<u>Questions 1 to 3</u>

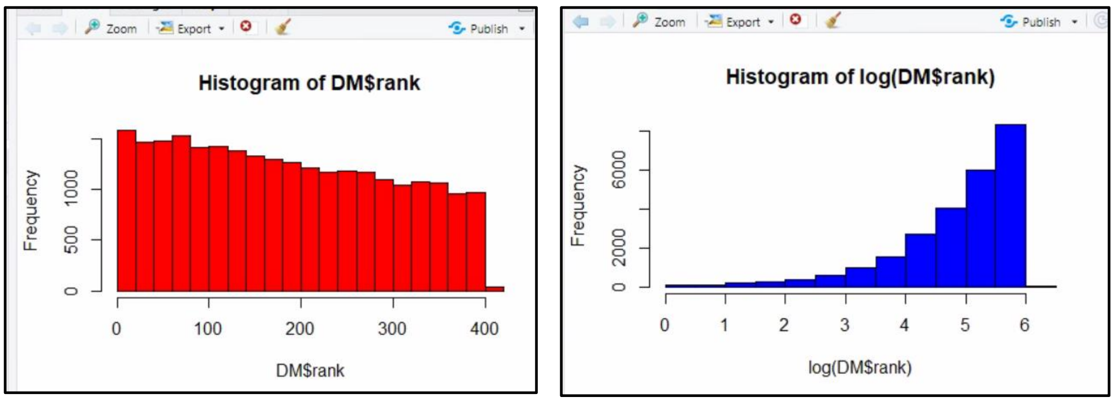
Note: The regression models for this testing have been done using R.

Below is a snapshot of the linear regression model used in R.

```
Linear_reg = lm(Log_rank~ Log_price+Log_rating_count+Log_filesize+Log_app_age_current_version+DM$deviceindex+
             DM$appstoreindex+DM$rindex+DM$apptypeindex+DM$num_screenshot+
             DM$average_rating+DM$inapp_addummy+DM$inapp_purchasedummy+
             DM$categoryindex+DM$developer)
```

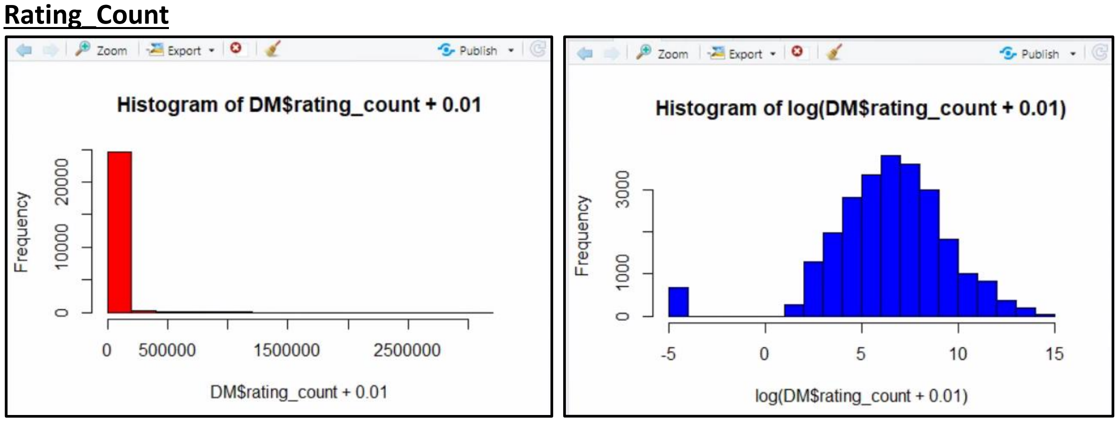
Rank has been used as the target variable. Log transformations have been applied on the following variables with explanations in chart.

1. **<u>Rank</u>**



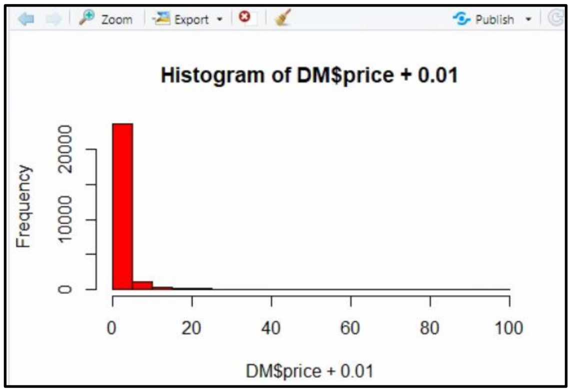
As can be seen from the charts above, log transformation helps in spreading out the data better for interpreting.

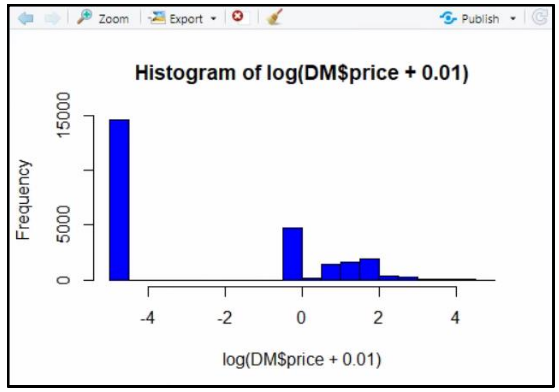
2. **<u>Rating_Count</u>**



Log transformation helps in improving the skewness of the data and interpretability for later modeling. There are rating_count with 0 values in the data provided which have been handled be adding 0.01 to each data cell. As a result there is one bar of data in the log transformed data with a value of -5 on the x axis

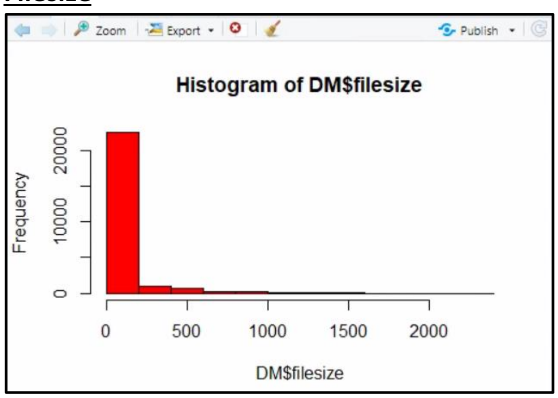
3. **<u>Price</u>**

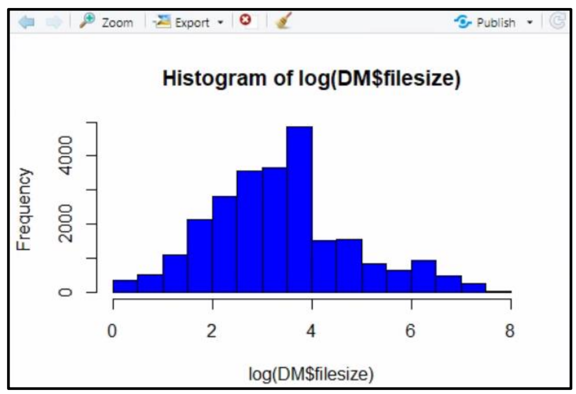




Log transformation helps in the interpretability of price and it's skewness. There are apps where the price is 0 which has been handled by adding each row of price with 0.01 which explains the extreme left bar on the x axis of the log transformed data.

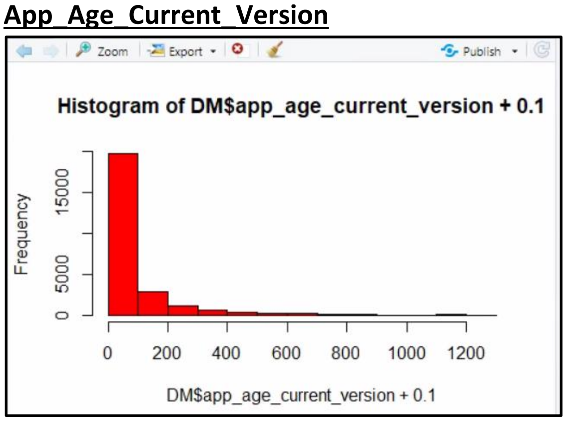
4. **<u>Filesize</u>**

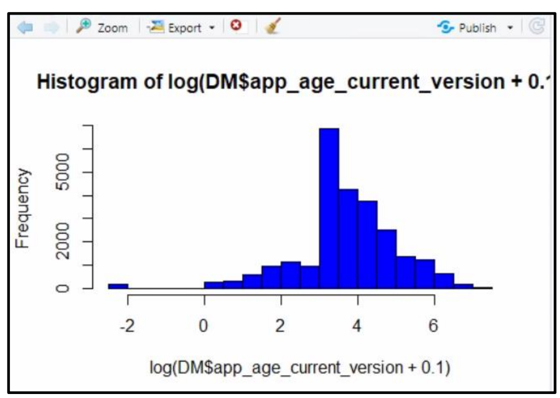




Log transformation helps in the interpretability of the data and helps with the skewness.

5. **<u>App_Age_Current_Version</u>**





Log transformation helps with the skewness of the data as well as it's interpretability. The values have been added with 0.1 to help with the log transformations of 0 values in this column.

The following variables have indexes placed which have been used in place of the original column. They are as follows:

Index variable	Original Variable	Dataframe reference
deviceindex	device	DM$device
appstoreindex	app_store	DM$app_store
rindex	region	DM$region
inapp_addummy	in_app_ads	DM$in_app_ads
inapp_purchasedummy	in_app_purchase	DM$in_app_purchase
categoryindex	category	DM$category
apptypeindex	app_type	DM$app_type

The variables num_screenshot and average_rating have been used as it is. Thus the linear regression model has 14 predictor variables and 1 predicted variable(Log of Rank).

Output

The output is in the format of a csv file which is attached. Let's look at the results.

	term	estimate	std_err	statistic	p_value	lower_c	upper_c
7	DM$appstoreindex: 2	2.68	0.842	3.181	0.001	1.029	4.331
8	DM$appstoreindex: 3	2.361	0.449	5.254	0	1.48	3.242
10	DM$apptypeindex: 2	-0.18	0.019	-9.466	0	-0.217	-0.143
11	DM$apptypeindex: 3	-1.155	0.025	-46.695	0	-1.203	-1.106
13	DM$average_rating	0.354	0.015	24.319	0	0.325	0.382
6	DM$deviceindex: 2	-0.104	0.012	-8.794	0	-0.127	-0.081
14	DM$inapp_addummy: 3	0.166	0.027	6.026	0	0.112	0.219
15	DM$inapp_purchasedummy: 3	-0.174	0.031	-5.683	0	-0.233	-0.114
12	DM$num_screenshot	0.049	0.007	6.656	0	0.034	0.063
9	DM$rindex: 2	0.106	0.021	5.15	0	0.066	0.146
1	intercept	2.175	0.701	3.103	0.002	0.801	3.548
5	Log_app_age_current_version	0.242	0.008	29.861	0	0.226	0.258
4	Log_filesize	-0.056	0.014	-4.019	0	-0.083	-0.029
2	Log_price	0.032	0.004	7.843	0	0.024	0.04
3	Log_rating_count	-0.241	0.006	-42.564	0	-0.252	-0.23

The above is a screen shot of all the variables except category and developer variables. The below table contains the interpretation of each of these variables. Note that given the p-value, these variables seem to be strong predictors of rank.

term	Interpretation	p_value
DM$appstoreindex: 2	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by e^2.68%	0.001
DM$appstoreindex: 3	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by e^2.361%	0
DM$apptypeindex: 2	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by e^-0.18%	0
DM$apptypeindex: 3	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by e^-1.155%	0
DM$average_rating	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by e^0.354%	0
DM$deviceindex: 2	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by e^-0.104%	0

DM$inapp_addummy: 3	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by e^0.166%	0
DM$inapp_purchasedummy: 3	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by e^-0.174%	0
DM$num_screenshot	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by e^0.049%	0
DM$rindex: 2	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by e^0.106%	0
intercept	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by e^2.175%	0.002
Log_app_age_current_version	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by 0.242%	0
Log_filesize	Increasing this term by 1 % while keeping all other variables constant decreases the value of rank by -0.056%	0
Log_price	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by 0.032%	0
Log_rating_count	Increasing this term by 1 % while keeping all other variables constant decreases the value of rank by -0.241%	0

Now let's look at the variable category.

term	Interpretation	p_value
DM$categoryindex: 10	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by **e^-0.133%**	0.093
DM$categoryindex: 13	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by **e^0.277%**	0
DM$categoryindex: 14	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by **e^0.302%**	0.55
DM$categoryindex: 17	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by **e^0.341%**	0.001
DM$categoryindex: 20	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by **e^-0.03%**	0.782
DM$categoryindex: 27	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by **e^0.41%**	0.607
DM$categoryindex: 31	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by **e^1.394%**	0.15
DM$categoryindex: 42	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by **e^0.041%**	0.66
DM$categoryindex: 43	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by **e^-1.788%**	0.035
DM$categoryindex: 47	Increasing this term by 1 % while keeping all other variables constant increases the value of rank by **e^0.124%**	0.16

Of these categories, however, only "categoryindex:13" and "categoryindex: 17" are significant as they have a small p value. Thus, the insight here is that the apps is that apps with categories other than categories 13 and 17 do not affect the rank or demand. However, if the category belongs to 13 or 17, there is a good possibility of having a better rank by the value mentioned in the above table.

Let's look at some of the top predictors from the developer's variable. We look at some of them as the number of developer types is a lot. However, this can be referred from the output csv file from R.

	term	estimate	std_err	statistic	p_value	lower_c	upper_c
689	DM$developer: Feng WuÂ© 2012 ä¸Šæµ·æ˜“ç‚¹æ—¶ç©ºç½‘ç›	-4.864	0.552	-8.812	0	-5.945	-3.782
1190	DM$developer: LOCOJOYÂ© LOCOJOY	-4.104	0.588	-6.978	0	-5.256	-2.951
1992	DM$developer: Xiangfei TuÂ©2012 Xiangfei Tu.All Rights Reser	-3.948	0.83	-4.758	0	-5.574	-2.321
1923	DM$developer: Wang XiaodanÂ© 4Game Studios	-3.762	0.514	-7.322	0	-4.769	-2.755
205	DM$developer: Beijing Baidu Netcom Science & Technology Co	-3.737	0.549	-6.808	0	-4.812	-2.661
1811	DM$developer: The Weather ChannelÂ© 2010-2012, The Wea	-3.647	0.624	-5.848	0	-4.869	-2.424
222	DM$developer: Beijing CloudNet Internet Co, LtdÂ© ï¼ 2011 m	-3.564	0.52	-6.855	0	-4.584	-2.545
900	DM$developer: he tiantianÂ© ç”°ç”°	-3.498	0.826	-4.236	0	-5.117	-1.879
1993	DM$developer: XiangJin LiuÂ© é˜¿åœŸå¦ˆå¦ˆå·¥ä½œå®¤	-3.498	0.517	-6.769	0	-4.511	-2.485
1403	DM$developer: Pang YufengÂ© Adult Fish LLC Copyright 2012	-3.427	0.541	-6.33	0	-4.488	-2.366

This is a list of the top 10 predictors to demand from the developer variable based on descending order of p-value. This shows that these developers usually have a negative affect on rank. Thus, an investor may choose against investing with these developers as it looks like they are not very well received in the market.

207	DM$developer: BeiJing BaoFengWangJi Technology Co., LtdÂ©	0.004	0.517	0.007	0.994	-1.009	1.017
1188	DM$developer: Liwei ZhengÂ© AppTao Inc,	0.004	0.521	0.008	0.994	-1.017	1.025
1297	DM$developer: MyDreamFactory	0.003	0.589	0.005	0.996	-1.152	1.158
1444	DM$developer: Pocket Gems, Inc.Â© Pocket Gems	0.004	0.826	0.005	0.996	-1.615	1.623
973	DM$developer: iLegendSoft IncÂ© iLegendSoft,Inc.	-0.003	0.68	-0.004	0.997	-1.336	1.33
653	DM$developer: ESPN Inc	-0.002	0.597	-0.004	0.997	-1.173	1.168
1608	DM$developer: Shanghai Gewara Business Info Consulting Co.,	0.002	0.679	0.003	0.998	-1.33	1.333
283	DM$developer: Beijing Zhangzhong MIG Information Technolo	-0.001	0.552	-0.001	0.999	-1.082	1.08
1944	DM$developer: WEIDONG LIÂ© 2012 Sanfarx	-0.001	0.548	-0.002	0.999	-1.075	1.074
1388	DM$developer: Oriented Games LimitedÂ© Oriented Games	0	0.825	0	1	-1.618	1.618

The above picture is a list of the bottom 10 developers in terms of p-value. The p-value here indicates that these developers do not really have an impact to the market. The apps made by these developers are not affected by the brand of developer.

Question 4

a) US vs China

```
> t.test(DM[rindex==1,]$rank, DM[rindex==2,]$rank, alternative = 'greater')

        Welch Two Sample t-test

data:  DM[rindex == 1, ]$rank and DM[rindex == 2, ]$rank
t = 12.909, df = 22966, p-value < 2.2e-16
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 16.41027      Inf
sample estimates:
mean of x mean of y 
 191.8767  173.0700 
```

rindex1: China has higher ranking in average than 2: USA as we reject the null hypothesis

```
> t.test(DM[rindex==1,]$price, DM[rindex==2,]$price, alternative = 'less')

        Welch Two Sample t-test

data:  DM[rindex == 1, ]$price and DM[rindex == 2, ]$price
t = -10.918, df = 17540, p-value < 2.2e-16
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
       -Inf -0.4564081
sample estimates:
mean of x mean of y 
 1.114881  1.652246 
```

rindex1: China has lower average prices than 2: USA as we reject the null hypothesis

```
> t.test(DM[rindex==1,]$average_rating, DM[rindex==2,]$average_rating, alternative = 'less')

        Welch Two Sample t-test

data:  DM[rindex == 1, ]$average_rating and DM[rindex == 2, ]$average_rating
t = -7.334, df = 23485, p-value = 1.153e-13
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
        -Inf -0.05947193
sample estimates:
mean of x mean of y 
 4.137830  4.214498 
```

rindex1: China has smaller average ratings than 2: USA as we reject the null hypothesis

```
> t.test(DM[rindex==1,]$rating_count, DM[rindex==2,]$rating_count, alternative = 'less')

        Welch Two Sample t-test

data:  DM[rindex == 1, ]$rating_count and DM[rindex == 2, ]$rating_count
t = -13.162, df = 17130, p-value < 2.2e-16
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
     -Inf -17426.4
sample estimates:
mean of x mean of y 
 10639.01  30554.35 
```

rindex1: China has smaller number of ratings than 2: USA as we reject the null hypothesis

b) tablets vs smart phones

```
> t.test(DM[deviceindex==1,]$rank, DM[deviceindex==2,]$rank, alternative = 'less')

        Welch Two Sample t-test

data:  DM[deviceindex == 1, ]$rank and DM[deviceindex == 2, ]$rank
t = -2.546, df = 22543, p-value = 0.005451
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
      -Inf -1.321124
sample estimates:
mean of x mean of y 
 182.3991  186.1319 
```

deviceindex: SmartPhones have lower rankings in average than 2: tablets as we reject the null hypothesis

```
> t.test(DM[deviceindex==1,]$price, DM[deviceindex==2,]$price, alternative = 'less')

        Welch Two Sample t-test

data:  DM[deviceindex == 1, ]$price and DM[deviceindex == 2, ]$price
t = -11.107, df = 15685, p-value < 2.2e-16
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
       -Inf -0.4774137
sample estimates:
mean of x mean of y
 1.108360  1.668767
```

deviceindex: SmartPhones have lower average prices than 2: tablets as we reject the null hypothesis

```
> t.test(DM[deviceindex==1,]$average_rating, DM[deviceindex==2,]$average_rating, alternative = 'greater')

        Welch Two Sample t-test

data:  DM[deviceindex == 1, ]$average_rating and DM[deviceindex == 2, ]$average_rating
t = 7.8115, df = 20906, p-value = 2.957e-15
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 0.07161441        Inf
sample estimates:
mean of x mean of y
 4.207855  4.117137
```

deviceindex: SmartPhones have greater value of average ratings than 2: tablets as we reject the null hypothesis

```
> t.test(DM[deviceindex==1,]$rating_count, DM[deviceindex==2,]$rating_count, alternative = 'greater')

        Welch Two Sample t-test

data:  DM[deviceindex == 1, ]$rating_count and DM[deviceindex == 2, ]$rating_count
t = 21.173, df = 16148, p-value < 2.2e-16
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 23702.06      Inf
sample estimates:
mean of x mean of y
29715.803  4017.226
```

deviceindex: SmartPhones have greater number of ratings than 2: tablets as we reject the null hypothesis

c) Apple vs Google

```
> t.test(DM[appstoreindex==2,]$rank, DM[appstoreindex==3,]$rank, alternative = 'greater')

        Welch Two Sample t-test

data:  DM[appstoreindex == 2, ]$rank and DM[appstoreindex == 3, ]$rank
t = 5.0326, df = 4811.2, p-value = 2.507e-07
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 6.977674      Inf
sample estimates:
mean of x mean of y
 185.5618  175.1953
```

Apple has higher ranking in average than Google as we reject the null hypothesis

```
> t.test(DM[appstoreindex==2,]$price, DM[appstoreindex==3,]$price, alternative = 'greater')

        Welch Two Sample t-test

data:  DM[appstoreindex == 2, ]$price and DM[appstoreindex == 3, ]$price
t = 13.573, df = 6693.7, p-value < 2.2e-16
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 0.5823405       Inf
sample estimates:
mean of x mean of y
1.4371148 0.7744555
```

Apple has higher prices in average than Google as we reject the null hypothesis

```
> t.test(DM[appstoreindex==2,]$average_rating, DM[appstoreindex==3,]$average_rating, alternative = 'less')

        Welch Two Sample t-test

data:  DM[appstoreindex == 2, ]$average_rating and DM[appstoreindex == 3, ]$average_rating
t = -19.046, df = 14137, p-value < 2.2e-16
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
       -Inf -0.1515828
sample estimates:
mean of x mean of y
 4.146873  4.312786
```

Apple has smaller average ratings than Google as we reject the null hypothesis

```
> t.test(DM[appstoreindex==2,]$rating_count, DM[appstoreindex==3,]$rating_count, alternative = 'less')

        Welch Two Sample t-test

data:  DM[appstoreindex == 2, ]$rating_count and DM[appstoreindex == 3, ]$rating_count
t = -22.836, df = 3553.6, p-value < 2.2e-16
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
      -Inf -97343.64
sample estimates:
 mean of x  mean of y
  4243.126 109144.863
```

Apple has smaller average rating count than Google as we reject the null hypothesis

d) free vs paid apps

```
> t.test(DM[apptypeindex==1,]$rank, DM[apptypeindex==3,]$rank, alternative = 'greater')

        Welch Two Sample t-test

data:  DM[apptypeindex == 1, ]$rank and DM[apptypeindex == 3, ]$rank
t = 4.0805, df = 16062, p-value = 2.258e-05
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 4.259819       Inf
sample estimates:
mean of x mean of y
 185.3459  178.2090
```

Free Apps have higher ranking in average than paid apps as we reject the null hypothesis

```
> t.test(DM[apptypeindex==1,]$average_rating, DM[apptypeindex==3,]$average_rating, alternative = 'less')

        Welch Two Sample t-test

data:  DM[apptypeindex == 1, ]$average_rating and DM[apptypeindex == 3, ]$average_rating
t = 9.9128, df = 11912, p-value = 1
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
      -Inf 0.1666346
sample estimates:
mean of x mean of y
 4.214788  4.071870
```

Free apps have higher average ratings than paid apps as we reject the null hypothesis.

```
> t.test(DM[apptypeindex==1,]$rating_count, DM[apptypeindex==3,]$rating_count, alternative = 'greater')

        Welch Two Sample t-test

data:  DM[apptypeindex == 1, ]$rating_count and DM[apptypeindex == 3, ]$rating_count
t = 20.701, df = 10111, p-value < 2.2e-16
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 30602.74      Inf
sample estimates:
mean of x mean of y
35496.756  2252.197
```

Free apps have greater number of ratings compared to the paid apps.

Conclusion

Although China has lower average prices, the average ratings and average rating counts are both low compared to USA. Thus, lowering prices may not be the right way to improve the demand in China. However, the rankings are higher in China pointing to a higher demand in China. Thus, there might be a possible avenue to increase prices without really affecting the demand.

There is an indication that the prices of apps are lower for smartphones than for tablets. This may also be the reason why the average ratings and rating count is greater in the smartphone platform. Also, additionally, as the number of ratings is quite low for tablets, it also probably shows that not many people have tablets, but a far larger set of people have smartphones. While this might seem obvious, one could draw an inference that the smartphones have more updates to apps and the apps are better optimized for smartphones in the current market which is why even the average ratings on the tablet apps are lower. It is also important to notice that the ranking in smartphone is lower than tablets, possibly showing that the customers are interested in good apps for tablets.

The demand in terms for rank in Apple seems to be higher compared to google. However, the prices are also higher in apple which may explain the demand in terms of higher rank. On the other hand, Apple has smaller number of ratings and smaller average ratings compared to google. This is possibly because Google has low-cost devices which are accessible to a larger number of people, making apps also available to a larger number of people.

Free apps seem to have higher demand in terms of ranking. The average ratings and average count of ratings also seem to be higher for free apps. Thus, maybe switching to a business model where the app appears to be free for download and use could improve demand.