Arsalan Syed
CDATE 3

Deep Learning Assignment 1

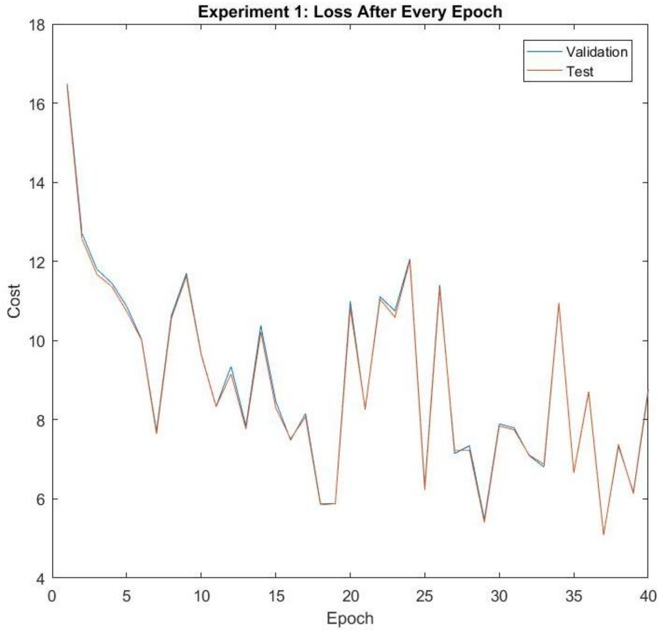
The purpose of this assignment was to train a neural network to classify 32x32 images from the CIFAR-10 dataset. The algorithm used to calculate the gradient of the cost function was successful and this evident from the fact that when compared to the numerically calculated gradients by the functions ComputeGradsNum and ComputeGradsNumSlow, the difference in the analytic result and the numerical result was _ and _ respectively. Using the function TestGradientCompute which calculates the relative error between the numerical and analytic results, the relative error for the gradient with respect to W is 9.3420e-08 and for the gradient with respect to b is 1.5131e-07. Both of these values are less than 1e-06 so it's safe to say that the analytic gradient was computed correctly.

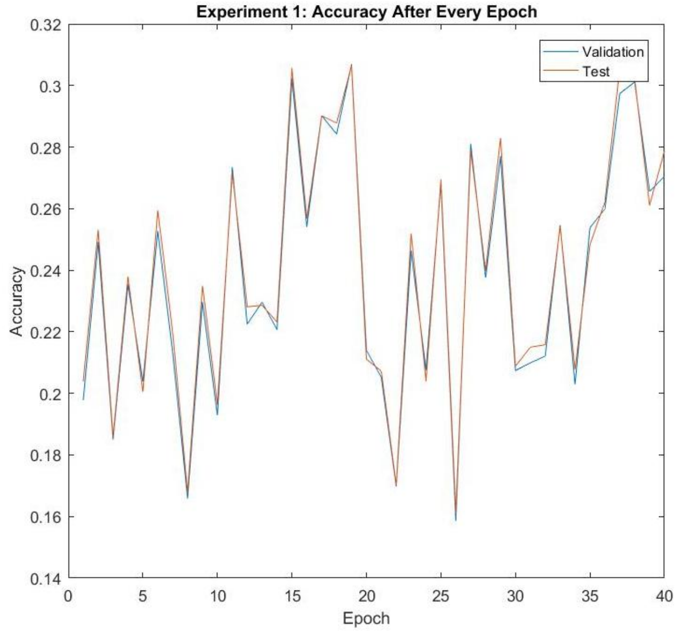
Results

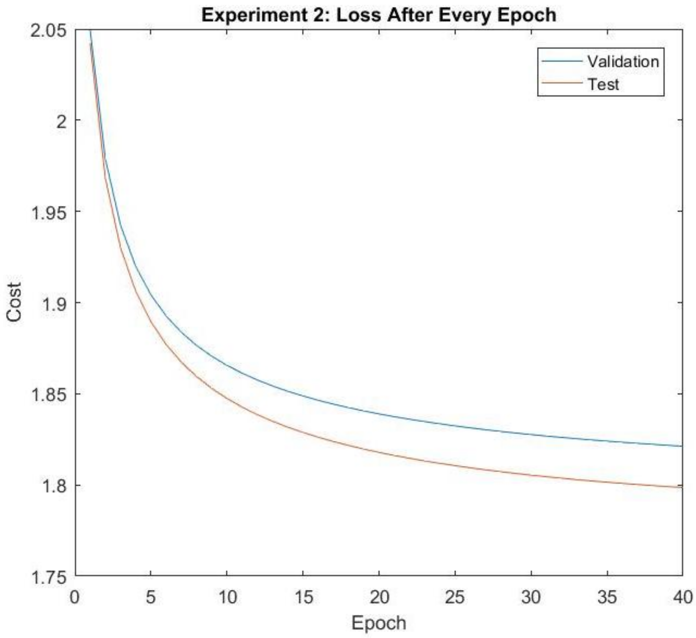
On the plain vanilla version of the learning algorithm which contained no improvements, the following accuracies were achieved given the parameters:

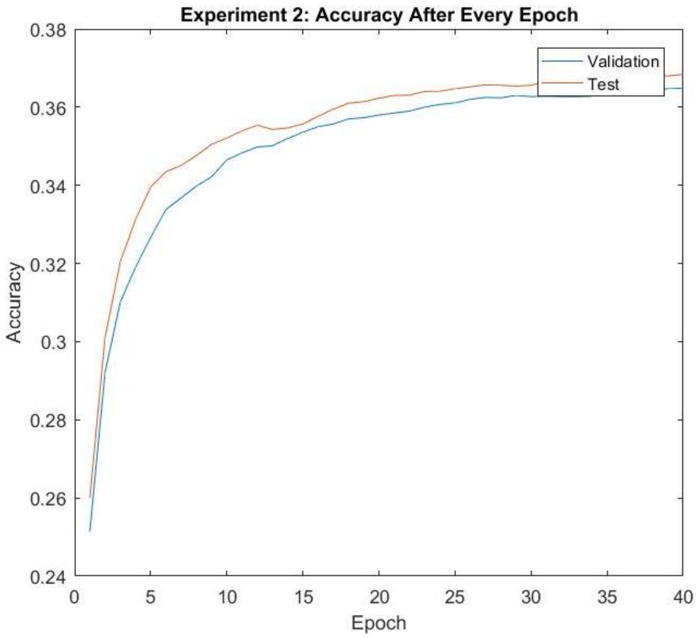
Experiment	Lambda	Number of Epochs	Batch size	Learning rate	Accuracy on test set (Cross entropy)
1	0	40	100	0.1	**0.2787**
2	0	40	100	0.01	**0.3685**
3	0.1	40	100	0.01	**0.3339**
4	1	40	100	0.01	**0.2187**

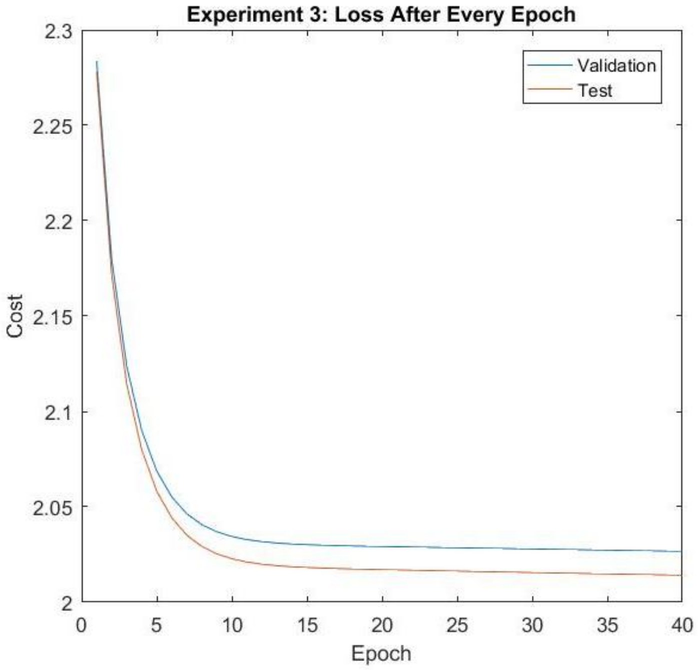
It is important to note that the last three experiments only differ by the parameter lambda which is a part of the regularisation term in the cost function. As lambda increases, the overall accuracy on the test set decrease. Although having a higher lambda would reduce overfitting, it can run the risk of making the model too simple and thus causing it to be underfitted. Having the correct learning rate is important because if it is too high, the gradients could be changed too much and completely miss an optimal point. Smaller learning rates are more likely to find these optimal points but they require more iterations to do so. For the results above, only 3 different files were used to form the datasets. The training set used data_batch_1, the validation set used data_batch_2 and the test set used test_batch. Below are the graphs of the total loss and cost function on each epoch for each of the experiments as well as a visual representation of the learned weights.

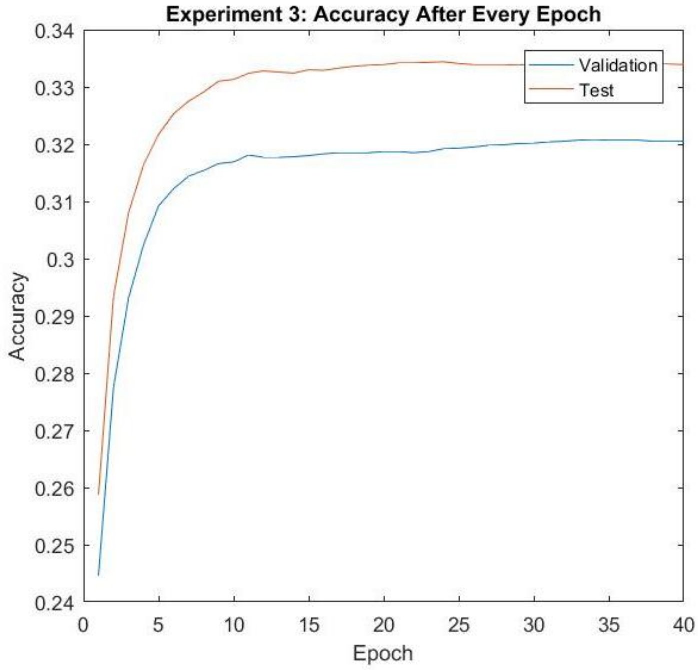

Experiment 1: Loss After Every Epoch
Validation
Test
Cost
Epoch

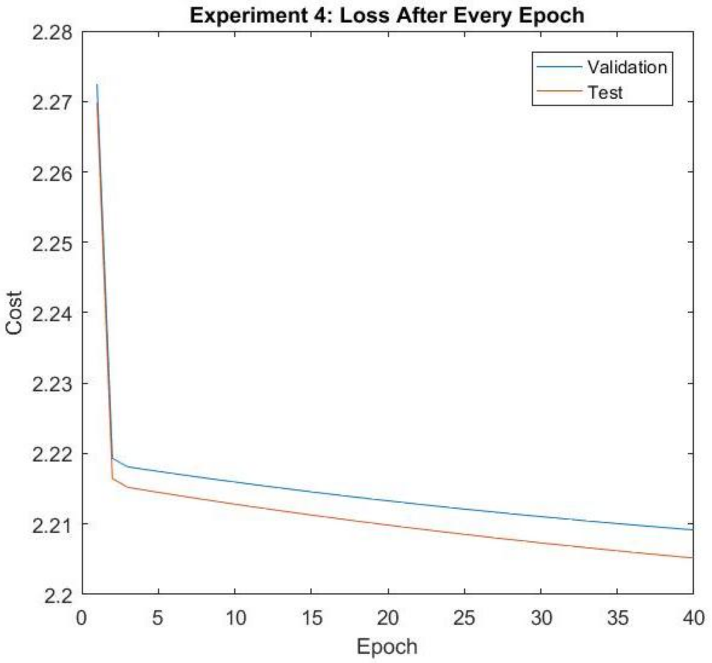

Experiment 1: Accuracy After Every Epoch
Validation
Test
Accuracy
Epoch

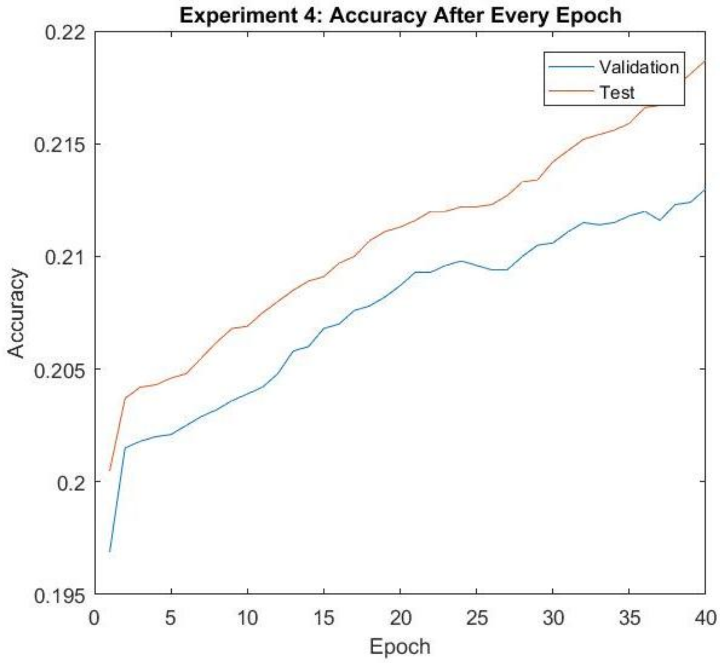

Experiment 2: Loss After Every Epoch
Validation
Test
Cost
Epoch


Experiment 2: Accuracy After Every Epoch
Validation
Test
Accuracy
Epoch

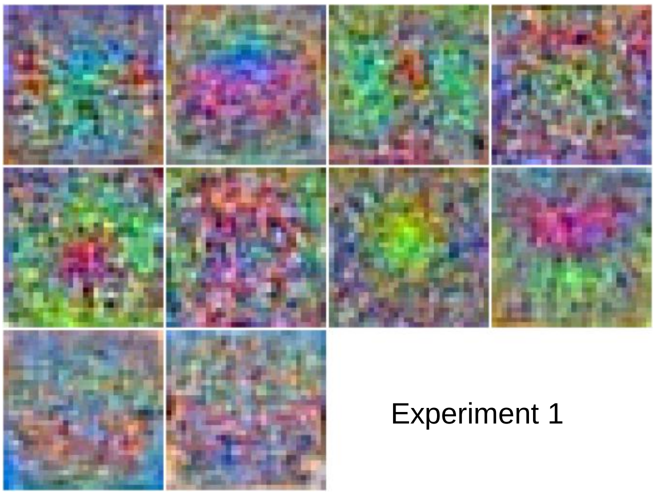

Experiment 3: Loss After Every Epoch
Validation
Test
Cost
Epoch


Experiment 3: Accuracy After Every Epoch
Validation
Test
Accuracy
Epoch

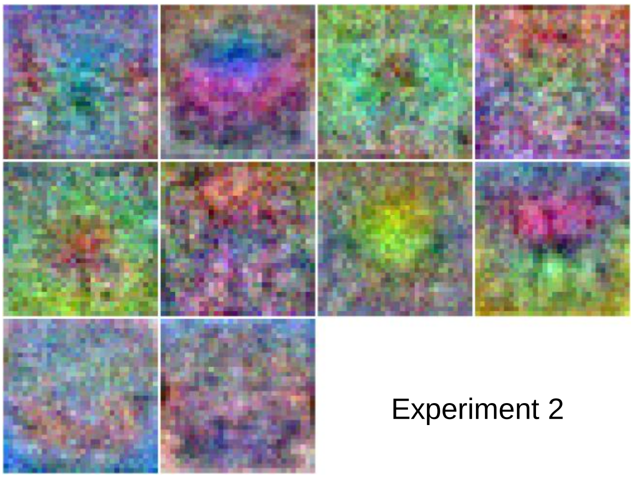

Experiment 4: Loss After Every Epoch
Validation
Test
Cost
Epoch


Experiment 4: Accuracy After Every Epoch
Validation
Test
Accuracy
Epoch

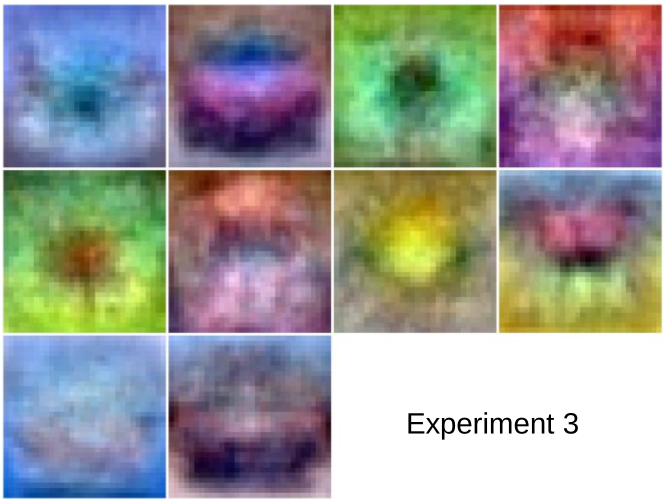
Arsalan Syed
CDATE 3



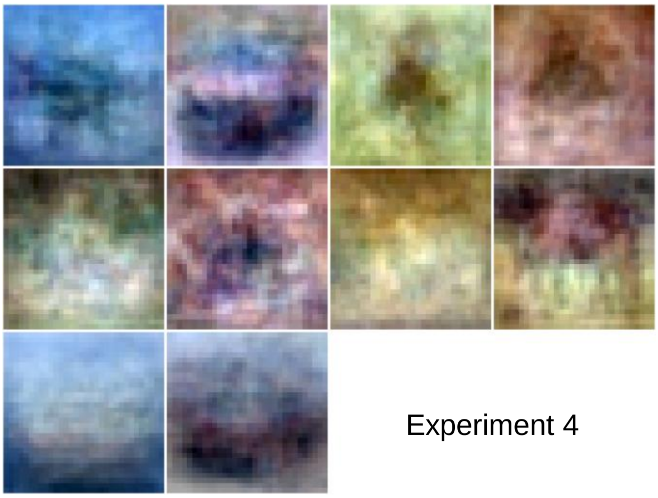
Experiment 1



Experiment 2



Experiment 3



Experiment 4

Something to note with the visualised weights is that as the regularisation term increases, the pictures become less grainy and form distinct shapes.