



**WeRateDogs™** @dog\_rates · Apr 16  
Say hello to Alice. I'm told she enjoys car rides and smells good. 12/10 would give her everything she could ever want

**WeRateDogs™** @dog\_rates · Apr 16  
A photographer took pictures before and after he told his bunny he's a good boy. Here are the results. 13/10



## Analysis and Visualization



```

In [176]: df_clean.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 7968 entries, 0 to 7967
Data columns (total 23 columns):
tweet_id                7968 non-null object
in_reply_to_status_id   92 non-null float64
in_reply_to_user_id     92 non-null float64
timestamp               7968 non-null datetime64[ns, UTC]
source                  7968 non-null object
text                    7968 non-null object
expanded_urls           7968 non-null object
rating_numerator         7968 non-null int64
rating_denominator       7968 non-null int64
name                    7968 non-null object
dog_type                 7968 non-null object
favorite_count           7968 non-null int64
jpg_url                  7968 non-null object
img_num                  7968 non-null int64
p1                       7968 non-null object
p1_conf                  7968 non-null float64
p1_dog                   7968 non-null bool
p2                       7968 non-null object
p2_conf                  7968 non-null float64
p2_dog                   7968 non-null bool
p3                       7968 non-null object
p3_conf                  7968 non-null float64
p3_dog                   7968 non-null bool
dtypes: bool(3), datetime64[ns, UTC](1), float64(5), int64(4), object(10)
memory usage: 1.3+ MB

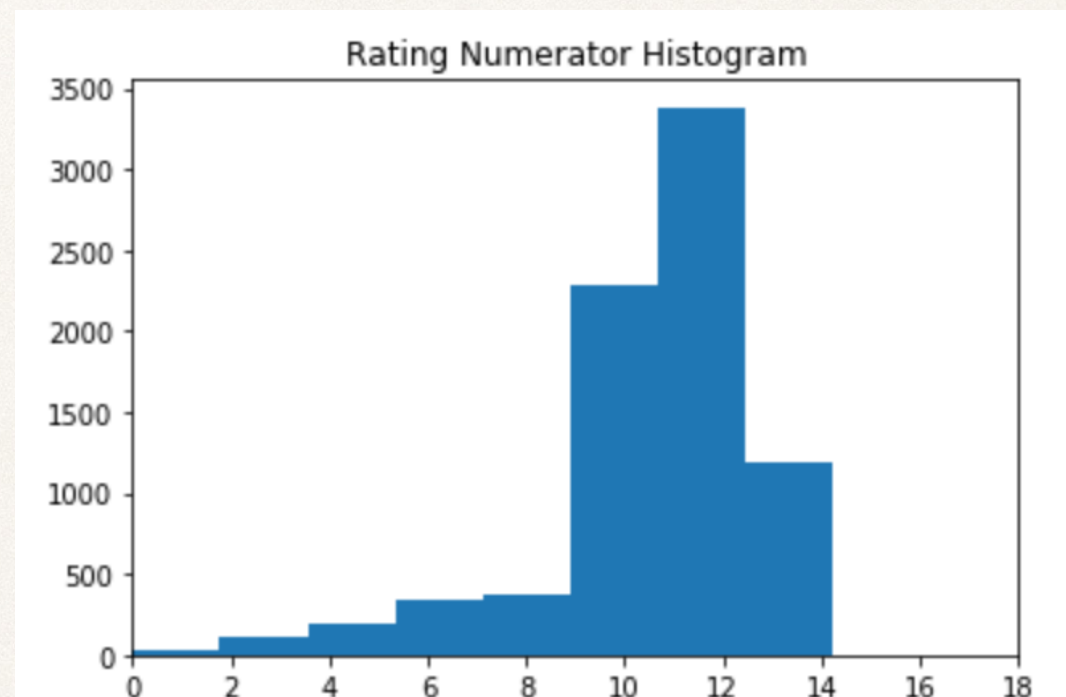
```

df.info() : show our new table information after cleaning we can see that the three table now become one organized table and dog types melt together in one column named dog\_type , Also timestamp type have been fixed to be datetime type and also tweet\_id type fixed .



	in_reply_to_status_id	in_reply_to_user_id	rating_numerator	rating_denominator	favorite_count	img_num	p1_conf	p2_conf	p3_conf
count	9.200000e+01	9.200000e+01	7968.000000	7968.000000	7968.000000	7968.000000	7968.000000	7.968000e+03	7.968000e+03
mean	6.978112e+17	4.196984e+09	12.280622	10.538153	8901.605422	1.203313	0.593669	1.345043e-01	6.028316e-02
std	4.286923e+16	0.000000e+00	41.509137	7.320081	12215.271422	0.560916	0.271904	1.006757e-01	5.089485e-02
min	6.671522e+17	4.196984e+09	0.000000	10.000000	81.000000	1.000000	0.044333	1.011300e-08	1.740170e-10
25%	6.717299e+17	4.196984e+09	10.000000	10.000000	1984.000000	1.000000	0.362775	5.401683e-02	1.616933e-02
50%	6.757073e+17	4.196984e+09	11.000000	10.000000	4136.000000	1.000000	0.587440	1.175370e-01	4.952715e-02
75%	7.032559e+17	4.196984e+09	12.000000	10.000000	11315.500000	1.000000	0.844247	1.952647e-01	9.162278e-02
max	8.558181e+17	4.196984e+09	1776.000000	170.000000	132810.000000	4.000000	1.000000	4.880140e-01	2.734190e-01

The basic statistics of the dataset after cleaning are described using `df_clean.describe()` method.



In this graph, we can see the frequency of the rating values, range of values as shown is between 0 -14 and the majority rating is 12.

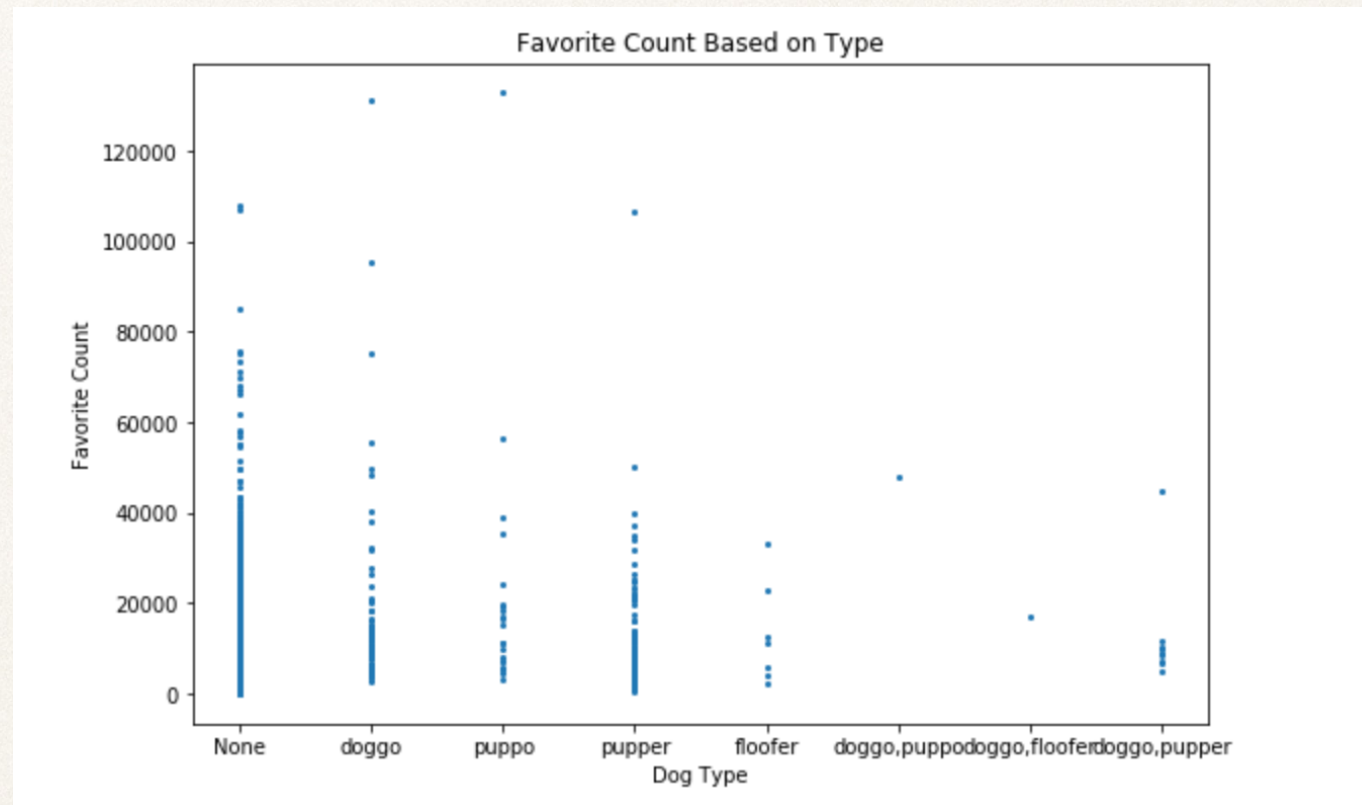


```
print ( 'Rating Numerator Mean =' )  
df_clean.rating_numerator.mean()
```

Rating Numerator Mean =

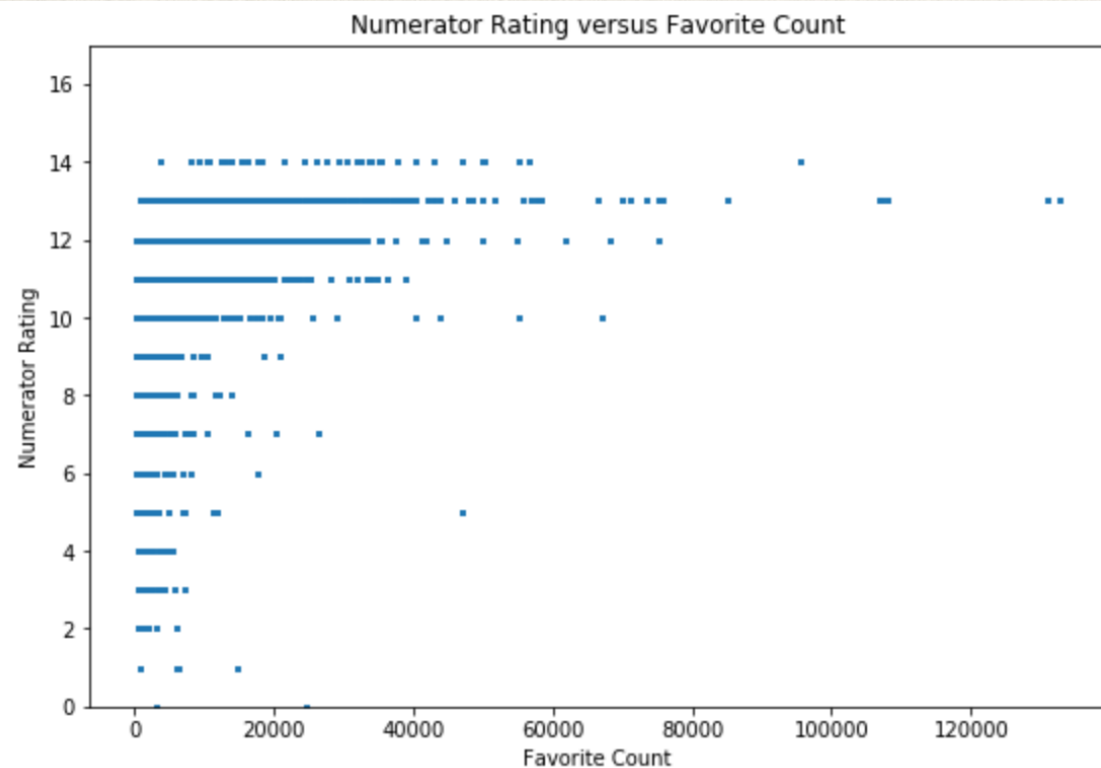
12.237101303911734

The mean value of the rating numerator is 12 .

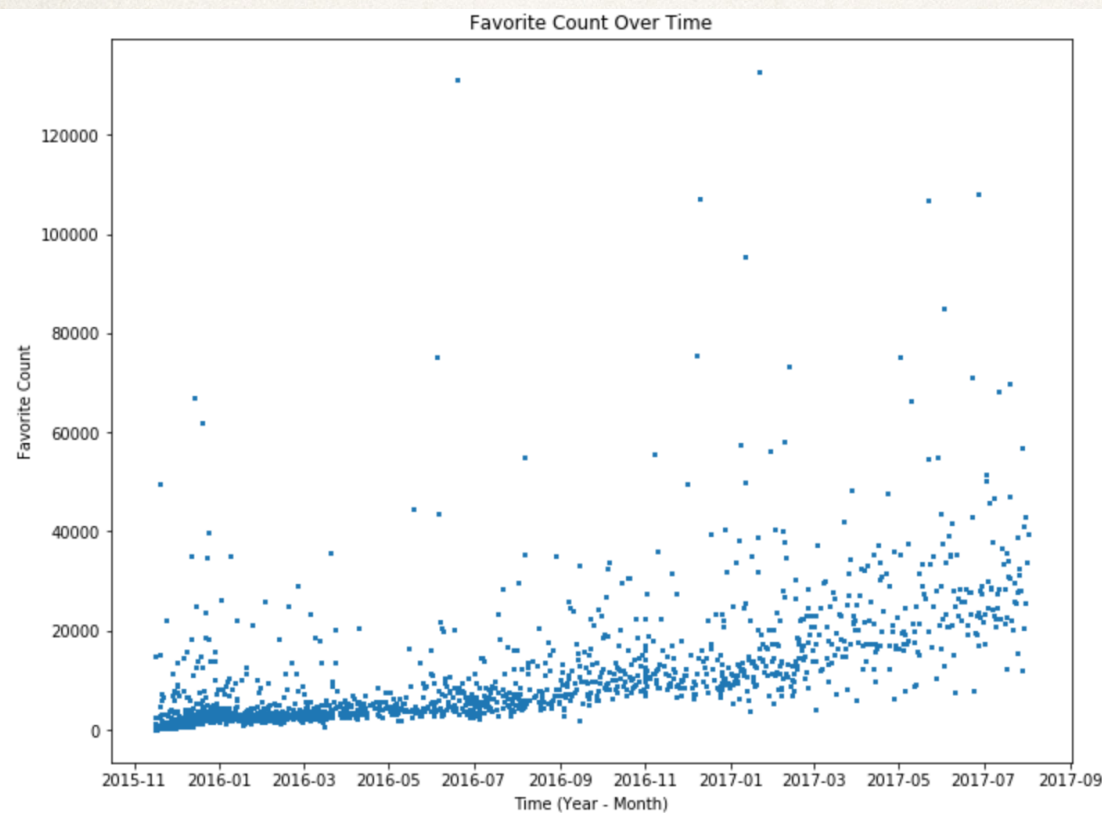


This distribution show us the favorite count based on dog type we can see that the pupper dog is the most favorite and the floofer is the less favorite, we ignore the none value .





This distribution show us the favorite count versus numerator rating and we can see that there is a direct relationship between them.



This distribution shows us the favorite count over time and we can see that the number favorite count decreasing over time which could mean people have become less interesting to the account or the account has become less tweet.