DD2424 Deep Learning in Data Science

Assignment 1 - Bonus Points

Ramona Häuselmann

April 2, 2021

1 Exercise 2.1

Starting from the results of Experiment 4 of Assignment 1 the following approaches to improve the network performance were tested:

(i) use all available training data

(ii) train for a longer time

The results after Experiment 4 of Assignment 1 were as follows:
training loss: 1.899
validation loss: 1.958
accuracy: 37.38%
and the following parameters were used:
lambda=0, n_epochs=40, n_batch=100, eta=0.1

1.1 Results Improvement (i)

final training loss 1.920
final validation loss 1.935
final accuracy 0.3786

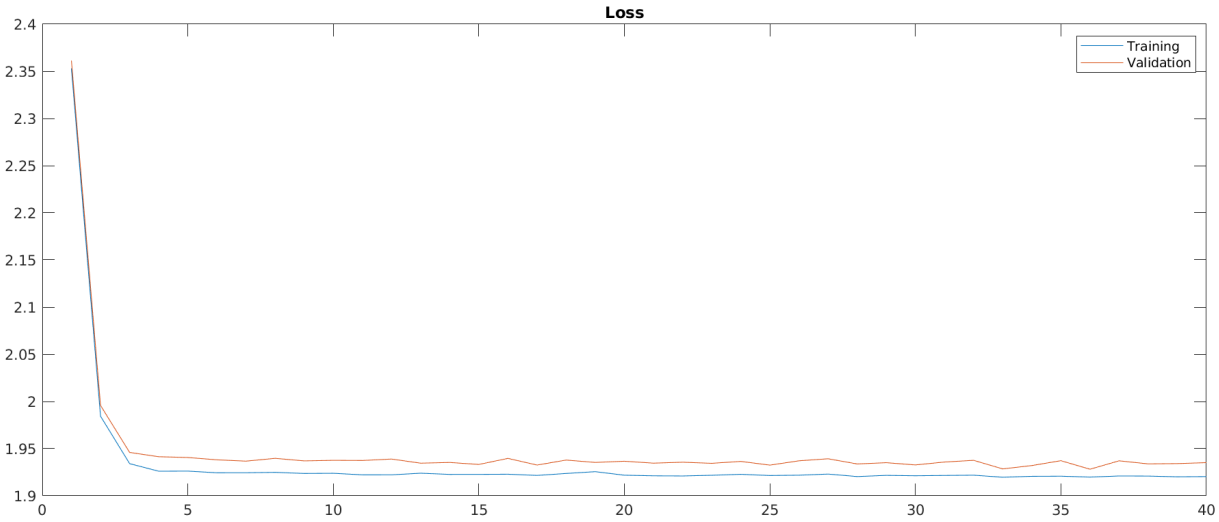

Figure 1: Improvement (i) Loss (lambda=1, n_epochs=40, n_batch=100, eta=0.001)

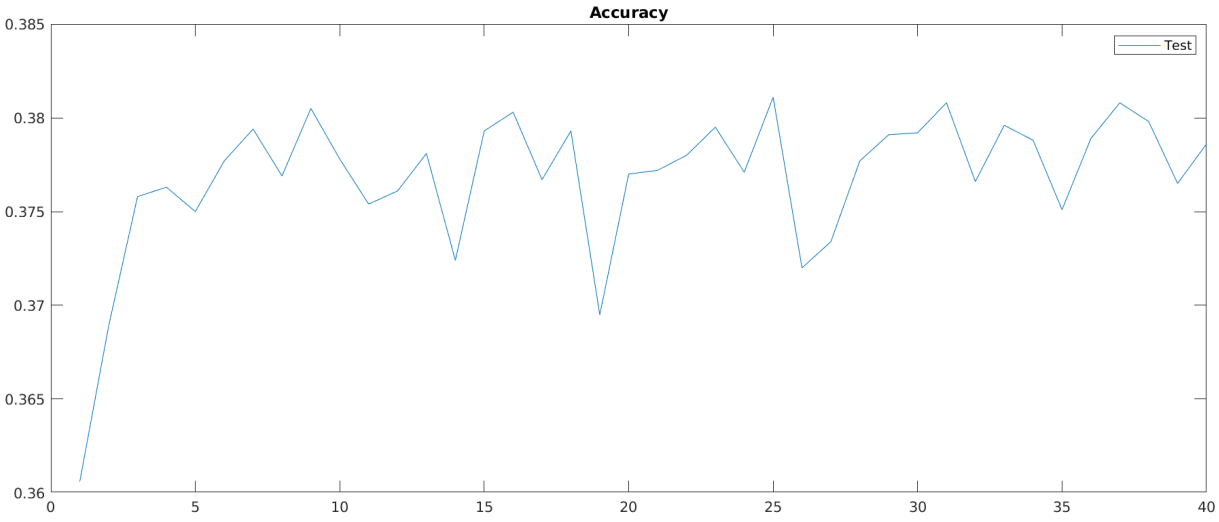

Figure 2: Improvement (i) Accuracy (lambda=1, n_epochs=40, n_batch=100, eta=0.001)

Figure 3: Improvement (i) Weights (lambda=1, n_epochs=40, n_batch=100, eta=0.001)

1.2 Experiment 1 diagrams

As seen in the diagrams with the parameters of this run the network is not really able to learn. The loss and accuracy look very random. This results in a very low accuracy (27.84%). The weight matrices look pretty random as well.

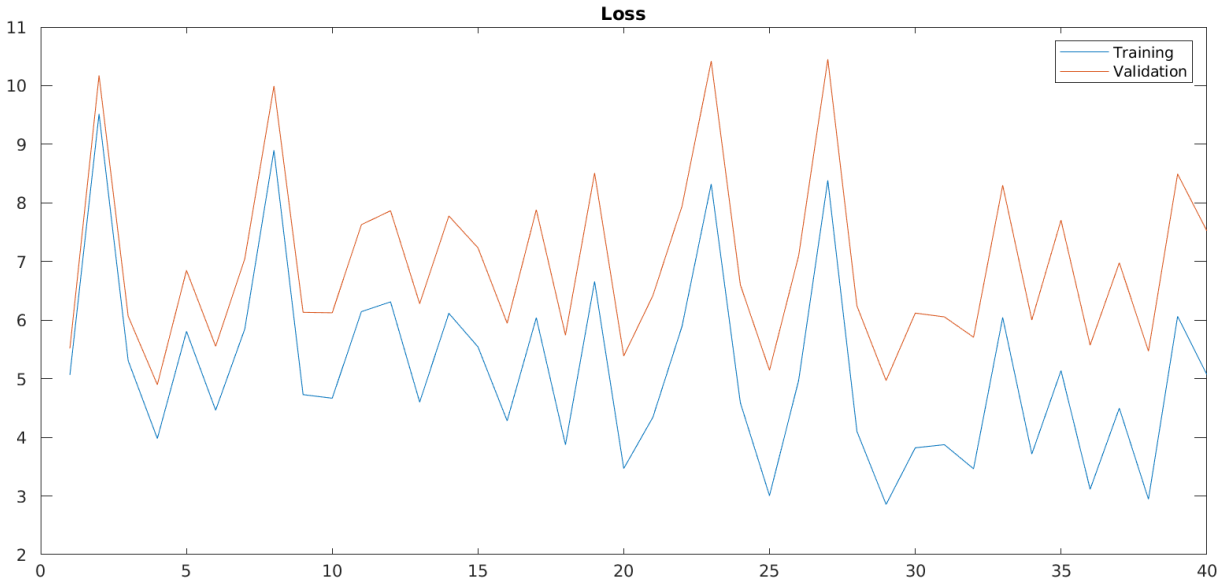

Figure 4: Experiment 1 Loss (lambda=0, n_epochs=40, n_batch=100, eta=0.1)

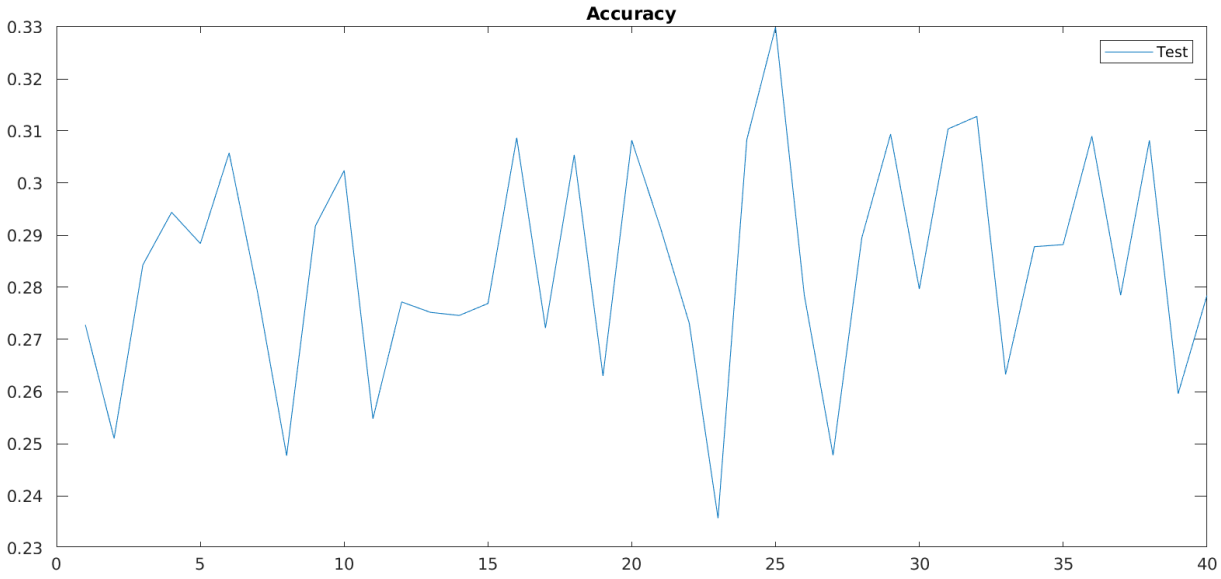

Figure 5: Experiment 1 Accuracy (lambda=0, n_epochs=40, n_batch=100, eta=0.1)

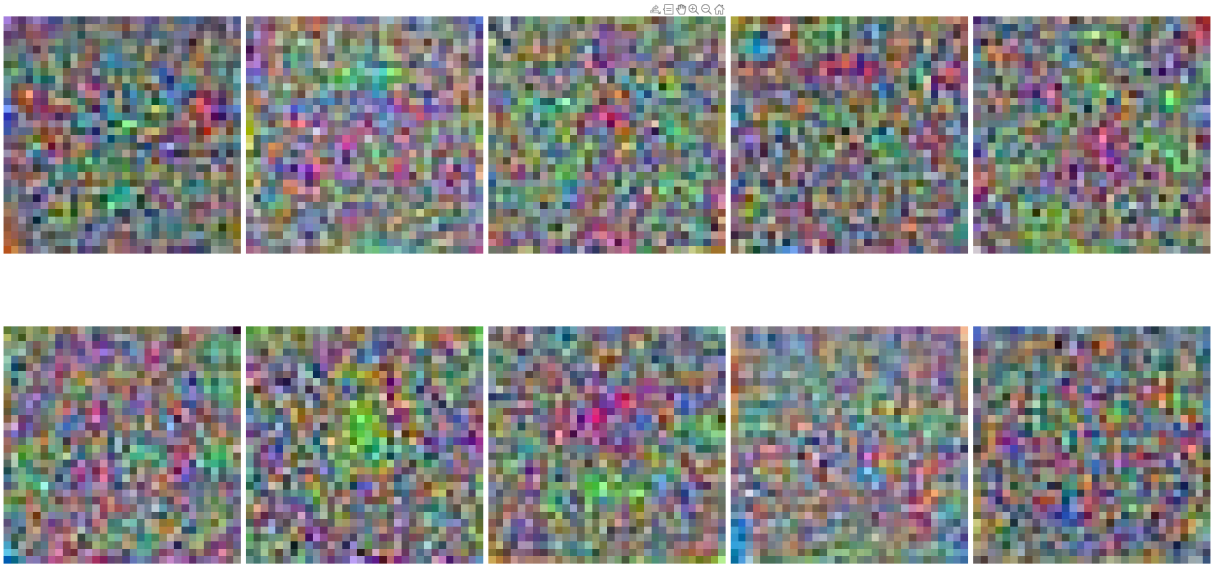

Figure 6: Experiment 1 Weights (lambda=0, n_epochs=40, n_batch=100, eta=0.1)

1.3 Experiment 2 diagrams

Compared to the first experiment in this run we use a much smaller learning rate. In this run it looks like the network learns better and it results in a higher accuracy (39.08%). If we look at the loss graphs we can see that there is a big difference between the training loss and validation loss. This could be an indicator for overfitting. The weight matrices look less random.

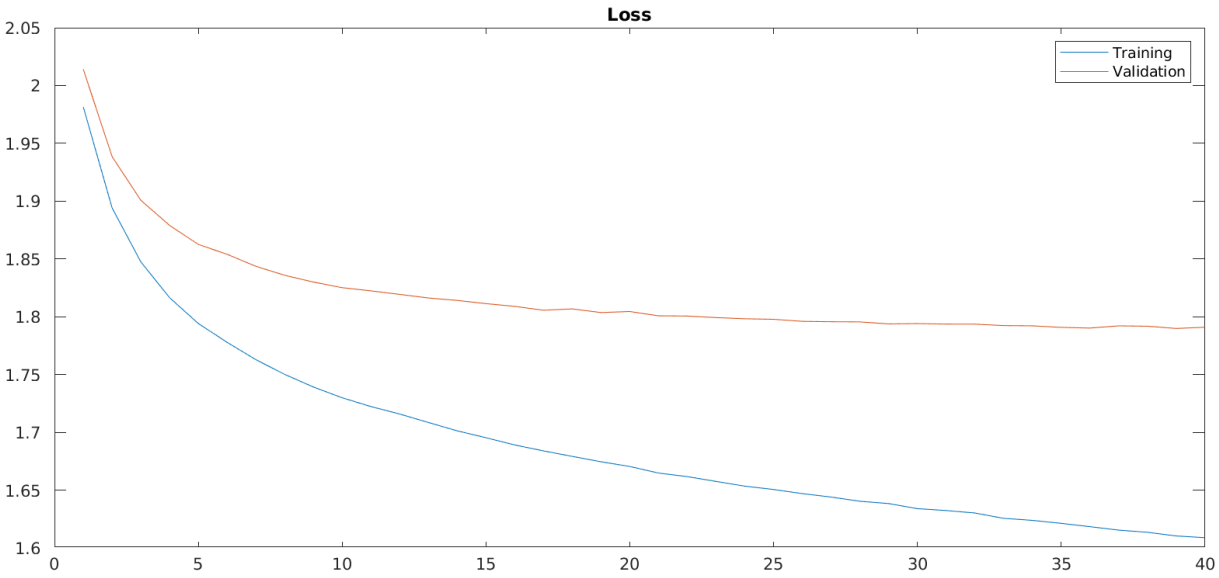

Figure 7: Experiment 2 Loss (lambda=0, n_epochs=40, n_batch=100, eta=0.001)

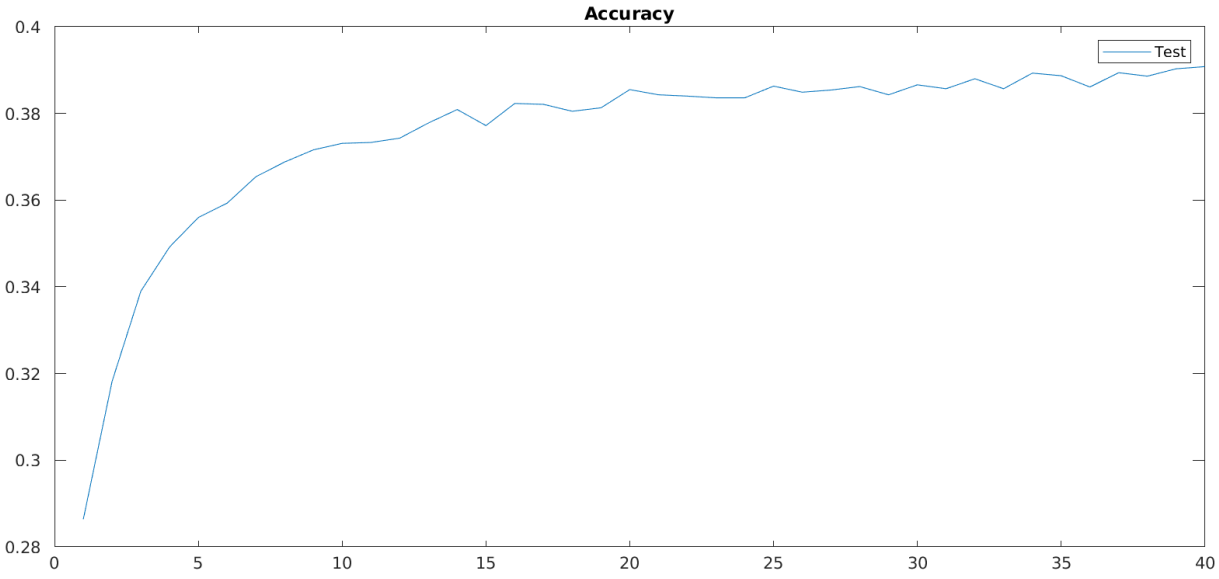

Figure 8: Experiment 2 Accuracy (lambda=0, n_epochs=40, n_batch=100, eta=0.001)

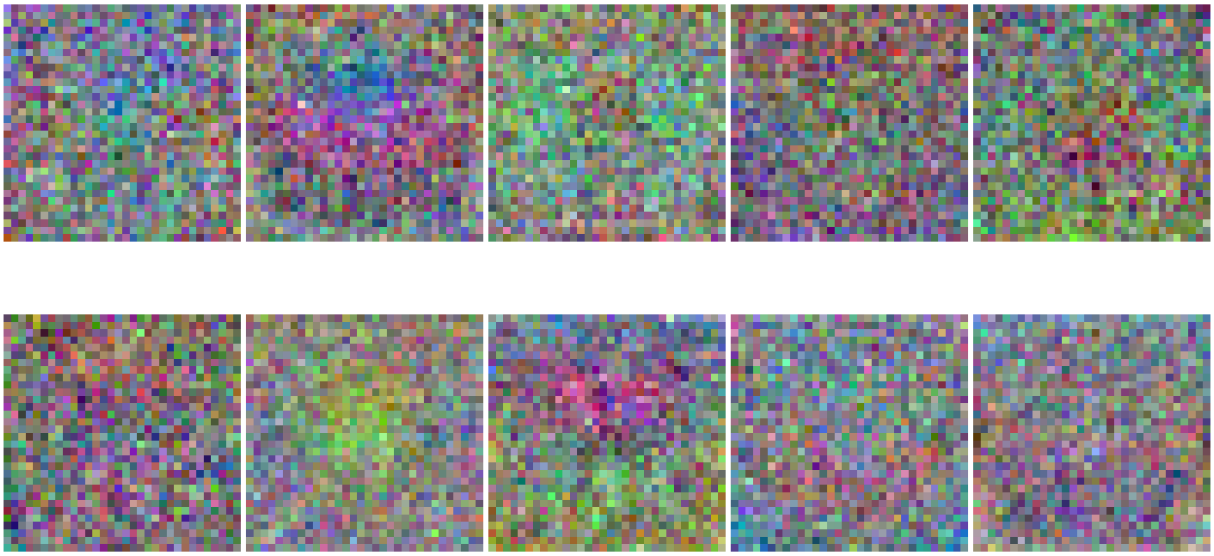

Figure 9: Experiment 2 Weights (lambda=0, n_epochs=40, n_batch=100, eta=0.001)

1.4 Experiment 3 diagrams

In this experiment we use a small regularization term. This results in a smaller training loss compared to the previous run. But we have still a big difference between training and validation which could mean that we overfit. We start to see some more distinct patterns in the weight matrices.

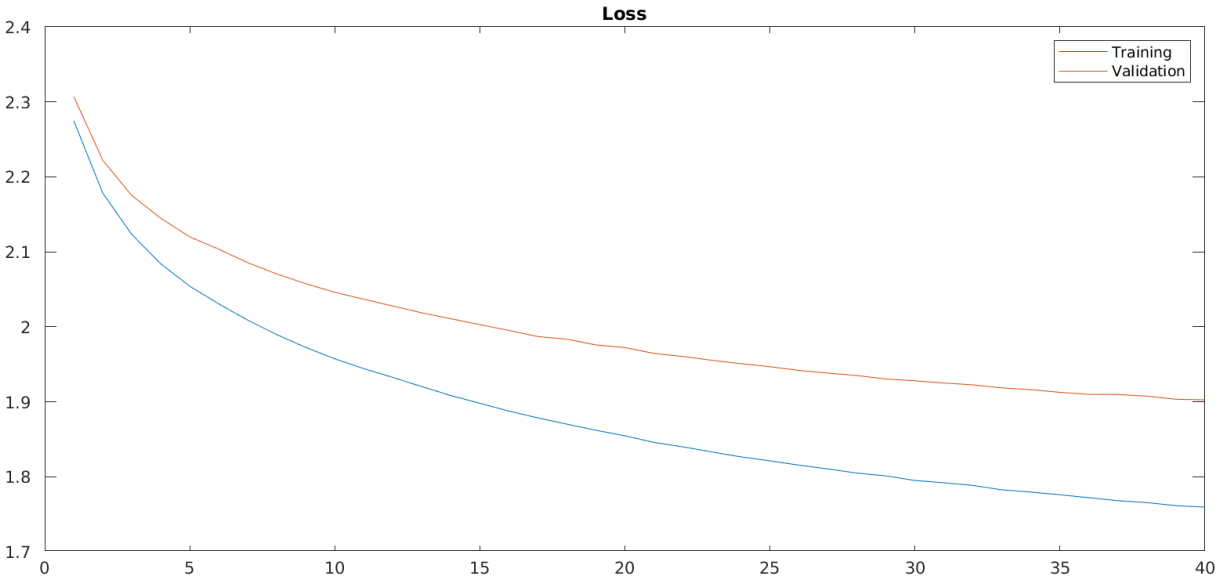

Figure 10: Experiment 3 Loss (lambda=0.1, n_epochs=40, n_batch=100, eta=0.001)

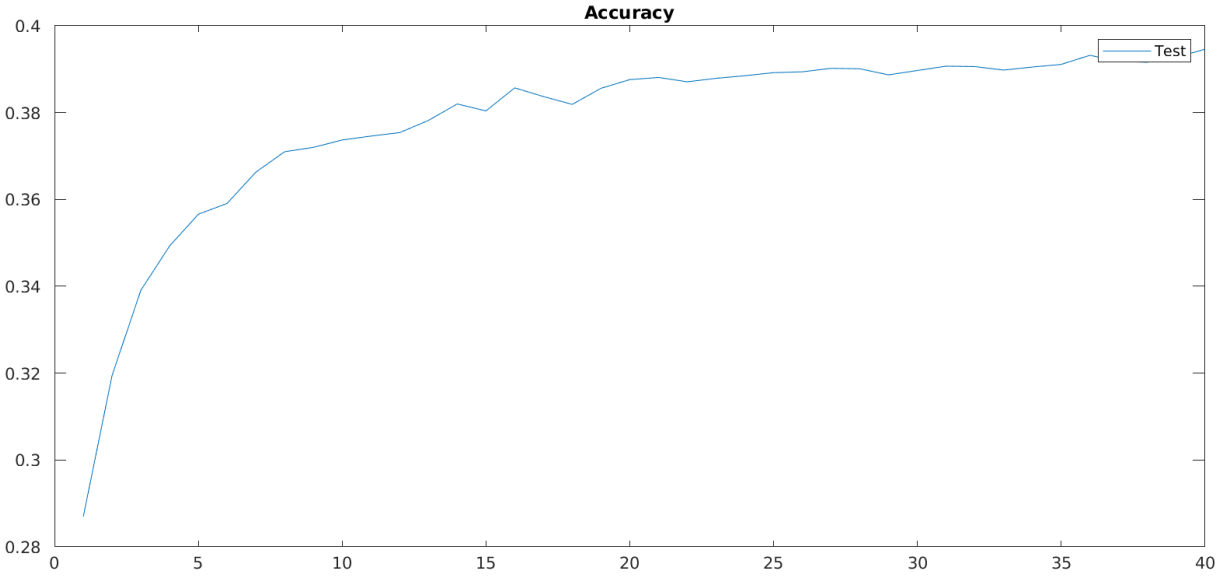

Figure 11: Experiment 3 Accuracy (lambda=0.1, n_epochs=40, n_batch=100, eta=0.001)

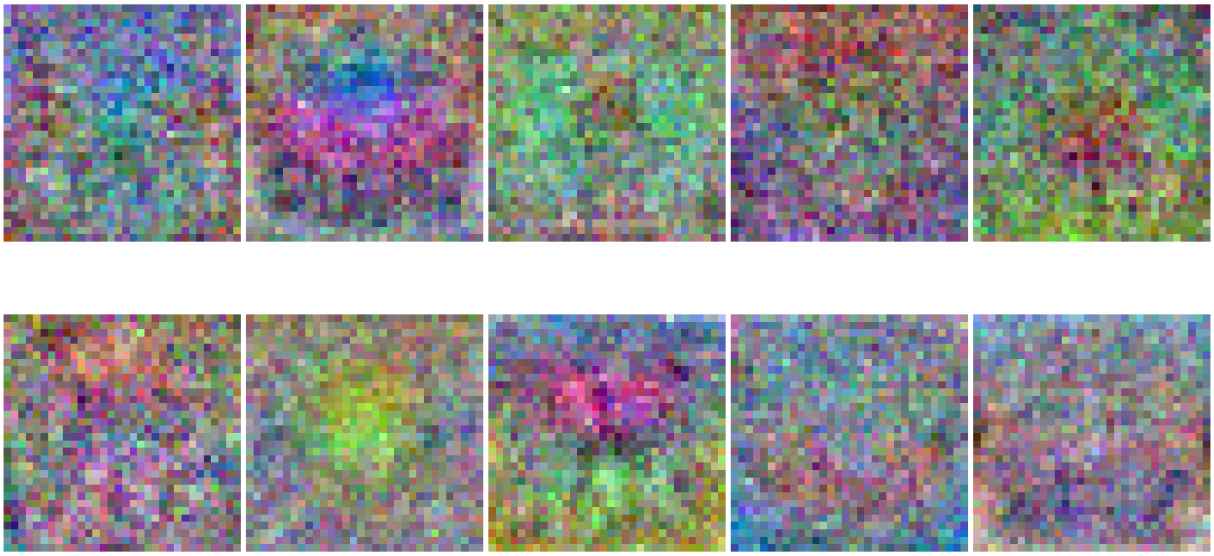

Figure 12: Experiment 3 Weights (lambda=0.1, n_epochs=40, n_batch=100, eta=0.001)

1.5 Experiment 4 diagrams

In this experiment we use an even bigger regularization term. That results in a slightly reduced accuracy but also the training and validation loss are close. That could mean that we are not overfitting anymore. In the weight matrix we see that we start to learn some patterns that looks similar to the images of the corresponding classes. Maybe with more training these patterns become even more visible.

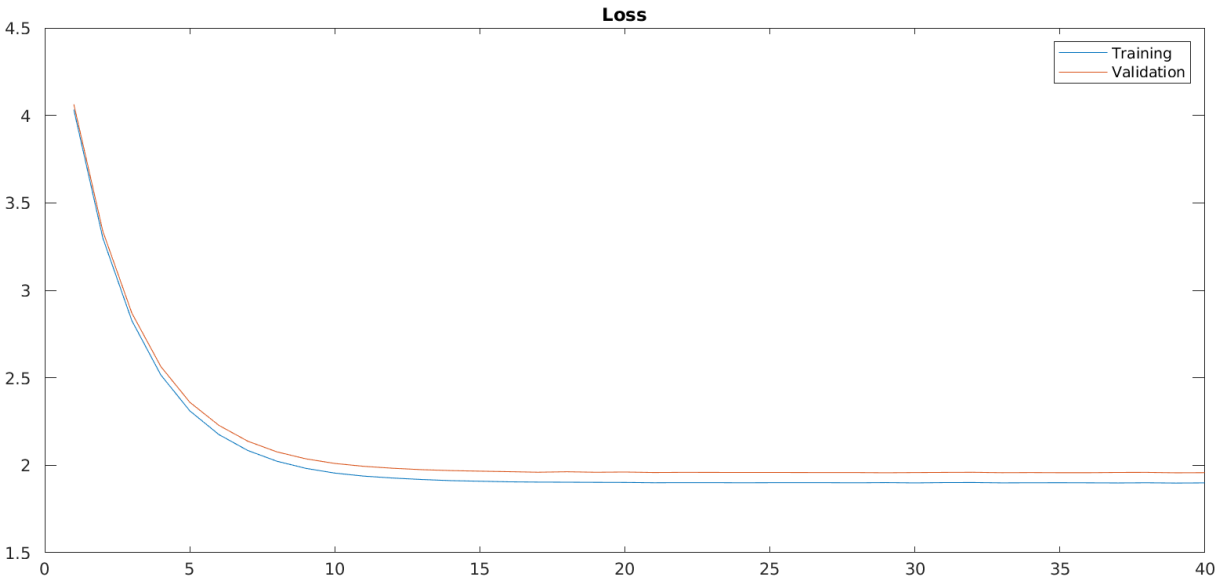

Figure 13: Experiment 4 Loss (lambda=1, n_epochs=40, n_batch=100, eta=0.001)

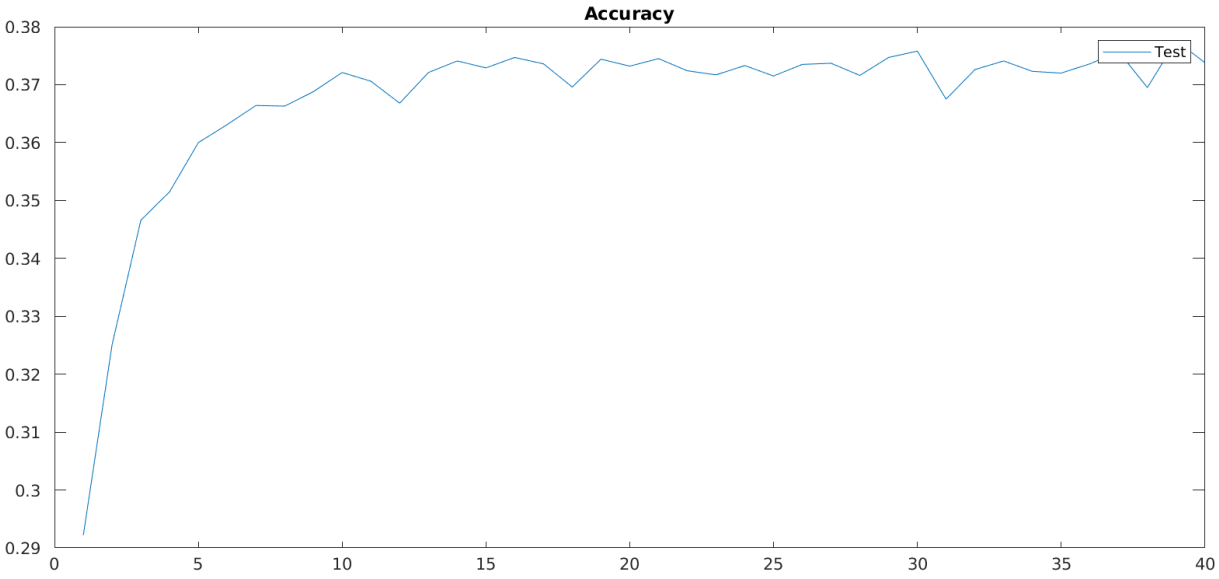

Figure 14: Experiment 4 Accuracy (lambda=1, n_epochs=40, n_batch=100, eta=0.001)

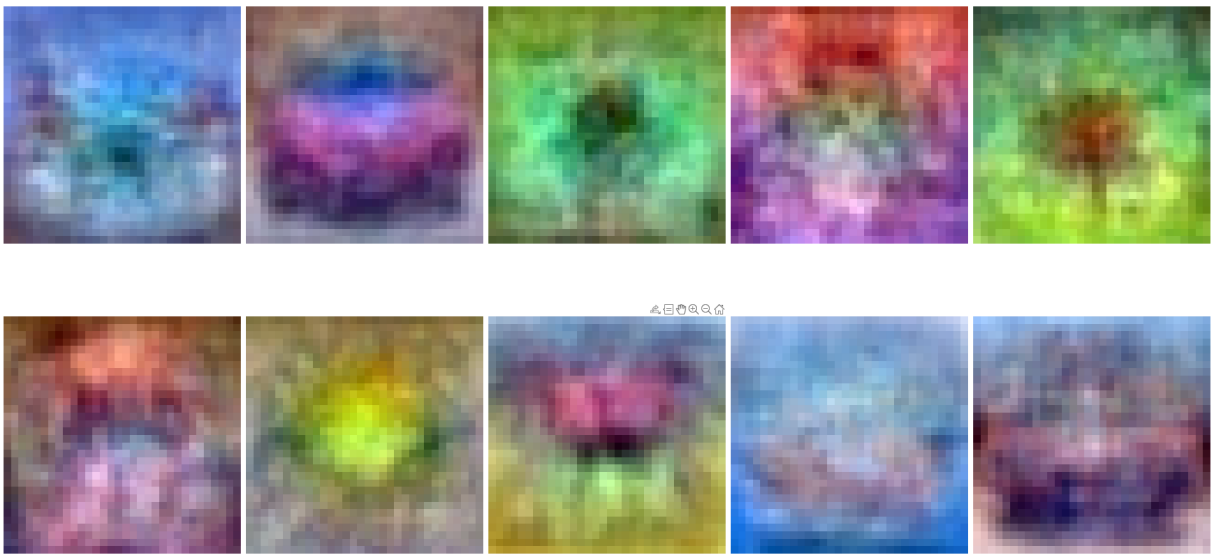

Figure 15: Experiment 4 Weights (lambda=1, n_epochs=40, n_batch=100, eta=0.001)

2 Conclusion

Increasing the amount of regularization reduces the accuracy but also prevents the network from overfitting. The choice of the correct learning rate is very important for the networks ability to learn, as we can see when comparing Experiment 1 and Experiment 2. The learning rate has a huge impact on the accuracy.