Kacy Espinoza

Vishnu Narayana

CSE 190 Data Mining

Assignment 2

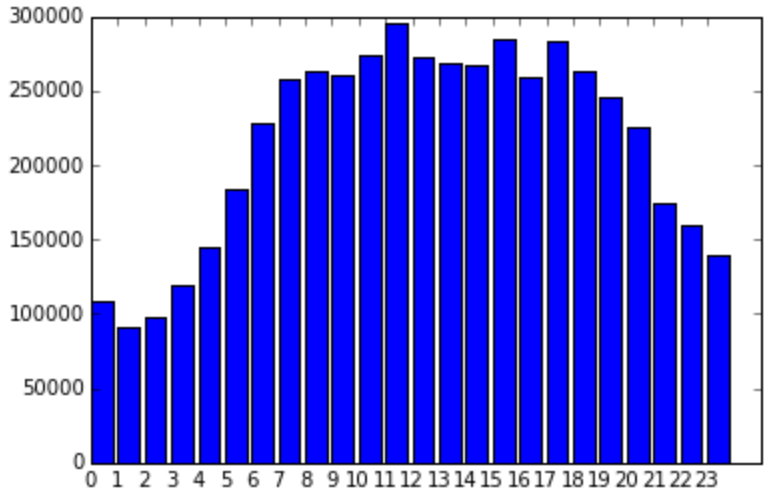
6/2/15

When posting in a social networking environment like Reddit, Tumblr, or Facebook, one might like to know whether their content will cause a discussion (possibly because they would like to bring attention to their profile). Going one step further, they may want to know what kind of discussion. That said, the problem we solve is predicting whether certain social media content will cause a discussion and classifying these discussions as controversial where appropriate.

The dataset we are using is a collection of 132,308 submissions to Reddit where 16,736 of those submissions are unique images. The average number of resubmissions is 7.9 and the timespan these images were submitted to Reddit was from July 2008 to June 2013.

Each submission represents an image a user posted into some community (aka subreddit). When a user creates a submission on Reddit, other users are able to comment and upvote or downvote it. The features collected for each submission in the dataset are things such as the number of upvotes and downvotes, the score (upvotes - downvotes), the time it was posted, the number of comments users posted on it, and the title. There are others, but these are the most relevant for our analysis on the data.

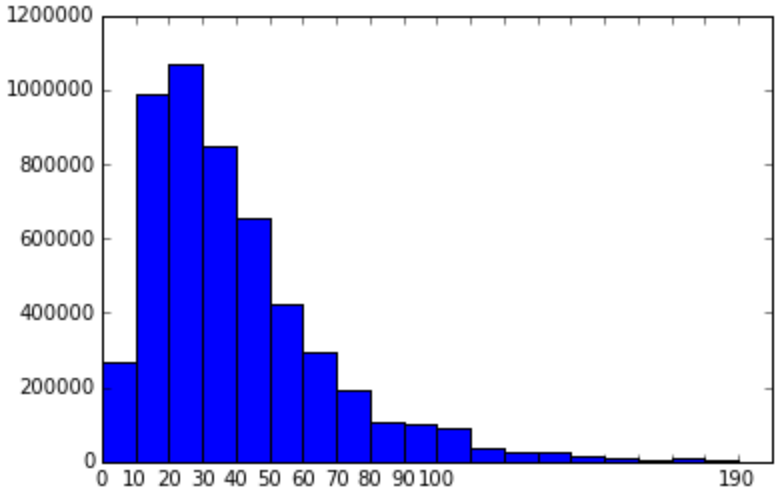
We performed some exploratory analysis on the dataset. The following is a bar graph showing the number of comments (y axis) versus the hour of the day (x axis) where 0 is midnight and 23 is 11:00 P.M.



Looking at the graph, we can see the activity peaks between 11 A.M. and 12 P.M., and is relatively high from 6 A.M. to 8 P.M. So it seems that if a particular user wants to generate a large discussion on their submission, their best bet is to submit their post at 11 A.M. (And they should definitely avoid posting in the wee hours of the night or posting before the birds wake up.) One other thing maybe not too significant but more like a fun fact, is that the 2nd highest peak in the graph happens to be around 5 to 6 P.M. which happens to be when most people get off work.

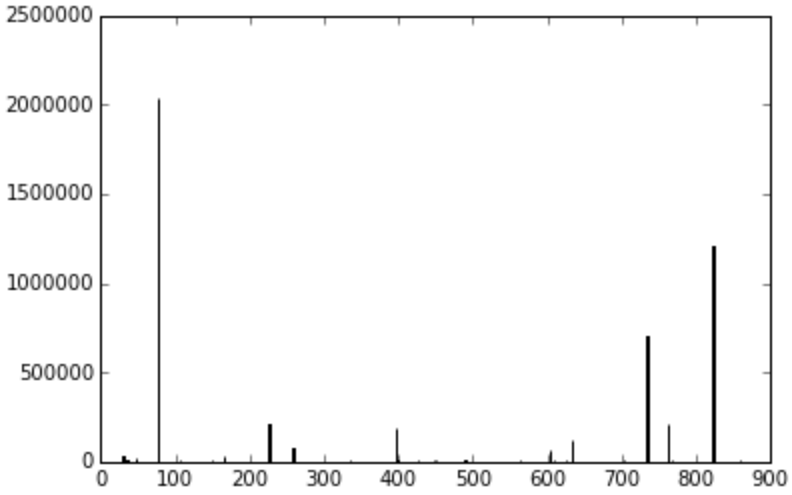
Another aspect to analyze is how discussions are affected by the lengths of titles. At first one might not think title length is important, but looking at the graph below one can see the major impact. This bar graph represents the number of comments (y axis) versus the number of characters in a title (x axis). For example, for titles in the range of 0 to 9

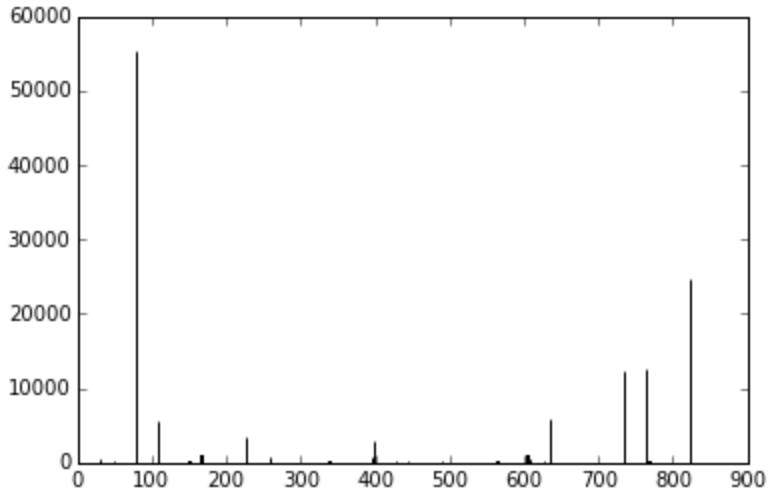
characters in length, the number of comments for those submissions sums up to about 300,000.



Analyzing the graph starting from the left, it is quite clear there's a huge jump in the number of comments from titles that are under 10 characters long to titles that are 10-19 characters. This is most likely because viewers have nothing to "go off of" when the title is so short. There is simply no "spark" for a discussion to start up with such a short title. The ideal title length appears to be between 20 and 29 characters since it produced the most discussion. After 29 characters, the number of comments consistently decreases as title length increases. The reason for this sudden decrease is most likely due to a "too long didn't read" mentality to the viewers. If people are on Reddit, they're most likely there to be entertained, not read novels. Lastly, the longest title was actually about 310 characters in length but there was no change in the pattern so it was sufficient to show the graph up to 190.

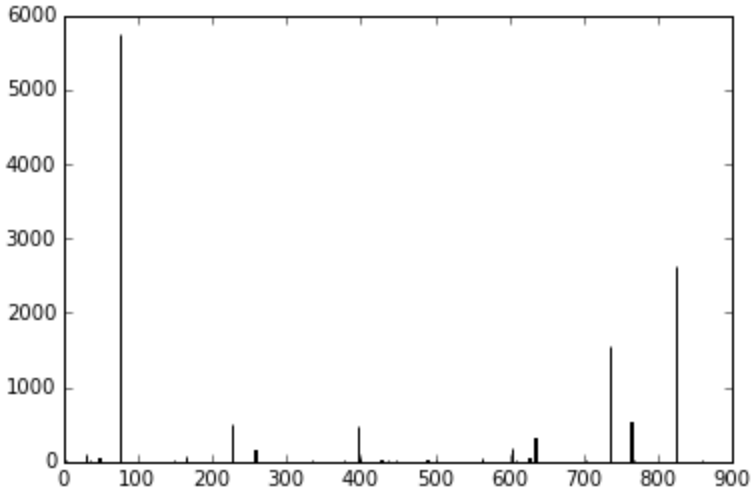
The next two graphs show the number of comments in each subreddit and the number of submissions in each subreddit. (The names of the subreddits are not important which is why they're represented by an id in the x axis for both graphs.) Together, these two graphs simply confirm the notion that the subreddits that receive the most submissions also result in having the most discussion.





But these 2 graphs pose a question: Do the submissions with the most comments exist in the subreddits with the most submissions? It is quite possible that the smaller communities (subreddits) actually result in more comments per submission because each submission is more likely to be seen by everyone in that subreddit. In the more popular subreddits, it is common for a submission to be drowned out by all the other submissions. A smaller subreddit simply has less competition for attention. It may be the case that the only reason a subreddit has the most comments is because it has the most submissions which each contain a small number of comments. For example, subreddit ‘A’ could have a billion submissions with 1 comment each while subreddit ‘B’ could have 10 submissions with 100 comments each. Although ‘B’ has less comments overall, it has the submissions with the most discussion.

Let us look at the next graph which again shows the subreddits in the x axis but with the number of most popular submissions (based on number of comments) in the y axis. In other words, the 13000 most popular submissions (about 10 percent of the total submissions) are distributed to the subreddits they belong to in the graph below.



Well that's exciting! This graph is almost a spitting image of the graph showing the number of comments versus subreddits. There is a direct correlation between being the subreddits with the most comments and subreddits with the most popular submissions. Tieing the 3 graphs together, it appears to be quite clear that if a user wants to generate a large discussion, he should post into the community that receives the most submissions.

This completes our exploratory analysis of how the number of comments generated by users is affected by the time a submission is posted, the length of the title, and which subreddit a submission is posted in. Now we would like to see if these factors play into the number of votes a submission receives.

If a post receives a large number of downvotes relative to the number of upvotes, and the post also receives a lot of comments, then that post is most likely filled with controversial discussion. It is probably safe to assume that as activity increases, the number of upvotes and downvotes increases (just like with comments). Rather than bore you with more graphs to show how downvotes are affected, let's just jump right into the predictor (the "exciting" stuff!).

Just to reiterate, the predictive task we are attempting to conquer is determining whether a submission is controversial or not. But that is not all! We want to determine whether a submission is controversial *before* a user even posts it! This means we are going in blind when it comes to the number of votes and comments. We cannot possibly know those factors before a user even makes the submission.

In order to tackle this beast, this required us to predict 2 or 3 things depending on which approach we wanted to take. We attempted 2 approaches (which I will discuss in a bit). Before we can decide how to evaluate our model at this predictive task, we must ask ourselves what kind of problem is this? Well, it is a classification problem! Is the submission controversial or not? Therefore we will look at the the classification accuracies and error rates of both our approaches.

In order to make sure we were on the right track, we used the average as a baseline. For example, when predicting the number of comments, we broke up the dataset into a training set and test set. We then computed what the average number of comments was on the training set. Then for every submission in the test set, we predicted the number of comments was the average we previously computed. Doing this, the mean squared error for the average was about 17375. Our model's mean squared error for predicting comments ended up being around 16424. We did this same thing for the score, downvotes, and upvotes. Using the averages, the mean squared errors were 215,163, 7,310,144, and 9,449,548 respectively. The mean squared error produced by our model when predicting these same features were 193,433, 6,861,415, and 8,808,558. This humble improvement was accomplished using linear regression. After performing the exploratory analysis, it seemed very easy to visualize what the feature vector would comprise of.

Since we want to classify a submission as controversial before it is posted, we decided to use the time the submission was posted, the length of the title, the subreddit/community it was posted into, the number of times it has previously been submitted, and the number of past submissions that resulted in being controversial. We used the time because depending on whether a submission is made in the middle of the afternoon or in the wee hours of the night, this will affect how much attention it gets (as seen in the exploratory analysis). We used the length of the title because if the title is an appropriate length for people to want to read it, then it is more likely to spark a discussion. The subreddit the image was submitted to was used because we believed some communities may simply tend to argue more than others. We used the number of resubmissions because original content is more likely to get more attention. The number of previous submissions that were controversial was used because chances are if it was controversial before, then it will be controversial the next time it is resubmitted.

Now, what does being controversial mean specifically in terms of the votes and comments? We define a controversial submission as having at least 100 comments, at least 10 downvotes, and at least a 1:3 ratio of downvotes to upvotes.

In the first approach for our model, we predicted the number of comments and the score (the downvotes minus the upvotes). Note that the score is included in each submission as its own feature so there was no need to predict both the upvotes and the downvotes here. The equations used here were:

$$numComments = \theta + \theta_{time} + \theta_{title} + \theta_{subreddit} + \theta_{resubmissions} + \theta_{controversialResubmissions}$$

$$score = \theta + \theta_{time} + \theta_{title} + \theta_{subreddit} + \theta_{resubmissions} + \theta_{controversialResubmissions}$$

where the individual thetas (except for the first one) represent several thetas each. For example, the time theta actually represents 23 thetas for 24 hours in the day. Why 23 and not 24? Because if the time is midnight, then all 23 thetas will be multiplied by a coefficient of 0 which implies midnight so having 24 would be redundant. The title length was categorized into multiples of 10, the subreddits were categorized based on the number of submissions made to them, the resubmissions were categorized by some multiple just like the title length, and same goes for the resubmissions that were controversial.
We then said if the predicted score was less than 200 and the number of comments was at least 100, then the submission could be classified as controversial.

In the second approach we predicted the number of comments, downvotes, and upvotes.

$$numComments = \theta + \theta_{time} + \theta_{title} + \theta_{subreddit} + \theta_{resubmissions} + \theta_{controversialResubmissions}$$

$$upvotes = \theta + \theta_{time} + \theta_{title} + \theta_{subreddit} + \theta_{resubmissions} + \theta_{controversialResubmissions}$$

$$downvotes = \theta + \theta_{time} + \theta_{title} + \theta_{subreddit} + \theta_{resubmissions} + \theta_{controversialResubmissions}$$

Here we said if the number of comments was at least 100, the number of downvotes was at least 10, and the ratio of downvotes to upvotes was at least 1:3, then the submission is controversial.

From the results of our model, we can reach several conclusions. We used both of these approaches when comparing our model to a baseline which made its predictions based on the average values of the training data. The baseline's predictions using the average number of comments and average score on the test set resulted in 0 true positives and false positives, and 60482 true negatives and 5667 false negatives. It had a classification accuracy

of 0.9143 and an error rate of 0.0856. Our model, on the other hand, predicted 420 true positives, 827 false positives, 59655 true negatives, and 5247 false negatives. It had a classification accuracy and error rate of 0.9081 and 0.0918 respectively. The baseline predicted none of the submissions to be controversial, the reason being because the average number of comments was 2. The model did a better job in predicting than the baseline, seeing as it actually marked some items as controversial, while the baseline marked none of them as such, as well as having a smaller MSE. Despite the number of false positives being about double that of the true positives and thus the model having a higher error rate, our model still predicted the number of non-controversial posts quite accurately, and it is definitely preferable to the baseline.

As for the results of running the baseline and the predictor using our second approach (number of comments, downvotes, and upvotes), the baseline's predictions were 0 true positives and false positives, and 60482 true negatives and 5667 false negatives, with a classification accuracy of 0.9143 and an error rate of 0.0856- the same values as the first baseline. In other words, it predicted every post as not controversial again. The predictor, on the other hand, had 419 true positives, 818 true negatives, 59664 false positives, and 5248 false negatives. This lead to a classification accuracy and error rate of 0.9083 and 0.0917. These results are very similar to those in the first approach, albeit small changes on each value. Our first conclusion is that what we have is definitely better than the alternative of predicting with the average.

Looking at the model itself in-depth leads to other conclusions. Consulting with the Professor's paper on the dataset, we took away that the biggest factors for predicting the *score* were the community the content was posted to, the time it was posted, the complexity of the title, and whether the content was being reposted. Our model relied on both the score,

as well as the number of comments of a post to calculate whether it was controversial or not. Despite the above stated factors being satisfactory to predict the number of comments, we decided to include another factor. In the case of a particular type of content having reposts, we calculated how many of those previous posts were controversial. The purpose of this factor was to determine the effect of reposts on controversy. The implementation of the factor into our model resulted in about 20 more true positives, so it definitely helped to better predict whether a post was controversial.

The model succeeded in predicting some controversial posts, but failed to be as accurate as we would have wanted when it came to predicting true positives, as there were many false positives and false negatives in comparison to the number of true positives. I hypothesize that our model might have run into some problems caused by practices of the reddit website itself, such as vote obfuscation, which might have lead to a not as accurate prediction on controversy. Vote obfuscation is the process of artificially adding false upvotes and downvotes to the real number in a popular post in order to prevent certain accounts from knowing what effect they have on them when they vote. This technique, despite inflating the upvotes and downvotes, keeps the score relatively the same. Thus, with popular posts, the true number of upvotes or downvotes, as well as the ratio between them, can be potentially inaccurate, leading to potential problems with the predictor.

Another potential reason for not achieving as many true positives as we would have liked was that we simply do not have the sufficient features to achieve more accurate results. If, for example, we knew the submitted image was related to race(which isn't always obvious from the title alone), then it might be easier to predict a controversial discussion. We also do not know when the spontaneous internet trolls (users who purposefully create controversy for their own enjoyment) are active.

As for pieces of literature related to the dataset we are studying, the most prominent one we know of is one of the Professor's papers. In it, he explored the factors that determine a post's success besides the content itself; namely the *way* in which it's presented. Factors such as the title's originality, the time the content was posted, and the sub-community it was posted to make some of the biggest differences. Linear regression in particular was used to determine the score, but other techniques were used to respond to other factors. For example, images posted multiple times become less popular, and thus an exponential decay function was used to take this into account. The data we used was obtained from Stanford University's website. It's represented as a csv file (the type you'd open with Microsoft Excel), where each row represents a post. Each column represents a particular bit of information on the post, such as a numerical representation of the content itself, its score, what sub-community it was posted to, and the number of comments the post received. Reddit, being a social media website, follows similar rules to other social media sites. Of course, each site has its own quirk. Twitter for example uses a system that focuses on directional connections between different users, and users with other users' content. Facebook focuses on connections between people, and content is presented to each user depending on its popularity compared to the content of other users they share connections with. Reddit, on the other hand, has little focus on connections between users, and focuses on sub-communities users can subscribe to and the content itself. Many studies performed on Facebook and Twitter focus on the connections between people, and the spread of content through them, but not on only content itself. Since the user themself doesn't really matter when it comes to how popular content is, only what the content is and how it is presented. The conclusions reached by the study on Reddit data discussed earlier correlate to what we found about a post's popularity, and we were able to extend that to predict how much discussion a post

would garner and if it was controversial. We used factors such as how many times a post had already been reposted, the sub-community the content was posted to, the time it was posted, and the complexity of the title (evaluated by its length). We also utilized the previous controversy of a post in our model.

If we were to do this again, a feature we would add to our linear regressor is the actual day (Monday, Tuesday, etc) the post was submitted. This is just a guess, but people are probably more likely to argue on days they have to work or be at class in the morning since they may not be in the best mood!