Kai Mcconnell

Project 3 Classification Analysis

Google App Store Data Set

For project 3 on classification analysis I chose a dataset which contained information about google play store apps. It contained data like the names of the apps and their ratings and downloads in one csv. In the other csv it had done sentiment analysis on a number of things such as reviews. I decided to focus on the csv containing more concrete data for ease of use and understandability. My first steps were to explore the data I had to work with and figure out what I wanted to target with my model. Towards this end I tried to gather information about the different features. My first step was to see what each row of the data looked like.

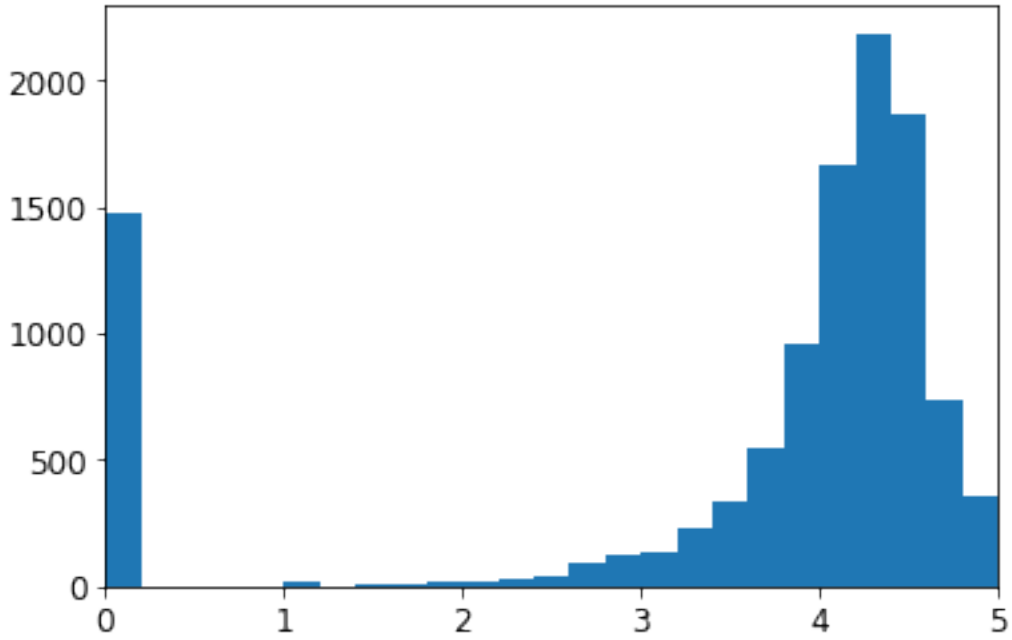
```
                                App        Category  Rating Reviews  Size  \
30  Pink Silver Bow Keyboard Theme  ART_AND_DESIGN     4.2    1120  9.2M
          Installs  Type Price Content Rating        Genres   Last Updated  \
30  100,000+  Free     0       Everyone  Art & Design  July 12, 2018
   Current Ver   Android Ver
30  6.7.12.2018  4.0.3 and up
```

This gave me a few ideas of what to explore and examine for being possible features I could use in my model. The first thing I did was look at how many apps of each category there were. These were my results:

[('FAMILY', 1972), ('GAME', 1144), ('TOOLS', 843), ('MEDICAL', 463), ('BUSINESS', 460), ('PRODUCTIVITY', 424), ('PERSONALIZATION', 392), ('COMMUNICATION', 387), ('SPORTS', 384), ('LIFESTYLE', 382), ('FINANCE', 366), ('HEALTH_AND_FITNESS', 341), ('PHOTOGRAPHY', 335), ('SOCIAL', 295), ('NEWS_AND_MAGAZINES', 283), ('SHOPPING', 260), ('TRAVEL_AND_LOCAL', 258), ('DATING', 234), ('BOOKS_AND_REFERENCE', 231), ('VIDEO_PLAYERS', 175), ('EDUCATION', 156), ('ENTERTAINMENT', 149), ('MAPS_AND_NAVIGATION', 137), ('FOOD_AND_DRINK', 127), ('HOUSE_AND_HOME', 88), ('AUTO_AND_VEHICLES', 85),

('LIBRARIES_AND_DEMO', 85), ('WEATHER', 82), ('ART_AND_DESIGN', 65), ('EVENTS', 64), ('COMICS', 60), ('PARENTING', 60), ('BEAUTY', 53)]

The second thing I looked at was the distribution of the review ratings. This feature interested me because it seemed like something which was important because of its likely connection to other features. These were my results:



As you can see there seems to be a trend towards reviews being between 4 and 4.5 which says that reviews don't have an even spread. I would guess this makes sense because people are less likely to review something they don't have a positive experience with. The reason there is a large jump at the 0 mark is because some of the data was unrated and so I filled in these values with 0's. Another feature I wanted to explore was the install numbers. These seemed to me as the best target for my models prediction because of their likely correlation to most other features and being a good metric to judge an app's success. It was difficult to explore this feature as it had a wide spread of numbers and was difficult to design how it should look. So in lieu of a graph I decided to do my best with some of the mathematical values of the feature. I found the min, max, and average of the values:

MIN VALUE:

0

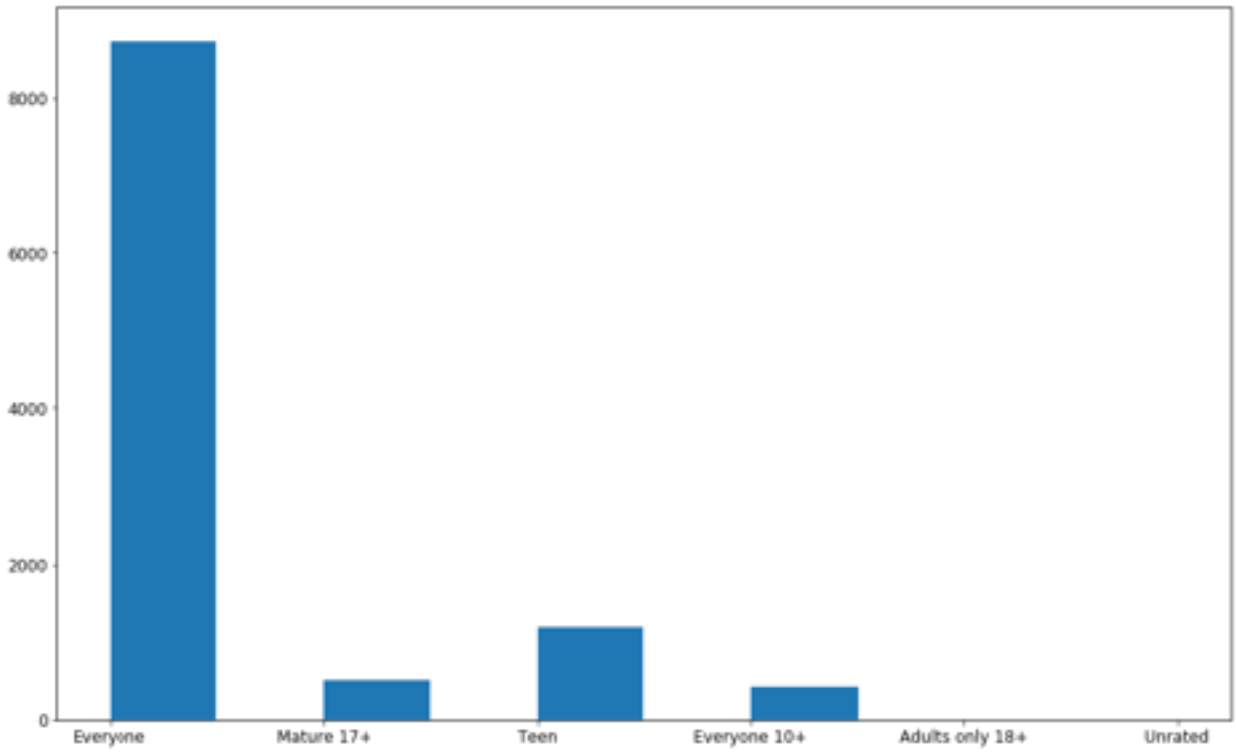
MAX VALUE:

1000000000

AVERAGE VALUE:

15464338.882564576

What this told me was the range I should expect my install feature to have as well as a possible cutoff for describing an app as successful or not. Another feature I explored was the content rating feature which acted as a suggestion of what age the consumers of these apps should be. These were my results:

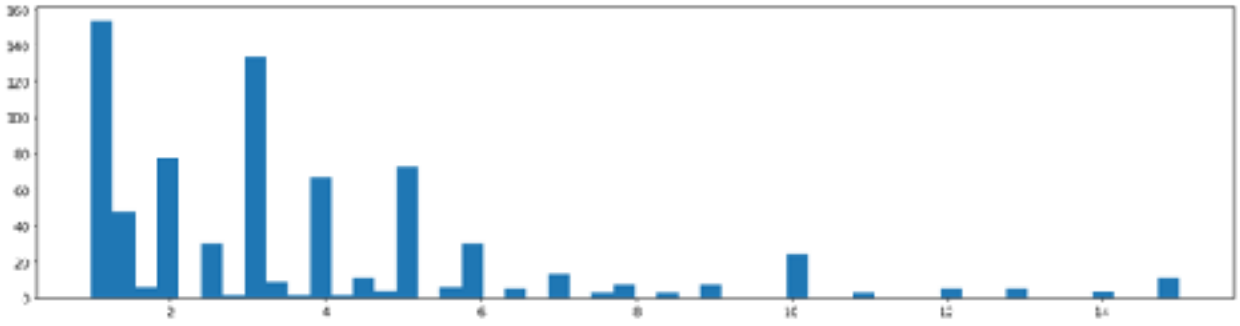


This showed me exactly what I suspected which was that most apps were aiming for the largest audience of everyone. In exploring this feature I also recognized that this would be feature that likely wouldn't be that useful in creating my model because it wasn't a numerical value and it had so few possible results. The next feature I explored was price.

These were my results:

AVERAGE COST:

$ 1.027368081180801



What this told me was that a large chunk of the apps were cheaper than 5 dollars. This lines up with my own experiences with the app store. Most apps are either free or very cheap. They often get their money through micro transactions not direct cost. Once I had explored these features I came to a decision on what features to use for creating my model. I chose to focus on the number of reviews, the apps rating based on those reviews and the apps price. These 3 features would allow me to predict the apps success represented by the number of installs the app had received. I chose these features because of their numerical nature and because from my analysis they appeared to be the features most likely to help predict success. I did a random forest classifier on each feature and then all 3 features together. These were my results:

#Reviews to installs:

Acurracy: 0.4986162361623616

Cost to installs:

Acurracy: 0.16605166051660517

Apps rating to installs:

Acurracy: 0.19280442804428044

#Reviews, cost, and rating to installs:

Acurracy: 0.5272140221402214

What these results told me was quite interesting. The only feature to have a significant accuracy was the # of reviews feature. In some ways this makes sense as the only way for an app to have a high number of reviews is if it has a high number of installs. It also made me realize that I should maybe simplify my prediction target to a boolean because it is hard to have the model predict the

exact number of installs an app would have. In making this change I found my accuracy increase by leaps and bounds as I expected.

Rating to install Boolean:

Acurracy: 0.9169741697416974

Price to install Boolean:

Acurracy: 0.9169741697416974

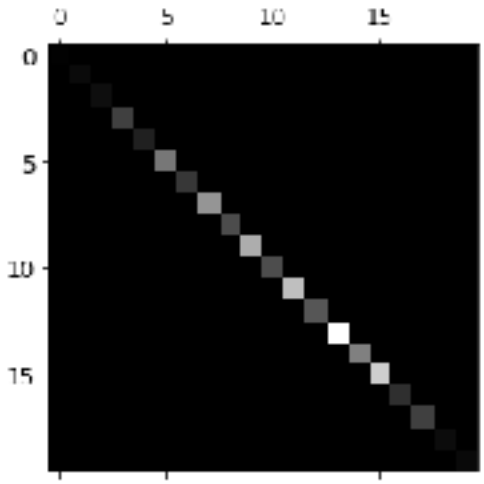
#Reviews to Install Boolean:

Acurracy: 0.9829335793357934

#Reviews, Cost, and Rating to install boolean:

Acurracy: 0.9838560885608856

After receiving such successful results I worked towards building a confusion matrix which resulted in this image representing the confusion matrix:



Overall this project showed me the value of exploring my features and evaluating what I was trying to predict. My most successful results came as a result of reevaluating what exactly I wanted to predict. While this can take away from a models detail when predicting It can also lead to a model which can accurately provide feedback on classifying your data.