MINI PROJECT

Project Title: Google Play Store Apps Data Analysis

Group details :

1. *Soumen Samanta (16010420133)*
2. *Himanshu Nimonkar (16010420139)*
3. *Omkar Karbhari (16010420141)*

Introduction:

It's difficult to manage the constantly shifting mobile landscape. Mobile's share of the market is steadily growing. For developers to work on and capture the Android market, actionable insights may be gathered. Mobile app analytics are an excellent approach to learn more about your current strategy for increasing user retention and growth.

Problem statement :

We plan to utilise Python NumPy, Python Pandas, Python Matplotlib, and a few more libraries to analyse and answer questions in this project. Hundreds of thousands of facts about programmes accessible on Google Play Store are contained in the data we selected.

About the Dataset:

The dataset contains the following information from 2008 to 2018:-

1. *App* – Name of the App
2. *Category* – The Category under which the App falls
3. *Rating* – Rating of the App on Play Store
4. *Reviews* – Number of user reviews for that App
5. *Size* – The of the App in KB
6. *Installs* – The number of downloads of the App
7. *Type* – The Type of App (Free/Paid)
8. *Price* – The Price of the App
9. *Content Rating* – The App is rated for which age group
10. *Genres* – Under which genre does app come under in the particular category
11. *Last Updated* – The date when the App was last updated
12. *Current Ver* – The current version of the App
13. *Android Ver* – The Android version which thee App supports

Why was this dataset selected ?

The data from Play Store applications has a lot of promise for helping app developers succeed. From 2008 to 2018, this dataset contains a diverse group of applications from many categories, providing developers with a comprehensive picture of the current condition of apps on the Play Store.

Some of the questions answered: -

- Which is the most popular category ?
- What are the various content ratings and their percentages ?
- Which type of app ‘paid ‘ or ‘free’ is more prevalent on Google Play store ?
- Which category has the maximum number of paid apps ?
- Which apps have maximum number of installs ?
- Which is the most reviewed app ?
- Which are the most popular apps in family category ?
- Which are the most popular apps in games category ?
- Which android version is most compatible with apps on google play store ?

Determining the outcomes from the dataset:-

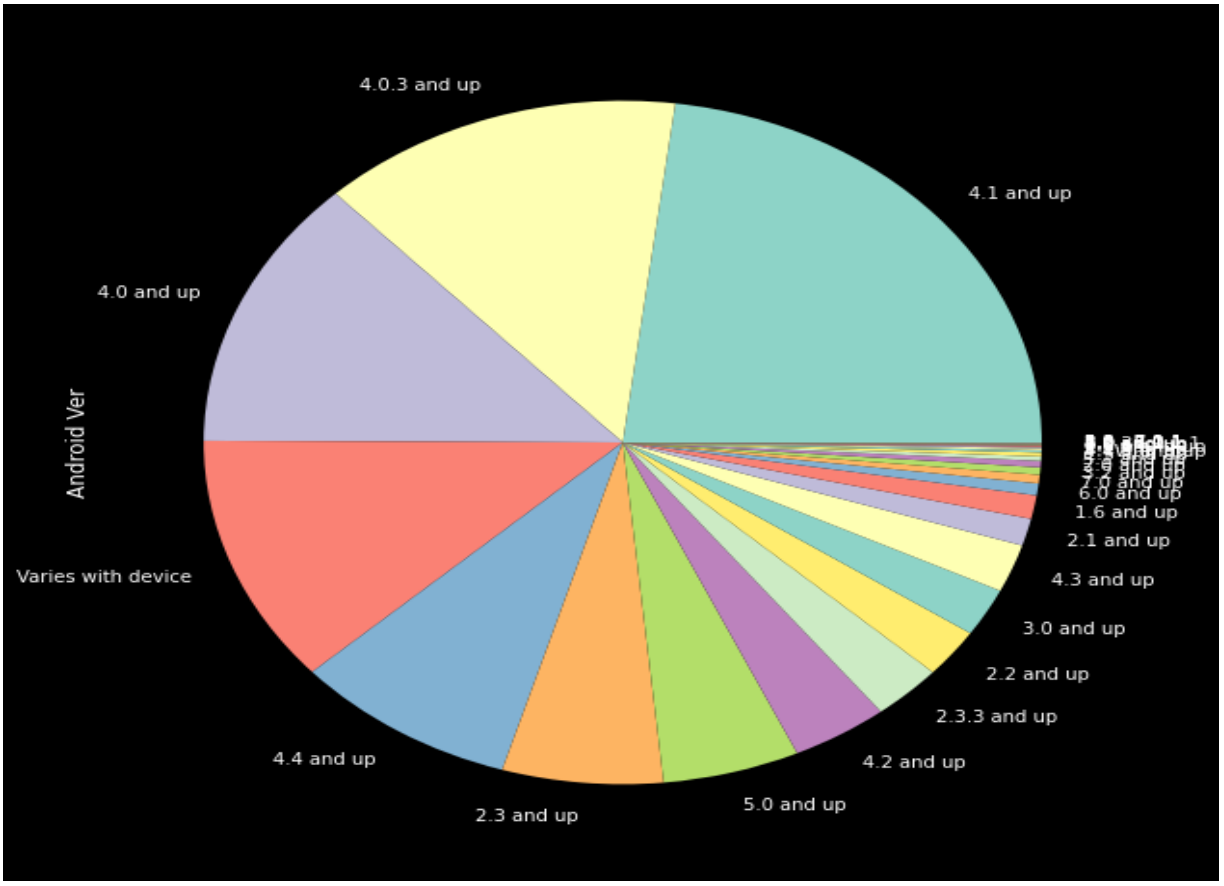
Pandas was used to answer the questions, while Matplotlib and Seaborn tools were used to create graphs and charts.

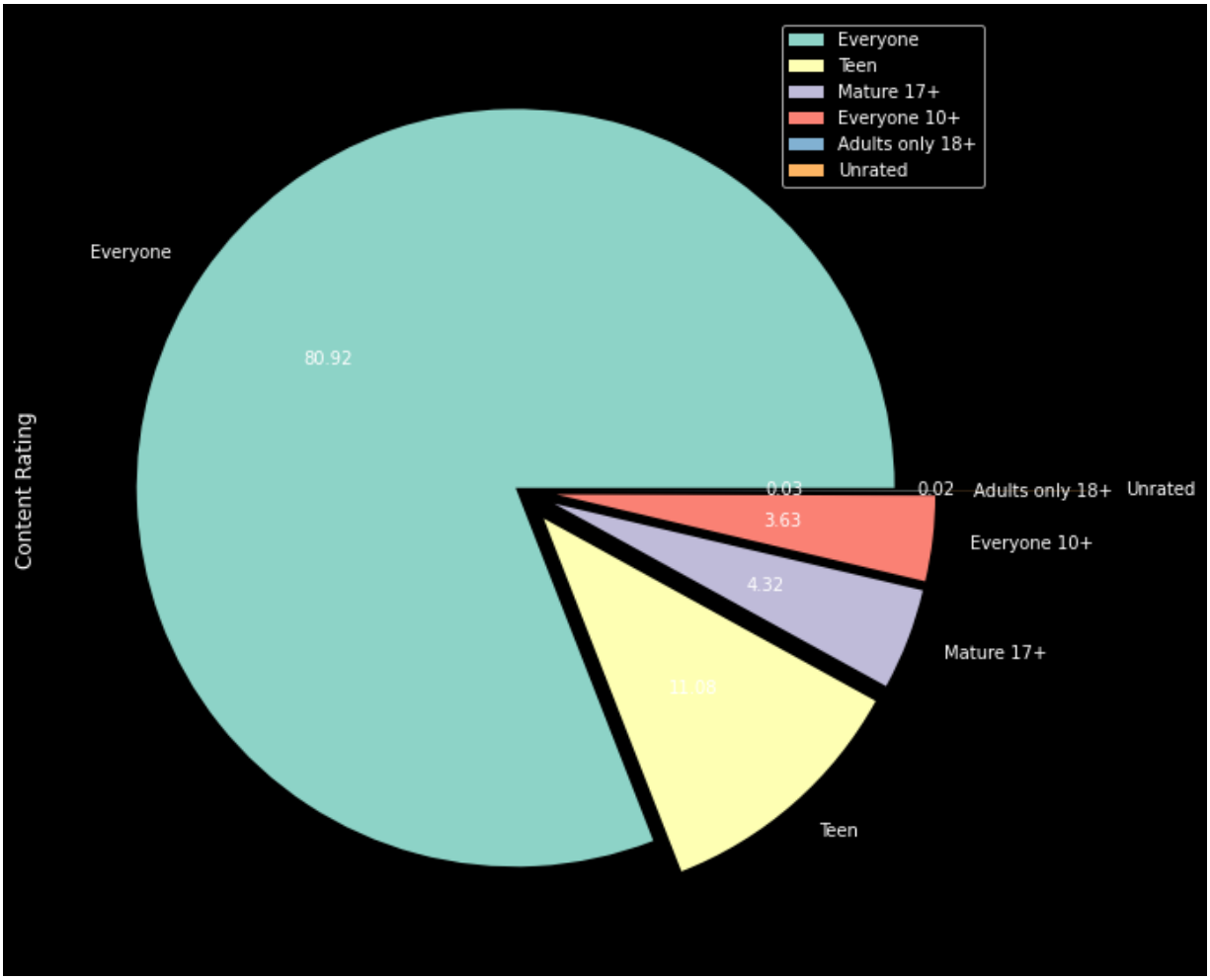
Wherever possible, we've utilised bar graphs and pie charts to properly depict our findings. These graphs aid in the visualisation of our results and the analysis of the data.

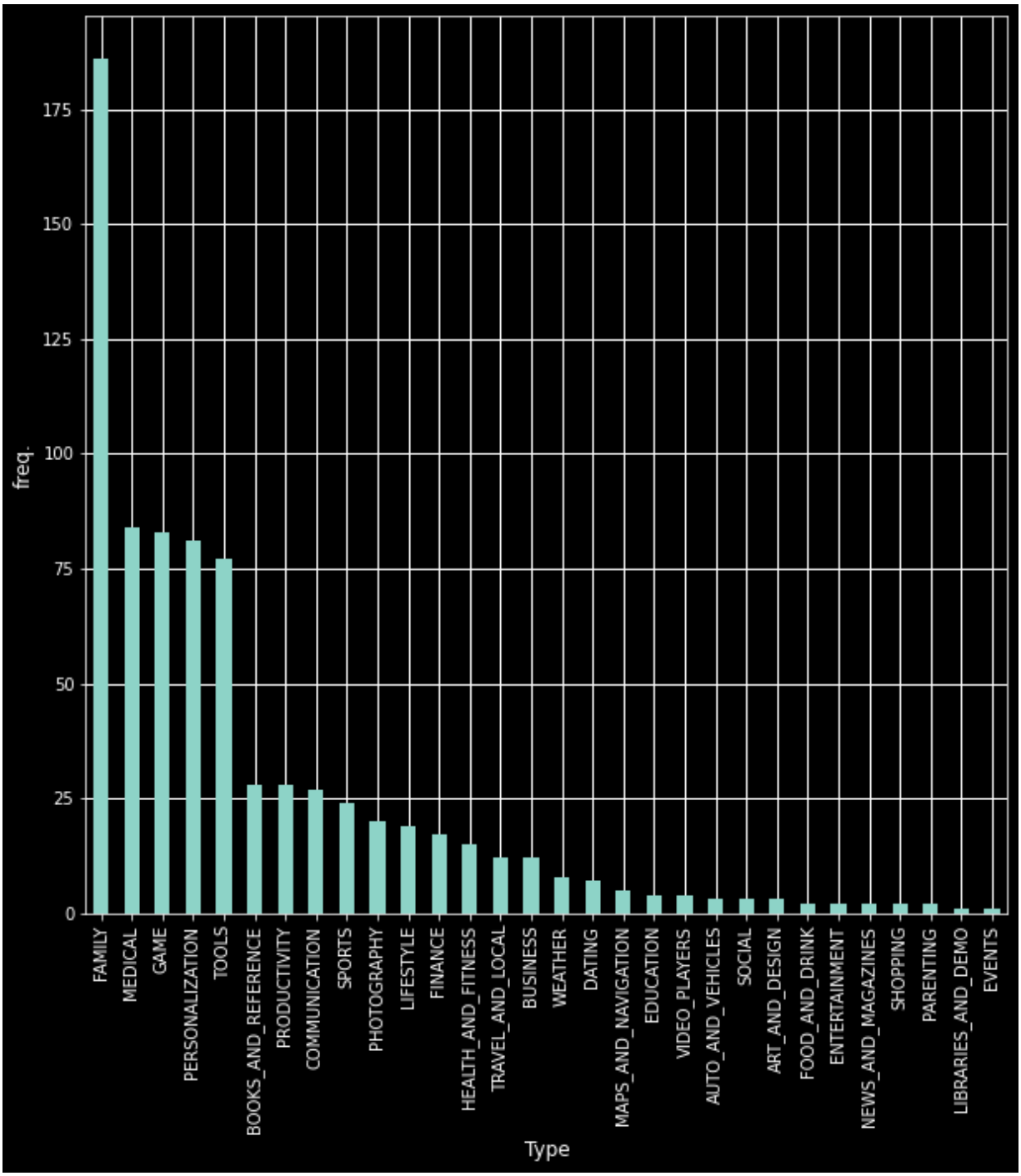
Additional Analysis:-

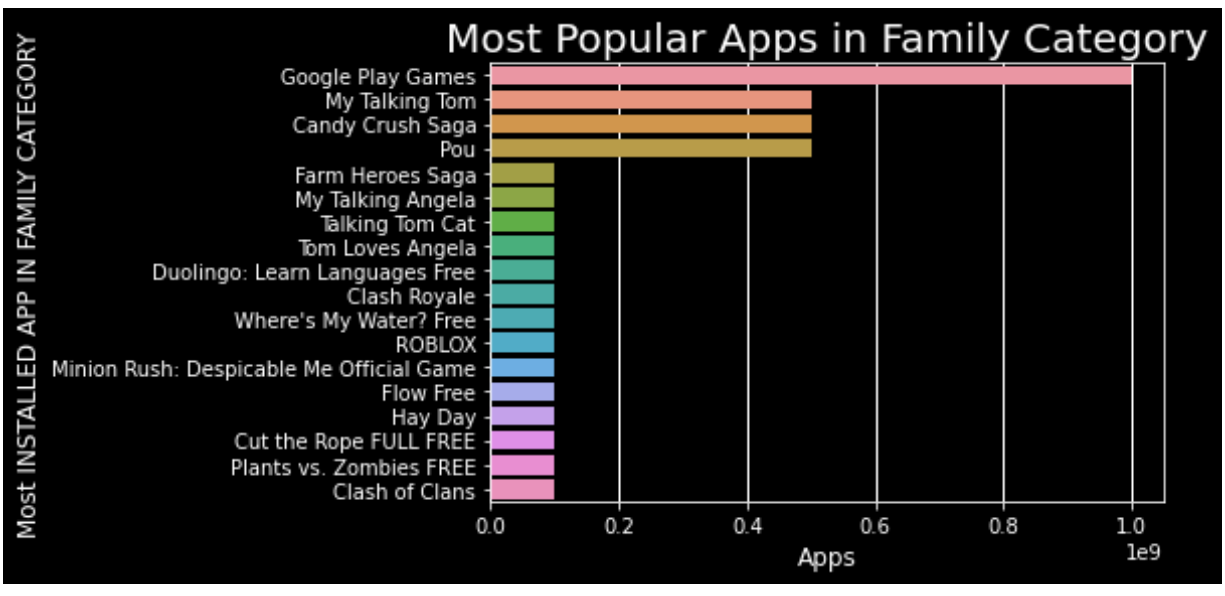
Additional Charts and Graphs have been supplied to clearly show which categories of Apps are the most popular, so that developers may utilise this knowledge to create Apps with greater potential.

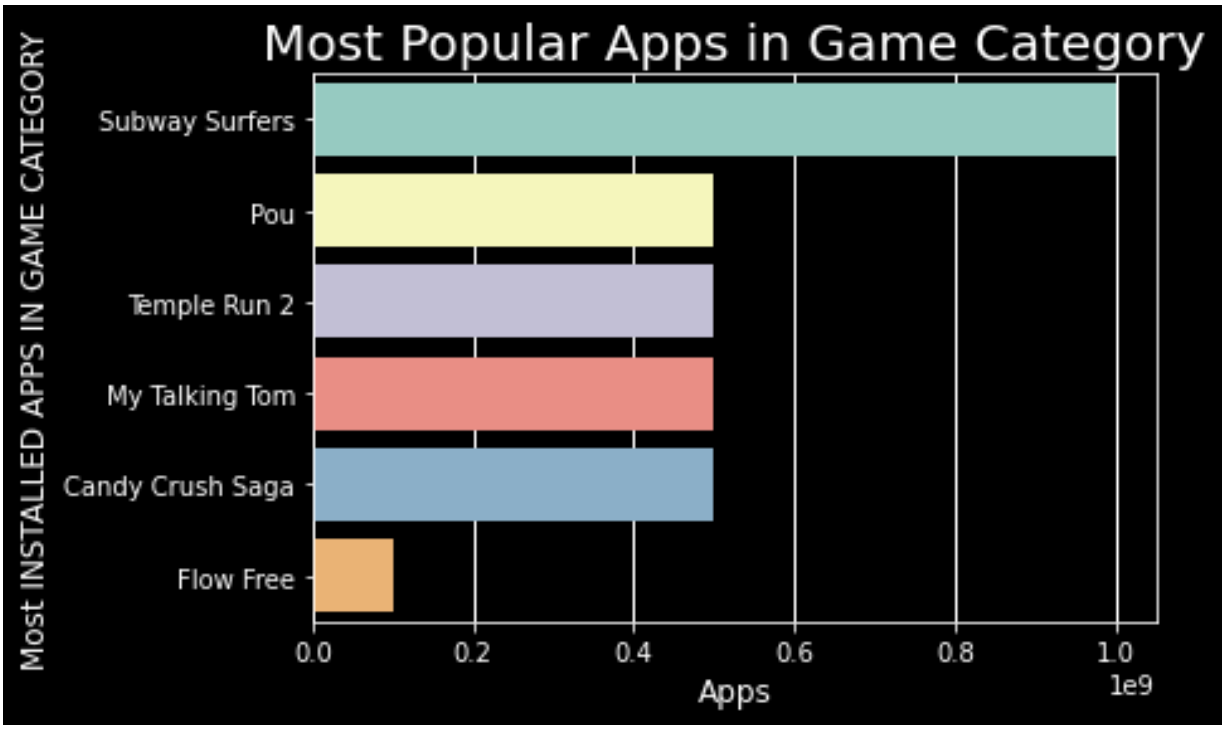
Functionality: -






freq.
0
25
50
75
100
125
150
175
FAMILY
MEDICAL
GAME
PERSONALIZATION
TOOLS
BOOKS_AND_REFERENCE
PRODUCTIVITY
COMMUNICATION
SPORTS
PHOTOGRAPHY
LIFESTYLE
FINANCE
HEALTH_AND_FITNESS
TRAVEL_AND_LOCAL
BUSINESS
WEATHER
DATING
MAPS_AND_NAVIGATION
EDUCATION
VIDEO_PLAYERS
AUTO_AND_VEHICLES
SOCIAL
ART_AND_DESIGN
FOOD_AND_DRINK
ENTERTAINMENT
NEWS_AND_MAGAZINES
SHOPPING
PARENTING
LIBRARIES_AND_DEMO
EVENTS
Type


Most Popular Apps in Family Category
Most INSTALLED APP IN FAMILY CATEGORY
Google Play Games
My Talking Tom
Candy Crush Saga
Pou
Farm Heroes Saga
My Talking Angela
Talking Tom Cat
Tom Loves Angela
Duolingo: Learn Languages Free
Clash Royale
Where's My Water? Free
ROBLOX
Minion Rush: Despicable Me Official Game
Flow Free
Hay Day
Cut the Rope FULL FREE
Plants vs. Zombies FREE
Clash of Clans
0.0
0.2
0.4
0.6
0.8
1.0
1e9
Apps



In [169…

```
# 'Rating' is cleaned.
t_data = data[pd.notnull(data['Rating'])]

# Mean is found
mean = np.mean(t_data['Rating'])

print("Mean of ratings excluding null values:", mean)
```

```
Mean of ratings excluding null values: 4.189657838778271
```

In [170…

```
# NaN is replaced with mean because there are many missing values in 'Rating' column.
data['Rating'].fillna(round(mean, 1), inplace = True)

# Missing value is not considerable in other columns so the NaN values are dropped.
data.dropna(inplace = True)
```

In [171…

```
# All are non-null here as we can observe.
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 10346 entries, 0 to 10345
Data columns (total 13 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   App             10346 non-null  object 
 1   Category        10346 non-null  object 
 2   Rating          10346 non-null  float64
 3   Reviews         10346 non-null  int64  
 4   Size            10346 non-null  float64
 5   Installs        10346 non-null  int64  
 6   Type            10346 non-null  object 
 7   Price           10346 non-null  float64
 8   Content Rating  10346 non-null  object 
 9   Genres          10346 non-null  object 
 10  Last Updated    10346 non-null  object 
 11  Current Ver     10346 non-null  object 
 12  Android Ver     10346 non-null  object 
dtypes: float64(3), int64(2), object(8)
memory usage: 1.1+ MB
```

Also the most occuring price is 0.99 dollars which is also the minimum price, hence we can say that most paid apps are inexpensive.

In [184…

```
# Finding the minimum, maximum, average, median, range, mode, count and variance of th
df = data
maximum = np.max(df['Installs'])
minimum = np.min(df['Installs'])
avg = np.mean(df['Installs'])
med = np.median(df['Installs'])
r = maximum - minimum
var = np.var(df['Installs'])

print("Maximum number of installs of an App on Play Store: " + str(maximum))
print("Minimum number of installs of an App on Play Store: " + str(minimum))
print("Average number of installs of an App on Play Store: " + str(round(avg, 2)))
print("Median of number of installs of Apps on Play Store: " + str(round(med, 2)))
print("Range of number of installs of  Apps on Play Store: " + str(r))
print("Variance of number of installs of Apps on Play Store: " + str(round(var, 2)))
```

```
Maximum number of installs of an App on Play Store: 1000000000
Minimum number of installs of an App on Play Store: 0
Average number of installs of an App on Play Store: 14172659.72
Median of number of installs of Apps on Play Store: 100000.0
Range of number of installs of  Apps on Play Store: 1000000000
Variance of number of installs of Apps on Play Store: 6444399933509701.0
```

This data shows us that there exist apps which have no installs.

There is a lot of discrepancy between average and the median value which shows us that the data for number of installs is not increasing linearly.

Thank-You!