|  |
| --- |
| Ryerson University—CKME 136 |
| Influencers in Online Social Network |
| By: Lyndsay Roach |

|  |
| --- |
| Roach  4/4/2016 |

Influencers in Online Social Network

# Introduction

Many entrepreneurs are using social media analytics to improve their marketing techniques and to expand their client base. One of the key components of social media analytics is utilizing patterns in online social networks for machine learning algorithms and predictive analysis. We have been given a dataset from the social media analytics company PeerIndex that contains multiple features describing the relationship between pairs of Twitter users, as well as, a label (or class) for each relationship. The label represents a human’s judgement of which user in the pair is more influential on Twitter. This information is useful for companies that are marketing to teenagers and young adults. Most of us are familiar with celebrity endorsements, but since the rise in popularity of social media many people who do not have conventional celebrity status are still able to have their messages reach millions of others. If companies can obtain information on who has a lot of influence online, then they will have an effective way of aiming their marketing efforts towards demographics that are heavily influenced by their peers.

The following will explore the effectiveness of two machine learning algorithms on their ability to predict the classification of Twitter data. The goal of this project is to answer the question: does a logistic regression model or a naïve Bayes model classify the Twitter data more accurately? To solve this problem, we will use RStudio to analyze the data to see if it reflects our everyday knowledge of Twitter activity and then to build both a logistic regression model and a naïve Bayes model. Finally, again in RStudio, we will compare the evaluation of the models accuracy and draw conclusions about the effectiveness of the models.

The entire project can be viewed on Github using the link: <https://github.com/lyndsayroach/Final-Project-Report.git>. The links for coding of the individual steps are provided in the Approach section below.

# Definitions

Before beginning, we shall define the Twitter terminology that we will be using. This will also refresh our general knowledge of the logistics of Twitter.

**Posts:** The main activity on Twitter is sending 140 character messages known as “posts” or “tweets”. They can include letters, numbers, photos or videos. Most tweets are public, but typically Twitter users will subscribe to the accounts they like so those posts will appear on their newsfeed. In our dataset, the posts for A users and for B users are calculated as real numbers; therefore, they are not to be interpreted as “number of posts”.

**Follower/Following:** When a Twitter user subscribes to another user’s posts it is called “following”. Likewise, if another user subscribes to your posts they are called your “follower”. In our dataset, the follower and following count for A users and B users are integers, so they are to be interpreted as an exact count.

**Retweet:** If a Twitter user likes a post by one of the users they follow and they want to share it with their followers, they can repost it and this is called “retweeting”. In our dataset, the retweets received and retweets sent for A users and for B users are calculated as real numbers; therefore, they are not to be interpreted as “number of retweets”.

**Mentions:** Twitter users have a username that begins with an ‘at’ symbol. They take the format @username. You can use someone’s username to reply to one of their posts by including it in your post. This is called a “mention”. In our dataset, the mentions received and mentions sent for A users and for B users are calculated as real numbers; therefore, they are not to be interpreted as “number of mentions”.

**Lists:** A Twitter user can follow a group of users, known as a “list”, instead of just following an individual user. In our dataset, the listed count for A users and B users are integers, so they are to be interpreted as an exact count.

# Literature Review

1. Benevenuto, Fabrício and Meeyong Cha, Krishna P. Gummadi, Hamed Haddadi. “Measuring User Influence in Twitter: The Million Follower Fallacy,” in *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media*, 10-17.AAAI, 2010.

Benevenuto and his colleagues examine the role of influence in sociology and marketing, focusing on influence in social networks on Twitter. In this article, they give a critical perspective on the theory that there are a minority of influential individuals in a network and by targeting the influentials that there will be far reaching word-of-mouth advertising at a low cost. They describe the influentials as “informed, respected and well-connected” (Benevenuto, 11, 2010). They consider three ways of measuring ones influence on Twitter: number of followers of a user, number of retweets of a user, and number of mentions container a user’s name.

1. Fürnkranz, Johannes and Eyke Hüllermeier. “Preference Learning: An Introduction,” in *Preference Learning*, 1-17. Berlin: Springer-Valen, 2010.

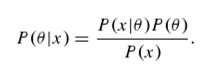
In Fürnkranz and Hüllermeier’s chapter introducing preference learning, they describe a preference as being a “relaxed constraint which, if necessary, can be violated to some degree” (Fürnkranz, 1, 2010). The goal of preference learning is to build models using empirical data to predict an individual’s or a group of individuals’ preference, where a set of items with known preferences to learn a model that will predict the preferences for a new set of items. Machine learning has advanced the computing of preference learning, specifically in regards to ranking tasks. Predicting preferences and the total order (ranking) of preferences is frequently used in e-commerce to provide personalized recommendations of products or services. This chapter goes beyond the scope of this project by detailing three different types of ranking tasks: label ranking, instance ranking and object ranking.

1. Li, Hang. “Data Labeling,” in *A Short Introduction to Learning to Rank*, 2. Tokyo: IEICE, 2010.

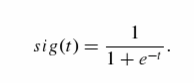
In this article, Hang discusses the theory and methods behind ranking the relevance of web pages. Although this is not the same problem we are exploring in this project, it provides insight on how the dataset was created. One of the methods in created page ranking training sets is to start by randomly selecting queries from a query log. Then imputing those queries into various search engines and noting the multiple documents that are associated with each query. Finally, human judges rank the documents based on relevance using an ordinal scale.

1. Pathak, Manas A. “Classification,” in *Beginning Data Science with R*, 115-136. Springer International Publishing: 2014.

This chapter explains how to preform classification techniques in R. For the purpose of this project, we will focus on naïve Bayes and logistic regression. Pathak summarizes Bayes theorem by stating that it is based on the prior probability from before seeing that data and the posterior probability from the data with updated information. In words, Bayes theorem is “that posterior probability is proportional to the data likelihood times the prior probability” (Pathak, 117-118, 2014). If the prior probability is P(θ) and the likelihood of the data x is P(x | θ), then the posterior probability is



Bayes theorem is be used as a classifier by calculating the conditional probability that a data point has a particular label (class). Then Pathak explains how to use the naiveBayes() function from the e1071 package in R. Next, Pathal begins explaining logistic regression is used to classify binary data and that its method is rooted in what is call the sigmoid function:



The function is within the range {0, 1} and it is monotonically increasing. Logistic regression is uses the sigmoid function to classify data based on the conditional probability of data point x being in the class y. Then Pathak explains how to use the generalized linear model function glm() to do logistic regression in R, using the parameter family = ‘binary’.

1. Rodríguez, Germán. “Logit Models for Binary Data,” in *Lecture Notes on Generalized Linear Models*. New Jersey: Princeton University, 2007.

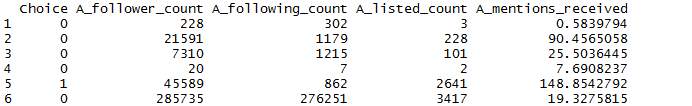
In this chapter, Rodríguez gives an example of how to build a regression model for dichotomous data, called logistic regression. It discusses the theory and formulas behind logistic regression, where the dependent variable is of the type ‘success or failure’ (derived from the Bernoulli trial) and the outcome is based on independent variables or attributes.

# Dataset

The dataset has been obtained from [www.kaggle.com](http://www.kaggle.com), originally provided by the London-based company PeerIndex. PeerIndex administers various services in social media analytics, including assigning a score to social media users to reflect their social capital. This dataset contains information about the relationship between 5500 pairs of Twitter users based on eleven pre-determined features of their Twitter activity, such as number of followers and number of retweets. Each record describes the relationship between an individual A and an individual B and is labelled either as ‘1’ if A is considered the more influential user or as ‘0’ if B is considered the more influential user. It is important to remember that the records are labelled based on a human interpretation of who is the more influential user.

The dataset has 23 attributes, 5500 records and no missing values. The first column is the dependent variable “Choice” and it is a dichotomous categorical variable. Each observation is either ‘0’ which represents choosing user A or ‘1’ which represents choosing user B, as stated above. The rest of the columns contain continuous quantitative data (some integers and some real numbers). Theses variables are of type ratio because their observations can be ordered and the value ‘0’ represents the quantity zero (unlike the column “Choice”). The particular characteristics of the dataset that makes it appropriate for logistic regression and naïve Bayes algorithms is that the dependent variable has mutually exclusive and exhaustive categories and the rest of the variables are independent of each other.

Table 1

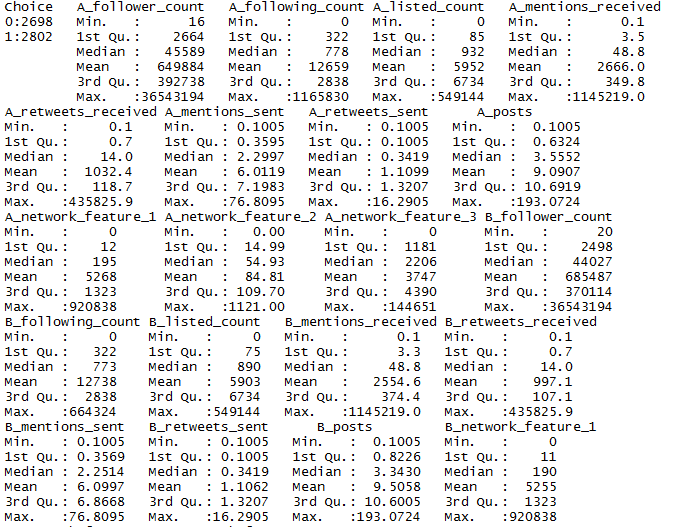


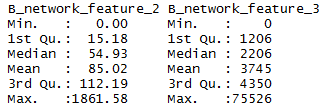
Above, in table 1, are the first 6 rows and 5 columns of the dataset. This table is to give an example of how the dataset looks like. Remember, the ‘1’ and ‘0’s in the first column are labels not integers.

One could make many assumptions about how this dataset will behave based on a general knowledge of how Twitter works, but since we have no information on how PeerIndex measures all the features of Twitter activity we should not rely too much on practical assumptions. There are two main difficulties with interpreting the data practically. Firstly, we are unable to interpret the attributes that are represented by real numbers. For example, if Twitter user A has the value of 1.111 sent retweets, we have no information on what 1.111 measures exactly. Secondly, the data set includes three attributes for both user A and user B that are simply called “Network Feature” 1, 2 and 3. The only hint about what these features could be that we can find is on [www.kaggle.com](http://www.kaggle.com) where it mentions *volume of interactions*; otherwise, we cannot speculate on what those features measure. However, we can check for linear correlations between the network features to see if they are worth analyzing further. Using RStudio we compute various combinations of network features; for example, user A’s network feature 1 with user B’s network feature 3 or A’s network feature 1 with B’s network feature 1 etc. All combinations had a very low to moderate linear correlation; therefore, we will use these features only in the model and not for the exploratory analysis.

An important aspect of this dataset to note is, as seen below in table 2, for each attribute the mean is larger than the median.

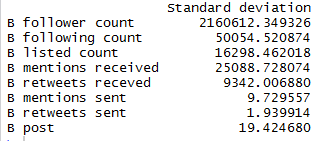
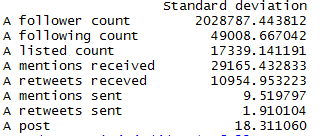
Table 2





Therefore, their distributions are skewed because of extreme values and we should consider the median as the appropriate measure of centre. This is also evident when we calculate the standard deviation, notably for the attributes “follower count”, “following count”, “listed count”, “mentions received”, and “mentions sent” for both users A and B. The standard deviation measures the spread of the data, where a high standard deviation indicates that the values in the data set are more spread out. Since the equation for the standard deviation uses the mean as a measure of centre, it is not surprising to get the large values we do in table 3.

Table 3



At this time, we should also pay attention to the fact that table 3 demonstrates well that the data behave similarly for both the A users and the B users.

# Approach

## Step 1: Import dataset into RStudio

Download dataset train.csv from [www.kaggle.com](http://www.kaggle.com) and import into RStudio using the command read.csv(), then check for missing values using anyNA(). Change dependent variable type from integer to factor and start with looking at the summary of the data.

<https://github.com/lyndsayroach/Final-Project-Report/blob/master/Step%201.R>

## Step 2: Explore dataset

### Step 2.1: Brief summary statistics

Begin with looking at the dimensions and the first 6 lines of the dataset using the commands dim() and head() respectively. Then use Pearson’s correlation to attempt to make a judgement on the “network features”. Finally, to wrap up the brief summary statistics, we will look at the mean, median and standard deviation.

### Step 2.2: Further analysis

Try to get an idea of which attributes are the most significant by splitting the dataset into two: those that are labelled ‘1’ and those that are labelled ‘0’. Then we shall compare medians and try to find patterns in the Twitter actions *done to* the user that is viewed as more influential and in the actions *done by* the user that is viewed as more influential.

### Step 2.3: Data visualization

To analyze “posts” and “listed count” we will use the lattice package in RStudio to visualize a sample of the dataset (first 100 records). We will look at how posts and listed count affects the number of retweets. Then we use a subset of the sample to visualize the skewness of the data using Q-Q plot and a density function.

<https://github.com/lyndsayroach/Final-Project-Report/blob/master/Step%202.R>

## Step 3: Build logistic regression model

Create formula where Choice is the dependent variable and split data into 70% for training and 30% for testing. Use the glm() command in R to build a logistic regression model that will help us predict the labels of the test dataset. Then use confusion matrix to evaluate accuracy and ROCR and caret packages to plot to receiver operating characteristic (ROC) curve.

<https://github.com/lyndsayroach/Final-Project-Report/blob/master/Step%203.R>

## Step 4: Build naïve Bayes model

Separate data into 70% for training and 30% for testing. Use the naiveBayes() command in RStudio to build a naïve Bayes model that will help us predict the labels of the test dataset. Then use confusion matrix to evaluate accuracy.

<https://github.com/lyndsayroach/Final-Project-Report/blob/master/Step%204.R>

## Step 5: Compare models

Compare accuracy results and discuss which model is more appropriate and why. Make suggestions of what could be improved.

<https://github.com/lyndsayroach/Final-Project-Report/blob/master/Logistic%20Regression%20(part%202).R>

<https://github.com/lyndsayroach/Final-Project-Report/blob/master/Naive%20Bayes%20(part%202).R>

## Step 6: Concluding remarks

Reiterate the research question and the problem that needed to be solved. Comment on whether or not the goal of the project was met. Provide insight for future exploration and research in social media analytics

# Results

If we think about our general knowledge of the inner workings of Twitter, we can make the assumption that volume of actions that are done to a user’s Twitter account (passive actions) are more indicative of their influence compared to the actions that they perform (active actions). The more people that want to see and share your activity give you more influence to spread information. To see if this is reflected in the dataset, we split the data based on their labels and compared the medians of each users’ passive and active actions. Tables 4 through 9 show that the user that is considered more influential has the highest medians when it comes to passive actions on their account. The row names in these tables indicate the type of action, the user and the label of those observations. For example, the row “followersA1” indicates the follower and following count medians for all user A’s where user A is viewed as the more influential individual. Whereas, the row “followersB1” indicates the follower and following count medians for all user B’s where user A is viewed as the more influential individual.

**Median follower and following count when A is viewed as the more influential user:**

Table 4



**Median follower and following count when B is viewed as the more influential user:**

Table 5



**Median mentions received and mentions sent count when A is viewed as the more influential user:**

Table 6



**Median mentions received and mentions sent count when B is viewed as the more influential user:**

Table 7



**Median retweets received and mentions sent count when A is viewed as the more influential user:**

Table 8



**Median retweets received and mentions sent count when B is viewed as the more influential user:**

Table 9



The two Twitter actions included in this analysis that are only able to be performed by the Twitter users themselves are “posts” and “listed count”. If once again we make an assumption based on our practical knowledge of Twitter, we can say that the amount of retweets a user receives is very telling of their influence because it shows that what they tweet is considered worthy of spreading. Using the xyplot() command from the lattice package, we will look at retweets in relation to both the users’ post and listed counts. The xyplot() command plots two variables on the xy-axis and outputs plots according to the variables labels. In our case, xyplot() outputs a plot of the data labelled ‘0’ and a plot of the data labelled ‘1’. For the purpose of this analysis we will use a subset of the first 100 records[[1]](#footnote-1).

In figure 1 and figure 2, we can see that the rate at which a user is posting effects how many retweets they get. The more the user posts, the more they get retweeted at an exponential rate which makes them appear more influential.

Figure 1

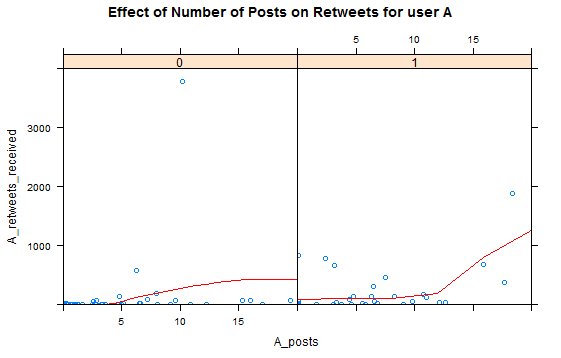
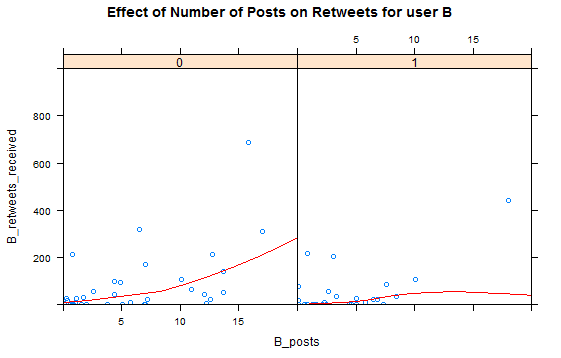


Figure 2



We can a similar effect in figure 3 and figure 4 with users’ list counts. However, the effect seems to be more dramatic that in the plots for posts.

Figure 3

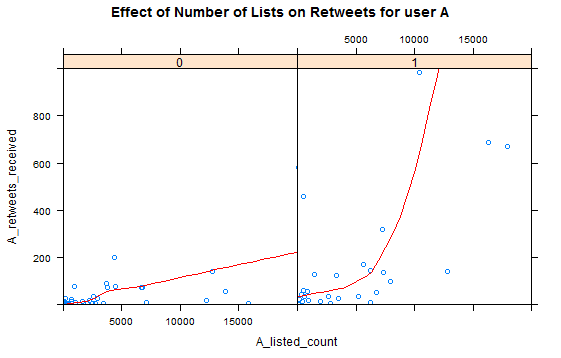
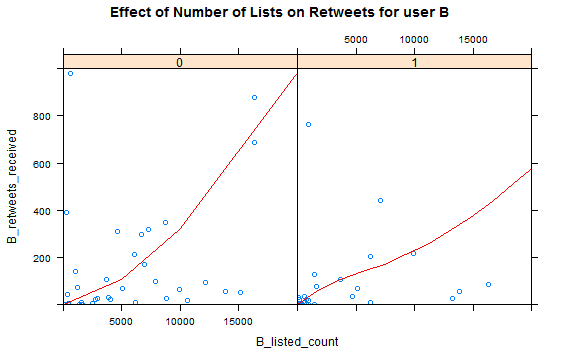
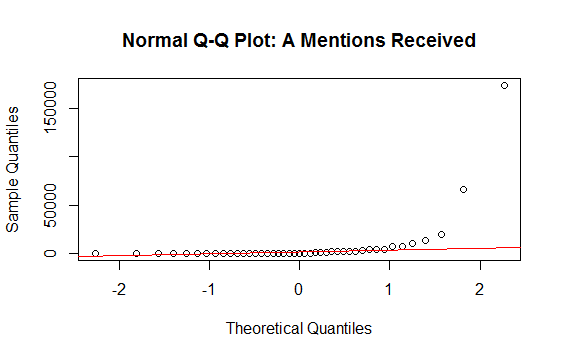


Figure 4



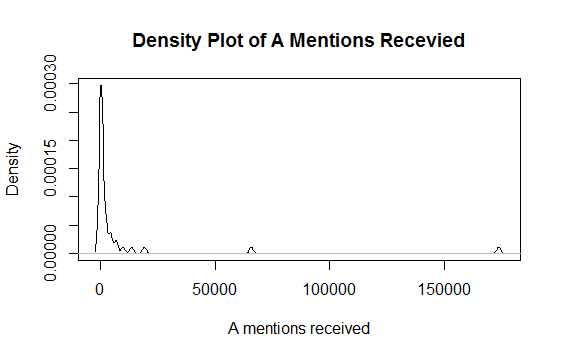
Simply for the purpose of visualizing the skewness of the data, we will plot a subset of the mentions received by individuals A when they are considered the more influential user. In figure 5 we have the normal Q-Q plot, where we can see that the points deviate from the red line that represents a normal distribution. The way that the points increase in a concave up shape means that the distribution of the data is right skewed.

Figure 5



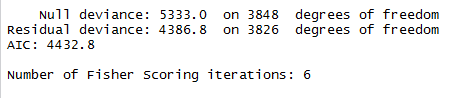
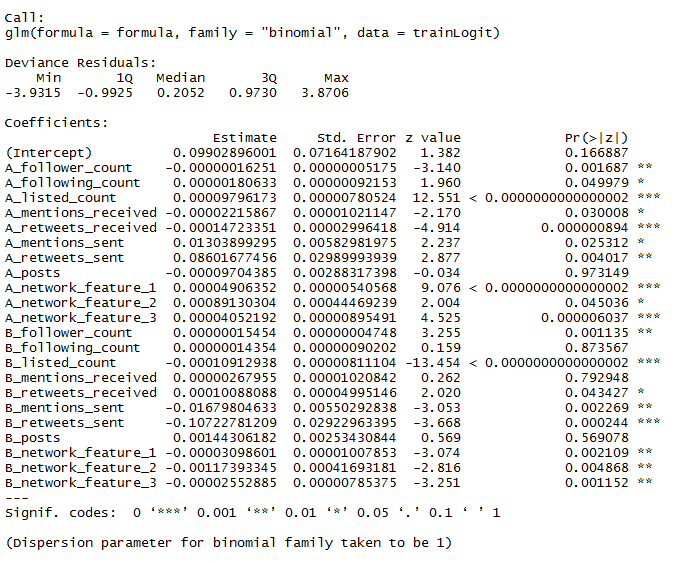
The skewness is more evident in the density plot below in figure 6. Here we see that most of the data lies before the value 25 000, but the tail is streched to the right because of extreme values.

Figure 6



The first classification method we will look at is the logistic regression model. We use RStudio to build the model and provide the results seen in figure 7, 8 and 9. In figure 7, we can see the output of the summary of the model. The column named “Estimate” contains the estimated coefficients of each of the attributes. Moreover, we should pay attention to 4 of the attributes (A posts, B following count, B retweets received, B post) because they are not statistically significant, since their p-values are greater than 0.05. This means that these attributes may not contribute to the accuracy of the predictive model.

Figure 7



According to figure 8 and figure 9, the logistic regression model was relatively accurate at predicting the labels. In figure 8 we can see that label ‘0’ was correctly predicted 543 times and label ‘1’ was correctly predicted 650 times. Figure 9 shows this a bit more clearly with around 72% of predictions being true and only around 28% of them being false.

Figure 8

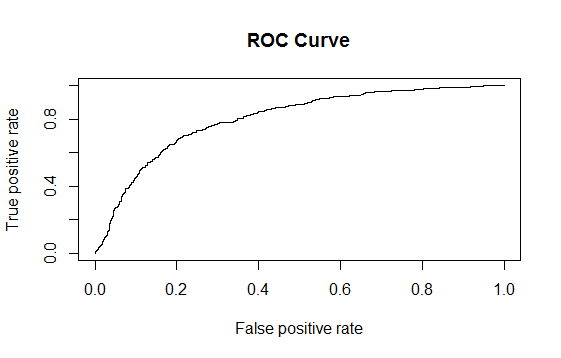


Figure 9



Figure 10 is a visual representation of the accuracy of the model. Ideally the curve would follow as close as possible to the upper left hand corner of the plot; however, in practice that is an unlikely outcome and we simply aim for a curve that is approaching the upper left hand corner.

Figure 10



The second classification method we are going to look at is the naïve Bayes model. We build the naïve Bayes model in RStudio and use a confusion matrix to document accuracy. Figure 11 and figure 12 show that the naïve Bayes model does not perform as well as the logistic regression model. Figure 11 shows that label ‘0’ was correctly predicted only 129 times, while label ‘1’ was correctly predicted 756 times. Moreover, figure 12 shows that only 53% of labels were predicted correctly and 47% of labels were predicted incorrectly.

Figure 11



Figure 12



While it is an easy model to implement, it is very fragile. It does not do well with complexity and it assumes zero dependence between attributes, which is unrealistic in practice. Since the logistic regression model is the appropriate choice in this case. Let us see if we can improve the model’s performance. By removing some of the statistically insignificant attribute may be we will be able to prevent over-fitting the data and increase the accuracy. First we shall try by removing “A\_posts” and “B\_posts” because it was the only attribute that was statistically insignificant for both users.

Figure 13

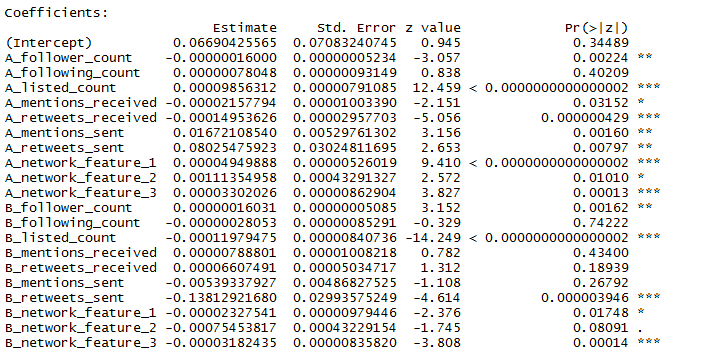


Figure 13 shows the estimated coefficients for the new logistic regression model, where there are now 5 attributes that are not statistically significant. However, below in figure 14, we see that around 74% of the labels were predicted correctly. In this model, both “A\_following\_count” and “B\_following\_count” have p-values greater than 0.05, so let us see if removing statistically insignificant attributes one more time will improve the model’s accuracy.

Figure 14



Figure 15 shows the accuracy of our final iteration of the logistic model. As we can see removing those last two attributes did not change the accuracy significantly.

Figure 15

# 

As seen earlier, the naïve Bayes model is sensitive to complex data. Therefore, let us explore the change in accuracy when we remove the same 4 attributes from our naïve Bayes model that we removed from our logistic regression model. Figure 16 shows that the accuracy does increase a bit, with 65% labels being predicted correctly and 35% being predicted incorrectly.

Figure 16

# 

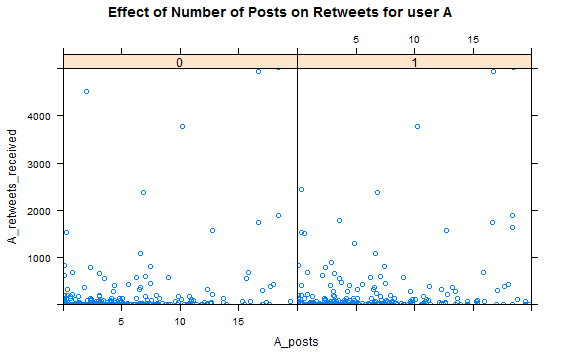
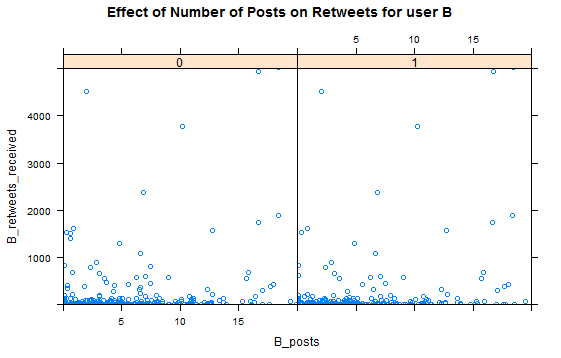
# Conclusions

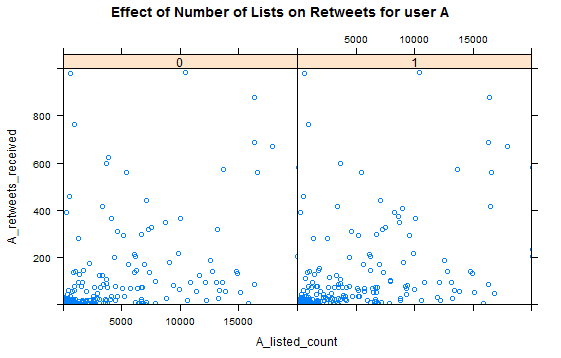
The purpose of this project was to compare how a logistic regression and a naïve Bayes model preformed when predicting the labels of a Twitter dataset. First we explored the dataset using summary statistics and visualization tools. We observed how the data behaves in similar ways to how we interpret Twitter actions in real life. We can conclude that actions done to a Twitter user’s account, such as follower count, retweets received and mentions received, have a bigger impact on how a human judges the influence of a user than the actions done to a user. A Twitter user’s posts are not going to spread quickly if they do not have many followers and do not get many retweets. When looking at the mean and standard deviation we noticed that the data has a large spread and is right skewed by extreme values, which leaves the data vulnerable to over-fitting. After executing the algorithms, we can see that the logistic regression model out-performed the naïve Bayes model with 19% more accuracy. Then when trying to improve our results we notice that by removing “A posts” and “B posts” from the logistic regression model we got 2% more accuracy. However, when we removed the variables “A following count” and “B following count” the results did not change significantly. To continue the comparison, we removed the same four variables from our naïve Bayes model and there was an 11% increase in accuracy. This is not surprising because the naïve Bayes algorithm is better for less complex data.

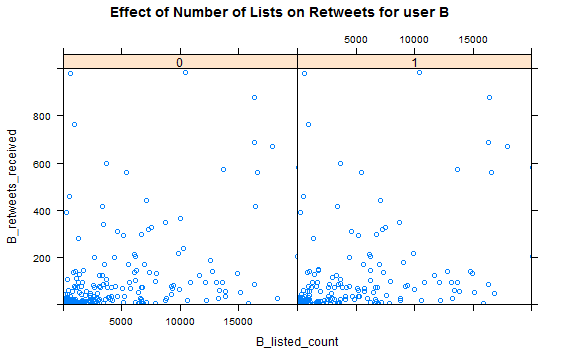
In regards to the original research question, the goals of the project were met. We have established that the logistic regression model was the more appropriate approach for this problem. Our analysis is good starting point for marketing research and social media analytics; however, it does not paint the whole picture. One of the downsides of our dataset is that we did not curate it; therefore, there was a lack of understanding of all the variables. We had no understanding of how the “network features” were measure, so we were missing vital information about how the information could be spread. Moreover, the attributes related to posts, retweets and mentions that were expressed as real numbers could have also been hiding important information. For example, if they were representing the rate at which users did those actions over a specific time period, having that information could have led to a wider range of analysis. Based on our results, we would recommend that future projects of this nature consider more closely the measurements of network interaction and the rates at which relevant Twitter actions are performed.

# Appendix A

The following plots show various relationships between posts and retweets, and list counts and retweets. These same plots were included earlier using only a sample of the data, but here we can see the spread of the full 5500 records.







1. To see plots using the full dataset, see Appendix A. [↑](#footnote-ref-1)