Data Analytics
HW 4

Ahmed Shahabaz

Original Model Stat

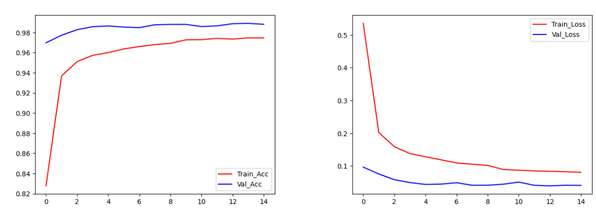
Test Accuracy: 0.98960

Question 1(a)

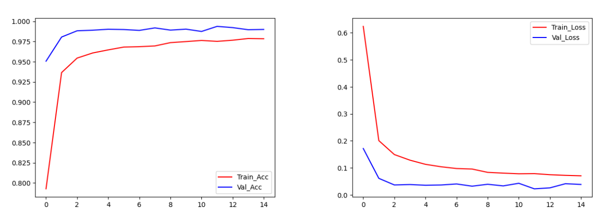
Test Accuracy with (64,128) CNN: 0.98660.

We can see that the test accuracy for the original model was a bit higher than that of the model described in question 1(a). By further comparing between the train accuracy of the last 5 epochs (reverse order 15-11) for the original model [0.9738, 0.9758, 0.9741, 0.9746, 0.9721] and the model in question 1(a) [0.9784, 0.9793, 0.9793, 0.9755, 0.9770] we can clearly see that the model in question 1(a) has higher train accuracy.

If we make the same observation for the validation accuracy we can see the same pattern that is validation accuracy for the model in Question 1(a) [0.9913, 0.9930, 0.9908, 0.9902, 0.9896] is higher than that of the original [0.9881, 0.9890, 0.9887, 0.9865, 0.9858] model.



(a) Accuracy (left) and Loss (right) plot for the Original model



(b) Accuracy (left) and Loss (right) plot for the model in question 1(a)

Fig. 1: Plots for Question 1(a)

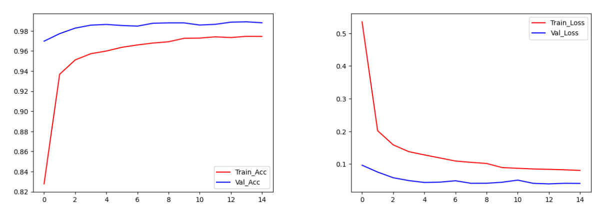
So after comparing the train, test and validation accuracy and the train and validation shown in figure [1], we can say that the model with CNN layers (64,128) as described in question 1(a) did overfit to the training data more compared to the original model.

Question 1(b)

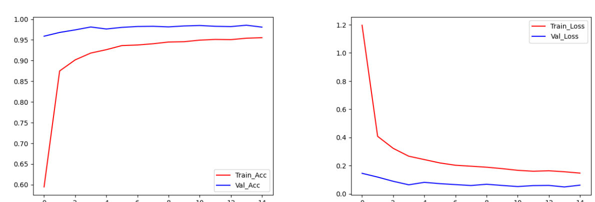
Test Accuracy with (32,64,32) CNN: 0.98571.

We can see that the test accuracy for the original model was a bit higher than that of the model described in question 1(b). By further comparing between the train accuracy of the last 5 epochs (reverse order 15-11) for the original model [0.9738, 0.9758, 0.9741, 0.9746, 0.9721] and the model in question 1(b) [0.9562, 0.9544, 0.9486, 0.9495, 0.9484] we can clearly see that the original model has higher train accuracy.

If we make the same observation for the validation accuracy we can see the same pattern that is validation accuracy for the model in Question 1(a) [0.98571, 0.9855, 0.9855, 0.9855, 0.9855] is lower than that of the original [0.9881, 0.9890, 0.9887, 0.9865, 0.9858] model.



(a) Accuracy (left) and Loss (right) plot for the Original model



(b) Accuracy (left) and Loss (right) plot for the model in question 1(b)

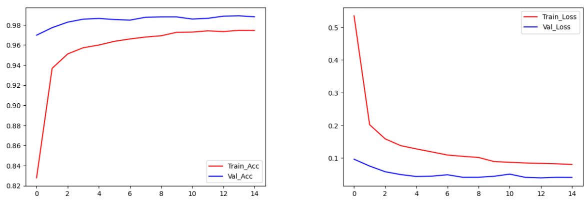
Fig. 2: Plots for Question 1(b)

So after comparing the train, test and validation accuracy and the train and validation shown in figure [2], we can say that the model with CNN layers (32,64,32) as described in question 1(b) is too complex for the data given to us.

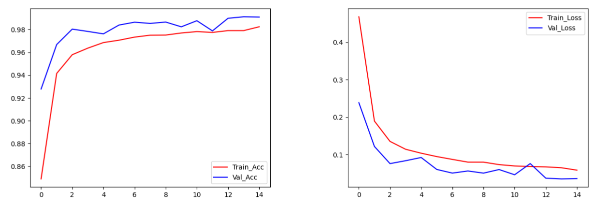
Question 1(c)

Test Accuracy with out dropout: 0.98903.

After looking at the figures shown in figure [3] we can see that the validation accuracy and loss for the original model was always higher and lower respectively than the train accuracy and loss. Whereas for the figure [3(b)] which shows the train and validation accuracy of the model without any drop out layer, we see that the validation accuracy sometimes goes lower than the train accuracy and as a result the validation loss goes higher than the train loss. So the training for the model in Question 1(c) is a bit more stochastic than the original model. And like wise the training for the original model is more smooth or less stochastic than the model without any drop out layer.



(a) Accuracy (left) and Loss (right) plot for the Original model



(b) Accuracy (left) and Loss (right) plot for the model in question 1(c)

Fig. 3: Plots for Question 1(c)

So from our above discussion and then observing the Test accuracy of the original model and the model in Question 1(c) we can say that, the model without any dropout layer as described in Question 1(c) overfits to the train data more than the original model.

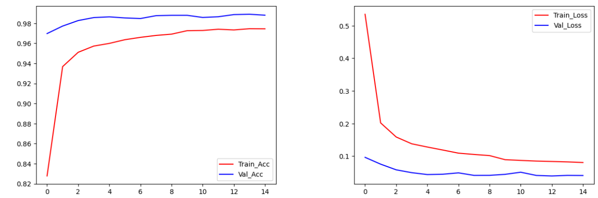
So as for the effect of the dropout layers, it makes the model more robust or not overfit to the training data.

QUESTION 1(D)

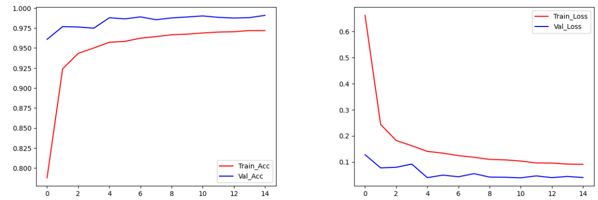
Test Accuracy with lr=0.2: 0.98925.

Figure [4] shows the Accuracy & Loss graphs between the original model and the model trained with learning_rate = 0.2 as described in Question 1(d). From the figure [4] we can see that the training accuracy for the model following the training method of Question 1(d) is a bit lower than the training accuracy of the original model. But in case of the validation accuracy we see that it is the other way around. That means, the validation accuracy for the model trained with learning_rate = 0.2 is higher than the original model.

Again, observing the Test Accuracy we can decide that the original models performs better than the same model trained with lr=0.2. So after considering the above discussion we can say that, the model suffered to converge with higher learning rate.



(a) Accuracy (left) and Loss (right) plot for the Original model



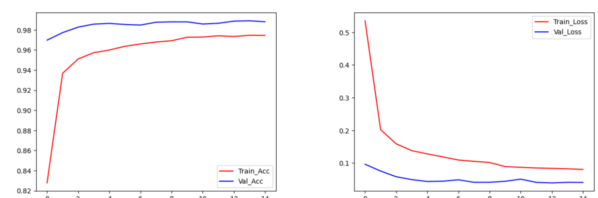
(b) Accuracy (left) and Loss (right) plot for the model in question 1(d)

Fig. 4: Plots for Question 1(d)

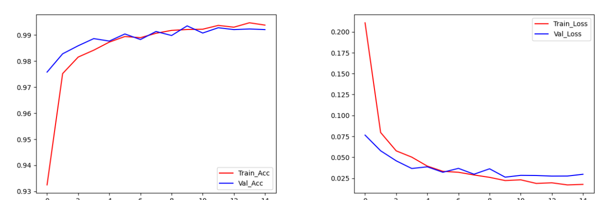
QUESTION 1(E)

Test Accuracy without data augmentation: 0.99092.

The test accuracy for the original model without any data augmentation is higher than that of the model with data augmentation.



(a) Accuracy (left) and Loss (right) plot for the Original model



(b) Accuracy (left) and Loss (right) plot for the model in question 1(e)

Fig. 5: Plots for Question 1(e)

As it can be seen from the figure [5] the training accuracy plot for the model without any augmentations goes over the validation accuracy and so the training loss goes lower than the validation loss. But as for the original model with augmentations the train accuracy and loss always kept lower and higher than the validation loss. Even though it seem that the model with out any augmentations is overfitting to the train data. But after observing the test performance we can understand that including data augmentations actually made the data too complex compared to the complexity of the model for the training to be finished in 15 epochs. So either the original model should be trained without data augmentation or if the augmentation is used then the model should be trained for more than 15 epochs.