CSE/ECE 343/543: Machine Learning
Assignment-3 SVM and MLP

Max Marks: 40 **Due Date**: 09/11/2022, 11:59PM

Instructions

- Keep collaborations at high level discussions. Copying/Plagiarism will be dealt with strictly.
- Late submission penalty: As per course policy.
- Your submission should be a single zip file **2020xxx_HW3.zip** (Where *2020xxx* is your roll number).
- Include only the **relevant files** arranged with proper names. A **.pdf report** explaining your codes with relevant graphs and visualization and theory questions.
- Do **NOT** include data files in your submission. It makes your files unnecessarily big while downloading.
- Ensure that everything required for a particular question is present in their respective files in terms of functions (not comments). Failure to do so would result in a penalty. Follow the following file structure for submission:

  2020xxx_HW3
  |– Q1.py
  |– Q2.py
  |– Q3.py
  |– Report.pdf
  |– Weights (folder)
  |– Plots (folder)

- Remember to **turn in** after uploading on google classroom.
- Resolve all your doubts from TA's in their office hours **two days before the deadline.**
- **Document** your code. Lack of comments and documentation or improper file names would result in loss of 20% of the *obtained* score.
- **Section A** is mandatory. You need to do one of **Section B** and **Secton C**.

Section A [Theoretical] (10 points)

1. (10 points) John has the following dataset:

Class	x1	x2
Ä	0	0
A	1	0
A	0	1
B	1	1
B	2	2
B	2	0

He is studying machine learning and encountered some questions which he is unable to solve. Help him solve the following questions.

1. (2 points) Are the points linearly separable? Support your answer by plotting the points.
2. (3 points) Find out the weight vector corresponding to the maximum margin hyperplane. Also find the support vectors present.
3. (2 points) What is the effect on the optimal margin if we remove any one of the support vectors in this question?
4. (3 points) In general, for any dataset, what can we say about the effect on optimal margin, if we remove any of the support vectors?

OR

2. (10 points) Consider the dataset:

Class	x
¨+	0
-	-1
-	+1

1. (4 points) Is the data linearly separable? If not, try to map every data point in three dimensional space using the feature vector $\phi(x) = [1, (\sqrt{2}\ )\text{x}, \text{x}^2]^T$ and find whether the points are linearly separable. If no, justify your answer and if yes, support your answer by finding a separating hyperplane.
2. (3 points) Consider a class variable $y_i \in \{-1, 1\}$ denoting the class of variables and assume $w = (w_1, w_2, w_3)^T$. The max margin classifier solves

$$\min_{w,b} \frac{||w||_2^2}{2}$$

s.t. $y_i(w^T\phi(xi) + b) \geq 1$ and $i = 1, 2, 3$ Show that the solution is $\hat{w} = (0, 0, -2)^T$, $b = 1$ and the margin is $\frac{1}{||\hat{w}||^2}$. [Hint: use Lagrange multipliers].

3. (3 points) Does the solution change if the constraints are changed to $y_i(w^T \phi x_i + b) \geq \rho$ for $i = 1, 2, 3$ and $\rho \geq 1$?

Section B [Scratch Implementation] (15 points)

1. (15 points) You must implement a general algorithm for Neural Networks. You can only use the NumPy library and the Pandas library for reading the dataset.

    1. (7.5 points) Implement a class named **NeuralNetwork** which takes as input the following parameters:

        - N: Number of layers in the network
        - A list of size N specifying the number of neurons in each layer
        - lr: Learning rate
        - Activation function (same activation function is used in all layers of the network except the last layer)
        - Weight initialization function
        - Number of epochs
        - Batch size

        The **NeuralNetwork** class should also implement the following functions:

        - fit(X, Y): trains a model on input data X and labels Y
        - predict(X): gives the prediction for input X
        - predict_proba(X): gives the class wise probability for input X
        - score(X, Y): gives the accuracy of the trained model in input X and labels Y

        Use of helper functions like forward, backward, reset_gradients is recommended to simplify implementation. You can also create helper classes.

    2. (2.5 points) You need to implement the following activation functions (along with their gradient functions): sigmoid, tanh, ReLU, Leaky ReLU, linear and softmax (only used in last layer).

    3. (2.5 points) You need to implement the following weight initialization functions: zero init, random init and normal init (Normal(0, 1)). Choose appropriate scaling factors.

    4. (2.5 points) Train the above implemented network on MNIST dataset. Perform appropriate preprocessing. Use the following configurations for training the network:

        - Number of hidden layers = 4
        - Layer sizes = [256,128,64,32]
        - Number of epochs = 100 (can be less if computation is taking too long)
        - Batch size = 128 (or any other appropriate batch size)

        Choose the remaining parameters appropriately. Plot training loss v/s epochs and validation loss v/s epochs for each activation function. Also, save all the trained

models as you might be asked to run them during the demo (TAs won't wait for the model to train). Note that it is possible that your training is stuck in a local minimum. Try fiddling around with the batch size.

Section C [Inbuilt Algorithms] (15 points)

1. (15 points) Using the sklearn implementation of MLP, train an MLP classifier on Fashion-MNIST dataset (divide training set into 85:15 train to validation set). The hidden layers should be of sizes (256, 32). Choose appropriate number of epochs and batch size.

    1. (3.5 points) Plot training loss v/s epochs and validation loss v/s epochs for activations simgoid, ReLU, tanh and linear (default learning rate). Which is the best activation function? Give analysis and comparison for each.
    2. (3.5 points) Using the best activation function obtained above, train models using learning rates $[0.1, 0.01, 0.001]$. Plot training loss v/s epochs and validation loss v/s epochs. Which is the best learning rate? Give explanations of the results obtained for each learning rate.
    3. (3 points) Decrease the number of neurons in each layer to various values. What do you observe? Plot training loss v/s epochs. Justify your answer.
    4. (5 points) Perform grid search on appropriate parameters of MLPClassifier. Choose the best parameters. Give an analysis of why you might have got those parameters.