Project Report

Date:

**Notebook link:**

<https://colab.research.google.com/drive/1Ru8CxE-cMQ7x0bu3cARUJZ3dFrR2n9_P?usp=sharing>

**Assignment details:**

<https://drive.google.com/file/d/1KoWyQx-mfjGP4CO6lIwBwNoVtZrkO1H4/view?usp=sharing>

**Dataset:**

[**https://drive.google.com/file/d/1\_kBtWDRjWEm0vV0V\_lzru5ULE20CQKCk/view?usp=sharing**](https://drive.google.com/file/d/1_kBtWDRjWEm0vV0V_lzru5ULE20CQKCk/view?usp=sharing)

**Important Concepts learned:**

* I learned how dataset and dataloader classes in pytorch work & how batches work.
* Learned how the transform function help to transform the images and how the normalize parameter work inside it.
* Learned how the dropout work and help to overcome the overfitting problem.
* Learned how nn.Modulelist() function works.
* Learned how loss function works and how it handle the output generated by model and how it compares it with target values to calculate losses.
* Also get insights into the output dimensions of data loaders.
* Learned confusion matrix and how it can help to visualize and check the accuracy of every class in a multiclass classifier and how we can check if model is confused between two or classes.

**Critical decisions made:**

* I was struggling to find a way to store fully connected layers in a any data structure so, I used nn.Modulelist() .
* I was also having issues about how to find accuracy explored different ways like accuracy\_score() function from sklearn and this leads me to learn about the dimensions of dataloader and the dimensions of output provided by the model.
* I faced a little problem while trying to build a main function with the optimizer argument it requires model parameters at the time of initialization so firstly I just initialize optimizer inside the main function besides taking it as an argument then I initialize a class of a optimizer function beside initializing the optimizer and then use pass this class to the main function along with learning rate and model parameters and initialize the optimizer inside main function as per the requirements.
* I faced an issue in forward propagation section having error related to size compatibility I found out that some of the images was having 3 channels and my model was expecting images with one channel or we can say simple greyscale images so I just change my data loader method to handle it.
* I faced another issue it was inside my test module I was just storing one batch data and predicted result and accurate results of just last batch and passing it to my visualize results to visualize the loss and confusion matrix etc. So, I just change my data loader function again to not to create any batches for my test data and also make a little changes inside my test and visualize functions as well and finally I was good to go.

**Conclusion:**

I found that normalization was not making much difference although the factor that effects the result most was the optimizer function I mostly used ‘Adam’ optimizer because it was even performing well from the start of training and other one was ‘SGD’ optimizer it starts by giving really poor accuracy and it was progressing really well but it was very slow even though I used 40 epochs and on the other hand I just 20 epochs with ‘Adam’ and gives me 98% accuracy but ‘SGD’ just reached something around 95% with same other parameters as I used to achieve best result with ‘Adam’ I may have reached same accuracy as ‘Adam’ but It already was taking double amount of epochs so I don’t further check it out and finally I used basically used two schemes in learning rate (0.001, 0.01) and batch size (64, 32) and no of hidden layers ([400,200,100,50], [100,50]) and drop out (0.2, 0.3) and got best results on the combination of learning rate ‘0.001’ , batch size ‘32’, hidden layers ‘[400,200,100,50]’, drop out ‘0.2’ and I used only cross entropy loss as a loss function as it’s the best fit for the classification models.

**Phase 1 (with Normalization):**

* **Parameters**:

loss\_fun = nn.CrossEntropyLoss()

hiddenlayers = 2

listof\_HL = [100,50]

drop = 0.2

batch = 32

optimizer = torch.optim.Adam

learning = 0.001

**Epoch**s = 20

* **Training credentials:**

**Epoch** 1/20 Train Loss: 0.361869 Train Acc: 0.893450 Valid Loss: 0.171219 Valid Acc: 0.950655

**Epoch** 2/20 Train Loss: 0.178859 Train Acc: 0.946933 Valid Loss: 0.103104 Valid Acc: 0.966767

**Epoch** 3/20 Train Loss: 0.144147 Train Acc: 0.955017 Valid Loss: 0.112444 Valid Acc: 0.965760

**Epoch** 4/20 Train Loss: 0.127661 Train Acc: 0.960967 Valid Loss: 0.094366 Valid Acc: 0.969789

**Epoch** 5/20 Train Loss: 0.112827 Train Acc: 0.965583 Valid Loss: 0.084923 Valid Acc: 0.968781

**Epoch** 6/20 Train Loss: 0.102600 Train Acc: 0.968333 Valid Loss: 0.102077 Valid Acc: 0.972810

**Epoch** 7/20 Train Loss: 0.096103 Train Acc: 0.970217 Valid Loss: 0.101722 Valid Acc: 0.973817

**Epoch** 8/20 Train Loss: 0.092932 Train Acc: 0.970833 Valid Loss: 0.087102 Valid Acc: 0.976838

**Epoch** 9/20 Train Loss: 0.084106 Train Acc: 0.973433 Valid Loss: 0.090762 Valid Acc: 0.972810

**Epoch** 10/20 Train Loss: 0.082911 Train Acc: 0.975050 Valid Loss: 0.093535 Valid Acc: 0.974824

**Epoch** 11/20 Train Loss: 0.078407 Train Acc: 0.975533 Valid Loss: 0.098648 Valid Acc: 0.976838

**Epoch** 12/20 Train Loss: 0.075375 Train Acc: 0.975850 Valid Loss: 0.090178 Valid Acc: 0.973817

**Epoch** 13/20 Train Loss: 0.073189 Train Acc: 0.976750 Valid Loss: 0.101134 Valid Acc: 0.976838

**Epoch** 14/20 Train Loss: 0.070431 Train Acc: 0.977483 Valid Loss: 0.092704 Valid Acc: 0.974824

**Epoch** 15/20 Train Loss: 0.068375 Train Acc: 0.978250 Valid Loss: 0.094356 Valid Acc: 0.971803

**Epoch** 16/20 Train Loss: 0.065244 Train Acc: 0.979500 Valid Loss: 0.087152 Valid Acc: 0.976838

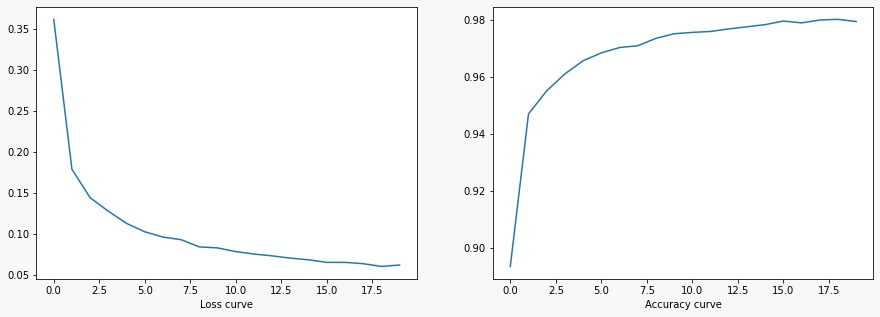
**Epoch** 17/20 Train Loss: 0.065251 Train Acc: 0.978867 Valid Loss: 0.073676 Valid Acc: 0.976838

**Epoch** 18/20 Train Loss: 0.063652 Train Acc: 0.979867 Valid Loss: 0.078486 Valid Acc: 0.976838

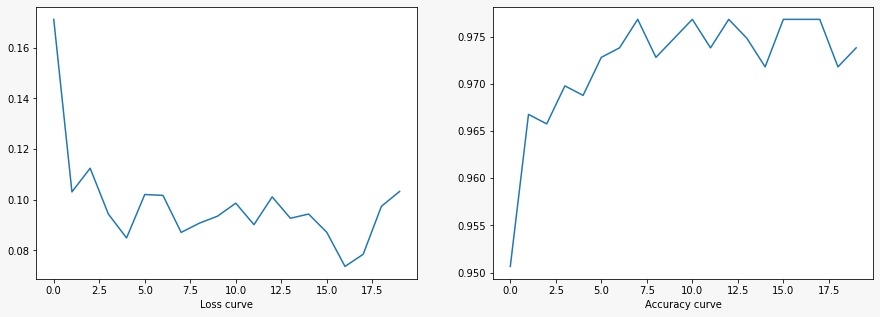
**Epoch** 19/20 Train Loss: 0.060278 Train Acc: 0.980100 Valid Loss: 0.097418 Valid Acc: 0.971803

**Epoch** 20/20 Train Loss: 0.062010 Train Acc: 0.979333 Valid Loss: 0.103286 Valid Acc: 0.973817

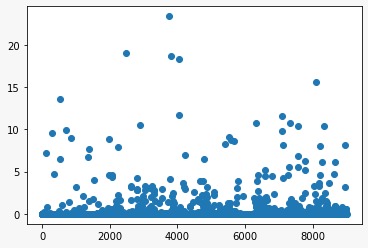
* **Training loss and accuracy curves:**



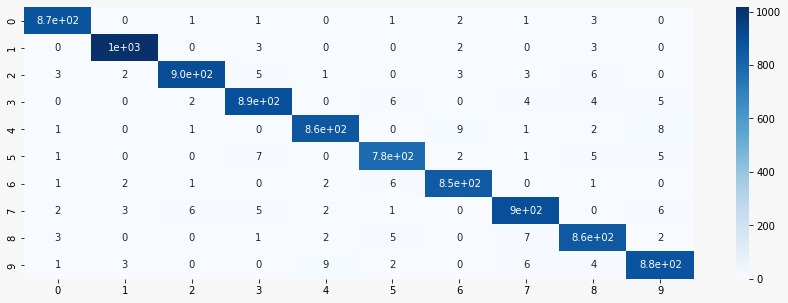
* **Validation loss and accuracy curves:**



* **Test Losses:**

****

* **Test Confusion matrix:**

****

* **Test Accuracy & F1 Score:**

F1 Score is: 0.9790441076639539

Accuracy is: 0.9792383701565449%

**Phase 1 (without Normalization):**

* **Parameters**:

Same as for phase1 (with normalizarion)

* **Training credentials:**

**Epoch** 1/20 Train Loss: 0.402609 Train Acc: 0.879667 Valid Loss: 0.169706 Valid Acc: 0.952669

**Epoch** 2/20 Train Loss: 0.193267 Train Acc: 0.942567 Valid Loss: 0.113909 Valid Acc: 0.958711

**Epoch** 3/20 Train Loss: 0.150488 Train Acc: 0.954767 Valid Loss: 0.119768 Valid Acc: 0.965760

**Epoch** 4/20 Train Loss: 0.130396 Train Acc: 0.961100 Valid Loss: 0.106637 Valid Acc: 0.970796

**Epoch** 5/20 Train Loss: 0.114926 Train Acc: 0.965217 Valid Loss: 0.109819 Valid Acc: 0.966767

**Epoch** 6/20 Train Loss: 0.103207 Train Acc: 0.968033 Valid Loss: 0.095094 Valid Acc: 0.968781

**Epoch** 7/20 Train Loss: 0.096261 Train Acc: 0.970400 Valid Loss: 0.102390 Valid Acc: 0.974824

**Epoch** 8/20 Train Loss: 0.089180 Train Acc: 0.971817 Valid Loss: 0.103354 Valid Acc: 0.969789

**Epoch** 9/20 Train Loss: 0.085033 Train Acc: 0.973700 Valid Loss: 0.088380 Valid Acc: 0.974824

**Epoch** 10/20 Train Loss: 0.081667 Train Acc: 0.974600 Valid Loss: 0.100847 Valid Acc: 0.971803

**Epoch** 11/20 Train Loss: 0.075026 Train Acc: 0.976600 Valid Loss: 0.084854 Valid Acc: 0.976838

**Epoch** 12/20 Train Loss: 0.072304 Train Acc: 0.976633 Valid Loss: 0.097208 Valid Acc: 0.974824

**Epoch** 13/20 Train Loss: 0.070642 Train Acc: 0.977800 Valid Loss: 0.096836 Valid Acc: 0.971803

**Epoch** 14/20 Train Loss: 0.068379 Train Acc: 0.978417 Valid Loss: 0.099694 Valid Acc: 0.973817

**Epoch** 15/20 Train Loss: 0.066902 Train Acc: 0.978750 Valid Loss: 0.100900 Valid Acc: 0.968781

**Epoch** 16/20 Train Loss: 0.062400 Train Acc: 0.980367 Valid Loss: 0.101564 Valid Acc: 0.970796

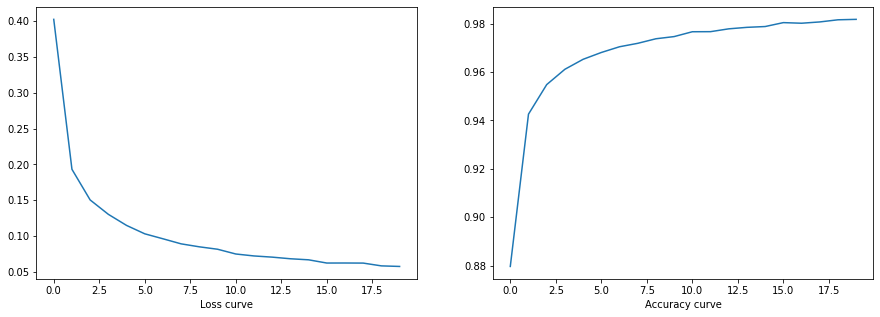
**Epoch** 17/20 Train Loss: 0.062443 Train Acc: 0.980117 Valid Loss: 0.102187 Valid Acc: 0.976838

**Epoch** 18/20 Train Loss: 0.062331 Train Acc: 0.980650 Valid Loss: 0.105287 Valid Acc: 0.971803

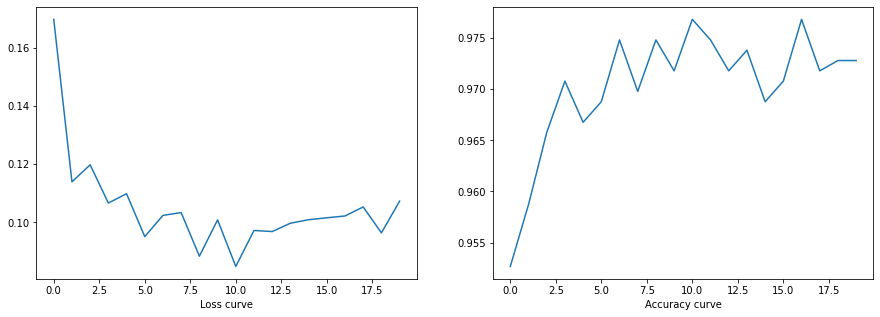
**Epoch** 19/20 Train Loss: 0.058442 Train Acc: 0.981533 Valid Loss: 0.096385 Valid Acc: 0.972810

**Epoch** 20/20 Train Loss: 0.057646 Train Acc: 0.981717 Valid Loss: 0.107281 Valid Acc: 0.972810

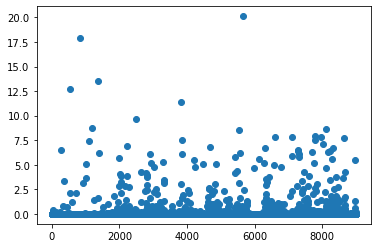
* **Training loss and accuracy curves:**



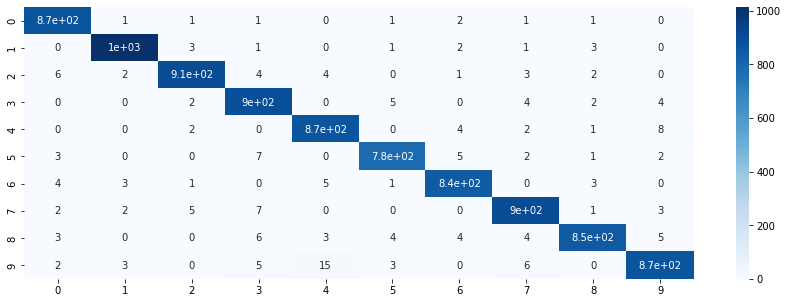
* **Validation loss and accuracy curves:**



* **Test Losses:**

****

* **Test Confusion matrix:**

****

* **Test Accuracy & F1 Score:**

F1 Score is: 0.9782055474278902

Accuracy is: 0.9783501720883757

**Phase 2 (with Normalization):**

* **Parameters**:

loss\_fun = nn.CrossEntropyLoss()

hiddenlayers = 2

listof\_HL = [100,50]

drop = 0.2

batch = 32

optimizer = torch.optim.Adam

learning = 0.01

**Epoch**s = 20

* **Training credentials:**

**Epoch** 1/20 Train Loss: 0.455856 Train Acc: 0.868217 Valid Loss: 0.270976 Valid Acc: 0.940584

**Epoch** 2/20 Train Loss: 0.374843 Train Acc: 0.900750 Valid Loss: 0.267337 Valid Acc: 0.943605

**Epoch** 3/20 Train Loss: 0.352367 Train Acc: 0.909917 Valid Loss: 0.269314 Valid Acc: 0.941591

**Epoch** 4/20 Train Loss: 0.337860 Train Acc: 0.913200 Valid Loss: 0.316670 Valid Acc: 0.939577

**Epoch** 5/20 Train Loss: 0.328801 Train Acc: 0.917583 Valid Loss: 0.306073 Valid Acc: 0.932528

**Epoch** 6/20 Train Loss: 0.332196 Train Acc: 0.918233 Valid Loss: 0.325600 Valid Acc: 0.945619

**Epoch** 7/20 Train Loss: 0.311292 Train Acc: 0.922117 Valid Loss: 0.272519 Valid Acc: 0.954683

**Epoch** 8/20 Train Loss: 0.308854 Train Acc: 0.924517 Valid Loss: 0.282082 Valid Acc: 0.944612

**Epoch** 9/20 Train Loss: 0.304976 Train Acc: 0.925550 Valid Loss: 0.279529 Valid Acc: 0.943605

**Epoch** 10/20 Train Loss: 0.311580 Train Acc: 0.924567 Valid Loss: 0.289903 Valid Acc: 0.946626

**Epoch** 11/20 Train Loss: 0.303816 Train Acc: 0.926867 Valid Loss: 0.289689 Valid Acc: 0.946626

**Epoch** 12/20 Train Loss: 0.303755 Train Acc: 0.925733 Valid Loss: 0.310323 Valid Acc: 0.949648

**Epoch** 13/20 Train Loss: 0.296512 Train Acc: 0.929817 Valid Loss: 0.281628 Valid Acc: 0.948640

**Epoch** 14/20 Train Loss: 0.293480 Train Acc: 0.930100 Valid Loss: 0.337343 Valid Acc: 0.952669

**Epoch** 15/20 Train Loss: 0.295342 Train Acc: 0.929017 Valid Loss: 0.251633 Valid Acc: 0.955690

**Epoch** 16/20 Train Loss: 0.288152 Train Acc: 0.931150 Valid Loss: 0.304592 Valid Acc: 0.955690

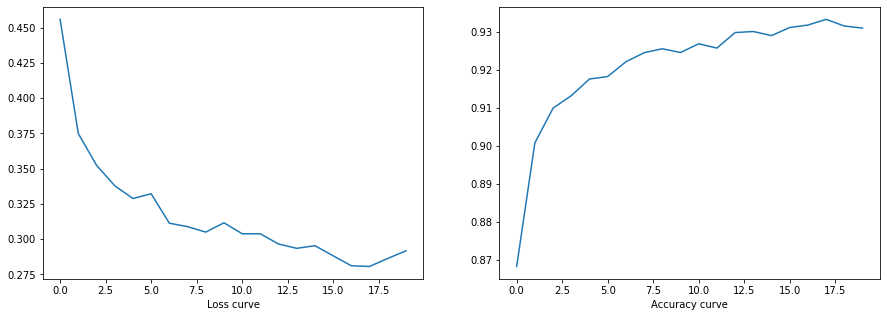
**Epoch** 17/20 Train Loss: 0.281081 Train Acc: 0.931800 Valid Loss: 0.273697 Valid Acc: 0.956697

**Epoch** 18/20 Train Loss: 0.280620 Train Acc: 0.933300 Valid Loss: 0.294012 Valid Acc: 0.937563

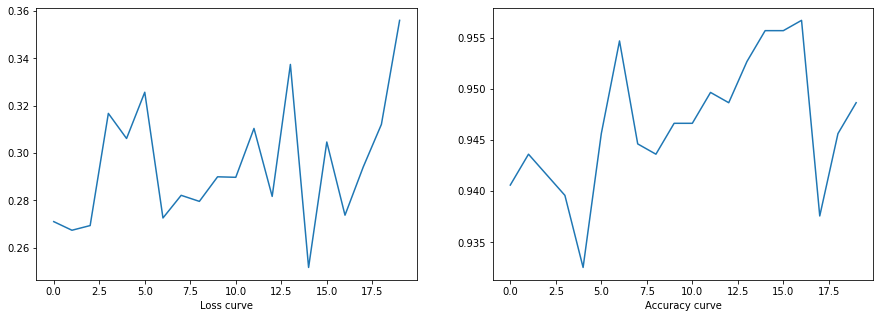
**Epoch** 19/20 Train Loss: 0.286312 Train Acc: 0.931550 Valid Loss: 0.312037 Valid Acc: 0.945619

**Epoch** 20/20 Train Loss: 0.291706 Train Acc: 0.931000 Valid Loss: 0.355949 Valid Acc: 0.948640

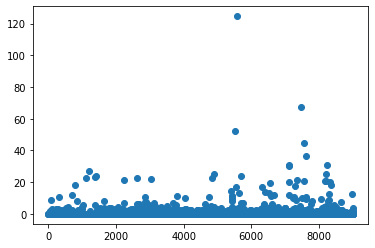
* **Training loss and accuracy curves:**



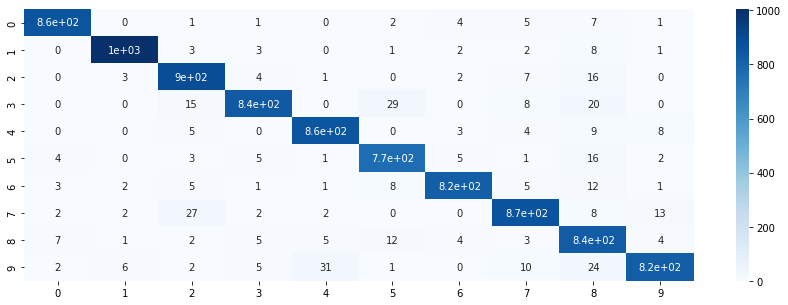
* **Validation loss and accuracy curves:**



* **Test Losses:**

****

* **Test Confusion matrix:**

****

* **Test Accuracy & F1 Score:**

F1 Score is: 0.9519634208327075

Accuracy is: 0.9522593538359054

**Phase 2 (without Normalization):**

* **Parameters**:

Same as in phase 2 (with normalization)

* **Training credentials:**

**Epoch** 1/20 Train Loss: 0.413002 Train Acc: 0.880950 Valid Loss: 0.294203 Valid Acc: 0.920443

**Epoch** 2/20 Train Loss: 0.317736 Train Acc: 0.915000 Valid Loss: 0.253011 Valid Acc: 0.930514

**Epoch** 3/20 Train Loss: 0.294960 Train Acc: 0.918267 Valid Loss: 0.221005 Valid Acc: 0.943605

**Epoch** 4/20 Train Loss: 0.282708 Train Acc: 0.925733 Valid Loss: 0.245902 Valid Acc: 0.943605

**Epoch** 5/20 Train Loss: 0.269876 Train Acc: 0.929183 Valid Loss: 0.217123 Valid Acc: 0.946626

**Epoch** 6/20 Train Loss: 0.264689 Train Acc: 0.932117 Valid Loss: 0.231706 Valid Acc: 0.948640

**Epoch** 7/20 Train Loss: 0.263087 Train Acc: 0.930900 Valid Loss: 0.211481 Valid Acc: 0.957704

**Epoch** 8/20 Train Loss: 0.261721 Train Acc: 0.932800 Valid Loss: 0.229407 Valid Acc: 0.953676

**Epoch** 9/20 Train Loss: 0.259072 Train Acc: 0.933433 Valid Loss: 0.202718 Valid Acc: 0.952669

**Epoch** 10/20 Train Loss: 0.256816 Train Acc: 0.935167 Valid Loss: 0.222564 Valid Acc: 0.940584

**Epoch** 11/20 Train Loss: 0.248428 Train Acc: 0.939100 Valid Loss: 0.236359 Valid Acc: 0.952669

**Epoch** 12/20 Train Loss: 0.255185 Train Acc: 0.936867 Valid Loss: 0.219393 Valid Acc: 0.957704

**Epoch** 13/20 Train Loss: 0.241388 Train Acc: 0.940650 Valid Loss: 0.243623 Valid Acc: 0.961732

**Epoch** 14/20 Train Loss: 0.245899 Train Acc: 0.938717 Valid Loss: 0.282316 Valid Acc: 0.948640

**Epoch** 15/20 Train Loss: 0.232641 Train Acc: 0.941917 Valid Loss: 0.265018 Valid Acc: 0.946626

**Epoch** 16/20 Train Loss: 0.242080 Train Acc: 0.940733 Valid Loss: 0.246587 Valid Acc: 0.953676

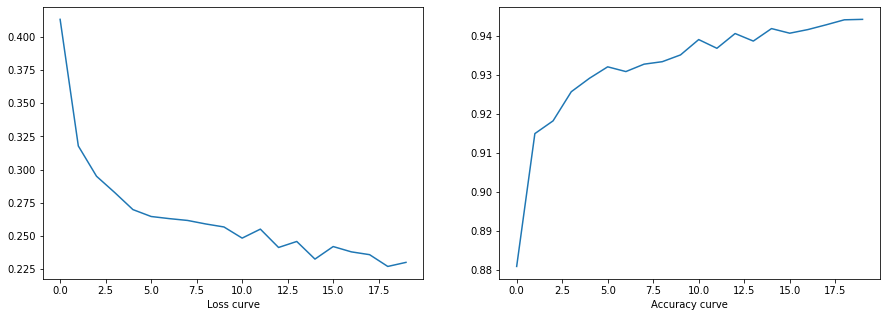
**Epoch** 17/20 Train Loss: 0.238081 Train Acc: 0.941667 Valid Loss: 0.245807 Valid Acc: 0.947633

**Epoch** 18/20 Train Loss: 0.235978 Train Acc: 0.942883 Valid Loss: 0.294278 Valid Acc: 0.944612

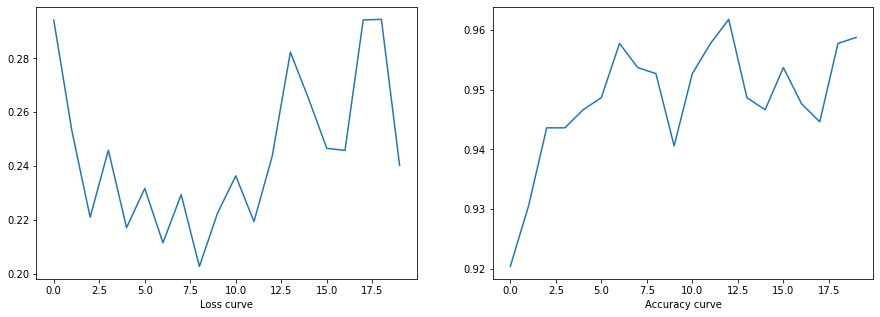
**Epoch** 19/20 Train Loss: 0.227112 Train Acc: 0.944167 Valid Loss: 0.294496 Valid Acc: 0.957704

**Epoch** 20/20 Train Loss: 0.230174 Train Acc: 0.944283 Valid Loss: 0.240283 Valid Acc: 0.958711

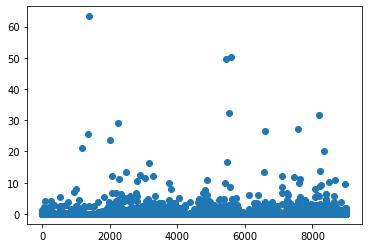
* **Training loss and accuracy curves:**



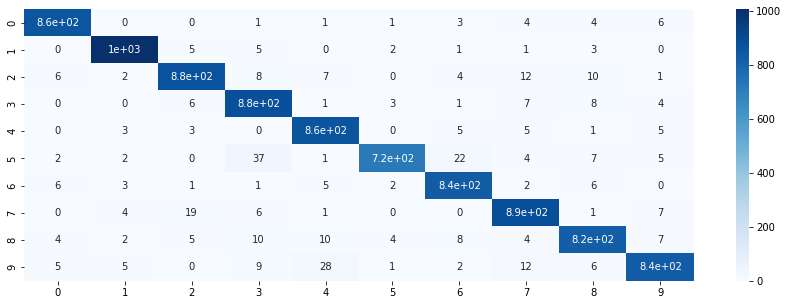
* **Validation loss and accuracy curves:**



* **Test Losses:**

****

* **Test Confusion matrix:**

****

* **Test Accuracy & F1 Score:**

F1 Score is: 0.9545073481874681

Accuracy is: 0.9550349727989341

**Phase 3 (with Normalization):**

* **Parameters**:

loss\_fun = nn.CrossEntropyLoss()

hiddenlayers = 4

listof\_HL = [400,200,100,50]

drop = 0.3

batch = 64

optimizer = torch.optim.Adam

learning = 0.01

epochs = 10

* **Training credentials:**

**Epoch** 1/10 Train Loss: 0.749605 Train Acc: 0.795383 Valid Loss: 0.484868 Valid Acc: 0.877140

**Epoch** 2/10 Train Loss: 0.625464 Train Acc: 0.848617 Valid Loss: 0.325918 Valid Acc: 0.919436

**Epoch** 3/10 Train Loss: 0.644855 Train Acc: 0.845700 Valid Loss: 0.349141 Valid Acc: 0.924471

**Epoch** 4/10 Train Loss: 0.626675 Train Acc: 0.851183 Valid Loss: 0.302991 Valid Acc: 0.923464

**Epoch** 5/10 Train Loss: 0.645091 Train Acc: 0.844200 Valid Loss: 0.405444 Valid Acc: 0.915408

**Epoch** 6/10 Train Loss: 0.648434 Train Acc: 0.845967 Valid Loss: 0.342654 Valid Acc: 0.917422

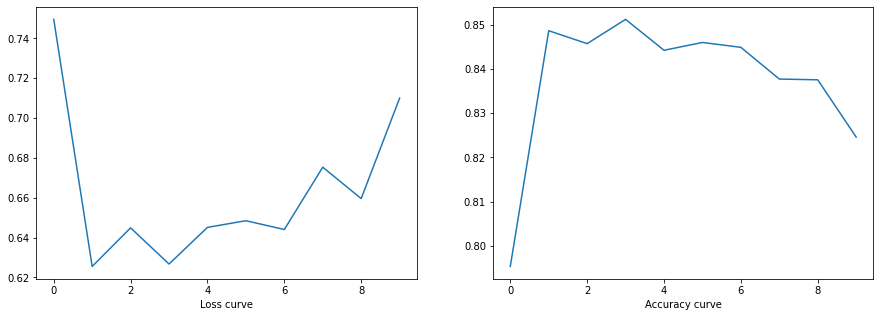
**Epoch** 7/10 Train Loss: 0.644022 Train Acc: 0.844867 Valid Loss: 0.271977 Valid Acc: 0.936556

**Epoch** 8/10 Train Loss: 0.675332 Train Acc: 0.837700 Valid Loss: 0.355497 Valid Acc: 0.928499

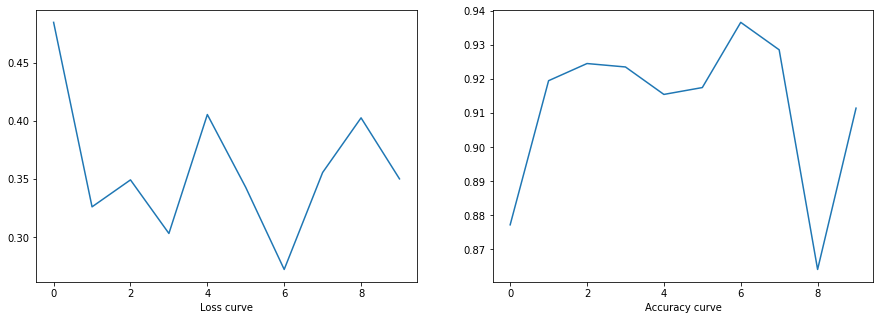
**Epoch** 9/10 Train Loss: 0.659562 Train Acc: 0.837533 Valid Loss: 0.402595 Valid Acc: 0.864048

**Epoch** 10/10 Train Loss: 0.710025 Train Acc: 0.824600 Valid Loss: 0.350022 Valid Acc: 0.911380

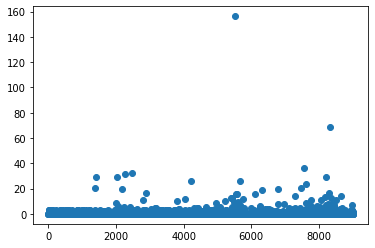
* **Training loss and accuracy curves:**



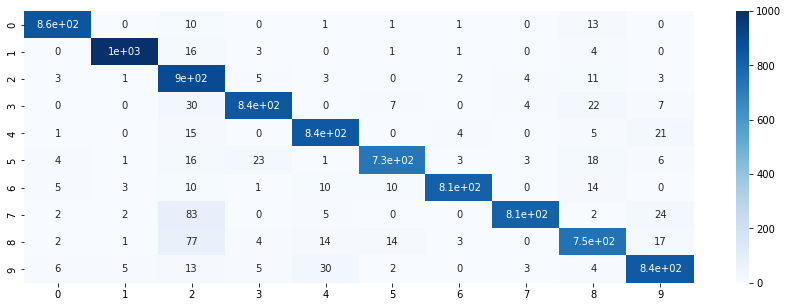
* **Validation loss and accuracy curves:**



* **Test Losses:**

****

* **Test Confusion matrix:**

****

* **Test Accuracy & F1 Score:**

F1 Score is: 0.9289256445559533

Accuracy is: 0.9283890307538581

**Phase 4 (with Normalization):**

* **Parameters**:

loss\_fun = nn.CrossEntropyLoss()

hiddenlayers = 4

listof\_HL = [400,200,100,50]

drop = 0.2

batch = 32

optimizer = torch.optim.Adam

learning = 0.001

epochs = 20

* **Training credentials:**

**Epoch** 1/20 Train Loss: 0.334240 Train Acc: 0.900650 Valid Loss: 0.147703 Valid Acc: 0.957704

**Epoch** 2/20 Train Loss: 0.154570 Train Acc: 0.958633 Valid Loss: 0.119410 Valid Acc: 0.970796

**Epoch** 3/20 Train Loss: 0.119711 Train Acc: 0.966867 Valid Loss: 0.107839 Valid Acc: 0.972810

**Epoch** 4/20 Train Loss: 0.100287 Train Acc: 0.972733 Valid Loss: 0.112135 Valid Acc: 0.969789

**Epoch** 5/20 Train Loss: 0.084467 Train Acc: 0.976633 Valid Loss: 0.079341 Valid Acc: 0.976838

**Epoch** 6/20 Train Loss: 0.075263 Train Acc: 0.979350 Valid Loss: 0.080713 Valid Acc: 0.979859

**Epoch** 7/20 Train Loss: 0.068618 Train Acc: 0.981567 Valid Loss: 0.090435 Valid Acc: 0.973817

**Epoch** 8/20 Train Loss: 0.063422 Train Acc: 0.982867 Valid Loss: 0.088221 Valid Acc: 0.977845

**Epoch** 9/20 Train Loss: 0.060011 Train Acc: 0.983033 Valid Loss: 0.082883 Valid Acc: 0.980866

**Epoch** 10/20 Train Loss: 0.058665 Train Acc: 0.983450 Valid Loss: 0.089946 Valid Acc: 0.975831

**Epoch** 11/20 Train Loss: 0.051295 Train Acc: 0.985600 Valid Loss: 0.083425 Valid Acc: 0.975831

**Epoch** 12/20 Train Loss: 0.049342 Train Acc: 0.985983 Valid Loss: 0.066253 Valid Acc: 0.978852

**Epoch** 13/20 Train Loss: 0.045166 Train Acc: 0.987483 Valid Loss: 0.098020 Valid Acc: 0.979859

**Epoch** 14/20 Train Loss: 0.045409 Train Acc: 0.987583 Valid Loss: 0.105048 Valid Acc: 0.982880

**Epoch** 15/20 Train Loss: 0.043461 Train Acc: 0.987717 Valid Loss: 0.096022 Valid Acc: 0.976838

**Epoch** 16/20 Train Loss: 0.040016 Train Acc: 0.989233 Valid Loss: 0.120140 Valid Acc: 0.976838

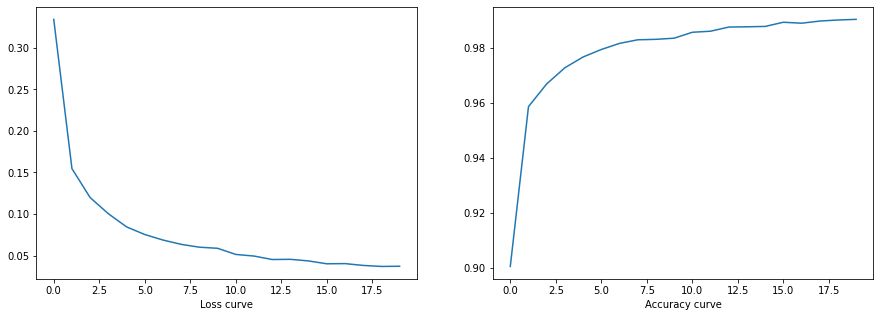
**Epoch** 17/20 Train Loss: 0.040253 Train Acc: 0.988883 Valid Loss: 0.091658 Valid Acc: 0.975831

**Epoch** 18/20 Train Loss: 0.038059 Train Acc: 0.989667 Valid Loss: 0.109046 Valid Acc: 0.976838

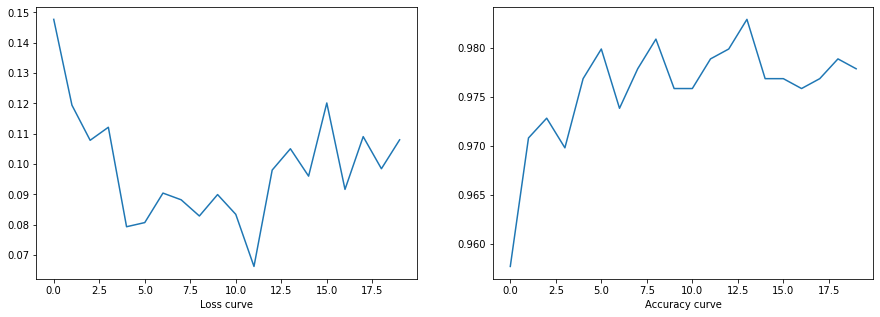
**Epoch** 19/20 Train Loss: 0.036831 Train Acc: 0.990050 Valid Loss: 0.098462 Valid Acc: 0.978852

**Epoch** 20/20 Train Loss: 0.037053 Train Acc: 0.990283 Valid Loss: 0.107990 Valid Acc: 0.977845

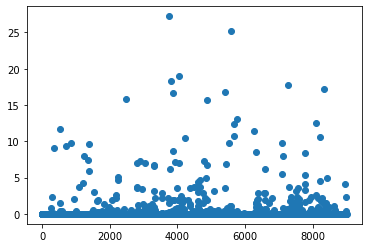
* **Training loss and accuracy curves:**



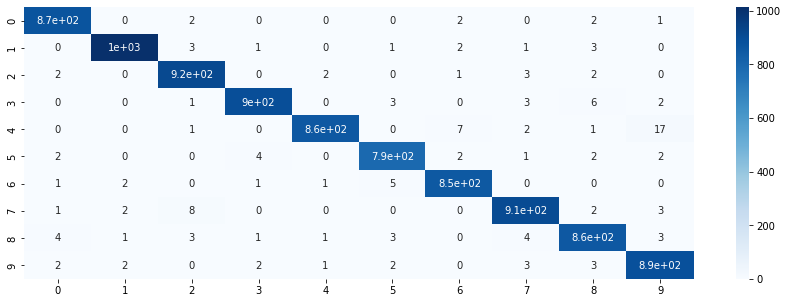
* **Validation loss and accuracy curves:**



* **Test Losses:**

****

* **Test Confusion matrix:**

****

* **Test Accuracy & F1 Score:**

F1 Score is: 0.9837948229283574

Accuracy is: 0.9839014100144332

**Phase 5 (with Normalization):**

* **Parameters**:

loss\_fun = nn.CrossEntropyLoss()

hiddenlayers = 4

listof\_HL = [400,200,100,50]

drop = 0.2

batch = 32

optimizer = torch.optim.SGD

learning = 0.001

epochs = 40

* **Training credentials:**

**Epoch** 1/40 Train Loss: 2.302361 Train Acc: 0.103450 Valid Loss: 2.300111 Valid Acc: 0.101712

**Epoch** 2/40 Train Loss: 2.296407 Train Acc: 0.134533 Valid Loss: 2.292299 Valid Acc: 0.215509

**Epoch** 3/40 Train Loss: 2.288360 Train Acc: 0.213083 Valid Loss: 2.280910 Valid Acc: 0.281974

**Epoch** 4/40 Train Loss: 2.274621 Train Acc: 0.266600 Valid Loss: 2.259179 Valid Acc: 0.320242

**Epoch** 5/40 Train Loss: 2.245332 Train Acc: 0.330817 Valid Loss: 2.207883 Valid Acc: 0.448137

**Epoch** 6/40 Train Loss: 2.155680 Train Acc: 0.386967 Valid Loss: 2.024178 Valid Acc: 0.463243

**Epoch** 7/40 Train Loss: 1.831907 Train Acc: 0.417617 Valid Loss: 1.506675 Valid Acc: 0.546828

**Epoch** 8/40 Train Loss: 1.428882 Train Acc: 0.502117 Valid Loss: 1.159974 Valid Acc: 0.662638

**Epoch** 9/40 Train Loss: 1.188064 Train Acc: 0.596467 Valid Loss: 0.938682 Valid Acc: 0.732125

**Epoch** 10/40 Train Loss: 0.995144 Train Acc: 0.675017 Valid Loss: 0.766857 Valid Acc: 0.789527

**Epoch** 11/40 Train Loss: 0.857028 Train Acc: 0.725667 Valid Loss: 0.651986 Valid Acc: 0.817724

**Epoch** 12/40 Train Loss: 0.752008 Train Acc: 0.764717 Valid Loss: 0.576002 Valid Acc: 0.841893

**Epoch** 13/40 Train Loss: 0.679458 Train Acc: 0.792583 Valid Loss: 0.528577 Valid Acc: 0.856999

**Epoch** 14/40 Train Loss: 0.634786 Train Acc: 0.807500 Valid Loss: 0.485899 Valid Acc: 0.860020

**Epoch** 15/40 Train Loss: 0.588471 Train Acc: 0.822833 Valid Loss: 0.452208 Valid Acc: 0.869084

**Epoch** 16/40 Train Loss: 0.553812 Train Acc: 0.835450 Valid Loss: 0.421325 Valid Acc: 0.875126

**Epoch** 17/40 Train Loss: 0.525200 Train Acc: 0.845900 Valid Loss: 0.390222 Valid Acc: 0.883182

**Epoch** 18/40 Train Loss: 0.497239 Train Acc: 0.853933 Valid Loss: 0.369228 Valid Acc: 0.891239

**Epoch** 19/40 Train Loss: 0.469780 Train Acc: 0.862417 Valid Loss: 0.344993 Valid Acc: 0.892246

**Epoch** 20/40 Train Loss: 0.449202 Train Acc: 0.868883 Valid Loss: 0.327057 Valid Acc: 0.900302

**Epoch** 21/40 Train Loss: 0.430058 Train Acc: 0.876417 Valid Loss: 0.307272 Valid Acc: 0.903323

**Epoch** 22/40 Train Loss: 0.409660 Train Acc: 0.883400 Valid Loss: 0.291746 Valid Acc: 0.912387

**Epoch** 23/40 Train Loss: 0.392519 Train Acc: 0.886667 Valid Loss: 0.277979 Valid Acc: 0.912387

**Epoch** 24/40 Train Loss: 0.378515 Train Acc: 0.891533 Valid Loss: 0.265158 Valid Acc: 0.917422

**Epoch** 25/40 Train Loss: 0.361012 Train Acc: 0.896917 Valid Loss: 0.250794 Valid Acc: 0.922457

**Epoch** 26/40 Train Loss: 0.344920 Train Acc: 0.902383 Valid Loss: 0.240595 Valid Acc: 0.923464

**Epoch** 27/40 Train Loss: 0.333951 Train Acc: 0.904650 Valid Loss: 0.228019 Valid Acc: 0.926485

**Epoch** 28/40 Train Loss: 0.319179 Train Acc: 0.909567 Valid Loss: 0.219610 Valid Acc: 0.931521

**Epoch** 29/40 Train Loss: 0.310785 Train Acc: 0.912833 Valid Loss: 0.211365 Valid Acc: 0.931521

**Epoch** 30/40 Train Loss: 0.297166 Train Acc: 0.917350 Valid Loss: 0.201226 Valid Acc: 0.937563

**Epoch** 31/40 Train Loss: 0.289305 Train Acc: 0.918550 Valid Loss: 0.194305 Valid Acc: 0.940584

**Epoch** 32/40 Train Loss: 0.276787 Train Acc: 0.922933 Valid Loss: 0.188229 Valid Acc: 0.941591

**Epoch** 33/40 Train Loss: 0.267589 Train Acc: 0.924617 Valid Loss: 0.181407 Valid Acc: 0.945619

**Epoch** 34/40 Train Loss: 0.260552 Train Acc: 0.925083 Valid Loss: 0.175504 Valid Acc: 0.943605

**Epoch** 35/40 Train Loss: 0.253338 Train Acc: 0.928467 Valid Loss: 0.170759 Valid Acc: 0.948640

**Epoch** 36/40 Train Loss: 0.244515 Train Acc: 0.931650 Valid Loss: 0.164527 Valid Acc: 0.952669

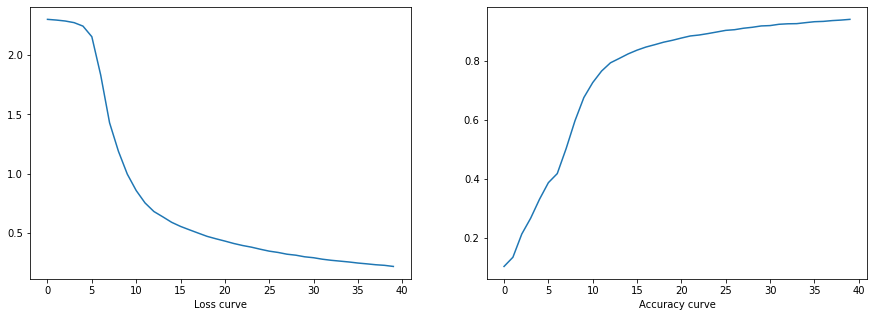
**Epoch** 37/40 Train Loss: 0.237701 Train Acc: 0.932733 Valid Loss: 0.160601 Valid Acc: 0.952669

**Epoch** 38/40 Train Loss: 0.230184 Train Acc: 0.935433 Valid Loss: 0.154694 Valid Acc: 0.951662

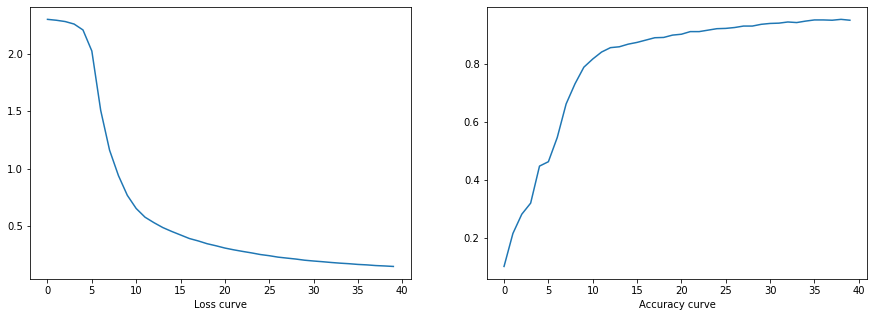
**Epoch** 39/40 Train Loss: 0.225043 Train Acc: 0.937183 Valid Loss: 0.151198 Valid Acc: 0.954683

**Epoch** 40/40 Train Loss: 0.215709 Train Acc: 0.939633 Valid Loss: 0.146922 Valid Acc: 0.951662

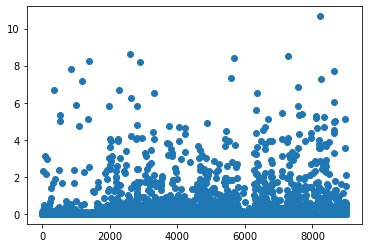
* **Training loss and accuracy curves:**



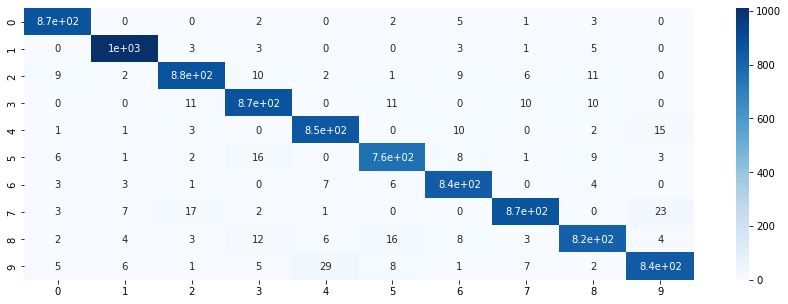
* **Validation loss and accuracy curves:**



* **Test Losses:**

****

* **Test Confusion matrix:**

****

* **Test Accuracy & F1 Score:**

F1 Score is: 0.9554441747828826

Accuracy is: 0.9559231708671033

* **Sample Predictions**:

