HW4 – Theoretical Questions

Notes:

- The code is written on the *HW4.py* file. Please do not check the *HW4.ipynb* notebook since it is not updated.
- We used the tutorials as references for some of the code.
- You may look at the *HW4_output1-5.txt* files for outputs of 5 runs of the code.

Part 1

Task2

We chose to change the activation function to *tanh*. This means that the output of each layer, is the *tanh* function of its input. This change affects the model by allowing only blocked set of inputs to each layer (blocked between ± 1), and by centering the numbers towards zero.

Task3

Accuracy results we obtained for two runs of the code were:

	Run1	Run2	Run3	Run4	Run5
25 epoch model	64.00%	66.86%	63.43%	63.43%	64.00%
40 epoch model	66.29%	64.57%	63.43%	63.43%	64.57%

We notice that higher number of epochs does not improve the model performance significantly, and even can worsen it by overfitting the training set. Thus, we conclude that 25 epochs are sufficient to train the model and obtain good performance.

Task4

When using SGD convergence of the model is much faster, than using mini-batch. However, it causes frequent updates with a high variance that cause overfitting. Overall performance of mini-batch is better and more stable at convergence of the loss function than SGD.

Task5

By using batch normalization, we obtained the following accuracies:

Run1 – 66.29%, Run2 – 66.29%, Run3 – 62.29%, Run4 – 62.86%, Run5 – 66.86%

We notice the results are generally worse than the other models learnt. However, we changed more parameters in this model such as batch size and number of epochs, rather than just adding batch normalization. Thus, we cannot have a concrete conclusion.

Part 2

Task1

The model has 8 layers.

The number of filters in each layer is: 64, 128, 128, 256, 256, and the last 3 layers are fully connected and thus have no number of filters.

The number of parameters learned by the model will not be the same as a fully connected NN. It will be much lower since only the filters are learned which have less parameters than all the weights and biases connecting a fully connected NN.

This NN is performing regularization (L_2 regularization).

Task2

We see that when using less filters, the accuracy gets worse, meaning the model is too simple with not enough parameters.

Here are the accuracies obtained:

	Run1	Run2	Run3	Run4	Run5
Original model	26.29%	26.29%	32.00%	33.71%	34.29%
Reduced filters	23.43%	25.14%	32.57%	25.14%	27.43%