



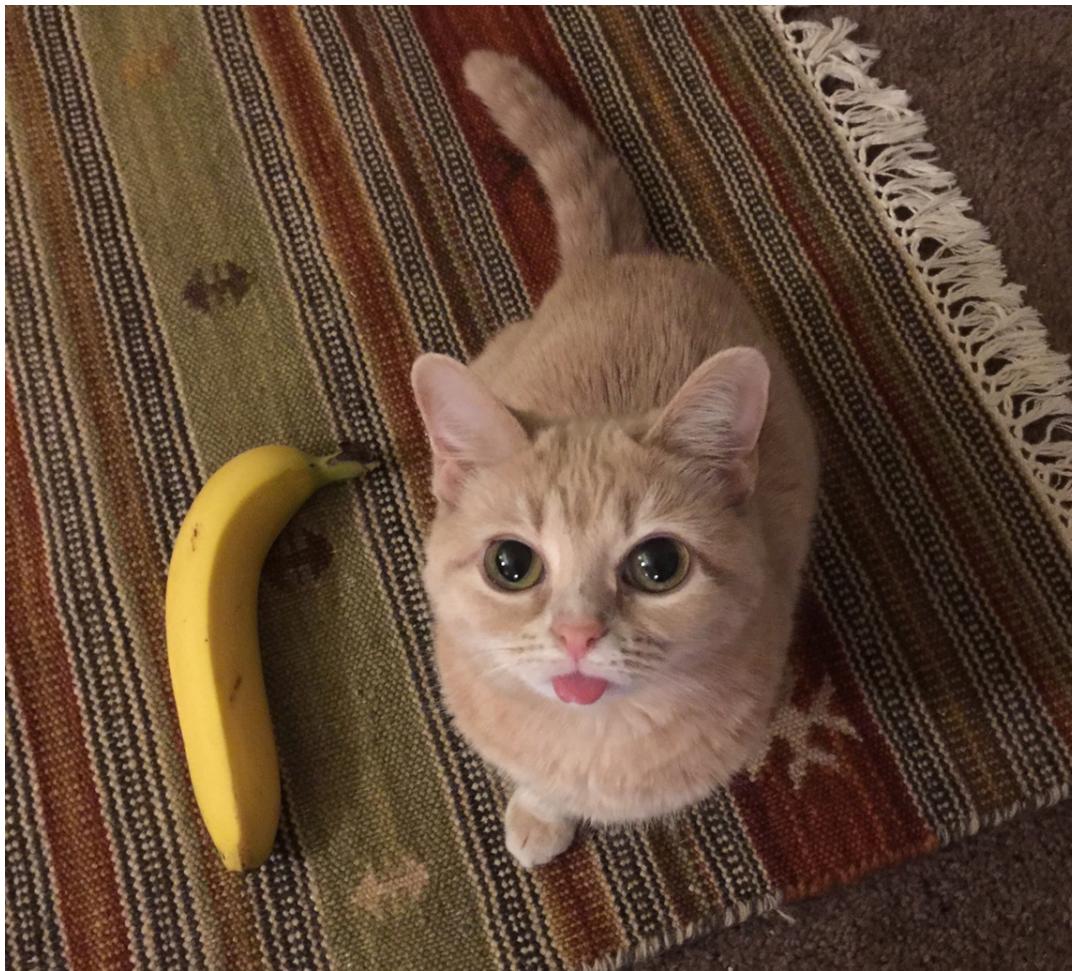
# The Best Words

Data mining Reddit to  
capture political moods



**reddit**

# Reddit as the internet's id



# Reddit as a political looking glass

The screenshot shows the homepage of the [r/Politics](#) subreddit. At the top, there's a banner featuring the Reddit logo with a patriotic theme (stars and stripes) and the text "r/Politics". Below the banner is a dark blue navigation bar with links for HOT, NEW, RISING, CONTROVERSIAL, TOP, GILDED, WIKI, and PROMOTED. A welcome message reads: "Welcome to /r/Politics! Please read [the wiki](#) before participating." The first post in the feed is a promotional one from "reddit\_exchanges" about a tennis exchange, with 63 upvotes. The second post is a "Sticky Post" from "PoliticsModeratorBot" titled "Megathread: Federal Court overturns President Trump's executive order regarding immigration", with 1.7k upvotes. The third post is from "The Atlantic" with 953 upvotes, titled "Trump Gives Stephen Bannon Access to the National Security Council". The fourth post is from "Slate" with 2.2k upvotes, titled "Paul Ryan Is Defending Trump's Discrimination. He Is an Embarrassment."

Welcome to /r/Politics! Please read [the wiki](#) before participating.

63 It's the first serve for the Reddit Gifts Tennis Exchange! Are you open to it? We thought so! Sign up today and exchange gifts with tennis lovers all around the world! ([redditgifts.com](#))  
promoted by reddit\_exchanges  
[promoted](#) [save](#) [hide](#) [report](#)

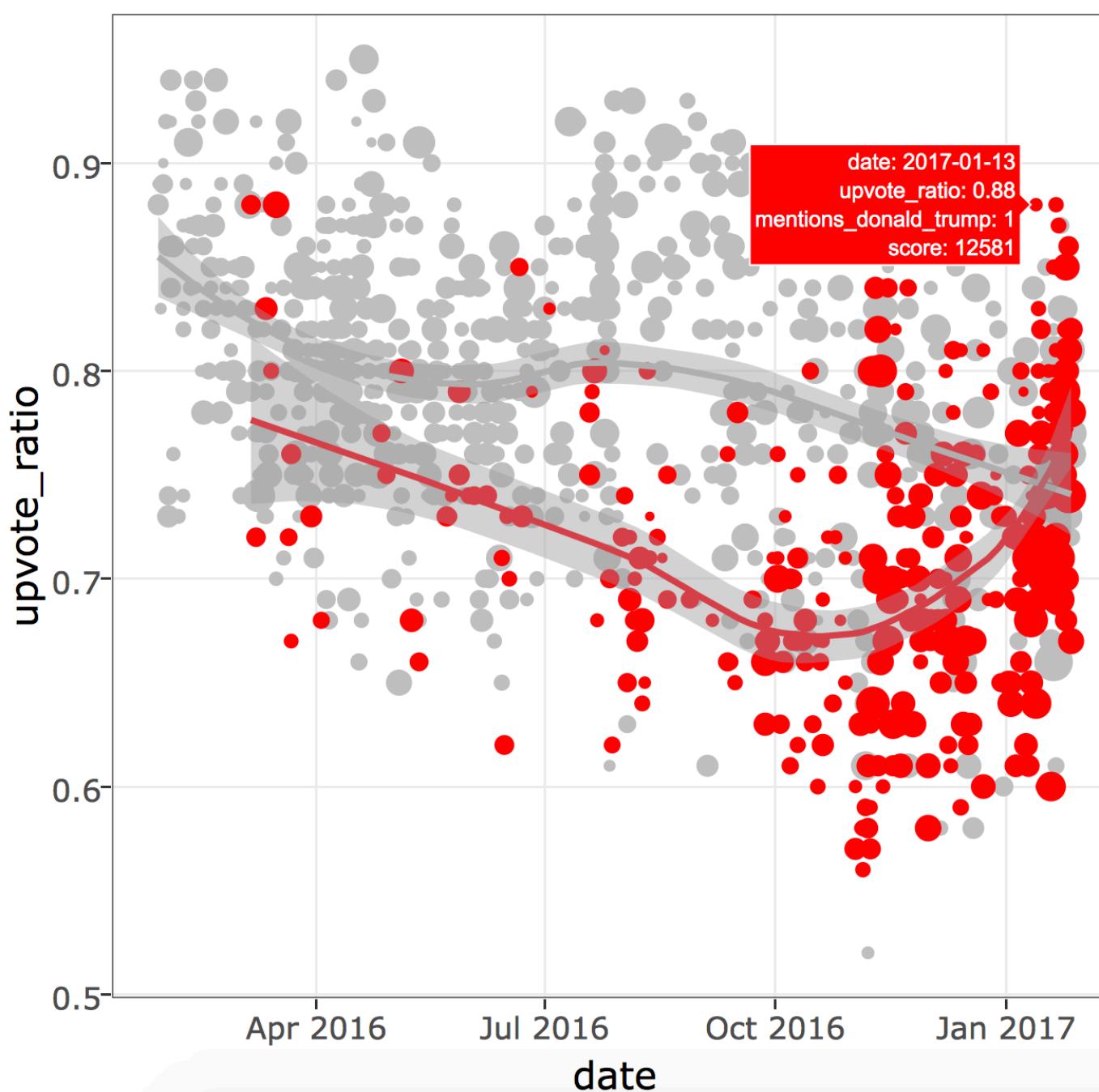
1.7k Megathread: Federal Court overturns President Trump's executive order regarding immigration self.politics  
submitted 9 hours ago \* by PoliticsModeratorBot [M] - POLITICS MOD POST  
[14580 comments](#) [share](#) [save](#) [hide](#) [report](#)

953 Trump Gives Stephen Bannon Access to the National Security Council theatlantic.com  
submitted 8 hours ago by Evolve\_or\_Bye Puerto Rico  
[1406 comments](#) [share](#) [save](#) [hide](#) [report](#)

2.2k Paul Ryan Is Defending Trump's Discrimination. He Is an Embarrassment. slate.com  
submitted 10 hours ago by buy\_iphone\_7 America  
[797 comments](#) [share](#) [save](#) [hide](#) [report](#)

# This project:

- Analyse the 1000 most “top” and “controversial” threads and comment trees on /r/ politics for previous 12 months
- Methods
  - Python package **PRAW**, a wrapper for Reddit’s API + Jupyter notebook
  - Data cleaning, analysis and sentiment mapping using R
    - tidyverse, stringr, purrr, tidytext
  - Microsoft Cognitive Services Azure **Text Analytics API**
  - Data visualisation
    - ggplot2, plotly

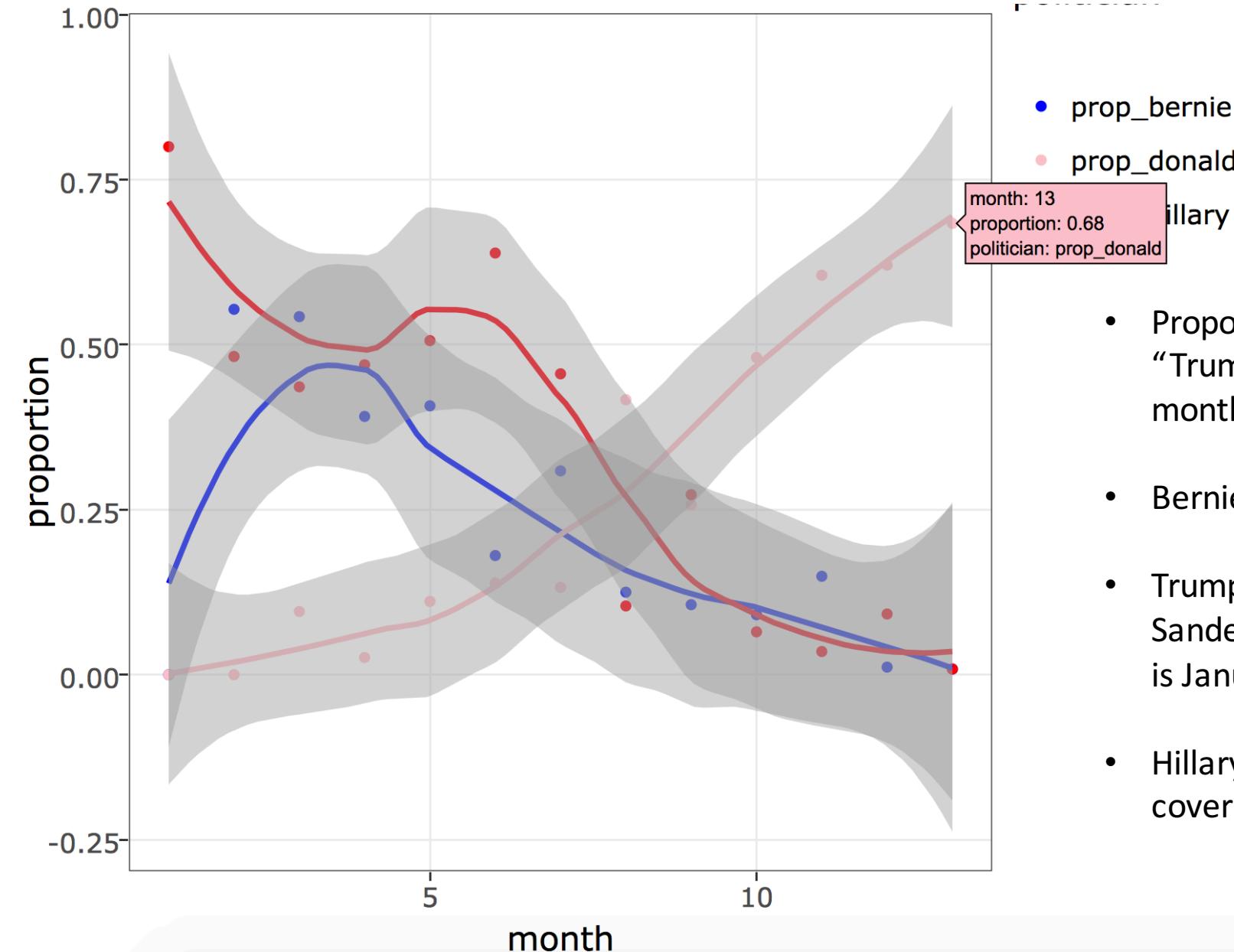


5/5

# Upvote For Trump 2016

- The top 1000 posts in /r/politics in last year, have gradually gotten more controversial (lower upvote ratio) over the past 12 months
- Tons more post titles mention Trump (red dots) after the election (too late!)
- Trump (red line) started to be viewed much more positively starting in October! Bad omen.

# The eclipse



- Proportion of top submissions containing "Trump" or "Donald" over the last 12 months. It's now (month=13) up to 69%!
- Bernie and Hillary down to ~ 0%.
- Trump publicity really overtook Sanders/Clinton in late march (month of 0 is January 2016)
- Hillary has always enjoyed higher coverage than Bernie.

# Trends in hate, disgust, fear, etc. towards Trump, Clinton, Sanders

	word	sentiment
	<chr>	<chr>
1	abacus	trust
2	abandon	fear
3	abandon	negative
4	abandon	sadness
5	abandoned	anger
6	abandoned	fear

:

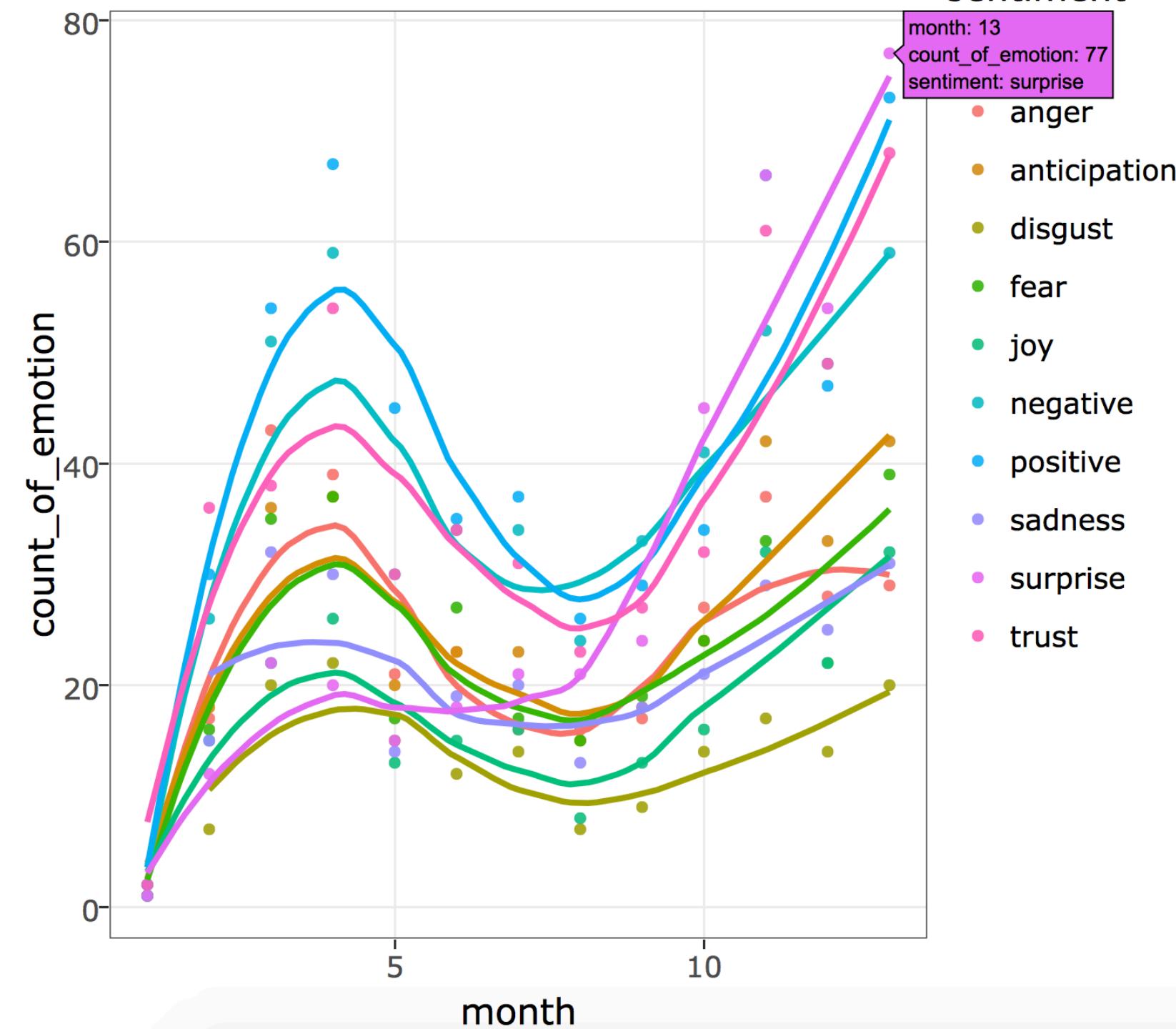
N=13901

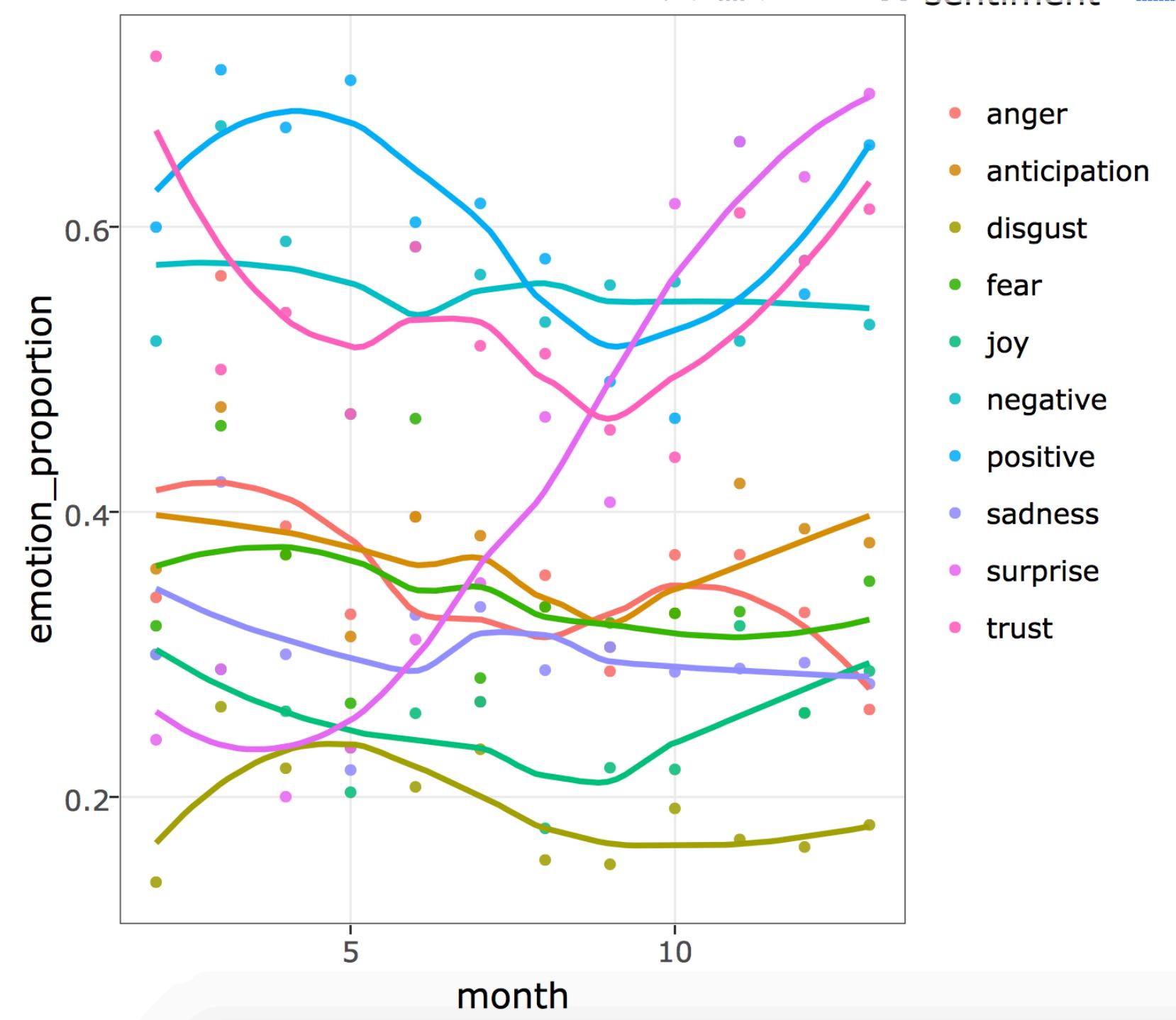
Titles of submissions

Text of the top comment  
per submission

# Emotions about politics in general

- Big correlations because raw counts, need to scale..



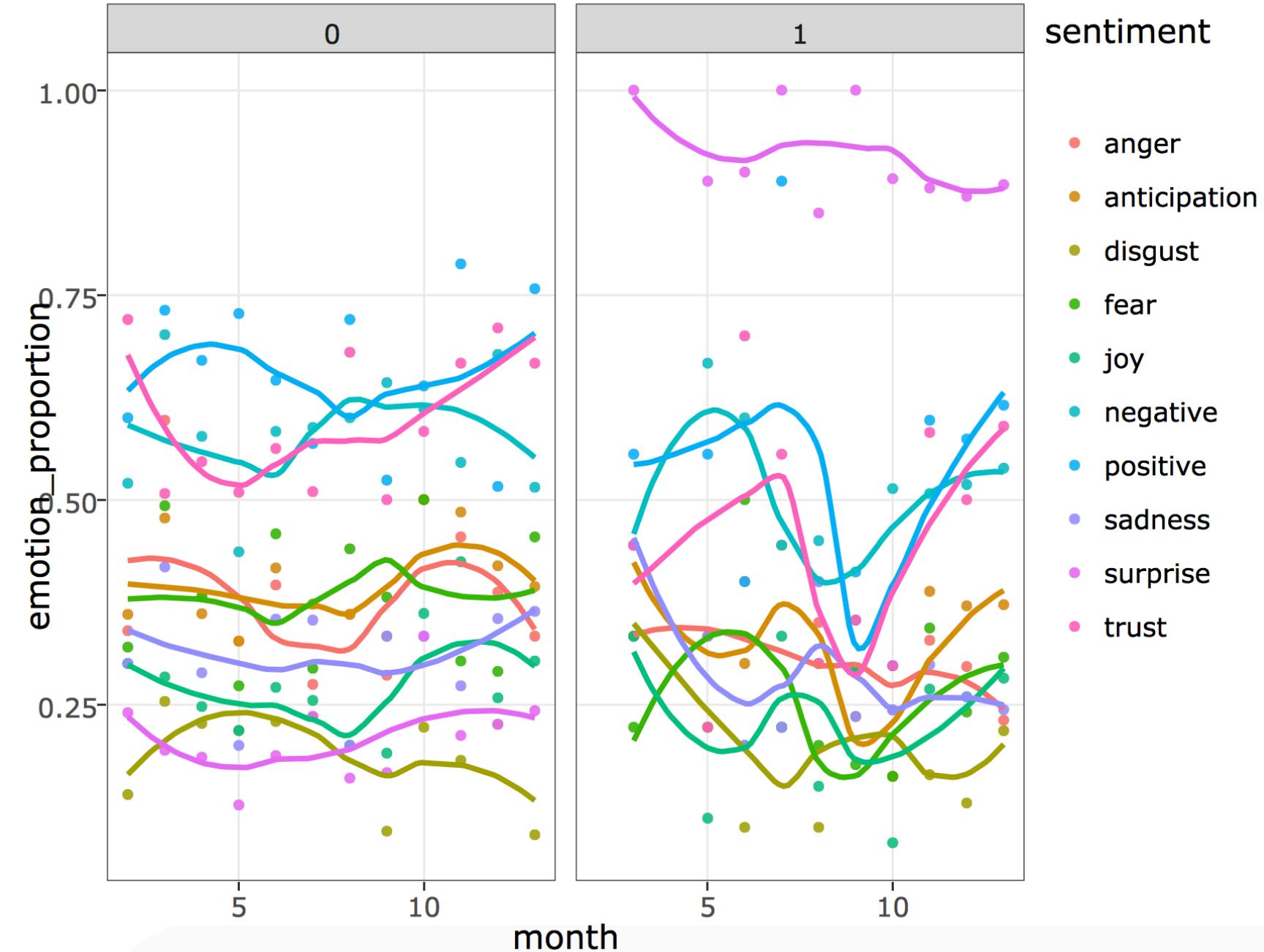


# “Surprise” is going up and up and up!

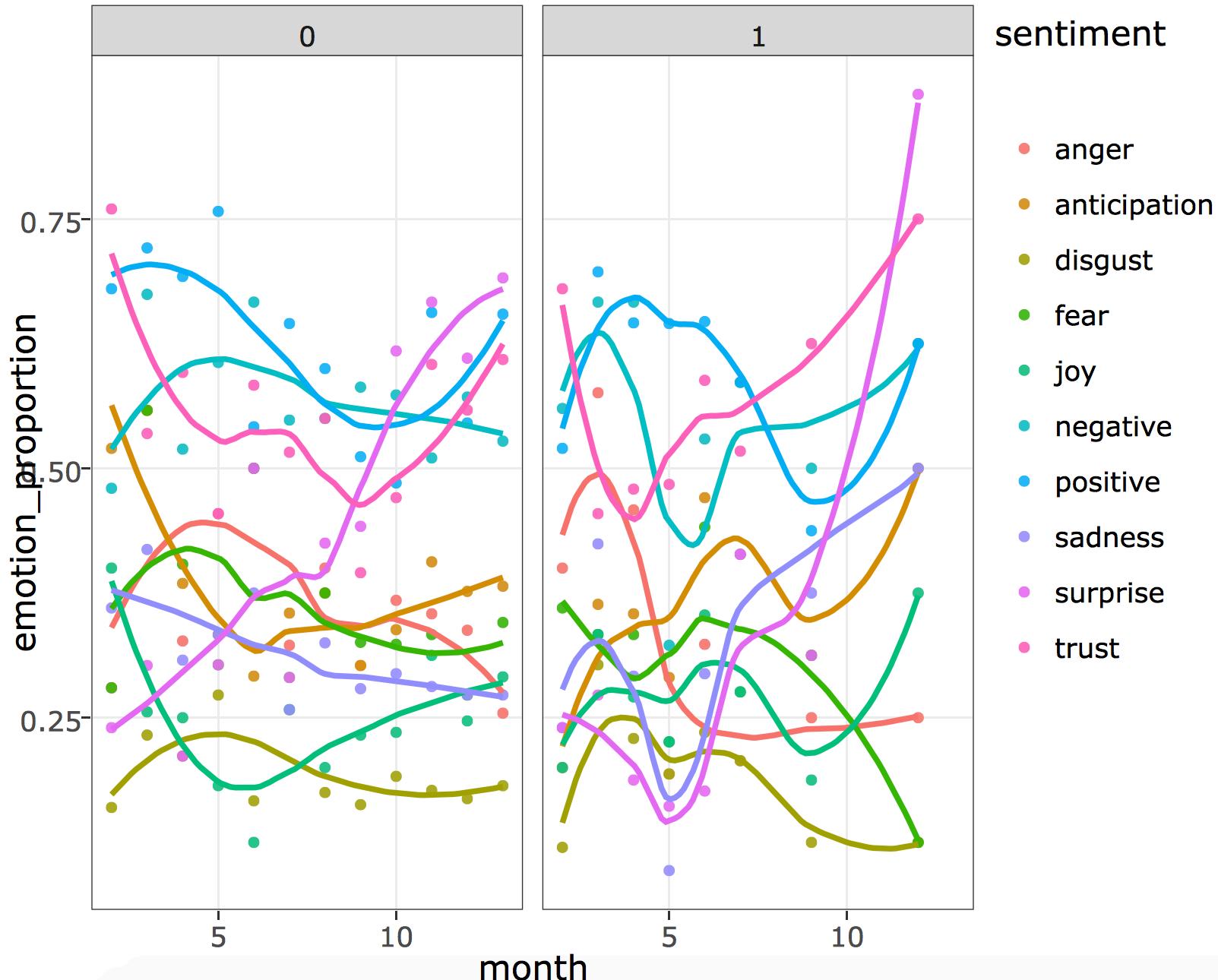
- Trust hit the lowest point at the month before the election
- Post-election, we are seeing a lot of fear (green)!
- Weirdly, post-election, trust and positivity are also going up?
- The year started off pretty positive and then nosedived, recovering now

# Emotional swings are due to The Donald

- Left panel (0) is posts not mentioning Donald, right panel mentions Donald
- The level of surprise is much higher in posts related to Trump
- The huge dips in trust, joy, happiness pre-election we saw before are largely due to Trump!



# Compared to Hillary..



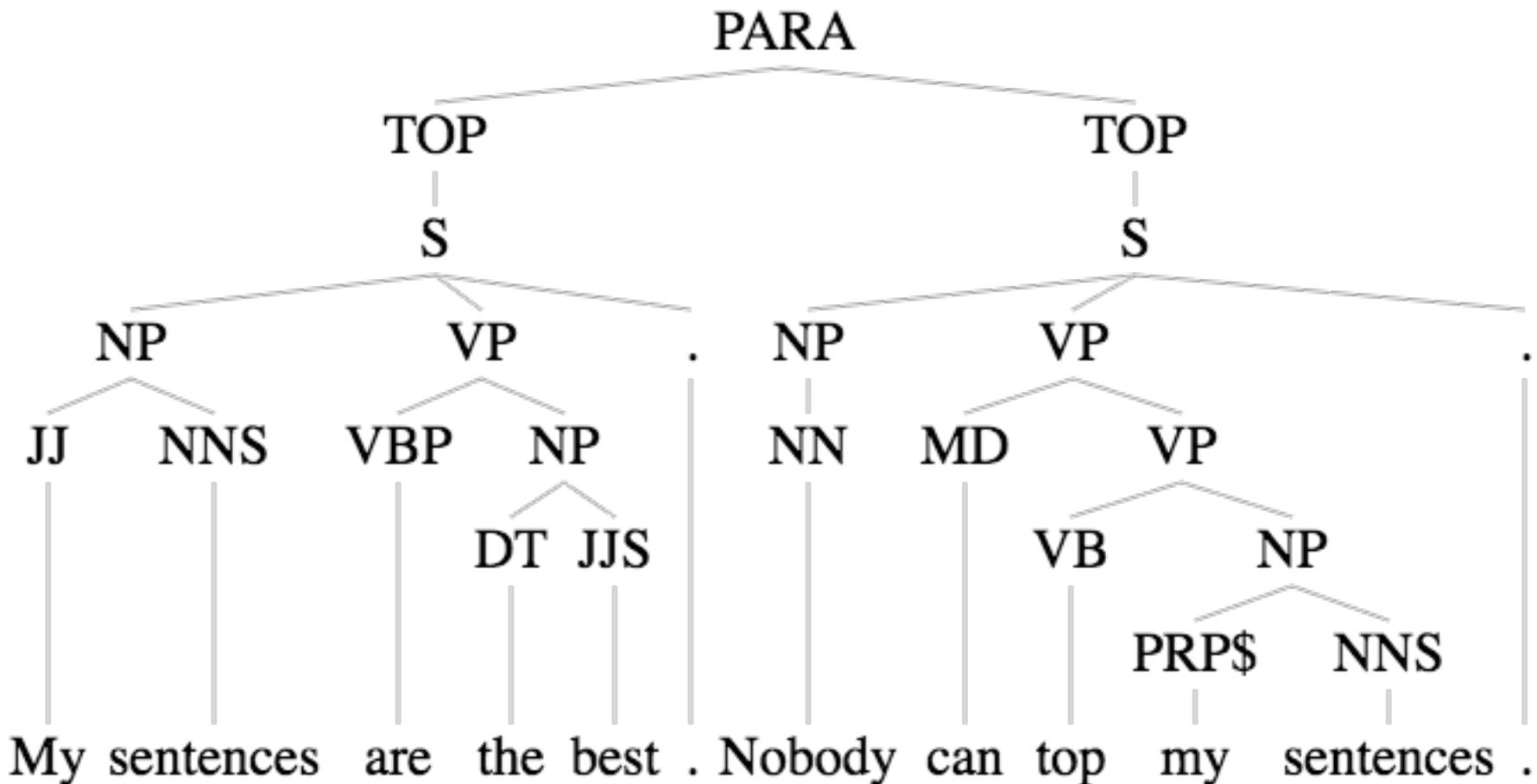
- Posts about Hillary Clinton (facet with '1') don't have such strong trends, mostly just follow the background levels of emotions about politics
- Main trend we do see here – anger/sadness/fear peaked right around the time that Donald was gaining lots of popularity (see plots 2)... related to the scandals, probably.

# Topic and sentiment detection with Azure

```
In [8]: # Azure portal URL.  
base_url = 'https://westus.api.cognitive.microsoft.com/'  
  
# Your account key goes here.  
account_key = 'ac455400c61d4c5792c1b4bc57b55715'  
  
headers = {'Content-Type': 'application/json', 'Ocp-Apim-Subscription-Key': account_key}  
  
input_texts = '{"documents":[' + df[['top_comment_body', 'title']].to_json() +']}'  
  
num_detect_langs = 1;  
  
# Detect key phrases.  
batch_keyphrase_url = base_url + 'text/analytics/v2.0/keyPhrases'  
req = urllib2.Request(batch_keyphrase_url, input_texts, headers)  
response = urllib2.urlopen(req)  
result = response.read()  
obj = json.loads(result)  
for keyphrase_analysis in obj['documents']:  
    kp=keyphrase_analysis['keyPhrases'][0]  
  
# Detect sentiment.  
batch_sentiment_url = base_url + 'text/analytics/v2.0/sentiment'  
req = urllib2.Request(batch_sentiment_url, input_texts, headers)  
response = urllib2.urlopen(req)  
result = response.read()  
obj = json.loads(result)  
for sentiment_analysis in obj['documents']:  
    print('Sentiment ' + ' score: ' + str(sentiment_analysis['score']))
```

- Convert pandas dataframe containing text of interest into a .json dictionary and pass as result to sentiment mapper as a url using urllib2
- Unfortunately I couldn't get this to work at scale. HTTP timeouts, too many requests sent?
- Still interesting for a few cases.
- Would be better if it could map multiple emotions instead of positive/negative.

# Future work



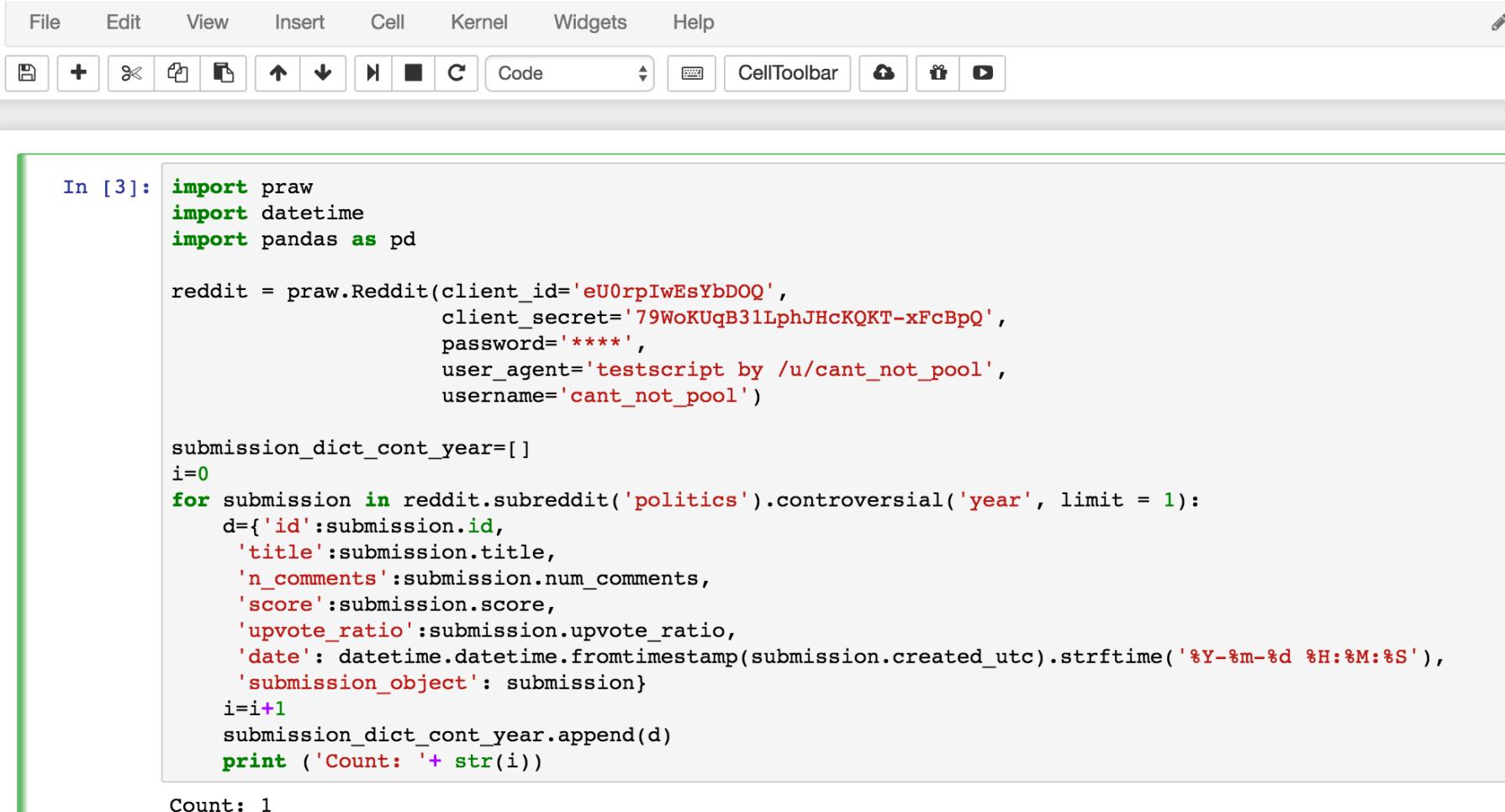
Microsoft syntactic structure detection... soon to be foiled by Trump's sentences?



# Appendix

## (Methods + Code)

# Mining reddit data



The screenshot shows a Jupyter Notebook interface. The top navigation bar includes File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. Below the toolbar are various icons for file operations like opening, saving, and running cells. The code cell (In [3]) contains Python code for mining reddit data. The output cell shows the result of the code execution.

```
In [3]: import praw
import datetime
import pandas as pd

reddit = praw.Reddit(client_id='eU0rpIwEsYbDOQ',
                     client_secret='79WoKUqB31LphJHcKQKT-xFcBpQ',
                     password='*****',
                     user_agent='testscript by /u/cant_not_pool',
                     username='cant_not_pool')

submission_dict_cont_year=[]
i=0
for submission in reddit.subreddit('politics').controversial('year', limit = 1):
    d={'id':submission.id,
       'title':submission.title,
       'n_comments':submission.num_comments,
       'score':submission.score,
       'upvote_ratio':submission.upvote_ratio,
       'date': datetime.datetime.fromtimestamp(submission.created_utc).strftime('%Y-%m-%d %H:%M:%S'),
       'submission_object': submission}
    i=i+1
    submission_dict_cont_year.append(d)
    print ('Count: '+ str(i))

Count: 1
```

- Used jupyter notebook + python to mine reddit data for /r/politics
- Can also collect the data live
- Can specify if you want “top”, or “controversial”
- Known issue – can’t get more than 1000 posts
- Rough/quick code at [https://github.com/nslatysheva/hack\\_cambidge\\_2017](https://github.com/nslatysheva/hack_cambidge_2017)

# Output is something like this

```
#df['top_comment_body'].apply(lambda x: x.encode('ascii', 'ignore'))  
df.to_csv('politics_controversial_1000.csv',encoding='utf-8')
```

```
df.head()
```

	date	id	n_comments	score	submission_object	title	upvote_ratio	top_comment_score	top_comment_gilded	top_comment_b
0	2016-11-07 11:19:09	5blmi4	12059	1554	5blmi4	I Was With Bernie Till the End; Now We All Mus...	0.50	10006.0	2.0	I'll vote for whoe the fuck I want\n\n**EDI...
1	2016-08-24 16:02:42	4zd07t	2081	0	4zd07t	My Writing in The Huffington Post, Salon, and ...	0.48	765.0	0.0	Thank you for doi this AMA! \n\nI'd to ...
2	2016-10-13 19:36:08	57buyp	537	52	57buyp	Hi Reddit, I am Maria Teresa Kumar, Emmy-nomin...	0.50	47.0	0.0	Are you surprised the record low pu tru...

- Data dictionary then converted to dataframe and exported into R

- Just because I am most comfortable in doing analysis with R

# R analysis took a while

The screenshot shows the RStudio interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Window, Help, and a status bar showing Sun 11:15. Below the menu is a toolbar with various icons. The main workspace has several tabs open: generate\_player\_transitions.R, stats.R, multitissue\_boxplots.R, trump\_tweets.R, more\_emotions, and test.R. The code editor (left pane) contains R script code for data manipulation and plotting. The Environment pane (right pane) lists global variables like links2, more\_emot..., and nrc. A plot pane (bottom right) displays a line chart titled "sentiment" showing "emotion\_proportion" over time ("month"). The plot includes four data series: anger (pink), anticipation (orange), disgust (purple), and fear (green).

```
52 # proportion of posts with the donald
53 # break down by month
54 head(top_year)
55 proportion_mentions <- top_year %>%
56   mutate(year = year(date),
57     month = month(date),
58     month = ifelse(year == 2017, 13, month)) %>%
59   group_by(year, month) %>%
60   summarise(prop_donald = sum(ifelse(mentions_donald_trump == "1", 1, 0))/n(),
61             prop_hillary = sum(ifelse(mentions_hillary_clinton == "1", 1, 0))/n(),
62             prop_bernie = sum(ifelse(mentions_bernie_sanders == "1", 1, 0))/n(),
63             count = n()) %>%
64   gather(key = politician, value = proportion, -c(year, month, count))
65
66 summary(proportion_mentions$month)
67 head(proportion_mentions)
68
69 ##### PLOT 2
70 ggplot(data = proportion_mentions, aes(x=month, y = proportion, colour=politician)) +
71   geom_point() + geom_smooth() + scale_color_manual(values = c("blue", "pink", "red"))
72 ggplotly()
73
74
```

68:1 (Top Level) ▾

Console R Markdown x

~/Downloads/polnet2016/ ↵

```
bernie sander wants raw vote count released after tight finish in iowa caucus
the clinton system: "clinton foundation data shows that during her term the state department authorize
$165 billion in commercial arms sales to twenty nations that had given money to the clinton foundation."
hillary makes a minimum of 225,000 for a speech, but she thinks a 15 dollar minimum wage is to high. well hillary it would
take a minimum wage worker 15 years to make as much money as you do for a one hour speech to goldman sachs
Female sander backers slam 'insulting' clinton supporters who say they're betraying their gender
score mentions_donald_trump mentions_hillary_clinton mentions_bernie_sanders year month
```

L 11883 0 1 0 2016 1

- Using Rstudio for R scripting
- Tons of data manipulation to do for every plot basically
- Visualisation with ggplot2, interactivity with ggplotly

# Topic and sentiment detection with Azure

```
In [8]: # Azure portal URL.  
base_url = 'https://westus.api.cognitive.microsoft.com/'  
  
# Your account key goes here.  
account_key = 'ac455400c61d4c5792c1b4bc57b55715'  
  
headers = {'Content-Type': 'application/json', 'Ocp-Apim-Subscription-Key': account_key}  
  
input_texts = '{"documents":[' + df[['top_comment_body', 'title']].to_json() +']}'  
  
num_detect_langs = 1;  
  
# Detect key phrases.  
batch_keyphrase_url = base_url + 'text/analytics/v2.0/keyPhrases'  
req = urllib2.Request(batch_keyphrase_url, input_texts, headers)  
response = urllib2.urlopen(req)  
result = response.read()  
obj = json.loads(result)  
for keyphrase_analysis in obj['documents']:  
    kp=keyphrase_analysis['keyPhrases'][0]  
  
# Detect sentiment.  
batch_sentiment_url = base_url + 'text/analytics/v2.0/sentiment'  
req = urllib2.Request(batch_sentiment_url, input_texts, headers)  
response = urllib2.urlopen(req)  
result = response.read()  
obj = json.loads(result)  
for sentiment_analysis in obj['documents']:  
    print('Sentiment ' + ' score: ' + str(sentiment_analysis['score']))
```

- Convert pandas dataframe containing text of interest into a .json dictionary and pass as result to sentiment mapper as a url using urllib2
- Unfortunately I couldn't get this to work at scale. HTTP timeouts, too many requests sent?
- Still interesting for a few cases.
- Would be better if it could map multiple emotions instead of positive/negative.

# Failed attempt: Pusher with Reddit API stream

```
Natashas-MBP:hack_cambidge_2017 nat$ node pusher  
/Users/nat/Projects/random_things/hack_cambidge_  
var app = new PusherPlatform.App({  
    ^  
ReferenceError: PusherPlatform is not defined  
    at Object.<anonymous> (/Users/nat/Projects/ra  
    at Module._compile (module.js:570:32)  
    at Object.Module._extensions..js (module.js:5  
    at Module.load (module.js:487:32)  
    at tryModuleLoad (module.js:446:12)  
    at Function.Module._load (module.js:438:3)  
    at Module.runMain (module.js:604:10)  
    at run (bootstrap_node.js:394:7)  
    at startup (bootstrap_node.js:149:9)  
    at bootstrap_node.js:509:3
```

```
1  var app = new PusherPlatform.App({  
2      appId: 'cbf29c93-840e-421c-af69-856cfcba01b5',  
3  });  
4  
5  var myFeed = app.feed('playground');  
6  
7  myFeed.subscribe({  
8      onOpen: () => console.log('Connection established'),  
9      onItem: item => console.log('Item:', item),  
10     onError: error => console.error('Error:', error),  
11  });  
12  
13  
14  myFeed.append('Hello, world!')  
15      .then(response => console.log('Success:', response))  
16      .catch(err => console.error('Error:', err));  
17  
18 // You're not limited to appending string values;  
19 // you can also append objects, arrays and numbers.  
20 myFeed.append({ yourKey: 'your value' })  
21     .then(response => console.log('Success:', response))  
22     .catch(err => console.error('Error:', err));  
23
```

- Main problem: I don't know javascript (???)
- Installed and tried to use node.js
- Error codes a mystery
- Not enough time
- Decided to just stick to Python and R and data ;P