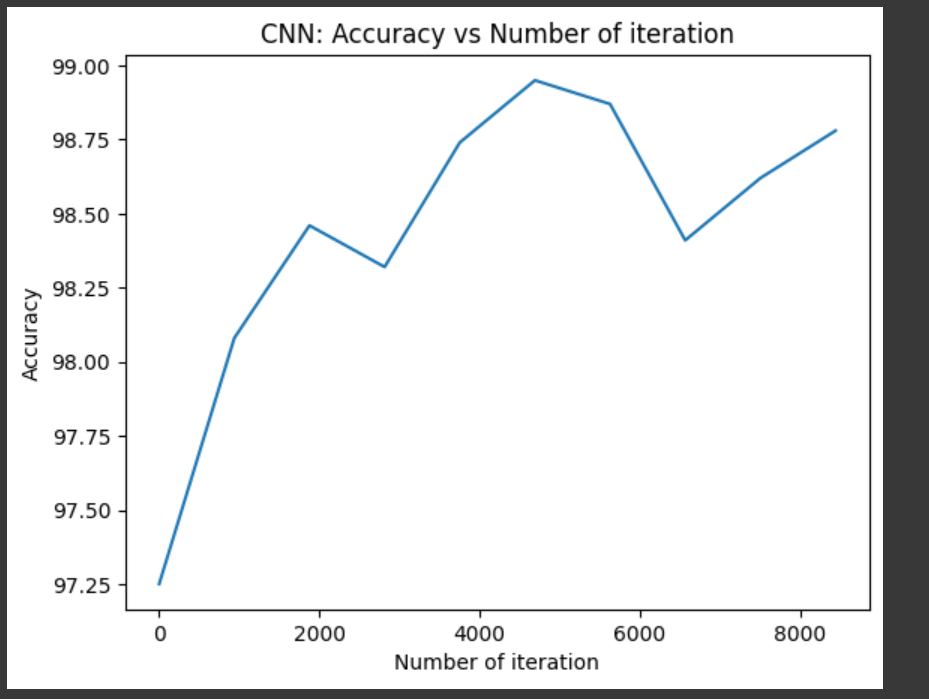
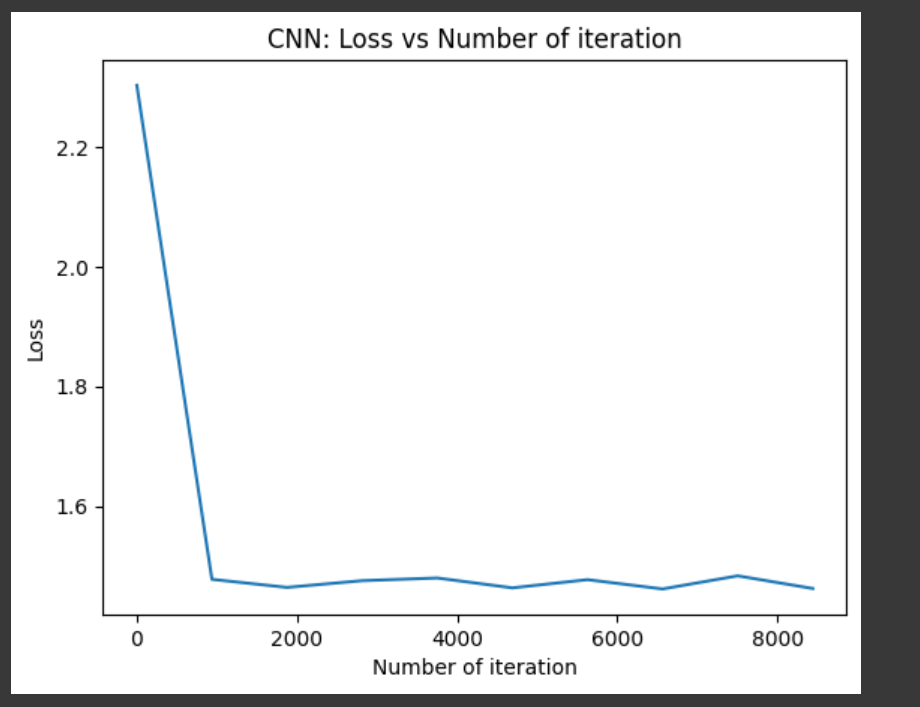
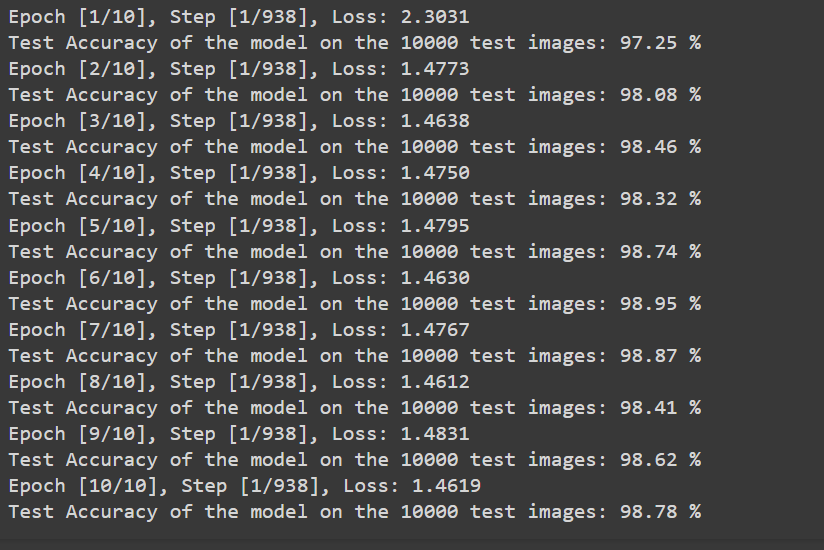
**CNN models on MNIST**

Imp observation- Use of CPU led to the running time of 10 min for 10 epochs but on the other hand using the GPU from the google colab helped to reduce the running time by 90% and decreased the time to just 1 min. Hence the GPU is used in the complete codes.

The results with hyper parameters as:

* Lr = 0.001
* OUT channels = 32
* Kernal size = 3
* Batch size = 64

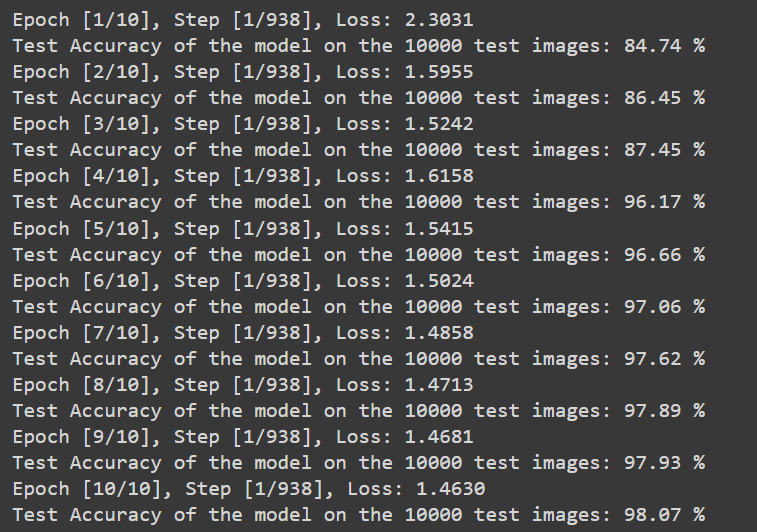


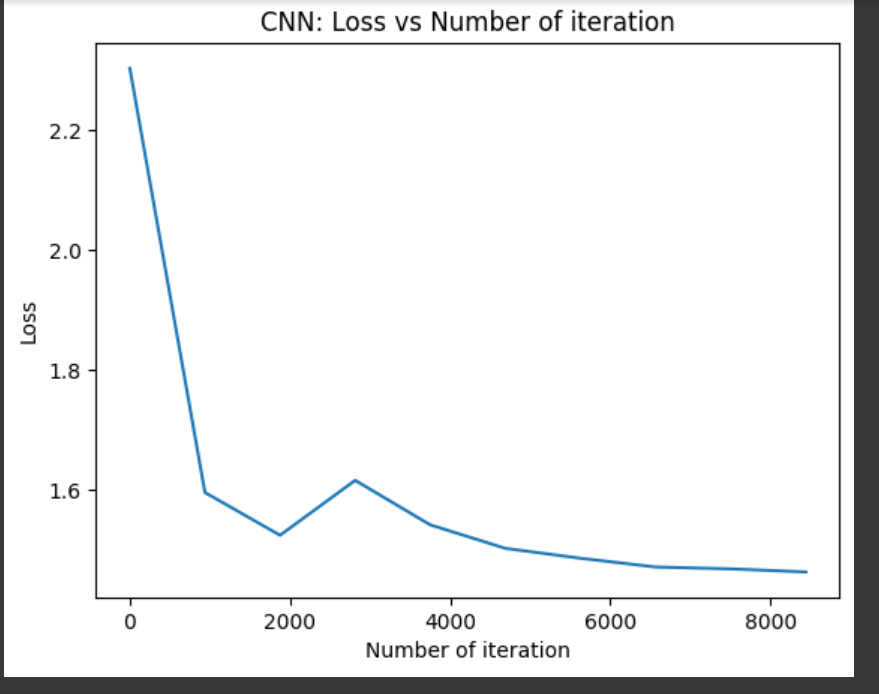
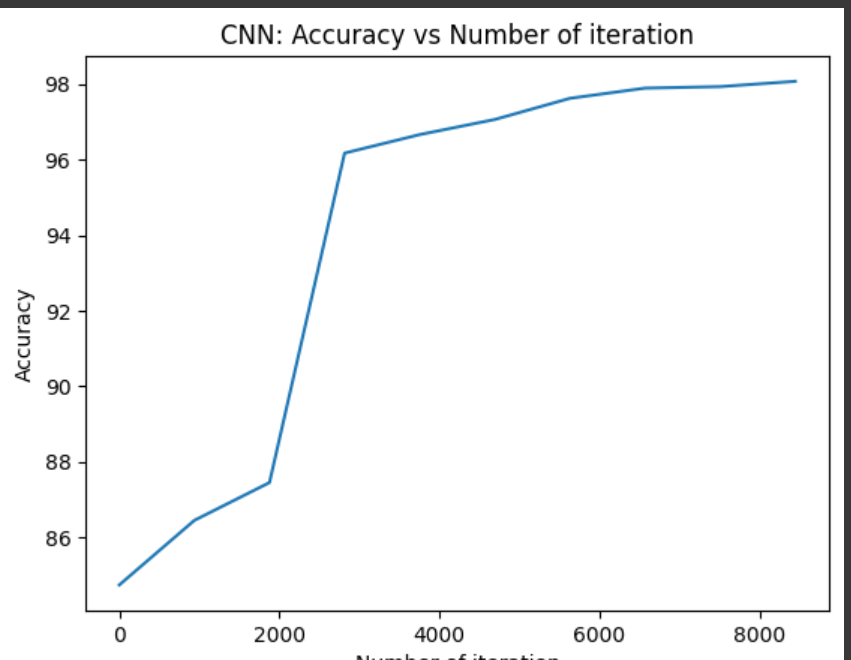
Conclusion- The output is quite good but some overfitting maybe is happening as the test accuracy is non monotonous.

**CNN models on MNIST**

The results with hyper parameters as:

* Lr = 0.0001
* OUT channels = 32
* Kernal size = 3
* Batch size = 64



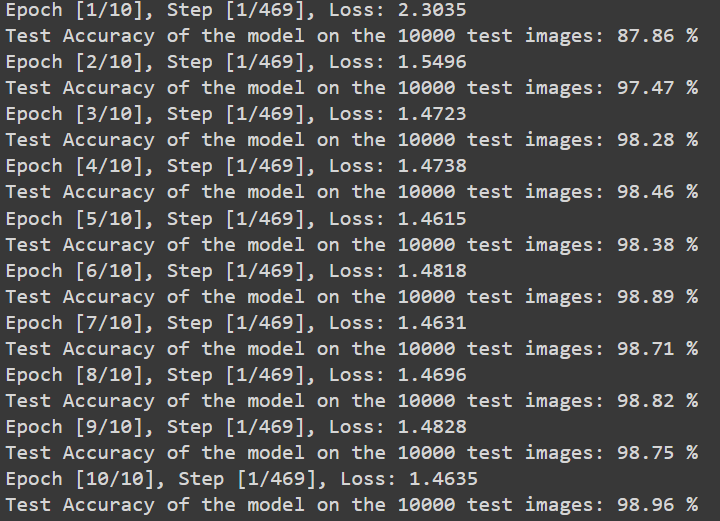
 

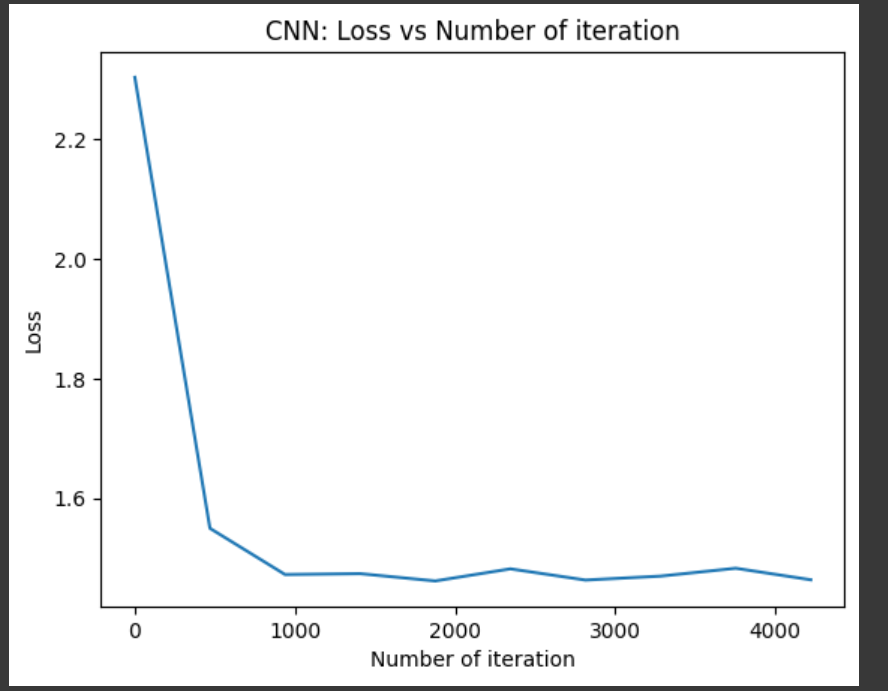
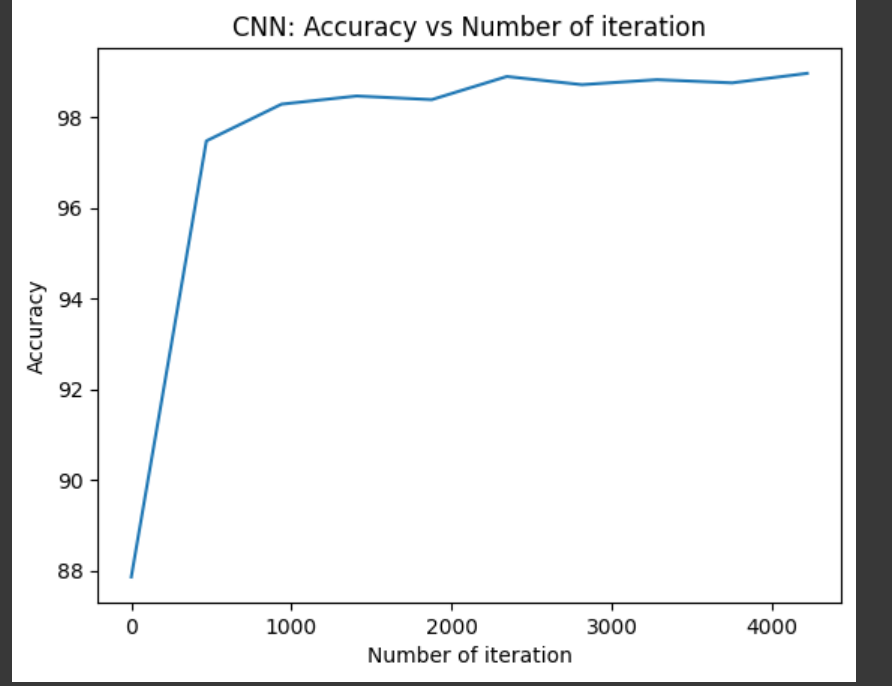
Conclusion- The learning process has become somewhat slow but the problem of overfitting has somewhat solved. So the above lr = 0.001 is better as we can proceed further.

**CNN models on MNIST**

The results with hyper parameters as:

* Lr = 0.001
* OUT channels = 32
* Kernal size = 3
* Batch size = 128



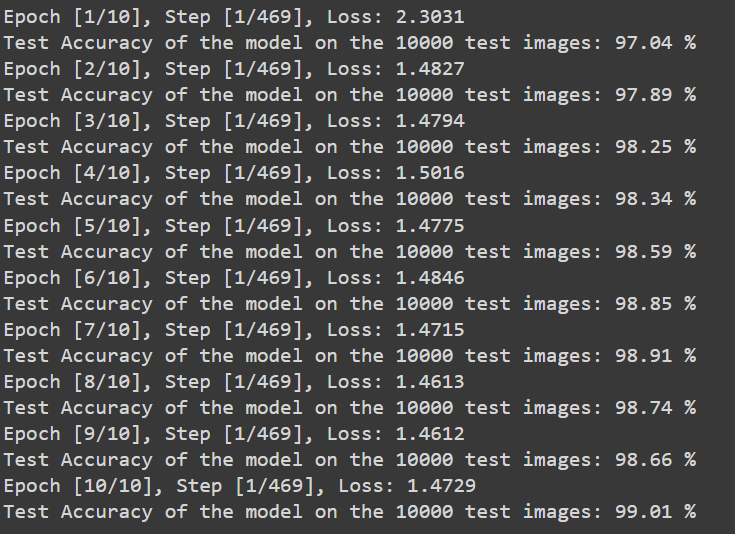
 

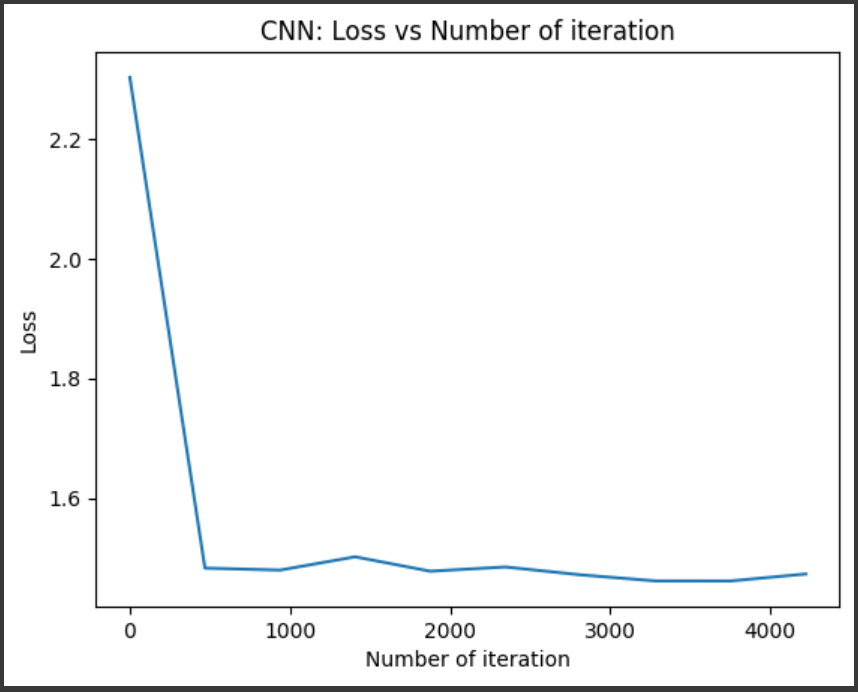
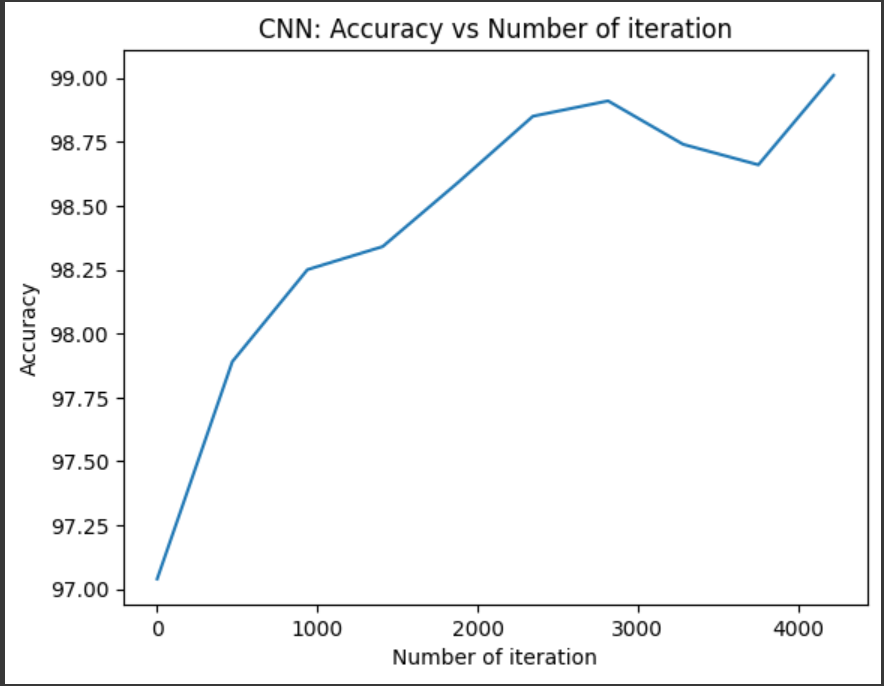
Conclusion- On some further research on the MNIST, I fount that the result with batch size= 128 gives quite better results than that for batch size of 64 or 256. Hence this batch size is finalized. The use of this batch size also speeds up the training time for 10 epochs.

**CNN models on MNIST**

The results with hyper parameters as:

* **Lr = 0.001**
* **OUT channels = 64**
* **Kernal size = 3**
* **Batch size = 128**



Final Conclusion

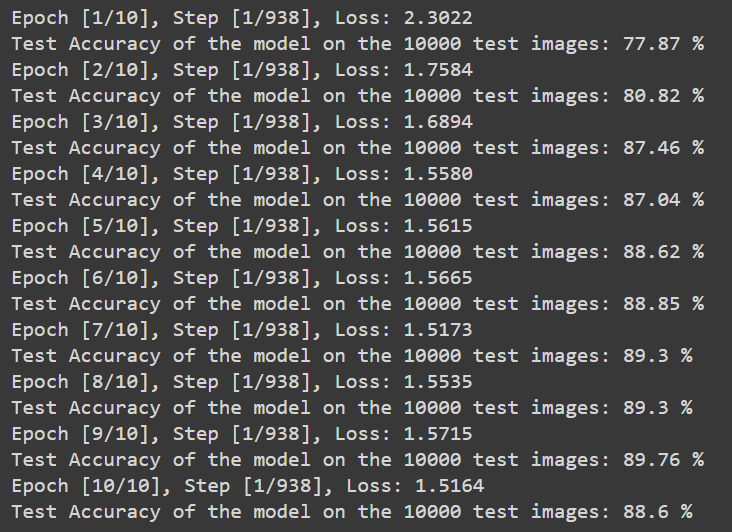
These are the final results that I obtained on using the above hyper parameters.

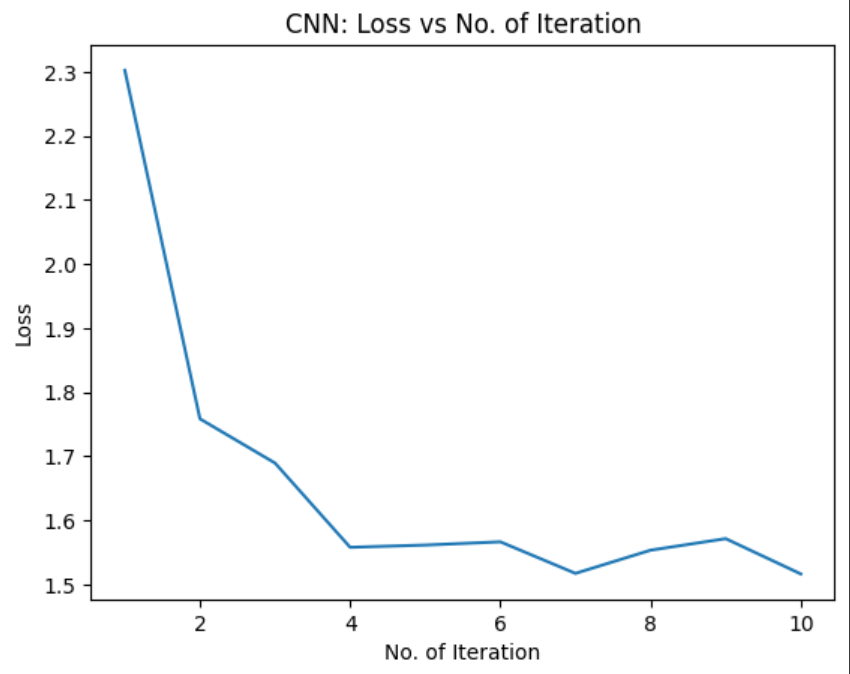
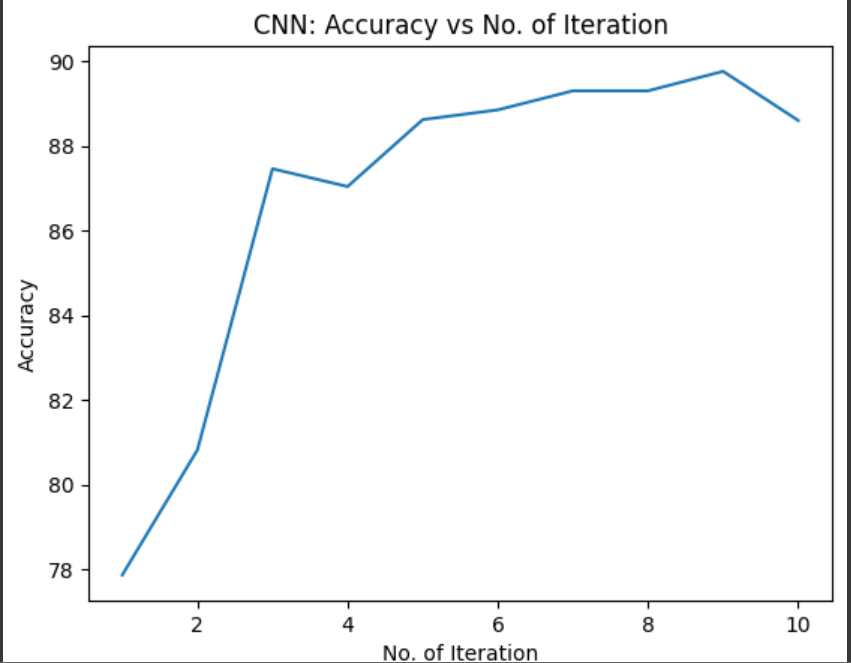
**The best Test Accuracy achieved= 99.01%**

**CNN models on Fashion\_MNIST**

The results with hyper parameters as:

* Lr = 0.001
* OUT channels = 32
* Kernal size = 3
* Batch size = 64



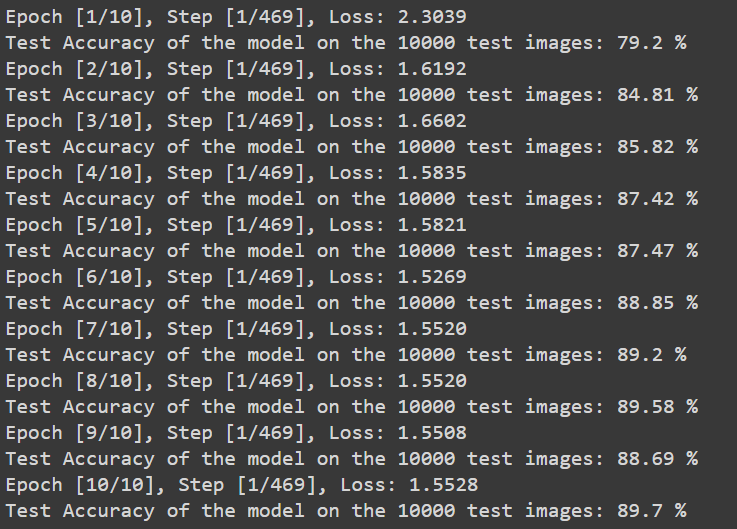
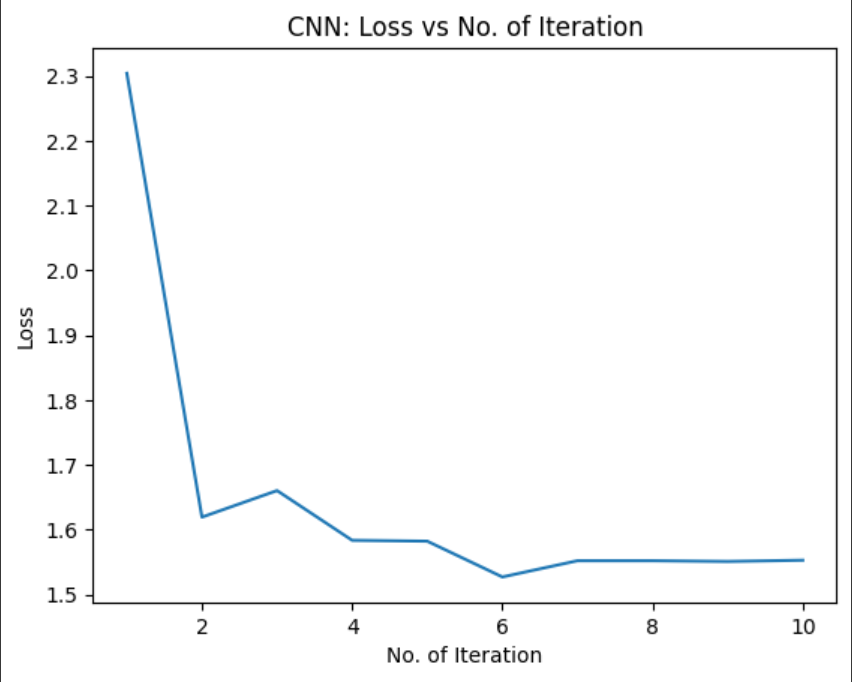
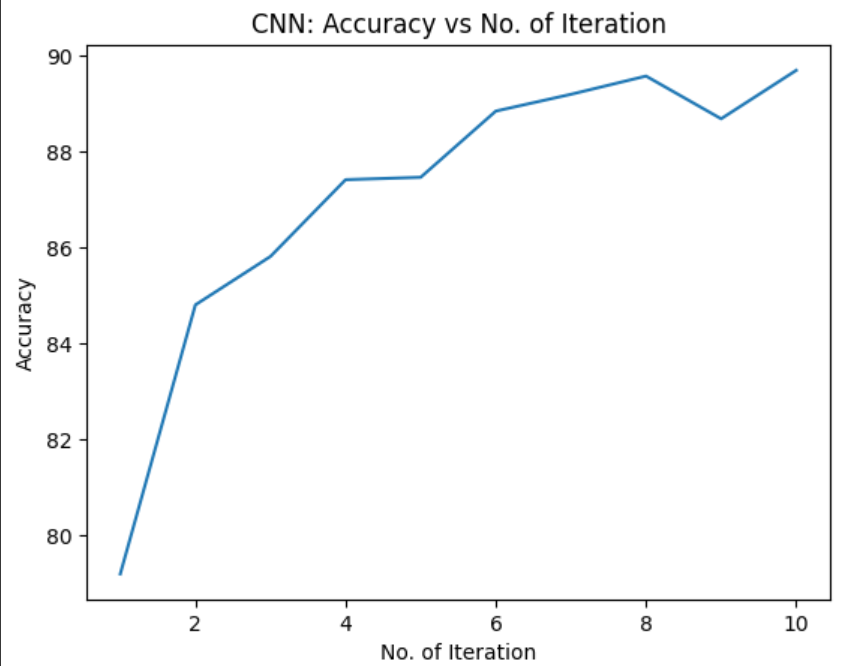
 

Conclusion- The output is quite good but some overfitting maybe is happening as the test accuracy is non monotonous.

**CNN models on Fashion\_MNIST**

The results with hyper parameters as:

* **Lr = 0.001**
* **OUT channels = 64**
* **Kernal size = 3**
* **Batch size = 128**

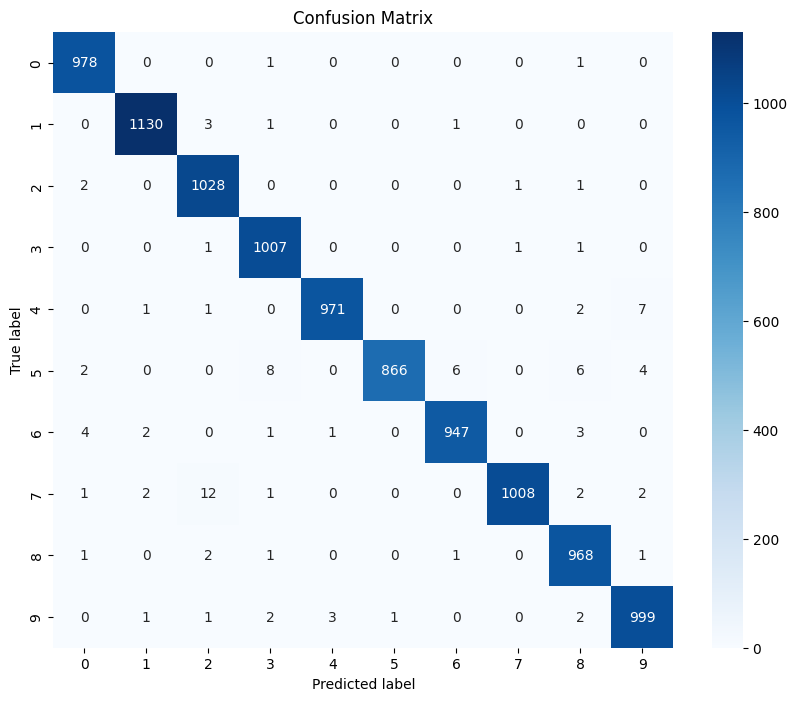
These are the final results that I obtained on using the above hyper parameters.

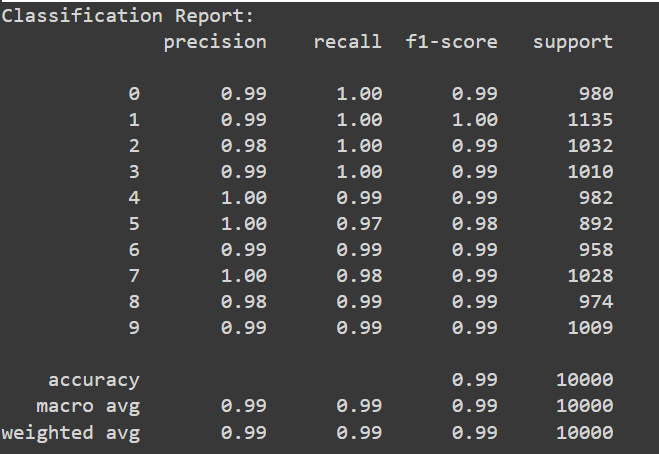
**The best Test Accuracy achieved= 89.70%**

**Overall Conclusion on selection of Hyper parameters->**

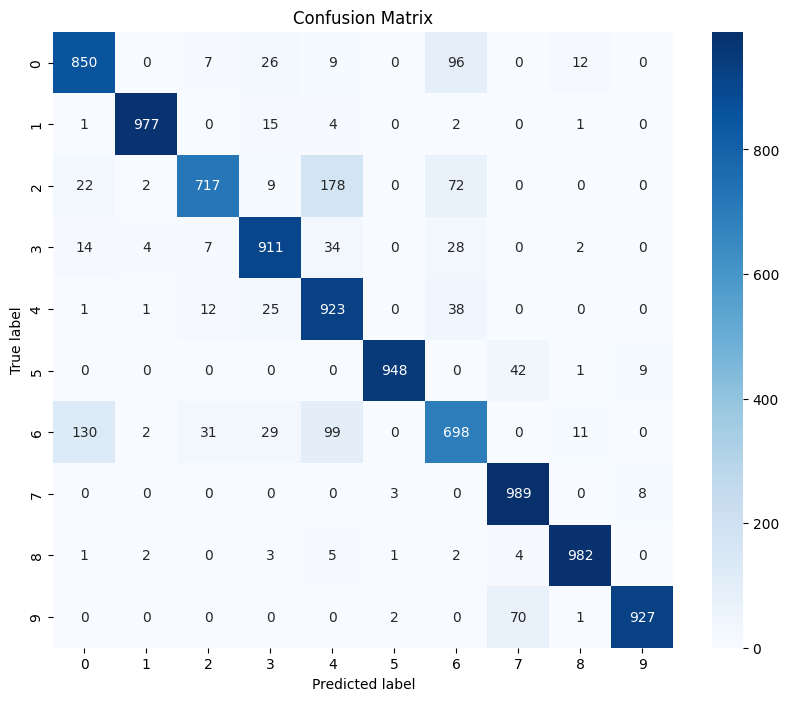
* In terms of Learning rate 0.001 seems to be the best suited one with some good speed and early convergence towards the most accurate result.
* In terms of Batch size 128 is selected to be the best as with lower sizes like 64 or 32 the training process gets somewhat slowed and whereas for higher sizes like 256 the learning process becomes too fast and it also hampers the overall accuracy of the result.
* Similarly, is the case with no of out channels and the kernel size which are selected to be 64 and 3 respectively after multiple hit and trials.
* The use of cuda or the GPU in the google colab helped a lot in speeding up the training time and decreasing it by 90% as compared to usual CPU.

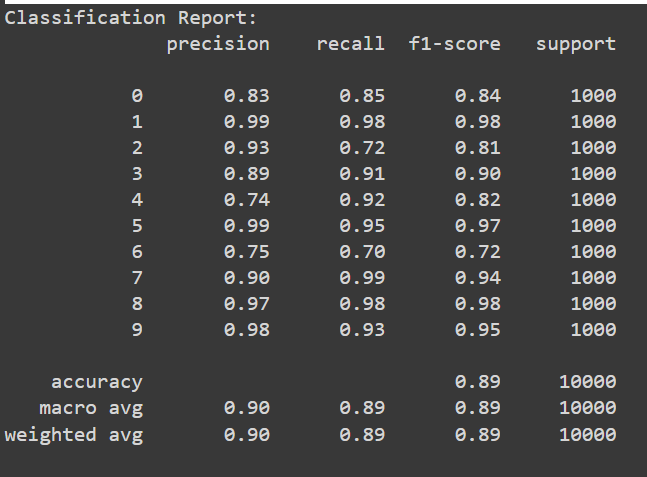
**The Confusion Matrix of CNN model on MNIST dataset**





**The Confusion Matrix of CNN model on Fashion\_MNIST dataset**



****