ASE 5016 PROJECT WORK

Hıdır Yüzügüzel, 244904

December 12, 2015

In this project work, I used red wine dataset. It contains $N = 1599$ samples, $D = 11$ feature dimensions and 11 different categories (quality score from 0 to 10). First of all, I plot the histogram of the dataset in order to see which classes contain how many samples. As can be seen from Figure 1, the dataset is unevenly distributed. We do not have any samples from quality scores of 0, 1, 2, 9 and 10. Further, we have very few samples from quality score class of 3, 4 and 8. The unevenly distributed dataset leads to difficulties in training.

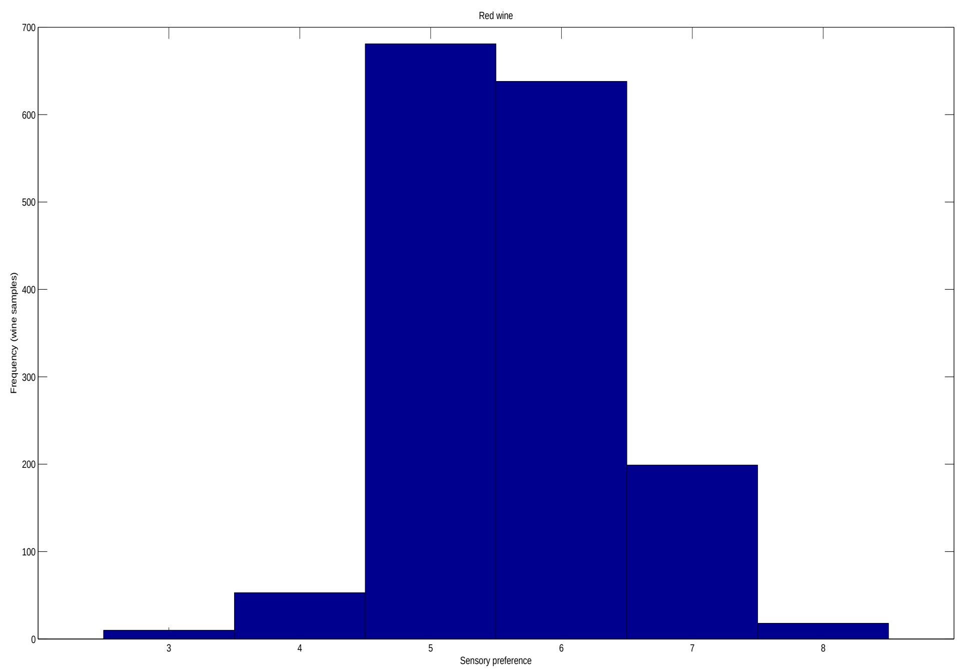

Figure 1: Red wine dataset histogram

I splitted the dataset randomly into three different subsets of training (70%), validation (15%) and test (15%) sets at the beginning. As nonlinear model, I chose multilayer perceptron (MLP) due to its great modelling capability (MLP is universal function approximator). I used all the input features. I used MATLAB Neural Network Toolbox *patternet* implementation for this task. Since Matlab implementation already performs a zero mean and unit variance feature standardization, I did not have to perform feature normalization beforehand. I used a single hidden layer in the MLP since it is not a problem to approximate any function even with a single hidden layer if it contains enough neurons. The number of hidden neurons is the hyperparameter of the model. I choose this hyperparameter based on k-fold cross validation. I search the hidden neuron number in a list of $(5, 6, \ldots, 25)$ neurons. I splitted the training set (70%) into $k = 5$ different folds. At each time, I trained the MLP with 4 folds and tested with remaining 1 fold. I repeated this procedure 5 times. At the end, for each parameter candidate, I computed the average correct classification rate over the folds. After that, I choose the hyperparameter which gives the maximum of these average classification rates. Here, note that I did not use test set which I created at the beginning. Lastly, I trained a new neural network, with best hyperparameter, using all training set (70%) and tested the trained network on unseen test data set (15%). The validation set is used for Neural Network early stopping regularization. The same validation set is used both in model selection and final training. Note that, we are interested in the performance of unseen test set. I performed two different experiments for two different hidden layer activation function, namely logistic sigmoid and tangent hyperbolic. In the output layer, softmax activation function is used.

In the experiment with logistic sigmoid activation, parameter search yielded 13 neurons as the best parameter. The network structure is shown in Figure 2. The confusion matrices for the dataset are presented in Figure 3. The classification accuracy is 63.3% on the test set. The receiver operating curve (ROC) is also presented in Figure 4.

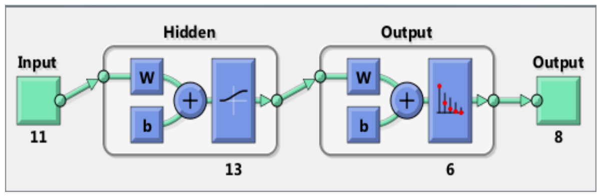

Figure 2: The Neural Network structure with sigmoid activation function

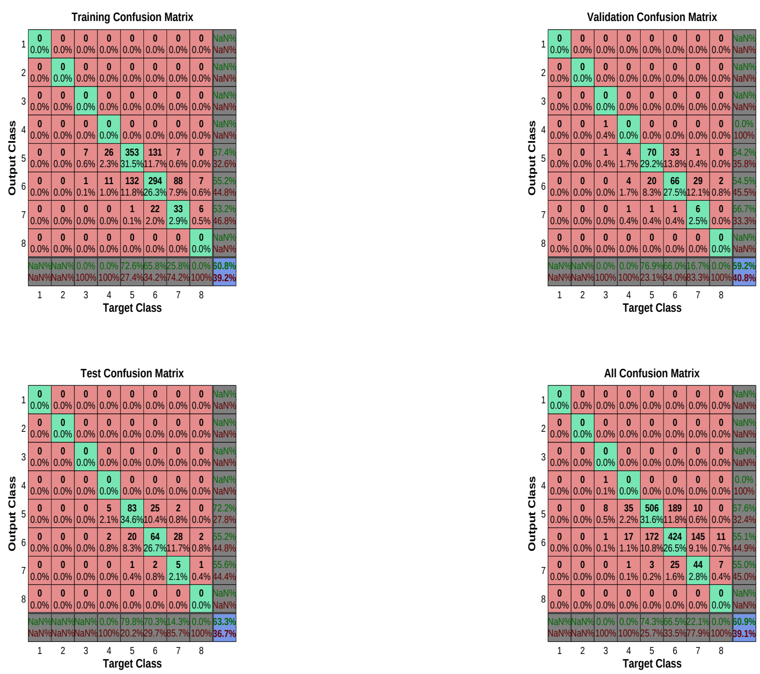

Figure 3: Confusion matrices for logistic sigmoid activation function

In the experiment with tangent hyperbolic activation, parameter search yielded 5 neurons as the best parameter. The network structure is shown in Figure 5. Compared to logistic sigmoid activation function, this is a much more simpler model. The confusion matrices for the dataset are presented in Figure 6. The classification accuracy is 64.6% on the test set. The receiver operating curve (ROC) is also presented in Figure 7.

As a comparison, I applied linear regression modelling to the data. I rounded the result of linear regression to convert the real valued output into categorical variables. The confusion matrix for the test set is presented in Figure 8. The classification accuracy is 60.4% on the test set. The nonlinear model of neural network outperforms the linear model. We can claim that the dataset is nonlinear and thus nonlinear model outperforms the linear model. However, the linear model is not very bad.

In summary, the neural network classification accuracies were lower than what I expected. The most probable reason for this could be that the dataset is not evenly distributed. However, as can be

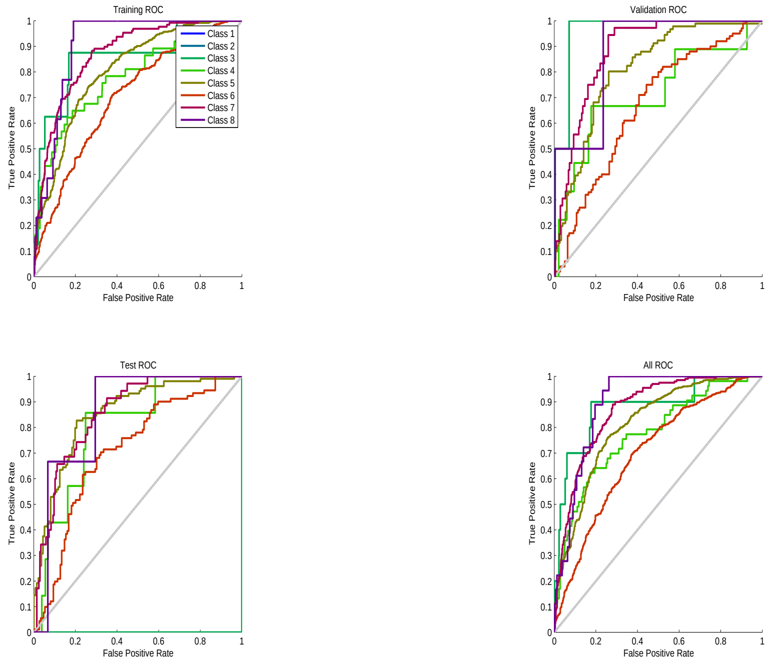

Figure 4: ROC for logistic sigmoid activation function

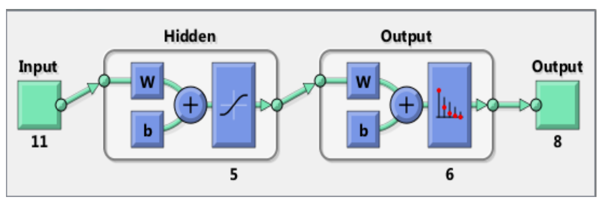

Figure 5: The Neural Network structure with tangent hyperbolic activation function

seen from the confusion matrices, the errors are mostly between the consecutive classes. For example, if a sample from class quality score 5 was misclassified, it is classified either as 4 or 6. If we consider top 3 classification accuracies, the results would be very high. In my opinion, a Deep Neural Network (DNN) with very few hidden layers can outperform the traditional MLPs. The dataset size is small for but data augmentation is possible. Furthermore, one can pretrain the DNN with white wine dataset.

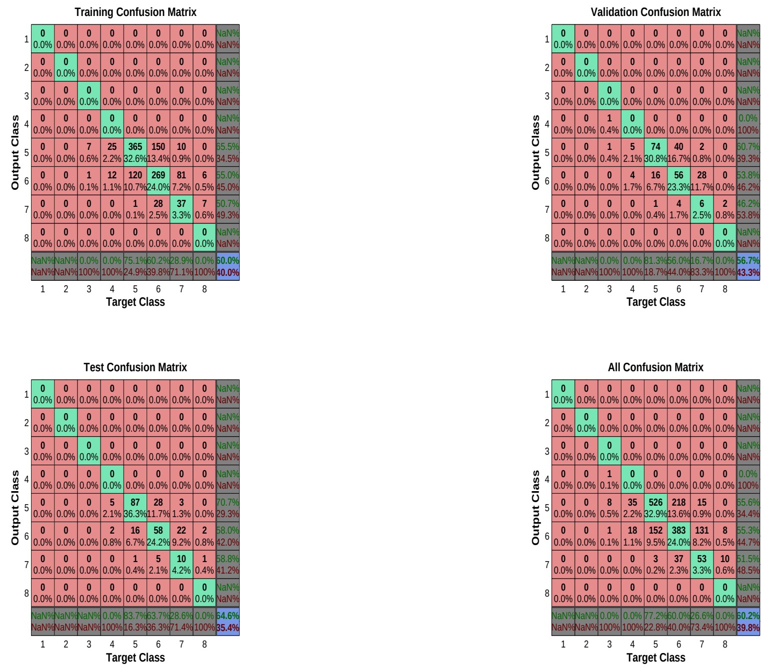

Figure 6: Confusion matrices for tangent hyperbolic activation function

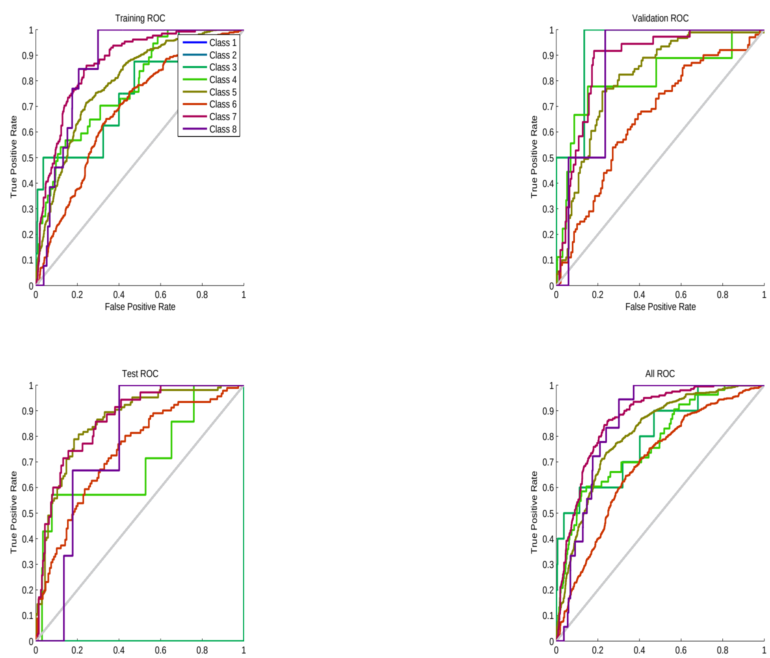

Figure 7: ROC for tangent hyperbolic activation function

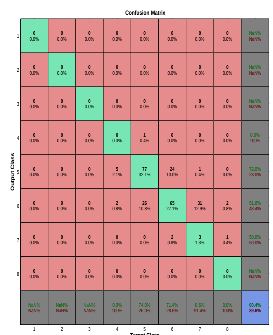

Figure 8: Confusion matrix of test set with linear regression