Google Play Store Visual Analytics
2023 1–6

Google Play Store Visual Analytics
Visualitation and Comparison

Paolo Caruso and Cristian Fioravanti

Abstract
The purpose of this project is to develop an interface for analyzing and comparing data from applications on the Google Play Store. Targeted towards developers, advertisers, and market analysts, it enables in-depth exploration of app features for a comprehensive understanding of the mobile application landscape.

Introduction

In an era where mobile applications dominate the digital landscape, understanding the dynamics of the app market can provide valuable insights. This project aims to explore and analyze data from applications available on the Google Play Store.
The main issue we sought to address involves comprehending trends and patterns within the Google Play Store. This encompasses the analysis of the most popular app categories, the distribution of user ratings, the correlation between reviews and the number of downloads, and so forth.
We chose to work with Google Play Store data for several reasons. Firstly, the Google Play Store is one of the largest app markets globally, making its data highly relevant for any analysis of the mobile application industry. Additionally, these data feature a wide range of attributes, such as category, rating, price, and number of downloads, which can be utilized for various interesting analyses.
Using the D3.js library, we have created a series of interactive visualizations that allow users to explore the data intuitively. This project not only provides insights into the app market but also serves as an example of how data visualization techniques can be employed to make vast amounts of information easily accessible and understandable.
Furthermore, we have provided users with two modes of application usage: in the ”Visualize” mode, users can view all data in the database, examine their features, and analyze trends. In the ”Compare” mode, users can select two groups of applications for a more detailed analysis.

Dataset

The dataset used for this project (*1*) contains data from 9660 applications, representing a reduced portion of the total number of applications currently available on the Google Play Store. In particular, for each application, data from 13 different features are reported, and we have chosen the 9 most interesting and useful for the data analysis:

- **App**: the name of the application.
- **Category**: the category of the application. Some examples include ”GAME,” ”BOOKS_AND_REFERENCE,” and ”EDUCATION.”
- **Rating**: the application’s rating on a scale from 1.0 to 5.0.
- **Reviews**: the number of reviews received.
- **Size**: the size of the application in kilobytes (kB).
- **Installs**: the number of installations of the application, rounded to the lower of the power of 10 or the power of 10 multiplied by 5.
- **Price**: the price of the application.
- **Type**: the type of application, indicating whether it is free or paid.
- **Content_Rating**: the age classification of the application, divided into Teen, Adults only, Mature, Everyone 10+, and Everyone.

The other 4 available features that we did not consider are:

- **Last_Updated**: The date of the last update of the application.
- **Current_Ver**: The current version of the application.
- **Genres**: The genres of the application.
- **Android_Ver**: The minimum required Android version for the application.

These data provide a detailed overview of the characteristics of applications in the Google Play Store, allowing for an in-depth analysis of trends and relationships between different variables.
For practical reasons, we manage the dataset in a MySQL database.

Data Manipulation and Preprocessing

We initiated the data preparation process by examining the dataset as it is.

Firstly, we noticed that some rows had incomplete data, specifically *NaN* values in the **Rating** and **Size** columns. For these cases, we decided to remove the respective rows to avoid skewing the analysis with fake data.

Subsequently, we cleaned the data by removing some unnecessary characters. In particular, in the **Size** column, there were size indicators such as *kb* or *MB*. We cleaned these redundant characters and standardized the data by converting all measurements to kilobytes. In the **Price** column, we removed the dollar symbol ($), and in the **Installs** column, we removed the plus symbol and commas between digits.

After these preliminary operations, we had a dataset of **7023** rows with consistent data for our purposes.

The next step was dimensionality reduction. In our case, we chose PCA, considering only the continuous columns (Rating, Reviews, Size, Installs, and Price) and omitting the categorical ones (App, Category, Type, Content_Rating). Therefore, the PCA process will only consider information related to the mentioned 5 features.

Upon loading the application page, data will be requested from the MySQL database and used for PCA computation. The result will be written to the dataset as two additional features for each row. This way, the data will be ready to populate the application's 2D scatterplot.

Furthermore, users can recalculate the PCA of the data as they please in the "Compare" section of the application to best visualize differences between the apps of their interest.

Visual Solution

The application utilizes multiple views to assist the user in understanding and analyzing the available data. The views are organized to make it easy for the user to leverage their potential and features. The various views will be presented in detail, distinguishing between the presentation in the "Visualize" mode and the "Compare" mode when needed.

Category Selection

For each row, a distinct category present in the applications is represented, followed by its corresponding checkbox. The application assigns a unique color to each of these categories, allowing for the unique distinction of applications in each specific category.

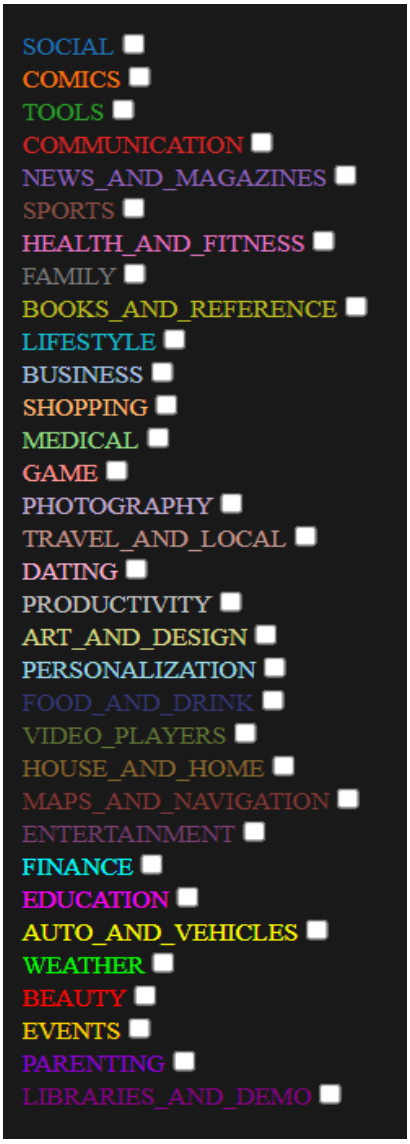

Figure 1. Screen category section

Scatterplot Section

In this section, thanks to dimensionality reduction PCA, all applications are represented in a Cartesian plane. Each application is visualized as a colored circle of the same color as the category to which it belongs. On the header of the application page, there are buttons that let the user zoom and brush the Scatterplot. In the "Compare" mode, there is an additional button that allows the user to re-run PCA for the selected data only, making it easier to visualize them in the Cartesian plane.

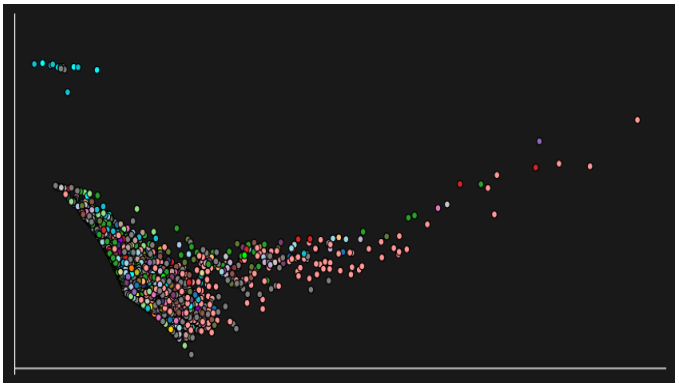

Figure 2. Screen category section

Parallel Coordinates Section

This view allows the representation of the features of all the data in our dataset. Each line in the graph represents an application with feature values as the values of the axes that the line crosses. The axes represented in this section are: Rating, Reviews, Size, Installs, Price, Content_Rating, and Type. Each line belonging to the represented application is colored the same as the category to which it belongs.

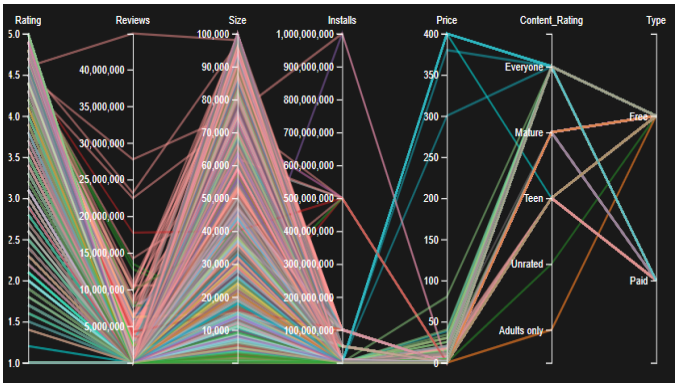

Figure 3. Screen category section

BoxPlot Sections

This view allows visualizing the distribution of data for the considered feature. This section hosts four different boxplots, each dedicated to specific attributes of the applications: Rating, Reviews, Installs, and Size. For the boxplot of the Rating attribute, a linear scale was chosen since the possible values range from 1 to 5. For the other boxplots, a logarithmic scale was chosen to best visualize the various sections of the boxplots. In ”Visualize” mode, the boxplots represent the distributions of all data in the dataset. In ”Compare” mode, a boxplot will be displayed for each group selected by the user. In ”Compare” mode, the sections of the boxplots assume the color of the group they represent: in the case of a category, the color of the category, and in the case of a selection from brushing, **red** is assigned to the first group, and **blue** is assigned to the second group.

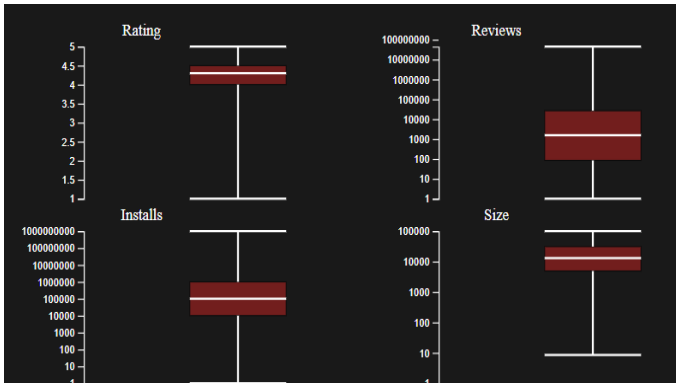

Figure 4. Screen category section

Histogram Section

This section hosts several histograms, each dedicated to specific attributes of the applications: Type, Content_Type, and Installs. For all histograms, a SymLog scale has been applied to best visualize all the data. In ”Compare” mode, the rectangles used to visualize the data will have, as in the case of the boxplots, the color of the group to which they belong. The data and scales of the three histograms are updated in real-time based on the data selected within the Scatterplot.

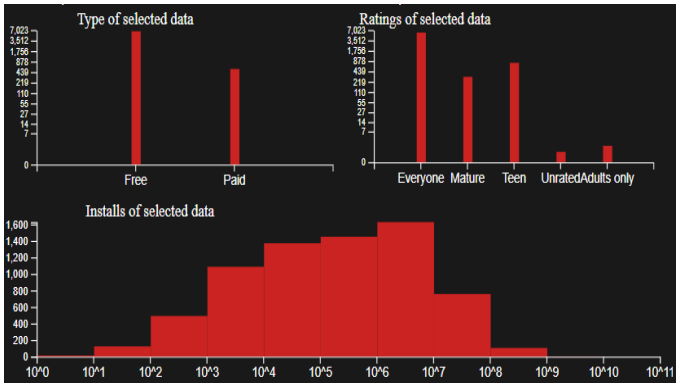

Figure 5. Screen category section

Interactions

We have planned several possible interactions between the user and the views available in the application.

The various possible interactions will be presented in detail below, along with their usefulness during use in both ”Visualize” and ”Compare” modes.

Visualize

1. **Category Selection**: Selecting a category from the list will highlight the applications in that category in the

Scatterplot that have not already been selected. Adding these applications to the selected ones will update the Histogram data in real time, updating the scales with the appropriate data if necessary. Deselecting a category will deselect all applications already selected with that category and remove the highlighting.

2. **Mouseover on Scatterplot Elements**: This action will show the user the features of the desired element.
3. **Brush on Scatterplot**: This action will select the applications included in the selection, adding them to the already selected applications. This will update the Histogram data in real time, updating the scales with the appropriate data if necessary. Only one brushing can be done in this mode.
4. **Brush on Parallel Coordinates**: This action will highlight in **yellow** the applications already selected in the Scatterplot. Performing multiple brushes on multiple features will highlight only the already selected applications that satisfy all the chosen criteria. It is possible to deselect and move the already made selection on each axis.
5. **Selection of a portion of BoxPlot**: Clicking one of the 4 sections of one of the BoxPlots will highlight in **yellow** the already selected applications that have the value of that category in the range of values that the selected section represents. Clicking multiple sections of the same or different BoxPlots will highlight only the already selected applications that agree with all the chosen value ranges. The portion chosen by the user is highlighted with the same color as the BoxPlot but in a more intense shade.
6. **Mouseover on portions of BoxPlot**: Performing this action on each of the 4 sections of one of the BoxPlots will show a tooltip that will show detailed information through a tooltip regarding the number of applications that satisfy or do not satisfy that particular constraint.
7. **Mouseover on Histogram rectangles**: Performing this action on each rectangle present in the 3 available views will show a tooltip that will show the value that that rectangle takes and therefore how many applications have the value represented on the abscissa.

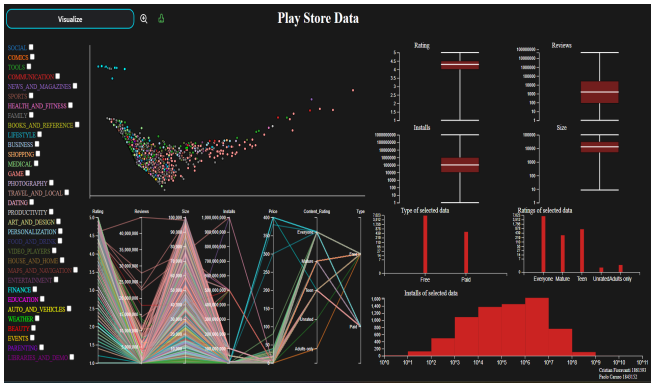

Figure 6. Screen of the application in Visualization Mode

Compare

1. **Selection and Deselection of a Category**: Selecting a category from the list will select the applications belonging to that category as a group to compare. These will be highlighted on the Scatterplot. Deselecting a category from the list will remove those applications as a group to compare; furthermore, these will be deselected in the Scatterplot as well.
2. **Mouseover on Scatterplot Elements**: The behavior of the Visualize mode is maintained.
3. **Brush on Scatterplot**: This action will set the applications included in the selection as a group to compare. At most, two brushes can be made in this mode (one for each group). If a category has already been selected as a group, only one brush can be made.
4. **Recompute PCA**: Once two groups have been selected, it is possible to click the button next to the Scatterplot, which will recalculate the PCA on the data of the applications of the two groups and update the Scatterplot with only the circles of those two groups.
5. **Brush on Parallel Coordinates**: The behavior of the Visualize mode is maintained.
6. **Mouseover on Histogram rectangles**: The behavior of the Visualize mode is maintained.

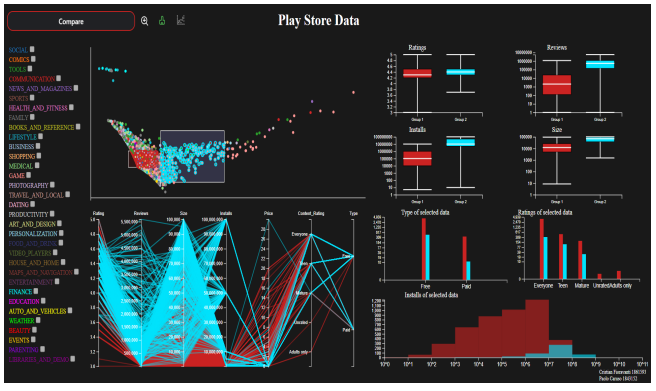

Figure 7. Screen of the application in Compare Mode

Analytics

Now we will list the Analytics tools provided by the application:

- **Parallel Coordinates Data in Compare Mode**: When applications are selected for a group, they will appear in real-time on the Parallel Coordinates chart with the corresponding group color.
- **Boxplot in Compare Mode**: As anticipated, minimums, maximums, and quantiles will be calculated in real-time for each group of applications. The values on the axes will also be consistent with the minimum and maximum values present in both groups. This way, the user will always see the difference in data distribution between the groups.
- **Real-time Update of Histograms**: In both modes, the Histograms show the data of the selected applications. Therefore, the values on the ordinate axes will be updated in real-time based on the values present in the selection or groups. Consequently, the data representing the Histograms will also be updated in real-time with user interactions.

Insights

Numerous insights can be derived from the application, and by analyzing different groups of apps, it's possible to discover common patterns as well as exceptions or new information present only in specific groups of apps. Here are some potential insights that can be gleaned using our application:

- In the analysis of the 'NEWS_AND_MAGAZINE' category, by selecting the checkbox in the compare mode and subsequently recomputing PCA exclusively for this category, intriguing findings come to light. Contrary to expectations, it appears that the installation dynamics of these applications are minimally influenced by overall ratings. Instead, a noteworthy trend emerges: the pivotal role of user reviews. Significantly, the most highly regarded applications, enjoying widespread user adoption, exhibit a substantial number of reviews, ranging from 100,000 to 900,000, coupled with ratings falling within the range of 4 to 5. In contrast, less-installed applications tend to accumulate fewer reviews (between 0 and 100,000) and showcase variable ratings ranging from 1 to 5. A notable exception that challenges the conventional pattern is represented by the Google News app. Despite sporting a rating of 3.8, it steadfastly maintains its position at the top of the ranking as the most-installed app by users. An exemplar of unique appeal and resilience in the competitive landscape of applications.
- In the examination of the 'GAME' category, achieved by selecting the category via the checkbox in both modes, a notable departure from the previous case is observed. Unlike the 'NEWS_AND_MAGAZINE' category, the pivotal characteristic distinguishing applications is not found in user reviews; rather, it is embodied in the rating. Notably, an optimal application is identified by a rating falling within the range of 4 to 5. Furthermore, the size of the application does not emerge as a substantial determinant in the decision-making process regarding app installations. In this context, reviews appear to exhibit a proportional relationship with the app's popularity.
- By selecting the category "FINANCE" through the checkbox in both modes, free apps are prevalent, with most of the paid ones appearing to be humorous apps that ask for €399.99 to be installed. Also, for this category, a rating from 4 to 5 represents the predominant trend, and the most popular apps are those with several reviews greater than about 30,000. The less popular apps have the common property of having a size ranging from 0 to 50 MB, while the more popular ones range from 30 to 100 MB fairly evenly.
- To examine the different characteristics that distinguish a highly installed app from a less installed one, we have delimited, in "Compare" mode, two distinct groups of applications. In the first group, we identified the applications with the highest number of installations using the Parallel Coordinates brush on our dataset. In the second group, we focused on applications with the lowest number of installations, also identified through a brush-based approach on Parallel Coordinates.
  Considering the boxplots related to Reviews generated by the two groups, significant contrasts emerge. If we define Q1 as the value of the first quartile and Q3 as the value of the third quartile of the boxplot of the first group, we can observe that the value of Q1 = 1,000,000 and Q3 = 5,000,000. Comparing these values with the boxplot generated by the second group, we observe a notable disparity in the quartiles.
  In particular, if we define Q1' as the value of the first quartile and Q3' as the value of the third quartile of the boxplot of the second group, we can define Q1' = 0.005% of Q1, while Q3' corresponds to about 0.1% of Q3. These calculations highlight a marked difference in the reviews attributed to less installed applications, showing a significantly lower number of reviews compared to their more popular counterparts. This suggests that less adopted apps receive an extremely reduced volume of reviews compared to more installed ones. Regarding reviews, it is evident how the distribution of more frequently installed applications is significantly concentrated. The values in the quartiles range from 3.7 to 4.8, indicating a narrow variation in ratings. This suggests that more popular apps largely receive reviews of similar quality, with scores falling within a narrow range. On the contrary,

the boxplot related to other applications shows a wider distribution, spanning all possible ratings from 1 to 5. This extended variety of ratings highlights greater diversity in user opinions. However, it is important to emphasize that this breadth in distribution may not be trivial. Less installed applications may find themselves in this position for various reasons. They could be of lower quality, receiving lower ratings and, consequently, occupying the lower end of the rating scale. At the same time, they could be voted on by only a limited number of users, often friends or acquaintances, leading to greater variability in scores. This situation makes it easier to achieve a maximum score (5) or a minimum (1) when the number of reviews is limited.

Related Work

We have identified numerous studies related to this dataset, but most of them have focused on Data Visualization analysis rather than a Visual Analytics approach. However, we cannot overlook the contribution of Aiastan Sherniiazov in his work titled "How To Make Your App Popular?" (*2*). His research has been inspiring, producing significant results through in-depth data analysis, providing a comprehensive view of user behavior and the success dynamics of applications.

Among the most relevant findings:

- A high number of reviews is correlated with a higher number of installations.
- User feedback is crucial for the success of an application.
- Applications with higher ratings tend to be installed more frequently.

The research approach focusing on global application trends has sparked our interest in uncovering hidden patterns in applications of a specific category or with particular specifications. From here arises the idea of our project: creating a space where users can visualize and compare even a limited number of applications, all on demand. By providing users with the ability to understand trends, patterns, and relationships among various application variables.

Conclusions

We are pleased with the outcome of our project. The goal of creating an effective tool to explore the dynamics of apps in the Google Store Data, identifying significant trends and patterns, and providing valuable insights for service users, has been fully achieved. The ability to directly compare groups of applications enriches the analytical experience as intended. This project manages to add new perspectives and confirm some of the relevant findings already present in other works such as (*2*).

Regarding future developments, we find it interesting to explore the possibility of:

- Allowing users to decide which types of histograms to display.
- Implementing different selection modes, including the use of Parallel Coordinates or Boxplots.
- Offering the ability to add or remove applications within the Scatterplot, customizing the visualization to make it lighter or more detailed.

These proposals aim to further improve the versatility and usability of the tool, allowing users to adapt it to their specific needs and preferences.

References

1. Dataset: https://www.kaggle.com/datasets/lava18/google-play-store-apps
2. Related work: https://www.kaggle.com/code/aiastansherniiazov/how-to-make-your-app-popular