PROJECT :

Goal: Wrangle WeRateDogs Twitter data to create interesting and trustworthy analyses and visualizations. The Twitter archive is great, but it only contains very basic tweet information. Additional gathering, then assessing and cleaning is required for "Wow!"-worthy analyses and visualizations.

Gathering Data

i) Downloading image-predictions file

In [189]:

```
import requests
```

In [190]:

```
url=" https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/i
response=requests.get(url)
```

In [191]:

```
with open("image_predictions.tsv",mode='wb') as file:
    file.write(response.content)
```

In [192]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from datetime import datetime, timedelta
%matplotlib inline
```

In [193]:

```
ip=pd.read_csv("image_predictions.tsv",sep="\t")
```

In [194]:

```
ip.head(2)
```

Out[194]:

	tweet_id	jpg_url	img_num	
0	666020888022790149	https://pbs.twimg.com/media/CT4udn0WwAA0aMy.jpg	1	Welsh_spring
1	666029285002620928	https://pbs.twimg.com/media/CT42GRgUYAA5iDo.jpg	1	

2. Twitter API file

Using API code :

import tweepy from tweepy import OAuthHandler import json from timeit import default_timer as timer

consumer_key = 'HIDDEN' consumer_secret = 'HIDDEN' access_token = 'HIDDEN' access_secret = 'HIDDEN'

auth = OAuthHandler(consumer_key, consumer_secret) auth.set_access_token(access_token, access_secret)

api = tweepy.API(auth, wait_on_rate_limit=True)

tweet_ids = df_1.tweet_id.values len(tweet_ids)

Query Twitter's API for JSON data for each tweet ID in the Twitter archive

count = 0 fails_dict = {} start = timer()

Save each tweet's returned JSON as a new line in a .txt file

with open('tweet_json.txt', 'w') as outfile:

```
    # This loop will likely take 20-30 minutes to run because of Twitter's rate limit
    for tweet_id in tweet_ids:
        count += 1
        print(str(count) + ": " + str(tweet_id))
        try:
            tweet = api.get_status(tweet_id, tweet_mode='extended')
            print("Success")
            json.dump(tweet._json, outfile)
            outfile.write('\n')
        except tweepy.TweepError as e:
            print("Fail")
            fails_dict[tweet_id] = e
            pass
```

end = timer() print(end - start) print(fails_dict)

In [195]:

```
import json
```

In [196]:

```
tweet=pd.read_json("tweet-json.txt",lines=True)
```

In [197]:

```
tweet.head(2)
```

Out[197]:

	created_at	id	id_str	full_text	truncated	display_text_ra
0	2017-08-01 16:23:56+00:00	892420643555336193	892420643555336192	This is Phineas. He's a mystical boy. Only eve...	False	[0
1	2017-08-01 00:17:27+00:00	892177421306343426	892177421306343424	This is Tilly. She's just checking pup on you....	False	[0,

2 rows × 31 columns

3. twitter-archive-enhanced file

In [198]:

```
archive=pd.read_csv("twitter-archive-enhanced.csv")
```

In [199]:

```
archive.head(2)
```

Out[199]:

	tweet_id	in_reply_to_status_id	in_reply_to_user_id	timestamp	
0	892420643555336193	NaN	NaN	2017-08-01 16:23:56 +0000	href="http://twitter.c
1	892177421306343426	NaN	NaN	2017-08-01 00:17:27 +0000	href="http://twitter.c

Columns description

For detailed column description click on the links below:

1. Description on twitter (https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/tweet-object)

2. sfm (https://sfm.readthedocs.io/en/1.4.3/data_dictionary.html)

Accessing Data

1. archive Dataset

In [200]:

```
archive.head()
```

Out[200]:

	tweet_id	in_reply_to_status_id	in_reply_to_user_id	timestamp	sc
0	892420643555336193	NaN	NaN	2017-08-01 16:23:56 +0000	href="http://twitter.com/download/iph
1	892177421306343426	NaN	NaN	2017-08-01 00:17:27 +0000	href="http://twitter.com/download/iph
2	891815181378084864	NaN	NaN	2017-07-31 00:18:03 +0000	href="http://twitter.com/download/iph

In [201]:

```
#Statistical Reference
archive.describe()
```

Out[201]:

	tweet_id	in_reply_to_status_id	in_reply_to_user_id	retweeted_status_id	retweeted_s
count	2.356000e+03	7.800000e+01	7.800000e+01	1.810000e+02	
mean	7.427716e+17	7.455079e+17	2.014171e+16	7.720400e+17	
std	6.856705e+16	7.582492e+16	1.252797e+17	6.236928e+16	
min	6.660209e+17	6.658147e+17	1.185634e+07	6.661041e+17	
25%	6.783989e+17	6.757419e+17	3.086374e+08	7.186315e+17	
50%	7.196279e+17	7.038708e+17	4.196984e+09	7.804657e+17	
75%	7.993373e+17	8.257804e+17	4.196984e+09	8.203146e+17	
max	8.924206e+17	8.862664e+17	8.405479e+17	8.874740e+17	

In [202]:

```
archive.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2356 entries, 0 to 2355
Data columns (total 17 columns):
 #   Column                      Non-Null Count  Dtype  
---  ------                      --------------  -----  
 0   tweet_id                    2356 non-null   int64  
 1   in_reply_to_status_id       78 non-null     float64
 2   in_reply_to_user_id         78 non-null     float64
 3   timestamp                   2356 non-null   object 
 4   source                      2356 non-null   object 
 5   text                        2356 non-null   object 
 6   retweeted_status_id         181 non-null    float64
 7   retweeted_status_user_id    181 non-null    float64
 8   retweeted_status_timestamp  181 non-null    object 
 9   expanded_urls               2297 non-null   object 
 10  rating_numerator            2356 non-null   int64  
 11  rating_denominator          2356 non-null   int64  
 12  name                        2356 non-null   object 
 13  doggo                       2356 non-null   object 
```

In [203]:

```
#Finiding null values in dataset
archive.isnull().sum()
```

Out[203]:

```
tweet_id                       0
in_reply_to_status_id       2278
in_reply_to_user_id         2278
timestamp                      0
source                         0
text                           0
retweeted_status_id         2175
retweeted_status_user_id    2175
retweeted_status_timestamp  2175
expanded_urls                 59
rating_numerator               0
rating_denominator             0
name                           0
doggo                          0
floofer                        0
pupper                         0
puppo                          0
dtype: int64
```

In [204]:

```
#Duplicate Values
archive[archive.tweet_id.duplicated()]
```

Out[204]:

tweet_id	in_reply_to_status_id	in_reply_to_user_id	timestamp	source	text	retweeted_status

Checking rating_numerator and rating_denominator

In [205]:

```
archive[archive.rating_denominator == 0]
```

Out[205]:

	tweet_id	in_reply_to_status_id	in_reply_to_user_id	timestamp	
313	835246439529840640	8.352460e+17	26259576.0	2017-02-24 21:54:03 +0000	href="http://twitte

In [206]:

```
archive[archive.rating_numerator==0]
```

Out[206]:

	tweet_id	in_reply_to_status_id	in_reply_to_user_id	timestamp	
315	835152434251116546	NaN	NaN	2017-02-24 15:40:31 +0000	href="http://twitt
1016	746906459439529985	7.468859e+17	4.196984e+09	2016-06-26 03:22:31 +0000	href="http://twitt

In [207]:

```
archive.rating_denominator.value_counts()
```

Out[207]:

```
10     2333
11        3
50        3
80        2
20        2
2         1
16        1
40        1
70        1
15        1
90        1
110       1
120       1
130       1
150       1
170       1
7         1
0         1
Name: rating_denominator, dtype: int64
```

In [208]:

```
archive.rating_numerator.value_counts()
```

Out[208]:

```
12      558
11      464
10      461
13      351
9       158
8       102
7        55
14       54
5        37
6        32
3        19
4        17
1         9
2         9
420       2
0         2
15        2
75        2
```

In [209]:

```
#Dogs Names- Not Known
archive[archive.name=="None"].name.value_counts()
```

Out[209]:

```
None    745
Name: name, dtype: int64
```

In [210]:

```
#Checking the dog names
dogs = []
dogs= archive['name'].unique()
dogs.sort()
dogs
```

Out[210]:

```
array(['Abby', 'Ace', 'Acro', 'Adele', 'Aiden', 'Aja', 'Akumi', 'Al',
       'Albert', 'Albus', 'Aldrick', 'Alejandro', 'Alexander',
       'Alexanderson', 'Alf', 'Alfie', 'Alfy', 'Alice', 'Amber',
       'Ambrose', 'Amy', 'Amélie', 'Anakin', 'Andru', 'Andy', 'Angel',
       'Anna', 'Anthony', 'Antony', 'Apollo', 'Aqua', 'Archie', 'Arlen',
       'Arlo', 'Arnie', 'Arnold', 'Arya', 'Ash', 'Asher', 'Ashleigh',
       'Aspen', 'Astrid', 'Atlas', 'Atticus', 'Aubie', 'Augie', 'Autumn',
       'Ava', 'Axel', 'Bailey', 'Baloo', 'Balto', 'Banditt', 'Banjo',
       'Barclay', 'Barney', 'Baron', 'Barry', 'Batdog', 'Bauer', 'Baxter',
       'Bayley', 'BeBe', 'Bear', 'Beau', 'Beckham', 'Beebop', 'Beemo',
       'Bell', 'Bella', 'Belle', 'Ben', 'Benedict', 'Benji', 'Benny',
       'Bentley', 'Berb', 'Berkeley', 'Bernie', 'Bert', 'Bertson',
       'Betty', 'Beya', 'Biden', 'Bilbo', 'Billl', 'Billy', 'Binky',
       'Birf', 'Bisquick', 'Blakely', 'Blanket', 'Blipson', 'Blitz',
       'Bloo', 'Bloop', 'Blu', 'Blue', 'Bluebert', 'Bo', 'Bob', 'Bobb',
       'Bobbay', 'Bobble', 'Bobby', 'Bode', 'Bodie', 'Bonaparte', 'Bones',
       'Bookstore', 'Boomer', 'Boots', 'Boston', 'Bowie', 'Brad',
       'Bradlay', 'Bradley', 'Brady', 'Brandi', 'Brandonald', 'Brandy',
```

Dataset Issues:

i.QUALITY :

1. Missing values in [in_reply_to_status_id, in_reply_to_user_id, retweeted_status_id , retweeted_status_user_id, retweeted_status_timestamp, expanded_urls] columns
2. Rating_numerator and rating_denominator have some inconsistent values in the numerator and denominator.
3. tweet id 835246439529840640 has a rating of denominator = 0
4. in_reply_to_status_id, in_reply_to_user_id, retweeted_status_id, retweeted_status_user_id must be integers instead of float (They have id's similar to tweet_id)
5. timestamp and retweeted_status_timestamp are object but they should be datetime.
6. The columns doggo, floofer, pupper, puppo have missing values has None instead of NaN
7. Dogs name such as - 'a', 'actually', 'all', 'an', 'by', 'getting','his', 'incredibly', 'infuriating', 'just', 'life', 'light', 'mad', 'my', 'not', 'officially', 'old', 'one', 'quite', 'space', 'such','the', 'this', 'unacceptable','very' are not likely.

ii.Tidiness :

1. Dog stages are found in multiple columns, They should be passed in under single column. This will reduce the dimensionality of the dataframe

2. tweet Dataset

In [211]:

```
tweet.head()
```

Out[211]:

	created_at	id	id_str	full_text	truncated	display_text_range	
0	2017-08-01 16:23:56+00:00	892420643555336193	892420643555336192	This is Phineas. He's a mystical boy. Only eve...	False	[0, 85]	{'h ' 'user_
1	2017-08-01 00:17:27+00:00	892177421306343426	892177421306343424	This is Tilly. She's just checking pup on you....	False	[0, 138]	{'h ' 'user_
2	2017-07-31 00:18:03+00:00	891815181378084864	891815181378084864	This is Archie. He is a rare Norwegian Pounein	False	[0, 121]	{'h ' 'user_

In [212]:

```
tweet.describe()
```

Out[212]:

	id	id_str	in_reply_to_status_id	in_reply_to_status_id_str	in_reply_to_
count	2.354000e+03	2.354000e+03	7.800000e+01	7.800000e+01	7.8000
mean	7.426978e+17	7.426978e+17	7.455079e+17	7.455079e+17	2.0141
std	6.852812e+16	6.852812e+16	7.582492e+16	7.582492e+16	1.2527
min	6.660209e+17	6.660209e+17	6.658147e+17	6.658147e+17	1.1856
25%	6.783975e+17	6.783975e+17	6.757419e+17	6.757419e+17	3.0863
50%	7.194596e+17	7.194596e+17	7.038708e+17	7.038708e+17	4.1969
75%	7.993058e+17	7.993058e+17	8.257804e+17	8.257804e+17	4.1969
max	8.924206e+17	8.924206e+17	8.862664e+17	8.862664e+17	8.4054

In [213]:

```
tweet.isnull().sum()
```

Out[213]:

```
created_at                    0
id                            0
id_str                        0
full_text                     0
truncated                     0
display_text_range            0
entities                      0
extended_entities           281
source                        0
in_reply_to_status_id      2276
in_reply_to_status_id_str  2276
in_reply_to_user_id        2276
in_reply_to_user_id_str    2276
in_reply_to_screen_name    2276
user                          0
geo                        2354
coordinates                2354
place                      2353
```

In [214]:

```
#Checking for duplicates
tweet[tweet.id.duplicated()]
```

Out[214]:

created_at	id	id_str	full_text	truncated	display_text_range	entities	extended_entities	sou

0 rows × 31 columns

In [215]:

```
#Finding Retweet
tweet[tweet.retweeted_status.notna()==True]
```

Out[215]:

	created_at	id	id_str	full_text	truncated	display_text_r
31	2017-07-15 02:45:48+00:00	886054160059072513	886054160059072512	RT @Athletics: 12/10 #BATP https://t.co/WxwJmv...	False	[(
35	2017-07-13 01:35:06+00:00	885311592912609280	885311592912609280	RT @dog_rates: This is Lilly. She just paralle...	False	[0,
67	2017-06-26 00:13:58+00:00	879130579576475649	879130579576475648	RT @dog_rates: This is Emmy. She was adopted t...	False	[0,
72	2017-06-24 00:09:53+00:00	878404777348136964	878404777348136960	RT @dog_rates: Meet Shadow. In an attempt to r...	False	[0,
73	2017-06-23	878316110768087041	878316110768087040	RT @dog_rates: Meet Terrance. He's	False	[0,

In [216]:

```
#Finding reply Tweet
tweet[tweet.in_reply_to_status_id.notna()==True]
```

Out[216]:

	created_at	id	id_str	full_text	truncated	display_text_ra
29	2017-07-15 16:51:35+00:00	886267009285017600	886267009285017600	@NonWhiteHat @MayhewMayhem omg hello tanner yo...	False	[27,
54	2017-07-02 21:58:53+00:00	881633300179243008	881633300179243008	@roushfenway These are good dogs but 17/10 is ...	False	[13
63	2017-06-27 12:14:36+00:00	879674319642796034	879674319642796032	@RealKentMurphy 14/10 confirmed	False	[16
112	2017-06-02 19:38:25+00:00	870726314365509632	870726314365509632	@ComplicitOwl @ShopWeRateDogs >10/10 is res...	False	[30
				@Jack_Septic_Eye		

In [217]:

```
tweet.user[0]
```

Out[217]:

```
{'id': 4196983835,
 'id_str': '4196983835',
 'name': 'WeRateDogs™ (author)',
 'screen_name': 'dog_rates',
 'location': 'DM YOUR DOGS, WE WILL RATE',
 'description': '#1 Source for Professional Dog Ratings | STORE: @ShopWeRa
teDogs | IG, FB & SC: WeRateDogs MOBILE APP: @GoodDogsGame | Business: dog
ratingtwitter@gmail.com',
 'url': 'https://t.co/N7sNNHAEXS',
 'entities': {'url': {'urls': [{'url': 'https://t.co/N7sNNHAEXS',
    'expanded_url': 'http://weratedogs.com',
    'display_url': 'weratedogs.com',
    'indices': [0, 23]}]},
  'description': {'urls': []}},
 'protected': False,
 'followers_count': 3200889,
 'friends_count': 104,
 'listed_count': 2784,
```

Dataset Issues:

i.QUALITY :

1. Missing values in [geo, coordinates, place, contributors, possibly_sensitive, possibly_sensitive_appealable, retweeted_status, quoted_status_id, quoted_status_id_str, quoted_status, in_reply_to_status_id, in_reply_to_user_id, retweeted_status_id , retweeted_status_user_id, extended_entities] columns
2. We need to remove retweet and replay

ii.Tidiness :

1. User column has data in dictionaries and have several unrequired data stored, we need followers_count separatedly so we can access them easily.
2. Retweets and Favorites has to be joined to the archive data table, becuase all the tweets information is found there.

image-predictions(ip) Dataset

In [218]:

```
ip.head(5)
```

Out[218]:

	tweet_id	jpg_url	img_num	
0	666020888022790149	https://pbs.twimg.com/media/CT4udn0WwAA0aMy.jpg	1	Welsh_spring
1	666029285002620928	https://pbs.twimg.com/media/CT42GRgUYAA5iDo.jpg	1	
2	666033412701032449	https://pbs.twimg.com/media/CT4521TWwAEvMyu.jpg	1	German
3	666044226329800704	https://pbs.twimg.com/media/CT5Dr8HUEAA-lEu.jpg	1	Rhodesian
4	666049248165822465	https://pbs.twimg.com/media/CT5IQmsXIAAKY4A.jpg	1	miniature

In [219]:

```
ip.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2075 entries, 0 to 2074
Data columns (total 12 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   tweet_id  2075 non-null   int64
 1   jpg_url   2075 non-null   object
 2   img_num   2075 non-null   int64
 3   p1        2075 non-null   object
 4   p1_conf   2075 non-null   float64
 5   p1_dog    2075 non-null   bool
 6   p2        2075 non-null   object
 7   p2_conf   2075 non-null   float64
 8   p2_dog    2075 non-null   bool
 9   p3        2075 non-null   object
 10  p3_conf   2075 non-null   float64
 11  p3_dog    2075 non-null   bool
dtypes: bool(3), float64(3), int64(2), object(4)
memory usage: 152.1+ KB
```

In [220]:

```
#Checking for duplicate
ip[ip.tweet_id.duplicated()]
```

Out[220]:

tweet_id	jpg_url	img_num	p1	p1_conf	p1_dog	p2	p2_conf	p2_dog	p3	p3_conf	p3_dc

In [221]:

```
#Null Values
ip.isnull().sum()
```

Out[221]:

```
tweet_id    0
jpg_url     0
img_num     0
p1          0
p1_conf     0
p1_dog      0
p2          0
p2_conf     0
p2_dog      0
p3          0
p3_conf     0
p3_dog      0
dtype: int64
```

Dataset Issues:

i.QUALITY :

1. Only 2075 tweet_id have images.

ii.Tidiness :

1. All the prediction outputs from different algorithms have to be joined with arc hive and tweet.

Creating Copy of Orginal Datasets

In [222]:

```
tweet_clean=tweet.copy()
archive_clean=archive.copy()
ip_clean=ip.copy()
```

Cleaning

Tweet Dataset

Finding retweets and removing

In [223]:

```
tweet_clean[tweet_clean.retweeted_status.notna()==True]
```

Out[223]:

	created_at	id	id_str	full_text	truncated	display_text_r
31	2017-07-15 02:45:48+00:00	886054160059072513	886054160059072512	RT @Athletics: 12/10 #BATP https://t.co/WxwJmv...	False	[(
35	2017-07-13 01:35:06+00:00	885311592912609280	885311592912609280	RT @dog_rates: This is Lilly. She just paralle...	False	[0,
67	2017-06-26 00:13:58+00:00	879130579576475649	879130579576475648	RT @dog_rates: This is Emmy. She was adopted t...	False	[0,
72	2017-06-24 00:09:53+00:00	878404777348136964	878404777348136960	RT @dog_rates: Meet Shadow. In an attempt to r...	False	[0,
73	2017-06-23	878316110768087041	878316110768087040	RT @dog_rates: Meet Terrance. He's	False	[0,

In [224]:

```
tweet_clean.drop(tweet_clean[tweet_clean.retweeted_status.notna()==True].index,inplace=True
```

In [225]:

```
tweet_clean[tweet_clean.retweeted_status.notna()==True]
```

Out[225]:

	created_at	id	id_str	full_text	truncated	display_text_range	entities	extended_entities	sou

0 rows × 31 columns

In [226]:

```
tweet_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2175 entries, 0 to 2353
Data columns (total 31 columns):
 #   Column                         Non-Null Count  Dtype
---  ------                         --------------  -----
 0   created_at                     2175 non-null   datetime64[ns, UTC]
 1   id                             2175 non-null   int64
 2   id_str                         2175 non-null   int64
 3   full_text                      2175 non-null   object
 4   truncated                      2175 non-null   bool
 5   display_text_range             2175 non-null   object
 6   entities                       2175 non-null   object
 7   extended_entities              1994 non-null   object
 8   source                         2175 non-null   object
 9   in_reply_to_status_id          78 non-null     float64
 10  in_reply_to_status_id_str      78 non-null     float64
 11  in_reply_to_user_id            78 non-null     float64
 12  in_reply_to_user_id_str        78 non-null     float64
 13  in_reply_to_screen_name        78 non-null     object
```

In [227]:

```
tweet_clean.drop(["created_at",'id_str','display_text_range',"entities","is_quote_status","
```

In [228]:

```
tweet.info()
```

```
 13  in_reply_to_screen_name        78 non-null     object
 14  user                           2354 non-null   object
 15  geo                            0 non-null      float64
 16  coordinates                    0 non-null      float64
 17  place                          1 non-null      object
 18  contributors                   0 non-null      float64
 19  is_quote_status                2354 non-null   bool
 20  retweet_count                  2354 non-null   int64
 21  favorite_count                 2354 non-null   int64
 22  favorited                      2354 non-null   bool
 23  retweeted                      2354 non-null   bool
 24  possibly_sensitive             2211 non-null   float64
 25  possibly_sensitive_appealable  2211 non-null   float64
 26  lang                           2354 non-null   object
 27  retweeted_status               179 non-null    object
 28  quoted_status_id               29 non-null     float64
 29  quoted_status_id_str           29 non-null     float64
 30  quoted_status                  28 non-null     object
dtypes: bool(4), datetime64[ns, UTC](1), float64(11), int64(4), object(11)
```

Finding Reply Tweet and removing it.

In [229]:

```
tweet_clean[tweet_clean.in_reply_to_status_id.notna()==True]
```

Out[229]:

	id	full_text	in_reply_to_status_id	user	retweet_count	favorite_co
29	886267009285017600	@NonWhiteHat @MayhewMayhem omg hello tanner yo...	8.862664e+17	{'id': 4196983835, 'id_str': '4196983835', 'na...	4	
54	881633300179243008	@roushfenway These are good dogs but 17/10 is ...	8.816070e+17	{'id': 4196983835, 'id_str': '4196983835', 'na...	7	
63	879674319642796034	@RealKentMurphy 14/10 confirmed	8.795538e+17	{'id': 4196983835, 'id_str': '4196983835', 'na...	10	

In [230]:

```
tweet_clean.drop(tweet_clean[tweet_clean.in_reply_to_status_id.notna()==True].index,inplace
```

In [231]:

```
tweet_clean[tweet_clean.in_reply_to_status_id.notna()==True]
```

Out[231]:

	id	full_text	in_reply_to_status_id	user	retweet_count	favorite_count

In [232]:

```
tweet_clean.drop("in_reply_to_status_id",axis=1,inplace=True)
```

In [233]:

```
tweet_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2097 entries, 0 to 2353
Data columns (total 5 columns):
 #   Column          Non-Null Count  Dtype 
---  ------          --------------  ----- 
 0   id              2097 non-null   int64 
 1   full_text       2097 non-null   object
 2   user            2097 non-null   object
 3   retweet_count   2097 non-null   int64 
 4   favorite_count  2097 non-null   int64 
dtypes: int64(3), object(2)
memory usage: 98.3+ KB
```

In [234]:

```
tweet_clean.user[0]
```

Out[234]:


```
{'id': 4196983835,
 'id_str': '4196983835',
 'name': 'WeRateDogs™ (author)',
 'screen_name': 'dog_rates',
 'location': 'DM YOUR DOGS, WE WILL RATE',
 'description': '#1 Source for Professional Dog Ratings | STORE: @ShopWeRa
teDogs | IG, FB & SC: WeRateDogs MOBILE APP: @GoodDogsGame | Business: dog
ratingtwitter@gmail.com',
 'url': 'https://t.co/N7sNNHAEXS',
 'entities': {'url': {'urls': [{'url': 'https://t.co/N7sNNHAEXS',
     'expanded_url': 'http://weratedogs.com',
     'display_url': 'weratedogs.com',
     'indices': [0, 23]}]},
  'description': {'urls': []}},
 'protected': False,
 'followers_count': 3200889,
 'friends_count': 104,
 'listed_count': 2784,
```


In [235]:

```
#Finding overall followers
tweet_clean["followers_counttweet"]=tweet_clean.user.apply(lambda x:x['followers_count'])
```

In [236]:

```
tweet_clean.drop("user",axis=1,inplace=True)
```

Finding Duplicate

In [237]:

```
tweet_clean[tweet_clean.duplicated()]
```

Out[237]:

	id	full_text	retweet_count	favorite_count	followers_counttweet

Changing column name 'id' to 'tweet_id'

In [238]:

```
tweet_clean.rename(columns={"id" : "tweet_id"},inplace=True)
```

2. Archive Dataset

In [239]:

```
archive_clean.drop(["in_reply_to_status_id","in_reply_to_user_id","source","retweeted_statu
```

Creating Dog Stages column

In [240]:

```
archive_clean["stages"]=archive.apply(lambda x: x.doggo if x.doggo=="doggo" else ("floofer"
```

In [241]:

```
archive_clean.drop(["doggo","floofer","pupper","puppo"],axis=1,inplace=True)
```

In [242]:

```
archive_clean[archive_clean.stages!="None"]
```

Out[242]:

	tweet_id	timestamp	rating_numerator	rating_denominator	name	stages
9	890240255349198849	2017-07-26 15:59:51 +0000	14	10	Cassie	doggo
12	889665388333682689	2017-07-25 01:55:32 +0000	13	10	None	puppo
14	889531135344209921	2017-07-24 17:02:04 +0000	13	10	Stuart	puppo
29	886366144734445568	2017-07-15 23:25:31 +0000	12	10	Roscoe	pupper
43	884162670584377345	2017-07-09 21:29:42 +0000	12	10	Yogi	doggo
...	...	...	...	...	...	...
1995	672594978741354496	2015-12-04 01:55:13 +0000	9	10	Scott	pupper
2002	672481316919734272	2015-12-03 18:23:34 +0000	12	10	Jazz	pupper
2009	672254177670729728	2015-12-03 03:21:00 +0000	11	10	Rolf	pupper
2015	672205392827572224	2015-12-03 00:07:09 +0000	9	10	Opal	pupper
2017	672160042234327040	2015-12-02 21:06:56 +0000	8	10	Bubba	pupper

380 rows × 6 columns

Changing timestamp object to datetime

In [243]:

```
archive_clean["timestamp"] = pd.to_datetime(archive_clean.timestamp)
```

In [244]:

```
archive_clean.query("name in ['a', 'actually', 'all', 'an', 'by', 'getting','his', 'incredi
```

Out[244]:

	tweet_id	timestamp	rating_numerator	rating_denominator	name	stages
22	887517139158093824	2017-07-19 03:39:09+00:00	14	10	such	None
56	881536004380872706	2017-07-02 15:32:16+00:00	14	10	a	pupper
118	869988702071779329	2017-05-31 18:47:24+00:00	12	10	quite	None
169	859196978902773760	2017-05-02 00:04:57+00:00	12	10	quite	None
193	855459453768019968	2017-04-21 16:33:22+00:00	12	10	quite	None
...	...	...	...	...	...	...
2349	666051853826850816	2015-11-16 00:35:11+00:00	2	10	an	None
2350	666050758794694657	2015-11-16 00:30:50+00:00	10	10	a	None
2352	666044226329800704	2015-11-16 00:04:52+00:00	6	10	a	None
2353	666033412701032449	2015-11-15 23:21:54+00:00	9	10	a	None
2354	666029285002620928	2015-11-15 23:05:30+00:00	7	10	a	None

109 rows × 6 columns

Changing unacceptable names to "unknown"

In [245]:

```
cr_name=["None",'a', 'actually', 'all', 'an', 'by', 'getting','his', 'incredibly', 'infuria
archive_clean.name=archive_clean.name.apply(lambda x:"Unknown" if (x in cr_name) else x)
```

In [246]:

```
archive_clean.query("name in ['a', 'actually', 'all', 'an', 'by', 'getting','his', 'incredi
```

Out[246]:

tweet_id	timestamp	rating_numerator	rating_denominator	name	stages

In [247]:

```
merge1=tweet_clean.merge(archive_clean,how='inner').reset_index(drop=True)
```

In [248]:

```
merge1=merge1.merge(ip_clean,how='inner').reset_index(drop=True)
```

In [249]:

```
#Complete Dataset
merge1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1971 entries, 0 to 1970
Data columns (total 21 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   tweet_id              1971 non-null   int64
 1   full_text             1971 non-null   object
 2   retweet_count         1971 non-null   int64
 3   favorite_count        1971 non-null   int64
 4   followers_counttweet  1971 non-null   int64
 5   timestamp             1971 non-null   datetime64[ns, UTC]
 6   rating_numerator      1971 non-null   int64
 7   rating_denominator    1971 non-null   int64
 8   name                  1971 non-null   object
 9   stages                1971 non-null   object
 10  jpg_url               1971 non-null   object
 11  img_num               1971 non-null   int64
 12  p1                    1971 non-null   object
 13  p1_conf               1971 non-null   float64
```

Fixing Non-Integer rating_numerator Issue

In [250]:

```
pattern = "(\d+\.\d+\/\d+)"

merge1.full_text.str.extract(pattern, expand = True)[0].dropna()
```

Out[250]:

```
39       13.5/10
499       9.75/10
549     11.27/10
1359    11.26/10
Name: 0, dtype: object
```

In [251]:

```
num = merge1.full_text.str.extract(pattern, expand = True)[0].dropna().str.split('/', n=1,
```

In [252]:

```
num_index = num.index
num_values = num.values.astype("float64")
```

In [253]:

```
merge1.rating_numerator = merge1.rating_numerator.astype("float64")
merge1.rating_denominator = merge1.rating_denominator.astype("float64")
merge1.loc[num_index, "rating_numerator"] = num_values
merge1.loc[num_index].rating_numerator
```

Out[253]:

```
39      13.50
499      9.75
549     11.27
1359    11.26
Name: rating_numerator, dtype: float64
```

Storing the Cleaned Dataset

In [254]:

```
merge1.to_csv('twitter_archive_master.csv', index=False)
```

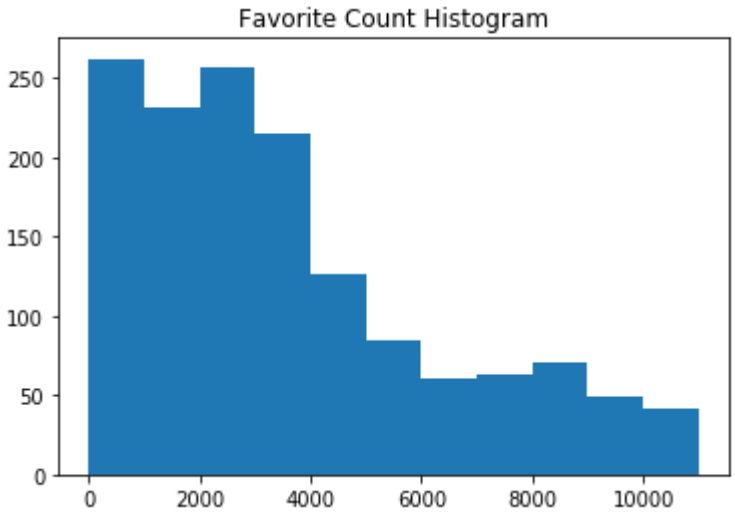
Analyzing Data

In [255]:

```
plt.hist(merge1.favorite_count,bins=np.arange(0,12000,1000))
plt.title("Favorite Count Histogram")
```

Out[255]:

```
Text(0.5, 1.0, 'Favorite Count Histogram')
```

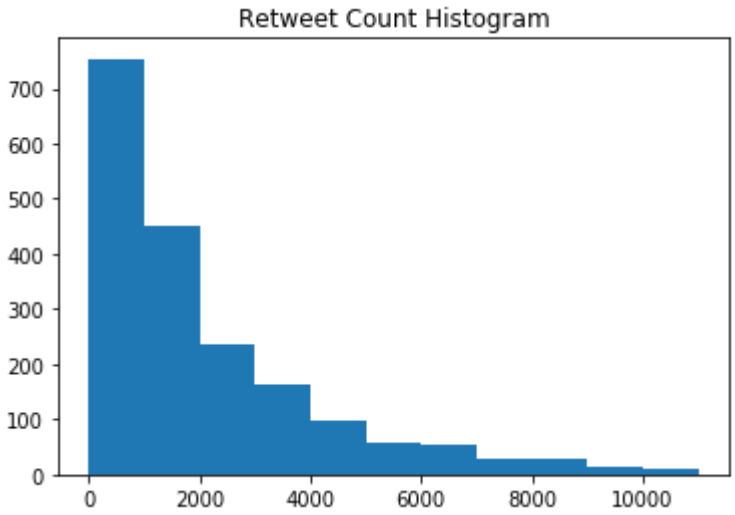


In [256]:

```
plt.hist(merge1.retweet_count,bins=np.arange(0,12000,1000))
plt.title("Retweet Count Histogram")
```

Out[256]:

```
Text(0.5, 1.0, 'Retweet Count Histogram')
```



Top 10 based on favorite count

In [257]:

```
top=merge1.sort_values(by=['favorite_count'],ascending=False).reset_index(drop=True)
top10=top[:10]
top10[["tweet_id","full_text","favorite_count","name","jpg_url"]]
```

Out[257]:

	tweet_id	full_text	favorite_count	name	
0	822872901745569793	Here's a super supportive puppo participating ...	132810	Unknown	https://pbs.twimg.com/media/C2tugX
1	744234799360020481	Here's a doggo realizing you can stand in a po...	131075	Unknown	https://pbs.twimg.com/ext_tw_video_
2	879415818425184262	This is Duddles. He did an attempt. 13/10 some...	107956	Duddles	https://pbs.twimg.com/ext_tw_video_
3	807106840509214720	This is Stephan. He just wants to help. 13/10 ...	107015	Stephan	https://pbs.twimg.com/ext_tw_video_
4	866450705531457537	This is Jamesy. He gives a kiss to every other...	106827	Jamesy	https://pbs.twimg.com/media/DAZAUfE
5	819004803107983360	This is Bo. He was a very good First Doggo. 14...	95450	Bo	https://pbs.twimg.com/media/C12whD
6	870374049280663552	This is Zoey. She really likes the planet. Wou...	85011	Zoey	https://pbs.twimg.com/media/DBQwlF
7	806629075125202948	"Good afternoon class today we're going to lea...	75639	Unknown	https://pbs.twimg.com/media/CzG425r
8	859196978902773760	We only rate dogs. This is quite clearly a smo...	75193	Unknown	https://pbs.twimg.com/ext_tw_video_

	tweet_id	full_text	favorite_count	name	
9	739238157791694849	Here's a doggo blowing bubbles. It's downright...	75163	Unknown	https://pbs.twimg.com/ext_tw_video_

In [258]:

```
print(top10.full_text[0])
top10.jpg_url[0]
```

```
Here's a super supportive puppo participating in the Toronto  #WomensMarch t
oday. 13/10 https://t.co/nTz3FtorBc (https://t.co/nTz3FtorBc)
```

Out[258]:

```
'https://pbs.twimg.com/media/C2tugXLXgAArJO4.jpg'
```

Top 10 based on retweet_count

In [259]:

```
retop=merge1.sort_values(by=['retweet_count'],ascending=False).reset_index(drop=True)
retop10=retop[:10]
retop10[["tweet_id","full_text","retweet_count","name","jpg_url"]]
```

Out[259]:

	tweet_id	full_text	retweet_count	
0	744234799360020481	Here's a doggo realizing you can stand in a po...	79515	Unk
1	807106840509214720	This is Stephan. He just wants to help. 13/10 ...	56625	Ste
2	739238157791694849	Here's a doggo blowing bubbles. It's downright...	52360	Unk
3	822872901745569793	Here's a super supportive puppo participating ...	48265	Unk
4	879415818425184262	This is Duddles. He did an attempt. 13/10 some...	45849	Du
5	819004803107983360	This is Bo. He was a very good First Doggo. 14...	42228	
6	806629075125202948	"Good afternoon class today we're going to lea...	37911	Unk
7	761672994376806400	Ohboyohboyohboyohboyohboyohboyohboyohboyohboyo...	33421	Unk
8	866450705531457537	This is Jamesy. He gives a kiss to every other...	32883	Ja
9	676219687039057920	This is Kenneth. He's stuck in a bubble. 10/10...	31989	Ke

In [260]:

```
print(retop10.jpg_url[0])
print(retop10.full_text[0])
```

```
https://pbs.twimg.com/ext_tw_video_thumb/744234667679821824/pu/img/1GaWmtJtd
qzZV7jy.jpg (https://pbs.twimg.com/ext_tw_video_thumb/744234667679821824/pu/
img/1GaWmtJtdqzZV7jy.jpg)
Here's a doggo realizing you can stand in a pool. 13/10 enlightened af (vid
 by Tina Conrad) https://t.co/7wE9LTEXC4 (https://t.co/7wE9LTEXC4)
```

Time period of Dataset

In [261]:

```
(merge1.timestamp[len(merge1)-1],merge1.timestamp[0])
```

Out[261]:

```
(Timestamp('2015-11-15 22:32:08+0000', tz='UTC'),
 Timestamp('2017-08-01 16:23:56+0000', tz='UTC'))
```

Change in No. of Followers during the time period '2015-11-15 22:32:08' - '2017-08-01 16:23:56'

In [262]:

```
merge1.followers_counttweet[0]-merge1.followers_counttweet[len(merge1)-1]
```

Out[262]:

```
-129
```

Note: The number of followers decreased by 129.

Number of tweets without dog names

In [263]:

```
named=merge1[merge1.name=="unknown"].tweet_id.count()
unnamed=merge1.shape[0]-named
unnamed
```

Out[263]:

```
1971
```

Note: There are 1349 tweets without dog name.

Correlation between numeric columns [retweet_count ,favorite_count, rating_numerator]

In [264]:

```
column=["retweet_count","favorite_count", "rating_numerator"]
merge1[column].corr()
```

Out[264]:

	retweet_count	favorite_count	rating_numerator
retweet_count	1.000000	0.913014	0.014238
favorite_count	0.913014	1.000000	0.010596
rating_numerator	0.014238	0.010596	1.000000

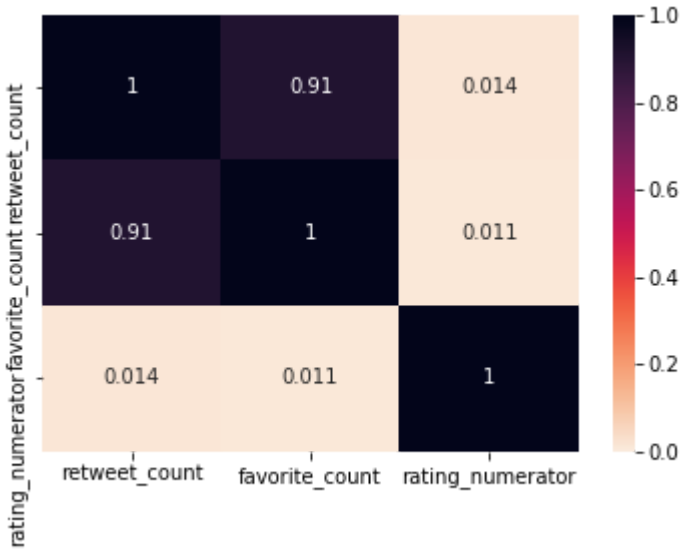
Heatmap for correlation

In [265]:

```
sns.heatmap(merge1[column].corr(),cmap="rocket_r",annot=True,vmin=0)
```

Out[265]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x2a1a8aba608>
```



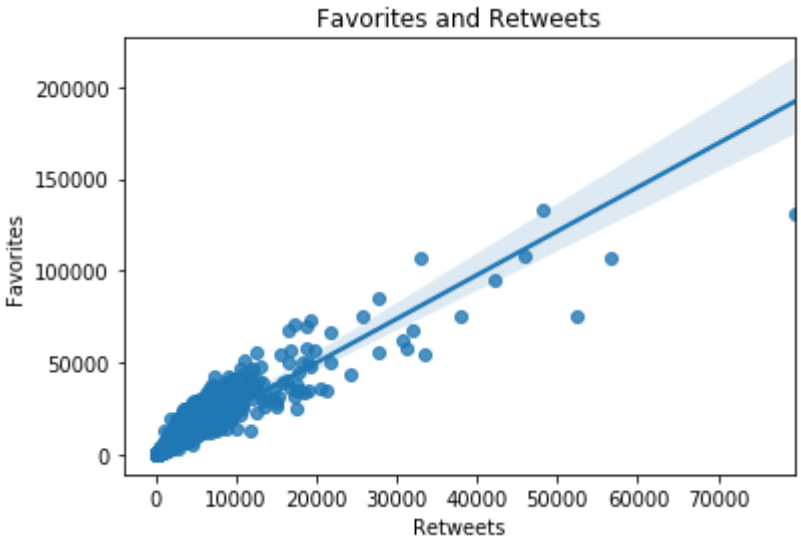
Note: There is strong correlation between retweet counts and favorite counts.

In [161]:

```
graph = sns.regplot(x=merge1.retweet_count, y=merge1.favorite_count)
plt.title("Favorites and Retweets")
plt.xlabel('Retweets')
plt.ylabel('Favorites')
```

Out[161]:

```
Text(0, 0.5, 'Favorites')
```



Note: As the number of retweets increases, the number of favorites also increases.