education
10837	Fr. Mike Schmitz Audio Teachings	FAMILY	5.0	4.0	3.6	100.0	Free	0.0	Everyone	Education
10838	Parkinson Exercices FR	MEDICAL	NaN	3.0	9.5	1000.0	Free	0.0	Everyone	Medical
10840	iHoroscope - 2018 Daily Horoscope & Astrology	LIFESTYLE	4.5	398307.0	19000.0	10000000.0	Free	0.0	Everyone	Lifestyle

9135 rows × 10 columns

In [26]:
```python
#seeing null values
missing_data=appdf.isnull()
missing_data
```

Out[26]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres
0	False	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...
10835	False	False	True	False	False	False	False	False	False	False
10836	False	False	False	False	False	False	False	False	False	False
10837	False	False	False	False	False	False	False	False	False	False
10838	False	False	True	False	False	False	False	False	False	False
10840	False	False	False	False	False	False	False	False	False	False

9135 rows × 10 columns

In [27]:
```python
#let us see the number of missing values in each column
for column in missing_data.columns.values.tolist():
  print(column,missing_data[column].value_counts())
  print("")
```

```
App False    9135
Name: App, dtype: int64

Category False    9135
Name: Category, dtype: int64

Rating False    7723
True     1412
Name: Rating, dtype: int64

Reviews False    9135
Name: Reviews, dtype: int64

Size False    9135
Name: Size, dtype: int64

Installs False    9135
Name: Installs, dtype: int64

Type False    9135
Name: Type, dtype: int64

Price False    9135
Name: Price, dtype: int64

Content Rating False    9135
Name: Content Rating, dtype: int64

Genres False    9135
Name: Genres, dtype: int64
```

In [28]:
```python
#seeing null values
appdf.isnull().sum()
```

Out[28]:
```
App                  0
Category             0
Rating            1412
Reviews              0
Size                 0
Installs             0
Type                 0
Price                0
Content Rating       0
Genres               0
dtype: int64
```

In [29]:
```python
#replacing missing values with nan(Not a number) that makes things simpler for python
appdf.replace("?",np.nan,inplace = True)
print(appdf.info())
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 9135 entries, 0 to 10840
Data columns (total 10 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   App             9135 non-null   object 
 1   Category        9135 non-null   object 
 2   Rating          7723 non-null   float64
 3   Reviews         9135 non-null   float64
 4   Size            9135 non-null   float64
 5   Installs        9135 non-null   float64
 6   Type            9135 non-null   object 
 7   Price           9135 non-null   float64
 8   Content Rating  9135 non-null   object 
 9   Genres          9135 non-null   object 
dtypes: float64(5), object(5)
memory usage: 785.0+ KB
None
```

In [30]:
```python
#taking mean of numerical values
avgrating=appdf['Rating'].mean()
avgprice=appdf['Price'].mean()
avginst=appdf['Installs'].mean()
avgrev=appdf['Reviews'].mean()
avgsize=appdf['Size'].mean()
```

In [31]:
```python
#replacing blank with mean
appdf['Rating'].replace(np.nan, avgrating, inplace=True)
```

In [32]:
```python
#seeing the outliers for Rating
sb.boxplot(appdf['Rating'])
```

```
stat=0.297, p=0.000
Probably not Gaussian
```

In [109…:
```python
#checking if price is dependent on app popularity
inscat=appdf['Installs-bin']
mod= ols("Price ~ inscat", data = appdf).fit()
anov_table = sm.stats.anova_lm(mod)
anov_table
```

Out[109]:

	df	sum_sq	mean_sq	F	PR(>F)
inscat	5.0	321.310924	64.262185	52.309415	1.101626e-53
Residual	9129.0	11214.988560	1.228501	NaN	NaN

In [85]:
```python
#checking if price is dependent on app size
sizcat=appdf['Size-bin']
mod= ols("Price ~ sizcat", data = appdf).fit()
anov_table = sm.stats.anova_lm(mod)
anov_table
```

Out[85]:

	df	sum_sq	mean_sq	F	PR(>F)
sizcat	2.0	21.357180	10.678590	8.383646	0.00023
Residual	8717.0	11103.196534	1.273741	NaN	NaN

In [86]:
```python
#checking if Rating is dependent on app popularity
inscat=appdf['Installs-bin']
mod = ols("Rating ~ inscat", data = appdf).fit()
anov_table = sm.stats.anova_lm(mod)
anov_table
```

Out[86]:

	df	sum_sq	mean_sq	F	PR(>F)
inscat	5.0	9.036342	1.807268	27.897998	3.805860e-28
Residual	9129.0	591.388409	0.064781	NaN	NaN

In [87]:
```python
appdf.head(0)
```

Out[87]:

App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Installs-bin	Size-bin

In [88]:
```python
#checking if Rating differs to apps being free or paid
Type=appdf['Type']
mod = ols("Rating ~ Type", data = appdf).fit()
anov_table = sm.stats.anova_lm(mod)
anov_table
```

Out[88]:

	df	sum_sq	mean_sq	F	PR(>F)
Type	1.0	0.929794	0.929794	14.164939	0.000169
Residual	9133.0	599.494957	0.065641	NaN	NaN

In [89]:
```python
#checking if Rating differs according to content type
Conra=appdf['Content Rating']
mod = ols("Rating ~ Conra", data = appdf).fit()
anov_table = sm.stats.anova_lm(mod)
anov_table
```

Out[89]:

	df	sum_sq	mean_sq	F	PR(>F)
Conra	5.0	2.219889	0.443978	6.775393	0.000003
Residual	9129.0	598.204862	0.065528	NaN	NaN

In [90]:
```python
#checking if Rating is different in different Category
Cat=appdf['Category']
mod = ols("Rating ~ Cat", data = appdf).fit()
anov_table = sm.stats.anova_lm(mod)
anov_table
```

Out[90]:

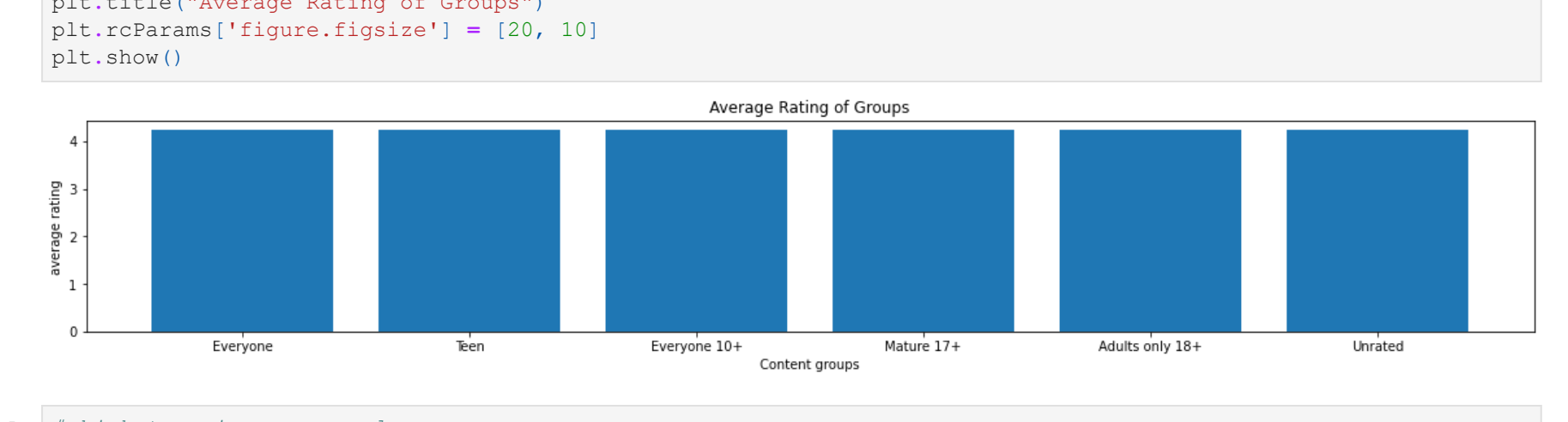
	df	sum_sq	mean_sq	F	PR(>F)
Cat	32.0	14.122668	0.441333	6.851445	2.276268e-29
Residual	9102.0	586.302083	0.064415	NaN	NaN

In [91]:
```python
#seeing average by content rating
appdf.groupby('Content Rating').mean().round(2)
```

Out[91]:

	Rating	Reviews	Size	Installs	Price
Content Rating					
Adults only 18+	4.55	37011.00	20502.45	750000.00	0.00
Everyone	4.22	14309.78	17215.27	2102136.54	0.35
Everyone 10+	4.29	29292.61	38949.73	3465137.88	0.38
Mature 17+	4.18	20198.65	23833.02	2164815.76	0.17
Teen	4.23	21291.71	29108.00	2968605.79	0.22
Unrated	4.14	593.50	4.45	25250.00	0.00

In [122…:
```python
#which content rating is more rated
x=appdf['Content Rating']
y=appdf['Rating'].mean()
#visualization stage
plt.bar(x, y)
plt.xlabel("Content groups")
plt.ylabel("average rating")
plt.title("Average Rating of Groups")
plt.rcParams['figure.figsize'] = [20, 10]
plt.show()
```



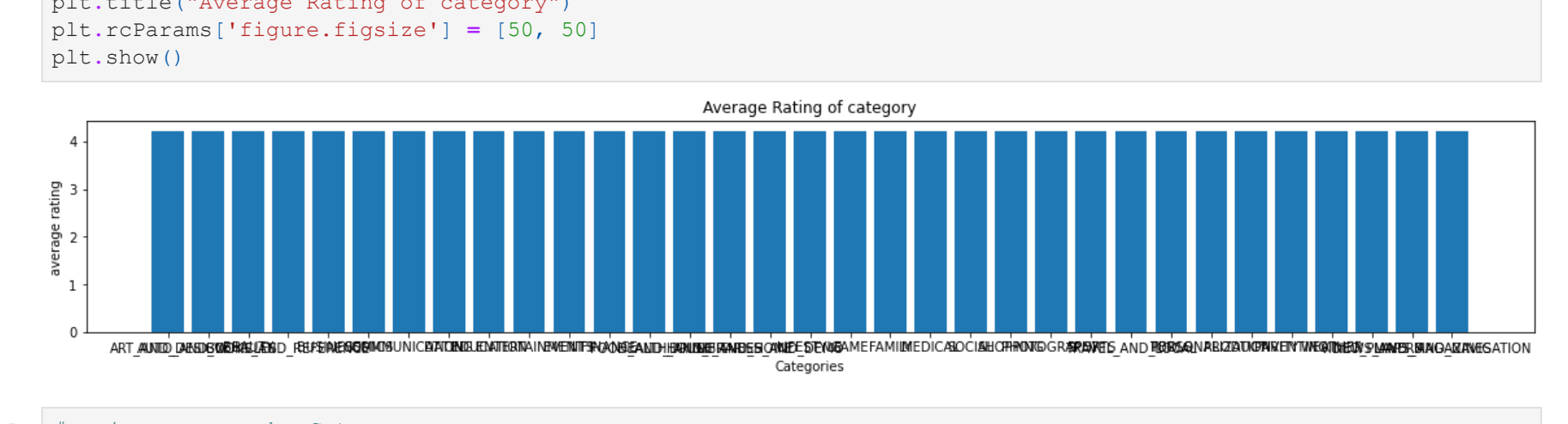
In [93]:
```python
#which type is more popular

partofday=appdf['Type'].unique()
incidents=appdf['Type'].value_counts()

plt.pie(incidents, labels = partofday, startangle = 90,autopct='%.0f%%')
plt.show()
```

In [94]:
```python
#seeing average by Category
appdf.groupby('Category').mean().round(2)
```

Out[94]:

	Rating	Reviews	Size	Installs	Price
Category					
ART_AND_DESIGN	4.32	17537.86	9470.65	1025538.09	0.20
AUTO_AND_VEHICLES	4.17	9605.64	17923.26	1438317.04	0.11
BEAUTY	4.26	3975.75	10939.03	442968.74	0.00
BOOKS_AND_REFERENCE	4.26	9320.97	10354.90	1795130.09	0.57
BUSINESS	4.20	6065.91	11962.18	2139010.27	0.14
COMICS	4.25	12082.24	11199.41	484484.92	0.00
COMMUNICATION	4.17	20788.97	9725.35	2877211.29	0.36
DATING	4.14	13204.53	14181.78	1307596.17	0.10
EDUCATION	4.36	43212.06	19299.53	2511729.73	0.15
ENTERTAINMENT	4.16	35470.03	20823.25	3977122.75	0.05
EVENTS	4.25	2222.35	11587.19	936980.45	0.07
FAMILY	4.22	14329.14	26666.52	1906236.09	0.40
FINANCE	4.23	9351.85	15957.58	1077802.26	0.23
FOOD_AND_DRINK	4.19	18182.41	21021.16	2417921.78	0.04
GAME	4.28	26108.27	43945.59	4033074.90	0.32
HEALTH_AND_FITNESS	4.27	19349.90	20449.56	2231866.51	0.19
HOUSE_AND_HOME	4.21	11893.46	12895.09	1418827.83	0.00
LIBRARIES_AND_DEMO	4.20	8382.88	9622.60	841689.40	0.05
LIFESTYLE	4.18	8412.70	12323.31	1598137.52	0.21
MAPS_AND_NAVIGATION	4.18	15337.37	14100.56	1553385.05	0.15
MEDICAL	4.22	3288.00	17908.88	1226053.58	0.94
NEWS_AND_MAGAZINES	4.22	14139.46	10346.70	1925119.03	0.04
PARENTING	4.26	4196.61	20113.51	575807.98	0.08
PERSONALIZATION	4.27	12374.41	8702.93	2050339.33	0.92
PHOTOGRAPHY	4.24	29029.71	14911.86	3656295.51	0.31
PRODUCTIVITY	4.22	15239.68	10569.64	2446136.08	0.33
SHOPPING	4.25	27315.52	14271.36	3466175.71	0.04
SOCIAL	4.23	15515.90	15192.95	2715672.41	0.06
SPORTS	4.23	22954.83	22730.01	2809365.12	0.30
TOOLS	4.18	11814.32	6174.50	1873521.70	0.38
TRAVEL_AND_LOCAL	4.19	19211.85	23263.18	2228605.78	0.22
VIDEO_PLAYERS	4.18	19839.93	13779.25	2498773.06	0.03
WEATHER	4.22	22181.95	10591.69	1941541.98	0.44

In [95]:
```python
#which content rating is more popular
x=appdf['Category']
y=appdf['Rating'].mean()
#visualization stage
plt.bar(x, y)
plt.xlabel("Categories")
plt.ylabel("average rating")
plt.title("Average Rating of category")
plt.rcParams['figure.figsize'] = [50, 50]
plt.show()
```



In [96]:
```python
#seeing average by Category
appdf.groupby('Genres').mean().round(2)
```

Out[96]:

	Rating	Reviews	Size	Installs	Price
Genres					
Action	4.28	22778.07	50164.34	4201242.30	0.31
Action;Action & Adventure	4.30	34379.34	53000.00	4876281.25	0.78
Adventure	4.19	27286.89	36160.36	2000915.84	0.73
Adventure;Action & Adventure	4.42	7866.89	49693.37	5404129.10	0.64
Adventure;Brain Games	4.60	7148.00	8.10	100000.00	4.17
...	...	...	...	...	...
Video Players & Editors	4.18	18764.68	13708.33	2479532.85	0.03
Video Players & Editors;Creativity	4.10	159620.50	23000.00	5000000.00	0.00
Video Players & Editors;Music & Video	4.00	39779.00	19000.00	5000000.00	0.00
Weather	4.22	22181.95	10591.69	1941541.98	0.44
Word	4.29	78167.07	34481.24	4736710.47	0.00

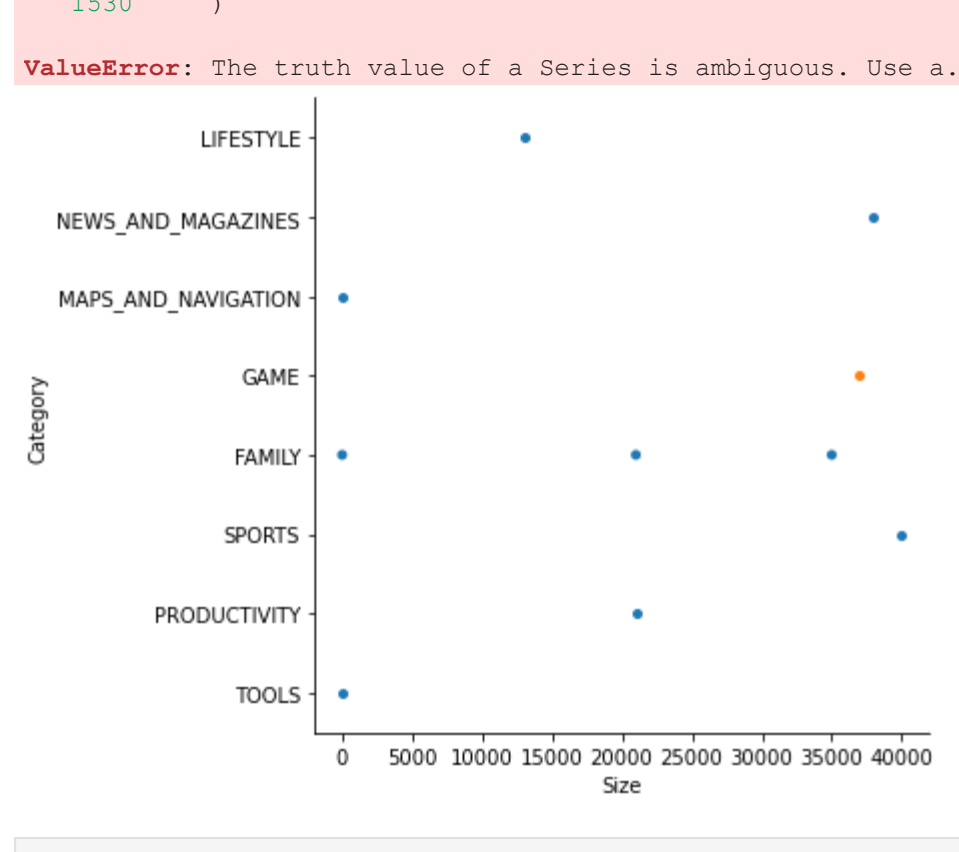
116 rows × 5 columns

In [97]:
```python
#top ten category by rating
top10cat=appdf.sort_values('Rating')['Category'].head(10)
top10cat
```

Out[97]:
```
1580              LIFESTYLE
7450     NEWS_AND_MAGAZINES
3867    MAPS_AND_NAVIGATION
7665                   GAME
8773                 FAMILY
6836                 FAMILY
6833                 SPORTS
7651           PRODUCTIVITY
9798                  TOOLS
9767                 FAMILY
Name: Category, dtype: object
```

In [98]:
```python
#which category apps are the heaviest
sb.catplot(data=appdf, x="Size", y=top10cat, hue="Type", kind="swarm")
plt.rcParams['figure.figsize'] = [50, 10]

"blue is free and yellow is paid"
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Input In [98], in <cell line: 2>()
      1 #which category apps are the heaviest
----> 2 sb.catplot(data=appdf, x="Size", y=top10cat, hue="Type", kind="swarm")
      3 plt.rcParams['figure.figsize'] = [50, 10]
      5 "blue is free and yellow is paid"

File ~\anaconda3\lib\site-packages\seaborn\_decorators.py:46, in _deprecate_positional_args.<locals>.inner_f(*args, **kwargs)
     36     warnings.warn(
     37         "Pass the following variable{} as {}keyword arg{}: {}. "
     38         "From version 0.12, the only valid positional argument "
   (...)
     43         FutureWarning
     44     )
     45 kwargs.update({k: arg for k, arg in zip(sig.parameters, args)})
---> 46 return f(**kwargs)

File ~\anaconda3\lib\site-packages\seaborn\categorical.py:3847, in catplot(x, y, hue, data, row, col, col_wrap, estimator, ci, n_boot, units, seed, order, hue_order, row_order, col_order, kind, height, aspect, orient, color, palette, legend, legend_out, sharex, sharey, margin_titles, facet_kws, **kwargs)
   3844 g = FacetGrid(**facet_kws)
   3846 # Draw the plot onto the facets
-> 3847 g.map_dataframe(plot_func, x=x, y=y, hue=hue, **plot_kws)
   3849 if p.orient == "h":
   3850     g.set_axis_labels(p.value_label, p.group_label)

File ~\anaconda3\lib\site-packages\seaborn\axisgrid.py:784, in FacetGrid.map_dataframe(self, func, *args, **kwargs)
    782 for i, val in enumerate(args[:2]):
    783     axis_labels[i] = val
--> 784 self._finalize_grid(axis_labels)
    786 return self

File ~\anaconda3\lib\site-packages\seaborn\axisgrid.py:813, in FacetGrid._finalize_grid(self, axlabels)
    811 def _finalize_grid(self, axlabels):
    812     """Finalize the annotations and layout."""
--> 813     self.set_axis_labels(*axlabels)
    814     self.set_titles()
    815     self.tight_layout()

File ~\anaconda3\lib\site-packages\seaborn\axisgrid.py:843, in FacetGrid.set_axis_labels(self, x_var, y_var, clear_inner, **kwargs)
    841 if y_var is not None:
    842     self._y_var = y_var
--> 843     self.set_ylabels(y_var, clear_inner=clear_inner, **kwargs)
    845 return self

File ~\anaconda3\lib\site-packages\seaborn\axisgrid.py:863, in FacetGrid.set_ylabels(self, label, clear_inner, **kwargs)
    861 label = self._y_var
    862 for ax in self._left_axes:
--> 863     ax.set_ylabel(label, **kwargs)
    864 if clear_inner:
    865     for ax in self._not_left_axes:

File ~\anaconda3\lib\site-packages\matplotlib\axes\_base.py:3852, in _AxesBase.set_ylabel(self, ylabel, fontdict, labelpad, loc, **kwargs)
   3849 elif loc == 'top':
   3850     kwargs.update(y=1, horizontalalignment='right')
-> 3852 return self.yaxis.set_label_text(ylabel, fontdict, **kwargs)

File ~\anaconda3\lib\site-packages\matplotlib\axis.py:1560, in Axis.set_label_text(self, label, fontdict, **kwargs)
   1547 """
   1548 Set the text value of the axis label.
   1549
   (...)
   1557     Merged into fontdict.
   1558 """
   1559 self.isDefault_label = False
-> 1560 self.label.set_text(label)
   1561 if fontdict is not None:
   1562     self.label.update(fontdict)

File ~\anaconda3\lib\site-packages\matplotlib\text.py:1223, in Text.set_text(self, s)
   1221 if s is None:
   1222     s = ''
-> 1223 if s != self._text:
   1224     self._text = str(s)
   1225     self.stale = True

File ~\anaconda3\lib\site-packages\pandas\core\generic.py:1527, in NDFrame.__nonzero__(self)
   1525 @final
   1526 def __nonzero__(self):
-> 1527     raise ValueError(
   1528         f"The truth value of a {type(self).__name__} is ambiguous. "
   1529         "Use a.empty, a.bool(), a.item(), a.any() or a.all()."
   1530     )

ValueError: The truth value of a Series is ambiguous. Use a.empty, a.bool(), a.item(), a.any() or a.all().
```

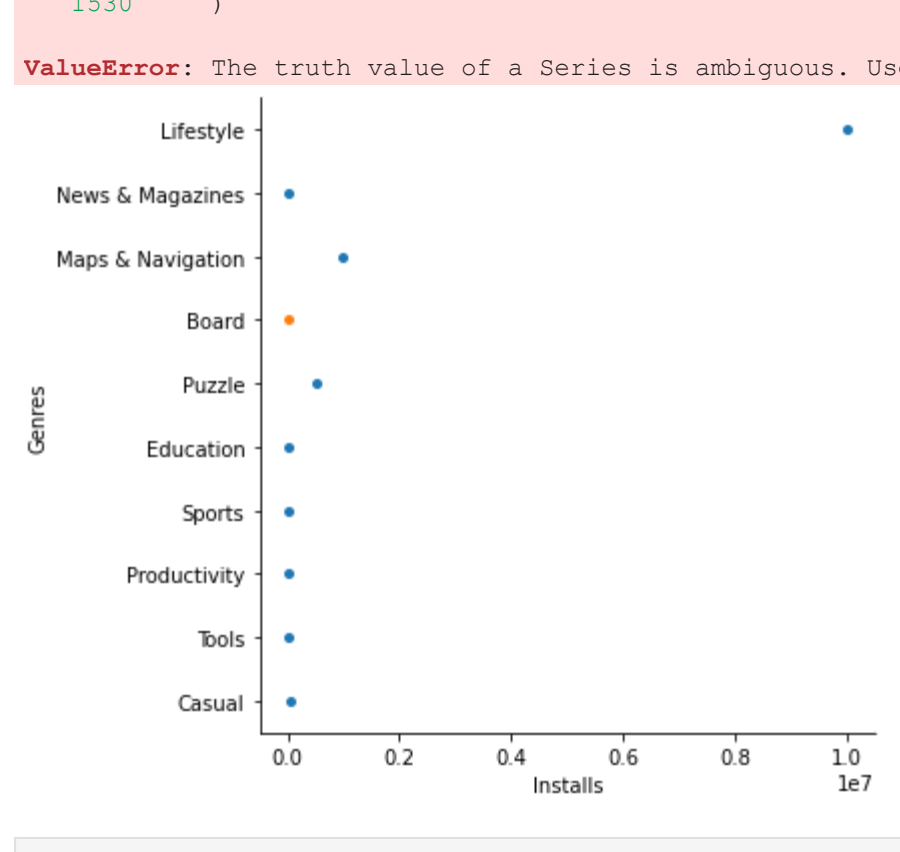


In [99]:
```python
#top ten Genre by rating
top10gen=appdf.sort_values('Rating')['Genres'].head(10)
top10gen
```

Out[99]:
```
1580            Lifestyle
7450     News & Magazines
3867    Maps & Navigation
7665                Board
8773               Puzzle
6836            Education
6833               Sports
7651         Productivity
9798                Tools
9767               Casual
Name: Genres, dtype: object
```

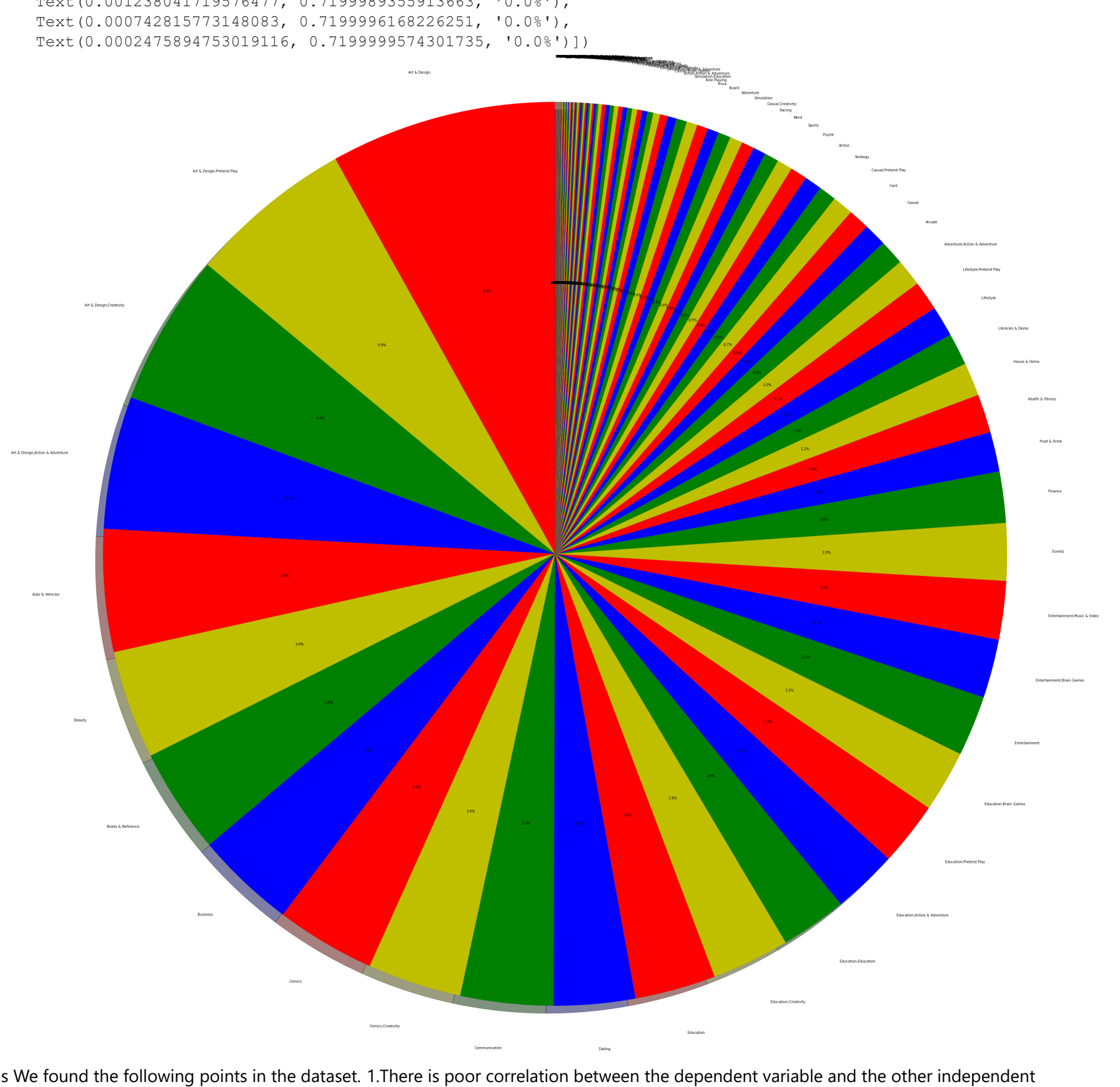
In [100…:
```python
#Number of installs for top category
sb.catplot(data=appdf, x="Installs", y=top10gen, hue="Type", kind="swarm")


"blue is free and yellow is paid"
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Input In [100], in <cell line: 2>()
      1 #Number of installs for top category
----> 2 sb.catplot(data=appdf, x="Installs", y=top10gen, hue="Type", kind="swarm")
      5 "blue is free and yellow is paid"

File ~\anaconda3\lib\site-packages\seaborn\_decorators.py:46, in _deprecate_positional_args.<locals>.inner_f(*args, **kwargs)
     36     warnings.warn(
     37         "Pass the following variable{} as {}keyword arg{}: {}. "
     38         "From version 0.12, the only valid positional argument "
   (...)
     43         FutureWarning
     44     )
     45 kwargs.update({k: arg for k, arg in zip(sig.parameters, args)})
---> 46 return f(**kwargs)

File ~\anaconda3\lib\site-packages\seaborn\categorical.py:3847, in catplot(x, y, hue, data, row, col, col_wrap, estimator, ci, n_boot, units, seed, order, hue_order, row_order, col_order, kind, height, aspect, orient, color, palette, legend, legend_out, sharex, sharey, margin_titles, facet_kws, **kwargs)
   3844 g = FacetGrid(**facet_kws)
   3846 # Draw the plot onto the facets
-> 3847 g.map_dataframe(plot_func, x=x, y=y, hue=hue, **plot_kws)
   3849 if p.orient == "h":
   3850     g.set_axis_labels(p.value_label, p.group_label)

File ~\anaconda3\lib\site-packages\seaborn\axisgrid.py:784, in FacetGrid.map_dataframe(self, func, *args, **kwargs)
    782 for i, val in enumerate(args[:2]):
    783     axis_labels[i] = val
--> 784 self._finalize_grid(axis_labels)
    786 return self

File ~\anaconda3\lib\site-packages\seaborn\axisgrid.py:813, in FacetGrid._finalize_grid(self, axlabels)
    811 def _finalize_grid(self, axlabels):
    812     """Finalize the annotations and layout."""
--> 813     self.set_axis_labels(*axlabels)
    814     self.set_titles()
    815     self.tight_layout()

File ~\anaconda3\lib\site-packages\seaborn\axisgrid.py:843, in FacetGrid.set_axis_labels(self, x_var, y_var, clear_inner, **kwargs)
    841 if y_var is not None:
    842     self._y_var = y_var
--> 843     self.set_ylabels(y_var, clear_inner=clear_inner, **kwargs)
    845 return self

File ~\anaconda3\lib\site-packages\seaborn\axisgrid.py:863, in FacetGrid.set_ylabels(self, label, clear_inner, **kwargs)
    861 label = self._y_var
    862 for ax in self._left_axes:
--> 863     ax.set_ylabel(label, **kwargs)
    864 if clear_inner:
    865     for ax in self._not_left_axes:

File ~\anaconda3\lib\site-packages\matplotlib\axes\_base.py:3852, in _AxesBase.set_ylabel(self, ylabel, fontdict, labelpad, loc, **kwargs)
   3849 elif loc == 'top':
   3850     kwargs.update(y=1, horizontalalignment='right')
-> 3852 return self.yaxis.set_label_text(ylabel, fontdict, **kwargs)

File ~\anaconda3\lib\site-packages\matplotlib\axis.py:1560, in Axis.set_label_text(self, label, fontdict, **kwargs)
   1547 """
   1548 Set the text value of the axis label.
   1549
   (...)
   1557     Merged into fontdict.
   1558 """
   1559 self.isDefault_label = False
-> 1560 self.label.set_text(label)
   1561 if fontdict is not None:
   1562     self.label.update(fontdict)

File ~\anaconda3\lib\site-packages\matplotlib\text.py:1223, in Text.set_text(self, s)
   1221 if s is None:
   1222     s = ''
-> 1223 if s != self._text:
   1224     self._text = str(s)
   1225     self.stale = True

File ~\anaconda3\lib\site-packages\pandas\core\generic.py:1527, in NDFrame.__nonzero__(self)
   1525 @final
   1526 def __nonzero__(self):
-> 1527     raise ValueError(
   1528         f"The truth value of a {type(self).__name__} is ambiguous. "
   1529         "Use a.empty, a.bool(), a.item(), a.any() or a.all()."
   1530     )

ValueError: The truth value of a Series is ambiguous. Use a.empty, a.bool(), a.item(), a.any() or a.all().
```



In [101…:
```python
#which Genre is more popular

genretypes=appdf['Genres'].unique()
number=appdf['Genres'].value_counts()


# color for each label
colors = ['r', 'y', 'g', 'b']

# plotting the pie chart
plt.pie(number, labels = genretypes, colors=colors,
        startangle=90, shadow = True,
        radius = 1.2, autopct = '%1.1f%%')
```

Out[101]:
```
([<matplotlib.patches.Wedge at 0x102d7863730>,
  <matplotlib.patches.Wedge at 0x102d786a100>,
  <matplotlib.patches.Wedge at 0x102d786aa90>,
  <matplotlib.patches.Wedge at 0x102d7875460>,
  <matplotlib.patches.Wedge at 0x102d78636d0>,
  <matplotlib.patches.Wedge at 0x102d787d790>,
  <matplotlib.patches.Wedge at 0x102d7888160>,
  <matplotlib.patches.Wedge at 0x102d7888af0>,
  <matplotlib.patches.Wedge at 0x102d78924c0>,
  <matplotlib.patches.Wedge at 0x102d7892e50>,
  <matplotlib.patches.Wedge at 0x102d789a820>,
  <matplotlib.patches.Wedge at 0x102d78a71f0>,
  <matplotlib.patches.Wedge at 0x102d78a7b80>,
  <matplotlib.patches.Wedge at 0x102d78b3550>,
  <matplotlib.patches.Wedge at 0x102d78b3ee0>,
  <matplotlib.patches.Wedge at 0x102d78bf8b0>,
  <matplotlib.patches.Wedge at 0x102d78ce280>,
  <matplotlib.patches.Wedge at 0x102d78cec10>,
  <matplotlib.patches.Wedge at 0x102d78dc5e0>,
  <matplotlib.patches.Wedge at 0x102d78dcf70>,
  <matplotlib.patches.Wedge at 0x102d78e8940>,
  <matplotlib.patches.Wedge at 0x102d78f3310>,
  <matplotlib.patches.Wedge at 0x102d78f3ca0>,
  <matplotlib.patches.Wedge at 0x102d7901670>,
  <matplotlib.patches.Wedge at 0x102d790c040>,
  <matplotlib.patches.Wedge at 0x102d790c9d0>,
  <matplotlib.patches.Wedge at 0x102d791b3a0>,
  <matplotlib.patches.Wedge at 0x102d791bd30>,
  <matplotlib.patches.Wedge at 0x102d79664c0>,
  <matplotlib.patches.Wedge at 0x102d7966e50>,
  <matplotlib.patches.Wedge at 0x102d7972820>,
  <matplotlib.patches.Wedge at 0x102d79801f0>,
  <matplotlib.patches.Wedge at 0x102d7980b80>,
  <matplotlib.patches.Wedge at 0x102d798c550>,
  <matplotlib.patches.Wedge at 0x102d798cee0>,
  <matplotlib.patches.Wedge at 0x102d79998b0>,
  <matplotlib.patches.Wedge at 0x102d79a5280>,
  <matplotlib.patches.Wedge at 0x102d79a5c10>,
  <matplotlib.patches.Wedge at 0x102d79b35e0>,
  <matplotlib.patches.Wedge at 0x102d79b3f70>,
  <matplotlib.patches.Wedge at 0x102d79be940>,
  <matplotlib.patches.Wedge at 0x102d79cb310>,
  <matplotlib.patches.Wedge at 0x102d79cbca0>,
  <matplotlib.patches.Wedge at 0x102d79d9670>,
  <matplotlib.patches.Wedge at 0x102d79e5040>,
  <matplotlib.patches.Wedge at 0x102d79e59d0>,
  <matplotlib.patches.Wedge at 0x102d79f13a0>,
  <matplotlib.patches.Wedge at 0x102d79f1d30>,
  <matplotlib.patches.Wedge at 0x102d79fe700>,
  <matplotlib.patches.Wedge at 0x102d7a0b0d0>,
  <matplotlib.patches.Wedge at 0x102d7a0ba60>,
  <matplotlib.patches.Wedge at 0x102d7a19430>,
  <matplotlib.patches.Wedge at 0x102d7a19dc0>,
  <matplotlib.patches.Wedge at 0x102d7a23790>,
  <matplotlib.patches.Wedge at 0x102d7a31160>,
  <matplotlib.patches.Wedge at 0x102d7a31af0>,
  <matplotlib.patches.Wedge at 0x102d7a3e4c0>,
  <matplotlib.patches.Wedge at 0x102d7a3ee50>,
  <matplotlib.patches.Wedge at 0x102d7a4b820>,
  <matplotlib.patches.Wedge at 0x102d7a57220>,
  <matplotlib.patches.Wedge at 0x102d7a57bb0>,
  <matplotlib.patches.Wedge at 0x102d7a63580>,
  <matplotlib.patches.Wedge at 0x102d7a63f10>,
  <matplotlib.patches.Wedge at 0x102d7a708e0>,
  <matplotlib.patches.Wedge at 0x102d7a7e2b0>,
  <matplotlib.patches.Wedge at 0x102d7a7ec40>,
  <matplotlib.patches.Wedge at 0x102d7a8d640>,
  <matplotlib.patches.Wedge at 0x102d7a8dfd0>,
  <matplotlib.patches.Wedge at 0x102d7a9a9a0>,
  <matplotlib.patches.Wedge at 0x102d7aa6370>,
  <matplotlib.patches.Wedge at 0x102d7aa6d00>,
  <matplotlib.patches.Wedge at 0x102d7ab46d0>,
  <matplotlib.patches.Wedge at 0x102d7ac10a0>,
  <matplotlib.patches.Wedge at 0x102d7ac1a30>,
  <matplotlib.patches.Wedge at 0x102d7ad0400>,
  <matplotlib.patches.Wedge at 0x102d7ad0d90>,
  <matplotlib.patches.Wedge at 0x102d7adc760>,
  <matplotlib.patches.Wedge at 0x102d7ae9130>,
  <matplotlib.patches.Wedge at 0x102d7ae9ac0>,
  <matplotlib.patches.Wedge at 0x102d7af9490>,
  <matplotlib.patches.Wedge at 0x102d7af9e20>,
  <matplotlib.patches.Wedge at 0x102d7b047f0>,
  <matplotlib.patches.Wedge at 0x102d7b121c0>,
  <matplotlib.patches.Wedge at 0x102d7b12b50>,
  <matplotlib.patches.Wedge at 0x102d7b1f520>,
  <matplotlib.patches.Wedge at 0x102d7b1feb0>,
  <matplotlib.patches.Wedge at 0x102d7b2b880>,
  <matplotlib.patches.Wedge at 0x102d7b3b250>,
  <matplotlib.patches.Wedge at 0x102d7b3bbe0>,
  <matplotlib.patches.Wedge at 0x102d7b475b0>,
  <matplotlib.patches.Wedge at 0x102d7b47f40>,
  <matplotlib.patches.Wedge at 0x102d7b54910>,
  <matplotlib.patches.Wedge at 0x102d7b612e0>,
  <matplotlib.patches.Wedge at 0x102d7b61c70>,
  <matplotlib.patches.Wedge at 0x102d7b6f640>,
  <matplotlib.patches.Wedge at 0x102d7b6ffd0>,
  <matplotlib.patches.Wedge at 0x102d7b7d9a0>,
  <matplotlib.patches.Wedge at 0x102d7b8b370>,
  <matplotlib.patches.Wedge at 0x102d7b8bd00>,
  <matplotlib.patches.Wedge at 0x102d7b996d0>,
  <matplotlib.patches.Wedge at 0x102d7ba40a0>,
  <matplotlib.patches.Wedge at 0x102d7ba4a30>,
  <matplotlib.patches.Wedge at 0x102d7bb3400>,
  <matplotlib.patches.Wedge at 0x102d7bb3d90>,
  <matplotlib.patches.Wedge at 0x102d7bc0760>,
  <matplotlib.patches.Wedge at 0x102d7bce130>,
  <matplotlib.patches.Wedge at 0x102d7bceac0>,
  <matplotlib.patches.Wedge at 0x102d7bdd490>,
  <matplotlib.patches.Wedge at 0x102d7bdde20>,
  <matplotlib.patches.Wedge at 0x102d7be87f0>,
  <matplotlib.patches.Wedge at 0x102d7bf71c0>,
  <matplotlib.patches.Wedge at 0x102d7bf7b50>,
  <matplotlib.patches.Wedge at 0x102d7c04520>,
  <matplotlib.patches.Wedge at 0x102d7c04eb0>,
  <matplotlib.patches.Wedge at 0x102d7c10880>,
  <matplotlib.patches.Wedge at 0x102d7c20250>],
 [Text(-0.3314354617438872, 1.2777130095207672, 'Art & Design'),
  Text(-0.8425534222544132, 1.0161219073749845, 'Art & Design;Pretend Play'),
  Text(-1.143153553593345, 0.6599999643234137, 'Art & Design;Creativity'),
  Text(-1.292507338184218, 0.26800145660042185, 'Art & Design;Action & Adventure'),
  Text(-1.3155622771543323, -0.10814756089948652, 'Auto & Vehicles'),
  Text(-1.2438504875629617, -0.4418551398245611, 'Beauty'),
  Text(-1.1031348320289724, -0.7249093338924605, 'Books & Reference'),
  Text(-0.9080927737375538, -0.9580018341765512, 'Business'),
  Text(-0.6723448601958608, -1.1359367891604746, 'Comics'),
  Text(-0.41157041954681456, -1.254196870413118, 'Comics;Creativity'),
  Text(-0.14136309487169338, -1.3124086541197055, 'Communication'),
  Text(0.11606310131277914, -1.3148875832228624, 'Dating'),
  Text(0.35204378470045167, -1.2721891265270986, 'Education'),
  Text(0.5719084314475822, -1.1896725373140151, 'Education;Creativity'),
  Text(0.7559119139050584, -1.0821262303522596, 'Education;Education'),
  Text(0.9064441794982532, -0.9595618528545925, 'Education;Action & Adventure'),
  Text(1.0354052460417142, -0.818740481757985, 'Education;Pretend Play'),
  Text(1.1401914183229227, -0.6651041494253076, 'Education;Brain Games'),
  Text(1.220279653584577, -0.5033066332242251, 'Entertainment'),
  Text(1.2765078734427449, -0.33604709348346007, 'Entertainment;Brain Games'),
  Text(1.309585720744224, -0.16548486342512236, 'Entertainment;Music & Video'),
  Text(1.3199878029841399, 0.005674502013701636, 'Events'),
  Text(1.309528719975661, 0.16593532342122655, 'Finance'),
  Text(1.28578446161318, 0.2985939019372387, 'Food & Drink'),
  Text(1.2545502309377061, 0.4104920438378167, 'Health & Fitness'),
  Text(1.2174917045251776, 0.5100136757111299, 'House & Home'),
  Text(1.1769004572774722, 0.5977502100878569, 'Libraries & Demo'),
  Text(1.1318697663597663, 0.6791692219179901, 'Lifestyle'),
  Text(1.0832947682531107, 0.7542363323756286, 'Lifestyle;Pretend Play'),
  Text(1.0331487018975918, 0.8215861243760884, 'Adventure;Action & Adventure'),
  Text(0.9833774775378126, 0.8805502465341594, 'Arcade'),
  Text(0.9344235738500605, 0.9323371625293505, 'Casual'),
  Text(0.8871250462409052, 0.9774503324118684, 'Card'),
  Text(0.8401047653062722, 1.0181473288820697, 'Casual;Pretend Play'),
  Text(0.7955735764159977, 1.0533103457712067, 'Strategy'),
  Text(0.7536773274825664, 1.0836837573936118, 'Action'),
  Text(0.7113877263509573, 1.1119026498741764, 'Puzzle'),
  Text(0.6707816006281347, 1.1368606089836861, 'Sports'),
  Text(0.6324909034691589, 1.1586005597395364, 'Word'),
  Text(0.5967380444779492, 1.1774139910297623, 'Racing'),
  Text(0.5628910610481059, 1.1939655159979026, 'Casual;Creativity'),
  Text(0.5290014176877568, 1.2093624353701184, 'Simulation'),
  Text(0.4946911073683413, 1.2237976582305934, 'Adventure'),
  Text(0.4616889881483361, 1.2366257632050877, 'Board'),
  Text(0.4305009885871955, 1.2478256684430913, 'Trivia'),
  Text(0.39990701171551396, 1.257964380251193, 'Role Playing'),
  Text(0.3699462352891115, 1.2670989633787146, 'Simulation;Education'),
  Text(0.3419699532874233, 1.2749339398763364, 'Action;Action & Adventure'),
  Text(0.31779497368306214, 1.281173819082244, 'Casual;Brain Games'),
  Text(0.29704592989453105, 1.2861429607680064, 'Simulation;Action & Adventure'),
  Text(0.2788821335912729, 1.290203377597338, 'Educational;Creativity'),
  Text(0.2624427184455307, 1.2936474865800651, 'Puzzle;Brain Games'),
  Text(0.24729870108693794, 1.2966276845882605, 'Educational;Education'),
  Text(0.2330145852830927, 1.2992706427243512, 'Card;Brain Games'),
  Text(0.21870224919943723, 1.3017562468431283, 'Educational;Brain Games'),
  Text(0.20481189800808003, 1.3040138367495675, 'Educational;Pretend Play'),
  Text(0.19179659325595572, 1.305991602888552, 'Casual;Action & Adventure'),
  Text(0.17921197603734737, 1.3077779122025228, 'Entertainment;Education'),
  Text(0.16661074144467453, 1.3094429582212643, 'Casual;Education'),
  Text(0.15444490649593504, 1.3109335493675727, 'Music;Music & Video'),
  Text(0.14271706011611968, 1.3122621082511725, 'Arcade;Pretend Play'),
  Text(0.1318812070602608, 1.313395350693891, 'Simulation;Pretend Play'),
  Text(0.12374838103041277, 1.314186569019921, 'Puzzle;Creativity'),
  Text(0.11741955512180141, 1.3147671459520878, 'Racing;Action & Adventure'),
  Text(0.1115403479693112, 1.3152789631005604, 'Educational;Action & Adventure'),
  Text(0.10611140617886738, 1.3157280758115422, 'Arcade;Action & Adventure'),
  Text(0.1006806571869905, 1.316154780133551, 'Entertainment;Action & Adventure'),
  Text(0.0952481934856928, 1.3165590687993123, 'Puzzle;Action & Adventure'),
  Text(0.09026700770708287, 1.3169099693295703, 'Role Playing;Action & Adventure'),
  Text(0.08573753405009335, 1.3172126158122726, 'Strategy;Action & Adventure'),
  Text(0.08120704635749044, 1.3174996833479287, 'Music & Audio;Music & Video'),
  Text(0.07667559821229142, 1.317771168541332, 'Health & Fitness;Education'),
  Text(0.07214324320887652, 1.31802706818157, 'Adventure;Education'),
  Text(0.06761003495234803, 1.318267379242065, 'Board;Brain Games'),
  Text(0.06352946232692995, 1.3184703286067727, 'Board;Action & Adventure'),
  Text(0.06035526089896428, 1.3186194456634628, 'Board;Pretend Play'),
  Text(0.057634237759571674, 1.318741178032245, 'Casual;Music & Video'),
  Text(0.05491296922528777, 1.3188572954686428, 'Education;Music & Video'),
  Text(0.05219146688272186, 1.3189677974782514, 'Role Playing;Pretend Play'),
  Text(0.049469742319478795, 1.3190726835905764, 'Entertainment;Pretend Play'),
  Text(0.046747807124110724, 1.3191719533590331, 'Video Players & Editors;Creativity'),
  Text(0.044025672886063134, 1.3192656063609518, 'Card;Action & Adventure'),
  Text(0.04175708398289599, 1.319339359655902, 'Medical'),
  Text(0.039942123617642517, 1.3193955535626583, 'Social'),
  Text(0.03812708766772364, 1.3194492507049969, 'Shopping'),
  Text(0.036311979567828875, 1.3195004509813044, 'Photography'),
  Text(0.0344968027527878, 1.3195491542946916, 'Travel & Local'),
  Text(0.03268156065755884, 1.3195953605529942, 'Travel & Local;Action & Adventure'),
  Text(0.030866256717223966, 1.3196390696687736, 'Tools'),
  Text(0.029050894366983354, 1.3196802815593172, 'Personalization'),
  Text(0.027235477042144185, 1.319718996146637, 'Productivity'),
  Text(0.025420008178121192, 1.3197552133574713, 'Parenting'),
  Text(0.023604491210425477, 1.3197889331232844, 'Parenting;Education'),
  Text(0.021788929574660325, 1.3198201553802664, 'Parenting;Brain Games'),
  Text(0.019973326706510044, 1.319848880069334, 'Parenting;Music & Video'),
  Text(0.018611599563469805, 1.3198687845243138, 'Weather'),
  Text(0.017703770372461913, 1.3198812736434287, 'Video Players & Editors'),
  Text(0.016795932805999628, 1.3198931383415766, 'News & Magazines'),
  Text(0.015888087293570677, 1.3199043786131448, 'Maps & Navigation'),
  Text(0.01498023426466655, 1.3199149944528155, 'Health & Fitness;Action & Adventure'),
  Text(0.01407237414878229, 1.3199249858555664, 'Music'),
  Text(0.01316450737541747, 1.3199343528166705, 'Educational'),
  Text(0.012256634374071285, 1.3199430953316966, 'Casino'),
  Text(0.011348755574249407, 1.319951213396509, 'Adventure;Brain Games'),
  Text(0.010440871405459072, 1.3199587070072665, 'Video Players & Editors;Music & Video'),
  Text(0.009532982297210049, 1.3199655761604243, 'Trivia;Education'),
  Text(0.008625088679014457, 1.319971820852733, 'Entertainment;Creativity'),
  Text(0.007717190980386544, 1.3199774410812377, 'Sports;Action & Adventure'),
  Text(0.006809289630842483, 1.319982436843284, 'Books & Reference;Creativity'),
  Text(0.005901385059901352, 1.319986808136496, 'Books & Reference;Education'),
  Text(0.004993477697080235, 1.3199905549588182, 'Puzzle;Education'),
  Text(0.004085567971901053, 1.3199936773084737, 'Role Playing;Education'),
  Text(0.0031776563138856723, 1.3199961751839855, 'Role Playing;Brain Games'),
  Text(0.0022697431525568745, 1.3199980485841718, 'Strategy;Education'),
  Text(0.0013618289174381522, 1.3199992975081463, 'Racing;Pretend Play'),
  Text(0.0004539140380535046, 1.3199999219553182, 'Strategy;Creativity')],
 [Text(-0.18078297913302938, 0.6969343688295093, '8.1%'),
  Text(-0.4595745939569526, 0.5542483131136279, '5.9%'),
  Text(-0.6235383019600064, 0.3599999805400438, '5.4%'),
  Text(-0.705004002645937, 0.1461826126911392, '4.7%'),
  Text(-0.717579423902363, -0.05898957867244719, '4.4%'),
  Text(-0.67846390230707, -0.2410118944497606, '3.9%'),
  Text(-0.6017099083794395, -0.3954050912140693, '3.8%'),
  Text(-0.4953233311295747, -0.5225464550053914, '3.6%'),
  Text(-0.3667335601068331, -0.6196018849966224, '3.6%'),
  Text(-0.2244929561164443, -0.6841073838617006, '3.4%'),
  Text(-0.0771071426572873, -0.7158592658834757, '3.3%'),
  Text(0.0633071461706068, -0.7172114090306522, '2.9%'),
  Text(0.19202388256388273, -0.6939213417420538, '2.9%'),
  Text(0.31195005351686295, -0.6489122930803718, '2.8%'),
  Text(0.4123155894027591, -0.5902506711012324, '2.4%'),
  Text(0.494424097908138, -0.5233973742843231, '2.3%'),
  Text(0.564766497840935, -0.4465857173225372, '2.3%'),
  Text(0.6219225918125032, -0.3627840815047132, '2.2%'),
  Text(0.6656070837734056, -0.2745308908495773, '2.2%'),
  Text(0.6962770218778608, -0.18329841462734184, '2.1%'),
  Text(0.7143194840423039, -0.09026447095915764, '2.1%'),
  Text(0.719993347082258, 0.0030951829165645284, '2.0%'),
  Text(0.7142883927139968, 0.0905101764115781, '1.8%'),
  Text(0.7013369790617344, 0.16286940105667566, '1.4%'),
  Text(0.6843001259660215, 0.22390475118426365, '1.4%'),
  Text(0.6640863842864605, 0.2781892776606163, '1.2%'),
  Text(0.6419457039695302, 0.32604556913883104, '1.1%'),
  Text(0.6173835089235088, 0.3704559392279946, '1.1%'),
  Text(0.5908880554107876, 0.41140163584125194, '1.1%'),
  Text(0.5635356555805046, 0.4481378860233209, '1.0%'),
  Text(0.536387715020625, 0.4803001344731778, '0.9%'),
  Text(0.5096855857363967, 0.5085475431978275, '0.8%'),
  Text(0.4838863888586755, 0.53315472677011, '0.8%'),
  Text(0.45823896289433025, 0.5553530884811289, '0.7%'),
  Text(0.43394922349963505, 0.5745329158752036, '0.6%'),
  Text(0.41109672408139986, 0.5911002313056064, '0.6%'),
  Text(0.38802966891870394, 0.6064923544768235, '0.6%'),
  Text(0.36588087306989164, 0.6201057867183742, '0.5%'),
  Text(0.34499503825590483, 0.6319639416761107, '0.5%'),
  Text(0.3254934788061541, 0.6422258132889612, '0.5%'),
  Text(0.3070314878444214, 0.6512539178170377, '0.4%'),
  Text(0.2885462278296855, 0.65965223747461, '0.4%'),
  Text(0.2698315131100043, 0.6675259953985054, '0.4%'),
  Text(0.2518303571718197, 0.6745231435664114, '0.4%'),
  Text(0.23481872104756113, 0.6806321827871407, '0.4%'),
  Text(0.21813109729937125, 0.6861623892279234, '0.4%'),
  Text(0.2017888556122426, 0.6911448891156625, '0.4%'),
  Text(0.18652906542950362, 0.6954185126598198, '0.3%'),
  Text(0.17334271291803388, 0.6988220831357693, '0.3%'),
  Text(0.16202505266974418, 0.7015325240552761, '0.2%'),
  Text(0.15211752741342158, 0.7037472968712752, '0.2%'),
  Text(0.14315057369756218, 0.7056259017709446, '0.2%'),
  Text(0.13489020059287524, 0.7072514643208693, '0.2%'),
  Text(0.12709886469986872, 0.708693077849646, '0.2%'),
  Text(0.11929213592696575, 0.7100488619144335, '0.2%'),
  Text(0.11171558073168002, 0.7112802745906731, '0.2%'),
  Text(0.10461632359415765, 0.7123590561210283, '0.2%'),
  Text(0.09775198692946219, 0.7133334066559215, '0.2%'),
  Text(0.09087858624254974, 0.7142416135752351, '0.2%'),
  Text(0.08424267627051002, 0.7150546632914032, '0.1%'),
  Text(0.07784566915424708, 0.7157793317733667, '0.1%'),
  Text(0.07193520385105134, 0.7163974640148495, '0.1%'),
  Text(0.06749911692567968, 0.7168290376472296, '0.1%'),
  Text(0.06404703006643712, 0.717145715973866, '0.1%'),
  Text(0.060840189801442464, 0.717424888963942, '0.1%'),
  Text(0.057878948824836754, 0.7176698595335684, '0.1%'),
  Text(0.05491672210199481, 0.7179026073455732, '0.1%'),
  Text(0.05195356008310516, 0.7181231284359885, '0.1%'),
  Text(0.04923654965840884, 0.7183145287252201, '0.1%'),
  Text(0.04676592766368728, 0.7184796086248758, '0.1%'),
  Text(0.04429475255863115, 0.718636190917052, '0.1%'),
  Text(0.041823053570340774, 0.7187842737498173, '0.1%'),
  Text(0.039350859932114464, 0.7189238553717654, '0.1%'),
  Text(0.03687820088309893, 0.7190549341320354, '0.1%'),
  Text(0.03465243399650724, 0.7191656337855123, '0.0%'),
  Text(0.032921051399435054, 0.7192469703618888, '0.0%'),
  Text(0.03143685695976636, 0.71931336983577, '0.0%'),
  Text(0.02995252866833878, 0.7193767066192597, '0.0%'),
  Text(0.028468072845121012, 0.7194369804426826, '0.0%'),
  Text(0.026983495810624793, 0.7194941910494052, '0.0%'),
  Text(0.025498803885878576, 0.7195483381958362, '0.0%'),
  Text(0.02401400339239807, 0.7195994216514282, '0.0%'),
  Text(0.02277659126339781, 0.719639650721401, '0.0%'),
  Text(0.021786612882350464, 0.7196703019432681, '0.0%'),
  Text(0.020796593273303802, 0.7196995912936346, '0.0%'),
  Text(0.019806534309724837, 0.7197275187170751, '0.0%'),
  Text(0.01881643786515698, 0.7197540841607407, '0.0%'),
  Text(0.01782630581321391, 0.7197792875743604, '0.0%'),
  Text(0.016836140027576708, 0.7198031289102401, '0.0%'),
  Text(0.01584594238199092, 0.7198256081232639, '0.0%'),
  Text(0.014855714750260464, 0.7198467251708928, '0.0%'),
  Text(0.013865459006247922, 0.7198664800131661, '0.0%'),
  Text(0.01287517702386844, 0.7198848726127005, '0.0%'),
  Text(0.011884870677087449, 0.7199019029346907, '0.0%'),
  Text(0.010894541839914569, 0.7199175709469094, '0.0%'),
  Text(0.010151781580074438, 0.7199284279223529, '0.0%'),
  Text(0.00965660202134286, 0.7199352401691428, '0.0%'),
  Text(0.009161417894181614, 0.7199417118226782, '0.0%'),
  Text(0.008666229432856732, 0.7199478428798971, '0.0%'),
  Text(0.0081710368716363, 0.7199536333378993, '0.0%'),
  Text(0.00767584044479034, 0.7199590831939452, '0.0%'),
  Text(0.007180640386591346, 0.7199641924454566, '0.0%'),
  Text(0.00668543693131161, 0.7199689610900163, '0.0%'),
  Text(0.006190230313226949, 0.7199733891253685, '0.0%'),
  Text(0.005695020766614038, 0.719977476549418, '0.0%'),
  Text(0.005199808525750935, 0.7199812233602314, '0.0%'),
  Text(0.004704593824916977, 0.719984629556036, '0.0%'),
  Text(0.0042093768983926605, 0.7199876951352205, '0.0%'),
  Text(0.003714157980459536, 0.7199904200963344, '0.0%'),
  Text(0.0032189373054007375, 0.7199928044380887, '0.0%'),
  Text(0.0027237151074983095, 0.7199948481593553, '0.0%'),
  Text(0.0022284916210369376, 0.7199965512591675, '0.0%'),
  Text(0.0017332670803012758, 0.7199979137367193, '0.0%'),
  Text(0.001238041719576477, 0.7199989355913663, '0.0%'),
  Text(0.000742815773148083, 0.7199996168226251, '0.0%'),
  Text(0.0002475894753019116, 0.7199999574301735, '0.0%')])
```



Observations We found the following points in the dataset. 1.There is poor correlation between the dependent variable and the other independent variables. 2.None of the numerical continuous variables are normally distributed. 3.The rating is same in all Categories/Genres/Content Rating of apps. 4.Almost 90% apps on the playstore are free and users dont have to pay for it. 5. Lifestyle apps have the largest number of installs. 6. Art and Design category has the largest number of apps in the Playstore. 7. Sports Apps were the heaviest in size and Family, Tools apps were lightweight. 8.Most of the Apps are light weight between 5 MB to 30MB. 9. Less than 10% of the apps are popular that is having more than 1 million installs. 10. The Ratings differ according to app size, popularity, Type, Content Rating and Popularity. 11. The number of installs is having a moderate positive correlation with the number of reviews and the app size.

Step 4: Model building

A linear regression model is going to be built in order to predict the rating of the app which is the dependent variable. The categorical variables will be converted into numerical using dummy variables method. The algorithm will predict the rating of an application.

In [123…
```
appdf.head(0)
```

Out[123]:

App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Installs-bin	Size-bin

In [136…
```
#initalizing encoder
label_encoder=preproc.LabelEncoder()
```

In [145…
```
#encoding categorical variables
appenc=label_encoder.fit_transform(appdf['App'])
catenc=label_encoder.fit_transform(appdf['Category'])
typenc=label_encoder.fit_transform(appdf['Type'])
corenc=label_encoder.fit_transform(appdf['Content Rating'])
genenc=label_encoder.fit_transform(appdf['Genres'])
inbenc=label_encoder.fit_transform(appdf['Installs-bin'])
szbenc=label_encoder.fit_transform(appdf['Size-bin'])
```

In [147…
```
#setting values of x and y
X=pd.DataFrame([appenc,
                catenc,
                typenc,
                corenc,
                genenc,
                inbenc,
                szbenc,
                appdf['Reviews'],
                appdf['Size'],
                appdf['Installs'],
                appdf['Price']]).T
y= appdf['Rating']
```

In [148…
```
# Create linear regression object
regressor = LinearRegression()
```

In [211…
```
# Train, test, split

X_train, X_test, y_train, y_test = train_test_split(X,y, test_size = .30, random_state= 1000000000)
```

In [212…
```
# Fit model to training data
regressor.fit(X_train,y_train)
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=-2, normalize=False)
```

Out[212]:
```
LinearRegression(n_jobs=-2, normalize=False)
```

In [213…
```
# Predict
# Predicting test set results
y_pred = regressor.predict(X_test)
y_pred
```

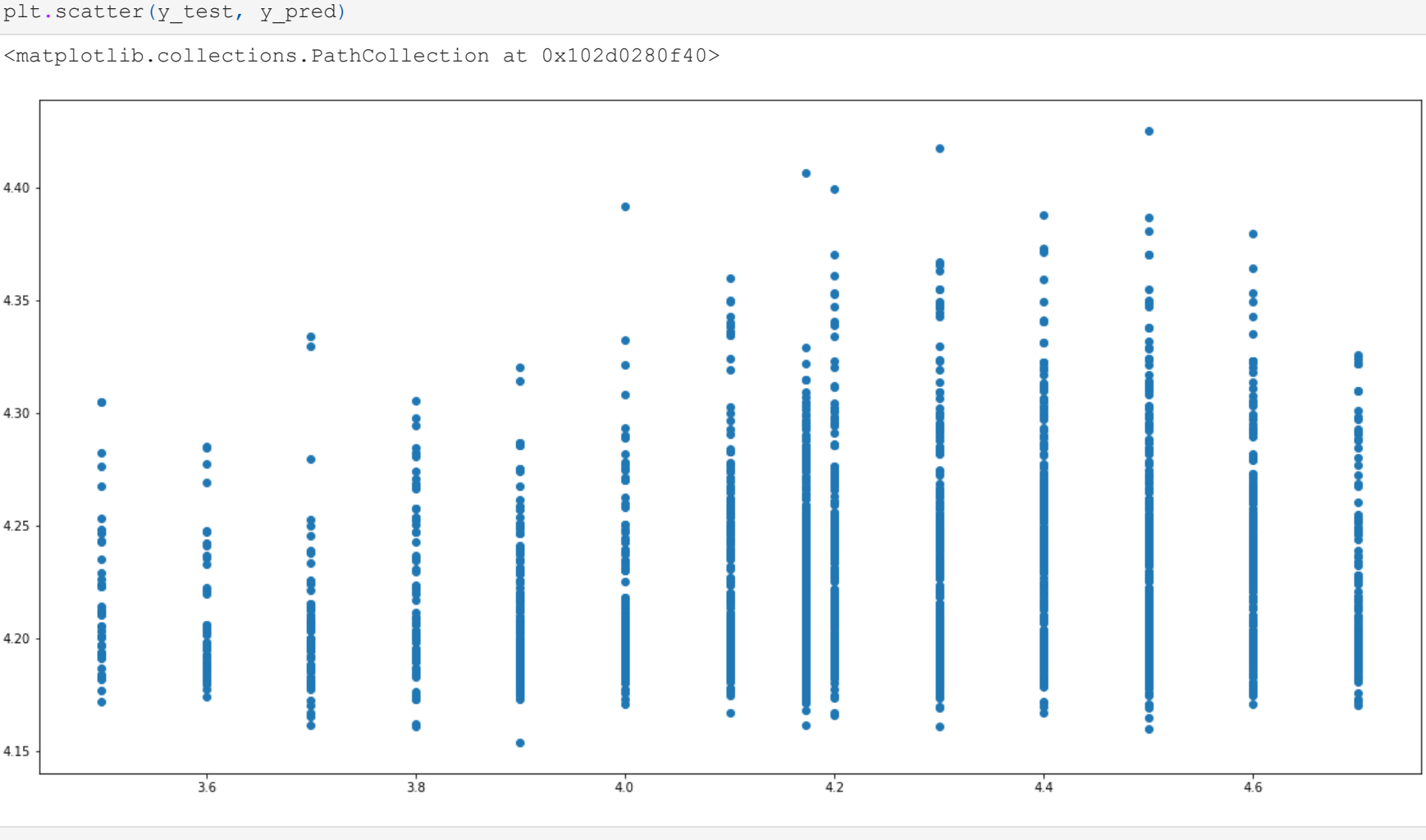
Out[213]:
```
array([4.21087461, 4.20586234, 4.22774308, ..., 4.22465065, 4.34727497,
       4.25132248])
```

In [214…
```
# Calculated R Squared
print('R^2 =',metrics.explained_variance_score(y_test,y_pred))
```

```
R^2 = 0.03343747472155478
```

In [215…
```
# Actual v predictions scatter
plt.scatter(y_test, y_pred)
```

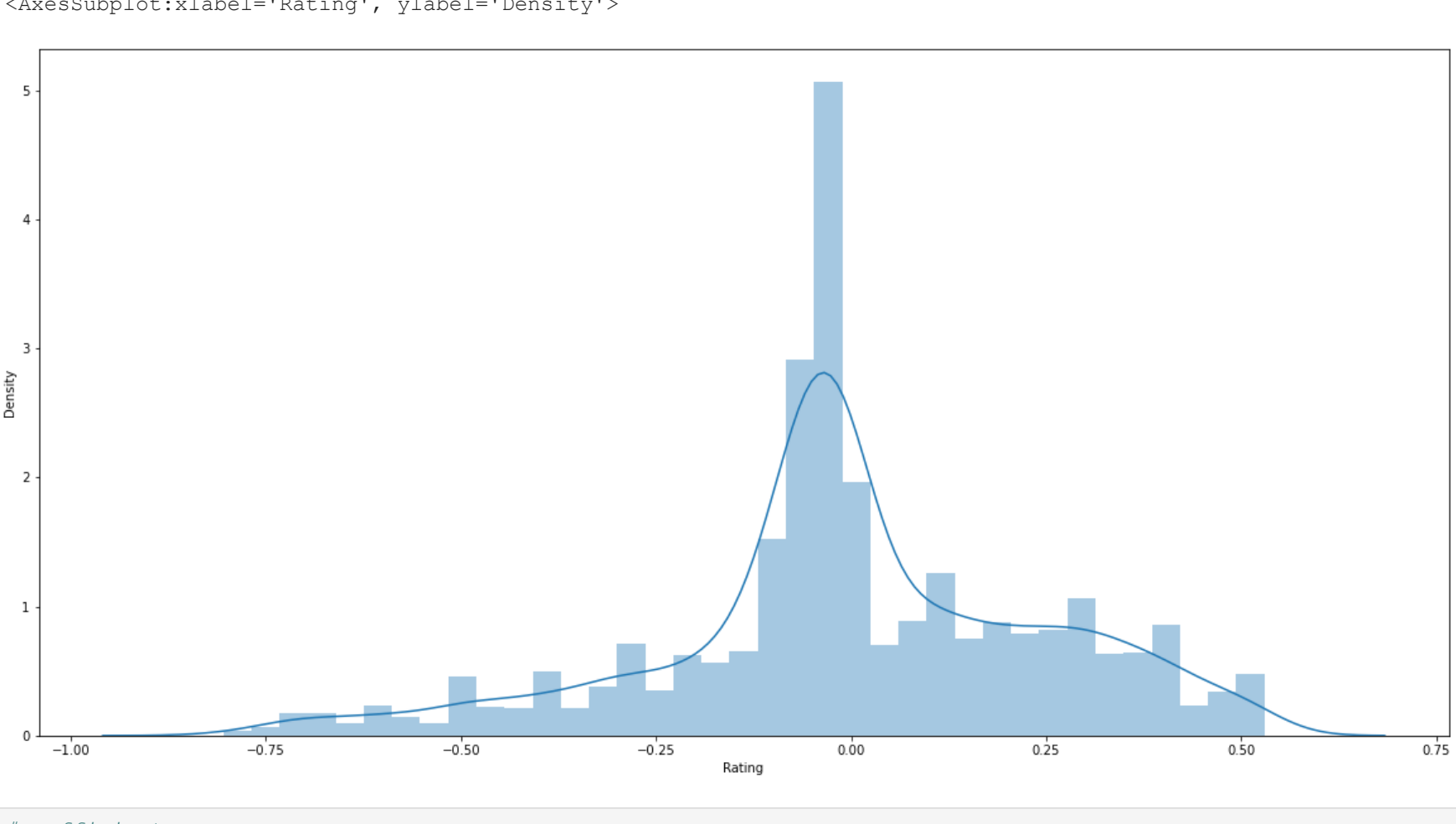
Out[215]:
```
<matplotlib.collections.PathCollection at 0x102d0280f40>
```



In [216…
```
# Histogram of the distribution of residuals
sb.distplot((y_test - y_pred))
```

```
C:\Users\sujoydutta\anaconda3\lib\site-packages\seaborn\distributions.py:2619: 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)
```

Out[216]:
```
<AxesSubplot:xlabel='Rating', ylabel='Density'>
```



In [217…
```
#coefficients
cdf = pd.DataFrame(data = regressor.coef_, index = X.columns, columns = ['Coefficients'])
cdf
```

Out[217]:

	Coefficients
0	3.178512e-06
1	5.860385e-04
2	2.793450e-03
3	-5.747790e-03
4	-3.325460e-04
5	2.891072e-03
6	6.238688e-03
7	4.439115e-07
8	7.053739e-07
9	3.805939e-09
10	1.165765e-02

Conclusion

The model is not very good as it can explain only 3% of the change in the dependent variable. The fit is not achieved. The variables are not enough and maybe another methodology needs to be used to develop this model.

In [ ]: