Project Report
On
Google Play Store Analysis

IE6600 – Computation and Visualization

Submitted by: Group 13

Team Members

Challagonda, Mrudula (NUID 002765266)

Chodisetti, Mohit (NUID 002766129)

Ghanta, Akash (NUID 002960065)

SVS, Aishwarya (NUID 002774851)

Term and Year: Fall 2022 [Full Term]

Submitted to: Sivarit Sultornsanee

Submitted Date: December 14th, 2022

ACKNOWLEDGEMENT

Apart from the efforts of team, the completion of this project wouldn't have been possible without the inputs and encouragement of the Teaching Assistants. We also take this opportunity to express our gratitude to Professor Sivarit Sultornsanee. This wouldn't have been materialized without his support and the knowledge he shared with us throughout the semester.

Our heartfelt appreciation goes to Northeastern University – College of Engineering for giving us this opportunity to test our knowledge and use it for practical purposes.

CONTENT

PART 1 - INTRODUCTION AND RESEARCH QUESTIONS

In recent times, technology has consumed and changed our aspect on all activities. Every industry – Media, Education, Medicine, Shopping, and Transport etc. are taking new shape and are being digitalized. In today's world, every in-person, interaction required activities can now be done with a click.

Nowadays we have an App for everything, which explains the exponential growth of the App release numbers. From a recent survey it has been observed that there are around 3750 Apps released in the Google play store every day, which are around 112,500 per month.

We as a team, decided to analyze the Google play store download data and understand the underlying trends. Our main objective is to build an interactive dashboard which can bring out the interconnected links among the various available factors of the Google Play Store download data.

We understood the different aspects and the links among the unique features of our dataset and gained interesting insights regarding the android market.

Our main intended target audience is:

- Application Investors
- App Developers

Research Questions

Q1. Which Category of Apps has more number of installs?

Q2. Which App Category has maximum Earning?

Q3. Which App has the highest earning?

Q4. What is the percentage of paid and free apps in the total market?

Q5. App Size vs Number of installs trend to understand what the users prefer.

Q6. Understanding the trend of App rating and download rate.

Q7. Spread of paid and free apps among each of the 'Content Rating' groups.

Q8. Analysis on Latest Update Date trend among each category.

These questions have been answered and additional visualizations have been created too. The detailed explanation regarding the purpose of answering these questions and the results have been explained section wise in Part 4 of this report.

PART 2 – SUMMARY OF RESULTS

- Based on 'Content Rating', ‘Everyone’ category has the highest number of apps.
- There are approximately 2% Paid Apps and 98% Free Apps.
- Action and Arcade are the top 2 categories when it comes to earnings
- The top five apps with the highest earnings found on Google play store are: (Minecraft, Poweramp, OffieSuite, GTA San Andreas, Hitman Sniper)
- Irrespective of Category all apps have seen a peak in the year 2020 with respect to last update date.
- The relationship between App Size and Number of installs is that they are inversely proportional
- Adults use the least number of paid Apps.

CONCLUSION:

So, as mentioned our 2 main target audience can utilize this dashboard for the following reasons.

App Developer:

If a developer is looking to create an App and their aim is to gain high earnings, they can create an ‘Arcade’ App as it’s the one with highest earning. They can also check the spread of the ‘Content Rating’ (Adults, Everyone, Teen) in that particular category. If the App is being created for the “Adult” category and is a Paid App, the chances of it being installed is low as it has been observed that Adults use the least amount of paid apps. For a Paid App to be installed more, it is better to put it in the “Everyone” Category.

Another conclusion that is of great use for the developer is that, The App Size is an important factor to be considered and can decide the success or failure of the App. The relation between App Size and Number of installs is inversely proportional. So, it is better to create an App with median size as it increases the chances of it being downloaded and used.

App Investors:

For a firm or a person to invest in something, they would definitely like to know if they are going to get good returns. Any investor would like to see the expected numbers and know if there's scope for profit. The Earnings Analysis Story that we developed was made to answer such questions.

This Analysis shows which category of Apps is making earnings, and further within that category which Apps are making more money. There is also a price rage analysis box plots for each category showing he median price and the Upper and Lower whiskers. Based on this an average price for paid app of each category can be seen.

With such information and understanding the features of similar Apps the success of a new app could be estimated which was the goal of this dashboard

PART 3 – DATA SOURCES

To analyze the Google play store download data, we tried to look for the right dataset. A dataset that has good number of records without having too many missing fields or corrupt data etc. Any dataset that we find is not usually perfect; the first step of analysis would be cleaning data, but there are certain factors and thresholds to consider when selecting the data.

We obtained our dataset from Kaggle (https://www.kaggle.com/code/sampathkumarlam/google-play-store-analysis/data). Our dataset initially contained 2.3 million records across 23 columns.

Our dataset initially had null values in a couple of columns, records of non-English language Apps, duplicates etc. which had to be cleaned as only with a clean and organized data can we perform analysis and come up with the right conclusions. After cleaning the data, we were left with a dataset that contained 1.8 million records.

Our final cleaned and structured data can be found at the link below.

https://drive.google.com/file/d/195EQ4RPdjJ9i4YKywzbgI9KF9R1gFJXg/view?usp=share_link

The dataset has 48 different categories of Apps and there are 23 columns, each giving information such as rating, price range etc. A detailed description i.e. the data dictionary of the Google Play Store dataset has been tabulated below.

Column Name	Data Type	Description
App Name	object	Name of the App
App Id	object	Unique App ID
Category	object	Category the app belongs to
Rating	float64	Overall user rating of the app
Rating Count	float64	No of ratings received
Installs	object	Number of user downloads/installs
Minimum Installs	float64	Approximate minimum app install count
Maximum Installs	int64	Approximate maximum app install count
Price	float64	App Price
Currency	object	App Currency
Size	float64	Size of application package
Minimum Android	object	Minimum android version supported
Developer Id	object	Developer Id in Google Play store
Developer Website	object	Website of the developer
Developer Email	object	Email-id of developer
Released	object	App launch date on Google Play store
Last Updated	object	Last app update date
Content Rating	object	Maturity level of app
Privacy Policy	object	Privacy policy from developer
Ad Supported	bool	Ad support in app
In App Purchases	bool	In-App purchases in app
Editor's Choice	bool	Whether rated as Editor Choice.
Scraped Time	object	Scraped date-time in GMT

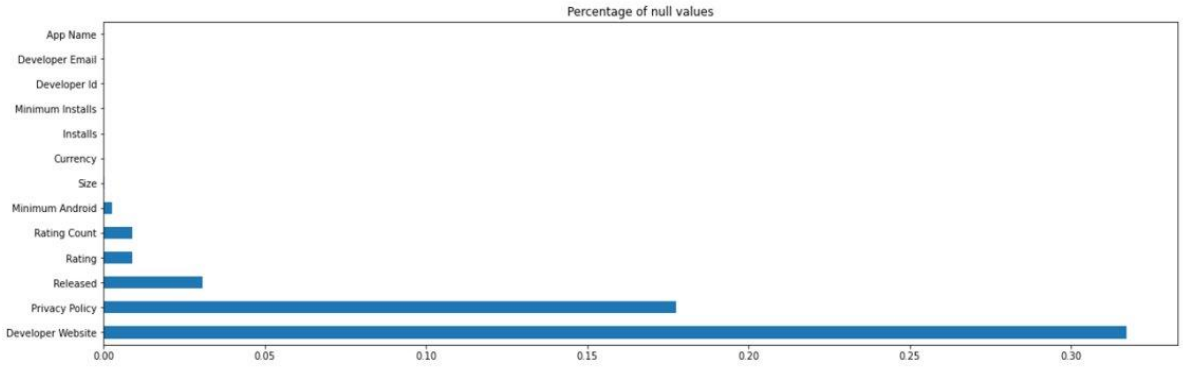
PART 4 – RESULTS AND METHODS

Exploratory Data Analysis:

EDA (Exploratory Data Analysis) is what needs to be done after acquiring the dataset. We transform unsorted raw data into structured data that is usable. Changes to empty columns, deletion of duplicate data, altering the data type in a column, and other actions are subparts of data preparation.

- **Null Value Analysis:**

Although there were no duplicates in the dataset, we saw that the dataset contains many Null or missing values. The column Developer Website, Privacy Policy, Released, Rating, Rating Count, Minimum Android, Size, Currency, Installs, Minimum Installs, Developer Id, Developer Email and App Name contains 760835, 420953, 71053, 22883, 22883, 6530, 196, 135, 107, 107,33,31 and 2 missing values respectively. The percentage of null values in each of the 23 columns is represented in the bar chart below.



- **Non-English Language Data**

We also had Non-English App data in our dataset. However, we only wanted to analyse English-language apps. We used a Gibberish model to clear out other language data. This algorithm detects whether the text is valid or randomly generated. It then returns a percentage value.

A low percentage indicates valid English text and a high one indicates random or other language text. We selected a particular threshold (33.4 to be specific) and filtered our data to end up with a dataset free from other language applications.

```
['Ampere Battery Info',
 'Vibook',
 'Smart City Trichy Public Service Vehicles 17UCS548',
 'GROW.me',
 'IMOCCI',
 'The Everyday Calendar',
 'WhatsOpen',
 '桃園機場捷運時刻表 – 捷運轉乘路線快速查詢(支援台北捷運)',
 'Caliway Conductor',
 'Readymade Grocery App',
 'OTENTIK Discovery FR',
 'All in one shopping app',
 'REDMOND  Robot',
 'Contemporary Love Wallpaper HD',
 'Nepali Congress',
 'Block Fill: Puzzle Game',
 'Coloring Book Barbaie',
```

The above snippet highlights the existence of non-English data which was present before cleaning the data. Below is a snippet of the Gib-Score model that we used to eliminate such data records.

```
[ ]  def word_to_char_ratio(text):
          chars = len(text)
          words = len([x for x in re.split(r"[\W_]",text) if x.strip() != ""])
          return words / chars * 100
```

```
     def deviation_score(percentage,lower_bound,upper_bound):
         if percentage < lower_bound:
             return math.log(lower_bound - percentage,lower_bound) * 100
         elif percentage > upper_bound:
             return math.log(percentage - upper_bound, 100 - upper_bound) * 100
         else:
             return 0
```

```
[ ]  def Gib_score(text):
         if text is None or len(text) == 0:
             return 0,0
         cp = chunk_per(text , 35)
         vp = vowels_per(text)
         wtcr = word_to_char_ratio(text)

         cp_dev = max(deviation_score(cp, 45,50),1)
         vp_dev = max(deviation_score(vp, 45,50),1)
         wtcr_dev = max(deviation_score(wtcr,15,20),1)


         return max(math.log10(cp_dev) + math.log10(vp_dev+
                    math.log10(wtcr_dev)) / 6 * 100, 1)
```

- **Re-Grouping For Better Filtering**

We believe that user readability is an important factor that needs to be considered while creating a dashboard. A lot of filters could make it look too cluttered and difficult to understand and compare trends. Therefore, we performed re-grouping in a couple of columns so that we can filter out on this limited small set of groups.

Initially in the “Content Rating” column we had 6 different age groups, in which a couple of them were even overlapping. We restricted them to just 3 categories of “Everyone”, “Teen” and “Adults”. Below is the code snippet of the re-grouping process.

```
[14] df['Content Rating']

0            Everyone
1            Everyone
2            Everyone
3            Everyone
4                Teen
             ...
1761344      Everyone
1761345      Everyone
1761346    Mature 17+
1761347          Teen
1761348      Everyone
Name: Content Rating, Length: 1761098, dtype: object
```

```
[19] df['Content Rating'].unique()

array(['Everyone', 'Teen', 'Mature 17+', 'Everyone 10+',
       'Adults only 18+', 'Unrated'], dtype=object)
```

```
[20] df["Content Rating"]=df["Content Rating"].replace("Unrated","Everyone")
     df["Content Rating"]=df["Content Rating"].replace("Everyone 10+","Teen")
     df["Content Rating"]=df["Content Rating"].replace("Mature 17+","Adults")
     df["Content Rating"]=df["Content Rating"].replace("Adults only 18+","Adults")
```

```
[21] df['Content Rating'].unique()

array(['Everyone', 'Teen', 'Adults'], dtype=object)
```

- **Column Addition**

To understand the trend of rating, we wanted to plot a couple of charts based on the rating vs other columns. But the rating column has too many unique values since they are decimals. So, we added a new column and categorized the ratings into 5 bins.

```
[ ] df['Rating Count'].unique()

array([6.4000e+01, 0.0000e+00, 5.0000e+00, ..., 3.1788e+04, 1.6042e+04,
       3.7104e+04])

[ ] df['Rating Type'] = 'NoRatingProvided'
    df.loc[(df['Rating Count'] > 0) & (df['Rating Count'] <= 10000.0),'Rating Type'] = 'Less than 10K'
    df.loc[(df['Rating Count'] > 10000) & (df['Rating Count'] <= 500000.0),'Rating Type'] = 'Between 10K and 500K'
    df.loc[(df['Rating Count'] > 500000) & (df['Rating Count'] <= 138557570.0),'Rating Type'] = 'More than 500K'
    df['Rating Type'].value_counts()

Less than 10K           875370
NoRatingProvided        865206
Between 10K and 500K     19606
More than 500K             916
Name: Rating Type, dtype: int64

[ ] df['Rating Type'].unique()

array(['Less than 10K', 'NoRatingProvided', 'Between 10K and 500K',
       'More than 500K'], dtype=object)
```

We later uploaded this cleaned dataset to Tableau and created multiple visualizations and below are the insights from our dashboard.

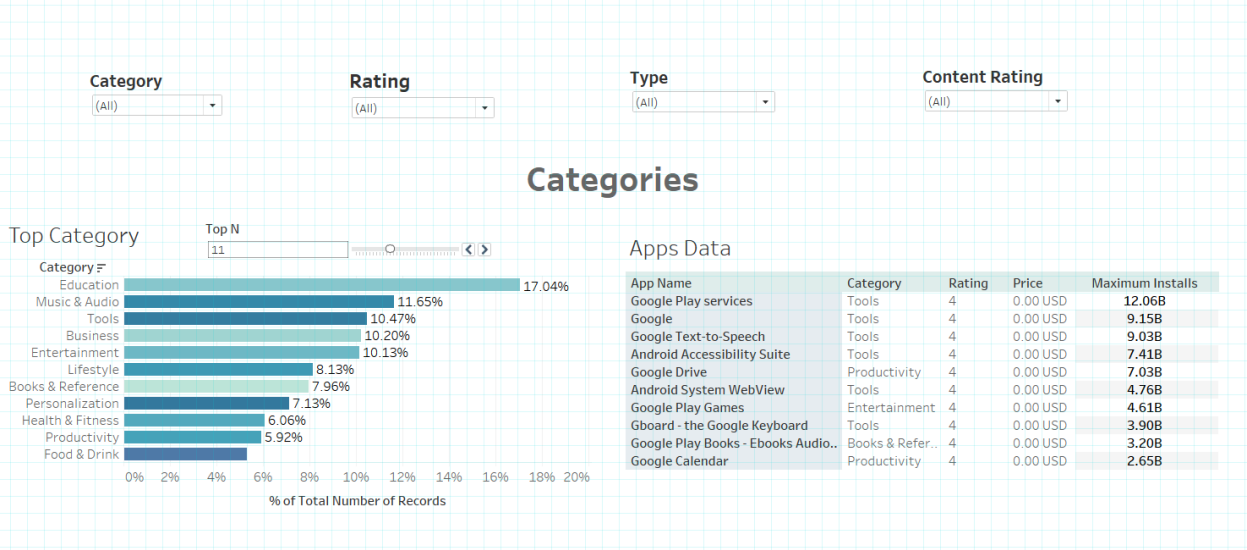
App Category (Top N Apps):

We created visualizations to find the Top categories with most number of installs. Although this is a straightforward visualization, which shows percentage of apps being downloaded from each category, this information is something that an App investor or Developer would be really interested in. If I am a developer trying to make an app then I would definitely want to understand the market size of the category to understand the number of users the application will be dealing with for multiple reasons, sometimes the backend code depends in the load of users too.

User can customize and select the Top N Apps in the graph. There are further common filtering opinions on Category, Rating, Content Rating and Type

Base Filtering Option:

The Next Visualization that we have provided is a multi-filter table that filters out based on the user's inputs. There are 5 columns with filter options for the user to get a list of a specific set. User can filter out based on App Name, Category, Rating, Price and Maximum Installs. Further filtering can be done using the top common filters of Type and Content Rating.



Rating, Category and Content Rating:

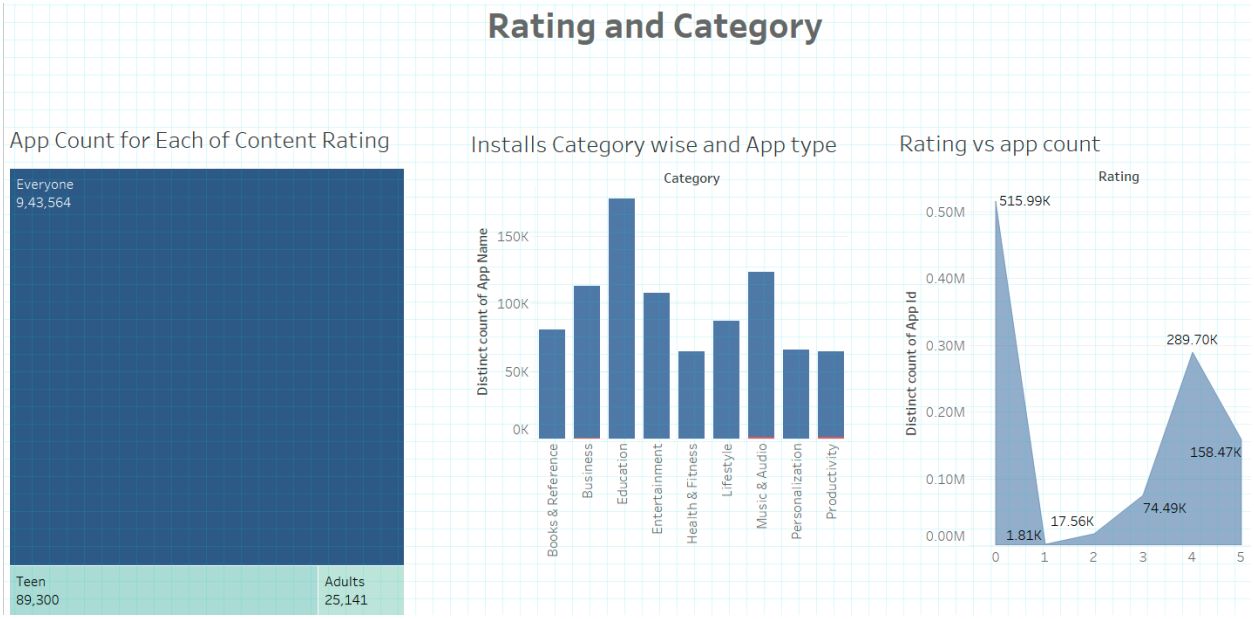
In our data these two factors have been used to create multiple visualizations. Rating is the numerical value assigned to n App, and the Category tells us which industry the app belongs to and Content Rating is the age group that is allowed to use the particular App.

We created visualizations to show how the apps in the market are spread among the three content rating categories.

The spread of Apps being downloaded with respect to Rating has also been plotted. This shows the spread of the Top N categories. The N value is user customizable.

These visualizations can be used to understand the type of Apps that are being rated more and which Category they belong to.

There is also a plot to understand the number of apps vs their rating. It shows how the apps are spread along the rating. With millions of Apps available, it is f no surprise that the unrated i.e. rating ‘0’ has the peak in the graph.



App Size vs Installation:

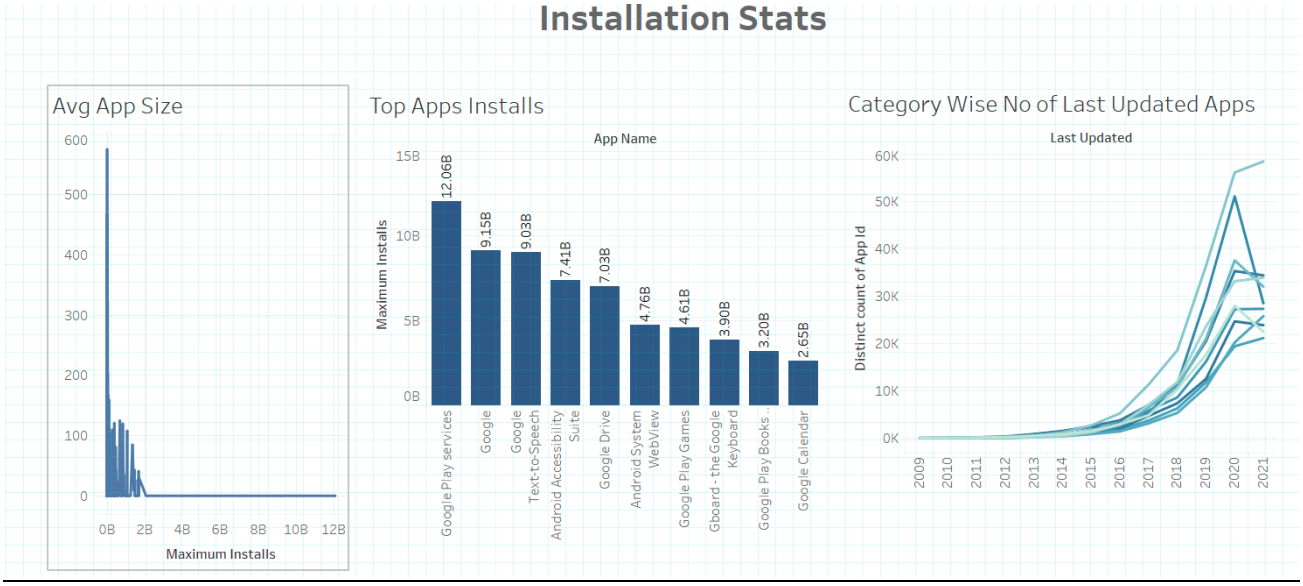
This plot is shows the trend among the App size and maximum installs. It has been observed that the maximum installs keep increasing as the size decreases. Developers must keep in mind that App size is a key factor and that users prefer smaller app size for them to download them.

Top N Maximum Installs vs App Name:

This plot shows the “Maximum Installs” on the y axis and the corresponding ‘App Name’ on the x axis. Keeping in mind that it’s the Android market, it is of no surprise that the Google Apps take majority of the Top 10 maximum installs.

Last Updated:

We plotted the count of Apps downloaded vs the year they were last updated. This shows the trend of category vs how they are being developed through years. One immediate result that be observed is that irrespective of category, all kinds of Apps observed a peak modification in the year 2020. This could be due to the outbreak of COVID as that's the ear when all apps had additional features or a change in the UI etc.

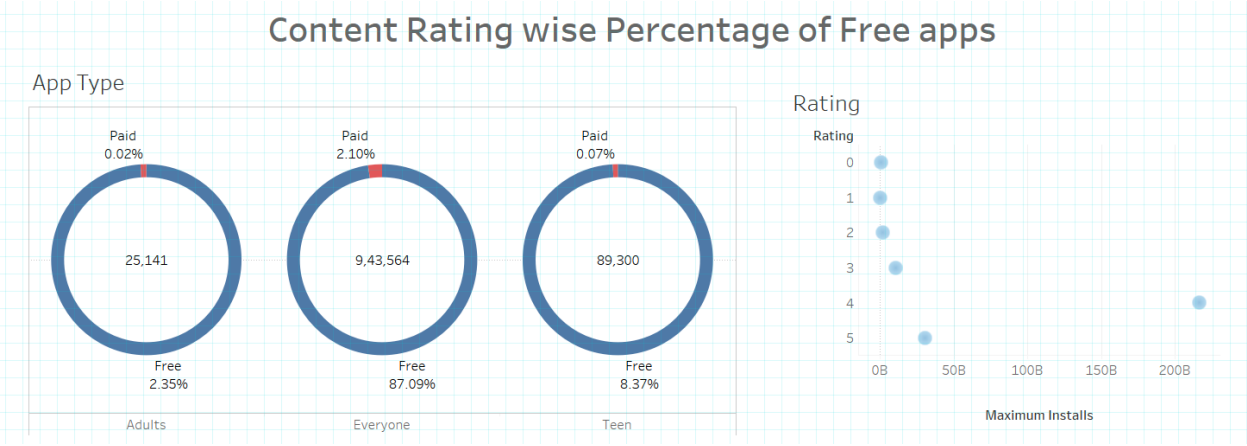


App Type and Content Rating:

This visualization shows the split that is the percentage of Paid apps and Free apps in each of the three content rating categories – Adults, Everyone and Teen. This information is of great importance for developers who are creating apps for a specific age group. If the developers target users are 'Teens' then he/she could understand the trend of just that category. The number of 'Adults' using 'paid' apps is the least.

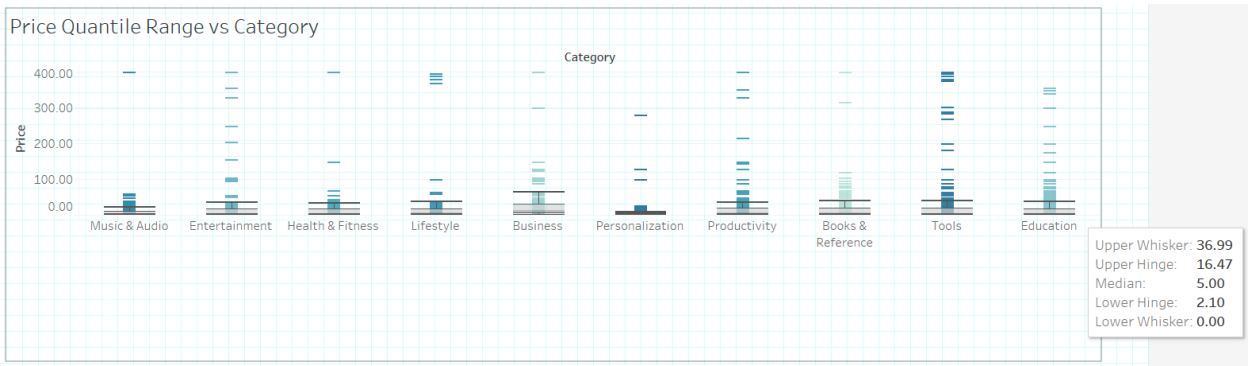
Rating vs Installs:

This visualization shows that Apps with rating ‘4’ have the maximum number of installs. But the first question that arises is, why aren’t the apps with an even higher rating i.e. 5 are downloaded in a significant low amount compared to those with rating ‘4’. One reason that we ended up with on further analysis is that, the number of paid apps number goes up in the category of rating 5 apps and even the size of these apps are somewhere in the middle or more inclined towards the right, i.e. bigger size.



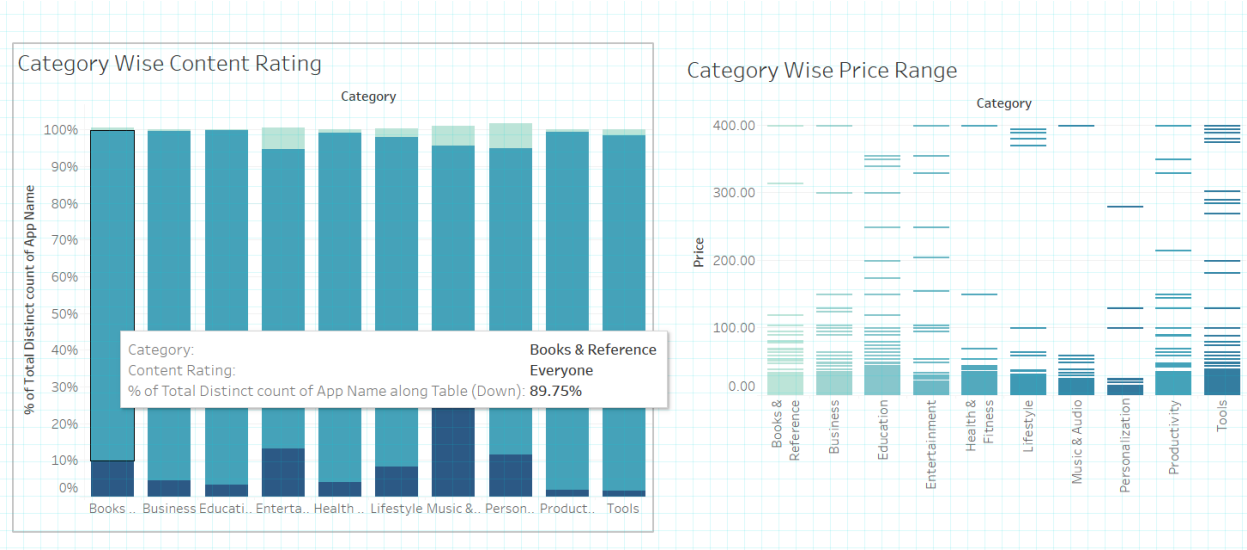
Price Quartile Range vs Category:

The box plots for the Top N categories can be visualized here; again, the N is user customizable. On hovering over each box plot of the price range.. It shows the 5 parameters in detail i.e. The Upper and Lower Whisker, Hinge and the median. This helps in understanding the profit margins and the price ranges that can be set with respect to category.

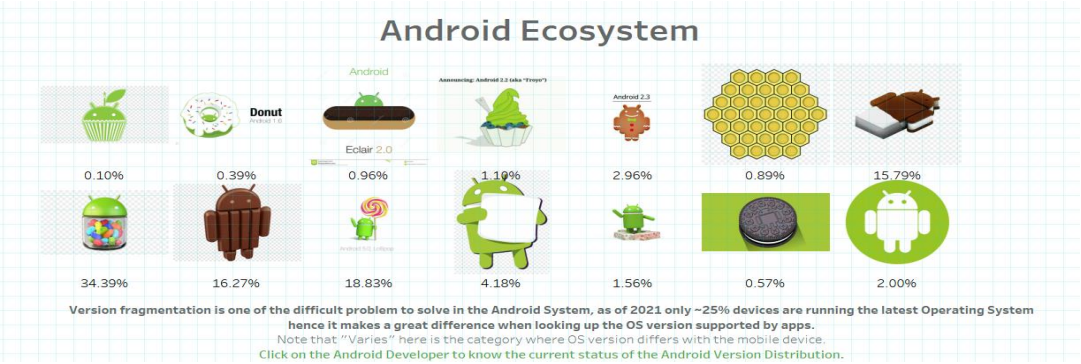


Content Rating percentage among each Category:

This plot shows the percentage of installs by Teens, Adults and Everyone in each Category. On hovering over the bar, the stats are clearly shown. This can be seen in the snippet below i.e. in the category “Books & References” 89.75% are of the Content Rating “Everyone”.

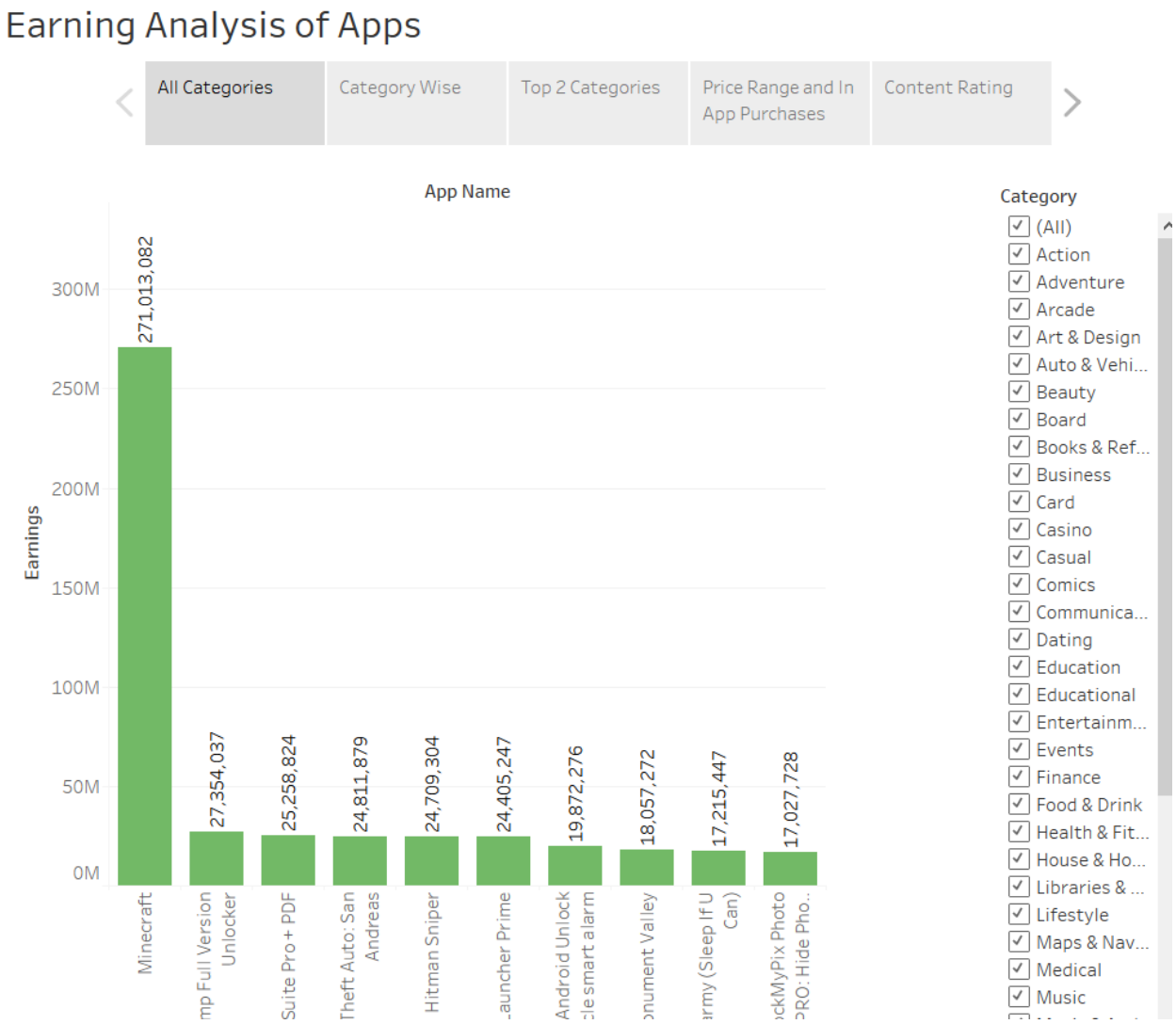


Android Ecosystem:



Earning Analysis of Apps (Story):

We calculated the profit i.e. the earning by using the 'Price' and number of installs column. The top 10 earnings have been plotted. This can be filtered to show the highest profit making Apps with respect to category, i.e. t=if you observe there is a checkbox filter towards the right.



"Minecraft" is the App that is making the most earnings is what we have analyzed.

Category wise bubble plot of earning has been plotted too. This was to understand which industry of apps is making more earnings. "Arcade" is clearly making the most earnings. If a

developer is interested in making high earning and want to understand which type of app they need to develop, then gaming apps are the best option followed by “Action”

Since these two are the categories with the highest earnings. We then performed an earning analysis within these two categories. Minecraft is the App with highest earning in the Arcade category and overall among all categories too. In Action, “Grand Theft Auto” is the app with highest earning.

PART 5 – LIMITATIONS AND FUTURE WORK

In Future, we will merge more datasets together and get a column for “Region”. This would be an important factor when it comes to performing country level analysis. A couple of apps are available only in select countries.

Future scope is by obtaining a time series dataset, i.e. with an added Date column. With this information we can provide analysis on the shift in the usage of Apps over time.
This kind of regional and time based analysis can give deeper insights which would be extremely useful for App developers and investors. Time series data will let us forecast future trends too which is of utmost importance to investors.

Coming to limitations, in our current dashboard we eliminated records that were not in english. This could impact the analysis. Example: Say we have Teens downloading Educational Apps the most now as the result. But since we dropped records which were in other languages, assuming all of them were of the category “Dating”, the actual result should have been Teens download Dating Apps the most.

Also the Android version fragmentation is an issue that still persists. As of 2021 only 25%of the users use the latest Operating system. And few Apps are supportedonly on the latest versions or on the old versions. So due to this inconsistency even though the Apps exist, a certain users cannot use them due to the version of their OS. This could lead to inconsistency in the results.

PART 6 – IMPLEMENTATION

1. The original unclean dataset is available at the following link. (https://www.kaggle.com/code/sampathkumarlam/google-play-store-analysis/data).

2. We used this dataset and performed phase 1 of cleaning. Below is the link to the .ipynb file with a step by step code of the cleaning process.

3. This cleaned dataset has been downloaded and used in the gib_score workbook to clear out the non-English language data.

4. Then we used the final dataset from this to generate visualizations in Tableau. Below is the link to the final cleaned dataset.

   (https://drive.google.com/file/d/195EQ4RPdjJ9i4YKywzbgI9KF9R1gFJXg/view?amp;usp=embed_facebook)

5. There are two Tableau versions that we uploaded. One has the Part 1 Analysis as a dashboard. The second one has the Earning Analysis Story.

   Dashboard Link –

   https://public.tableau.com/app/profile/mohit8363/viz/Playstoreappsanalysis-updated/FinalDashboard

   Story Link –

   https://public.tableau.com/app/profile/mohit8363/viz/FinalStory_16710594413660/FinalStory?publish=yes