Yingzhi Hao 12/9/2019

ECE-469

The data that I use is the “Wine Quality Dataset” from <https://archive.ics.uci.edu/ml/datasets/Wine+Quality>. Specifically, I use the “winequality-white.csv” dataset. Each row has 12 values. The first 11 values are attributes of the wine sample and the 12th value is the quality score result. The original dataset contains 4898 data sets. I used 3500 sets as training sets and the rest 1398 sets as testing sets. The goal is to train a network so that it can predict the quality score of a wine according to its 11 attributes.

I modified the dataset before using it in my neural-network program. Each column of attributes was normalized by dividing with the maximum number of that column and then being truncate to three decimal points. The quality score of each set was converted from decimal number (0-11) to binary number sequence under inversion of one-hot-encoding (decimal 1 in this case will be encoded to 1011111111111).

I used 100 epochs, learning rate 0.1 and 10 hidden nodes. The result shows an F1 score of 95% accuracy.

I wrote a script, “dataset\_processer.py”, to automatically modify the original dataset from “winequality-white.csv” and generate training-set file “wine\_quality.train” and testing-set file “wine\_quality.test”. The script also generates initial weights for my neural network and save them in the file “wine\_quality\_weight.init”. The weights are generated with uniform random distribution, which is one of the common ways to generate initial weights in deep-learning field.

The network was trained with 100 epochs and 0.1 learning rate. The trained weights were saved in “wine\_quality\_weight.trained”. The trained network was tested with the testing set, and the results were saved in “wine\_quality.result”.