SMAI Assignment 2

Naila Fatima
201530154

Question 2

Part 1a)

The CNN architecture that I have constructed consists of the following seven layers in the particular order.

1) Convolutional 2D layer with 32 filters, kernel size = (1,1), strides = (1,1), activation function = 'ReLu' and input shape = (32,32,3)
2) Max pooling 2D layer with pool size = (2,2) and strides = (2,2)
3) Convolutional 2D layer with 64 filters, kernel size = (1,1), activation function = 'ReLu'
4) Max pooling 2D layer with pool size = (2,2)
5) Convolutional 2D layer with 32 filters, kernel size = (1,1), activation function = 'ReLu'
6) Flatten layer
7) Dense layer with number of outputs = 10 and activation function = 'softmax'

The time taken for this to converge was around 25 minutes on a cpu. The accuracy was 61.7%.

In my experiments, I have noticed that a smaller kernel size gives faster results which are slightly more accurate. Also, a pool size of (2,2) was optimum although it took more time than a pool size of (1,1) but was better in terms of accuracy. For larger pool sizes, I found the trade off between accuracy and size to be poor. The activation function is chosen to be ReLu since it has given the best outputs.

i) Adding Batch Normalization

On adding one batch normalization layer between the first maxpool layer and the second conv2D layer, the accuracy decreases to become 56%. The convergence time is around 45 minutes.

ii) Adding dropout

On adding one dropout layer before the flatten layer where the rate is 0.25, the accuracy further decreases to 52.5%. The convergence time is around 25 minutes.

Part 1b)

i) Using the ReLu activation

Since we have used an activation function of 'ReLu' in the previous subparts, we can see that the accuracy is around 52.5 %. The convergence time is around 25 minutes.

ii) Using the tanh activation

On switching to a 'tanh' activation function, the accuracy increases to 58.5 %. The convergence time is around 20 minutes.

Iii) Using the sigmoid activation

On switching to a 'sigmoid' activation function, the accuracy was 51.1%. The convergence time was around 30 minutes.

iV) Using the linear activation

On switching to the linear activation function, the accuracy was 39.96 %. The convergence time was around 30 minutes.

Part 2)

I have used the above CNN architecture shown. I am using 3 Conv2D nets, 2 maxpool layers and 1 dense layer. I have omitted the batch normalization layer as it was reducing my

accuracy. Also, I have used the ReLu activation function as I have noticed that it is giving me the highest accuracy among the other functions. The accuracy I am getting is around 61%.