MNIST Handwritten digit Classification

Use Support Vector Machines (SVMs) to build a handwritten digit classifier. Solve the SVM optimization problem using the Pegasos algorithm and also use a customized solver known as LIBSVM.

Each row in the (train/test) data file corresponds to an image of size 28x28, represented as a vector of grayscale pixel followed by the label associated with the image. Every column represents a feature where the feature value denotes the grayscale value (0-255) of the corresponding pixel in the image. There is a feature for every pixel in the image. Last column gives the corresponding label.

Given a training dataset D = $\{(x^{(i)},y^{(i)})\}^{m}_{i=1}$, the unconstrained SVM optimization problem can be written as:

$$\frac{1}{2}w^T w + C\sum_{i=m}^{m} \max\,(0,1 - t_i)$$

where, $t_i = y^{(i)}\left(w^T x^{(i)} + b\right)$

(a) Use mini-batch version of Pegasos algorithm to optimize above function and solve for w, b. Use a batch size of 100 in SGD implementation.

(b) Implement one-vs-one multi-class SVM

(c) Train a multi-class SVM using the LIBSVM library using a linear Kernel as well as a Gaussian kernel with γ = 0.05 (i.e. γ in K(x,z) = $\exp^{-\gamma * \|x-z\|^2}$). Use C = 1.0 in both cases.

(d) Perform 10 fold cross-validation to estimate the value of the C parameter for the Gaussian kernel case. Fix γ to be 0.05. Vary the value of C in the set $\{10^{-5}, 10^{-3}, 1, 5, 10\}$.

(e) Draw a confusion matrix for your best results in part (d).