

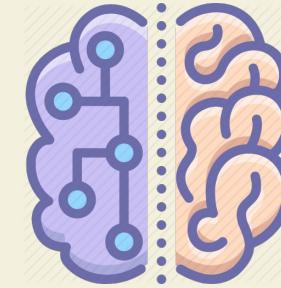


Google Play Store Apps

Phatpicha Y. (2020)



OVERVIEW OF ANALYSIS



Data Cleaning

Understand the structure of dataset and clean data to right format

Data Exploration

Find significant patterns and trends

Predictive Modeling

Construct models to predict and forecast



DATA CLEANING



Dataset Description

Googleplaystore.csv

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
5209	Adobe Illustrator Draw	PHOTOGRAPHY	4.4	65766	Varies with device	5,000,000+	Free	0	Everyone	Photography	April 2, 2018	3.4.23	5.0 and up
9138	DZ Blagues	FAMILY	4.2	580	2.1M	10,000+	Free	0	Everyone	Entertainment	December 19, 2012	1.3	2.1 and up
7873	Trauma CT Head Rules	MEDICAL	Nan	2	1.4M	100+	Free	0	Everyone	Medical	April 23, 2017	1.1	4.0.3 and up
9978	EV Calculator	TOOLS	4.9	85	19M	1,000+	Free	0	Everyone	Tools	July 5, 2018	1.20	4.1 and up
10282	FD Shift Calendar Widget	TOOLS	4.1	981	73k	100,000+	Free	0	Everyone	Tools	March 16, 2011	1.2.4	1.6 and up

googleplaystore_user_reviews.csv

	App	Translated_Review	Sentiment	Sentiment_Polarity	Sentiment_Subjectivity
43409	Fake Call - Fake Caller ID	I problem app.but amazing Thanks	Positive	0.400000	0.550000
271	11st	Open Market Revolution A section of the Holo-S...	Positive	0.166667	0.500000
62834	Home Street – Home Design Game	This game amazing. NO ads. NO bad gameplay . A...	Positive	0.163889	0.700000
32870	Daily Workouts - Exercise Fitness Routine Trainer	NaN	NaN	NaN	NaN
10409	BBW Dating & Curvy Singles Chat- LargeFriends	There literally point free account. You can't ...	Positive	0.325000	0.566667



DATASET DESCRIPTION: Googleplaystore.csv

```
apps.describe()
```

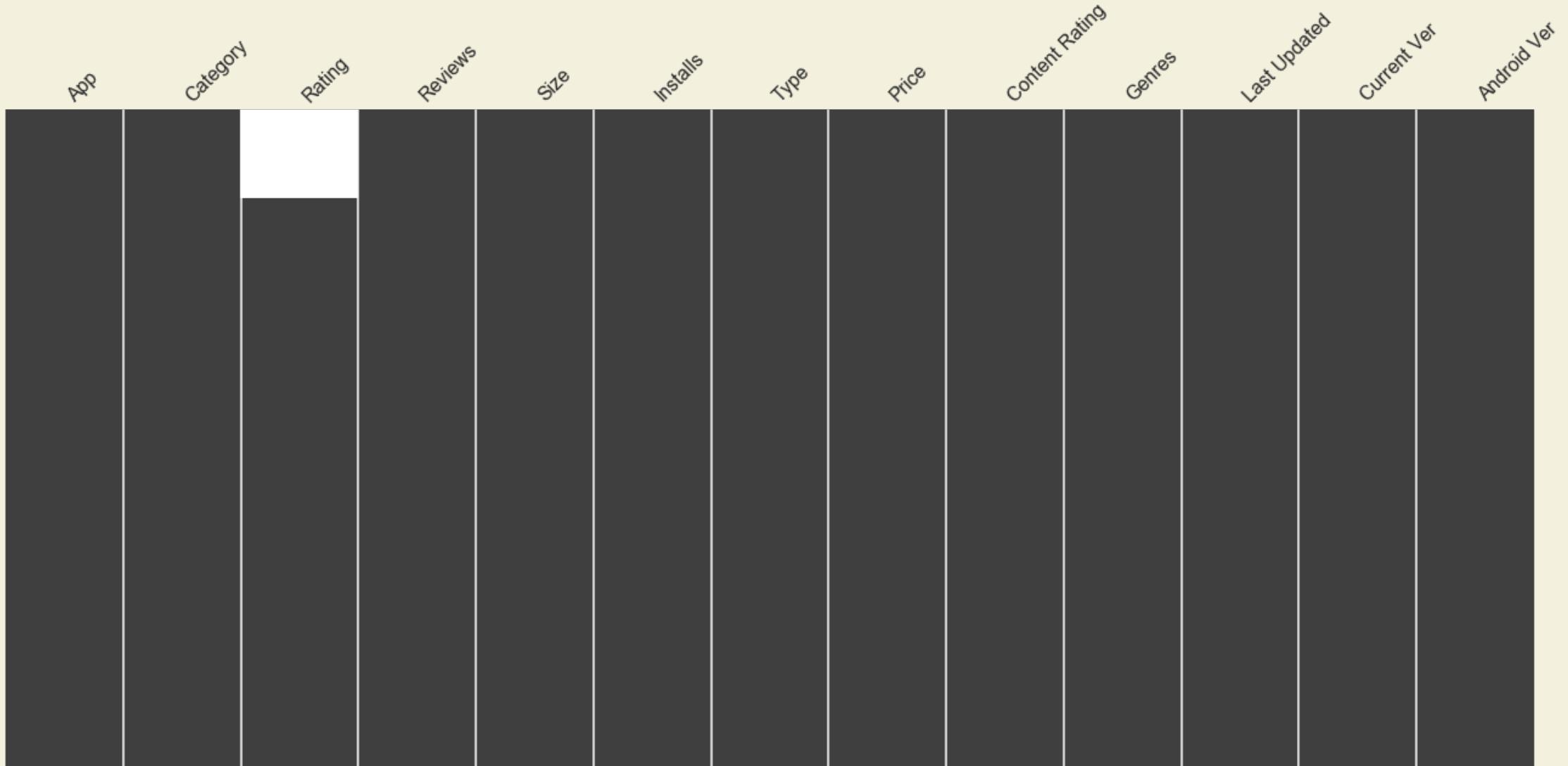
Rating	
count	9367.000000
mean	4.193338
std	0.537431
min	1.000000
25%	4.000000
50%	4.300000
75%	4.500000
max	19.000000

```
apps.describe(include=["O"])
```

	App	Category	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
count	10841	10841	10841	10841	10841	10840	10841	10840	10841	10841	10833	10838
unique	9660	34	6002	462	22	3	93	6	120	1378	2832	33
top	ROBLOX	FAMILY	0	Varies with device	1,000,000+	Free	0	Everyone	Tools	August 3, 2018	Varies with device	4.1 and up
freq	9	1972	596	1695	1579	10039	10040	8714	842	326	1459	2451

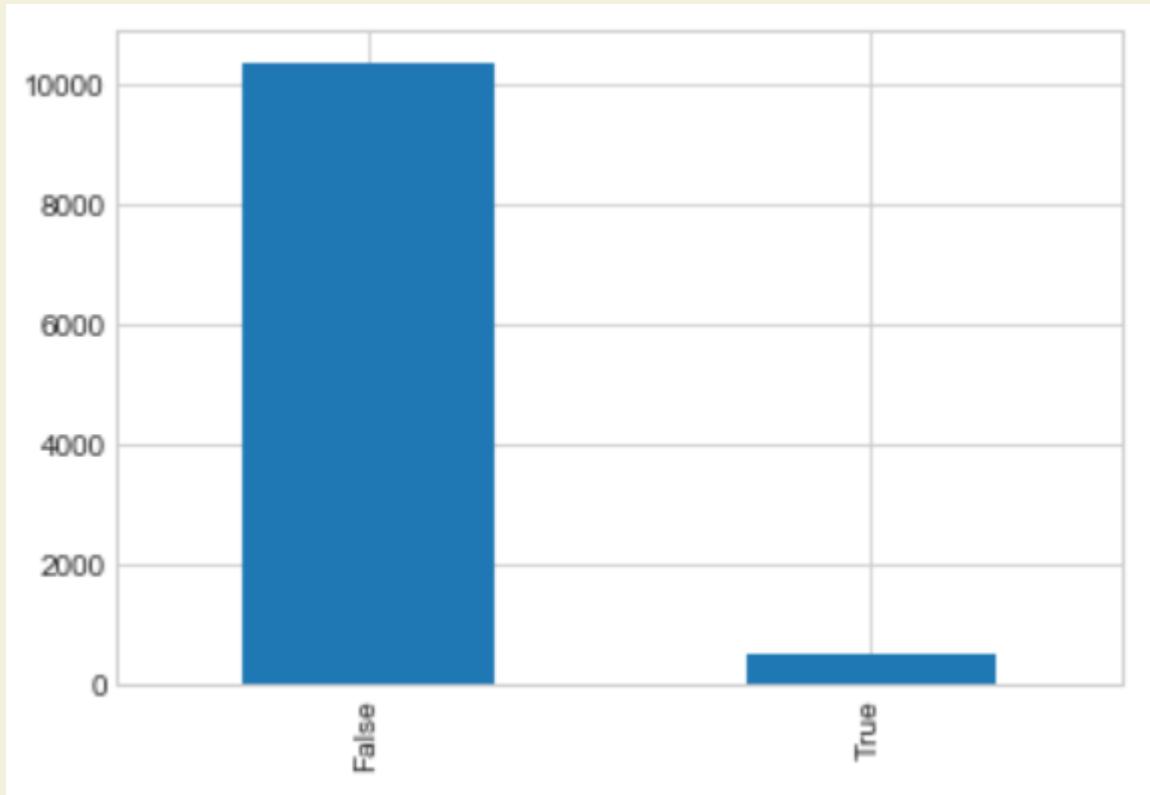


MISSING VALUES





APP Column Problem



Overall Duplicated: 483



App Duplicated: 698



9660



Category Column Problem

Category	
1.9	1
ART_AND_DESIGN	84
AUTO_AND_VEHICLES	85
BEAUTY	53
BOOKS_AND_REFERENCE	222
BUSINESS	420
COMICS	56
COMMUNICATION	315
DATING	171
EDUCATION	119
ENTERTAINMENT	102
EVENTS	64
FAMILY	1832
FINANCE	345
FOOD_AND_DRINK	112
GAME	959
HEALTH_AND_FITNESS	288
HOUSE_AND_HOME	74
LIBRARIES_AND_DEMO	84
LIFESTYLE	369
MAPS_AND_NAVIGATION	131

```
# which row is category 1.9 ?  
i = apps[apps["Category"] == "1.9"].index  
apps.loc[i]
```

App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
10472	Life Made Wi-Fi Touchscreen Photo Frame	1.9	19.0	3.0M	1,000+	Free	0	Everyone	NaN	February 11, 2018	1.0.19	4.0 and up

Category is abnormal
Rating is over
so “Drop”



Reviews Column Problem

```
: apps.loc[apps["Reviews"].str.isnumeric()]
```

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver	Cat
0	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	
1	Coloring book moana	ART_AND DESIGN	3.9	967	14M	500,000+	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	4.0.3 and up	
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	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	

Not numerical number
Too high values



pd.to_numeric(apps["Reviews"])
np.log(apps["Reviews"])



Size Column Problem

```
apps["Size"].unique()

array(['19M', '14M', '8.7M', '25M', '2.8M', '5.6M', '29M', '33M', '3.1M',
       '28M', '12M', '20M', '21M', '37M', '2.7M', '5.5M', '17M', '39M',
       '31M', '4.2M', '7.0M', '23M', '6.0M', '6.1M', '4.6M', '9.2M',
       '5.2M', '11M', '24M', 'Varies with device', '9.4M', '15M', '10M',
       '1.2M', '26M', '8.0M', '7.9M', '56M', '57M', '35M', '54M', '201k',
       '3.6M', '5.7M', '8.6M', '2.4M', '27M', '2.5M', '16M', '3.4M',
       '8.9M', '3.9M', '2.9M', '38M', '32M', '5.4M', '18M', '1.1M',
       '2.2M', '4.5M', '9.8M', '52M', '9.0M', '6.7M', '30M', '2.6M',
       '7.1M', '3.7M', '22M', '7.4M', '6.4M', '3.2M', '8.2M', '9.9M',
       '4.9M', '9.5M', '5.0M', '5.9M', '13M', '73M', '6.8M', '3.5M',
       '4.0M', '2.3M', '7.2M', '2.1M', '42M', '7.3M', '9.1M', '55M',
       '23k', '6.5M', '1.5M', '7.5M', '51M', '41M', '48M', '8.5M', '46M',
       '8.3M', '4.3M', '4.7M', '3.3M', '40M', '7.8M', '8.8M', '6.6M',
       '5.1M', '61M', '66M', '79k', '8.4M', '118k', '44M', '695k', '1.6M',
       '6.2M', '18k', '53M', '1.4M', '3.0M', '5.8M', '3.8M', '9.6M',
       '45M', '63M', '49M', '77M', '4.4M', '4.8M', '70M', '6.9M', '9.3M',
       '10.0M', '8.1M', '36M', '84M', '97M', '2.0M', '1.9M', '1.8M',
       '5.3M', '47M', '556k', '526k', '76M', '7.6M', '59M', '9.7M', '78M',
       '72M', '43M', '7.7M', '6.3M', '334k', '34M', '93M', '65M', '79M',
       '100M', '58M', '50M', '68M', '64M', '67M', '60M', '94M', '232k',
       '99M', '624k', '95M', '8.5k', '41k', '292k', '11k', '80M', '1.7M',
       '74M', '62M', '69M', '75M', '98M', '85M', '82M', '96M', '87M'],
      ...)
```

Problems:

1. Various character words
2. Null values

Solve:

```
def size_convert(x):
    x = str(x)
    if "M" in x:
        return float(x.replace("M", ""))
    elif "k" in x:
        return float(x.replace("k", ""))/1000
    elif x == "Varies with device":
        return np.NaN
    else:
        return float(x)
```

```
# fill NA
mean_Size = apps.groupby("Category")["Size"].transform("mean")
apps["Size"] = apps["Size"].fillna(mean_Size)
```



Installs Column Problem

Installs
0
0+
1+
1,000+
1,000,000+
1,000,000,000+
10+
10,000+
10,000,000+
100+
100,000+
100,000,000+
5+
5,000+
5,000,000+



```
def installs_convert(x):
    if "+" in x:
        x = x.replace("+", "")
    if "," in x:
        x = x.replace(",", "")
    return int(x)
```

```
pd.to_numeric(apps["Installs"])
```

```
np.log(apps["Installs"])
```



Type Column Problem

```
apps["Type"].unique()  
array(['Free', 'Paid', nan], dtype=object)
```



```
apps[apps["Type"].isnull()]
```

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver	Category Code	Review_Log	Install_Lc
9148	Command & Conquer: Rivals	FAMILY	Nan	0	27.187988	0	NaN	0	Everyone 10+	Strategy	June 28, 2018	Varies with device	Varies with device	11	0.0	0



Price is zero then Type should be Free

```
apps["Type"].fillna("Free", inplace=True)
```



Price Column Problem

```
apps["Price"].unique()  
  
array(['0', '$4.99', '$3.99', '$6.99', '$1.49', '$2.99', '$7.99', '$5.99',  
       '$3.49', '$1.99', '$9.99', '$7.49', '$0.99', '$9.00', '$5.49',  
       '$10.00', '$24.99', '$11.99', '$79.99', '$16.99', '$14.99',  
       '$1.00', '$29.99', '$12.99', '$2.49', '$10.99', '$1.50', '$19.99',  
       '$15.99', '$33.99', '$74.99', '$39.99', '$3.95', '$4.49', '$1.70',  
       '$8.99', '$2.00', '$3.88', '$25.99', '$399.99', '$17.99',  
       '$400.00', '$3.02', '$1.76', '$4.84', '$4.77', '$1.61', '$2.50',  
       '$1.59', '$6.49', '$1.29', '$5.00', '$13.99', '$299.99', '$379.99',  
       '$37.99', '$18.99', '$389.99', '$19.90', '$8.49', '$1.75',  
       '$14.00', '$4.85', '$46.99', '$109.99', '$154.99', '$3.08',  
       '$2.59', '$4.80', '$1.96', '$19.40', '$3.90', '$4.59', '$15.46',  
       '$3.04', '$4.29', '$2.60', '$3.28', '$4.60', '$28.99', '$2.95',  
       '$2.90', '$1.97', '$200.00', '$89.99', '$2.56', '$30.99', '$3.61',  
       '$394.99', '$1.26', '$1.20', '$1.04'], dtype=object)
```

```
apps["Price"].str.replace("$", "")  
pd.to_numeric(apps.Price)
```



Content Rating Column Problem

```
apps["Content Rating"].value_counts()
```

Everyone	7903
Teen	1036
Mature 17+	393
Everyone 10+	322
Adults only 18+	3
Unrated	2

Name: Content Rating, dtype: int64

```
apps[apps["Content Rating"] == "Unrated"]
```

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver	Category Code	Review_Log	Install_L
731	Best CG Photography	FAMILY	NaN	1	2.5	500	Free	0.0	Unrated	Entertainment	June 24, 2015	5.2	3.0 and up	11	0.000000	6.2146
826	DC Universe Online Map	TOOLS	4.1	1186	6.4	50000	Free	0.0	Unrated	Tools	February 27, 2012	1.3	2.3.3 and up	29	7.078342	10.8197

```
# Unrated to Everyone
```

```
apps["Content Rating"] = apps["Content Rating"].apply(lambda x: "Everyone" if x == "Unrated" else x)
```

```
apps["Content Rating"] = apps["Content Rating"].str.replace("+", "")
```



Genres Column Problem

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver	Category Code
2042	Draw.ly - Color by Number Pixel Art Coloring	FAMILY	4.4	18616	10M	1,000,000+	Free	0	Everyone	Casual;Education	August 5, 2018	1.0.9	4.0.3 and up	11
8176	King of Math	FAMILY	4.4	766	14M	10,000+	Paid	\$2.99	Everyone	Education;Education	May 22, 2018	1.0.11	2.2 and up	11
10694	FO SODEXO	COMMUNICATION	NaN	0	16M	100+	Free	0	Everyone	Communication	March 13, 2018	1.0	4.1 and up	6
3318	Free & Premium VPN - FinchVPN	TOOLS	4.2	19096	10M	1,000,000+	Free	0	Everyone	Tools	July 5, 2018	2.0.2	4.1 and up	29
4706	V BTS Wallpaper	PERSONALIZATION	4.7	127	9.5M	5,000+	Free	0	Everyone	Personalization	May 4, 2018	8.0	3.0 and up	23

```
sep = ";"  
main_cat = apps["Genres"].apply(lambda x: x.split(sep)[0])  
apps["Genres Category"] = main_cat  
  
child = apps["Genres"].apply(lambda x: x.split(sep)[-1])  
apps["Genres Subcategory"] = child
```

1. " ; "

2. Separate

Genres Subcategory	Action	Action & Adventure	Adventure	Arcade	Art & Design	Auto & Vehicles	Beauty	Board	Books & Reference	Brain Games
Genres Category	Action	Action & Adventure	Adventure	Arcade	Art & Design	Auto & Vehicles	Beauty	Board	Books & Reference	Brain Games
Action	299	12	0	0	0	0	0	0	0	0
Adventure	0	5	73	0	0	0	0	0	0	1
Arcade	0	14	0	184	0	0	0	0	0	0
Art & Design	0	1	0	0	57	0	0	0	0	0
Auto & Vehicles	0	0	0	0	0	85	0	0	0	0

3. Get Main-genres & Sub-genres



Last Updated Column Problem

```
apps["Last Updated"] = pd.to_datetime(apps["Last Updated"])
```

```
# Year attribute of a date
apps["Year"] = apps["Last Updated"].map(lambda x: x.year)
```

```
def find_date(item):
    time = item
    now = datetime.today()
    td = now - time
    return td.days
```

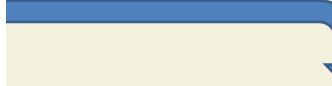
```
apps["Last Updated Day"] = apps["Last Updated"].apply(find_date)
```



Current Ver Column Problem

```
apps.groupby(apps["Current Ver"]).size()
```

```
Current Ver
0.0.0.2           1
0.0.1             15
0.0.10            1
0.0.2             4
0.0.3             2
..
v7.0.7.1.0625.1_06_0629 1
v7.0.9.1.0526.1_06_0704 1
v8.0.1.8.0629.1       1
v8[1.0.10]           1
version 0.994         1
Length: 2817, dtype: int64
```



```
# Various Year and Version so various
apps["Current Ver"].fillna("Varies with device", inplace=True)
```



```
#Deal with various character
apps["Current Ver"].apply(lambda x: "99.99" if x == "Varies with device" else re.findall("^[0-9]\.[0-9]|[\d]|\w*", str(x))[0])
```



Android Ver Column Problem

```
apps[apps["Android Ver"].isnull()]
```

stalls	Type	Price	Content Rating	Genres	...	Review_Log	Install_Log	Type Code	Content Rating Code	Genres Category	Genres Subcategory	Genres Category Code	Genres Subcategory Code	Year	Last Updated Day
1000	Paid	1.49	Everyone	Personalization	...	5.438079	6.907755	1	1	Personalization	Personalization	30	34	2018	656
10000	Free	0.00	Everyone	Personalization	...	5.241747	9.210340	0	1	Personalization	Personalization	30	34	2018	771

```
cross_anv = pd.crosstab(index=apps[(apps["Year"] == 2018)]["Year"], columns=apps[(apps["Year"] == 2018)]["Android Ver"])  
cross_anv
```

Android Ver	1.0 and up	1.5 and up	1.6 and up	2.0 and up	2.0.1 and up	2.1 and up	2.2 and up	2.3 and up	2.3.3 and up	3.0 and up	...	5.0 - 7.1.1	5.0 and up	5.1 and up	6.0 and up	7.0 - 7.1.1	7.0 and up	7.1 and up	8.0 and up	Varies with device	
Year																					
2018	1	1	10	5	1	27	24	167	56	56	...	1	1	459	16	41	1	39	2	6	831

```
most_frequent_anv = apps["Android Ver"].value_counts().idxmax()  
apps["Android Ver"].fillna(most_frequent_anv, inplace=True)
```

```
apps["Android Ver"].apply(lambda x: "99.99" if x == "Varies with device" else re.findall("^[0-9]\.[0-9]|\[\d\]|\W*", str(x))[0])
```



Rating Column Problem

1463

```
apps["Rating"].unique()
```

```
array([4.1, 3.9, 4.7, 4.5, 4.3, 4.4, 3.8, 4.2, 4.6, 3.2, 4. , nan, 4.8,
       4.9, 3.6, 3.7, 3.3, 3.4, 3.5, 3.1, 5. , 2.6, 3. , 1.9, 2.5, 2.8,
       2.7, 1. , 2.9, 2.3, 2.2, 1.7, 2. , 1.8, 2.4, 1.6, 2.1, 1.4, 1.5,
       1.2])
```

Data columns (total 24 columns):

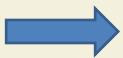
#	Column	Non-Null Count	Dtype
0	App	9659	non-null
1	Category	9659	non-null
2	Reviews	9659	non-null
3	Size	9659	non-null
4	Installs	9659	non-null
5	Type	9659	non-null
6	Price	9659	non-null
7	Content Rating	9659	non-null
8	Genres	9659	non-null
9	Last Updated	9659	non-null
10	Current Ver	9659	non-null
11	Android Ver	9659	non-null
12	Category Code	9659	non-null
13	Review_Log	9659	non-null
14	Install_Log	9659	non-null
15	Type Code	9659	non-null
16	Content Rating Code	9659	non-null
17	Genres Category	9659	non-null
18	Genres Subcategory	9659	non-null
19	Genres Category Code	9659	non-null
20	Genres Subcategory Code	9659	non-null
21	Year	9659	non-null
22	Last Updated Day	9659	non-null
23	Rating	9659	non-null

Data columns (total 51 columns):

#	Column	Non-Null Count	Dtype
0	App	9659	non-null
1	Rating	9659	non-null
2	Reviews	9659	non-null
3	Size	9659	non-null
4	Installs	9659	non-null
5	Price	9659	non-null
6	Last Updated	9659	non-null
7	Current Ver	9659	non-null
8	Android Ver	9659	non-null
9	Category Code	9659	non-null
10	Review_Log	9659	non-null
11	Install_Log	9659	non-null
12	Type Code	9659	non-null
13	Content Rating Code	9659	non-null
14	Genres Category Code	9659	non-null
15	Genres Subcategory Code	9659	non-null
16	Year	9659	non-null
17	Last Updated Day	9659	non-null
18	ART_AND_DESIGN	9659	non-null
19	AUTO_AND_VEHICLES	9659	non-null
20	BEAUTY	9659	non-null
21	BOOKS_AND_REFERENCE	9659	non-null
22	BUSINESS	9659	non-null

Step 1: LabelEncoder()

Step 2: get_dummies()



Step 3: KNNImputer()



DATASET DESCRIPTION: googleplaystore_user_reviews.csv

```
reviews.describe()
```

	Sentiment_Polarity	Sentiment_Subjectivity
count	37432.000000	37432.000000
mean	0.182146	0.492704
std	0.351301	0.259949
min	-1.000000	0.000000
25%	0.000000	0.357143
50%	0.150000	0.514286
75%	0.400000	0.650000
max	1.000000	1.000000

```
reviews.describe(include=["O"])
```

	App	Translated_Review	Sentiment
count	64295	37427	37432
unique	1074	27994	3
top	Bowmasters	Good	Positive
freq	320	247	23998



Merge 2 Tables

1

```
a = apps[['App', 'Category']]
diff_df = pd.merge(reviews, a, on="App", how="left")
diff_df
```

	App	Translated_Review	Sentiment	Sentiment_Polarity	Sentiment_Subjectivity	Category
0	10 Best Foods for You	I like eat delicious food. That's I'm cooking ...	Positive	1.00	0.533333	HEALTH_AND_FITNESS
1	10 Best Foods for You	This help eating healthy exercise regular basis	Positive	0.25	0.288462	HEALTH_AND_FITNESS
2	10 Best Foods for You		NaN	NaN	NaN	HEALTH_AND_FITNESS
3	10 Best Foods for You	Works great especially going grocery store	Positive	0.40	0.875000	HEALTH_AND_FITNESS
4	10 Best Foods for You	Best idea us	Positive	1.00	0.300000	HEALTH_AND_FITNESS
...
64290	Houzz Interior Design Ideas		NaN	NaN	NaN	HOUSE_AND_HOME
64291	Houzz Interior Design Ideas		NaN	NaN	NaN	HOUSE_AND_HOME
64292	Houzz Interior Design Ideas		NaN	NaN	NaN	HOUSE_AND_HOME
64293	Houzz Interior Design Ideas		NaN	NaN	NaN	HOUSE_AND_HOME
64294	Houzz Interior Design Ideas		NaN	NaN	NaN	HOUSE_AND_HOME

2

```
diff_df.drop_duplicates(inplace=True)
```

3

```
diff_df.isnull().any()
```

App		False
Translated_Review		True
Sentiment		True
Sentiment_Polarity		True
Sentiment_Subjectivity		True
Category		True
dtype:	bool	

4

```
diff_df.dropna(axis=0, inplace=True)
```

Result

	App	Translated_Review	Sentiment	Sentiment_Polarity	Sentiment_Subjectivity	Category	
4587	Agar.io	This worst game ever teamer don't take game im...	Negative	-0.600	0.6	GAME	
10623	BBWCupid - BBW Dating App		good	Positive	0.700	0.6	DATING
61091	Hello Kitty Nail Salon	SO MANY ADS I CAN'T EVEN TOUCH MY SCREEN WITHO...	Positive	0.625	0.5	GAME	
13305	BeSoccer - Soccer Live Score	Best app. Value money	Positive	1.000	0.3	SPORTS	
51345	GO SMS Pro - Messenger, Free Themes, Emoji	The resnts attachments recipient. Any attachm...	Positive	0.500	0.5	COMMUNICATION	

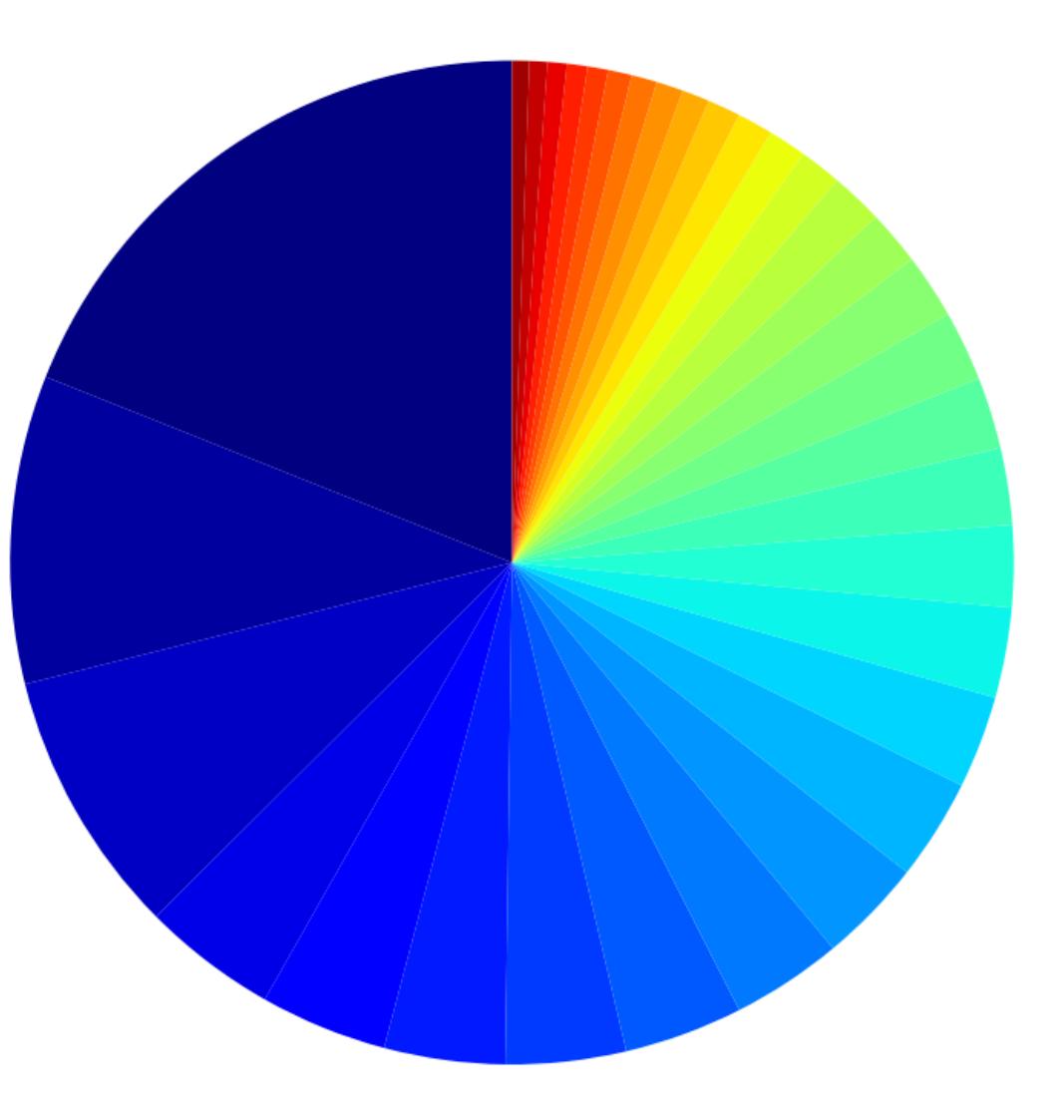


EXPLORATORY DATA ANALYSIS



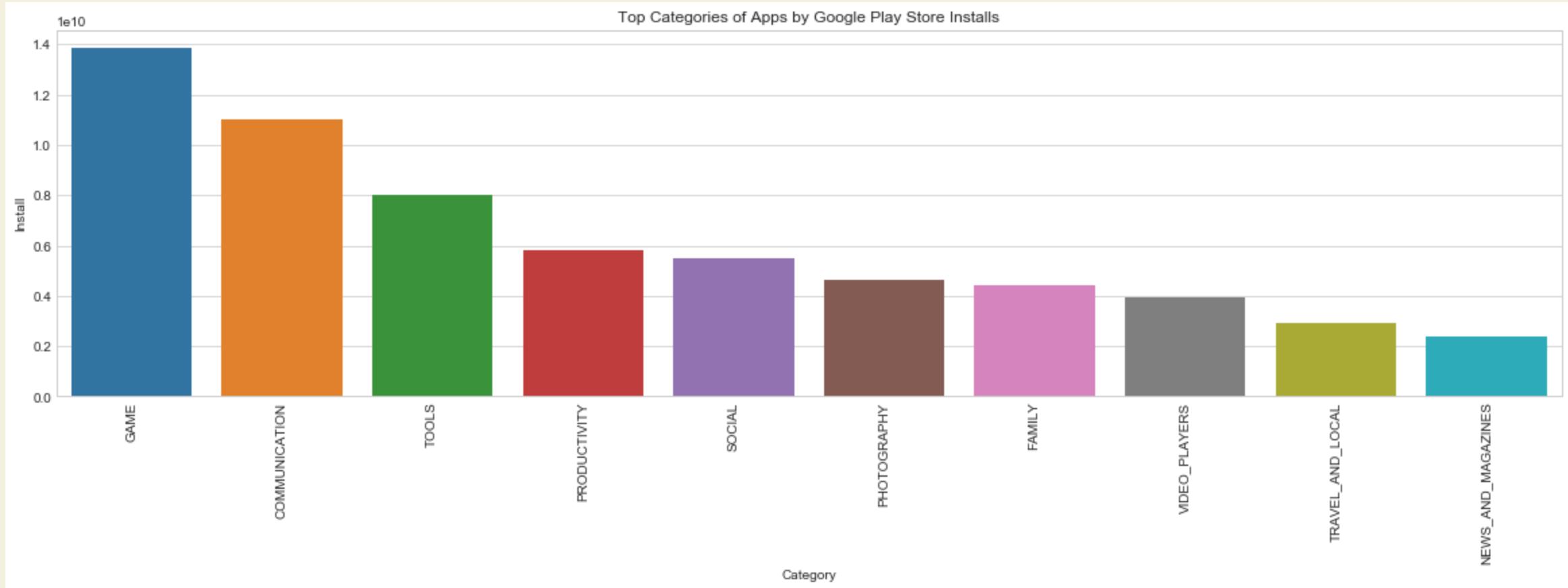
Overview of Apps' Categories in Google Play Store

FAMILY, 19.0%
GAME, 9.9%
TOOLS, 8.6%
BUSINESS, 4.3%
MEDICAL, 4.1%
PERSONALIZATION, 3.9%
PRODUCTIVITY, 3.9%
LIFESTYLE, 3.8%
FINANCE, 3.6%
SPORTS, 3.4%
COMMUNICATION, 3.3%
HEALTH_AND_FITNESS, 3.0%
PHOTOGRAPHY, 2.9%
NEWS_AND_MAGAZINES, 2.6%
SOCIAL, 2.5%
BOOKS_AND_REFERENCE, 2.3%
TRAVEL_AND_LOCAL, 2.3%
SHOPPING, 2.1%
DATING, 1.8%
VIDEO_PLAYERS, 1.7%
MAPS_AND_NAVIGATION, 1.4%
EDUCATION, 1.2%
FOOD_AND_DRINK, 1.2%
ENTERTAINMENT, 1.1%
AUTO_AND_VEHICLES, 0.9%
LIBRARIES_AND_DEMO, 0.9%
WEATHER, 0.8%
HOUSE_AND_HOME, 0.8%
ART_AND DESIGN, 0.7%
EVENTS, 0.7%
PARENTING, 0.6%
COMICS, 0.6%
BEAUTY, 0.5%





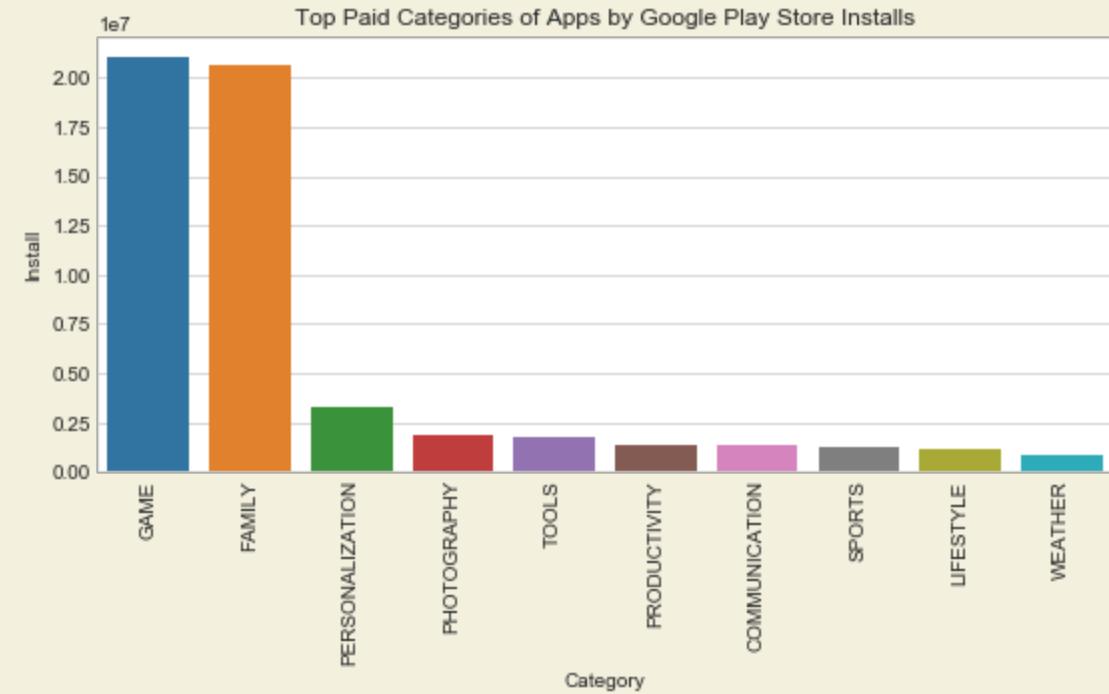
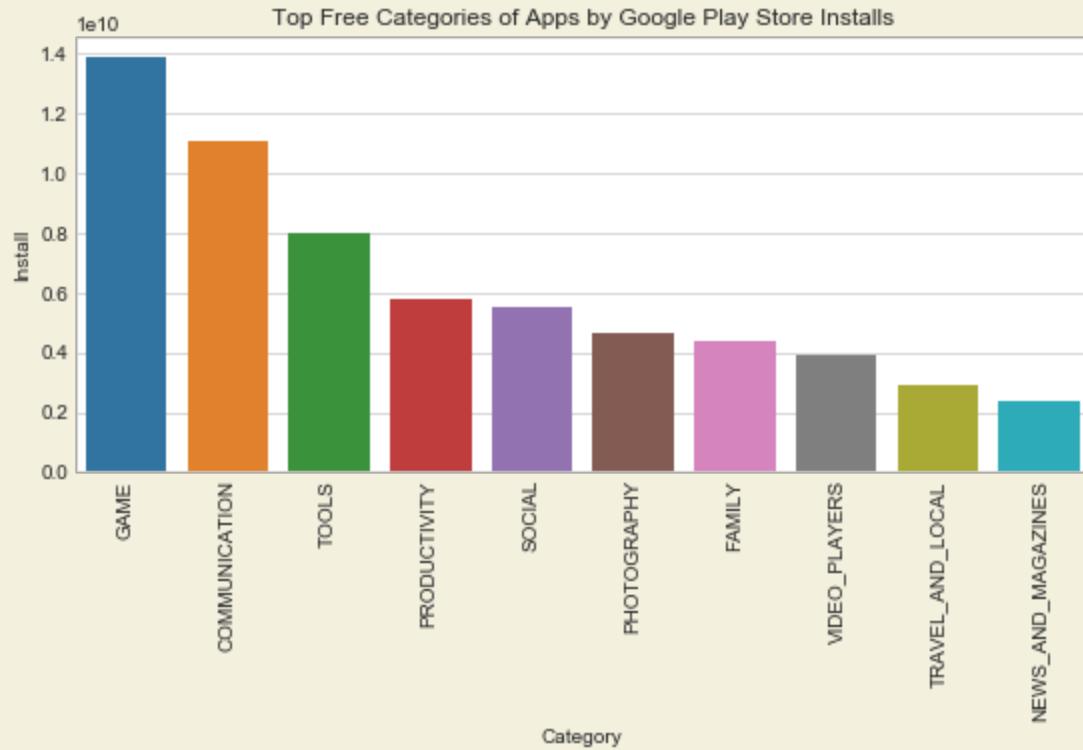
Install Strategy



- Game applications have the largest installations.



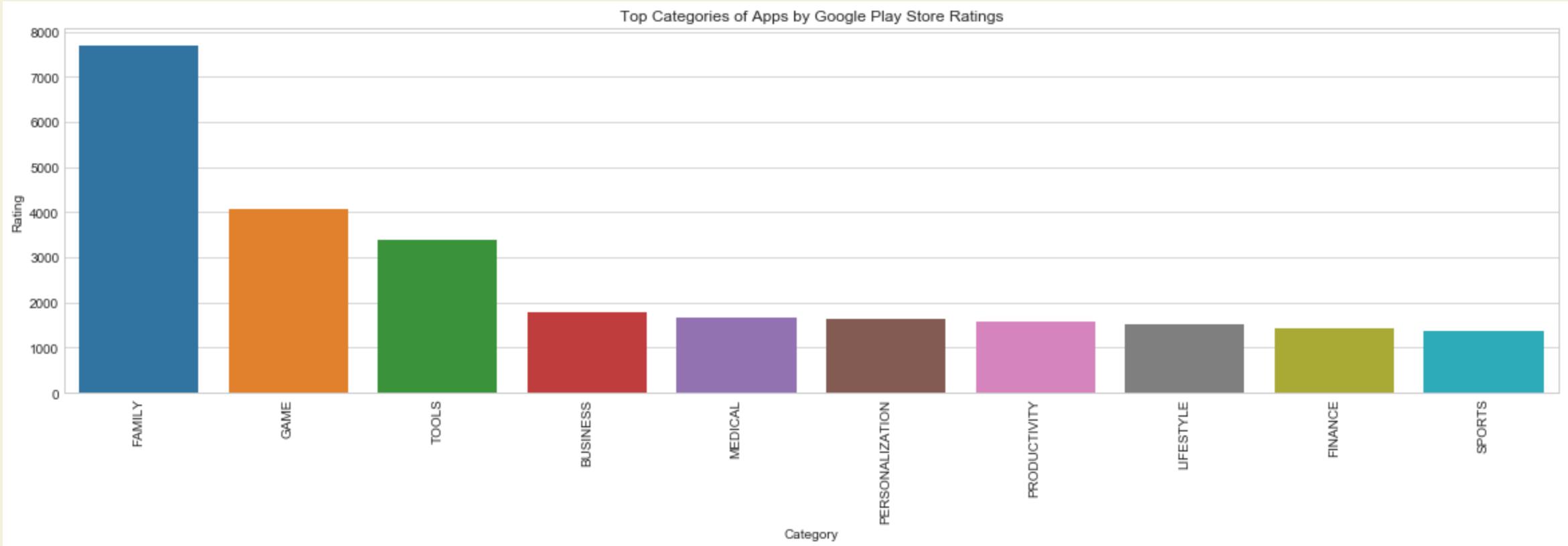
Install Strategy



- For both free and paid, Game application is the most popular installed.



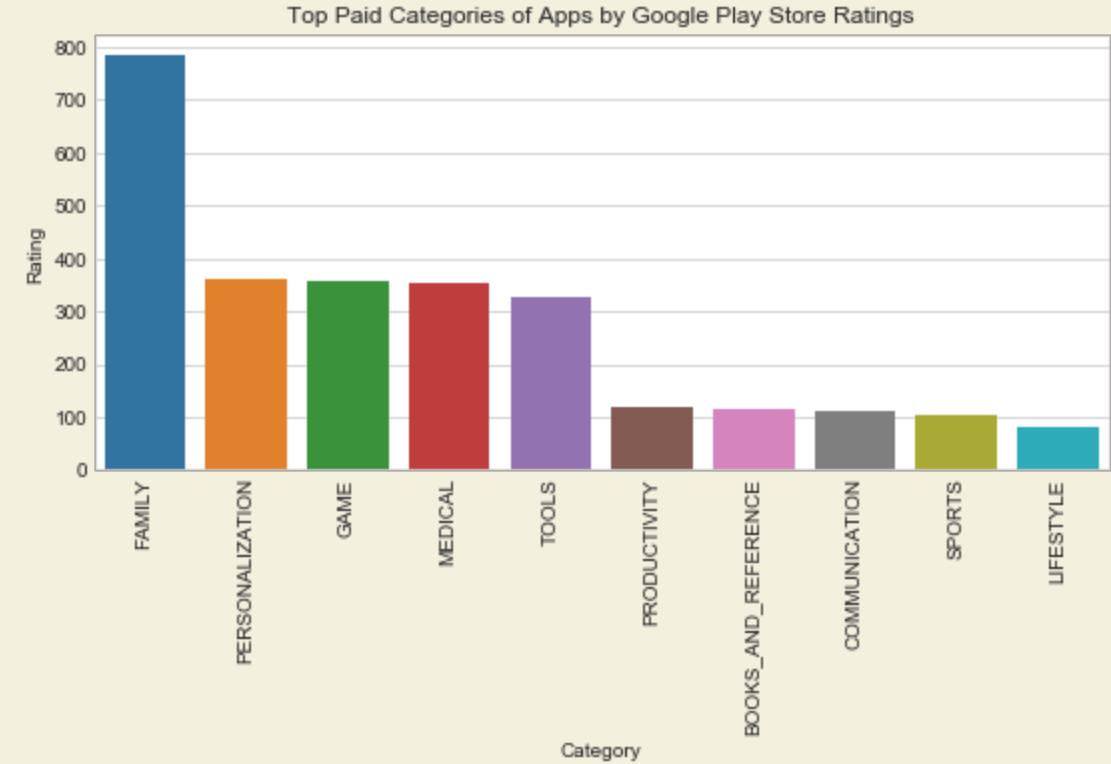
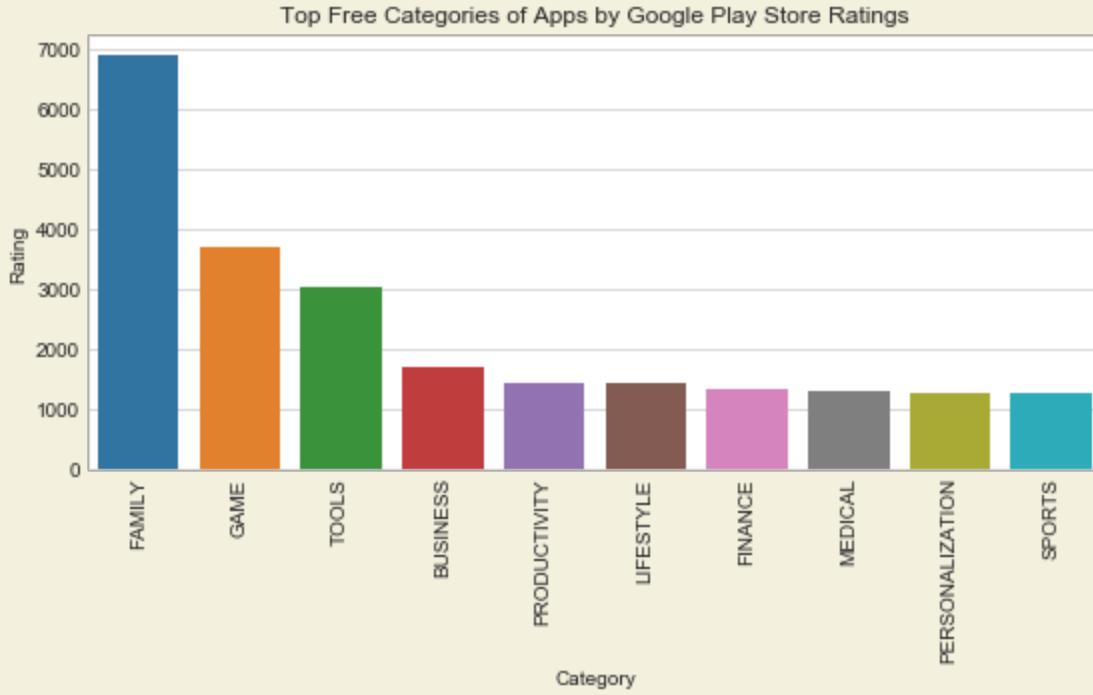
Rating Strategy



- Family applications is the top rated applications.



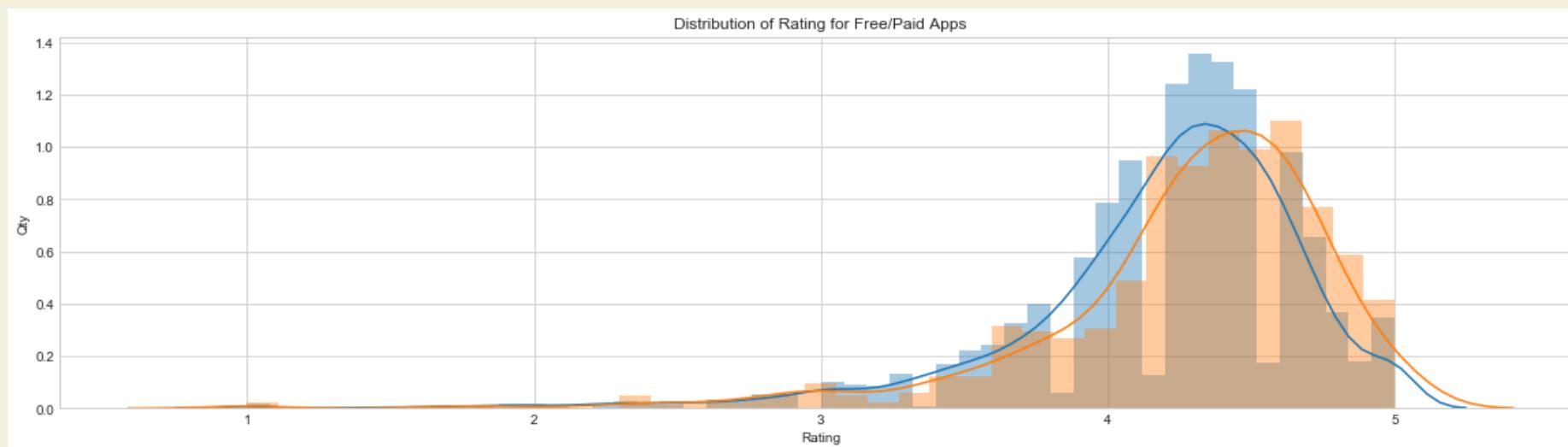
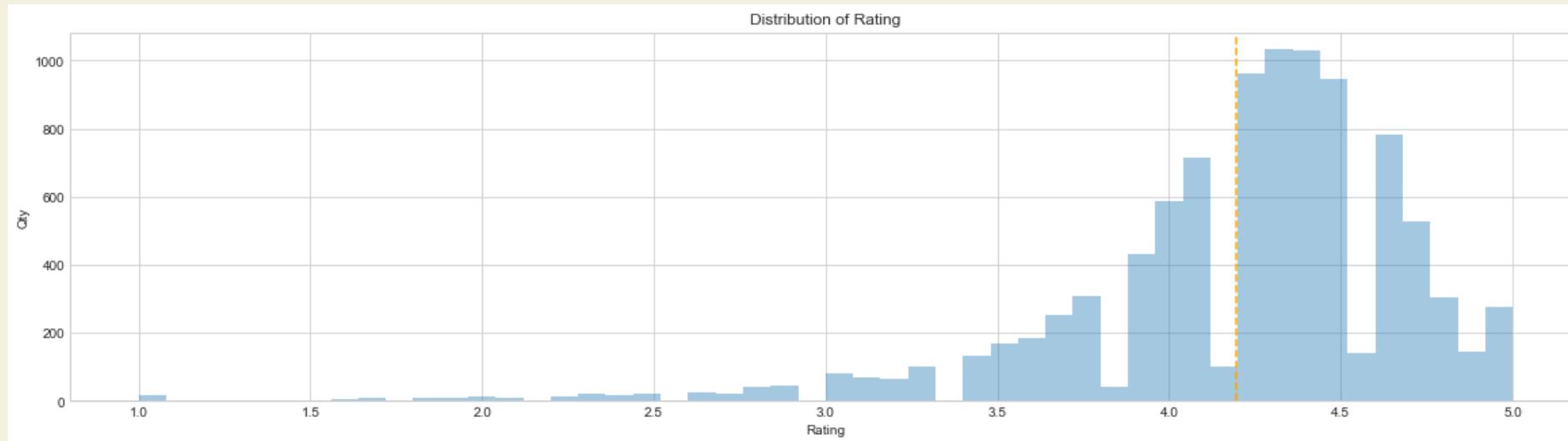
Rating Strategy



- Family applications are the top rated applications both free and paid.

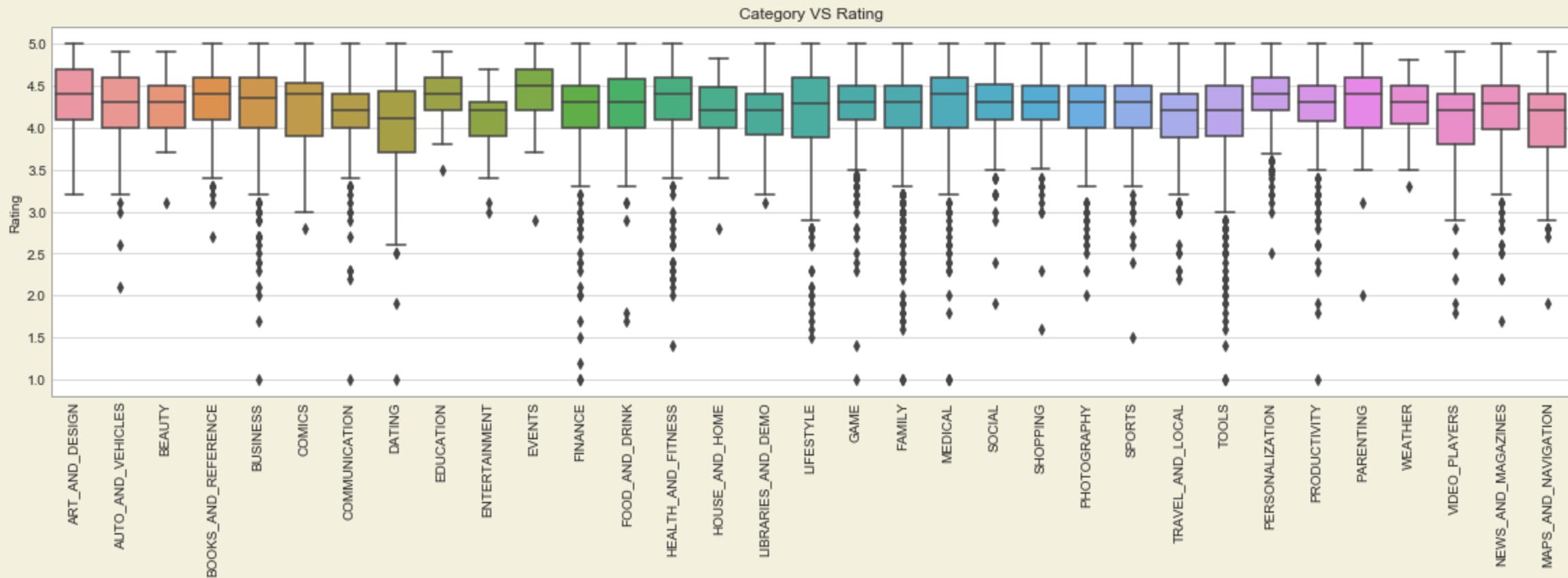


Rating Strategy





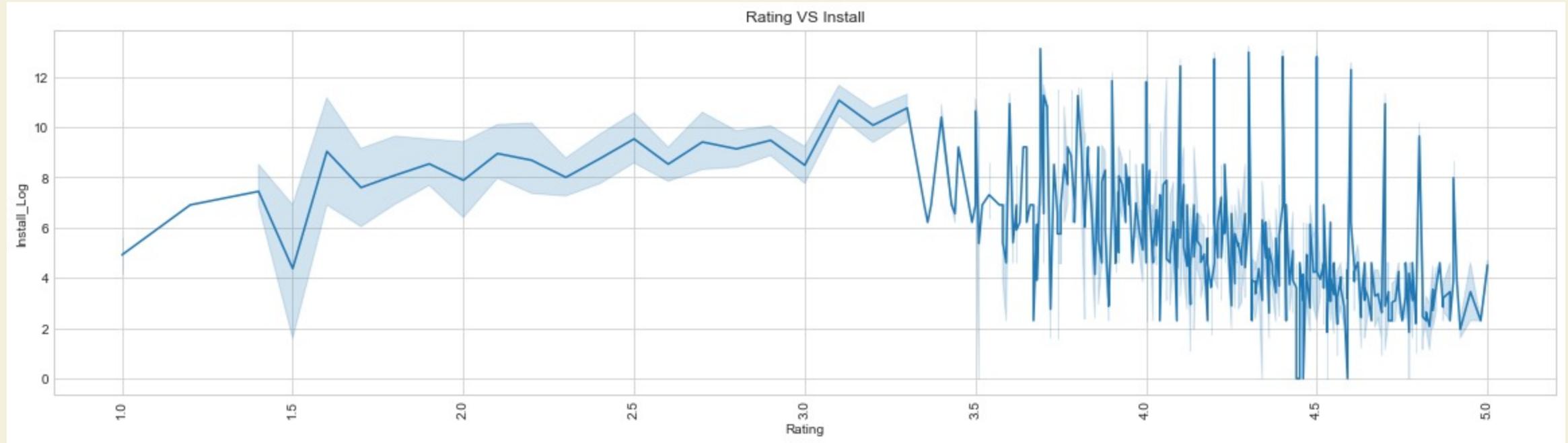
OVERVIEW OF ANALYSIS



- The rated for each category is not much different.
- Most category has rated between 4 to 5 ratings.



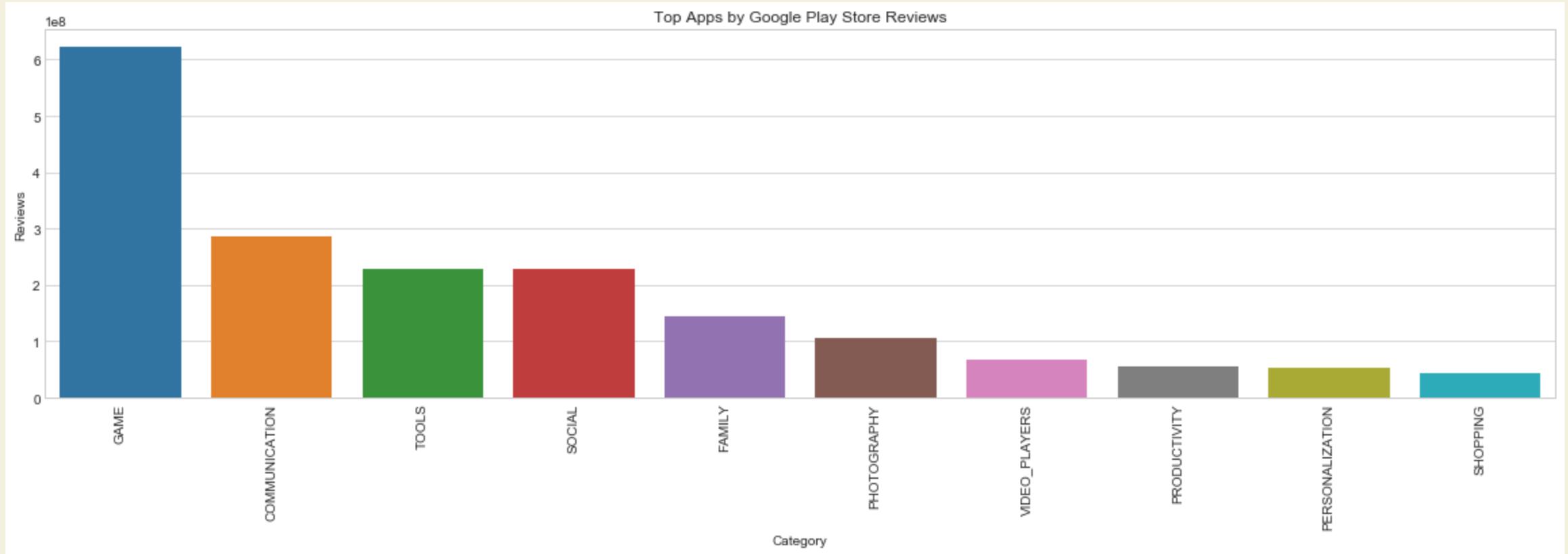
Rating Strategy



- High rating is not always high installed. (\neq effect between them)



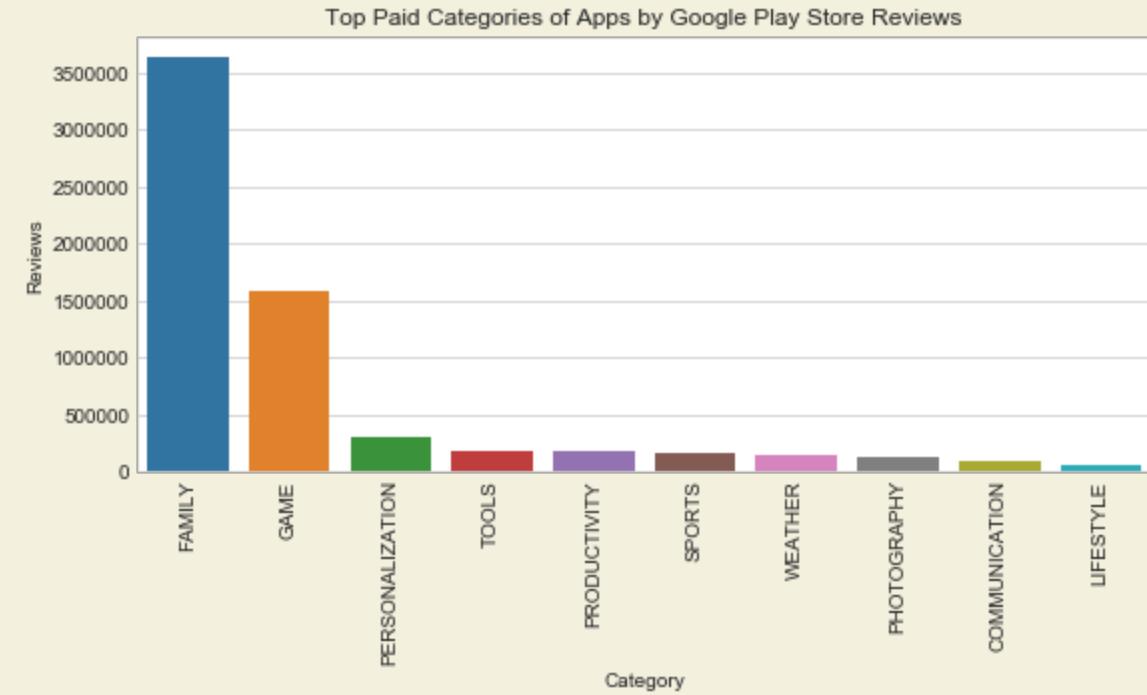
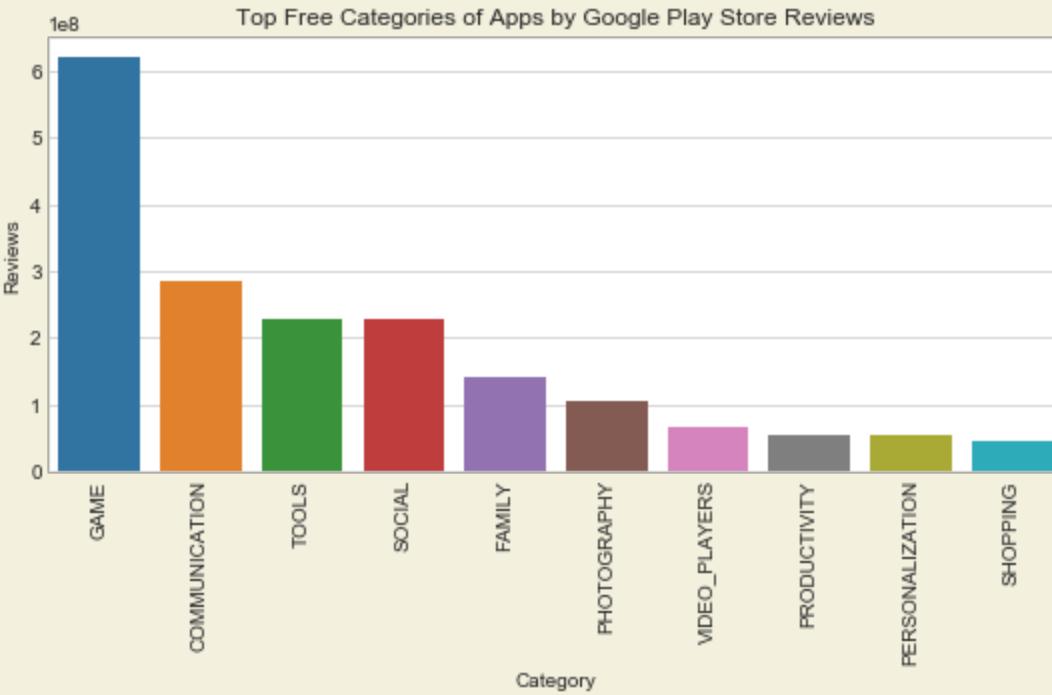
Review Strategy



- Game applications is the top reviewed applications.



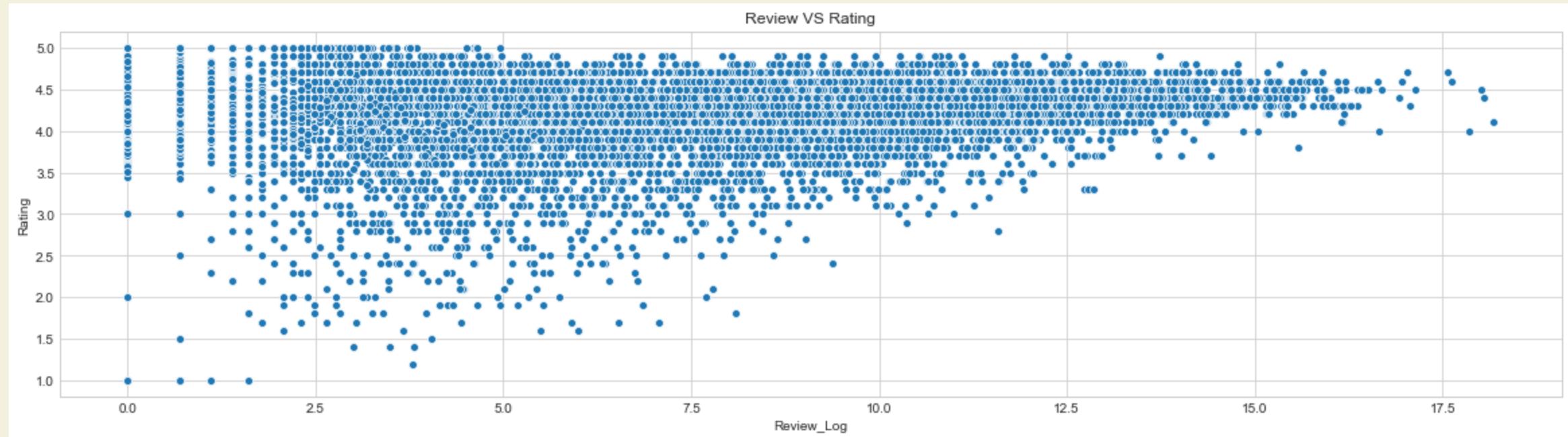
Review Strategy



- Family and Game are the most reviewed categories.



Review Strategy

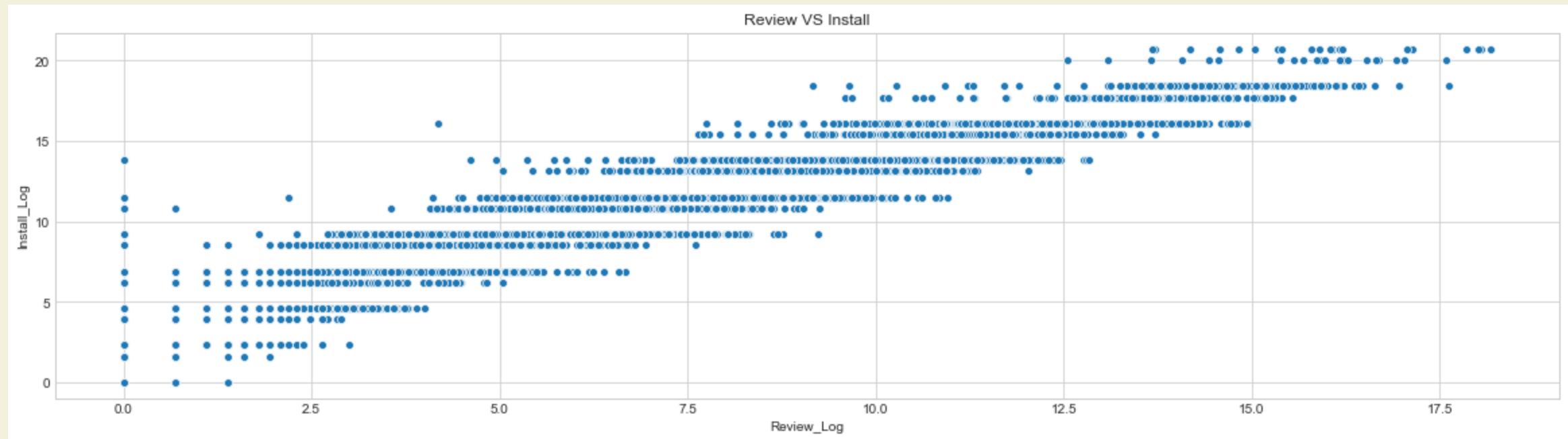


- Most applications have high rating, also have good reviews.

[[Rating VS Reviews in each category]]

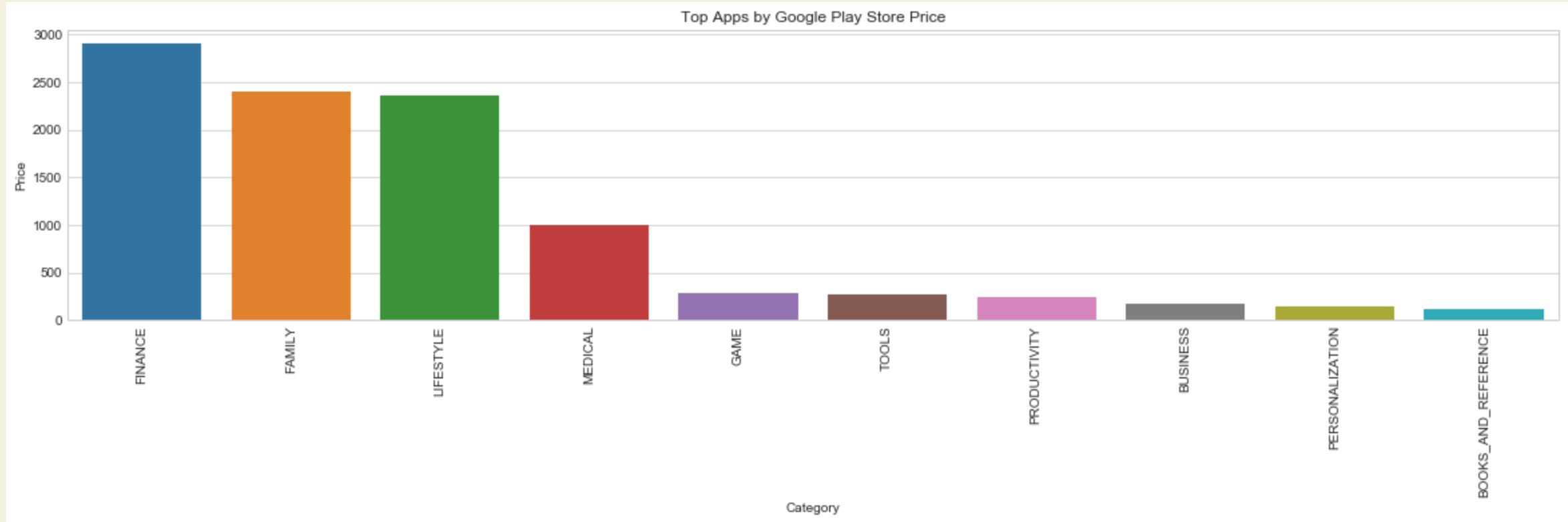


Review Strategy





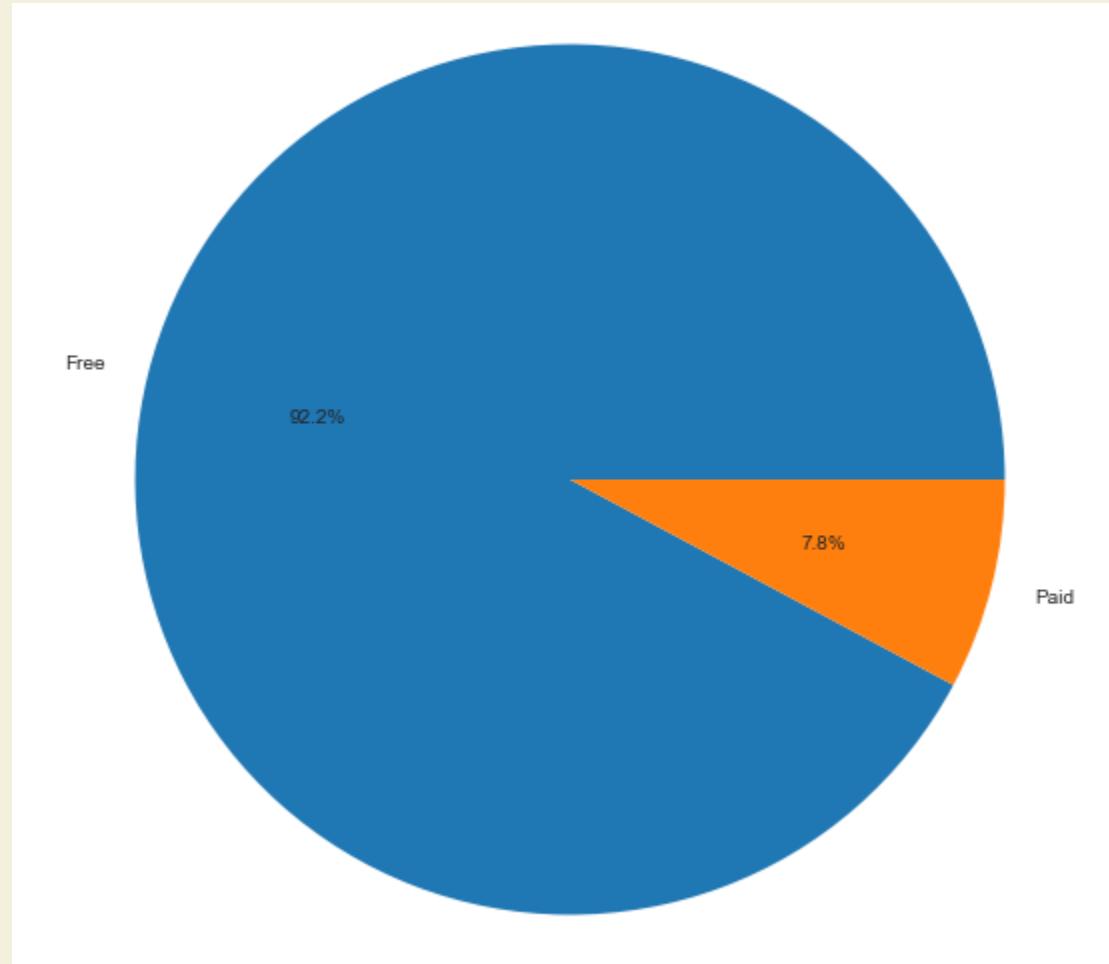
Price Strategy



- Finance applications have high price.

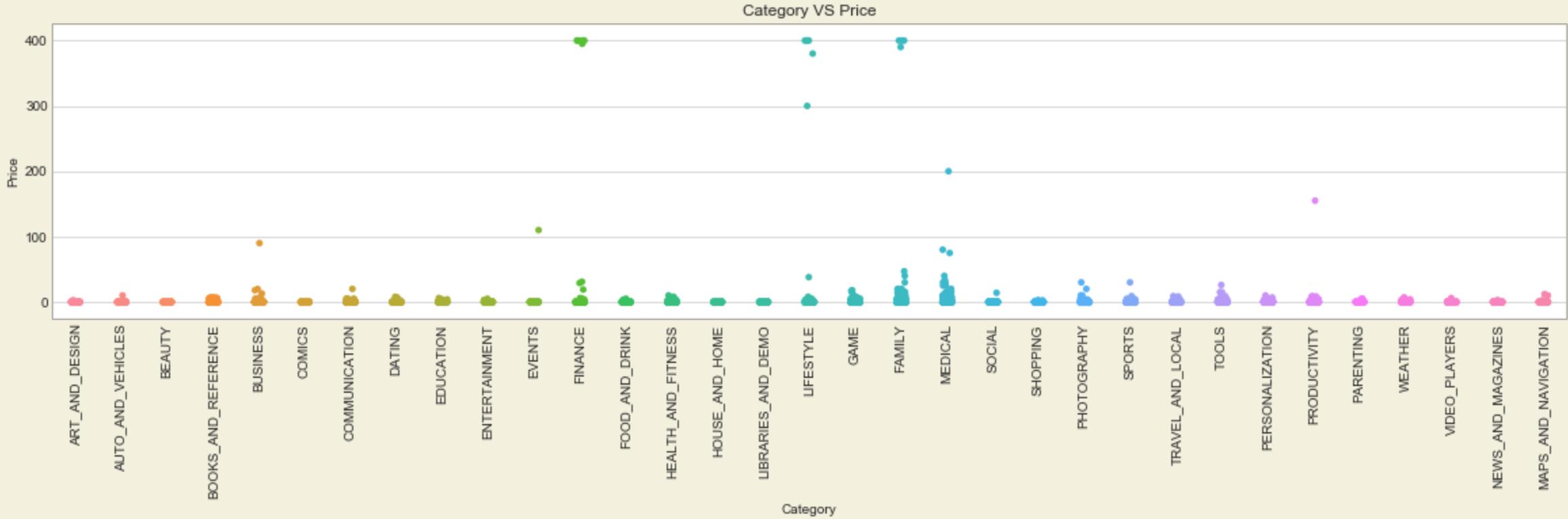


Price Strategy





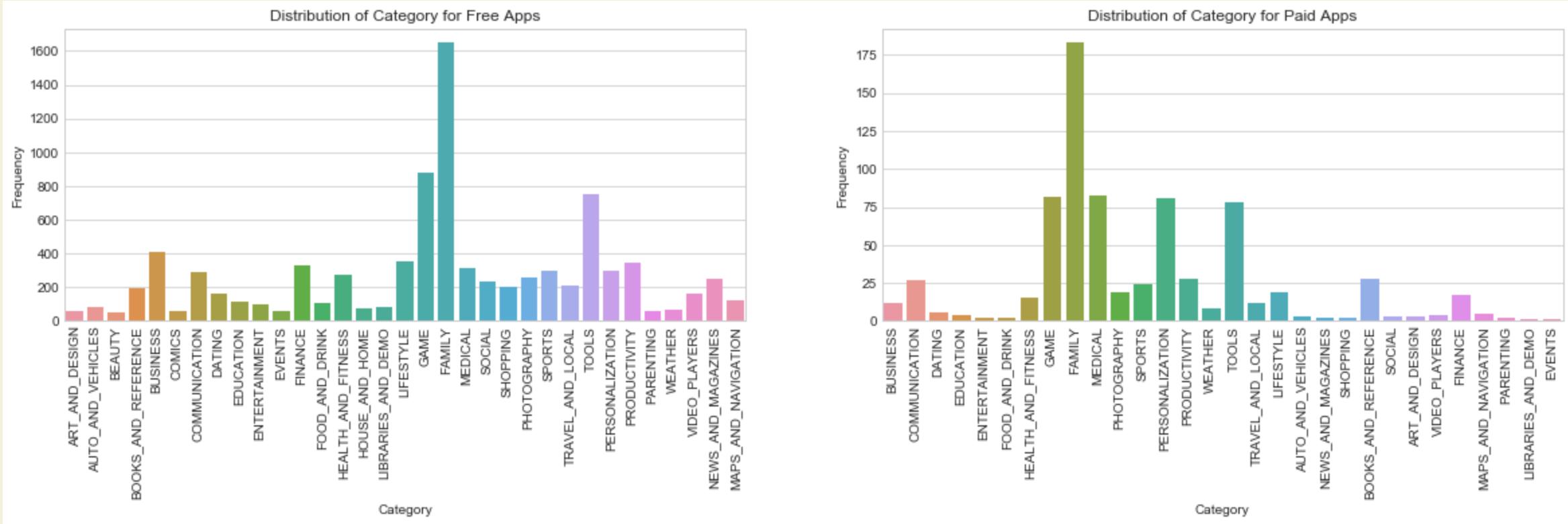
Price Strategy



- Few applications have high price.

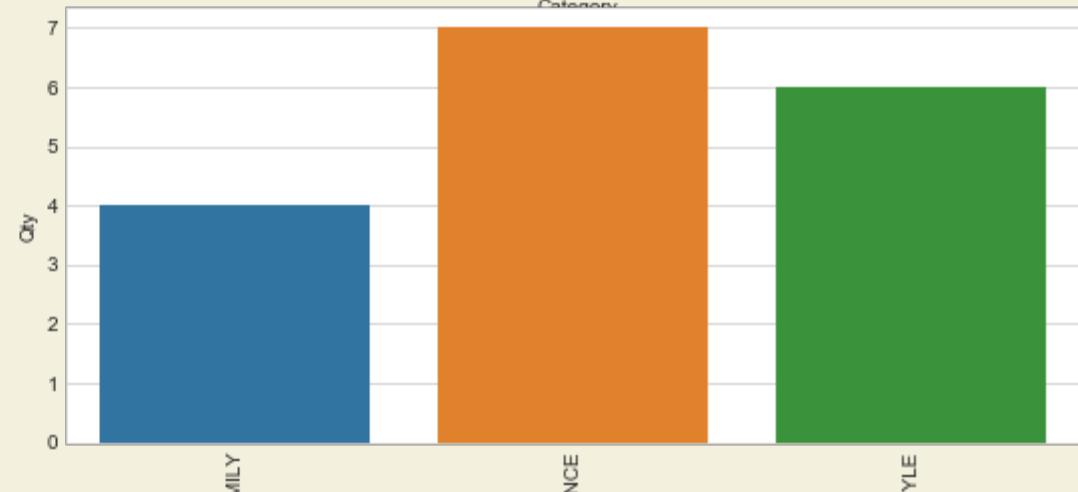
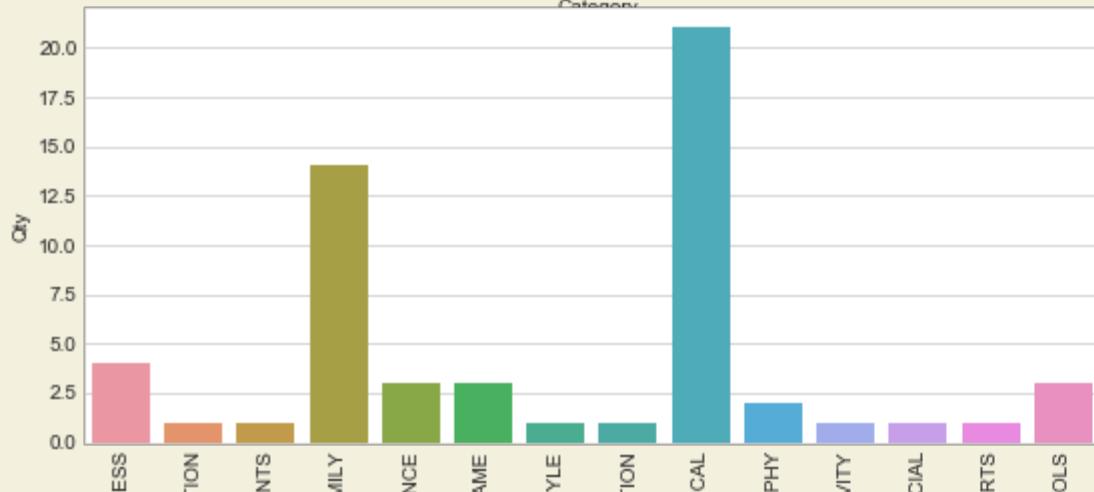
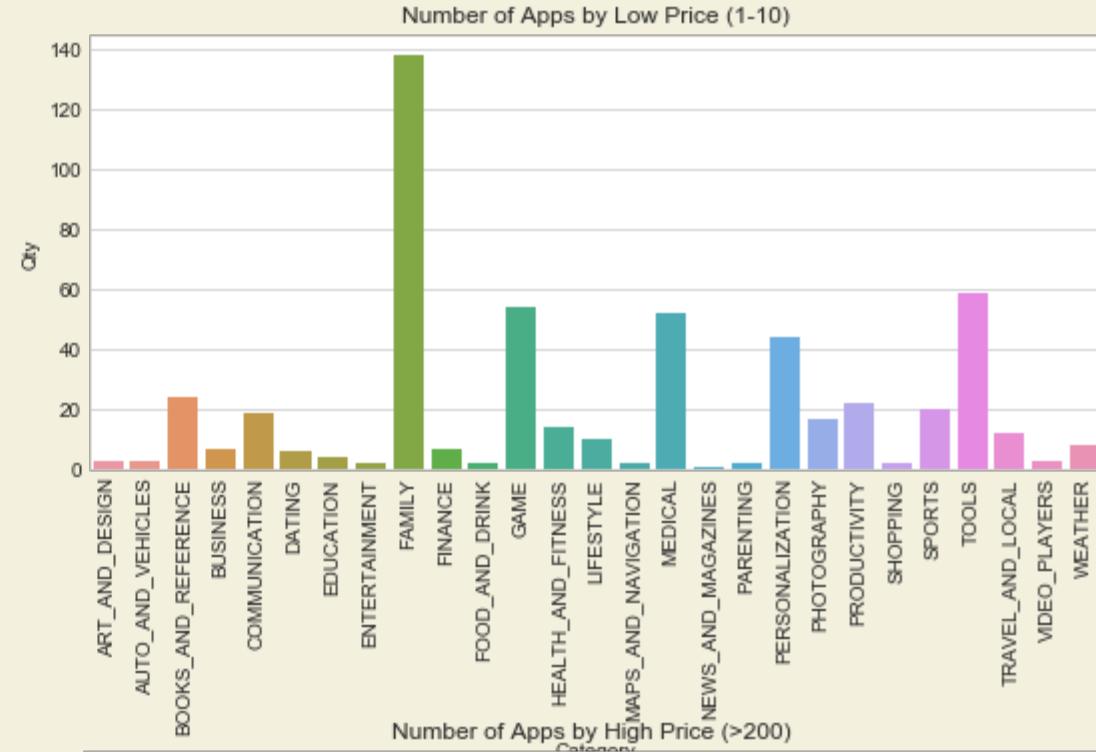
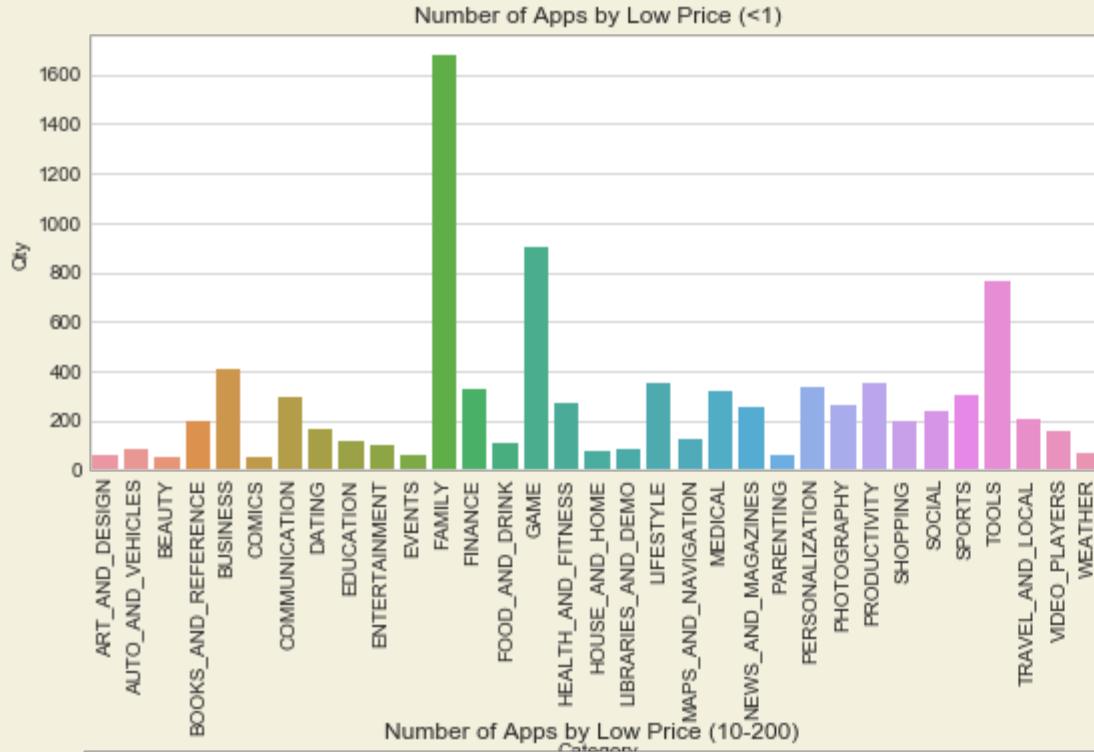


Price Strategy





Price Strategy





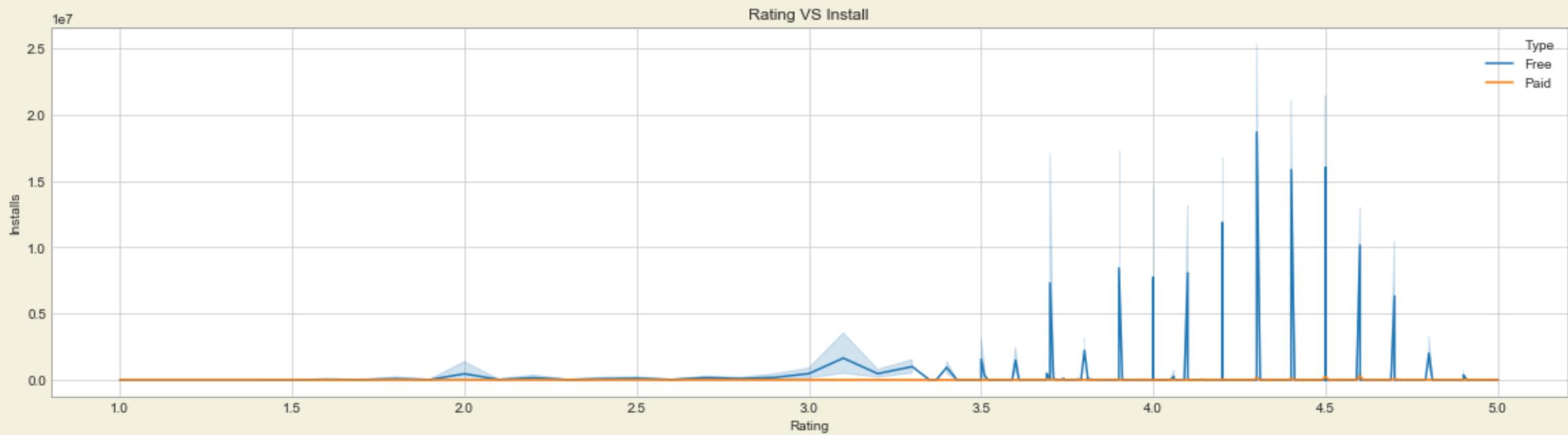
Price Strategy

	App	Category	Installs	Price
336	WhatsApp Messenger	COMMUNICATION	1000000000	0.00
152	Google Play Books	BOOKS_AND_REFERENCE	1000000000	0.00
1654	Subway Surfers	GAME	1000000000	0.00
3117	Maps - Navigate & Explore	TRAVEL_AND_LOCAL	1000000000	0.00
340	Gmail	COMMUNICATION	1000000000	0.00
...
9917	Eu Sou Rico	FINANCE	0	394.99
6692	cronometra-br	PRODUCTIVITY	0	154.99
5486	AP Series Solution Pro	FAMILY	0	1.99
9719	EP Cook Book	MEDICAL	0	200.00
9148	Command & Conquer: Rivals	FAMILY	0	0.00

[[Top paid apps in each category]]

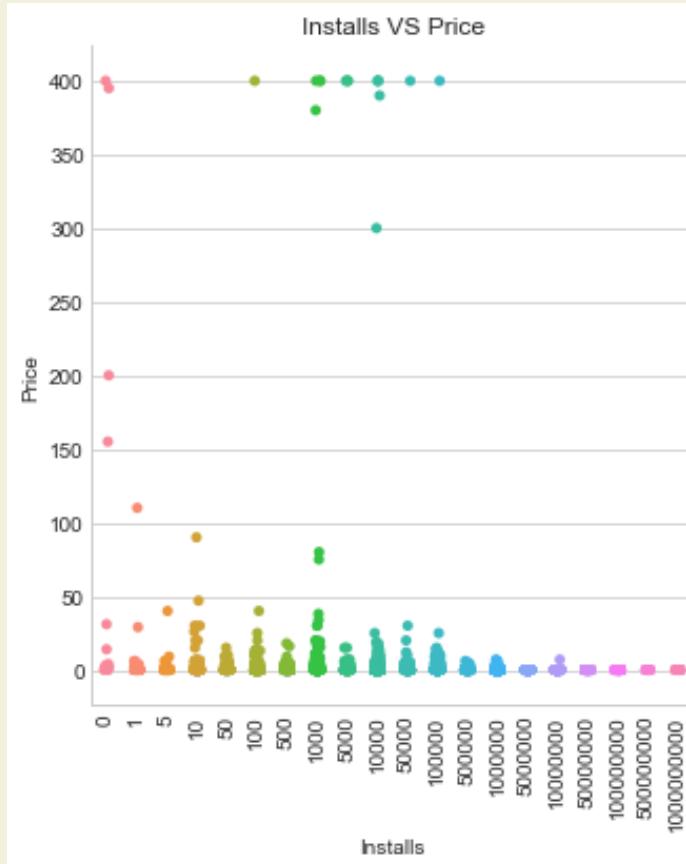


Price Strategy





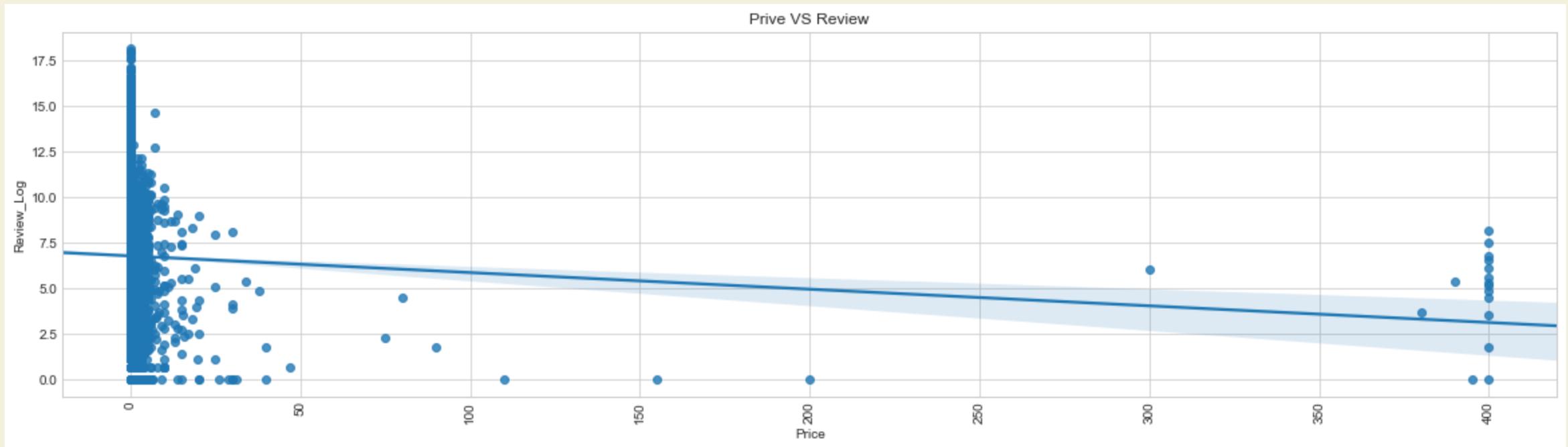
Price Strategy



- High price applications are less installed.



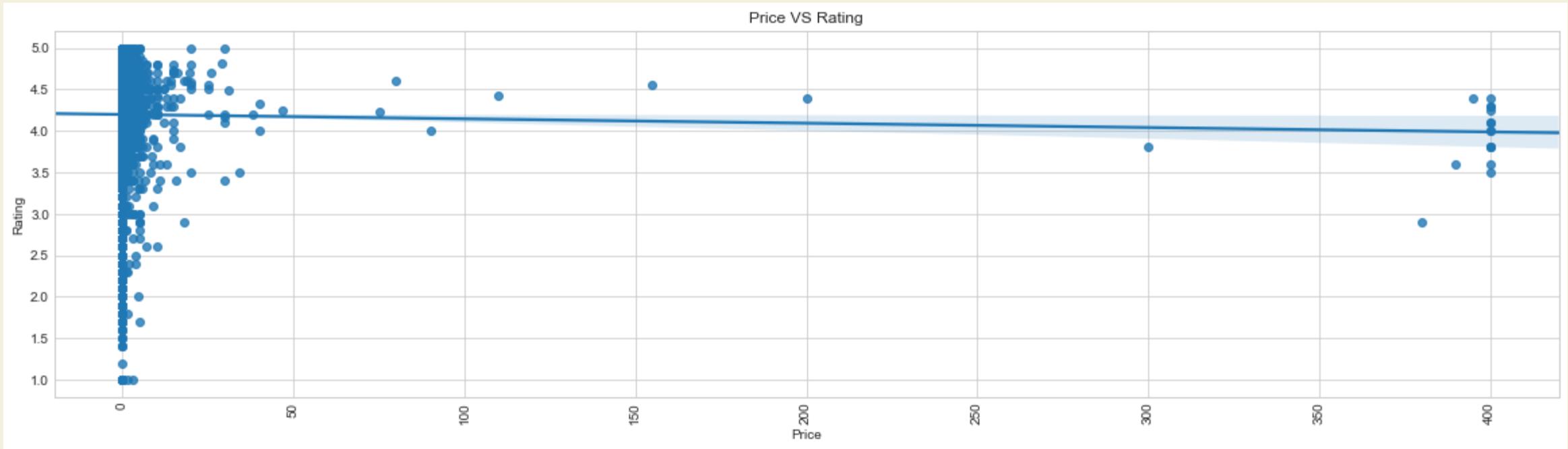
Price Strategy



- High price applications have not get high reviewed.



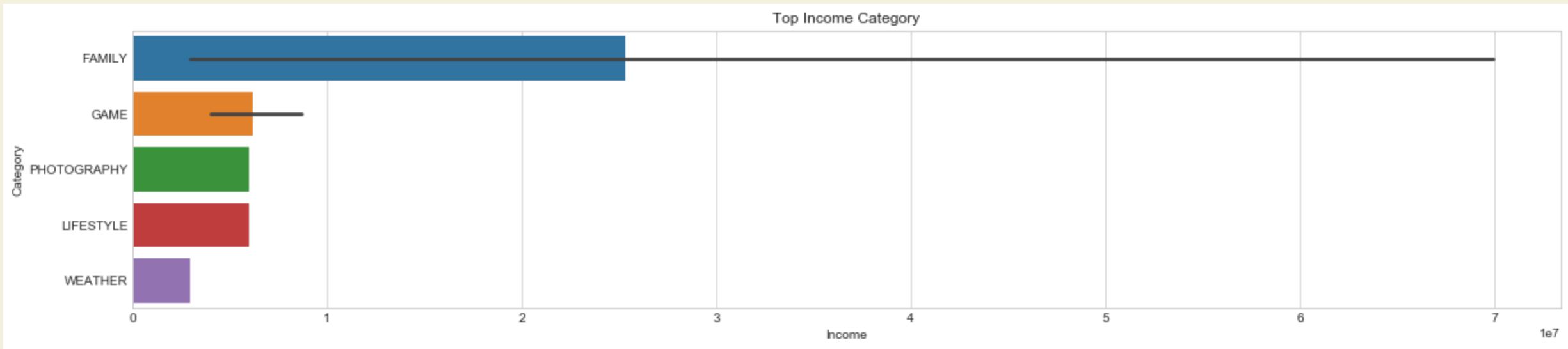
Price Strategy



- High price applications may not match user expectations, then the rated applications are a bit low.



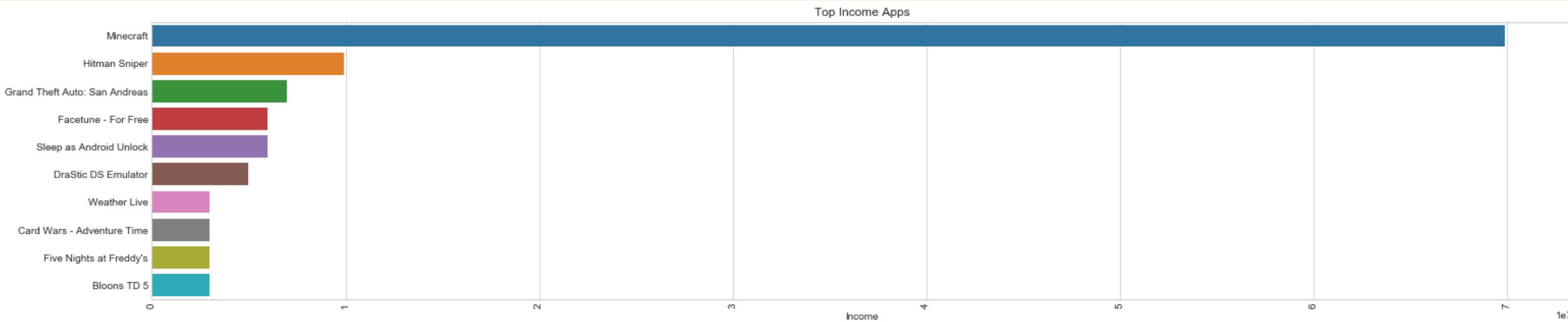
Price Strategy



- Income = install x price



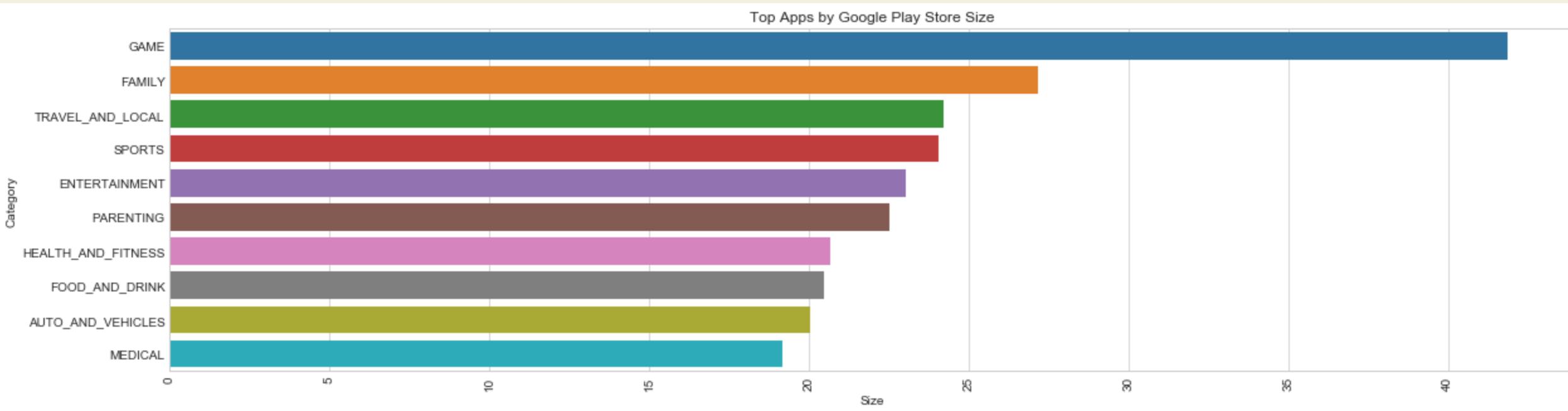
Price Strategy



[[Top Income of Apps in each category]]



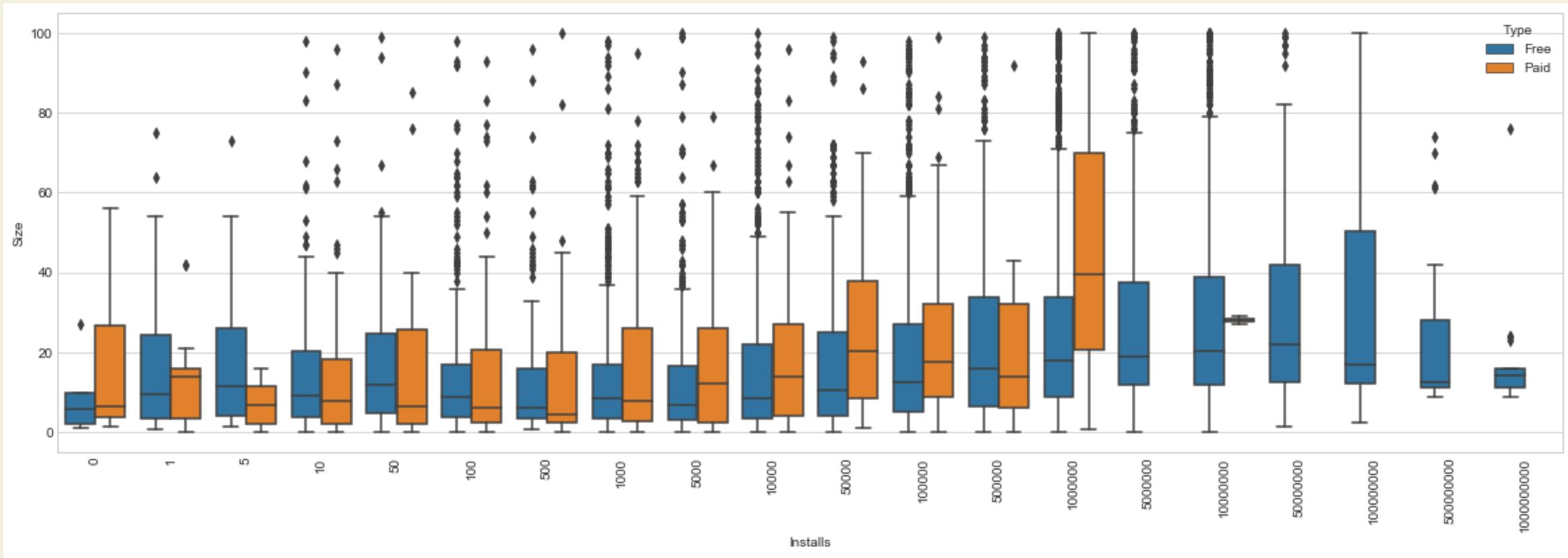
Size Strategy



- Game applications have the largest size compared to other application.



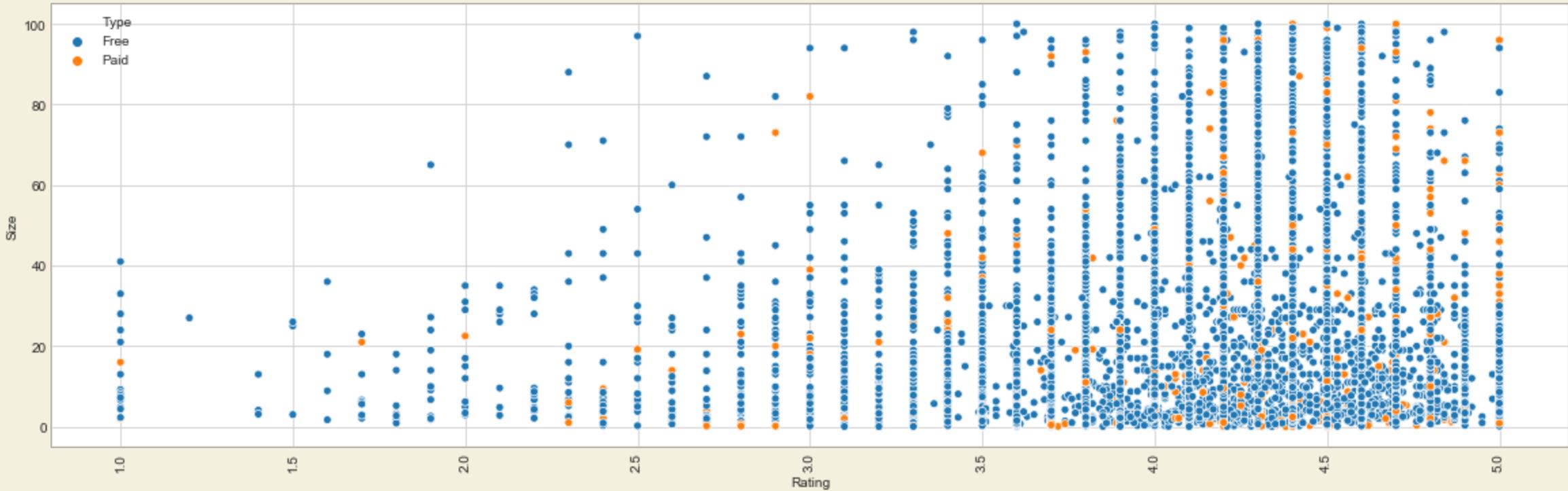
Price Strategy



- Average size is 20 MB for free applications and 19 MB for paid applications.
- More installed are increase, size also increases. (Installation depends on Size)
- Compared to free applications, paid applications have most installed in each size.



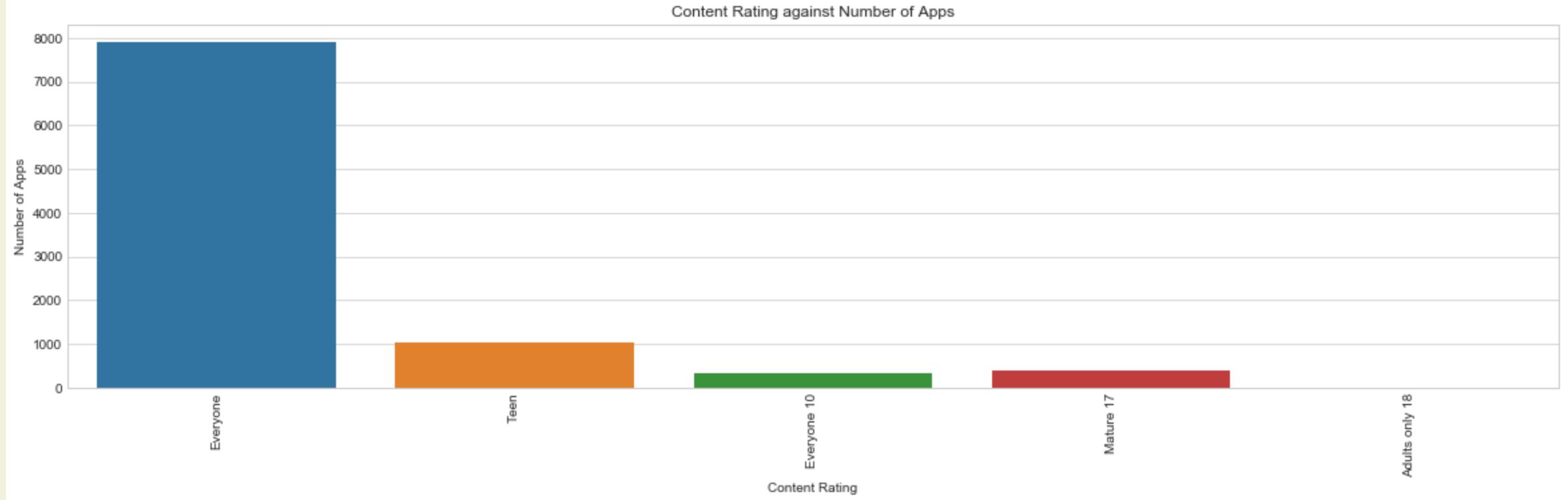
Price Strategy



- Applications have high rated, size also has bigger. (Rating depends on Size)
- Size between 0-20 MB get more score in 4-5 rated.



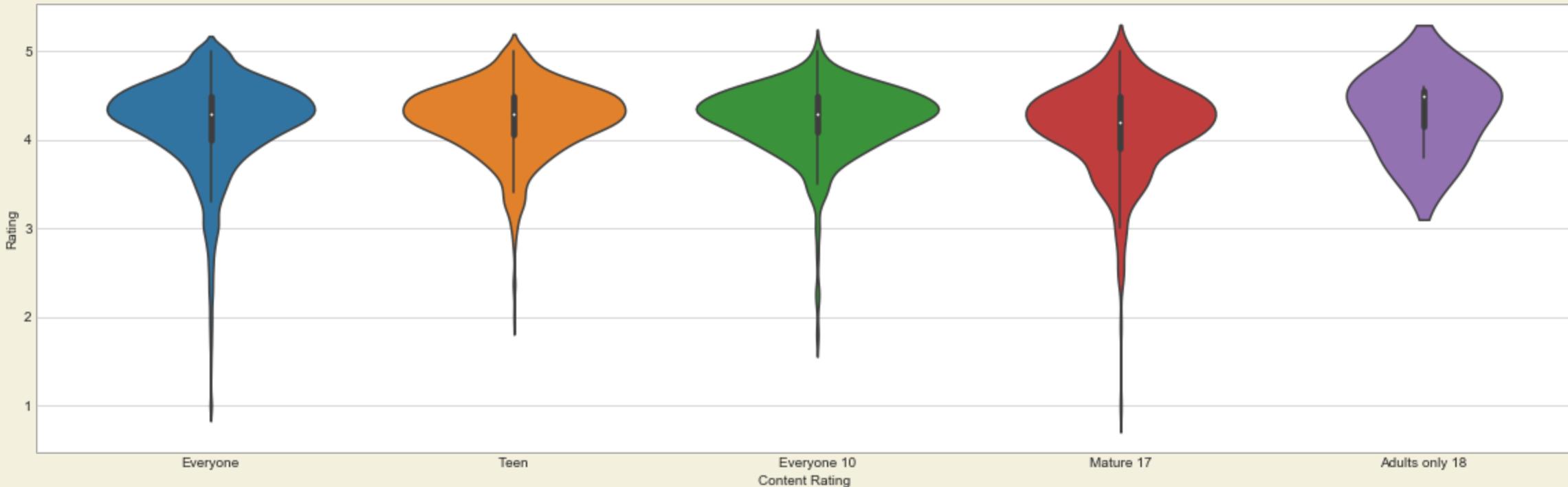
Content Rating Strategy



- Most of applications is developed for everyone.



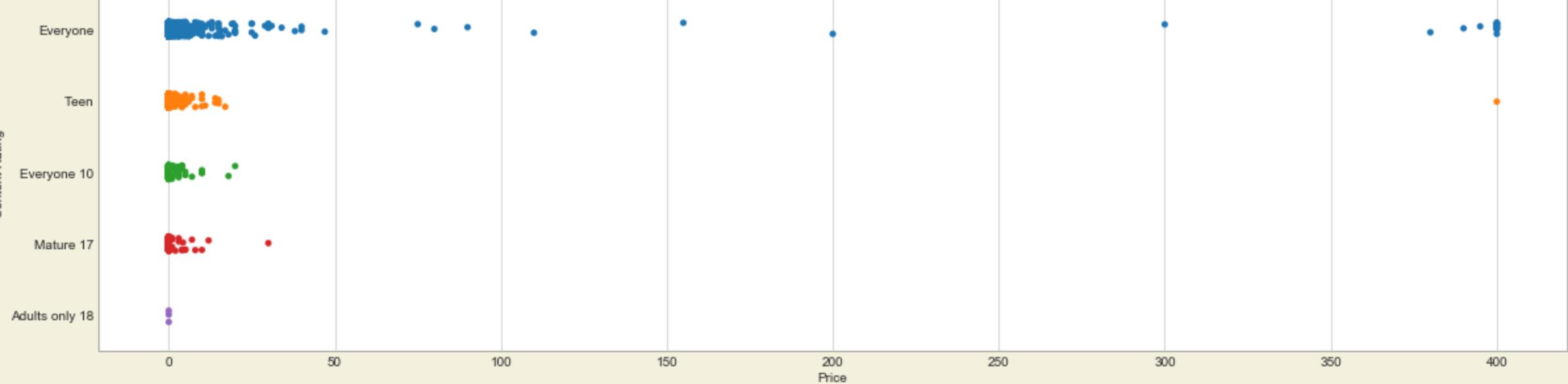
Content Rating Strategy



- Each content rating has high rating.



Content Rating Strategy

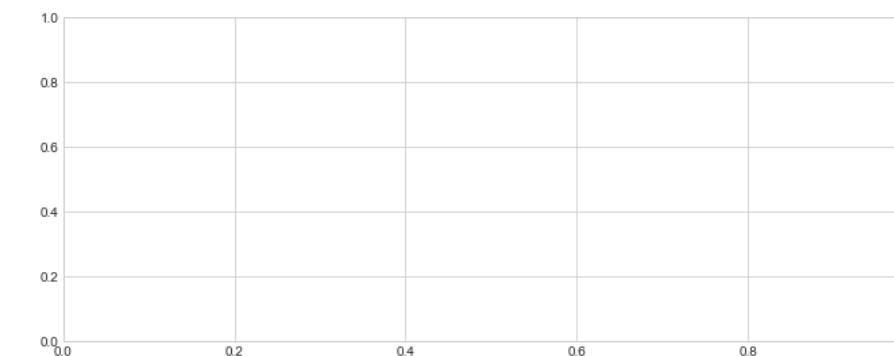
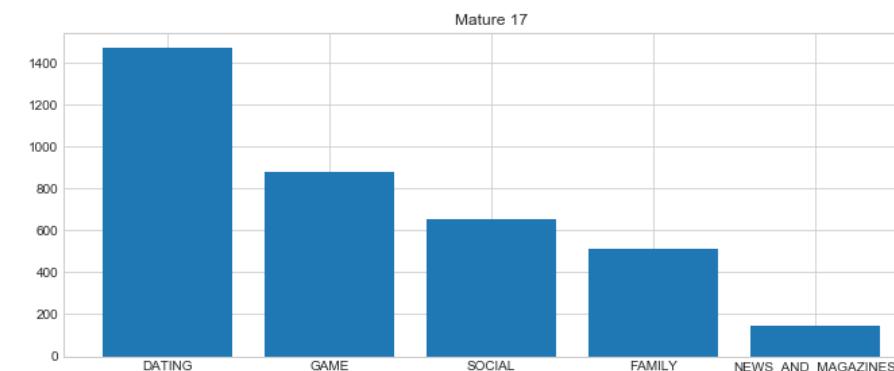
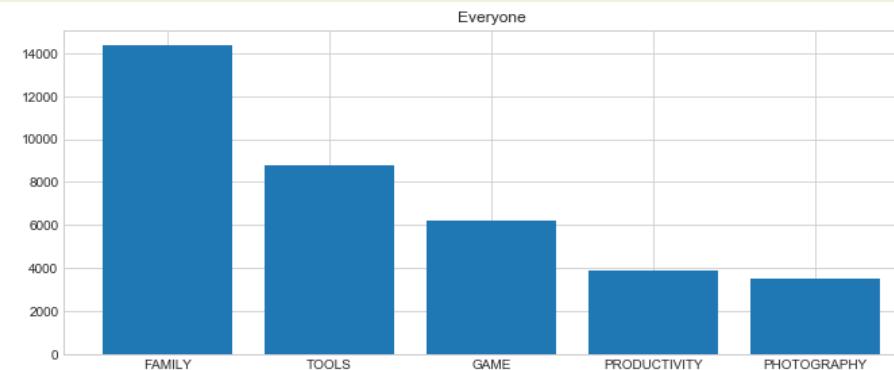
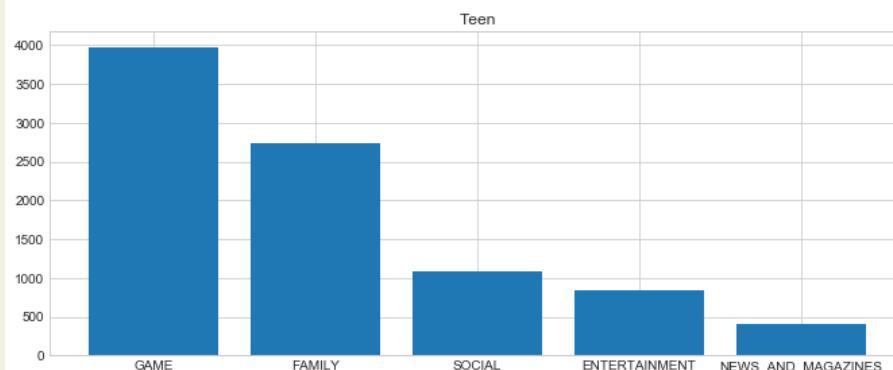
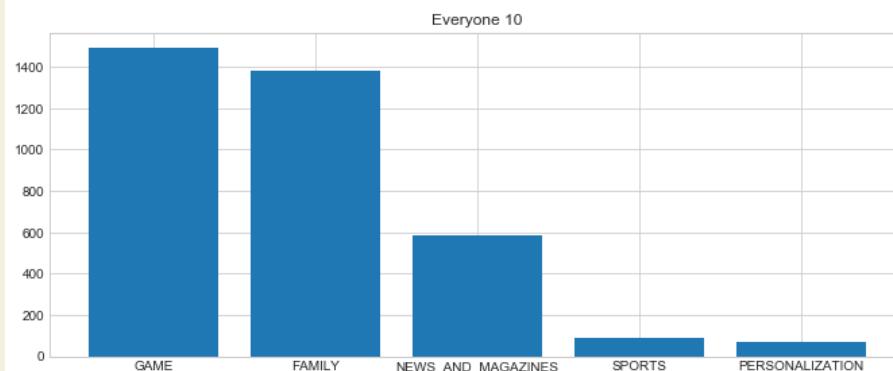
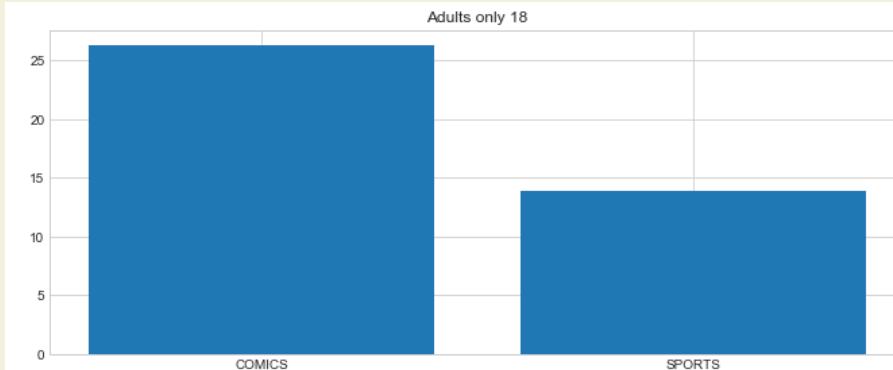


- Applications for everyone have higher price.

[[Size VS Rating in each category]]

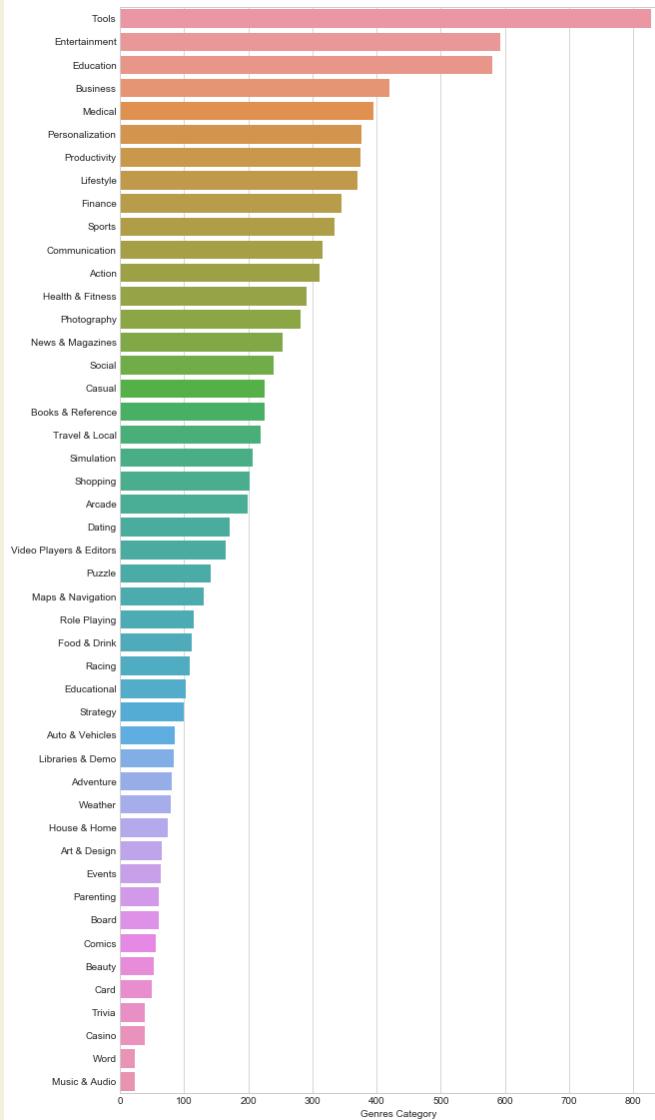


OVERVIEW OF ANALYSIS



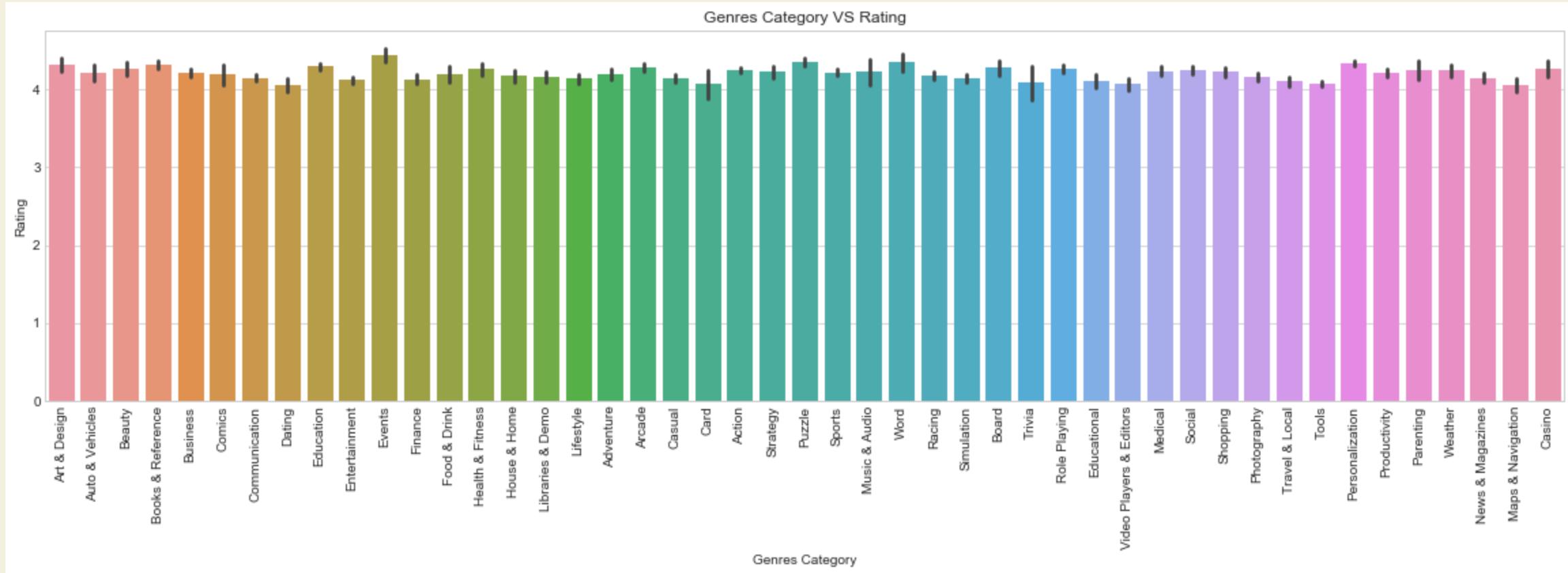


Genres Category Strategy



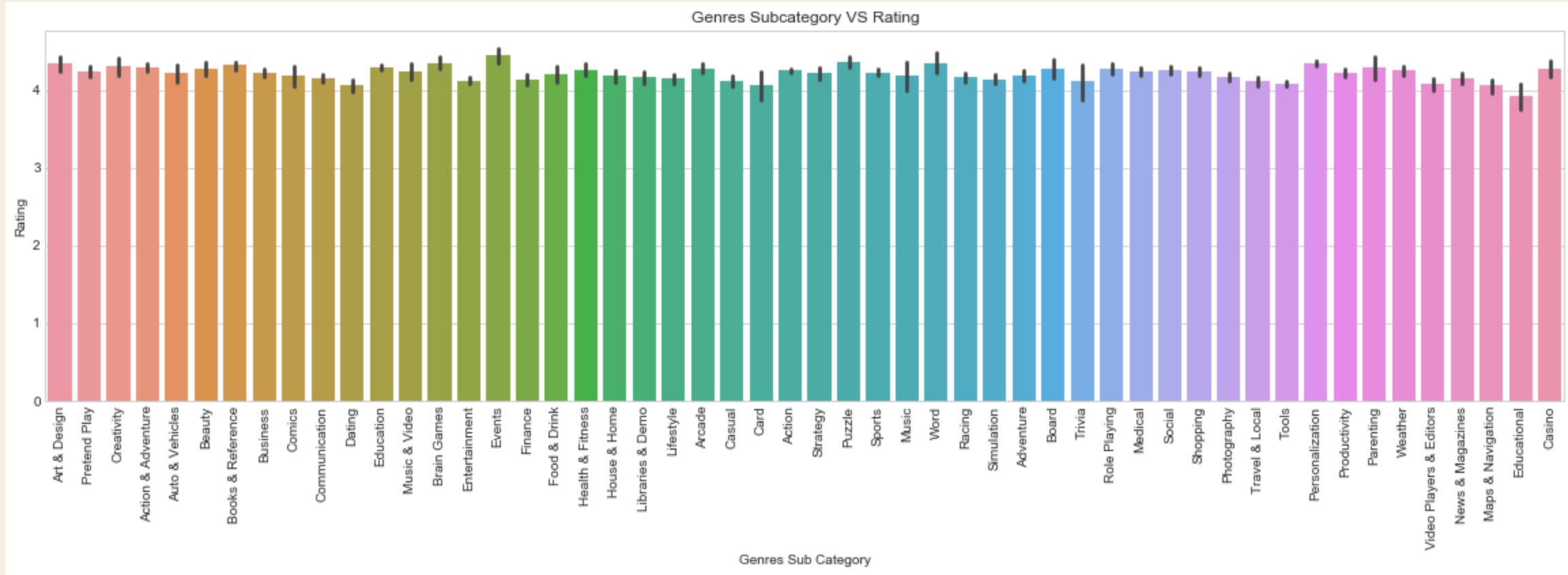


Genres Category Strategy



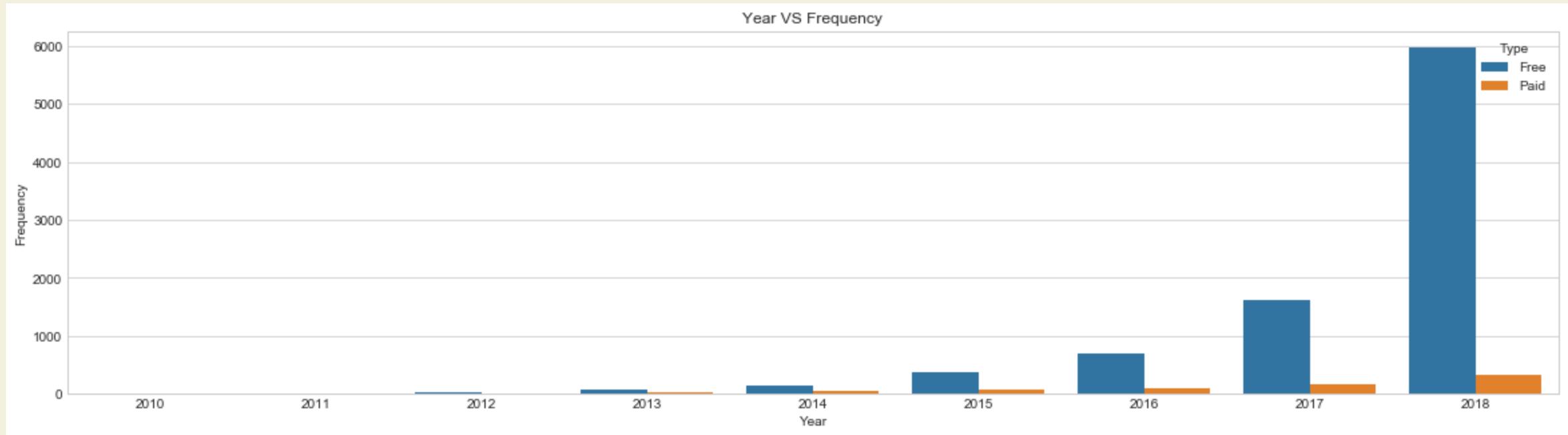


Genres Category Strategy





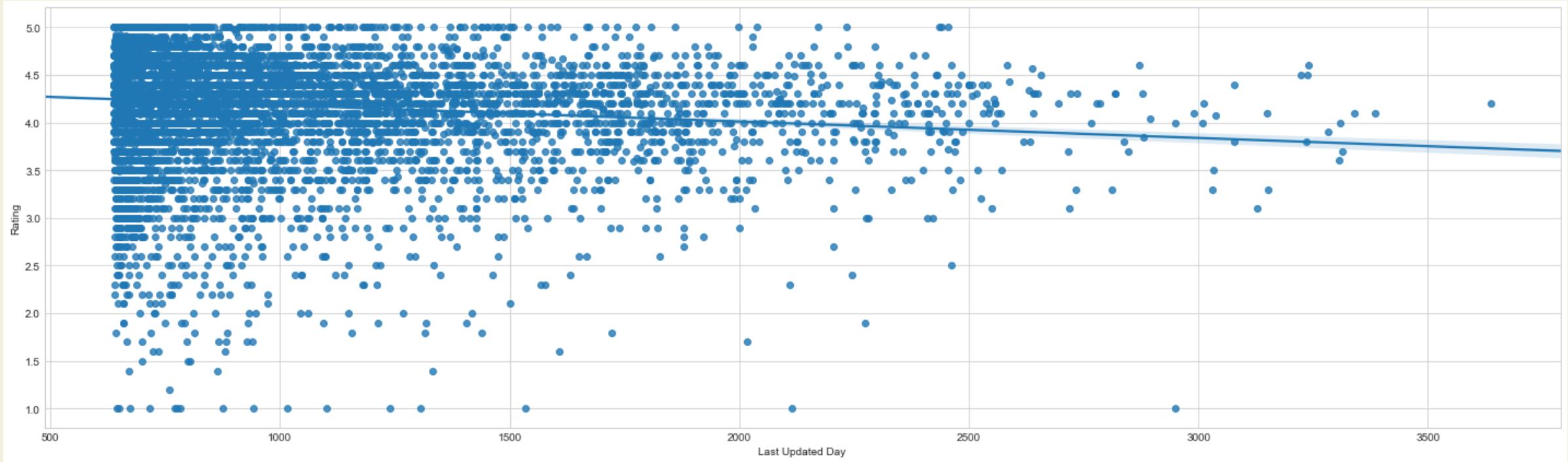
Last Updated Strategy



- Applications have increased update in every year.



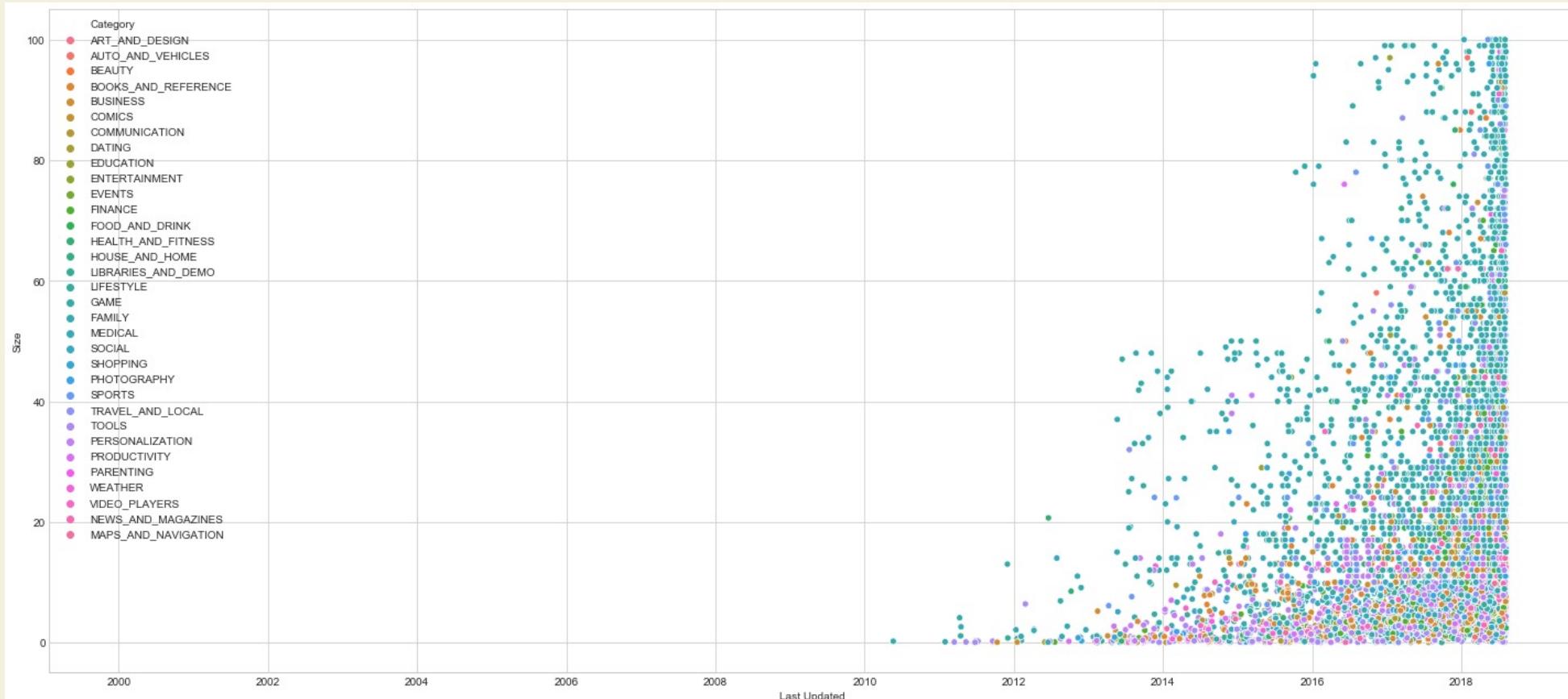
Last Updated Strategy



- Nearest updated applications are high ratings.



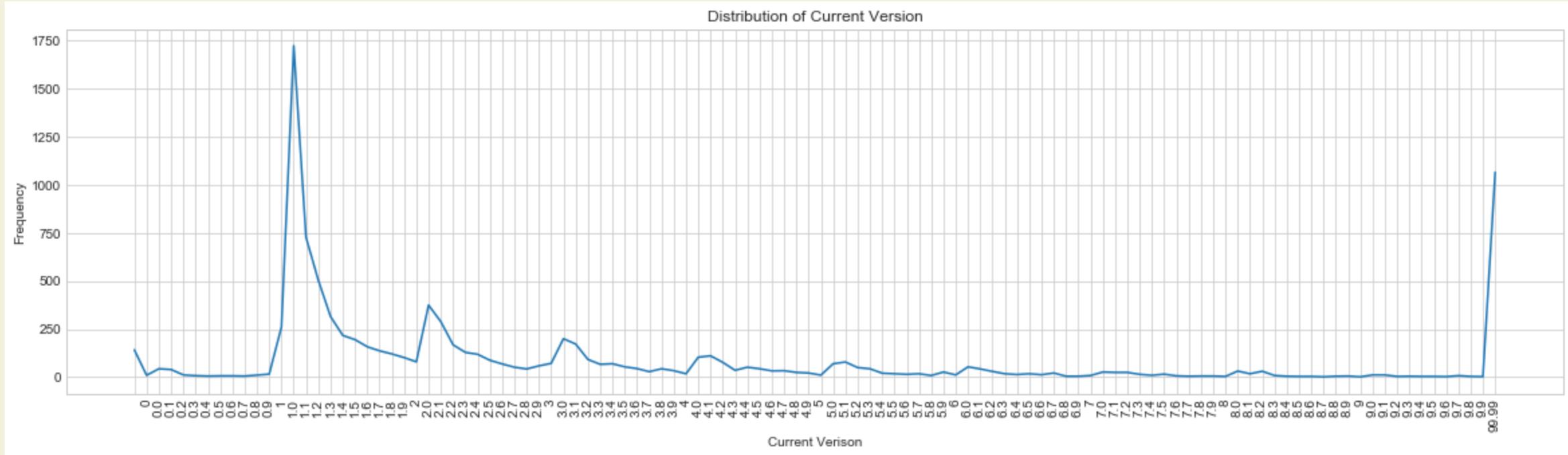
Last Updated Strategy



- Size of applications are increase in every year.

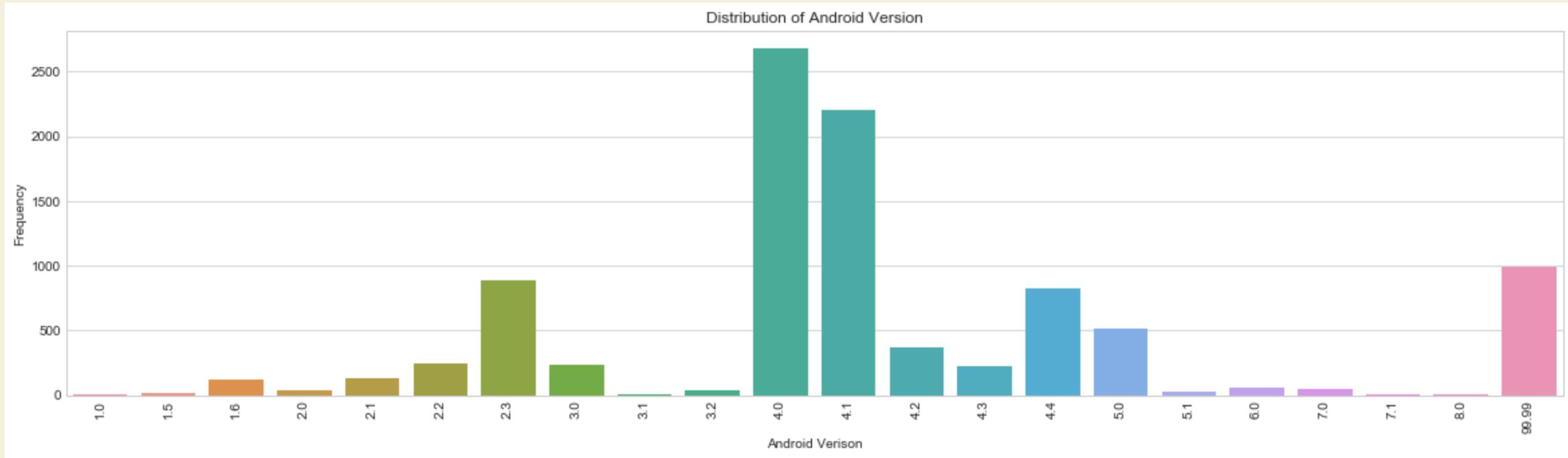


Current Version Strategy





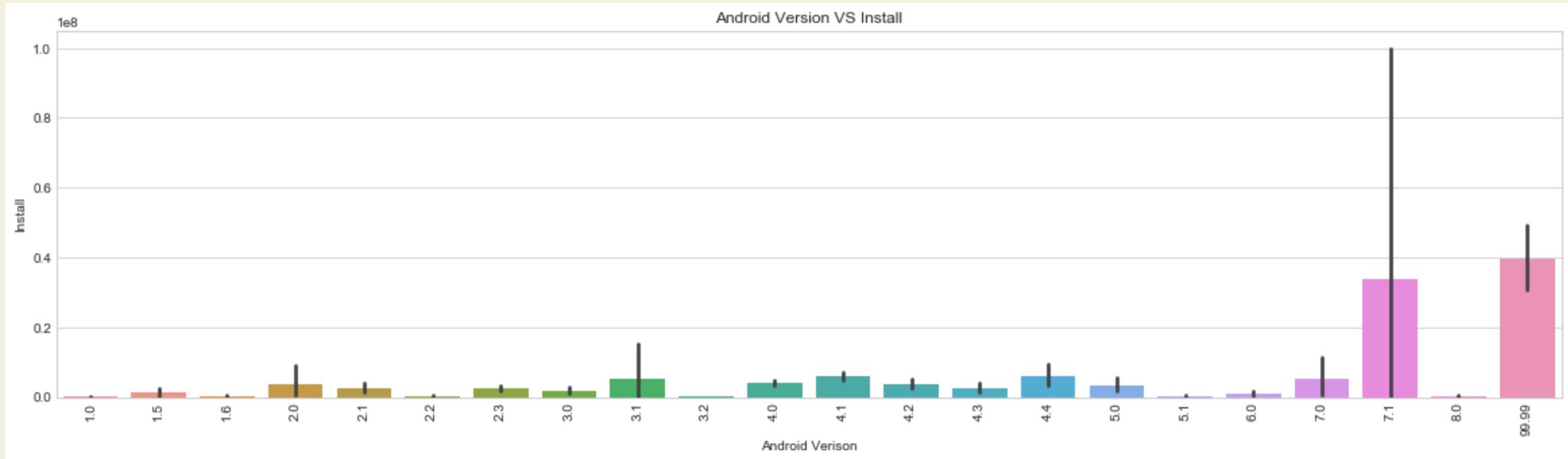
Android Version Strategy



- Most of applications are supported from android version 4.0 and above



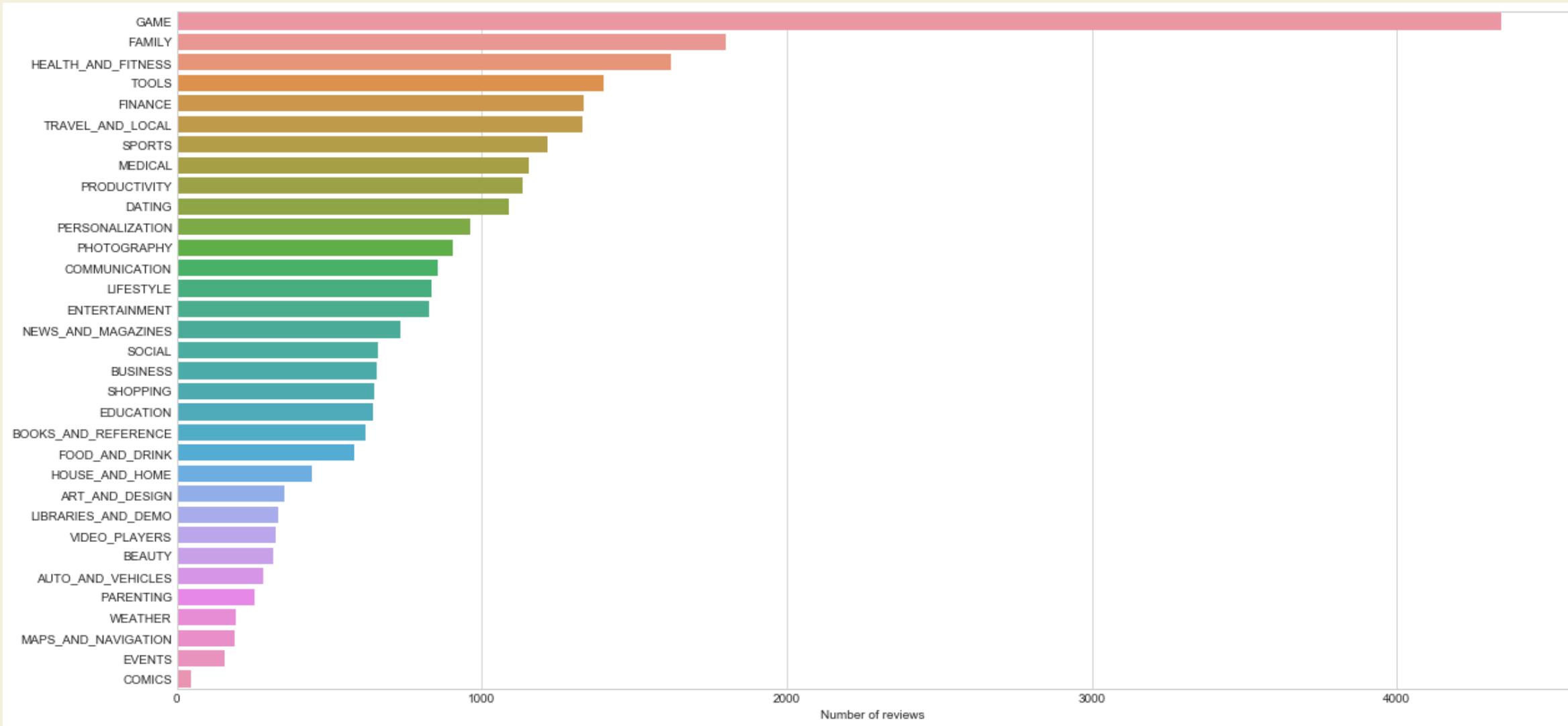
Android Version Strategy



- Most installed android version are not specific version.
- Android version 7.1 is second

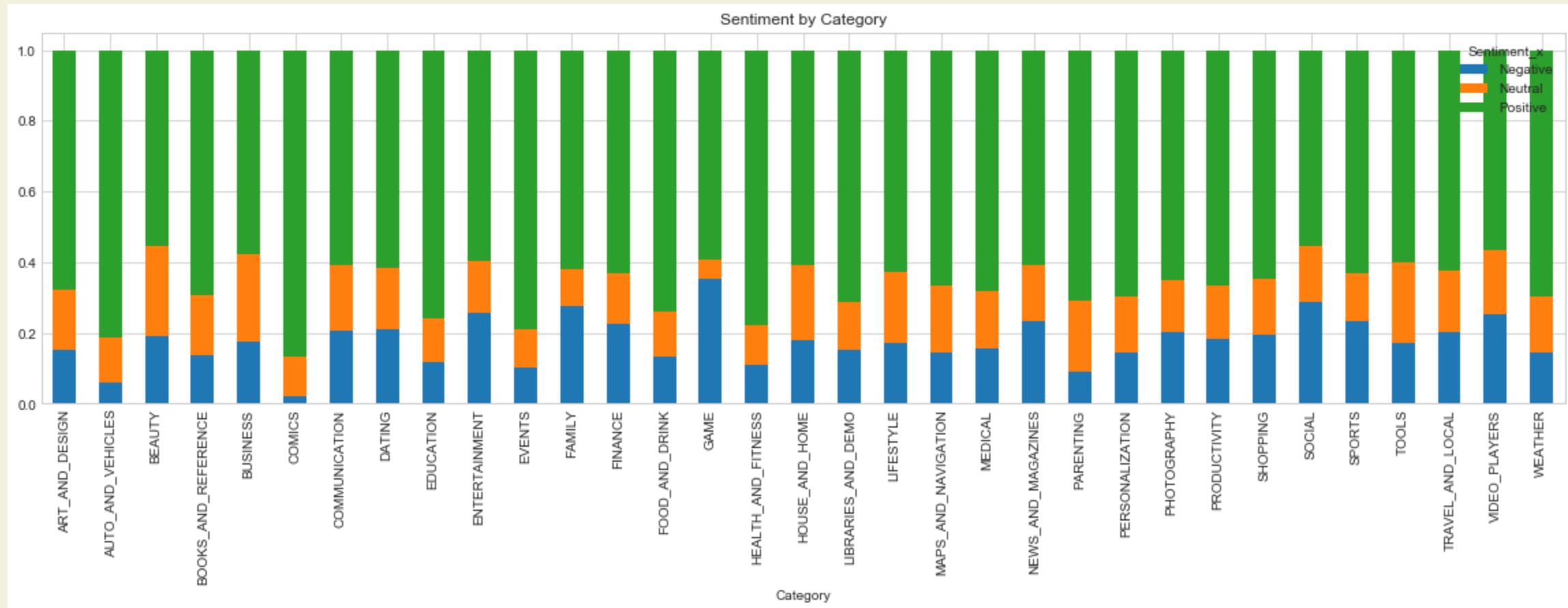


Sentiment Strategy



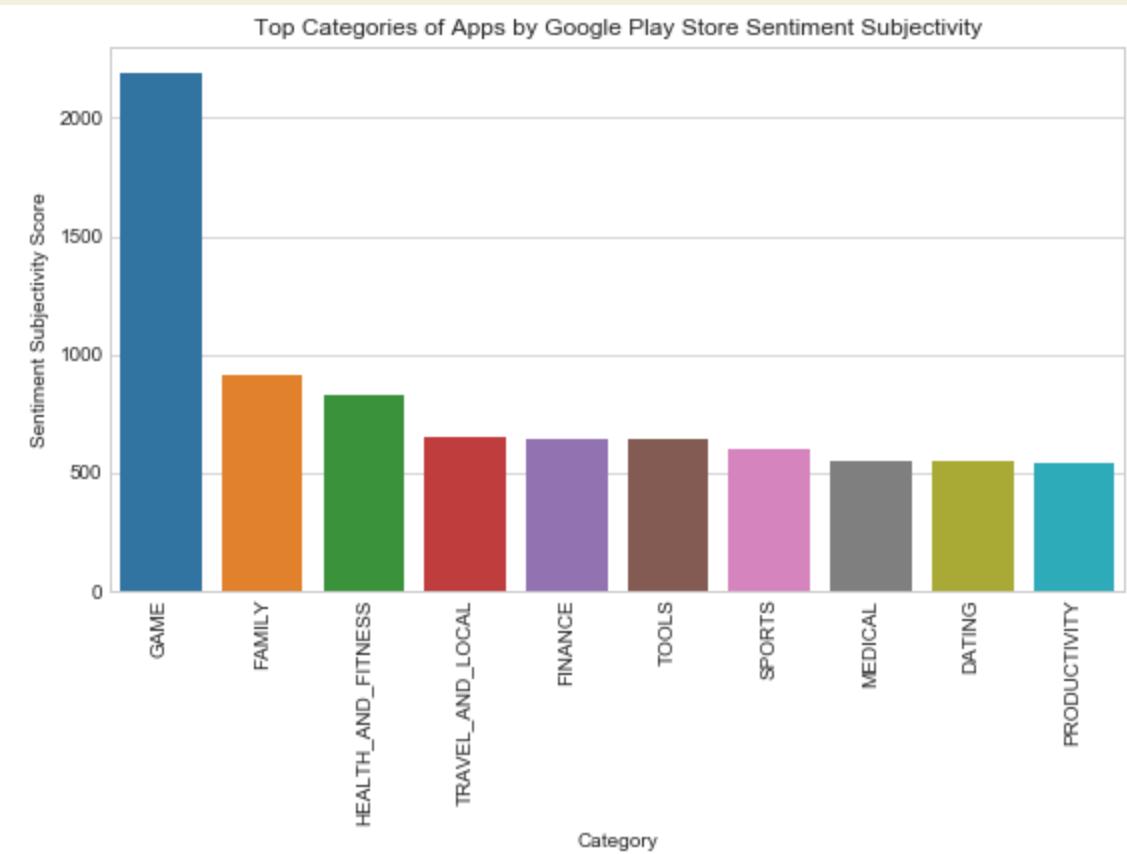
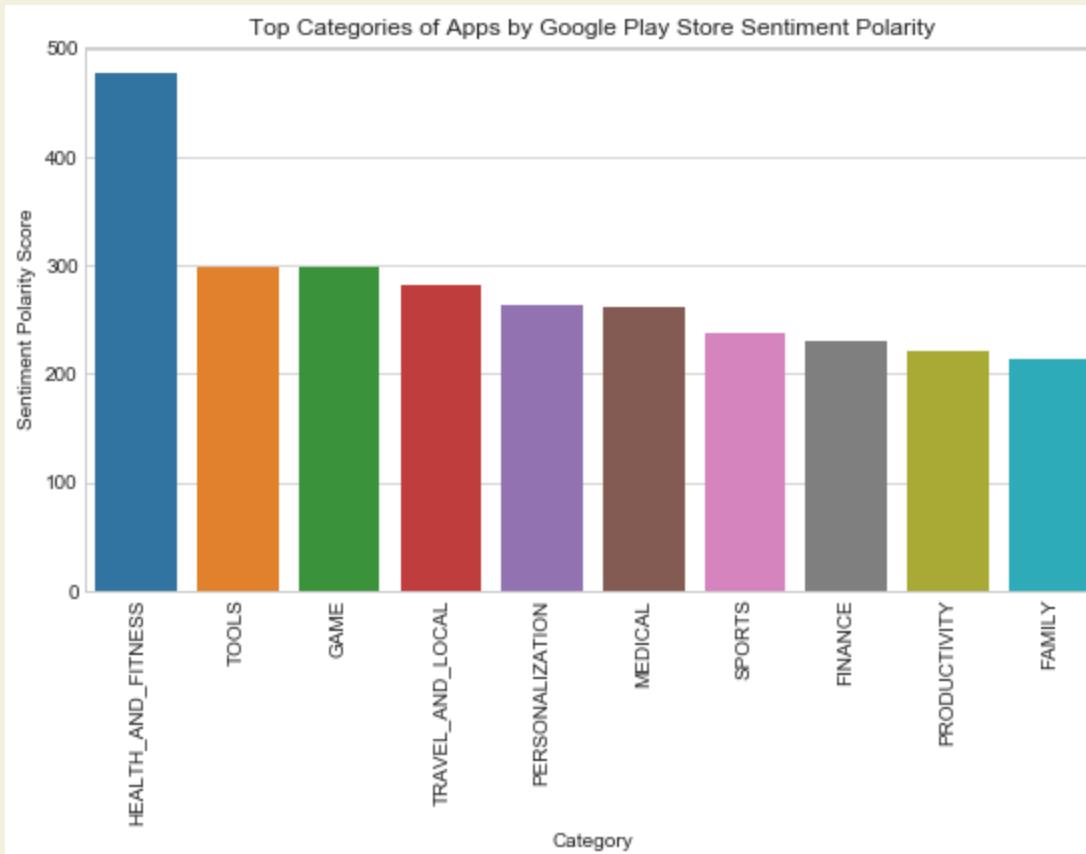


Sentiment Strategy



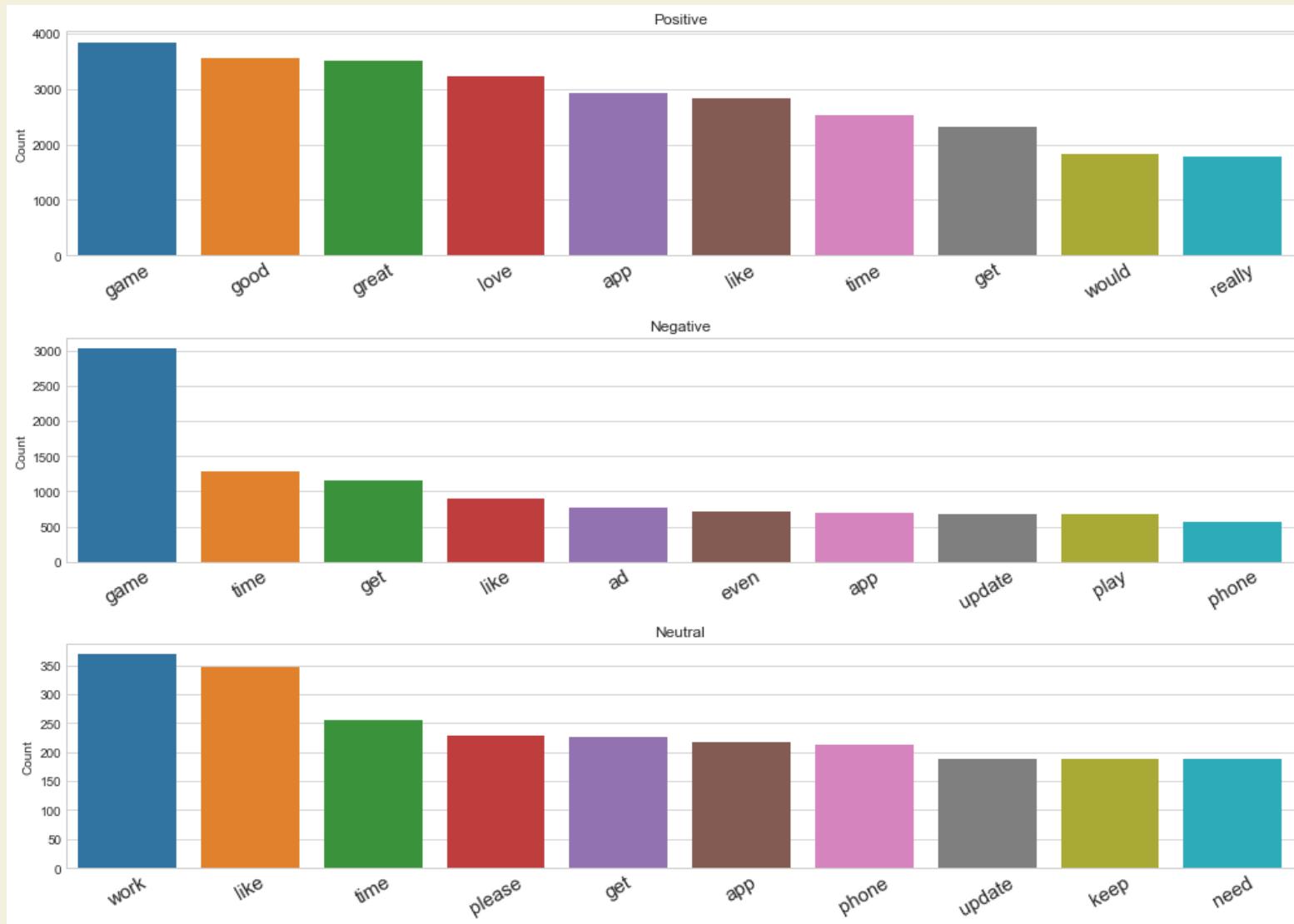


Sentiment Strategy



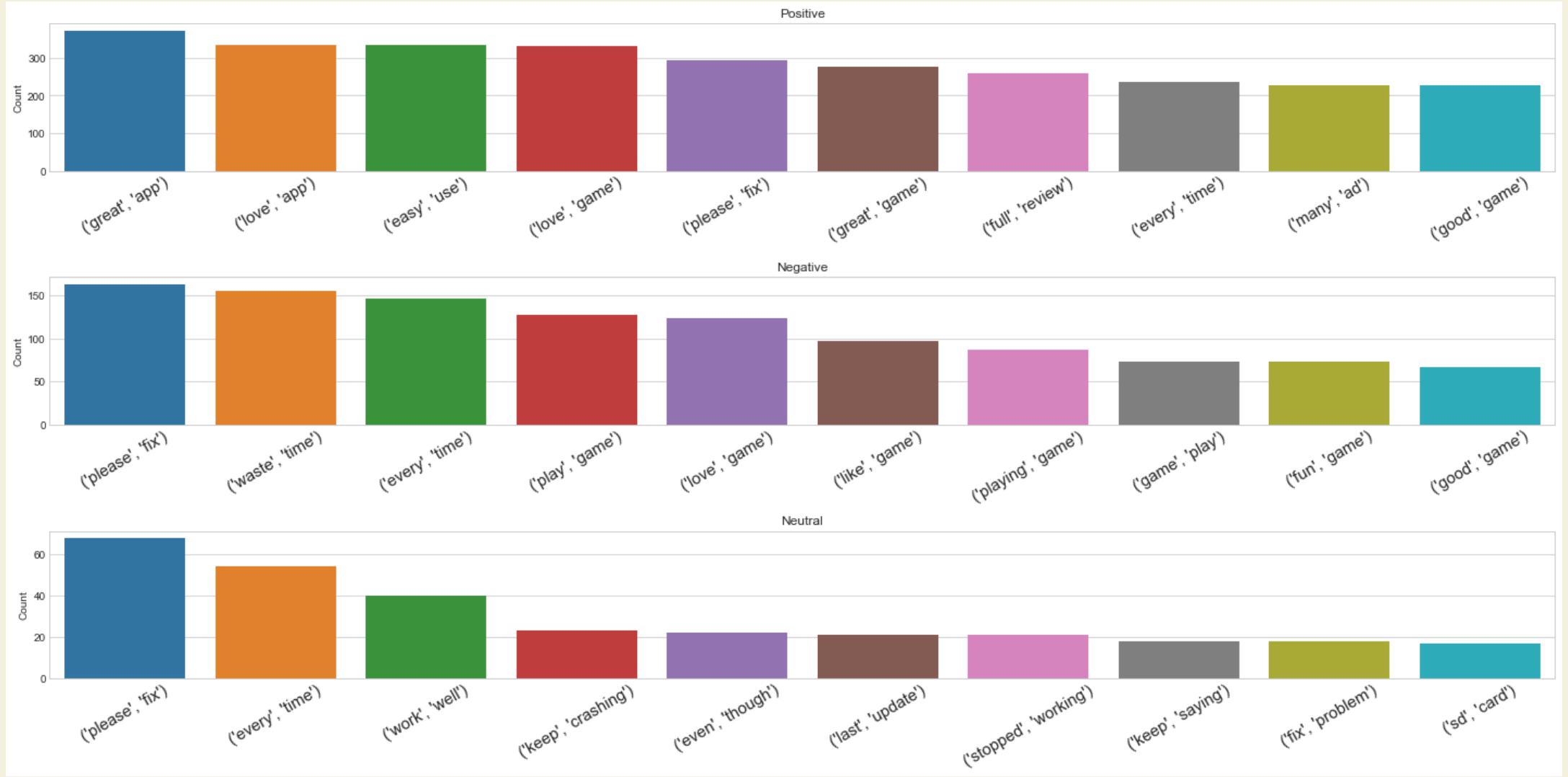


Sentiment Strategy





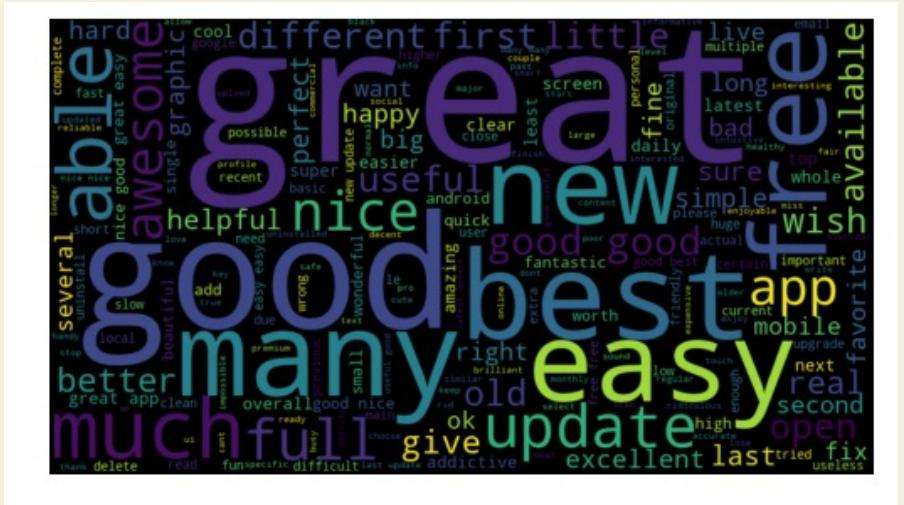
Sentiment Strategy



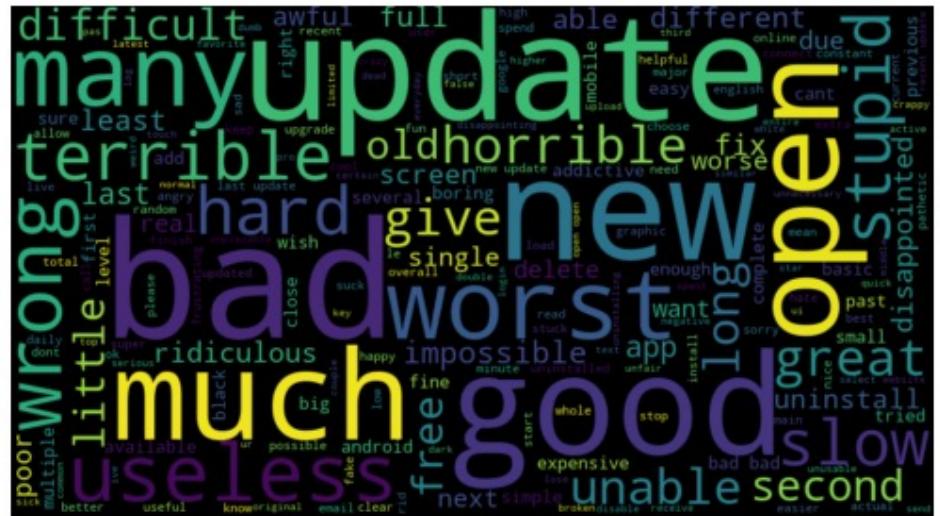


Sentiment Strategy

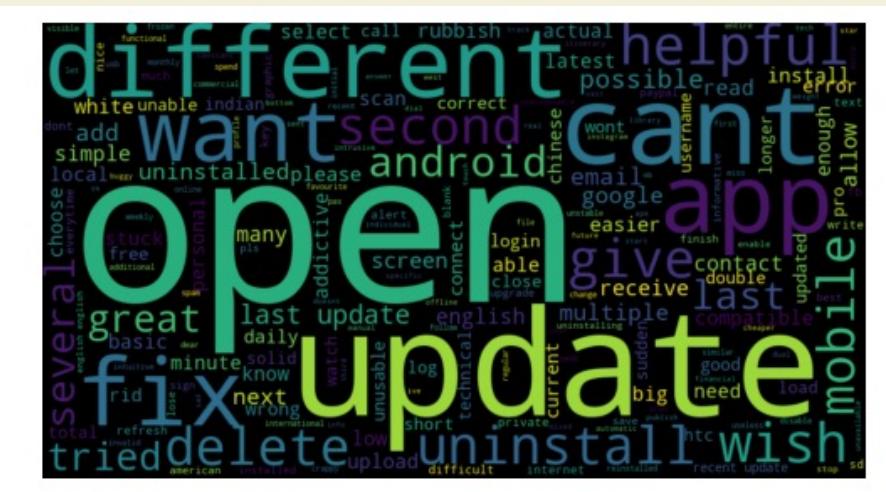
Adjective for Positive



Adjective for Negative



Adjective for Neutral





Analysis three categories based on top values

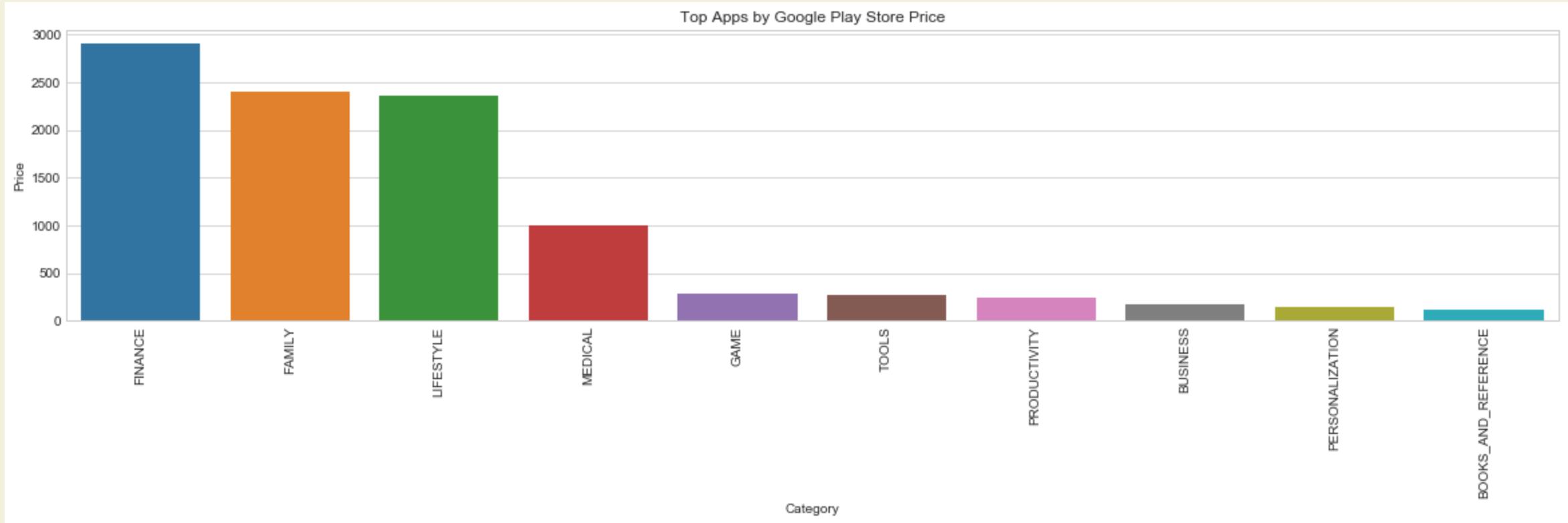
"Finance" High Price

"Family" High Rating, High Income

"Game" High Install, High Review, High Comment, High Size



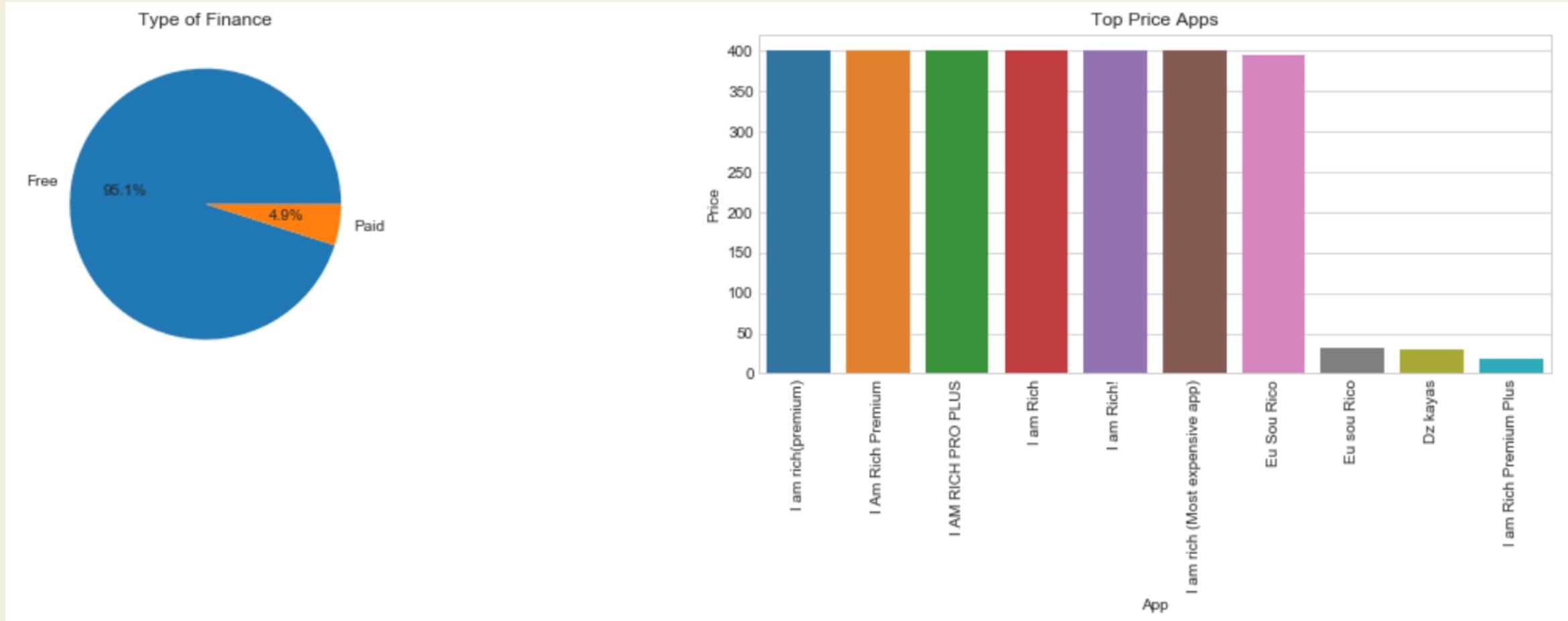
"Finance" = High Price



- From visualization, Finance category is top price.

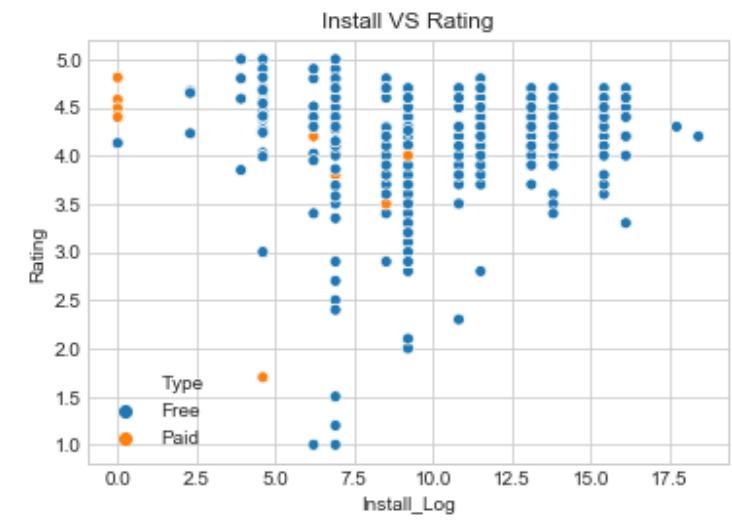
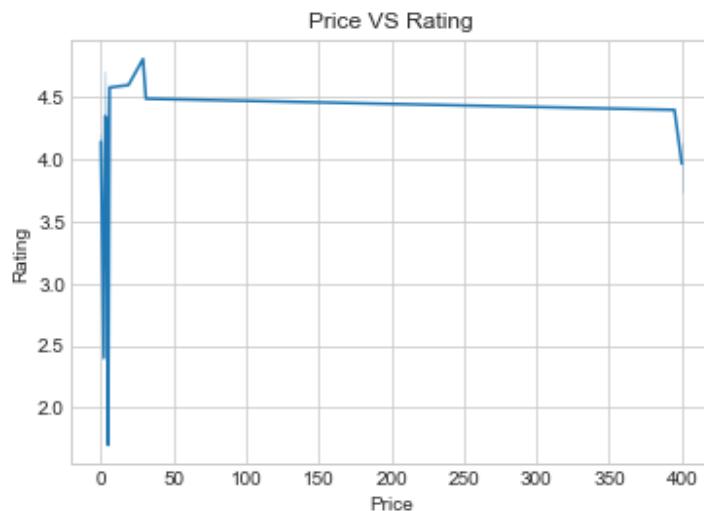
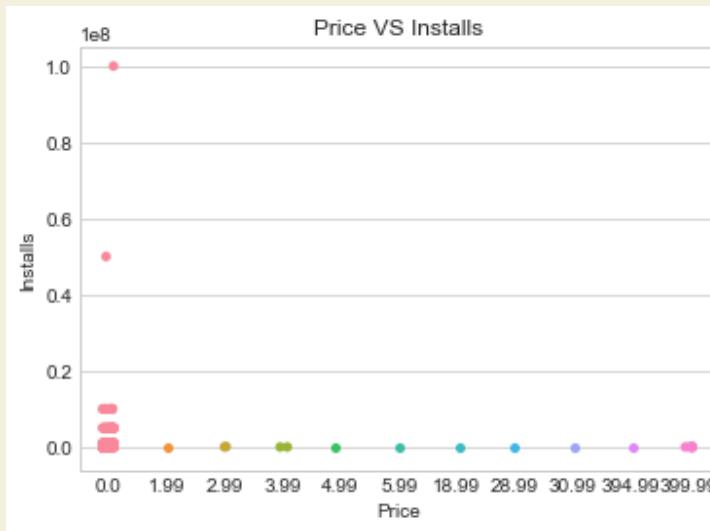


"Finance" = High Price



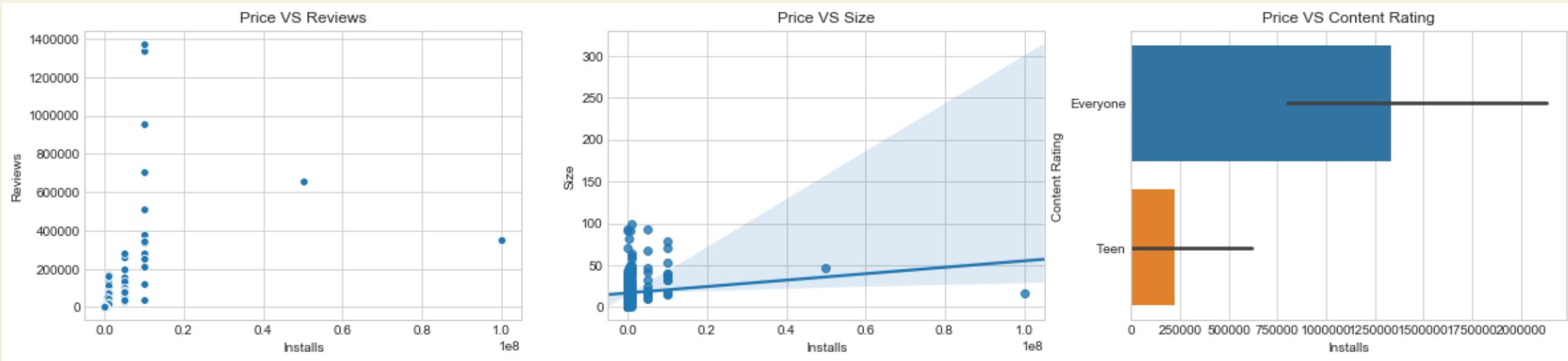


"Finance" = High Price



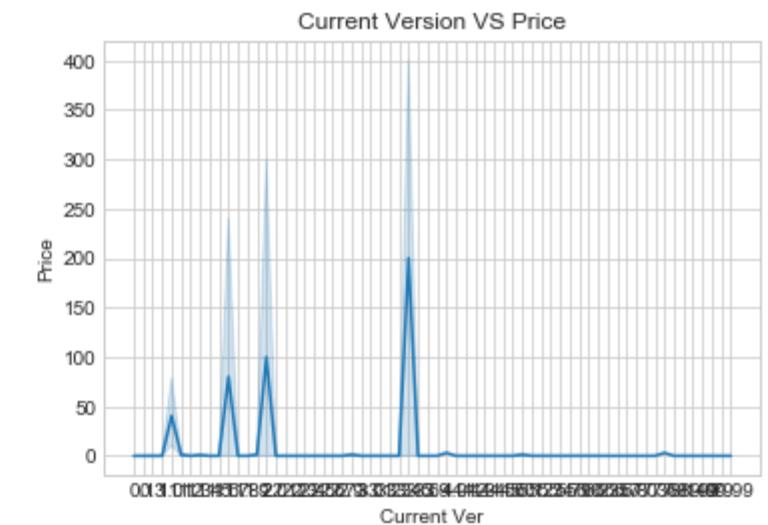
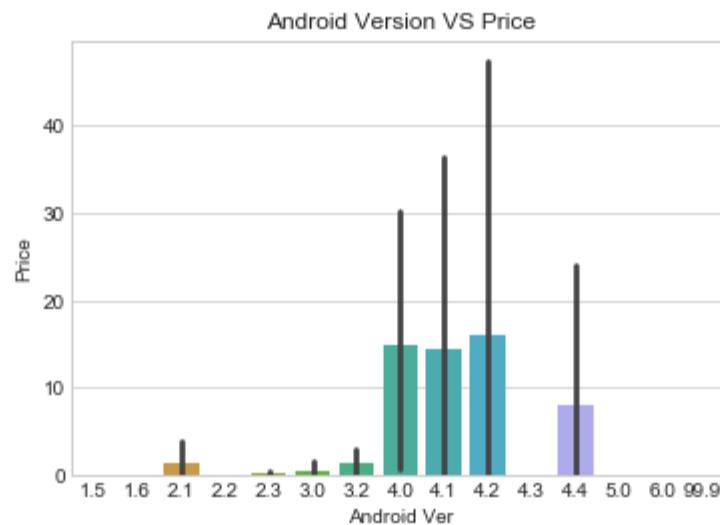
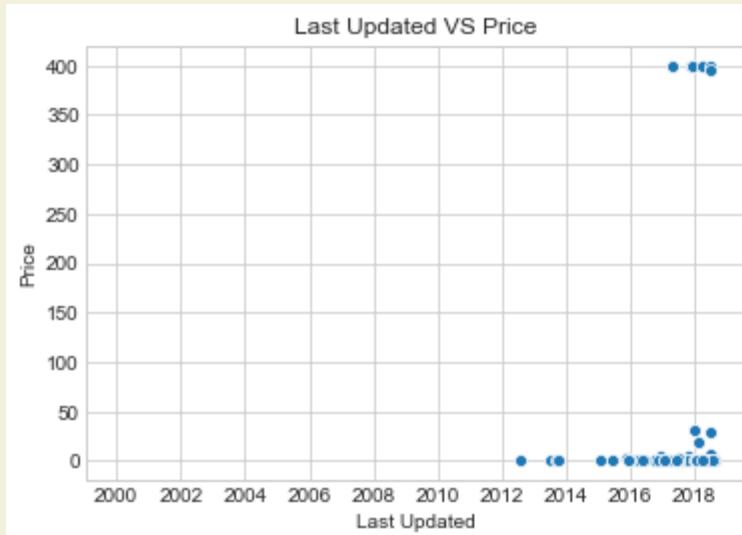


"Finance" = High Price



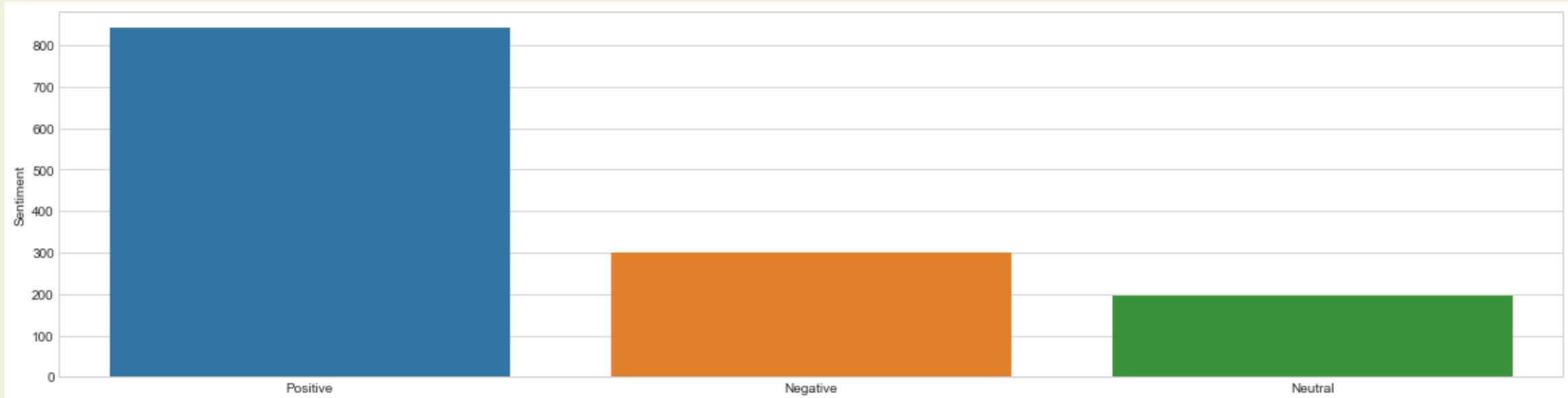


"Finance" = High Price



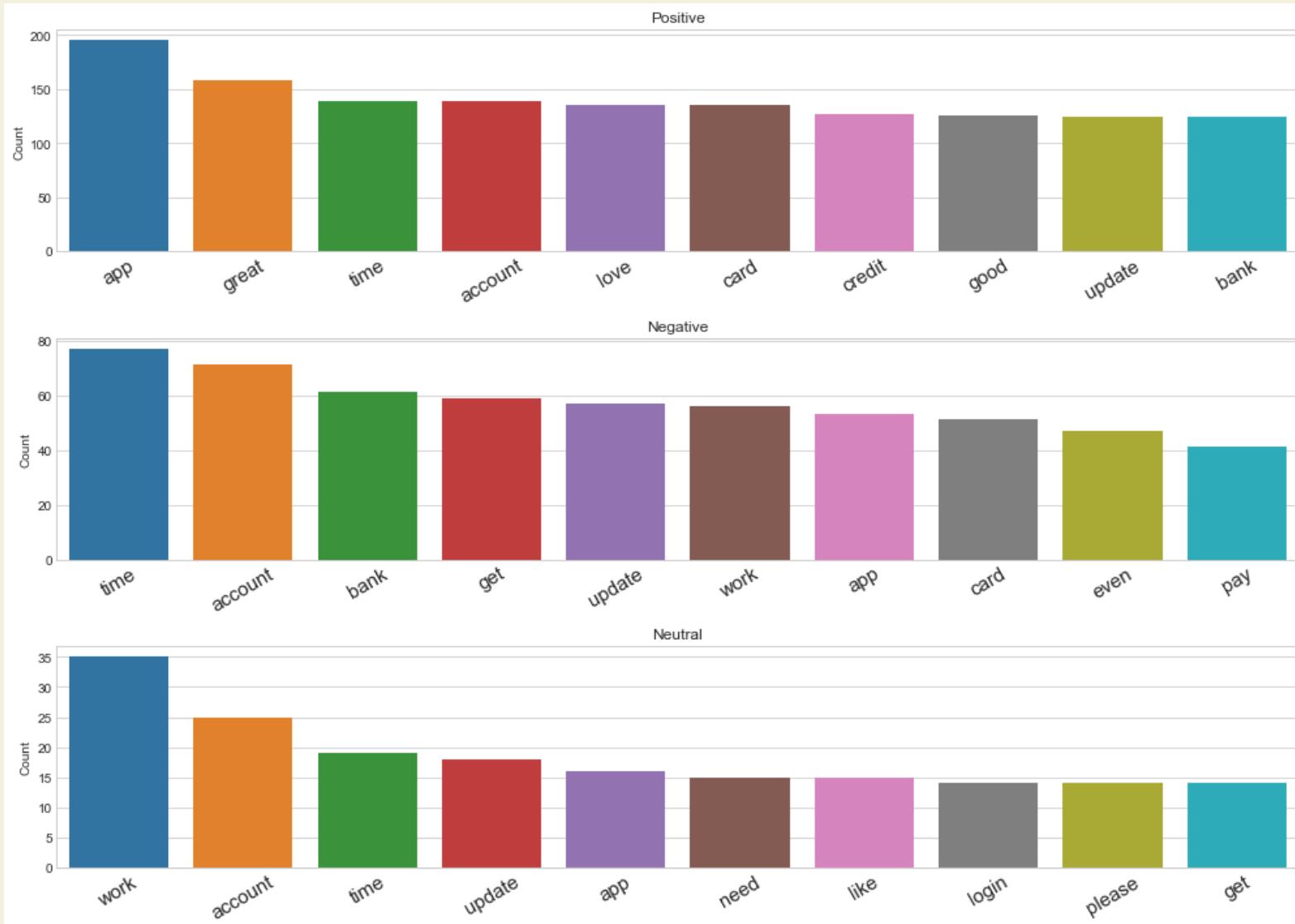


"Finance" = High Price





"Finance" = High Price





"Finance" = High Price



"Finance" = High Price



	App	Category	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	...	Type Code	Content Rating Code	Genres Category	Genres Subcategory
5359	I am rich(premium)	FINANCE	472	0.965	5000	Paid	399.99	Everyone	Finance	2017-05-01	...	1	1	Finance	Finance
5356	I Am Rich Premium	FINANCE	1867	4.700	50000	Paid	399.99	Everyone	Finance	2017-11-12	...	1	1	Finance	Finance
5373	I AM RICH PRO PLUS	FINANCE	36	41.000	1000	Paid	399.99	Everyone	Finance	2018-06-25	...	1	1	Finance	Finance
5369	I am Rich	FINANCE	180	3.800	5000	Paid	399.99	Everyone	Finance	2018-03-22	...	1	1	Finance	Finance
5358	I am Rich!	FINANCE	93	22.000	1000	Paid	399.99	Everyone	Finance	2017-12-11	...	1	1	Finance	Finance
...
1171	Citi Mobile®	FINANCE	78306	46.000	5000000	Free	0.00	Everyone	Finance	2018-07-31	...	0	1	Finance	Finance
1170	Amex Mobile	FINANCE	24729	32.000	1000000	Free	0.00	Everyone	Finance	2018-08-03	...	0	1	Finance	Finance
1167	Discover Mobile	FINANCE	87951	68.000	5000000	Free	0.00	Everyone	Finance	2018-07-23	...	0	1	Finance	Finance



"Finance" = High Price

		App	Category	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	...	Type Code	Rating Code	Genres Category	Genres Subcategory	Category Code	Subcate...
9917	Eu Sou Rico	FINANCE		0	1.4	0	Paid	394.99	Everyone	Finance	2018-07-11	...	1	1	Finance	Finance	19	
9905	Eu sou Rico	FINANCE		0	2.6	0	Paid	30.99	Everyone	Finance	2018-01-09	...	1	1	Finance	Finance	19	
9104	Dz kayas	FINANCE		0	14.0	1	Paid	28.99	Everyone	Finance	2018-07-12	...	1	1	Finance	Finance	19	
9101	amm dz	FINANCE		0	14.0	1	Paid	5.99	Everyone	Finance	2018-07-08	...	1	1	Finance	Finance	19	
6948	Bitcoin BX Thailand PRO	FINANCE		21	21.0	100	Paid	4.99	Everyone	Finance	2017-10-22	...	1	1	Finance	Finance	19	
...	
1171	Citi Mobile®	FINANCE		78306	46.0	5000000	Free	0.00	Everyone	Finance	2018-07-31	...	0	1	Finance	Finance	19	
1170	Amex Mobile	FINANCE		24729	32.0	1000000	Free	0.00	Everyone	Finance	2018-08-03	...	0	1	Finance	Finance	19	
1167	Discover Mobile	FINANCE		87951	68.0	5000000	Free	0.00	Everyone	Finance	2018-07-23	...	0	1	Finance	Finance	19	
USE																		

- Remove words: RICH, Rich, rich



"Finance" = High Price

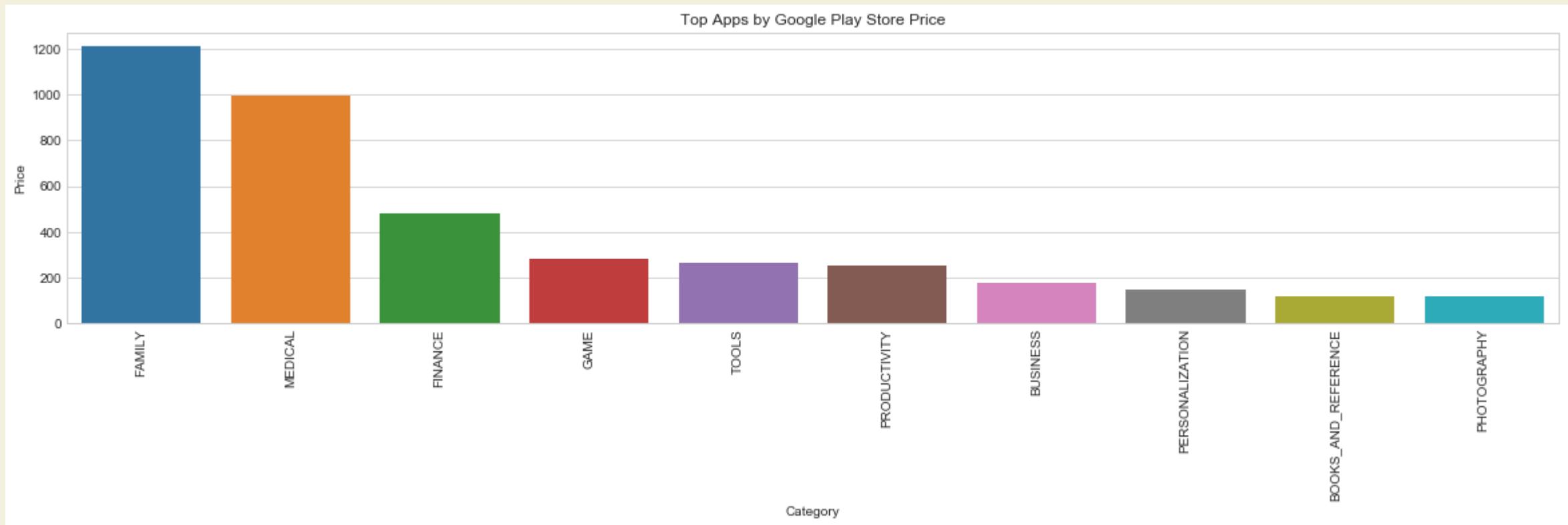
		App	Category	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	...	Type Code	Rating Code	Genres Category	Genres Subcategory	Category Code	Subcate...
9917	Eu Sou Rico	FINANCE		0	1.4	0	Paid	394.99	Everyone	Finance	2018-07-11	...	1	1	Finance	Finance	19	
9905	Eu sou Rico	FINANCE		0	2.6	0	Paid	30.99	Everyone	Finance	2018-01-09	...	1	1	Finance	Finance	19	
9104	Dz kayas	FINANCE		0	14.0	1	Paid	28.99	Everyone	Finance	2018-07-12	...	1	1	Finance	Finance	19	
9101	amm dz	FINANCE		0	14.0	1	Paid	5.99	Everyone	Finance	2018-07-08	...	1	1	Finance	Finance	19	
6948	Bitcoin BX Thailand PRO	FINANCE		21	21.0	100	Paid	4.99	Everyone	Finance	2017-10-22	...	1	1	Finance	Finance	19	
...	
1171	Citi Mobile®	FINANCE		78306	46.0	5000000	Free	0.00	Everyone	Finance	2018-07-31	...	0	1	Finance	Finance	19	
1170	Amex Mobile	FINANCE		24729	32.0	1000000	Free	0.00	Everyone	Finance	2018-08-03	...	0	1	Finance	Finance	19	
1167	Discover Mobile	FINANCE		87951	68.0	5000000	Free	0.00	Everyone	Finance	2018-07-23	...	0	1	Finance	Finance	19	
USE																		

- Remove words: RICH, Rich, rich



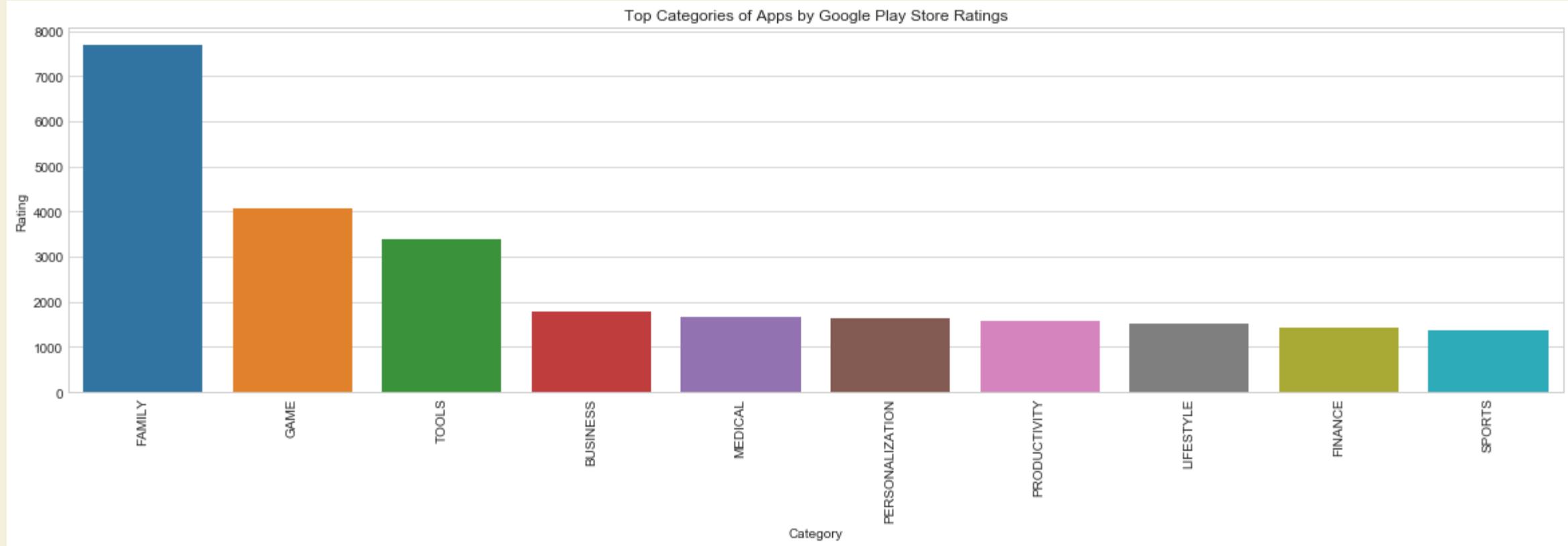
"Finance" = High Price

Does finance category have really high price among all categories?



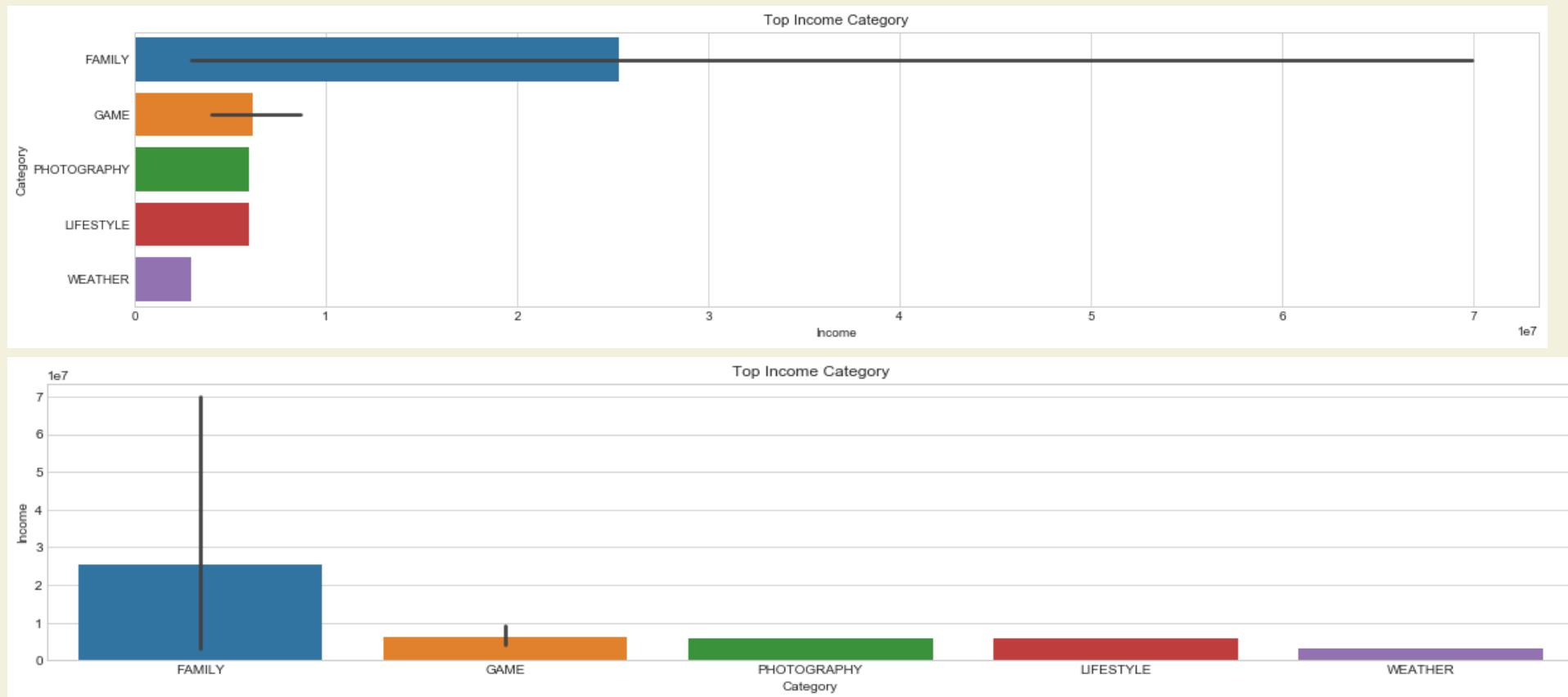


"Family" = High Rating, High Income



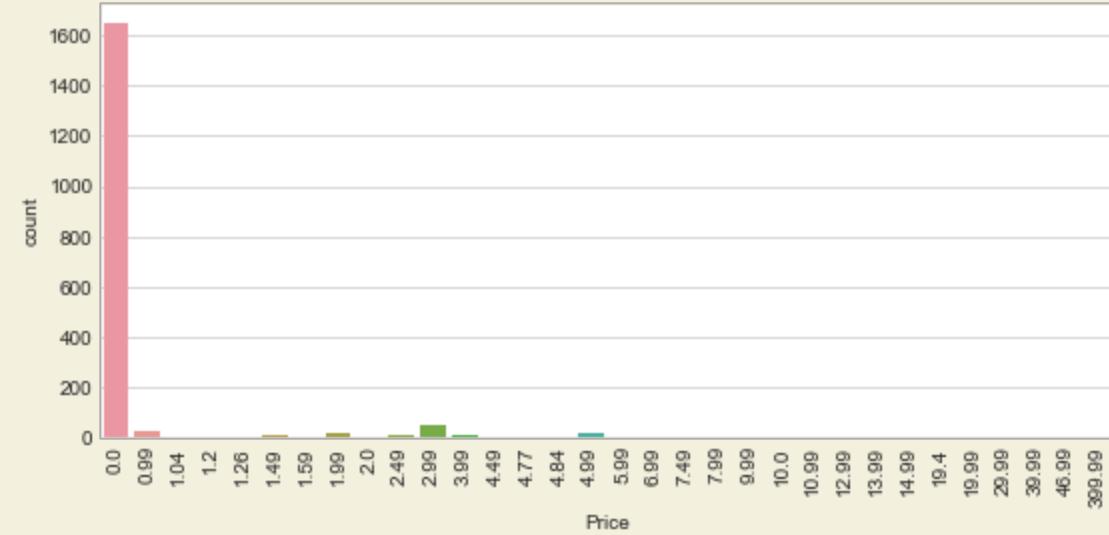
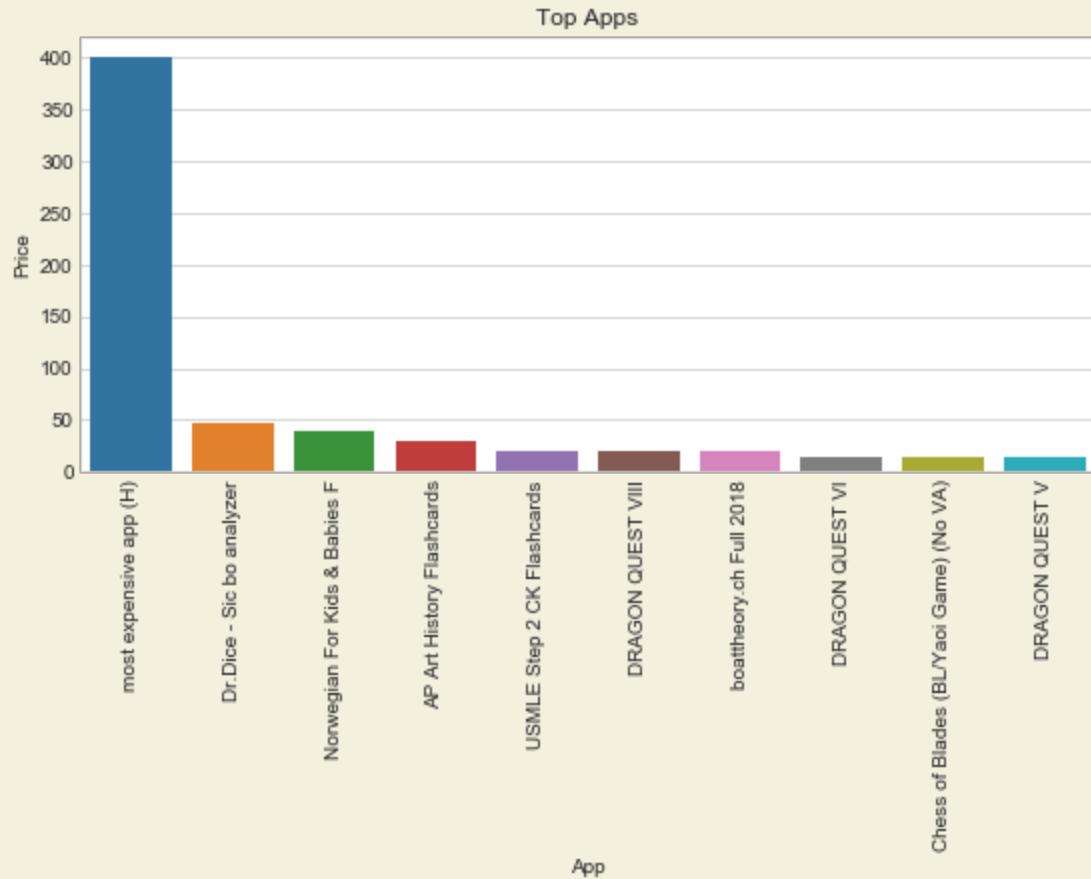


"Family" = High Rating, High Income



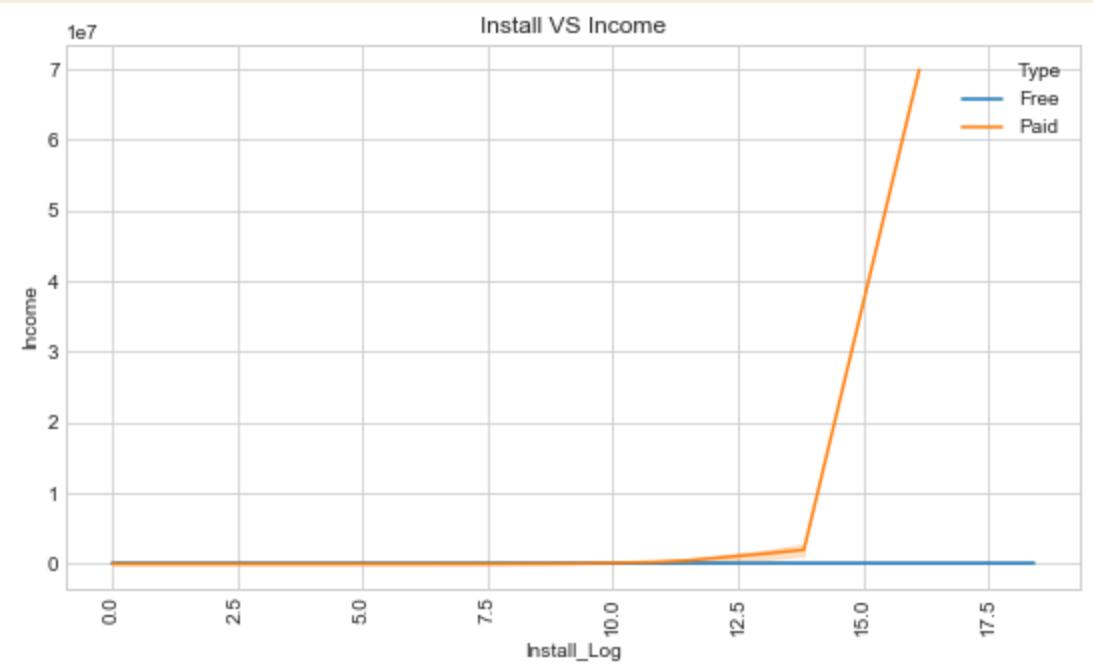
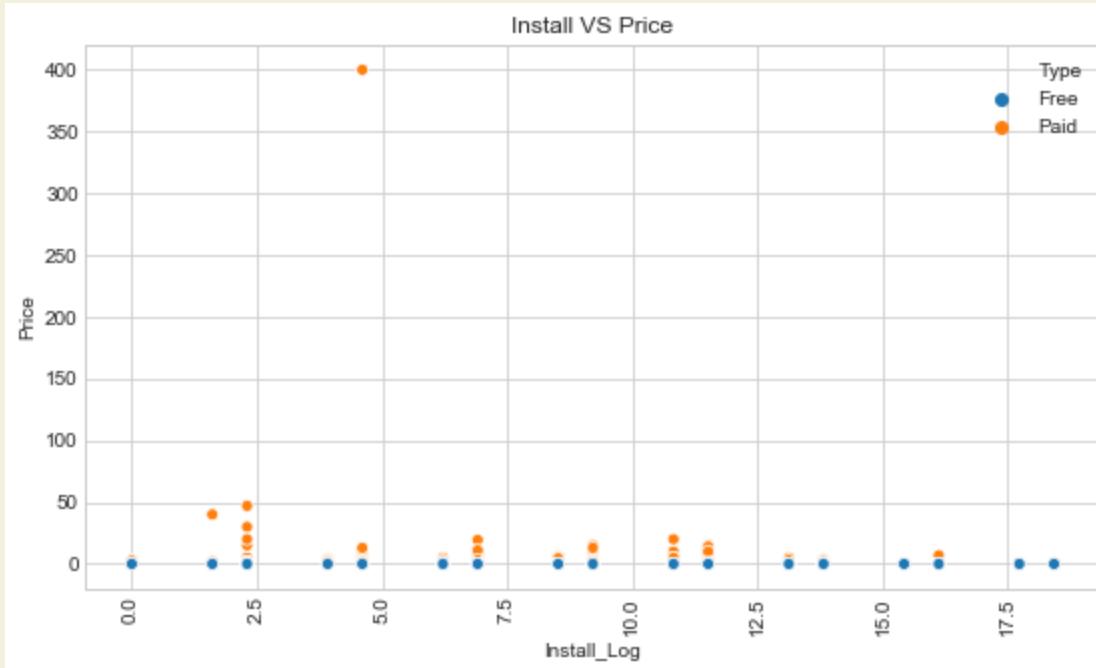


"Family" = High Rating, High Income



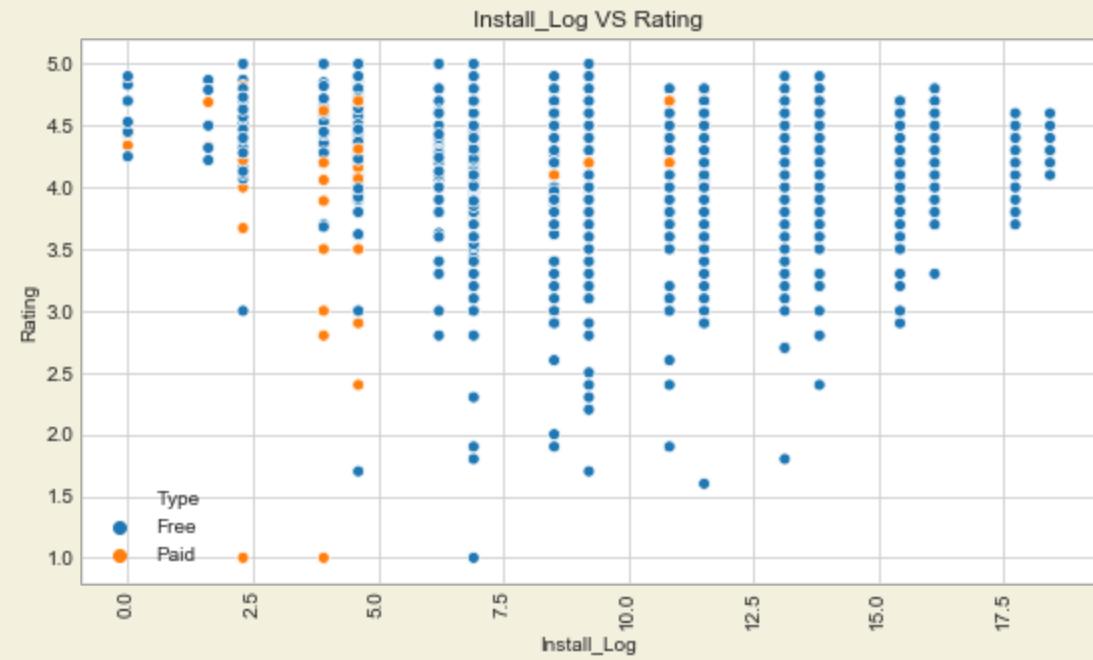


"Family" = High Rating, High Income



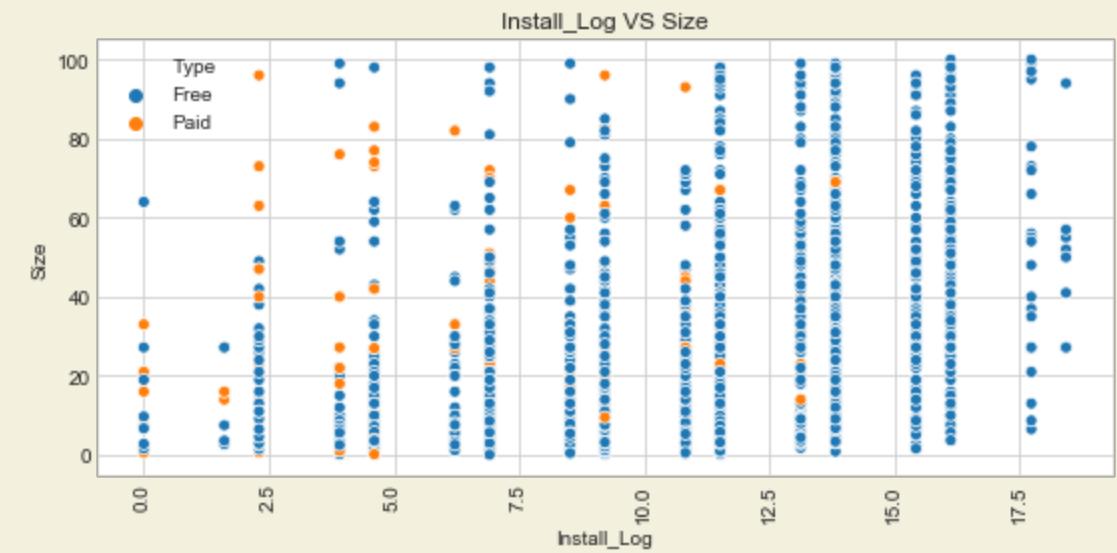
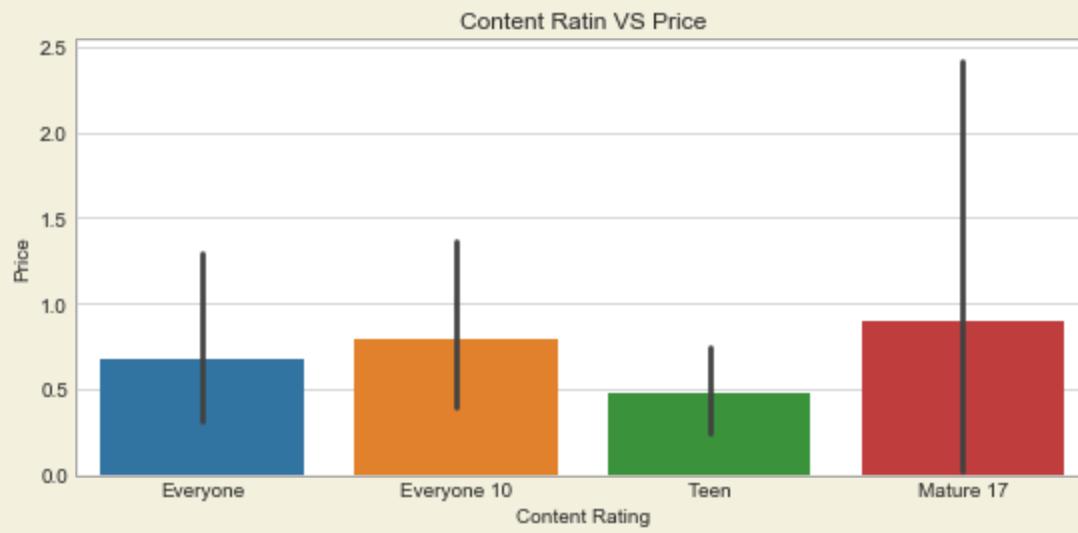


"Family" = High Rating, High Income



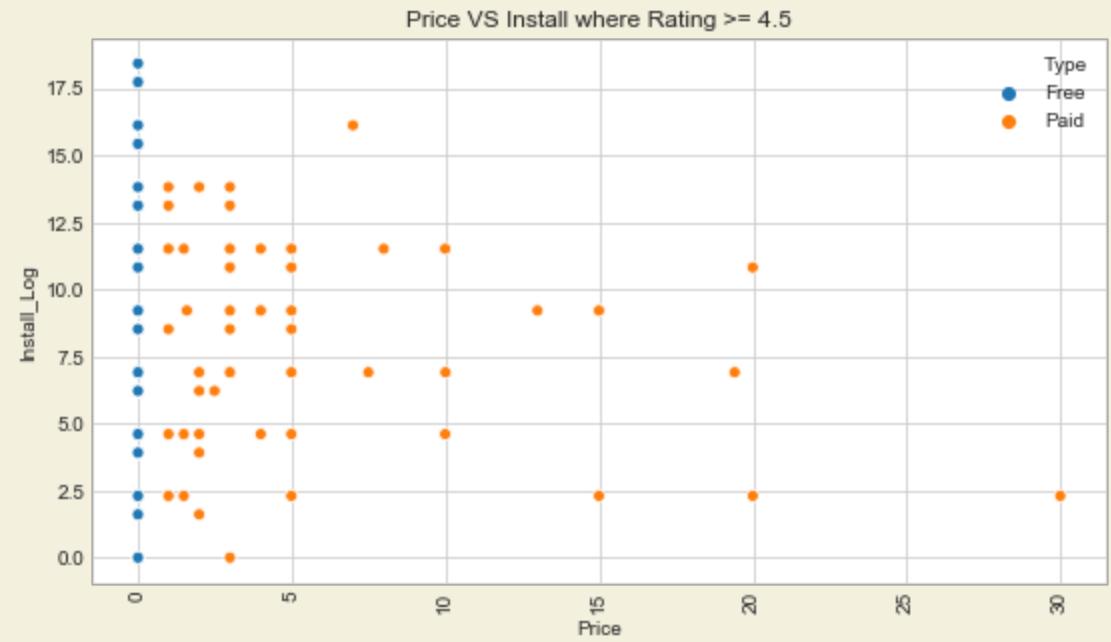


"Family" = High Rating, High Income



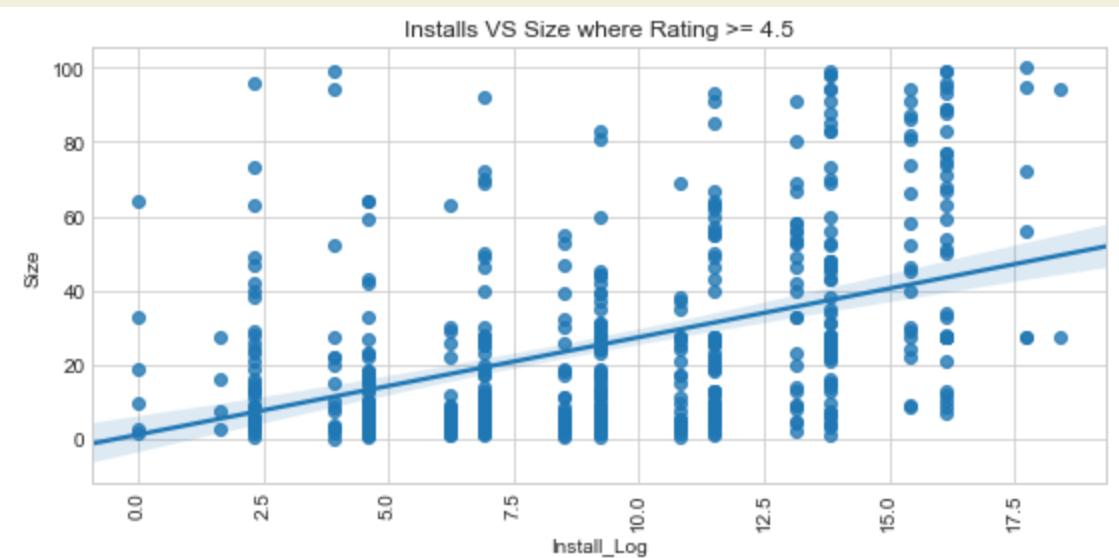
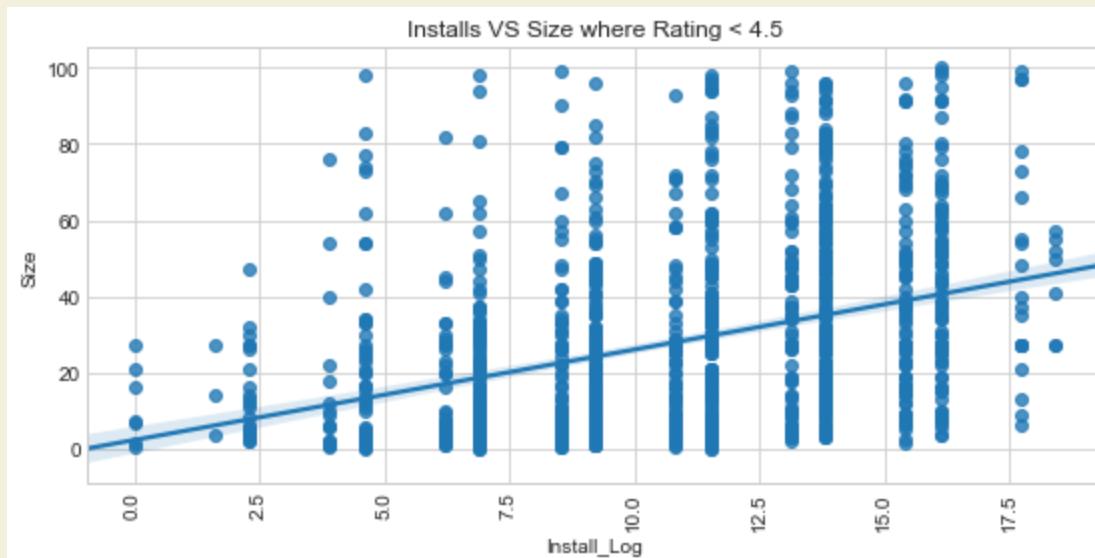


"Family" = High Rating, High Income



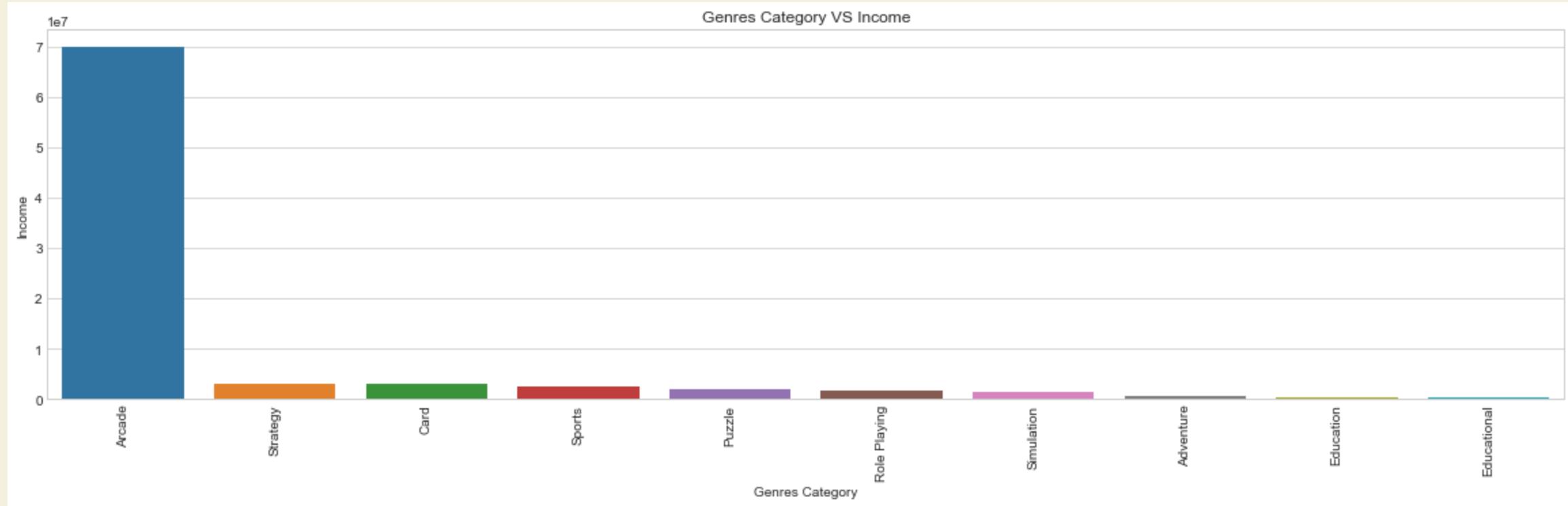


"Family" = High Rating, High Income



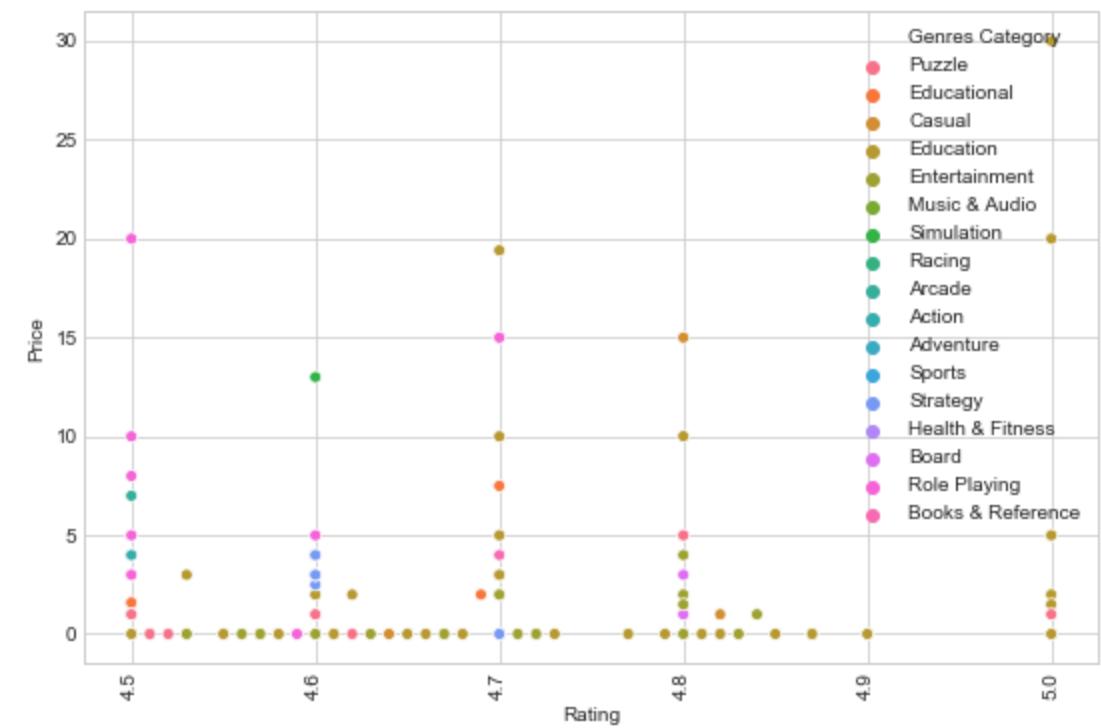
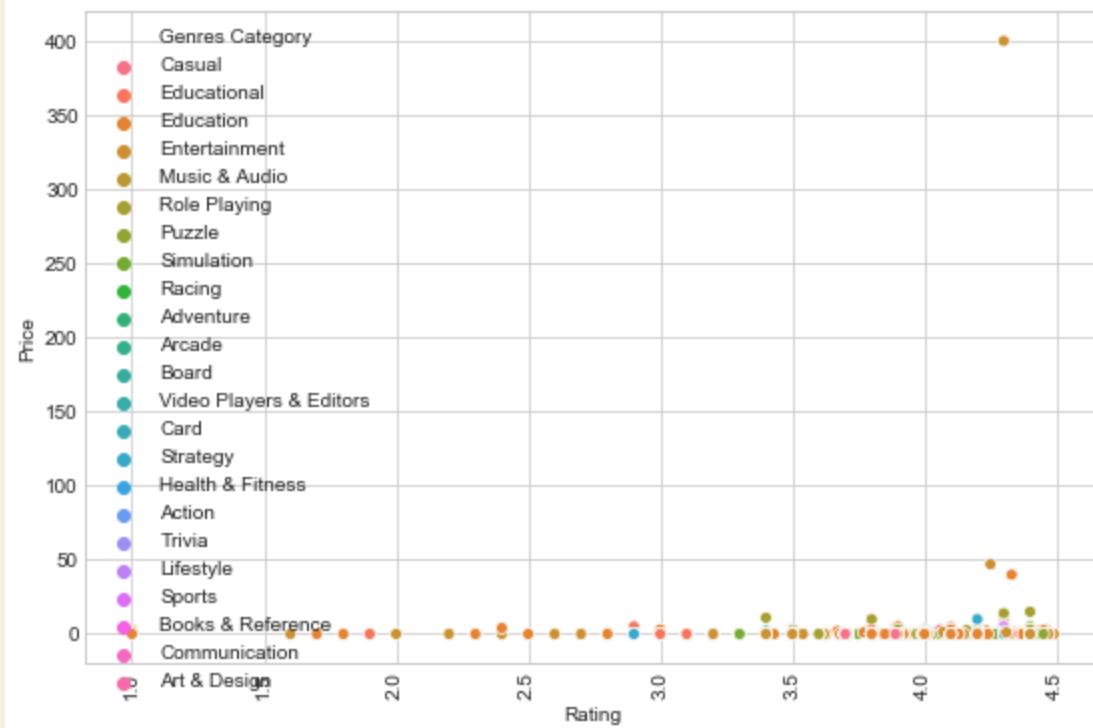


"Family" = High Rating, High Income



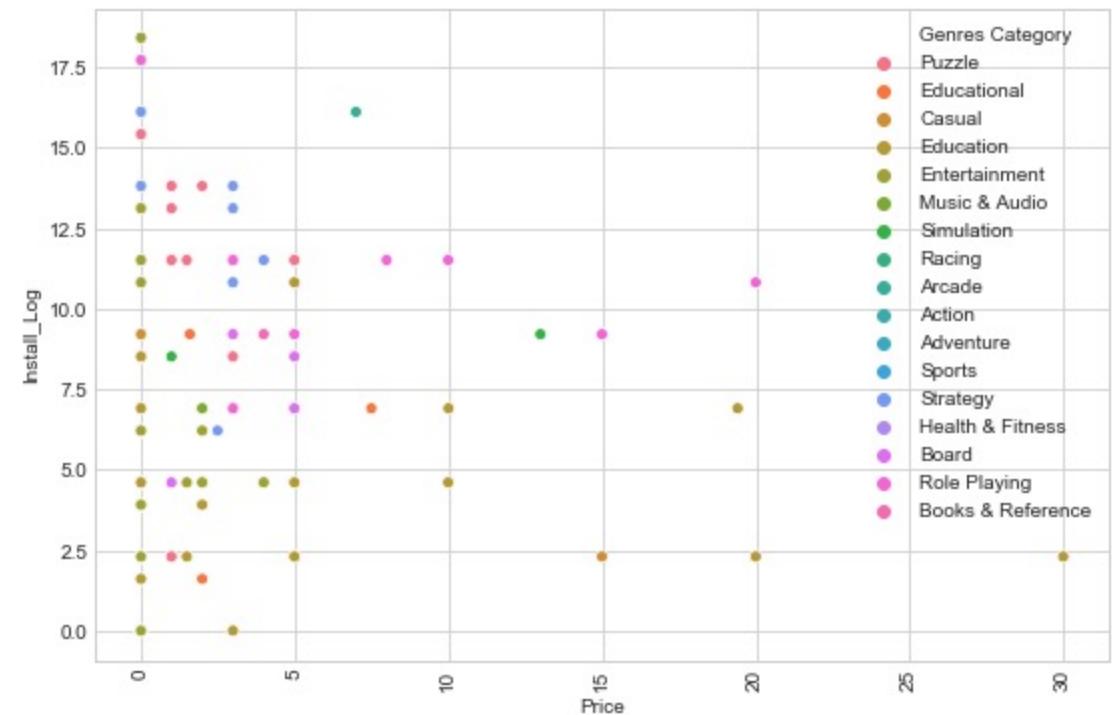
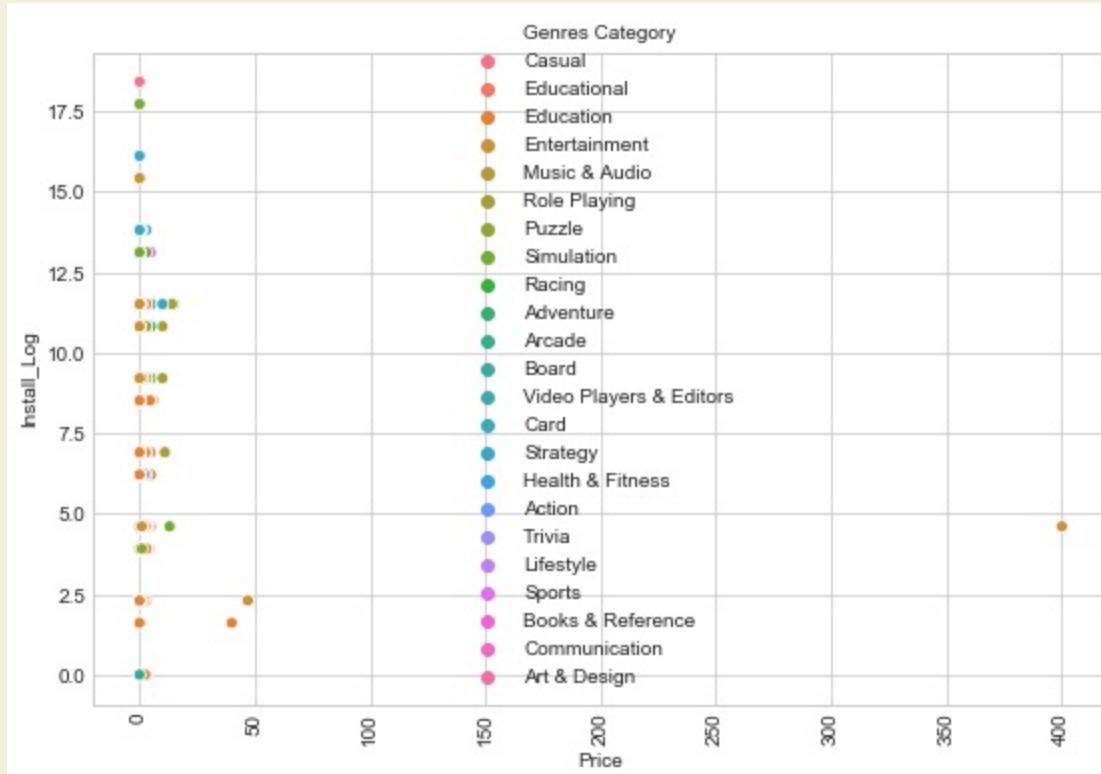


"Family" = High Rating, High Income



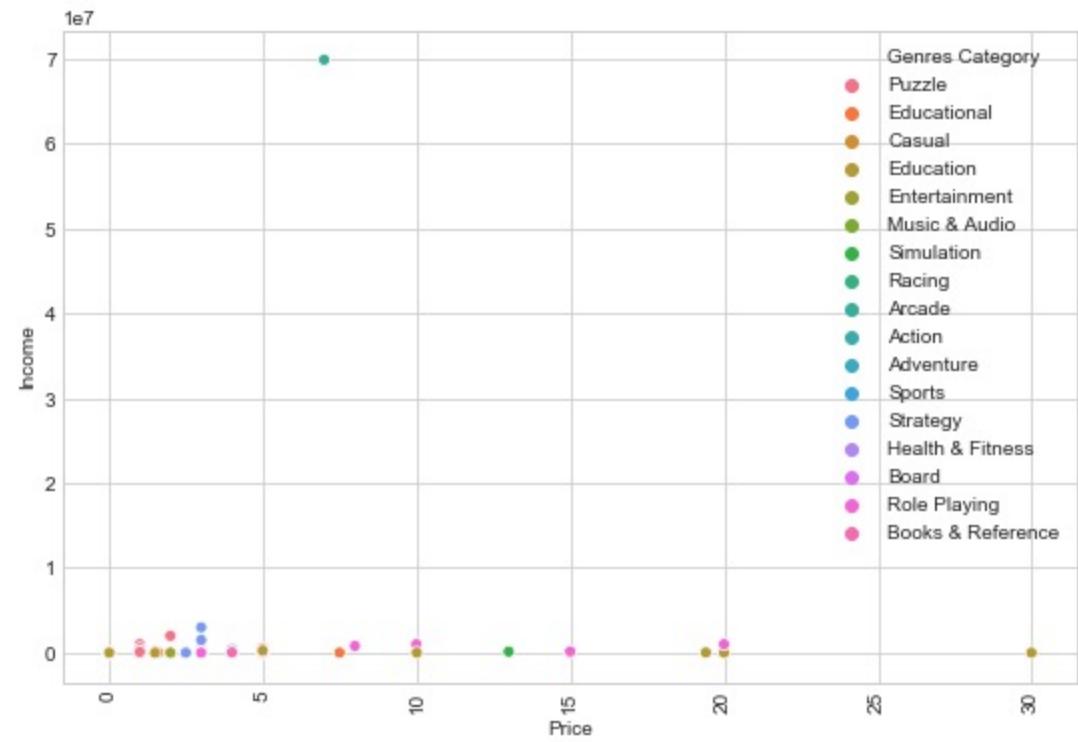
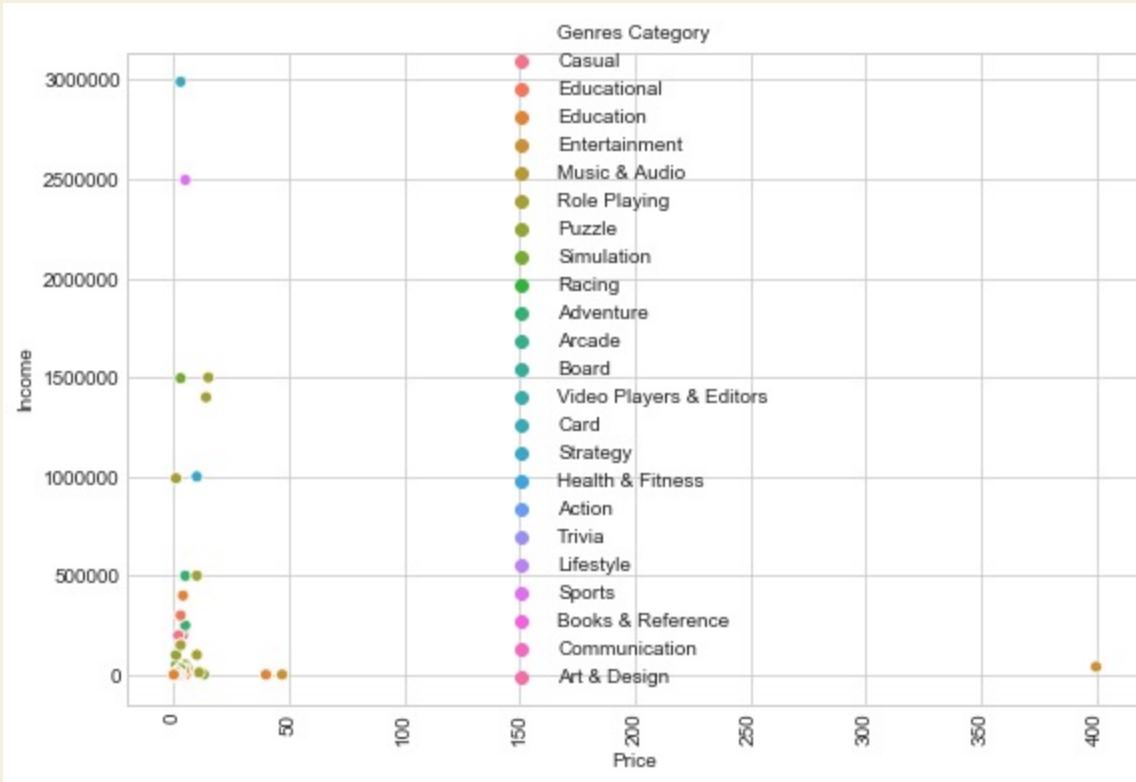


"Family" = High Rating, High Income



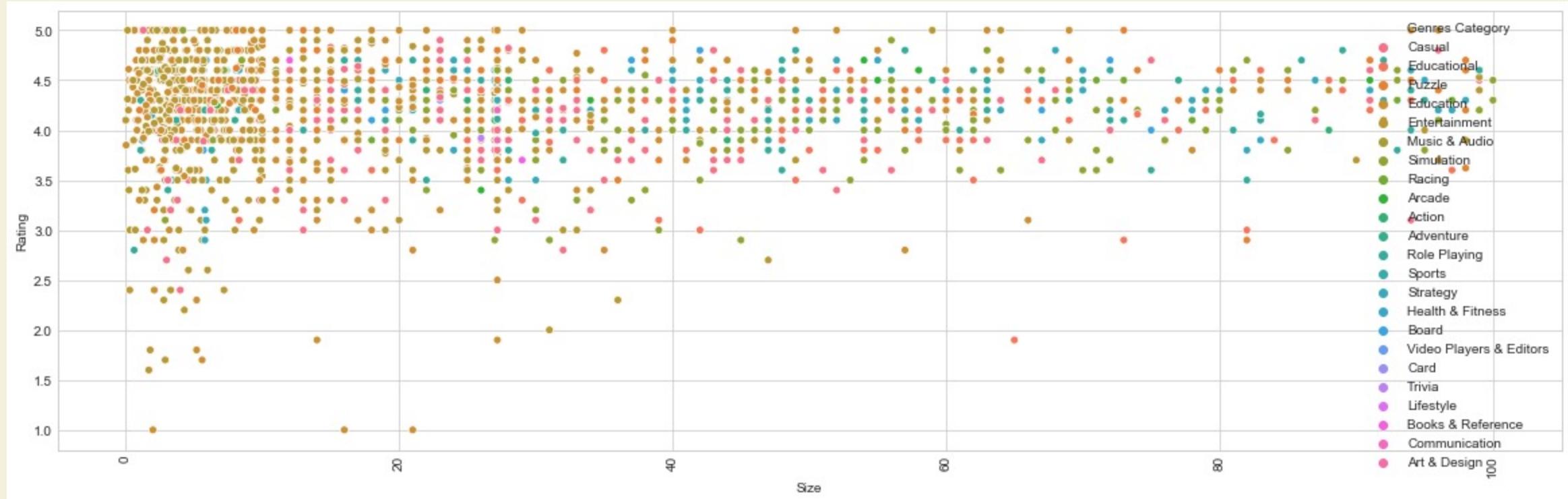


"Family" = High Rating, High Income



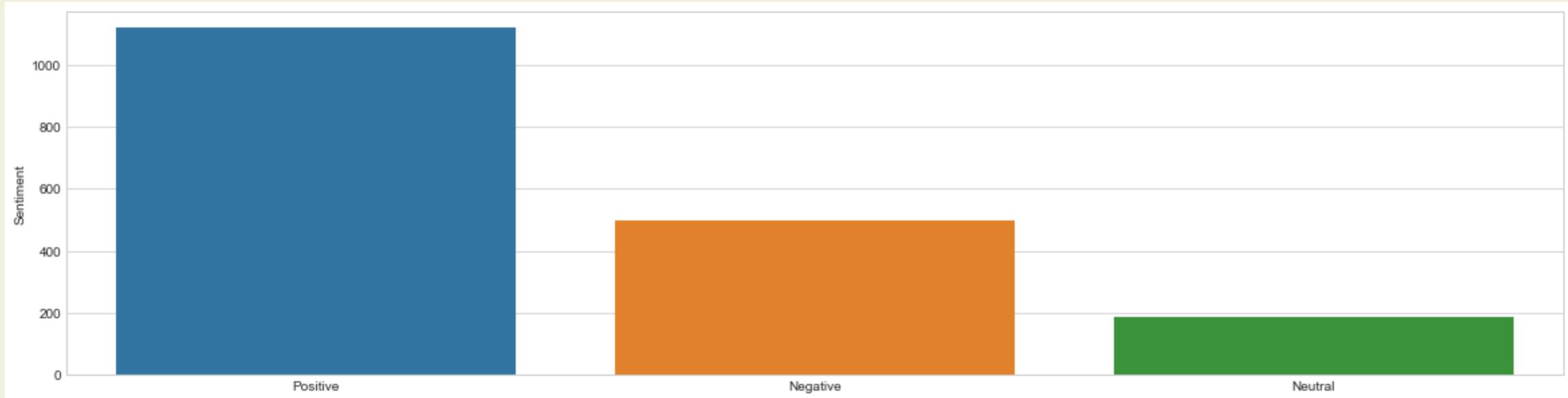


"Family" = High Rating, High Income



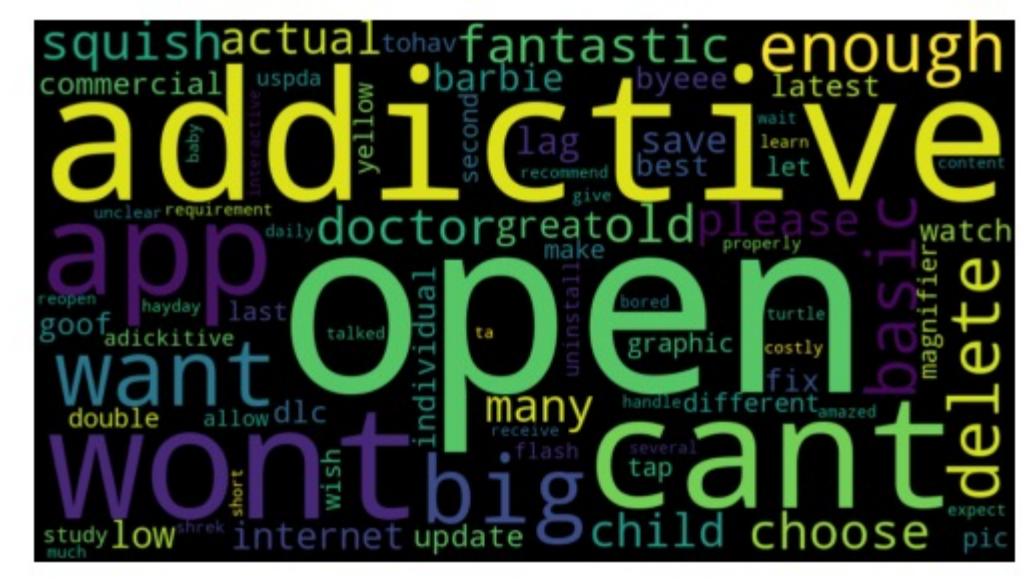
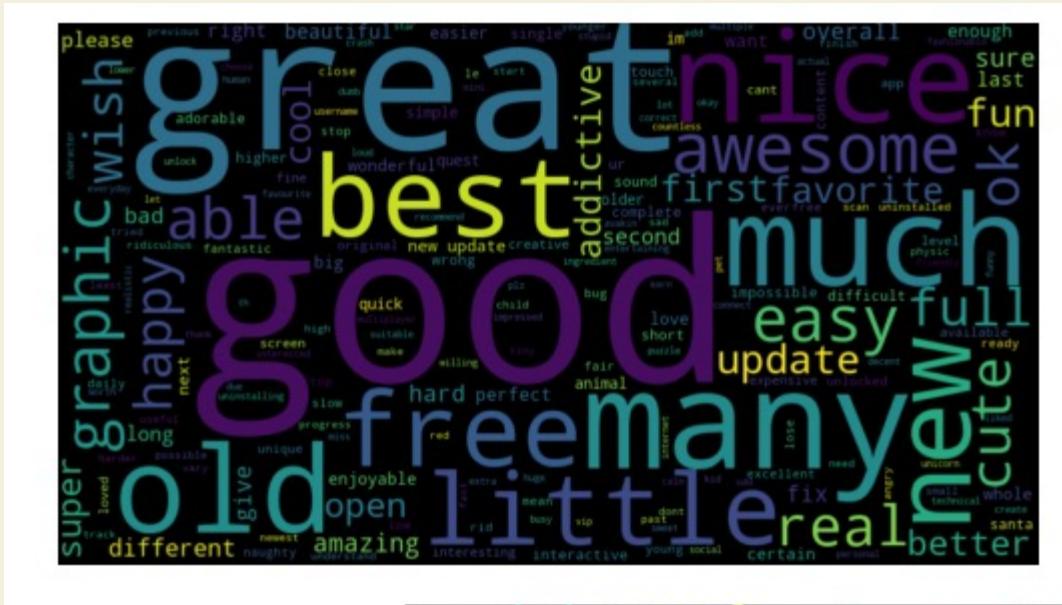


"Family" = High Rating, High Income





"Family" = High Rating, High Income





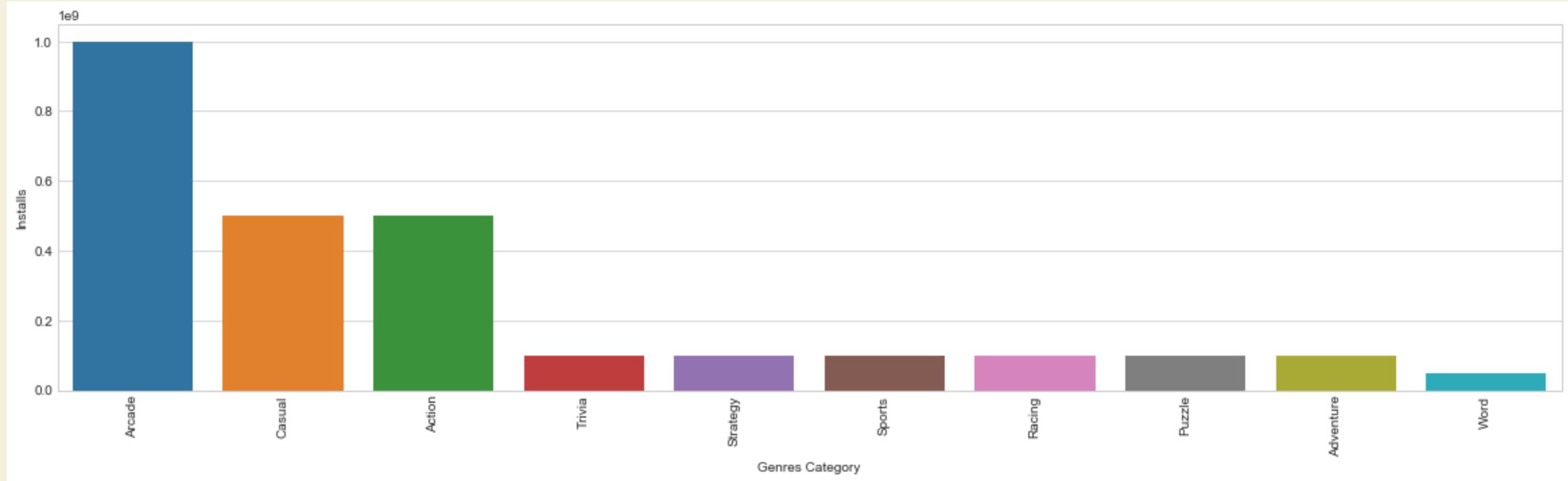
"Game" = High Install, High Review, High Comment, High Size

```
df_game = apps[(apps["Category"] == "GAME") & (apps["Installs"] > 10000000)]  
df_game.sort_values(by="Installs", ascending=False)[:5]
```

	App	Category	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	...	Install_Log	Type Code	Content Rating Code	Genres Category	Genres Subcategory
1654	Subway Surfers	GAME	27722264	76.000000	1000000000	Free	0.0	Everyone 10	Arcade	2018-07-12	...	20.723266	0	2	Arcade	Arcade
1655	Candy Crush Saga	GAME	22426677	74.000000	500000000	Free	0.0	Everyone	Casual	2018-07-05	...	20.030119	0	1	Casual	Casual
1722	My Talking Tom	GAME	14891223	41.866609	500000000	Free	0.0	Everyone	Casual	2018-07-19	...	20.030119	0	1	Casual	Casual
1661	Temple Run 2	GAME	8118609	62.000000	500000000	Free	0.0	Everyone	Action	2018-07-05	...	20.030119	0	1	Action	Action
1662	Pou	GAME	10485308	24.000000	500000000	Free	0.0	Everyone	Casual	2018-05-25	...	20.030119	0	1	Casual	Casual

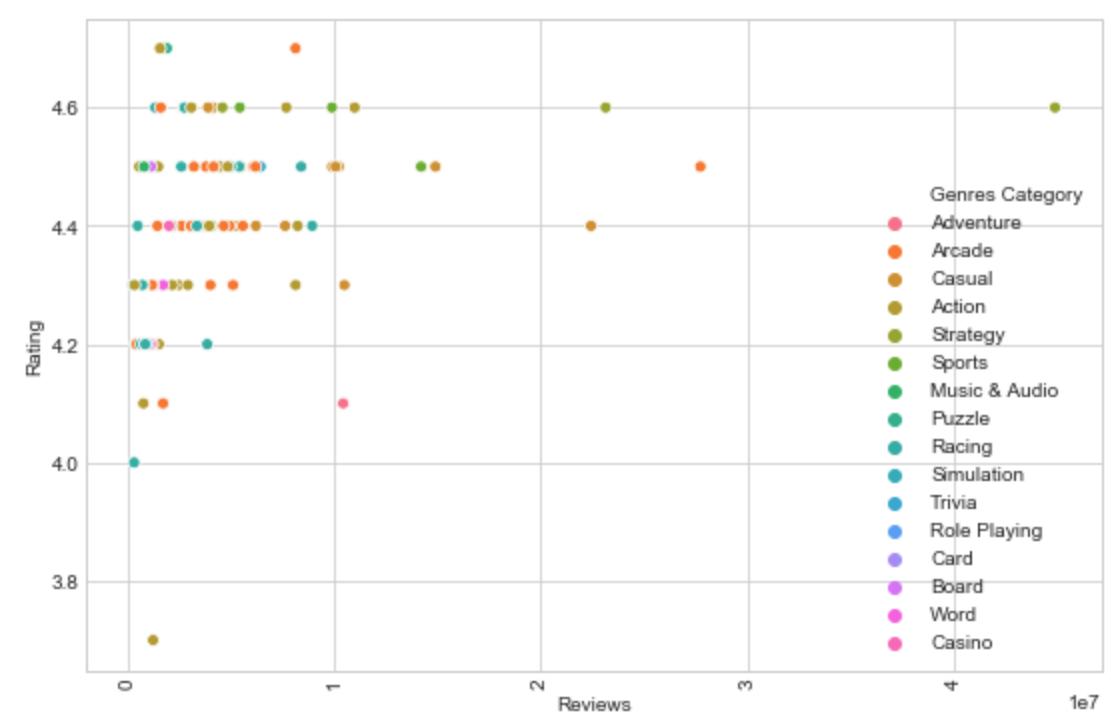
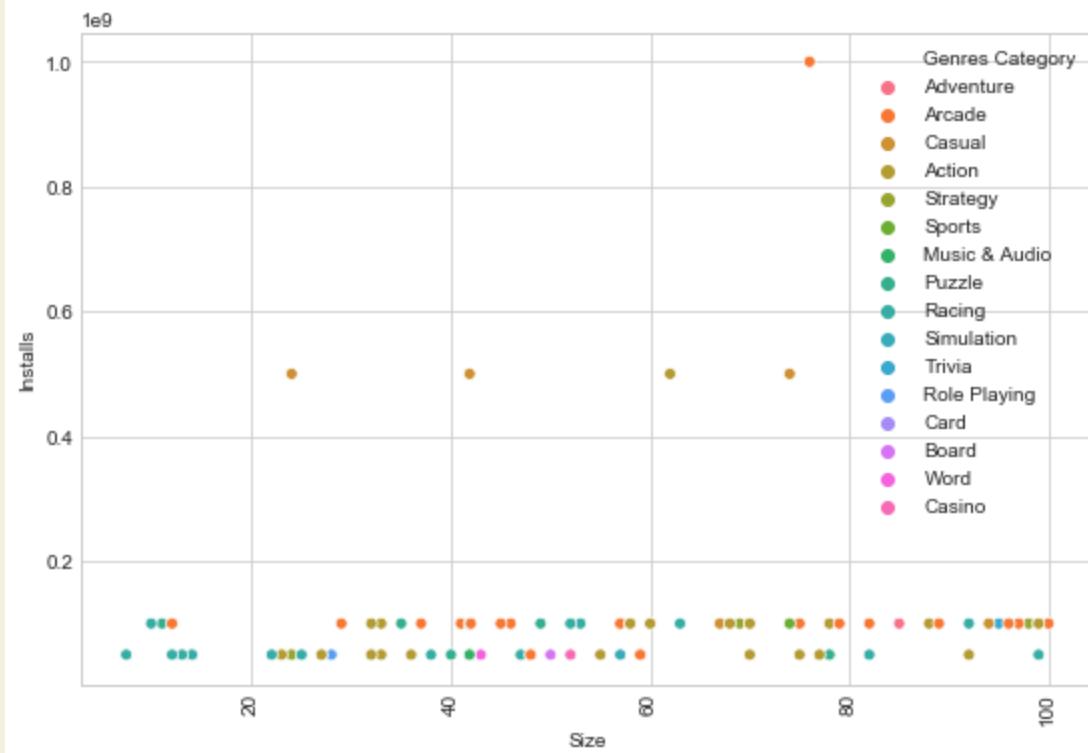


"Game" = High Install, High Review, High Comment, High Size



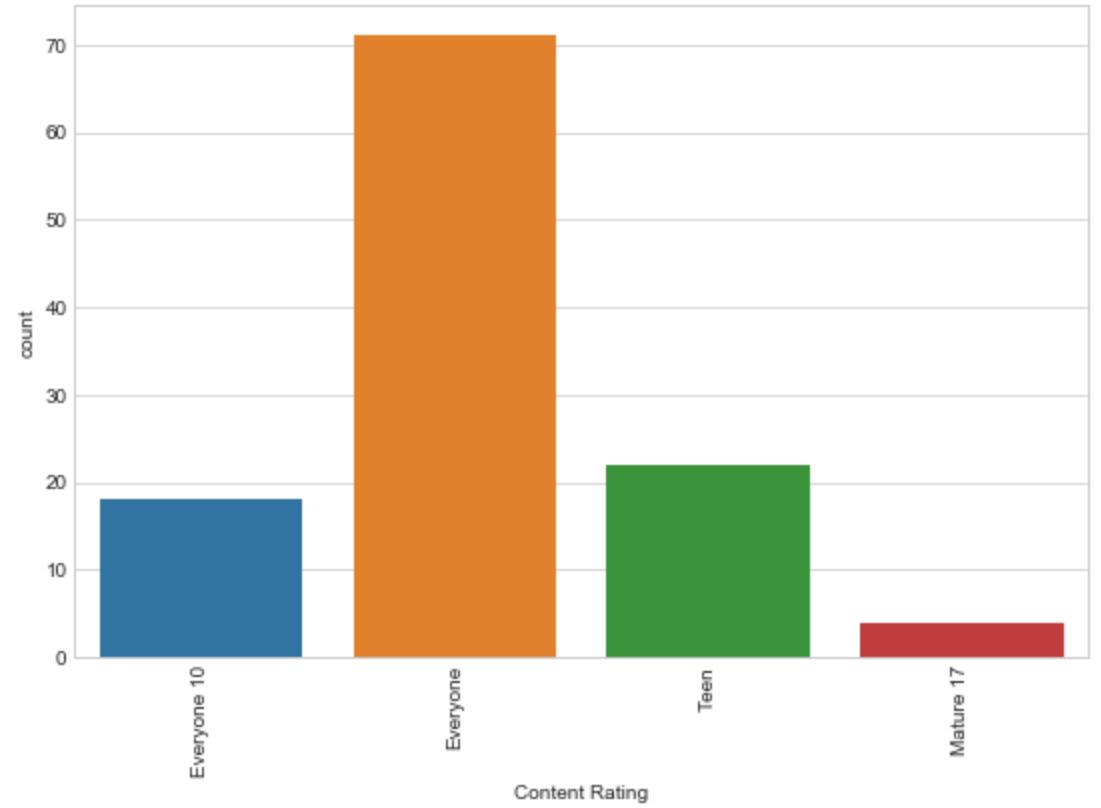
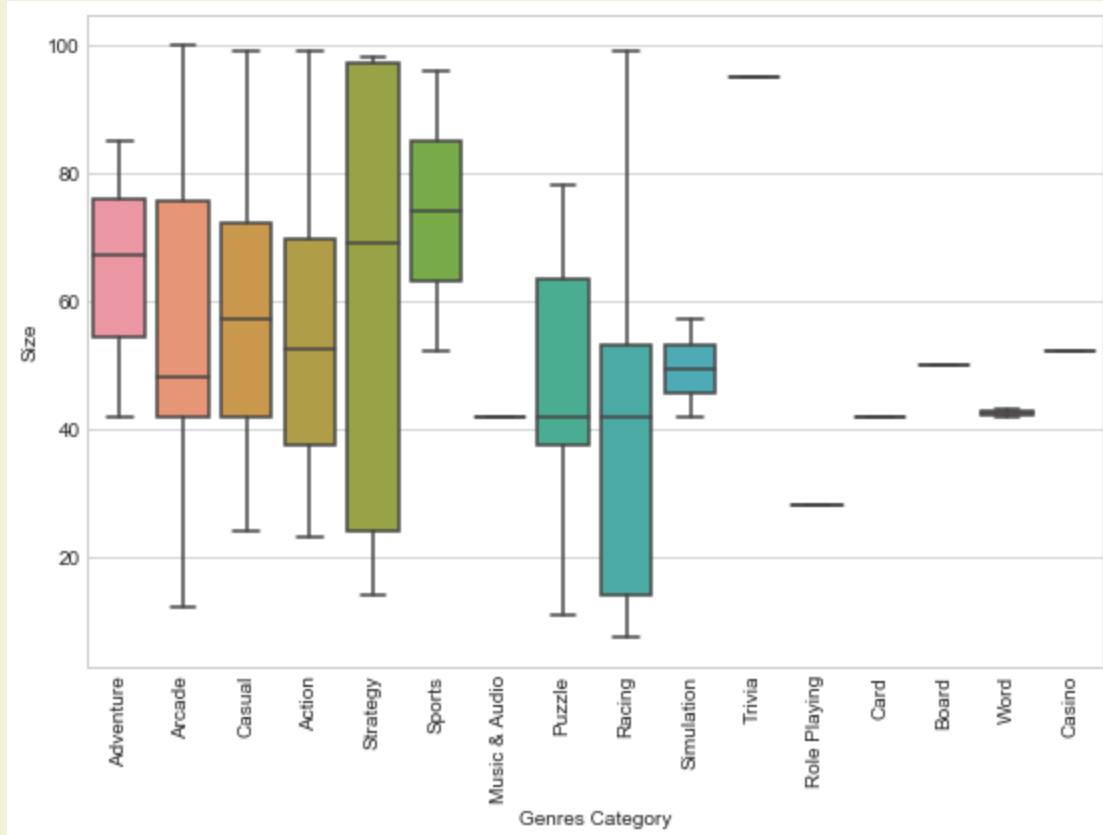


"Game" = High Install, High Review, High Comment, High Size



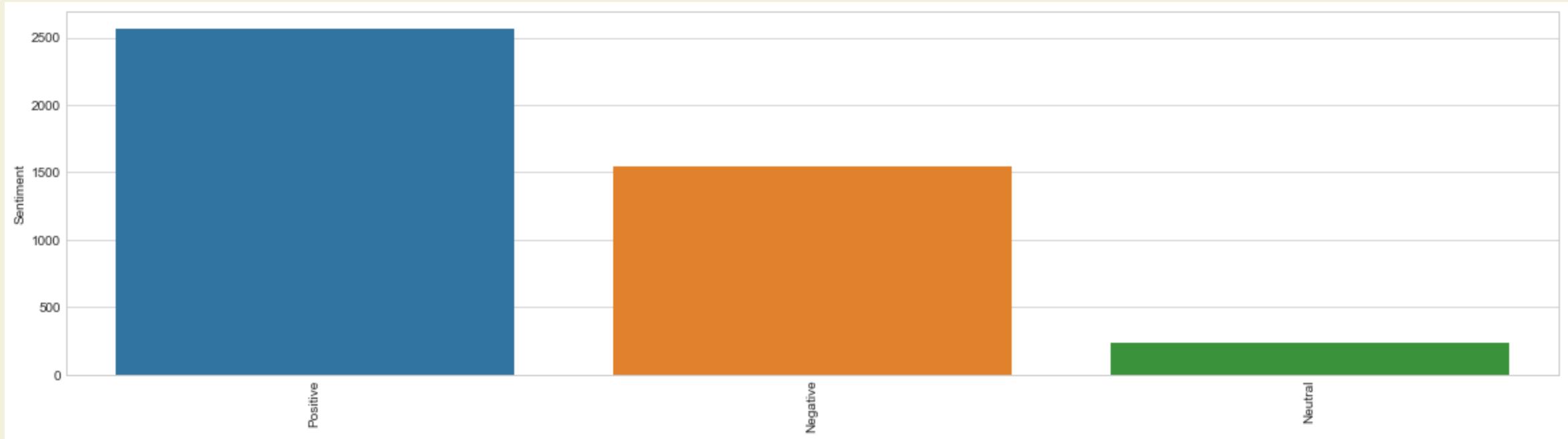


"Game" = High Install, High Review, High Comment, High Size



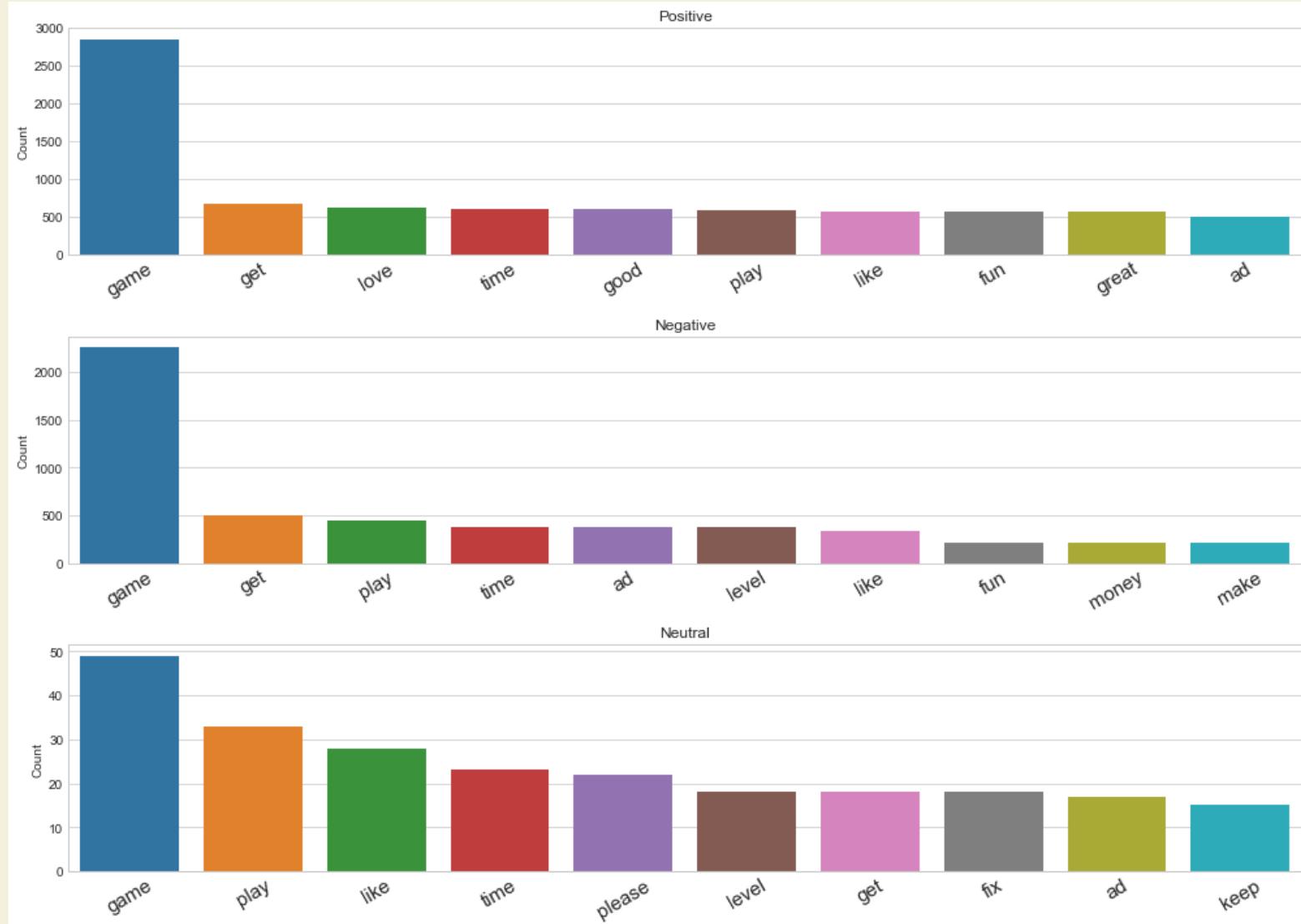


"Game" = High Install, High Review, High Comment, High Size



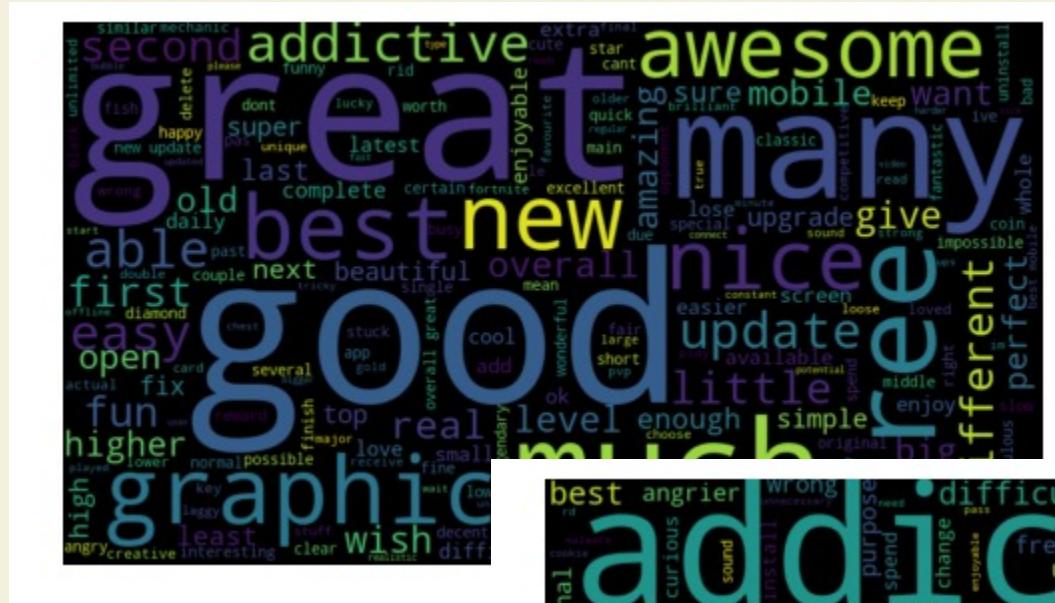


"Game" = High Install, High Review, High Comment, High Size





"Game" = High Install, High Review, High Comment, High Size





PREDICTIVE MODELING



Regression Model

- Predict Install

```
X = apps[["Rating", "Size", "Price", "Category Code", "Review_Log"]]  
y = apps["Install_Log"]
```



Regression Model

No tuning

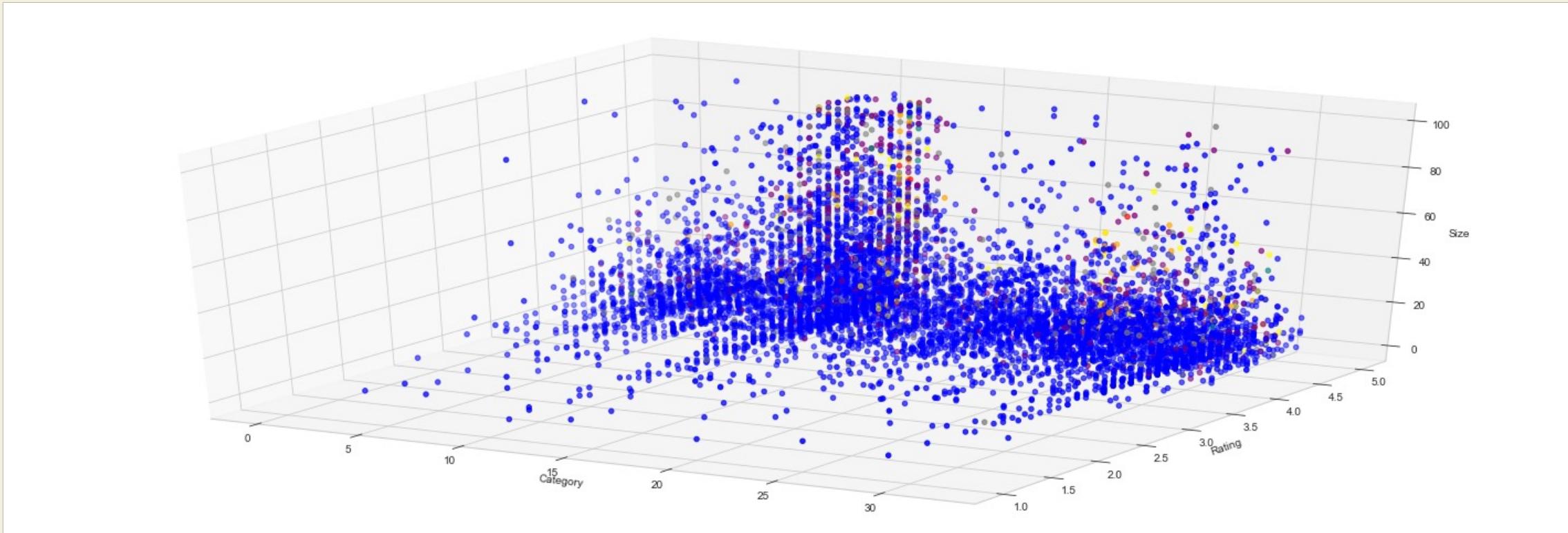
	MAE	MSE	RMSE	Score
DecisionTreeRegressor	0.9865	2.3079	1.5192	87.78%
RandomForestRegressor	0.8157	1.2120	1.1009	93.58%
XGBRegressor	0.8134	1.2019	1.0656	1.0963

Tuning

	MAE	MSE	RMSE	Score
DecisionTreeRegressor	1.0678	1.8425	1.3573	90.24%
RandomForestRegressor	0.7870	1.1355	1.0656	93.98%
XGBRegressor	0.7899	1.1335	1.0656	93.63%

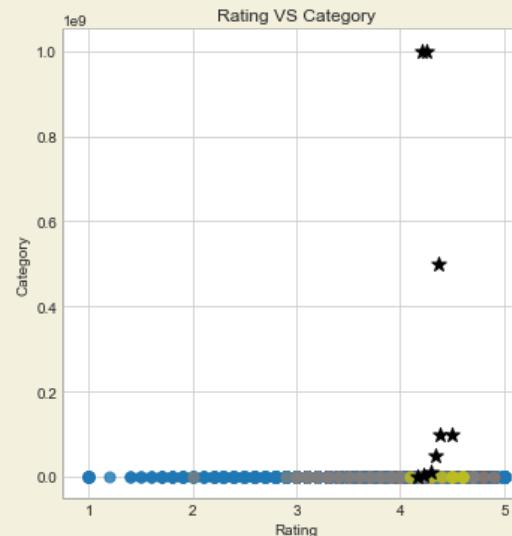
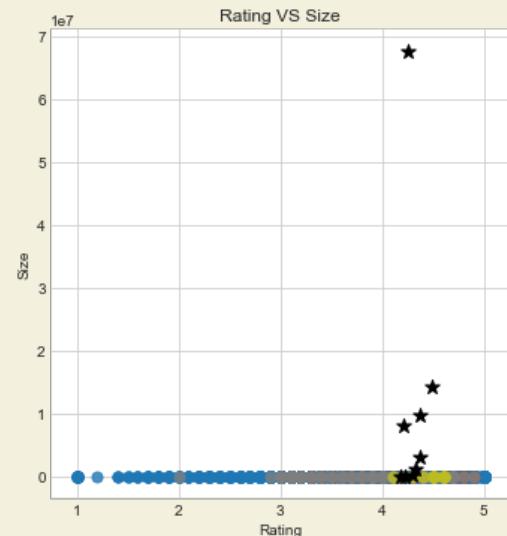
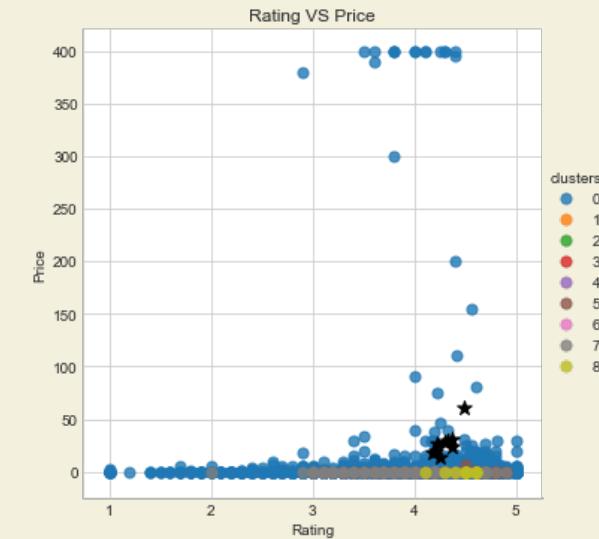
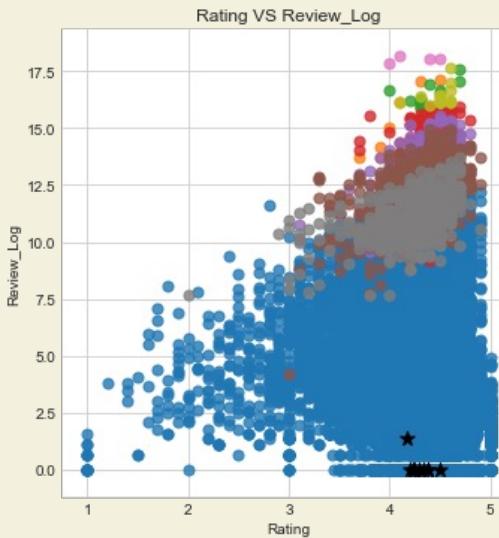
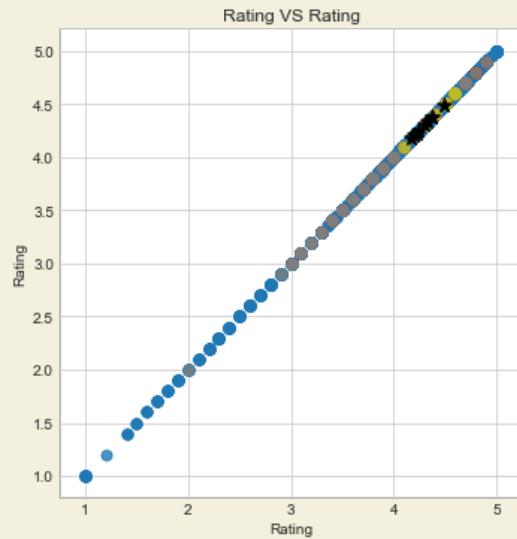


K-mean





K-mean



- Some results and this model is not well finished.



Related papers

- KNN Imputation Missing Value For Predictor App Rating on Google Play Using Random Forest Method
- App Store Analysis:Using Regression Model for App Downloads



THANK YOU