Letter Recognition and Data Preperation

Abstract:
In this report the method of preprocessing data and its impact on accuracy is discussed. The data set chosen is that of 'Letter Recognition' and the neural network is set up using PyTorch. The study was conducted by running the neural network against both prepared and unprepared data with different epochs and the results are compared.

The results obtained were far worse than that of the paper "Letter Recognition Using Holland-style Adaptive Classifiers". An accuracy of 80% was achieved using the same data, but the best accuracy achieved by this study was 35%.

Introduction:
The problem chosen was a classification problem, to identify the letter based on the black and white rectangular pixel displays. This problem is an interesting one as many Optical Character Recognition libraries are built on this and this study is an attempt to achieve the best results of classification based on the method suggested in the paper "
Classifying dry Sclerophyll Forest from augmented satellite data: Comparing Neural Network, Decision Tree & Maximum likelihood".

Data Set:
The data set of 'Letter Recognition' was obtained from the archives of UCI Machine Learning Repository (ref). The data is that of character images, black and white rectangular pixel displays of 26 capital letters, which were based on twenty different fonts. Each letter within these 20 fonts was randomly distorted to produce a file of 20,000 unique stimuli. Each stimulus was converted into 16 primitive numerical attributes which were then scaled to fit a range of integers from 0 through 15.

This data was used in the paper "Letter Recognition Using Holland-style Adaptive Classifiers", in which an accuracy of 80% was achieved using the same data.

In this study the data was slightly manipulated for the sake of code reusability and performance improvement. The following modifications are done to the data. In the original data the first column is the English letter and the following columns its corresponding attribute values, to make use of the code from the course lab sessions, this column is made the last column. Another modification is the converting the 26 different classes from strings to numbers to better support the neural network. The final modification is to split the data into two, one with 16,000 rows to train the network and the other with 4000 rows to test the network. This is to normalize both the data separately instead normalizing the entire data and then splitting it as the first 16,000 rows were meant for training and the remaining 4,000 for testing.

Model Design:
The neural network design is adopted from one of the lab sessions, as it was used with a similar classification problem of identifying the type of glass from its chemical composition.

A normalization function was used to prepare the data. The network was ran with different number of epochs from 15 - 35. Since the instance of data are too large, increasing the number of epochs resulted in consumption of more time for program execution.

Results:
Without normalizing the data the accuracy obtained for 35 epochs was 19.02%. After normalizing both training and testing data the accuracy obtained for 35 epochs was 34.55%, which is a significant rise but not near perfect.

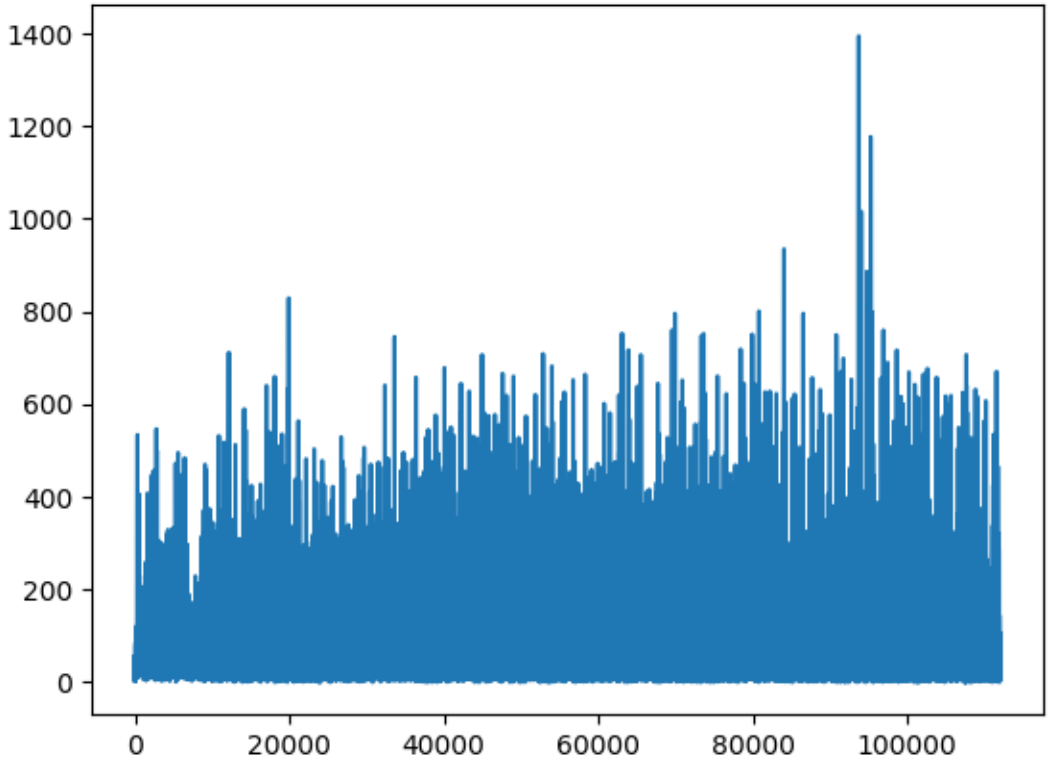

Fig 1 – Loss at 35 epochs