Unveiling the Android App Market: Analyzing Google Play Store Data

Clean, categorize, and visualize Google Play Store data to understand app market dynamics.

Will gain insights into the Android app market by leveraging data analytics, visualization, and enhanced interpretation skills.

Import Library

```
import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns
```

Import Dataset

```
gp=pd.read_csv('/content/drive/MyDrive/kaggle_API/apps.csv')
```

Displaying first 10 rows

```
gp.head(10)
```

	Unnamed: 0	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Curren Ve
0	0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19.0	10,000+	Free	0	Everyone	Art & Design	January 7, 2018	1.0
1	1	Coloring book moana	ART_AND_DESIGN	3.9	967	14.0	500,000+	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0
2	2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7	5,000,000+	Free	0	Everyone	Art & Design	August 1, 2018	1.2
3	3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25.0	50,000,000+	Free	0	Teen	Art & Design	June 8, 2018	Varie wi devic
4	4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8	100,000+	Free	0	Everyone	Art & Design;Creativity	June 20, 2018	1
5	5	Paper flowers instructions	ART_AND_DESIGN	4.4	167	5.6	50,000+	Free	0	Everyone	Art & Design	March 26, 2017	

Next steps: View recommended plots

Data Preparation

Checking for any missing values

```
gp.isnull().sum()
```

```
Unnamed: 0          0
App                 0
Category            0
Rating           1463
Reviews             0
Size             1227
Installs            0
Type                0
Price               0
Content Rating      0
Genres              0
Last Updated        0
Current Ver         8
Android Ver         2
dtype: int64
```

Now handling the missing values

for example: filling with median for numerical columns,mode for categorial columns

```
gp['Rating'].fillna(gp['Rating'].median(),inplace=True)
```

```
gp['Current Ver'].fillna('unknown',inplace=True)
```

```
gp['Android Ver'].fillna('unknown',inplace=True)
```

Ensuring correct data types

```
gp['Reviews']=gp['Reviews'].astype('int')
```

```
gp['Price']=gp['Price'].apply (lambda x: x.replace('$','')if isinstance(x,str)else x).astype('float')
```

```
gp['Installs']=gp['Installs'].apply (lambda x: x.replace('+','').replace(',','')if isinstance(x,str)else x).astype(int)
```

Now we will remove the duplicates

```
gp.drop_duplicates(inplace=True)
```

```
gp.head()
```

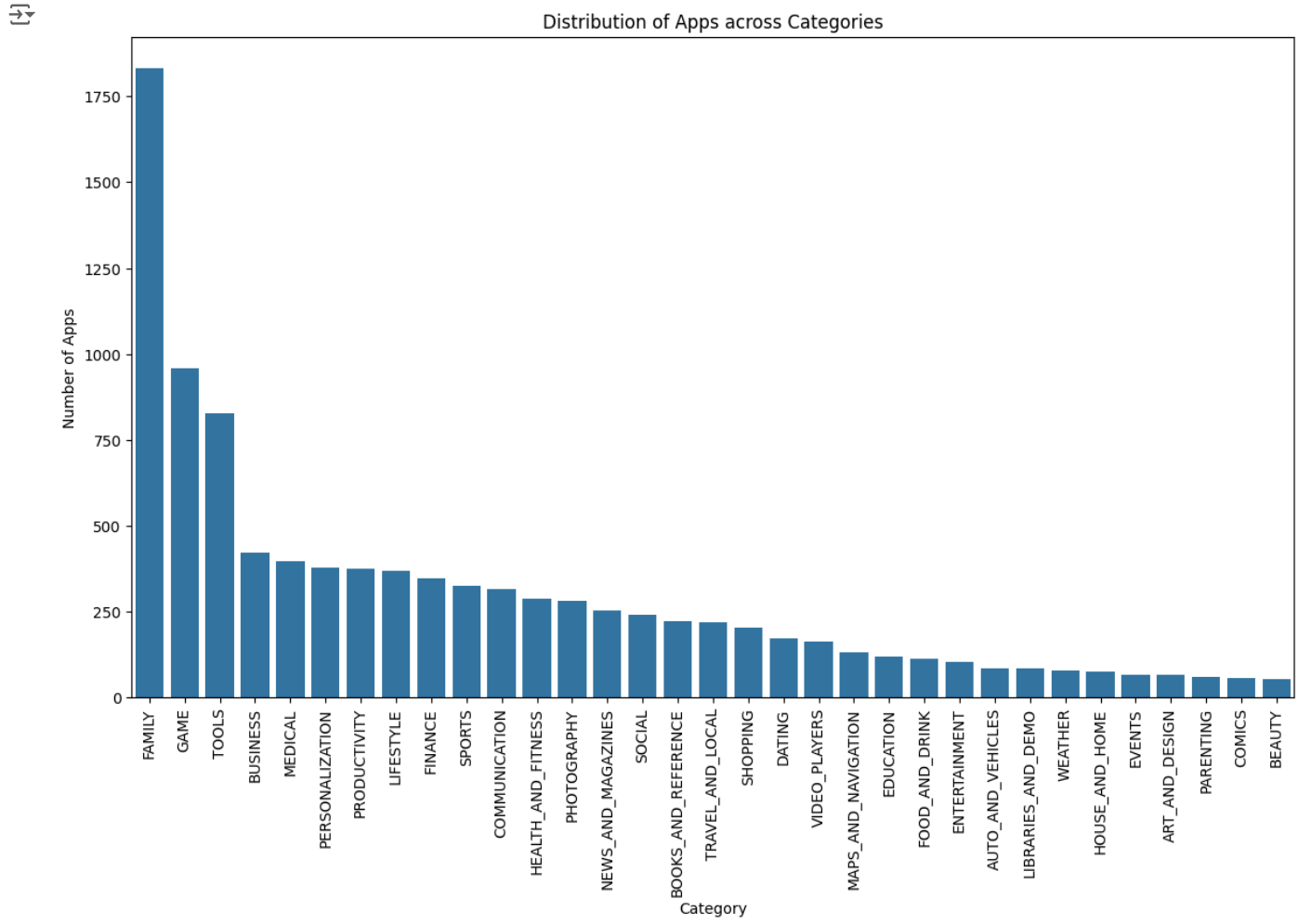
	Unnamed: 0	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	A
0	0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19.0	10000	Free	0.0	Everyone	Art & Design	January 7, 2018	1.0.0	
1	1	Coloring book moana	ART_AND_DESIGN	3.9	967	14.0	500000	Free	0.0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	
2	2	U Launcher Lite – FREE Live Cool Themes	ART_AND_DESIGN	4.7	87510	8.7	5000000	Free	0.0	Everyone	Art & Design	August 1, 2018	1.2.4	

Next steps: View recommended plots

Categorial Exploration

Plotting the distribution of apps across categories

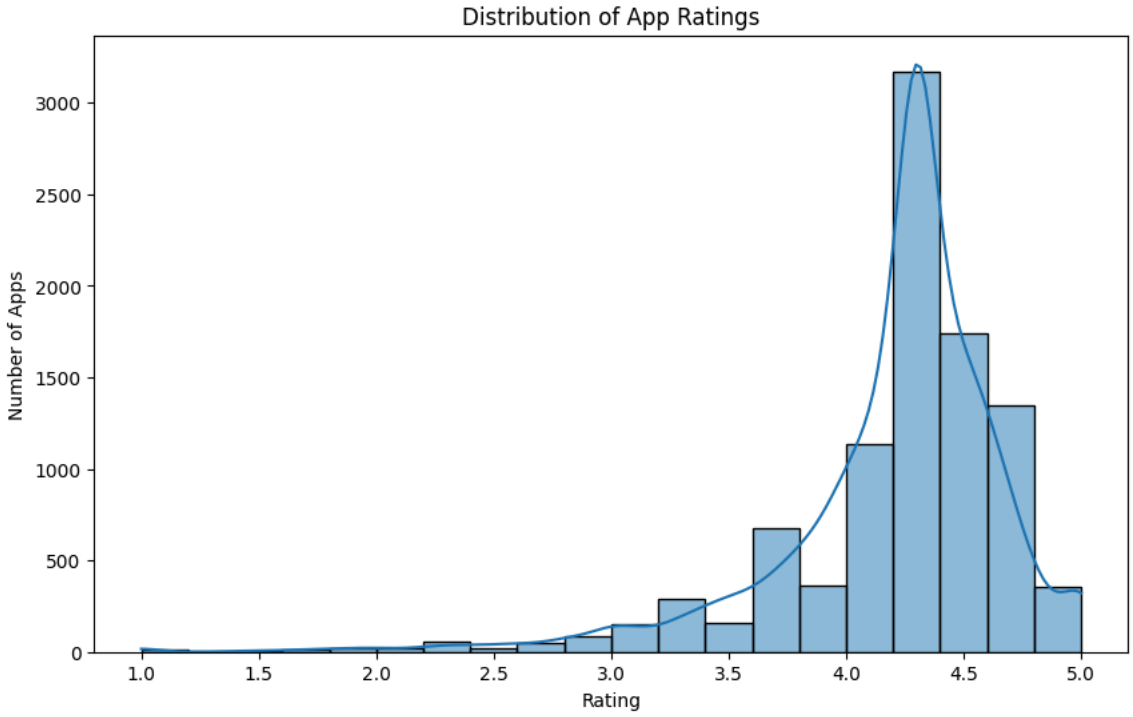
```
plt.figure(figsize=(14,8))
sns.countplot(x='Category',data=gp,order=gp['Category'].value_counts().index)
plt.title('Distribution of Apps across Categories')
plt.ylabel('Number of Apps')
plt.xlabel('Category')
plt.xticks(rotation=90)
plt.show()
```



Metrics Analysis

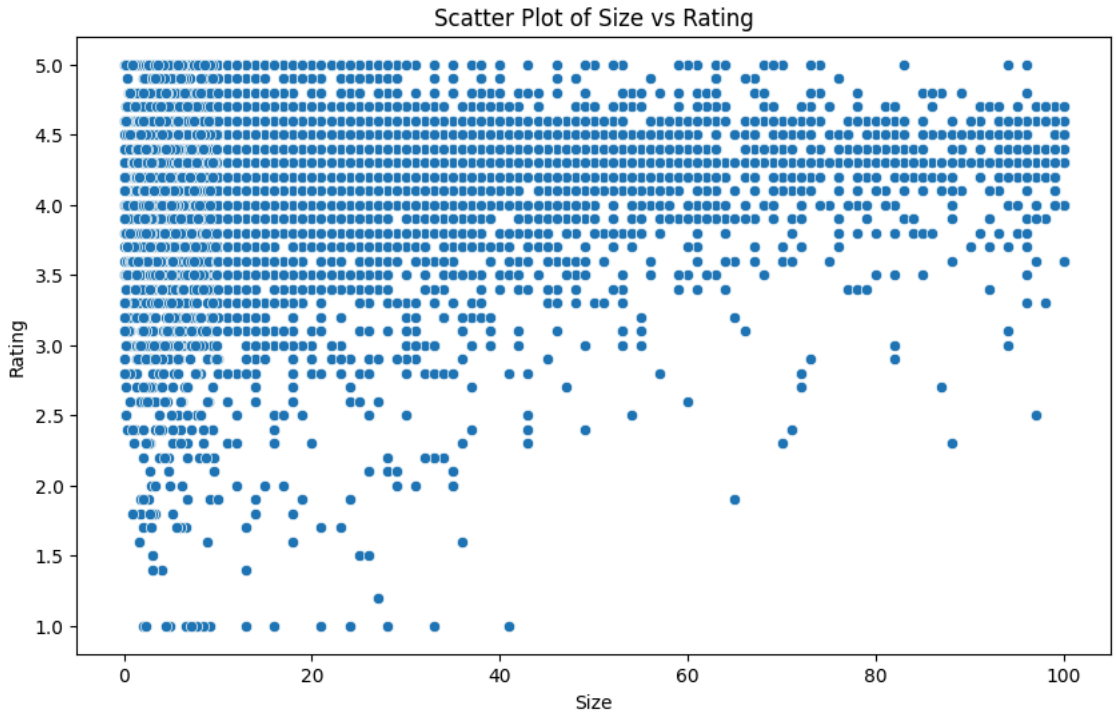
Distribution of app ratings

```
plt.figure(figsize=(10,6))
sns.histplot(gp['Rating'],bins=20,kde=True)
plt.title('Distribution of App Ratings')
plt.xlabel('Rating')
plt.ylabel('Number of Apps')
plt.show()
```



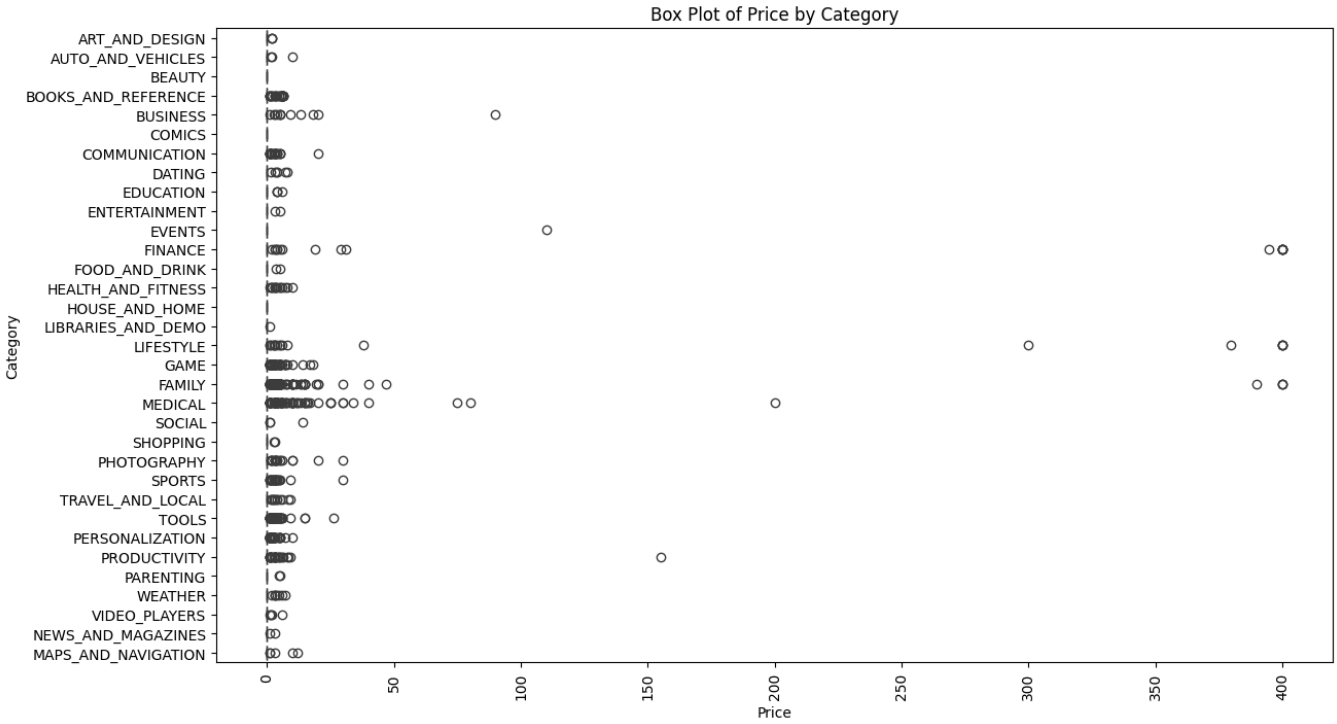
Scatter Plot of size vs rating

```
plt.figure(figsize=(10,6))
sns.scatterplot(x='Size',y='Rating',data=gp)
plt.title('Scatter Plot of Size vs Rating')
plt.xlabel('Size')
plt.ylabel('Rating')
plt.show()
```



Box plot of price by category

```
plt.figure(figsize=(14,8))
sns.boxplot(x='Price',y='Category',data=gp)
plt.title('Box Plot of Price by Category')
plt.ylabel('Category')
plt.xlabel('Price')
plt.xticks(rotation=90)
plt.show()
```



Sentiment Analysis

Functions to calculate sentiment polarity

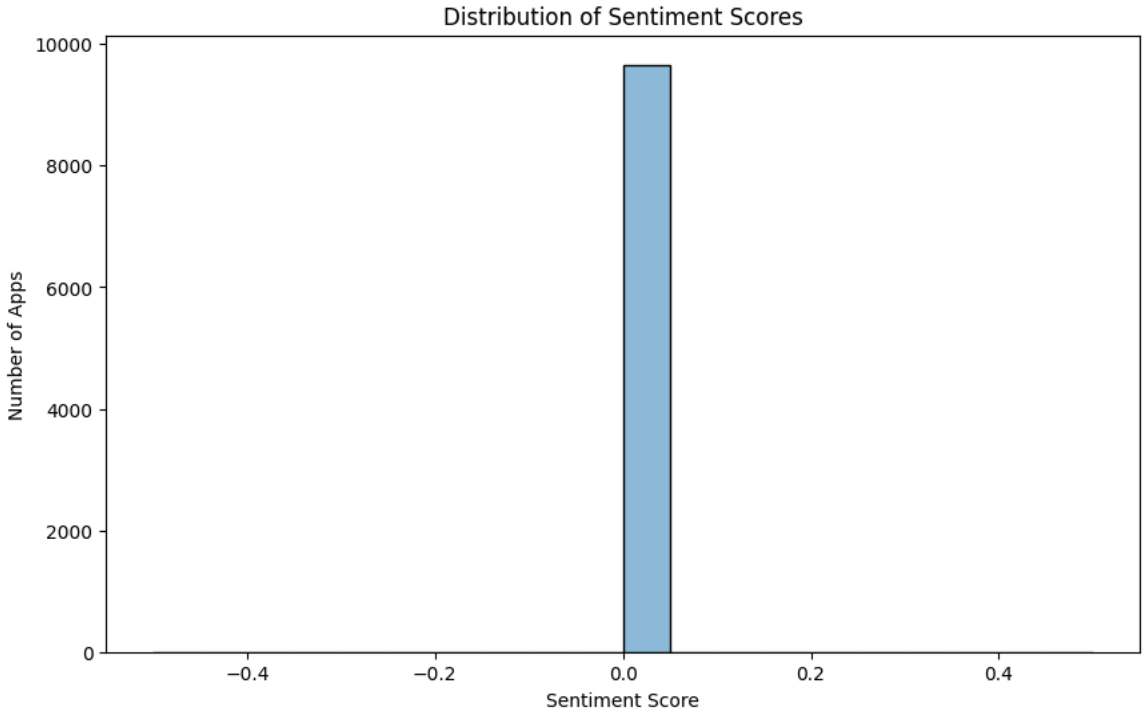
```
def get_sentiment(review):
    analysis= TextBlob(review)
    return analysis.sentiment.polarity
```

Apply sentiment analysis on reviews

```
gp['Sentiment']=gp['Reviews'].apply(lambda x: get_sentiment(str(x)))
```

Plotting the distribution of sentiment scores

```
plt.figure(figsize=(10,6))
sns.histplot(gp['Sentiment'],bins=20,kde=True)
plt.title('Distribution of Sentiment Scores')
plt.xlabel('Sentiment Score')
plt.ylabel('Number of Apps')
plt.show()
```

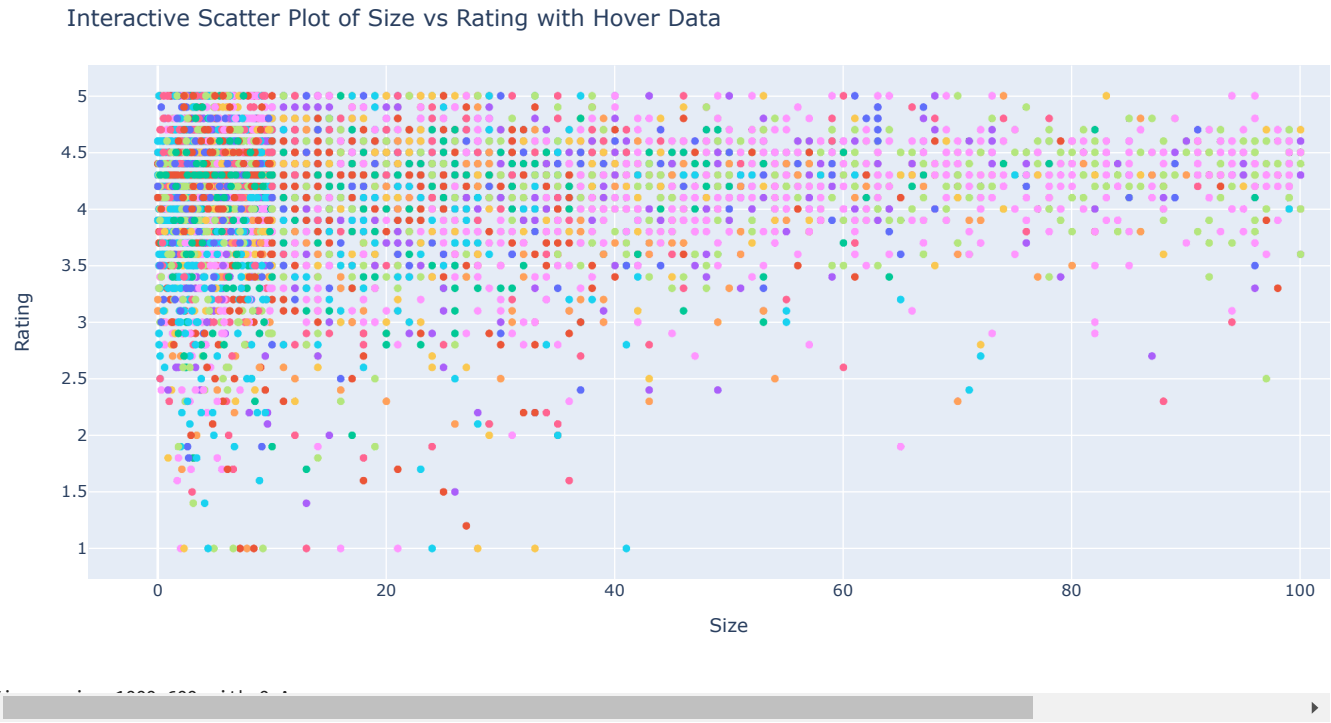


Interactive visualization

```
import plotly.express as px
```

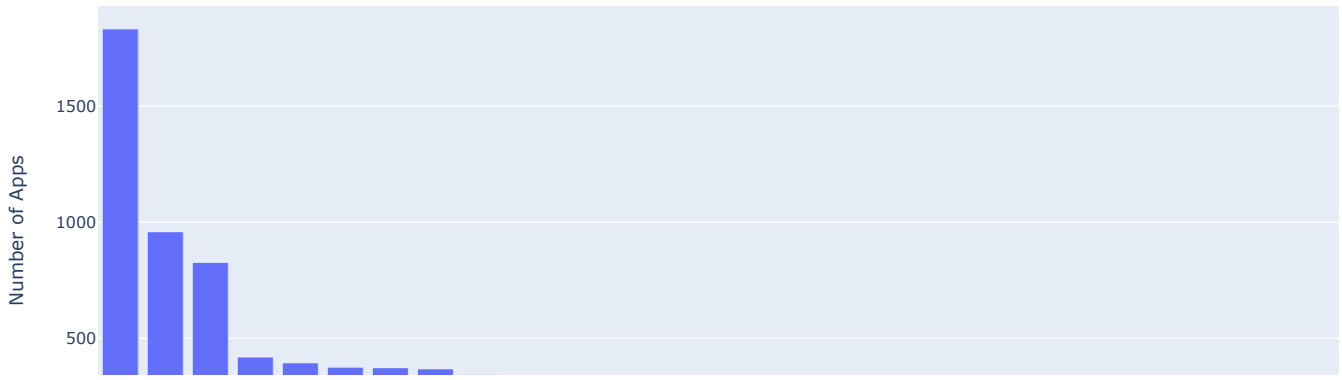
Interactive scatter plot of size vs. rating with hover data

```
plt.figure(figsize=(10,6))
fig=px.scatter(gp,x='Size',y='Rating',color='Category',hover_data=['App','Installs'])
fig.update_layout(title='Interactive Scatter Plot of Size vs Rating with Hover Data')
fig.show()
```



Interactive bar chart of app distribution across categories

```
fig= px.bar(gp['Category'].value_counts().reset_index(), x='Category', y='count', labels={'Category':'Category', 'count': 'Number of App
fig.update_layout(title='Interactive Bar Chart of App Distribution Across Categories')
fig.show()
```


Interactive Bar Chart of App Distribution Across Categories
Number of Apps
1500
1000
500