STATS 231A Project 0 Report

Name: Xiaofeng Gao
ID: 304880356

09 Oct. 2017

1 Learning LeNet

The first part of the projects aims at learning LeNet, a five-block ConvNet introduced by LeCun et al. in 1998, on CIFAR-10 dataset. The training and validation error is shown as below.

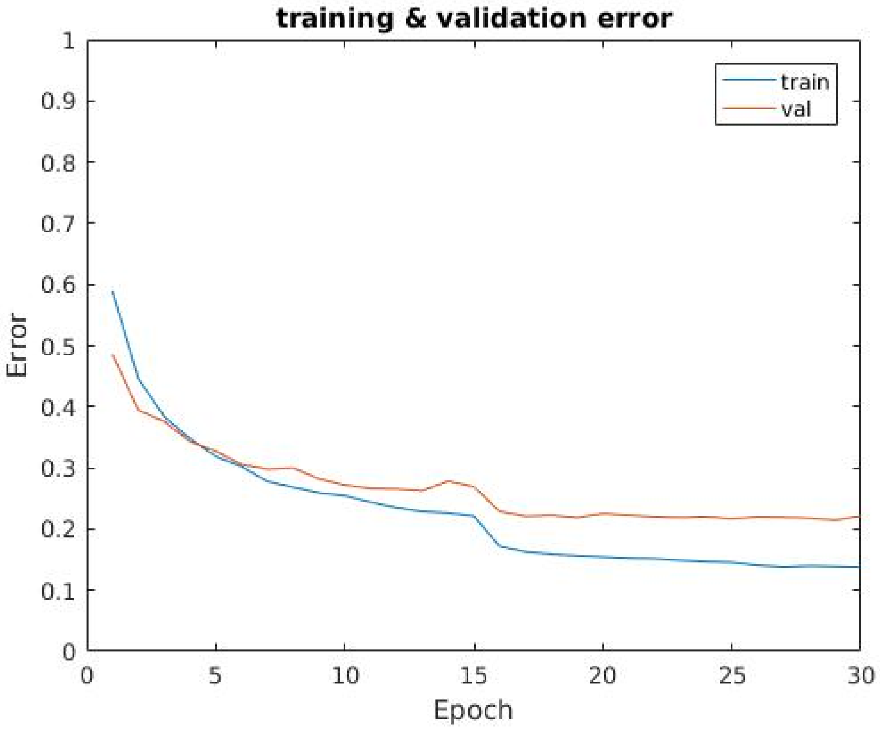

Figure 1: Training and Validation Error for LeNet

We can see that training and validation error drop as the number of epoch increases. In most of the time, the training error is lower than the validation error, which means we somehow over-fit the data.

2 Cutting Some Blocks

After learning the original LeNet, we are required to keep Block5 and train three new ConvNets, with a) Block1; b) Block1 and Block2; c) Block1, Block2 and Block3 respectively. The training and validation error is shown as below.

As number of hidden layer increases, we can see that both training error and validation error decrease. Compared to the full LeNet, however, cutting off single block (Block4) doesn't seem to have too much influence on the performance. Since CIFAR-10 is a small dataset, it is not necessary to have many layers, which may cause over-fitting.

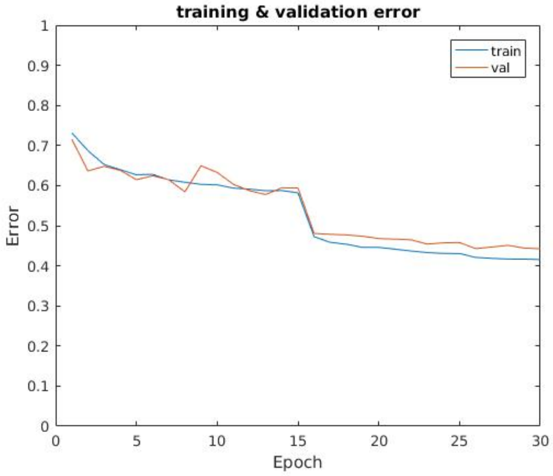

Figure 2: Training and Validation Error for ConvNet with Block1+Block5

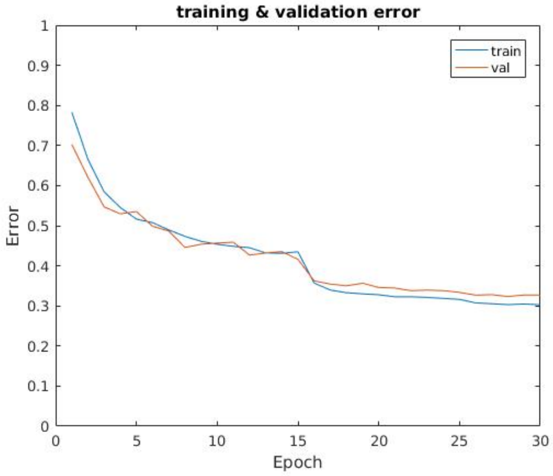

Figure 3: Training and Validation Error for ConvNet with Block1+Block2+Block5

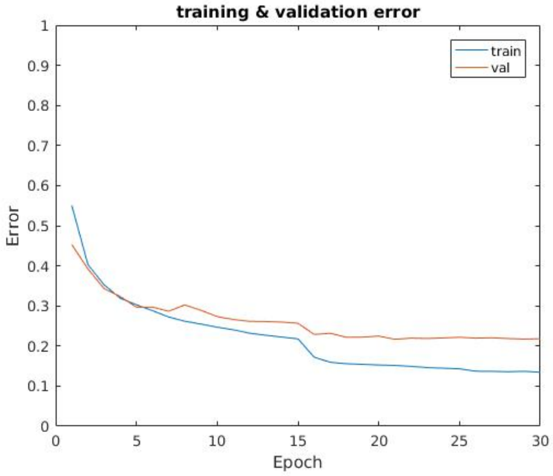

Figure 4: Training and Validation Error for ConvNet with Block1+Block2+Block3+Block5

Block	(a)	(b)	(c)	(full)
Training Error	0.4156	0.3028	0.1347	0.1380
Validation Error	0.4425	0.3270	0.2182	0.2213

Table 1: Final Training and Validation Error for different ConvNets

3 Visualizing Filter Response

The final step is to visualize the filter response of the first layer. It will give us some ideas about the features we have learnt using the ConvNet. The result is shown below.
We can see that the first layer learns low level patterns (edges, corners, colors, etc.) from the picture. Different filters focus on different patterns. For example, the 15th filter (row 2, col 7) will be activated by green pixels.