COMP 551 Kaggle Competition: Classification of Modified MNIST*

Yingnan Zhao[1], Vincent d'Orsonnens[2] and Hamed Layeghi[3]

***Abstract*— This paper provides the report for the Kaggle Competition (assignment 4) of COMP 551 using the provided Modified MNIST dataset. The dataset includes a set of 8-bit grayscale images that include 2 or 3 digits of different sizes that are rotated and scaled from the classic MNIST dataset. The goal is to design Machine Learning algorithms that identify the biggest digit in each image. Several algorithms have been used in the report. First, the logistic regression and linear SVM are used which lead to relatively lower precisions. Second, a forward neural network completely developed by the team was implemented. Finally, a convoluted neural network was trained and tested on the preprocessed dataset which showed the best performance.**

I. INTRODUCTION

The MNIST database [1] is a set of handwritten images that is popular for training and testing of Machine Learning algorithms [2].

II. PREPROCESSING

The provided images in the Modified MNIST include 3 numbers that are rotated and scaled from the MNIST dataset and are written on random backgrounds. Some samples of the train dataset with their associated outputs are shown in Figure 1.

The format for the images is 8-bit grayscale image, thus each pixel has 256 shades of gray represented by numbers 0 (black) to 255 (white) as shown in Figure 2.

Before, the data are used for training, the following preprocessing steps are used.

A. *Thresholding*

Since the numbers in the dataset match the 255 shade, a simple idea for preprocessing is to use *image thresholding*. The idea of thresholding is to compare the pixel values of the input image f with some threshold T and make a binary decision for the output binary image g as below

$$g(i,j) = \begin{cases} 1 & f(i,j) \geq T \\ 0 & f(i,j) \leq T \end{cases} \tag{1}$$

for all i, j where i, j represent the coordinates of the ij^{th} pixel [3].

The output of this filter is shown in Figure 3

*Kaggle Team name: BetaGo, Best Score: 0.95899

[1]Yingnan Zhao, Student ID: 260563769, Electrical Engineering, McGill University, nan.zhao2@mail.mcgill.CA

[2]Vincent d'Orsonnens, Student ID: 260746099, Software Engineering, McGill University, vincent.dorsonnens@mail.mcgill.ca

[3]Hamed Layeghi, Student ID: 260524499, Electrical Engineering, McGill University, hamed.layeghi@mail.mcgill.ca

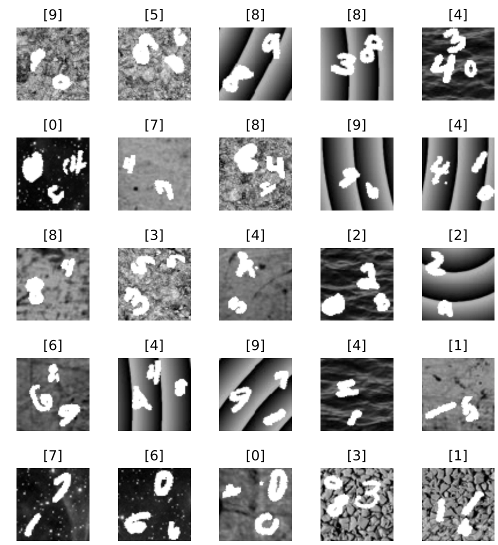

Fig. 1. 25 Random Samples of the original train dataset

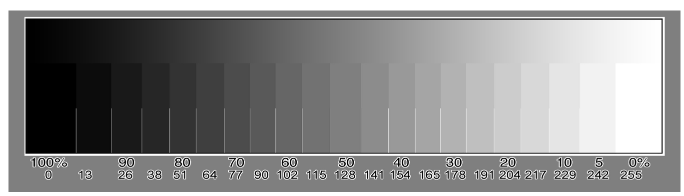

Fig. 2. 8-bit Grayscale Shades of Gray

B. *Median Filter*

As can be seen in Figure 3, there are some small white areas in some of the images that can act as undesirable noise. Median filtering is one method to remove the noise from the images and smoothen the edges. The main disadvantage of median filter is the damaging of thin lines and sharp corners [3].

The idea of median filter is to choose the median of the pixel values in a neighborhood of the given coordinate. These neighborhoods could be defined as disk, square, or any other shape of interest.

The output of the median filter using a disk of radius 1 applied on thresholded images are shown in Figure 4.

C. *Biggest Number*

The output of this filter on thresholded images are shown in Figure 5.

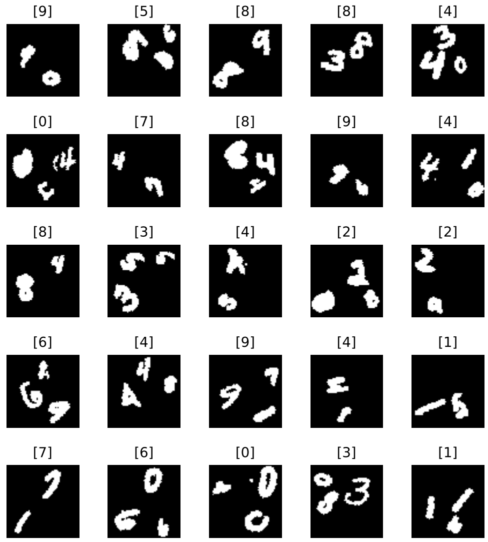

Fig. 3. Output of thresholding on images from Figure 1

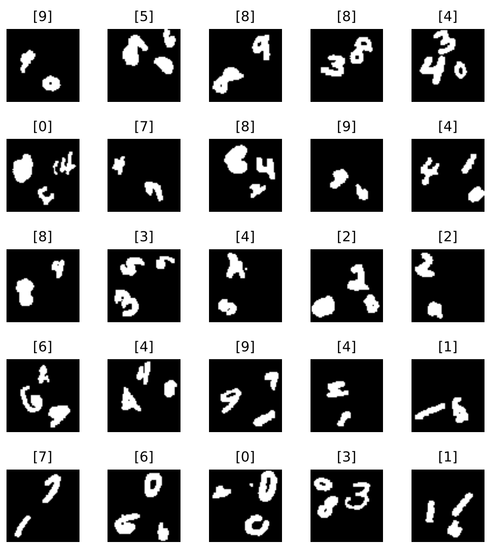

Fig. 4. Output of median filter on thresholded images from Figure 3

III. ALGORITHMS

A. Logistic Regression

B. Fully-connected Neural Network

The second algorithm we used to solve the problem is a fully-connected neural network. It was implemented by hand using the Numpy library. We trained the network using backpropagation and minibatch gradient-descent. Because this is a classification problem, the last layer of the network is a softmax layer, allowing the network to output a probability distribution.

C. Convolutional Neural Network

IV. METHODOLOGY

To start with, we had to split the original dataset into a training and validation set. We decided to go with a simple 90/10 split. Then we applied our preprocessing steps defined in the Feature Design section to generate 4 new datasets based on the original one.

- The first new dataset is the thresholded version of the original one.
- The second is the one where the biggest number from each image in the thresholded dataset set are extracted.
- The third one is the augmented version of the first new dataset, where the images are rotated 27°and -27°.
- The fourth one is the augmented version of the second new dataset, with the same rotations as the third one.

All the transformations were applied to the original training, validation and test set, except for the 3rd and 4th dataset where the transformations were only applied to the training set, because we wanted the validation set to stay as close as possible to the test set. So the validation and test sets for the 1st dataset are identical to the validation and test sets of the 3rd dataset, same for the 2nd and the 4th. This gave us two datasets with 45,000 training and 5,000 validation examples, and two datasets with 135,000 training and 5,000 validation examples.

A. Logistic Regression

B. Fully Connected Neural Network

C. Convolutional Neural Network

V. CONCLUSIONS

APPENDIX

Appendixes should appear before the acknowledgment.

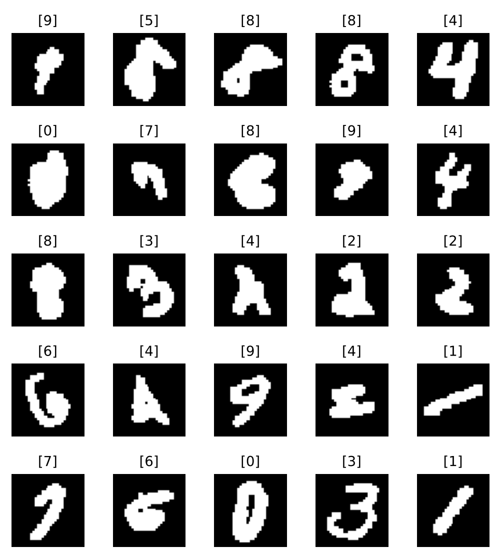

Fig. 5. Output of biggest number filter on thresholded images from Figure 3

STATEMENT OF CONTRIBUTIONS

Yignan implemented the convolutional neural network. Vincent implemented the fully-connected feedforward neural network and the linear classifiers. Hamed implemented preprocessing and tested the linear classifiers. Yignan and Vincent implemented the algorithms on Amazon AWS. All three contributed to the report.

REFERENCES

[1] Y. LeCun, C. Cortes, and C. J. Burges. Mnist handwritten digit database, yann lecun, corinna cortes and chris burges. [Online]. Available: http://yann.lecun.com/exdb/mnist/

[2] W. contributors, "Mnist database — wikipedia, the free encyclopedia," 2018, [Online; accessed 16-March-2018]. [Online]. Available: https://en.wikipedia.org/w/index.php?title=MNIST_database&oldid=829836442

[3] A. C. Bovik, *The essential guide to image processing*. Academic Press, 2009.