NAME : BHAVIK RANSUBHE

CLASS : TE(B) COMP

ROLL NO : 39055

Course Name – Data Science Honours

PROBLEM STATEMNET

Title: - Case Study on open source visualization tool.

1. Download the dataset from standard repositories like UCM, kaggle etc.
2. Import the dataset.
3. Apply preprocessing and visualization techniques to explore the dataset.

Google PlayStore Dataset Analysis (From Kaggle):

IMPORTING PACKAGES :

In [1]:
```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import matplotlib.pyplot as plt #plotting charts
import seaborn as sns #plotting good-looking charts
#To hide Warning messages.
import warnings
warnings.filterwarnings('ignore')
```

READING DATA :

In [2]:
```
df = pd.read_csv("C:\\Users\\bhavi\\Downloads\\archive\\googleplaystore.csv")
df.head(10)
```

Out[2]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Cur
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
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14M	500,000+	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2
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
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25M	50,000,000+	Free	0	Teen	Art & Design	June 8, 2018	Va de
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	
5	Paper flowers instructions	ART_AND_DESIGN	4.4	167	5.6M	50,000+	Free	0	Everyone	Art & Design	March 26, 2017	
6	Smoke Effect Photo Maker - Smoke Editor	ART_AND_DESIGN	3.8	178	19M	50,000+	Free	0	Everyone	Art & Design	April 26, 2018	
7	Infinite Painter	ART_AND_DESIGN	4.1	36815	29M	1,000,000+	Free	0	Everyone	Art & Design	June 14, 2018	6.1.
8	Garden Coloring Book	ART_AND_DESIGN	4.4	13791	33M	1,000,000+	Free	0	Everyone	Art & Design	September 20, 2017	2
9	Kids Paint Free - Drawing Fun	ART_AND_DESIGN	4.7	121	3.1M	10,000+	Free	0	Everyone	Art & Design;Creativity	July 3, 2018	

In [3]:
```
df.tail(10)
```

Out[3]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Ge
10831	payermonstationnement.fr	MAPS_AND_NAVIGATION	NaN	38	9.8M	5,000+	Free	0	Everyone	Ma Naviga
10832	FR Tides	WEATHER	3.8	1195	582k	100,000+	Free	0	Everyone	Wea
10833	Chemin (fr)	BOOKS_AND_REFERENCE	4.8	44	619k	1,000+	Free	0	Everyone	Boo Refere
10834	FR Calculator	FAMILY	4.0	7	2.6M	500+	Free	0	Everyone	Educa
10835	FR Forms	BUSINESS	NaN	0	9.6M	10+	Free	0	Everyone	Busi
10836	Sya9a Maroc - FR	FAMILY	4.5	38	53M	5,000+	Free	0	Everyone	Educa
10837	Fr. Mike Schmitz Audio Teachings	FAMILY	5.0	4	3.6M	100+	Free	0	Everyone	Educa
10838	Parkinson Exercices FR	MEDICAL	NaN	3	9.5M	1,000+	Free	0	Everyone	Me
10839	The SCP Foundation DB fr nn5n	BOOKS_AND_REFERENCE	4.5	114	Varies with device	1,000+	Free	0	Mature 17+	Boo Refere
10840	iHoroscope - 2018 Daily Horoscope & Astrology	LIFESTYLE	4.5	398307	19M	10,000,000+	Free	0	Everyone	Lifes

Data Cleaning & Preprocessing :

In [4]:
```
#drop off irrelevant columns
df = df.drop(columns=['Last Updated','Current Ver','Content Rating'])
```

In [5]:
```
#check out the type and counts for each attribute
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10841 entries, 0 to 10840
Data columns (total 10 columns):
App            10841 non-null object
Category       10841 non-null object
Rating         9367 non-null float64
Reviews        10841 non-null object
Size           10841 non-null object
Installs       10841 non-null object
Type           10840 non-null object
Price          10841 non-null object
Genres         10841 non-null object
Android Ver    10838 non-null object
dtypes: float64(1), object(9)
memory usage: 847.1+ KB
```

In [6]:
```
print(df.isnull().sum())
df.dropna(inplace=True) #Dropping Rows with Null values
```

```
App               0
Category          0
Rating         1474
Reviews           0
Size              0
Installs          0
Type              1
Price             0
Genres            0
Android Ver       3
dtype: int64
```

In [7]:
```
#Removing Duplicate entries
df.drop_duplicates(inplace=True)
```

In [8]:
```
#fill the missing values of Rating by the mean value of their corresponding app categories
df['Rating'] = df.groupby("Category").Rating.transform(lambda x: x.fillna(x.mean()))
```

In [9]:
```
#Double check if any nan value still exist
df.Rating.isna().sum()
```

Out[9]: 0

In [10]:
```
print(df.isnull().sum())
```

```
App            0
Category       0
Rating         0
Reviews        0
Size           0
Installs       0
Type           0
Price          0
Genres         0
Android Ver    0
dtype: int64
```

In [11]:
```
df.shape
```

Out[11]: (8888, 10)

The data types of each feature must be changed to a proper format that can be used for analysis.

In [12]:
```
df['Installs']=df['Installs'].str.replace(',','').str.replace('+','').astype('int')
```

In [13]:
```
# converting review to int
df['Reviews']=df['Reviews'].astype('int')
```

In [14]:
```
df['Price']=df['Price'].str.replace('$','').astype('float')
```

In [15]:
```
# Changing the feature, Android Ver
newVer = []

for row in df['Android Ver']:
    try:
        newrow = float(row[:2])
    except:
        newrow = 0  # When the value is - Varies with device

    newVer.append(newrow)

df['Android Ver'] =  newVer

df['Android Ver'].value_counts()
```

Out[15]:
```
4.0    5604
0.0    1178
2.0    1160
5.0     499
3.0     246
1.0     105
6.0      46
7.0      45
8.0       5
Name: Android Ver, dtype: int64
```

In [16]:
```
df.head(10)
```

Out[16]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Genres	Android Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19M	10000	Free	0.0	Art & Design	4.0
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14M	500000	Free	0.0	Art & Design;Pretend Play	4.0
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7M	5000000	Free	0.0	Art & Design	4.0
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25M	50000000	Free	0.0	Art & Design	4.0
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8M	100000	Free	0.0	Art & Design;Creativity	4.0
5	Paper flowers instructions	ART_AND_DESIGN	4.4	167	5.6M	50000	Free	0.0	Art & Design	2.0
6	Smoke Effect Photo Maker - Smoke Editor	ART_AND_DESIGN	3.8	178	19M	50000	Free	0.0	Art & Design	4.0
7	Infinite Painter	ART_AND_DESIGN	4.1	36815	29M	1000000	Free	0.0	Art & Design	4.0
8	Garden Coloring Book	ART_AND_DESIGN	4.4	13791	33M	1000000	Free	0.0	Art & Design	3.0
9	Kids Paint Free - Drawing Fun	ART_AND_DESIGN	4.7	121	3.1M	10000	Free	0.0	Art & Design;Creativity	4.0

DATA VISUALIZATION

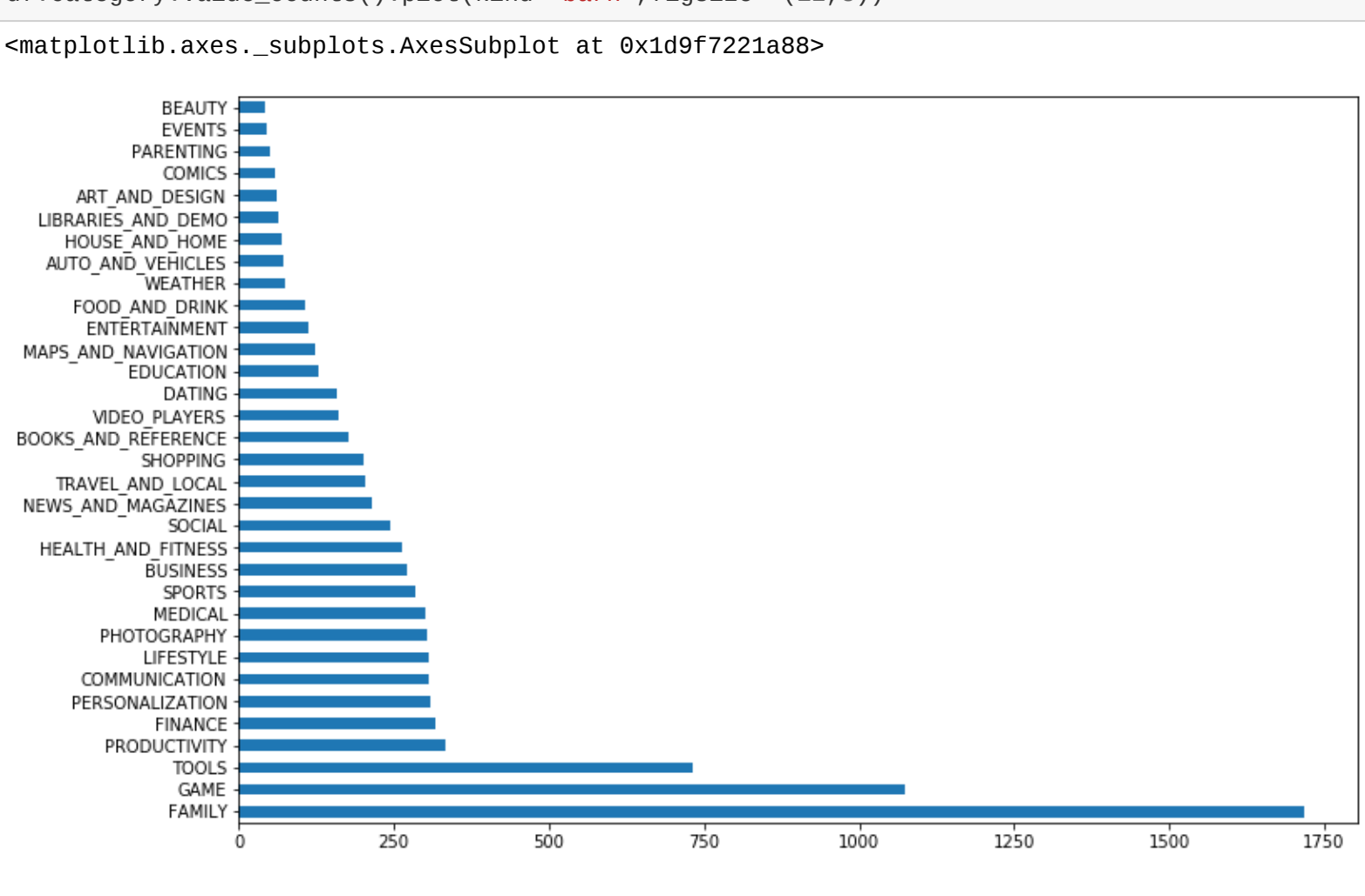
1. Displaying all the categories and their counts

In [17]:
```
df.Category.value_counts()
```

Out[17]:
```
FAMILY                 1718
GAME                   1074
TOOLS                   733
PRODUCTIVITY            334
FINANCE                 317
PERSONALIZATION         308
COMMUNICATION           307
LIFESTYLE               305
PHOTOGRAPHY             304
MEDICAL                 302
SPORTS                  286
BUSINESS                270
HEALTH_AND_FITNESS      262
SOCIAL                  244
NEWS_AND_MAGAZINES      214
TRAVEL_AND_LOCAL        205
SHOPPING                201
BOOKS_AND_REFERENCE     177
VIDEO_PLAYERS           160
DATING                  159
EDUCATION               129
MAPS_AND_NAVIGATION     124
ENTERTAINMENT           111
FOOD_AND_DRINK          106
WEATHER                  75
AUTO_AND_VEHICLES        73
HOUSE_AND_HOME           68
LIBRARIES_AND_DEMO       65
ART_AND_DESIGN           62
COMICS                   58
PARENTING                50
EVENTS                   45
BEAUTY                   42
Name: Category, dtype: int64
```

In [18]:
```
df.Category.value_counts().plot(kind='barh',figsize= (12,8))
```

Out[18]: <matplotlib.axes._subplots.AxesSubplot at 0x1d9f7221a88>



Insight : Maximum Number of Apps belong to the Family and Game Category.

2. Rating

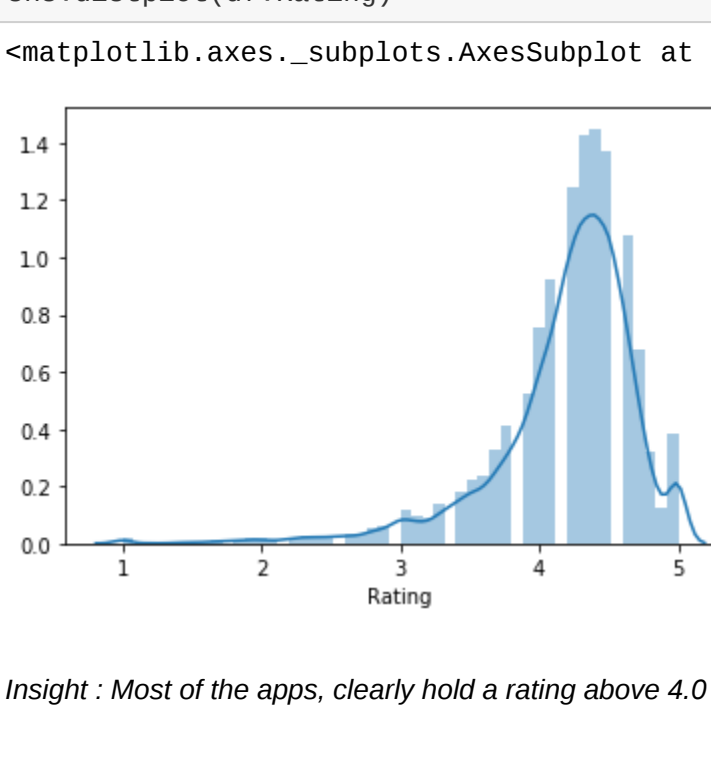
In [19]:
```
df.Rating.describe()
```

Out[19]:
```
count    8888.000000
mean        4.187826
std         0.522478
min         1.000000
25%         4.000000
50%         4.300000
75%         4.500000
max         5.000000
Name: Rating, dtype: float64
```

Distribution Plot of 'Rating'

In [20]:
```
sns.distplot(df.Rating)
```

Out[20]: <matplotlib.axes._subplots.AxesSubplot at 0x1d9f7848dc8>

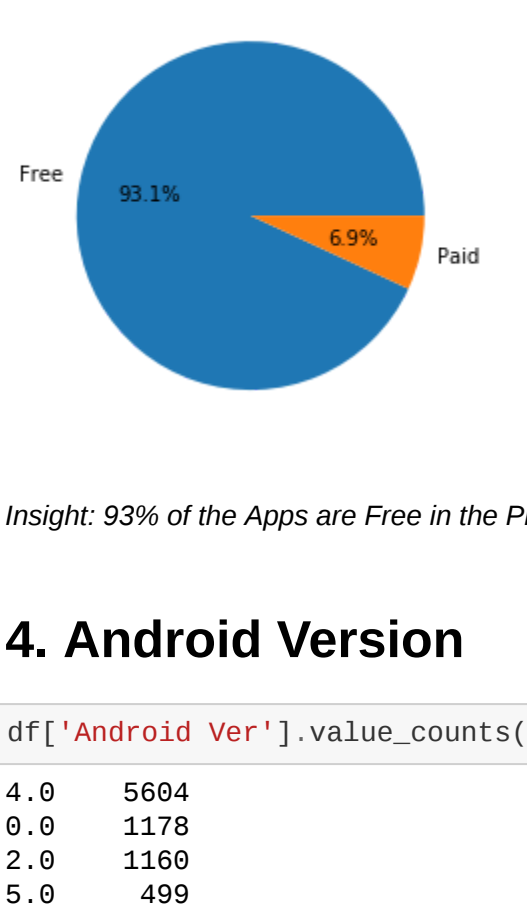


Insight : Most of the apps, clearly hold a rating above 4.0 !

3. TYPE

In [21]:
```
plt.pie(df.Type.value_counts(), labels=['Free', 'Paid'], autopct='%1.1f%%')
```

Out[21]:
```
([<matplotlib.patches.Wedge at 0x1d9f7748248>,
  <matplotlib.patches.Wedge at 0x1d9f7748d48>],
 [Text(-1.0743632425327427, 0.23610087482796627, 'Free'),
  Text(1.074363237006403, -0.23610089997525263, 'Paid')],
 [Text(-0.5860163141087688, 0.12878229536070884, '93.1%'),
  Text(0.5860163110944014, -0.1287823090774105, '6.9%')])
```



Insight: 93% of the Apps are Free in the Play Store

4. Android Version

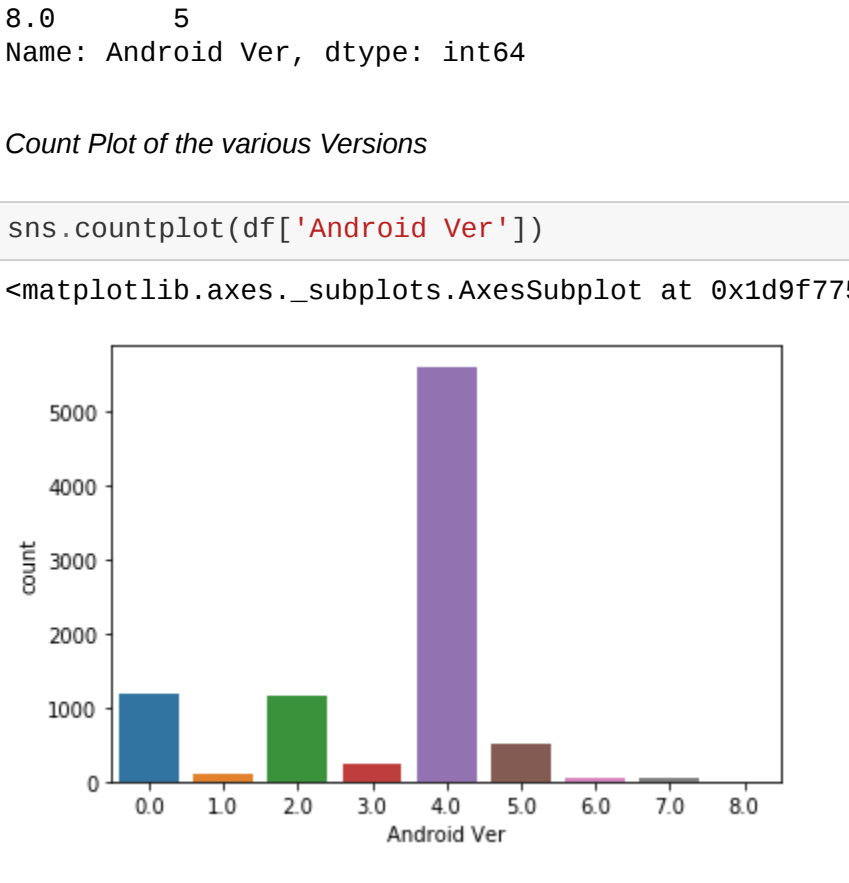
In [22]:
```
df['Android Ver'].value_counts()
```

Out[22]:
```
4.0    5604
0.0    1178
2.0    1160
5.0     499
3.0     246
1.0     105
6.0      46
7.0      45
8.0       5
Name: Android Ver, dtype: int64
```

Count Plot of the various Versions

In [23]:
```
sns.countplot(df['Android Ver'])
```

Out[23]: <matplotlib.axes._subplots.AxesSubplot at 0x1d9f7758988>



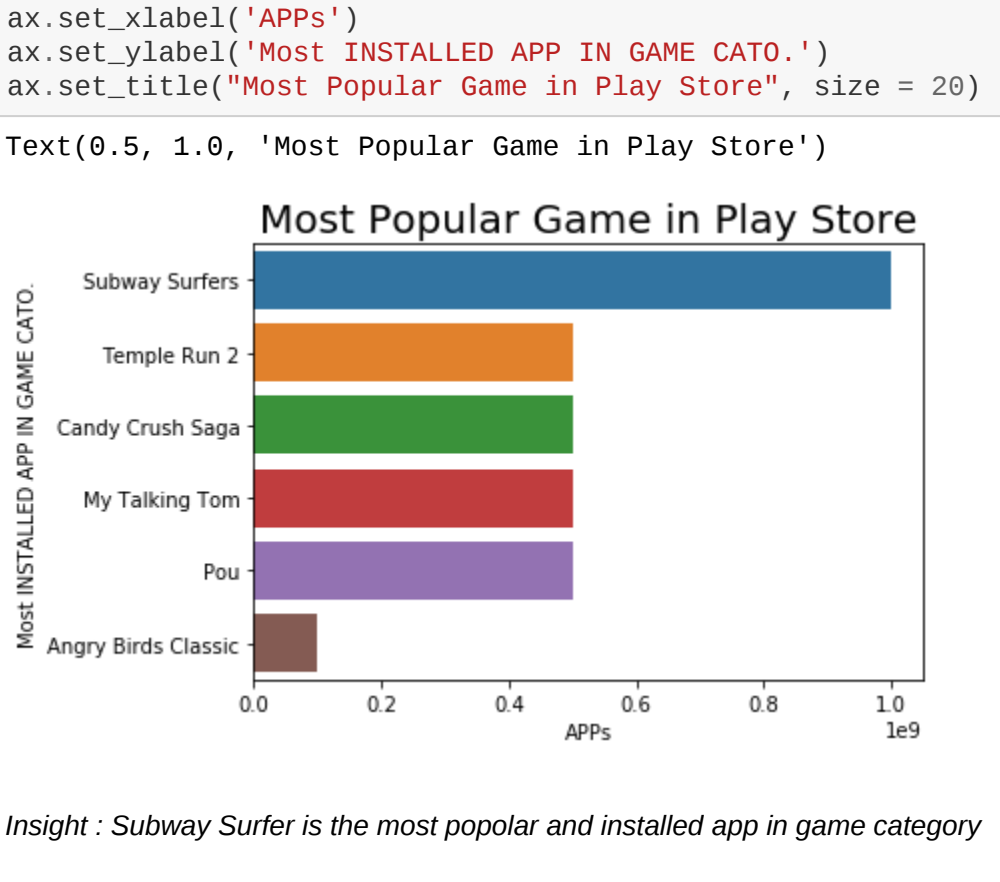
Insight : Most of the apps support Android 4.0 and above

5. Most Popular Game in Play Store

In [24]:
```
data_cat=df[df['Category']=='GAME'].sort_values(['Installs'],ascending=0)[:20]

ax = sns.barplot(x = 'Installs' , y = 'App' , data = data_cat )
ax.set_xlabel('APPs')
ax.set_ylabel('Most INSTALLED APP IN GAME CATO.')
ax.set_title("Most Popular Game in Play Store", size = 20)
```

Out[24]: Text(0.5, 1.0, 'Most Popular Game in Play Store')



Insight : Subway Surfer is the most popolar and installed app in game category

6. Furthur Analysis

Looking at the Apps with 5.0 ratings:

In [25]:
```
df_full = df[df.Rating == 5]
df_full.head()
```

Out[25]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Genres	Android Ver
329	Hojiboy Tojiboyev Life Hacks	COMICS	5.0	15	37M	1000	Free	0.0	Comics	4.0
612	American Girls Mobile Numbers	DATING	5.0	5	4.4M	1000	Free	0.0	Dating	4.0
615	Awake Dating	DATING	5.0	2	70M	100	Free	0.0	Dating	4.0
633	Spine- The dating app	DATING	5.0	5	9.3M	500	Free	0.0	Dating	4.0
636	Girls Live Talk - Free Text and Video Chat	DATING	5.0	6	5.0M	100	Free	0.0	Dating	4.0

Insight : There are many Apps that have full ratings but less downloads/installs. So we can't really consider those apps as the best ones

Therefore we consider the Apps with 5.0 Ratings and Maximum Installs :

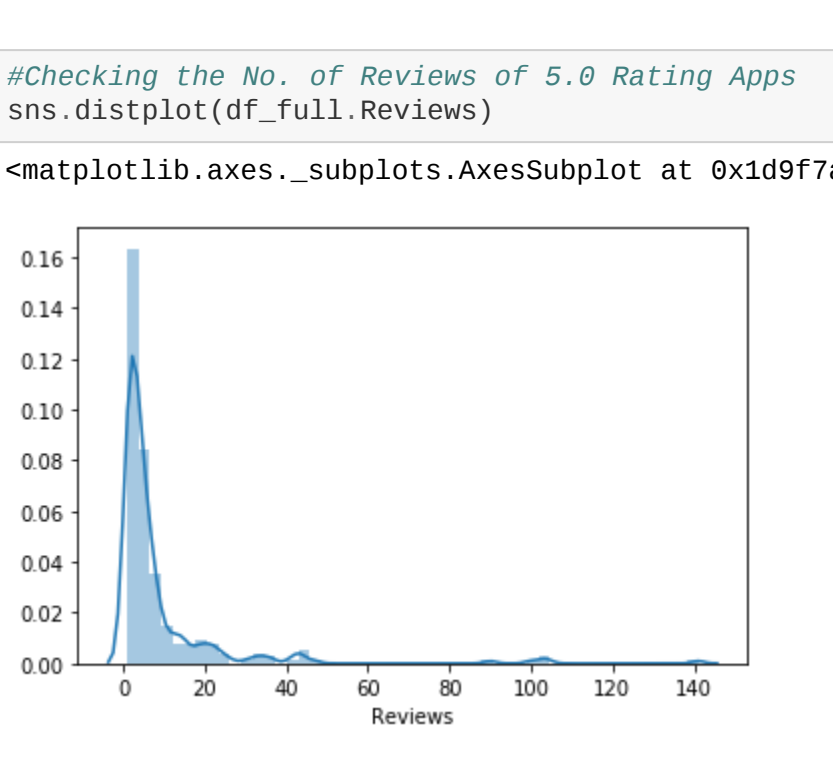
In [26]:
```
df_full_maxinstalls = df_full[df.Installs > 1000]
df_full_maxinstalls[['App', 'Category', 'Installs']]
```

Out[26]:

	App	Category	Installs
7514	CL Keyboard - Myanmar Keyboard (No Ads)	TOOLS	5000
8058	Oración CX	LIFESTYLE	5000
8260	Superheroes, Marvel, DC, Comics, TV, Movies News	COMICS	5000
9511	Ek Bander Ne Kholi Dukan	FAMILY	10000

In [27]:
```
#Checking the No. of Reviews of 5.0 Rating Apps
sns.distplot(df_full.Reviews)
```

Out[27]: <matplotlib.axes._subplots.AxesSubplot at 0x1d9f7a2dd48>



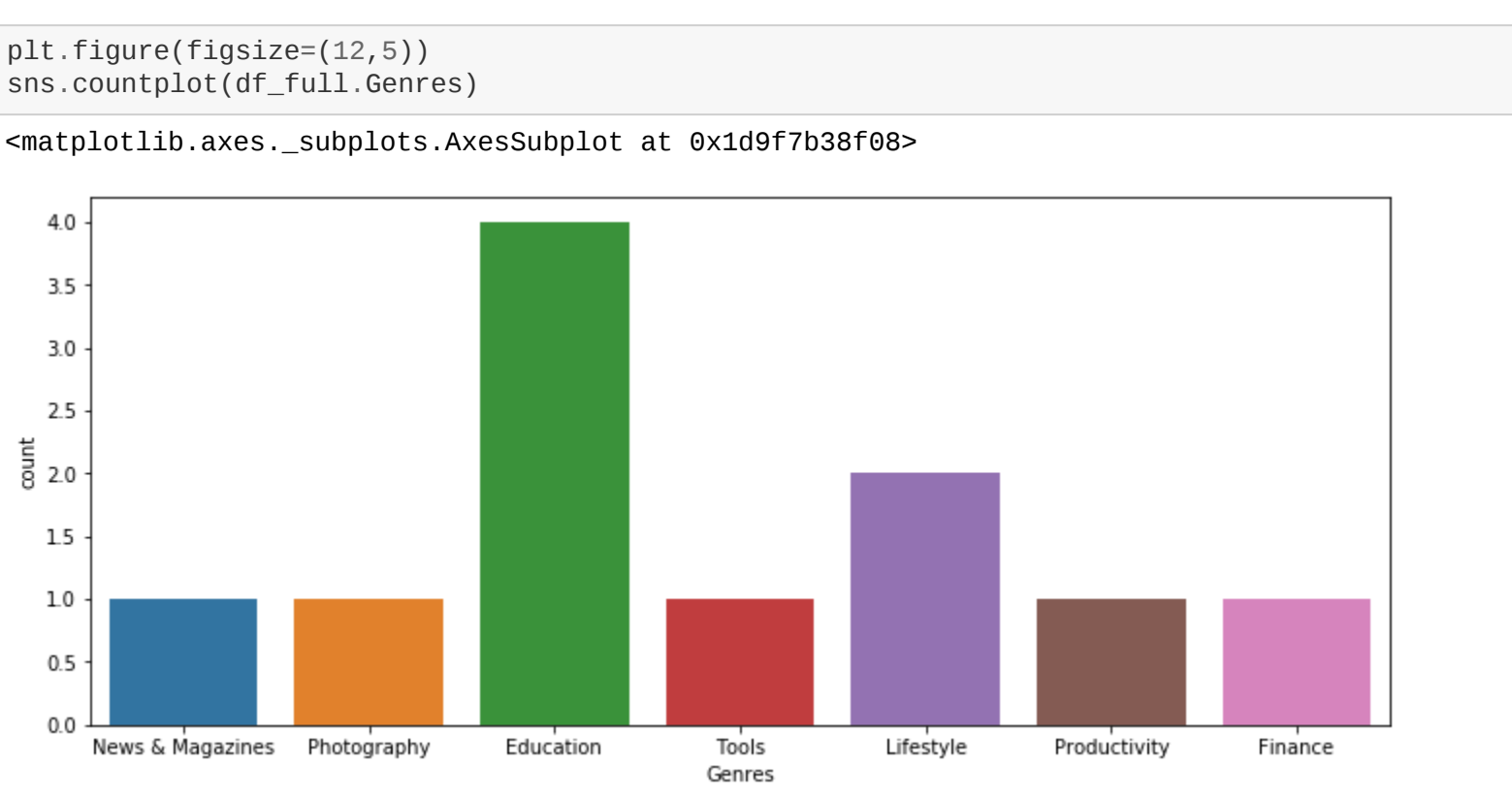
Apps with very few reviews easily managed to get 5.0 ratings which can be misleading. So let's filter out the ones with more than 40 reviews.

In [28]:
```
df_full = df_full[df.Reviews > 40]
print("No. of Apps having 5.0 Rating with sufficient Reviews: ",df_full.App.count())
```

```
No. of Apps having 5.0 Rating with sufficient Reviews:  11
```

In [29]:
```
plt.figure(figsize=(12,5))
sns.countplot(df_full.Genres)
```

Out[29]: <matplotlib.axes._subplots.AxesSubplot at 0x1d9f7b38f08>



Insight : Apps related to Education, LifeStyle and Tools seem to fetch full Ratings with sufficient number of reviews.

In [ ]: