

Machine Learning Practical Coursework 4

1 Research Questions

As part of this coursework the performance of a more complex neural network is evaluated on the given CIFAR-10 and CIFAR-100 data sets. The neural network used is a Convolutional Neural Network(CNN) proposed by Yann LeCun and Yoshio Bengio [1]. Some variations were carried out to better understand the power of CNN and results are compared with the baseline model from Coursework 3 in similar instances.

Experiment 1: Vary the Number of Convolution layers - max pool layers

The idea of having a convolution layer is central to learning a particular set of features in the given data. It is hypothesised that with increasing the number of convolution layer a better representation of the input could be learned. Max pool layers reduce convolution layers to form a feature map by selection only maximum values in the given filter size. In this experiment different architectures are tried, one with two max pool layers and another with just one max pool layer. An architecture with only one convulution and one max pool layer is also evaluated.

Experiment 2: Vary the size of window for max pool layer

The prime motivation for this experiment is to reduce the number of parameters for downstream layers and record the effect that it might have on the accuracy of the model. Such an increase in window size might also reduce the time taken by the model to train.

Experiment 3: Different optimizers

The optimizers that were evaluated for the architecture include Adagrad, Adam and Gradient Descent, with learning rates 0.01 each. Motivation behind this experiment was to evaluate how different optimisers perform on the convolutional neural networks. Each optimisation technique uniquely calculates the minimum loss value and has has different set of hyperparameters, thus offering more control.

Experiment 4: Different function for feature maps: max-pool, avg-pool

A max pool layer reduces the features learned by the previous layer by taking a max over a grid size. This new formed representation is known as feature map and highlights distinct properties in block units of the feature representation. Another approach called average pooling takes the average of the feature values in the grid block instead of selecting only the highest value feature. Both the techniques have been explored in this experiment for different grid sizes.

Experiment 5: Different number of filters

The motivation behind this experiment is the fact that each filter correspond to learning a feature about the given data and hence increasing the number of features could help learn a more complex model. For this purpose a set of different filter in convolution layers are experimented with.

Experiment 6: Dropout

The results from above experiments expressed a sense of overfitting. Dropout is a powerful technique for reducing overfitting. In this technique randomly some values of the learned parameters are set to zero. Hence inducing a 'drop-out'. This leads to some loss in learning but would be effective if the model sees a large amount of similar data and later might fail to generalize over a different settings.

2 Method

A general architecture of the Convolution network is used for all the experiments detailed as follow:

Convolution – ReLu – maxpool – Convolution – ReLu – maxpool – fullyconnected – output (1)

32 5X5 filters are used for the first convolution layer and 200 5X5 filters are used for the second convolution layer. Each convolution layer is followed by a rectified linear unit(ReLu) layer and a max pool layer with window size of k=4. A fully connected layers of ReLu non-linearity having 1000 neurons is used. Adam Optimizer is used with default learning rate of 0.01. The weights for each layer are initialized using the Xavier initialization method and a constant value of 0.1 is used to initialize the bias. This is the general architecture followed for all the experiments unless the experiments deals about changing a specific parameter and recording its effects.

A varied number of epochs were tried to find where the convergence occurs for on CIFAR-10 data for the architecture. It was found out that this number is less than 10 epochs for given model. Thus, all the models have been trained for 20 epochs, since there are some complex architectures tried which might require more time to converge. The same has been visualised in Figure 1. This technique of stopping the training early, when overfitting starts to occur is also known as early stopping and one of the widely used techniques to avoid overfitting on the training data. Other reason for keeping the number of epochs was the time taken for the model to run. The experiments were conducted on a mix of personal computer having RAM 8GB and Intel i5 processor, DICE machine and University of Edinburgh msccluster.

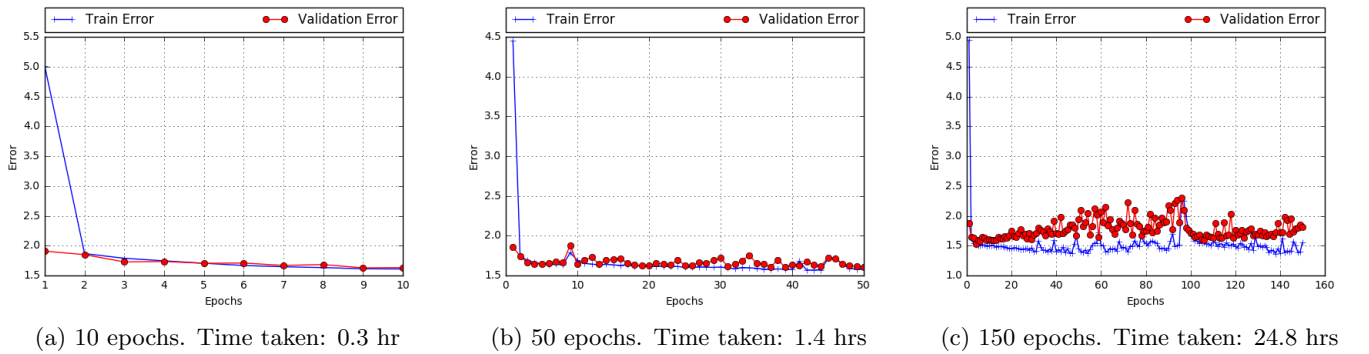


Figure 1: Error graph for different number of epochs

Experiment 1: Vary number of max pool layer

In this experiment different architecture were explored. One had a two convolution layers followed by a max pool layer each and the second had two convolution layers and only the last one followed by a max pool layer. This kind of architecture was motivated by success of many deeper networks such as AlexNet [2]. Not the full extent of Alexnet has been implemented but the idea of putting two convolution layers in succession has been borrowed and implemented. The layers have same configuration as explained previously. CIFAR 100 data set is used to experiment on one convolution layer-one max pool layer architecture and two convolution-two maxpool layers architecture.

Experiment 2: Vary the window size of max pool filter

In this experiment two different window size of filter are explored, $k=2$ and $k=4$. This means that a max operation would be done over a grid of 2×2 and 4×4 respectively, the remaining architecture of the model remains the same. For CIFAR 10, window size of 2 and 4 are evaluated, and for CIFAR 100 only 2

Experiment 3: Different optimizers

Optimisation is a technique by which a minima or maxima of a given equation is calculated. In the experiments for this coursework, error equation is the softmax error function. Three different optimisers are tried: Adam, Adagrad and Gradient Descent. The optimisers search for the smaller value in the vicinity of the starting point and then move towards the direction where error is minimal. The step length of movement is controlled by learning rate, a larger learning rate means a larger step in that direction. This may often lead to missing the minima. Learning rate is a hyperparameter and has to be tweaked to get a good setting. In this experiment three different optimization techniques each with learning rate 0.01. CIFAR-10 is experimented with all three and CIFAR-100 is experimented with Adam and Gradient Descent optimisation techniques.

Experiment 4: Different function for feature maps: max-pool and avg-pool.

In the above architecture the max pool layer is replaced by a average pool layer and experiments for two sets of grid size 2 and 4 are conducted for each of the algorithms. For CIFAR 100 average pool with $k=2$ is tried.

Experiment 5: Different number of filters

In this experiment the affect of changing the number of filters in the convolution layer is explored. The hypothesis is with increasing the number of filters more feature could be learned. This also mean that the computation time would be increased. Hence this presents a good opportunity to compare the results of increasing the number of filters with different number of max pool layer as in Experiment 1. For CIFAR-10 dataset the values of filters tried is 100,200,300 and 400, similarly for CIFAR 100 filter sizes of 200 and 400 are evaluated.

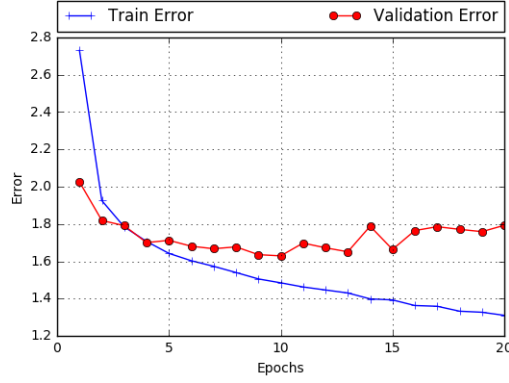
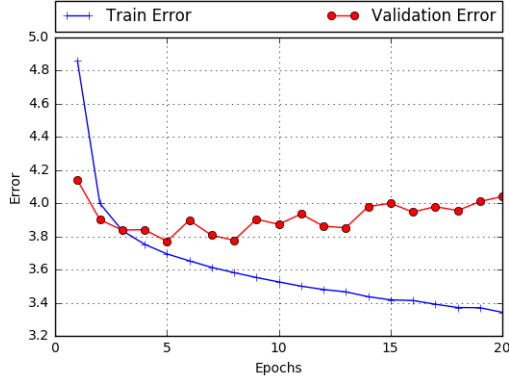
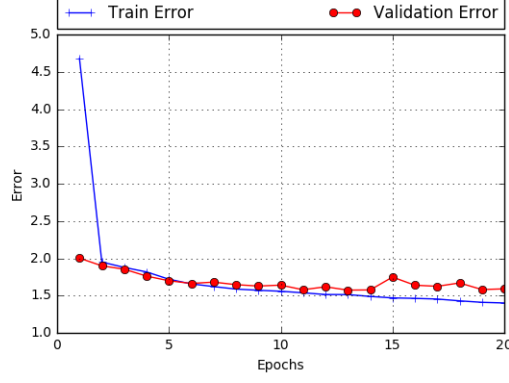
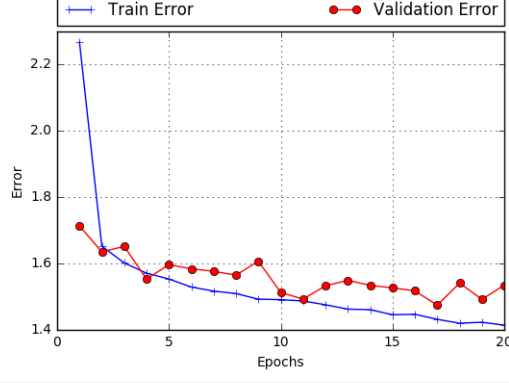
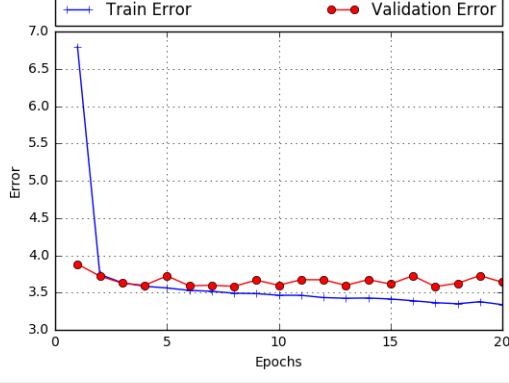
Experiment 6: Dropout

Dropout is a powerful regularization technique to reduce overfitting in the model. The neurons are randomly selected and ignored during training. For this experiment four settings are considered. Two values of dropout: 0.5 and 0.25 are applied before the max pool layer and after max pool layer. It is expected that applying a dropout after a max pool layer is not a good idea since this would lead to loss of vital information that was condensed using the pooling layer. For CIFAR-100 data set, the setting with dropout value 0.5 before each max pool layer is tried.

3 Results and Discussion

Experiment 1: Vary number of max pool layer

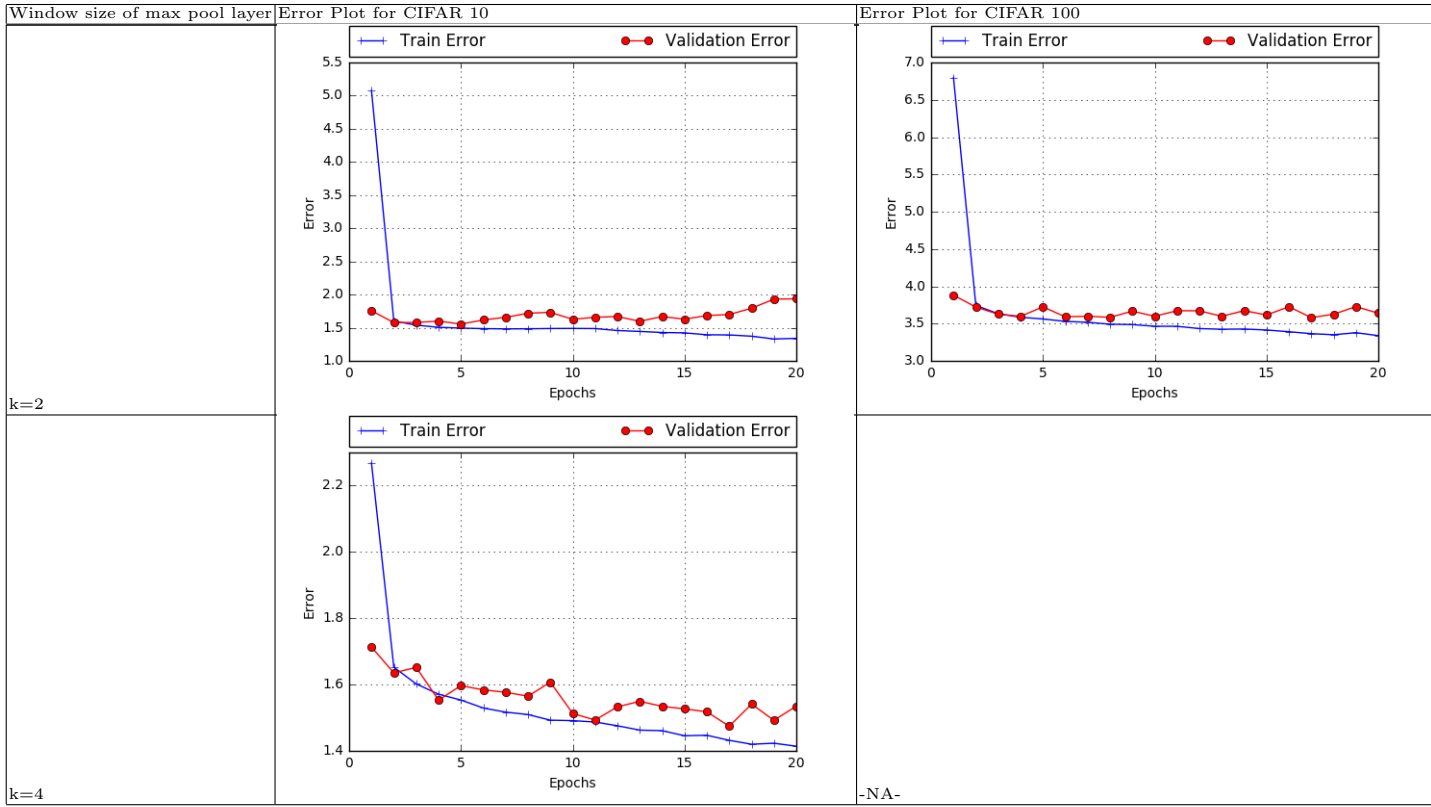
Table 1: Training and Validation Error plots for Experiment 1

Layer Variation	Error Plot for CIFAR 10	Error Plot for CIFAR 100
One Conv-One Max pool		
Two Conv-One Max pool		-NA-
Two Conv-Two Max pool		

Discussion: Table 1 plots the error plots for different convolution and max pool layer combinations. For first 20 epochs of training it is observed that the one convolution layer and one max pool layer model tends to overfit quickly for both CIFAR-10 and CIFAR-100 data sets. The effect of overfitting increases, however the fall in error rate is not much for the validation set.

Experiment 2: Vary the window size of max pool filter

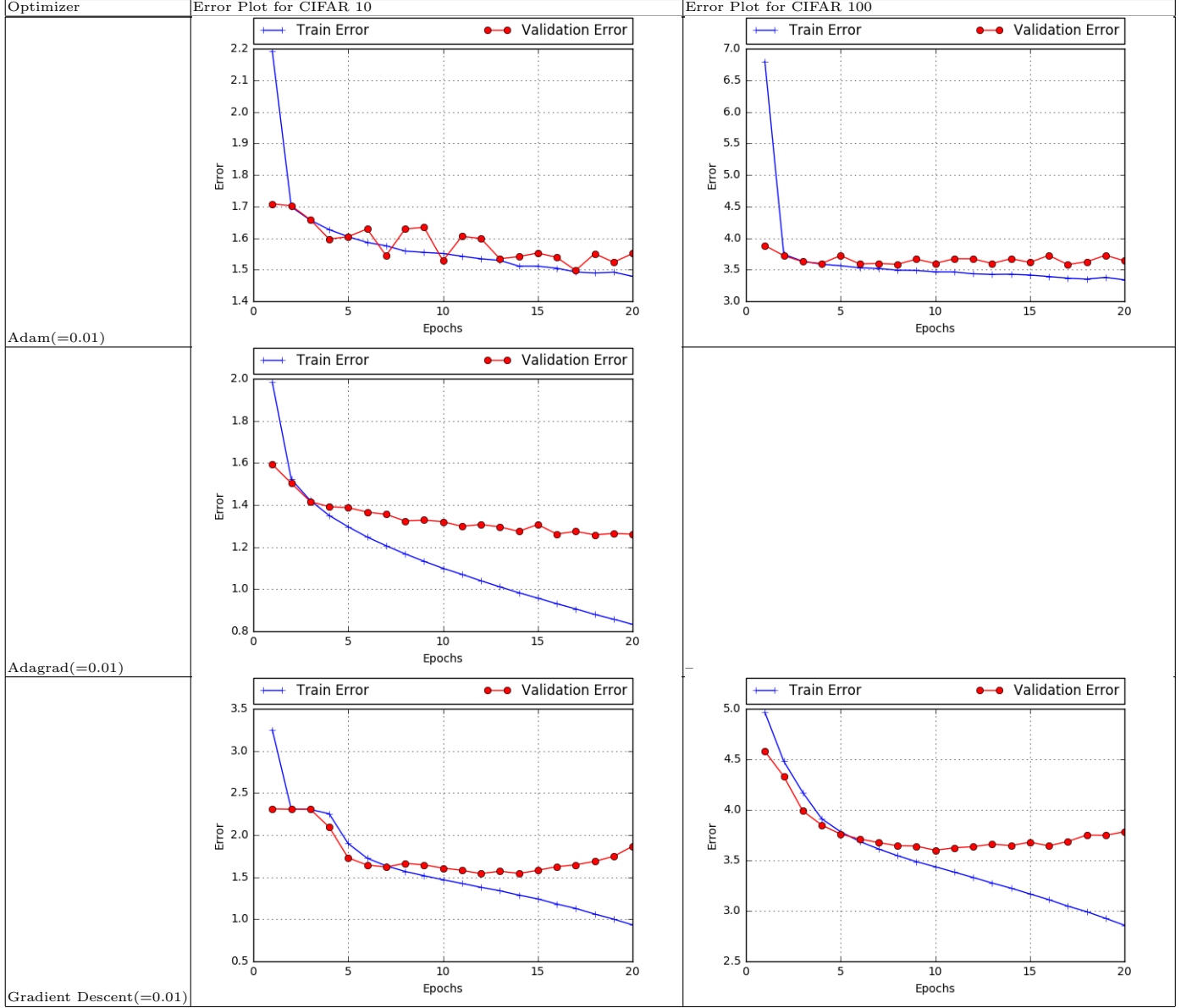
Table 2: Training and Validation Error plots for Experiment 2



Discussion: From Table 2 it can be concluded that the difference in size of max pool window had little impact in terms of overfitting or error loss value. However the time taken by each is greatly different. With k=2 setting it took 1.35 hours on the DICE machine to train the model and with k=4, it took 0.8 hrs to train. Due to such significant difference in training time, further experiments were conducted with k=4 value.

Experiment 3: Different optimizers

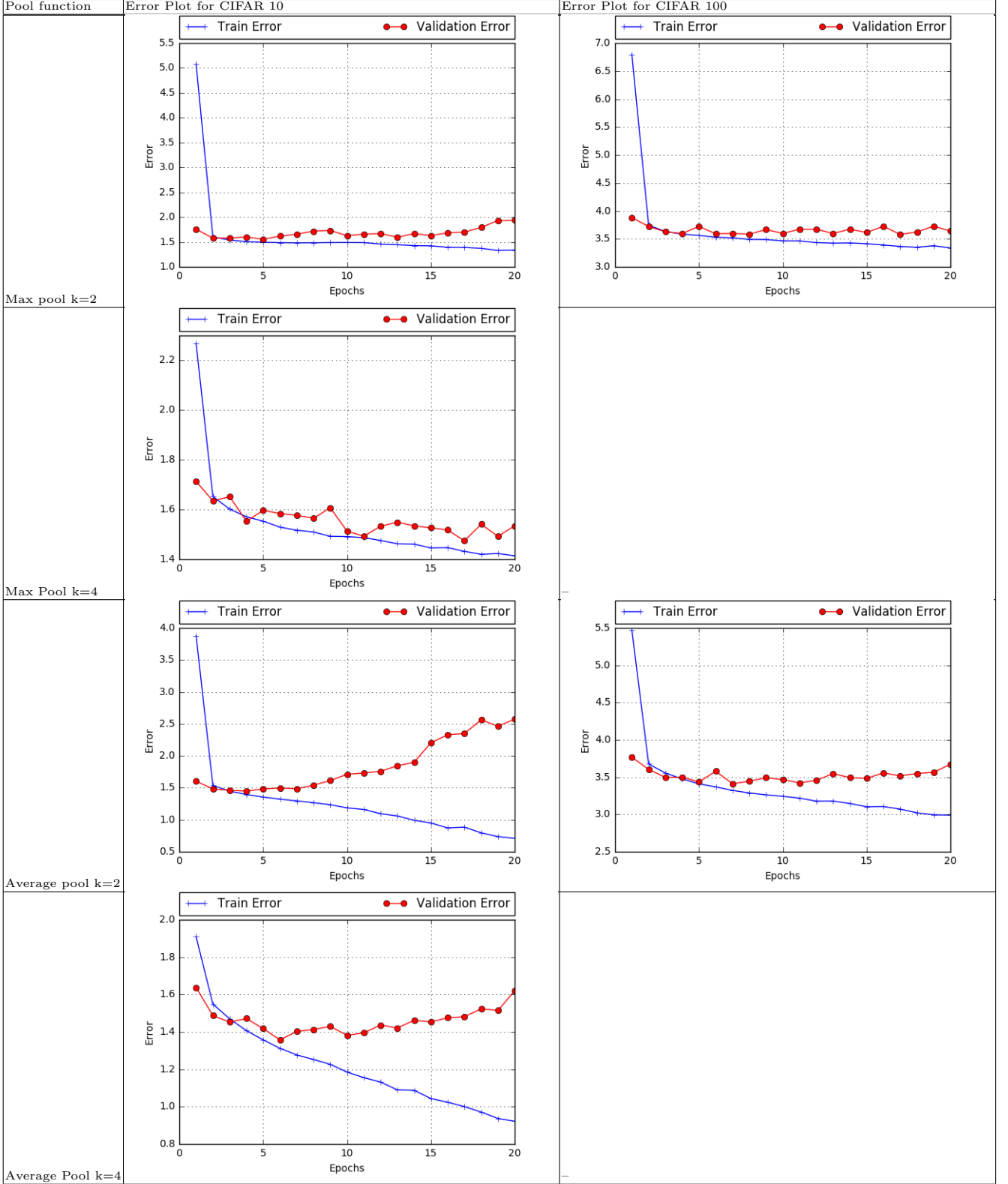
Table 3: Training and Validation Error plots for Experiment 3



Discussion: From Table 3, it can be inferred that Adagrad and Gradient Descent find a minima fairly quick and clearly overfit based on training data set, whereas Adam optimiser did not get stuck at any such local minima for both CIFAR 10 and CIFAR-100 dataset. This also motivated the decision to choose Adam as the default optimiser for all the experiments despite Adagrad optimiser having a lower error.

Experiment 4: Different function for feature maps: max-pool and avg-pool.

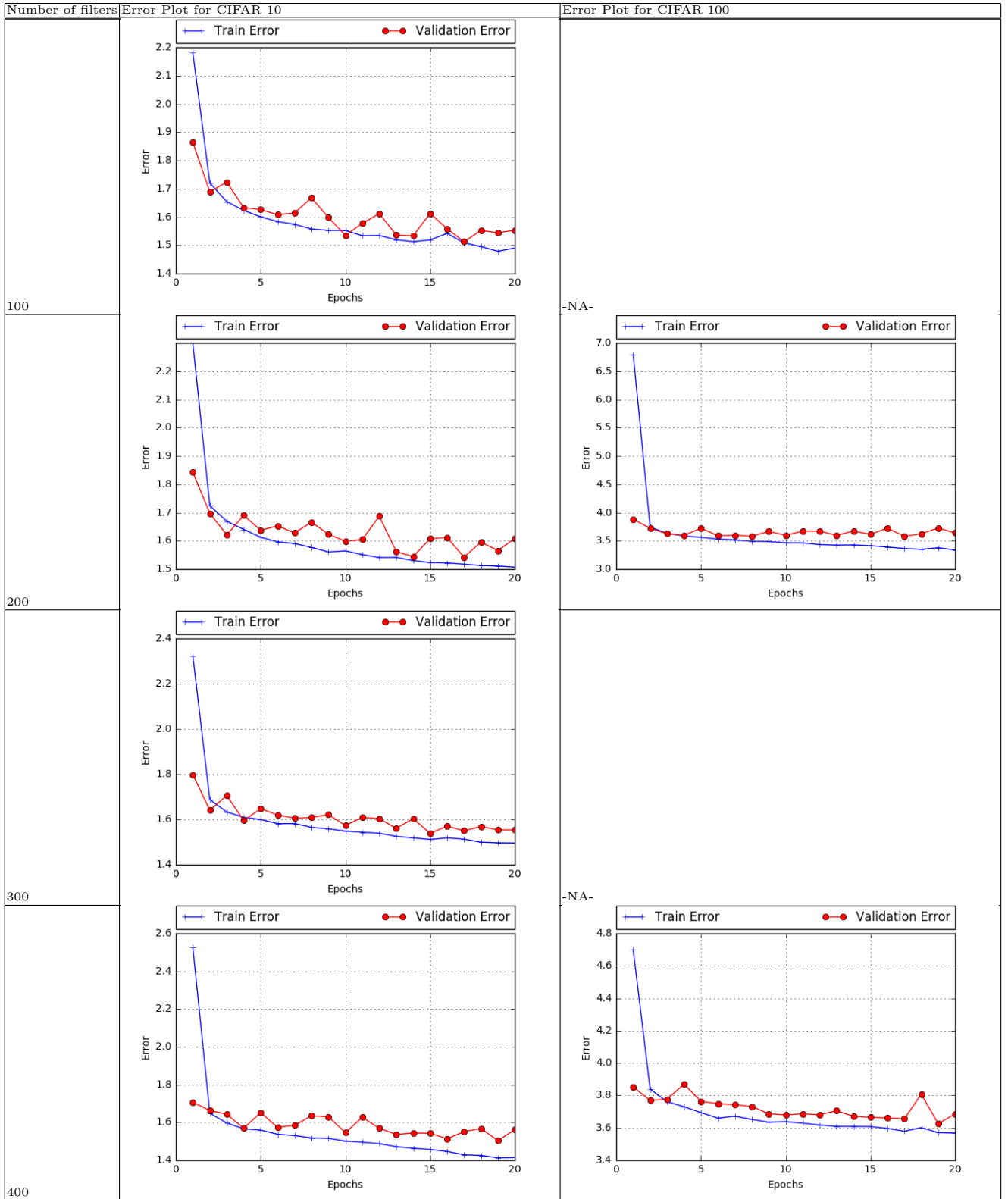
Table 4: Training and Validation Error plots for Experiment 4



Discussion: Table 4 details the results of using average pool and max pool for different window sizes. Average pool has a lower accuracy but tends to overfit early. This does show that average pooling is a good technique to replace max pool operation if overfitting could be controlled by various regularization or dropout techniques. This inference is valid for both CIFAR 10 and CIFAR 100 datasets.

Experiment 5: Different number of filters

Table 5: Training and Validation Error plots for Experiment 5



Discussion: The results from Table 5 for different number of filters in the convolution layer highlight that error decreases marginally on increasing the number of filters. Also there is not much effect on overfitting for both CIFAR 10 and CIFAR 100 data sets. Following this conclusion, the number of filters was kept fixed to 200 for all the other experiments.

Experiment 6: Dropout

Table 6: Training and Validation Error plots for Experiment 6

Dropout Rules	Error Plot for CIFAR 10	Error Plot for CIFAR 100
Before maxpool(=0.25)		-NA-
Before maxpool(=0.50)		
After maxpool(=0.25)		-NA-
After maxpool(=0.50)		-NA-

Discussion: From the results in Table 6 for different settings of dropout it can be concluded that in all cases dropout significantly reduced overfitting for both CIFAR-10 and CIFAR-100. The gain in error drop is not so significant in first 20 epochs when compared to previous experiments, but reduction in overfitting provides a good motivation to use dropout and train it for longer epochs.

4 Conclusion

Some comparisons with Coursework 3 [3]:

1. L2 regularization was an effective technique to reduce the overfitting for a multilayer perceptron neural model. Other techniques such as L1 regularization did not work so well. In this coursework, introduction of dropout technique highlighted that for all the values of dropout there was a significant drop in overfitting.
2. Overfitting seems to be largely present in most of the experiments conducted in Coursework 3. This has been major focus of this assignment and many techniques were adopted from the start to avoid such a thing in this Coursework. This included techniques such as early stopping, wherein training was stopped after 20 epochs against 100 epochs for Coursework 3

The contrast between a multilayer perceptron architecture and convolutional neural network architecture for same data set and similar experiments helped understand presence of issues such as overfitting and how different regularization techniques work in different architectures.

References

- [1] LeCun, Y. and Bengio, Y., 1995. Convolutional networks for images, speech, and time series. The handbook of brain theory and neural networks, 3361(10), p.1995.
- [2] Krizhevsky, A., Sutskever, I. and Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).
- [3] s1579563, Coursework 3, Machine Learning Practical(INFR11132), 2016-17