Step 1:

Github Link: <https://github.com/Tuhinmajumder/S5/blob/master/S5_MNIST_version1_TuhinMajumder_.ipynb>

Target:

1. Creating the model architecture in a way not to exceed 10k parameters to reach 99.4 test accuracy within 15 epochs.
2. The training and test loss should keep on reducing with number of epochs, so we can infer the model is getting trained properly and with test accuracy the generalizability power will also be checked.

Reaching accuracy on test data close to 99.4(if possible even more), but the accuracy value should be consistent and also there should not be large gap between train and test accuracy.

Results:

* 1. Parameters: 9594
  2. Best Training Accuracy: 99.48
  3. Best Test Accuracy: 99.15
  4. Epoch:14

1. Analysis:
   1. Test accuracy 99.4 was not achieved , hence in the next step the first task is to reach this target and staying on the value consistently.
   2. Model is slightly over-fitting as there is some gap between train and test accuracy values, for the epoch 13 also test accuracy is 99.07 but train accuracy is 99.49 , so for the next steps this issue also can be taken care of.

Step 2:

Github Link:

<https://github.com/Tuhinmajumder/S5/blob/master/S5_MNIST_version2_TuhinMajumder.ipynb>

Target:

1. Creating the model architecture in a way not to exceed 10k parameters to reach 99.4 test accuracy within 15 epochs.
2. The training and test loss should keep on reducing with number of epochs, so we can infer the model is getting trained properly and with test accuracy the generalizability power will also be checked.
3. The model should perform better than the previous trained model.

Results:

* 1. Parameters: 9594
  2. Best Training Accuracy: 99.51
  3. Best Test Accuracy: 99.30
  4. Epoch:14

1. Analysis:
   1. Test accuracy 99.4 was not achieved , hence in the next step the first task is to reach this target .
   2. Model is trained better as it reached 99.30 test accuracy.From the previous model nothing is changed in terms of model architecture , just step schedular introduced, which boosted the accuracy value , though the model is still performing better in training data than test data as there is subtle gap is present.

Step 3:

Github Link:

<https://github.com/Tuhinmajumder/S5/blob/master/S5_MNIST_version3_TuhinMajumder.ipynb>

Target:

1. Creating the model architecture in a way not to exceed 10k parameters to reach 99.4 test accuracy within 15 epochs.
2. The training and test loss should keep on reducing with number of epochs, so we can infer the model is getting trained properly and with test accuracy the generalizability power will also be checked.
3. The model should perform better than the previous trained model.

Results:

* 1. Parameters: 9594
  2. Best Training Accuracy: 98.93
  3. Best Test Accuracy: 98.80
  4. Epoch:10

1. Analysis:
   1. Test accuracy 99.4 was not achieved , hence in the next step the task is to reach this target .
   2. Model accuracy is actually gone degraded, after introducing the dropout (without changing the model architecture)for all the layers , though for most of the epochs and overall the gap between train and test accuracy is minimal. So again from this step the focus should be to increase accuracy upto at least 99.40.

Step 4:

Github Link:

<https://github.com/Tuhinmajumder/S5/blob/master/S5_MNIST_version4_TuhinMajumder.ipynb>

Target:

1. Creating the model architecture in a way not to exceed 10k parameters to reach 99.4 test accuracy within 15 epochs.
2. The training and test loss should keep on reducing with number of epochs, so we can infer the model is getting trained properly and with test accuracy the generalizability power will also be checked.
3. The model should perform better than the previous trained model.

Results:

* 1. Parameters: 9996
  2. Best Training Accuracy: 99.34
  3. Best Test Accuracy: 98.78
  4. Epoch:10

1. Analysis:
   1. Test accuracy 99.4 was not achieved , though it reached 99.34 as test accuracy with 98.78 as training accuracy. Which is actually a good model having fair chance to train model even better or for higher epochs there is a propensity to reach higher training accuracy, that will definitely giving better result on test data also.
   2. Model accuracy is improved with introducing random rotational transforms for training data and increased model’s capacity, but target still not met.

Step 5:

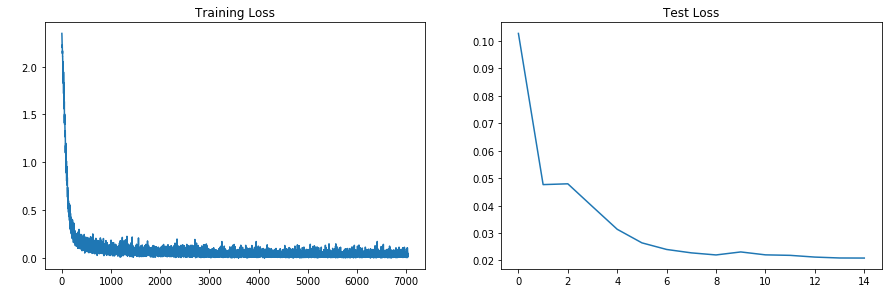
Github Link:

<https://github.com/Tuhinmajumder/S5/blob/master/S5_MNIST_Final_version_TuhinMajumder.ipynb>

Results:

* 1. Parameters: 9996
  2. Best Training Accuracy: 99.43
  3. Best Test Accuracy: 98.85
  4. Epoch:13

1. Analysis:

Test accuracy 99.4 is achieved , not only it reached 99.43 as test accuracy at epoch 13, but also in epoch 14 the test accuracy is 99.41. From the previous model ,only change is made that is step size of the step schedular increased to 8 , as from the previous visuals of model performance for the test loss, it was found that there is almost horizontal line after steep decrease of test loss , though training loss kept on decreasing in a regular steep pattern, so if the period of learning rate decay is increased, then the model will be trained better ,as gradual decay of LR is expected when training happens , it converges too fast, to a crappy loss/accuracy, if it’s decayed rapidly.

* 1. Model accuracy is improved with introducing random rotational transforms for training data and increased model’s capacity, but target still not met.