**App Retention Data Analysis Report**

Laurence T. Burden

Purdue Global University

IN498: Bachelor’s Capstone in Analytics

Robert Kayl

December 17, 2023

**Introduction**

The analysis in this paper focuses on the retention rate of a mobile app from the Google Play Store.  The data is analyzed to determine the number of installs retained at the one, seven, fifteen, and thirty-day marks.  Several probability models are also created and reported on.

This work aims to give the app creator the knowledge necessary to improve the number of installs retained by customers.  The creator could use the data in this report to justify adjusting the advertisement for the application, or they could use the insights to work on changes to the app that will entice more people to retain the installation.  Either way, the goal will be to increase the persistent customer base.

**Methods**

The data was analyzed using a machine running Windows 11 and the internals include an Intel i7, 16GB of RAM, and a GTX1650M. The primary programming language used is Python 3.12 on PyCharm 2023.2.5 (Community Edition). Microsoft Excel® was also used for cleaning and initially exploring the data and combining the original nine files into one CSV file.

**Table 1: Python Package Details**

|  |  |
| --- | --- |
| Package Name | Version Number |
| Pandas | 2.1.3 |
| SciKit-Learn | 1.3.2 |
| Numpy | 1.26.2 |
| Matplotlib | 3.8.2 |
| Seaborn | 0.13.0 |

Numpy and Pandas were used for formatting the data during the analysis and for replacing missing values with zeros. Matplotlib and Seaborn were the plotting packages used to create visuals and charts. Finally, SciKit-Learn was used for its machine learning algorithms. The algorithms used include logistic regression, decision tree, random forest, and linear regression. SciKit-Learn’s train\_test\_split method was also utilized with an 80%/20% split.

**Results**

The following section summarizes key attributes of the data. Included is information on the data counts, trends over time, machine learning predictions, and the probabilities of those predictions.

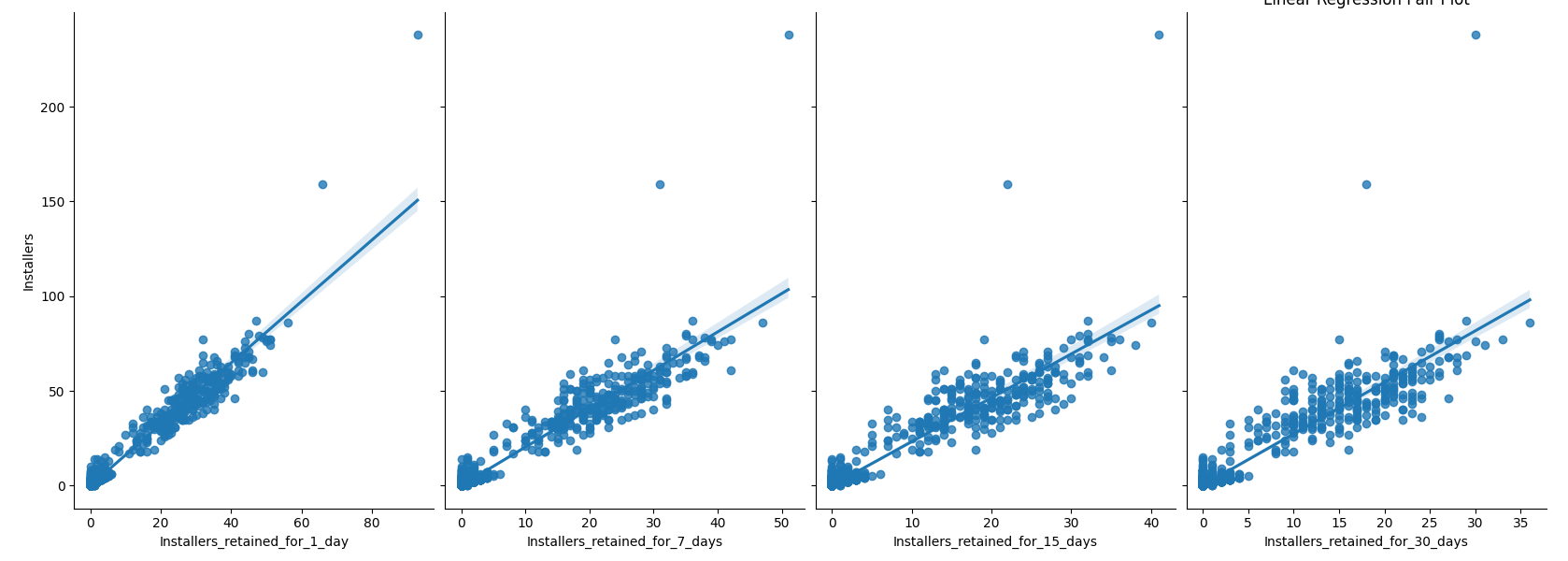
**Table 2: Attribute Counts by Month**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Month | Visitors | Installers | Retained 1 Day | Retained 7 Days | Retained 15 Days | Retained 30 Days |
| April | 8,222 | 2,217 | 1,294 | 1,010 | 858 | 720 |
| May | 7,673 | 2,196 | 1,372 | 1,082 | 924 | 769 |
| June | 6,914 | 1,912 | 1,227 | 980 | 842 | 707 |
| July | 6946 | 1,869 | 1,057 | 802 | 688 | 589 |
| August | 7,482 | 2,160 | 1,207 | 933 | 809 | 668 |
| September | 3,622 | 1,134 | 763 | 621 | 550 | 470 |
| October | 4,831 | 1,795 | 1,054 | 807 | 699 | 600 |
| November | 4,742 | 1,444 | 933 | 740 | 662 | 547 |
| December | 4,850 | 1,430 | 858 | 671 | 575 | 476 |
| **Total** | **55,282** | **16,157** | **9,765** | **7,646** | **6,607** | **5,546** |

Table 2 represents the high-level findings for the entire data set. There is a nine-month period covered from April 2021 – December 2021. 55,282 visitors viewed the app within the app store, with an average of 6,142 visitors per month. The retention rates are those customers who kept the app installed after the given period (1, 7, 15, and 30 days).

Linear regression was completed on the data after replacing all missing values with zeros. The number of retained users shows a positive correlation with the total number of installations. Various metrics used on the linear regression model back up this finding.

**Figure 1: Linear Regression Pair Plot (Installers vs. Installer Retention)**



**Table 3: Retention Predictions**

|  |  |
| --- | --- |
| Linear regression prediction for 1 install | ~0.3711 |
| Linear regression prediction for 2 installs | ~0.6951 |
| Linear regression prediction for 4 installs | ~1.3428 |
| Linear regression intercept | ~0.0473 |
| Linear regression coefficient | ~0.3239 |

**Table 4: Predicted Probabilities**

|  |  |  |
| --- | --- | --- |
|  | Retention | Non-Retention |
| Probability of keeping 1 user for 30 days with 1 install | 17.3% | 82.7% |
| Probability of keeping 1 user for 30 days with 2 installs | 42.9% | 57.1% |
| Probability of keeping 1 user for 30 days with 4 installs | 91% | 9% |

The machine learning algorithms used for this study were the decision tree and random forest methods. Both methods boasted a high accuracy score. The decision tree method was found to be ~89.4% accurate and the random forest method was ~87% accurate. The mean absolute error was found to be 0.13 and the receiver operator characteristic area under the curve (ROC AUC) was ~0.9232. This gives us confidence that the model can successfully distinguish between the two states of retention and non-retention.

**Conclusions**

The linear regression calculations showed that as installations increased, the predicted retention rate for 30 days also went up in a linear method. The relationship is a positive one in a linear fashion as shown in the pair plot of Figure 1. The installs by month show us that most installs happen during April and May. There is a noticeable drop off of installers during the fall and winter months, as well.

The logistic regression model shows us that two installs are the break-even area. There is a ~57% chance of retaining at least one user if we reach two installs. As the number of installers increases, so does our probability of retention.

**Recommendations**

Any organization that is developing mobile applications can use the information here to assist in tracking and planning their release schedule. The yearly install cycle can help plan when to spend more on advertising throughout the year to maintain a steady installation base. The number of installs needed to increase the probability of retention will also help determine when and where to focus energy on new features to increase adoption rates. These metrics can be continuously monitored and help guide the organization in a successful direction.