Scientific Programming In Python
App Store Data Analysis

Table Of Contents

Datasets Details

We delve into two comprehensive datasets that provide data on Google and Apple app stores. These datasets offer information about smartphone applications, encompassing essential metrics such as download counts, user ratings, categories, and more.

1. Google app store: "**Google-Playstore.csv**"
   Features:

	Name	**Type**	**Description**
1	App Name	String	The app name
2	App Id	String	The app ID
3	Category	String	The app category
4	Rating	Float	The rating of the app
5	Rating Count	Float	How many rated the app
6	Installs	String + number	Amount of installs
7	Minimum Installs	Int	App's min installs
8	Maximum Installs	Int	App's max installs
9	Free	Bool	Is the app free
10	Price	Float	The price of the app
11	Currency	String	App currency
12	Size	String + float	The size of the app (in M / G / K)
13	Minimum Android	String + float	Minimum version of the Android operating system required to run the app
14	Developer Id	String	The developer ID
15	Developer Website	String	The developer website

16	Developer Email	String	The developer email
17	Released	Date	App release date
18	Last Updated	Date	App update date
19	Content Rating	String	Specific audiences
20	Privacy Policy	String	Link to privacy policy
21	Ad Supported	Bool	Does the app support ad
22	In App Purchases	Bool	If there is purchases in the app
23	Editors Choice	Bool	Does the editors choose this app
24	Scraped Time	Date + Time	Data timestamp

2. Apple app store: "**appleAppData.csv**"
   Features:

	Name	**Type**	**Description**
1	App_Id	String	The app ID
2	App_Name	String	The app name
3	AppStore_Url	String	The app URL
4	Primary_Genre	String	The primary genre of the app
5	Content_Rating	String+number	App content rating
6	Size_Bytes	Int	The size of the app in bytes
7	Required_IOS_Version	Float	Required IOS version for the app
8	Released	Date	App release date
9	Updated	Date	App update date
10	Version	Float	The app version
11	Price	Float	The price of the app

12	Currency	String	App currency
13	Free	Bool	Is the app free or not
14	DeveloperId	Float	The ID of the app developer
15	Developer	String	The name of the app developer
16	Developer_Url	String	Apple URL for App Store Preview
17	Developer_Website	String	The developer website
18	Average_User_Rating	Float	The average rating of the app
19	Reviews	Float	Amount of reviews
20	Current_Version_Score	Float	App’s current version score
21	Current_Version_Reviews	Float	App’s current version reviews

Preprocessing

We made a series of data preparation steps in order to ultimately unite the two data sets into one data set on which we will perform the data analysis.

We made preprocessing for each dataset separately before merging, and then another preprocessing of the compressed data set after merging.

The steps that we made:

1. **Standardizing Column Names**

   To differentiate between features from the Google and Apple datasets after merging, we added a suffix indicating their respective sources: "_google" and "_apple." This will help distinguish between the data after merge.

2. **Data Combination**

   We opted to merge datasets based on the "App_Id" rather than the "App_Name" to ensure reliability and accuracy of our analysis while minimizing the risk of data mismatching.
   For that we had to rename the google ID column to “App_Id” instead of “App Id”.

3. **Handeling null values**

   Handling Null Values with Precision - handling null values by leveraging the "dropna" function.

4. **Reordering Columns**

   Rearranged the columns across both Google and Apple app store datasets. This strategic reordering ensures that columns with identical names, existing in both datasets, are positioned adjacent to each other.

5. **Remove Unnecessary Features**

   Streamlined the dataset by removing unnecessary attributes.

   The attributes we choose to remove **before** the merge:

   'Last_Updated_google', 'Updated_apple', 'Currency_google', 'Currency_apple', 'Installs_google', 'Developer Website_google', 'Developer Email_google', 'Privacy Policy_google','Scraped Time_google', 'AppStore_Url_apple', 'Required_IOS_Version_apple', 'Version_apple','Developer_apple', 'Developer_Url_apple', 'Developer_Website_apple', 'Current_Version_Score_apple', 'Current_Version_Reviews_apple'

   The attributes we choose to remove **after** the merge:

   'Size_google', 'Rating Count_google', 'Content_Rating_apple', 'Content Rating_google','Reviews_apple','Minimum Android_google'

6. **Converte Data Types**

   6.1. Convert 'Rating_google' and 'Average_User_Rating_apple' to numeric.

   6.2. Convert ‘Price_google' and 'Price_apple' to numeric.

   6.3. Convert 'Size_google' to numeric **after processing** size values and **removing** units ('G', 'M', 'k') using custom conversion functions (convert_to_gigabyte, convert_to_megabyte, convert_to_kilobyte) and **replacing** 'Varies with device' with '0'.

   6.4. Converts the 'Total_rating' to numeric.

   6.5. Converted to numeric : 'Size_Bytes_apple', 'Rating_google', 'Rating Count_google', 'Minimum Installs_google', 'Maximum Installs_google', 'Average_User_Rating_apple', 'Reviews_apple'.

6.6. 'Date_google_agg' and 'Date_apple_agg' converted to quarter-year format ('Q{quarter}{year}') using custom function quarter_of_year.

6.7. 'Released_google' converted to datetime objects and assigned to the **new column** 'Date_google_agg'. 'Released_apple': Converted to datetime objects and assigned to the **new column** 'Date_apple_agg'.

7. Adding Features

7.1. 'Total_rating' that contains the average rating.

7.2. ‘Date_google_agg’ and ‘Date_apple_agg’ that contains the released date of the app in google and apple.

8. Handling Missing Values

Using “dropna”.

9. Removing Duplicates

Usign “drop_duplicates”.

10. Handling Outliers

Calculat the interquartile range (IQR) for the 'Rating_google' column.

11. Scaling Numerical Features

Using ‘MinMaxScaler’.

At the end, the new data set saved as "**ready_dataset.csv"**

Describe of the dataset and features after processing it:

	Name	**Type**	**Description**
1	App_Id	String	The app ID
2	App_Name_google	String	The app name in google
3	App_Name_Apple	String	The app name in apple
4	Free_google	Bool	Is the app free on google
5	Free_apple	Bool	Is the app free on apple
6	Price_google	Float	The price of the app in google
7	Price_apple	Float	The price of the app in apple
8	Category_google	String	The category of the app in google
9	Primary_Genre_apple	String	The category of the app in apple
10	Released_google	Date	App release date in google
11	Released_apple	Date	App release date in apple
12	Size_Bytes_apple	Float	App size in apple (in bytes)
13	Rating_google	Float	App rate in google
14	Developer Id_google	String	The developer ID in google
15	DeveloperId_apple	Int	The developer ID in apple
16	Minimum Installs_google	Float	The min installs in google
17	Average_User_Rating_apple	Float	App rate in apple
18	Ad Supported_google	Bool	Does the app support ads(google)
19	In App Purchases_google	Bool	Can in-app purchases be made(google)
20	Editors Choice_google	Bool	Does the editors choose the app

			(google)
21	Maximum Installs_google	Int	The max installs in google
22	Total_rating	Float	The sum of app rate
23	Date_google_agg	Date	Release date in quarter format (google)
24	Date_apple_agg	Date	Release date in quarter format (apple)
25	Size_google_scaled	Float	App size in apple (in bytes)

Features Distribution

Describe the distribution of interesting features and what can be learned about them:

1. **Price distribution**

   Price Range:
   Highest price: 249.99
   Lowest non-zero price: 0.99
   Lowest price: 0.0

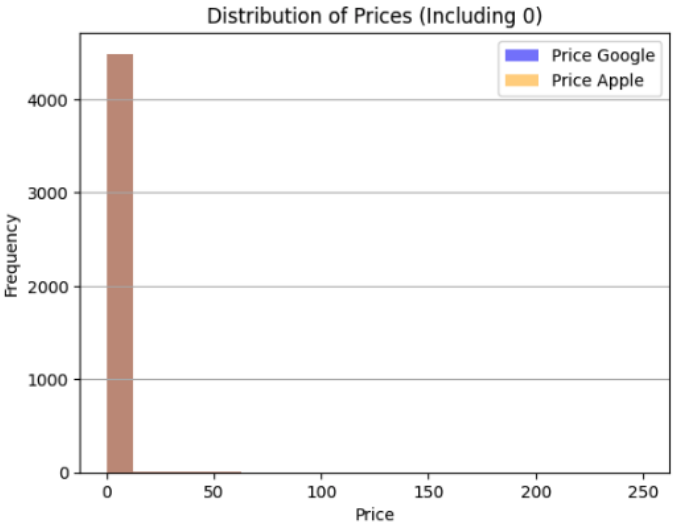
   Total Prices On Google Play Store:
   Number of zero-priced apps: 80548
   Number of non-zero-priced apps: 1527

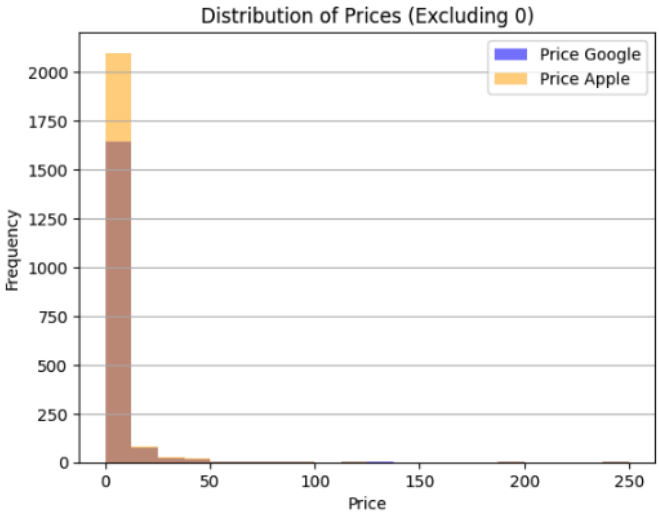
   Total Prices On Apple App Store:
   Number of zero-priced apps: 80163
   Number of non-zero-priced apps: 1912

   Total:
   Number of zero-priced apps: 160711
   Number of non-zero-priced apps: 3439

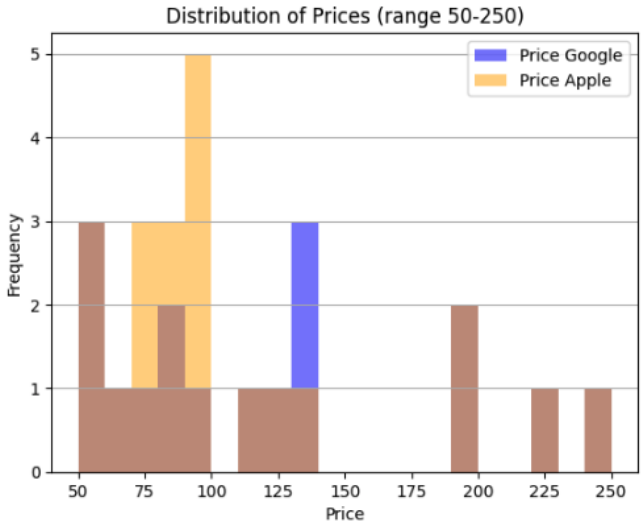
Graph:

Range 0 – 250 :





Range 50-250:



Google Play has more free apps compared to Apple's App Store.
Explore further details in the 'Distribution of Free vs. Paid Apps' section below.

Consistency between "Price" and "Free":

We wanted to assess the compatibility between the 'price' and 'free' features to ensure their coherence and the reliability of the data.

There are no inconsistencies between 'Free' and 'Price' columns. This indicates the reliability of the data, providing confidence to proceed with further data analysis.

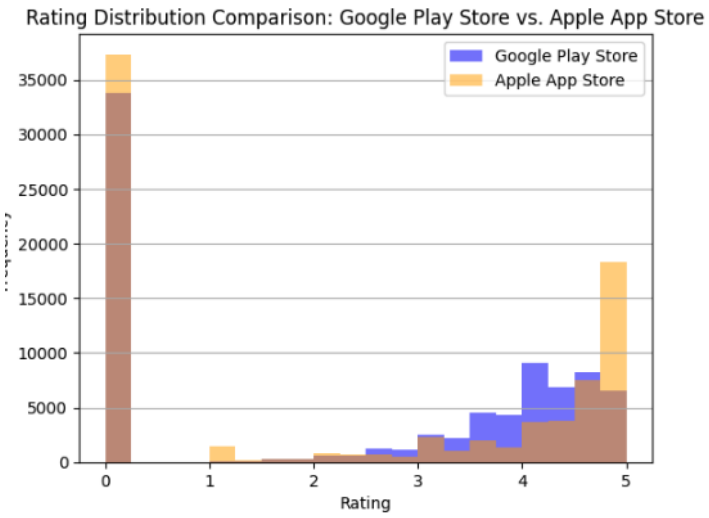
Market Share Analysis

Market Share of Free Apps on Google Play: 98.14%

Market Share of Free Apps on Apple App Store: 97.67%

2. Rating distribution

The range is 0-5:



App with the highest rating on Google Play Store: **Buzzer Beater**
App with the highest rating on Apple App Store: **RePlayer**
Both have **5.0** rating

The most rated categorie on google store – **Business**.

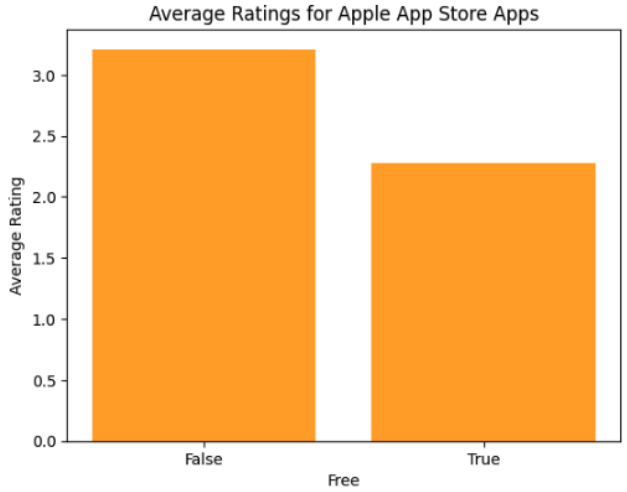
The most rated categorie on apple store – **Games**.

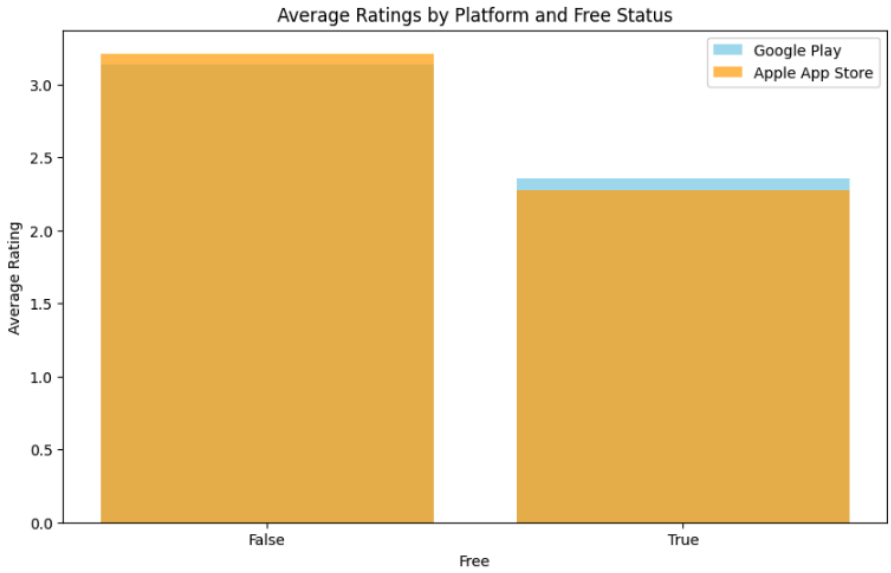
3. Average Ratings Comparison

Average Ratings for Google Play Apps:
Paid apps: 3.1067749160134377
Free apps: 2.356199880499387

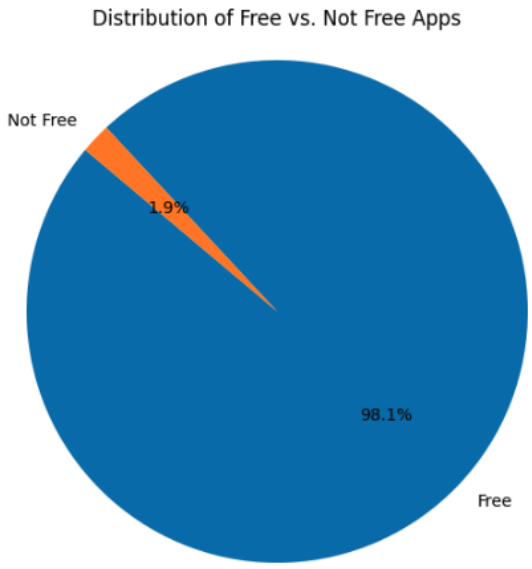
Average Ratings for Apple App Store Apps:
Paid apps: 3.1665661130742047
Free apps: 2.2726355969826906

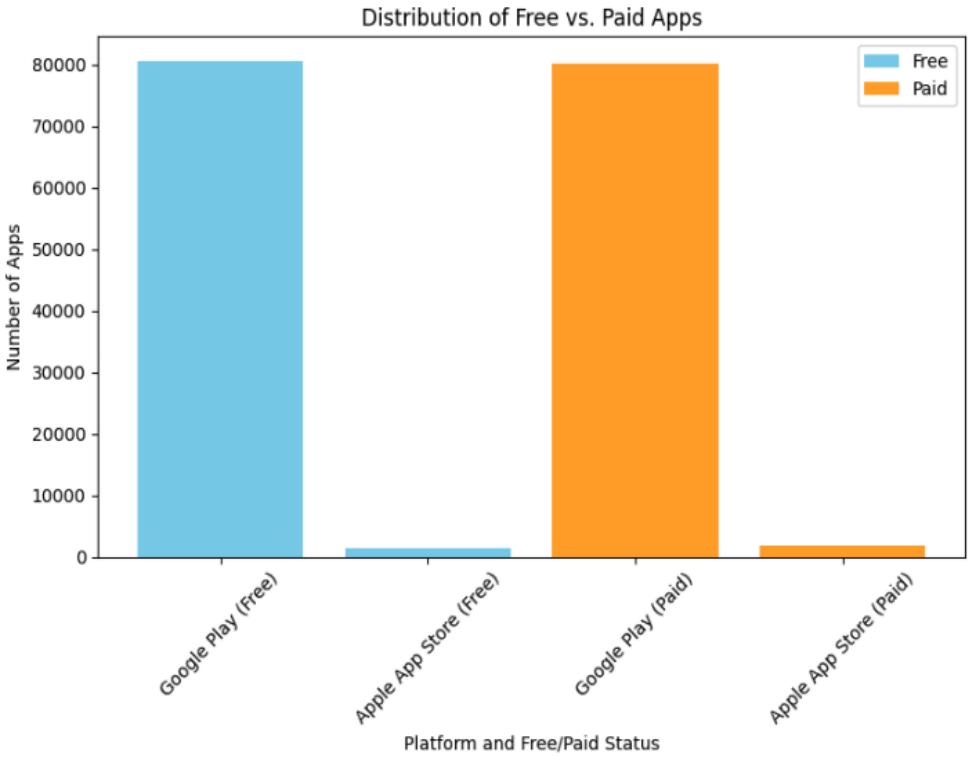
Overall Average Rating for Google Play and Apple App Store Apps Together:
2.335405778617118





4. Distribution of Free vs. Paid Apps

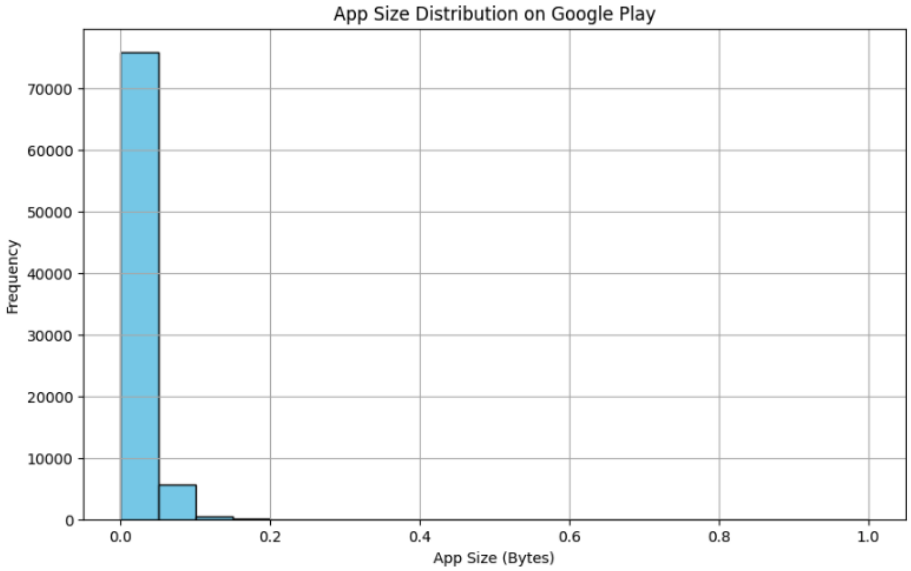




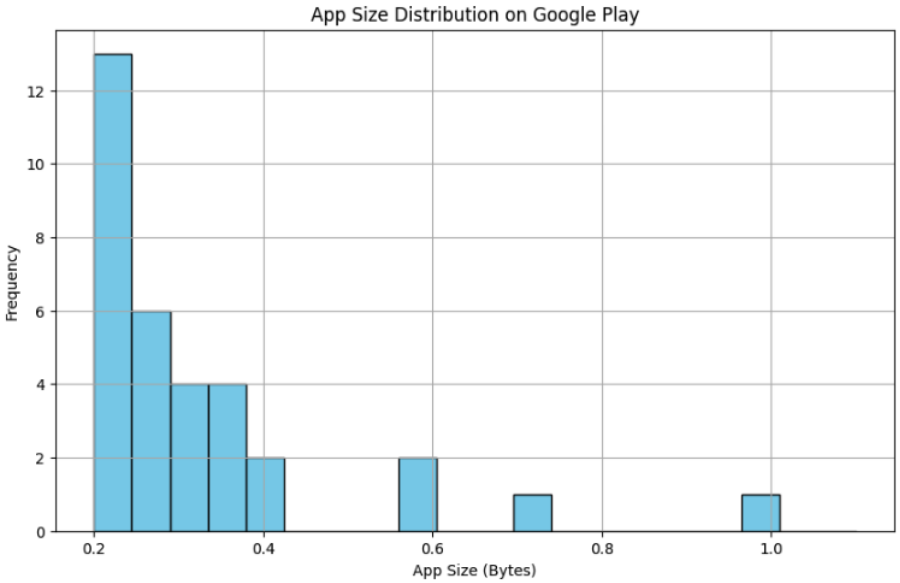
The bar plot shows that there are more free apps on both Google Play and the Apple App Store compared to paid apps. This indicates that the majority of apps on both platforms are available for free.
Google Play has a higher proportion of free apps compared to the Apple App Store.

5. **App Size Distribution**

On Google Store:
Full Range:

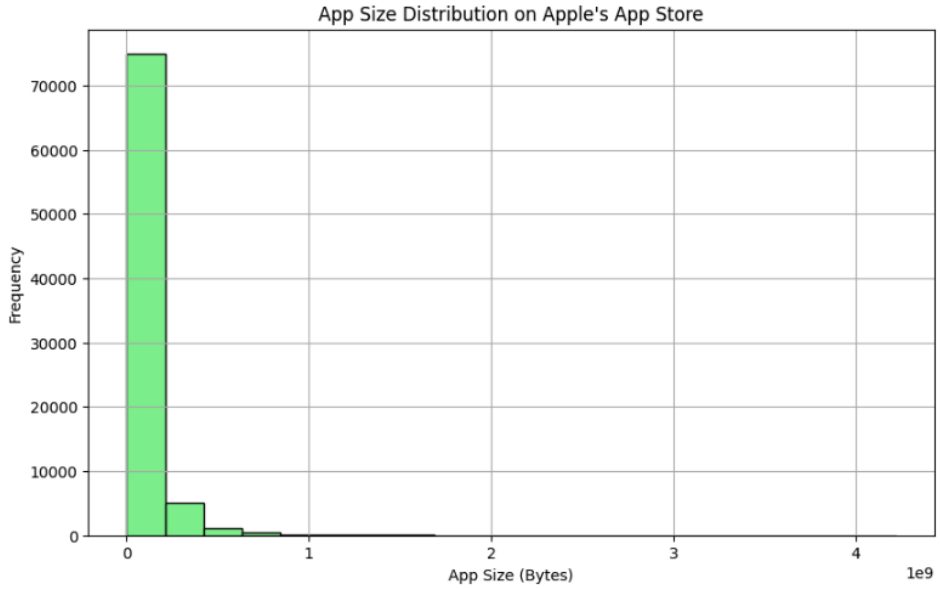


Range 0.2-1.1:



On Apple Store:

Full Range:



Average app size on Google Play: 0.020186648574084646
Average app size on Apple's App Store: 96703908.63714895

App with Maximum Size in Apple Dataset: DanMachi - MEMORIA FREESE
Size: 4221705216.0

App with Minimum Size in Apple Dataset: Smart Guard Control
Size: 260096.0

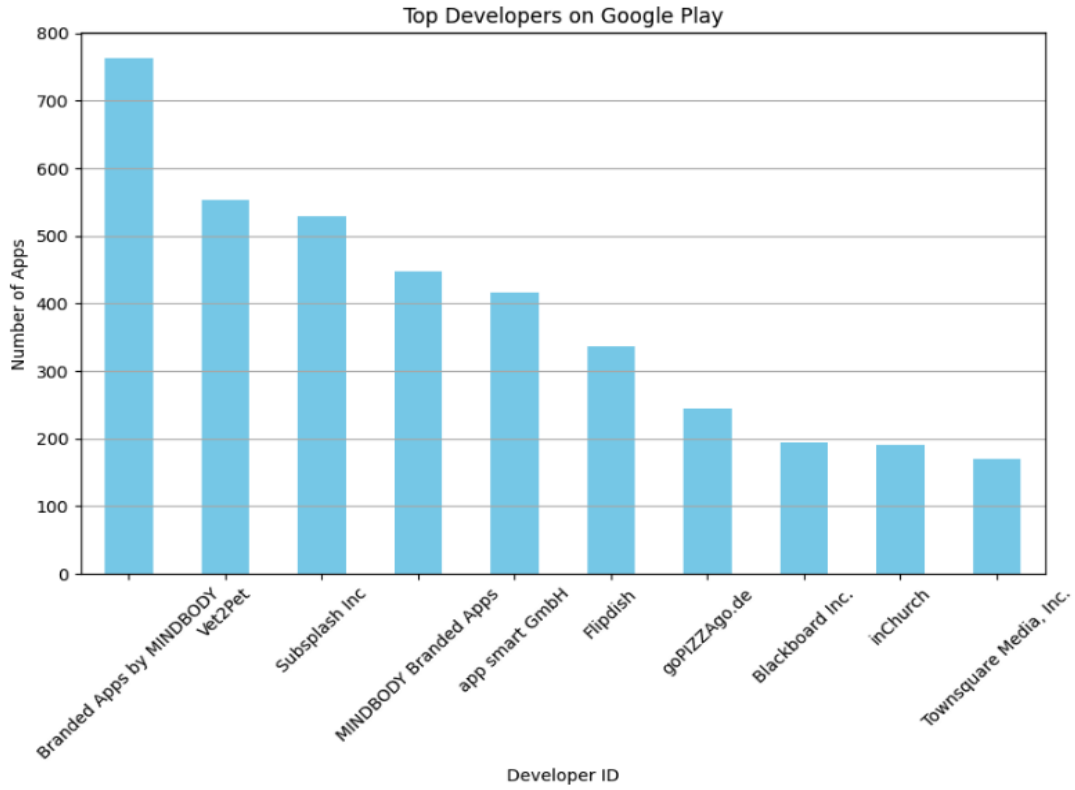
App with Maximum Size in Google Dataset: Legendary Edition
Size: 1.0

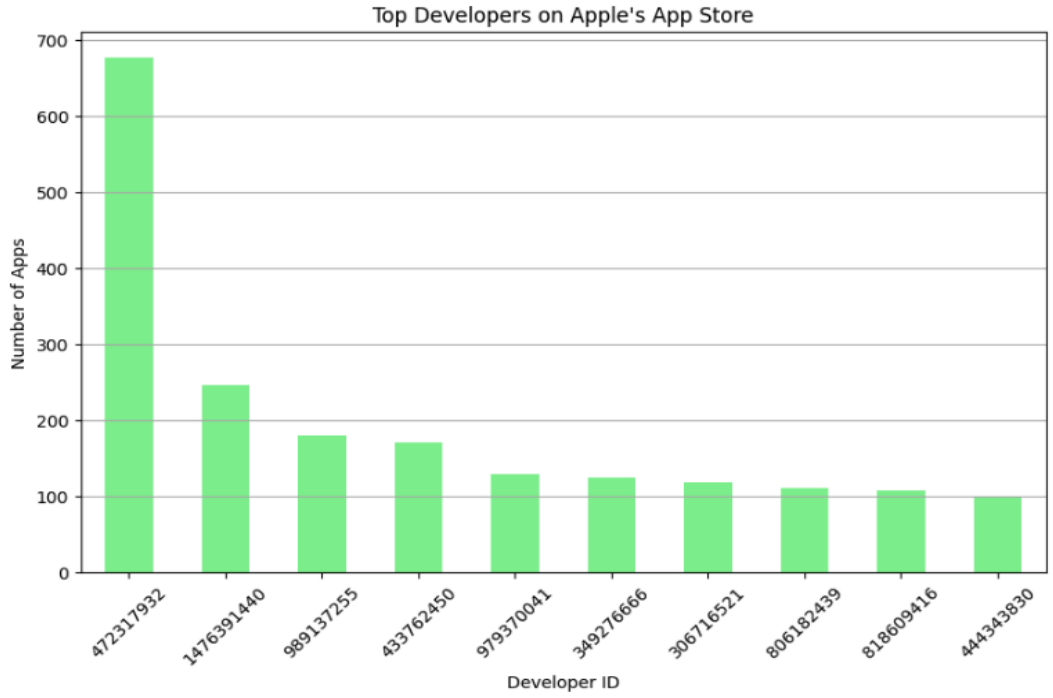
App with Minimum Size in Google Dataset: Delta Ontario
Size: 0.0

App with the Most Minimum Size: Delta Ontario
Size: 0.0
Dataset: Apple

App with the Most Maximum Size: DanMachi - MEMORIA FREESE
Size: 4221705216.0
Dataset: Apple

6. 10 Top developers





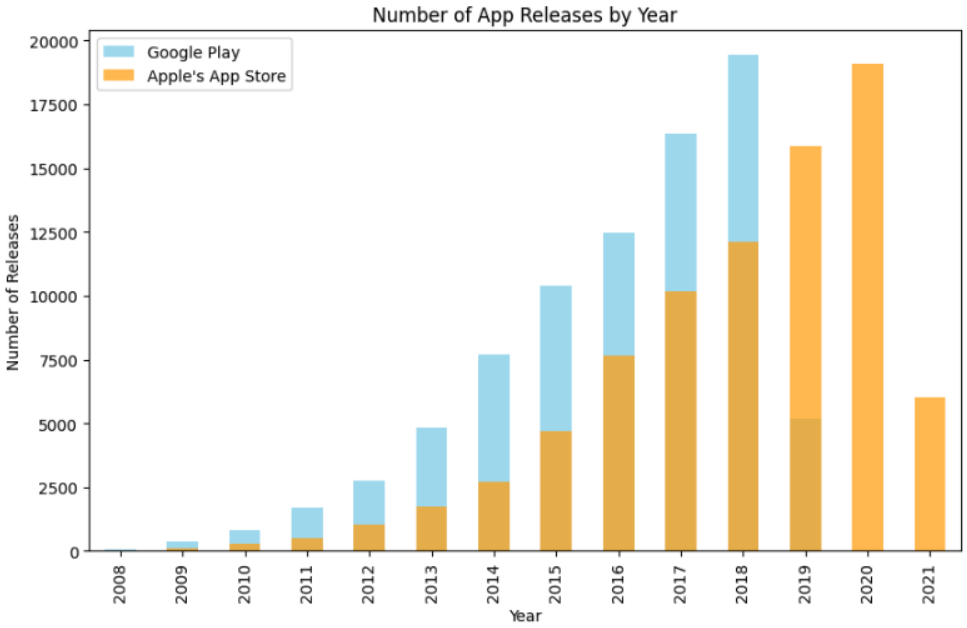
Top developers on Google Play:

Branded Apps by MINDBODY	**763**
Vet2Pet	553
Subsplash Inc	529
MINDBODY Branded Apps	447
app smart GmbH	416
Flipdish	337
goPIZZAgo.de	245
Blackboard Inc.	194
inChurch	190
Townsquare Media, Inc.	170

Top developers on Apple's App Store:

472317932	**677**
1476391440	245
989137255	180
433762450	170
979370041	128
349276666	124
306716521	118
806182439	110
818609416	107
444343830	100

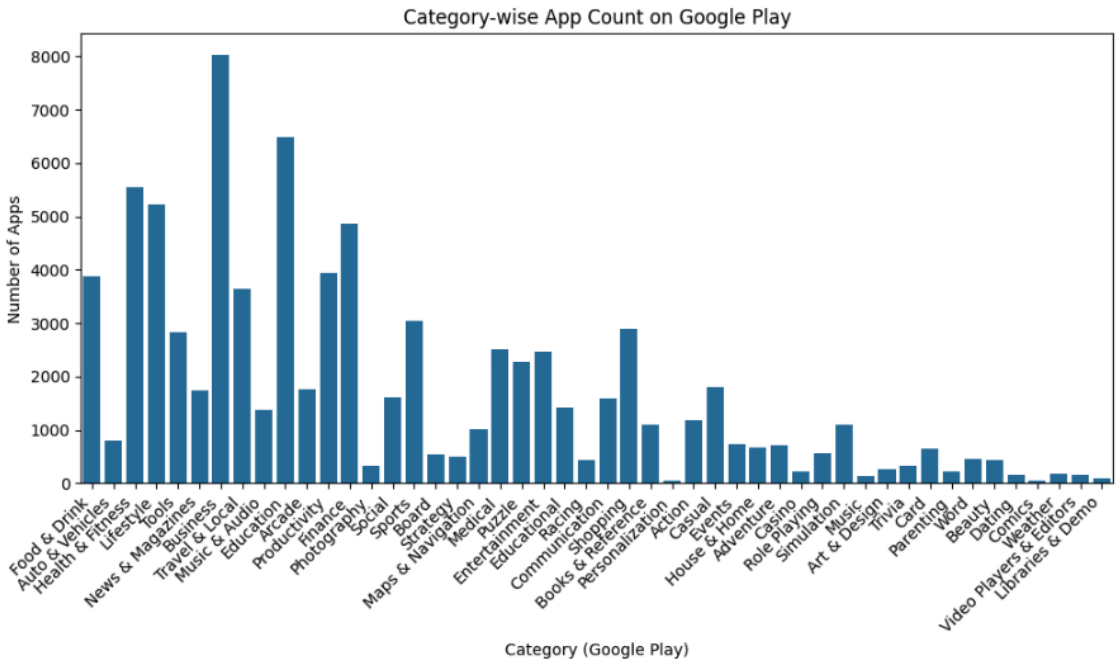
7. Number of App Releases by Year

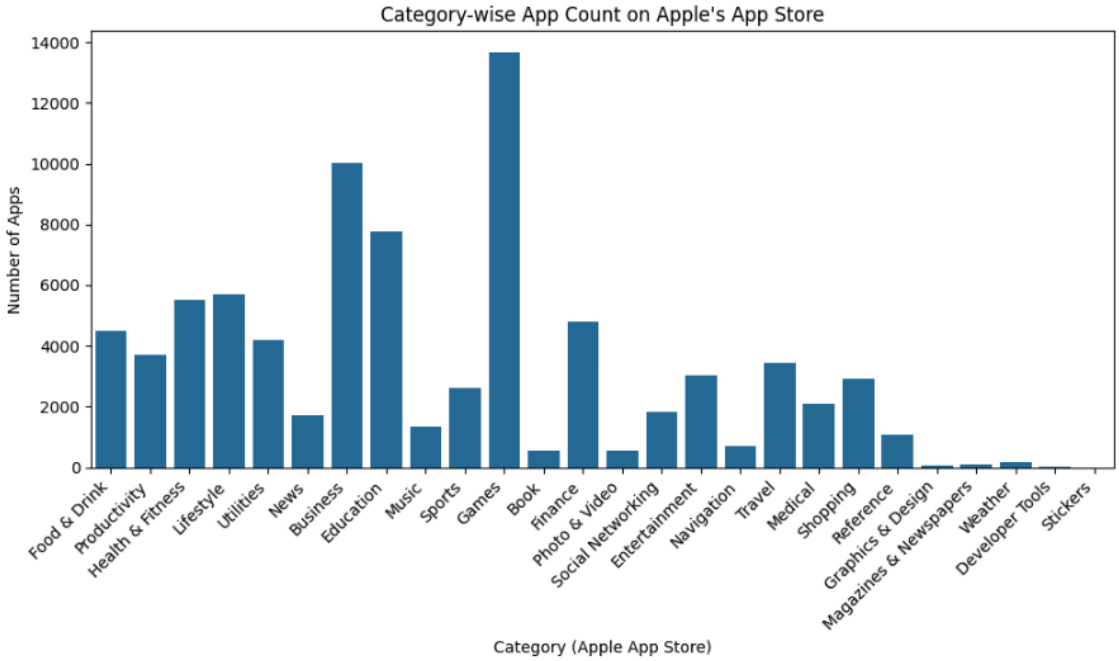


* The information of Google Play is up to date until 2020

The bar chart above illustrates the number of app releases by year on Google Play and Apple's App Store. Both platforms experienced steady growth in app releases over the years, with **Google Play showing a slightly higher number** of releases in recent years compared to Apple's App Store.

8. Category-wise App Count





The count plots above display the distribution of apps across different categories on Google Play and Apple's App Store.

On Google Play, the '**Business'** category appears to have the highest number of apps, followed by **'Education'** and **'Health & Fitness'**.

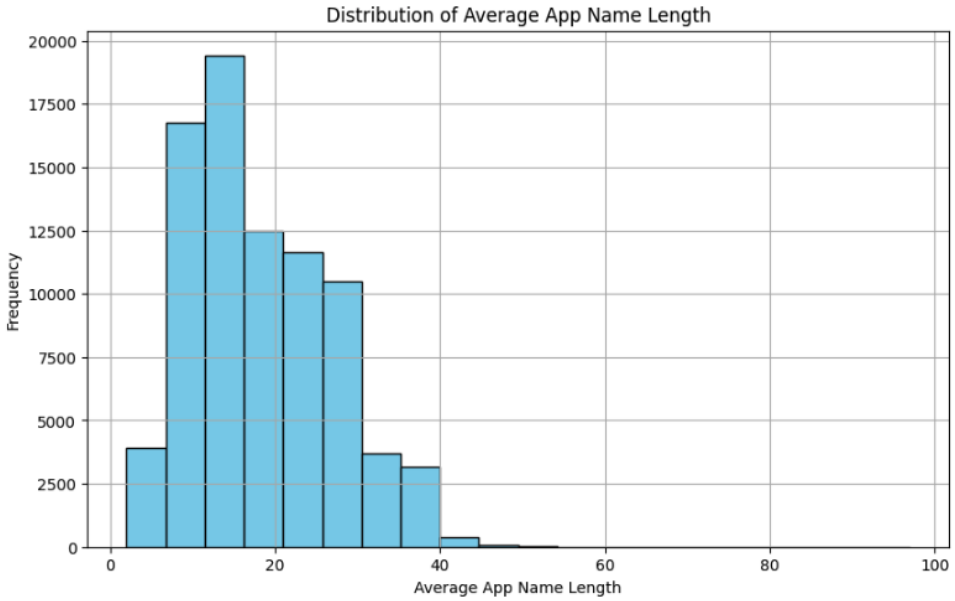
On Apple's App Store, the **'Games'** category dominates with the highest number of apps, followed by '**Business'** and **'Education'**.

9. **App Names**

Number of names not the same in both columns: 25980

(That's why we chose to merge by unique identifier)

Distribution of average app name length:



Both in the Google store and in the Apple store the longest App Name is
Best Credit Card Reader & Swiper App - Process Credit Cards Fast on Your Mobile Phone with this Point of Sale (POS) System - Download Now for Free
With length of 146.

Correlations Between Features

Contrary to our expectations , we haven't identified a perfect positive or negative linear relationship (1 or -1).

The relationships we found are:

Positive Linear Relationship:

1. **Maximum installs google** & **Minimum installs google** – It indicates that apps with higher maximum installs tend to also have higher minimum installs, suggesting consistent popularity across different user segments.

2. **Total Rating** & **Rating google** – As could be expected, apps with higher overall ratings tend to also have higher individual ratings on Google Play.

3. **Total Rating** & **Average User Rating apple** – Similarly , apps with higher overall ratings across both platforms also tend to have higher individual ratings on the Apple App Store.

4. **Year google** & **Year apple** – Suggests that the release years of apps on both platforms tend to follow similar trends over time.

5. **Price apple** & **Price google** - It appears that as the price rises in one company, it also increases in the other.

6. **Released google year** & **year apple** - There is a strong linear relationship between the year a particular app was released on Google Play and the year it was released on the Apple App Store. It means that if an app was released more recently on Google Play, it is highly likely to have been released more recently on the Apple App Store as well.

7. **Released apple year** & **year google** - Similarly, the correlation indicates a strong linear relationship which implies that if an app was released more recently on the Apple App Store, it is highly likely to have been released more recently on Google Play.

8. **Released apple year** & **Released google year** - This suggests that the release dates of apps tend to be consistent across both platforms.

Negative Linear Relationship:

1. **Year apple** & **rating google** - There is a weak negative correlation which implies that apps released more recently on the Apple App Store tend to have slightly lower ratings on Google Play.

2. **Year google** & **rating google** - Similarly, there is a weak negative correlation which suggests that newer apps on Google Play may receive slightly lower ratings.

3. **Released google year** & **rating google** - indicates that apps released more recently on Google Play may tend to have slightly lower ratings.

4. **Released apple year** & **rating google** - Apps released more recently on the Apple App Store may have slightly lower ratings on Google Play.

5. **Year apple** & **total rating** - that implies that newer apps on the Apple App Store may have slightly lower total ratings.

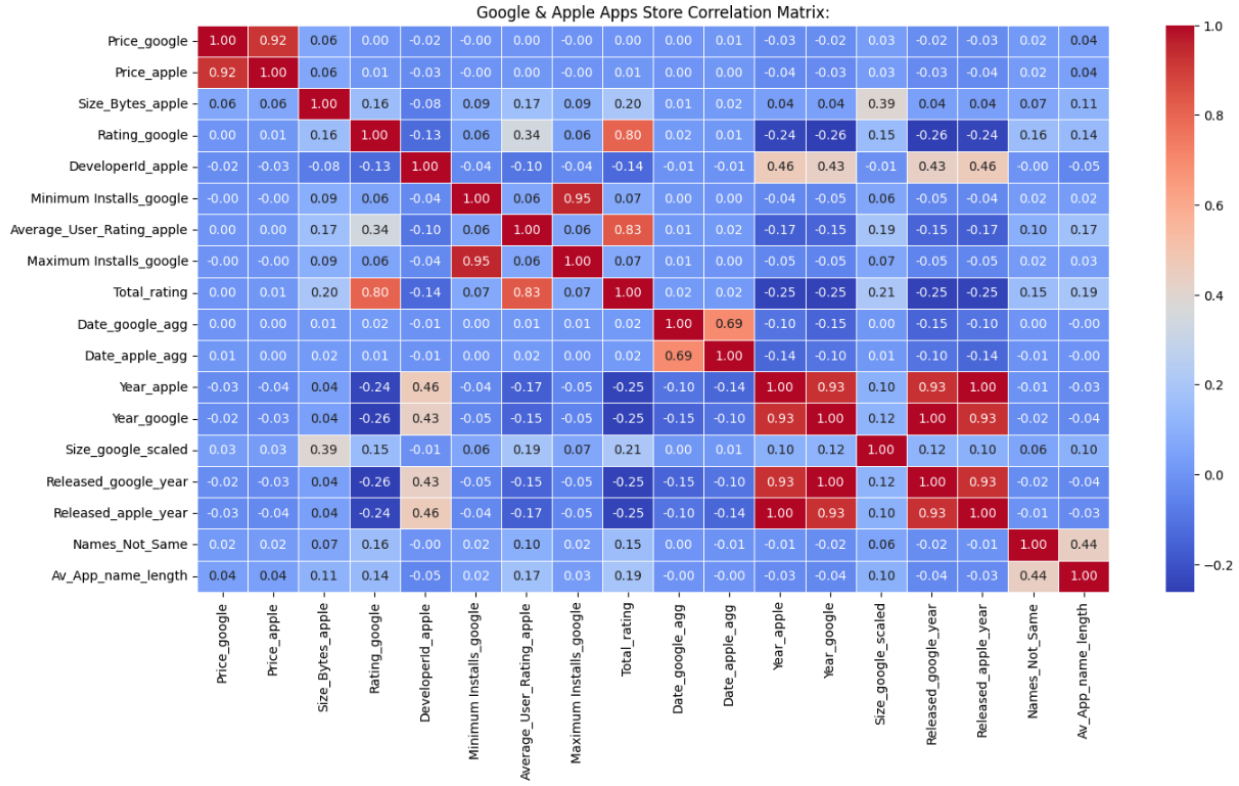
6. **Year google** & **total rating** - Similarly, the newer apps on Google Play may receive slightly lower total ratings.

7. **Released apple year** & **total rating** - Apps released more recently on the Apple App Store may have slightly lower total ratings.

8. **Released google year** & **total rating** - Suggests that newer apps on Google Play may tend to have slightly lower total ratings.

These negative linear relationships suggest a weak inverse correlation.

Contrary to our expectations, there appears to be no correlation between price and year.


Google & Apple Apps Store Correlation Matrix:
Price_google 1.00 0.92 0.06 0.00 -0.02 -0.00 0.00 -0.00 0.00 0.00 0.01 -0.03 -0.02 0.03 -0.02 -0.03 0.02 0.04
Price_apple 0.92 1.00 0.06 0.01 -0.03 -0.00 0.00 -0.00 0.01 0.00 0.00 -0.04 -0.03 0.03 -0.03 -0.04 0.02 0.04
Size_Bytes_apple 0.06 0.06 1.00 0.16 -0.08 0.09 0.17 0.09 0.20 0.01 0.02 0.04 0.04 0.39 0.04 0.04 0.07 0.11
Rating_google 0.00 0.01 0.16 1.00 -0.13 0.06 0.34 0.06 0.80 0.02 0.01 -0.24 -0.26 0.15 -0.26 -0.24 0.16 0.14
DeveloperId_apple -0.02 -0.03 -0.08 -0.13 1.00 -0.04 -0.10 -0.04 -0.14 -0.01 -0.01 0.46 0.43 -0.01 0.43 0.46 -0.00 -0.05
Minimum Installs_google -0.00 -0.00 0.09 0.06 -0.04 1.00 0.06 0.95 0.07 0.00 0.00 -0.04 -0.05 0.06 -0.05 -0.04 0.02 0.02
Average_User_Rating_apple 0.00 0.00 0.17 0.34 -0.10 0.06 1.00 0.06 0.83 0.01 0.02 -0.17 -0.15 0.19 -0.15 -0.17 0.10 0.17
Maximum Installs_google -0.00 -0.00 0.09 0.06 -0.04 0.95 0.06 1.00 0.07 0.01 0.00 -0.05 -0.05 0.07 -0.05 -0.05 0.02 0.03
Total_rating 0.00 0.01 0.20 0.80 -0.14 0.07 0.83 0.07 1.00 0.02 0.02 -0.25 -0.25 0.21 -0.25 -0.25 0.15 0.19
Date_google_agg 0.00 0.00 0.01 0.02 -0.01 0.00 0.01 0.01 0.02 1.00 0.69 -0.10 -0.15 0.00 -0.15 -0.10 0.00 -0.00
Date_apple_agg 0.01 0.00 0.02 0.01 -0.01 0.00 0.02 0.00 0.02 0.69 1.00 -0.14 -0.10 0.01 -0.10 -0.14 -0.01 -0.00
Year_apple -0.03 -0.04 0.04 -0.24 0.46 -0.04 -0.17 -0.05 -0.25 -0.10 -0.14 1.00 0.93 0.10 0.93 1.00 -0.01 -0.03
Year_google -0.02 -0.03 0.04 -0.26 0.43 -0.05 -0.15 -0.05 -0.25 -0.15 -0.10 0.93 1.00 0.12 1.00 0.93 -0.02 -0.04
Size_google_scaled 0.03 0.03 0.39 0.15 -0.01 0.06 0.19 0.07 0.21 0.00 0.01 0.10 0.12 1.00 0.12 0.10 0.06 0.10
Released_google_year -0.02 -0.03 0.04 -0.26 0.43 -0.05 -0.15 -0.05 -0.25 -0.15 -0.10 0.93 1.00 0.12 1.00 0.93 -0.02 -0.04
Released_apple_year -0.03 -0.04 0.04 -0.24 0.46 -0.04 -0.17 -0.05 -0.25 -0.10 -0.14 1.00 0.93 0.10 0.93 1.00 -0.01 -0.03
Names_Not_Same 0.02 0.02 0.07 0.16 -0.00 0.02 0.10 0.02 0.15 0.00 -0.01 -0.01 -0.02 0.06 -0.02 -0.01 1.00 0.44
Av_App_name_length 0.04 0.04 0.11 0.14 -0.05 0.02 0.17 0.03 0.19 -0.00 -0.00 -0.03 -0.04 0.10 -0.04 -0.03 0.44 1.00
Price_google
Price_apple
Size_Bytes_apple
Rating_google
DeveloperId_apple
Minimum Installs_google
Average_User_Rating_apple
Maximum Installs_google
Total_rating
Date_google_agg
Date_apple_agg
Year_apple
Year_google
Size_google_scaled
Released_google_year
Released_apple_year
Names_Not_Same
Av_App_name_length
1.0
0.8
0.6
0.4
0.2
0.0
−0.2

Most influential features

Our Steps:

1. At the beginning of the project, our target was to characterize applications with a high rate (over 4.5). During the project we noticed that it is possible to combine the scores from the two app stores and conduct research on them (“Rating_google”, “Average_User_Rating_apple”).

2. So we created new feature “Total Rating” that sum both of the rating (Google & Apple) and try again to search for any correlation between features for total rating 9 and above.

3. Fortunately there are two features for categories (One for Apple and One for Google), so we transform the data into dummies so it can be correlated with the rating feature.

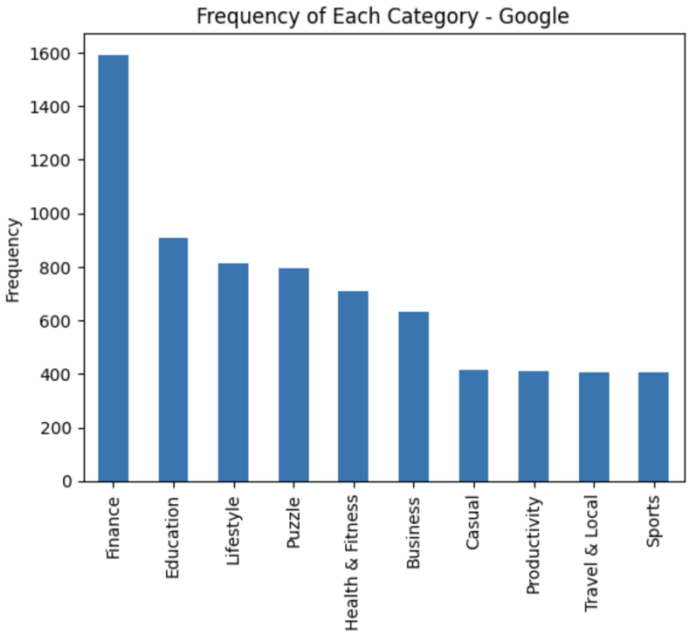
The top results:

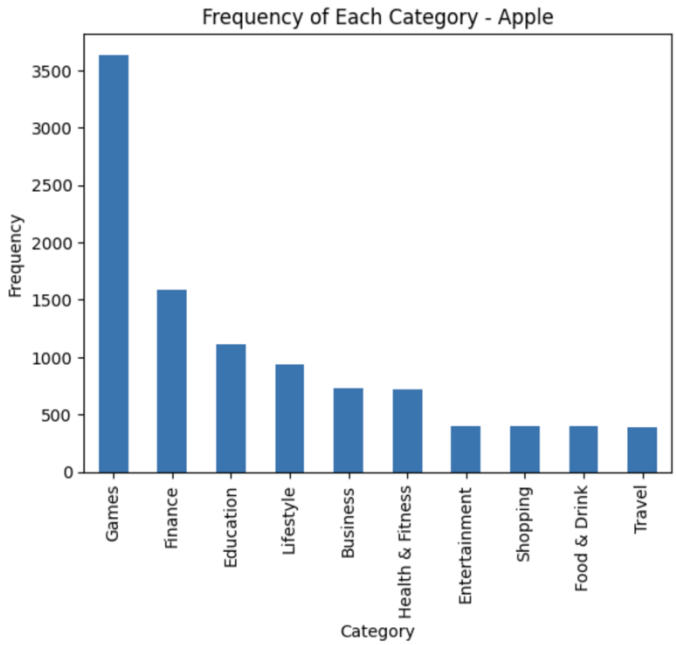
```
Total_rating:                    1.0000
on_apple_Games:                  0.3049
on_google_Puzzle:                0.1135
on_google_Simulation:            0.1033
on_apple_Finance:                0.0886
on_google_Finance:               0.0871
on_google_Casual:                0.0857
on_google_Action:                0.0843
on_google_Arcade:                0.0786
on_google_Role Playing:          0.0767
on_google_Adventure:             0.0698
on_google_Card:                  0.0677
```

```
on_google_Strategy:                    0.0673
```

No correlation was found between the categories and the Total rating.

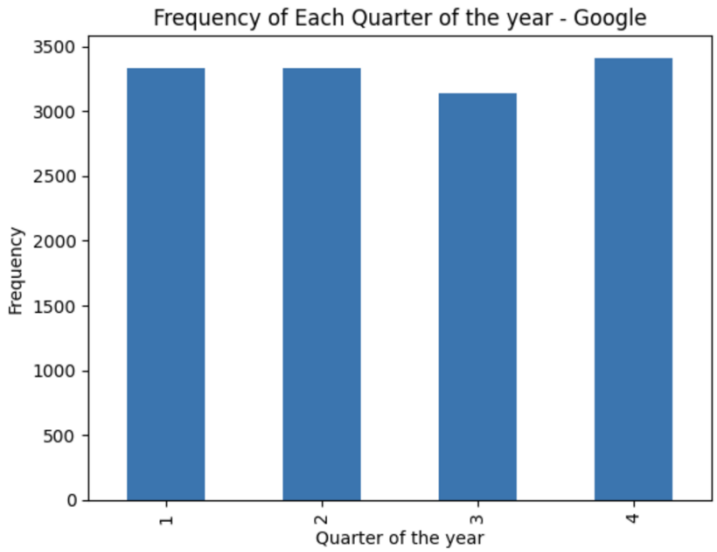
We continue to check for the best categories that have ranked **9 and above**:

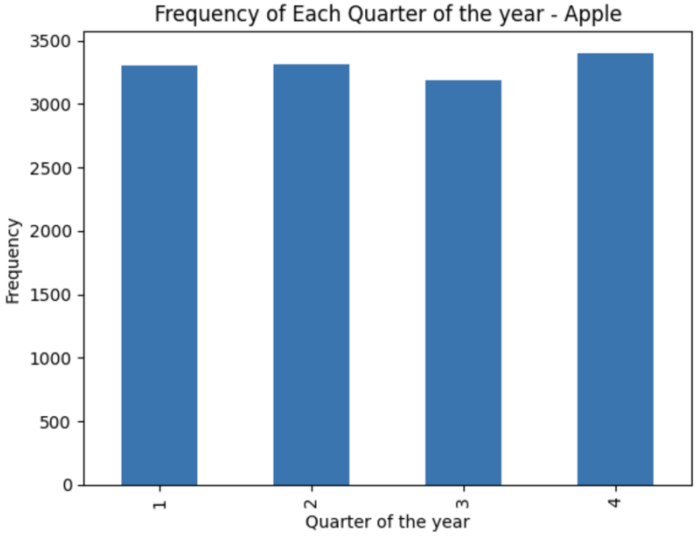




In Google, the **'Finance'** category is the highest, while in Apple, it's **'Games'**

We attempted to determine the most relevant quarter of the year for publishing apps, and it appears that the distribution is fairly balanced across quarters:





Conclusion

While both datasets originated from the same publisher, it's important to note the variations in terminology, feature types, and data formats encountered during the analysis. These disparities posed challenges in data consolidation and required preprocessing before meaningful insights could be derived.

Despite extensive analysis, no definitive formula for achieving high app ratings emerged. However, noteworthy observations included the relatively high ratings received by financial applications. Additionally, the absence of concrete recommendations regarding factors such as publication timing, app size, or pricing underscores the complex and multifaceted nature of app development and user preferences.

One key insight gleaned from the analysis was the significant synchronization observed in app release timelines across Google Play and Apple's App Store. This alignment suggests a deliberate strategy by developers to capitalize on dual-platform launches, potentially maximizing audience reach and engagement.

Further investigation into the relatively high ratings received by financial applications revealed intriguing insights into user preferences and expectations within this category. Features such as secure transaction processing, personalized financial insights, or user-friendly budgeting tools may play a pivotal role in driving positive user experiences and ratings within the financial app domain.

General Conclusions:

1. Google Play has more free apps compared to Apple's App Store.
2. Most apps on both platforms (google and apple) are available for free.
3. Google Play showing a slightly higher number of releases in recent years compared to Apple's App Store.
4. If an app is recently launched on one platform, it's highly likely to be new on the other platform as well. This synchronization suggests that developers often release their apps simultaneously or closely following each other on both platforms, potentially aiming to expand their audience and user base.
5. There appears to be a synchronization or coordination in the release timelines of apps across both platforms. This alignment may signal developers' intentions to launch simultaneously or closely in tandem on both platforms, aiming to maximize

their outreach and influence, respond to market dynamics, or leverage promotional prospects.

Based on the findings, developers may benefit from adopting a coordinated release strategy across multiple platforms to leverage the synergistic effects observed in app adoption and user engagement. Additionally, prioritizing features or characteristics that contribute to higher ratings, such as intuitive user interfaces or robust functionality, could enhance overall app quality and user satisfaction.