MIE1624 Assignment 2

Twitter Sentiment Analysis

Sanjif Rajaratnam || 999091986 || rajara24

1.0 Classification Algorithms

The two types of classification algorithms that were chosen for this assignment were the Logistic regression algorithm and the Naïve Bayes algorithm. Both algorithms are well known supervised machine learning algorithms. Linear regression is not appropriate for classification because it is difficult to fit the data well. The SVM algorithm was not chosen because it took a very long time to train in practice.

1.1 Logistic Regression Algorithm

The Logistic regression algorithm is often used to predict binary categorical labels [1]. Logistic regression models the probability that the input data belongs to a certain category. The goal is to predict the probability that the given data point belongs to class "1" or class "0" [1]. This is done by trying to learn the Sigmoid function seen in **Equation 1** [1].

$$\sigma(z) = \frac{1}{1 + \exp(-z)} \quad (1)$$

This function takes an input of $z = \theta^T x$ and returns a value between the range of 0 and 1 that can be interpreted as a probability [1]. Here x is the input data. The goal of learning is to find θ such that the probability that x belonging to class "1" is large, i.e. $\sigma(\theta^T x)$ is near 1, if x belongs to class "1", and the probability that x belonging to class "0" is small, i.e. $\sigma(\theta^T x)$ is near 0, if x belongs to class "0" [1]. For a given set of training data with labels: $(x_i, y_i): i = 1, \dots, m$, the optimal θ can be found by minimizing the following cost function [1]:

$$J(\theta) = -\sum_i \left(y_i \cdot \log\left(\sigma(\theta^T x_i)\right) + (1 - y_i) \cdot \log\left(1 - \sigma(\theta^T x_i)\right)\right) \quad (2)$$

1.2 Naïve Bayes Classification Algorithm

The underlying principle behind the Naïve-Bayes algorithm is the Bayes' theorem [2]:

$$P(c|x) = \frac{P(x|c) \cdot P(c)}{P(x)} \quad (3)$$

Here, $P(x|c)$ is the probability of the data given the class, $P(x)$ is the prior probability of the data, $P(c)$ is the prior probability of the class, and $P(c|x)$ is the posterior probability of the class given the data [2]. The Naïve Bayes algorithm uses this theorem while assuming that the data given the class is conditionally independent [2]. The learning of the classes is fast because only the probability of each class (**Equation 4**) and the conditional probability of inputs belonging to each class is used to built the model (**Equation 5**) [3].

$$Class: P(class = 1) = \frac{count(class = 1)}{count(class = 1) + count(class = 0)} \quad (4)$$

$$Conditional: P(x = C|class = 1) = count\ (\ x = C\ \&\&\ class = 1) \mid count(class = 1) \quad (5)$$

Here C represents some constant, and P represents probability.

2.0 Results

2.1 K-Fold Cross Validation

The algorithms were all implemented via the sci-kit learn libraries in Python. To determine the ideal algorithm, a greedy grid search technique was used to built models with various parameters. The models were than evaluated using the K-Fold cross validation technique with 10 folds. The models built with the optimal parameters were then used for analysis. The statistical results can be seen in the error plot in **Figure 1**. Here it is quite clearly seen that the Logarithmic model had the best mean accuracy with the lowest standard deviation of all three models from K-Fold cross validation. This means that it will have better performance more consistently than the other two models.

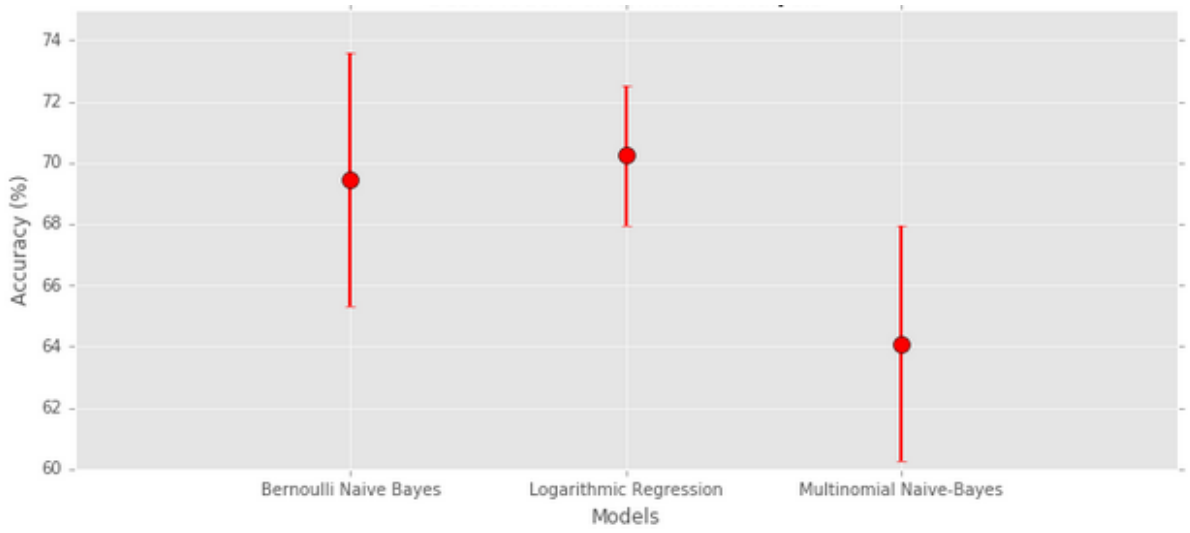

Figure 1: Optimal Model Performance Analysis

2.2 Train-Test-Split

Then each model was built with the optimal parameters from the grid search. The models were all trained with 67% of the classified data and tested with the remaining 33%. Then the accuracy and misclassification rate of each model was calculated. The accuracy is defined by **Equation 6**, and the misclassification rate is defined by **Equation 7**.

$$accuracy = \frac{TP + TN}{TP + TN + FN + FP} \qquad (6)$$

$$misclassification\ rate = \frac{FP + FN}{TP + TN + FN + FP} \qquad (7)$$

Here, TP (True Positive) represents the times the model predicted positive and it was positive, TN (True Negative) represents the times it predicted negative and it was negative, FP (False Positive) represents the times it predicted positive but it was negative, and FN (False Negative) represents the times it predicted negative but it was positive. The results of this analysis can be seen in **Table 1**.

Table 1: Train-Test-Split analysis

	Logarithmic Regression	**Multinomial Naïve Bayes**	**Bernoulli Naïve Bayes**
Accuracy	76.4%	72.1%	73.8%
Misclassification Rate	23.6%	27.9%	26.2%

Again, the Logarithmic model had the highest accuracy with the lowest misclassification rate. Therefore, it can be concluded that the Logarithmic model is the ideal algorithm for this problem.

3.0 2015 Canadian Election Sentiment Analysis

The chosen Logarithmic regression classifier was then used to give scores to the unclassified set. A party was also assigned to each tweet if mostly one party's related words showed up within the tweet. These party related words were found by looking up party related hashtags on Twitter. The results of the analysis can be seen in **Table 2**.

Table 2: Party Tweets Sentiment Analysis

	Conservative	Liberal	NDP
Mean Score	0.608163	0.783557	0.740854
Score Standard Deviation	0.488659	0.412166	0.438836
Count	490	596	328
Tweet %	34.65%	42.15%	23.20%

The Liberal party had the most buzz with about 42.15% of the tweets, and the NDP party had the least buzz with about 34.65% of the tweets. The Liberal party was the most liked with a positive-negative tweet ratio of 78.36%, and the Conservative party was the least liked with a positive-negative tweet ratio of 60.82%. The Liberal party also had the lowest standard deviation in their score whereas the Conservative party had the most.

The Liberal party and the NDP party had similar distributions of scores but the Liberal party was more liked and had almost twice as many party related tweets. Overall, it shows that the Liberal party was the most likely to win the election. This is reflected in the actual election as Justin Trudeau won and the Liberal party also had the most seats [4]. The Conservative party got more positive attention (tweet count) than the NDP party even though its positive-negative ratio was worse. This is also reflected in the election as Steven Harper got the second most votes but the Conservative party lost the most seats during the election [4]. The NDP party had the least tweets overall, and they got the least votes and the least seats of the three parties in the election [5].

4.0 References

[1] http://ufldl.stanford.edu/tutorial/supervised/LogisticRegression/
[2] http://www.saedsayad.com/naive_bayesian.htm
[3] http://machinelearningmastery.com/naive-bayes-for-machine-learning/
[4] https://en.wikipedia.org/wiki/Results_of_the_Canadian_federal_election,_2015