**ENGR421 Introduction to Machine Learning Assignment #3 Report**

Orçun Özdemir

54020

I have started my project by importing fundamental machine learning libraries such as numpy, pandas and matplotlib since I have coded my homework with Python. Then, I have imported all given csv files with pandas and converted them into dataframes. Labels of clothes are started from 1 to 5 which can cause issue for python since indexing starts from 0, therefore, to overcome this issue I have encoded the dataframe that contains labels with label encoder which provided by sklearn library. Then, I have split my data into train and test parts, to do this I have taken first 500 data points and set them to train for both images and labels, and last 500 data points and set them to test for both images and labels as well. Then, I have implemented sigmoid and safelog functions as in the lab session and implemented the softmax function which takes Z and v, dot products it and divides into row sum of dot products of them which will be our activation function just like sigmoid. Then, I have implemented a function which takes two input a dataframe and a numpy array respectively. The function concatenates them into one dataframe and transform into an array. This function was useful since during our activation functions and calculating delta\_v we need this columnwise binded arrays to match our given matrices (W and v). To handle the matrix multiplication operations, I had to one hot encode the y\_train by using one hot encoder from sklearn. Then, I indicated the hyperparameters and started to iteration. Just before the iteration, I have created an objective\_values list that appends loss function throughout the iteration. Then, I activated hidden layers with sigmoid function by using colum binded X\_train and W, then I activated output layer by using softmax function with the output of first activation function and v. Then, as we were described in the book part 11.7.3, I have summed the encoded y\_train times safelogged y\_pred, output of softmax function, and multiplied with -1. Then, I have appended the result to objective values. Again, I have calculated delta\_v and delta\_W in the scope of 11.7.3 and printed the iteration to observe whether the iteration was working smooth and sound. After the iteration, I have plotted the objective values, and observed the decrease of error throughout the iteration. Then, I have taken the argmax of the y\_pred and appended into a different list to get the classification outputs. Then, I have one hot encoded the y\_test and activated X\_test to find y\_pred\_test and did the same operation to get the classified output. Then I have put these lists to two different confusion matrix with related test and train data (encoded\_y\_train and encoded\_y\_test) with the help of sklearn library and observed great results.