Mobile Application Market Visualization

September 19, 2018

Importing Necessary Libraries

```
pkgTest <- function(x)
  {
    if (!require(x,character.only = TRUE))
    {
      install.packages(x,dep=TRUE)
        if(!require(x,character.only = TRUE)) stop("Package not found")
    }
  }
pkgTest('tidyr')
pkgTest('dplyr')
pkgTest('ggplot2')
pkgTest('stringr')
pkgTest('data.table')
pkgTest('rmarkdown')
pkgTest('forcats')
pkgTest('gridExtra')
```

Basic Summary Statistics of Input Data

```
setwd("M:/MSBA6410 EDA/HW1/HW 1/HW 1")
apps <- read.csv("mobileApps.csv", header = TRUE, stringsAsFactors = F)
summary(apps)
```

```
##     device              t_day        crawl_date             rank
##  Length:25129       Min.   :113.0   Length:25129       Min.   :  1
##  Class :character   1st Qu.:114.0   Class :character   1st Qu.: 84
##  Mode  :character   Median :116.0   Mode  :character   Median :175
##                     Mean   :115.9                      Mean   :184
##                     3rd Qu.:118.0                      3rd Qu.:280
##                     Max.   :119.0                      Max.   :408
##
##   app_store            region          release_date
##  Length:25129       Length:25129       Length:25129
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  app_age_current_version  developer           app_type
##  Min.   :   0.00         Length:25129       Length:25129
##  1st Qu.:  24.00         Class :character   Class :character
##  Median :  38.00         Mode  :character   Mode  :character
##  Mean   :  81.67
##  3rd Qu.:  83.00
##  Max.   :1275.00
```

```
##  NA's   :1256
##      price           filesize..MB.    num_screenshot    rating_count
##  Min.   :    0.000   Min.   :   1.0   Min.   : 1.000   Min.   :      0
##  1st Qu.:    0.000   1st Qu.:  11.0   1st Qu.: 5.000   1st Qu.:    116
##  Median :    0.000   Median :  26.1   Median : 6.000   Median :    707
##  Mean   :    2.141   Mean   :  93.0   Mean   : 7.092   Mean   :  19029
##  3rd Qu.:    0.990   3rd Qu.:  50.5   3rd Qu.:10.000   3rd Qu.:   4102
##  Max.   :12448.344   Max.   :2300.0   Max.   :10.000   Max.   :3077855
##
##  average_rating    category          in_app_ads        in_app_purchase
##  Min.   : 0.00   Length:25129       Length:25129       Length:25129
##  1st Qu.: 4.00   Class :character   Class :character   Class :character
##  Median : 4.50   Mode  :character   Mode  :character   Mode  :character
##  Mean   :10.19
##  3rd Qu.: 4.50
##  Max.   :50.00
##
##  num_issues_reported
##  Min.   :0.00000
##  1st Qu.:0.00000
##  Median :0.00000
##  Mean   :0.02945
##  3rd Qu.:0.00000
##  Max.   :4.00000
##
```

Data Cleaning

Changing the column headers to be more intuitive and devoid of special characters for ease of access.

```
setnames(apps , "filesize..MB.", "filesize")
```

Correcting for inconsistent data in the average_rating column.
Treatment : Replace entries containing average_rating as 50 with 5.0 *Reason:* Since the rating range is from 1-5, it is not possible to have values like 50.
Impact: ~3500 records have this rating of 50.
Assumption : We are assuming that all the people who have voted for an app, have given a rating as high as 5.0.

```
apps$average_rating[apps$average_rating == 50] <- 5.0
```

Changing the datatype of crawl_date from character to Date.
Treatment: Format the date column as YYYY-MM-DD.
Reason: The column type was looked upon in the summary of the data and had string as the format.
Assumption: If we have to perform any day level analysis, R or any other language will not be able to recognize it as a date rather will consider this as a string entry.

```
apps$crawl_date <- as.Date(apps$crawl_date,format = "%m/%d/%y")
```

Changing the datatype of release_date from character to Date.
Treatment: Format the date column as YYYY-MM-DD.
Reason: The column type was looked upon in the summary of the data and had string as the format.
Assumption: If we have to perform any day level analysis, R or any other language will not be able to recognize it as a date rather will consider this as a string entry.

```
apps$release_date <- as.Date(apps$release_date,format = "%m/%d/%y")
```

Replacing special characters with blanks to condense categories to be intuitive.
Treatment: Replace "åÊ" with ""
Reason: Categories like Games had two entries in the category column (Games, Games åÊ). Also, having special characters might make it difficult while performing search operations.
Assumption: the categories with and without the "åÊ" special character are comparable and can be condensed into a single category.

```
apps$category <- str_replace(apps$category,"åÊ","")
apps$category <- str_trim(apps$category)
```

Correcting for NAs in the age of current version of the app.
Treatment: App age = Crawl date - Release date
Reason: We checked the valid entries in this column and found that the value populated in this column is a difference in days between crawl date and release date.

```
apps$app_age_current_version[is.na(apps$app_age_current_version)] <- as.numeric(apps$crawl_date[is.na(ap
```

Removing price outliers (greater than $100).
Treatment: Filter for price <= 100
Reason: When sorted the dataset by price in descending order, we found 5-6 instances of price > $1500 which felt very unusual. Also, such rows had a very similar entry in terms of all other columns and this looked like a duplicate entry.
Assumption: The assumption is that this row has been captured by mistake and can be safely removed.

```
apps <- filter(apps, price <= 100)
```

```
## Warning: package 'bindrcpp' was built under R version 3.5.2
```

Removing duplicates in the data at Crawl date, Device, Rank, Category, App store, Region, Developer and App type level.
Treatment: Keep the first occurrance of each combination of datapoints at the mentioned level.
Reason: It is not possible for an app at that level to have two entries with potentially different values.
Assumption: The first entry at the mentioned level is the correct entry and the further occurrances are erreneous.

```
apps <- apps[!duplicated(apps[,c("crawl_date","device","rank","category","app_store","region","developer
```

Filtering out rows with NAs in the region field.
Treatment: Exclude rows having region as NA.
Assumption: Removing 1.5% of datapoints having NAs would not affect our data adversely.

```
apps <- filter(apps, region != 'NA')
```

Replacing 0 rating count and average rating with the mean rating count and average rating of the developer respectively.
Treatment: Replace 0 with mean rating value (count and/or average rating) of developer.
Reason: Alternate options would be to remove the rows or give an average value at an overall level. But we voted against removing the rows for the worry of losing data and did not impute it with overall mean value as this data comprises of multiple categories and all will not present the correct picture.
Assumption: We assume that the developer's mean rating value is the right representative of a 0 rating value.

```
apps <- apps %>%
  group_by(developer) %>%
  mutate(rating_count= ifelse(rating_count == 0, mean(rating_count, na.rm=TRUE), rating_count),
         average_rating= ifelse(average_rating == 0, mean(average_rating, na.rm=TRUE), average_rating))
```

Remove entries with developers having 0 rating count or and average rating even after the replacement in the previous steps.
Treatment: Filter out entries having 0 rating count or 0 average rating.
Assumption: We assumed that using average rating or rating counts having 0 values would not add value to identify the success of an app, and considering that there are inly 148 such entries, it is safe to remove these datapoints.

```
apps <- filter(apps, average_rating != 0 & rating_count != 0)
```

Subset for the last crawl date from the data.
Treatment: We have filtered data which belongs to the final crawl date for our analyses.
Assumption: Since we are not comparing the performance of the same apps over a period of time and just rather understanding how the overall app market is performing, this one cross-section of data will suffice as the success measurement attributes are a factor of the same day.

```
#12. Filtering out for the last "crawl_date":
apps_last_day <- apps[which(apps$crawl_date == max(apps$crawl_date)),]
```

Converting categorical columns into dummies for further analysis.
Treatment: Replace NO_IN_APP_ADS as 0 and IN_APP_ADS as 1 in in_app_ads column and replace NO_IN_APP_PURCHASE as 0 and PLUGIN_PURCHASE as 1 in in_app_purchase column.

```
apps_last_day$in_app_ads <- ifelse(apps_last_day$in_app_ads == "NO_IN_APP_ADS",0,1)
apps_last_day$in_app_purchase <- ifelse(apps_last_day$in_app_purchase == "NO_IN_APP_PURCHASE",0,1)
```

Analyses

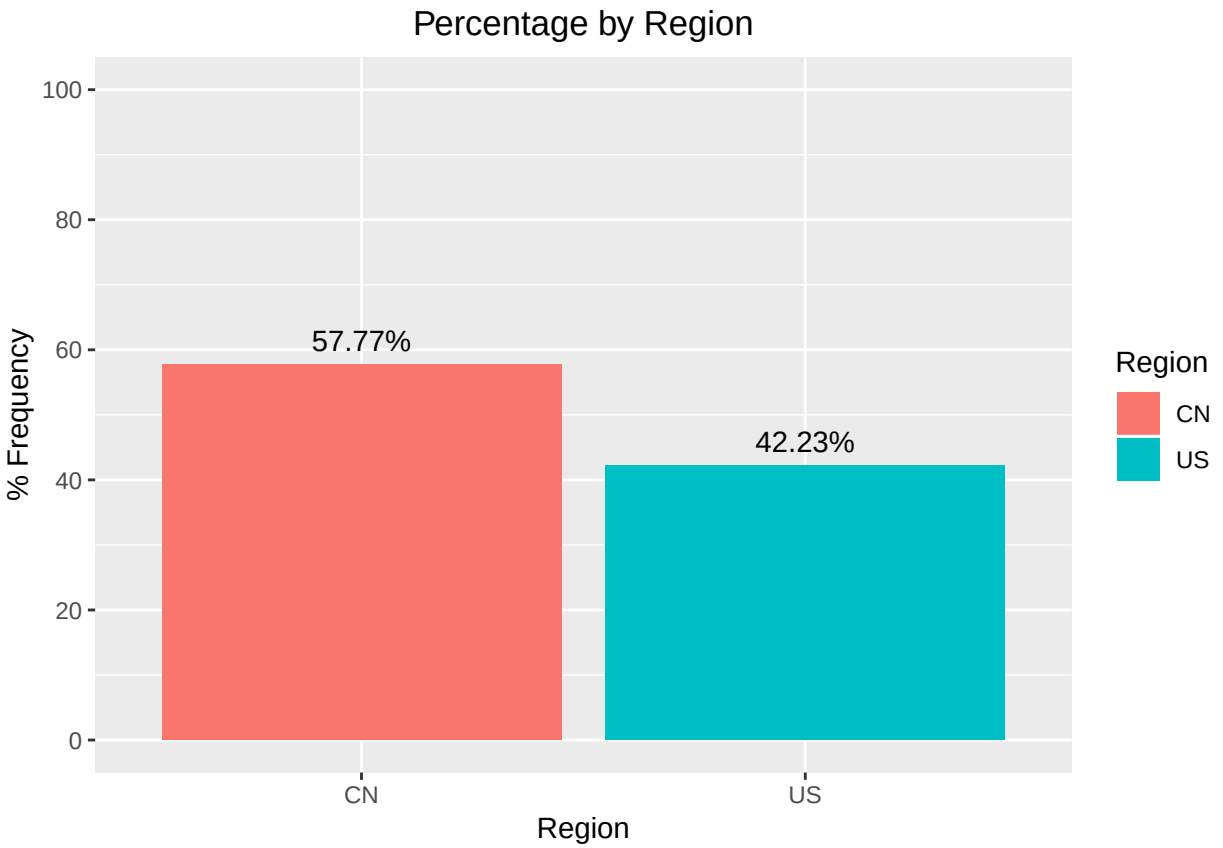
Part I: Region - Deciding which region to focus on

Reason: Region would be one of the key factors to consider while trying to enter into the mobile application market.
US and China are two of the most emerging regions in the global application market where opportunities are in abundance. In order to enter and succeed in the market, the client has to compare these two regions and choose between them given the limited availability of resources at hand.
Recommendation: We recommend that the client should focus on US region over China.
Sub-analyses: To make this recommendation, we have looked at three factors - level of competition, market awareness, and maintenance efforts in terms of app updates.

Level of competition

Assumption: Since we are looking at number of apps present in a region on a given day, comparison of number of apps in the regional market becomes straightforward.
Insight: Based on this analysis, we realize that the competition in the US market is lower than that of China. Hence entering the US market would be more beneficial.

```
ggplot(apps_last_day, aes(x = region,y = (..count..)/sum(..count..)*100,fill = region)) +
  geom_bar(aes(y = prop.table(..count..)*100),position = "dodge") +
  geom_text(aes(y = prop.table(..count..) * 100 + 0.5,
                label = paste0(round(prop.table(..count..),4) * 100, '%')),
            stat = 'count', vjust = -0.5,
            position = position_dodge(1),
            size = 4) +
  labs(x = 'Region', y = '% Frequency', fill = 'Region') +
  ggtitle("Percentage by Region") +
  scale_y_continuous(limits = c(0,100),breaks =seq(0,100,20)) +
  theme(plot.title = element_text(hjust = 0.5))
```

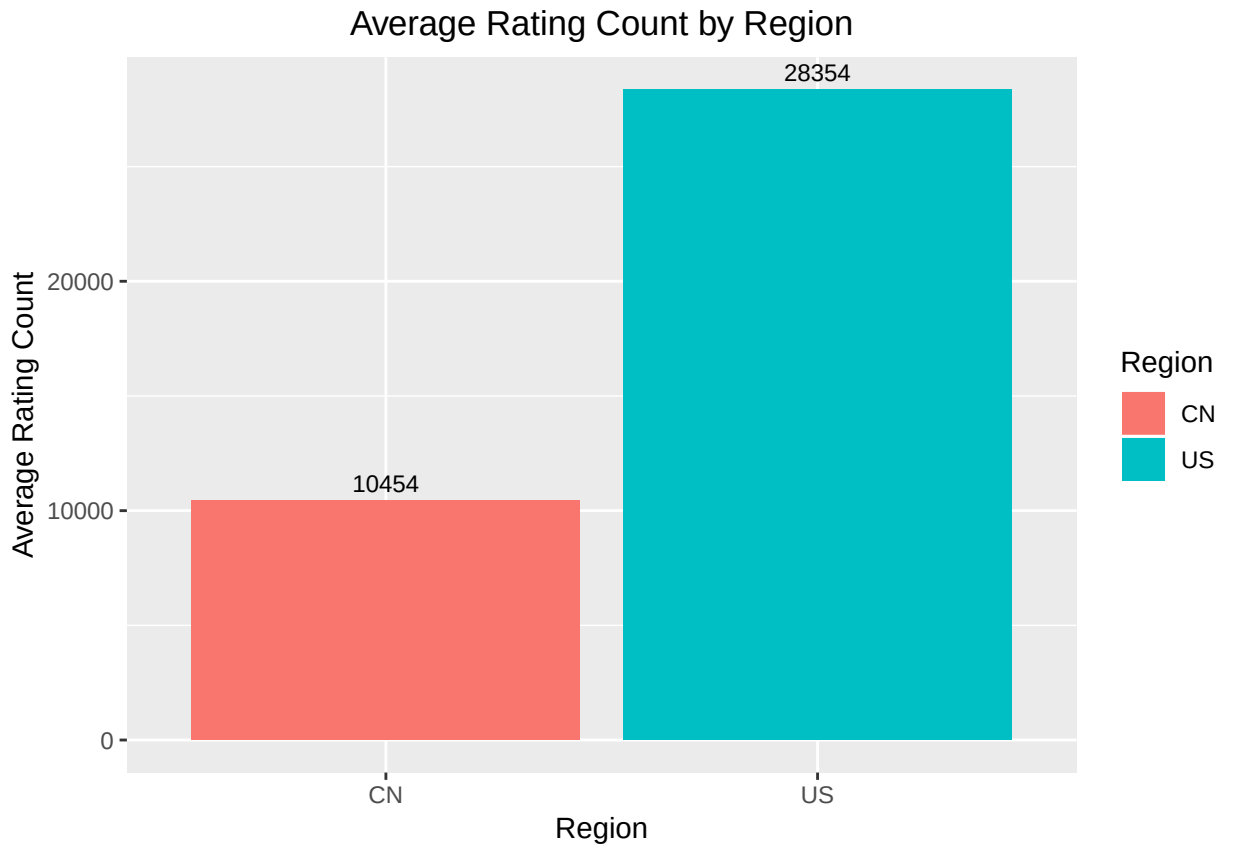


Market awareness

Assumption: Since we are looking at number of apps present in a region on a given day, comparison of number of apps in the regional market becomes straightforward. We also assumed that the rating count is a good indicator of the awareness among the users in the market.

Insight: Based on this analysis, we realize that there are more user ratings in the US app stores that that of China. Hence we can conclude that the user engagement and market awareness in better in US than in China.

```
ggplot(apps_last_day, aes(x=region, y=rating_count, fill = region)) +
  stat_summary(fun.y="mean", geom="bar") +
  stat_summary(aes(label=round(..y..,0)), fun.y=mean, geom="text", size=3,
               vjust = -0.5) +
  labs(x = 'Region', y = 'Average Rating Count', fill = 'Region') +
  ggtitle("Average Rating Count by Region") +
  theme(plot.title = element_text(hjust = 0.5))
```

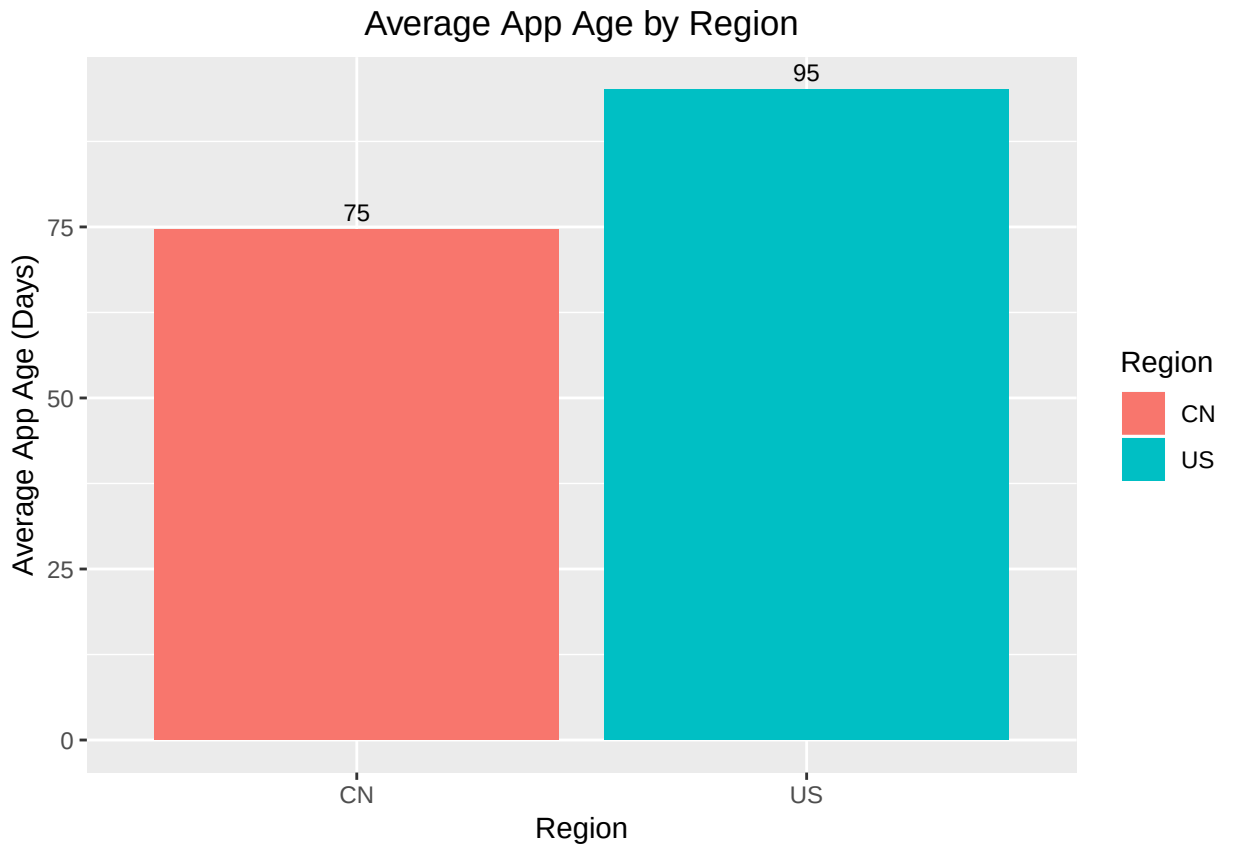


Maintenance efforts in terms of app updates

Assumption: We are assuming that frequency of developing updates/apps represents the effort in developing/maintaining apps. So, higher the frequency of developing apps/updates, higher the effort in developing/maintaining apps.

Recommendation: The average app update period is shorter in China than US (75 days Vs. 95 days). Hence targeting the US market will require less effort on the updates.

```
ggplot(apps_last_day, aes(x=region, y=app_age_current_version, fill = region)) +
  stat_summary(fun.y="mean", geom="bar") +
  stat_summary(aes(label=round(..y..,0)), fun.y=mean, geom="text", size=3,
               vjust = -0.5) +
  labs(x = 'Region', y = 'Average App Age (Days)', fill = 'Region') +
  ggtitle("Average App Age by Region") +
  theme(plot.title = element_text(hjust = 0.5))
```

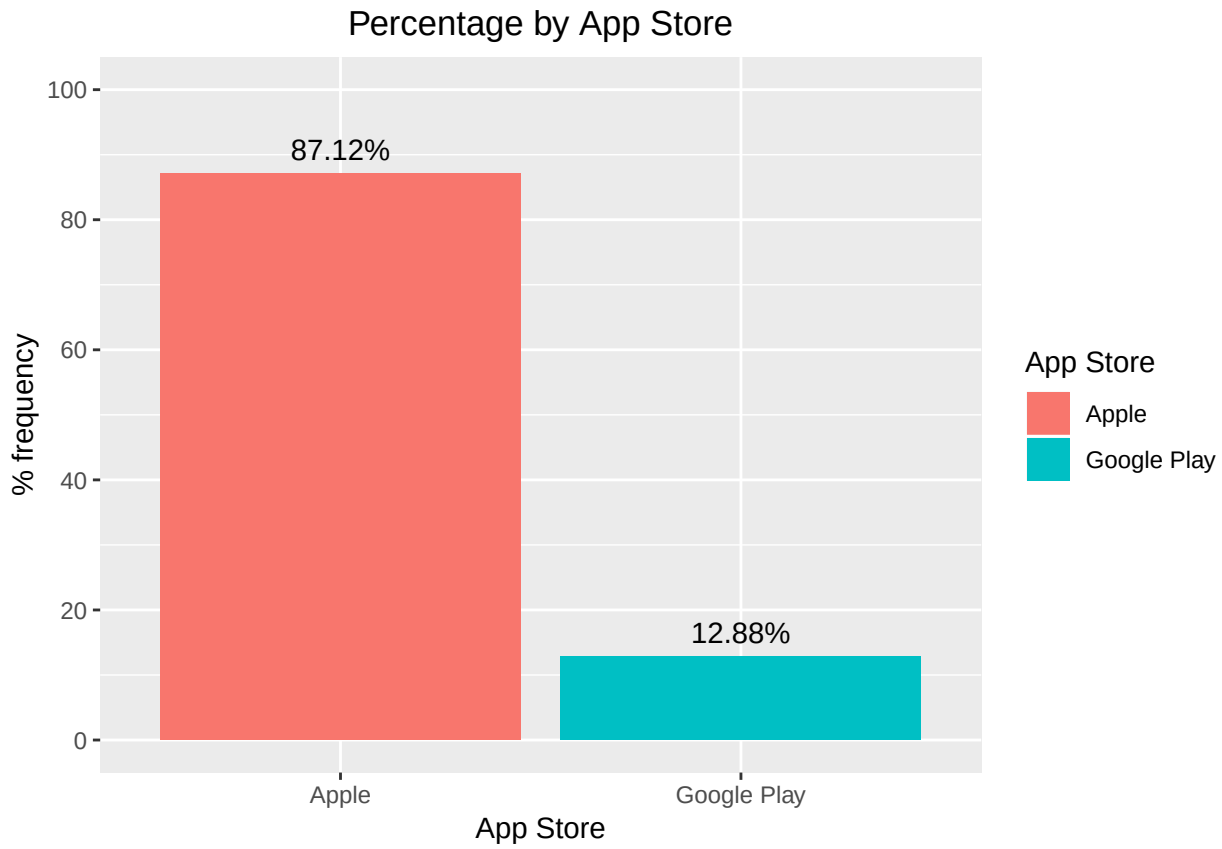


Part II: Platform - Identify which app store to target

Reason: The Apple Store is highly competitive (defined by number of apps) as compared to Google Play store. As per the graph below, we see that 87% of apps are currently present in Apple App Store while only 13% are present in Google Play store.
Recommendation: New player in Mobile App business should target to launch Apps in Google Play store.
Sub-analyses: To make this recommendation, we have looked at three factors - level of competition, market awareness, and maintenance efforts in terms of app updates.

Level of competition
Assumption: Since we are looking at number of apps present across app stores on a given day, comparison of number of apps across app stores becomes straightforward.
Insight: Based on this analysis, we realize that the competition in the US market is lower than that of China. Hence entering the US market would be more beneficial.

```
ggplot(apps_last_day, aes(x = app_store,y = (..count..)/sum(..count..)*100,fill = app_store)) +
  geom_bar(aes(y = prop.table(..count..)*100),position = "dodge") +
  geom_text(aes(y = prop.table(..count..) * 100 + 0.5,
                label = paste0(round(prop.table(..count..),4) * 100, '%')),
            stat = 'count', vjust = -0.5,
            position = position_dodge(1),
            size = 4) +
  labs(x = 'App Store', y = '% frequency', fill = 'App Store') +
  ggtitle("Percentage by App Store") +
  scale_y_continuous(limits = c(0,100),breaks =seq(0,100,20)) +
  theme(plot.title = element_text(hjust = 0.5))
```

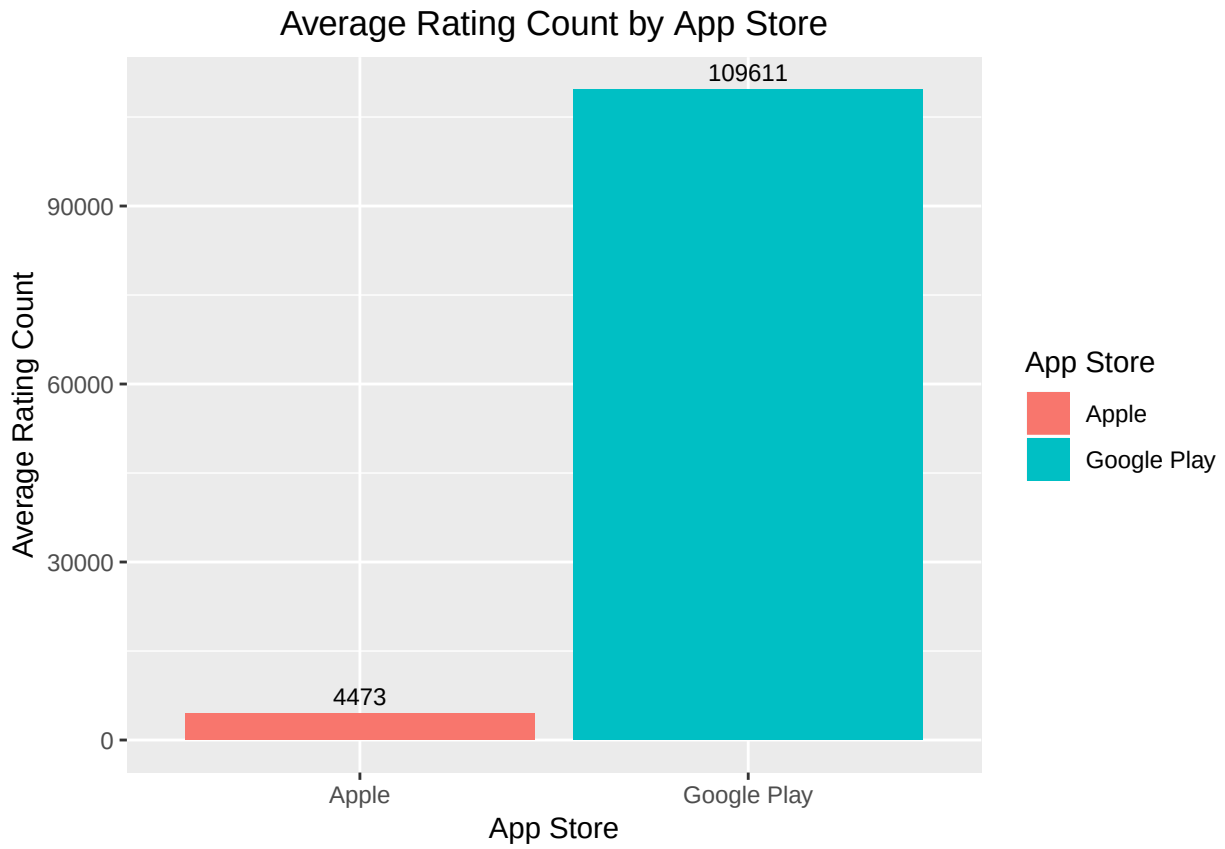


Market Awareness

Assumption: Since we are looking at number of apps present in an app store on a given day, comparison of number of apps across app stores regional market becomes straightforward. We also assumed that the rating count is a good indicator of the awareness among the users in the market.

Insight: Based on the analysis, we can conclude that Google Play provides better understanding of the market as compared to Apple App Store for the developers since there are more 96% of rating counts are occurring in the Google play store.

```
ggplot(apps_last_day, aes(x=app_store, y=rating_count, fill = app_store)) +
  stat_summary(fun.y="mean", geom="bar") +
  stat_summary(aes(label=round(..y..,0)), fun.y=mean, geom="text", size=3,
               vjust = -0.5) +
  labs(x = 'App Store', y = 'Average Rating Count', fill = 'App Store') +
  ggtitle("Average Rating Count by App Store") +
  theme(plot.title = element_text(hjust = 0.5))
```

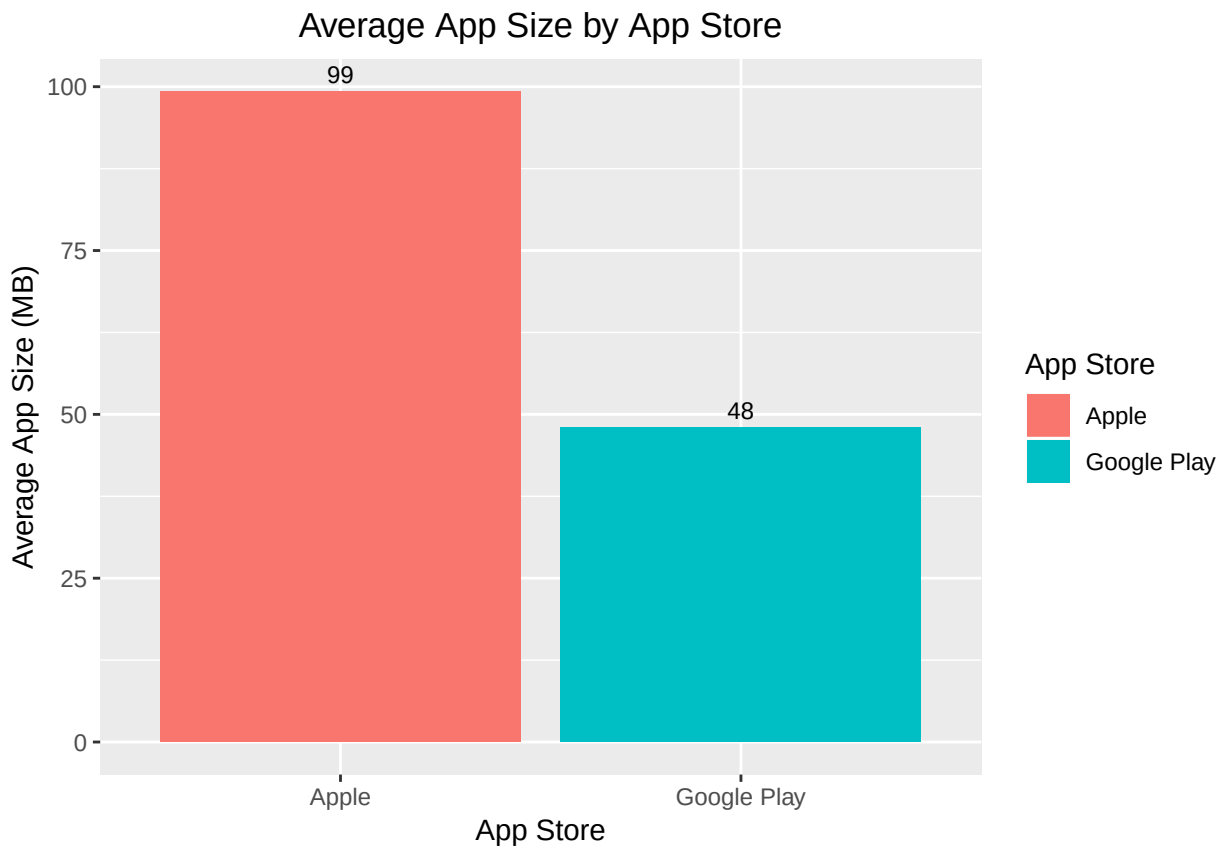


Maintenance effort

Assumption: We assumed that app size is directly proportional to effort on development.

Insight: With average app size much smaller in Google Play store as compared to Apple app store (48 vs 99), launching apps (including updates) in Google Play store requires relatively lower effort on development.

```
ggplot(apps_last_day, aes(x=app_store, y=filesize, fill = app_store)) +
  stat_summary(fun.y="mean", geom="bar") +
  stat_summary(aes(label=round(..y..,0)), fun.y=mean, geom="text", size=3,
               vjust = -0.5) +
  labs(x = 'App Store', y = 'Average App Size (MB)', fill = 'App Store') +
  ggtitle("Average App Size by App Store") +
  theme(plot.title = element_text(hjust = 0.5))
```

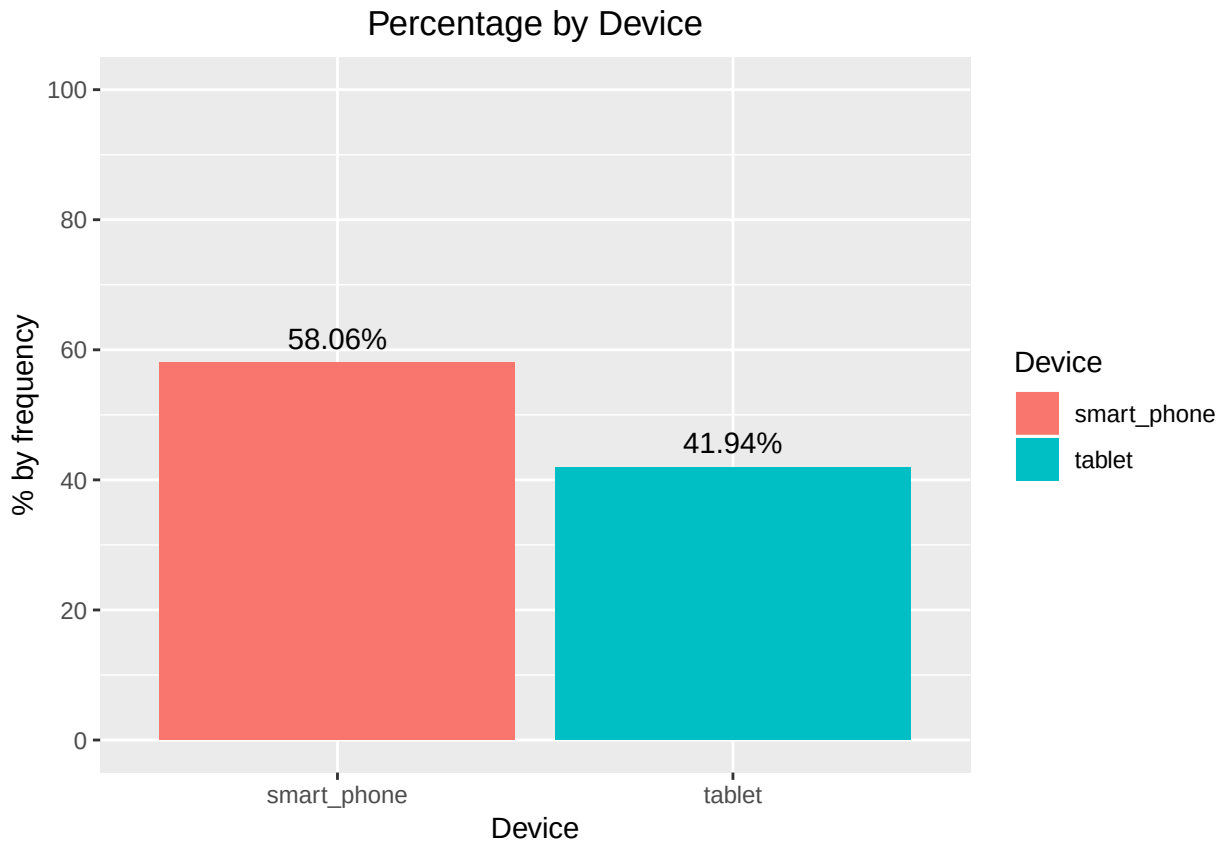


Part III: Device - Identify which device to target

Reason: Smart phones and tablet are the two devices available to target but the charecteristics for the same app is different for both device types. Also tablet device is present only in Apple store.
Recommendation: As a new player in Mobile App business, we recommend launching Apps suitable for Smartphones instead of Tablet because of the user base, market awareness and maintenance effort.
Sub-analyses: To make this recommendation, we have looked at three factors - level of competition, market awareness, and Creating efforts/costs in terms of app size.

Level of competition
Assumption: people are exposed to both smart phones and tablets equally across both app stores.
Insight: More smart phone apps are present in the app market compared to tablets.

```
ggplot(apps_last_day, aes(x = device,y = (..count..)/sum(..count..)*100,fill = device)) +
  geom_bar(aes(y = prop.table(..count..)*100),position = "dodge") +
  geom_text(aes(y = prop.table(..count..) * 100 + 0.5,
                label = paste0(round(prop.table(..count..),4) * 100, '%')),
            stat = 'count', vjust = -0.5,
            position = position_dodge(1),
            size = 4) +
  labs(x = 'Device', y = '% by frequency', fill = 'Device') +
  ggtitle("Percentage by Device") +
  scale_y_continuous(limits = c(0,100),breaks =seq(0,100,20)) +
  theme(plot.title = element_text(hjust = 0.5))
```

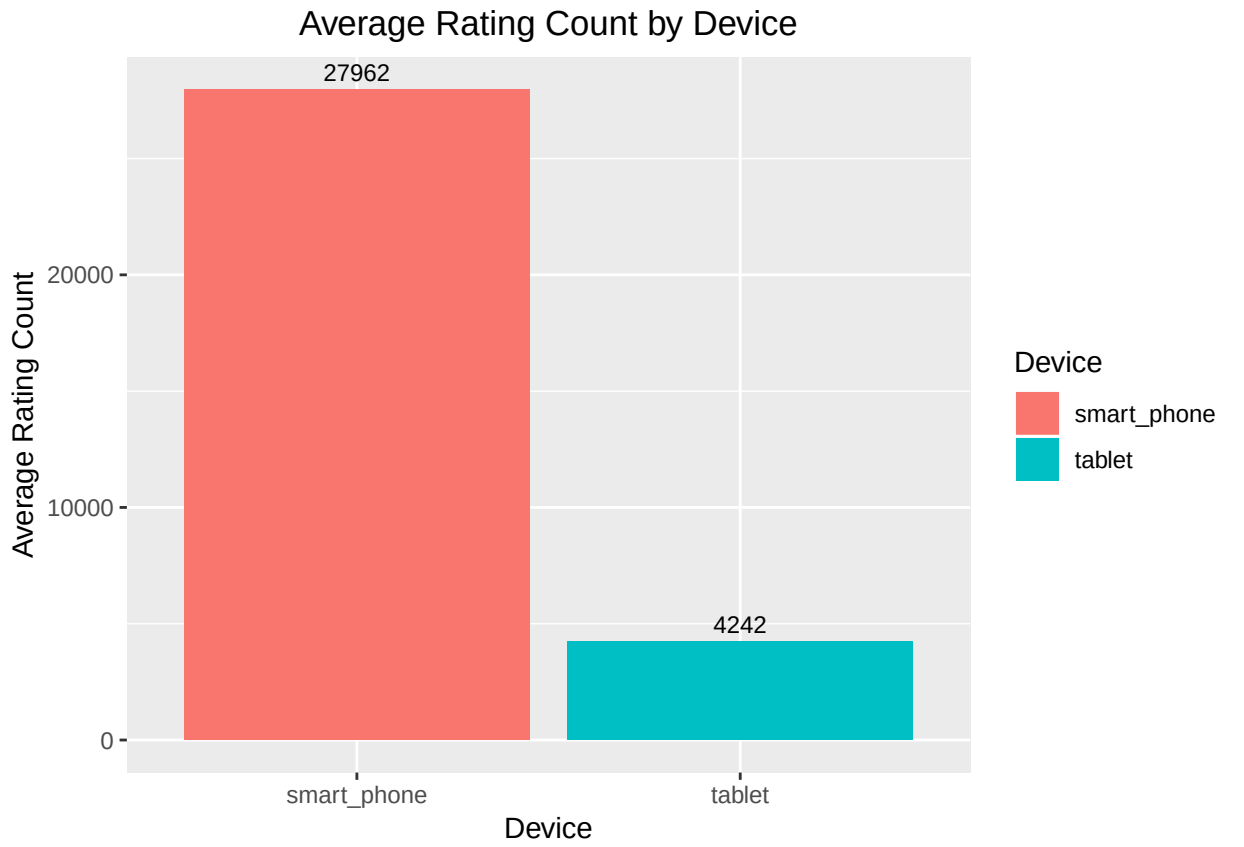


Market awareness

Assumption: We assumed that the rating count is a good indicator of the awareness among the users in the market.

Insight: On an average, smart phone apps are rated by more people compared to the tablet apps, suggesting a higher customer interaction.

```
ggplot(apps_last_day, aes(x=device, y=rating_count, fill = device)) +
  stat_summary(fun.y="mean", geom="bar") +
  stat_summary(aes(label=round(..y..,0)), fun.y=mean, geom="text", size=3,
               vjust = -0.5) +
  labs(x = 'Device', y = 'Average Rating Count', fill = 'Device') +
  ggtitle("Average Rating Count by Device") +
  theme(plot.title = element_text(hjust = 0.5))
```



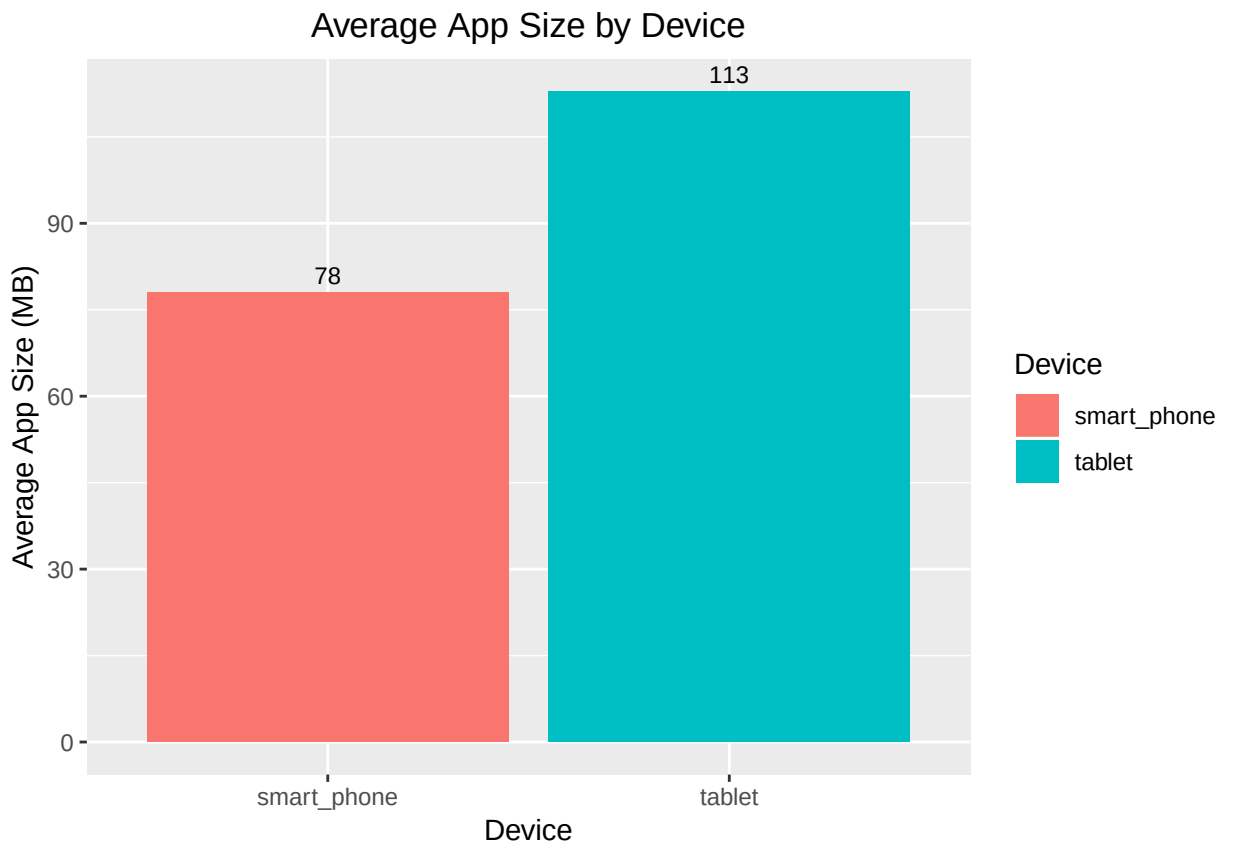
Creating efforts/costs in terms of app size

Assumption: We have assumed that the size of the app is directly propotional to the cost incurred and effort required to develop an app.

Insight: Average app size of tablet is comparitively higher than a smart phone app.

Recommendation: Since smart phone apps are smaller sized, and are more in use, it is recommended to have a smart phone app over a tablet app. Also since the customer interaction is high, we can have more points to improve on it.

```
ggplot(apps_last_day, aes(x=device, y=filesize, fill = device)) +
  stat_summary(fun.y="mean", geom="bar") +
  stat_summary(aes(label=round(..y..,0)), fun.y=mean, geom="text", size=3,
               vjust = -0.5) +
  labs(x = 'Device', y = 'Average App Size (MB)', fill = 'Device') +
  ggtitle("Average App Size by Device") +
  theme(plot.title = element_text(hjust = 0.5))
```



Part IV: Number of Apps to be deployed (including updates)

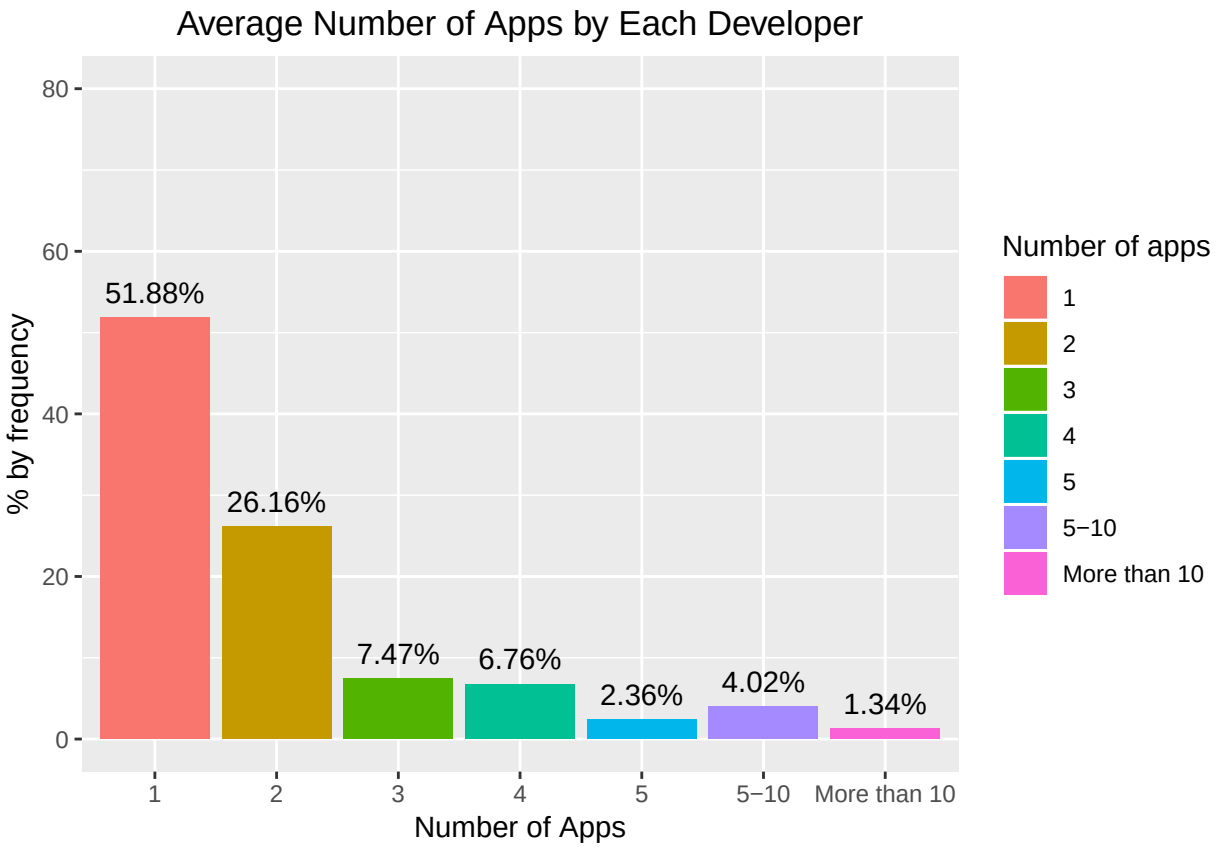
Reason: For a new player in the market, the number of apps to develop is one of the major decisions to make. For the client to make informed decision regarding the number, we looked at the average number of apps developed by each developer.
Recommendation: We recommend the client to focus on launching a single app in the market as compared to multiple apps.
Assumption: We assumed that each new app released (could be a new app or an update) as a separate entity in our analysis.
Insight: As shown in Figure 9, most of the players (51.88%) have only launched one single app in the market. Launching one single app will make the client more efficient in terms of resource allocation and more focused in terms of maintenance given the relative less experience that the client has, as a new entrant.

```
apps_by_developer <- apps_last_day %>%
  group_by(developer) %>%
  summarise(avg_count = n())

apps_by_developer$avg_count <- ifelse(apps_by_developer$avg_count <= 5,apps_by_developer$avg_count,
                                      ifelse(apps_by_developer$avg_count <= 10,"5-10","More than 10"))

ggplot(apps_by_developer, aes(x = avg_count,y = (..count..)/sum(..count..)*100,fill = avg_count)) +
  geom_bar(aes(y = prop.table(..count..)*100),position = "dodge") +
  geom_text(aes(y = prop.table(..count..) * 100 + 0.5,
                label = paste0(round(prop.table(..count..),4) * 100, '%')),
            stat = 'count', vjust = -0.5,
            position = position_dodge(1),
```

```
          size = 4) +
  labs(x = 'Number of Apps', y = '% by frequency', fill = 'Number of apps') +
  ggtitle("Average Number of Apps by Each Developer") +
  scale_y_continuous(limits = c(0,80),breaks =seq(0,80,20)) +
  theme(plot.title = element_text(hjust = 0.5))
```



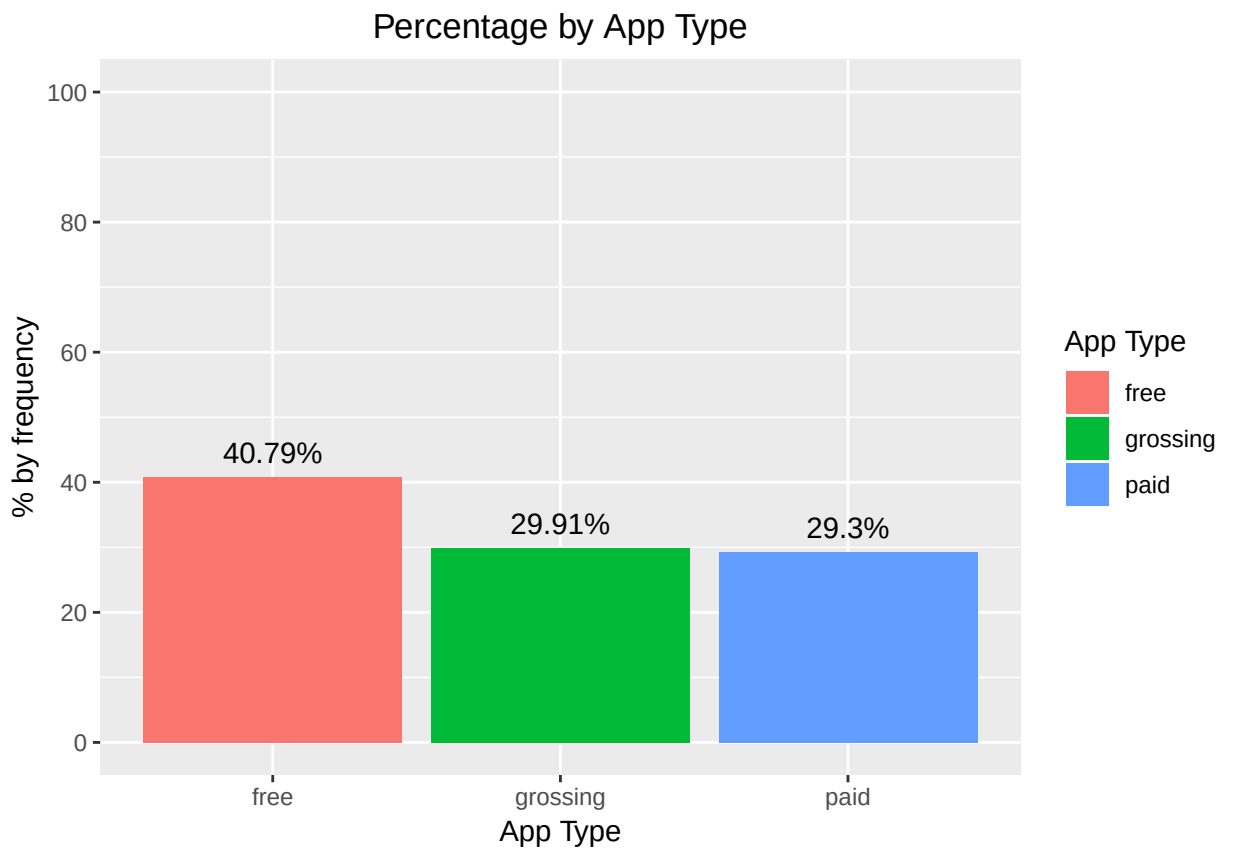
Part V: Free Vs. Grossing Vs. Paid - Choosing the right app type for entering the market

Reason: Being new in the app development arena, it is necessary to determine the ideal type of app to develop and deploy.
Recommendation: New player in Mobile App business should develop a free version to begin with.
Sub-analyses: To make this recommendation, we have looked at three factors - level of competition, market awareness, efforts required for app development, and maintenance efforts in terms of app updates.

Level of competition
Insight: Based on this analysis, in terms of the number of apps, the market is almost evenly distributed (as shown in plot. However the free apps are the mainstream in the market.

```
ggplot(apps_last_day, aes(x = app_type,y = (..count..)/sum(..count..)*100,fill = app_type)) +
  geom_bar(aes(y = prop.table(..count..)*100),position = "dodge") +
  geom_text(aes(y = prop.table(..count..) * 100 + 0.5,
                label = paste0(round(prop.table(..count..),4) * 100, '%')),
            stat = 'count', vjust = -0.5,
            position = position_dodge(1),
            size = 4) +
  labs(x = 'App Type', y = '% by frequency', fill = 'App Type') +
  ggtitle("Percentage by App Type") +
```

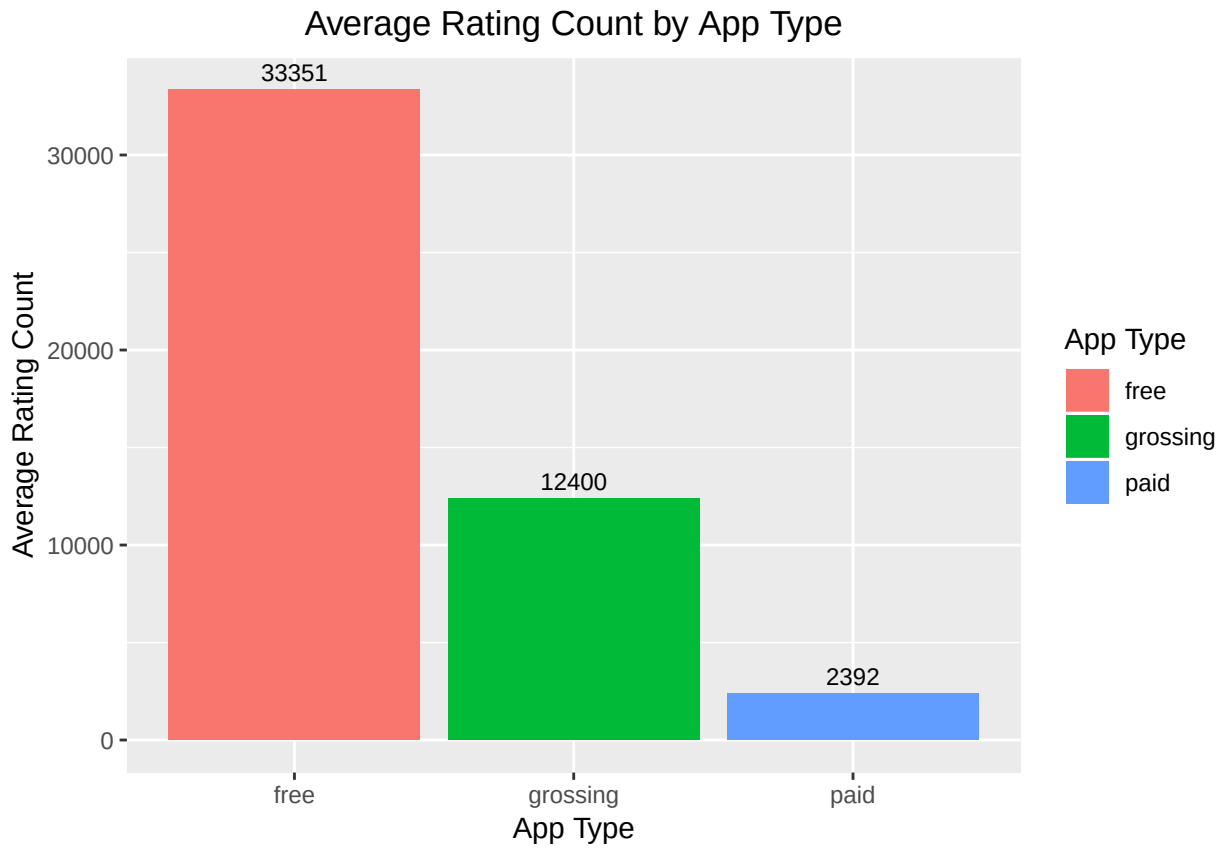
```
  scale_y_continuous(limits = c(0,100),breaks =seq(0,100,20)) +
  theme(plot.title = element_text(hjust = 0.5))
```



Market awareness

Insight: : Based on the analysis, in terms of popularity, free apps are way more popular than paid and grossing apps according to the rating counts.

```
ggplot(apps_last_day, aes(x=app_type, y=rating_count, fill = app_type)) +
  stat_summary(fun.y="mean", geom="bar") +
  stat_summary(aes(label=round(..y..,0)), fun.y=mean, geom="text", size=3,
               vjust = -0.5) +
  labs(x = 'App Type', y = 'Average Rating Count', fill = 'App Type') +
  ggtitle("Average Rating Count by App Type") +
  theme(plot.title = element_text(hjust = 0.5))
```

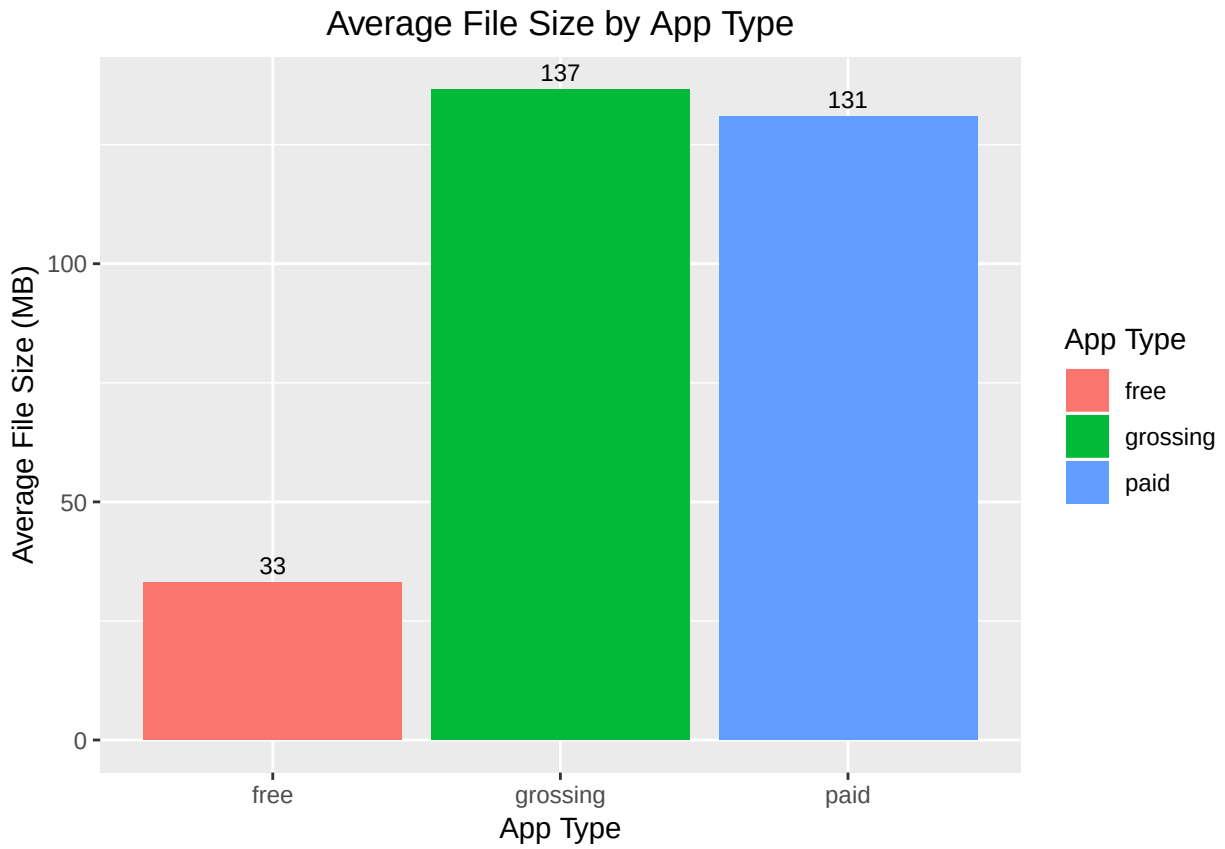


Efforts required for app development

Assumption: We assume that the development effort is directly proportional to the filesize of the app. Also, all three app types are present across all regions, app stores and categories.

Insight: : Based on the analysis, the average filesize is lowest for free apps and hence we should launch an app of free type to have the least maintenance efforts.

```
ggplot(apps_last_day, aes(x=app_type, y=filesize, fill = app_type)) +
  stat_summary(fun.y="mean", geom="bar") +
  stat_summary(aes(label=round(..y..,0)), fun.y=mean, geom="text", size=3,
               vjust = -0.5) +
  labs(x = 'App Type', y = 'Average File Size (MB)', fill = 'App Type') +
  ggtitle("Average File Size by App Type") +
  theme(plot.title = element_text(hjust = 0.5))
```

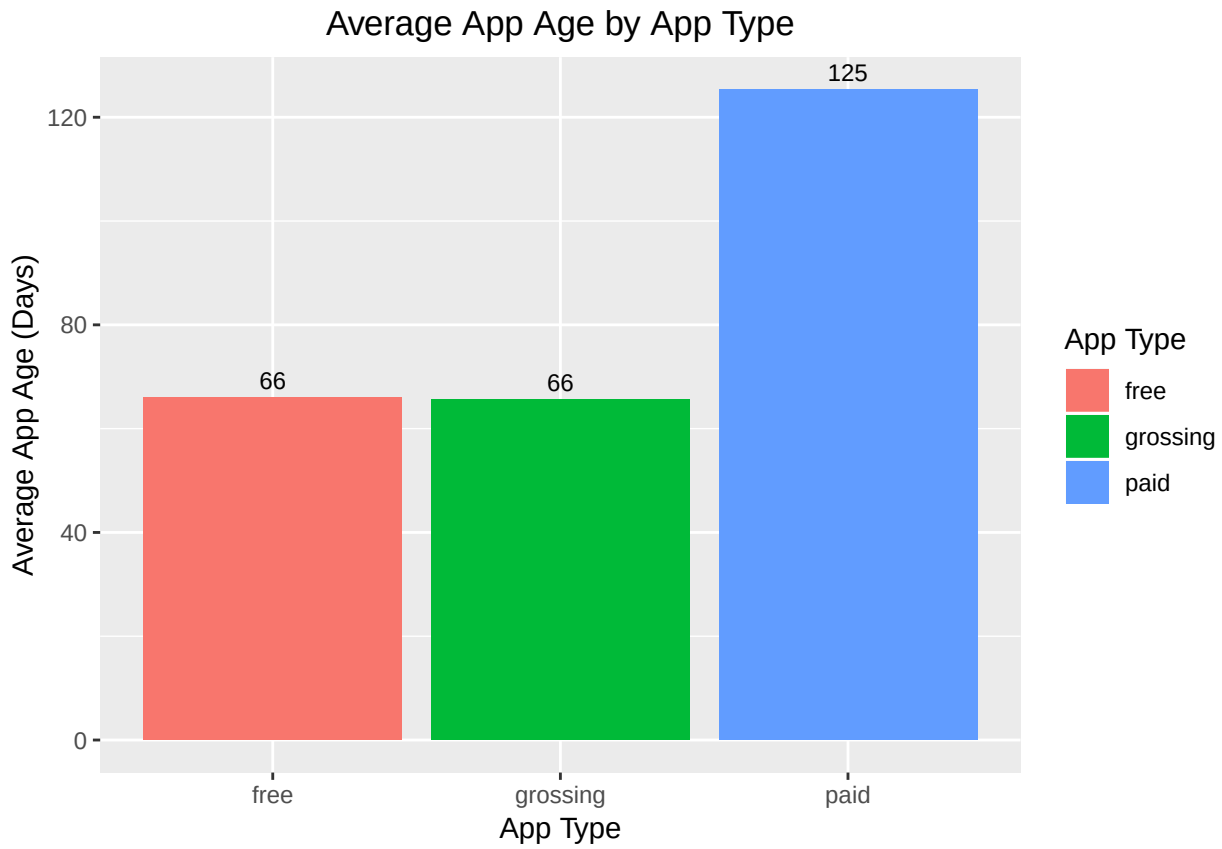


Maintenance efforts

Assumption: All three app types are present across all regions, app stores and categories. Also, We assume that the development effort is directly proportional to the filesize of the app.

Insight: : Based on the analysis, free apps are significantly sized smaller compared to paid and grossing apps. Though the frequency at which updates are launched for free and grossing apps is the same, developing free apps is relatively easier due to their lower size.

```
ggplot(apps_last_day, aes(x=app_type, y=app_age_current_version, fill = app_type)) +
  stat_summary(fun.y="mean", geom="bar") +
  stat_summary(aes(label=round(..y..,0)), fun.y=mean, geom="text", size=3,
               vjust = -0.5) +
  labs(x = 'App Type', y = 'Average App Age (Days)', fill = 'App Type') +
  ggtitle("Average App Age by App Type") +
  theme(plot.title = element_text(hjust = 0.5))
```


Average App Age by App Type
Average App Age (Days)
120
80
40
0
66
66
125
free
grossing
paid
App Type
App Type
free
grossing
paid