Indian Institute of Technology Hyderabad
Deep Learning (AI2100/AI5100): Assignment-1

Assigned on: 20^{th} **January, 2023**
Deadline: 30^{th} **January, 2023**
Maximum Marks: 40

1 Instructions

- Answer all questions. We encourage best coding practices by not penalizing (i.e. you may not get full marks if you make it difficult for us to understand. Hence, use intuitive names for the variables, and comment your code liberally. You may use the text cells in the notebook for briefly explaining the objective of a code cell.)
- It is **expected** that you work on these problems individually. If you have any doubts please contact the TA or the instructor no later than 2 days prior to the deadline.
- You may use built-in implementations only for the basic functions such as `sqrt, log`, etc. from libraries such as `numpy` or `PyTorch`. Other high-level functionalities are expected to be implemented by the students. (Individual problem statements will make this clear.)
- For plots, you may use `matplotlib` and generate clear plots that are complete and easy to understand.
- You are expected to submit the Python Notebooks saved as <your-roll-number>.ipynb
- If you are asked to report your observations, use the mark down text cells in the notebook.

2 Problems

1. **Perceptron learning algorithm**
   [10 = dataset creation 2 + perceptron learning algorithm 5 + analysis 3]
   Implement perceptron learning algorithm for classifying a linearly separable dataset in 2D. Note that you have to create such a dataset with at least 1000 data points.
   Discuss your observations with respect to number of iterations required for perfect classification (k) by varying the level of separability (γ from the class discussions) in the dataset. (Hint: compute the average value of k for each level of γ, and do this for about 5 values of γ. Observe if you can relate to the result discussed in class)
2. **Gradient descent for training a linear classifier**
   [10 = loss formulation 4 + gradient computation 3 + update equation 3]
   Consider solving the above problem (training a line for classifying a linearly separable 2D dataset) using Gradient Descent algorithm. Think of a loss function based on our classroom discussion. You may implement the analytical way of finding gradient for it. You may implement the basic version of gradient descent update equation.
3. **MLP with a single hidden layer**
   [20 = dataset creation 3 + MLP definition 3 + backprop 10 + elegance/recursion 4]
   Consider a binary classification dataset that is not linearly separable in 2D (e.g. data lying on the circumference two concentric circles). Train a Multi layer perceptron (MLP) with a single hidden layer for classifying the same. You may use the loss function used in problem 2. You have to implement the gradient descent (i.e. backpropagation) algorithm yourself.