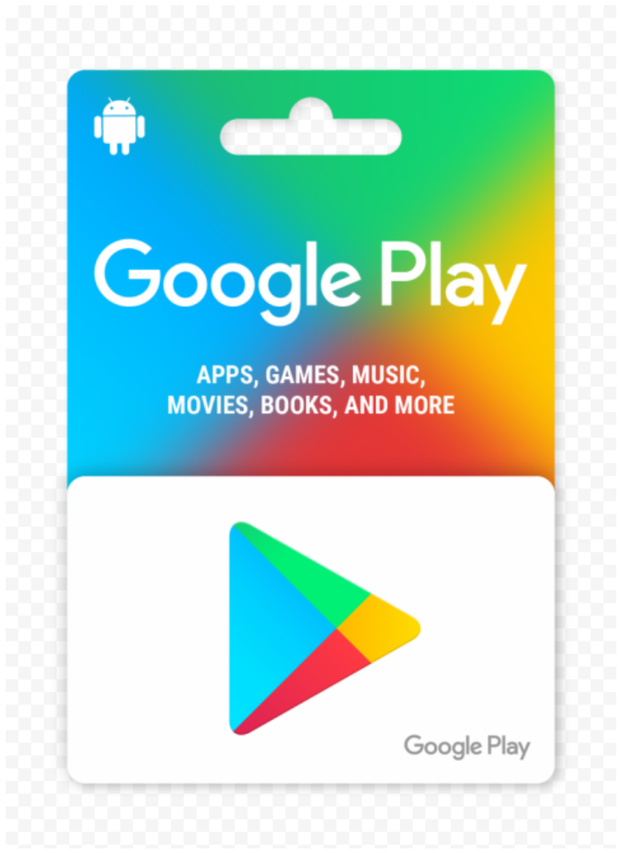
Project Name - Playstore App review EDA



Project Type - EDA

Contribution - Individual

Team Member - Aman Mulla

Project Summary -

📱 Let's Begin !

We are working on a Play Store app review dataset. First, let's understand what the Play Store is. Basically, the Play Store is a kind of application market (according to Wikipedia). When we visit any market (vegetable/fruit market or mall), we simply purchase items, search for required items, compare with other sellers, take reviews for those items, rate those items, and also check the demand for those items. Similarly, when we use a smartphone, we require different types of applications to meet various needs. According to Wikipedia, the Play Store is a kind of market. So, in the Play Store, we download applications, check their type (free or paid), examine their uses, genres of applications, write reviews for applications, give ratings for applications, and check the demand for applications (installs).

So, the given dataset (Dataset name: Play Store Data) is all about applications and their category, rating, reviews, size, installs, type, price, content rating, genres, and versions.

Along with that, we have one more dataset (Dataset name: User Reviews) in which we have applications and their reviews by users, sentiment (Positive, Negative, Neutral), sentiment polarity, and sentiment subjectivity.

Now, let's begin with understanding the dataset and its column names and data types.

For the Play Store Data dataset, we have the following columns:

- App: The name of the application in text format.
- Category: The category or type of the application.
- Rating: User ratings for the application, typically on a scale from 1 to 5.(Float64)
- Reviews: User-written reviews for the application in text format.
- Size: The size of the application.
- Installs: The number of installations or downloads of the application.
- Type: Indicates whether the application is free or paid.
- Price: The price of the application.
- Content Rating: The content rating of the application, such as "Everyone" or "Teen."
- Genres: The genre or category of the application.
- Last Updated: The date when the application was last updated.
- Current Ver: The current version of the application in text format.
- Android Ver: The required Android version for the application.

For the User Reviews dataset, we have the following columns:

- App: The name of the application in text format.
- Translated_Review:User-written reviews for the application in text format.
- Sentiment: The sentiment type in text format (Positive, Negative, Neutral).
- Sentiment_Polarity: Contains sentiment polarity for the app.
- Sentiment_Subjectivity:Contains sentiment subjectivity for the app."

GitHub Link -

Provide your GitHub Link here.

Problem Statement

1. What is the shape of both datasets?
2. How many different types of datatypes are there in both datasets?
3. How many applications are available in the dataset?
4. How many application categories are available, and which category has the maximum number of installs?
5. Show category-wise installs using a pie chart.
6. What is the category-wise average rating of applications?
7. Which applications have the maximum and minimum numbers of reviews?
8. Which applications have the maximum and minimum number of installs?
9. How many free and paid applications are there?
10. Which applications have the maximum and minimum prices?
11. What are the types of content ratings, and how many applications are in each type? Show this with a pie chart.
12. What are the types of genres, and how many applications are in each type? Show this with a pie chart.
13. Which application is the latest and oldest in the dataset?
14. On which Android version do most applications work?
15. How important is the rating of an application?

16. How does the count of apps vary by Genres?
17. How does the last update affect the rating?
18. How are ratings affected when the app is paid?
19. How are reviews and ratings correlated?
20. Discuss the sentiment subjectivity.
21. What is the proportion between subjectivity and polarity?
22. What is the percentage distribution of review sentiments?
23. What is the relationship between the type of application and sentiment polarity?
24. How does Content Rating affect the application?
25. Find the relationship between the last updated date and the rating.
26. Examine the app updation frequency over time (yearly).
27. Analyze the distribution of types of app updates over the months.

Define Your Business Objective?

The objective is to gain actionable insights from the Play Store dataset to optimize app performance and user satisfaction. This involves understanding various aspects of app categories, user sentiments, pricing strategies, and user engagement.

More for Business objective we can analyze for below points from dataset,

- Type of application Demand.
- Number of aplications.
- Applications prefered category by users.
- Applications prefered Genres by users.
- Sentiment Analysis for ratings, category.
- Distribution of application over ratings and reviews.

Let's Begin !

1. Know Your Data

Import Libraries

Firstly we will import necessary libraries. Which will help us to write code.

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

import warnings
warnings.filterwarnings("ignore")
```

Dataset Loading

Basically, we have two dataset, will load both dataset with two differnt variable.

```
from google.colab import drive
drive.mount('/content/drive')

playstore = pd.read_csv('/content/drive/MyDrive/DataSets/Playstore Dataset/PlayStoreData.csv')

user_review = pd.read_csv('/content/drive/MyDrive/DataSets/Playstore Dataset/User Reviews.csv')
```

```
Mounted at /content/drive
```

Dataset First View

```
# Dataset First Look

playstore.head()
```

	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

```
user_review.head()
```

	App	Translated_Review	Sentiment	Sentiment_Polarity	Sentiment_Subjectivity
0	10 Best Foods for You	I like eat delicious food. That's I'm cooking ...	Positive	1.00	0.533333
1	10 Best Foods for You	This help eating healthy exercise regular basis	Positive	0.25	0.288462
2	10 Best Foods for You	NaN	NaN	NaN	NaN
3	10 Best Foods for You	Works great especially going grocery store	Positive	0.40	0.875000
4	10 Best Foods for You	Best idea us	Positive	1.00	0.300000

Dataset Rows & Columns count

```
# Dataset Rows & Columns count
playstore.shape
```

```
(10841, 13)
```

Play-store Data dataset having shape (10841, 13), that is 10840 rows and 13 columns.

```
user_review.shape
```

```
(64295, 5)
```

user_review Data dataset having shape (64295,5), that is 64295 rows and 5 columns.

Dataset Information

```
# Dataset Info
playstore.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
```

```
user_review.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 64295 entries, 0 to 64294
Data columns (total 5 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   App                     64295 non-null  object
 1   Translated_Review       37427 non-null  object
 2   Sentiment               37432 non-null  object
 3   Sentiment_Polarity      37432 non-null  float64
 4   Sentiment_Subjectivity  37432 non-null  float64
dtypes: float64(2), object(3)
memory usage: 2.5+ MB
```

From .info(), we will obtain all the information about the dataset. This includes all column names, along with their respective non-null value counts and datatypes. In the Play Store dataset, most of the data entries are in the 'object' datatype, which represents text format. Only the 'rating' column has a datatype of 'float.' Additionally, we need to check for NaNs/Null values in the next few steps.

Duplicate Values

```
# Dataset Duplicate Value Count
# Will check for duplicate value present in App column. Why App column selected because this dataset is all
# about application and its all information. If we having any duplictes values in app column, will drop for same so it will get dropped from other column also.
# Below line of code gives us count of each entry in app column.

playstore['App'].value_counts()

# Below line of code is show us all entries with 'ROBLOX'. We can check for all by changing attribute.
playstore[playstore['App']=='ROBLOX']

# So, besically we are having lots of duplicte entries.We can verify by observing them all are having same entries.
# Below line of code give us all data which are duplictes.

playstore[playstore.duplicated()]

# By Using .drop_duplicates will drop all duplicte values with axis=1.

playstore.drop_duplicates(subset='App',inplace=True)
```

Missing Values/Null Values

```
# Missing Values/Null Values Count

playstore.isnull().sum()

# With above we can say most no. of Null values are in rating only. In other column having null values but its very less.Will check column wise NaNs and work on it

# Start will highest number of NaNs which is in rating column.

playstore[playstore['Rating'].isnull()]

# Above output will give you entries with value NaN. Total 1474 rows are with NaNs,
# So it will not practically healty to drop this much of column, we will loss much amount of data by dropping.
# To fill NaN will fill with Mean and median.

mean_rating = playstore['Rating'].mean()

median_rating =playstore['Rating'].median()

playstore['Rating'] = playstore['Rating'].fillna(playstore['Rating'].median())


playstore.dropna(subset=['Type'],inplace=True)

playstore.dropna(subset=['Content Rating'],inplace=True)

playstore.dropna(subset=['Current Ver'],inplace=True)

playstore.dropna(subset=['Android Ver'],inplace=True)

playstore.isnull().sum()
```

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

By using .dropna(), We have dropped NaNs present in perticuler columns. In this columns NaNs count was less hence we have dropped them. It will not affect to any of analysis

```
playstore.shape
```

```
(9648, 13)
```

After dropping NaNs and duplictes will check for all datatype and will change datatype as per our requirement.

```
# Will Change data type of 'last updated' column by datetime from string as 'last updated' column contain all date entries.

playstore['Last Updated'] = pd.to_datetime(playstore['Last Updated'])

# Will Change data type of 'Price' column by string from flote as 'Price' column contain all numerical values.

playstore[playstore['Price']!='0']

# As we can chek in price column, price is given with $ sign. For changing in Flote type we have to $ sign.

def convert_dollar(val):
  '''
  This funtion drops the $ symbol if present and returns the value with float datatype.
  '''
  if '$' in val:
    return float(val[1:])
  else:
    return float(val)

playstore['Price']=playstore['Price'].apply(lambda x: convert_dollar(x))
```

```
# Will change data type of 'Installs' to integer from string. Also for Installs we have entry with '+' sign.

def convert_plus(val):
  '''
  This function drops the + symbol if present and returns the value with int datatype.
  '''
  if '+' and ',' in val:
    new = int(val[:-1].replace(',',''))
    return new
  elif '+' in val:
    new1 = int(val[:-1])
    return new1
  else:
    return int(val)

playstore['Installs'] = playstore['Installs'].apply(lambda x: convert_plus(x))

# Similerly we have size column with 'M'. Will convery size column to float from strig Datatype.

def convert_kb_to_mb(val):
  '''
  This function converts all the valid entries in KB to MB and returns the result in float datatype.
  '''
  try:
    if 'M' in val:
      return float(val[:-1])
    elif 'k' in val:
      return round(float(val[:-1])/1024, 4)
    else:
      return val
  except:
    return val

playstore['Size'] = playstore['Size'].apply(lambda x: convert_kb_to_mb(x))
```

What did you know about your dataset?

Answer Here

2. Understanding Your Variables

```
# Dataset Columns

playstore.columns
```

```
Index(['App', 'Category', 'Rating', 'Reviews', 'Size', 'Installs', 'Type',
       'Price', 'Content Rating', 'Genres', 'Last Updated', 'Current Ver',
       'Android Ver'],
      dtype='object')
```

```
# Dataset Describe

playstore.describe()
```

	Rating	Installs	Price
count	9648.000000	9.648000e+03	9648.000000
mean	4.192465	7.786211e+06	1.100193
std	0.496552	5.378830e+07	16.861727
min	1.000000	0.000000e+00	0.000000
25%	4.000000	1.000000e+03	0.000000
50%	4.300000	1.000000e+05	0.000000
75%	4.500000	1.000000e+06	0.000000
max	5.000000	1.000000e+09	400.000000

From .describe we can understad differnt type of outcome for numerical column. Like Mean, standard Vaviation, count, Min, Max.

Check Unique Values for each variable.

```
# Check Unique Values for each variable.

playstore['App'].unique()

playstore['Category'].unique()

playstore['Rating'].unique()

playstore['Reviews'].unique()

playstore['Size'].unique()

playstore['Installs'].unique()

playstore['Type'].unique()

playstore['Price'].unique()

playstore['Content Rating'].unique()

playstore['Genres'].unique()

playstore['Last Updated'].unique()

playstore['Current Ver'].unique()

playstore['Android Ver'].unique()
```

```
array(['4.0.3 and up', '4.2 and up', '4.4 and up', '2.3 and up',
       '3.0 and up', '4.1 and up', '4.0 and up', '2.3.3 and up',
       'Varies with device', '2.2 and up', '5.0 and up', '6.0 and up',
       '1.6 and up', '1.5 and up', '2.1 and up', '7.0 and up',
       '5.1 and up', '4.3 and up', '4.0.3 - 7.1.1', '2.0 and up',
       '3.2 and up', '4.4W and up', '7.1 and up', '7.0 - 7.1.1',
       '8.0 and up', '5.0 - 8.0', '3.1 and up', '2.0.1 and up',
       '4.1 - 7.1.1', '5.0 - 6.0', '1.0 and up', '2.2 - 7.1.1',
       '5.0 - 7.1.1'], dtype=object)
```

3. *Data Wrangling*

What all manipulations have you done and insights you found?

Below Manipulation done,

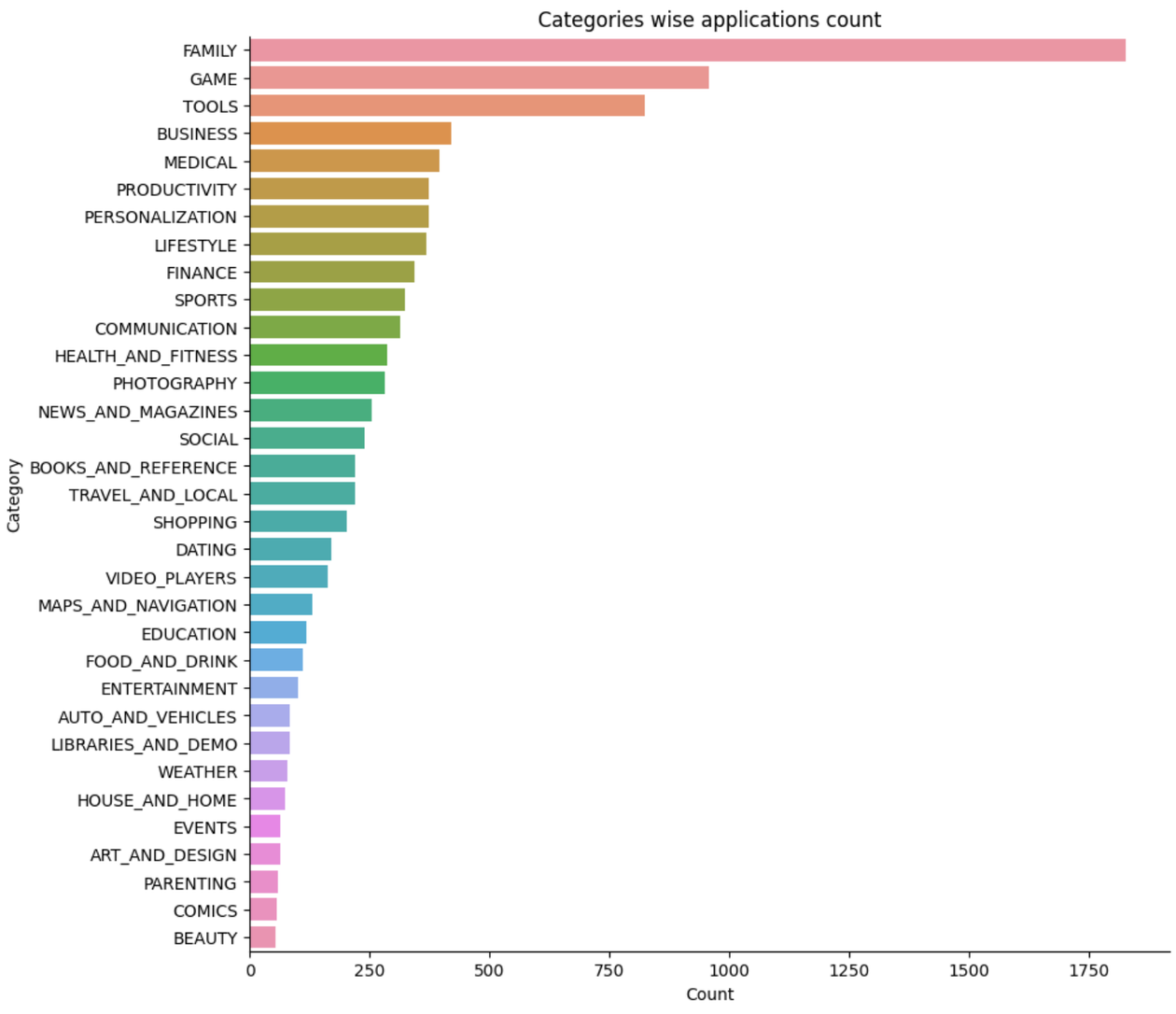
1. Treaded for Nulls/NaNs.
2. Changed datatype as per requirement.
3. Treated for duplicate Value.

4. Data Vizualization, Storytelling & Experimenting with charts : Understand the relationships between variables

Chart - 1

Categories and thier application Count


```
Categories_app_count = playstore['Category'].value_counts()
Categories_app_count

f,ax = plt.subplots(figsize=(10,10))
sns.despine(f)
sns.barplot(x = Categories_app_count.values, y = Categories_app_count.index ,data=playstore )
plt.xlabel('Count')
plt.ylabel('Category')
plt.title('Categories wise applications count')
plt.show()
```



1. Why did you pick the specific chart?

This bar plot is a suitable choice when want to compare the number of applications in different categories easily. The length of each bar directly represents the count of applications in each category, making it simple to analyze which categories have more or fewer apps.

2. What is/are the insight(s) found from the chart?

Most Popular Categories: Family, Games, tools**

App Diversity : Few Categories with large number of application and few with less numbers.

Opportunities : For Which categories having less number of application its having opportinuities for development.

Market Fall : Can check market is falling towards the categories like Family type, Games, tools and Business while there is less market fall towards categories like beauty, comics,parnting and Art_and_design

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

- Insights that highlight categories with a lower number of apps can provide opportunities for businesses for leading to positive growth
- Creating apps that align with popular categories can lead to increased downloads and revenue.

We can raise awareness for applications that have a smaller user base. By doing so, we can share this information with application developers, giving them the opportunity to create applications focused on beauty, comics, and parenting

Chart 1.2 Above Graph with differnt type of representation

```
Categories_app_count = playstore['Category'].value_counts().reset_index()
plt.figure(figsize=(10,10))
ax = plt.subplot(111)
plt.pie(x = Categories_app_count['Category'], labels= Categories_app_count['index'],autopct= '%1.1f%%')
plt.legend()
ax.legend(bbox_to_anchor=(1.4, 1))
plt.show()
```

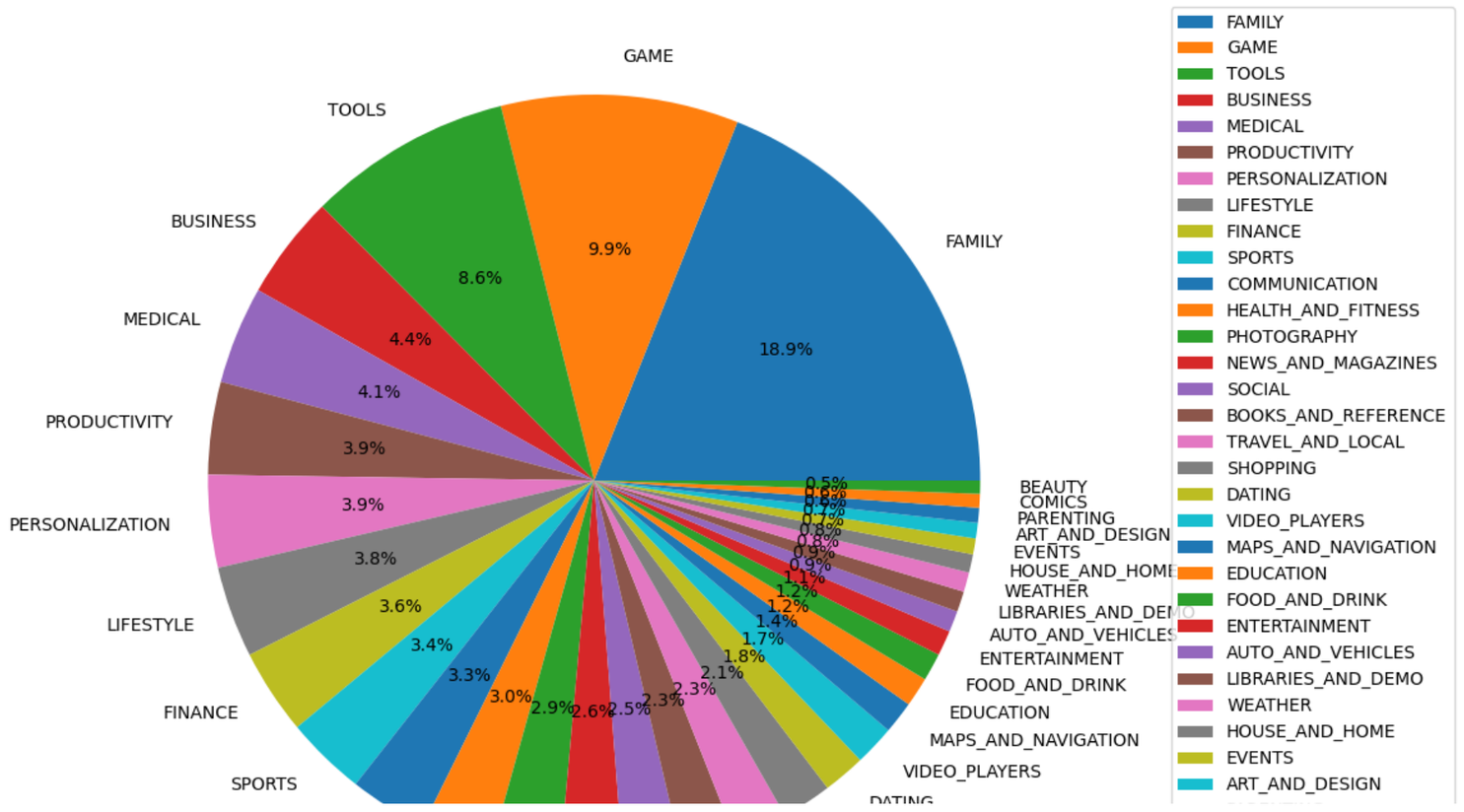
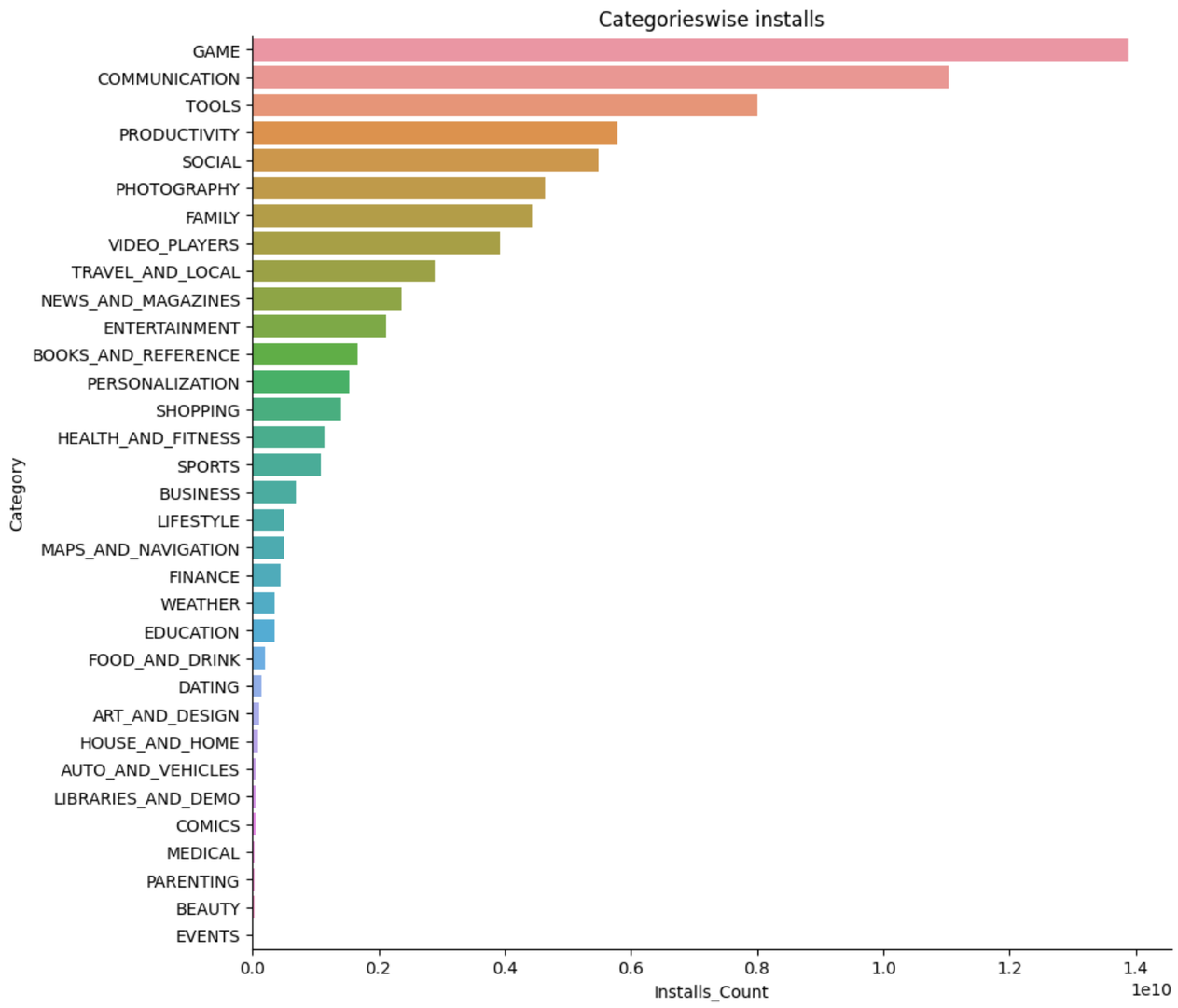


Chart - 2

Categories and thier application Installs

```
Category_installs = playstore.groupby('Category')['Installs'].sum().sort_values(ascending=False)
Category_installs

f,ax = plt.subplots(figsize=(10,10))
sns.despine(f)
sns.barplot(x = Category_installs.values, y = Category_installs.index ,data=playstore,orient='h' )
plt.xlabel('Installs_Count')
plt.ylabel('Category')
plt.title('Categorieswise installs')
plt.show()
```



1. Why did you pick the specific chart?

This bar plot is a suitable choice when want to compare the total number of installs in different categories easily. The length of each bar directly represents the count of installs in each category, making it simple to analyze which categories have more or fewer apps.

2. What is/are the insight(s) found from the chart?

Most installed Categories: Game, Communication, Tools.

App Diversity : Few Categories with large number of installs and few with less numbers.

Opportunities : For Which categories having less number of installs its having opportinuities for development.

Market Fall : Can check market is falling towards the categories like Family Game, Communication, Tools while there is less market fall towards categories like Event, Beauty ,parnting and Medical.

3. Will the gained insights help creating a positive business impact?

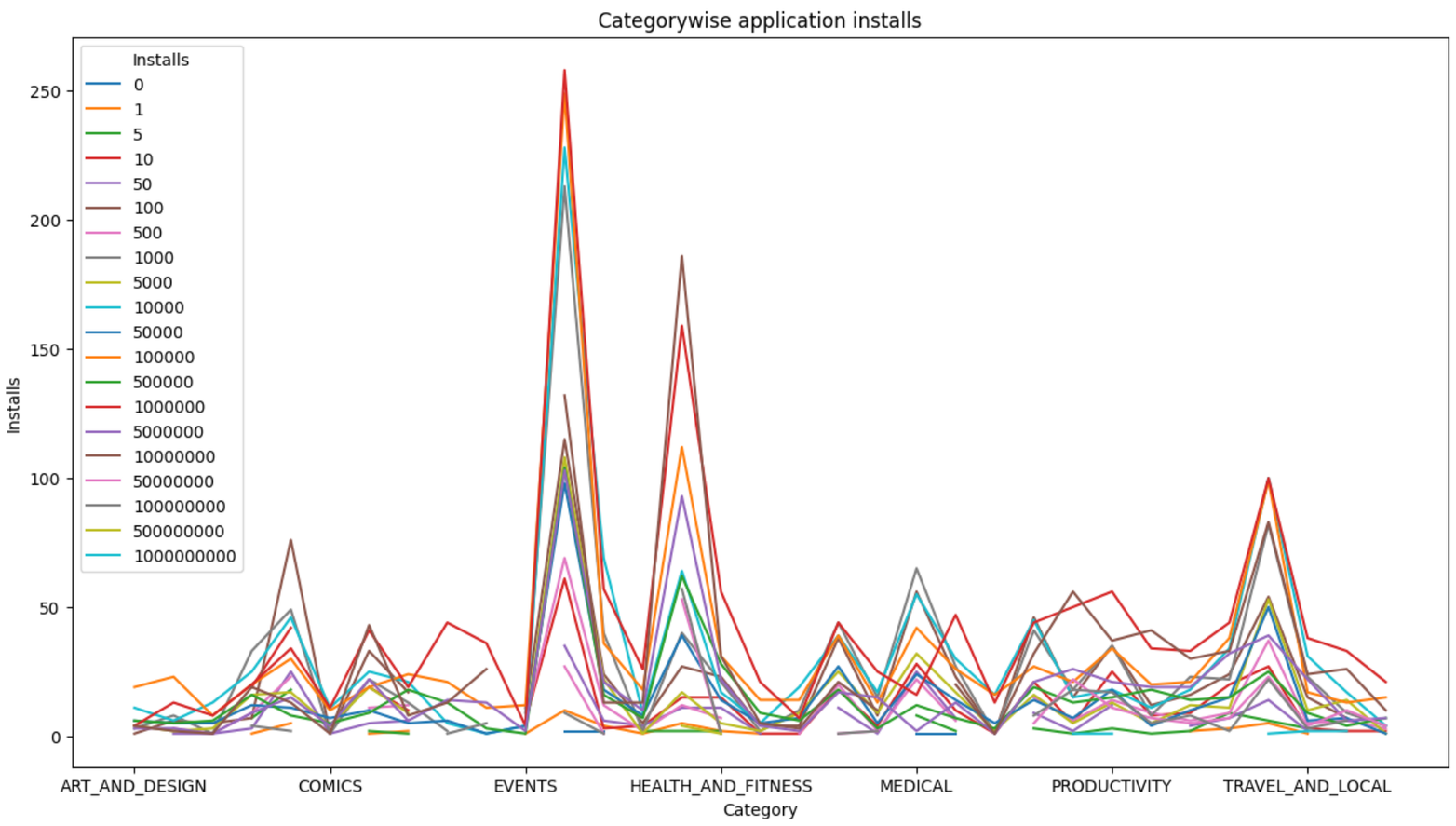
Are there any insights that lead to negative growth? Justify with specific reason.

- Insights that highlight categories with a lower number of installs can provide opportunities for businesses for leading to positive growth
- Creating apps that align with popular categories can lead to increased downloads and revenue.

We can raise awareness for categories that have a smaller user base. By doing so, we can share this information with application developers, giving them the opportunity to create applications focused on beauty, comics, and parenting

Chart 2.1 Above Graph with differnt type of representation

```
cat_install = playstore.groupby(['Category','Installs'])['Category'].count().unstack()
cat_install

cat_install.plot(figsize=(15,8))
plt.title('Categorywise application installs')
plt.xlabel('Category')
plt.ylabel('Installs')
plt.show()
```



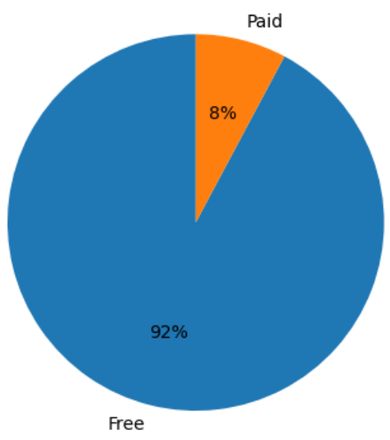
Above Graph seems confusing and unable to give clear picture while bar graph gives clar picture what wanted to showcase.

Chart - 3

Type of app and its count

```
Type_installs = playstore.groupby('Type')['Installs'].sum()
print(Type_installs)

Type = playstore['Type']
Type_count = playstore['Type'].value_counts()
plt.pie(Type_count,labels =Type_count.index , autopct='%.0f%%',startangle=90)
plt.show()
```

```
Type
Free    75064046546
Paid       57313871
Name: Installs, dtype: int64
```



1. Why did you pick the specific chart?

The specific chart chosen in the code is a pie chart. The choice of a pie chart is typically based on the specific data and the kind of information you want to communicate.

2. What is/are the insight(s) found from the chart?

Proportion of Free and Paid Apps: Free apps 92% and Paid Apps 8%.

Comparison of App Types:"Free" segment is much larger than the "Paid" segment, it indicates that there are more free apps available in the dataset.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

- **This pie chart shows that the majority of apps in the dataset are free and they are popular among users, a business may decide to focus on offering more free apps. This could lead to increased user engagement, higher download numbers, and potential revenue from ads, in-app purchases, or premium versions of the free apps.**
- **Free apps are more popular, the business can focus on advertising and promoting these apps to attract more users. For paid apps, the marketing strategy might involve emphasizing their unique features or benefits.**

Chart 3.1 Top 10 paid applications

```
paid_apps = playstore[playstore['Type'] == 'Paid']

top_10_paid_apps = paid_apps.sort_values(by='Price', ascending=False).head(10)

print(top_10_paid_apps[['App','Price']])

plt.figure(figsize=(12, 6))
plt.plot(top_10_paid_apps['Price'], top_10_paid_apps['App'], color='skyblue')
plt.title('Top 10 paid applicaation')
plt.xlabel('Price')
plt.ylabel('App')
plt.xticks(rotation=45)
```

```
                                App   Price
4367          I'm Rich - Trump Edition  400.00
5356                I Am Rich Premium  399.99
5373               I AM RICH PRO PLUS  399.99
5369                          I am Rich  399.99
5364     I am rich (Most expensive app)  399.99
5362                      I Am Rich Pro  399.99
4197            most expensive app (H)  399.99
5359                I am rich(premium)  399.99
5358                          I am Rich!  399.99
9934  I'm Rich/Eu sou Rico/أنا غني/我很有錢  399.99
(array([399.988, 399.99 , 399.992, 399.994, 399.996, 399.998, 400.   ,
       400.002]),
 [Text(399.988, 0, '−0.002'),
  Text(399.99, 0, '0.000'),
  Text(399.992, 0, '0.002'),
  Text(399.99399999999997, 0, '0.004'),
  Text(399.996, 0, '0.006'),
  Text(399.998, 0, '0.008'),
  Text(400.0, 0, '0.010'),
  Text(400.002, 0, '0.012')])
```

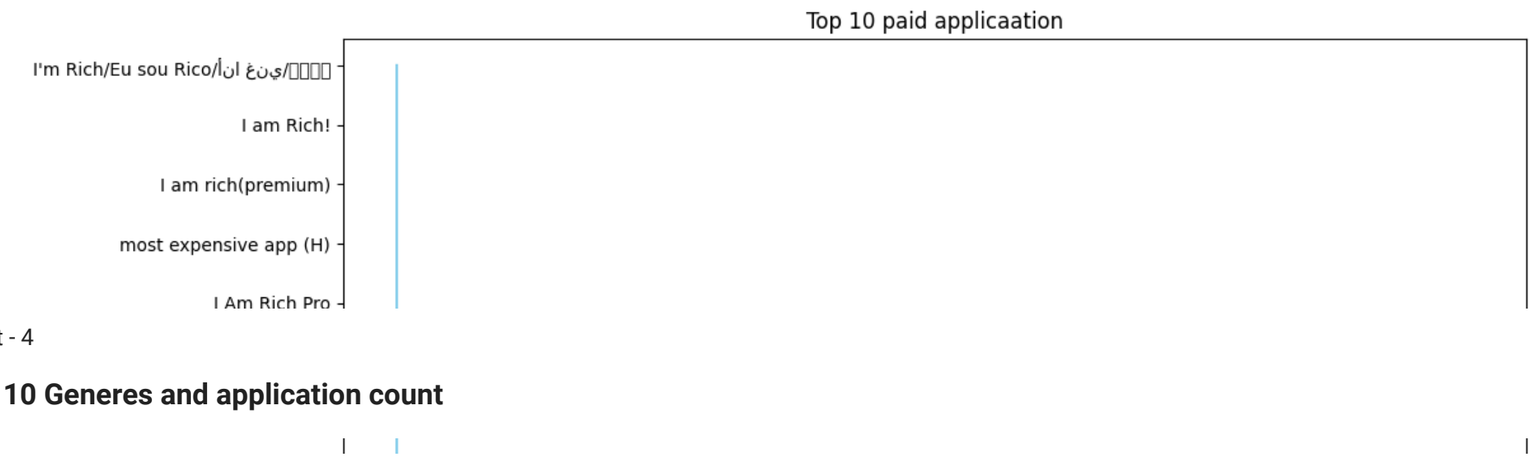
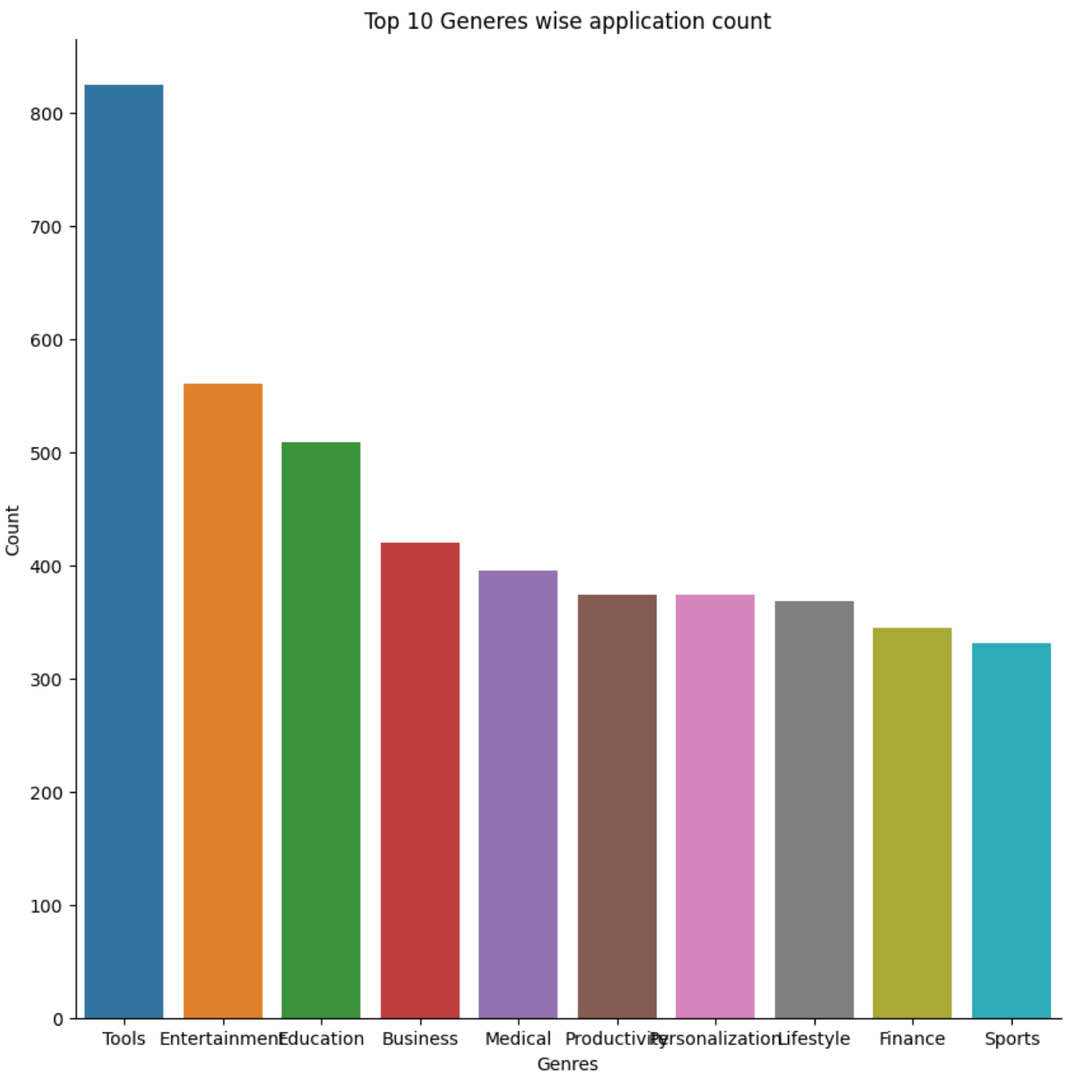


Chart - 4

Top 10 Generes and application count

```
playstore.head()

playstore['Genres'].value_counts().shape

Top_10_Genres = playstore['Genres'].value_counts().head(10)

f,ax = plt.subplots(figsize=(10,10))
sns.despine(f)
sns.barplot(y = Top_10_Genres.values, x = Top_10_Genres.index ,data=playstore )
plt.xlabel('Genres')
plt.ylabel('Count')
plt.title('Top 10 Generes wise application count')
plt.show()
```



1. Why did you pick the specific chart?

The chart is use to visualize the distribution of application genres. Bar plots are well-suited for count of different categories or groups, making it easy to see which genres are the most common.

2. What is/are the insight(s) found from the chart?

Most Popular Genre:Tools,Entertaainment,Education.

App Diversity : Few Genre with large number of installs and few with less numbers.

Opportunities : For Which Genre having less number of installs its having opportinuities for development.

Market Fall : Can check market is falling towards the Genre like Tools,Entertaainment,Education while there is less market fall towards categories like Sport, Finance and Lifestyle.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

- **Most popular genres can guide businesses in developing new applications that cater to these genres. This targeted approach can lead to higher user engagement and increased downloads.**
- **Chart shows that the "Games" genre is highly saturated, it might be more strategic to explore less competitive genres**
- **There's a less common genre with a significant user base, entering that niche could lead to positive growth.**

Chart 4.1 Top 10 Generes and application count

```
Top_10_Genres = playstore['Genres'].value_counts().head(10)

f,ax = plt.subplots(figsize=(12,6))
sns.despine(f)
sns.lineplot(x=Top_10_Genres.index, y=Top_10_Genres.values, marker='o')
plt.xlabel('Genres')
plt.ylabel('Count')
```

```
plt.title('Top 10 Generes wise application count')
plt.show()
```

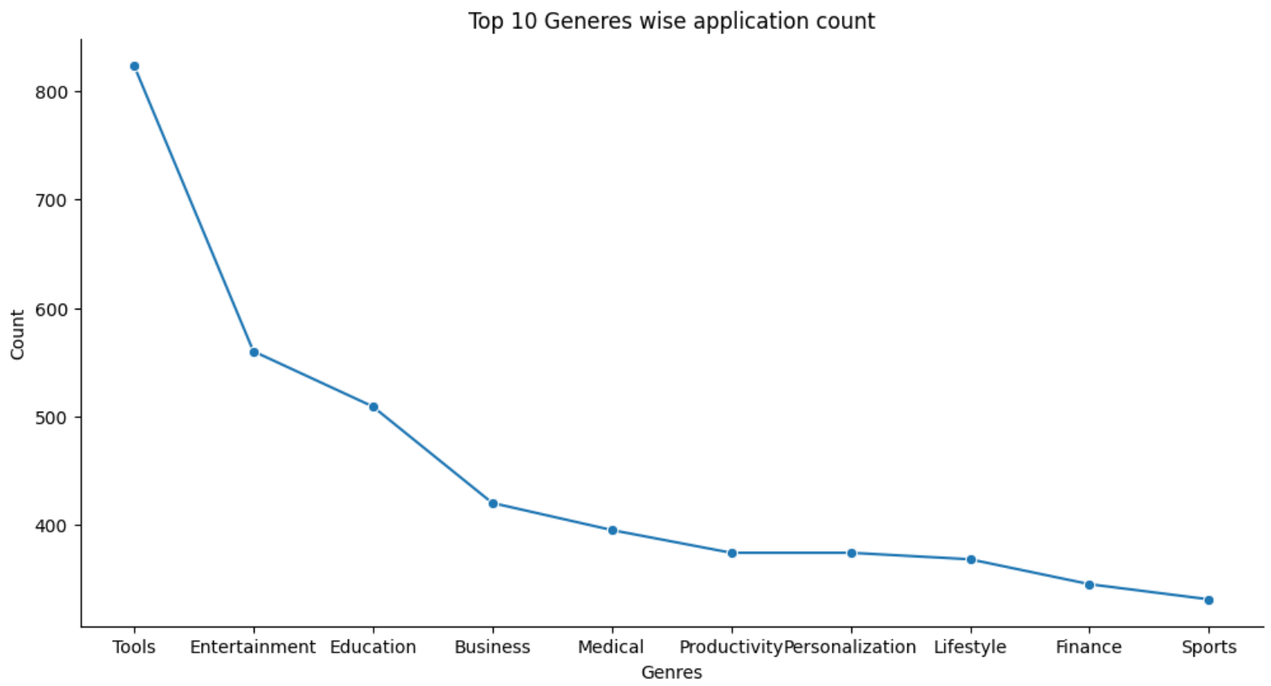


Chart 4.2 Top 10 Generes For Paid and Free app

```
paid_free_apps = playstore[playstore['Type'].isin(['Paid', 'Free'])]
top_10_paid_genres = paid_free_apps[paid_free_apps['Type'] == 'Paid']['Genres'].value_counts().head(10)
top_10_free_genres = paid_free_apps[paid_free_apps['Type'] == 'Free']['Genres'].value_counts().head(10)

print(top_10_paid_genres)
print(top_10_free_genres)

fig, axes = plt.subplots(nrows=1, ncols=2, figsize=(14, 6))
sns.barplot(x=top_10_paid_genres.values, y=top_10_paid_genres.index, ax=axes[0], palette='magma')
axes[0].set_xlabel('Count')
axes[0].set_ylabel('Genres')
axes[0].set_title('Top 10 Genres for Paid Apps')

sns.barplot(x=top_10_free_genres.values, y=top_10_free_genres.index, ax=axes[1], palette='magma')
axes[1].set_xlabel('Count')
axes[1].set_ylabel('Genres')
axes[1].set_title('Top 10 Genres for Free Apps')
```

```
Medical             83
Personalization     80
Tools               77
Education           30
Productivity        28
Books & Reference   28
Communication       27
Sports              24
Action              24
Role Playing        22
Name: Genres, dtype: int64
Tools              747
Entertainment      541
Education          479
Business           408
Lifestyle          349
Productivity       346
Finance            328
Medical            312
Sports             307
Personalization    294
Name: Genres, dtype: int64
Text(0.5, 1.0, 'Top 10 Genres for Free Apps')
```

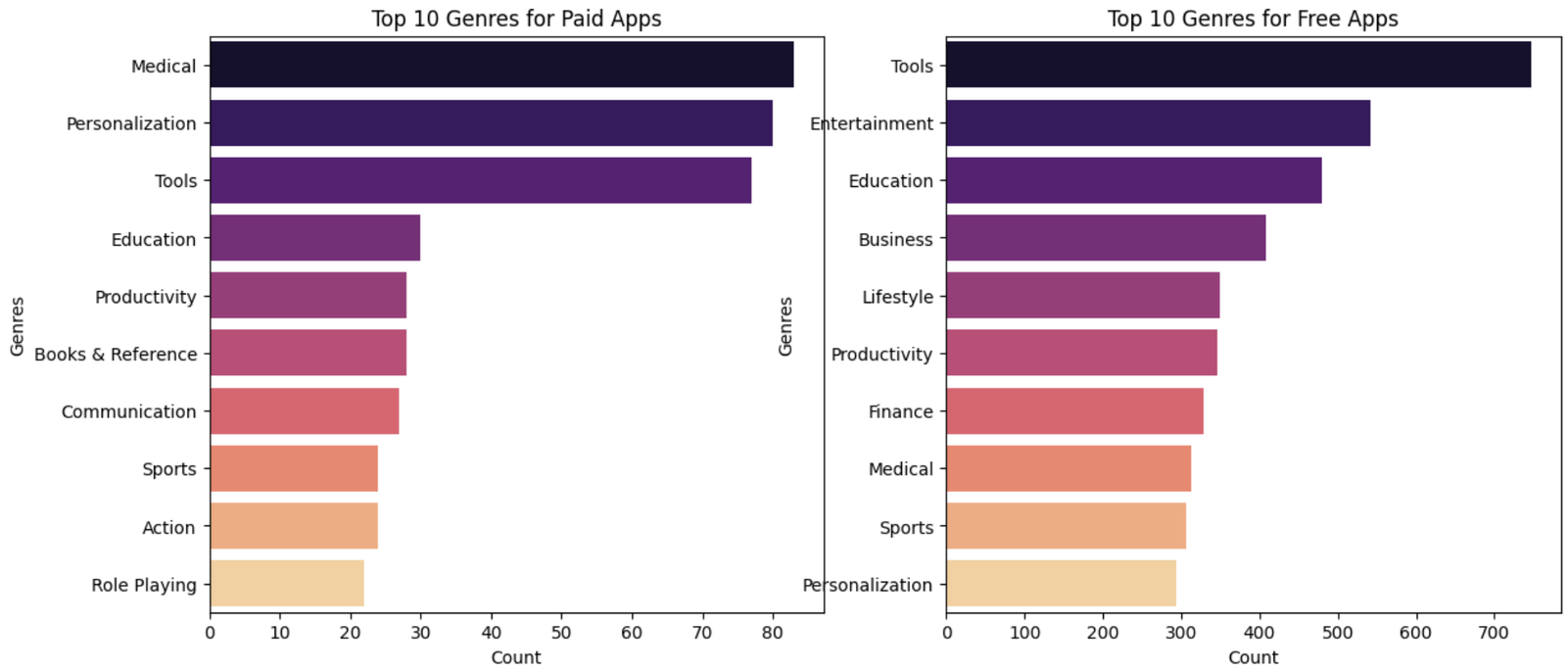
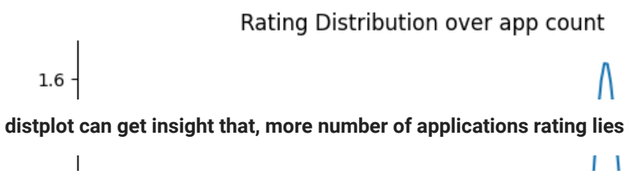


Chart - 5

Ratings Distribution Plot

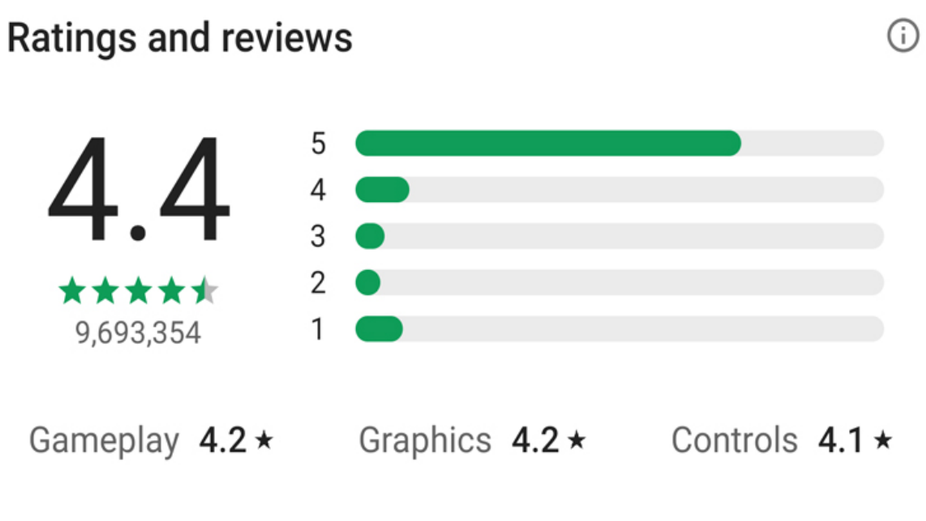
```
Rating_App_count = playstore['Rating'].value_counts().head(10)
print(Rating_App_count)

f,ax = plt.subplots(figsize=(7,5))
sns.despine(f)
sns.distplot(playstore['Rating'],bins=10)
plt.title('Rating Distribution over app count')
plt.xlabel('Rating')
plt.show()
```

```
4.3    2355
4.4     894
4.5     847
4.2     809
4.6     683
4.1     620
4.0     512
4.7     442
3.9     359
3.8     286
Name: Rating, dtype: int64
```

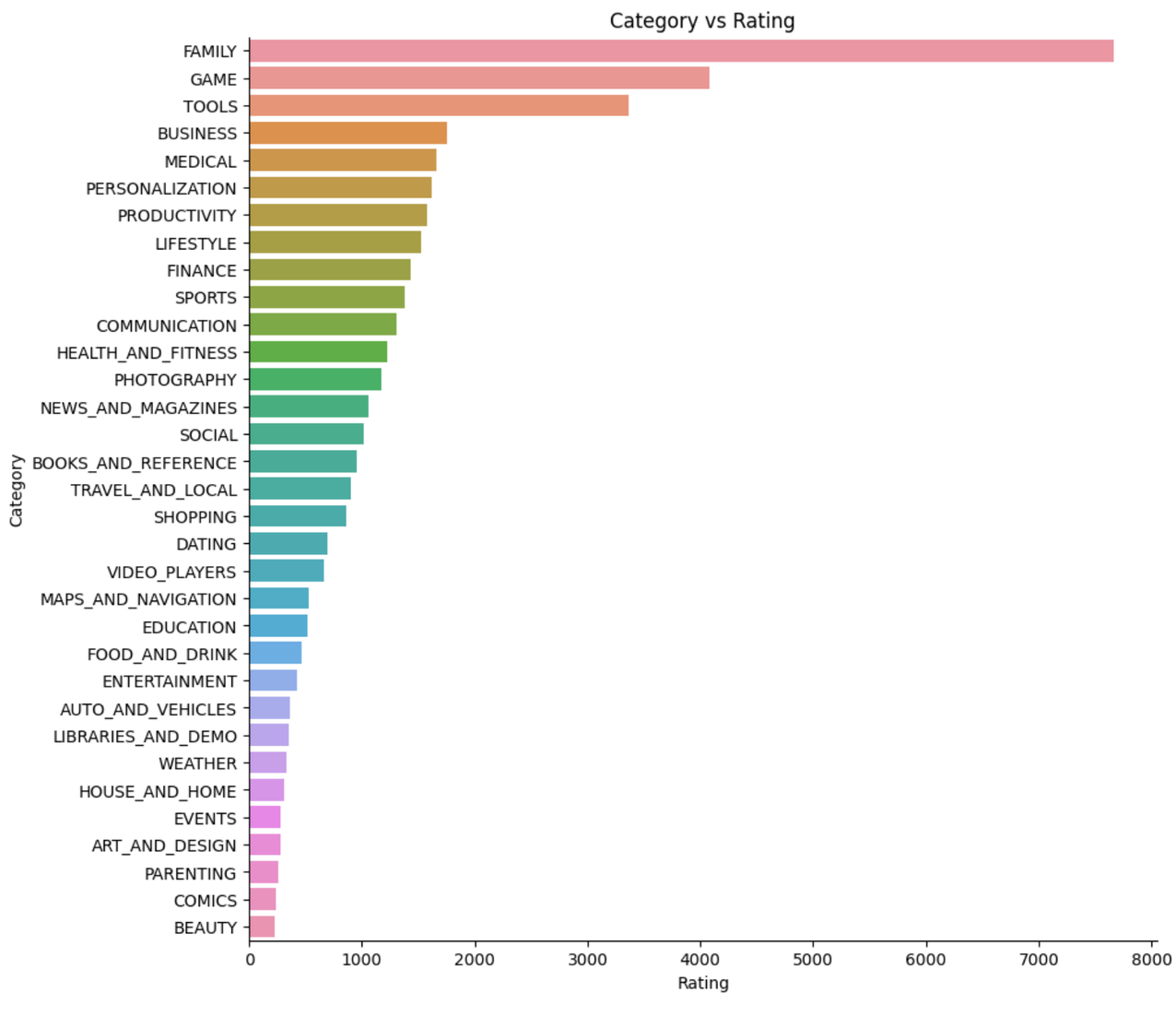


From distplot can get insight that, more number of applications rating lies near 4 to 4.5. Very Few application with rating from 1 to 3.

Category Vs Rating




```
Category_rating = playstore.groupby('Category')['Rating'].sum().sort_values(ascending=False)
Category_rating

f,ax = plt.subplots(figsize=(10,10))
sns.despine(f)
sns.barplot(x = Category_rating.values, y = Category_rating.index ,data=playstore,orient='h' )
plt.xlabel('Rating')
plt.ylabel('Category')
plt.title('Category vs Rating')
plt.show()
```



1. Why did you pick the specific chart?

This bar plot is a suitable choice when want to compare the Sum of Rating in different categories easily. The length of each bar directly represents the Sum of Rating in each category, making it simple to analyze which categories have more or fewer apps.

2. What is/are the insight(s) found from the chart?

Top Rated Categories: Family, Game, Tools.

Bottom Rated Categories: Beauty, Comics,Parenting

App Diversity : Few categories with large number of rating and few with less numbers.

Opportunities : For Which categories having less number of rating its having opportinuities for development.

Market Fall : Can check market is falling towards the Genre Family, Game, Tools. while there is less market fall towards categories like Beauty, Comics,Parenting.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

- Insights that highlight categories with a lower number of Rating can provide opportunities for businesses for leading to positive growth
- Creating apps that align with popular categories can lead to increased Rating

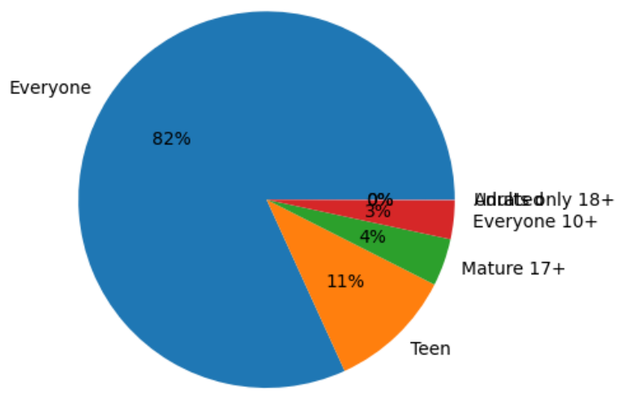
We can raise awareness for categories that have a smaller user base. By doing so, we can share this information with application developers, giving them the opportunity to create applications focused on beauty, comics, and parenting

Chart - 6

Containt Rating vs Applicaion Count

```
Content_Rating = playstore['Content Rating']
Content_Rating_count = playstore['Content Rating'].value_counts()

print(Content_Rating_count)

plt.pie(Content_Rating_count,labels =Content_Rating_count.index , autopct='%.0f%%')
plt.show()
```

```
Everyone           7893
Teen               1036
Mature 17+          393
Everyone 10+        321
Adults only 18+       3
Unrated               2
Name: Content Rating, dtype: int64
```



1. Why did you pick the specific chart?

The specific chart chosen in the code is a pie chart. The choice of a pie chart. This will shows Proportions of containt rating for number of application. Pie chart effectively communicates the distribution of content ratings and helps you make your point, it can be a suitable choice.

2. What is/are the insight(s) found from the chart?

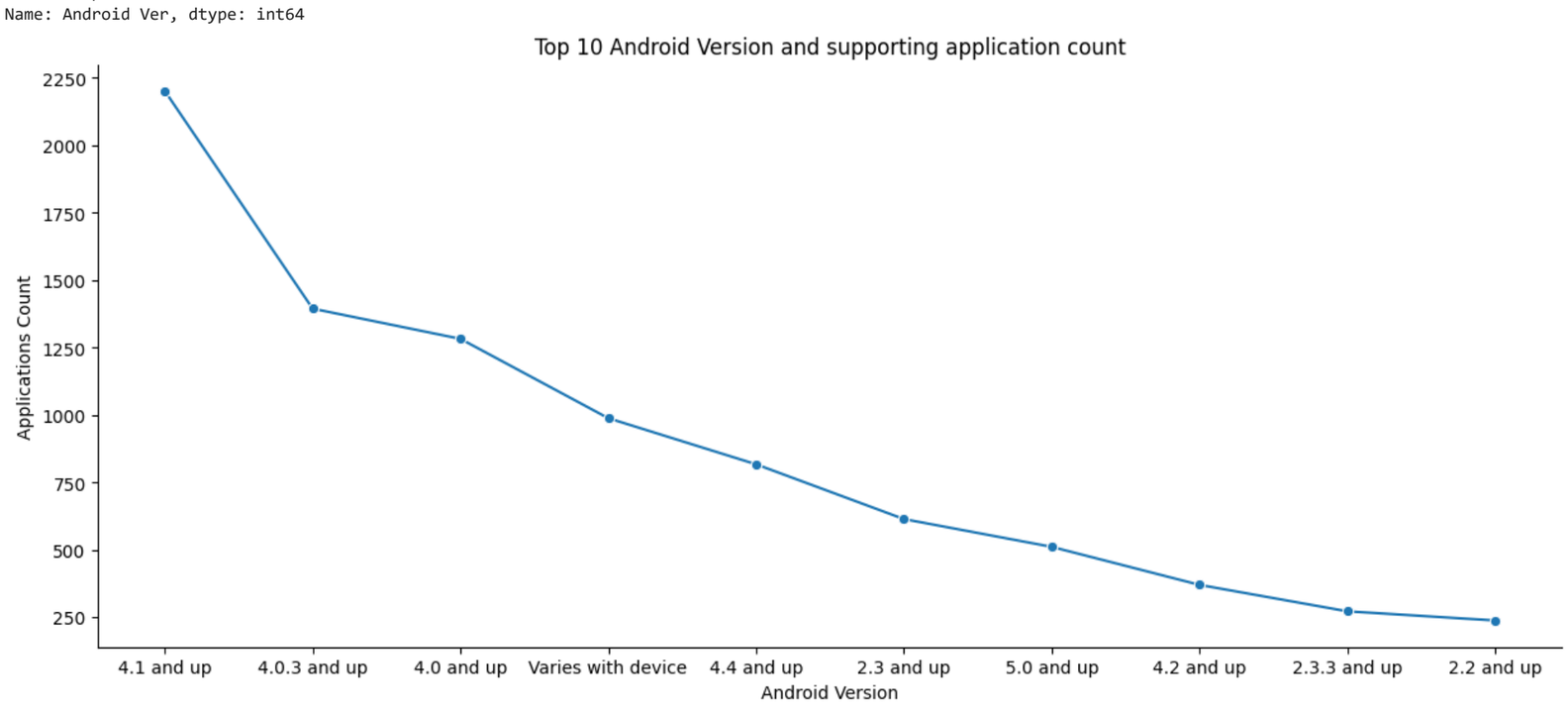
- **There are 82% applications which are belong to 'Everyone' while 11%,4% belogs to 'teen' and 'Mature17+' Content Rating. There ae very less applications for 'Adults only 18+'.**
- **Opportunities : More Opportunities for Content rating for 'Adults only 18+','Everyone 10+'.**

Chart - 7

Count of application and supporting android Version

```
android_ver = playstore['Android Ver'].value_counts().head(10)
print(android_ver)

f,ax = plt.subplots(figsize=(15,6))
sns.despine(f)
sns.lineplot(x=android_ver.index, y=android_ver.values, marker='o')
plt.xlabel('Android Version')
plt.ylabel('Applications Count')
plt.title('Top 10 Android Version and supporting application count')
plt.show()
```

```
4.1 and up            2202
4.0.3 and up          1395
4.0 and up            1283
Varies with device     989
4.4 and up             818
2.3 and up             615
5.0 and up             512
4.2 and up             371
2.3.3 and up           273
2.2 and up             239
Name: Android Ver, dtype: int64
```



1. Why did you pick the specific chart?

This graph helps you visualize the distribution of Android versions among the applications in your dataset. It shows which Android versions are most commonly supported by these applications.can see which versions are most popular and whether there are any outliers or unusual patterns.

2. What is/are the insight(s) found from the chart?

- **Most used Android Versions: '4.1 and up' and '4.0.3 and up'.**
- **Most Unused Android Version : '2.2 and up' and '2.3.3 and up'**
- **Most users have moved on to supporting more recent versions.**

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

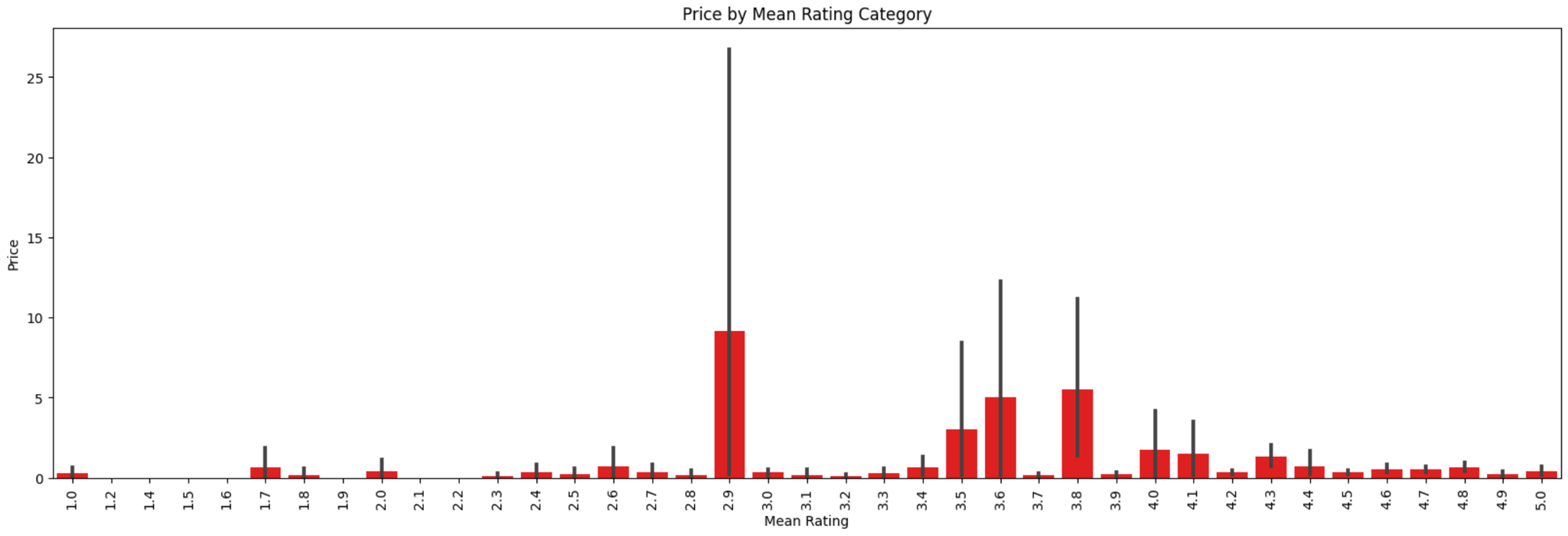
As we check from graph as the android version is upgrading or incresing count of supporting application also increasing. We can say users are using letest version of android on there phones.

Chart - 8

Price Vs Rating Comparision

```
Price_rating = playstore.groupby('Price')['Rating'].mean()


print(Price_rating)

plt.figure(figsize=(20, 6))
sns.barplot(y='Price', x='Rating', data=playstore, color='red')
plt.xlabel('Mean Rating')
plt.ylabel('Price')
plt.title('Price by Mean Rating Category')
plt.xticks(rotation=90)
plt.show()
```

```
Price
0.00      4.186082
0.99      4.300699
1.00      4.400000
1.04      4.300000
1.20      4.200000
            ...
379.99    2.900000
389.99    3.600000
394.99    4.300000
399.99    4.033333
400.00    3.600000
Name: Rating, Length: 92, dtype: float64
```



1. Why did you pick the specific chart?

Answer Here.

2. What is/are the insight(s) found from the chart?

- **Free apps tend to have lower ratings compared to paid apps, it may suggest that users have higher expectations for paid apps.**
- **Bars show a clear trend of higher ratings for higher-priced apps, it suggests that users may associate higher prices with better quality.**
- **There is no clear trend or if free apps have similar ratings to paid apps**
- **We can see that highly expensive apps are not necessarily well rated.**

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

- **Pricing Strategy: If might increase the price if users associate higher prices with better quality.**

We also have one more dataset 'user_review'. For this we will merge with 'playstore' dataset and will find insights from newly merged dataset.

```
user_review.shape

user_review.shape

# Merging 2 datasets

merged_data = pd.merge(playstore,user_review, on='App')
merged_data.shape

# Dropping for NaNs for newly merged dataset.

merged_data.dropna(subset=['Translated_Review'],inplace=True)
merged_data.dropna(subset=['Sentiment'],inplace=True)
merged_data.dropna(subset=['Sentiment_Polarity'],inplace=True)
merged_data.dropna(subset=['Sentiment_Subjectivity'],inplace=True)

merged_data.isna().sum()
merged_data.shape

# Finally will have newly merged dataset.

merged_data.head()
```

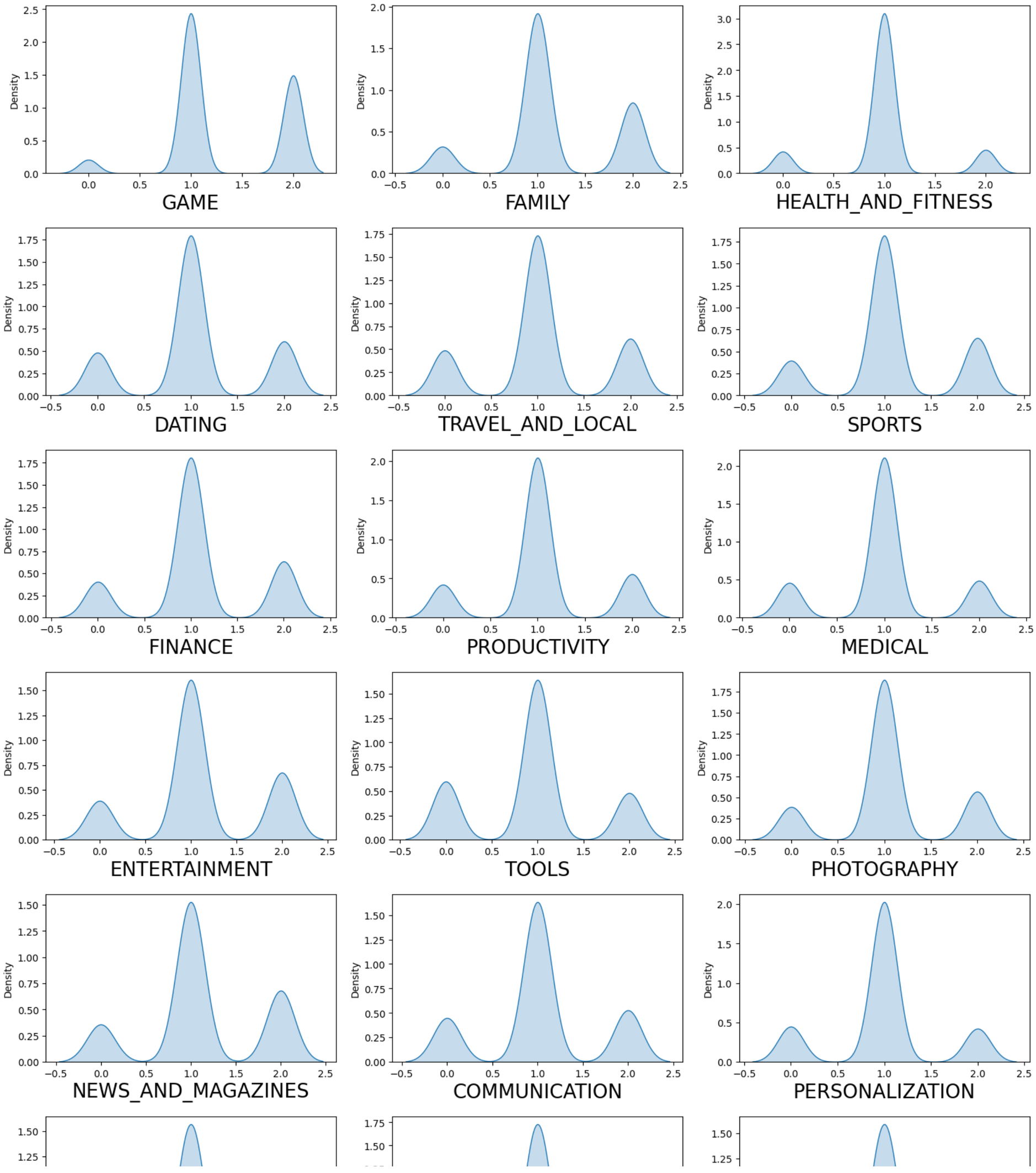
	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver	Translated_Review	Sentiment	Sentiment_Polarity	Sentiment_Subjectivity
0	Coloring book moana	ART_AND_DESIGN	3.9	967	14.0	500000	Free	0.0	Everyone	Art & Design;Pretend Play	2018-01-15	2.0.0	4.0.3 and up	A kid's excessive ads. The types ads allowed a...	Negative	-0.250	1.000000
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14.0	500000	Free	0.0	Everyone	Art & Design;Pretend Play	2018-01-15	2.0.0	4.0.3 and up	It bad >:(	Negative	-0.725	0.833333
2	Coloring book moana	ART_AND_DESIGN	3.9	967	14.0	500000	Free	0.0	Everyone	Art & Design;Pretend Play	2018-01-15	2.0.0	4.0.3 and up	like	Neutral	0.000	0.000000
4	Coloring book moana	ART_AND_DESIGN	3.9	967	14.0	500000	Free	0.0	Everyone	Art & Design;Pretend Play	2018-01-15	2.0.0	4.0.3 and up	I love colors inspyering	Positive	0.500	0.600000
5	Coloring book moana	ART_AND_DESIGN	3.9	967	14.0	500000	Free	0.0	Everyone	Art & Design;Pretend Play	2018-01-15	2.0.0	4.0.3 and up	I hate	Negative	-0.800	0.900000

Chart - 9

Sentiment Analysis for Categories

```
# Firstly will encode for sentiment column to findout relation of Sntiments over various column

encoding_nums = {"Sentiment": {"Negative": 2, "Positive": 1, "Neutral": 0}}
encoding_nums
merged_data = merged_data.replace(encoding_nums)

merged_data.head()

# Ploting KDEPLOT

rating_sentiment = merged_data.groupby('Category')['Sentiment'].sum().sort_values(ascending=False)
rating_sentiment
```

```
plt.figure(figsize=(15, 35))
plotnumber = 1

for category in rating_sentiment.index:
    if plotnumber <= 33:
        ax = plt.subplot(11, 3, plotnumber)
        sns.kdeplot(merged_data[merged_data['Category'] == category]['Sentiment'], shade=True)
        plt.xlabel(category, fontsize=20)
    plotnumber += 1

plt.tight_layout()
plt.show()
```



1. Why did you pick the specific chart?

- **The graph allows you to visualize how sentiment scores are distributed across different categories**
- **This can provide valuable insights into how sentiments vary across various aspects or topics, helping you identify trends or patterns.**
- **This plots can help you to spot unusual sentiment distributions within specific categories.**

2. What is/are the insight(s) found from the chart?

- **Positive Sentiment Categories: Game, Health&Fitness, Education this Categories have more positive sentiment and have less negative sentiment.**
- **Negative Sentiment Categories: Beauty, Video_playes,Lifestyle, News and Magzine have more negative sentiment compaired to others**

3. Will the gained insights help creating a positive business impact?

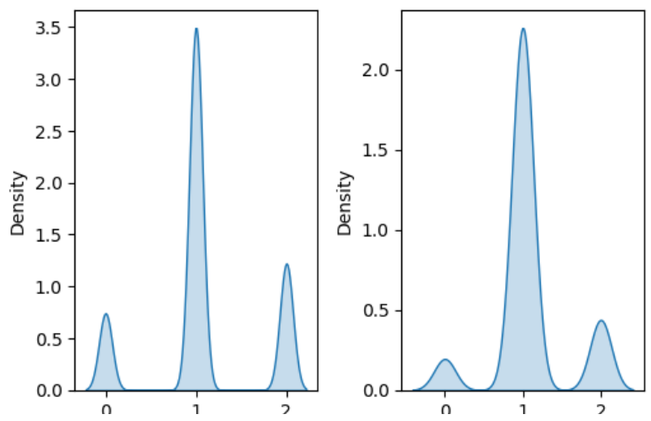
Are there any insights that lead to negative growth? Justify with specific reason.

- **Certain categories consistently have high positive sentiment scores and are positively skewed in the KDE plots, this insight can help the business focus on and invest more in those areas.**
- **Identifying areas with consistently negative sentiment or unusual patterns, the business can take targeted actions to improve the customer experience in those categories.**

Chart - 10

Sentiment Analysis for Type(Free and Paid)

```
type_sentiment = merged_data.groupby('Type')['Sentiment'].sum().sort_values(ascending=False).head(10)
type_sentiment


plt.figure(figsize=(13,7))
plotnumber = 1

for type_label in type_sentiment.index:
    if plotnumber <10:
        ax = plt.subplot(2, 5, plotnumber)
        sns.kdeplot(merged_data[merged_data['Type'] == type_label]['Sentiment'], shade=True)
        plt.xlabel(type_label, fontsize=20)
    plotnumber += 1

plt.tight_layout()
plt.show()
```



1. Why did you pick the specific chart?

- **The graph allows you to visualize how sentiment scores are distributed across type of app that is for free and paid**
- **This can provide valuable insights into how sentiments vary across various aspects and helping you identify trends or patterns.**
- **This plots can help you to spot unusual sentiment distributions for type of app**

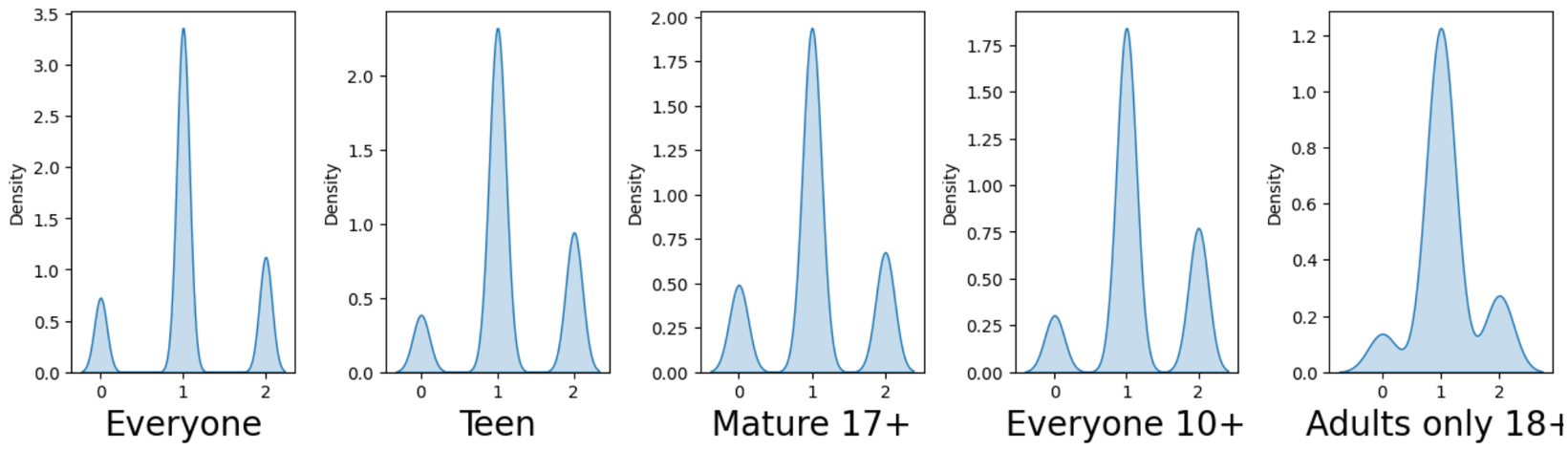
2. What is/are the insight(s) found from the chart?

- **For Free type negative sentimant is more as compaired to paid type.**
- **Also, for Free nutral sentiment is more as compaired to paid type.**
- **Positive sentiment is having more density for free type rather that paid type.**

Chart - 11

Sentiment Analysis for Content Rating

```
ContentRating_sentiment = merged_data.groupby('Content Rating')['Sentiment'].sum().sort_values(ascending=False).head(10)
ContentRating_sentiment


plt.figure(figsize=(13,7))
plotnumber = 1

for Content_Rating_label in ContentRating_sentiment.index:
    if plotnumber <10:
        ax = plt.subplot(2, 5, plotnumber)
        sns.kdeplot(merged_data[merged_data['Content Rating'] == Content_Rating_label]['Sentiment'], shade=True)
        plt.xlabel(Content_Rating_label, fontsize=20)
    plotnumber += 1

plt.tight_layout()
plt.show()
```



1. Why did you pick the specific chart?

- **The graph allows you to visualize how sentiment scores are distributed across Content Rating.**
- **This can provide valuable insights into how sentiments vary across various aspects and helping you identify trends or patterns.**
- **This plots can help you to spot unusual sentiment distributions for Content Rating**

2. What is/are the insight(s) found from the chart?

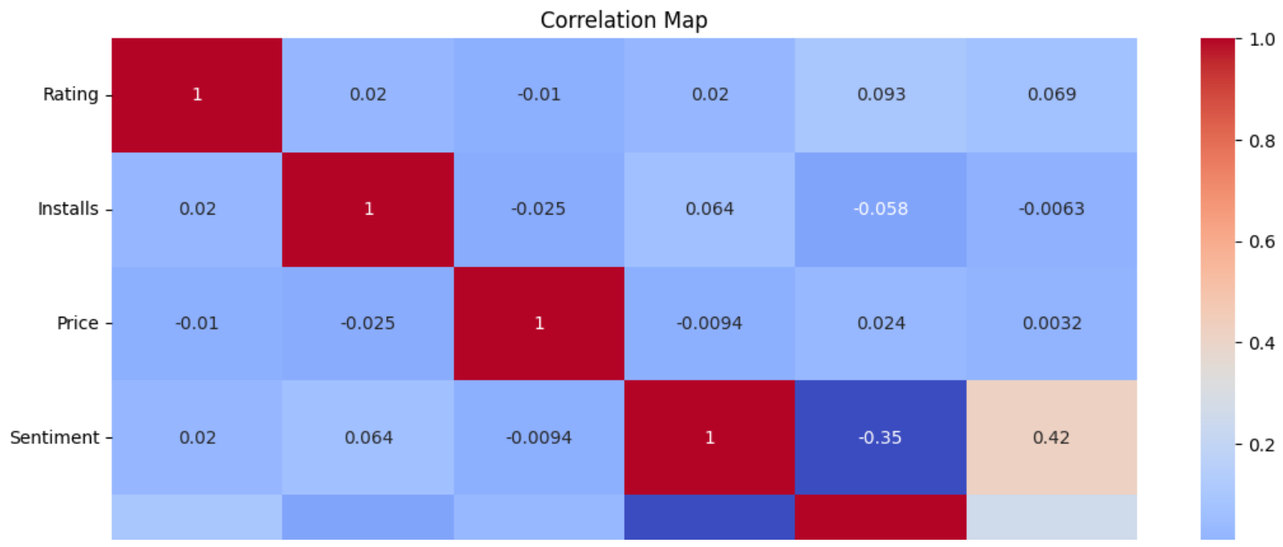
- **Almost for every Content Rating sufficent negative sentiment**
- **For 'Everyone' have more positive sentiment as compaired to others.**

Chart - 13

Correlation Heatmap for Merged dataset

Correlation heatmap can be plot with numerical columns only hence check for numerical columns in dataset

```
# From info there are column like 'Rating','Reviews','Installs','Price','Sentiment','Sentiment_Polarity','Sentiment_Subjectivity' this are numerical type

numerical_merged = ['Rating','Reviews','Installs','Price','Sentiment','Sentiment_Polarity','Sentiment_Subjectivity']

numerical_merged

# built correlaion

correlation_data = merged_data[numerical_merged]

# Correlation Matrix

correlation_matrix = correlation_data.corr()

plt.figure(figsize=(13,7))

sns.heatmap(correlation_matrix,annot=True,cmap='coolwarm')
plt.title('Correlation Map')
plt.show()
```



▼ 1. Why did you pick the specific chart?

This graph will show us correlation between numerical columns from dataset.

▼ 2. What is/are the insight(s) found from the chart?

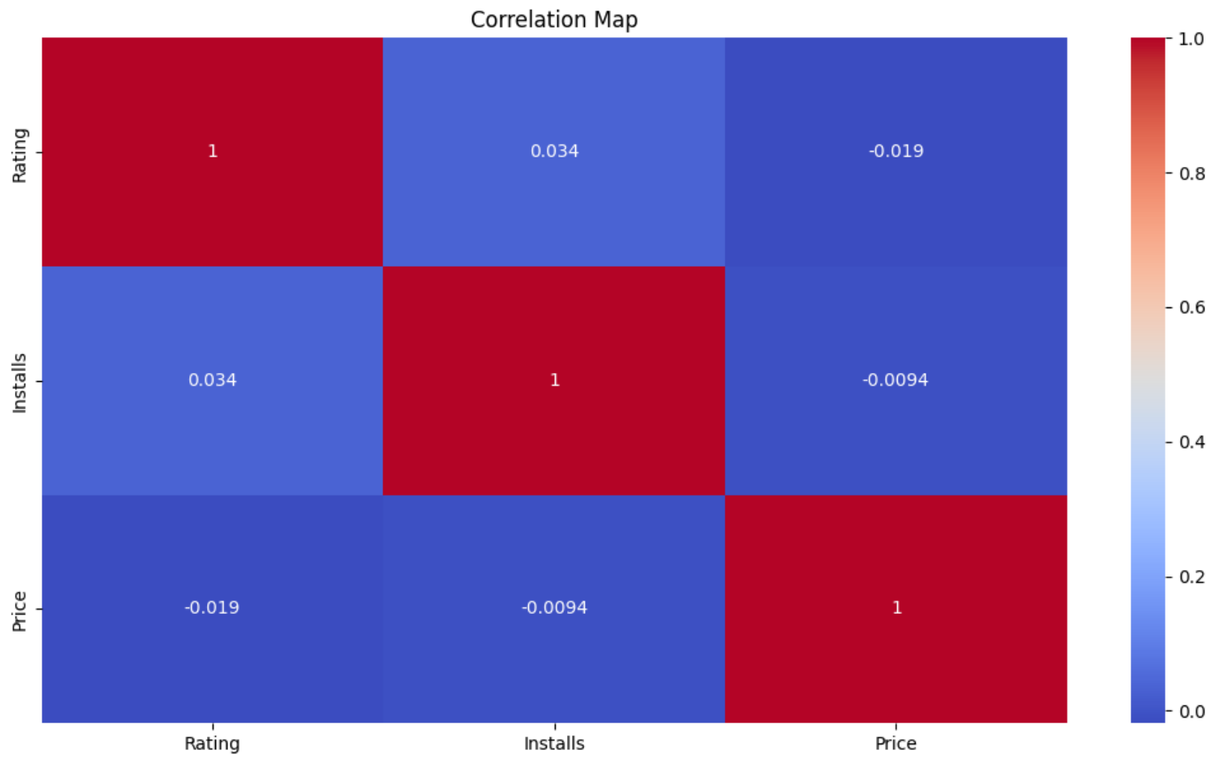
- **Strong Positive Correlations for Reviews and Installs, Sentiment_Polarity and Sentiment_Subjectivity.**
- **Strong Negative Correlations for Price and Rating**

▼ Chart - 14

Correlation Heatmap for playstore Dataset

Correlation heatmap can be plot with numerical columns only hence check for numerical columns in dataset

```
# From info there are column like 'Rating','Reviews','Installs','Price' this are numerical type

numerical_columns = ['Rating','Reviews','Installs','Price']

numerical_columns

# built correlaion

correlation_data = playstore[numerical_columns]

# Correlation Matrix

correlation_matrix = correlation_data.corr()

plt.figure(figsize=(13,7))

sns.heatmap(correlation_matrix,annot=True,cmap='coolwarm')
plt.title('Correlation Map')
plt.show()
```



▼ 1. Why did you pick the specific chart?

This graph will show us correlation between numerical columns from dataset.

▼ 2. What is/are the insight(s) found from the chart?

From above graph got below insights,

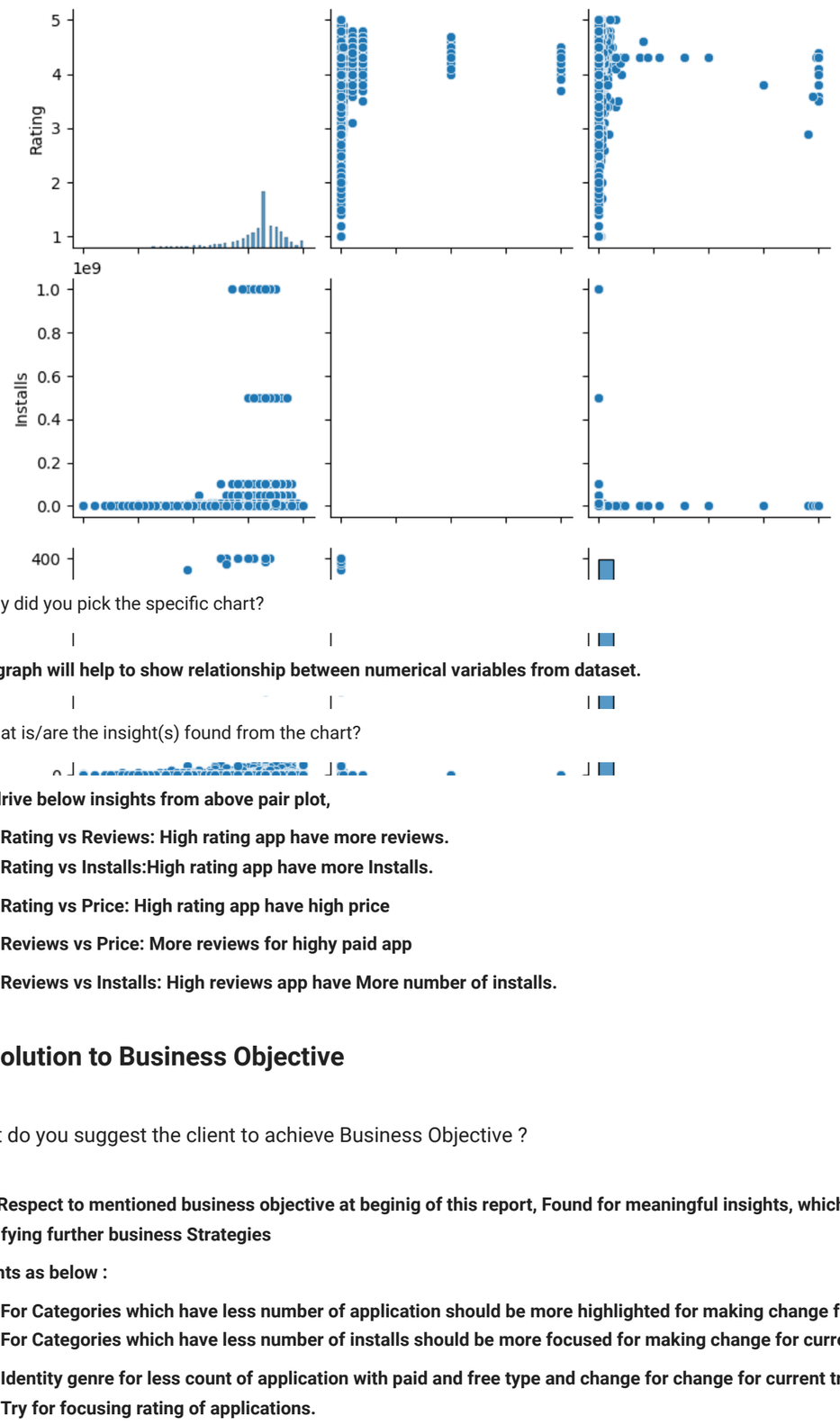
- **Positive Correlation between Reviews and Installs**
- **No Strong Correlation with Rating**
- **Price and Reviews are Weakly Correlated.**

▼ Chart - 15 - Pair Plot

Pair Plot for playstore dataset

```
numerical_columns = ['Rating','Reviews','Installs','Price']

numerical_columns

pairplot_data = playstore[numerical_columns]
plt.figure(figsize=(10,8))
sns.pairplot(pairplot_data)
plt.show()
```

```
<Figure size 1000x800 with 0 Axes>
```



1. Why did you pick the specific chart?

This graph will help to show relationship between numerical variables from dataset.

2. What is/are the insight(s) found from the chart?

Can drive below insights from above pair plot,

- **Rating vs Reviews: High rating app have more reviews.**
- **Rating vs Installs:High rating app have more Installs.**
- **Rating vs Price: High rating app have high price**
- **Reviews vs Price: More reviews for highy paid app**
- **Reviews vs Installs: High reviews app have More number of installs.**

5. Solution to Business Objective

What do you suggest the client to achieve Business Objective ?

With Respect to mentioned business objective at beginig of this report, Found for meaningful insights, which will be helpful to clint for identifying further business Strategies

Insights as below :

- **For Categories which have less number of application should be more highlighted for making change for current trend.**
- **For Categories which have less number of installs should be more focused for making change for current trend.**
- **Identity genre for less count of application with paid and free type and change for change for current trend.**
- **Try for focusing rating of applications.**
- **For Categories which have lowrating should be more highlighted for making change for current trend.**
- **Focus for negative and nuetral sentiment of Categories, rating and type of application.**

Conclusion

Overall Conclusion from entire report

1. **Most Popular Categories: Family, Games, tools.**
2. **Most Installed Categories: Family, Games, tools**
3. **Proportion of Free and Paid Apps: Free apps 92% and Paid Apps 8%. Comparison of App Types:"Free" segment is much larger than the "Paid" segment, it indicates that there are more free apps available in the dataset.**
4. **Most Popular Genre:Tools,Entertaainment,Education.**
5. **Top 3 Genre for Free apps : Medical,Personalization and Tools**
6. **Top 3 Genre for Paid apps : Tools,Entertainment and Education**
7. **Chart 5 can get insight that, more number of applications rating lies near 4 to 4.5. Very Few application with rating from 1 to 3.**
8. **Top Rated Categories: Family, Game and Tools.**
9. **There are 82% applications which are belong to 'Everyone' while 11%,4% belogs to 'teen' and 'Mature17+' Content Rating. There ae very less applications for 'Adults only 18+'.**
10. **Most used Android Versions: '4.1 and up' and '4.0.3 and up'.**
11. **Free apps tend to have lower ratings compared to paid apps, it may suggest that users have higher expectations for paid apps.There is no clear trend or if free apps have similar ratings to paid apps.**
12. **Positive Sentiment Categories: Game, Health&Fitness, Education this Categories have more positive sentiment and have less negative sentiment.**
13. **Negative Sentiment Categories: Beauty, Video_playes,Lifestyle, News and Magzine have more negative sentiment compaired to others**
14. **For Free type negative sentimant is more as compaired to paid type.for Free nutral sentiment is more as compaired to paid type.**
15. **Almost for every Content Rating sufficent negative sentiment**
16. **Strong Positive Correlations for Reviews and Installs, Sentiment_Polarity and Sentiment_Subjectivity.**
17. **Positive Correlation between Reviews and Installs**
18. **Rating vs Reviews: High rating app have more reviews**
19. **Reviews vs Installs: High reviews app have More number of installs**

Thank-you