Programming Assignment 1

CIS4930/5930

Due Wednesday Nov. 30, 1:25 pm EST

Instructions: All students must complete all problems. All implementations may be completed using the programming environment of your choice; the source code of all of your implementations must be submitted.

1. SVM

   (a) Implement the soft margin SVM classification algorithm. To solve the convex optimization problem use an external library, such as CVXPY (`https://www.cvxpy.org/install/index.html`).

   (b) Discuss the results of running your implementation on random data generated in the following way: for class A (corresponding to $+1$), generate 200 points where according to a two-dimensional gaussian with mean $(-1, -1)$ and covariance matrix

   $$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

   For class B, generate 200 points using the same distribution except change the mean to $(1, 1)$. Seed your random number generator with the last 5 digits of your Lib# (located on your FSU card).

   Plot the decision boundary of the classifier along with the training points. What are the support vectors? What is the margin? Compute the leave-one-out cross validation error. Vary the tradeoff parameter C between the misclassification error and the margin; plot the decision boundary for different values of C.

   (c) Download the MNIST dataset `http://yann.lecun.com/exdb/mnist/`. Use the training patterns to train your SVM implementation to distinguish between the classes corresponding to 0 and 1. What is the generalization error on the testing patterns?

2. Regression

   (a) Implement linear regression, given an input set of vectors in $\mathbb{R}^d$. Create a binary classifier by thresholding the output at 0. Discuss the results of running your implementation on the sample data generated for 1(b).

Plot the decision boundary of the classifier along with the training points. Compute the leave-one-out cross validation error.

(b) Implement logistic regression, given an input set of vectors in $\mathbb{R}^d$. Create a binary classifier by thresholding the output at 0. Repeat part (a) with this classifier. Add a regularization term that penalizes the ℓ_1-norm of the linear weight vector and repeat the experiment.

(c) Compare the generalization errors of distinguishing between class 0 and class 1 on the MNIST dataset of linear regression, logistic regression, Fisher's linear discriminant, and your SVM implementation.