Wrangle report

- **Data source:**

  1- File [" twitter-archive-enhanced.csv ".]

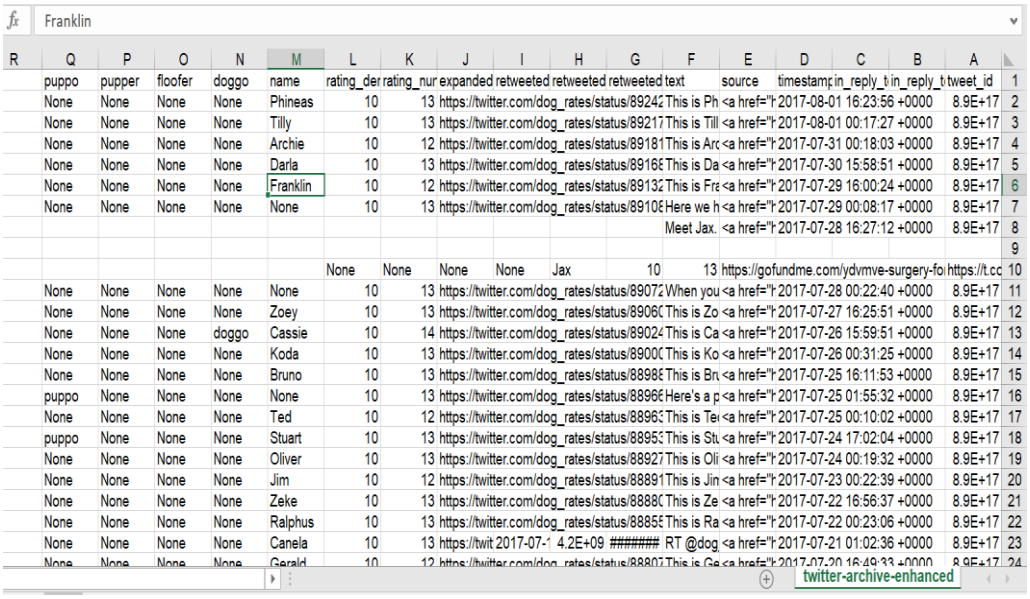
  2- The tweet image predictions, https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/image-predictions.tsv

  3- Using the tweet IDs in the WeRateDogs Twitter archive, query the Twitter API for each tweet's JSON data using Python's Tweepy library and store each tweet's entire set of JSON data in a file called tweet_json.txt file.

- **Assessing data:**

  The second step after collecting the data is to evaluate the data, and it is as follows:

  1- Visual evaluation, by using Excel, through which data can be viewed and problems identified as in the picture:



	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A	
	puppo	pupper	floofer	doggo	name	rating_den	rating_nur	expanded	retweeted	retweeted	retweeted	text	source	timestamp	in_reply_t	in_reply_t	tweet_id	1
	None	None	None	None	Phineas	10	13	https://twitter.com/dog_rates/status/89242				This is Ph	<a href="h	2017-08-01	16:23:56	+0000	8.9E+17	2
	None	None	None	None	Tilly	10	13	https://twitter.com/dog_rates/status/89217				This is Till	<a href="h	2017-08-01	00:17:27	+0000	8.9E+17	3
	None	None	None	None	Archie	10	12	https://twitter.com/dog_rates/status/89181				This is Arc	<a href="h	2017-07-31	00:18:03	+0000	8.9E+17	4
	None	None	None	None	Darla	10	13	https://twitter.com/dog_rates/status/89168				This is Da	<a href="h	2017-07-30	15:58:51	+0000	8.9E+17	5
	None	None	None	None	Franklin	10	12	https://twitter.com/dog_rates/status/89132				This is Fra	<a href="h	2017-07-29	16:00:24	+0000	8.9E+17	6
	None	None	None	None	None	10	13	https://twitter.com/dog_rates/status/89108				Here we h	<a href="h	2017-07-29	00:08:17	+0000	8.9E+17	7
												Meet Jax.	<a href="h	2017-07-28	16:27:12	+0000	8.9E+17	8
																		9
						None	None	None	None	Jax	10	13	https://gofundme.com/ydvmve-surgery-fo				https://t.co	10
	None	None	None	None	None	10	13	https://twitter.com/dog_rates/status/89072				When you	<a href="h	2017-07-28	00:22:40	+0000	8.9E+17	11
	None	None	None	None	Zoey	10	13	https://twitter.com/dog_rates/status/89060				This is Zo	<a href="h	2017-07-27	16:25:51	+0000	8.9E+17	12
	None	None	None	doggo	Cassie	10	14	https://twitter.com/dog_rates/status/89024				This is Ca	<a href="h	2017-07-26	15:59:51	+0000	8.9E+17	13
	None	None	None	None	Koda	10	13	https://twitter.com/dog_rates/status/89000				This is Ko	<a href="h	2017-07-26	00:31:25	+0000	8.9E+17	14
	None	None	None	None	Bruno	10	13	https://twitter.com/dog_rates/status/88988				This is Bru	<a href="h	2017-07-25	16:11:53	+0000	8.9E+17	15
	puppo	None	None	None	None	10	13	https://twitter.com/dog_rates/status/88966				Here's a p	<a href="h	2017-07-25	01:55:32	+0000	8.9E+17	16
	None	None	None	None	Ted	10	12	https://twitter.com/dog_rates/status/88963				This is Ted	<a href="h	2017-07-25	00:10:02	+0000	8.9E+17	17
	puppo	None	None	None	Stuart	10	13	https://twitter.com/dog_rates/status/88953				This is Stu	<a href="h	2017-07-24	17:02:04	+0000	8.9E+17	18
	None	None	None	None	Oliver	10	13	https://twitter.com/dog_rates/status/88927				This is Oli	<a href="h	2017-07-24	00:19:32	+0000	8.9E+17	19
	None	None	None	None	Jim	10	12	https://twitter.com/dog_rates/status/88891				This is Jim	<a href="h	2017-07-23	00:22:39	+0000	8.9E+17	20
	None	None	None	None	Zeke	10	13	https://twitter.com/dog_rates/status/88880				This is Ze	<a href="h	2017-07-22	16:56:37	+0000	8.9E+17	21
	None	None	None	None	Ralphus	10	13	https://twitter.com/dog_rates/status/88855				This is Ra	<a href="h	2017-07-22	00:23:06	+0000	8.9E+17	22
	None	None	None	None	Canela	10	13	https://twit	2017-07-1	4.2E+09	########	RT @dog	<a href="h	2017-07-21	01:02:36	+0000	8.9E+17	23
	None	None	None	None	Gerald	10	12	https://twitter.com/dog_rates/status/88807				This is Ge	<a href="h	2017-07-20	16:49:33	+0000	8.9E+17	24

  2- Programmatic evaluation and this is through the use of codes that show the data accurately and easily, and from these symbols

  Df_3 - df_2_clean22.head() - df_1.tail(50)

In [6]:
```
# See the data for the WeRateDogs table
df_1
```

Out[6]:

	tweet_id	in_reply_to_status_id	in_reply_to_user_id	timestamp	source	text	retweeted_status_id	retweete
0	892420643555336193	NaN	NaN	2017-08-01 16:23:56 +0000	<a href="http://twitter.com/download/iphone" r...	This is Phineas. He's a mystical boy. Only eve...	NaN	
1	892177421306343426	NaN	NaN	2017-08-01 00:17:27 +0000	<a href="http://twitter.com/download/iphone" r...	This is Tilly. She's just checking pup on you....	NaN	
2	891815181378084864	NaN	NaN	2017-07-31 00:18:03 +0000	<a href="http://twitter.com/download/iphone" r...	This is Archie. He is a rare Norwegian Pouncin...	NaN	
3	891689557279858688	NaN	NaN	2017-07-30 15:58:51 +0000	<a href="http://twitter.com/download/iphone" r...	This is Darla. She commenced a snooze mid meal...	NaN	
				2017-07-		This is		

- Cleaning data:

3- Merging and preservation

Quality:

- **<u>twitter_archive_enhanced</u>**

These are some of the problems with quality:

- need to remove all rows that have values (not blank or non-null) in retweeted.
- Missing data (NaN):
  1- in_reply_to_status_id
  2- in_reply_to_user_id
  3- retweeted_status_id
  4- retweeted_status_user_id
  5- retweeted_status_timestamp
  6- expanded_urls .
- The data type changed to string:
  1- tweet_id
- The type of timestamp need to change from object to datetype.
- There are non-dog names in name column such as 'a', ' such' , 'the', etc. These cases are in lowercase as a result of the way the names were extracted.
- remove the outliers in column data in rating_numerator
-
-

- **<u>The tweet image predictions</u>**

- The data type changed to string:
  1- tweet_id
  2- img_num

- **tweet_json.txt**

1- The data type changed to string: 1- tweet_id

- The type of timestamp need to change from object to datetype
- Column names change “favorites to favorite_count”

Tidiness:

We find some of the problems with Tidiness:

- **twitter_archive_enhanced**

- dog stages can be organized into a column for doggo, pupper, pupper and floofer

- **The tweet image predictions**

- df_1, df_2 and df_3 Combine them into one table **tweet_json.txt**

Codes that were used:

- **del df_1_clean11['in_reply_to_status_id']**
- **del df_1_clean11['in_reply_to_user_id']**
- **del df_1_clean11['retweeted_status_id']**
- **del df_1_clean11['retweeted_status_user_id']**
- **del df_1_clean11['retweeted_status_timestamp']**
- **df_1_clean11['tweet_id']=df_1_clean11['tweet_id'].astype(object)**
- **df_1_clean11['timestamp'] = pd.to_datetime(df_1_clean11['timestamp'])**
- **df_1_clean11['timestamp'] = pd.to_datetime(df_1_clean11['timestamp'])**
- **mask = df_1_clean11.name.str.islower()**
  column_name = 'name'
  df_1_clean11.loc[mask, column_name] = np.nan
- **df_1_clean11.loc[df_1.rating_numerator >=100, 'rating_numerator'] = df_1.rating_numerator.mean()**
- **df_1_clean11.doggo.replace('None','',inplace=True)**
- **df_1_clean11.floofer.replace('None','',inplace=True)**
- **df_1_clean11.pupper.replace('None','',inplace=True)**
- **df_1_clean11.puppo.replace('None','',inplace=True)**
- **df_1_clean11['stage'] = df_1_clean11.doggo + df_1_clean11.floofer +df_1_clean11.pupper + df_1_clean11.puppo**
- **df_1_clean11.loc[df_1_clean11.stage == 'doggopupper', 'stage'] = 'doggo, pupper'**

- **df_1_clean11.loc[df_1_clean11.stage == 'doggopuppo', 'stage'] = 'doggo, puppo'**
- **df_1_clean11.loc[df_1_clean11.stage == 'doggofloofer', 'stage'] = 'doggo, floofer'**
- **df_1_clean11.loc[df_1_clean11.stage == '', 'stage'] = np.nan**
- **del df_1_clean11['doggo']**
- **del df_1_clean11['floofer']**
- **del df_1_clean11['pupper']**
- **del df_1_clean11['puppo']**
- **df_2_clean22['tweet_id']=df_2_clean22['tweet_id'].astype(object)**
- **df_2_clean22['img_num']=df_2_clean22['img_num'].astype(object)**
- **def predict_breed(row):**

  if row.p1_dog:

  return row.p1

  elif row.p2_dog:

  return row.p2

  elif row.p3_dog:

  return row.p3
- **df_2_clean22['dog_breed'] = df_2_clean22.apply(lambda row: predict_breed(row),axis=1)**
- **df_3_clean33['tweet_id']=df_3_clean33['tweet_id'].astype(object)**
- **df_3_clean33['timestamp'] = pd.to_datetime(df_3_clean33['timestamp'])**
- **df_3_clean33=df_3_clean33.rename(columns={'favorites':'favorite-count'})**
- **df_temp_clean = pd.merge(df_1_clean11, df_3_clean33, on='tweet_id', how='inner')**
- **df_master_clean = pd.merge(df_temp_clean, df_2_clean22, on='tweet_id', how='inner')**

- **After cleaning, the data is combined into one file and saved to the file by use the code:**

  df_master_clean.to_csv('twitter_archive_collected.csv')