Mobile Apps Rating and Rating Counts Analysis Project Report

By Zhengyu Li (lizhengy@umich.edu)

1. Motivation and Summary

The ever-changing mobile landscape is a challenging space to navigate. The percentage of mobile over the desktop is only increasing. Android holds about 53.2% of the smartphone market, while iOS is 43%.[1] To get more people to download your app, you need to make sure they can easily find your app. Mobile app analytics is a great way to understand the existing strategy to drive growth and retention of future users. With millions of apps around nowadays, it is an interesting topic to analyze the feedback of users to make apps more practical.

This project will explore which factors have an impact on app ratings (reputation) as well as rating counts (popularity) by users. The general question is **how does the App details contribute to the user ratings and user rating counts?**

According to the general question, this analysis will focus on three major research questions:

1) What is the distribution of average app rating counts among app genres in different content (age) ratings?

Content rating has 4 groups: "4+", "9+", "12+" and "17+", I can use a for loop to plot the relationship between average app ranting counts among app genres in each group.

2) Is there any difference among ratings on different price levels?

Use factor to cut the price of apps into three levels ("free", "Medium", "high"), then utilize proportional stacked area chart and ANOVA test to verify the conclusion.

3) What is an optimal regression model (OLS, Lasso, and Ridge) to predict user ratings?

Create OLS, Lasso and Ridge regressions models, calculate MSE (mean squared error) for training and test data respectively and utilize bar chart to visualize MSE results and determine optimal models

2. Data Sources

The following dataset is concentrated on getting top trending apps in iOS app store. This data set contains more than 7000 Apple iOS mobile application details. The data was extracted from the iTunes Search API at the Apple Inc website in July 2017. Here is the dataset link from Kaggle:
https://www.kaggle.com/ramamet4/app-store-apple-data-set-10k-apps#AppleStore.csv

It is a csv file called "AppleStore.csv" and includes 7197 records (apps) in total.

Here are several fields that I am interested in:

name	type in r	explanation
size_bytes	int	Size (in Bytes)
price	num	Price amount
rating_count_tot	int	User Rating counts (for all version)
user_rating	num	Average User Rating value (for current version)
cont_rating	factor	Content (Age) Rating
prime_genre	factor	Primary Genre
sup_devices.num	int	Number of supporting devices
lang.num	int	Number of supported languages

For content rating (age rating), Apple's rating system for the App Store follows the following rubric:[2]

Rated 4+: Contains no objectionable material.

Rated 9+: May contain content unsuitable for children under the age of 9.

Rated 12+: May contain content unsuitable for children under the age of 12.

Rated 17+: May contain content unsuitable for children under the age of 17.

3. Methods

3.1 Question 1: What is the distribution of average app rating counts among app genres in different age ratings?

3.1.1 Question 1: Manipulation

For this question, it focuses on the user rating counts, content rating as well as primary genres. First, I extracted the subset including these three columns to conduct the following analysis. Then I used aggregate() function to calculate the mean user rating counts grouping by different genres and different content ratings, which is a required step to plot distribution.

3.1.2 Question 1: Missing/Incomplete/Noise

I utilized sum(is.na()) function to check if there exist any NA values in the subset dataset and concluded that there is no missing data in the dataset. For this question, I did not take incomplete and noise data into account.

3.1.3 Question 1: Challenge

This part was not terribly challenging since it mostly involved exploratory analysis. However, if there was a challenge, it would be making the final bar chart more sense. Initially, the mean user rating counts have great differences among different genres and highly skewed in several popular genres, such as games and music. In order to solve this, I adopted the log transformation for mean user rating counts. It is helpful to make highly skewed distributions less skewed. This can be valuable for making patterns in the data more interpretable.

3.2 Question 2: Is there any difference among ratings on different price levels?

3.2.1 Question 2: Manipulation

For this question, it concentrates on the user rating and price. First, I extracted the subset including these two columns to conduct the following analysis. Then I used geom_histogram() function to visualize the price and its density to determine critical value for each price level. As a result, I executed cut() function to cut price into three levels, here is the range for each group:

Price Level	Price Range ($)
Free	0
Medium	(0, 10]
High	(10,300]

3.2.2 Question 2: Missing/Incomplete/Noise

I utilized sum(is.na()) function to check if there exist any NA values in the subset dataset and concluded that there is no missing data in the dataset. For this question, I did not take incomplete and noise data into account.

3.2.3 Question 2: Challenge

The challenge for this question is the choice of chart type to visualization ratings on different price levels. First, I tried boxplot but the average user rating for each app is also categorical data. Rating levels contain 0, 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5 and 5. For boxplot, all price levels have the same median 4 and difficult to discover the difference on price levels. After searched on Google, I found an awesome chart to show, which is called the proportional stacked area chart. In a proportional stacked area graph, the sum of each price level is always equal to hundred and rating of each price level is represented through percentages.

3.3 Question 3: What is an optimal regression model (OLS, Lasso, and Ridge) to predict user ratings?

3.3.1 Question 3: Manipulation

For this question, it concentrates on the average user rating, price, user rating counts, content rating, supported languages, size (in Bytes) and supporting devices for every app. First, I extracted the subset including these columns to conduct the following analysis. Then I used sample() function to select 80% rows of dataset as a training dataset and the rest is a test dataset to test the fitness of each model.

3.3.2 Question 3: Missing/Incomplete/Noise

I utilized sum(is.na()) function to check if there exist any NA values in the subset dataset and concluded that there is no missing data in the dataset. As for noise data, I noticed that almost 56% of apps' price is 0, which might influence the accuracy of the regression model, so I converted the price column from continuous data to categorical data containing two levels: free and non-free.

3.3.3 Question 3: Challenge

The difficult part of this question was executing lasso and ridge regression in R. I only learned basic theoretical knowledge for these two dimension reduction regressions before. After searched related information on google and watched relevant videos on Youtube, I plotted ridge and lasso regression, found the best lambda through cross-validation and computed training and testing error step by step.

4. Analysis and Results

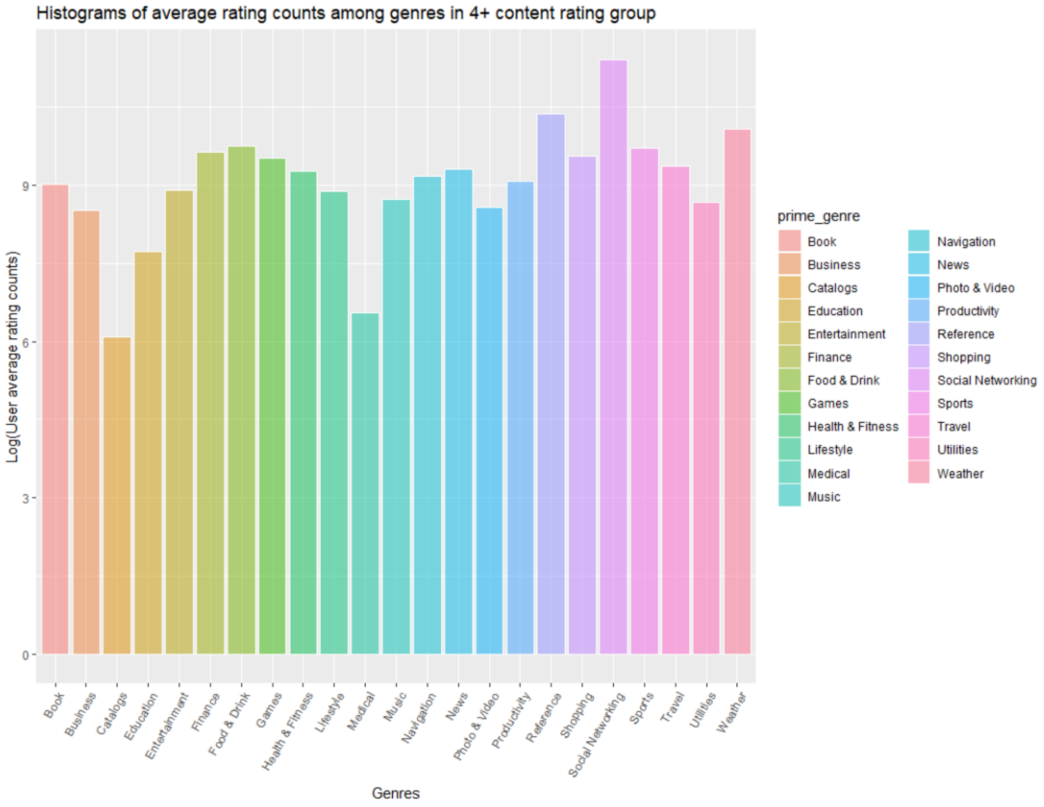
4.1 Question 1: What is the distribution of average app rating counts among app genres in different age ratings?

4.1.1 Question 1: Code Flow

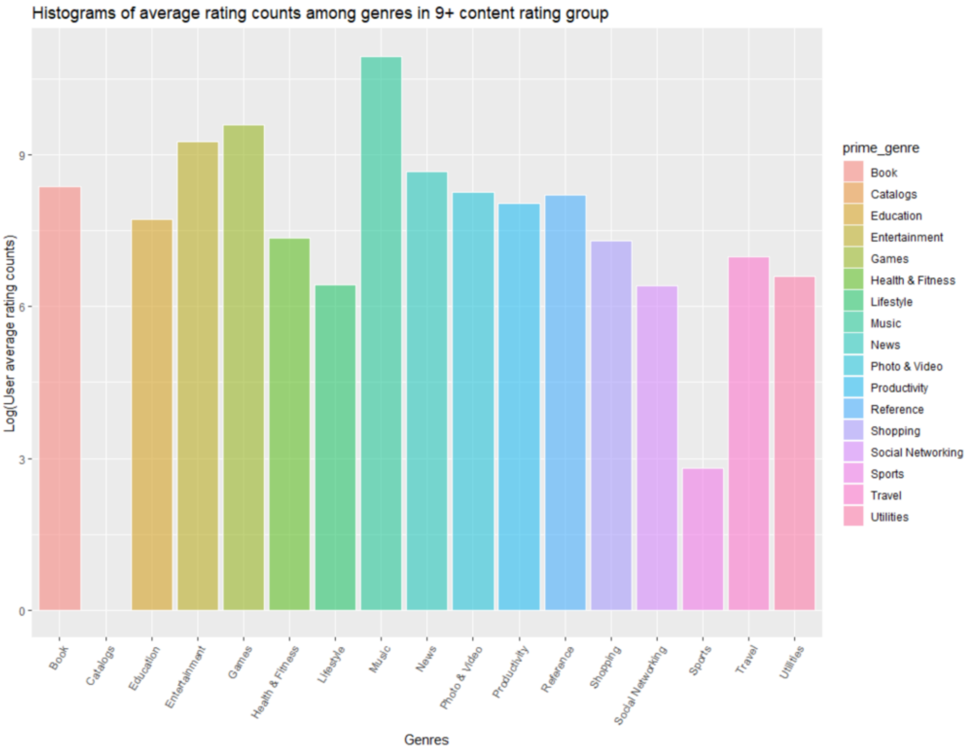
First, I extracted the subset including user rating counts, content rating as well as primary genres columns. Next I used aggregate() function to calculate the mean user rating counts grouping by different genres and different content ratings. Then I utilized levels() function to get a vector of content rating. Finally, I conducted a for loop to plot the relationship between average app ranting counts among app genres in each group.

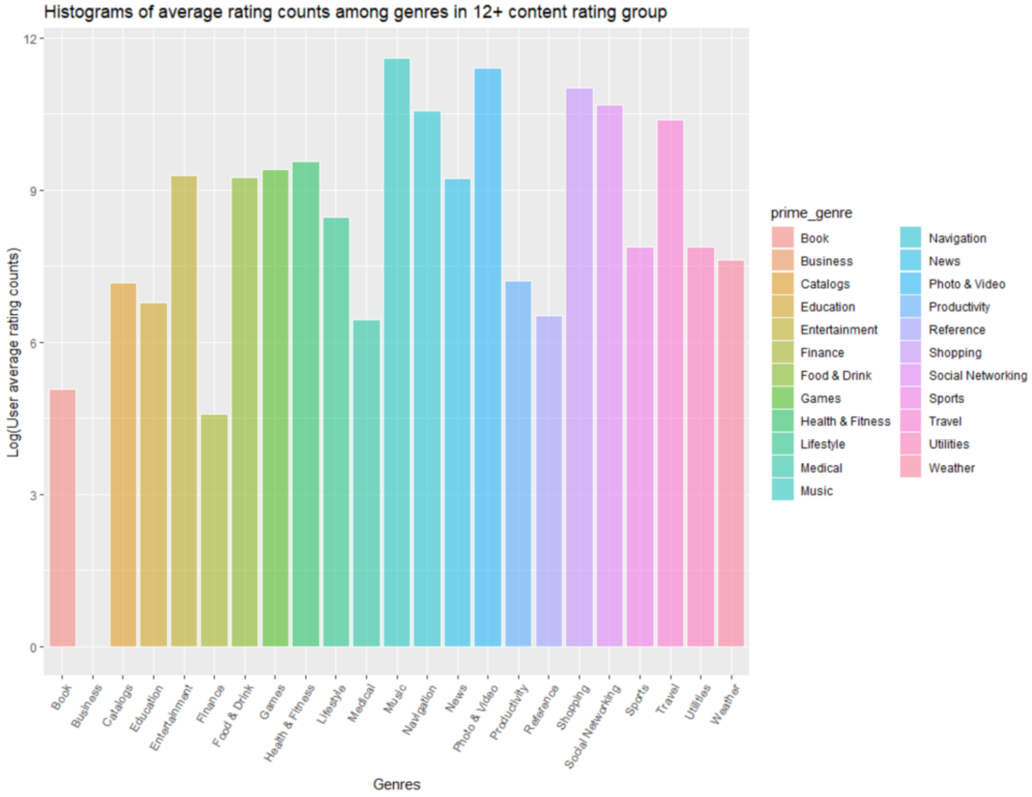
4.1.2 Question 1: Results and Visualization

I adopted the log transformation for mean user rating counts. It is helpful to make highly skewed distributions less skewed and valuable for making patterns in the data more interpretable.

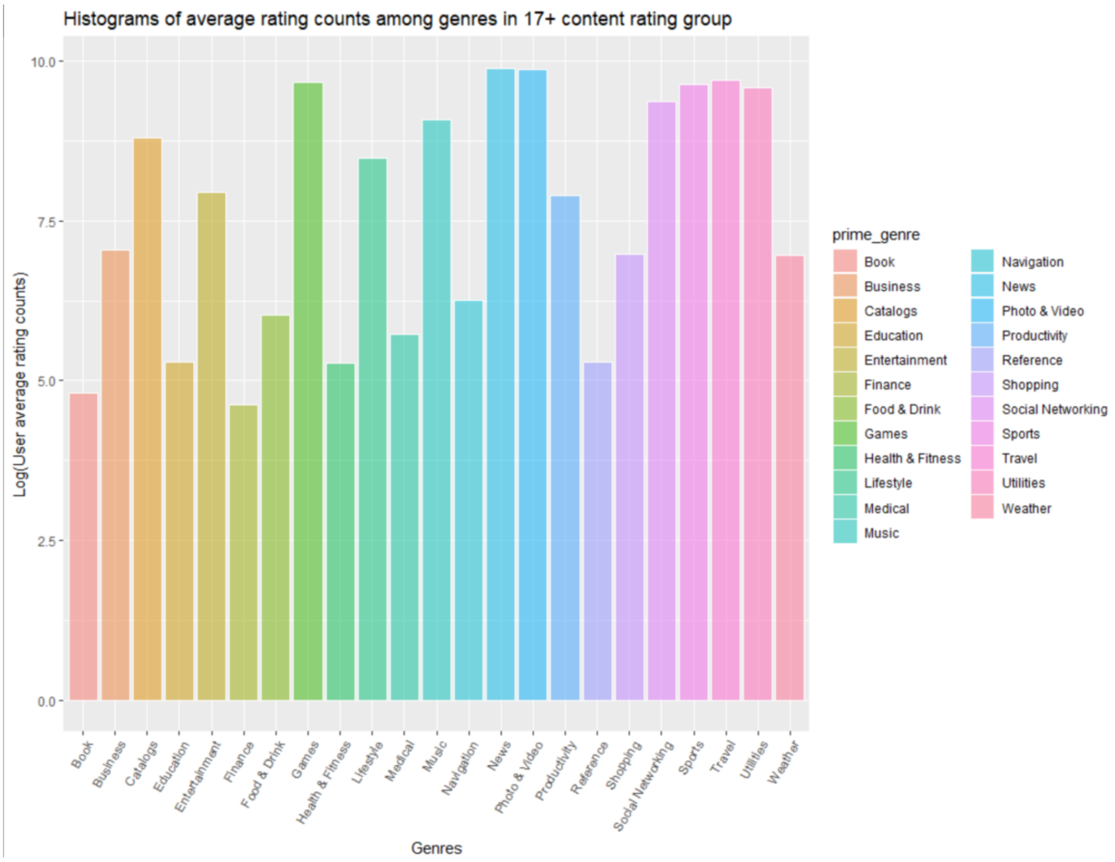


For 4+ content (age) rating group, social network apps have the highest average rating count while catalogs apps are lowest. It fits with common sense because several age ratings of popular social network apps are 4+, such as GroupMe and Life360. I even cannot find apps whose class is catalogs in the app store.





For 9+ content (age) rating group and 12+ content (age) group, music apps become the most popular apps. Most of the top music apps are 9+ and 12+ including Spotify. I also find an interesting result that sports apps are not prevalent in these age groups. Then I searched in the app store and find the age contents of top sports apps, such as FOX and ESPN, are all 4+.



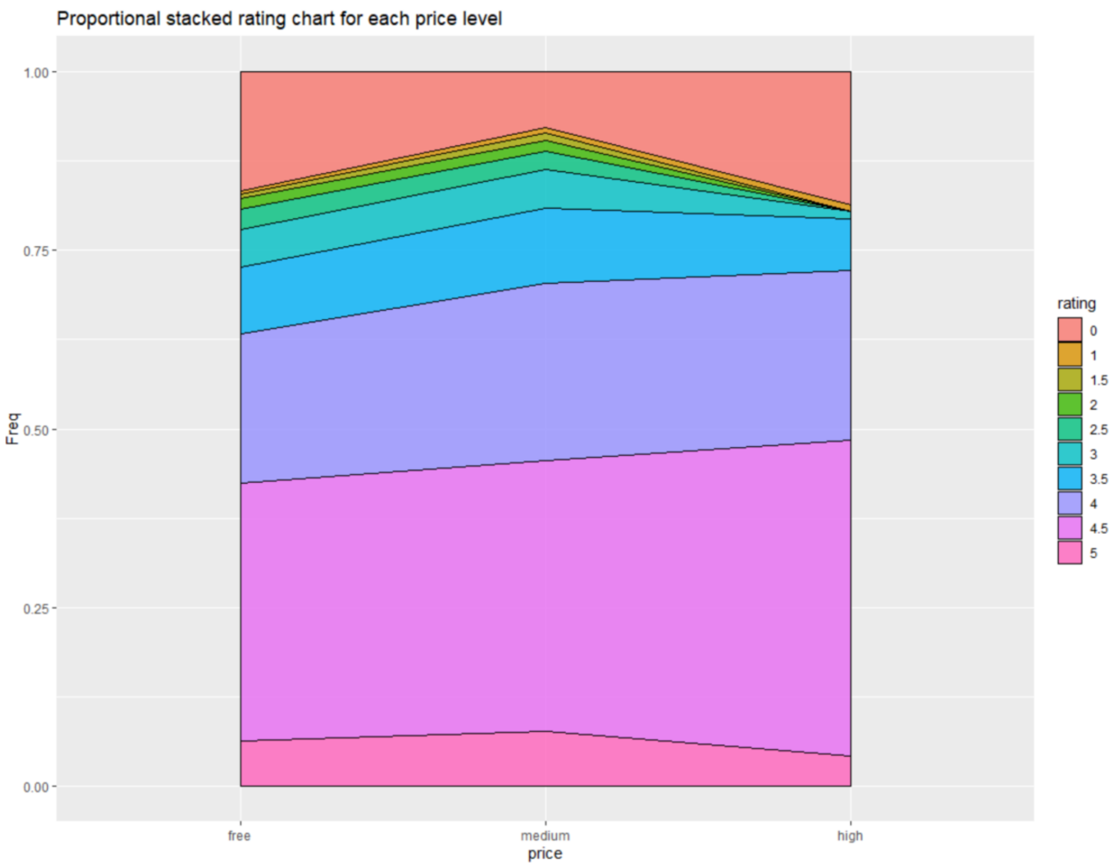
For 17+ content (age) rating group, some recreational apps containing travel, games and travel are more popular compared to other apps.
In conclude the genres and age ratings have a great impact on apps rating counts.

4.2 Question 2: Is there any difference among ratings on different price levels?

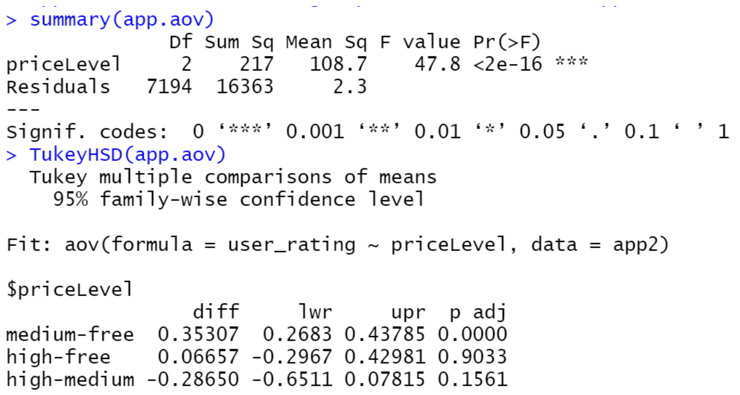
4.2.1 Question 2: Code Flow

First, I extracted the subset including user rating and price columns to conduct the following analysis. Then I visualized the price and its density through histogram to determine critical value for each price level. As a result, I executed cut() function to cut price into three levels: free, medium and high. Next, I adopted prop.table() function to compute percentages for each price level and user rating group and geom_area() function to finishing proportional stacked area graph. Finally, other than visualization, I did ANOVA test as well as TukeyHSD test to test the different mathematically.

4.2.2 Question 2: Results and Visualization



From the proportional stacked area graph, it is clear to observe the proportion among different rating levels on each price level. High price level seems to contain less rating 5 and more rating 4.5 and 0 compared to other groups. Medium price level is more likely to have rating 5 as well as rating between 1 and 3.5.



```
> summary(app.aov)
              Df Sum Sq Mean Sq F value Pr(>F)    
priceLevel     2    217   108.7    47.8 <2e-16 ***
Residuals   7194  16363     2.3                   
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> TukeyHSD(app.aov)
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = user_rating ~ priceLevel, data = app2)

$priceLevel
                diff       lwr     upr  p adj
medium-free  0.35307  0.2683 0.43785 0.0000
high-free    0.06657 -0.2967 0.42981 0.9033
high-medium -0.28650 -0.6511 0.07815 0.1561
```

Based on ANOVA test, p-value is less than 0.05, so we can reject the null hypothesis and it means that at least one of the group means is different from the other groups. According to the result of TukeyHSD test, it shows the price medium level is significantly different from free because their p-value is less than 0.05. And medium price apps tend to have a higher user rating compared to free.

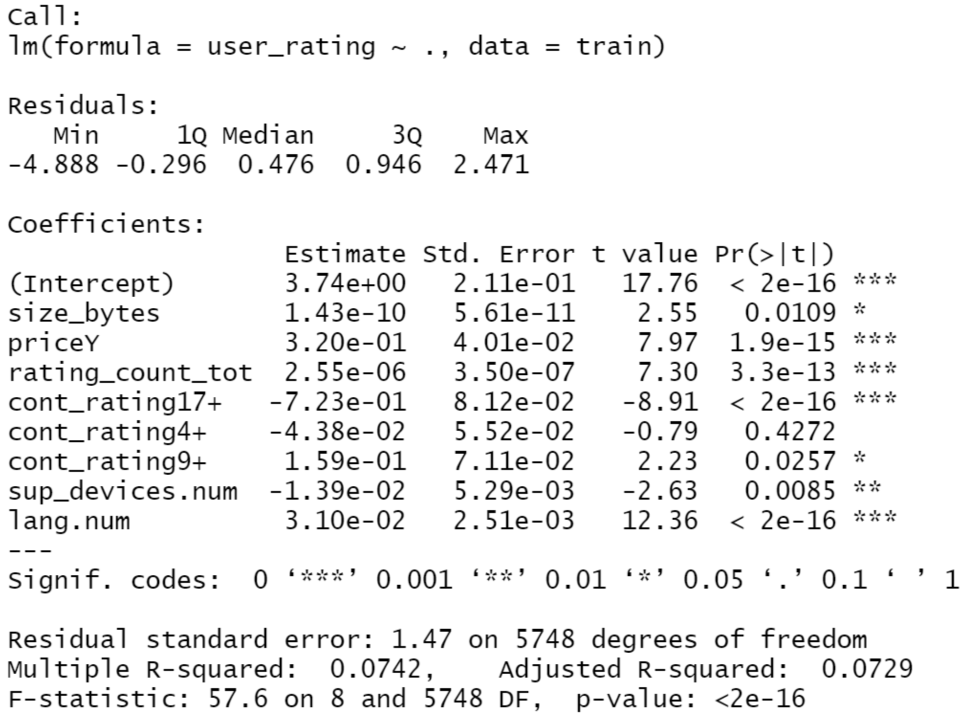
4.3 Question 3: What is an optimal regression model (OLS, Lasso, and Ridge) to predict user ratings?

4.3.1 Question 3: Code Flow

First, I extracted the subset including average user rating, price, user rating counts, content rating, supported languages, size (in Bytes) and supporting devices for every app to conduct the following analysis. Next, I converted the price column from continuous data to categorical data containing two levels: free and non-free to make predictors more reasonable. Then I used sample() function to select 80% rows of dataset as a training dataset and the rest is a test dataset to test the fitness of each model. For the next section, I created OLS, Lasso and Ridge regressions models, calculated MSE (mean squared error) for training and test data respectively and utilized bar chart to visualize MSE results and determine optimal models.

4.3.2 Question 3: Results and Visualization

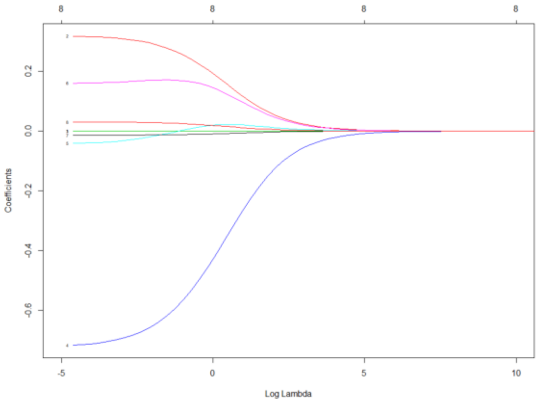
1) OLS summary

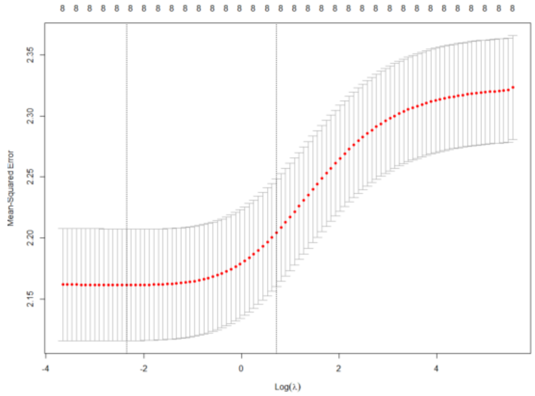


```
Call:
lm(formula = user_rating ~ ., data = train)

Residuals:
   Min     1Q Median     3Q    Max 
-4.888 -0.296  0.476  0.946  2.471 

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)    
(Intercept)       3.74e+00   2.11e-01   17.76  < 2e-16 ***
size_bytes        1.43e-10   5.61e-11    2.55   0.0109 *  
priceY            3.20e-01   4.01e-02    7.97  1.9e-15 ***
rating_count_tot  2.55e-06   3.50e-07    7.30  3.3e-13 ***
cont_rating17+   -7.23e-01   8.12e-02   -8.91  < 2e-16 ***
cont_rating4+    -4.38e-02   5.52e-02   -0.79   0.4272    
cont_rating9+     1.59e-01   7.11e-02    2.23   0.0257 *  
sup_devices.num  -1.39e-02   5.29e-03   -2.63   0.0085 ** 
lang.num          3.10e-02   2.51e-03   12.36  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.47 on 5748 degrees of freedom
Multiple R-squared:  0.0742,	Adjusted R-squared:  0.0729 
F-statistic: 57.6 on 8 and 5748 DF,  p-value: <2e-16
```

OLS is a common regression method. As we can see in the summary, even though R^2 is low, the independent variables are statistically significant. I can still draw important conclusions about the relationships between the variables. Other than cont-rating is equal to 4+, other predictors are statistically significant with the response.

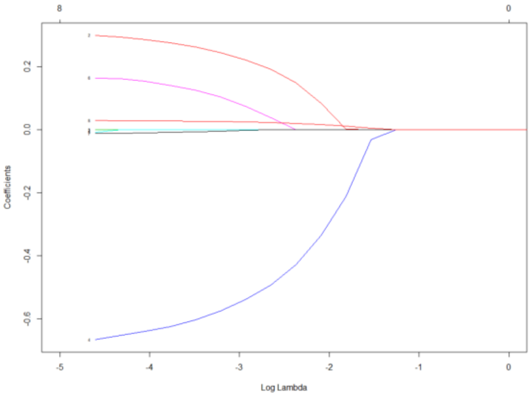
2) Ridge Summary

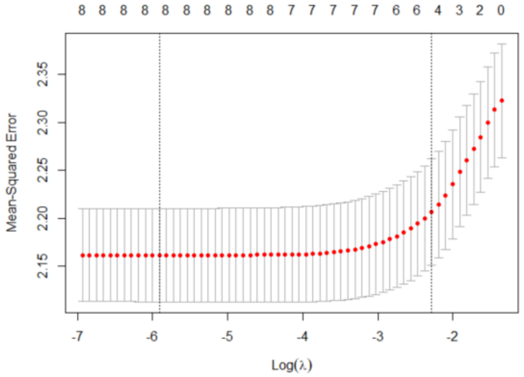




The left plot is the coefficient for each predictor among different lambda values. They tend to shrink to 0 (but not equal to 0) with the increase of lambda. The right plot is the relationship between lambdas and mean square error to determine lambda. Optimal lambda should have the least mean square error and be equal to 0.095.

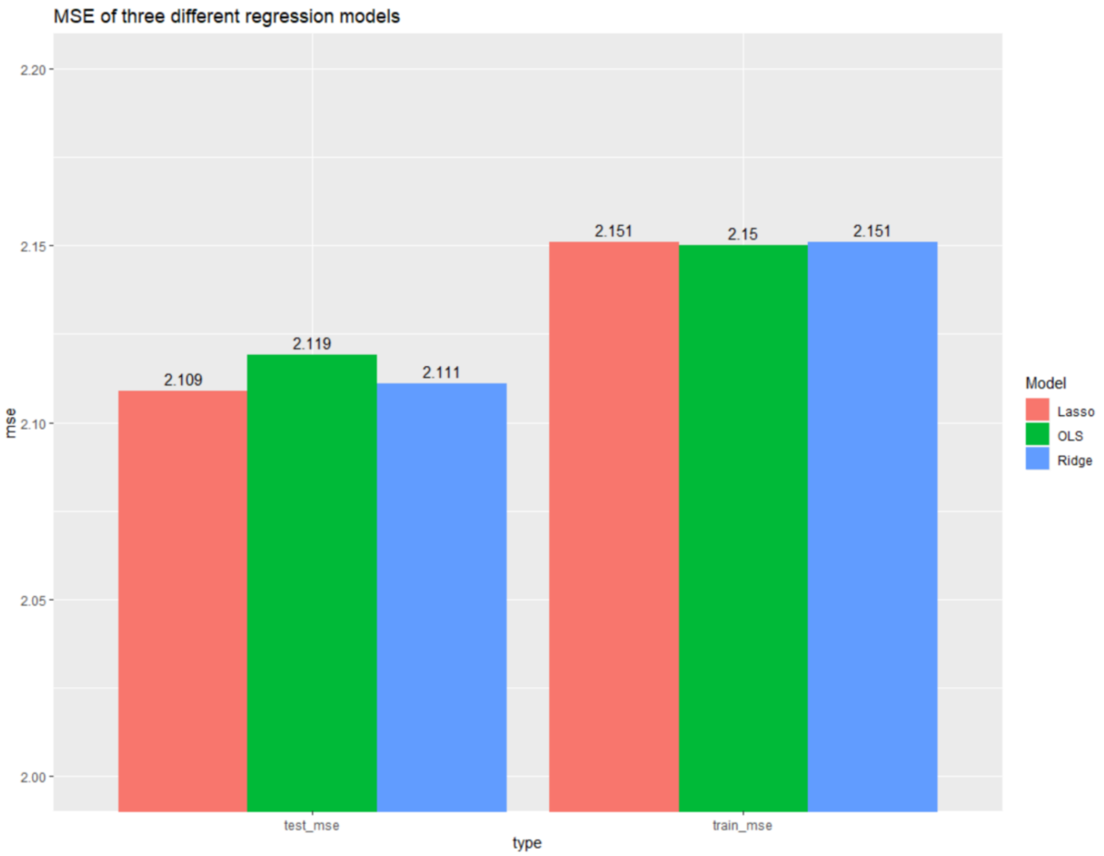
3) Lasso Summary





The left plot is the coefficient for each predictor among different lambda values. They tend to shrink to 0 (equal to 0 finally) with the increase of lambda. The right plot is the relationship between lambdas and mean square error to determine lambda. Optimal lambda should have the least mean square error and be equal to 0.003.

4) MSE comparison



As we can see from the bar chart, test MSEs are all less than train MSEs, which indicates that models have great fitness for test data compared to training data. For train data, three models have similar mean squared errors. People are inclined to select the model with lower MSE for test data, so in this situation, lasso regression is an optimal choice to predict user rating. The conclusion also corresponds with theoretical knowledge. Since Lasso Regression can exclude useless variables from models, it is a little bit better than ridge regression at reducing the variance in models that contain useless variables.

5. Reference

[1] Mobile App Statistics (Apple iOS app store), https://www.kaggle.com/ramamet4/app-store-apple-data-set-10k-apps#AppleStore.csv

[2] "Identifying Your App in iTunes Connect: Set App Ratings". Apple Inc. Retrieved 19 May 2017.