

Becca Getto  
February 25<sup>th</sup>, 2016

## Text Mining & Analysis: Weather Sentiment Analysis vis Twitter

### Project Overview:

I created a weather preference analyzer by scraping tweets from Twitter with various weather conditions as the hashtag then running them through a Markov sentiment analyzer to determine if people responded positively or negatively to the weather.

### Implementation:

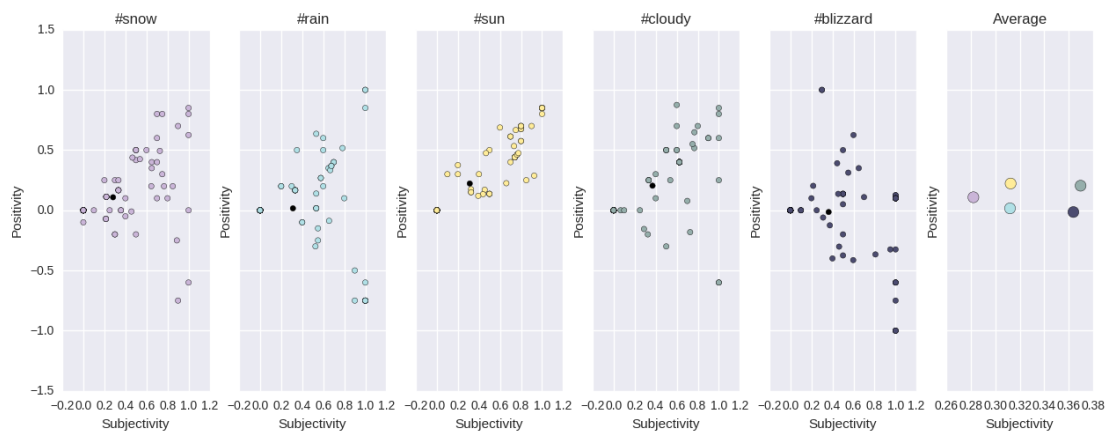
I broke my analyzer up into two main programs, the first two acquire and pickle the data, and the second to process and represent the data. In order to create my data set, I scraped Twitter for tweets with hashtags relating to certain weather conditions (ex. #snow, #cold, #sun). I collected 1000 tweets for each of the weather conditions I was interested in. I stored this data in a dictionary format. Each weather condition hashtag served as a key and the corresponding value was a list of the 1000 collected tweets represented as strings. The dictionary format seemed to be the best way to organize the data as it made it easy to search through the various weather conditions and manipulate only the tweets that I was interested in at that time.

Once the data was collected, I began to process the tweets to determine the sentiment associated with them. I found that some of weather condition hashtags themselves had a positivity or negativity value using the sentiment analysis in the pattern module, so I chose to remove the hashtag from the body of the tweet. Once the hashtag was removed from the text of the tweet I used the sentiment analysis available in pattern.en to determine a positivity and subjectivity value. I stored this information in a dictionary organized in the same way as the tweet dictionary. In order to represent the data, I created a scatter plot of the sentiment for each tweet associated with various weather conditions as well as they average sentiment.

### Results:

As a person who revels in the sunshine and is not at all fond of the cold dreary winter weather, I was quite surprised to find that overall, the people of Twitter do not feel the same way. The only weather conditioned I analyzed that had a negative average sentiment was #blizzard, and it was barely below zero. The blizzard condition also had the greatest variability in sentiment for the individual tweets with some as low as -1 and other as high as 1. I found that #sun and #cloudy had the highest positivity

values yet the #cloudy condition was much more subjective. The #sun condition was the condition in which every individual tweet was either neutral or positive, so at least people agree with me that sunshine is pretty wonderful!



## Reflection:

I believe that my project was well scoped for my current level of python experience. As I do not have an extensive coding background nor do I have particularly high self-efficacy in the area, I wanted to select a project that felt manageable, but also had potential areas for extension. Before I began writing any code, I drew out how I wanted my data to be organized on paper, which helped in deciding what data type to use. The process of data collection was a bit more difficult than I had anticipated as I ran into trouble using the API, but I was able to receive help from instructors. As I went through the processing section of my program, I continued to write out on paper what I wanted the program to do and then moved to pseudo-code before attempting to actually implement code and get bogged down in the syntax and errors and get discouraged. I also made sure I had someone nearby who could help answer my questions and encourage me when I got stuck. I wish I had known more about the sentiment analyzer and how it assigns positivity or negativity before I started. Given that many tweets are simply a sequence of emojis and other non-english characters followed by a collection of hashtags, the sentiment analyzer often had only a few words to work with. Many of the tweets were deemed neutral because of this lack of information. Overall, this project went well and I am happy that I was able to create a program that met my initial goal.