COL 341: Assignment 4

Notes:

- You are advised to use vector operations (wherever possible) for best performance.
- Include a report of maximum 5 pages which should be a brief description explaining what you did. Include any observations and/or plots required by the question in the report.
- You should use Python for all your programming solutions.
- Your assignments will be auto-graded, make sure you test your programs before submitting. We will use your code to train the model on training data and predict on test set.
- Input/output format, submission format and other details are included. Your programs should be modular enough to accept specified parameters.
- You should submit work of your own. You should cite the source, if you choose to use any external resource. You will be awarded F grade or DISCO in case of plagiarism.
- You can use total of 7 buffer days across all assignments.
- Data is available at this link

1. **Support Vector Machine (50 points, Release date: Oct. 24, 2019, Due date: Nov. 04, 2019)**

   In this problem, we will use Support Vector Machines (SVMs) to build a handwritten digit classifier. We will be solving the SVM optimization problem using the Pegasos algorithm. You are provided with separate training and test example files. Each row in the (train/test) data file corresponds to an image of size 28x28, represented as a vector of grayscale pixel intensities 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. You are provided with a subset of the original MNIST dataset. We will work with this subset for the purpose of this assignment.

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

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

   where, $t_i = y^{(i)}(w^T x^{(i)} + b)$. Other symbols are as described in the class. Use mini-batch version of Pegasos algorithm described in "Pegasos: Primal Estimated sub-GrAdient SOlver for SVM" to optimize above function and solve for w, b. You should start with Algorithm 1 given in the paper. Further, Algorithm 1 ignores the intercept term b. To incorporate b, use the description provided in Section 6 of the paper (use Equation 23). Your final implementation should compute the values of both w and b. In order to extend this to the multi-class setting, we train a model on each pair classes to get $\binom{k}{2}$ classifiers k being the number of classes. During prediction time, we output the class which has the maximum number of wins among all the $\binom{k}{2}$ classifiers.

Evaluation:

- For part-a you can get 0 (error), half (code runs fine but predictions are incorrect within some predefined threshold) and full (works as expected).

Submission Instructions:

Support Vector Machine

Submit your code in 1 executable python file called svm.py

> *python svm.py trainfile.csv testfile.csv testpred.txt*
> Here you have to write the predictions (1 per line) and create 1 line aligned file testpred.txt for test file.

Coding Guidelines:

- Grading will be based on test set accuracy.
- Unfair means in this assignment include use of external libraries like sklearn or use of any other technique other than pegasos algorithm. Be assured your submissions will be carefully checked and any such attempt will be **severely punished** with appropriate disciplinary measures.