Data Analysis showing Profitable Apps for the Google Play and App Store Markets

The aim of this project is to find mobile app profiles that are profitable for the App Store and Google Play markets. I am working for a company that builds Android and iOS mobile apps that are free to download and install.

Our main source of revenue consists of in-app ads, which means our revenue is mostly influenced by the number of users who use our app - the more users that see and engage with the ads, the better. As a data analyst, my goal for this project is to analyze data to help our developers understand what type of apps are likely to attract more users.

Opening and Exploring the Data

As of September 2018, there were approximately 2 million iOS apps available on the App Store, and 2.1 million Android apps on Google Play. We will try to analyze a sample of data set containing data about approximately 7,000 iOS apps from the App Store collected in July 2017 and 10,000 Android apps fom the Google play collected in August 2018.

In [1]:
```
from csv import reader

### The Google Play data set ###
opened_file = open('googleplaystore.csv')
read_file = reader(opened_file)
android = list(read_file)
android_header = android[0]
android = android[1:]

### The App Store data set ###
opened_file = open('AppleStore.csv')
read_file = reader(opened_file)
ios = list(read_file)
ios_header = ios[0]
ios = ios[1:]
```

In [2]:
```
def explore_data(dataset, start, end, rows_and_columns=False):
    dataset_slice = dataset[start:end]
    for row in dataset_slice:
        print(row)
        print('\n') # adds a new (empty) line between rows

    if rows_and_columns:
        print('Number of rows:', len(dataset))
        print('Number of columns:', len(dataset[0]))

print(android_header)
print('\n')
explore_data(android, 0, 5, True)
```

```
['App', 'Category', 'Rating', 'Reviews', 'Size', 'Installs', 'Type', 'Price',
'Content Rating', 'Genres', 'Last Updated', 'Current Ver', 'Android Ver']


['Photo Editor & Candy Camera & Grid & ScrapBook', 'ART_AND_DESIGN', '4.1',
'159', '19M', '10,000+', 'Free', '0', 'Everyone', 'Art & Design', 'January 7,
2018', '1.0.0', '4.0.3 and up']


['Coloring book moana', 'ART_AND_DESIGN', '3.9', '967', '14M', '500,000+', 'F
ree', '0', 'Everyone', 'Art & Design;Pretend Play', 'January 15, 2018', '2.0.
0', '4.0.3 and up']


['U Launcher Lite – FREE Live Cool Themes, Hide Apps', 'ART_AND_DESIGN', '4.
7', '87510', '8.7M', '5,000,000+', 'Free', '0', 'Everyone', 'Art & Design',
'August 1, 2018', '1.2.4', '4.0.3 and up']


['Sketch - Draw & Paint', 'ART_AND_DESIGN', '4.5', '215644', '25M', '50,000,0
00+', 'Free', '0', 'Teen', 'Art & Design', 'June 8, 2018', 'Varies with devic
e', '4.2 and up']


['Pixel Draw - Number Art Coloring Book', 'ART_AND_DESIGN', '4.3', '967', '2.
8M', '100,000+', 'Free', '0', 'Everyone', 'Art & Design;Creativity', 'June 2
0, 2018', '1.1', '4.4 and up']


Number of rows: 10841
Number of columns: 13
```

From above, Google Play data set has 10841 apps and 13 columns. I decided to print the first few rows of the data set. At a quick glance, the columns that might be useful for our analysis are 'App', 'Category', 'Reviews', 'Installs', 'Type', 'Price', and 'Genres'.

We take a look at the App Store data set below:

In [3]:
```
print(ios_header)
print('\n')
explore_data(ios, 0, 5, True)
```

```
['id', 'track_name', 'size_bytes', 'currency', 'price', 'rating_count_tot',
'rating_count_ver', 'user_rating', 'user_rating_ver', 'ver', 'cont_rating',
'prime_genre', 'sup_devices.num', 'ipadSc_urls.num', 'lang.num', 'vpp_lic']


['284882215', 'Facebook', '389879808', 'USD', '0.0', '2974676', '212', '3.5',
'3.5', '95.0', '4+', 'Social Networking', '37', '1', '29', '1']


['389801252', 'Instagram', '113954816', 'USD', '0.0', '2161558', '1289', '4.
5', '4.0', '10.23', '12+', 'Photo & Video', '37', '0', '29', '1']


['529479190', 'Clash of Clans', '116476928', 'USD', '0.0', '2130805', '579',
'4.5', '4.5', '9.24.12', '9+', 'Games', '38', '5', '18', '1']


['420009108', 'Temple Run', '65921024', 'USD', '0.0', '1724546', '3842', '4.
5', '4.0', '1.6.2', '9+', 'Games', '40', '5', '1', '1']


['284035177', 'Pandora - Music & Radio', '130242560', 'USD', '0.0', '112687
9', '3594', '4.0', '4.5', '8.4.1', '12+', 'Music', '37', '4', '1', '1']


Number of rows: 7197
Number of columns: 16
```

From above, iOS data set has 7197 apps and 16 columns. I decided to print the first few rows of the data set. The columns that seem related to our analysis are: 'track_name', 'currency', 'price', 'rating_count_tot', 'rating_count_ver', and 'prime_g

Not all columns are self-explanatory. Details of each column description can be found in the dataset Documentation (https://www.kaggle.com/ramamet4/app-store-apple-data-set-10k-apps/home)

Data Cleaning

Deleting wrong data entry

Below, I check for row with incomplete column entry in the android data set and remove the row from the data set:

In [4]:
```
for row in android:
    headerlenght = len(android_header)
    rowlenght = len(row)
    if rowlenght != headerlenght:
        print(row)
        print(android.index(row))
```

```
['Life Made WI-Fi Touchscreen Photo Frame', '1.9', '19', '3.0M', '1,000+', 'F
ree', '0', 'Everyone', '', 'February 11, 2018', '1.0.19', '4.0 and up']
10472
```

Index row 10472 has a missing 'Category' value and below we delete the row as part of our data cleaning.

In [5]:
```
opened_file = open('googleplaystore.csv')
read_file = reader(opened_file)
android = list(read_file)
android_header = android[0]
android = android[1:]

print(len(android))
del android[10472]
print(len(android))
```

```
10841
10840
```

Finding duplicating entries and removing them

We don't want to count certain apps more than once when we analyze data, so we need to remove the duplicate entries and keep only one entry per app.

In [6]:
```
duplicate_apps = []
unique_apps = []

for app in android:
    name = app[0]
    if name in unique_apps:
        duplicate_apps.append(name)
    else:
        unique_apps.append(name)

print('Number of duplicate apps:', len(duplicate_apps))
print('\n')
print('Examples of duplicate apps:', duplicate_apps[:10])
print('\n')
print('Number of unique apps:', len(unique_apps))
```

```
Number of duplicate apps: 1181


Examples of duplicate apps: ['Quick PDF Scanner + OCR FREE', 'Box', 'Google M
y Business', 'ZOOM Cloud Meetings', 'join.me - Simple Meetings', 'Box', 'Zene
fits', 'Google Ads', 'Google My Business', 'Slack']


Number of unique apps: 9659
```

There are 1,181 duplicated entries and 9,659 unique entries. If we choose 'Google Ads' app for example and want to know how many entries, we will as below:

In [7]:
```
for app in android:
    name = app[0]
    if name == 'Google Ads':
        print(app)
```

```
['Google Ads', 'BUSINESS', '4.3', '29313', '20M', '5,000,000+', 'Free', '0',
'Everyone', 'Business', 'July 30, 2018', '1.12.0', '4.0.3 and up']
['Google Ads', 'BUSINESS', '4.3', '29313', '20M', '5,000,000+', 'Free', '0',
'Everyone', 'Business', 'July 30, 2018', '1.12.0', '4.0.3 and up']
['Google Ads', 'BUSINESS', '4.3', '29331', '20M', '5,000,000+', 'Free', '0',
'Everyone', 'Business', 'July 30, 2018', '1.12.0', '4.0.3 and up']
```

We will not remove duplicates at random. Looking at the Google Ads printed above, the main difference is on the 4th position of each row which corresponds to the number of reviews. We will use this as a criterion in removing duplicates by keeping the row with the highest number of reviews which suggests that to be the more recent data collected. In the case above with Google Ads, we will remove row 1 and 2 as duplicates and keep row 3.

We proceed by

- creating a dictionary where each key is a unique app name, and the value is the highest number of reviews of that app and
- Use the dictionary to create a new data set, which will have only one entry per app (and we only select the apps with the highest number of reviews)

In [8]:
```
reviews_max = {}

for app in android:
    name = app[0]
    n_reviews = float(app[3])

    if name in reviews_max and reviews_max[name] < n_reviews:
        reviews_max[name] = n_reviews

    elif name not in reviews_max:
        reviews_max[name] = n_reviews

print(len(reviews_max))
```

```
9659
```

Notice that as expected, the length of reviews_max in our dictionary is equals to the number of unique apps calcaulated previously as 9,659.

Now we can proceed to remove the duplicate using the reviews_max dictionary.

In [9]:
```
android_clean = []
already_added = []

for app in android:
    name = app[0]
    n_reviews = float(app[3])

    if (reviews_max[name] == n_reviews) and (name not in already_added):
        android_clean.append(app)
        already_added.append(name)

explore_data(android_clean, 0, 2, True)
```

```
['Photo Editor & Candy Camera & Grid & ScrapBook', 'ART_AND_DESIGN', '4.1',
'159', '19M', '10,000+', 'Free', '0', 'Everyone', 'Art & Design', 'January 7,
2018', '1.0.0', '4.0.3 and up']


['U Launcher Lite – FREE Live Cool Themes, Hide Apps', 'ART_AND_DESIGN', '4.
7', '87510', '8.7M', '5,000,000+', 'Free', '0', 'Everyone', 'Art & Design',
'August 1, 2018', '1.2.4', '4.0.3 and up']


Number of rows: 9659
Number of columns: 13
```

Our clean Android data set with no duplication consists of 9659 rows and 13 columns as expected.

For the App Store, we check to find duplicates as below:

In [10]:
```
duplicate_ios_apps = []
unique_ios_apps = []

for app in ios:
    name = app[0]
    if name in unique_ios_apps:
        duplicate_ios_apps.append(name)
    else:
        unique_ios_apps.append(name)

print('Number of duplicate ios apps:', len(duplicate_ios_apps))
print('\n')
print('Examples of duplicate ios apps:', duplicate_ios_apps[:10])
print('\n')
print('Number of unique ios apps:', len(unique_ios_apps))
```

```
Number of duplicate ios apps: 0


Examples of duplicate ios apps: []


Number of unique ios apps: 7197
```

From above, we can confirm that there are no duplicates with the App Store data.

Removing Non English - Speaking apps

All the characters that are specific to English texts are encoded using the ASCII standard. Each ASCII (American Standard Code for Information Interchange) system character has a corresponding number between 0 and 127 associated with it, and we can take advantage of that to build a function that checks an app name and tells us whether it contains non-ASCII characters.

In [11]:
```
def is_english(string):

    for character in string:
        if ord(character) > 127:
            return False
        else:
            return True

print(is_english('Instagram'))
print(is_english('爱奇'))
```

```
True
False
```

The function seems to work fine, but some English app names use emojis or other symbols (™, — (em dash), – (en dash), etc.) that fall outside of the ASCII range. Because of this, we'll remove useful apps if we use the function in its current form.

To minimize the impact of data loss, we'll only remove an app if its name has more than three non-ASCII characters:

In [12]:
```
def is_english(string):
    non_ascii = 0

    for character in string:
        if ord(character) > 127:
            non_ascii += 1

    if non_ascii > 3:
        return False
    else:
        return True

print(is_english('Docs To Go™ Free Office Suite'))
print(is_english('Instachat 😜'))
```

```
True
True
```

In [13]:
```
android_english = []
ios_english = []

for app in android_clean:
    name = app[0]
    if is_english(name):
        android_english.append(app)

for app in ios:
    name = app[1] # rep the track name
    if is_english(name):
        ios_english.append(app)

explore_data(android_english, 0, 3, True)
print('\n')
explore_data(ios_english, 0, 3, True)
```

```
['Photo Editor & Candy Camera & Grid & ScrapBook', 'ART_AND_DESIGN', '4.1',
'159', '19M', '10,000+', 'Free', '0', 'Everyone', 'Art & Design', 'January 7,
2018', '1.0.0', '4.0.3 and up']


['U Launcher Lite – FREE Live Cool Themes, Hide Apps', 'ART_AND_DESIGN', '4.
7', '87510', '8.7M', '5,000,000+', 'Free', '0', 'Everyone', 'Art & Design',
'August 1, 2018', '1.2.4', '4.0.3 and up']


['Sketch - Draw & Paint', 'ART_AND_DESIGN', '4.5', '215644', '25M', '50,000,0
00+', 'Free', '0', 'Teen', 'Art & Design', 'June 8, 2018', 'Varies with devic
e', '4.2 and up']


Number of rows: 9614
Number of columns: 13


['284882215', 'Facebook', '389879808', 'USD', '0.0', '2974676', '212', '3.5',
'3.5', '95.0', '4+', 'Social Networking', '37', '1', '29', '1']


['389801252', 'Instagram', '113954816', 'USD', '0.0', '2161558', '1289', '4.
5', '4.0', '10.23', '12+', 'Photo & Video', '37', '0', '29', '1']


['529479190', 'Clash of Clans', '116476928', 'USD', '0.0', '2130805', '579',
'4.5', '4.5', '9.24.12', '9+', 'Games', '38', '5', '18', '1']


Number of rows: 6183
Number of columns: 16
```

We are left with 9614 Android apps and 6183 iOS apps.

Free Apps

As mentioned in the introduction, we only build apps that are free to download and install, and our main source of revenue consists of in-app ads. Our data sets contain both free and non-free apps; we'll need to isolate only the free apps for our analysis.

In [14]:
```
android_final = []
ios_final = []

for app in android_english:
    price = app[7]
    if price == '0':
        android_final.append(app)

for app in ios_english:
    price = app[4]
    if price == '0.0':
        ios_final.append(app)

print(len(android_final))
print(len(ios_final))
```

```
8864
3222
```

We are left with 8864 Free Android apps and Free 3222 iOS apps for our analysis.

Most common apps by Genre

Our aim is to determine the kinds of apps that are likely to attract more users because our revenue is highly influenced by the number of people using our apps.

We begin our analysis by getting a sense of the most common genres for each market:

We will build two functions we can use to analyze the frequency tables:

1. One function to generate frequency tables that show percentages
2. Another function that we can use to display the percentages in a descending order

In [15]:
```
def freq_table(dataset, index):
    table = {}
    total = 0

    for row in dataset:
        total += 1
        value = row[index]
        if value in table:
            table[value] += 1
        else:
            table[value] = 1

    table_percentages = {}
    for key in table:
        percentage = (table[key] / total) * 100
        table_percentages[key] = percentage

    return table_percentages


def display_table(dataset, index):
    table = freq_table(dataset, index)
    table_display = []
    for key in table:
        key_val_as_tuple = (table[key], key)
        table_display.append(key_val_as_tuple)
        table_sorted = sorted(table_display, reverse = True)
    for entry in table_sorted:
        print(entry[1], ':', entry[0])
```

We examine the frequency table for the prime_genre column of the App Store data set.

In [16]:
```
display_table(ios_final, -5)
```

```
Games : 58.16263190564867
Entertainment : 7.883302296710118
Photo & Video : 4.9658597144630665
Education : 3.662321539416512
Social Networking : 3.2898820608317814
Shopping : 2.60707635009311
Utilities : 2.5139664804469275
Sports : 2.1415270018621975
Music : 2.0484171322160147
Health & Fitness : 2.0173805090006205
Productivity : 1.7380509000620732
Lifestyle : 1.5828677839851024
News : 1.3345747982619491
Travel : 1.2414649286157666
Finance : 1.1173184357541899
Weather : 0.8690254500310366
Food & Drink : 0.8069522036002483
Reference : 0.5586592178770949
Business : 0.5276225946617008
Book : 0.4345127250155183
Navigation : 0.186219739292365
Medical : 0.186219739292365
Catalogs : 0.12414649286157665
```

We can see that among the free English apps for iOS, more than a half (58.16%) are games. Entertainment apps are close to 8%, followed by photo and video apps, which are close to 5%. Only 3.66% of the apps are designed for education, followed by social networking apps which amount for 3.29% of the apps in our data set.

The general impression is that App Store (at least the part containing free English apps) is dominated by apps that are designed for fun (games, entertainment, photo and video, social networking, sports, music, etc.)

Let's continue by examining the Genres and Category columns of the Google Play data set (two columns which seem to be related).

In [17]:
```
display_table(android_final, 1) # Category
```

```
FAMILY : 18.907942238267147
GAME : 9.724729241877256
TOOLS : 8.461191335740072
BUSINESS : 4.591606498194946
LIFESTYLE : 3.9034296028880866
PRODUCTIVITY : 3.892148014440433
FINANCE : 3.7003610108303246
MEDICAL : 3.531137184115524
SPORTS : 3.395758122743682
PERSONALIZATION : 3.3167870036101084
COMMUNICATION : 3.2378158844765346
HEALTH_AND_FITNESS : 3.0798736462093865
PHOTOGRAPHY : 2.944494584837545
NEWS_AND_MAGAZINES : 2.7978339350180503
SOCIAL : 2.6624548736462095
TRAVEL_AND_LOCAL : 2.33528880866426
SHOPPING : 2.2450361010830324
BOOKS_AND_REFERENCE : 2.1435018050541514
DATING : 1.861462093862816
VIDEO_PLAYERS : 1.7937725631768955
MAPS_AND_NAVIGATION : 1.3989169675090252
FOOD_AND_DRINK : 1.2409747292418771
EDUCATION : 1.1620036101083033
ENTERTAINMENT : 0.9589350180505415
LIBRARIES_AND_DEMO : 0.9363718411552346
AUTO_AND_VEHICLES : 0.9250902527075812
HOUSE_AND_HOME : 0.8235559566787004
WEATHER : 0.8009927797833934
EVENTS : 0.7107400722021661
PARENTING : 0.6543321299638989
ART_AND_DESIGN : 0.6430505415162455
COMICS : 0.6204873646209386
BEAUTY : 0.5979241877256317
```

Google Play: there are not that many apps designed for fun, and it seems that a good number of apps are designed for practical purposes (family, tools, business, lifestyle, productivity, etc.).

However, the family category (which accounts for almost 19% of the apps) may means mostly games and entertainment for kids.

Even so, practical apps seem to have a better representation on Google Play compared to App Store. This is also confirmed by the frequency table we see for the Genres column:

In [18]:
```
display_table(android_final, -4)
```

```
Tools : 8.449909747292418
Entertainment : 6.069494584837545
Education : 5.347472924187725
Business : 4.591606498194946
Productivity : 3.892148014440433
Lifestyle : 3.892148014440433
Finance : 3.7003610108303246
Medical : 3.531137184115524
Sports : 3.463447653429603
Personalization : 3.3167870036101084
Communication : 3.2378158844765346
Action : 3.1024368231046933
Health & Fitness : 3.0798736462093865
Photography : 2.944494584837545
News & Magazines : 2.7978339350180503
Social : 2.6624548736462095
Travel & Local : 2.3240072202166067
Shopping : 2.2450361010830324
Books & Reference : 2.1435018050541514
Simulation : 2.0419675090252705
Dating : 1.861462093862816
Arcade : 1.8501805054151623
Video Players & Editors : 1.7712093862815883
Casual : 1.7599277978339352
Maps & Navigation : 1.3989169675090252
Food & Drink : 1.2409747292418771
Puzzle : 1.128158844765343
Racing : 0.9927797833935018
Role Playing : 0.9363718411552346
Libraries & Demo : 0.9363718411552346
Auto & Vehicles : 0.9250902527075812
Strategy : 0.9138086642599278
House & Home : 0.8235559566787004
Weather : 0.8009927797833934
Events : 0.7107400722021661
Adventure : 0.6768953068592057
Comics : 0.6092057761732852
Beauty : 0.5979241877256317
Art & Design : 0.5979241877256317
Parenting : 0.4963898916967509
Card : 0.45126353790613716
Casino : 0.42870036101083037
Trivia : 0.41741877256317694
Educational;Education : 0.39485559566787
Board : 0.3835740072202166
Educational : 0.3722924187725632
Education;Education : 0.33844765342960287
Word : 0.2594765342960289
Casual;Pretend Play : 0.236913357400722
Music : 0.2030685920577617
Racing;Action & Adventure : 0.16922382671480143
Puzzle;Brain Games : 0.16922382671480143
Entertainment;Music & Video : 0.16922382671480143
Casual;Brain Games : 0.13537906137184114
Casual;Action & Adventure : 0.13537906137184114
Arcade;Action & Adventure : 0.12409747292418773
Action;Action & Adventure : 0.10153429602888085
```

```
Educational;Pretend Play : 0.09025270758122744
Simulation;Action & Adventure : 0.078971119133574
Parenting;Education : 0.078971119133574
Entertainment;Brain Games : 0.078971119133574
Board;Brain Games : 0.078971119133574
Parenting;Music & Video : 0.06768953068592057
Educational;Brain Games : 0.06768953068592057
Casual;Creativity : 0.06768953068592057
Art & Design;Creativity : 0.06768953068592057
Education;Pretend Play : 0.056407942238267145
Role Playing;Pretend Play : 0.04512635379061372
Education;Creativity : 0.04512635379061372
Role Playing;Action & Adventure : 0.033844765342960284
Puzzle;Action & Adventure : 0.033844765342960284
Entertainment;Creativity : 0.033844765342960284
Entertainment;Action & Adventure : 0.033844765342960284
Educational;Creativity : 0.033844765342960284
Educational;Action & Adventure : 0.033844765342960284
Education;Music & Video : 0.033844765342960284
Education;Brain Games : 0.033844765342960284
Education;Action & Adventure : 0.033844765342960284
Adventure;Action & Adventure : 0.033844765342960284
Video Players & Editors;Music & Video : 0.02256317689530686
Sports;Action & Adventure : 0.02256317689530686
Simulation;Pretend Play : 0.02256317689530686
Puzzle;Creativity : 0.02256317689530686
Music;Music & Video : 0.02256317689530686
Entertainment;Pretend Play : 0.02256317689530686
Casual;Education : 0.02256317689530686
Board;Action & Adventure : 0.02256317689530686
Video Players & Editors;Creativity : 0.01128158844765343
Trivia;Education : 0.01128158844765343
Travel & Local;Action & Adventure : 0.01128158844765343
Tools;Education : 0.01128158844765343
Strategy;Education : 0.01128158844765343
Strategy;Creativity : 0.01128158844765343
Strategy;Action & Adventure : 0.01128158844765343
Simulation;Education : 0.01128158844765343
Role Playing;Brain Games : 0.01128158844765343
Racing;Pretend Play : 0.01128158844765343
Puzzle;Education : 0.01128158844765343
Parenting;Brain Games : 0.01128158844765343
Music & Audio;Music & Video : 0.01128158844765343
Lifestyle;Pretend Play : 0.01128158844765343
Lifestyle;Education : 0.01128158844765343
Health & Fitness;Education : 0.01128158844765343
Health & Fitness;Action & Adventure : 0.01128158844765343
Entertainment;Education : 0.01128158844765343
Communication;Creativity : 0.01128158844765343
Comics;Creativity : 0.01128158844765343
Casual;Music & Video : 0.01128158844765343
Card;Action & Adventure : 0.01128158844765343
Books & Reference;Education : 0.01128158844765343
Art & Design;Pretend Play : 0.01128158844765343
Art & Design;Action & Adventure : 0.01128158844765343
Arcade;Pretend Play : 0.01128158844765343
Adventure;Education : 0.01128158844765343
```

We will proceed using the Category value to analyse the Android data

Most Popular Apps by Genre on the App Store

One way to find out what genres are the most popular (have the most users) is to calculate the average number of installs for each app genre. For the Google Play data set, we can find this information in the Installs column, but for the App Store data set this information is missing. As a workaround, we'll take the total number of user ratings as a proxy, which we can find in the rating_count_tot app.

Below, we calculate the average number of user ratings per app genre on the App Store:

In [19]:
```
genres_ios = freq_table(ios_final, -5)

for genre in genres_ios:
    total = 0
    len_genre = 0
    for app in ios_final:
        genre_app = app[-5]
        if genre_app == genre:
            n_ratings = float(app[5])
            total += n_ratings
            len_genre += 1
    avg_n_ratings = total / len_genre
    print(genre, ':', avg_n_ratings)
```

```
News : 21248.023255813954
Navigation : 86090.33333333333
Food & Drink : 33333.92307692308
Shopping : 26919.690476190477
Social Networking : 71548.34905660378
Games : 22788.6696905016
Travel : 28243.8
Sports : 23008.898550724636
Entertainment : 14029.830708661417
Photo & Video : 28441.54375
Utilities : 18684.456790123455
Weather : 52279.892857142855
Business : 7491.117647058823
Medical : 612.0
Book : 39758.5
Education : 7003.983050847458
Health & Fitness : 23298.015384615384
Lifestyle : 16485.764705882353
Reference : 74942.11111111111
Music : 57326.530303030304
Finance : 31467.944444444445
Catalogs : 4004.0
Productivity : 21028.410714285714
```

On average, navigation apps have the highest number of user reviews, but this figure is heavily influenced by Waze and Google Maps, which have close to half a million user reviews together:

In [20]:
```
for app in ios_final:
    if app[-5] == 'Navigation':
        print(app[1], ':', app[5]) # print name and number of ratings
```

```
Waze - GPS Navigation, Maps & Real-time Traffic : 345046
Google Maps - Navigation & Transit : 154911
Geocaching® : 12811
CoPilot GPS – Car Navigation & Offline Maps : 3582
ImmobilienScout24: Real Estate Search in Germany : 187
Railway Route Search : 5
```

The same pattern applies to social networking apps, where the average number is heavily influenced by a few giants like Facebook, Pinterest, Skype, etc. Same applies to music apps, where a few big players like Pandora, Spotify, and Shazam heavily influence the average number.

Our aim is to find popular genres, but navigation, social networking or music apps might seem more popular than they really are. The average number of ratings seem to be skewed by very few apps which have hundreds of thousands of user ratings, while the other apps may struggle to get past the 10,000 threshold. We could get a better picture by removing these extremely popular apps for each genre and then rework the averages, but we'll leave this level of detail for later.

Reference apps have 74,942 user ratings on average, but it's actually the Bible and Dictionary.com which skew up the average rating:

In [21]:
```
for app in ios_final:
    if app[-5] == 'Reference':
        print(app[1], ':', app[5])
```

```
Bible : 985920
Dictionary.com Dictionary & Thesaurus : 200047
Dictionary.com Dictionary & Thesaurus for iPad : 54175
Google Translate : 26786
Muslim Pro: Ramadan 2017 Prayer Times, Azan, Quran : 18418
New Furniture Mods - Pocket Wiki & Game Tools for Minecraft PC Edition : 1758
8
Merriam-Webster Dictionary : 16849
Night Sky : 12122
City Maps for Minecraft PE - The Best Maps for Minecraft Pocket Edition (MCP
E) : 8535
LUCKY BLOCK MOD ™ for Minecraft PC Edition - The Best Pocket Wiki & Mods Inst
aller Tools : 4693
GUNS MODS for Minecraft PC Edition - Mods Tools : 1497
Guides for Pokémon GO - Pokemon GO News and Cheats : 826
WWDC : 762
Horror Maps for Minecraft PE - Download The Scariest Maps for Minecraft Pocke
t Edition (MCPE) Free : 718
VPN Express : 14
Real Bike Traffic Rider Virtual Reality Glasses : 8
教えて!goo : 0
Jishokun-Japanese English Dictionary & Translator : 0
```

However, this niche seems to show some potential. One thing we could do is take another popular book and turn it into an app where we could add different features besides the raw version of the book. This might include daily quotes from the book, an audio version of the book, quizzes about the book, etc. On top of that, we could also embed a dictionary within the app, so users don't need to exit our app to look up words in an external app.

Now let's analyze the Google Play market a bit.

For the Google Play market, we actually have data about the number of installs, so we should be able to get a clearer picture about genre popularity. However, the install numbers don't seem precise enough — we can see that most values are open-ended (100+, 1,000+, 5,000+, etc.):

In [22]:
```
display_table(android_final, 5) # the Installs columns
```

```
1,000,000+ : 15.726534296028879
100,000+ : 11.552346570397113
10,000,000+ : 10.548285198555957
10,000+ : 10.198555956678701
1,000+ : 8.393501805054152
100+ : 6.915613718411552
5,000,000+ : 6.825361010830325
500,000+ : 5.561823104693141
50,000+ : 4.7721119133574
5,000+ : 4.512635379061372
10+ : 3.5424187725631766
500+ : 3.2490974729241873
50,000,000+ : 2.3014440433213
100,000,000+ : 2.1322202166064983
50+ : 1.917870036101083
5+ : 0.78971119133574
1+ : 0.5076714801444043
500,000,000+ : 0.2707581227436823
1,000,000,000+ : 0.22563176895306858
0+ : 0.04512635379061372
0 : 0.01128158844765343
```

One problem with this data is that is not precise. For instance, we don't know whether an app with 100,000+ installs has 100,000 installs, 200,000, or 350,000. However, we don't need very precise data for our purposes — we only want to get an idea which app genres attract the most users, and we don't need perfect precision with respect to the number of users.

We're going to leave the numbers as they are, which means that we'll consider that an app with 100,000+ installs has 100,000 installs, and an app with 1,000,000+ installs has 1,000,000 installs, and so on.

In [23]:
```
categories_android = freq_table(android_final, 1)

for category in categories_android:
    total = 0
    len_category = 0
    for app in android_final:
        category_app = app[1]
        if category_app == category:
            n_installs = app[5]
            n_installs = n_installs.replace(',', '')
            n_installs = n_installs.replace('+', '')
            total += float(n_installs)
            len_category += 1
    avg_n_installs = total / len_category
    print(category, ':', avg_n_installs)
```

```
EVENTS : 253542.22222222222
SPORTS : 3638640.1428571427
TOOLS : 10801391.298666667
EDUCATION : 1833495.145631068
AUTO_AND_VEHICLES : 647317.8170731707
COMICS : 817657.2727272727
COMMUNICATION : 38456119.167247385
HEALTH_AND_FITNESS : 4188821.9853479853
HOUSE_AND_HOME : 1331540.5616438356
FINANCE : 1387692.475609756
BUSINESS : 1712290.1474201474
GAME : 15588015.603248259
PHOTOGRAPHY : 17840110.40229885
BOOKS_AND_REFERENCE : 8767811.894736841
MAPS_AND_NAVIGATION : 4056941.7741935486
LIBRARIES_AND_DEMO : 638503.734939759
SHOPPING : 7036877.311557789
TRAVEL_AND_LOCAL : 13984077.710144928
MEDICAL : 120550.61980830671
FOOD_AND_DRINK : 1924897.7363636363
DATING : 854028.8303030303
ART_AND_DESIGN : 1986335.0877192982
PRODUCTIVITY : 16787331.344927534
FAMILY : 3695641.8198090694
BEAUTY : 513151.88679245283
VIDEO_PLAYERS : 24727872.452830188
PARENTING : 542603.6206896552
NEWS_AND_MAGAZINES : 9549178.467741935
PERSONALIZATION : 5201482.6122448975
LIFESTYLE : 1437816.2687861272
WEATHER : 5074486.197183099
SOCIAL : 23253652.127118643
ENTERTAINMENT : 11640705.88235294
```

On average, communication apps have the most installs: 38,456,119. This number is heavily skewed up by a few apps that have over one billion installs (WhatsApp, Facebook Messenger, Skype, Google Chrome, Gmail, and Hangouts), and a few others with over 100 and 500 million installs:

In [24]:
```
for app in android_final:
    if app[1] == 'COMMUNICATION' and (app[5] == '1,000,000,000+'
                                      or app[5] == '500,000,000+'
                                      or app[5] == '100,000,000+'):
        print(app[0], ':', app[5])
```

```
WhatsApp Messenger : 1,000,000,000+
imo beta free calls and text : 100,000,000+
Android Messages : 100,000,000+
Google Duo - High Quality Video Calls : 500,000,000+
Messenger – Text and Video Chat for Free : 1,000,000,000+
imo free video calls and chat : 500,000,000+
Skype - free IM & video calls : 1,000,000,000+
Who : 100,000,000+
GO SMS Pro - Messenger, Free Themes, Emoji : 100,000,000+
LINE: Free Calls & Messages : 500,000,000+
Google Chrome: Fast & Secure : 1,000,000,000+
Firefox Browser fast & private : 100,000,000+
UC Browser - Fast Download Private & Secure : 500,000,000+
Gmail : 1,000,000,000+
Hangouts : 1,000,000,000+
Messenger Lite: Free Calls & Messages : 100,000,000+
Kik : 100,000,000+
KakaoTalk: Free Calls & Text : 100,000,000+
Opera Mini - fast web browser : 100,000,000+
Opera Browser: Fast and Secure : 100,000,000+
Telegram : 100,000,000+
Truecaller: Caller ID, SMS spam blocking & Dialer : 100,000,000+
UC Browser Mini -Tiny Fast Private & Secure : 100,000,000+
Viber Messenger : 500,000,000+
WeChat : 100,000,000+
Yahoo Mail – Stay Organized : 100,000,000+
BBM - Free Calls & Messages : 100,000,000+
```

If we removed all the communication apps that have over 100 million installs, the average would be reduced roughly ten times:

In [25]:
```
under_100_m = []

for app in android_final:
    n_installs = app[5]
    n_installs = n_installs.replace(',', '')
    n_installs = n_installs.replace('+', '')
    if (app[1] == 'COMMUNICATION') and (float(n_installs) < 100000000):
        under_100_m.append(float(n_installs))

sum(under_100_m) / len(under_100_m)
```

Out[25]:
```
3603485.3884615386
```

We see the same pattern for the video players category, which is the runner-up with 24,727,872 installs. The market is dominated by apps like Youtube, Google Play Movies & TV, or MX Player. The pattern is repeated for social apps (where we have giants like Facebook, Instagram, Google+, etc.), photography apps (Google Photos and other popular photo editors), or productivity apps (Microsoft Word, Dropbox, Google Calendar, Evernote, etc.).

Again, the main concern is that these app genres might seem more popular than they really are. Moreover, these niches seem to be dominated by a few giants who are hard to compete against.

The books and reference genre looks fairly popular as well, with an average number of installs of 8,767,811. It's interesting to explore this in more depth, since we found this genre has some potential to work well on the App Store, and our aim is to recommend an app genre that shows potential for being profitable on both the App Store and Google Play.

Let's take a look at some of the apps from this genre and their number of installs:

In [27]:
```
for app in android_final:
    if app[1] == 'BOOKS_AND_REFERENCE':
        print(app[0], ':', app[5])
```

```
E-Book Read - Read Book for free : 50,000+
Download free book with green book : 100,000+
Wikipedia : 10,000,000+
Cool Reader : 10,000,000+
Free Panda Radio Music : 100,000+
Book store : 1,000,000+
FBReader: Favorite Book Reader : 10,000,000+
English Grammar Complete Handbook : 500,000+
Free Books - Spirit Fanfiction and Stories : 1,000,000+
Google Play Books : 1,000,000,000+
AlReader -any text book reader : 5,000,000+
Offline English Dictionary : 100,000+
Offline: English to Tagalog Dictionary : 500,000+
FamilySearch Tree : 1,000,000+
Cloud of Books : 1,000,000+
Recipes of Prophetic Medicine for free : 500,000+
ReadEra – free ebook reader : 1,000,000+
Anonymous caller detection : 10,000+
Ebook Reader : 5,000,000+
Litnet - E-books : 100,000+
Read books online : 5,000,000+
English to Urdu Dictionary : 500,000+
eBoox: book reader fb2 epub zip : 1,000,000+
English Persian Dictionary : 500,000+
Flybook : 500,000+
All Maths Formulas : 1,000,000+
Ancestry : 5,000,000+
HTC Help : 10,000,000+
English translation from Bengali : 100,000+
Pdf Book Download - Read Pdf Book : 100,000+
Free Book Reader : 100,000+
eBoox new: Reader for fb2 epub zip books : 50,000+
Only 30 days in English, the guideline is guaranteed : 500,000+
Moon+ Reader : 10,000,000+
SH-02J Owner's Manual (Android 8.0) : 50,000+
English-Myanmar Dictionary : 1,000,000+
Golden Dictionary (EN-AR) : 1,000,000+
All Language Translator Free : 1,000,000+
Azpen eReader : 500,000+
URBANO V 02 instruction manual : 100,000+
Bible : 100,000,000+
C Programs and Reference : 50,000+
C Offline Tutorial : 1,000+
C Programs Handbook : 50,000+
Amazon Kindle : 100,000,000+
Aab e Hayat Full Novel : 100,000+
Aldiko Book Reader : 10,000,000+
Google I/O 2018 : 500,000+
R Language Reference Guide : 10,000+
Learn R Programming Full : 5,000+
R Programing Offline Tutorial : 1,000+
Guide for R Programming : 5+
Learn R Programming : 10+
R Quick Reference Big Data : 1,000+
V Made : 100,000+
Wattpad 📖 Free Books : 100,000,000+
Dictionary - WordWeb : 5,000,000+
```

```
Guide (for X-MEN) : 100,000+
AC Air condition Troubleshoot,Repair,Maintenance : 5,000+
AE Bulletins : 1,000+
Ae Allah na Dai (Rasa) : 10,000+
50000 Free eBooks & Free AudioBooks : 5,000,000+
Ag PhD Field Guide : 10,000+
Ag PhD Deficiencies : 10,000+
Ag PhD Planting Population Calculator : 1,000+
Ag PhD Soybean Diseases : 1,000+
Fertilizer Removal By Crop : 50,000+
A-J Media Vault : 50+
Al-Quran (Free) : 10,000,000+
Al Quran (Tafsir & by Word) : 500,000+
Al Quran Indonesia : 10,000,000+
Al'Quran Bahasa Indonesia : 10,000,000+
Al Quran Al karim : 1,000,000+
Al-Muhaffiz : 50,000+
Al Quran : EAlim - Translations & MP3 Offline : 5,000,000+
Al-Quran 30 Juz free copies : 500,000+
Koran Read &MP3 30 Juz Offline : 1,000,000+
Hafizi Quran 15 lines per page : 1,000,000+
Quran for Android : 10,000,000+
Surah Al-Waqiah : 100,000+
Hisnul Al Muslim - Hisn Invocations & Adhkaar : 100,000+
Satellite AR : 1,000,000+
Audiobooks from Audible : 100,000,000+
Kinot & Eichah for Tisha B'Av : 10,000+
AW Tozer Devotionals - Daily : 5,000+
Tozer Devotional -Series 1 : 1,000+
The Pursuit of God : 1,000+
AY Sing : 5,000+
Ay Hasnain k Nana Milad Naat : 10,000+
Ay Mohabbat Teri Khatir Novel : 10,000+
Arizona Statutes, ARS (AZ Law) : 1,000+
Oxford A-Z of English Usage : 1,000,000+
BD Fishpedia : 1,000+
BD All Sim Offer : 10,000+
Youboox - Livres, BD et magazines : 500,000+
B&H Kids AR : 10,000+
B y H Niños ES : 5,000+
Dictionary.com: Find Definitions for English Words : 10,000,000+
English Dictionary - Offline : 10,000,000+
Bible KJV : 5,000,000+
Borneo Bible, BM Bible : 10,000+
MOD Black for BM : 100+
BM Box : 1,000+
Anime Mod for BM : 100+
NOOK: Read eBooks & Magazines : 10,000,000+
NOOK Audiobooks : 500,000+
NOOK App for NOOK Devices : 500,000+
Browsery by Barnes & Noble : 5,000+
bp e-store : 1,000+
Brilliant Quotes: Life, Love, Family & Motivation : 1,000,000+
BR Ambedkar Biography & Quotes : 10,000+
BU Alsace : 100+
Catholic La Bu Zo Kam : 500+
Khrifa Hla Bu (Solfa) : 10+
```

```
Kristian Hla Bu : 10,000+
SA HLA BU : 1,000+
Learn SAP BW : 500+
Learn SAP BW on HANA : 500+
CA Laws 2018 (California Laws and Codes) : 5,000+
Bootable Methods(USB-CD-DVD) : 10,000+
cloudLibrary : 100,000+
SDA Collegiate Quarterly : 500+
Sabbath School : 100,000+
Cypress College Library : 100+
Stats Royale for Clash Royale : 1,000,000+
GATE 21 years CS Papers(2011-2018 Solved) : 50+
Learn CT Scan Of Head : 5,000+
Easy Cv maker 2018 : 10,000+
How to Write CV : 100,000+
CW Nuclear : 1,000+
CY Spray nozzle : 10+
BibleRead En Cy Zh Yue : 5+
CZ-Help : 5+
Modlitební knížka CZ : 500+
Guide for DB Xenoverse : 10,000+
Guide for DB Xenoverse 2 : 10,000+
Guide for IMS DB : 10+
DC HSEMA : 5,000+
DC Public Library : 1,000+
Painting Lulu DC Super Friends : 1,000+
Dictionary : 10,000,000+
Fix Error Google Playstore : 1,000+
D. H. Lawrence Poems FREE : 1,000+
Bilingual Dictionary Audio App : 5,000+
DM Screen : 10,000+
wikiHow: how to do anything : 1,000,000+
Dr. Doug's Tips : 1,000+
Bible du Semeur-BDS (French) : 50,000+
La citadelle du musulman : 50,000+
DV 2019 Entry Guide : 10,000+
DV 2019 - EDV Photo & Form : 50,000+
DV 2018 Winners Guide : 1,000+
EB Annual Meetings : 1,000+
EC - AP & Telangana : 5,000+
TN Patta Citta & EC : 10,000+
AP Stamps and Registration : 10,000+
CompactiMa EC pH Calibration : 100+
EGW Writings 2 : 100,000+
EGW Writings : 1,000,000+
Bible with EGW Comments : 100,000+
My Little Pony AR Guide : 1,000,000+
SDA Sabbath School Quarterly : 500,000+
Duaa Ek Ibaadat : 5,000+
Spanish English Translator : 10,000,000+
Dictionary - Merriam-Webster : 10,000,000+
JW Library : 10,000,000+
Oxford Dictionary of English : Free : 10,000,000+
English Hindi Dictionary : 10,000,000+
English to Hindi Dictionary : 5,000,000+
EP Research Service : 1,000+
Hymnes et Louanges : 100,000+
```

```
EU Charter : 1,000+
EU Data Protection : 1,000+
EU IP Codes : 100+
EW PDF : 5+
BakaReader EX : 100,000+
EZ Quran : 50,000+
FA Part 1 & 2 Past Papers Solved Free – Offline : 5,000+
La Fe de Jesus : 1,000+
La Fe de Jesús : 500+
Le Fe de Jesus : 500+
Florida - Pocket Brainbook : 1,000+
Florida Statutes (FL Code) : 1,000+
English To Shona Dictionary : 10,000+
Greek Bible FP (Audio) : 1,000+
Golden Dictionary (FR-AR) : 500,000+
Fanfic-FR : 5,000+
Bulgarian French Dictionary Fr : 10,000+
Chemin (fr) : 1,000+
The SCP Foundation DB fr nn5n : 1,000+
```

The book and reference genre includes a variety of apps: software for processing and reading ebooks, various collections of libraries, dictionaries, tutorials on programming or languages, etc. It seems there's still a small number of extremely popular apps that skew the average:

In [28]:
```
for app in android_final:
    if app[1] == 'BOOKS_AND_REFERENCE' and (app[5] == '1,000,000,000+'
                                            or app[5] == '500,000,000+'
                                            or app[5] == '100,000,000+'):
        print(app[0], ':', app[5])
```

```
Google Play Books : 1,000,000,000+
Bible : 100,000,000+
Amazon Kindle : 100,000,000+
Wattpad 📖 Free Books : 100,000,000+
Audiobooks from Audible : 100,000,000+
```

However, it looks like there are only a few very popular apps, so this market still shows potential. Let's try to get some app ideas based on the kind of apps that are somewhere in the middle in terms of popularity (between 1,000,000 and 100,000,000 downloads):

In [29]:
```
for app in android_final:
    if app[1] == 'BOOKS_AND_REFERENCE' and (app[5] == '1,000,000+'
                                            or app[5] == '5,000,000+'
                                            or app[5] == '10,000,000+'
                                            or app[5] == '50,000,000+'):
        print(app[0], ':', app[5])
```

```
Wikipedia : 10,000,000+
Cool Reader : 10,000,000+
Book store : 1,000,000+
FBReader: Favorite Book Reader : 10,000,000+
Free Books - Spirit Fanfiction and Stories : 1,000,000+
AlReader -any text book reader : 5,000,000+
FamilySearch Tree : 1,000,000+
Cloud of Books : 1,000,000+
ReadEra – free ebook reader : 1,000,000+
Ebook Reader : 5,000,000+
Read books online : 5,000,000+
eBoox: book reader fb2 epub zip : 1,000,000+
All Maths Formulas : 1,000,000+
Ancestry : 5,000,000+
HTC Help : 10,000,000+
Moon+ Reader : 10,000,000+
English-Myanmar Dictionary : 1,000,000+
Golden Dictionary (EN-AR) : 1,000,000+
All Language Translator Free : 1,000,000+
Aldiko Book Reader : 10,000,000+
Dictionary - WordWeb : 5,000,000+
50000 Free eBooks & Free AudioBooks : 5,000,000+
Al-Quran (Free) : 10,000,000+
Al Quran Indonesia : 10,000,000+
Al'Quran Bahasa Indonesia : 10,000,000+
Al Quran Al karim : 1,000,000+
Al Quran : EAlim - Translations & MP3 Offline : 5,000,000+
Koran Read &MP3 30 Juz Offline : 1,000,000+
Hafizi Quran 15 lines per page : 1,000,000+
Quran for Android : 10,000,000+
Satellite AR : 1,000,000+
Oxford A-Z of English Usage : 1,000,000+
Dictionary.com: Find Definitions for English Words : 10,000,000+
English Dictionary - Offline : 10,000,000+
Bible KJV : 5,000,000+
NOOK: Read eBooks & Magazines : 10,000,000+
Brilliant Quotes: Life, Love, Family & Motivation : 1,000,000+
Stats Royale for Clash Royale : 1,000,000+
Dictionary : 10,000,000+
wikiHow: how to do anything : 1,000,000+
EGW Writings : 1,000,000+
My Little Pony AR Guide : 1,000,000+
Spanish English Translator : 10,000,000+
Dictionary - Merriam-Webster : 10,000,000+
JW Library : 10,000,000+
Oxford Dictionary of English : Free : 10,000,000+
English Hindi Dictionary : 10,000,000+
English to Hindi Dictionary : 5,000,000+
```

This niche seems to be dominated by software for processing and reading ebooks, as well as various collections of libraries and dictionaries, so it's probably not a good idea to build similar apps since there'll be some significant competition.

We also notice there are quite a few apps built around the book Quran, which suggests that building an app around a popular book can be profitable. It seems that taking a popular book (perhaps a more recent book) and turning it into an app could be profitable for both the Google Play and the App Store markets.

However, it looks like the market is already full of libraries, so we need to add some special features besides the raw version of the book. This might include daily quotes from the book, an audio version of the book, quizzes on the book, a forum where people can discuss the book, etc.

Conclusions

In this project, we analyzed data about the App Store and Google Play mobile apps with the goal of recommending an app profile that can be profitable for both markets.

We concluded that taking a popular book (perhaps a more recent book) and turning it into an app could be profitable for both the Google Play and the App Store markets. The markets are already full of libraries, so we need to add some special features besides the raw version of the book. This might include daily quotes from the book, an audio version of the book, quizzes on the book, a forum where people can discuss the book, etc.