COMP 551 Assignment 2

Li, Haoxi, Qiu Zhuo, Peng Linya

March 9, 2023

Abstract

The CIFAR-10 dataset is a popular dataset which are presented in many research papers of machine learning and used to test various training models. In this project, the normal multi-layer perceptron(MLP) model and a model with convolutional neural network(CNN) is applied to learn this dataset and their performance are analyzed with different hyperparameters used. After several times of training and predicting with different models and hyperparameters, it is discovered that the CNN models with convolutional layers and fully connected layers perform better than normal MLP model which only has fully connected layer. Additionally, different activation functions, hyperparameters, regularization and normalization all have effect on the final performance of the models.

1 Introduction

The purpose of this project is testing the performance of different MLP and CNN models on the CIFAR-10 dataset. Several different models with different structures and hyperparameters are used to be trained on the CIFAR-10 dataset, which consists of 60000 images stored as an array of their pixel values and separated as training and testing sets. The MLP model is built from scratch and can take number and width of hidden layers and activation function as input. The CNN model is built with PyTorch package. The expectation of accuracy of MLP and CNN models is respectively 50% and 85% based on the result of the GitHub repository "CIFAR10 _CNN _MLP" [1]. Overall, the 2-layered MLP model has an accuracy nearly 50%, and the CNN models have an accuracy of 70% approximately, since the structure of the trained CNN models are relatively simple comparing to the CNN model used in the repository mentioned above.

2 Datasets

The CIFAR-10 dataset consists of 60000 32x32 colour images in 10 classes, with 6000 images per class. There are 50000 training images and 10000 test images. The dataset comprises six batches, five dedicated to training and one for testing, each containing 10,000 images. The test batch has precisely 1000 images randomly selected from each category. In our project, we combine the five training batches into one big training set for a total of 50000 images, and each image represents by a vector of length 3072, which is the combination of RGB values ($32^2 \times 3 = 3072$).

3 Results

3.1 MLP with different number of hidden layers

During the implementation, we used grid search to determine the best learning rate and max iteration for each MLP. The MLP with nonlinearity has significantly improved the accuracy of predicting the category. But after one hidden layer, increasing the depth doesn't significantly affect the accuracy.

3.2 MLP with different activation function

In these three (ReLU, leaky-ReLU, tanh) activation functions, ReLU has the best performance. Leaky-ReLU has only a 2% difference lower than ReLU, and tanh has more than a 5% difference lower than

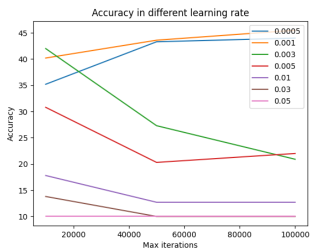

Figure 1: grid search for one hidden layer

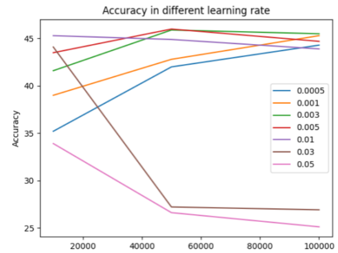

Figure 2: grid search for two hidden layer

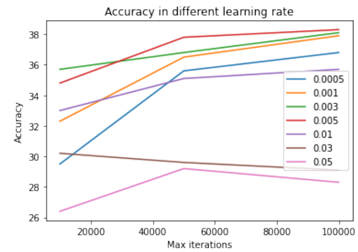

Figure 3: grid search for no hidden layer and linear activation function

ReLU. The ReLU activations are better than tanh and sigmoid. One of the reasons that ReLU has the best performance is that ReLU has a sparsity property, which means it can activate only a subset of the neurons in the network, resulting in more efficient use of network resources.

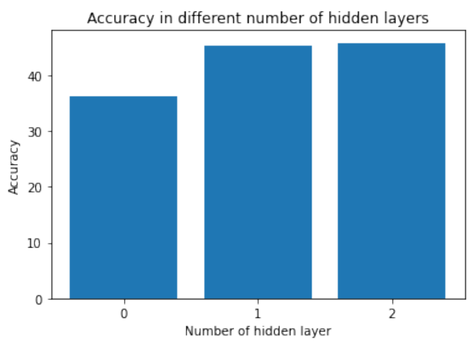

Figure 4: Accuracy in different number of hidden layer

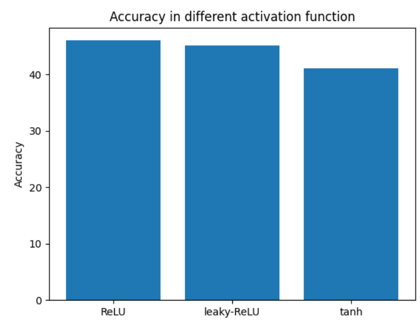

Figure 5: accuracy in different activation function

3.3 MLP with different regularization

Applying L2 regularization in MLP results in about a 5% improvement in accuracy compared to non-regularization MPL. Using L1 regularization doesn't improve accuracy. Instead, it has a 20% to 30% difference lower than not using regularization.

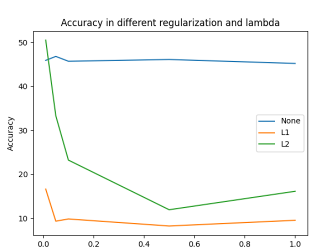

Figure 6: accuracy in different lambda

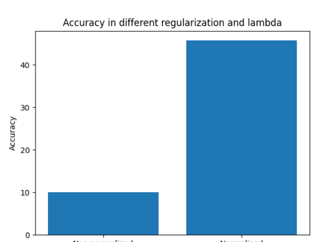

Figure 7: accuracy in normalized or non-normalized data

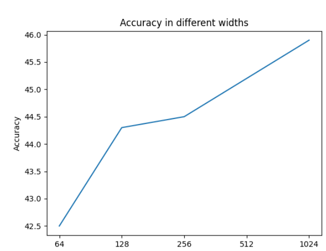

Figure 8: Accuracy in different width of hidden layer

3.4 Normalizing data

Normalizing data give a significant improvement in accuracy; non-normalizing data result accuracy of 10% no matter what learning rate is being used and how many hidden layers are in MLP (Figure 5).

3.5 MLP with different width

The width of the hidden layer is not a decisive factor in the accuracy, but increasing the number of nodes in the hidden layer will still raise a small amount of accuracy. In our project, we test one-hidden layer MLP with a width of [64, 128, 256, 512, 1024]. After 64 nodes, every time we double the number of nodes, the accuracy will increase by about 0.5% to 1% from the original accuracy (Figure 8). However, the runtime of the gradient descent will also double when we double the number of nodes.

3.6 CNN model

The CNN model with 2 convolutional layers and 2 fully connected layers, trained with the SGD optimizer with a batch size of 32 has a significantly higher accuracy than MLP models, which is around 70% (Figure 9). The ideal learning rate of this CNN model is between 0.03 and 0.05. Due to hardware constraint, the max training epochs is set to be 50, but the improvement of higher training epochs will not be significant.

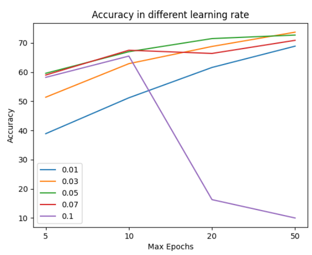

Figure 9: grid search for CNN model

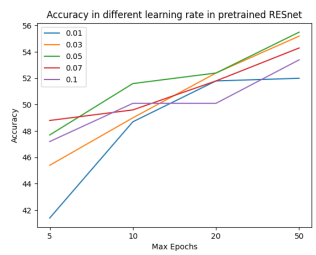

Figure 10: grid search for pretrained ResNet model

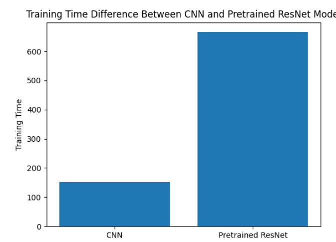

Figure 11: running time comparison between CNN and pretrained ResNet model

3.7 Pretrained ResNet model

The pretrained ResNet model with convolutional layers froze and 2 fully connected layers have a relatively lower performance than the CNN model used in the section above, which is about 56%(Figure 10). The number of fully connected layers are chose to be 2 for better contrast to the normal MLP model with 2 hidden layers and a reasonable training time. The effect of adding and removing fully connected layers is similar in MLP models and CNN models. Higher max epochs can possibly lead to a higher accuracy since the curve is still observed to have a high tendency to increase when the max epochs are set to be 50. The pretrained ResNet model has a significantly longer training time than CNN model, since it has a more complex structure(Figure 11).

3.8 Discussion and Conclusion

After training and testing various MLP and CNN models, the project provides an overall image of their approximate performance on the CIFAR-10 dataset. With optimal hyperparameters and activation function, as well as other optimizing techniques (regularization and normalization of data), the SGD CNN model used in this project can achieve a successful rate of 50%. In the meanwhile, the CNN model and pretrained ResNet model used in this project have significantly higher performance than the normal MLP model, with a successful rate of 70%. In this project, the variation of batch size, descent method(SGD) and structure of CNN models are not deeply investigated due to hardware constraints. The models with more convolutional and fully connected layers, more training epochs, an optimal batch size and better descent method(ADAM) will possibly have a longer training time and higher performance.

3.9 Statement of Contributions

The implementation of MLP model was done together. The training and testing of MLP model in different conditions were done by Zhuo Qiu. The training and testing of CNN and pretrained ResNet models with different hyperparameters were done by Haoxi Li. The report was written by Linya Peng.

References

[1] AnupamMicrosoft. *CIFAR10 _CNN _MLP*. 2019. URL: https://github.com/AnupamMicrosoft/CIFAR10_CNN_MLP.