Introduction to Pattern Recognition and Machine Learning - Spring 2016 - Project #1

Project Due: 10:50am Thursday, March 24, 2016

In project 1, you will implement a support vector machine in Python 3. *You are expected to implement the SVM yourselves (do not use code you find on the web or another source.* You can use a quadratic programming solver/package. *Be sure to get approval from me for any packages you use besides numpy, math, and matplotlib!*

Contest: There will be a contest! Your goal is to train your SVM to distinguish between hand-written characters. The teams with the best classfication accuracy will get extra credit.

Please have your code output a class label that matches the class value in the provided training data.

There will be test data points from classes that do not appear in the training data. So, you will want to come up with a way to identify when a test point class is "unknown" or what not in the training data. The label you should return for this case is -1.

Place code in your group GitHub Project1 Repository. Set it up here: `https://classroom.github.com/group-assignment-invitations/0319f63132ba93823adc930b9e86b47d`

Project Report: You should write a report that includes the sections listed below. Your report should follow the IEEE transactions format. Focus your report on your training strategies for the contest and any unique implementations. Templates for the IEEE transactions format can be found here:
http://www.ieee.org/publications_standards/publications/authors/author_templates.html

The maximum number of pages for the report is 3. If there are any pages beyond page 3, they will discarded and not read or graded. It should be written with correct English grammar and spelling. Be precise - use pseudo-code or equations to be precise.

Abstract A summary description of the contents of the report and your findings
Introduction Overview of your experiment and a literature review. For the literature review, include any references to any relevant papers for your experiment.
Implementation Describe and outline any specific implementation details for your project. A reader should be able to recreate your implementation and experiments from your project report. How will you use the SVM to do multi-class classification? How will you identify faces that were not in the training data?
Experiments Carefully describe your experiments with the training dataset and any other small toy data sets you constructed. Include a description of what the goal each individual experiment is and what your findings are.
Conclusions Describe any conclusions or things you learned from the project. Your conclusions must follow from what you did. Do not copy something out of a paper or say something that has no experimental support in the Experiments section.
References Listing of all references in IEEE bibliography format.

Submission Details: March 24 at 10:50am turn in your project and your code on GitHub. Be sure your code contains the following files: train.py (includes a *function* that will run your SVM training on an input data set X and desired output vector Y. Any parameter settings must be easy to find and modify.), test.py (includes a *function* that will run your testing code on an input data set X. Any parameter settings must be easy to find and modify. It should return a vector with the class label associated with each input data point X) and a concise README.txt file that clearly illustrates how to run your code. In class on March 24, we will hold a contest on test data that I will distribute. Your classification accuracy on this test data will factor into your project grade. The teams with the top 2 scores will get extra credit.

Grade Details: Your grade will be determined using the following breakdown: 50% implementation and classification accuracy on the data set and 50% project report.