Google Play Store: User Story 98 Report

**User would like to know the mean price and mean number of installs for Health and Fitness**

KenzieM

1. Introduction

For this report, I used data on apps from the Google Play Store to answer the user story: “User would like to know how the mean price and mean number of installs for Health and Fitness.” The data contains 10,841 rows and 13 columns. The columns of this data set include information such as name, category, genre, rating, size, number of installations, price, content rating and more. Every row of the set represents an individual application available on the Google Play Store. In the following report, I will discuss how I transformed the data in order to answer the question, and provide some visual aids to better understand that answer.

1. Data

To begin, I needed to clean the master csv. This involved deleting several duplicate rows, as well as one row that had all of its information entered incorrectly. Once these rows were removed, I reset the index and inspected the columns. I found four columns with numerical data that were not converted upon download. I removed unnecessary symbols from this data, like plus signs, units, commas, and dollar signs, and converted these to the correct datatypes. The size column needed an extra conversion as some of the apps were measured in kilobytes and others in megabytes. I’ve included the code I used for that transformation below.

|  |
| --- |
| def new\_size(x):  if 'k' in x:  return (float(x.replace('k', ''))) \* 1000  if 'M' in x:  return (float(x.replace('M', ''))) \* 1000000   df['Size'] = df['Size'].apply(new\_size) |

After completing these data type conversions, I saved the new clean csv in a separate file to work on this story.

For this story, I began working with a cleaned csv containing the data mentioned above.This csv contains 10,357 rows and 13 columns. I used boolean indexing to create a new dataframe from the cleaned csv containing only apps in the health and fitness category. I also dropped most columns, keeping only the App name, Type (paid or free), Installs, and Price. This new data set contained 306 entries and 4 columns representing 306 different apps in this category.

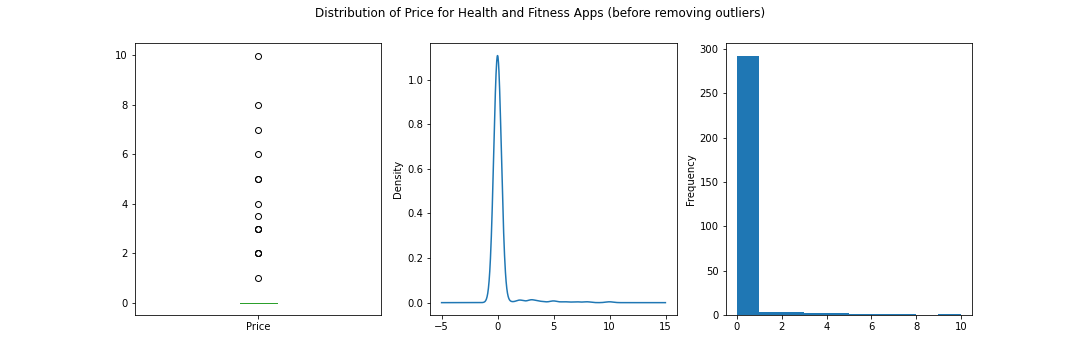
I used the DataFrame.describe() method to take a quick look at some descriptive statistics, and I can immediately see that the installs column has a huge difference between the mean and the median, and that a very large percentage of these apps are free (price = 0). I decided it may be more useful to separate these apps into paid apps and free apps to better examine the means of installs and prices.

The set of free apps contains 291 rows and the paid apps contains 15. I performed descriptive statistical analysis on both the entire set and the grouped sets, and I will describe this further in the following section.

1. Methods

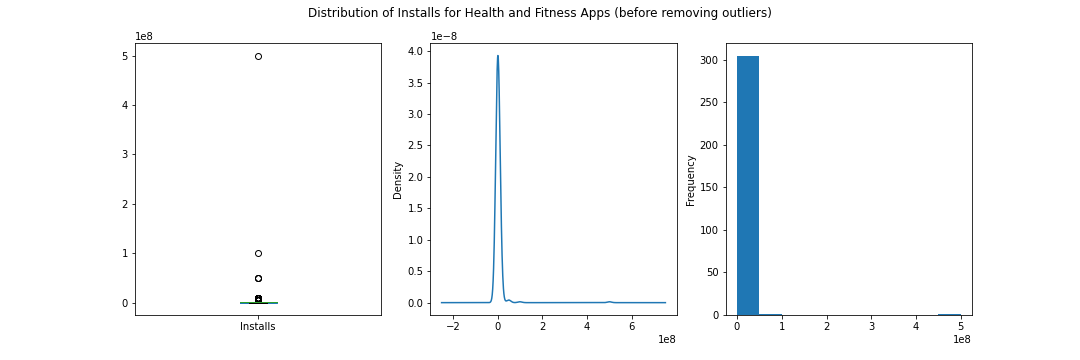
In this section, I will describe the methods I used to derive my results and reach my conclusions. As stated above, I performed the same general process on three different sets of data. One contained all of the Apps in the health and fitness category, and the other two were these same apps, but divided into two data sets based on whether their type was free or paid. Here I will describe the general methods I used, and in the following Results section, I will discuss how these differed based on the different data sets.

The first column I examined was the price column. In order to visualize the distribution of the data, I plotted a histogram, kde density plot, and a box plot.



Due to the nature of the data already being ‘pre-binned’ in a sense, the histograms did not give me better information than the others, so most of the visualizations you see from now on will resemble kde density plots and box and violin plots. As we can see above, most of these apps are free. From a visual examination, our box plot is telling us that any app that isn’t free is considered an outlier in this group. Since we still want to consider the paid apps (removing outliers in the Price category would result in removing all of the paid apps), I split the set in two and performed another analysis. I also kept the set together for an analysis of only the installs.

Taking a look at the same plots as above, but for the installs category, we can see similar distributions with even more extreme positive outliers.



The first order of business for each set of data was to remove the outliers. There are many different methods for doing this, and I chose to use the interquartile range (IQR) because it’s a pretty simple and straightforward calculation. This method takes the range between which 50% of the data falls and uses a multiplier (in this case 1.5) and then adds or subtracts that value from the 75th percentile or the 25th percentile respectively to create an even larger range under which we can be confident that most of the data falls. Here is an example of the code I used to do this for the installs column:

|  |
| --- |
| q1\_installs = haf['Installs'].quantile(.25) q3\_installs = haf['Installs'].quantile(.75) iqr\_installs = q3\_installs - q1\_installs max\_installs = q3\_installs + 1.5\*iqr\_installs min\_installs = q1\_installs - 1.5\*iqr\_installs print(min\_installs, max\_installs) |

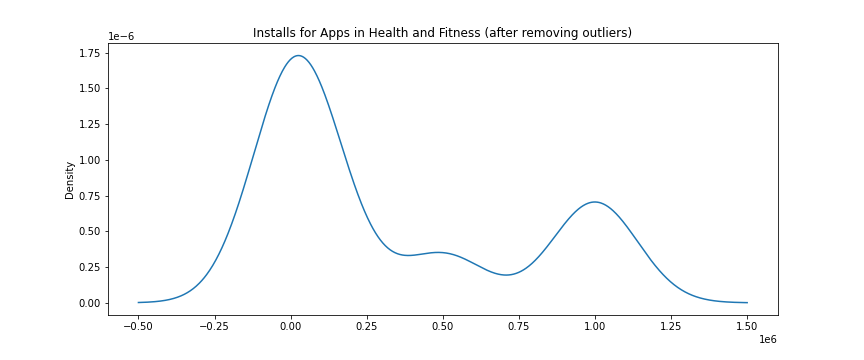
Any value that out of the range of (min\_installs, max\_installs) was then removed from the data set. I found many outliers in the installs column of every data set. I did not remove price outliers from the main data set, because that would remove all free apps. I also did not examine outliers for the free apps because they all have the same price. I did check for price outliers in the paid apps, but the highest price was still within the range determined by the IQR.

Once I had my data sets free of outliers, I was better able to determine realistic means for the apps in this category and I will discuss those findings in the next section.

1. Results

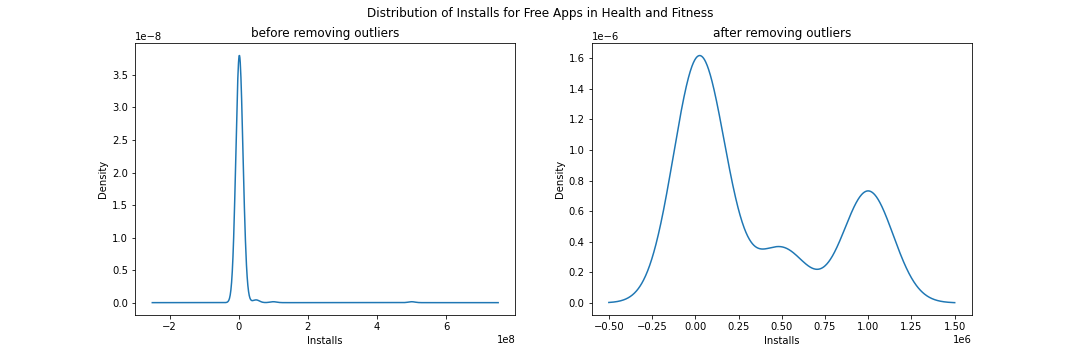
Before splitting up the data set or removing outliers, our mean price for apps in the health and fitness category is about $0.21, and our mean installs is about 4.4 million. To get an idea of how much our data is being influenced by outliers, it’s useful to take a look at median as well. The median price for all apps in health and fitness is $0, and the median installs is about 500,000. At a glance it seems that both of these values are positively skewed, which makes sense, since negative installs and price aren’t possible. Furthermore it seems that the installs column is far more influenced by outliers than the price column.

After removing outliers from the whole set, our new mean for installs is about 324,000 (with a median of 75,000 which is much closer than before), and the price mean did increase slightly to about $0.28. The figure below is a new distribution curve of the installs with the outliers removed, and is much more readable and informative than the original.



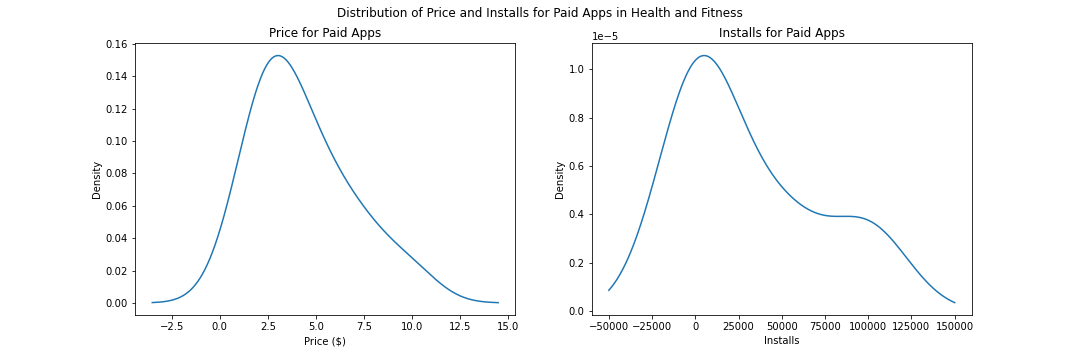
I believe splitting the data into paid and free will give us more useful information, and I will share the results of that analysis here. I did an analysis of installs on both groups and an analysis of price on the paid group. I will start with the analysis of the free apps. Before removing outliers, the free apps had a mean installs of about 4.68 million (median 500,000).

Using the same method as before, I inspected each group for outliers and found many in the free group. After removing the outliers, the new mean for installs was about 345,000 (median 100,000). Below are the density plots of the free group before and after removing outliers (please note the differences in axes of each plot).



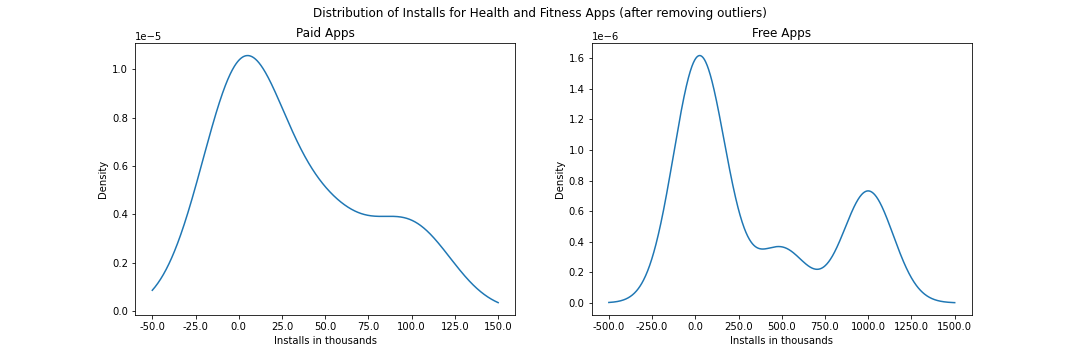
We can see that this density function and the new mean estimate are very similar to the ones for the whole group, which makes sense, as most of these apps are free.

Finally I examined the installs and prices of the paid apps. There were no outliers for either column in this group. The mean price for these apps is $4.29 (median $3.49) and the mean installs is about 32,000 (median 10,000). The plots below show density functions for both price and installs for the paid apps (please note the differences in axes of each plot).



1. Conclusion

In conclusion, an overwhelming majority of apps in this category are free. Since it is rare for apps to have more than 2,500,000 installs, we removed these outliers to give the user a better estimate of how many downloads they can expect for their app. The adjusted average number of installs for free apps is 345,000 compared to about 32,000 for paid apps. Below are two density plots showing installs for each category: paid and free.



Furthermore, the user should keep in mind that the average price for the paid apps is $4.29. I would recommend the app or very inexpensive to install. Perhaps the user can look into in-app purchases or ads to cover some of the cost of creation, but I don’t have data about those to provide more insight.