Assignment 2 , 159.740, 2020 S2

Family Name: JIANG , Given Name: TAO ,

Student ID:20004769

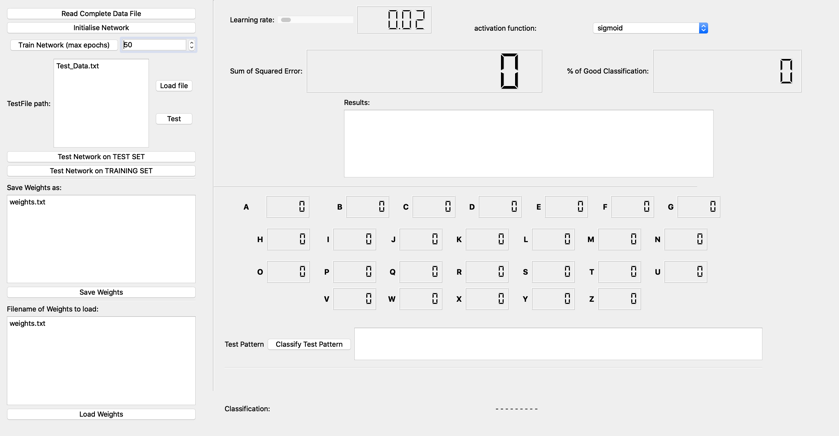
**Ass2** Letter Recognition using Deep Neural Nets with Softmax Units

The experiment uses a deep neural network to classifier the dataset of Letter+Recognition in the UCI.

* **User’s Guide**

**Environment**

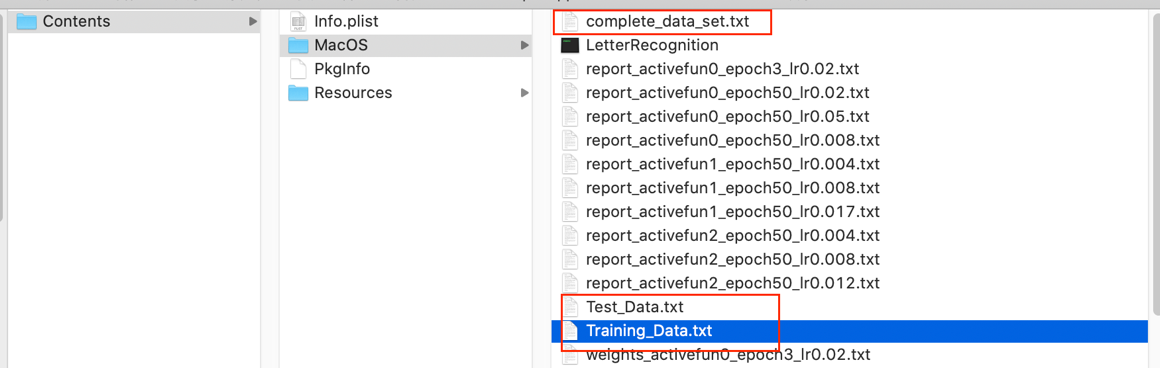
I use macOS as a development platform. And use *qmake* to compile the project in the Qt5.1 integrated environment.



**Preparation**

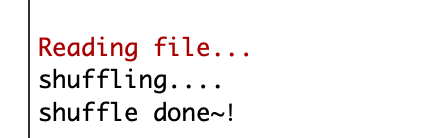
Compile the project first and copy the dataset files to :

*./build-LetterRecognition-Desktop\_Qt\_5\_15\_1\_clang\_64bit-Debug/LetterRecognition.app/Contents/MacOS/*

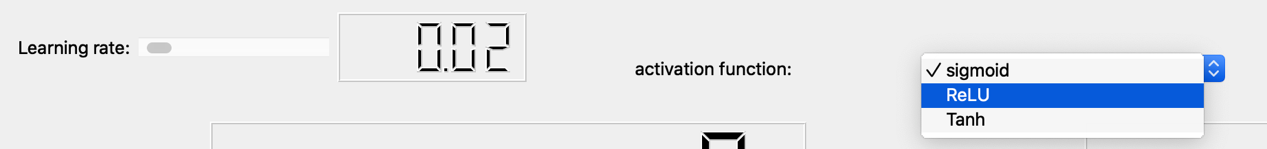


**Training**

Execute the program and click the button of **Read Complete Data File** to read the file of *complete\_data\_set.txt.* The program will automatically perform random shuffle operations for the data set.

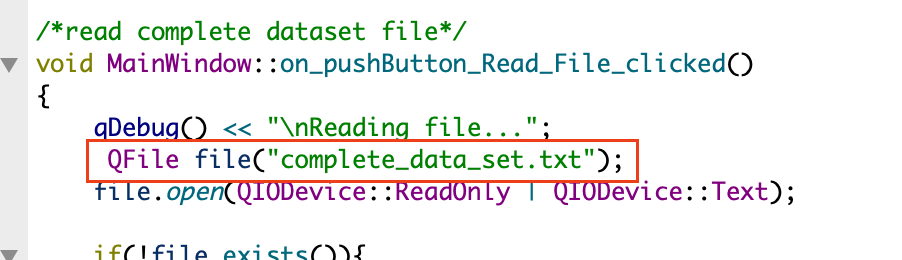


Next, click the button of **Initialize Network** to initialize the network and weight. After that, adjust the required parameter **Learning rate** and the **Activation function** of the model.



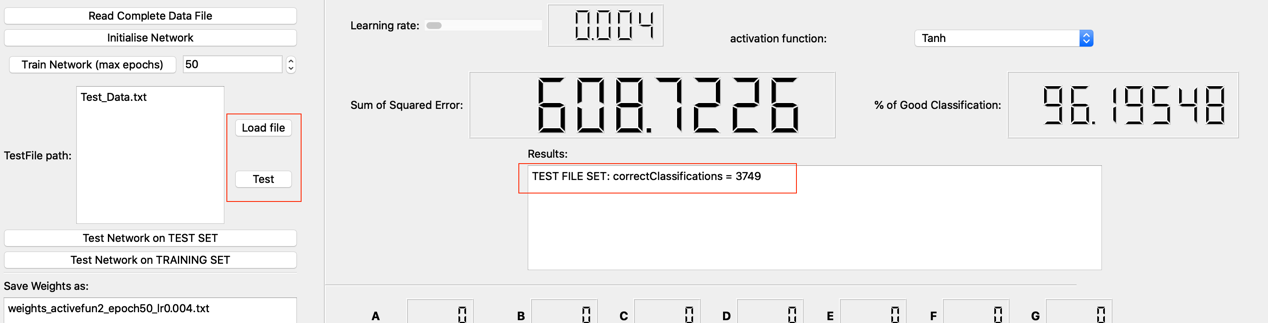
The last selected the size of **Max epochs** and click the button of **Train Network**. And then the Network will training.

Changing the training data set requires changing the path of the data set.



**Testing**

The program implements the function of verifying a separate test dataset. Enter test dataset file name and click the button of **Load file**. Click the **Test** button will verify the model by test dataset as below.



**Save and load weight**

Save and load weight functions implement properly as well. The files will save at *./build-LetterRecognition-Desktop\_Qt\_5\_15\_1\_clang\_64bit-Debug/LetterRecognition.app/Contents/MacOS/* . At the same time, Each the model of epoch SSE and MSE will record.

* **Experiment results**

The experiment uses different activation functions and learning rates to train and verify the model. The input size of the model is a vector of 16, and the output layer is 26 neurons and passes through the SoftMax classifier. Among them, the model uses 2 hidden layers, each with 128 and 256 neurons connected. I trained the models by using different learning rates and activation functions. Get the three best performing models. The results are as follows:

