

# Act report

In this report we will try to show the final results of the project, and some basic visualizations on the data worked

Sample twitter\_archive\_master.csv

```
In [10]: twitter_archive_master.info()
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1365 entries, 0 to 1364
Data columns (total 18 columns):
#   Column                Non-Null Count  Dtype
---  -
0   tweet_id              1365 non-null   int64
1   timestamp              1365 non-null   datetime64[ns, UTC]
2   source                 1365 non-null   object
3   text                   1365 non-null   object
4   expanded_urls          1365 non-null   object
5   name                   1365 non-null   object
6   ratings                1365 non-null   float64
7   dog_stages             182 non-null    object
8   hour                   1365 non-null   int64
9   full_week              1365 non-null   object
10  jpg_url                 1365 non-null   object
11  favorite_count          1365 non-null   float64
12  retweet_count           1365 non-null   float64
13  tweet_length            1365 non-null   int64
14  language                1365 non-null   object
15  predicted_dog            1365 non-null   object
16  predicted_precision      1365 non-null   float64
17  is_dog                  1365 non-null   object
dtypes: datetime64[ns, UTC](1), float64(4), int64(3), object(10)
memory usage: 202.6+ KB
```

As we can see, new columns were created, of which we will make a general explanation below.

## ratings

This column was created from the rating\_numerator and denominator columns. First, it was checked that there were no zeros in the denominator. After this step, the numerator was divided by the denominator.

dog\_stages

This column is the union of the columns doggo, pupper, puppo, and floof(er), the value "None" was also changed to None to have a more congruent structure.

hour

we get the tweet time with the timestamp column

full\_week

We get the days from the timestamp column and create a function to get weekday and weekend.

predicted\_dog and predicted\_precision

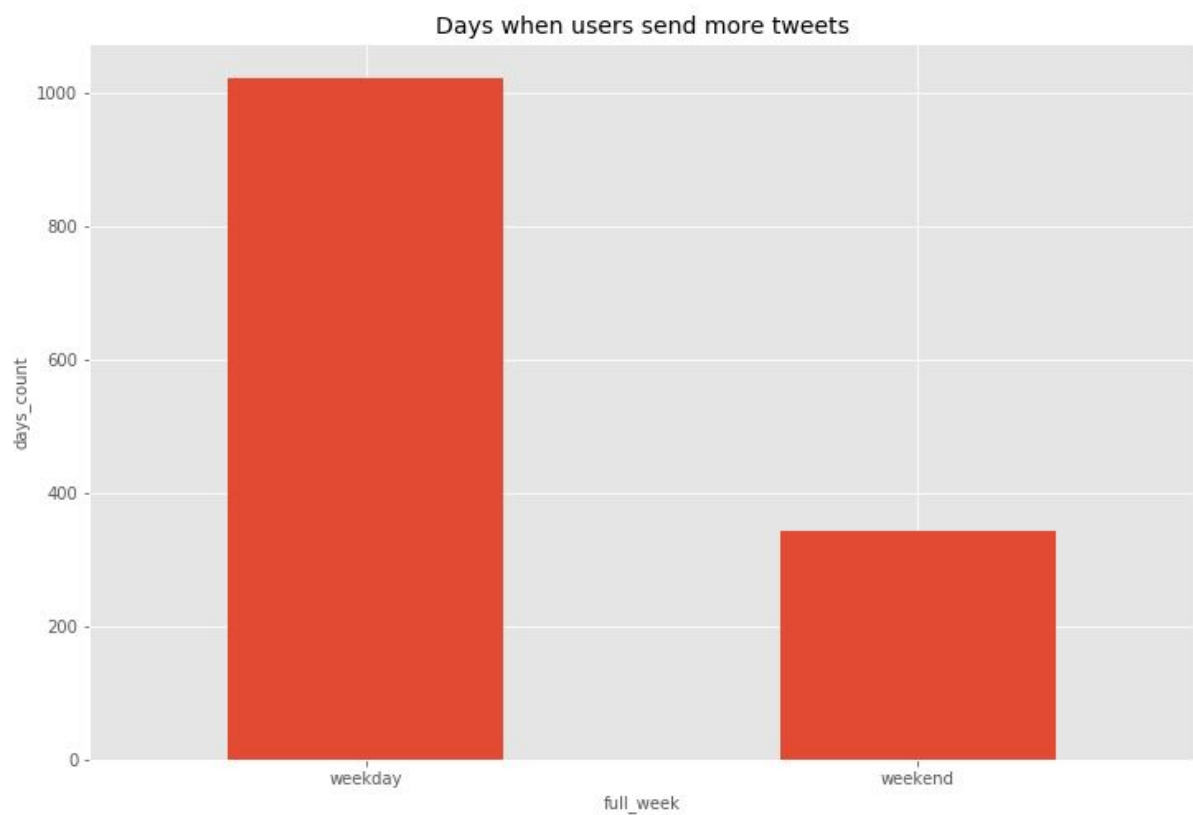
When we merge between the 3 dataframes, we create a new column for the first value in the prediction column that is True

is\_dog

When we merge between the 3 dataframes, we create a new column that checks if any result in the prediction column was True (It is a dog) or False (It is not a dog)

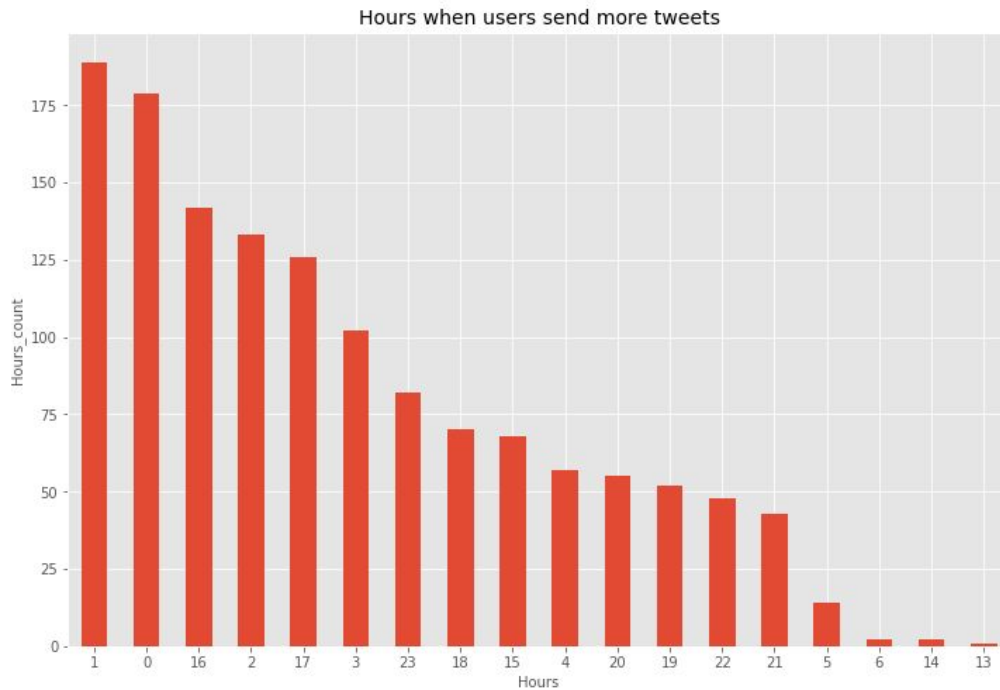
## Visualizing data

full week



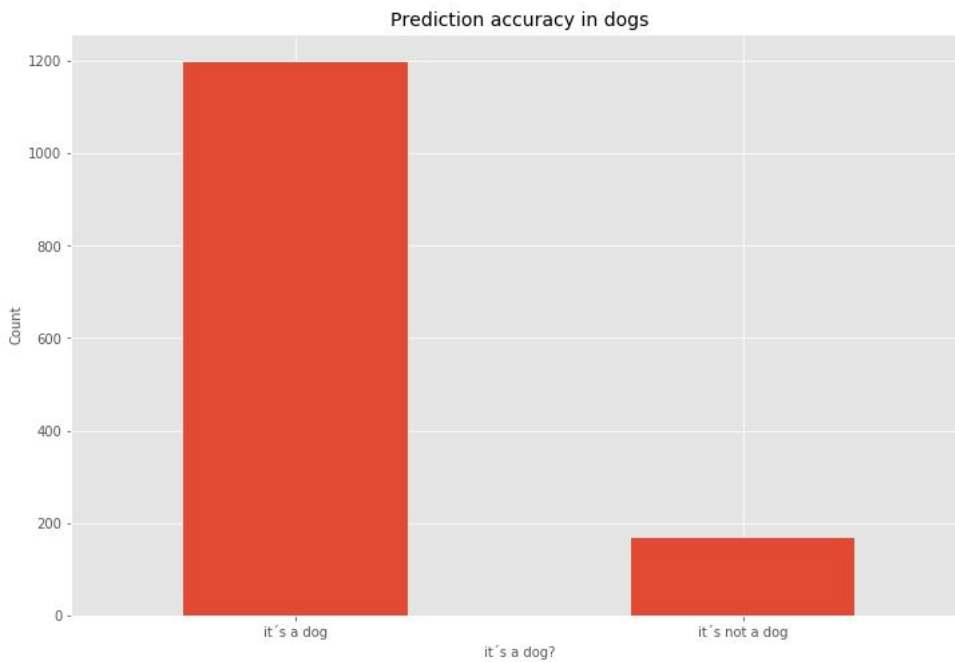
Here we can see that the sample we have, people always take time to tweet on business days.

hours



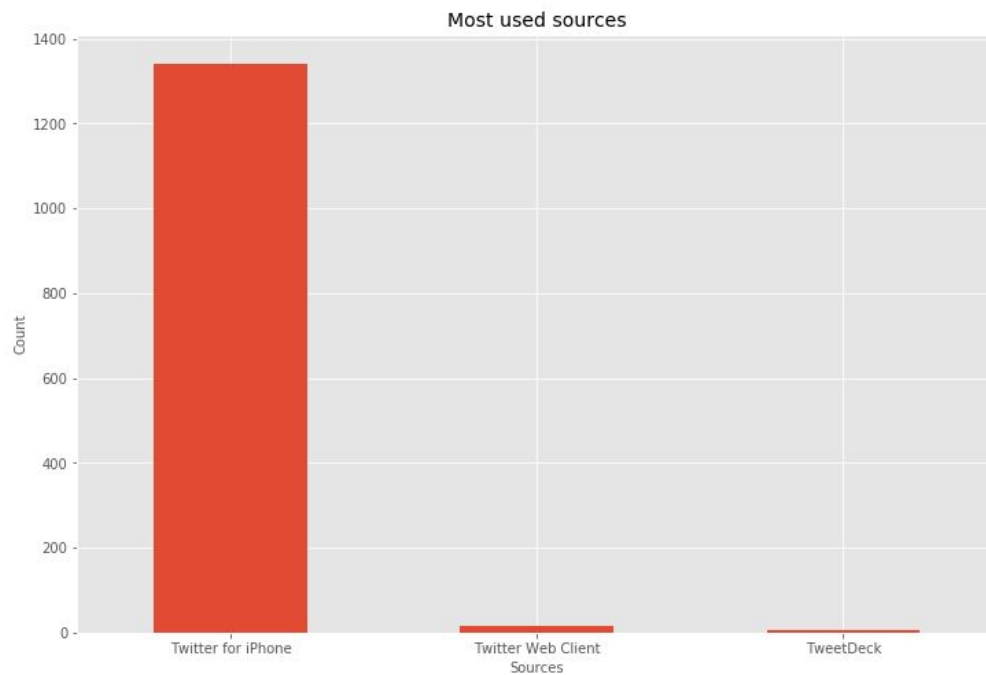
In this graph we can see that the hours of most tweets are at dawn, insomnia?

id\_dog



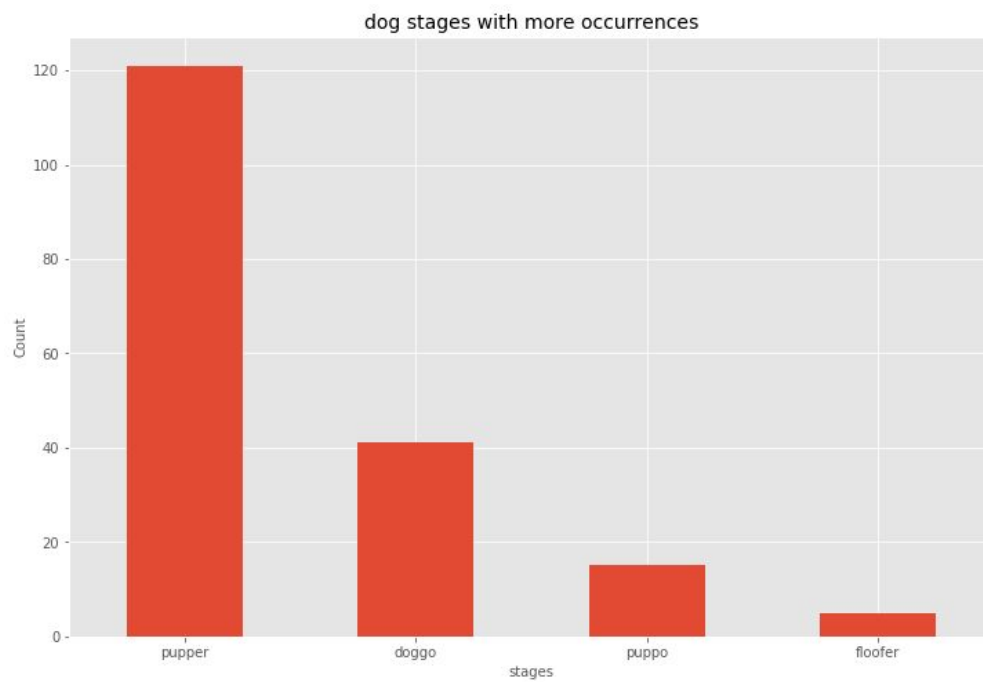
Here it is observed that the predictions are quite accurate with almost 12% failures to recognize dogs.

source



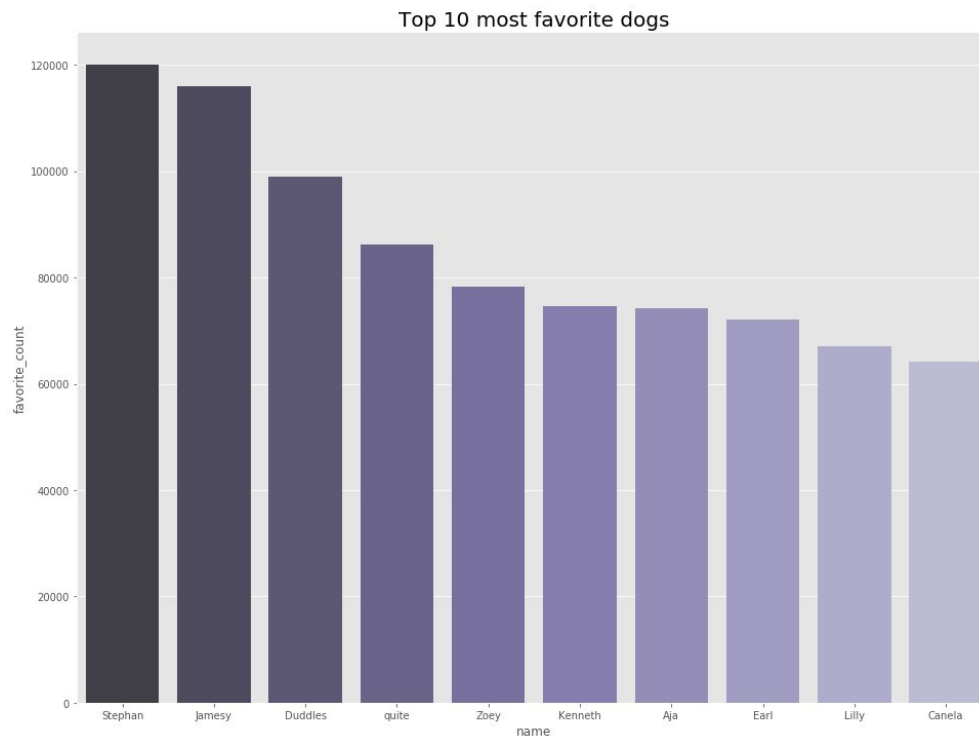
Overwhelming difference between tweet sources, iphone above by far.

dog\_stages

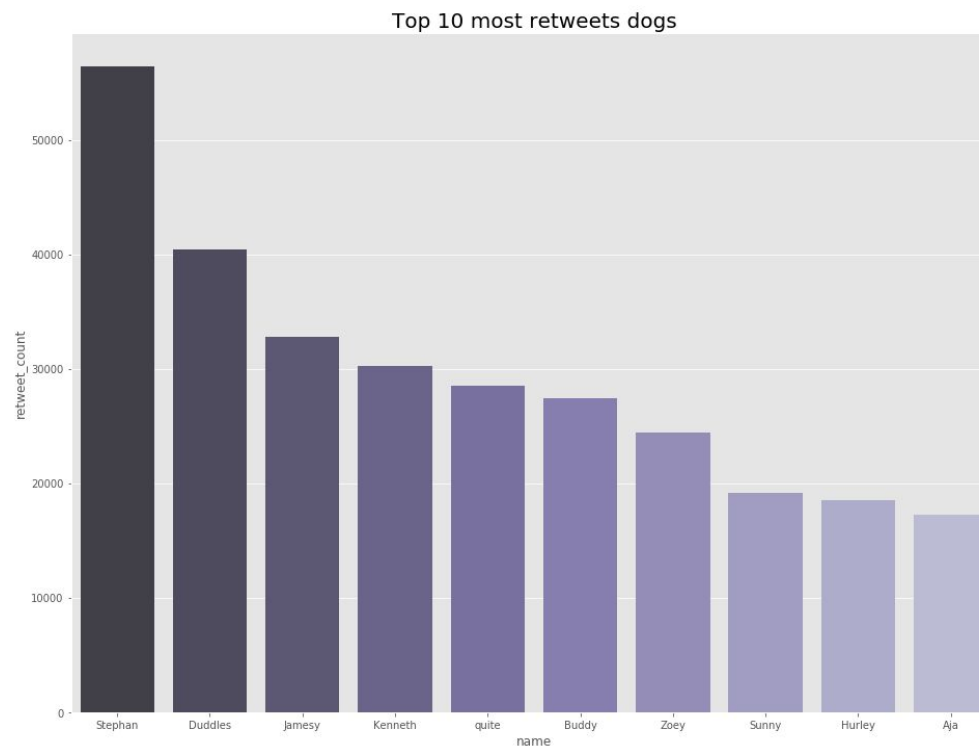


In this graph we observe that small dogs are mostly followed by larger ones.

## Top 10 favorite dogs

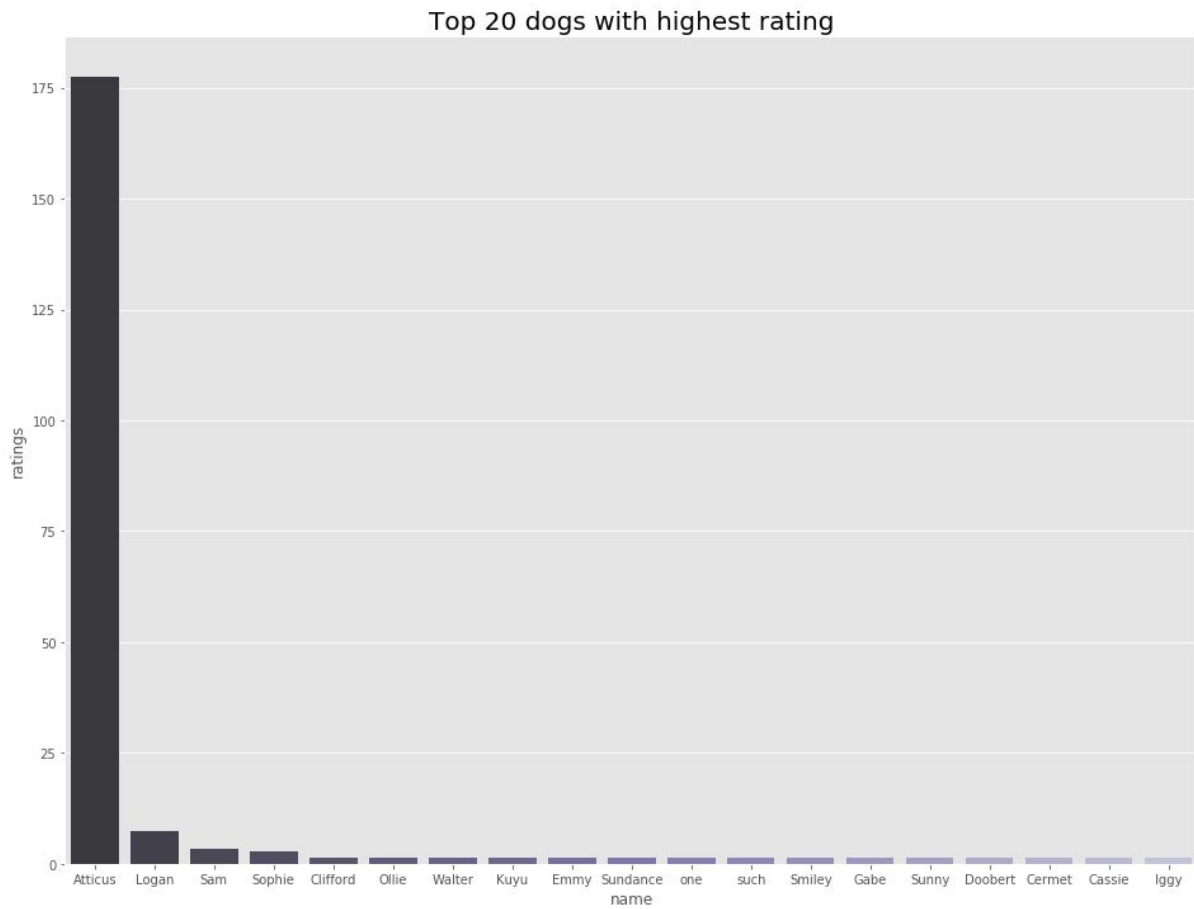


## Top 10 retweets dogs



As we observe in the tops, a relationship between favorites and retweets is almost always maintained.

## Ratings



In this graph there is an unusual value, but it is the fun of @weratedogs, right? But otherwise, we note that they remain almost at the same rating