The Retweet Propagation of News

Jonathan Campbell

Contents

1 Abstract

With the eruption of social media, there has been a large amount of social research done to look into and analyze the data created through the interaction of users with various types of media. Not only has this been an area of interest in academic circles, corporations have been adding to and taking advantage of these analyses to monetize these interactions though the use of viral marketing. In both respects, many have attempted to apply various regression models and machine learning classification algorithms in order to find predictive indicators for user influence, trend forecasting, and information diffusion. Additionally, the applications of these models go well beyond a commercial setting but into news media as well. With the decline of print journalism, news moved towards online media where it integrated social media into the dissemination of breaking stories. Thus, it is interesting to see how we can use these types of models in order to describe the consumption of news. Therefore, area of interest in this paper will be information diffusion: the transmission of information across a digital social network through the direct actions of its component users. More specifically, I will be analyzing the propensity of news media to be dispersed across Twitter though the retweeting of news organizations' tweets. While most research up to this point has analyzed the propensity of individual users to be retweeted or how the content of a tweet affects its propensity to be retweeted, this paper will focus on using a continuous time survival analysis model to describe both the total reach and temporal cycle of news. This analysis will show the difference between news organizations and how the survival rates themselves can be used to classify the network's interpretation of news.

2 Introduction

"I realize I am inviting blowback from passionate Tweeters, from aging academics who stoke their charisma by overpraising every novelty and from colleagues at The Times who are refining a social-media strategy to expand the reach of our journalism. So let me be clear that Twitter is a brilliant device, a megaphone for promotion, a seine for information, a helpful organizing tool for everything from dog-lover meet-ups to revolutions. It restores serendipity to the flow of information." **- Bill Keller**

In recent years we have seen most major news organizations include social media commentary, most specifically during political events and breaking news, when information regarding events may be scarce. Additionally, there is a general mood that traditional print journalism, including its online print counterpart, is dying. However, media organization's use of social media is not limited to creating news and making it more immersive. Social media sites are also used to promote traditional news stories in the daily news cycle and stimulate virtual discourse. We have the unique capacity, through the use of social media data, to track this flow of news consumption: how it spreads, evolves, and trends: how discussion of current events either progresses on or lingers on important issues. The goal is to analyze consumer consumption of news assets via social media promotion.

Answering this question requires acquiring social media data connected with large national news organizations. The main methods of social promotions tends to be three fold: direct promotion to news outlet subscribers (either through app notifications or email digests), Facebook, and Twitter. Acquiring data for the first method is extremely difficult. This would require gaining access to internal user databases from different news organizations, and this is information that companies do not easily part with. The next method of social promotion, Facebook, fairs slightly better. Through the use of the Facebook API, we can gain access to information regarding stories publicly promoted on news pages. However, this method also has a considerable drawback. Without security permissions to the individual pages,

the data pulls are limited to aggregate overviews of posts. While this information would give insight into the total reach of certain stories, we have no way to analyze the aspects of growth. Additionally, we will not have information regarding the people consuming this information, limiting the analysis further. The final method of social promotion, Twitter, seems like the most optimal choice among the three. Due to Twitter's public nature, we are provided a wider range of information, such as: full original posting, user identifiers, full post and comment text, geographic location, and event based time stamps.

Twitter has been shown to help organize individuals, provide instantaneous information to people in disaster zones, and allows users all over the world to contribute to a global conversation about current events. While, there are pleanty of tweets out there containing the random thoughts of bored adolescents and angry customer service compaints, Twitter, through its inate design, is a medium for sharing news. My thesis seeks to examine the ways in which news is shared and then spread on social media. Having chosen Twitter as a data source due to access and data quality, the primary aim of this examination will be to study the social propagation of news though the general analysis of the retweeting of news.

3 Literature Review

The news media's use of Twitter has been discussed by various authors since it was integrated into the daily reporting of events. Alex Bruns and Jean Burgess in their article, "Researching News Discussion on Twitter," cover various aspects of it's use and analysis. At it's very base, we can consider Twitter to interact with news in three particular ways. First, Twitter is used for the spread of information as instantaneously as it occurs. "Such activities no include not only the reporting of events by actual eyewitnesses on the ground . . . , but also second-hand live discussion of unfolding events as they are covered by other media" (Bruns & Burgess 2012). In this respect, the individual is thrown directly into the news cycle by participating in the conversation from its outset. According to the authors, an added driver here is the ease

at which different pieces of media can be uploaded and shared in the form of links, photos, and videos, extending the impact of a tweet beyond its allowed 140 characters. The second way in which Twitter interacts with the creation and spread of news is to instantly evaluate how newsworthy an event is. Bruns describes this interactions as *gatewatching*: "highlighting, sharing, and evaluating relevant material released by other sources, in order to develop a more comprehensive understanding" (Bruns & Burgess 2012). In other words, Twitter will decide what is news and who the primary source should be by promoting and sharing content. Lastly, "Twitter's new coverage also consists of significant amounts of broader commentary on current events, reflecting mainly the senders' own perspectives and intended more as markers of those perspectives than as formal contributions to debate" (Bruns & Burgess 2012). Through discussing news topics in aggregate, users can shift the focus of the entire network to highlight to bring it to the forefront of public attention.

The ability of Twitter to function in this way lies mainly in the way it allows users to interact with one another. To begin with, Twitter uses a directed network structure. Users' follow others to receive updates and information updates, but this information only flows one way unless the other user decides to follow in return. Beyond that, there are three special ways in which one can send out updates to their followers: retweets, mentions, and hashtags. Retweets forward information directly from another used to all of the users within your own network. Mentions allow a user to direct a comment to a specific user who may not be one of their followers; this message will be able to be seen by all of they user's followers. Lastly, hashtags can be used to contribute to a larger conversation outside one's network. "To the extent that Twitter users consciously understand this network structure, their responses to newsworthy events address and interact with their own immediate community of followers, and may also attempt to overcome the barriers dividing specific clusters in the network." (Bruns & Burgess 2012). The authors here are implying that by retweeting, you explicitly are seeking to both make your users aware of certain information and extend the informational reach of the tweet. Using a hashtag is a deliberate effort to engage in the conversation of the

larger Twitter community.

Through the knowledge of the way that Twitter works we can attempt to identify key participants in discussions in order to establish a community structure. "In doing so, it is usually less important to examine the total number of tweets *sent* by each user, but rather to focus on the number of responses and retweets *received*." (Bruns & Burgess 2012). Intuitively, we know that the initial reach of a tweet is a functions of the size of the network. When looking at Twitter as a whole, according to Kwak et al. (2010) in their analysis of a crawl of the entire Twitter network encompassing 2009, the number of followers that a random user has follows a power-law distribution e.g. there are a disproportionally small section of users that have a disproportionately high number of followers. Unsurprisingly, they found that the users with the larges follower bases were either celebrities or accounts associated with mass media companies. These accounts were also generally associated with high levels of retweets, and according to them, "[t]he number of retweets for a certain tweet is a measure of that tweet's popularity and in turn of the tweet writer's popularity" (Kwak et al. 2010).

Garnering many retweets is how topics may begin to trend in Twitter. This is partly how Twitter functions as a determinant for the newsworthiness of a topic, as mentioned earlier, but how do media organizations compare? "When comparing trending topics on Twitter to that of CNN headlines, more than half the time CNN was ahead in reporting" (Kwak et al. 2010). Thus, despite Twitter's ability to identify trends, news media in most instances serves as key conveyer of information. This information almost seems to imply that media companies could hold a trending topic in place for an extended period of time or steer conversation forever away from certain topics; however, "[a] trending topic does not last forever nor dies never to come back" (Kwak et al. 2010). The majority of active trends are shorter than a week, with close to a third being only a day or less, but there are close to 7% that lasted longer than 10 days. Upon inspecting trends further, Kwak et al. looked into the marginal effect of a users retweet and was able to find something slightly surprising: "Up to about 1,000 followers, the average number of additional recipients is not affected by

the number of followers of the tweet source. That is, no matter how many followers a user has, the tweet is likely to reach a certain number of audience, once the user's tweet starts spreading via retweets" (Kwak et al. 2010). The authors suggest that once again reinforces the use of Twitter as a barometer for the newsworthiness of a topic as the decision of each user to retweet effectively determines how newsworthy the originating tweet was. The authors go a bit far in suggesting that, "[i]n a way, we are witnessing the emergence of collective intelligence" (Kwak et al. 2010).

Meeyoung Cha and her coauthors for, "Measuring User Influence in Twitter: The Million Follower Fallacy," explore the idea of user influence in Twitter just a bit further. They analyzed three different, yet common identifications of influence in Twitter: network indegree (the number of followers that a users has), the number of retweets that they get, and mentions (the number of times that any user tries to direct their attention towards their own tweet). "Analysis of the tree influence measures provides a better understanding of the different roles users play in social media. Indegree represents popularity of a user; retweets represent the content value of one's tweets; and mentions represent the name value of a user" (Cha et al. 2010). Fairly quickly, the author were able to discount indegree as a signification measure of influence. Previously coined as *the million dollar fallacy* due to anecdotal evidence regarding the etiquette of following someone who is following you, but they were able to find that the prior rate of garnering retweets was a far more accurate predictor than the total number of users. We can also look to the evidence provided by Kwak et al. regarding the relationship between indegree and retweet propensity to further validate this claim. With indegree mostly invalidated as a measure of influence, Cha et al. turned to a traditional communications theory which states that there exists certain users, *influentials*, who excel in the persuasion of others. However, there also exists a more modern theory called *collaborative filtering*, which, "de-emphasizes the role of influentials. Instead, it posits that the key factors determining influence are (i) the interpersonal relationship among ordinary users and (ii) the readiness of a society to adopt an innovation" (Chat et al. 2010). What collaborative filtering implies is

that people are more likely to follow the advice of their peers that the influentials. Under the idea of collaborative filtering, "a trend can be initiated by any one, and if the environment is right, it will spread" (Cha et al. 2010). These two theories were hard to be fairly weighed against one another as there is no universal definition for influence. It could, for instance, be defined as the effect of an individual in the dispersion of news.

Cha and her associates analyzed the different measures of influence themselves, using their own crawl of the Twitter network. After calculating the different influence measures for all users, they then used Spearman's rank correlation coefficient

$$\rho = 1 - \frac{6 \sum (x_i - y_i)^2}{N^3 - N}$$

as a measure of the correlation between the pairs for indegree, mentions, and retweets. "Spearman's rank correlation coefficient calculation is a nonparametric test; the coefficient assesses how well an arbitrary monotonic function could describe the relationship between two variables, without making any other assumptions about the particular nature of the relationship between the variables" (Cha et al. 2010). Very quickly, they were able to find that among the three measures of influence only mentions and retweets seemed to share a correlation. However, when looking at the most highly ranked profiles in each of these categories, there was very little overlap, indicating different values of influence:

1. Indegree influence, the number of followers of a user, directly indicates the size of the audience for that user.

2. Retweet influence, which we measure through the number of retweets containing one's name, indicates the ability of that user to generate content with pass-along value.

3. Mention influence, which we measure through the number of mentions containing one's name, indicates the ability of that user to engage others in a conversation.

- (Cha et al. 2010)

We can note the slight change in the scope with which each influence measure was slightly changed to more accurately portray how each of these measures relates to the real world. The most significant change lies in the implication in the change in the definition of retweet influence. "[R]etweets represent influence of a user beyond one's one-to-one interaction domain; popular tweets could propagate multiple hops away from the source before they are retweeted throughout the network" (Cha et al. 2010). According to their results, the most retweeted users were content aggregation services, businessmen, and news sites. The most mentioned users tended to be celebrities as, "[o]rdinary users showed a great passion for celebrities, regularly posting messages to them or mentioning them, without necessarily retweeting their posts" (Cha et al. 2010).

Lastly, the authors looked to see how different factors might affect the influence of users. First, they evaluated the difference in influence across three different but largely encompassing events:

- The Iranian Presidential Election
- The Outbreak of H1N1 (Swine Flu) &
- The Death of Michael Jackson

Each of these three topics reached over 40% of users on Twitter. Despite the far reach of these events, fewer than 2% of users discussed all three topics. Unsurprisingly, authoritative news sources were able to maintain popularity across all three topics. Cha et al. made two intermediary conclusions: (1) the users with the most influence are trusted over a wide range of topics and (2) it is more effective to target high-level influential than collaborative filtering in order to spread information. Second, the authors looked at how the levels of influence

changed over time for different types of users. For established influential, their levels of retweeting, on average grew throughout 2009; however, mentions decreased for the most influential users and increased for more ordinary influentials. Indicating that news media is more focused to the influential in disseminating information, but more ordinary users are more influential in engaging users in a conversation.

With more evidence supporting the influential model of information dispersion, we then want to know how information actually disperses beyond one's personal social network. Kristina Lerman and Rumi Ghosh studied exactly that in their analysis of news contagion in the Digg and Twitter Social networks in their article, "Information Contagion: An Empirical Study of the Spread of News on Digg and Twitter Social Networks." Digg and Twitter, are both social networks that allow for the dispersion of news within and across their networks. Digg is a news aggregator that allows each of its 3 million registered users to submit links to news stories and *Digg* them (e.g. vote on stories to increase their importance). While the exact method is unknown, it is assumed that they prioritize stories based on how many of their users *digg* their stories and how quickly they *digg* them. Lerman and Ghosh collected their Twitter data from Tweetmeme, a cite that aggregates the most retweeted posts over all of Twitter. The data for Digg and Twitter are very similar when we think of retweeting as the act of propagating news further into the network. Additionally, both of the cites are formed of a directed network, where a user can follow the actions of another they deem to be interesting or influential. One difference, however, is that, "the Digg social network is denser, more tightly knit than the Twitter social network" (Lerman & Ghosh 2010).

From their analysis, they were able to study a similar trend, which they termed as a *cascade*. "[T]he cascade ("information contagion" in the title of this article) starts with story's submitter and grows as the story accrues fan votes" (Lerman & Ghosh 2010). These information cascades are then used to create a model of the dynamics of information on networks. "The mechanism for the spread of information is as follows: "users watch their friends' activities — what they tweet or vote for — and by their own tweeting and voting

actions they make this information visible to their own fans or followers" (Lerman & Ghosh 2010). At the end of the study, there ended up being substantive quantitative differences in the structure and function of the Digg and Twitter networks. As stated earlier, the Digg network is much more dense and interconnected that that of the Twitter network."A story posted on Digg initially spreads quickly through the network, with users who are following the submitter also likely to follow other voters. After the story is promoted to Digg's front page, however, it is exposed to a large number of unconnected users. The spread of the story on the network slows significantly, though the story may still generate a large response from Digg audience" (Lerman & Ghosh 2010). This mechanism implies that once a story reaches the front page of Digg, users no longer will have the incentive to promote the story further. The less dense network of Twitter functions in a different way. "[S]tories spread through the network slower than Digg stories do initially, but they continue spreading at this rate as the story ages and generally penetrate the network farther than Digg stories" (Lerman & Ghosh 2010). It seems as though there is a higher satisfaction threshold for Twitter users in their decisions to retweet. Through the study of these cascades, the authors were able to gain more information about exactly how information is being dispersed throughout each respective network.

Another method of analyzing the dispersion of information in a network is to analyze whether particular types of information are more likely to be dispersed farther than others. Nasir Naveed and his coauthors applied a series of text analysis logistic regressions in order to find factors that affected the probability of tweet to get retweeted. Taking a different view from Lerman & Gosh, "a network-based analysis of retweets may give hints into *who* tends to write interesting messages, but cannot give insights into *what* the community is interested in" (Naveed et al. 2011). According to the authors, previous researcher have been mostly focused on how a user's position in a network (e.g. their general level of influence, closeness within his immediate network, and his position within the larger network as a whole) and not on the particular aspects of the tweets themselves. The authors posit that the tendency

of a particular user to retweet is more closely based on how interesting they find the tweet in question that from whom the tweet comes from. It must be interesting **enough** to relay onto their followers. "[W]hether a particular tweet actually is retweeted depends heavily on context, such as the user's position in the social graph or the time of day the tweet is posted" (Naveed et al. 2011) as the tweet must reach those followers that find it interesting.

Naveed and his associates looked at a variety of features: direct messages, URLs, usernames and hashtags, exclamation and question marks, positive and negative terms, emoticons, sentiments, and word selection. In order to get results indicative of probability, they ran a series of logistic regressions in order to obtain weights (w_i) under the equation model:

$$P(retweet_j|f) = \frac{1}{1 + e^{-(w_0 + \sum_i w_i f_{ij})}}.$$

The resulting weights showed that direct messages are very unlikely to be retweeted; an intuitive results as conversational exchanges would be unlikely to be universally appealing. Messages with hashtags, usernames, and URLs increase the odds of being retweeted; however, these factors, in and of themselves, aren't enough in order to substantially increase the likelihood of being retweeted without being combined with other factors. Punctuation turned out to be a significant factor as well; where exclamation points decreased the likelihood of being retweeted and question marks increased the likelihood. Lastly, it seems as though word choice and sentiment were predictors as well. Positive sentiments or topics concerning everyday live decreased the likelihood; whereas, negative sentiments or topics concerning events and updates increased the likelihood. Overall, when including all of the factors in making a prediction, the likelihoods produces accurate results. "As a general rule, a tweet is likely to be retweeted when it is about a general, public topic instead of a narrow personal topic. For instance, a tweet is unlikely to be retweeted when it is addressed to another Twitter user directly, while our topic analysis revealed that general topics affecting many users like

social media or Christmas are more likely to be retweeted. This can be understood as the Twitter platform being better suited as a news and announcement channel rather than a personal communication platform" (Naveed et al. 2011).

While Naveed was successful in his creation of a probabilistic model to predict retweeting tendancies, researcher's Riley Crane and Didier Sornette in their article, "Robust dynamic classes revealed by measuring the response function of a social system," looked at social sharing as Poisson process, analyzing the number of times an event is likely to occur in a given time interval. Crane and Sornette looked at an eight-month span of data regarding daily human activity on YouTube. According to the authors, "uncovering rules governing collective human behavior is a difficult task because of the myriad of factors that influence an individual's decision to take action. Investigations into the timing of individual activity, as a basis for understanding more complex collective behavior, have reported statistical evidence that human actions range from random to highly correlated" (Crane & Sornette 2008). This task is very similar to the task faced by the previously mentioned researchers: trying to identify all of the factors (influence measure, network structure, tweet composition, etc) that are influencing a user to retweet. However, if you could identify a collective model of the timing of behavior; we could explain the average tendency of an individual to retweet over time. Crane and Sornette begin by describing the ways in which a social action might occur: "At the simplest level, viewing activity can occur one of three ways: randomly, exogenously (when a video is featured, or endogenously (when a video is shared)" (Crane & Sornette 2008). The authors then go on to define a two part model of human interaction. The first component is a power law distribution of waiting time (e.g. the time it takes for a person to view a video). The second component is an epidemic branching process to describe the diffusion of influence throughout a social network; similar to Lerman & Ghosh's information cascades. "This process captures how previous attention from one individual can spread to others and become the cause that triggers their future attention" (Crane & Sornette 2008). This epidemic process can be modeled using the self0excited Hawkes conditional Poisson

process:

$$\lambda(t) = V(t) + \sum_{i,t_i \leq t} \mu_i \phi(t - t_i)$$

According to the Crane and Sornette, using all of these considerations, they were able to come up with four models for viewership: exogenous subcritical, exogenous critical, endogenous subcritical, and endogenous critical. Here critical and subcritical refer to the ability of individuals to influence others into action or not respectively, and exo/endogeneity refers to whether the influence comes from a function of the system or a function of your network. Upon fitting these models to the data, they found that close to 90% of video dynamics, "either do not experience much activity or can be described statistically as a Poisson process ([as] verified using a Chi-Squared test)" (Crane & Sornette 2008). The fit of these models show that general epidemic models can be used to approximate average tendencies of human behavior within a social network when combined with information cascades. Their study was found two other interesting conclusions. First, the more sensitive a network is to an external shock, the faster the network will identify the event, but also the longer the event will linger around the network. Second, one could, "natural[ly] suggest a qualitative labeling that is quantitatively consistent with the three classes derived from the model: Viral videos, quality videos, and junk videos" (Crane & Sornette). In the case of twitter, these classifications could be such as: Breaking news, persistent news, and regular news. This method of classification would be highly robust as it would not rely on qualitative judgement, using only information revealed by the dynamics of human activity as its indicator.

Amit Goyal, Francesco Bonchi, and Laks V. S. Lakshmanan implemented a similar model regarding influence propagation in social networks in their article, "Learning Influence Probabilities in Social Networks." According to the authors, most viral marketers assume that they are going to have access to a complete social graph labeled with conditional influence probabilities, but there is no determination as to how these probabilities are to be calculated.

If we incorporated time into our influence model, we could use a general threshold model. "At a given timestamp, each node is either active or inactive, and each node's tendency to become active increases monotonically as more of its neighbors become active" (Goyal et al. 2010). When applying a temporal element, this becomes a model of how long until all of the nodes *activate*. They also found that buy using a continuous time model, as opposed to a discrete time model, for besides a more robust prediction, the model also allows to predict the time at which a user is most likely to perform the action.

The works of previous researchers has shown that within Twitter, there are several measures of influence (or of information dispersion). News networks, specifically, are broadly considered one of the most influential of actors within the network, and their influence is best measured through the tracking an analysis of retweets. Additionally, among the different types of influence analysis, using a generalized probability model in conjuncture with a continuous time distribution should be able to produce robust estimates for the behavior of individuals within a given network.

4 Data

4.1 Data Collection

In turn, in order to examine this aim, there will be two steps of data collection: collecting the shared news stories of different news organizations and collecting the retweet trees for the individual tweets from each organization. In order to accomplish this, I will collect all of the most recent tweets from each news organization's timeline. My first aim is to identify how original content is spread; therefore, I will set aside all posts that are retweets themselves or relies to other posts. At this point, I will cycle through each tweet and collect all information from each tweet's retweets and so on until an end to each branch of the tree is reached.

Thus far in researching data collection methods, I have looked into various ways of

accessing the Twitter API. The methods that I have been able to use have utilized packages in R that were created for this purpose; however, I have discovered that these packages have a few limitations. These limitations are ones that originate in the limits set in place by the Twitter API. "Changes made late in 2010 mean that even for the purposes of publicly funded, non-commercial research, it is no longer possible to gain access to the full "firehose" of all tweets, or to substantial subsets of this full feed" (Bruns & Burgess 2012). Therefore, each query's return is limited to a set number of records and each user is limited to a set number of queries per day. The more challenging of these two limits is the record limit per query. One of the major pitfalls of using packages is the inability to pick a query up after the record limit has been reached, making the collection or records time sensitive. It has become evident that It will be best to accomplish this data collection through connecting to the Twitter API though Python. The goals for the script are to: 1) make the code well defined enough such that new data pulls can be done simply by running the same script without modification, 2) prevent data loss when encountering data limits, 3) maximizing the amount of information retrieved from each pull, and 4) construct it such that information can be gathered multiple hops down a network graph.

The implementation of the data collection script uses just a few modules. [datetime] was used to create time differences in seconds. The [sleep] function from the [time] module was used in order delay the next query to the API until the time limit was reached. The [tweepy] module was used to create the secure connection to the API, and make each functional call. Lastly, the [numpy] and [pandas] modules were used in order to reformat the incoming data into a data frame so that it could be exported as a `.csv` file.

The first section of the code, despite being relatively short, it is the most crucial component which allowed for the data collection. While it initially seems as there is an overall limit to the number of queries per period of time, the limits are actually imposed based on the type of query involved. Because this was a method that needed to be tracked in various places in the script, I defined it as a class so that each limit, with its varying limit

values, could be tracked concisely and efficiently.

With the rate limit problem taken care of, the remainder of the code handles interacting with the Twitter API in order to pull the necessary data. After setting up an OAuth connection, the user just needs to enter the name of the user it wishes to collect data on. The script will then pull the maximum amount of information from the initial timeline, and the recursively iterate through all of the tweets in order to obtain all recognized retweets available.

4.2 Case Selection

With the data collection set up, our concerns can then move to choosing cases of analysis. Because of the large time intensive commitment towards each data-pull, I have decided to select three different major news organizations based on differences in market space: CBS News, Yahoo! News, and BBC News. I have chosen these three different news organizations due to my belief that (1) there won't be a substantial difference in the networks and (2) they each represent a slightly different marketshare: CBS represents the typical American media corporation, Yahoo! represents digital technology, and BBC represents the international news market.

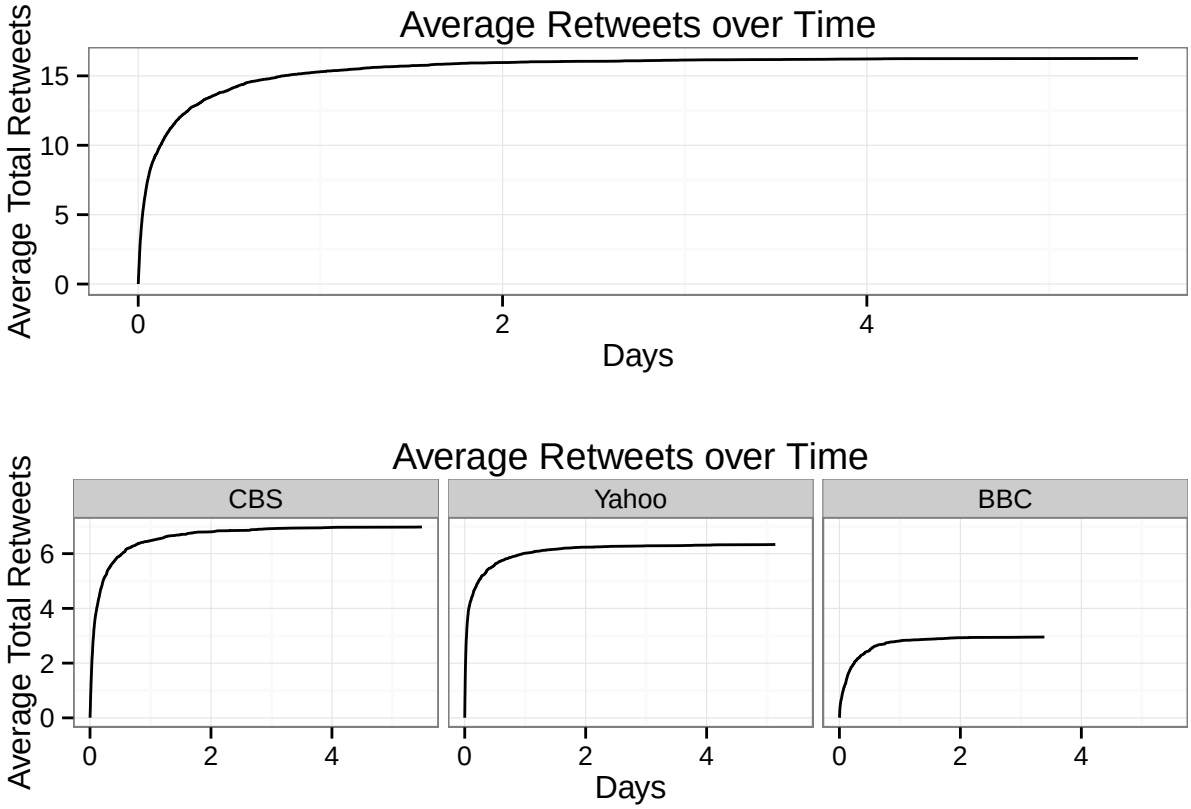
4.3 Data Import and Exploration

After collected a sufficient amount of retweet trees, we can then take our analysis further and begin applying other methods of data transformation and analysis. With all of the information that I am able to collect, there are certain transformations necessary to gain greater insight. First and foremost, we must transform the data into a continuous time series. To do that I've split the text formatted date into sections, and aggregated time into total number of seconds. With the aggregated time calculated, our variables of interest are:

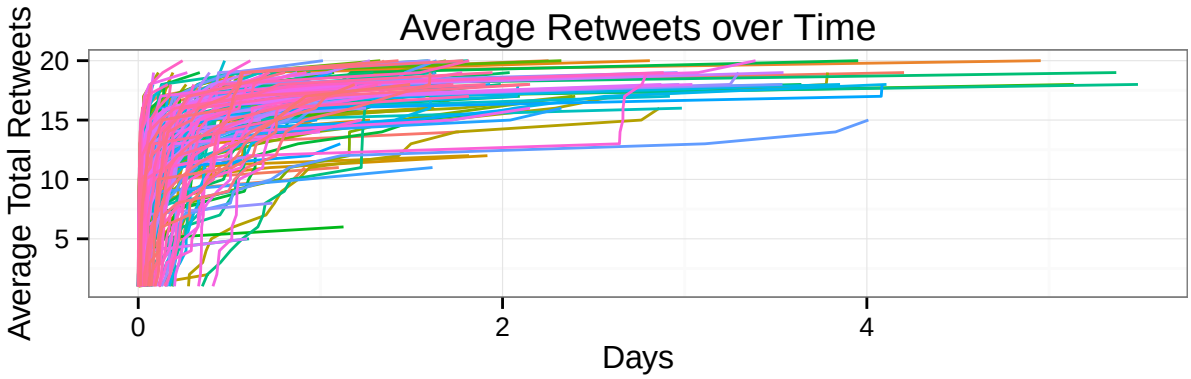
- NewsTweet: The unique tweet ID of the originating tweet
- Org: The user account responsible for the originating tweet
- tmtotal: The number of seconds elapsed since the original tweet was posted

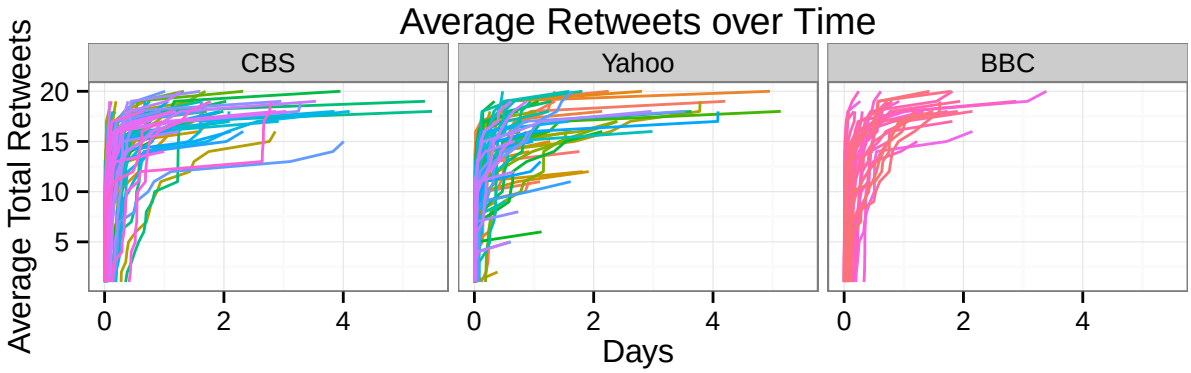
5 Methods

5.1 Data Exploration



This first plot shows the cumulative density curve for the collection of all tweets along with the components that each news organization adds to the density curve. While we can see that CBS seems to have a higher level of impact on the curve due to its greater magnitude, each of the curves seem to share the same general shape.





This second plot displaus the retweet density foe each of the tweets in the entire dataset. We can notice that the individual lines seem to share the same general pattern. BBC seems to have th emost consistent trends among its tweets, but it also seems to have put out the smallest number of tweets over the timeframe. The next most consistent seems to be CBS. Yahoo, seems to be the least consistent, with some stories garnering a minimal number of retweets.

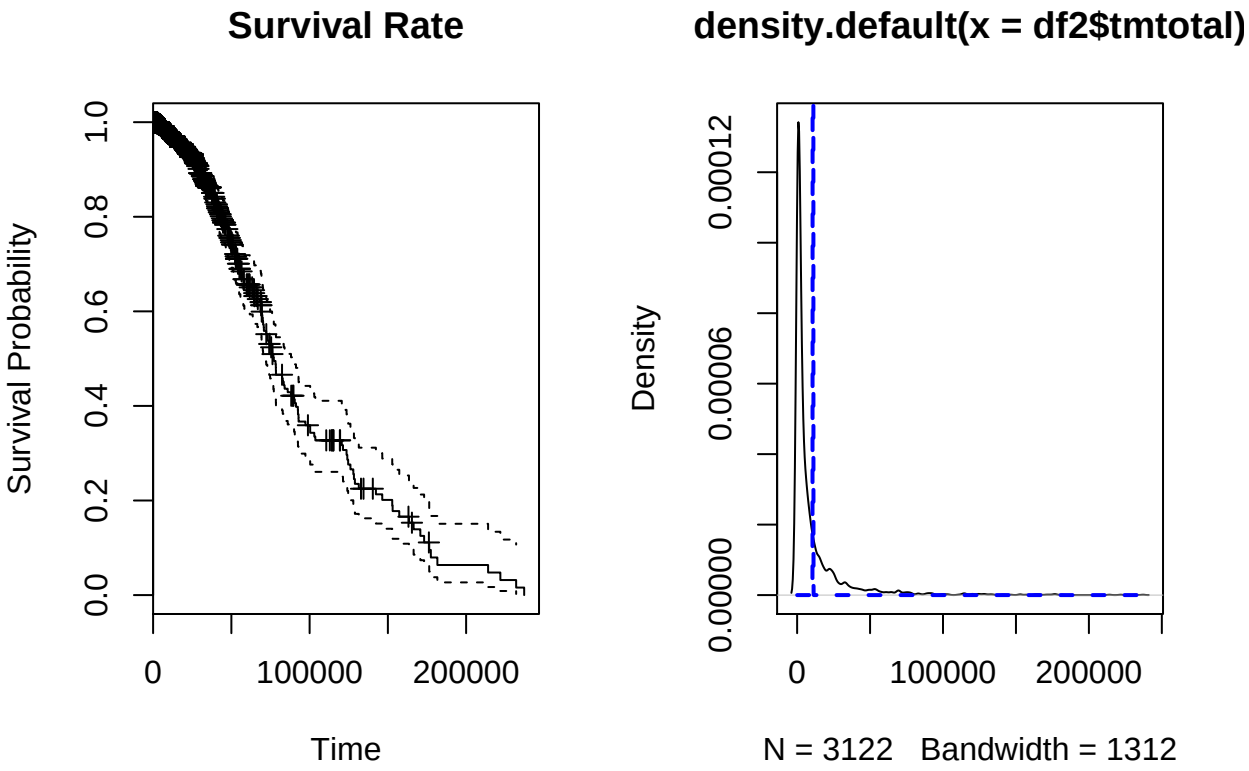
5.2 Survival Analysis

The main method of analysis to analyze the propensity of a Twitter user to retweet news over time, will be the application of a continuous time series analysis. Looking at the data, we can note that there are no cases in the dataset in which a tweet was unable to garner at least a single retweet. Beacause there is a lack of a negative case, we are unable to use logistic regression in order to estimate the probability over time. However, a cox proportional hazard

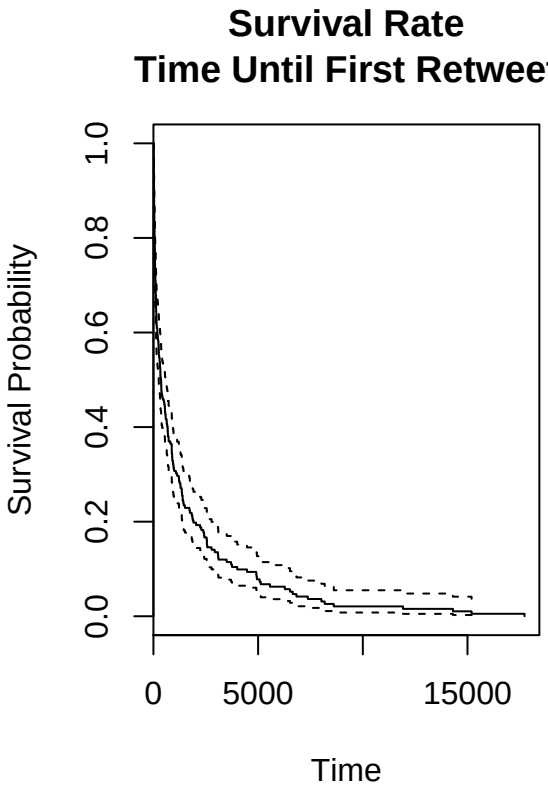
survival analysis does not need a negative case in the data, so that is the model that we will use.

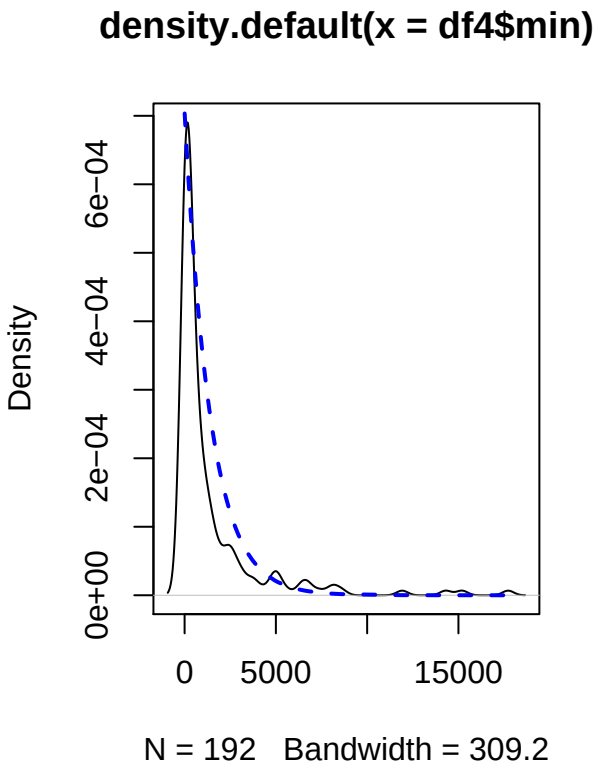
For this analysis, we will need to give each retweet tree a unique identifier and aggregate each tree into the same data frame. With that dataset, I will run a survival analysis to analyze the lifetime of tweets. For this analysis, I will run three different model:

(1) The Survival rate until the last retweet



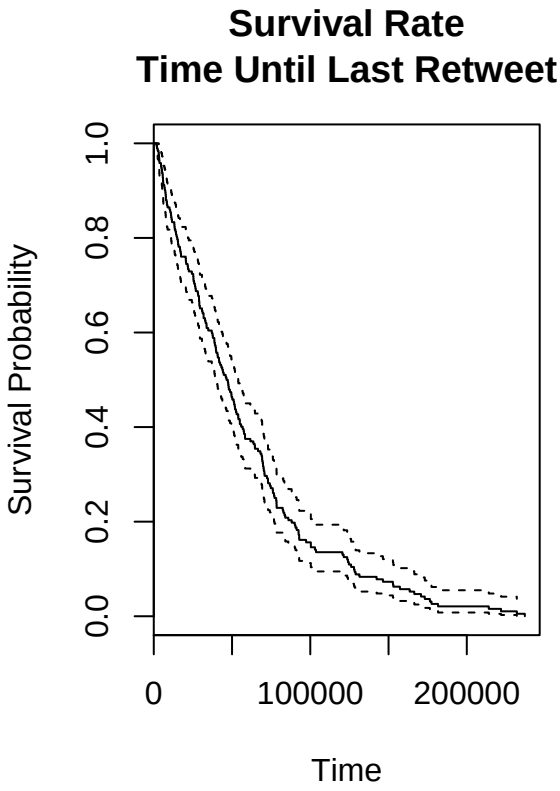
(2) The Survival until the first retweet

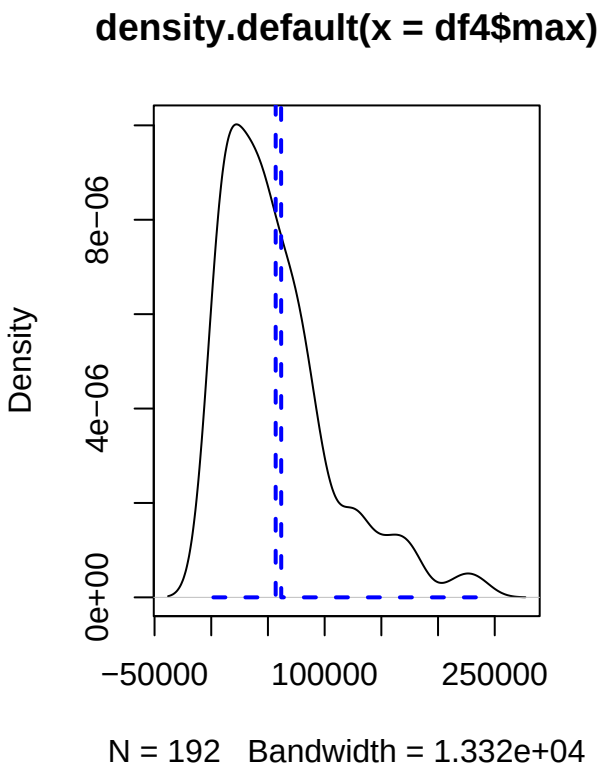




&

(3) The survival until the last retweet





6 Results

The Survival Analysis was able to show that the retweeting trends tended to follow probability distributions associated with waittimes. The best fits occurs with the overall survival rate, fitted with a poisson distribution with lambda = 10867.97 and eith the survivial rate until the first retweet, ditted with an exponential distribution with the rate = 0.0007053818 . Unfortunately, the last model didn't easily fit with a probability distribution, but the best fit consisted of a poisson distribution with lambda = 57898.63 .

7 Limitations

The lack of negative cases within the dataset was a severe restriction in being able to run a more robust model with a larger number of predictors, as we could have done with a logistic

regression. Additionally, due to the time constraints of pulling data under the restrictrions of the Twitter API limited the size of the dataset, potentially causing inconsistency with density functions.

8 Conclusion

In this analysis, I was able to confirm that the waiting times for retweets for news organizations follows an exponential probability model, and the combined overall survival rate is similiar to that of a poisson distribution. While I was not able to fit an accurate distributin with the thrird distribution, I believe that this was due to omititng the influence of previous retweeters when calculating the marginal probability of survival.

9 Appendix

9.1 Twitter Data Extraction

Written in Python

```
# -*- coding: utf-8 -*-
"""

@author: JonathanCampbell
"""
import datetime as dt
from time import sleep

class query_limits:
    def __init__(self, limit, n_min):
        self.limit = limit # n queries / time period
        self.minutes = n_min # time period
        self.start = dt.datetime.now()
        self.q_times = []
        self.q_count = 0

    def difftime(self):
        delta = dt.datetime.now() - self.start
        return(delta.seconds)

    def timecheck(self):
        # Assign times based on API query limits
```

```
        cycle_time = 60*self.minutes + 1 # minutes into seconds
        waittime = cycle_time - self.difftime()
        if self.q_count <= self.limit-1:
            return(sleep(0.1))
        else:
            # shift starttimes with query runs
            self.start = self.q_times[-self.limit]
            # Add in waittimes to avoid pull errors
            waittime = cycle_time - self.difftime()
            if waittime > 0:
                # Exceeded limit
                if self.q_count < self.limit:
                    self.start = dt.datetime.now()
                print('Limit reached: %s s till next query' % waittime)
                return(sleep(waittime+1)) # Waittime + buffer
            else: # Within limit
                return(sleep(0.1))
    def query(self):
        self.timecheck()
        self.q_count += 1
        self.q_times.append(dt.datetime.now())

# Get info from user:
consumer_key = input('Enter your consumer key: ')
consumer_secret = input('Enter secret key: ')
access_token = input('Enter your access token: ')
access_token_secret = input('Enter secret token: ')
```

```
user = input("Enter Twitter handle you'd like to search: ")

import tweepy

auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_token_secret)
api = tweepy.API(auth)

tmline = api.user_timeline(screen_name = user, count = 200)

q_retweet = query_limits(15, 15)
q_retweeters = query_limits(15,15)
retweets = {}
retweeters = {}
for tweet in reversed(tmline[-100:-1]):
    if tweet.retweet_count <= 100:
        q_retweet.query()
        resultset = api.retweets(id=tweet.id)
        retweets[tweet.id] = {}
        retweets[tweet.id]['id'] = []
        retweets[tweet.id]['tmd'] = []
        retweets[tweet.id]['user'] = []
        retweets[tweet.id]['tree'] = []
        retweets[tweet.id]['branch'] = []
        retweets[tweet.id]['text'] = []
        for result in resultset:
            retweets[tweet.id]['id'].append(result.id)
```

```
            retweets[tweet.id]['tmd'].append(result.created_at)
            retweets[tweet.id]['user'].append(result.user.id)
            retweets[tweet.id]['tree'].append(result.retweeted)
            retweets[tweet.id]['branch'].append(result.retweet_count)
            retweets[tweet.id]['text'].append(result.text[0:19])
    else:
        q_retweeters.query()
        c_query = api.retweeters(id=tweet.id, cursor = -1)
        retweeters[tweet.id] = c_query[0]
        next_c = c_query[1][1]
        while next_c != 0:
            q_retweeters.query()
            c_query = api.retweeters(id=tweet.id, cursor=next_c)
            retweeters[tweet.id] += c_query[0]
            next_c = c_query[1][1]

tweets = {}
for tweet in tmline:
    tweets[tweet.id] = tweet

retweets1 = {'NewsTweet': [],
        'tmd': [],
        'text': [],
        'retweet_id': [],
        'retweet_user': [],
        'tree': [],
        'branch': [],
```

```
        'difftime': []}

for retweet in retweets:
    retweets1['NewsTweet'] += [retweet]*len(retweets[retweet]['id'])
    retweets1['retweet_user'] += retweets[retweet]['user']
    retweets1['retweet_id'] += retweets[retweet]['id']
    retweets1['tmd'] += retweets[retweet]['tmd']
    retweets1['text'] += retweets[retweet]['text']
    retweets1['branch'] += retweets[retweet]['branch']
    retweets1['tree'] += retweets[retweet]['tree']
    mydate1 = retweets[retweet]['tmd']
    mydate2 = []
    for dtm in mydate1:
        x = tweets[retweet].created_at
        x = dtm - x
        mydate2.append(x)
    retweets1['difftime'] += mydate2

import numpy as np
import pandas as pd

df = pd.DataFrame(retweets1)
df.to_csv('/Users/JonathanCampbell/Desktop/Columbia/2_Thesis/2_DataCollection/pydata5.cs
```

9.2 R Code

```
library(plyr)
library(dplyr)
library(ggplot2)
library(gridExtra)
library(survival)
library(KMsurv)
library(survival)
library(KMsurv)
```

```
df_cbs <- cbind(read.csv("CBS_retweets.csv"), Org="CBS")
df_yahoo <- cbind(read.csv("Yahoo_retweets.csv"), Org="Yahoo")
df_bbc <- cbind(read.csv("BBC_retweets.csv"), Org="BBC")
df1 <- rbind(df_cbs, df_yahoo, df_bbc)
df1$NewsTweet <- as.character(df1$NewsTweet)
df1$retweet_id <- as.character(df1$retweet_id)
df1$retweet_user <- as.character(df1$retweet_user)
df1$difftime <- as.character(df1$difftime)
df1$day <- as.numeric(substr(df1$difftime,1,1))
df1$hrs <- as.numeric(substr(df1$difftime,8,9))
df1$min <- as.numeric(substr(df1$difftime,11,12))
df1$sec <- as.numeric(substr(df1$difftime,14,18))
df1$tmtotal <- df1$day*(24*(60^2)) + df1$hrs*(60^2) + df1$min*60 + df1$sec
```

```
df2 <- df1 %>%
  arrange(NewsTweet,tmtotal) %>%
  mutate(no.rt = 1) %>%
```

```
  group_by(NewsTweet) %>%
  mutate(no.rt = cumsum(no.rt))
df3 <- df1 %>%
  group_by(tmtotal) %>%
  summarize(n=n()) %>%
  mutate(avg.no.rt = cumsum(n)/length(unique(df1$NewsTweet)))
df3.1 <- df1 %>%
  group_by(Org, tmtotal) %>%
  summarize(n=n()) %>%
  mutate(avg.no.rt = cumsum(n)/length(unique(df1$NewsTweet)))
df4 <- df1 %>%
  group_by(NewsTweet) %>%
  summarize(max=max(tmtotal), min=min(tmtotal))
groups <- df2 %>%
  group_by(NewsTweet) %>%
  group_size()
event <- c(rep(0,groups[1]-1),1)
for(i in 2:length(groups)){
  event <- c(event,rep(0,groups[i]-1),1)
}
df2$event <- event
```

```
library(ggplot2)
ggplot(df3,aes(x=tmtotal/(60^2*12),y=avg.no.rt)) +
  geom_line() +
  theme_bw() +
  ylab("Average Total Retweets") + xlab("Days") +
```

```
  ggtitle("Average Retweets over Time")
ggplot(df3.1,aes(x=tmtotal/(60^2*12),y=avg.no.rt)) +
  geom_line() +
  theme_bw() +
  ylab("Average Total Retweets") + xlab("Days") +
  ggtitle("Average Retweets over Time") +
  facet_grid(.~Org)
```

```
ggplot(df2, aes(x=tmtotal/(60^2*12), y=no.rt, color=NewsTweet)) +
  geom_line() +
  theme_bw() +
  theme(legend.position="NULL") +
  ylab("Average Total Retweets") + xlab("Days") +
  ggtitle("Average Retweets over Time")
ggplot(df2, aes(x=tmtotal/(60^2*12), y=no.rt, color=NewsTweet)) +
  geom_line() +
  theme_bw() +
  theme(legend.position="NULL") +
  ylab("Average Total Retweets") + xlab("Days") +
  ggtitle("Average Retweets over Time") +
  facet_grid(.~Org)
```