Documentation of Analysis and Insights into final data

For Analysing the tweet data I conveted the given dates into Day of the week so that I can see how many tweets are posted each day of the week

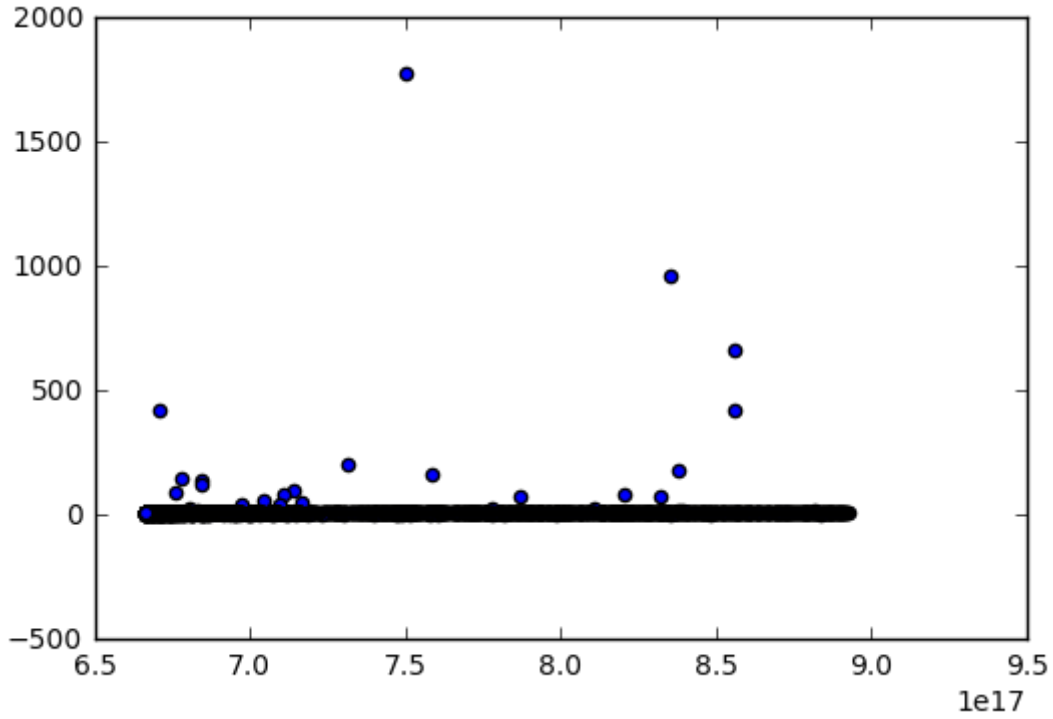
In [37]:
```
import pandas as pd
Final_tweet_data= pd.read_csv('twitter_archive_master.csv', index_col=0)
```

In [38]:
```
Final_tweet_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2356 entries, 0 to 2355
Data columns (total 27 columns):
tweet_id              2356 non-null int64
source                2356 non-null object
text                  2356 non-null object
expanded_urls         2297 non-null object
rating_numerator      2356 non-null int64
rating_denominator    2356 non-null int64
name                  2356 non-null object
doggo                 2356 non-null object
floofer               2356 non-null object
pupper                2356 non-null object
puppo                 2356 non-null object
Date                  2356 non-null object
Time                  2356 non-null object
favorite_count        2356 non-null float64
retweet_count         2356 non-null float64
jpg_url               2356 non-null object
img_num               2356 non-null float64
p1                    2356 non-null object
p1_conf               2356 non-null float64
p1_dog                2356 non-null object
p2                    2356 non-null object
p2_conf               2356 non-null float64
p2_dog                2356 non-null object
p3                    2356 non-null object
p3_conf               2356 non-null float64
p3_dog                2356 non-null object
Day_of_week           2356 non-null object
dtypes: float64(6), int64(3), object(18)
memory usage: 515.4+ KB
```

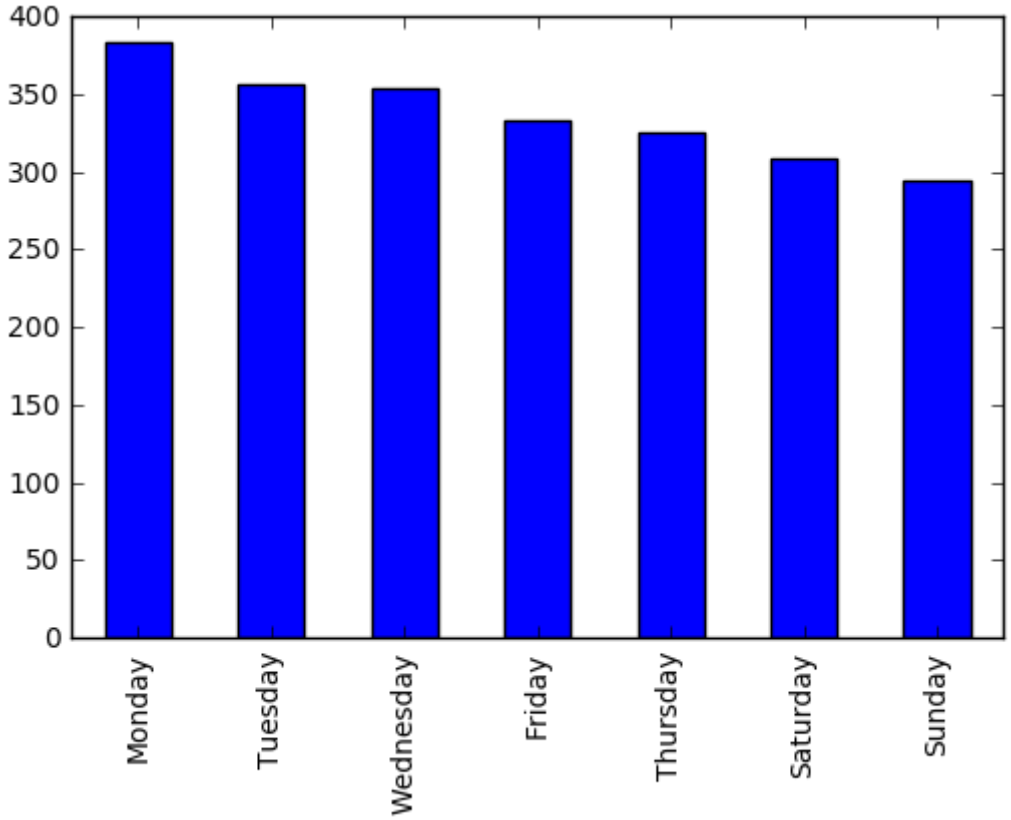
In [39]:
```
import matplotlib.pyplot as plt
%matplotlib inline
plt.scatter(Final_tweet_data.tweet_id,
Final_tweet_data.rating_numerator)
```

Out[39]: <matplotlib.collections.PathCollection at 0x1187b13d0>



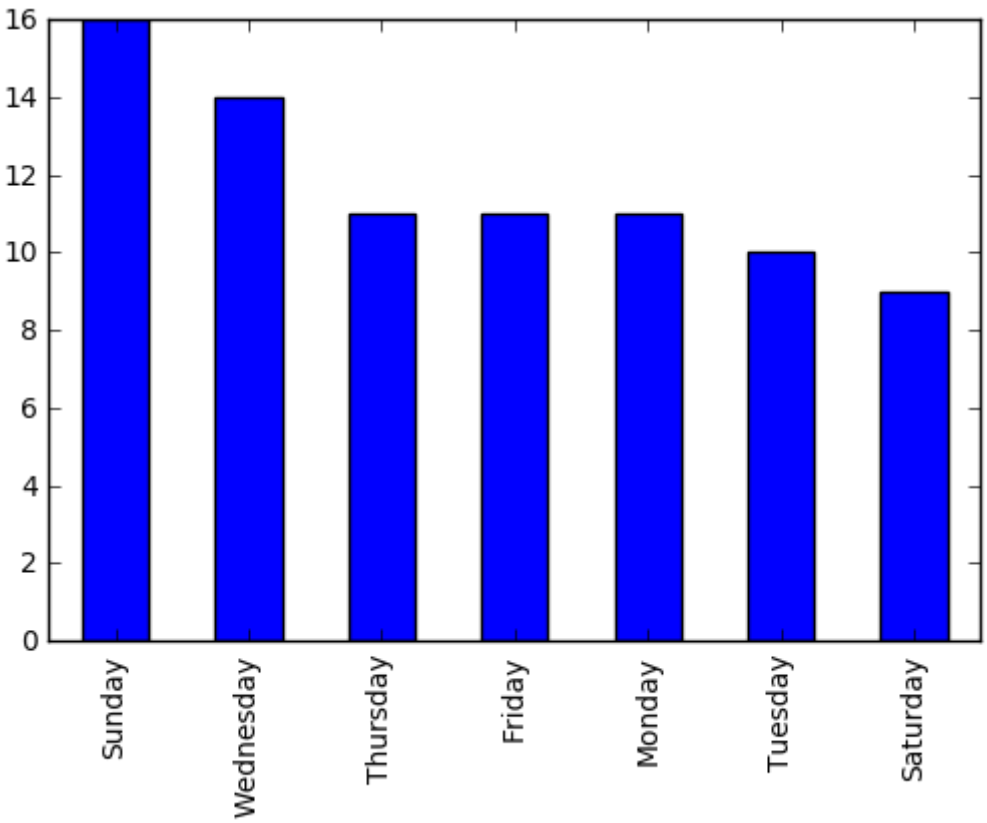
In [25]:
```
Final_tweet_data.Date[0]
# Final_tweet_data.Date.map(lambda x:x.strftime('%A'))
```

Out[25]: <matplotlib.axes._subplots.AxesSubplot at 0x116774250>



In [26]:
```
greater_than_average_ratings= Final_tweet_data[Final_tweet_data.rating_numerator >13]
greater_than_average_ratings.Day_of_week.value_counts().plot(kind='bar')
```

Out[26]: <matplotlib.axes._subplots.AxesSubplot at 0x1160d7d90>



Below are the insights that I find during the Analysis

1) Most of the tweets are posted on Mondays.

2) In general we can see people used to tweet more on Mondays as compared to other days and least on Sundays. The mean value of rating_numerator on Mondays is 15.208333 and total tweet count on Monday is 384.

3) However, when I check the tweets having rating_numerator values greater than average then the graph is different as people used to post tweets on Sunday. Most of the tweet having rating_numerator greater than 13 are on Sunday.

In [29]:
```
tweet_days= Final_tweet_data.groupby('Day_of_week')
tweet_days.size()
```

Out[29]:
```
Day_of_week
Friday       333
Monday       384
Saturday     309
Sunday       294
Thursday     326
Tuesday      356
Wednesday    354
dtype: int64
```

In [32]:
```
tweet_days.mean()
```

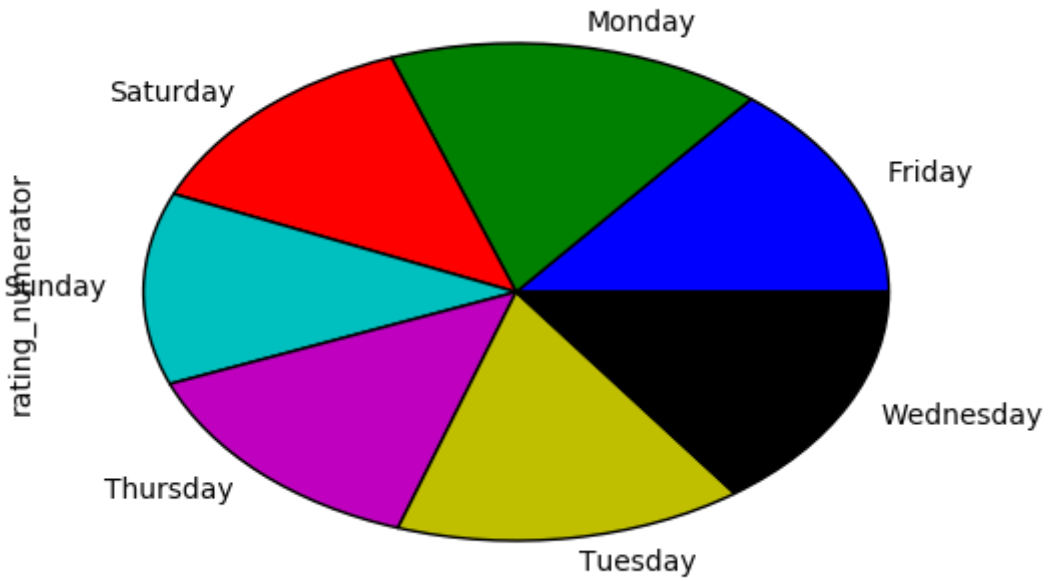
Out[32]:

	tweet_id	rating_numerator	rating_denominator	favorite_count	retwe
Day_of_week					
Friday	7.461903e+17	14.960961	11.012012	7803.246246	3075.
Monday	7.406242e+17	15.208333	10.075521	7932.697917	2881.
Saturday	7.450709e+17	14.822006	10.000000	7975.317152	3182.
Sunday	7.398319e+17	13.044218	11.023810	8057.071429	2907.
Thursday	7.383225e+17	11.699387	10.475460	7722.027607	2941.
Tuesday	7.445165e+17	11.469101	10.617978	7894.064607	3173.
Wednesday	7.446619e+17	10.711864	10.087571	9070.692090	3630.

The mean of the rating_numerator is greater on Mondays 15.2 as compared to other days and second day of psoting more tweets is Friday and Saturday having mean 14.96 and 14.82.

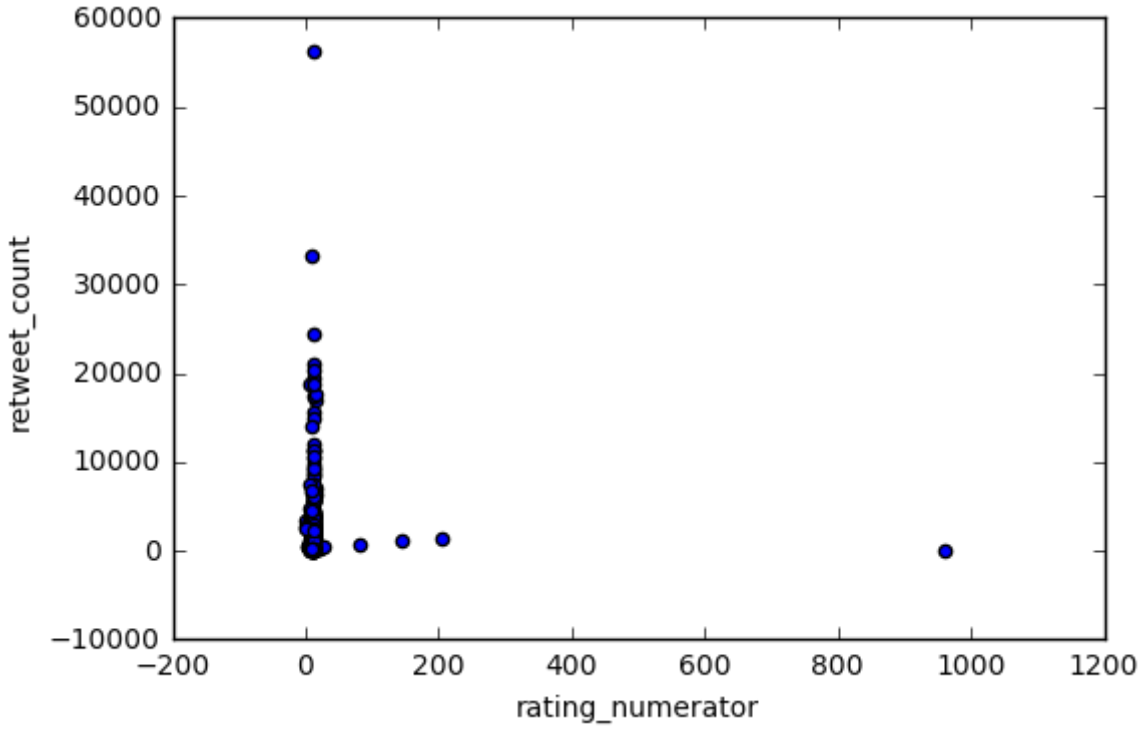
```
In [33]: tweet_days.rating_numerator.count().plot(kind='pie')
```

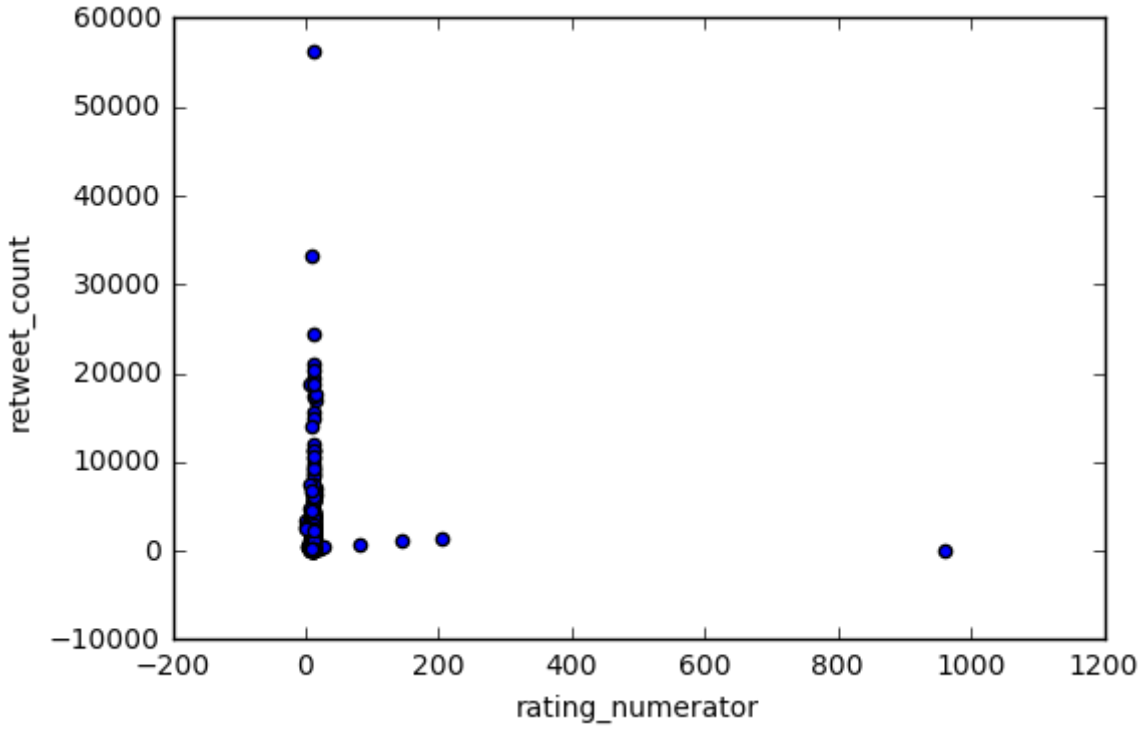
```
Out[33]: <matplotlib.axes._subplots.AxesSubplot at 0x11623f410>
```

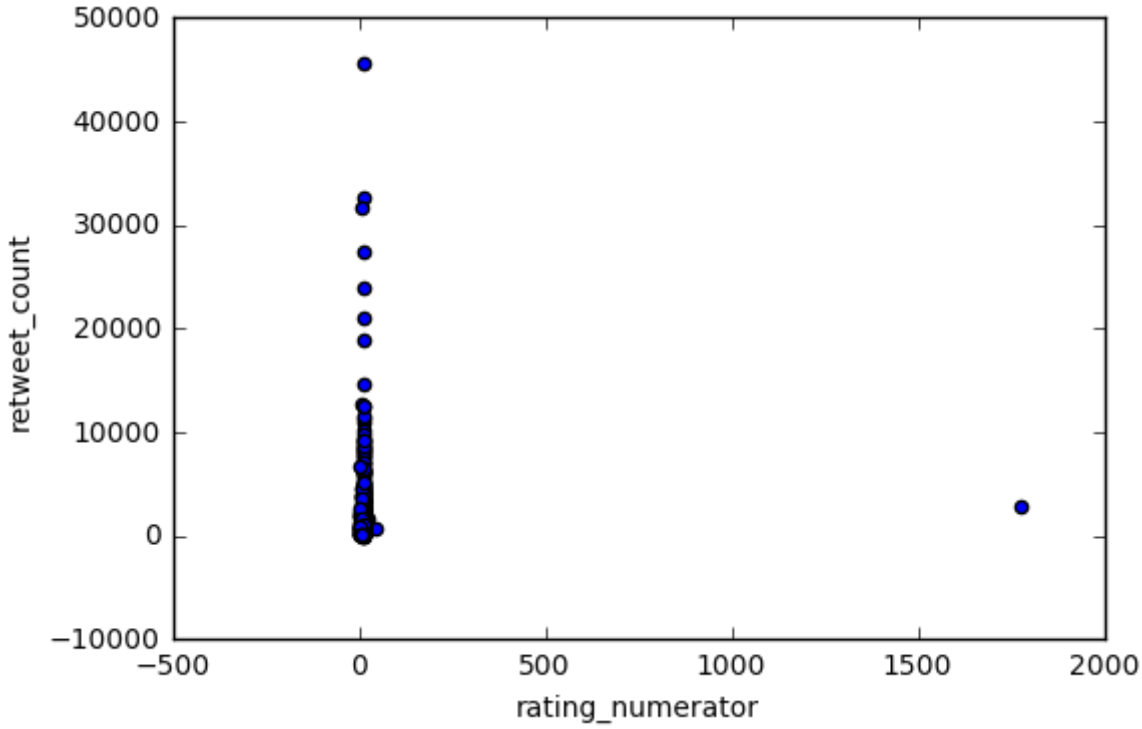


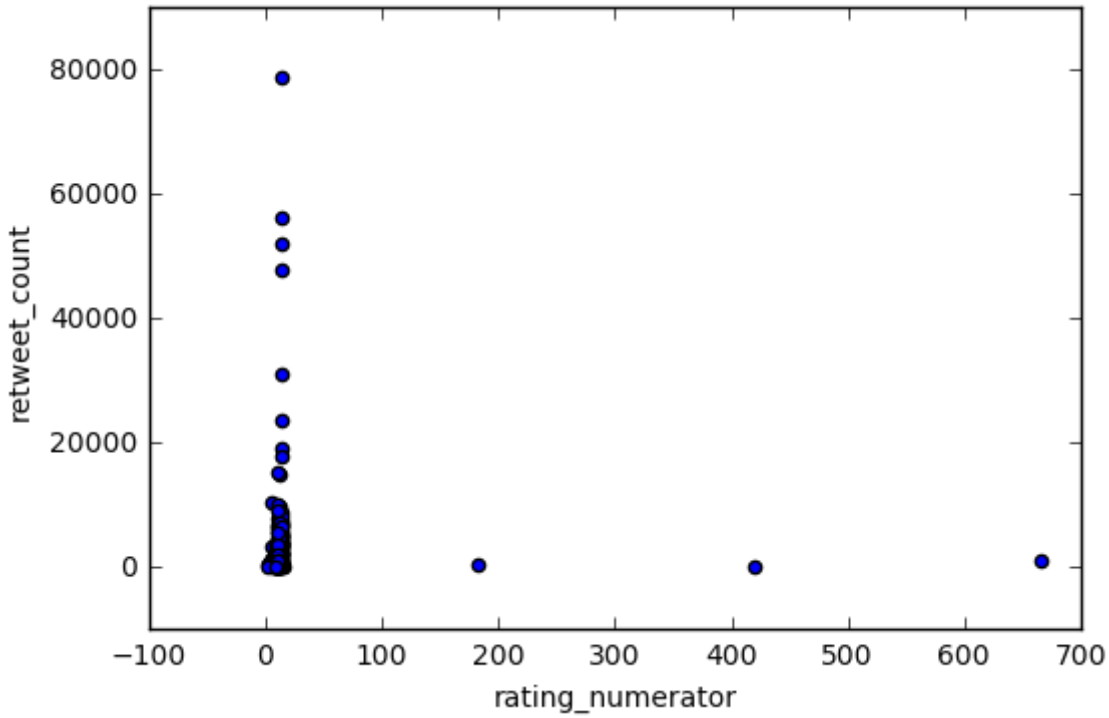
retweet_count and favorite_count Analysis

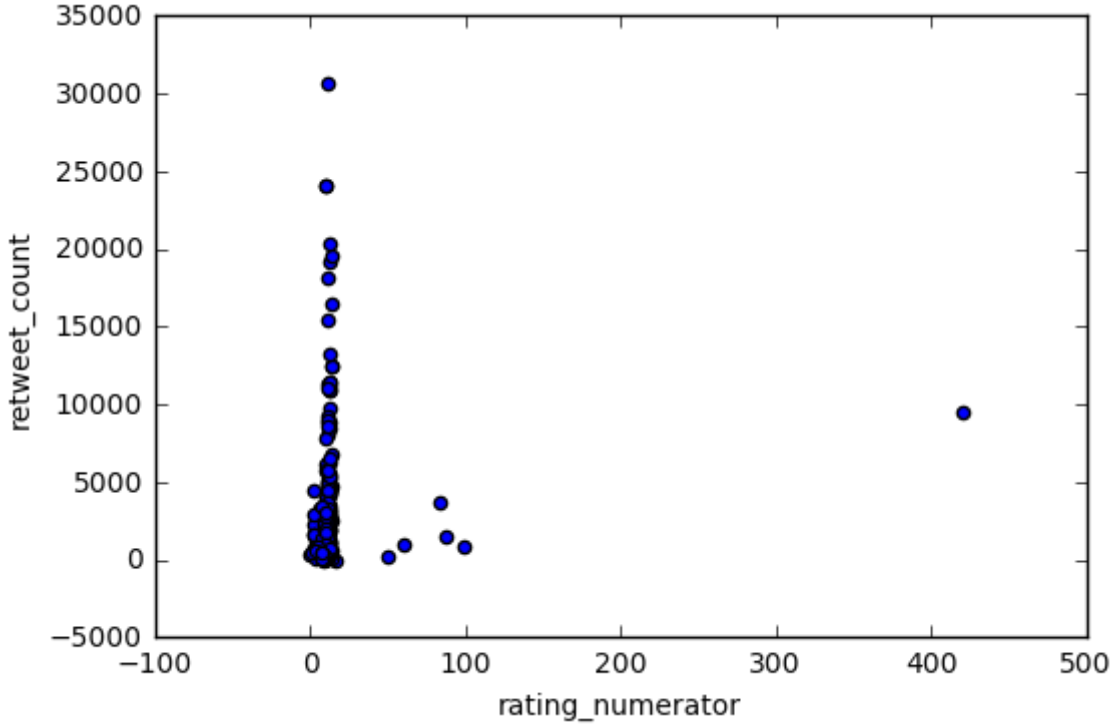
In [35]:
```
tweet_days.plot.scatter(x='rating_numerator', y='retweet_count');
```

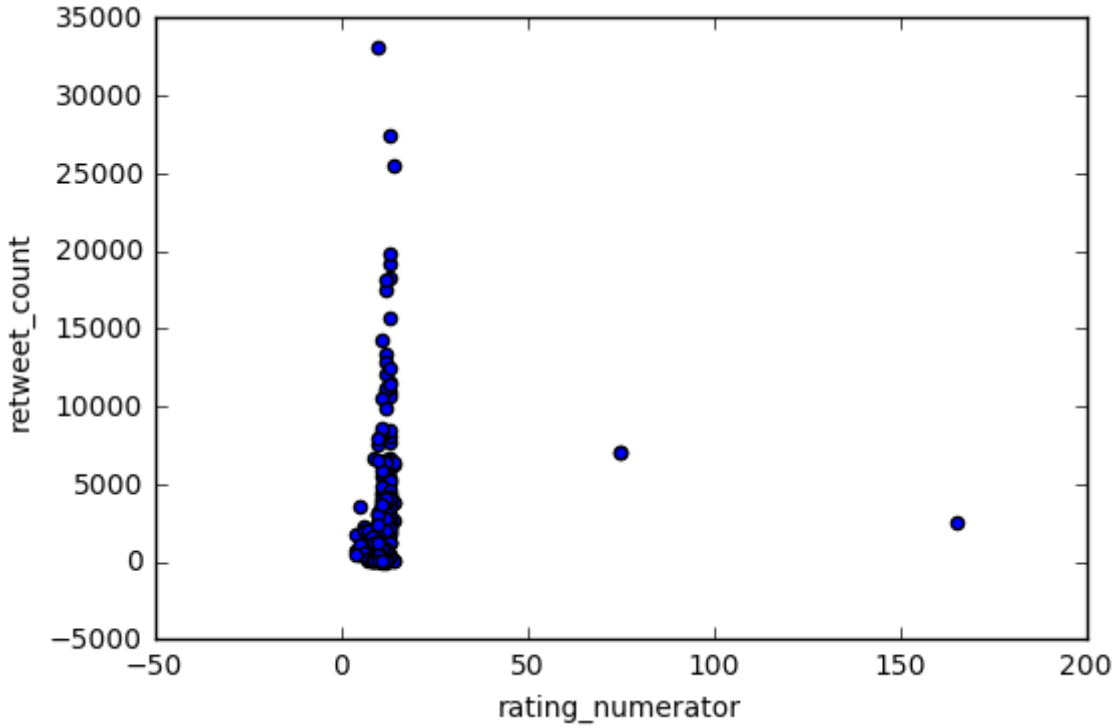

60000
50000
40000
30000
20000
10000
0
−10000
retweet_count
−200
0
200
400
600
800
1000
1200
rating_numerator

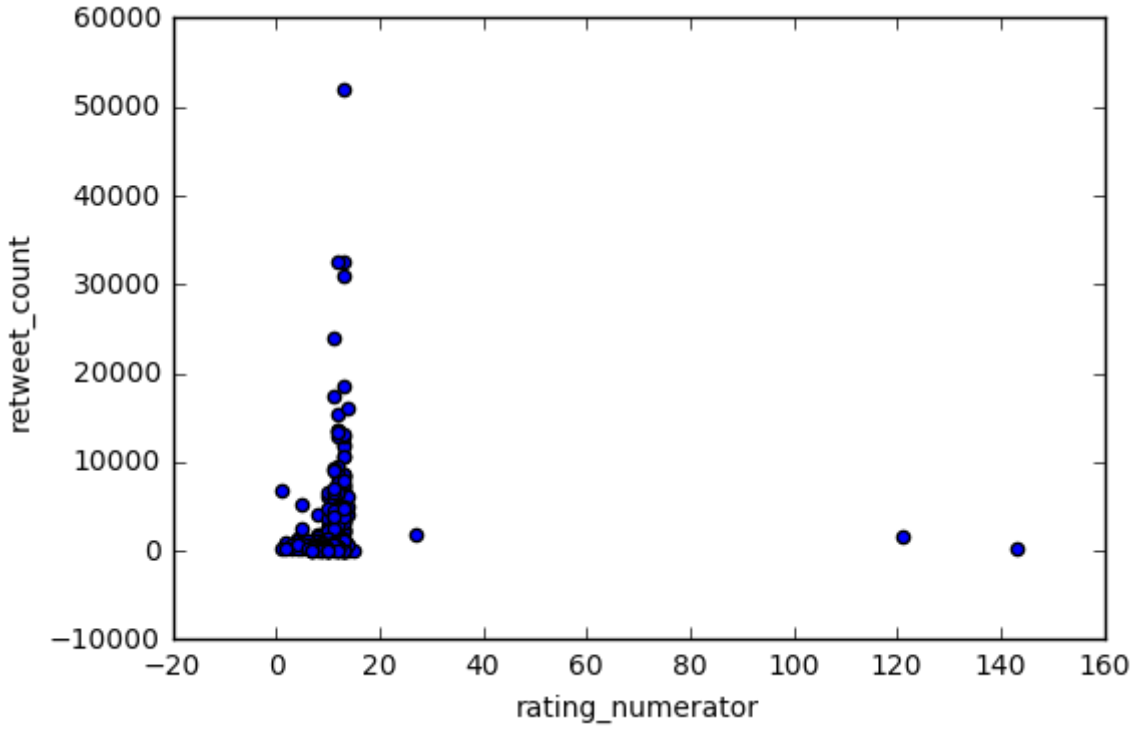

60000
50000
40000
30000
20000
10000
0
−10000
retweet_count
−200
0
200
400
600
800
1000
1200
rating_numerator

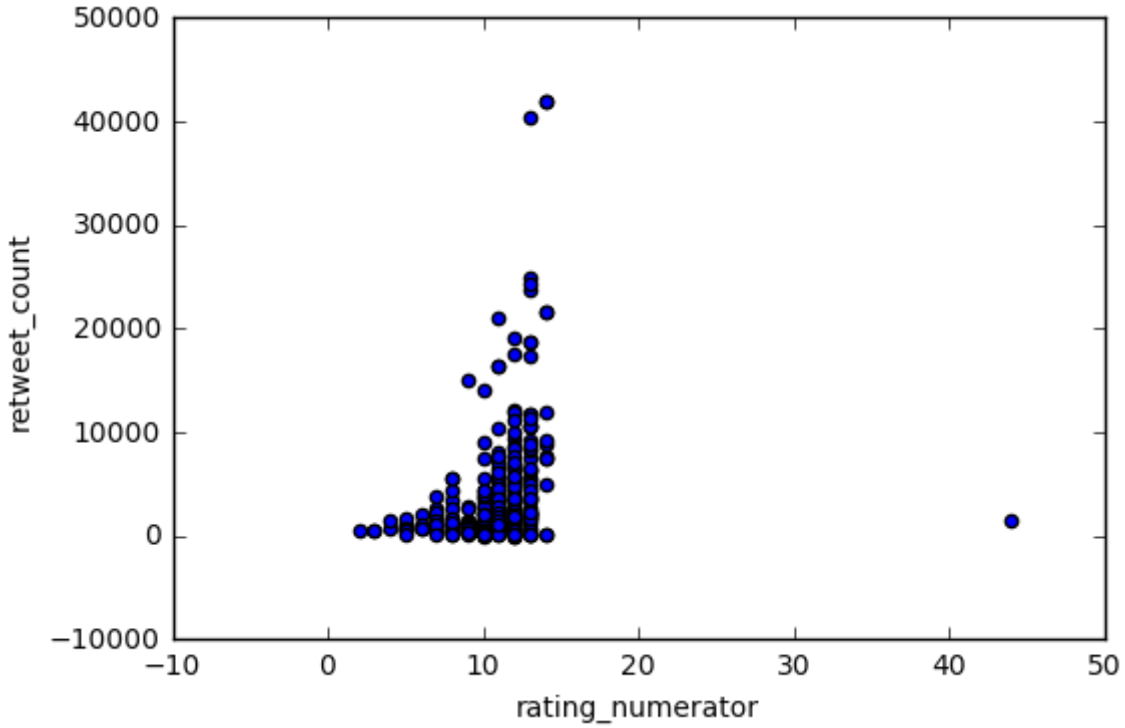

50000
40000
30000
20000
10000
0
−10000
retweet_count
−500
0
500
1000
1500
2000
rating_numerator


80000
60000
40000
20000
0
retweet_count
−100
0
100
200
300
400
500
600
700
rating_numerator

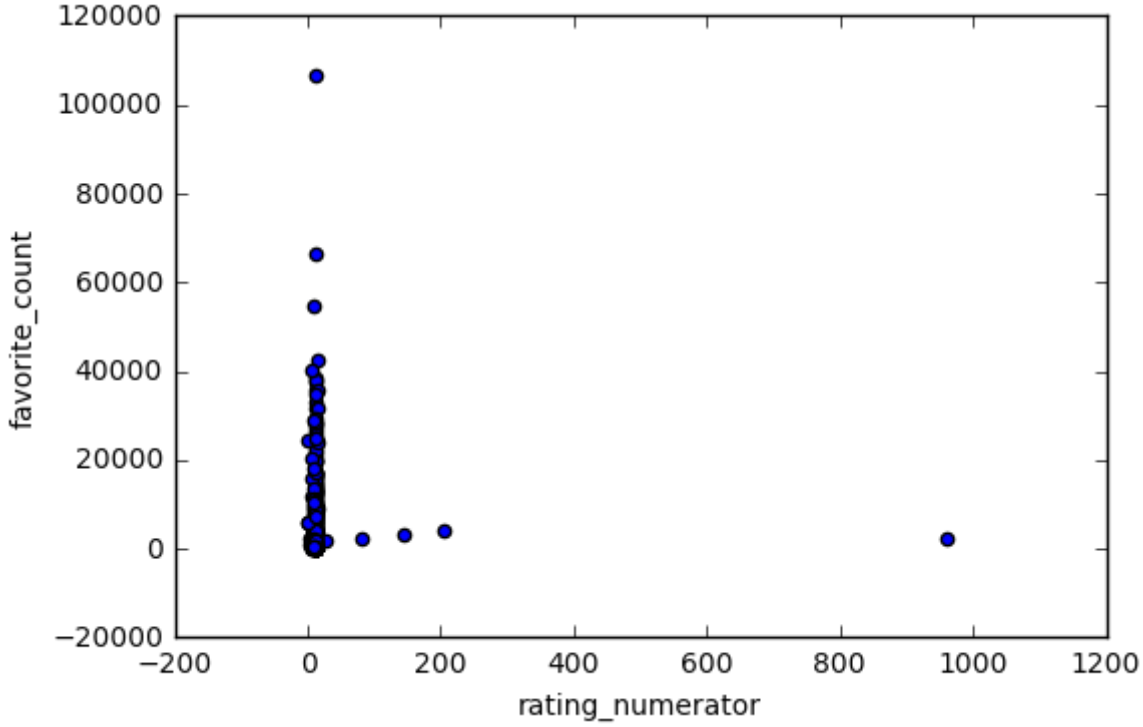

35000
30000
25000
20000
15000
10000
5000
0
−5000
retweet_count
−100
0
100
200
300
400
500
rating_numerator

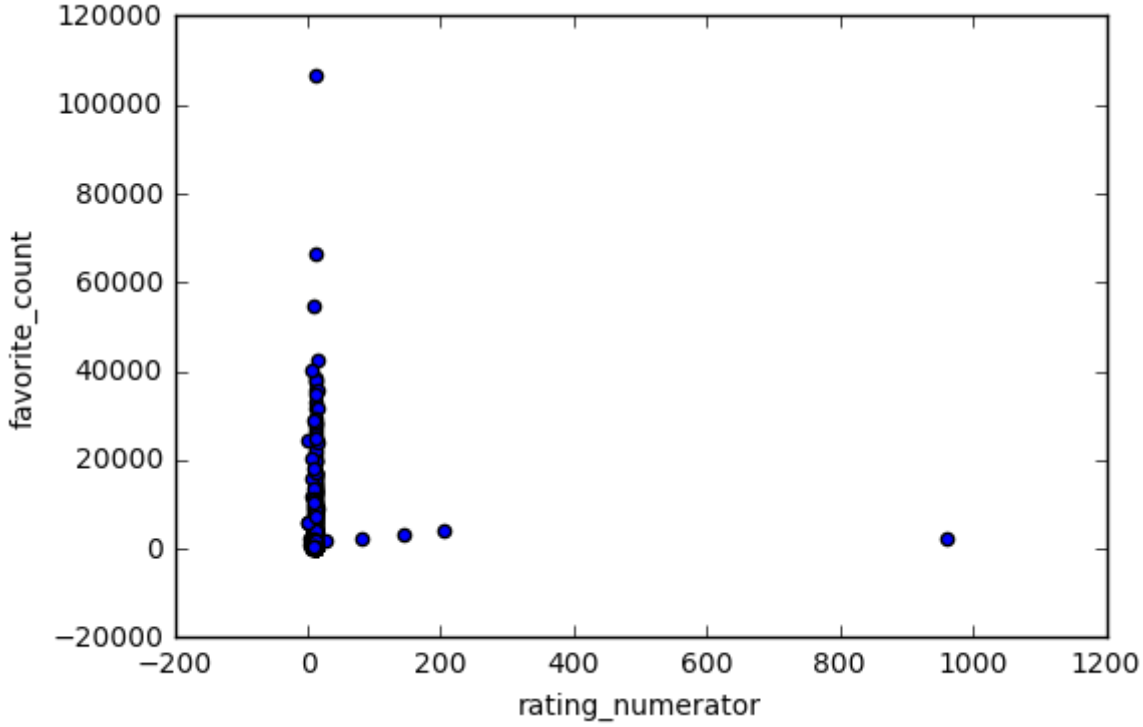

35000
30000
25000
20000
15000
10000
5000
0
−5000
retweet_count
−50
0
50
100
150
200
rating_numerator

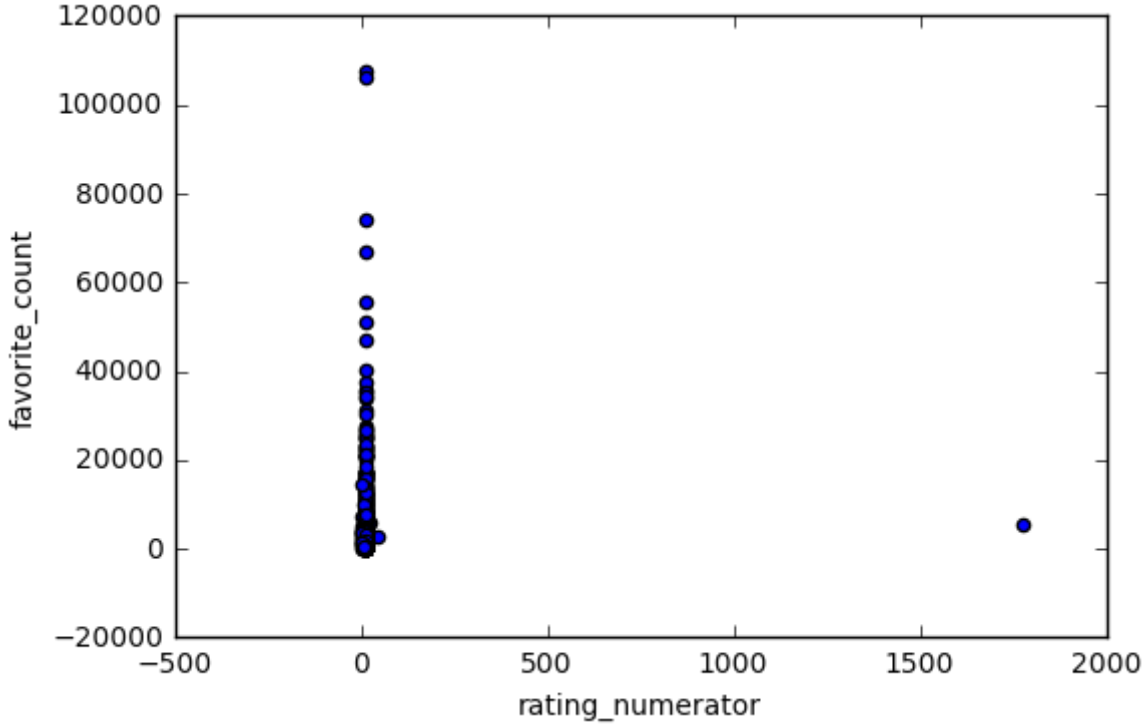

60000
50000
40000
30000
20000
10000
0
−10000
retweet_count
−20
0
20
40
60
80
100
120
140
160
rating_numerator

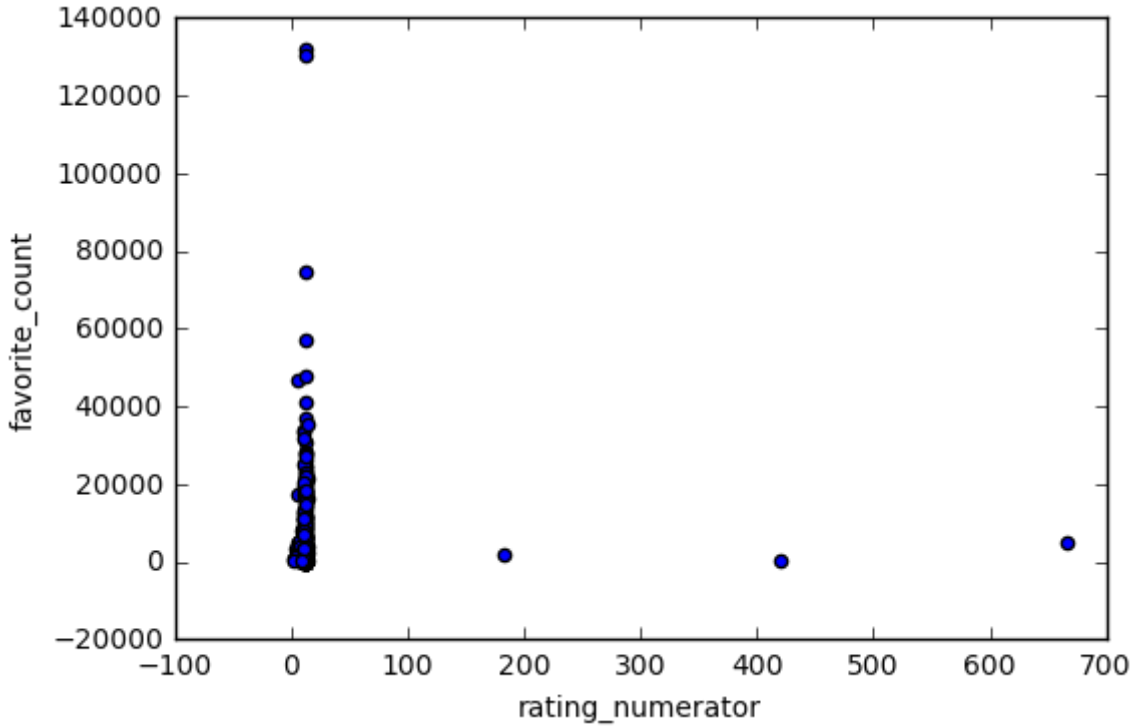

50000
40000
30000
20000
10000
0
−10000
retweet_count
−10
0
10
20
30
40
50
rating_numerator

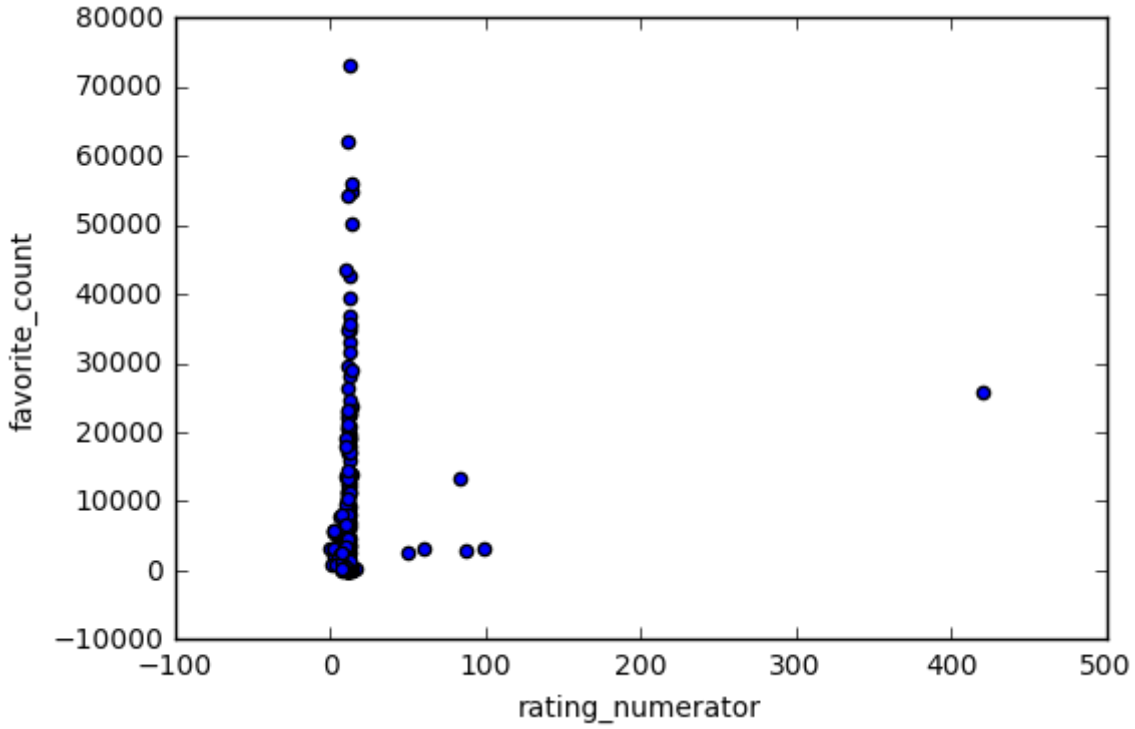
In [36]:
```
tweet_days.plot.scatter(x='rating_numerator', y='favorite_count');
```

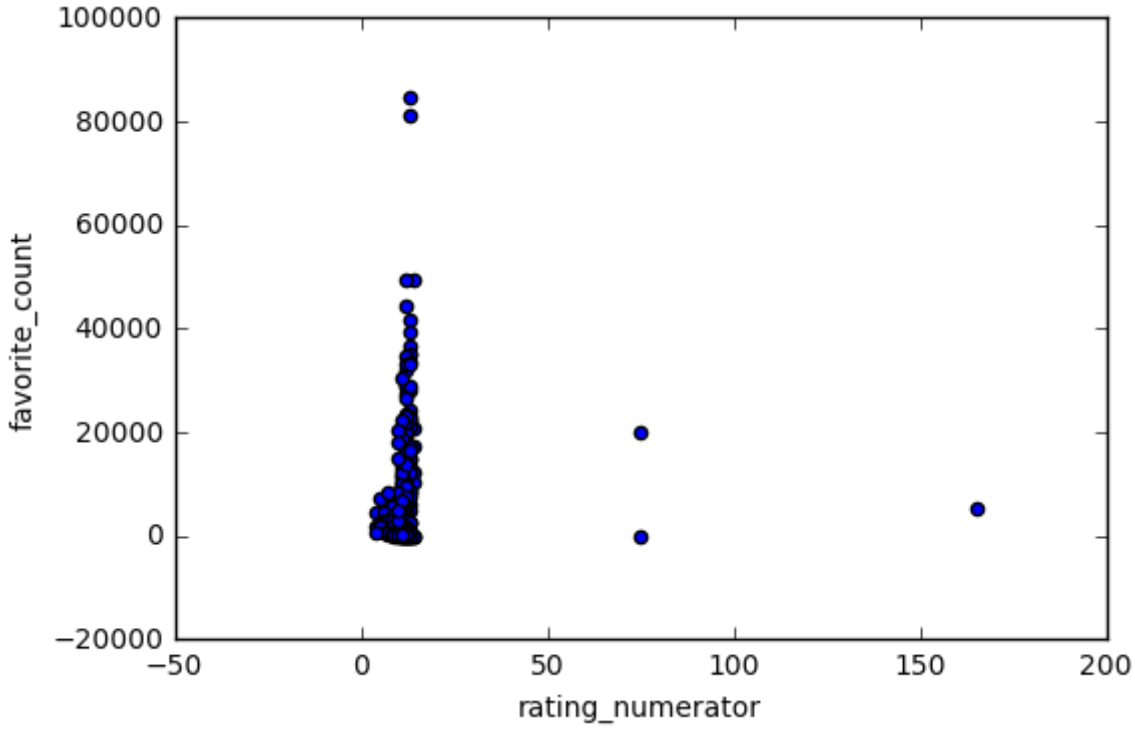

120000
100000
80000
60000
40000
20000
0
−20000
favorite_count
−200
0
200
400
600
800
1000
1200
rating_numerator

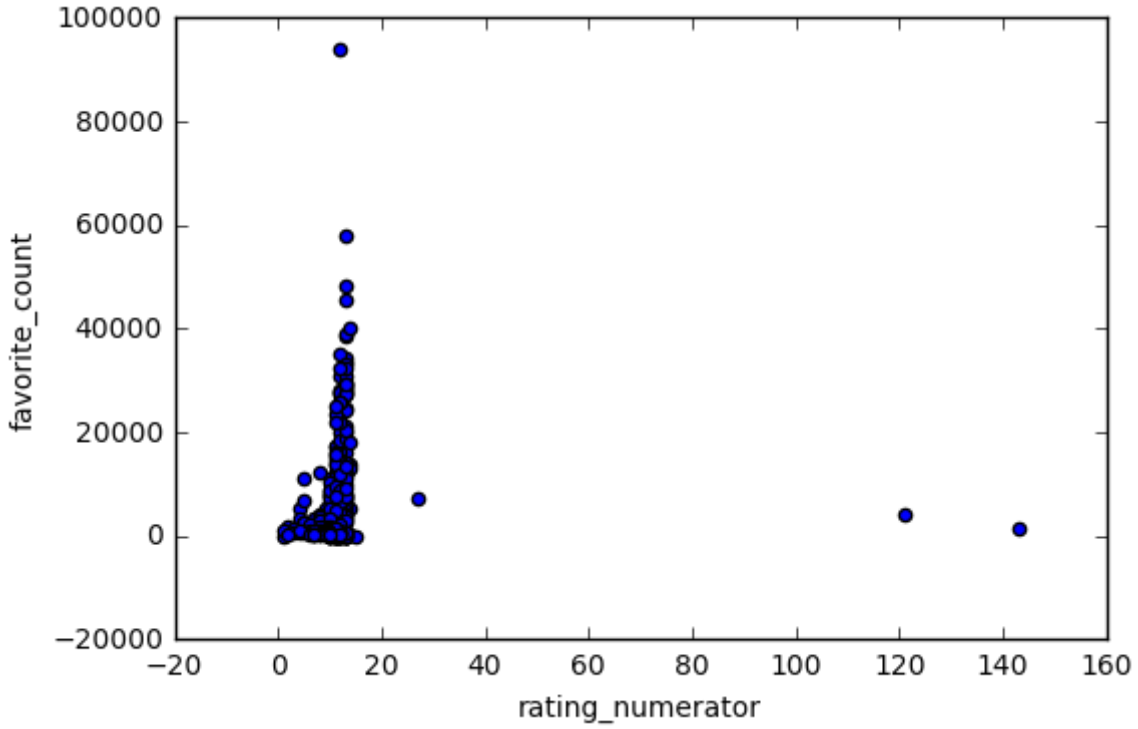

120000
100000
80000
60000
40000
20000
0
−20000
favorite_count
−200
0
200
400
600
800
1000
1200
rating_numerator

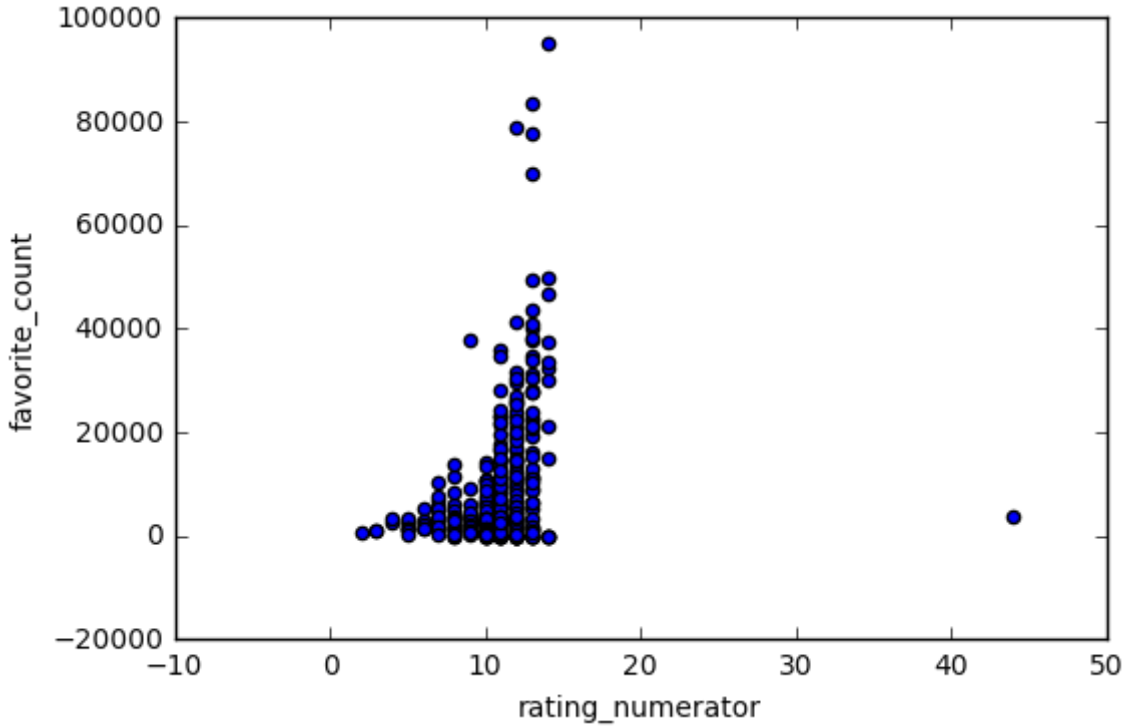

120000
100000
80000
60000
40000
20000
0
−20000
favorite_count
−500
0
500
1000
1500
2000
rating_numerator


140000
120000
100000
80000
60000
40000
20000
0
−20000
favorite_count
−100
0
100
200
300
400
500
600
700
rating_numerator


80000
70000
60000
50000
40000
30000
20000
10000
0
−10000
favorite_count
−100
0
100
200
300
400
500
rating_numerator


100000
80000
60000
40000
20000
0
−20000
favorite_count
−50
0
50
100
150
200
rating_numerator





1) Most of the retweets are for rating_numerator between 8 to 13.

2) favorite_count are also somewhat like retweets.

In [ ]: