

Neural Network for Handwritten Digits Recognition

Li-Yi Lin (llin34, llin34@jhu.edu), Chenyang Su (csu8, csu8@jhu.edu)

1 Abstract

Neural network has been successfully applied to many machine learning topics such as speech and object recognitions, and machine translation in recent years. Neural network can learn new features automatically instead of doing feature engineering by hand, making it very powerful of dealing with complex problems.

In this project, we are going to adopt neural network techniques to do handwritten digits recognition. The shapes of handwritten digits varies from people to people, thus are difficult for simple machine learning algorithms to classify them. We will investigate how to apply neural network to this recognition problem and get hands-on experience by implementing the algorithm and tuning parameters. Finally we hope our result can be comparable to the result from applying external libraries which implement other machine learning algorithms suitable to this problem.

2 Methods

Neural network produces hidden layers by passing the linear combination of original features to a nonlinear function (e.g. logistic function). Then by passing the linear combination of hidden nodes to a nonlinear function, we produce outputs. The hidden layers can be thought as new feature representations, which is helpful for nonlinear classification. Classifying handwritten digits is a nonlinear problem, so neural network is a suitable algorithm to apply.

To implement the neural network, in the training process we will apply the backpropagation to learn the parameters and in the prediction we will apply feedforward propagation.

3 Resources

We plan to use the "Modified National Institute of Standards and Technology (MNIST)" dataset. The training and testing data files will be downloaded from Kaggle: <https://www.kaggle.com/c/digit-recognizer/data>. The data has already been preprocessed into csv formats so we can use them directly. Every image in the MNIST dataset is 28 pixels in width and 28 pixels in height. So every instance in the training and testing data files has 784 features corresponding to the 784 pixels. The value of the feature is 0-255 to indicate the darkness of the pixel. The training data file also contains a label column indicating the digits 0-9.

For comparison, we plan to use libraries from Python scikit-learn to perform the same recognition task.

4 Milestones

4.1 Must achieve

We will implement a multi-classes classifier built on neural network on our own. It can fully perform the training and predict task on our training data and testing data. We will also get familiar with some external library implementing some other machine learning algorithm (e.g. SVMs, KNN) and apply it to our dataset to get good results.

4.2 Expected to achieve

We will tune the parameters to make the performance of our classifier as good as other off-the-shelf machine learning libraries.

4.3 Would like to achieve

We will try to do some rigorous analysis on the two results got from our implemented neural network and the external library. Suppose there is some big difference in the performance, we will try to find out the reason.

5 Final Writeup

In the final writeup, we will first include the detailed mathematical representation of this neural network for solving this problem. The training and testing processes will be described in the final report as well. For the performance, we will include the training and testing results and the comparison of different machine learning algorithms. Furthermore, we will analyze the result (If we finished the "Would like to achieve" item) to find out possible reasons for one algorithm to outperform other algorithms.

6 Bibliography

1. Stefan Knerr, LCon Personnaz, and GCrard Dreyfus, "Handwritten Digit Recognition by Neural Networks with Single-Layer Training," IEEE.
2. Chris Bishop, "Pattern Recognition and Machine Learning," 2006, Chapter 5.