



Middle East Technical University
Electrical-Electronics Engineering Department



EE 583 Pattern Recognition Homework 6

Due Date: 03.01.2021, 23:55 via odtuclass.metu.edu.tr

This homework is an online application of PyTorch library on Google Colaboratory, by the help of provided codes written in Python language. You can find more information about the syntax and functions at <https://pytorch.org/docs/stable>. Although not required, if you are interested, you can also check the tutorials at <https://pytorch.org/tutorials>.

In order to work in Google Colaboratory, you should use your Google Account. If you do not have a Google Account, you can create an account from [here](#).

Click the following Google Colaboratory project [link](#) (prepared by Assist. Aybora Köksal aybora@metu.edu.tr). Click to Copy to Drive in order to save file on your own local/personal drive, to make permanent changes. You can copy the main project more than once, since in each part you will make changes to original structure.

Before start, check your code is running on GPU. Therefore, follow the options Runtime -> Check runtime type -> Hardware accelerator -> Select GPU -> Save. Also note that at every question, you should start running the code from beginning (initial random network) to make a better comparison.

- 1) Click Runtime -> Run all. Alternatively, you can run cell by cell with clicking Play buttons at the left of each cell row. Observe the dataset by checking the plot on xy-axis. Examine the *network structure, learning rate, optimizer, loss function*. Plot the learning curve for *training data* and find *validation* accuracy and loss values after the last epoch.
- 2) Increase the number of epochs to 500. Plot the learning curve for training data and find validation accuracy and loss values for the last epoch. What is the effect of increasing the number of epochs?
- 3) Add two additional hidden fully connected layers into the current model. Specify the input sizes and indicate your modified `FullyConnected` class. Plot the learning curve for training data and find validation accuracy and loss values for the last 500th epoch. What is the effect of this larger network?
- 4) Increase the learning rate to 0.01. Plot the learning curve for training data and find validation accuracy and loss values for the last 500th epoch. Compare the results two networks of the previous questions.
- 5) Now, return to the initial network in Q1. Add one *dropout* layer (with probability of 0.5) to the output of ReLU activation function. Indicate your modified `FullyConnected` class. Plot the learning curve for training data and find validation accuracy and loss values for the last epoch. Analyze the effects of dropout layer.
- 6) Verify that optimizer of the model is Stochastic Gradient Descent (SGD). Introduce *momentum* to the optimizer. Plot the learning curve for training data and find validation accuracy and loss values for the last epoch for the network in Q1 with momentum (Hint: Default value for momentum in SGD is 0.9. You are encouraged to try other values to get higher validation accuracy)