Neural Network

Implementation of neural network analysed in this section is found under the *neural_network* directory in the NeuralNetwork.m file and runnable from NeuralNetworkRun.m file at the same directory.

Git repository: https://gitlab.cs.ttu.ee/Allar.Viinamae/iti8565-hw3.

Properties and hyperparameters used:

1. Neural network architecture consist of 2 hidden layers and an output layer each with 2 neurons. 2 neurons at output layer represent each class value. Hot encoding technique is used by transforming the output layer to binary vectors [0, 1] and [1, 0], so one column per class.
2. Logistic function is used as the transfer function for each neuron in the network after the neuron is activated. Sigmoid (logistic) transfer function derivative is calculated when back propagating errors through the network.
3. Within each epoch each row in the training dataset is used to update the network (stochastic gradient descent).
4. Banknote validation is performed the dataset from:

   https://archive.ics.uci.edu/ml/datasets/banknote+authentication
5. For evaluation KFolds cross-validation technique was used. Each fold was evaluated by calculating the accuracy metric.

Improvement options include better initial weights initialization, batch gradient descent (update weights only after each epoch) and different architecture.

AdaBoost

Implementation of AdaBoost algorithm analysed in this section is found under the *adaboost* directory in the AdaBoost.m file and runnable from AdaBoostRun.m file at the same repository. Archieved accuracies 84.111% and 82.799% with the first two features from the same banknote dataset using Decision Tree Stump as the weak classifiers with M=5 and learning rate=0.01.

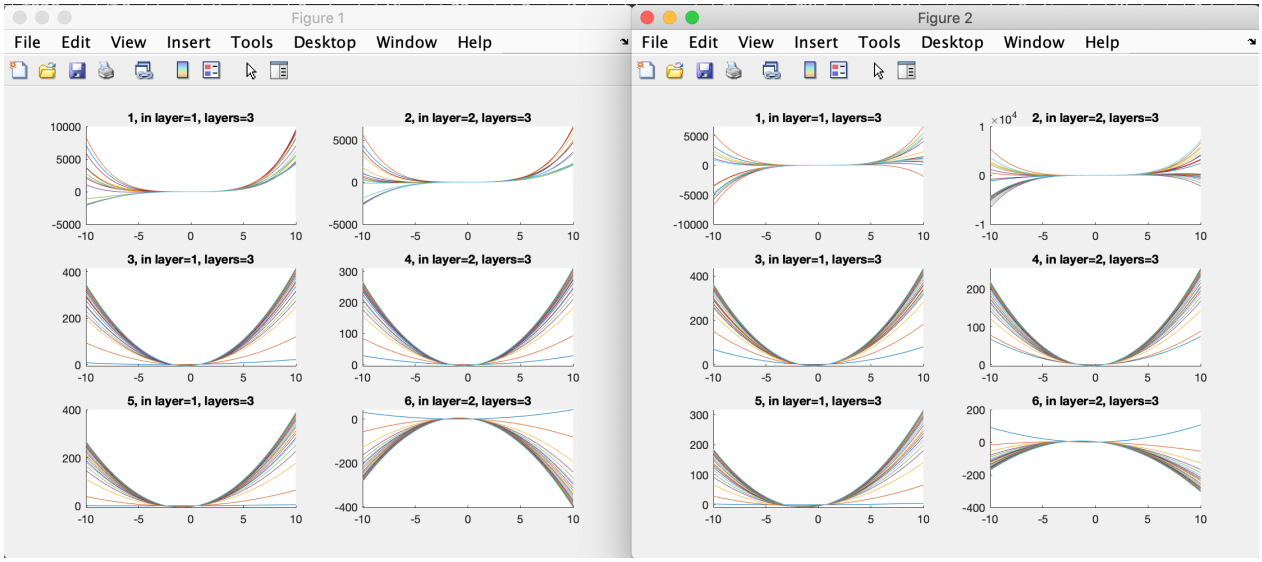

Figure 1: Graph lines density represent the direction the algorithm is converging. Achieved accuracies 98.667% and 100.000% with RSE 1.317 and 1.244 with 20 iterations and learning rate 0.3. Number of features = 4. Probably overfitting.

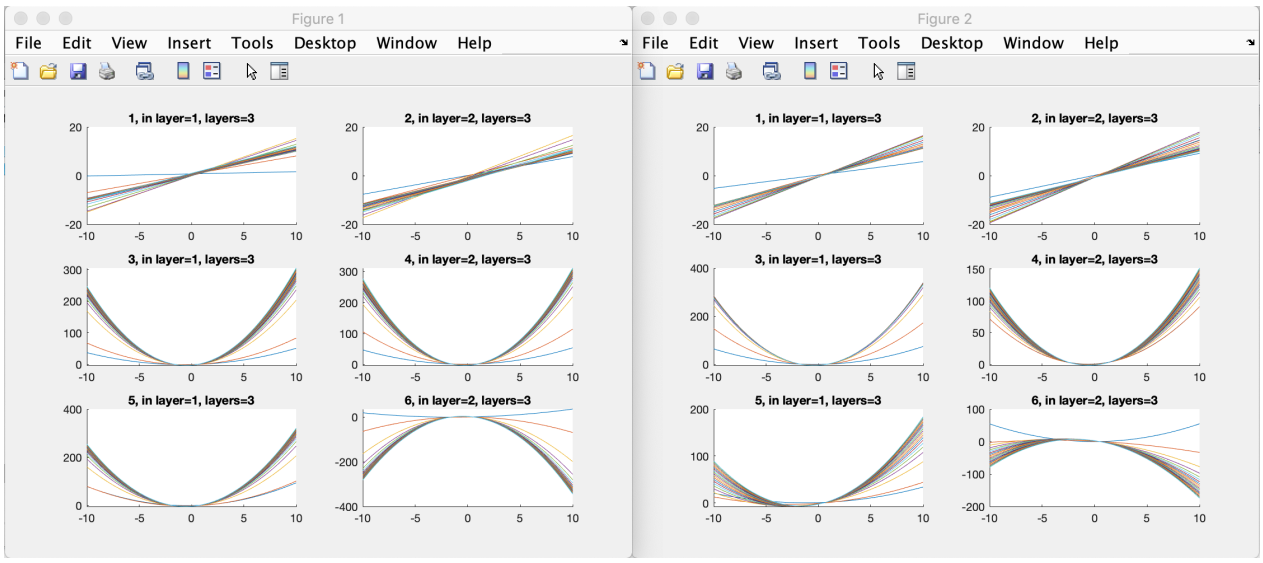

Figure 2: Achieved accuracies 86.297% and 82.945% with RSE 166.668 and 145.463 with 20 iterations and learning rate 0.3. Number of features = 1.