Project 1

Following is a binary classification task. Consider we have 4 dimensional gaussian samples, one is $\frac{1}{\sqrt{2\pi\sigma_1^2}}e^{-\frac{(\vec{x}-\vec{\mu}_1)^2}{2\sigma_1^2}}$, another is $\frac{1}{\sqrt{2\pi\sigma_2^2}}e^{-\frac{(\vec{x}-\vec{\mu}_2)^2}{2\sigma_2^2}}$, where $\vec{\mu_1} = (1, 1, 1, 1)$, $\sigma_1 = 0.5$, $\vec{\mu_2} = (0, 0, 0, 0)$ and $\sigma_2 = 2$, each one has 10000 points for training. For simplicity, you can label the first type of distribution $\hat{y} = 1$ and another is $\hat{y} = 0$. Denote $\vec{x}$ as the input of the network, $\hat{y}$ is the true label and y is the output of network. To complete the task, please:

- Write a neural network in PyTorch to do the forward propogation.
- Write a custom loss function $L = \frac{1}{m}\sum_{batch}(-\hat{y}\log y - (1-\hat{y})\log(1-y) + |\vec{x} - \vec{\mu_1}|)$, where m is the batch size and $|\vec{x} - \vec{\mu}_1|$ is the Euclidean distance $\vec{x}$ and $\vec{\mu}_1$.
- Implement a train function and set your favorite parameters to train the network.
- Generate test samples, 2000 for each distribution and evaluate your network to display the distribution of the output y for each type of test samples.
- Implement a function to calculate separation, which is $\frac{1}{2}\sum_{i}^{bins}\frac{(n_i^a - n_i^b)^2}{n_i^a + n_i^b}$, where n_i^a and n_i^b is the frequency of y in the i-th bin for each two type of test samples(you can choose the same number of bins as in the distribution of y you displayed.). Evaluate the separation of test samples.