CS492C – Introduction to Deep Learning – Programming Assignment #1

School of Computing 20183309 Giyeon Shin

1. Unit of Layers in fully-connected network

The goal of this assignment is to construct the fully-connected network architecture to classify the image dataset to numeric digits (0-9). Since there was only a fully-connected layer available, I thought it would perform better if I preprocessed the features through dropout, regularizer and batch normalization operations on the layer. Thus, in one layer, the regularizer, batch normalization and dropout operations were performed in order, and the final activation function was relu function. Each of the corresponding parameters was determined through several experiments.

Index	1-unit	2-unit	3-unit	Dropout rate	L2-reg scale	Accuracy
1	256	256	10	0.20	0.005	0.456
2	256	256	10	0.25	0.005	0.4645
3	**256**	**256**	**10**	**0.30**	**0.005**	**0.4725**
4	256	256	10	0.35	0.005	0.467
5	256	256	10	0.40	0.005	0.466

Table 1 Accuracy change with dropout rate change

Index	1-unit	2-unit	3-unit	Dropout rate	L2-reg scale	Accuracy
1	256	256	10	0.30	0.006	0.4520
2	**256**	**256**	**10**	**0.30**	**0.005**	**0.4725**
3	256	256	10	0.30	0.004	0.4645
4	256	256	10	0.30	0.003	0.4710
5	256	256	10	0.30	0.002	0.4645

Table 2 Accuracy change with L2-reg scale change

Tables show that the absolute difference value by dropout rate or L2-regularizer scale change is not large. However, I thought that the relative value of accuracy was meaningful, so the dropout rate was fixed at 0.3 and the L2-regulizer scale was fixed at 0.005. Then, the experiments were conducted to find the best unit for the first and the second layer with fixed hyperparameter values.

Index	1-unit	2-unit	3-unit	Dropout rate	L2-reg scale	Accuracy
1	550	256	10	0.30	0.005	0.4615
2	**500**	**256**	**10**	**0.30**	**0.005**	**0.4835**
3	450	256	10	0.30	0.005	0.4650
4	400	256	10	0.30	0.005	0.4665

5	350	256	10	0.30	0.005	0.4615

Table 3 Accuracy change with the unit of the first layer change

Index	1-unit	2-unit	3-unit	Dropout rate	L2-reg scale	Accuracy
1	**500**	**256**	**10**	**0.30**	**0.005**	**0.4835**
2	500	300	10	0.30	0.005	0.4645
3	500	350	10	0.30	0.005	0.4710
4	500	400	10	0.30	0.005	0.4645
5	500	450	10	0.30	0.005	0.4680

Table 4 Accuracy Change with the unit of the second layer change

In fully-connected layer network with depth 3, I fixed each units of each layers to find the best unit of the models. When the first layer has 500 units and the second layer has 256 units, the model showed the best performance for accuracy.

However, the deeper layer of the model has the lower of the accuracy. This means that test data has a lot of different data compared to training data. As the layer becomes deeper, higher level features can be found. On the other hand, the decrease in accuracy means that the model tends to overfitting.

Index	1-unit	2-unit	3-unit	4-unit	5-unit	Dropout rate	L2-reg scale	Accuracy
1	500	256	256	256	10	0.30	0.005	0.434
2	500	256	256	128	10	0.30	0.005	0.4260
3	500	256	256	64	10	0.30	0.005	0.4380
4	500	256	128	128	10	0.30	0.005	0.4410
5	**500**	**256**	**128**	**64**	**10**	**0.30**	**0.005**	**0.4485**

Table 5 Accuracy of the 5-layer fully connected network

To overcome this overfitting problem, I needed to look at the shape of the dataset.

2. Analysis of datasets

The main task of this assignment is to classify the image dataset to numeric digits (0-9). Unlike the numeric digit dataset used by MNIST in general, we should deal with more complex dataset to work on. As I look at the dataset, there were some noises in the images, and there were many cases where the numeric digits were not in the vertical direction correctly. Since the numeric digits were also inverted upside down or reversed left and right, the features seemed to be insufficient to just learn it as a fully-connected layer. This institution comes from the fact that there are only 10,000 train dataset, while the test dataset has 50,000 image data. I decided to proceed with the data augmentation of the training dataset based on the judgment that the amount of training dataset is insufficient to learn the architecture by deep learning method.

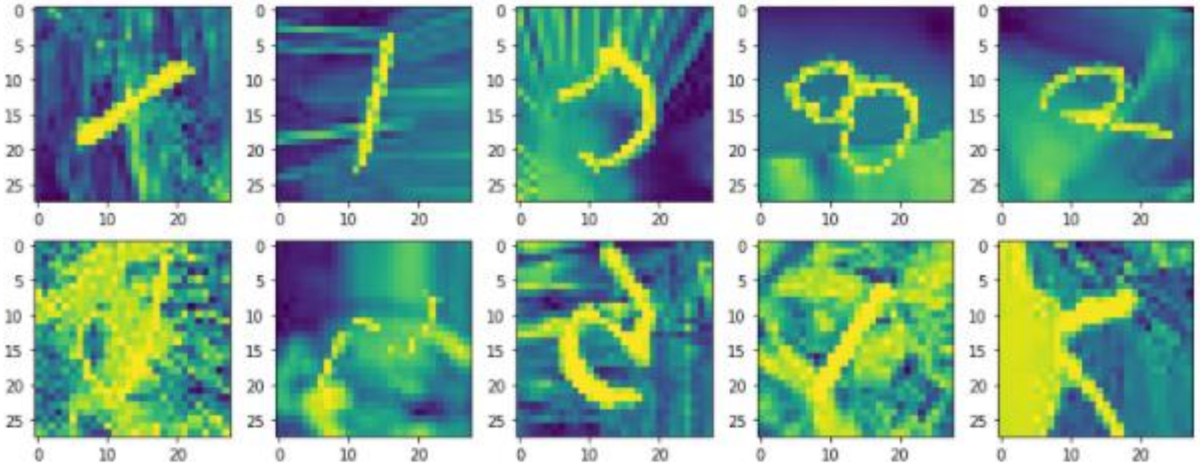

Figure 1 Plots of the training dataset

As a result of plotting the input data by matplotlib, the numeric digit is usually located at the center of the images. So, in the data augmentation phase, the image dataset was preprocessed through rotation or vertical flip. The condition for the data augmentation shows the best performance to create 15 new data and concatenate it with the original data. In this case, the rotate degree was determined from –pi to pi by Normal distribution, and the existence of vertical flip was determined by Bernoulli distribution. I chose Normal distribution and Bernoulli distribution respectively, because they showed better performance than the data generated in the Uniform distribution.

# of Generated Data	1-unit	2-unit	3-unit	Dropout rate	L2-reg scale	Accuracy
3	500	256	10	0.30	0.005	0.5960
7	500	300	10	0.30	0.005	0.6380
15	**500**	**350**	**10**	**0.30**	**0.005**	**0.6490**

Table 6 Accuracy of the number of generated data change (Uniform distribution)

First, I can see how the accuracy changes according to the number of data generated in the uniform distribution. When the number of generated data is 15, that is when a total data including original data is augmented with 16 data, the accuracy is increased. When more data is generated, the learning time becomes very long and the accuracy is similar to the model that has 16 generated data.

However, assuming that the probability of rotation follows a normal distribution, I modified the source code to follow the normal distribution because when one data is extended to 16 data, it has higher accuracy.

# of Generated Data	1-unit	2-unit	3-unit	Dropout rate	L2-reg scale	Accuracy
15	**500**	**256**	**10**	**0.30**	**0.005**	**0.6735**

Table 7 Accuracy of the rotation that is referred by Normal distribution

3. Final performance and Analysis

1-unit	2-unit	3-unit	4-unit	5-unit	6-unit	7-unit	Accuracy
500	256	10					0.6735
500	256	128	64	10			0.6685
500	256	128	64	50	35	10	0.6720

Table 8 Final performance by the depth change

As a result of integrating the above experiments, the model which is learned by the data extended through data augmentation processing has better performance than the model which is learned by only original data. The accuracy was highest when the first fully connected layer has 500 units and the second fully connected layer has 256 units and the remaining fully connected layer unit was also the unit with the highest accuracy in the experiment.

The noticeable one in the final experiment is that the depth of the network does not significantly affect the accuracy. This result is in contrast to the experiment described in No.1 session, and the experiment conducted in No.1 session tended to overfit the model as the layer became deeper. Because of the diversity of the number of data and the lack of the number of data, the test data showed poor performance. However, if we extend the data pool through data augmentation processing, the diversity of the data is sufficient. So, the model which extend the data pool had better performance, but the accuracy did not change as the layer became deeper. It seems that the fully-connected layer does not extract enough features to classify the training data. So I think using different kinds of layers like convolutional layers in this situation will give better results.