

Google Apps - Datacamp Project

February 18, 2022

0.1 1. Google Play Store apps and reviews

Mobile apps are everywhere. They are easy to create and can be lucrative. Because of these two factors, more and more apps are being developed. In this notebook, we will do a comprehensive analysis of the Android app market by comparing over ten thousand apps in Google Play across different categories. We'll look for insights in the data to devise strategies to drive growth and retention.

Let's take a look at the data, which consists of two files:

apps.csv: contains all the details of the applications on Google Play. There are 13 features that describe a given app.

user_reviews.csv: contains 100 reviews for each app, most helpful first. The text in each review has been pre-processed and attributed with three new features: Sentiment (Positive, Negative or Neutral), Sentiment Polarity and Sentiment Subjectivity.

```
[657]: # Read in dataset
import pandas as pd
apps_with_duplicates = pd.read_csv("datasets/apps.csv")

# Drop duplicates from apps_with_duplicates
apps = apps_with_duplicates.drop_duplicates()

# Print the total number of apps
print('Total number of apps in the dataset = ', apps)

# Have a look at a random sample of 5 rows
print(apps.head())
```

```
Total number of apps in the dataset =          Unnamed: 0
App \
0          0      Photo Editor & Candy Camera & Grid & ScrapBook
1          1                               Coloring book moana
2          2  U Launcher Lite - FREE Live Cool Themes, Hide ...
3          3                               Sketch - Draw & Paint
4          4          Pixel Draw - Number Art Coloring Book
5          5                               Paper flowers instructions
6          6      Smoke Effect Photo Maker - Smoke Editor
7          7                               Infinite Painter
8          8                               Garden Coloring Book
```

9	9	Kids Paint Free - Drawing Fun
10	10	Text on Photo - Fontee
11	11	Name Art Photo Editor - Focus n Filters
12	12	Tattoo Name On My Photo Editor
13	13	Mandala Coloring Book
14	14	3D Color Pixel by Number - Sandbox Art Coloring
15	15	Learn To Draw Kawaii Characters
16	16	Photo Designer - Write your name with shapes
17	17	350 Diy Room Decor Ideas
18	18	FlipaClip - Cartoon animation
19	19	ibis Paint X
20	20	Logo Maker - Small Business
21	21	Boys Photo Editor - Six Pack & Men's Suit
22	22	Superheroes Wallpapers 4K Backgrounds
23	23	Mcqueen Coloring pages
24	24	HD Mickey Minnie Wallpapers
25	25	Harley Quinn wallpapers HD
26	26	Colorfit - Drawing & Coloring
27	27	Animated Photo Editor
28	28	Pencil Sketch Drawing
29	29	Easy Realistic Drawing Tutorial
...
9629	10811	FR Plus 1.6
9630	10812	Fr Agnel Pune
9631	10813	DICT.fr Mobile
9632	10814	FR: My Secret Pets!
9633	10815	Golden Dictionary (FR-AR)
9634	10816	FieldBi FR Offline
9635	10817	HTC Sense Input - FR
9636	10818	Gold Quote - Gold.fr
9637	10819	Fanfic-FR
9638	10820	Fr. Daoud Lamei
9639	10821	Poop FR
9640	10822	PLMGSS FR
9641	10823	List iptv FR
9642	10824	Cardio-FR
9643	10825	Naruto & Boruto FR
9644	10826	Frim: get new friends on local chat rooms
9645	10827	Fr Agnel Ambarnath
9646	10828	Manga-FR - Anime Vostfr
9647	10829	Bulgarian French Dictionary Fr
9648	10830	News Minecraft.fr
9649	10831	payermonstationnement.fr
9650	10832	FR Tides
9651	10833	Chemin (fr)
9652	10834	FR Calculator
9653	10835	FR Forms
9654	10836	Sya9a Maroc - FR

9655	10837	Fr. Mike Schmitz Audio Teachings
9656	10838	Parkinson Exercices FR
9657	10839	The SCP Foundation DB fr nn5n
9658	10840	iHoroscope - 2018 Daily Horoscope & Astrology

	Category	Rating	Reviews	Size	Installs	Type	Price	\
0	ART_AND_DESIGN	4.1	159	19.0	10,000+	Free	0	
1	ART_AND_DESIGN	3.9	967	14.0	500,000+	Free	0	
2	ART_AND_DESIGN	4.7	87510	8.7	5,000,000+	Free	0	
3	ART_AND_DESIGN	4.5	215644	25.0	50,000,000+	Free	0	
4	ART_AND_DESIGN	4.3	967	2.8	100,000+	Free	0	
5	ART_AND_DESIGN	4.4	167	5.6	50,000+	Free	0	
6	ART_AND_DESIGN	3.8	178	19.0	50,000+	Free	0	
7	ART_AND_DESIGN	4.1	36815	29.0	1,000,000+	Free	0	
8	ART_AND_DESIGN	4.4	13791	33.0	1,000,000+	Free	0	
9	ART_AND_DESIGN	4.7	121	3.1	10,000+	Free	0	
10	ART_AND_DESIGN	4.4	13880	28.0	1,000,000+	Free	0	
11	ART_AND_DESIGN	4.4	8788	12.0	1,000,000+	Free	0	
12	ART_AND_DESIGN	4.2	44829	20.0	10,000,000+	Free	0	
13	ART_AND_DESIGN	4.6	4326	21.0	100,000+	Free	0	
14	ART_AND_DESIGN	4.4	1518	37.0	100,000+	Free	0	
15	ART_AND_DESIGN	3.2	55	2.7	5,000+	Free	0	
16	ART_AND_DESIGN	4.7	3632	5.5	500,000+	Free	0	
17	ART_AND_DESIGN	4.5	27	17.0	10,000+	Free	0	
18	ART_AND_DESIGN	4.3	194216	39.0	5,000,000+	Free	0	
19	ART_AND_DESIGN	4.6	224399	31.0	10,000,000+	Free	0	
20	ART_AND_DESIGN	4.0	450	14.0	100,000+	Free	0	
21	ART_AND_DESIGN	4.1	654	12.0	100,000+	Free	0	
22	ART_AND_DESIGN	4.7	7699	4.2	500,000+	Free	0	
23	ART_AND_DESIGN	NaN	61	7.0	100,000+	Free	0	
24	ART_AND_DESIGN	4.7	118	23.0	50,000+	Free	0	
25	ART_AND_DESIGN	4.8	192	6.0	10,000+	Free	0	
26	ART_AND_DESIGN	4.7	20260	25.0	500,000+	Free	0	
27	ART_AND_DESIGN	4.1	203	6.1	100,000+	Free	0	
28	ART_AND_DESIGN	3.9	136	4.6	10,000+	Free	0	
29	ART_AND_DESIGN	4.1	223	4.2	100,000+	Free	0	
...	
9629	AUTO_AND_VEHICLES	NaN	4	3.9	100+	Free	0	
9630	FAMILY	4.1	80	13.0	1,000+	Free	0	
9631	BUSINESS	NaN	20	2.7	10,000+	Free	0	
9632	FAMILY	4.0	785	31.0	50,000+	Free	0	
9633	BOOKS_AND_REFERENCE	4.2	5775	4.9	500,000+	Free	0	
9634	BUSINESS	NaN	2	6.8	100+	Free	0	
9635	TOOLS	4.0	885	8.0	100,000+	Free	0	
9636	FINANCE	NaN	96	1.5	10,000+	Free	0	
9637	BOOKS_AND_REFERENCE	3.3	52	3.6	5,000+	Free	0	
9638	FAMILY	5.0	22	8.6	1,000+	Free	0	
9639	FAMILY	NaN	6	2.5	50+	Free	0	

9640	PRODUCTIVITY	NaN	0	3.1	10+	Free	0
9641	VIDEO_PLAYERS	NaN	1	2.9	100+	Free	0
9642	MEDICAL	NaN	67	82.0	10,000+	Free	0
9643	SOCIAL	NaN	7	7.7	100+	Free	0
9644	SOCIAL	4.0	88486	NaN	5,000,000+	Free	0
9645	FAMILY	4.2	117	13.0	5,000+	Free	0
9646	COMICS	3.4	291	13.0	10,000+	Free	0
9647	BOOKS_AND_REFERENCE	4.6	603	7.4	10,000+	Free	0
9648	NEWS_AND_MAGAZINES	3.8	881	2.3	100,000+	Free	0
9649	MAPS_AND_NAVIGATION	NaN	38	9.8	5,000+	Free	0
9650	WEATHER	3.8	1195	0.6	100,000+	Free	0
9651	BOOKS_AND_REFERENCE	4.8	44	0.6	1,000+	Free	0
9652	FAMILY	4.0	7	2.6	500+	Free	0
9653	BUSINESS	NaN	0	9.6	10+	Free	0
9654	FAMILY	4.5	38	53.0	5,000+	Free	0
9655	FAMILY	5.0	4	3.6	100+	Free	0
9656	MEDICAL	NaN	3	9.5	1,000+	Free	0
9657	BOOKS_AND_REFERENCE	4.5	114	NaN	1,000+	Free	0
9658	LIFESTYLE	4.5	398307	19.0	10,000,000+	Free	0

	Content Rating	Genres	Last Updated \
0	Everyone	Art & Design	January 7, 2018
1	Everyone	Art & Design;Pretend Play	January 15, 2018
2	Everyone	Art & Design	August 1, 2018
3	Teen	Art & Design	June 8, 2018
4	Everyone	Art & Design;Creativity	June 20, 2018
5	Everyone	Art & Design	March 26, 2017
6	Everyone	Art & Design	April 26, 2018
7	Everyone	Art & Design	June 14, 2018
8	Everyone	Art & Design	September 20, 2017
9	Everyone	Art & Design;Creativity	July 3, 2018
10	Everyone	Art & Design	October 27, 2017
11	Everyone	Art & Design	July 31, 2018
12	Teen	Art & Design	April 2, 2018
13	Everyone	Art & Design	June 26, 2018
14	Everyone	Art & Design	August 3, 2018
15	Everyone	Art & Design	June 6, 2018
16	Everyone	Art & Design	July 31, 2018
17	Everyone	Art & Design	November 7, 2017
18	Everyone	Art & Design	August 3, 2018
19	Everyone	Art & Design	July 30, 2018
20	Everyone	Art & Design	April 20, 2018
21	Everyone	Art & Design	March 20, 2018
22	Everyone 10+	Art & Design	July 12, 2018
23	Everyone	Art & Design;Action & Adventure	March 7, 2018
24	Everyone	Art & Design	July 7, 2018
25	Everyone	Art & Design	April 25, 2018
26	Everyone	Art & Design;Creativity	October 11, 2017

27	Everyone	Art & Design	March 21, 2018
28	Everyone	Art & Design	July 12, 2018
29	Everyone	Art & Design	August 22, 2017
...
9629	Everyone	Auto & Vehicles	July 24, 2018
9630	Everyone	Education	June 13, 2018
9631	Everyone	Business	July 17, 2018
9632	Teen	Entertainment	June 3, 2015
9633	Everyone	Books & Reference	July 19, 2018
9634	Everyone	Business	August 6, 2018
9635	Everyone	Tools	October 30, 2015
9636	Everyone	Finance	May 19, 2016
9637	Teen	Books & Reference	August 5, 2017
9638	Teen	Education	June 27, 2018
9639	Everyone	Entertainment	May 29, 2018
9640	Everyone	Productivity	December 1, 2017
9641	Everyone	Video Players & Editors	April 22, 2018
9642	Everyone	Medical	July 31, 2018
9643	Teen	Social	February 2, 2018
9644	Mature 17+	Social	March 23, 2018
9645	Everyone	Education	June 13, 2018
9646	Everyone	Comics	May 15, 2017
9647	Everyone	Books & Reference	June 19, 2016
9648	Everyone	News & Magazines	January 20, 2014
9649	Everyone	Maps & Navigation	June 13, 2018
9650	Everyone	Weather	February 16, 2014
9651	Everyone	Books & Reference	March 23, 2014
9652	Everyone	Education	June 18, 2017
9653	Everyone	Business	September 29, 2016
9654	Everyone	Education	July 25, 2017
9655	Everyone	Education	July 6, 2018
9656	Everyone	Medical	January 20, 2017
9657	Mature 17+	Books & Reference	January 19, 2015
9658	Everyone	Lifestyle	July 25, 2018

	Current Ver	Android Ver
0	1.0.0	4.0.3 and up
1	2.0.0	4.0.3 and up
2	1.2.4	4.0.3 and up
3	Varies with device	4.2 and up
4	1.1	4.4 and up
5	1	2.3 and up
6	1.1	4.0.3 and up
7	6.1.61.1	4.2 and up
8	2.9.2	3.0 and up
9	2.8	4.0.3 and up
10	1.0.4	4.1 and up
11	1.0.15	4.0 and up

12	3.8	4.1 and up
13	1.0.4	4.4 and up
14	1.2.3	2.3 and up
15	NaN	4.2 and up
16	3.1	4.1 and up
17	1	2.3 and up
18	2.2.5	4.0.3 and up
19	5.5.4	4.1 and up
20	4	4.1 and up
21	1.1	4.0.3 and up
22	2.2.6.2	4.0.3 and up
23	1.0.0	4.1 and up
24	1.1.3	4.1 and up
25	1.5	3.0 and up
26	1.0.8	4.0.3 and up
27	1.03	4.0.3 and up
28	6	2.3 and up
29	1	2.3 and up
...
9629	1.3.6	4.4W and up
9630	2.0.20	4.0.3 and up
9631	2.1.10	4.1 and up
9632	1.3.1	3.0 and up
9633	7.0.4.6	4.2 and up
9634	2.1.8	4.1 and up
9635	1.0.612928	5.0 and up
9636	2.3	2.2 and up
9637	0.3.4	4.1 and up
9638	3.8.0	4.1 and up
9639	1	4.0.3 and up
9640	1	4.4 and up
9641	1	4.0.3 and up
9642	2.2.2	4.4 and up
9643	1	4.0 and up
9644	Varies with device	Varies with device
9645	2.0.20	4.0.3 and up
9646	2.0.1	4.0 and up
9647	2.96	4.1 and up
9648	1.5	1.6 and up
9649	2.0.148.0	4.0 and up
9650	6	2.1 and up
9651	0.8	2.2 and up
9652	1.0.0	4.1 and up
9653	1.1.5	4.0 and up
9654	1.48	4.1 and up
9655	1	4.1 and up
9656	1	2.2 and up
9657	Varies with device	Varies with device

9658 Varies with device Varies with device

[9659 rows x 14 columns]

	Unnamed: 0	App \
0	0	Photo Editor & Candy Camera & Grid & ScrapBook
1	1	Coloring book moana
2	2	U Launcher Lite - FREE Live Cool Themes, Hide ...
3	3	Sketch - Draw & Paint
4	4	Pixel Draw - Number Art Coloring Book

	Category	Rating	Reviews	Size	Installs	Type	Price \
0	ART_AND_DESIGN	4.1	159	19.0	10,000+	Free	0
1	ART_AND_DESIGN	3.9	967	14.0	500,000+	Free	0
2	ART_AND_DESIGN	4.7	87510	8.7	5,000,000+	Free	0
3	ART_AND_DESIGN	4.5	215644	25.0	50,000,000+	Free	0
4	ART_AND_DESIGN	4.3	967	2.8	100,000+	Free	0

	Content Rating	Genres	Last Updated \
0	Everyone	Art & Design	January 7, 2018
1	Everyone	Art & Design;Pretend Play	January 15, 2018
2	Everyone	Art & Design	August 1, 2018
3	Teen	Art & Design	June 8, 2018
4	Everyone	Art & Design;Creativity	June 20, 2018

	Current Ver	Android Ver
0	1.0.0	4.0.3 and up
1	2.0.0	4.0.3 and up
2	1.2.4	4.0.3 and up
3	Varies with device	4.2 and up
4	1.1	4.4 and up

0.2 2. Data cleaning

Data cleaning is one of the most essential subtask any data science project. Although it can be a very tedious process, it's worth should never be undermined.

By looking at a random sample of the dataset rows (from the above task), we observe that some entries in the columns like Installs and Price have a few special characters (+ , < /code >)duetothewaythenumbershavebeenrepresented.Thispreventsthecolumnsfrombeingpurelynumeric,makingitdif
9]. < /p >< p > Hence,wenowproceedtocleanourdata.Specificially,thespecialcharacters < code >,< /code > and < code > + < /code > presentin < code > Installs < /code > columnand < code > present in Price column need to be removed.

It is also always a good practice to print a summary of your dataframe after completing data cleaning. We will use the info() method to acheive this.

```
[659]: # List of characters to remove
chars_to_remove = ["+", ",", "$"]
# List of column names to clean
```

```
cols_to_clean = ["Installs", "Price"]

# Loop for each column in cols_to_clean
for col in cols_to_clean:
    # Loop for each char in chars_to_remove
    for char in chars_to_remove:
        # Replace the character with an empty string
        apps[col] = apps[col].apply(lambda x: x.replace(char, ""))

# Print a summary of the apps dataframe
print(apps.info())
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 9659 entries, 0 to 9658
Data columns (total 14 columns):
Unnamed: 0      9659 non-null int64
App             9659 non-null object
Category        9659 non-null object
Rating          8196 non-null float64
Reviews         9659 non-null int64
Size            8432 non-null float64
Installs        9659 non-null object
Type            9659 non-null object
Price           9659 non-null object
Content Rating  9659 non-null object
Genres          9659 non-null object
Last Updated    9659 non-null object
Current Ver     9651 non-null object
Android Ver     9657 non-null object
dtypes: float64(2), int64(2), object(10)
memory usage: 1.1+ MB
None
```

0.3 3. Correcting data types

From the previous task we noticed that Installs and Price were categorized as object data type (and not int or float) as we would like. This is because these two columns originally had mixed input types: digits and special characters. To know more about Pandas data types, read this.

The four features that we will be working with most frequently henceforth are Installs, Size, Rating and Price. While Size and Rating are both float (i.e. purely numerical data types), we still need to work on Installs and Price to make them numeric.

```
[661]: import numpy as np

# Convert Installs to float data type
apps["Installs"] = apps["Installs"].astype(float)
```



```
# Convert Price to float data type
apps["Price"] = apps["Price"].astype(float)

# Checking dtypes of the apps dataframe
print(apps.info())
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 9659 entries, 0 to 9658
Data columns (total 14 columns):
Unnamed: 0      9659 non-null int64
App             9659 non-null object
Category        9659 non-null object
Rating          8196 non-null float64
Reviews         9659 non-null int64
Size            8432 non-null float64
Installs        9659 non-null float64
Type            9659 non-null object
Price           9659 non-null float64
Content Rating  9659 non-null object
Genres          9659 non-null object
Last Updated    9659 non-null object
Current Ver     9651 non-null object
Android Ver     9657 non-null object
dtypes: float64(4), int64(2), object(8)
memory usage: 1.1+ MB
None
```

0.4 4. Exploring app categories

With more than 1 billion active users in 190 countries around the world, Google Play continues to be an important distribution platform to build a global audience. For businesses to get their apps in front of users, it's important to make them more quickly and easily discoverable on Google Play. To improve the overall search experience, Google has introduced the concept of grouping apps into categories.

This brings us to the following questions:

Which category has the highest share of (active) apps in the market?

Is any specific category dominating the market?

Which categories have the fewest number of apps?

We will see that there are 33 unique app categories present in our dataset. Family and Game apps have the highest market prevalence. Interestingly, Tools, Business and Medical apps are also at the top.

```
[663]: import plotly
plotly.offline.init_notebook_mode(connected=True)
import plotly.graph_objs as go
```

```

# Print the total number of unique categories
num_categories = apps['Category'].nunique()
print('Number of categories = ', num_categories)

# Count the number of apps in each 'Category'.
num_apps_in_category = apps['Category'].value_counts()

# Sort num_apps_in_category in descending order based on the count of apps in
↳ each category
sorted_num_apps_in_category = num_apps_in_category.sort_values(ascending =
↳ False)

data = [go.Bar(
    x = num_apps_in_category.index, # index = category name
    y = num_apps_in_category.values, # value = count
)]

plotly.offline.iplot(data)

```

Number of categories = 33

0.5 5. Distribution of app ratings

After having witnessed the market share for each category of apps, let's see how all these apps perform on an average. App ratings (on a scale of 1 to 5) impact the discoverability, conversion of apps as well as the company's overall brand image. Ratings are a key performance indicator of an app.

From our research, we found that the average volume of ratings across all app categories is 4.17. The histogram plot is skewed to the left indicating that the majority of the apps are highly rated with only a few exceptions in the low-rated apps.

```

[665]: # Average rating of apps
avg_app_rating = apps['Rating'].mean()
print('Average app rating = ', avg_app_rating)

# Distribution of apps according to their ratings
data = [go.Histogram(
    x = apps['Rating']
)]

# Vertical dashed line to indicate the average app rating
layout = {'shapes': [{
    'type' : 'line',
    'x0': avg_app_rating,
    'y0': 0,
    'x1': avg_app_rating,

```

```

        'y1': 1000,
        'line': { 'dash': 'dashdot' }
    }]
}

```

```
plotly.offline.iplot({'data': data, 'layout': layout})
```

Average app rating = 4.173243045387994

0.6 6. Size and price of an app

Let's now examine app size and app price. For size, if the mobile app is too large, it may be difficult and/or expensive for users to download. Lengthy download times could turn users off before they even experience your mobile app. Plus, each user's device has a finite amount of disk space. For price, some users expect their apps to be free or inexpensive. These problems compound if the developing world is part of your target market; especially due to internet speeds, earning power and exchange rates.

How can we effectively come up with strategies to size and price our app?

Does the size of an app affect its rating?

Do users really care about system-heavy apps or do they prefer light-weighted apps?

Does the price of an app affect its rating?

Do users always prefer free apps over paid apps?

We find that the majority of top rated apps (rating over 4) range from 2 MB to 20 MB. We also find that the vast majority of apps price themselves under \$10.

```

[667]: %matplotlib inline
import seaborn as sns
sns.set_style("darkgrid")
import warnings
warnings.filterwarnings("ignore")

# Select rows where both 'Rating' and 'Size' values are present (ie. the two
→ values are not null)
apps_with_size_and_rating_present = apps[(~apps['Rating'].isnull()) &
→ (~apps['Size'].isnull())

# Subset for categories with at least 250 apps
large_categories = apps_with_size_and_rating_present.groupby('Category').
→ filter(lambda x: len(x) >= 250)

# Plot size vs. rating
plt1 = sns.jointplot(x = large_categories['Size'], y =
→ large_categories['Rating'])

# Select apps whose 'Type' is 'Paid'

```

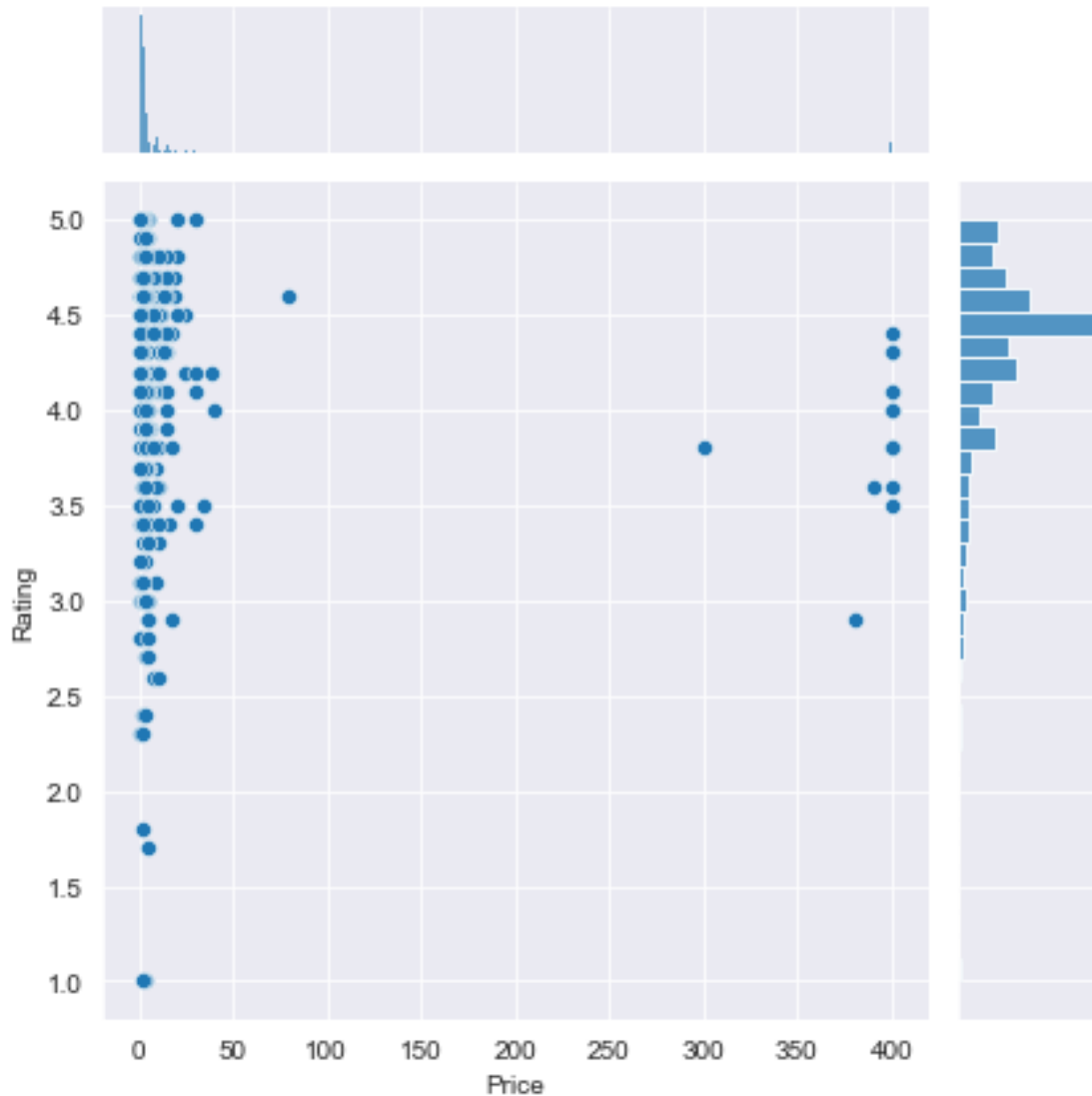
```

paid_apps =
    ↪ apps_with_size_and_rating_present[apps_with_size_and_rating_present['Price']
    ↪ != 0]

# Plot price vs. rating
plt2 = sns.jointplot(x = paid_apps['Price'], y = paid_apps['Rating'])

```





0.7 7. Relation between app category and app price

So now comes the hard part. How are companies and developers supposed to make ends meet? What monetization strategies can companies use to maximize profit? The costs of apps are largely based on features, complexity, and platform.

There are many factors to consider when selecting the right pricing strategy for your mobile app. It is important to consider the willingness of your customer to pay for your app. A wrong price could break the deal before the download even happens. Potential customers could be turned off by what they perceive to be a shocking cost, or they might delete an app they've downloaded after receiving too many ads or simply not getting their money's worth.

Different categories demand different price ranges. Some apps that are simple and used daily, like the calculator app, should probably be kept free. However, it would make sense to charge for a

highly-specialized medical app that diagnoses diabetic patients. Below, we see that Medical and Family apps are the most expensive. Some medical apps extend even up to \$80! All game apps are reasonably priced below \$20.

```
[669]: import matplotlib.pyplot as plt
fig, ax = plt.subplots()
fig.set_size_inches(15, 8)

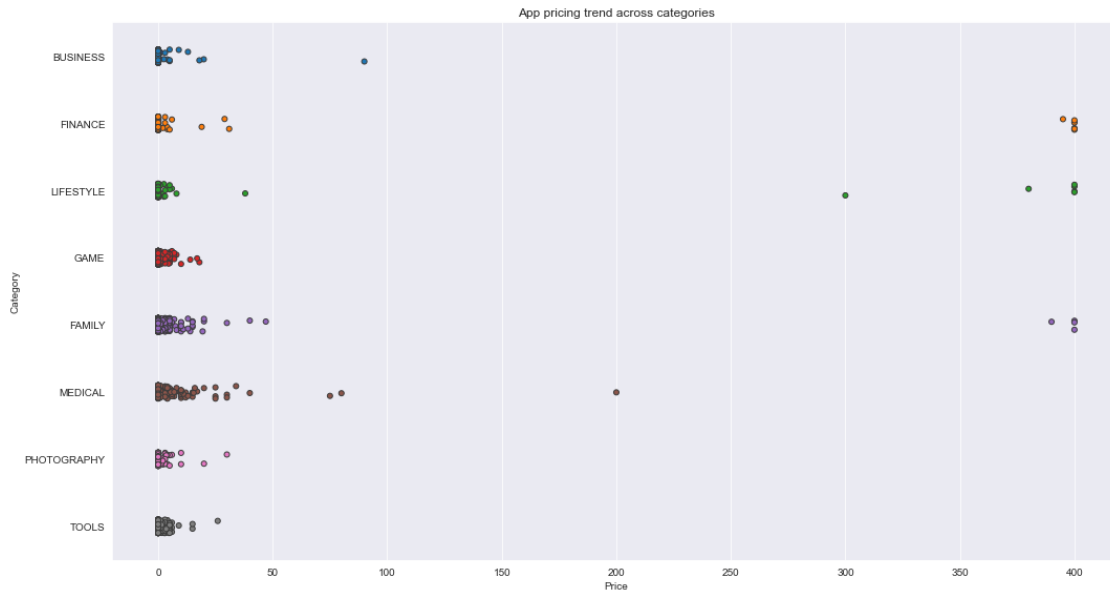
# Select a few popular app categories
popular_app_cats = apps[apps.Category.isin(['GAME', 'FAMILY', 'PHOTOGRAPHY',
                                            'MEDICAL', 'TOOLS', 'FINANCE',
                                            'LIFESTYLE', 'BUSINESS'])]

# Examine the price trend by plotting Price vs Category
ax = sns.stripplot(x = popular_app_cats['Price'], y = _
    →popular_app_cats['Category'], jitter=True, linewidth=1)
ax.set_title('App pricing trend across categories')

# Apps whose Price is greater than 200
apps_above_200 = popular_app_cats[popular_app_cats['Price'] > 200]
apps_above_200[['Category', 'App', 'Price']]
```

```
[669]:
```

	Category	App	Price
3327	FAMILY	most expensive app (H)	399.99
3465	LIFESTYLE	I'm rich	399.99
3469	LIFESTYLE	I'm Rich - Trump Edition	400.00
4396	LIFESTYLE	I am rich	399.99
4398	FAMILY	I am Rich Plus	399.99
4399	LIFESTYLE	I am rich VIP	299.99
4400	FINANCE	I Am Rich Premium	399.99
4401	LIFESTYLE	I am extremely Rich	379.99
4402	FINANCE	I am Rich!	399.99
4403	FINANCE	I am rich(premium)	399.99
4406	FAMILY	I Am Rich Pro	399.99
4408	FINANCE	I am rich (Most expensive app)	399.99
4410	FAMILY	I Am Rich	389.99
4413	FINANCE	I am Rich	399.99
4417	FINANCE	I AM RICH PRO PLUS	399.99
8763	FINANCE	Eu Sou Rico	394.99
8780	LIFESTYLE	I'm Rich/Eu sou Rico/ /	399.99



0.8 8. Filter out “junk” apps

It looks like a bunch of the really expensive apps are “junk” apps. That is, apps that don’t really have a purpose. Some app developer may create an app called I Am Rich Premium or most expensive app (H) just for a joke or to test their app development skills. Some developers even do this with malicious intent and try to make money by hoping people accidentally click purchase on their app in the store.

Let’s filter out these junk apps and re-do our visualization.

```
[671]: # Select apps priced below $100
apps_under_100 = popular_app_cats[popular_app_cats['Price'] < 100]

fig, ax = plt.subplots()
fig.set_size_inches(15, 8)

# Examine price vs category with the authentic apps (apps_under_100)
ax = sns.stripplot(x = 'Price', y = 'Category', data = apps_under_100, jitter = True, linewidth = 1)
ax.set_title('App pricing trend across categories after filtering for junk apps')
```

```
[671]: Text(0.5, 1.0, 'App pricing trend across categories after filtering for junk apps')
```



0.9 9. Popularity of paid apps vs free apps

For apps in the Play Store today, there are five types of pricing strategies: free, freemium, paid, paymium, and subscription. Let's focus on free and paid apps only. Some characteristics of free apps are:

Free to download.

Main source of income often comes from advertisements.

Often created by companies that have other products and the app serves as an extension of those products.

Can serve as a tool for customer retention, communication, and customer service.

Some characteristics of paid apps are:

Users are asked to pay once for the app to download and use it.

The user can't really get a feel for the app before buying it.

Are paid apps installed as much as free apps? It turns out that paid apps have a relatively lower number of installs than free apps, though the difference is not as stark as I would have expected!

```
[673]: trace0 = go.Box(
    # Data for paid apps
    y = apps[apps['Type'] == 'Paid']['Installs'],
    name = 'Paid'
)

trace1 = go.Box(
    # Data for free apps
```



```

    y = apps[apps['Type'] == 'Free']['Installs'],
    name = 'Free'
)

layout = go.Layout(
    title = "Number of downloads of paid apps vs. free apps",
    yaxis = dict(title = "Log number of downloads",
                  type = 'log',
                  autorange = True)
)

# Add trace0 and trace1 to a list for plotting
data = [trace0, trace1]
plotly.offline.iplot({'data': data, 'layout': layout})

```

0.10 10. Sentiment analysis of user reviews

Mining user review data to determine how people feel about your product, brand, or service can be done using a technique called sentiment analysis. User reviews for apps can be analyzed to identify if the mood is positive, negative or neutral about that app. For example, positive words in an app review might include words such as ‘amazing’, ‘friendly’, ‘good’, ‘great’, and ‘love’. Negative words might be words like ‘malware’, ‘hate’, ‘problem’, ‘refund’, and ‘incompetent’.

By plotting sentiment polarity scores of user reviews for paid and free apps, we observe that free apps receive a lot of harsh comments, as indicated by the outliers on the negative y-axis. Reviews for paid apps appear never to be extremely negative. This may indicate something about app quality, i.e., paid apps being of higher quality than free apps on average. The median polarity score for paid apps is a little higher than free apps, thereby syncing with our previous observation.

In this notebook, we analyzed over ten thousand apps from the Google Play Store. We can use our findings to inform our decisions should we ever wish to create an app ourselves.

```

[675]: # Load user_reviews.csv
reviews_df = pd.read_csv('datasets/user_reviews.csv')

# Join the two dataframes
merged_df = apps.merge(reviews_df)

# Drop NA values from Sentiment and Review columns
merged_df = merged_df.dropna(subset = ['Sentiment', 'Review'])

sns.set_style('ticks')
fig, ax = plt.subplots()
fig.set_size_inches(11, 8)

# User review sentiment polarity for paid vs. free apps
ax = sns.boxplot(x = 'Type', y = 'Sentiment_Polarity', data = merged_df)
ax.set_title('Sentiment Polarity Distribution')

```

[675]: Text(0.5, 1.0, 'Sentiment Polarity Distribution')

