**Google Play Store Report and Analysis**

1. **Introduction**

This report will include data and information regarding the Google Play Store and will be used to answer a user story. The data utilized for answering the user story consists of 10,841 rows and 13 columns all related to Google Play Store statistics and metrics. These ranged from the category of the app, the overall rating, installs, current version, price, and other similar factors. Meaning there are 10,841 apps and relevant information contained within the data.

The question we are tasked with is, “User would like to know how price affects the amount of downloads for the ENTERTAINMENT category”. In this report I will be discussing more in-depthly in the following sections about the data, methods used, analysis performed, and the overall results. Since this user story asks about a relationship or a correlation between price and the amount of downloads for the entertainment category, I will be using two statistical tests in this report to determine if there is indeed a relationship between these two variables.

1. **Body**

**Data:**

To start off this analysis I initially gathered all of the data into a data frame for inspection of rows which will be relevant to the user story and to prepare statistical testing on the relationship between price and downloads. After creating a data frame for all of the relevant data I decided to start inspecting the data and determine areas which would need cleaning for the analysis. Since the most important variables in this report will be Price and Installs (Downloads based on user story) I utilized the isnull method offered by Pandas on both of these columns. Fortunately, there were no missing values so the next step was to drop all of the irrelevant columns which would be unneeded for the analysis later on in the report. Based upon the user story, the only columns I need to keep are the Category, Price, and Installs.

At this point there was a data frame consisting of Category, Price, and Installs and the next step to take was to further clean these columns data to be better prepared for analysis. I did this by checking the unique values of each column to find the next steps which should be taken. In the installs column this resulted in the need to remove commas and addition symbols which was done by utilizing the .replace method. Additionally, there was one unique value which I knew needed to be removed, which was “Free”. The reason this row was removed is “Free” is irrelevant to the data in this column and needed to be a numerical value. The next step was to convert the column to a float data type. The reason I decided upon float instead of integer was to make any mathematical operation work smoothly because I knew that a price column would need to be a float data type as well (this is because of prices like 0.99, 1.99, etc.). The next step was then inspecting the Price column, I did this by also checking the unique values in the column and found one value which did not fit. This value was “Everyone” and did not fit in with the numerical values in the column so I removed the row from the data frame. The next thing I noticed was each value had a dollar sign symbol which would need to be removed for mathematical operations, I did this by utilizing the .replace method just like in the installs column. The last step to preparing this column was to assign the data type as a float.

The final step to preparing the data for analysis was to prepare the data frame with only Category values which were entertainment apps. Now that the data was prepared we can now move onto the methods used for analysis and my reasoning behind them.

**Method:**

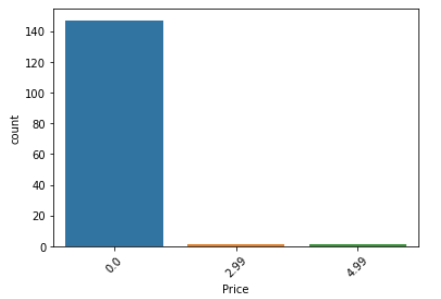
To refresh your mind, I will remind you of the user story “User would like to know how price affects the amount of downloads for the ENTERTAINMENT category”. Since this question asks about a correlation or relationship between price and downloads (or in the data’s case Installs) I needed to utilize a statistical test which best represented this problem. After researching I found that the most proper methods for this would be a correlational test and a regression test. I decided for the correlational test I would utilize Pearson Correlation to test for the strength of the association between price and downloads. I decided for the regression test that I should utilize simple regression in order to test how a change in the price variable predicts the level of change in the installs variable.

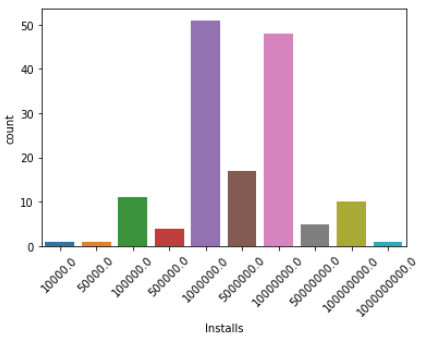
I decided against normalizing the data in my updated data frame for a few reasons. The first reason being the type of statistical tests I am utilizing do not require normalization. While I could normalize it will not have an effect on the end result for this data since the tests describe the nature of relationship between the price and install variables. Secondly, by utilizing the Pearson Correlation the correlation is already normalized by standard deviation so any normalization which is needed will be done through the test. For the Simple Regression test it has a similar reasoning that in theory that normalization should have no effect in the final result but can speed up the calculations which will be performed, and since the data which will be used is relatively small I found this to be unnecessary.

**Analysis:**

Prior to doing the analysis I knew there was still a small step to be performed and this was to create two new variables, one that would be assigned to the values in the Installs column and one that would be assigned to the values in the Price column. Now that all of the data is prepared for analysis, I will discuss the formulas and methods used. For the Pearson Coefficient test I decided to utilize the stats method from the Scipy library. This would result in an r-value and a p-value which represent the degree of correlation and probability of the result respectively. For the Simple Regression test I also used the Scipy library and utilized the stats.linregression method. This would result in an r squared value which tells us how well price can predict the amount of installs an app receives.

A limitation to this analysis, however, is the count of unique values in the Price variable. Many of the apps in this particular category (Entertainment) are at a price of 0, with only 2 out of 149 belonging to other unique price values. The vast majority of apps in this category have a price of 0 and below I will show a visualization which represents the count of each unique price value.



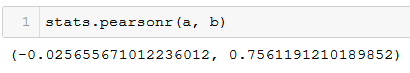
Additionally, I will provide a visualization of the count for unique Install values, in order for the viewer to gain a better understanding of the data.

Based upon this information it may become more difficult to determine if price does indeed impact the amount of installs since there are limited occurrences outside of the price of 0. Meaning if there were more occurrences the results which will be discussed in the next section may differ.

Now, it is time to discuss the results of the analysis.

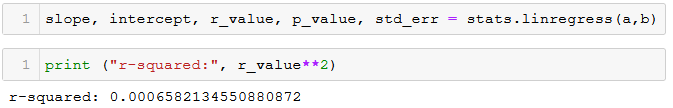
**Results:**

From the results of the Pearson Coefficient test the R-value was negative, but close to zero while the P-value is not less than 0.05 meaning it is not statistically significant. Like previously mentioned, I utilized the Scipy library and the stats method belonging to it and below I will show the formula used and the results.



In this picture the “a” variable represents price while the “b” value represents installs. Underneath are the results of the test, which on the left hand side shows the R-value that shows a slight negative correlation between variables. While the right hand side shows the P-value which is not less than 0.05 meaning the correlation is not statistically significant.

I wanted to test these variables with an additional test to make sure my findings were indeed accurate, and this was with the Simple Regression method. Like previously mentioned I utilized the Scipy library and the stats.linregress method and below I will show the formula used and the results.



In this picture the “a” variable still represents price and the “b” value still represents installs. At the bottom you can notice the r-squared value which explains how well the price can predict the installs value. Based upon the r-squared value which resulted from this formula since it is not close to 1 we can determine that price can not accurately be used to predict the installs value for the category of Entertainment apps.

1. **Conclusions**

To conclude this report for the app category of Entertainment to determine if price affects the amount of downloads I can safely make the statement that in this specific circumstance price does not affect the amount of downloads. However, based upon the amount of data available and the limitations discussed in the method section this does not mean the result is completely accurate. If there were to be more applications in the category of Entertainment which were not free and there were a sample of over 1000 apps the result could be completely different, since 98% (147/149) of the data consists of free apps . With the data available we can assume that there is a slight negative correlation between the price of an app in this category and the amount of downloads it receives, but not to a significant extent.

Had more data been available and additional time I believe the next step to take to improve this analysis would be to create a machine learning model for linear regression to better determine if price can indeed predict the amount of downloads an app would receive. However, with the amount of information available I believe that the two statistical tests utilized would give enough information to provide a proper answer for the user story.