

## Assignment 3

This assignment intends to test your understanding of Multilayer Perceptron concepts and how to apply it using Tensorflow gradient tape.

### Dataset generation

Use following code to generate the training and the testing dataset. This dataset is similar to XOR but have 4 classes. X\_train and y\_train are features and labels respectively for the training dataset. Similarly, X\_test and y\_test are features and labels for the testing dataset.

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split

num_observations = 10000

x1 = np.random.multivariate_normal([0,0], [[0.1, 0.2],[0.2, 0.1]], num_observations)
x2 = np.random.multivariate_normal([3,3], [[0.1, 0.2],[0.2, 0.1]], num_observations)
x3 = np.random.multivariate_normal([3,0], [[0.1, 0.2],[0.2, 0.1]], num_observations)
x4 = np.random.multivariate_normal([0,3], [[0.1, 0.2],[0.2, 0.1]], num_observations)

features = np.vstack((x1,x2,x3,x4)).astype(np.float32)
labels = np.hstack((np.full(num_observations,0),
                    np.full(num_observations,1),
                    np.full(num_observations,2),
                    np.full(num_observations,3)))
X_train, X_test, y_train, y_test = train_test_split(features,labels, test_size=0.33,random_state=42)
```

Task 1 – Visualize the complete dataset where samples from one class have same color. Show the legend and axes name as well.

Task 2 – Train a neural network with more than 1 hidden layer. Keep all the hidden layers as dense (fully connected). Use any non-linear activation functions (ReLU, Sigmoid, etc.) of your choice to find non-linear decision boundary. Make sure your model is neither overfitted nor underfitted. Use regularization to overcome overfitting.

Task 3 – Plot the final decision boundary for each class.

Task 4 – Show the training and testing results for final model and also analyze the effect of changing the hidden layers and regularization.

You can use TensorFlow library for the implementation