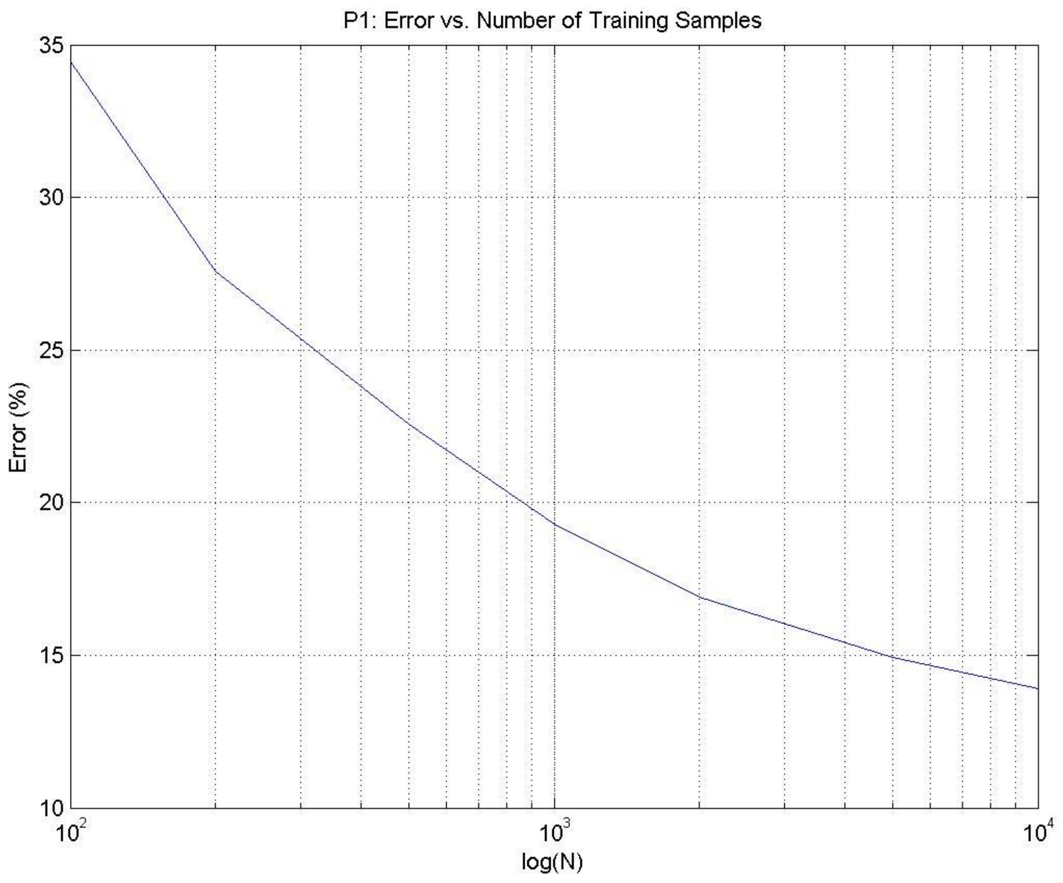
CS 289

JAvila

HW1

P1

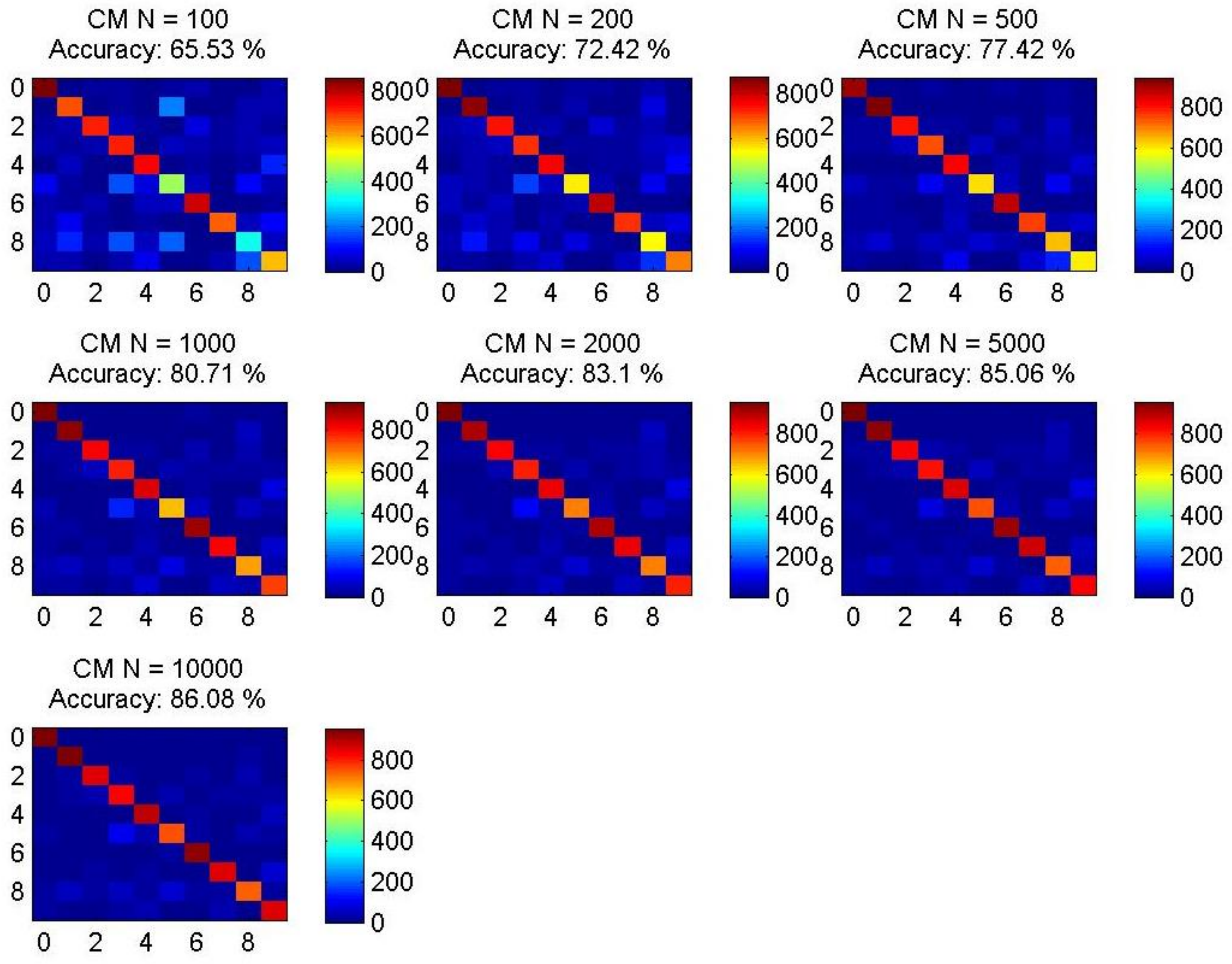
Here we built a simple support vector machine (SVM) using the default SVM of LIBLINEAR. From the digits data, I evenly sampled 10k images for validation. Training data were sampled from the remaining 50k images. Below is a plot of the error rate vs. the number of training samples.



HW1

P2

In this problem, we built confusion matrices for each of the experiments in P1. Below are the matrices. We can see that the biggest problems we have are misclassifying "5" for "2" and "8" for "9". Confusion matrices can give us great intuition about misclassification, and could help us tune our models.

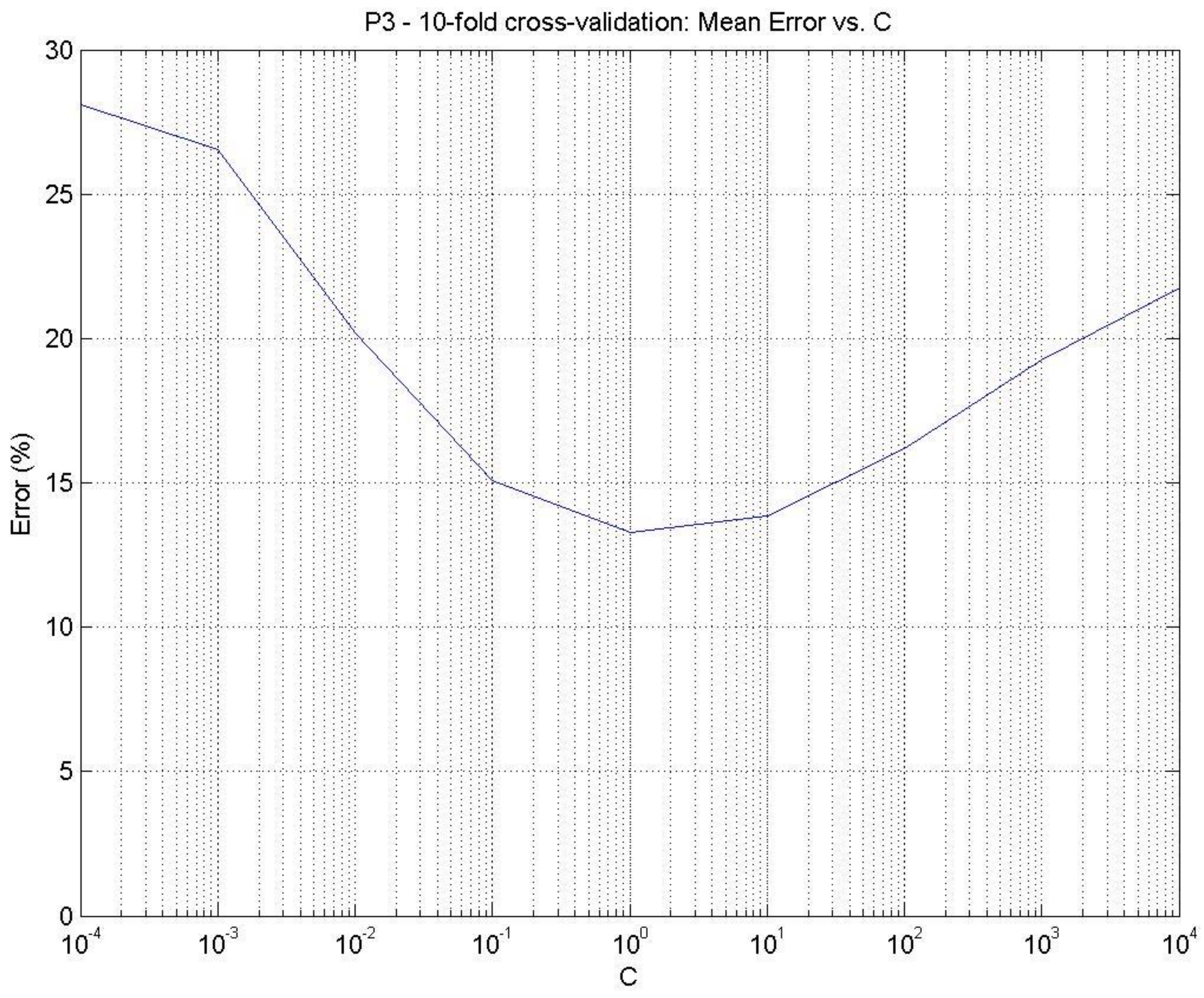


HW1

P3

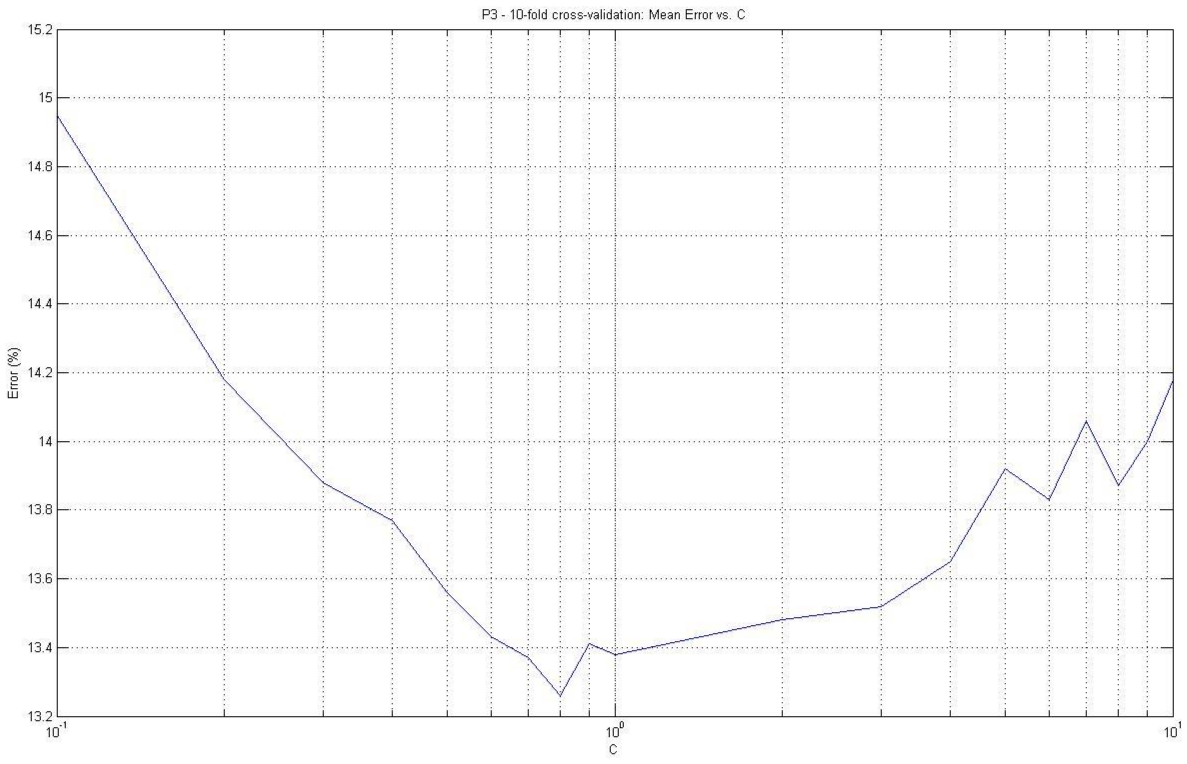
Here we were tasked to use 10-fold cross-validation to optimize the parameter C of our model. Cross-validation randomly samples from a training data and splits in into k parts for each C parameter. Randomly sampling ensures that we are not biasing our sample, and validating k-fold times and computing the mean error makes the current C parameter being tested robust against outliers. helps in two ways: optimize for model parameters and

Initial Run: C = [.0001 .001 .01 .1 1 10 100 1e+3 1e+4]. We can see that the optimal value seems to be between .1 and 10.



HW1

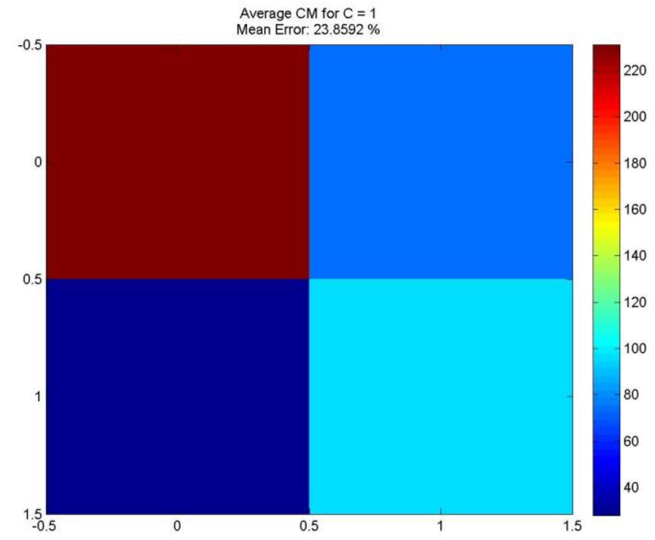
Second Run: C = [.1 .2 .3 .4 .5 .6 .7 .8 .9 1 2 3 4 5 6 7 8 9 10]. Here the minimum C occurs at C = 0.7 (mean_err ~ 13.2 %).



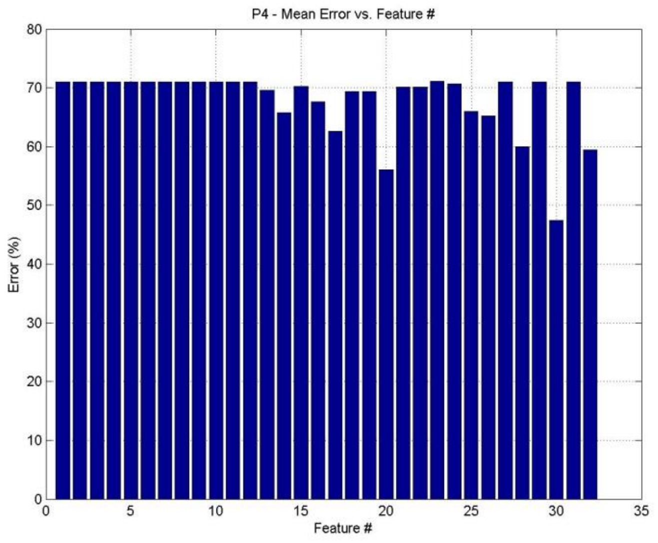
HW1

P4

I started with all features, and the average error and average confusion matrix for a 12-fold cross-validation and C = 1 are as follows:

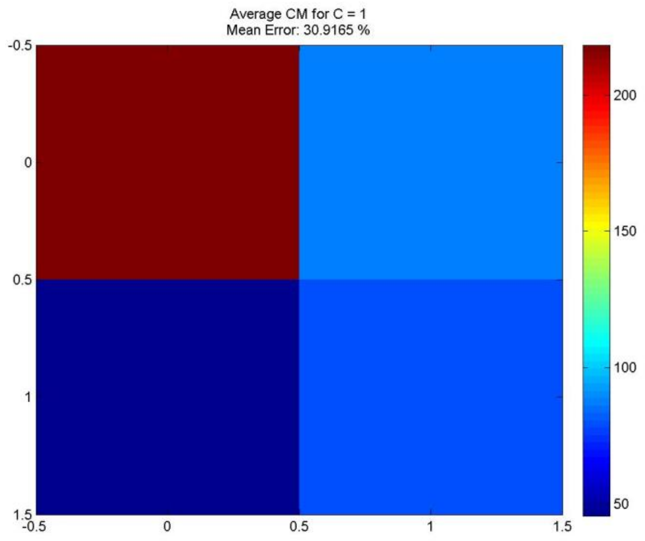


Next, I decided to see how each feature individually predicts. To make sure that I won't suffer from biasing, I sampled randomly but also evenly across each label. I ran the same code a few times and made sure that the plot stayed consistently the same. Below is the result from one of the runs.

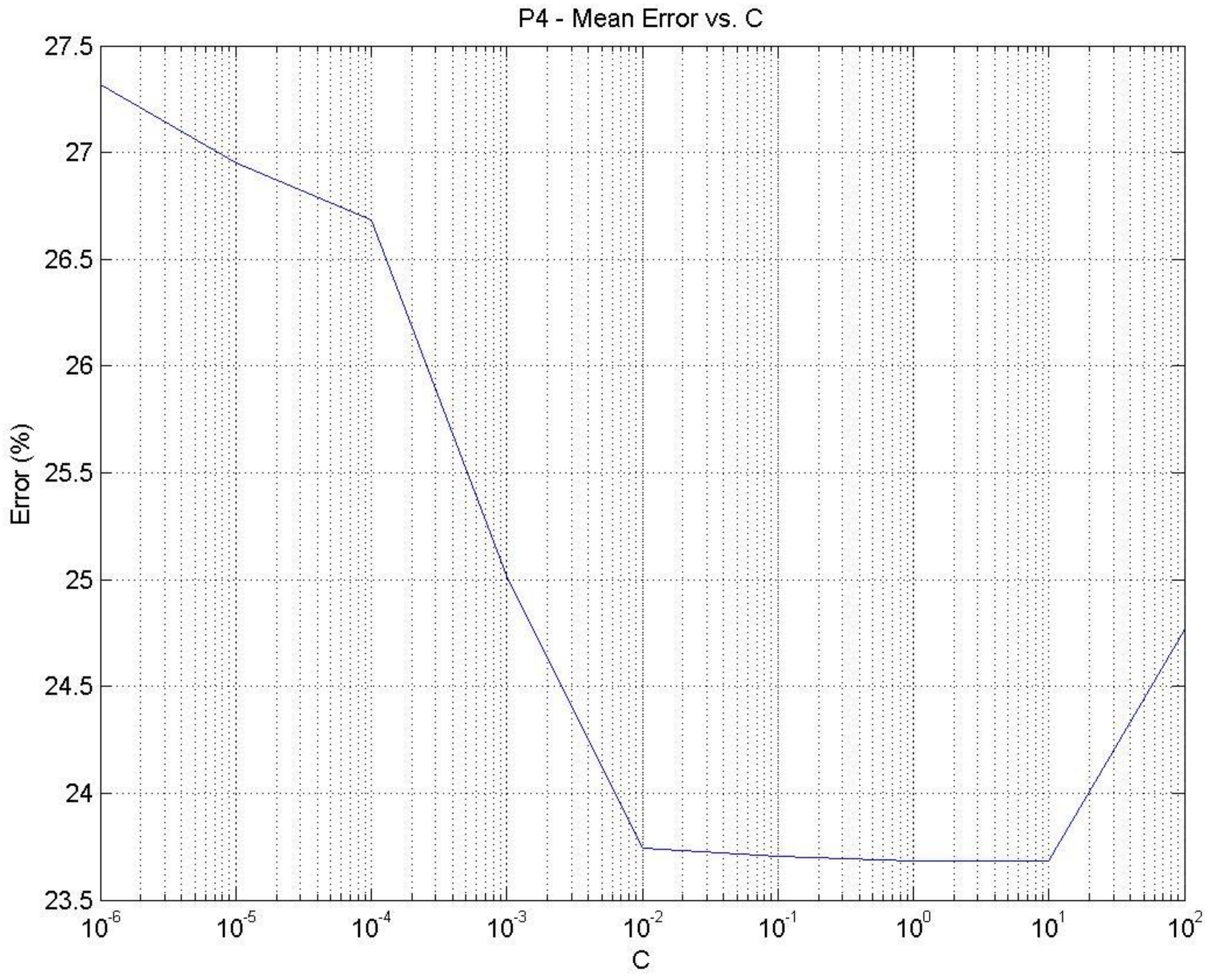


It would seem logical to remove the features that had the biggest error at classifying, so I set a threshold at 70% error and removed those features. Unfortunately, this performed worse. Below is the result.

HW1



I decided not to remove any features, and instead optimize the hyperparameter C. Below is the semilog plot.



CS 289

JAvila

HW1

My optimized code is for C = 1. The final model had no features modified.