

# Machine Learning Contest 2016 Solution Document

# Abstract

This paper presents a method to recognize the handwritten digits, which utilize the deep neural network. Recognizing handwritten digits isn’t easy, many machine learning methods have been developed, like neural network, SVM. As this paper is not academic, we simply choose the most common and efficient algorithm for handwritten digit, which is the deep learning. Also, no proof will be given.

The data is divided into two parts, 9000 images and 1000 images. The 9000 data is expanded by affine and elastic distortion.

The deep neural network is constructed with two convolutional layers, two fully connected layers and one soft max layers for output. One deep neural network can obtain the accuracy about 99.3%. With five or more different networks vote for the final result, it can improve to about 99.4%.

# Platform

The platform is local.

Lenovo Y400N laptop 2.6GHz

8G RAM

Nvidia GT750M

OS: Windows

Laguage: Python

Packages: Theano, numpy, cv2

# Features

1. Take use of the GPU to accelerate training process, it is simply by importing the Theano library, and enable GPU function. It can save about 80% training time.
2. As the training data given is not enough, it is necessary to expand the data. The simplest method is shifting, each image is shifted with 1 pixel in 4 directions. Then each image is rotated by +/-12 degree. The last and most important expanding method is elastic distortion. By changing the kernel, alpha and intensity parameters of elastic distortion, we can generate different dataset for each training.
3. Use five different neural network to vote for the final result, as each neural network is trained with different expanded dataset (use the above method), the parameters should be different. And the submitted result proved what we have assumed.

# Models

The algorithm we utilized is deep learning.

The deep neural network is constructed with two convolutional layers, two fully connected layers and one soft max layer for output.

The first convolutional layer has 20 feature maps, each associated to 5x5 local receptive field. The second convolutional layer has 40 feature maps, each associated to 5x5 local receptive field. In addition to the convolutional layers just described, each layer also contains pooling layer immediately after them. The pooling layer take the root of the sum of the squares of the activations in the 2x2 region, which is used to simplify outputs.

The next two layers are fully connected layers, the first one has 640 input neurons, 1000 output layers. The second one has 1000 input neurons, and 1000 output neurons.

we use rectified linear units as activation function, that is , also the l2 regularization method is used, the . The regularized cost function is

Where is the original cost.

Besides the above regularization methods, the dropout technique is also used to reduce overfitting, which removes individual activations at random while training the network.

For soft max layer, the activation of the th output neuron is

The learning rate for the whole model is 0.03.

# Training process

1. GPU acceleration is quite important, it saves us a lot of time. With GPU, we can train 4-5 deep networks in one day, where each network can have an accuracy rate > 99.2%.
2. Expanding the training data is also important, we have tried different ways to expand, among them, the elastic distortion is the most efficient one. We also tried to use tangent vectors with precomputed deformation fields, which is used by the infiMNIST project, but failed to implement. The expanding data improves about 0.2%-0.3%.
3. The number of training data is 9000\*5\*3\*2, the number of validation data is 1000. To save time, we use a mini batch size = 10. During training, for each 3000 training data, the program checks the accuracy on validation data, once it is larger than 99.2%, the program stops and starts a new training process. This because we don’t know the highest accuracy of current model, but to reach an accuracy 99.3% or higher, it will take a quite long time (> 12 hours), the time spent on accuracy 99.2% is acceptable. Finally, we use 5 different networks to vote for the result, which can achieve 99.35%-99.4% accuracy.
4. The accuracy validation data,

# Conclusion

1. Convolutional layers greatly reduces the number of parameters of the system
2. Regularization techniques are necessary to reduce overfitting
3. GPU saves a lot of time
4. Algorithmically expanding the data can make use of sufficiently original data
5. Use right cost function to avoid learning slowly

Actually the algorithm for recognizing handwritten digit now is mature, most researchers use deep neural network to train, and have a similar structure like convolutional layers. The highest record official we know is 99.7%, but this record is achieved with much more training data. Our team thinks that the most difficult problem of this competition is lack of training data.

# Reference

[1] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” Proc. IEEE, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[2] D. Ciresan, U. Meier, L. M. Gambardella, and J. Schmidhuber, “Convolutional neural network committees for handwritten character classification,” in Proc. ICDAR, 2011.

[3] D. Ciresan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber, “Flexible, high performance convolutional neural networks for image classification,” in Proc. IJCAI, 2011.