Charles Bolton
5/15/2020
Machine Learning

Spam Classification with Naive Bayes

This program is actually a relatively short and simple program. Here's what I did (you can follow along in the code if you wish):

1. After loading the data, shuffle and split the data into training and test sets based on the size of the data and the known ratio of spam/no spam.

2. Separate the spam and no spam parts of the training set and then calculate the priors.

3. Calculate the means and standard deviations for the spam and no spam parts (the positive and negative parts of the training set). Adjust the standard deviations in case of divide by zero.

4. Loop over the rows of the test data calculating the logarithmic pdf for each of the data examples in the test set.

5. Classify each data example based on whether or not the probability that it is spam or not

6. Add this prediction to the confusion matrix.

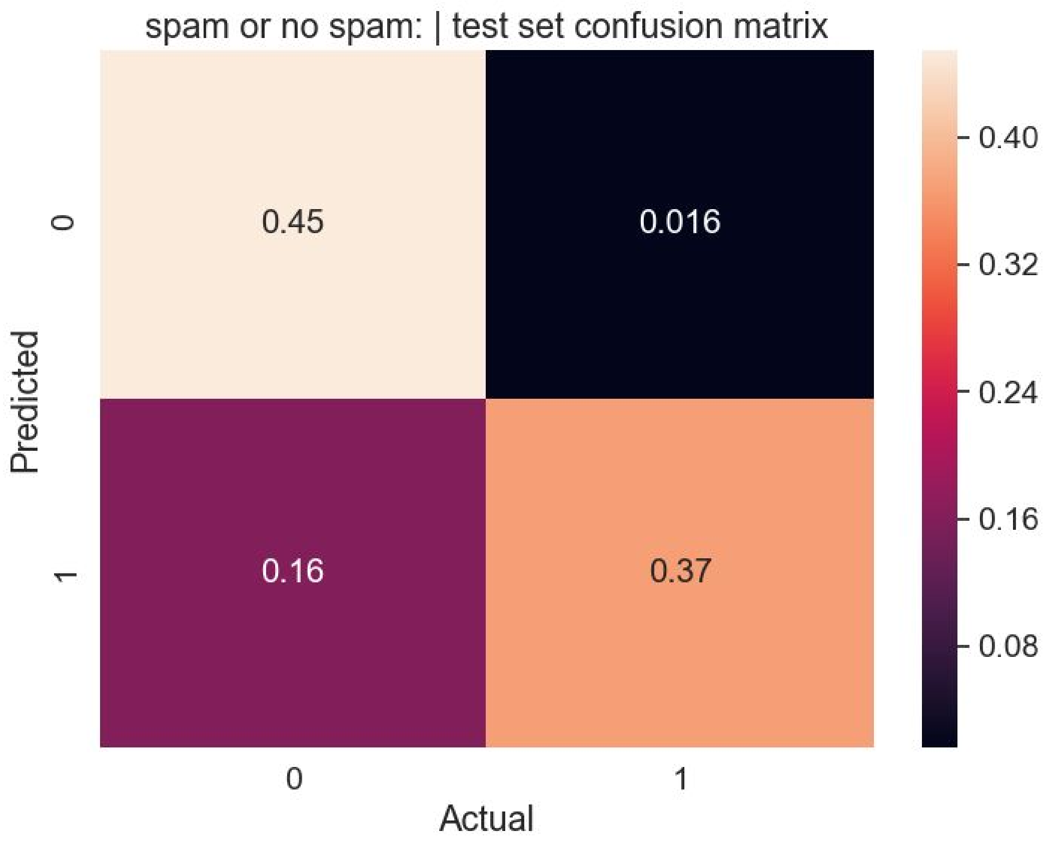
7. After looping across all rows in the test set, calculate the accuracy, precision, and recall, print them, and plot the confusion matrix.

Example output:

```
Confusion Matrix:

[[0.45498043 0.01609395]
 [0.16093954 0.36798608]]

accuracy  = 0.8229665071770336
precision = 0.9658356417359187
recall    = 0.7387005649717515
```



This is an example of the average output for this program. You can feel free to run it as well. The average accuracy was around 81-84%. Precision was often close to 95% while recall was not as good, usually around 70-75%.

Do you think the attributes here are independent, as assumed by Naïve Bayes?

It's obviously pretty unlikely that these words are independent. The features we assumed to be independent include the frequency of words such as "business," "credit," "project," and "meeting." These words have word vecs (if you will) which are probably not completely orthogonal, and would likely appear near each other in various texts and contexts. It should also be no surprise that one cannot really claim that any word or frequency of occurrence in a given text thereof is ever truly independent of some other word, even words in other languages. This is because words in and of themselves mean nothing, and it is only by their surroundings, syntactic and semantic couplings, and other cues that they carry any meaning at all. So insofar as we are considering words to be our primary data points, the assumption that they are independent is clearly absurd and totally naive indeed.

Does Naïve Bayes do well on this problem in spite of the independence assumption?

Of course, despite the naivete, Naïve Bayes performs pretty well on this data set, as shown by the above results. While these results show that this method performs pretty well on the spam data, it still makes a number of mistakes, classifying spam as non-spam 15% of the time, which in practice is not ideal, because then more than 1 out of ten spam emails are mistaken for real emails But as a demonstration of the Bayesian predictive power, it does show that it works pretty well. For example, it was really good at identifying real emails, hardly ever classifying a real email as fake; it just wasn't as good at identifying spam itself, which is clearly pretty important. But at least it wasn't likely to throw away important emails. It might be better to get a few spam messages here and there than to get all your emails thrown away.

Anyway, Bayesian classifiers are in fact used in spam filtration, so this particular algorithm may not be perfect, but it proves that it's possible.

Speculate on other reasons Naïve Bayes might do well or poorly on this problem.

One advantage to this model is that, unlike MLPs or certain other predictive models, this approach requires fewer examples to train on in general, which means that the process of data collection and labelling for a given problem using Bayesian classification is less arduous. This model may perform poorly if it were given an example which contained zero occurrences of a given word, since then the probabilities, when multiplied, would equal zero and would throw the model out of order. However, this could be fixed using Laplace's principle of indifference, for example.