# Assignment

Anand Mishra
Machine Learning - 1
AY 2020, Sem-2
IIT Jodhpur

May 11, 2021

Note:

1. **Due date:** May 30: 11:55 PM (early bird deadline), June 8: 11:55 PM (20% penalty)
2. This is a mix of programming and conceptual assignment. There are a total of three problems.
3. One of the objectives of this assignment to provide you hands-on experience with neural networks.
4. TAs will communicate regarding submission instructions soon.

## Problem 1: Perceptron [40 points]

Following training samples are given:

| $x_1$ | $x_2$ | Class |
|---|---|---|
| 1 | 1 | +1 |
| -1 | -1 | -1 |
| 0 | 0.5 | -1 |
| 0.1 | 0.5 | -1 |
| 0.2 | 0.2 | +1 |
| 0.9 | 0.5 | +1 |

Table 1: Sample data

Assuming weight vector of initial decision boundary $\mathbf{w}^T\mathbf{x} = 0$ as w=[1, 1], solve the following:

1. In how many steps perception learning algorithm will converge. (15 points)
2. What will be the final decision boundary? Show step-wise-step update of weight vector using computation as well as hand-drawn plot. (15 points)
3. Prove that Perceptron Learning Algorithm converges in a finite number of steps. (10 points)

Resources:
1. Refer to Fractal-3, Lecture-2.
2. `http://page.mi.fu-berlin.de/rojas/neural/neuron.pdf`

# Problem 2: Learning to implement Neural Network [40 points]

1. We have seen in class how backpropagation is used to compute gradients of cost functions with respect to weights. In class example we took Mean Square Error as cost function. Replace this cost function with cross-entropy loss and demonstrate backpropoagation. Submit both handwritten explanation as well as modified code. Code link (10 points)

   Resources: `http://neuralnetworksanddeeplearning.com/chap2.html`

2. Please go through the following blog to learn how to recognize handwritten digits using Neural Network. Here Neural Network is coded using PyTorch Library in Python.
   `https://towardsdatascience.com/handwritten-digit-mnist-pytorch-977b5338e627`
   Use above code and report your observation based on the following:(15 points)
   (i) Change loss function,
   (ii) Change in learning rate, and
   (iii) Change in Number of hidden layers

3. **Gurmukhi Handwritten Digit Classification:** Gurmukhi is one of the popular Indian scripts widely used in Indian state of Punjab. In this part of the assignment, our goal is to develop a neural network solution for classifying Gurmukhi digits. We provide you Handwritten Gurmukhi digit dataset here:

   Dataset link

   Modifying the code provided in 2, and develop a robust neural network to classify the Gurmukhi digits. Higher performance on test set will have bonus point. Briefly write your observation and submit your code so that we can evaluate your implementation at our end. (15 points)

# Problem 3: MISC [20 point]

1. Compute derivative of following activation functions: (3 points)
   (i) Sigmoid
   (ii) tanh
   (iii) ReLU

2. What are the strategies you will follow to avoid over-fitting in a neural network. (2 point)

3. Let x = $[1,1]^T$ , y = $[1,1]^T \in \mathbb{R}^2$ and let $f: \mathbb{R}^2 \implies \mathbb{R}^2$ with *f(z)* $= z_1.x + z_2.y$ for any z = $[z_1, z_2]^T \in \mathbb{R}^2$. Further, $z = g(r) = [r^2, r^3]$ where $r \in \mathbb{R}$. Show how chain rule is applied here giving major steps of the calculation, write down the expression for $\frac{\partial f}{\partial r}$, and also evaluate $\frac{\partial f}{\partial r}$ at r=2. (2 points)

4. Define following loss functions in brief. (3 points)
   (i) Mean Squared Error (MSE)
   (ii) Binary Cross-entropy (BCE)
   (iii) Categorical Cross-entropy (CC)

5. Explain the following variants of Gradient Descent in brief: (10 points)
   (i) Batch Gradient Descent
   (ii) Stochastic Gradient Descent
   (iii) Mini-batch Gradient Descent
   (iv) Momentum based Gradient Descent
   (v) Adam solver

   Resources:
   1. Variants of GD
   2. Loss functions