Classification Models and Dimensionality Reduction for Detection of Twitter Bots

Kate Duffy and Burhan Sadliwala

Abstract

Automated social media accounts, known as bots, have earned increasing attention for their role in spreading misinformation and manipulating politics and the stock market. In this project, we compare the accuracy of several bot-versus-human classification models and investigate the effect of dimensionality reduction on model accuracy. We use cresci-2017 dataset and perform preprocessing using Principal Component Analysis. We evaluate K-Nearest Neighbors, Logistic Regression, Naive Bayes and Support Vector Machine. We find that Dimensionality Reduction (PCA) should be done for Ridge Regression since it increases accuracy for Ridge Regression. However, we recommend PCA should not be implemented for KNN, SVM or Naive Bayes since PCA decreases the accuracy and increases the running time of the entire process. Gaussian Naive Bayes gave the highest accuracy (>99%) whereas KNN gave the highest performance considering running time.

1 Introduction

In an increasingly digitized and interconnected world, social networking sites are a growing source of news and information. Automated accounts, referred to henceforth as bots, may represent between 9% and 15%, of the active Twitter population [1].

Bots on social media platforms have increasingly been implicated in influencing political opinion, spreading conspiracy theories and manipulating the stock market. One publicly available service, BotOrNot, offers to detect fake Twitter followers using 1000 features extracted from account meta data and tweet content [2]. In this project, we evaluate classification methods to identify automated twitter accounts.

2 Related Work

Manual review of Twitter users is prohibitively labor intensive and detecting bots can be difficult even to humans. Several studies have addressed the challenge of automatic bot detection. Varol et al. process data pulled from the Twitter API to produce over 1000 features. They use four classification models and find random forest performs best [1]. Random forest is also used by the service BotOrNot, now called Botometer [2].

3 Technical Approach

3.1 Dataset

Our real-world dataset is provided in .csv format by Cresci et al. [3]. The dataset consists of 1000 traditional spambot accounts and 377 genuine user accounts. The raw dataset includes over 25 metadata features per account as well as tweet text and tweet features. From the available features, we perform an initial manual feature selection based on (1) completeness and (2) intuitive relevance to the problem. We discard features that were missing for some users and features with no relevance to our classification, such as ID number. After this selection, we have 13 features. Of the 13 features, 11 are discrete variables and 2 are categorical. In data preprocessing, features of tweets are aggregated as an average for each individual user, e.g. average number of retweets User 1 recieves. Additionally, all features are rescaled to take values in the range [0, 1].

Our set of 13 features can be summarized as the following:

Twitter account features:

statuses_count
followers_count
friends_count
favorites_count
listed_count
verified
geo_enabled
account_age

Tweet features:

favorite_count
retweet_count
number_hashtags
number_urls
number_mentions

3.2 Principcal Component Analysis.

PCA converts a set of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components.Eliminating components that contain low variation, decreases the dimension of the observation without losing significant amount of information.

By inspection of the data, we realized that some attributes viz. followers_count, friends_count and favorites_count measure related properties and so are redundant. To reduce this redundancy and describe each account with less properties, we used PCA.

The reduced data is obtained by the following:

$$y^{(i)} = U^T(x^{(i)} - \bar{x}) \tag{1}$$

$\bar{x}$ and U are computed as:

$$\bar{x} = \frac{1}{N}\sum_{i}^{N} x^{(i)} \tag{2}$$

$$x = U_x \sum_{x} V_x^T \tag{3}$$

$$U = U_x(:, 1:d) \tag{4}$$

where d is the reduced dimension of the data.

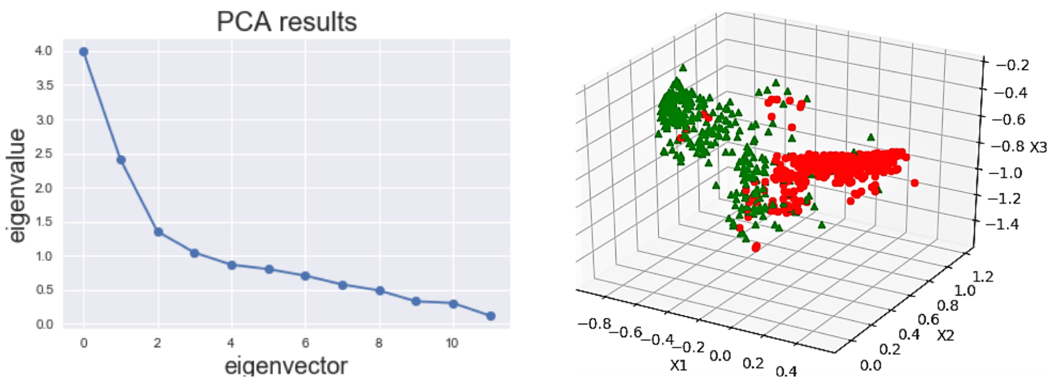

Figure 1: PCA results and visualization.

3.3 Methods

We evaluate the performance of K-Nearest Neighbors, Logistic Regression, Naive Bayes and Support Vector Machine with dimensionality reduction.

K-Nearest Neighbors. KNN, is a lazy, non-parametric model, meaning there are no discrete training phase and no assumptions need to be made about the structure of the data. The user chooses a positive integer k. The k closest entries to a point are identified and the common classification among the entries is assigned to the point. We implement KNN with k values as odd integers in the range of 1 to 17. Classification accuracy is calculated as the percent of sample points which are correctly classified as bot or non-bot by this method.

Logistic Regression. In logistic regression, the log probability of a discrete dependent variable is expressed as a linear combination of independent variables. The logistic loss function, Equation 5, is asymptotic and penalizes increasingly large errors at a constant cost. Ridge regression is used to discourage overfitting. The cost function for Ridge regression, Equation 6 includes a weight regularization term, λ. We test values of λ in the set [0.000001, 0.00001, 0.0001, 0.001, 0.1, 0.5, 1, 1.5, 10] using k-fold cross validation with k = 10. The logistic regression models are trained using gradient descent with a learning rate of 0.001.

$$J = -y\log(p) - (1-y)\log(1-p) \tag{5}$$

$$J = -y\log(p) - (1-y)\log(1-p) + \lambda * \frac{1}{2}||\theta||_2^2 \tag{6}$$

Naive Bayes. Naive Bayes, is a generative model that is called naive because it (potentially wrongly) considers features to be independent of each other. Naive Bayes uses likelihood and probability values to calculate the posterior probability. The number of parameters required is linear in the number of features in the problem. MLE training can be done by evaluating a closed form solution rather than using an expensive iterative approach as used in Logistic Regression or SVM.

Since our training data contains continuous attributes, we use a Gaussian Naive Bayes implementation. The data is first segmented by class and then mean and variance is computed for each attribute in each class. The probability function is then computed by plugging in the values in the equation for a Normal distribution 7.

$$P(x,\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{7}$$

Support Vector Machine. SVM models data as existing in some p-1 dimensional space, where each data point has p features. The objective of the SVM is to learn a separating hyperplane that best divides the classes.Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest data point of any class, since as large the margin gets, the lower the training error.

We implement the Sequential Minimal Optimization (SMO) algorithm as it breaks down the large quadratic programming (QP) optimization problems into smaller problems solved analytically which reduces the space and time complexity drastically. We train the SVM model for a range 1 to 1000, for the hyperparameter C.

4 Experimental Results

Among our four selected models, Naive Bayes resulted in the highest classification accuracy. Overall, using a lower-dimension feature space had the expected effect of reducing computation time, but with one exception resulted in poorer classification accuracy. Results are summarized in Table 4.

Classification model	Full feature space (d=13)		PCA (d=3)	
	train accuracy (%)	test accuracy (%)	train accuracy (%)	test accuracy (%)
K-Nearest Neighbors	99.7	–	97.4	–
Logistic regression	63.5	63.3	62.7	62.3
Ridge regression	90.2	88.8	90.5	89.5
Naive Bayes	99.2	100	94.4	92.0
Support Vector Machine	99.2	99.6	95.9	95.2

Table 1: Summary of results for classification models. Accuracy is presented as percent of accounts correctly classified as human or spambot.

K-Nearest Neighbors. For K-Nearest Neighbors, the highest accuracy resulted for a k value of 1, as reflected in Figure 2. Accuracy was higher for the full dimensionality than for the PCA product for all values of k, and for both training sets accuracy decreased as k increased. A k value of 1 results in the highest accuracy, but increases the risk of overfitting, as it makes the classification of each sample point highly sensitive to outliers. Selecting a greater value for k would increase the robustness of the model.

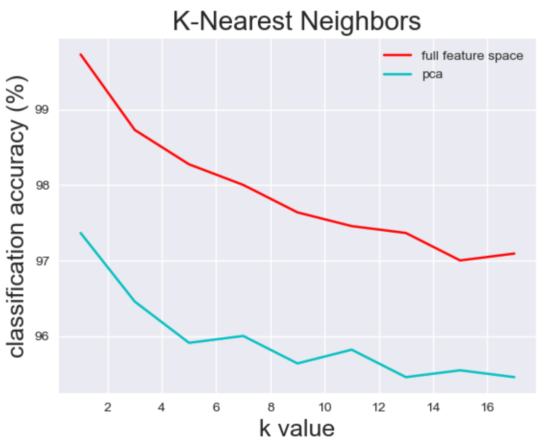

Figure 2: K-Nearest Neighbors results for varying selections of k.

Logistic Regression. This method achieved the lowest classification accuracy among the four methods tested. Results for varying regularization parameter values are presented in Figure 3. Cross-validation accuracy was maximized for the full feature dataset with with C $= 1E - 5$ then generally decreased as C increased. For the reduced dimensionality dataset cross-validation accuracy was maximized for C $= 1E - 6$. Interestingly, Ridge regression improved classification accuracy by nearly 30% across datasets. Ridge Regression is also the only method we tested for which dimensionality reduction improved classification error.

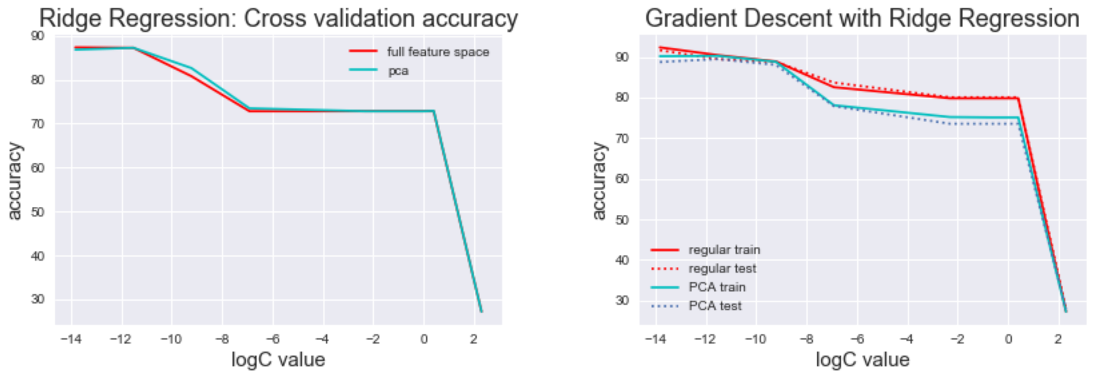

Figure 3: Ridge regression results with varying weight regularization. On the left, k-fold cross validation accuracy with k=10. On the right, training and testing error. Values of regularization parameter, C, are plotted on a logarithmic scale.

Naive Bayes. Comparing other models with the Gaussian Naive Bayes, we realized that this model is pretty accurate.Accuracy for the reduced dimensional data is much lower than for full feature space. In our tests, we obtained a >99% accuracy for the full dimensional data but the accuracy reduced by

8% for PCA data.

Support Vector Machine. By inspection, our dataset with d=3 is not perfectly linearly separable. Likewise, we do not have reason to believe the the dataset is linearly separable in the full feature space. The SVM model took the longest time to train and the training time increased as we increased the value of C. We obtained the lowest mean training and test error for C=256. For the PCA data, the classification error did not vary much with respect to the hyperparameter C.

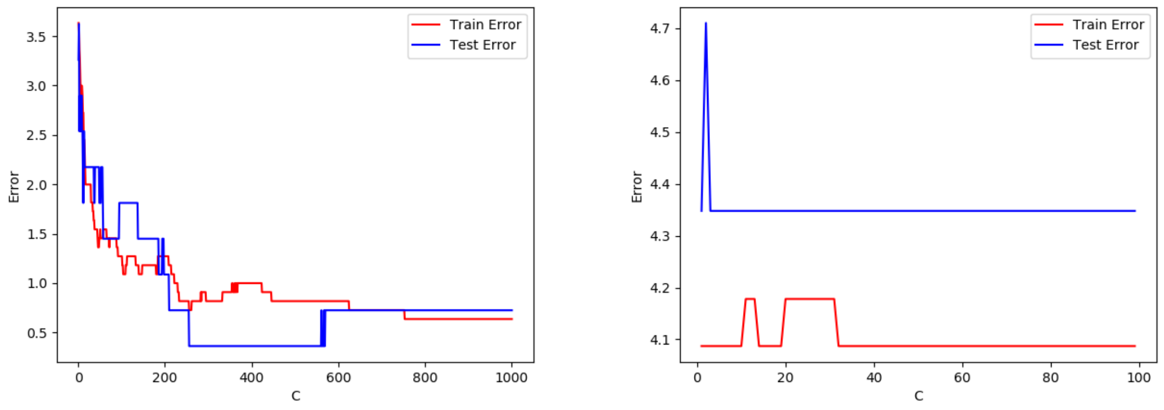

Figure 4: SVM Results: Training and Test Error with varying values of C from 1 to 1000

5 Conclusion

Based on our results, we would not recommend using dimensionality reduction for this problem unless computational cost is a limiting factor. For all models tested, dimensionality reduction via principal component analysis decreased classification accuracy, with the exception of Ridge Regression. The SVM, KNN and Naive Bayes models provided more than 99% accuracy for full feature space, however, by considering running time, KNN gave the highest performance over other models.

6 Participants Contribution

The participants in this project are Kate Duffy and Burhan Sadliwala. The original dataset was preprocessed by Kate, including selecting relevant features and aggregating tweet data by user. PCA was implemented by Burhan, as was division of the dataset into train and test sets. Kate implemented KNN and Logistic/Ridge Regression. Burhan implmented Naive Bayes and SVM. Results for each model were prepared by the respective team member.

References

[1] Onur Varol, Emilio Ferrara, Clayton A. Davis, Filippo Menczer, and Alessandro Flammini. Online human-bot interactions: Detection, estimation, and characterization. *CoRR*, abs/1703.03107, 2017.

[2] Clayton A. Davis, Onur Varol, Emilio Ferrara, Alessandro Flammini, and Filippo Menczer. Botornot: A system to evaluate social bots. *CoRR*, abs/1602.00975, 2016.

[3] Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. The paradigm-shift of social spambots: Evidence, theories, and tools for the arms race. In *Proceedings of*

the 26th International Conference on World Wide Web Companion, WWW '17 Companion, pages 963–972. International World Wide Web Conferences Steering Committee, 2017.