

For a detailed explanation of how the coding portion was accomplished, see the comments in svm.py.

Analysis 1

The scikit implementation of SVM was implemented using the SVC function (support vector classification). This function was used to distinguish 3's from 8's in the MNIST data. The main function from the knn homework was used to import the MNIST data and limit the number of training data points processed. The knn script was modified to build only datasets consisting of 3's and 8's that were then labeled appropriately.

The SVC function was passed a regularization value and kernel type as parameters. These parameters were varied and the resulting accuracy of the classifier on the test data set was recorded.

Analysis 2

A number of regularization values and kernels were tried. The results are recorded in the table bellow:

Run	Kernel	C Value	Train Size	Accuracy on
1	Linear	1	5000	0.948
2	Linear	0.8	5000	0.951
3	Linear	0.6	5000	0.953
4	Linear	0.4	5000	0.954
5	Linear	0.2	5000	0.955
6	Linear	0.2	20000	0.97
7	rbf	0.4	5000	0.951
8	poly	0.4	5000	0.505
9	poly	0.4	20000	0.505
10	poly	0.1	20000	0.505
11	poly	0.9	20000	0.555
12	poly	0.9999	20000	0.589
13	sigmoid	0.4	5000	0.505
14	sigmoid	0.9999	5000	0.505
15	rbf	0.01	5000	0.505
16	rbf	0.01	20000	0.589
17	rbf	0.99	20000	0.968

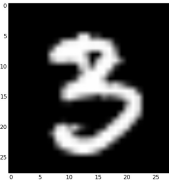
For a linear kernel, a smaller regularization term was found to improve accuracy for the same number of training points. More training points were found to improve accuracy for a the linear kernel using the same regularization value. The rbf kernel was found to perform better when the regularization term was closer to a value of 1.0. The sigmoid and polynomial kernels were found to perform poorly for a number of different regularization terms and training sizes. A linear kernel with $C = 0.2$ using 20k training points was found to have the best performance with a test data accuracy of 0.97.

Analysis 3

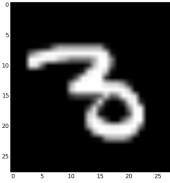
The index of various support vectors were found by calling SVC's support_ attribute. Support vectors were found for the SVC svm using $C = 0.4$, an rbf kernel, and 5000 training points. The resulting support vectors are those closest to the margin and therefore appeared to be those that were most difficult to discern:



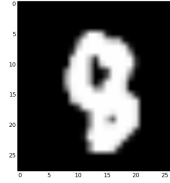
Support vector for 3, smudge visible



Support vector for 3, extra line



Support vector for 3, lower tail curves in



*Illustration 1:
Support vector for 8, lower lobe is collapsed*

In addition to these, other support vectors were found to be elongated or at an angle. All of them appeared to be atypical representations of 3's and 8's.