Homework 3: SVMs & Feature Selection

CS 545: Machine Learning, Winter 2017

Ben Wilson

Introduction

This homework assignment focuses on support vector machines (SVMs) and feature selection by using an SVM classifier to determine whether or not an email is spam based on 57 various features. The data for this assignment comes from https://archive.ics.uci.edu/ml/datasets/Spambase and needed to be processed prior to being used in the SVM. Once the SVM was working properly, a ROC curve was generate to display the results and subsequent experiments on feature selection were performed.

Experiment 1: Training the SVM

Using the scikit-learn Python module, an SVM object was instantiated and passed the training set of the pre-processed spambase data. Once the model was trained, the test data was passed to the `predict` method and the following results were obtained:

- Accuracy = 92.65%
- Precision = 91.57%
- Recall = 89.20%
- False Positive Rate = 5.17%

A score vector was created by taking the dot product of the weights with the test samples. The signum function was applied to the score vector at 200 evenly spaced threshold and evaluated for true and false positive rates. The results of this is displayed in the ROC curve in Figure 1 (page 2).

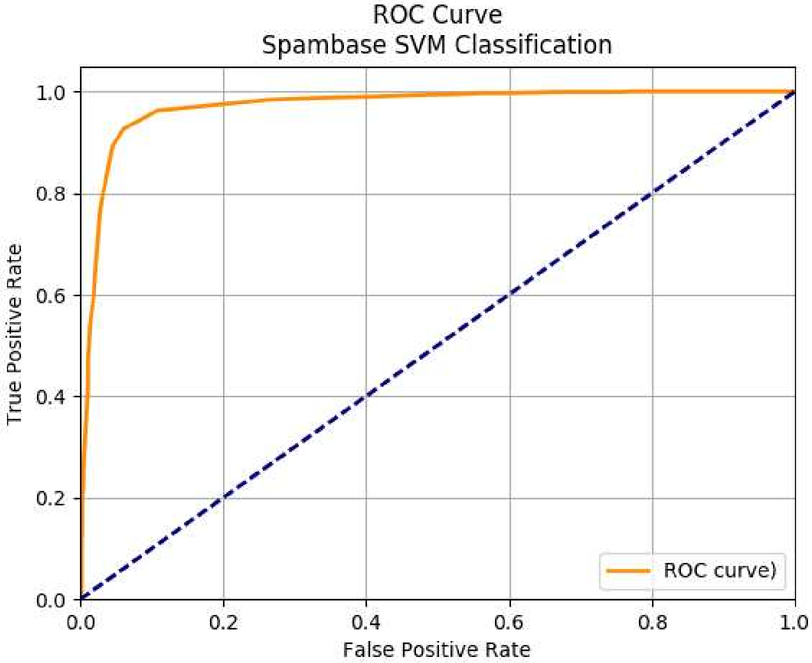

Fig. 1: ROC curve of the scikit-learn SVM on the Spambase dataset

Experiment 2: Feature Selection with Linear SVM

In experiment 2 we were asked to train an SVM with an increasing number of features starting with the one with the largest weights. After running several trials, the top five largest weights were associated with the following features:

1. word_freq_george
2. word_freq_hp
3. word_freq_cs
4. capital_run_length_longest
5. word_freq_edu

These results were not expected and had to be double checked to verify that they were correct. It can make intuitive sense that an email containing the recipients name (in this case, George) multiple times would probably be spam. It also makes sense that a long streak of capital letters would be a strong indicator of spam. However, I personally, couldn't understand why the frequency of "hp", "cs" or "edu" would have a strong weight in determining if an email was spam or not.

In Figure 2 (page 4) it can be seen that the largest increase in accuracy is achieved at the beginning when the most heavily weighted features are used to train the SVM. The accuracy continues to increase as features with lower weights are used, but the accuracy doesn't increase nearly as fast as it did initially. This indicates that careful consideration of features can have a considerable effect on the results of the SVM: Finding the most relevant features will greatly increase accuracy, while non-relevant features may not increase the results and may only make the processing needlessly more complicated.

Experiment 3: Random Feature Selection

The final experiment is similar to experiment 2 in that the accuracy is measured after training the SVM on more and more features. However, in this experiment, the features are chosen at random. Figure 3 (page 5) shows the effect on accuracy from selecting random features to train the SVM. The accuracy seems to steadily increase towards the accuracy measured in experiment 1 with an occasional small jump. By contrasting this result with experiment 2, were we saw a very steep initial jump leveling off towards our final accuracy, we can see that adding a randomly selected feature may not add much to the SVMs ability to classify the samples.

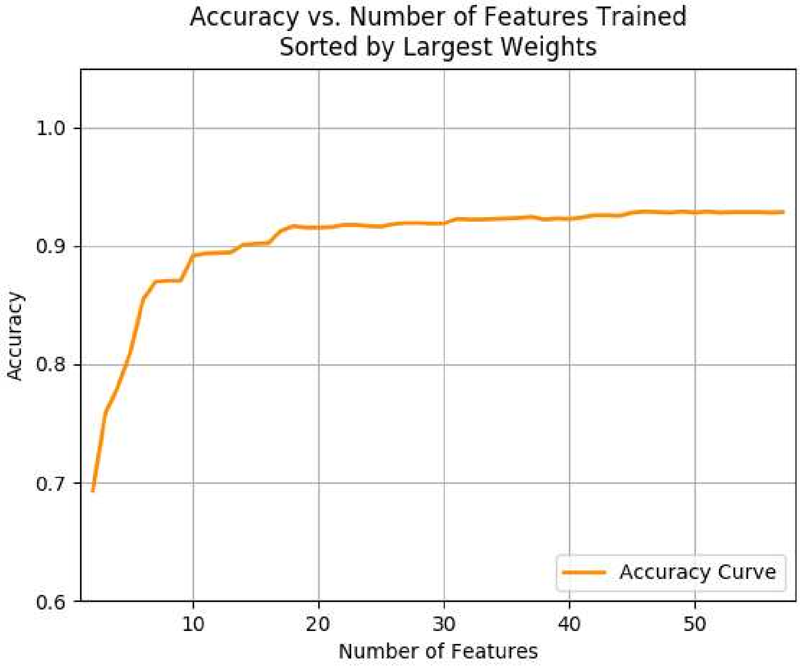

Fig. 2: Results of Experiment 2. The SVM was trained using an increasing number of features starting with the ones with the largest weights. It can be seen that the largest increase happens with the first ten features, indicating that these have the largest effect on classification.

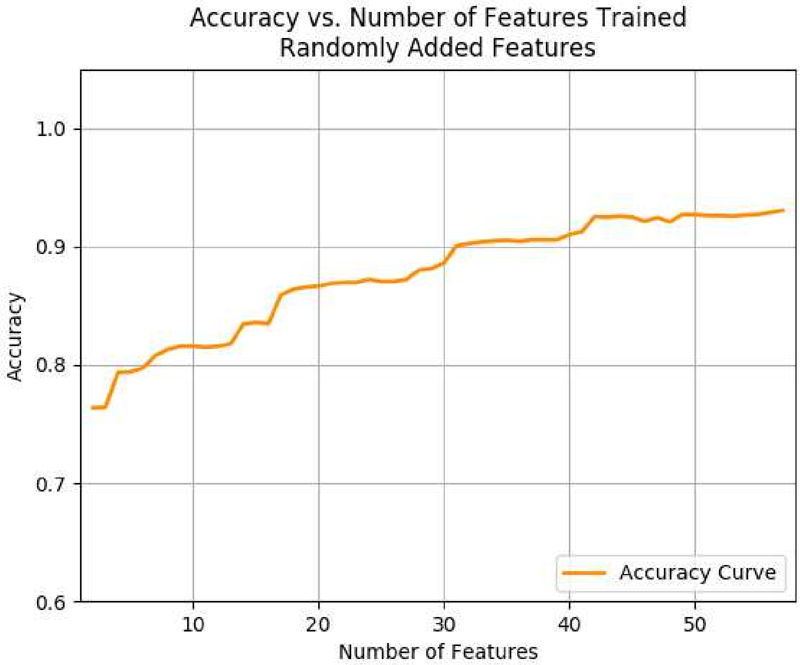

Fig. 3: Results of Experiment 3. The SVM was trained using an increasing number of features using randomly selected features. Since the feature selection is random, we don't see the steep jump that was seen in Experiment 2. Instead, the accuracy slowly increases towards the final accuracy we measured in Experiment 1.