Final Project Report
Group members: John Tejeda, Mauricio Ferrato, Alvin Tang

1. Introduction

The problem that our group is addressing is the issue of who is more influential on social media, in this case Twitter. This is an issue because advertising companies could be wasting money on an individual who they believe to be influential but is not that influential compared to someone else. Also finding out who is most influential on social media can be used to sway the opinions on a mass group of individuals following that influencer, obviously this is great for advertising, but it could also have other uses. The ability to detect who the influencers are is vital for advertising agencies as influencers have a large following of dedicated fans that view their content on a daily basis. Being able to get an influencer to display some type of product or clothing item allows the item to be shown to thousands and or millions of people. The bigger the influencer, the larger the exposure of the item.

For this project we have used three different machine learning algorithms to help us figure out who is the most influential out of a Person A and a Person B. The three algorithms we have used are: Naive Bayes, Logistic Regression, and Perceptron Neural Network. In this report we will show and explain how we applied these algorithms to the public dataset and what accuracies we have obtained from each. We will also select the best algorithm that has the highest accuracy as the algorithm that advertising companies, or really anyone trying to figure out who is most influential should use.

2. Problem Definition and Algorithm

2.1 Task Definition

Our task is to be able to correctly classify who is the most influential person between two people, Person A and Person B. The classification models we use will take as input a set of attributes (number of followers, tweets counts, mentions, etc) from Person A and Person B and will output a 1 or 0 value, 1 if Person A is more influential than Person B and 0 if it's the opposite. With these outcomes, we should be able to see which person is the most influential.

2.2 Algorithm Definition

We are using three different algorithms to classify our data points. These algorithms have been proven to be effective at supervised classification in the past and we thought that they would provide us with good insight for our tasks.

<u>Logistic Regression:</u> Linear regression is a common approach to solving lots of supervised machine learning tasks. They are efficient at predicting numerical values, but fall short when they are used as classifier. A modified version of this model, called logistic regression, solves this problem. Logistic Regression is good for classification because it uses the log function to

squeeze the output of a linear equation into 1s or 0s. The model then follows a similar approach to linear regression to find the minimum distance to predict the closest outcome.

Naive Bayes: A probabilistic machine learning model for classification based mainly on the Bayes theorem. This algorithm is fast and easy to implement and is great for recommendation systems, spam filtering and analysis.

Perceptron Neural Network: This is a algorithm that utilizes multi-layer perceptron which are linear classifiers(binary). These perceptrons utilize a set of weights along with the feature vector to make a prediction. This algorithm can be used to extract patterns and detect trends that are too complex to be recognized by other algorithms.

3. Experimental Evaluation

3.1 Methodology

The criteria we are using to evaluate our method is the percentage of correct classification of whether Person A or Person B is more influential. Our hypothesis is between Naive Bayes, Logistic Regression, and Perceptron Neural Network with one of these algorithms being more efficient than the rest. Our experimental methodology was using each of these algorithms on the same dataset and comparing the correct classification percentages between them. The dependent variable is the correct classification percentage and the independent variable is which algorithm we will use. We used the provided training and test dataset that was gathered from the twitter API. We collected accuracy based on correct labeling (number of correct labels / total) and we also have the number of mislabeled data. We presented it as a percentage and analyze this data by checking which algorithm produces the higher number.

3.2 Results

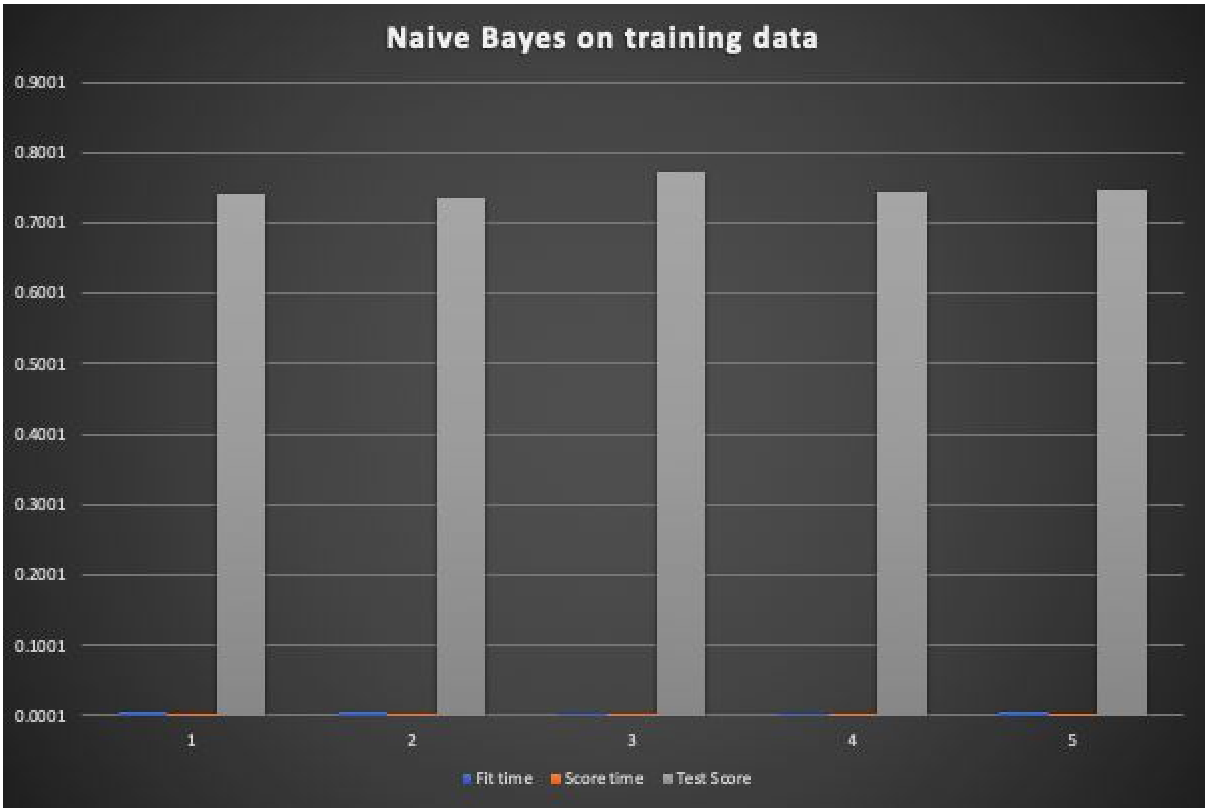
Initially we began to code our experiments in Java, beginning with Naive Bayes. But after finishing the code, we noticed a glaring issue, our accuracy for Naive Bayes was very low and just was not cutting it. We were getting an overall accuracy of 38%, which was abysmal, so we decided to move over to Python and use the Sklearn package for our machine learning algorithms.

Here are the results of our experiment done in Python:

Naive Bayes

Cross Validation 5 fold:

Fit time	0.00467896	0.00492215	0.00370193	0.00360084	0.0045011
Score time	0.00128984	0.0017581	0.00129199	0.00115681	0.00189376
Test score	0.74114441	0.73660309	0.77363636	0.74340309	0.74704277



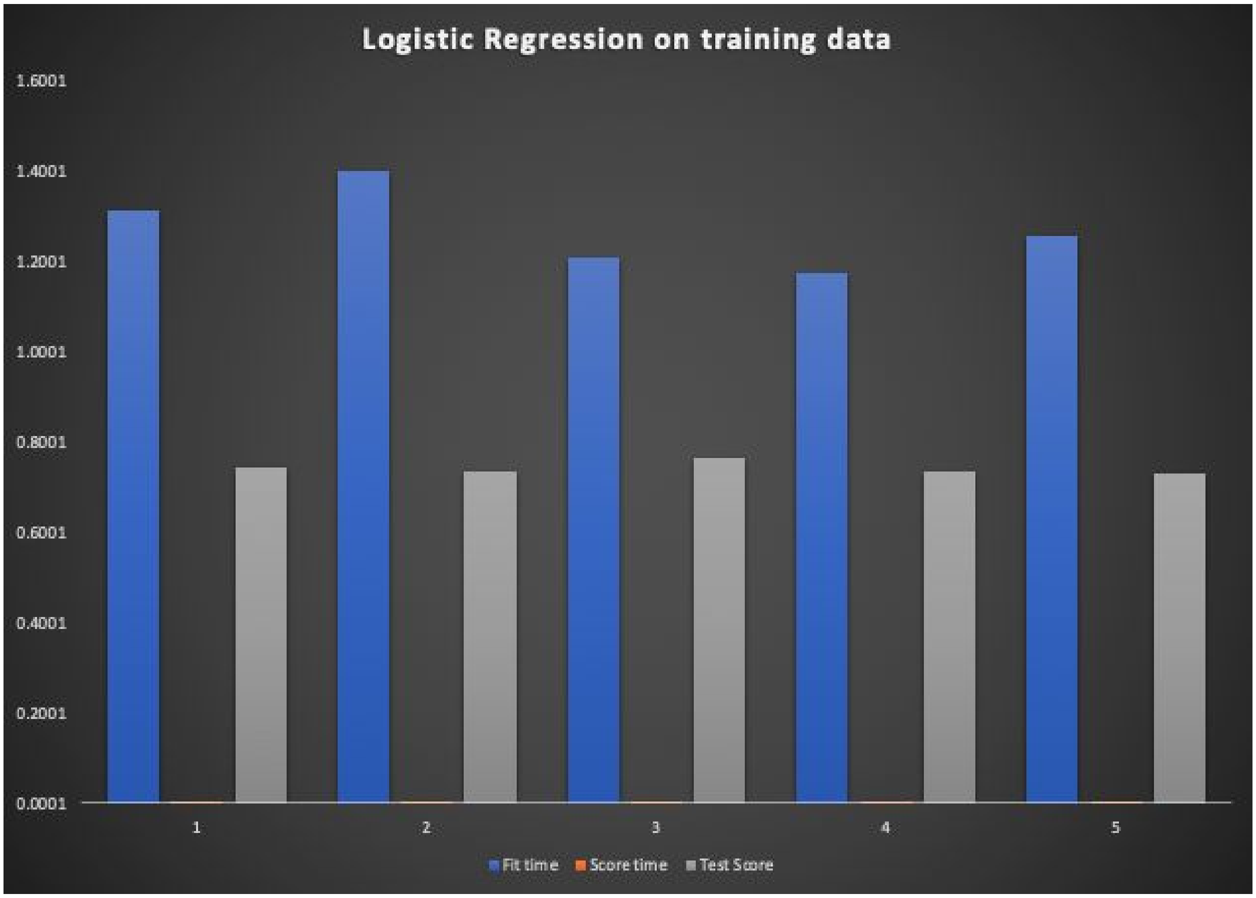
Accuracy on test data:

Naive Bayes	0.883232527

Logistic Regression

Cross Validation 5 fold:

Fit time	1.31662321	1.4033947	1.21183205	1.17781305	1.25866199
Score time	0.0017128	0.00199723	0.00137925	0.00151086	0.002563
Test score	0.74659401	0.73387829	0.76545455	0.73612375	0.73157416



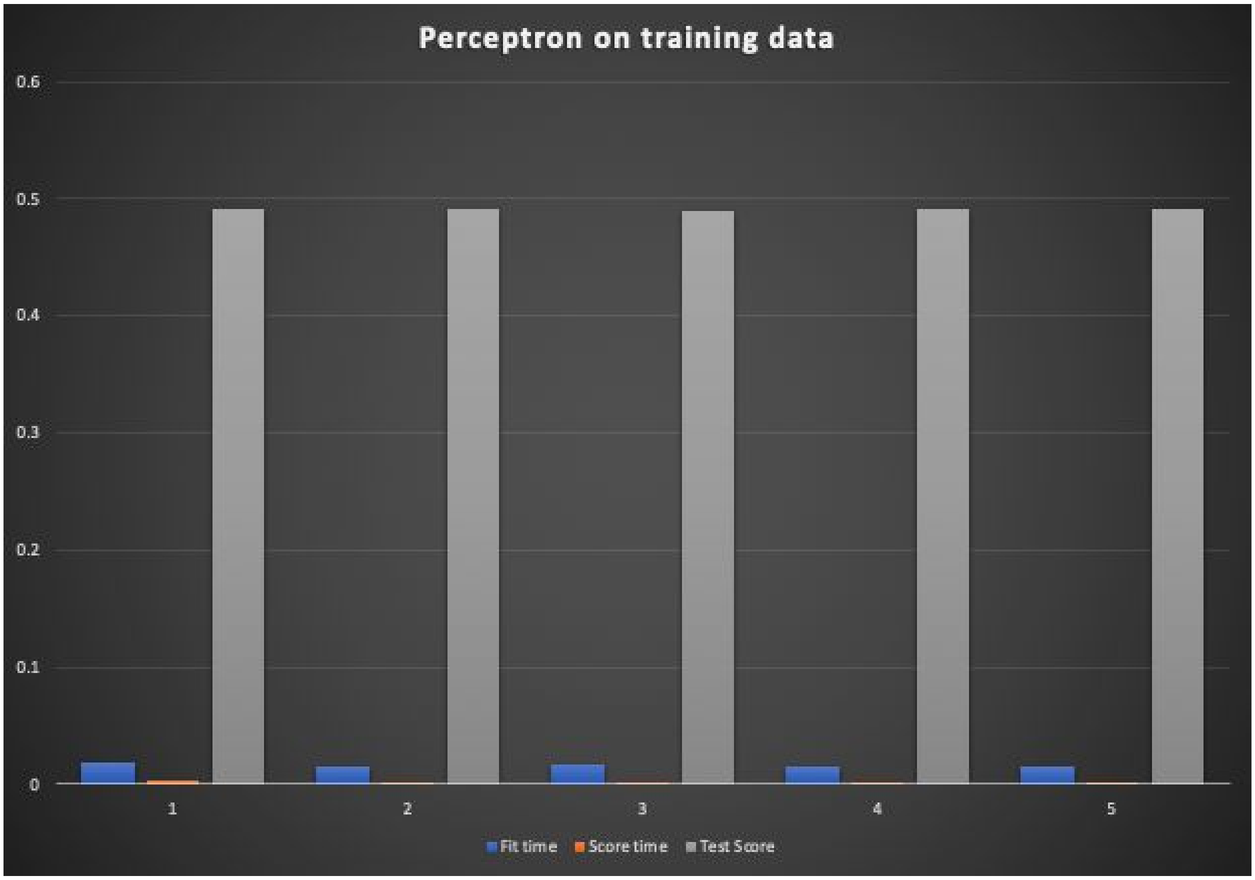
Accuracy on test data:

Logistic Regression	0.85987903

Perceptron Neural Network

Cross Validation 5 fold:

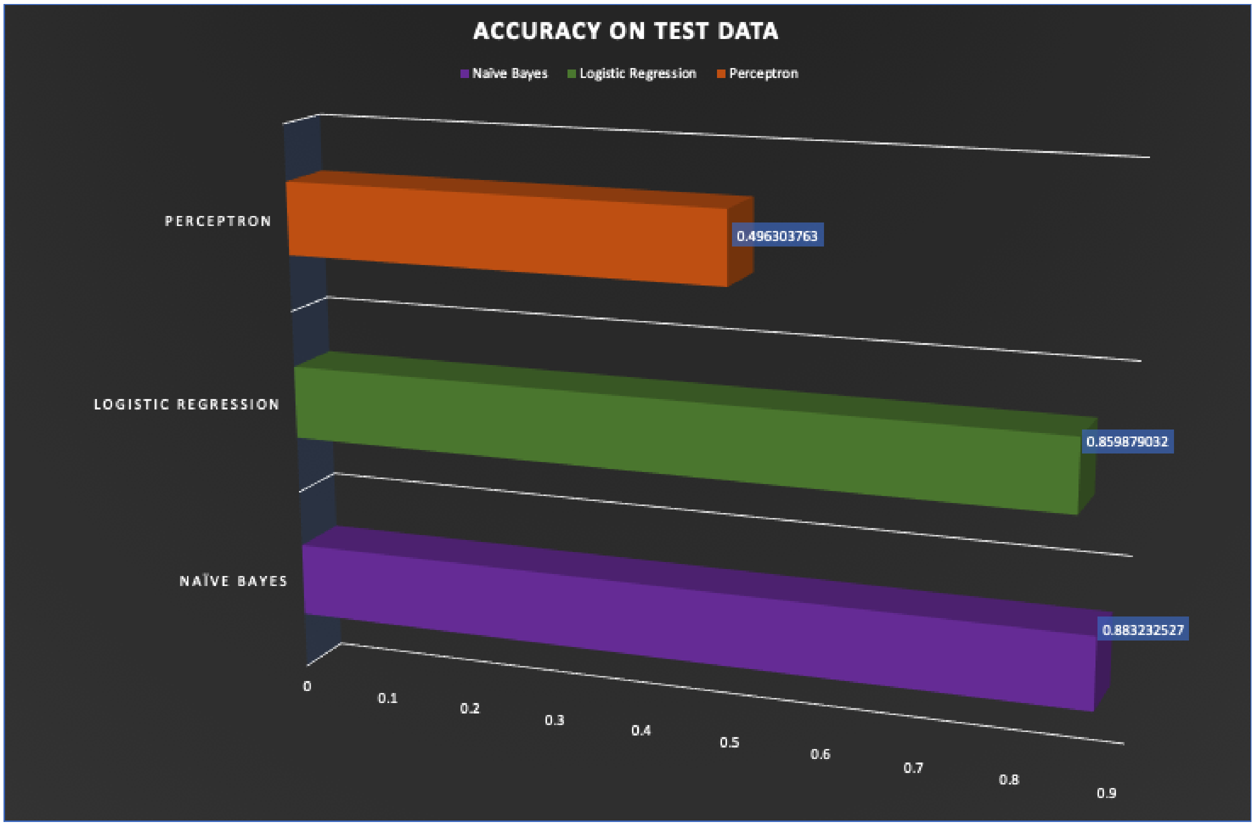
Fit time	0.01784897	0.01521802	0.016119	0.01527715	0.0158689
Score time	0.00283599	0.00139213	0.00139499	0.0014658	0.00143003
Test score	0.49046322	0.49046322	0.49	0.49044586	0.49044586



<u>Accuracy on test data:</u>

Perceptron	0.49630376

All accuracies compared:



Based on the results above, we can conclude that Naive Bayes was the most accurate in classifying whether Person A or Person B was more influential. Though it was almost neck and neck with Logistic Regression having 85.99% and Naive Bayes having 88.32%.

3.3 Discussion

Our original hypothesis was between Naive Bayes, Logistic Regression, and Perceptron Neural Network with one of these algorithms being more efficient than the rest. And after evaluating the results, it is clear that the best algorithm for this task of identifying whether or not person A or person B is more influential is Naive Bayes. We are basing this assumption on how well Naive Bayes performed in contrast to Logistic Regression and the neural network. The reason a Bayesian technique might be better for these kinds of tools is tasks is that Naive Bayes reaches/converges at a faster rate compared to Logistic Regression, and thus we can gather a better solution when using smaller datasets like ours. This paper *"On Discriminative vs. Generative classifiers: A comparison of logistic regression and naive Bayes"* (NG, A. Y., & Jordan, M. I. (2001) confirms that discriminative models like Logistic Regression perform better when training size approaches infinity, but generative classifiers like Naive Bayes reach a solution faster. If our dataset was bigger or we trained it for longer, maybe Logistic Regression could have performed better. When comparing to the Perceptron Neural Network, we saw a weird discrepancy in the results between the team members. Alvin and John were only able to reach the 49% accuracy reported above, but somehow with the same code Mauricio was getting accuracy closer to 84%. We tried to look as to why this was happening but couldn't find an explanation, so as it stands we assumed that Naive Bayes is the better performer of the three.

4. Related Work

Our task was part of a Keggle competition, so looking at the members that competed we were able to find other people's approach at solving this problem. One approach a person had was using the Keras library to create a neural network classifier, Grewal, B. (2018, May 31). His approach was different to ours because we used Sklearn instead and looked at different, less complex, classification models. Our method was more interesting because we compared different techniques instead of only looking at a single one. Another approach we saw was a person used different ensemble networks such as random forest and gradient boosting, O. (2019, February). Their approach saw better performance because the ensemble models are combination of smaller weaker models (similar to the ones we used). After looking at their project, we thought it would be a good idea to try ensemble models for future work.

5. Future Work

Currently our method does not account for when both persons share the same amount of influence. This could become a problem when the model tries to classify two people who share the same amount of followers, mentions, etc. A solution to this could be giving predetermined importance to certain characteristics a follower could have, or bringing other attributes into account such as demographic influence, or targeted influence to certain market fields (sports, fashion, video games, etc). Future work that could be done on these models would be to use ensemble models to see if there could be an improvement to the predictive performance of the models. Another technique that could also improve the models would be to look into preference learning models. These models base their preference on certain rankings and are used for deciding when one thing is better than another. This sounds like a better fit our task and would be something worth looking into.

6. Conclusion

To conclude, we were able to narrow the three algorithms that we used down to one ultimate algorithm that would be the best at predicting influencers. In our case, this algorithm ended up being Naive Bayes, as it provided us with the best predictive performance based on correctly classifying the test cases. We think that these models, with some improvements, could be useful to marketing and advertising professionals, and that this could improve the way advertisement is seen online.

Bibliography

NG, A. Y., & Jordan, M. I. (2001). *On Discriminative vs. Generative Classifiers: A comparison of logistic regression and naive Bayes* (Research Paper. University of California, Berkeley). Neural Information Processing Systems.

Influencers in Social Networks. (2013). Retrieved May 22, 2019, from https://www.kaggle.com/c/predict-who-is-more-influential-in-a-social-network/data

Grewal, B. (2018, May 31). Bigrewal/Influencers-in-Social-Networks. Retrieved May 22, 2019, from https://github.com/bigrewal/Influencers-in-Social-Networks

O. (2019, February). Bayesian Optimized LGBM. Retrieved May 22, 2019, from https://www.kaggle.com/sdoliver/bayesian-optimized-lgbm

Gandhi, R. (2018, May 05). Naive Bayes Classifier. Retrieved May 22, 2019, from https://towardsdatascience.com/naive-bayes-classifier-81d512f50a7c