Project

Jacob Aronoff

11/12/2017

Make the queries

Getting the data

I decided to get started with my project by only looking about posts relating to a movie, later in the project I want to get into comments and sentiment analysis.

In constructing the query, I ran into a couple of problems. At first the query I was trying to run was returning all NULL values, I then changed the query from doing it all at once, to doing the queries individually. Making this change also allowed me to make it so that each query would only get data up until the movie's release date.

```
if(exists("movieData", inherits = T)) {
    # Pass
  } else {
    movieData = getMovieData(movies)
  }
```

```
## Warning in strptime(x, "%Y/%m/%d"): unknown timezone 'zone/tz/2017c.1.0/
## zoneinfo/America/New_York'
```

```
## [1] "Getting data for Allied"
## [1] "Getting data for Ben-Hur"
## [1] "Getting data for The BFG"
## [1] "Getting data for Deepwater Horizon"
## [1] "Getting data for The Finest Hours"
## [1] "Getting data for Ghostbusters"
## [1] "Getting data for Gods of Egypt"
## [1] "Getting data for The Great Wall"
## [1] "Getting data for The Huntsman: Winter's War"
## [1] "Getting data for Live by Night"
## [1] "Getting data for Monster Trucks"
## [1] "Getting data for Captain America: Civil War"
## [1] "Getting data for Rogue One: A Star Wars Story"
```

```
## Warning: NAs introduced by coercion
```

```
## [1] "Getting data for Finding Dory"
## [1] "Getting data for Zootopia"
## [1] "Getting data for The Jungle Book"
## [1] "Getting data for The Secret Life of Pets"
## [1] "Getting data for Batman v Superman: Dawn of Justice"
## [1] "Getting data for Fantastic Beasts and Where to Find Them"
## [1] "Getting data for Deadpool"
## [1] "Getting data for Suicide Squad"
```

```
## Warning in getMovieData(movies): NAs introduced by coercion
```

```
movieQueries = list()
for(i in 1:nrow(movieData))
{
  movieQueries <- append(movieQueries, moviePostQuery(movieData[i,]))
}

if(exists("bigQueryData", inherits = T)) {
    # Pass
} else if(file.exists("bigQueryData.csv")) {
    bigQueryData <- read.csv("bigQueryData.csv", header = TRUE)
    class(bigQueryData$created_utc) <- class(Sys.time())
  } else {
    bigQueryData <- data.frame(created_utc=numeric(0),
                               subreddit=character(0),
                               author=character(0),
                               domain=character(0),
                               num_comments=numeric(0),
                               score=numeric(0),
                               ups=numeric(0),
                               downs=numeric(0),
                               title=character(0),
                               selftext=character(0),
                               id=character(0),
                               gilded=numeric(0),
                               movie=character(0),
                               stringsAsFactors=FALSE)
    for(i in 1:length(movieQueries))
    {
      post.data <- query_exec(movieQueries[[i]][1], project = project, useLegacySql = FALSE, max_pages =
      post.data$movie = movieData[i,]$movie
      print(paste("The response has",nrow(post.data), "rows"))
      for(x in 1:nrow(post.data))
      {
        bigQueryData[nrow(bigQueryData)+1,] = post.data[x,]
      }
    }
    write.csv(bigQueryData, file = "bigQueryData.csv", na="NA")
  }
```

Creating an Analytics Base Table

```
checkDataQuality(data= bigQueryData, out.file.num="dq_num.csv", out.file.cat= "dq_cat.csv")
```

```
## Check for numeric variables completed // Results saved to disk // Time difference of 0.2717209 secs
## Check for categorical variables completed // Results saved to disk // Time difference of 1.461943 se
```

```
numericalQuality <- read.csv("dq_num.csv", header = TRUE)
categoricalQuality <- read.csv("dq_cat.csv", header = TRUE)

print(numericalQuality)
```

```
##               X non.missing missing missing.percent unique     mean min
## 1             X       42267       0            0.00  42267 21134.00   1
## 2 num_comments       42267       0            0.00    488    13.66   0
## 3        score       42267       0            0.00    846    36.25   0
## 4          ups       41814     453            1.07    841    35.68   0
## 5        downs       41814     453            1.07      2     0.00   0
## 6       gilded       42267       0            0.00      3     0.00   0
##       p1     p5    p10     p25   p50     p75     p90     p95      p99
## 1 423.66 2114.3 4227.6 10567.5 21134 31700.5 38040.4 40153.7 41844.34
## 2   0.00    0.0    0.0     0.0     1     4.0    17.0    36.0   169.00
## 3   0.00    0.0    0.0     1.0     1     5.0    34.0    91.0   561.00
## 4   0.00    0.0    0.0     1.0     1     5.0    34.0    92.0   557.00
## 5   0.00    0.0    0.0     0.0     0     0.0     0.0     0.0     0.00
## 6   0.00    0.0    0.0     0.0     0     0.0     0.0     0.0     0.00
##     max
## 1 42267
## 2 10389
## 3 13129
## 4  9424
## 5     0
## 6     2
```

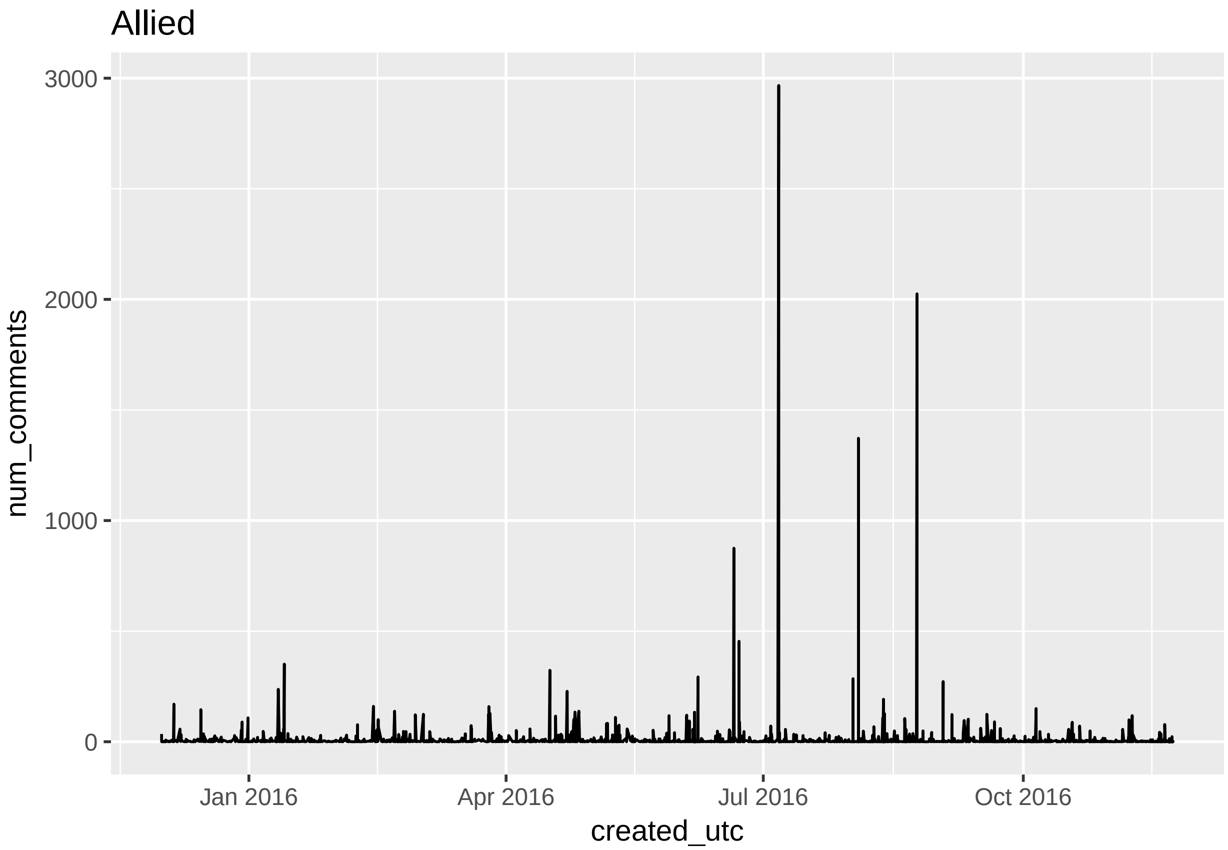
```
print(categoricalQuality)
```

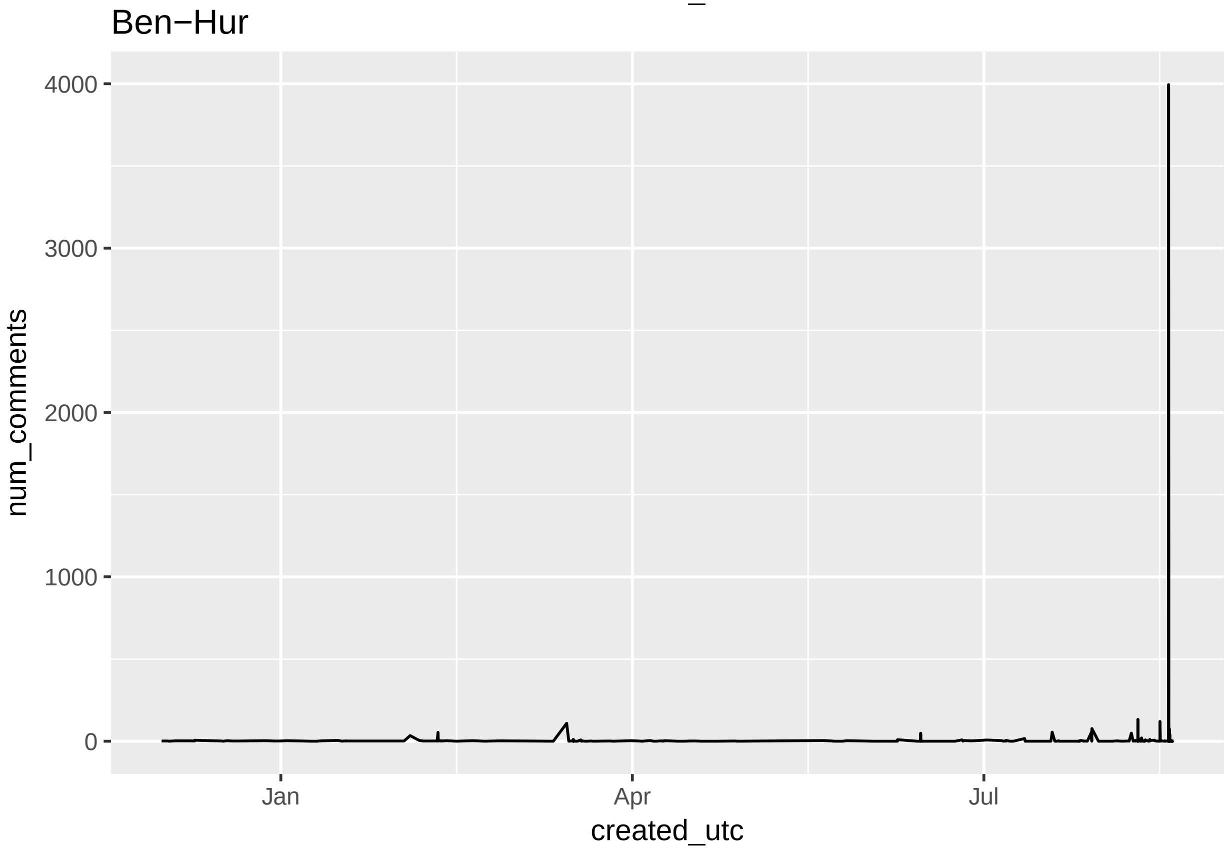
```
##           X n.non.miss n.miss n.miss.percent n.unique
## 1 subreddit      42233     34           0.08     4237
## 2    author      42267      0           0.00    15987
## 3    domain      42267      0           0.00     5143
## 4     title      42267      0           0.00    34847
## 5  selftext      13945  28322          67.01     4818
## 6        id      42267      0           0.00    42206
## 7     movie      42267      0           0.00       21
##                                              cat_1 freq_1         cat_2
## 1                                           movies   6218  DC_Cinematic
## 2                                        [deleted]   7743  ell_computer
## 3                                      youtube.com   7504     imgur.com
## 4 Rogue One: A Star Wars Story Trailer (Official)     61 Suicide Squad
## 5                                        [deleted]   6181     [removed]
## 6                                           3zfoiz      3        3zfp94
## 7                                     Ghostbusters   8937 Suicide Squad
##   freq_2
## 1   1983
## 2    635
## 3   2524
## 4     50
## 5   2870
## 6      3
## 7   8773
##
## 1
## 2
## 3
## 4
## 5 Watch... Batman v Superman: Dawn of Justice... Full... Movie... Free... Streaming... Online... wit
## 6
## 7
##   freq_3
## 1   1151
## 2    427
## 3   2115
## 4     49
## 5     12
## 6      2
## 7   6365
##  
## 1
## 2
## 3
## 4
## 5 **Goals: FUN, Community, and Dank Memes**\n\n**Information:**\nTired of all the mil-sim bullshit?
## 6
## 7
##   freq_4
## 1   1146
## 2    406
## 3   1970
## 4     44
## 5      6
## 6      2
## 7   3961
##  
## 1
## 2
## 3
## 4
## 5 **Goals: FUN, Community, and Dank Memes**\n\n**Information:**\nTired of all the mil-sim bullshit?
## 6
## 7
##   freq_5
## 1    788
## 2    384
## 3   1042
## 4     41
## 5      5
## 6      2
## 7   2549
##  
## 1
## 2
## 3
## 4
## 5 **Goals: FUN, Community, and Dank Memes**\n\n**Information:**\nTired of all the mil-sim bullshit?
## 6
## 7
##   freq_6                                  cat_7 freq_7
## 1    769                                 Marvel    767
## 2    369                                ImaBlue    351
## 3    759                      self.DC_Cinematic    733
## 4     40 Suicide Squad - Official Trailer 1 [HD]     40
## 5      4                                 Title.      4
## 6      2                                 3yxbxf      2
## 7   2340      Batman v Superman: Dawn of Justice   1499
##  
## 1
## 2
## 3
## 4
## 5 ****\n|*|*|*|\n:---|:---|:--:|\n**[Steam profile](http://steamcommunity.com/id/ZenchiZennou/#btn)**
## 6
## 7
##   freq_8
## 1    652
## 2    333
## 3    453
## 4     36
## 5      3
## 6      2
## 7   1333
##
## 1
## 2
## 3
## 4
## 5 **Synopsis:** A former Special Forces operative turned mercenary is subjected to a rogue experiment
## 6
## 7
##   freq_9                              cat_10 freq_10
## 1    602                            DCcomics     536
## 2    268                     ImagesOfNetwork     264
## 3    440                   cinematographe.it     409
## 4     32 Rogue One: A Star Wars Story (2016)      32
## 5      3                      Maria Williams       3
## 6      2                              41ly3i       2
## 7   1188                        Finding Dory    1023
```

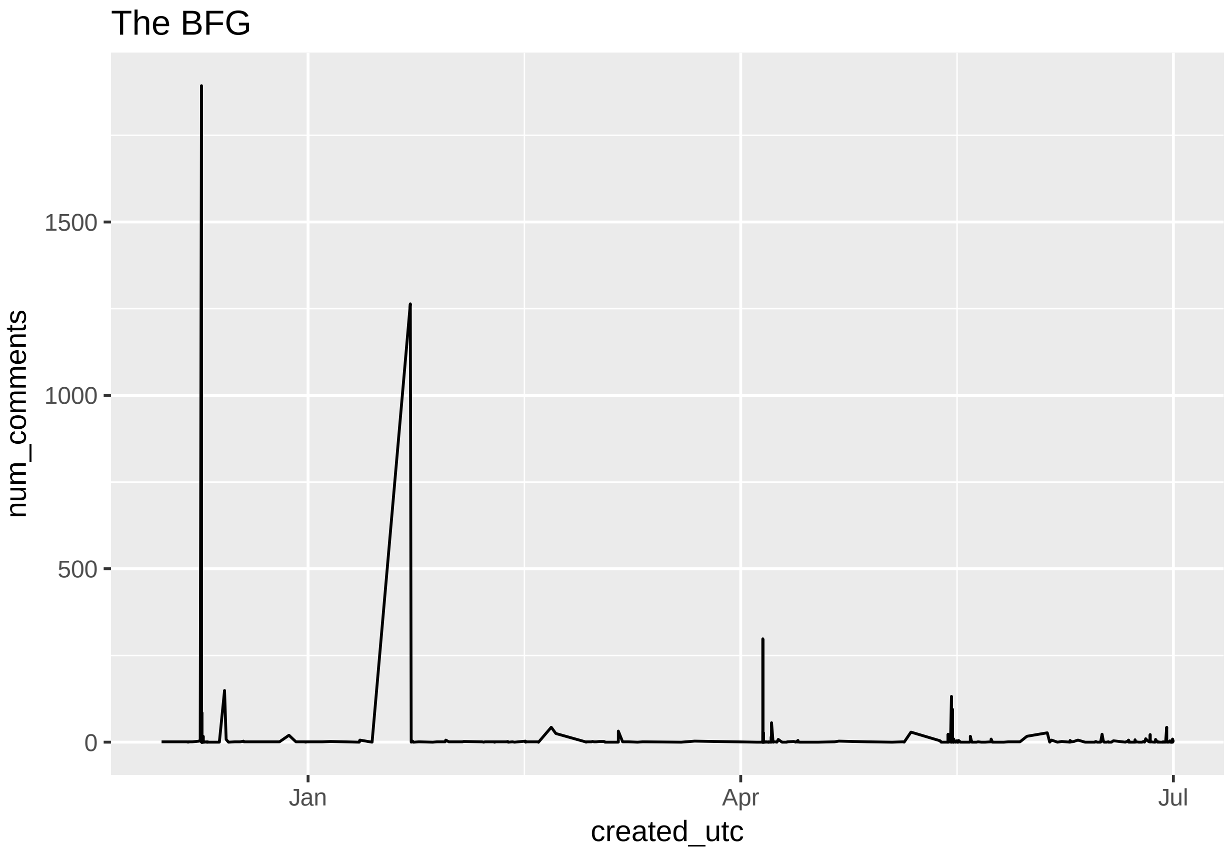
Exploring Data

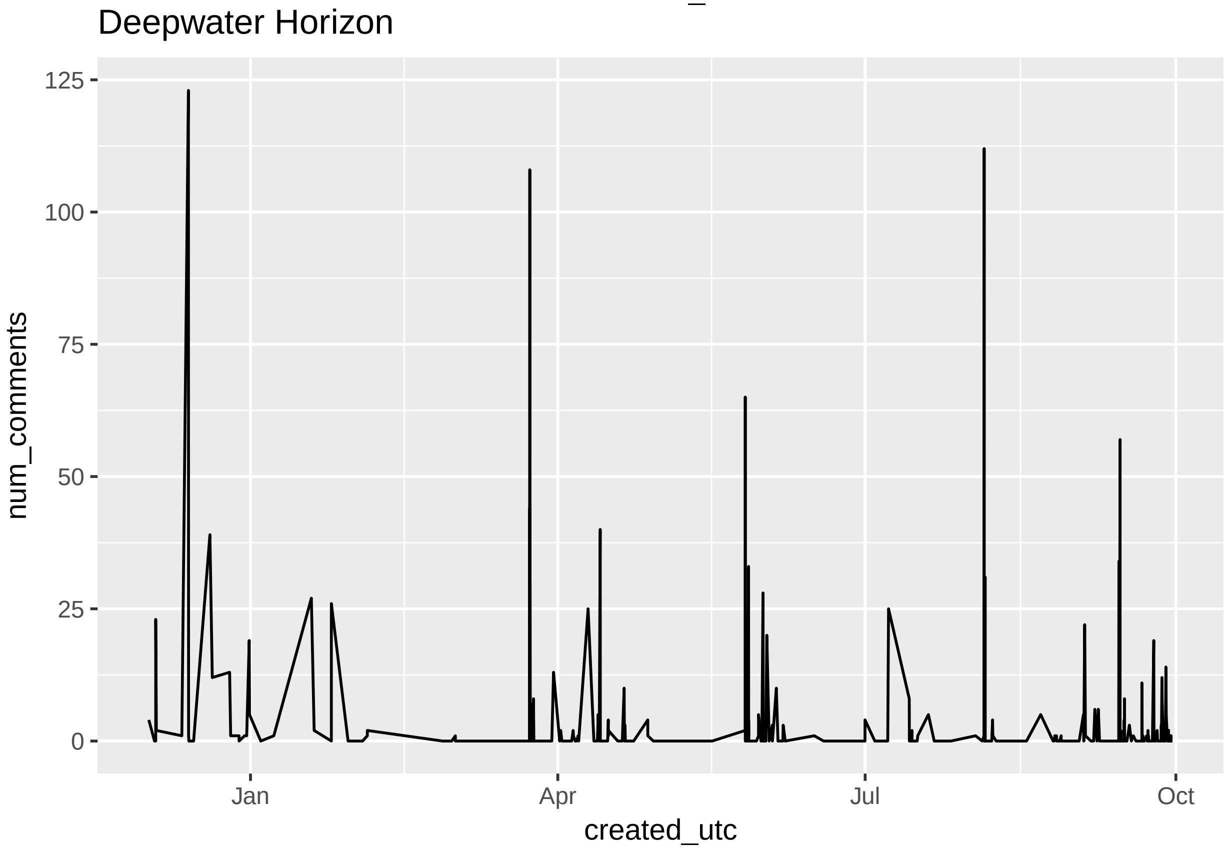
In exploring my data I wanted to just look at basic patterns in the data, and it looks like there are some general trends in a few of the fields. I'll be able to do some better analysis later, when I implement Plotly so I can easily change around the data.

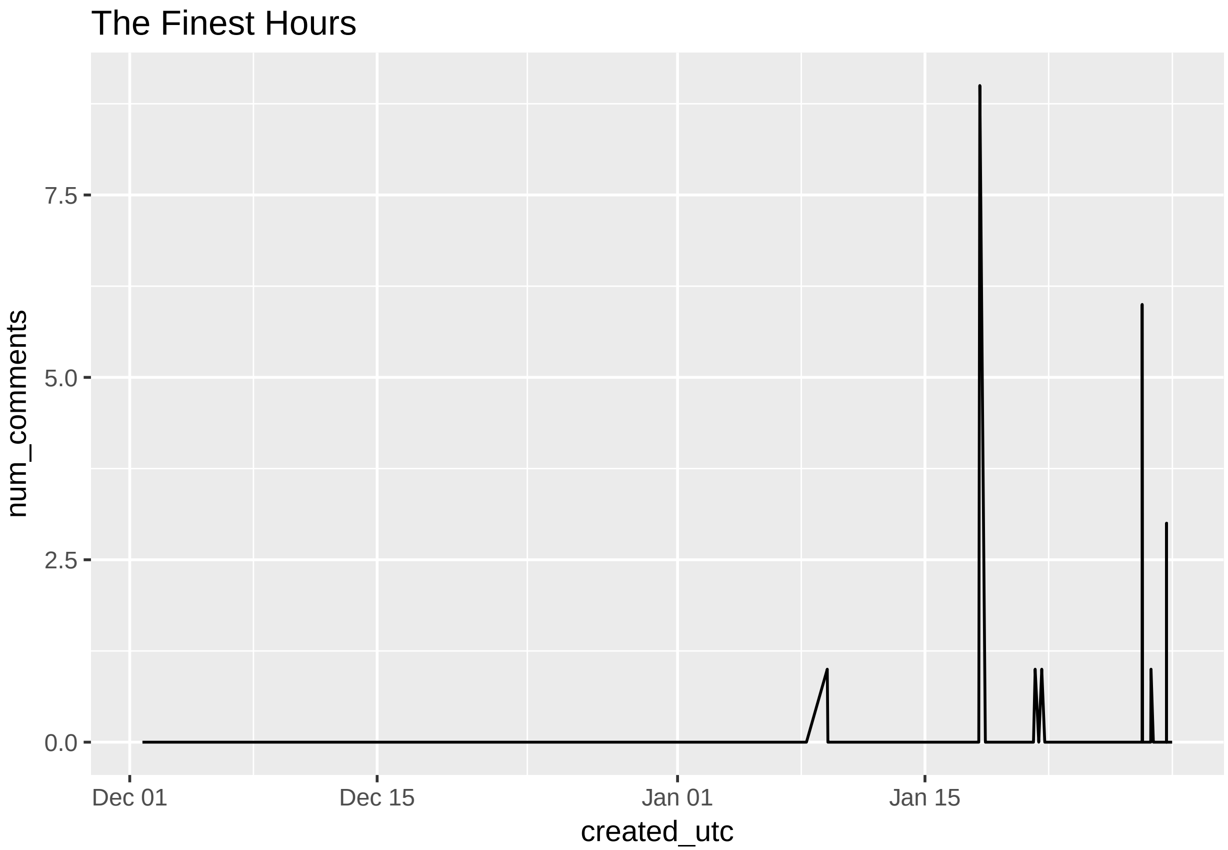
```
for(movie in movies)
{
  p <- ggplot(bigQueryData[bigQueryData$movie == movie,], aes(x = created_utc, y = num_comments)) + geom
  print(p)
}
```

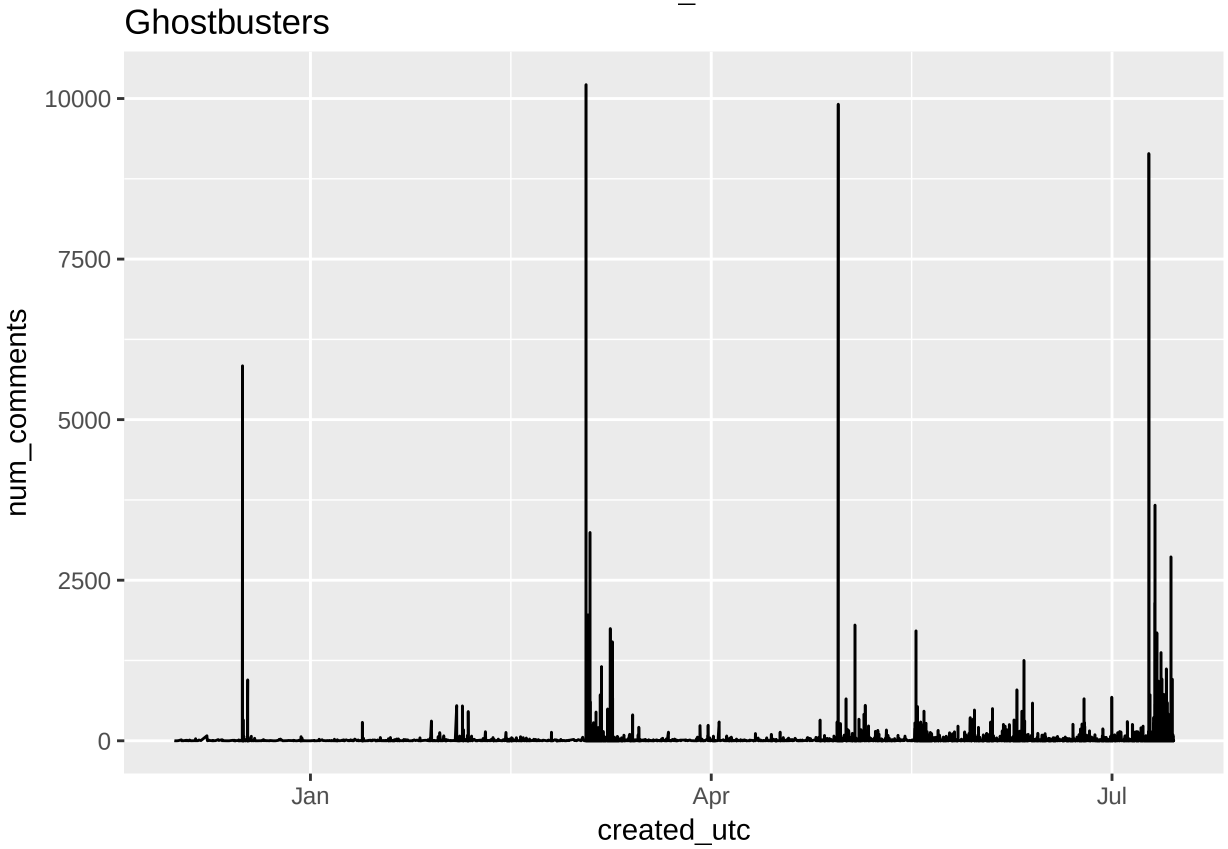

Allied
3000
2000
1000
0
num_comments
Jan 2016
Apr 2016
Jul 2016
Oct 2016
created_utc

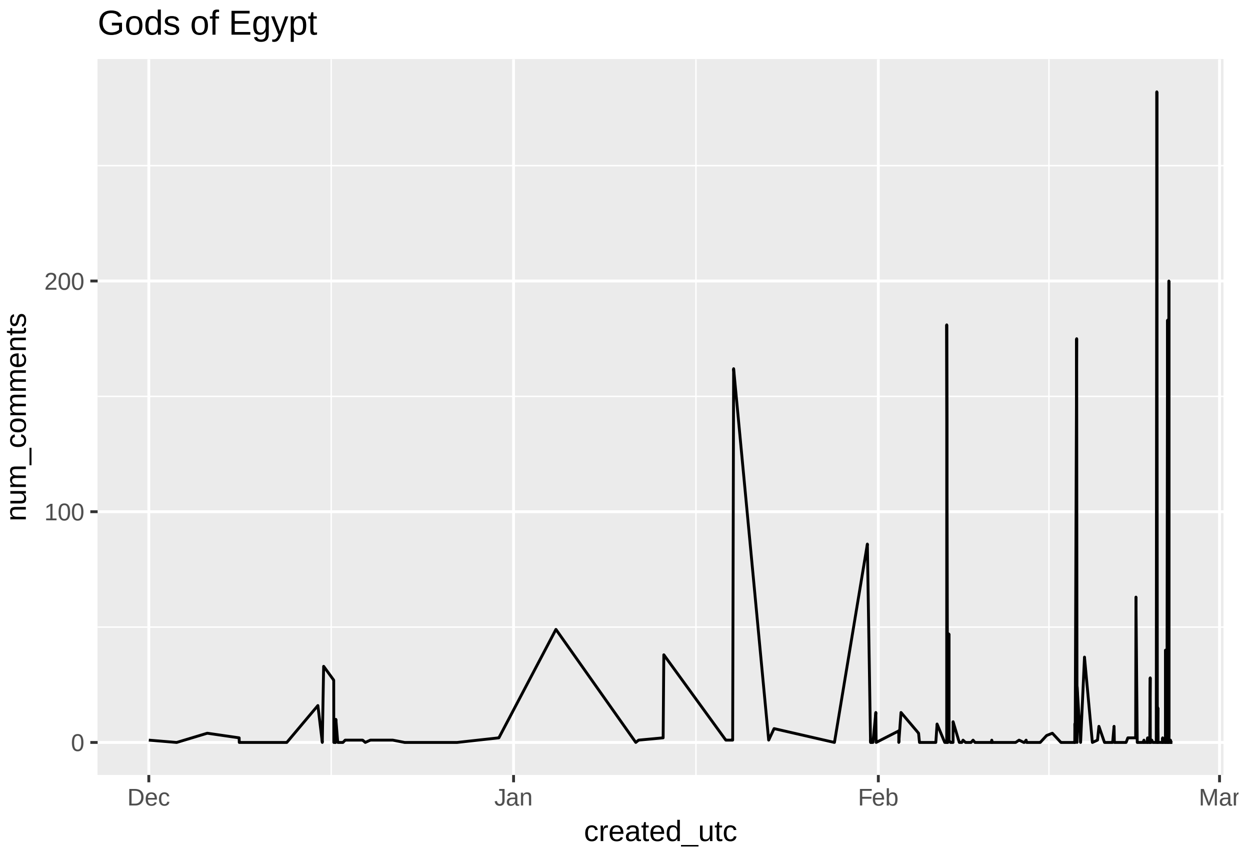

Ben−Hur
4000
3000
2000
1000
0
num_comments
Jan
Apr
Jul
created_utc

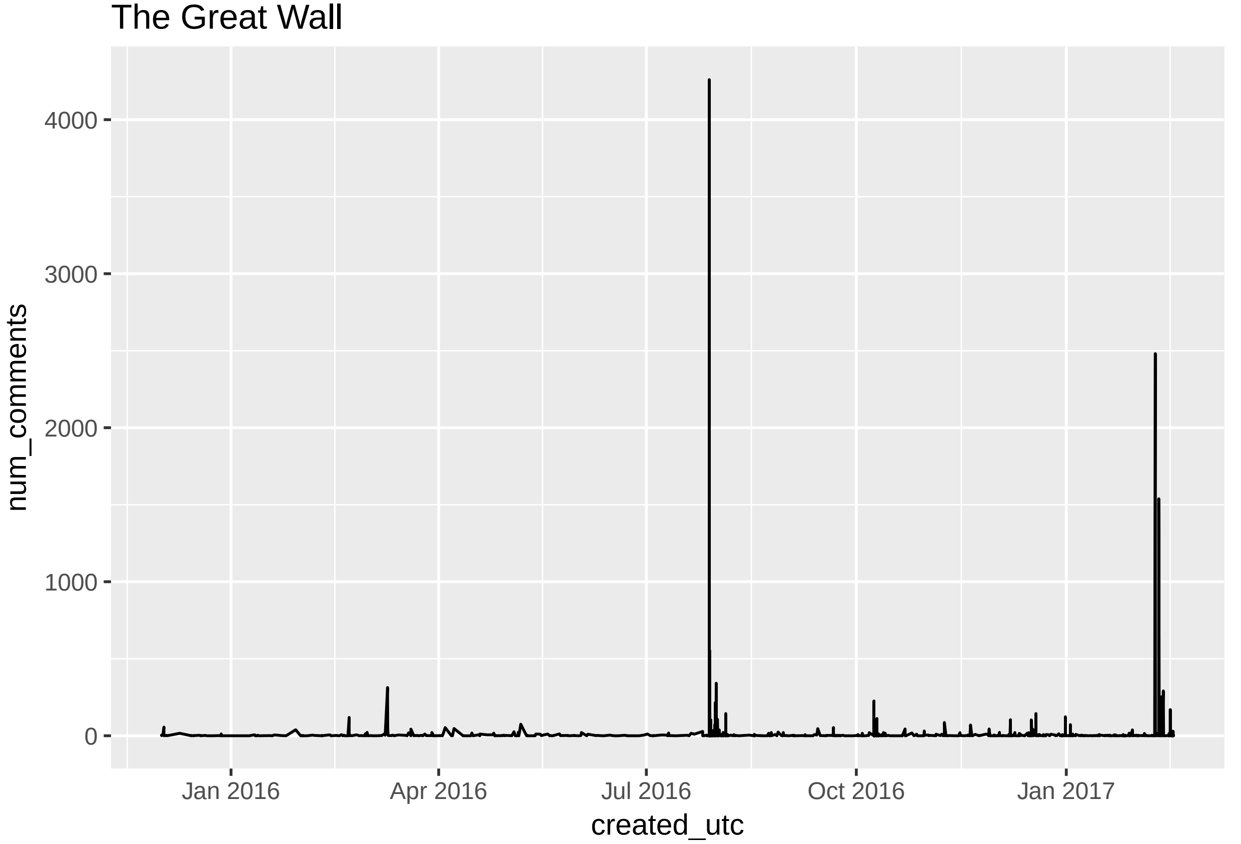

The BFG
num_comments
1500
1000
500
0
Jan
Apr
Jul
created_utc

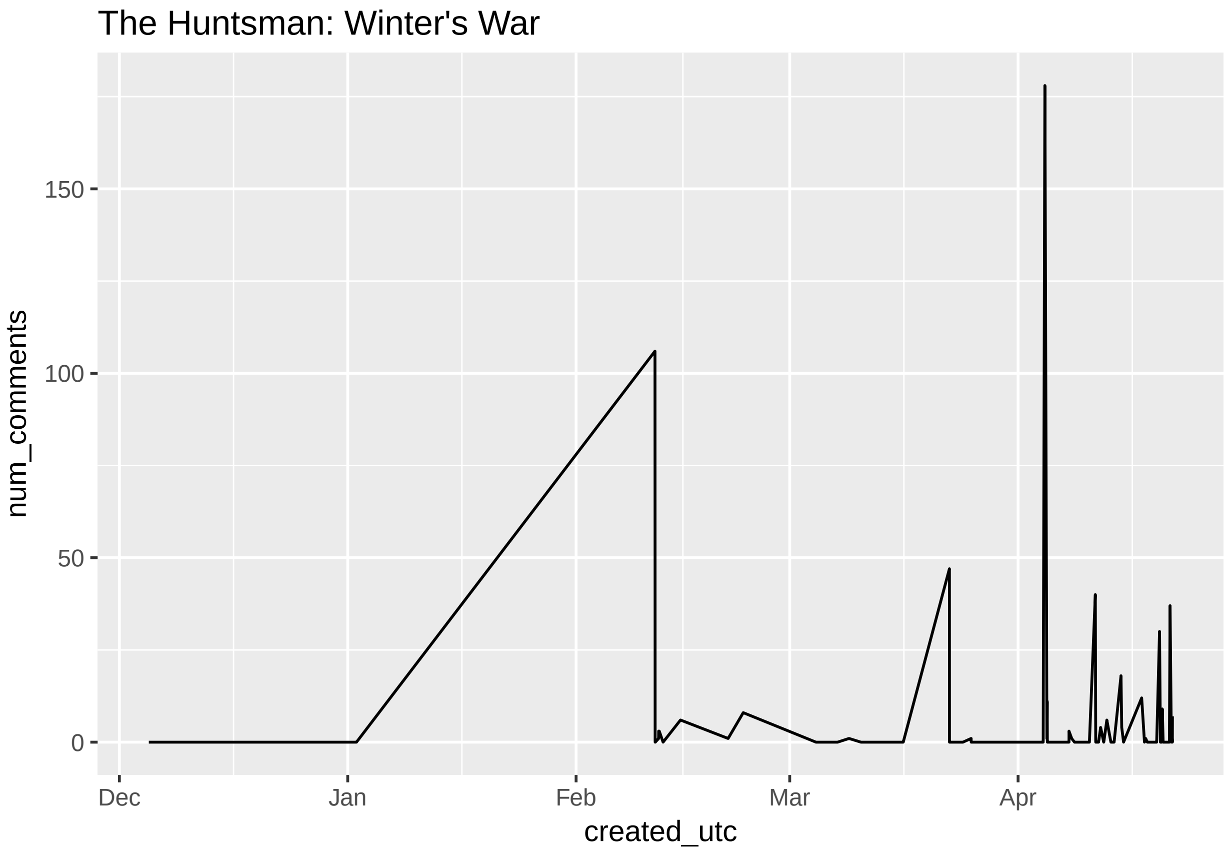

Deepwater Horizon
num_comments
125
100
75
50
25
0
Jan
Apr
Jul
Oct
created_utc


The Finest Hours
num_comments
7.5
5.0
2.5
0.0
Dec 01
Dec 15
Jan 01
Jan 15
created_utc

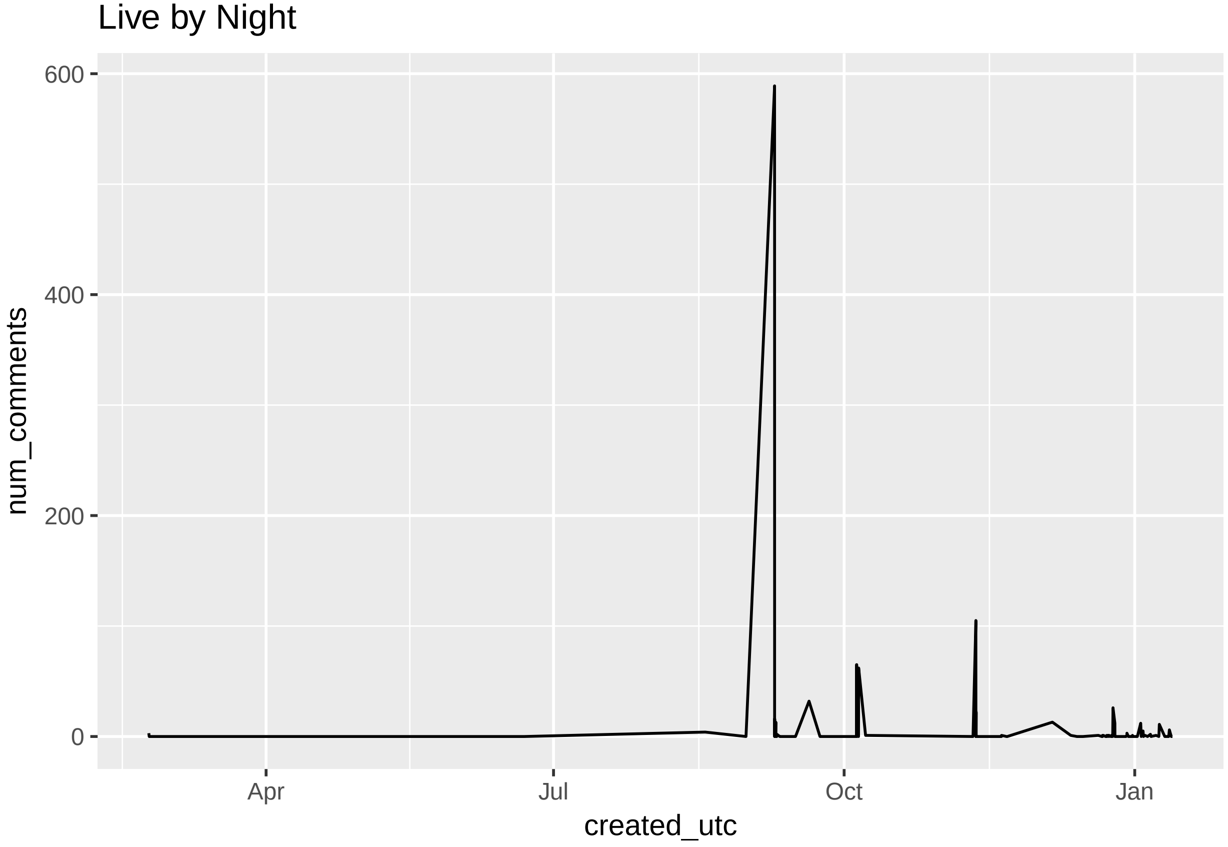

Ghostbusters
num_comments
10000
7500
5000
2500
0
Jan
Apr
Jul
created_utc


Gods of Egypt
num_comments
0
100
200
Dec
Jan
Feb
Mar
created_utc

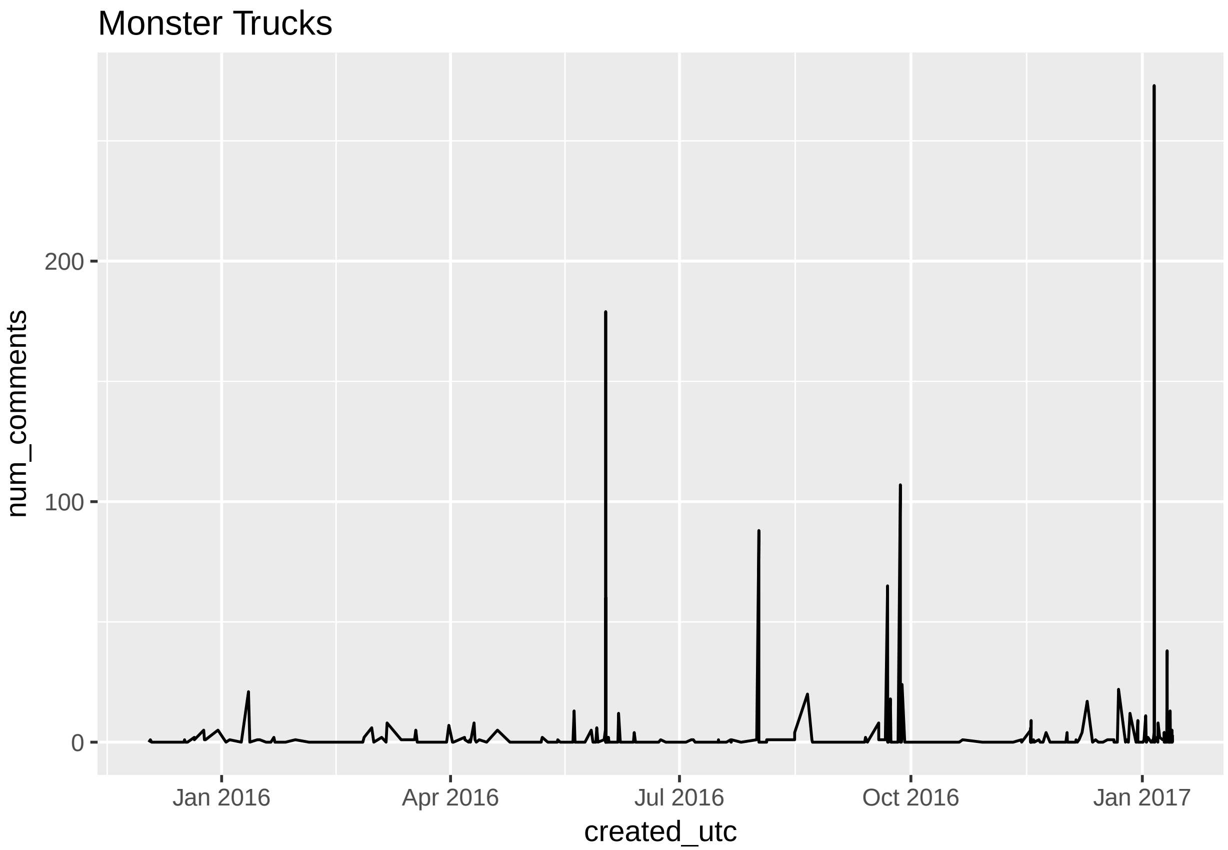

The Great Wall
num_comments
0
1000
2000
3000
4000
Jan 2016
Apr 2016
Jul 2016
Oct 2016
Jan 2017
created_utc

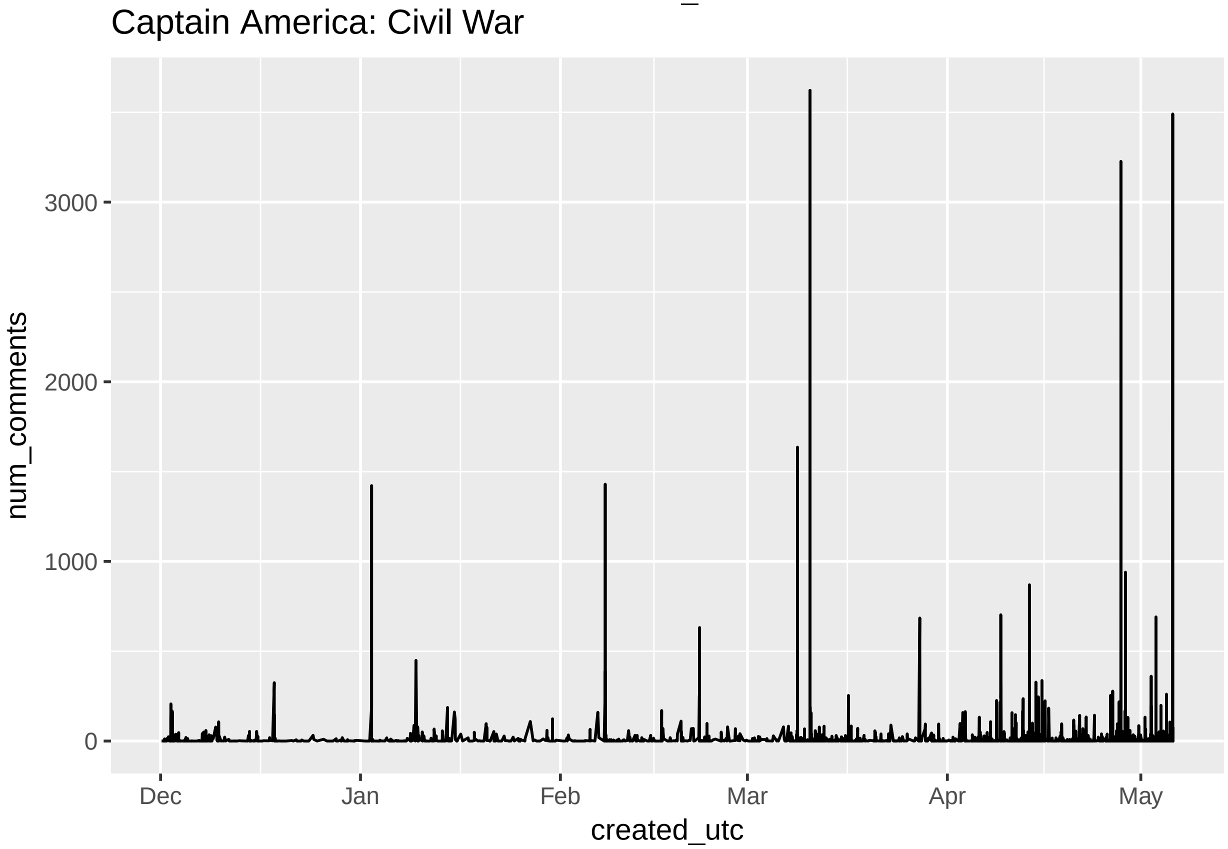


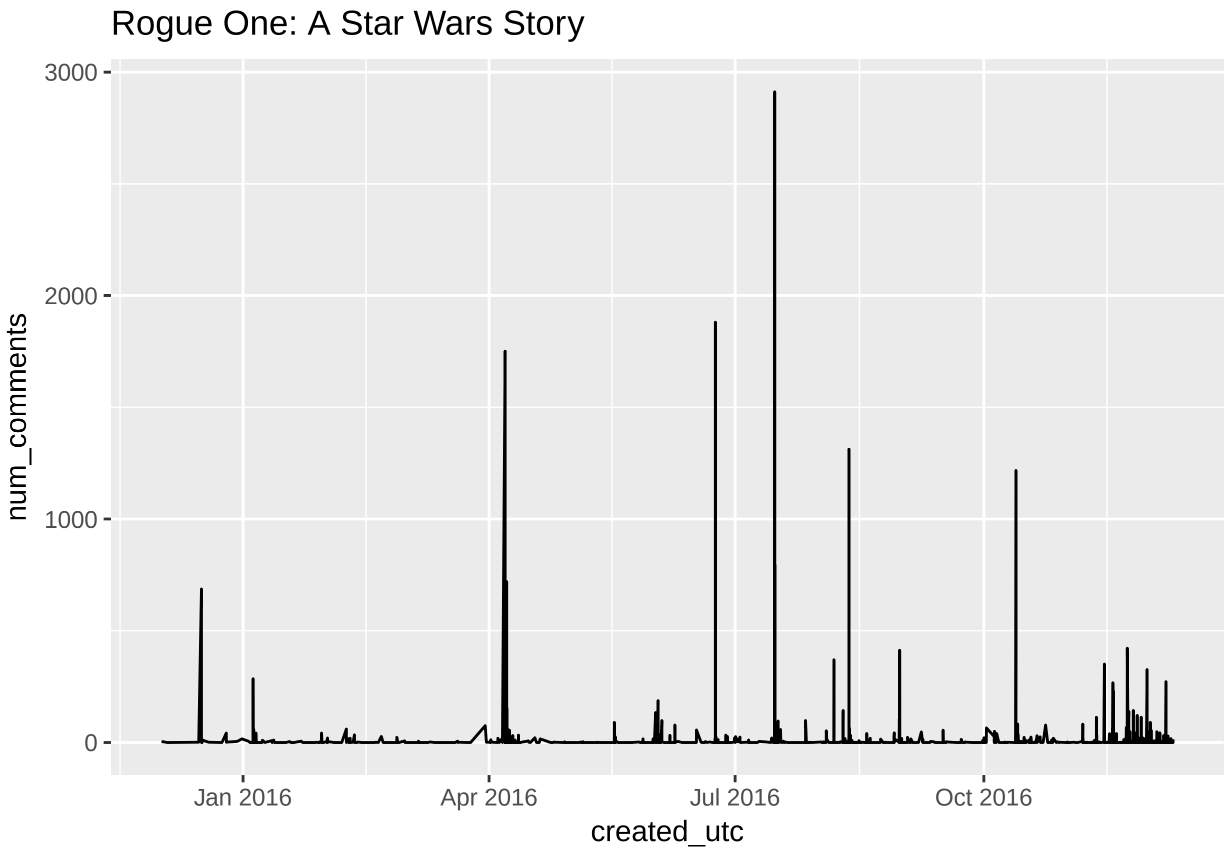
The Huntsman: Winter's War

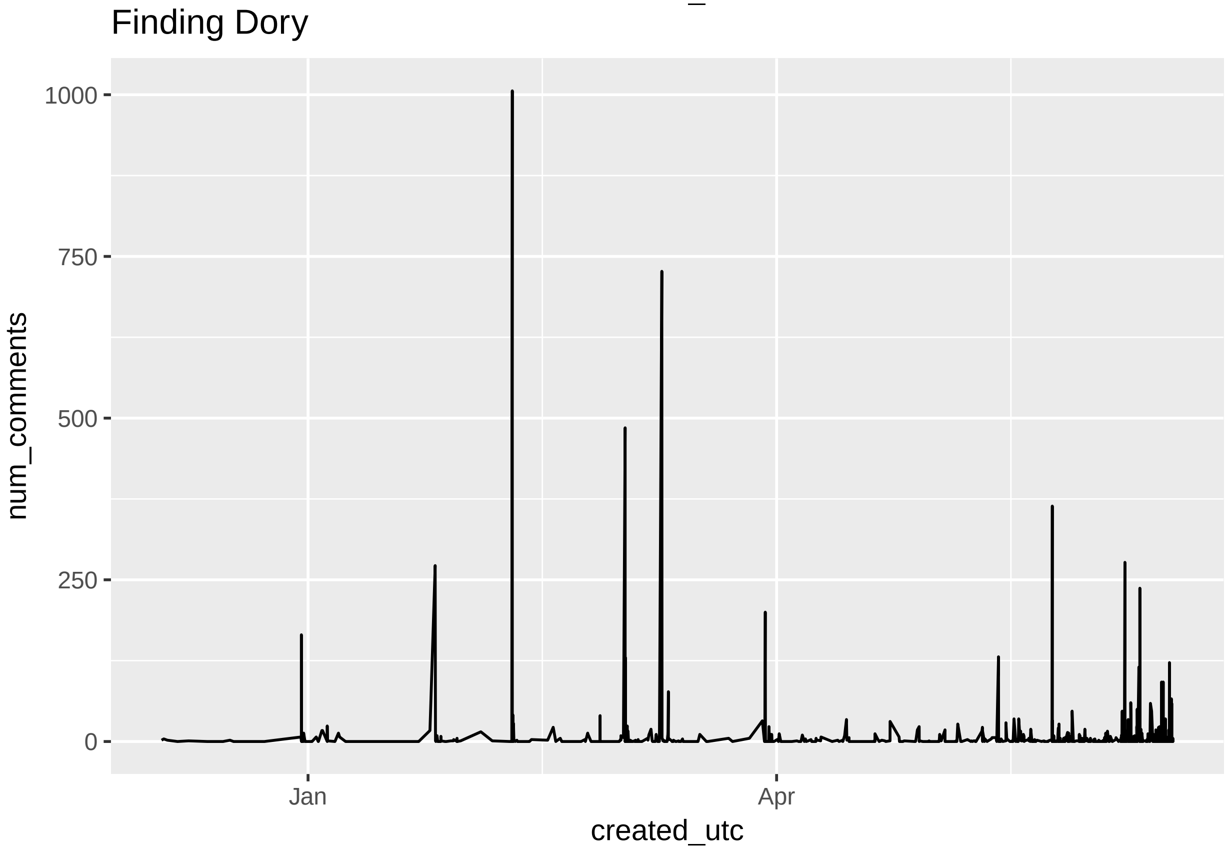


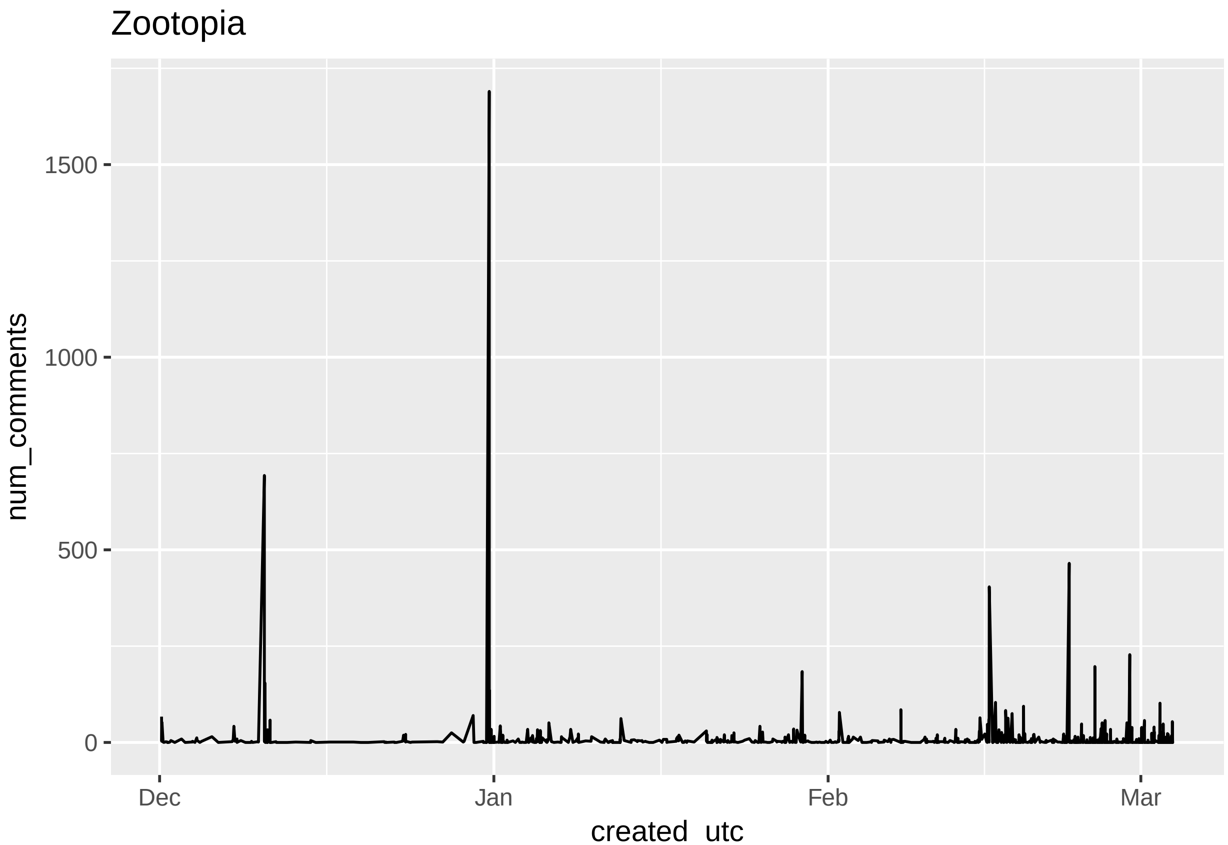
Live by Night


Monster Trucks
num_comments
0
100
200
Jan 2016
Apr 2016
Jul 2016
Oct 2016
Jan 2017
created_utc

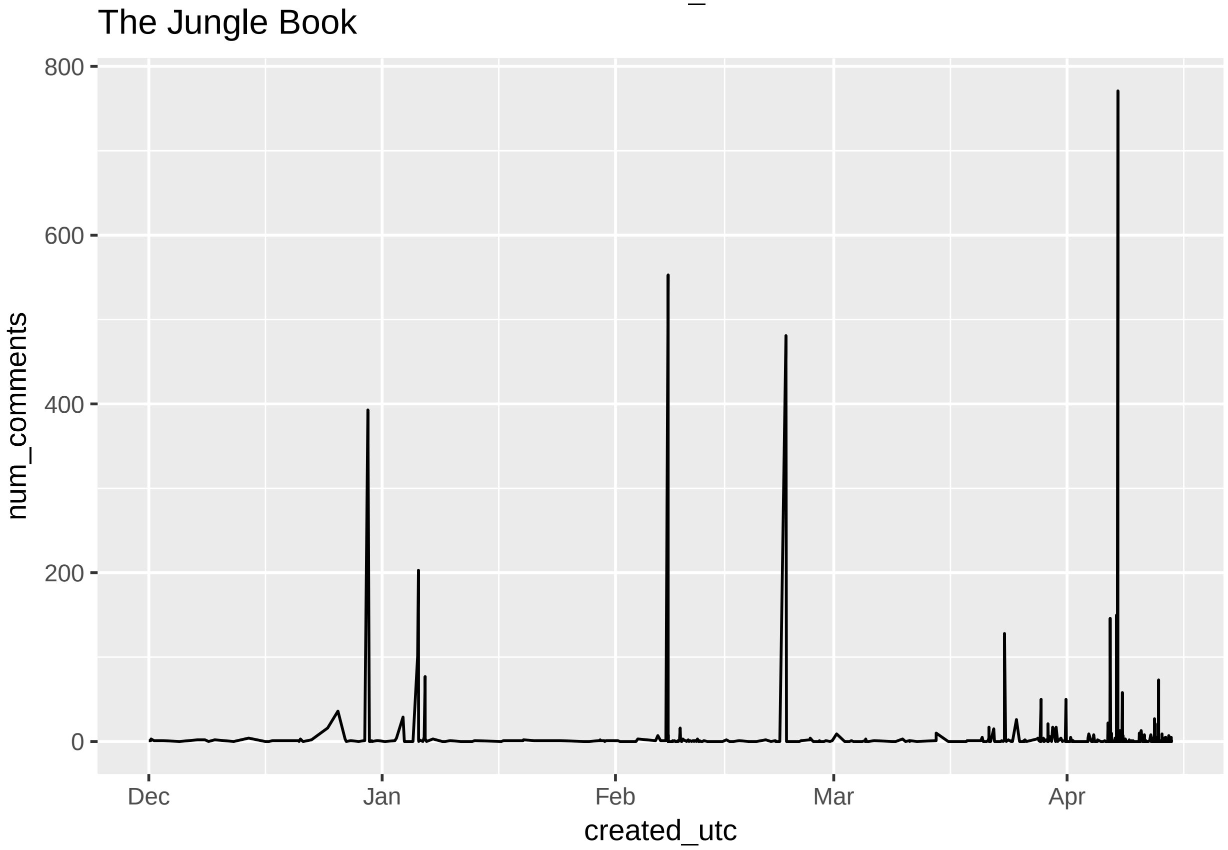

Captain America: Civil War
num_comments
0
1000
2000
3000
Dec
Jan
Feb
Mar
Apr
May
created_utc


Rogue One: A Star Wars Story
num_comments
3000
2000
1000
0
Jan 2016
Apr 2016
Jul 2016
Oct 2016
created_utc

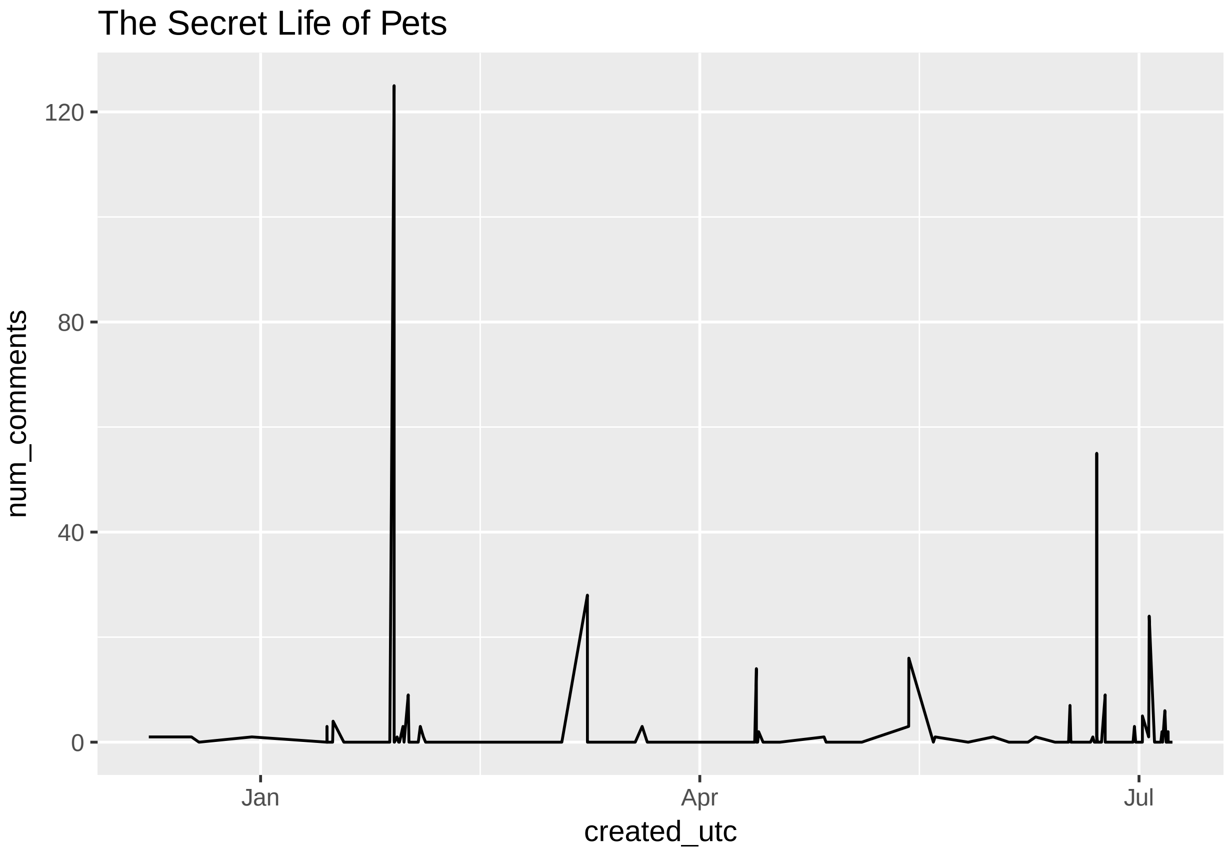

Finding Dory
num_comments
1000
750
500
250
0
Jan
Apr
created_utc

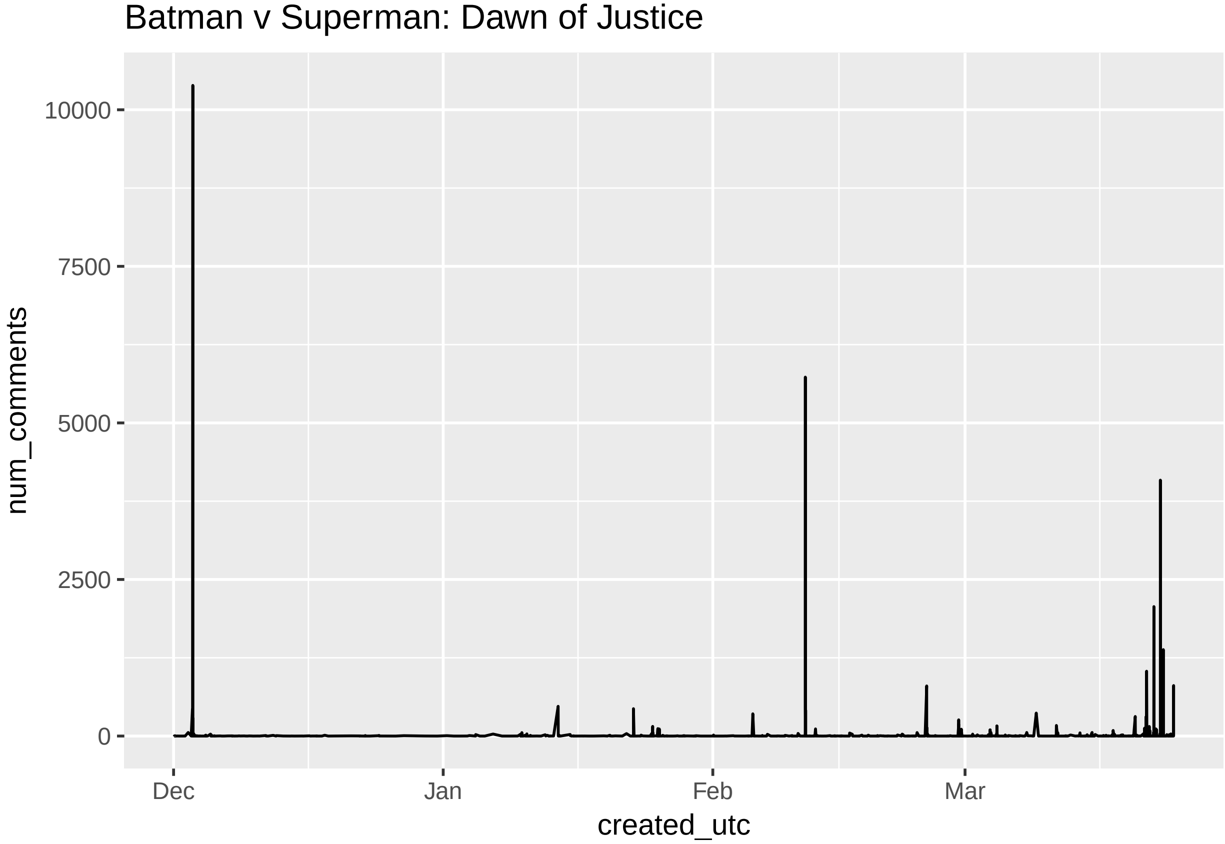


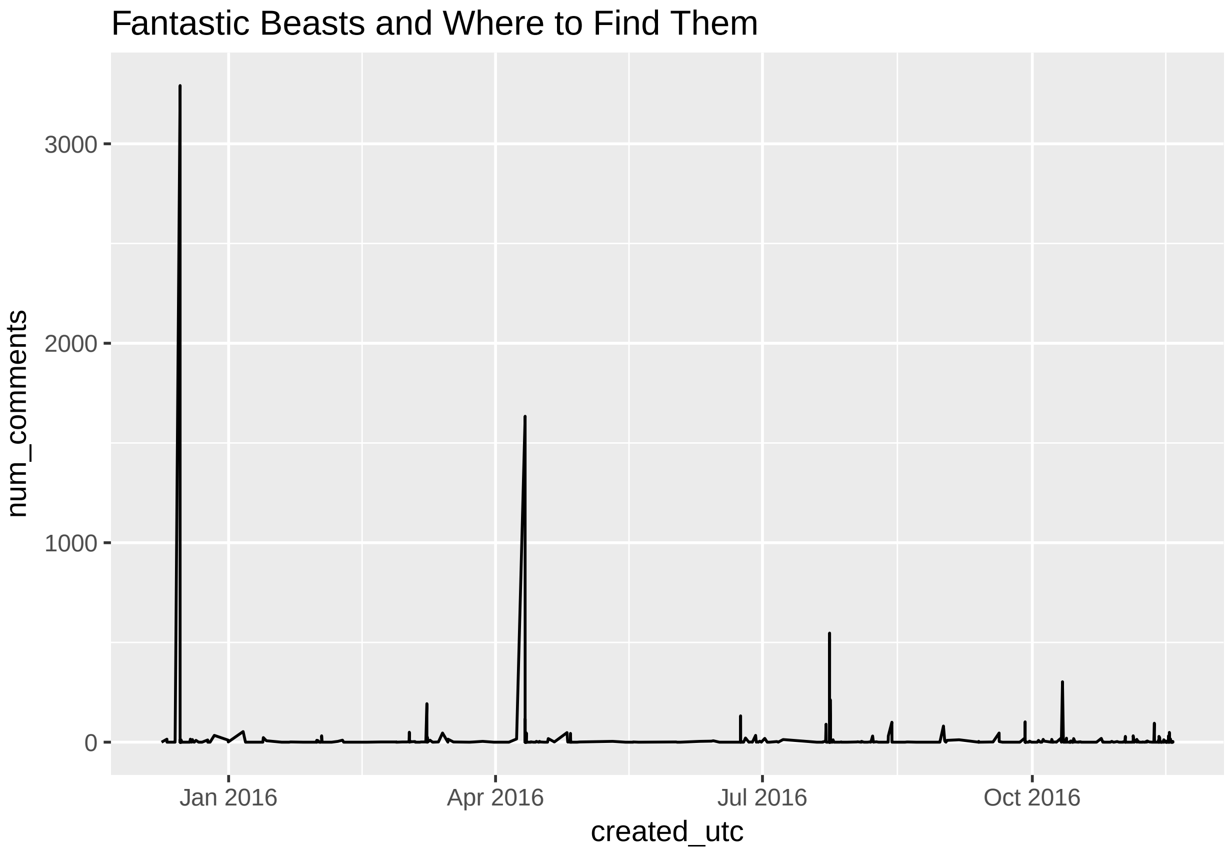
Zootopia

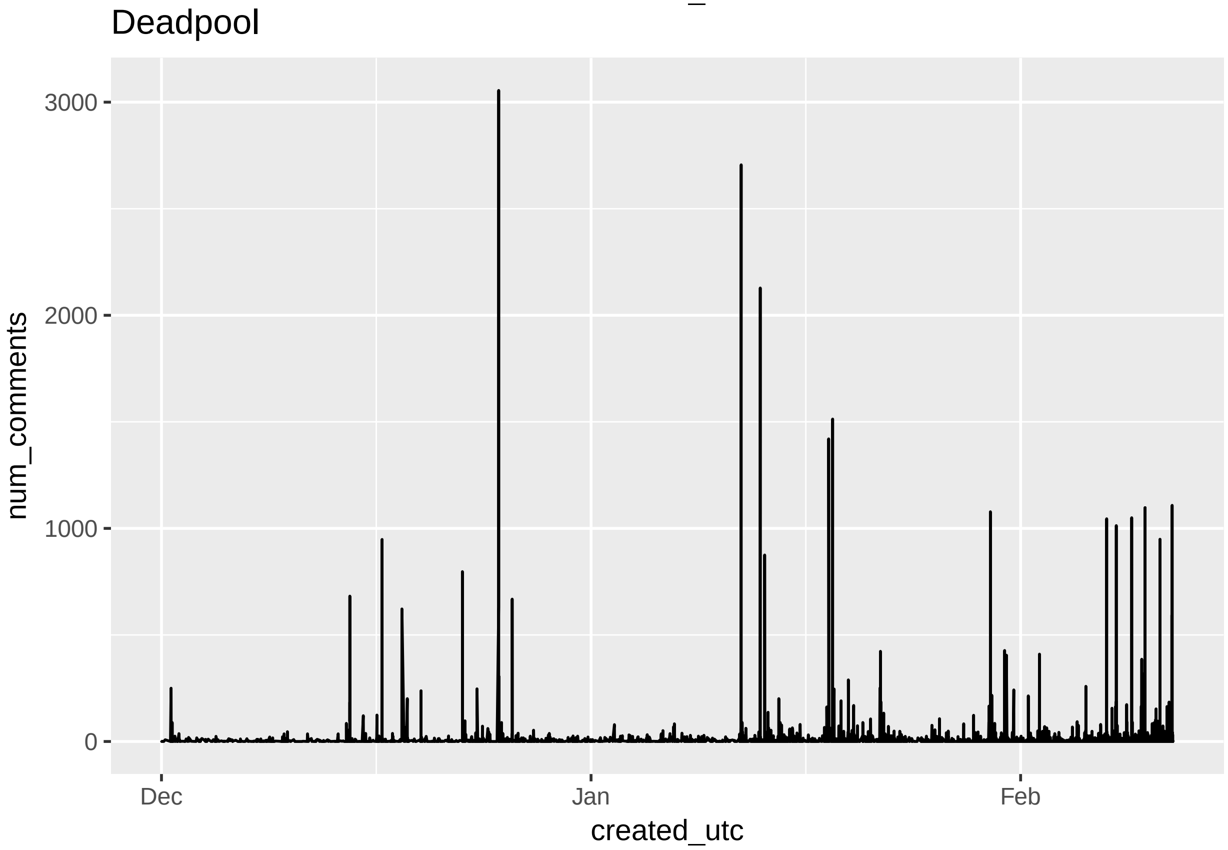


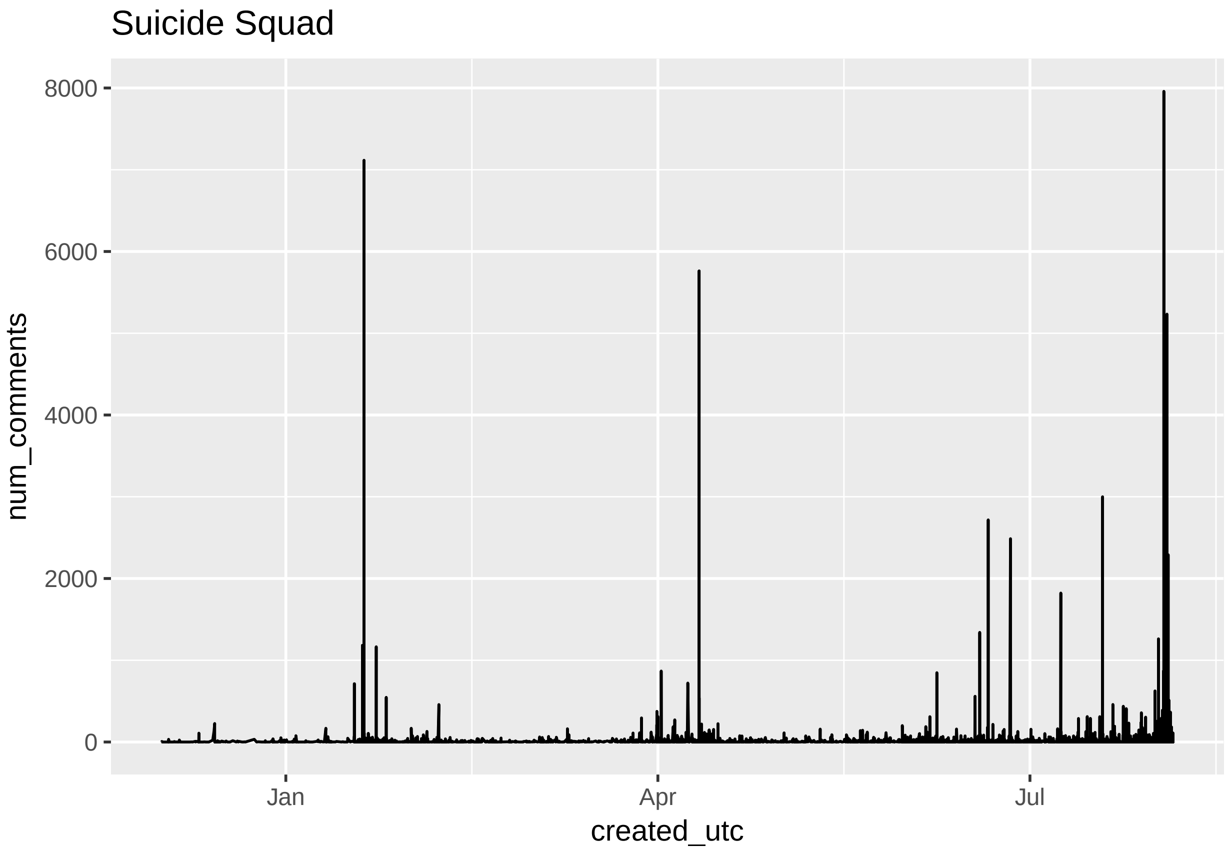
The Jungle Book


The Secret Life of Pets


Batman v Superman: Dawn of Justice


Fantastic Beasts and Where to Find Them
num_comments
3000
2000
1000
0
Jan 2016
Apr 2016
Jul 2016
Oct 2016
created_utc


Deadpool
num_comments
3000
2000
1000
0
Dec
Jan
Feb
created_utc



Techniques to be used in predictions

I believe the two best techniques to be used for my predictions is going to be either a random forest or using a naive bayesian model. It also may be useful to use a classification algorithm to simplify my problem; rather than trying to predict an exact box office outcome, I could also try and predict whether the movie is a flop, breakeven, or hit. Breaking it up into a categorical variable would allow me to use a support vector machine.