Nathan McKenna

SVM Homework Analysis

I have used 1 of the 5 assignment extension days on this homework.

1)

To train the classifier to distinguish between 8â€™s and 3â€™s I used the sklearn SVC() model and the built in .fit method to train the model on the training set. I added addition command line arguments for different kernels and varying values of C.

```
parser.add_argument('--kernel', type=str, default =rbf')
parser.add_argument('--C', type=float, default=1)

model = SVC(kernel=args.kernel, C=args.C)
model.fit(data.x_train[:args.limit],data.y_train[:args.limit])
```

2)

For this problem I created a grid of the various parameters a linear, polynomial, and rbf kernel could take. I then used the sklearn GridSearchCV to test every permutation of those parameters through cross validation in order to determine which configuration had the highest training accuracy. The grid of parameters is below.

```
modelGrid = [
        {"kernel":["linear"], 'C':[.0001,.01,.1,.2,.3,.4,.5,.75,1,2,3,4,5,25]},
        {"kernel":["poly"], 'C':[.0001,.01,.1,.2,.3,.4,.5,.75,1,2,3,4,5,25],
         "degree":[1,2,3,4,5,6,7,20]},
        {"kernel":["rbf","sigmoid"],'C':[.0001,.01,.1,.2,.3,.4,.5,.75,1,2,3,4,5,25],
         "gamma":[.0001,.001,.01,.1,.5,"auto"]}]
```

Below are the lines that actually perform the grid search on the training data set. It is important to note that I used a training data size limit of 1250 because there are many permutations of the parameters that must be tested.

```
grid = GridSearchCV(SVC(), param_grid=modelGrid, cv=4, scoring='accuracy', verbose=10)
grid.fit(data.x_train[:args.limit],data.y_train[:args.limit])
```

The results of the grid search are below. An rbf kernel with C = 25 and gamma = 0.01 ended up being the best with an accuracy of 97.44%. The best polynomial kernel was quadratic and also had C = 25. Its accuracy was 96.56%. The best linear kernel had a C = 0.1 and and accuracy of 95.84%.

```
('The best linear permutation is: ', {'kernel': 'linear', 'C': 0.1}, 0.95840000000000003)

('The best polynomial permutation is: ', {'kernel': 'poly', 'C': 25, 'degree': 2}, 0.96560000000000001)

('The best rbf permutation is: ', {'kernel': 'rbf', 'C': 25, 'gamma': 0.01}, 0.97440000000000004)
```

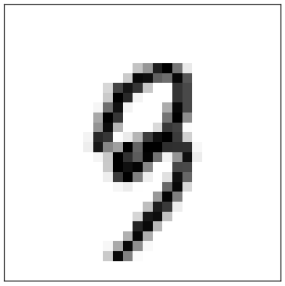
3)

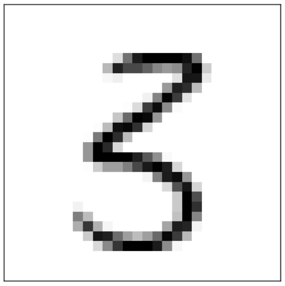
The first thing I did was to train the most accurate model determined above (rbf kernel) with the entire training data set.

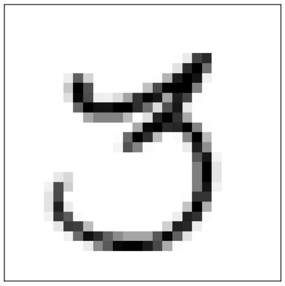
The support vectors will be the images that are classified correctly while also being closest to the decision boundary. What this mean is that the support vectors are the images classified with the least confidence. These will be the images that are the most vague,such as having extra or unclosed loops. These features make them difficult to classify with confidence.

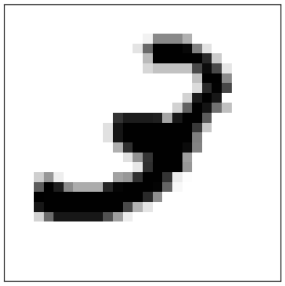
Below are a few images from the support vectors for 3 and 8. They were found by retrieving the indices of the pictures on the support vector determined by SVC().support_.

Support vector images for 3









Support vector images for 8