**Approach**

**Search correlation of epoch and batch-size which leads to higher accuracy**

In order to find a high accuracy, we try to find the correlation between the epoch and batch-size with the standard configuration of the “cifar10\_cnn.py”. We run the program with different values for epoch and batch-size. For epoch we choose 10, 25, 50, 75 and 100. A high batch size requires much memory, thus in order to fit the data into the graphic memory we take 8, 16, 32, 64 and 128 for the batch size.

We ran the program with the different configurations for epoch and batch-size and stored the accuracy for train and test set in a stylesheet. We can see in Figure 1 that with increasing epochs the accuracy of the CNN with batch size 8 decreases, which tells us, that from epochs greater than 10 the model starts to overfit on the trainings data.

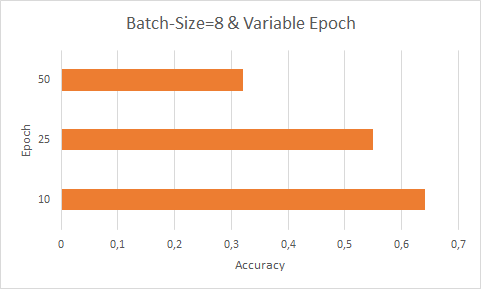


Figure 1 Cifar10\_cnn.py trained with batch size 8 and variable epochs.

For batch size 16 we observe, that the accuracy starts to increase from 10 epochs to 20 epochs, but overfits at 50 epochs [Figure 2]. Figure 3 depicts that for batch size 32 the highest accuracy can be achieved with 75 epochs. When we analyze Figure 1 and Figure 2 we see that for small batch sizes the highest accuracy is reached with small epochs, thus for batch size 8 and 16 we will not be able to increase the accuracy further by increasing the epoch. However, for batch size 32 we got a good accuracy, where the model achieved the best result with 75 epochs and overfit at 100 epochs.

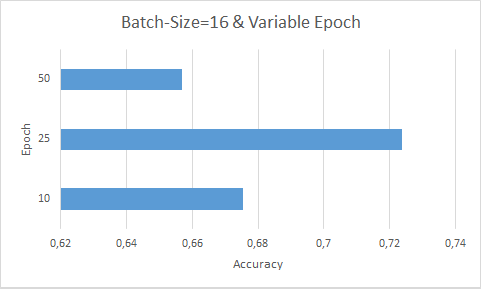


Figure 2 Cifar10\_cnn.py trained with batch size 16 and variable epochs.

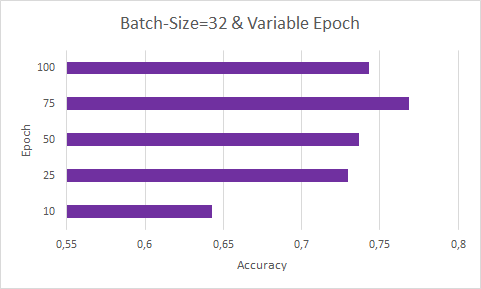


Figure 3 Cifar10\_cnn.py trained with batch size 32 and variable epochs.

As our goal is to find the highest possible accuracy, we discard the configuration with small accuracy. Figure 4 shows all the different configuration and its accuracy, which helps us to compare them and decide which configuration to take and to discard. The best accuracy could be achieved for batch size 64 with epoch 100, where the accuracy is 0.7902, but also batch size 32 and 128 reached a high accuracy. As the accuracy for batch size 8 and 16 is lower in contrast to 32, 64 and 128 we discard them.

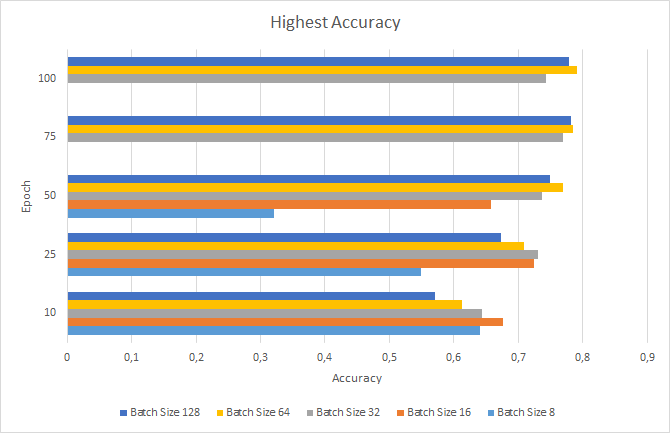


Figure 4 Accuracy of Cifar10\_cnn.py with different configurations for epoch and batch-size.

**Find good value for number of filters**

We continue our experiment with the batch sizes 32, 64 and 128, by trying to find out the optimal number of filters for the convolution layers. First, we will run the program with only 10 epochs and if we detect an increase in the accuracy, then we apply it to epoch 75 or 100.

By increasing the number of the filters of the first and second convolution layer to 64 and 128 we could observe an increase of the accuracy for batch sizes 32, 64 and 128 as shown in Figure 5. This means as the number of filters increases our model can make better predictions, because probably more descriptive features in the images are found. [Figure 4] With 75 iterations we achieved a high accuracy for the batch size 32, 64 and 128, thus we try to increase the accuracy with larger number of filters for 75 epochs. [Figure 6] shows an increase so the accuracy got higher than 0.8 for batch size 64 and 128 for both increased filter sizes, but for batch size 32 we see that it increased with filter size 64 and decreased for filter size 128. Our approach of increasing the number of filters showed an increase of the accuracy.

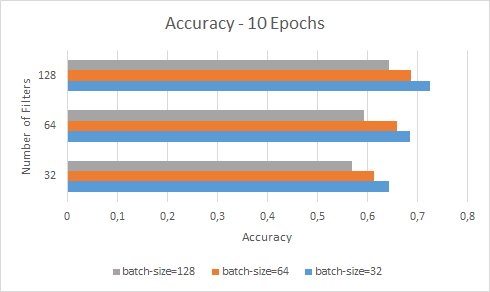


Figure 5 Accuracy of the model with different number of filters. Model trained with 10 iterations.

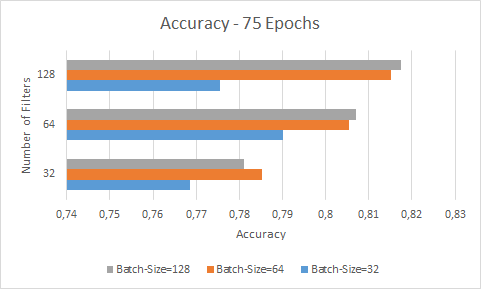


Figure 6 Accuracy of the model with different number of filters. Model trained with 75 iterations.