Group Task 2.d, CNN on permutated MNIST

2.1 Comparison of results for some different parameters

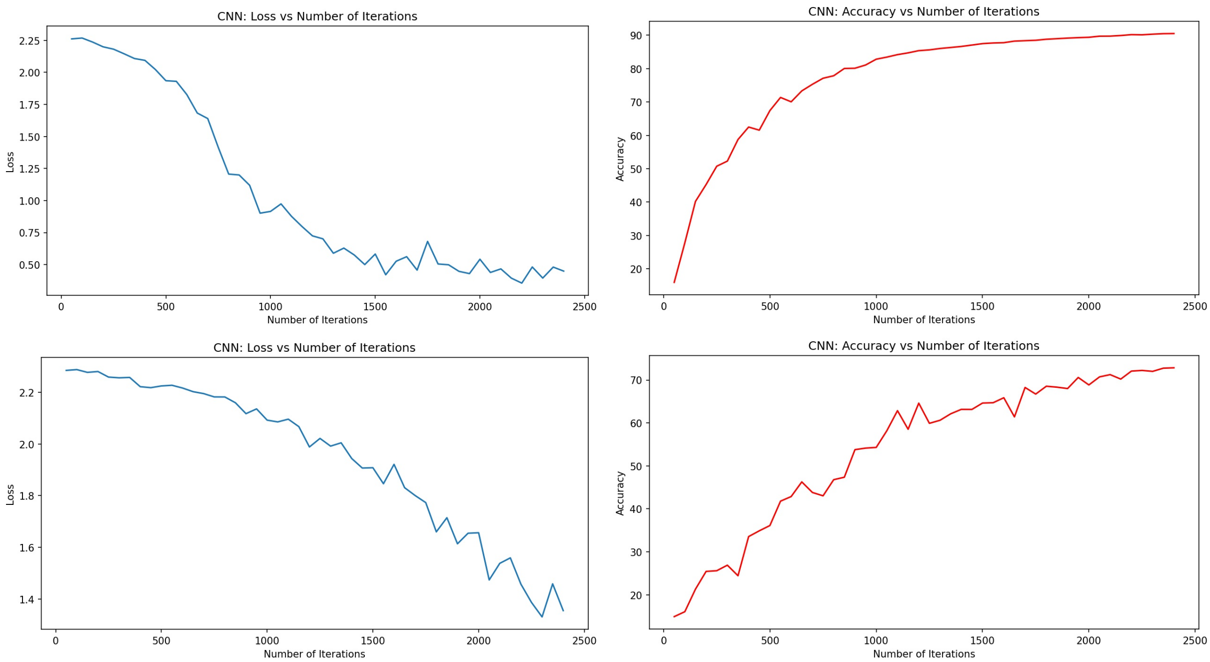

Figure 1. Loss function and Accuracy at learning rate of 0.002 and 2500 iterations. Top row: original MNIST, bottom row: permutated MNIST

All figures show the difference in the loss function and the accuracy between the original MNIST data set and the permutated one.

In figure 1 we see the results with the parameters we found to be working best in exercise 2.c), i.e. a learning rate of 0.002 and 2500 iterations. We can clearly see that the results for the permutated data set are considerably worse than for the original data set, in every respect.

Figure 2 shows the effect of computing more iterations. While the results get better for both data sets with more iterations, the original data set still outperforms the permutated data set.

Figures 3 and 4 show the effects of adapting the learning rate. With a learning rate of 0.001, the difference is even greater, the results for the permutated data set are a lot worse than for the original data set. With a higher learning rate of 0.003, the results are also not better than with the original learning rate of 0.002, though, in this case, also not much worse.

In conclusion, we can say that the CNN performs significantly worse on the permutated data set than on the original one. The optimal parameters seem to be approximately the same.

A possible explanation for this behaviour is the similarity of a CNN to a human brain. It does not just take the pixel values seperatedly for the training and labelling, but takes into account the neighbourhood of the pixels and their position in the greater image. In the permutated images, the information in the neighbourhood of each pixel is just noise, as it is in no way connected to the object it represents.

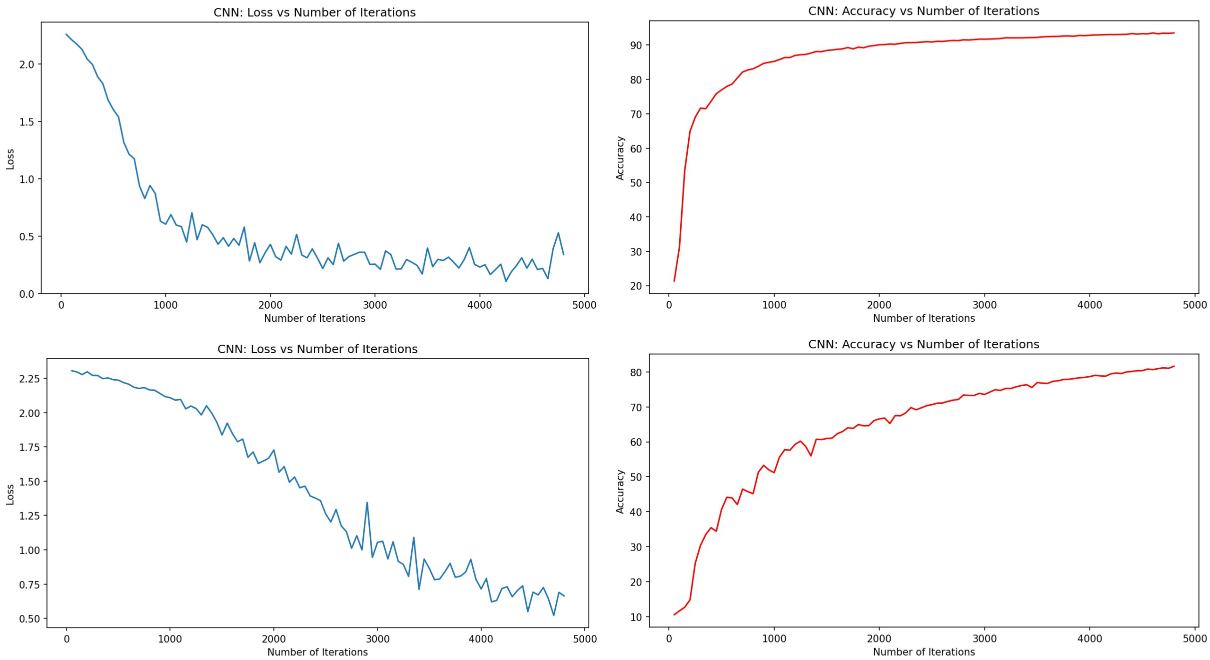

Figure 2. Loss function and Accuracy at learning rate of 0.002 and 5000 iterations. Top row: original MNIST, bottom row: permutated MNIST

However, the CNN still works to some extent, even on the permutated images. This might be explained by the fact that the isolated value of each pixel has still an important influence on the labelling, thus images with similar values in the same pixels still get the same label.

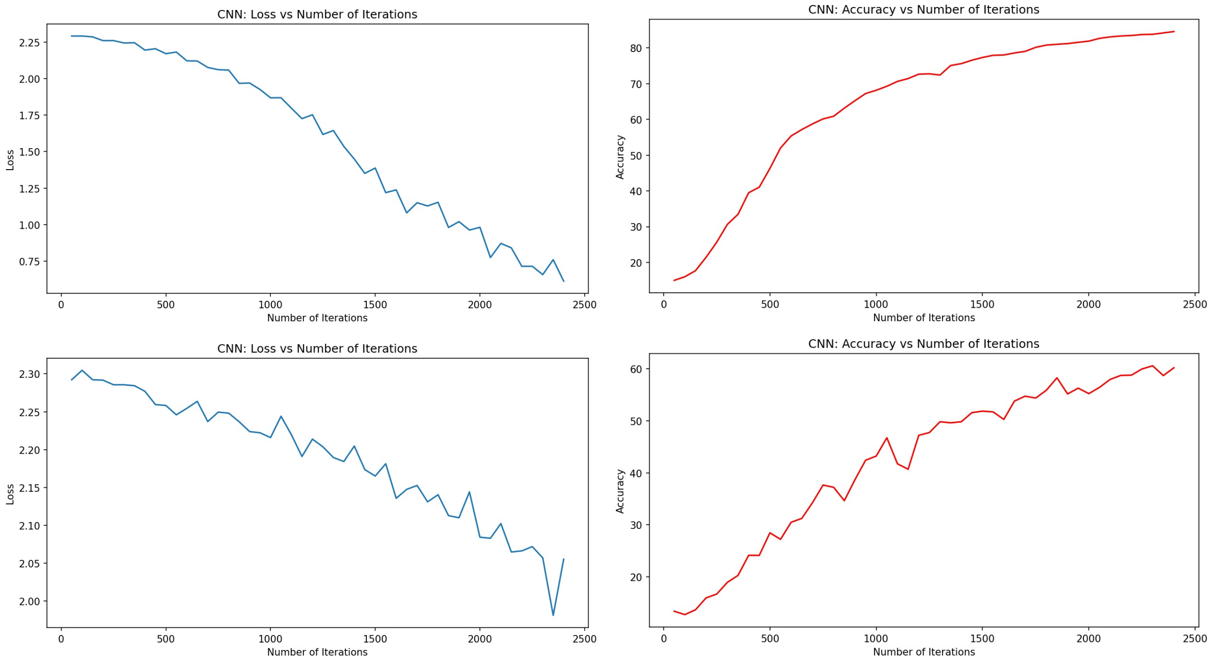

Figure 3. Loss function and Accuracy at learning rate of 0.001 and 2500 iterations. Top row: original MNIST, bottom row: permutated MNIST

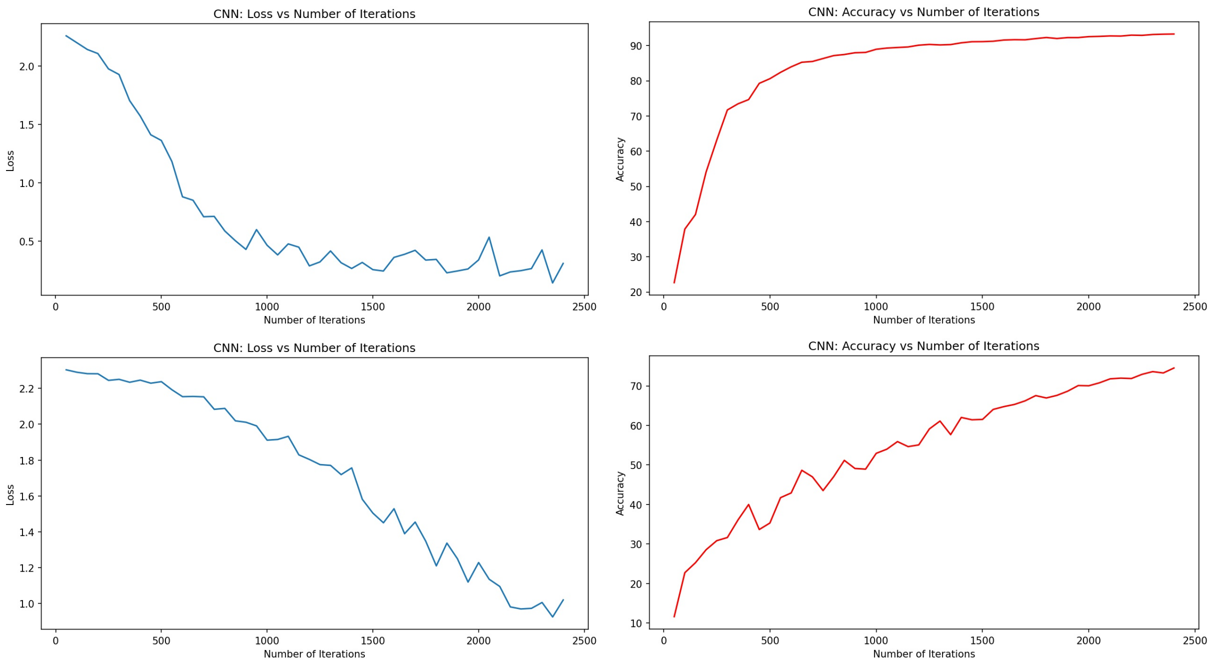

Figure 4. Loss function and Accuracy at learning rate of 0.003 and 2500 iterations. Top row: original MNIST, bottom row: permutated MNIST