AIML Project

Sohail Abbasi-206631, Omer Usman Uppal-203515, M Yasir Ali Khan-206226

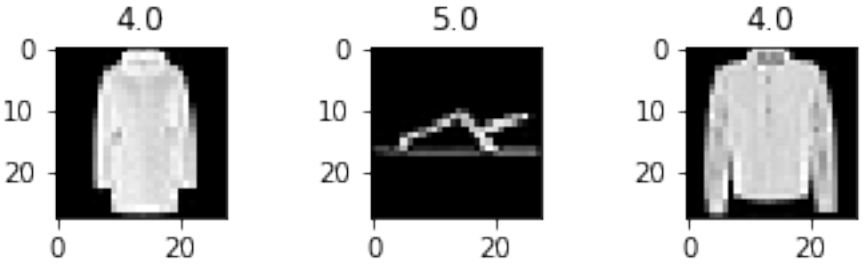
May 29, 2018

1 K Nearest Neighbour Classifier

- DATASET has 60,000 training samples and 10,000 test samples.
- First model is trained using the training dataset.
- Prediction of labels is calculated both for test and training samples.
- Due to very high computation power we have predicted labels for 10,000 training samples
- All samples of Test dataset are used for predictions.
- kNN classifier is very slow as it has to find the similary of each sample of test data with each sample of data used for model. Once all computation is done than labels are predicted based for K nearest neigbours

1.1 Display few images

After reading the data in pandas dataframe, images are displayed to check if data is read correctly. Data is loaded in the dataframe using the fashion-mnist-train csv file and fashion-mnist-test csv file.



1.2 Results of kNN Classifier when k = 3

1.2.1 Accuracy when k = 3

```
Accuracy of the system on training dataset (10000 samples) when k=3 is 0.9215
Accuracy of the system on test dataset when k=3 is 0.8584
```

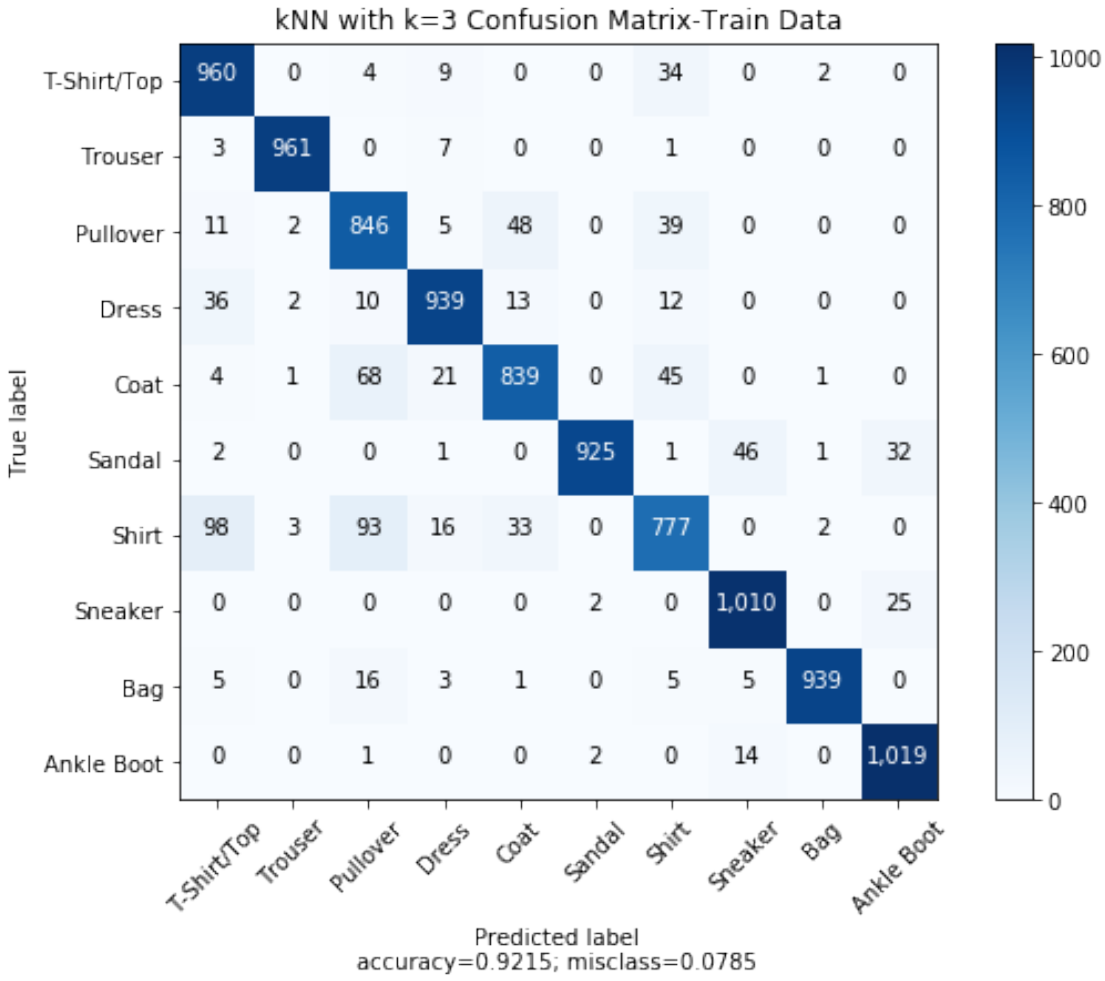
1.2.2 Precision, Recall and F1-score at k = 3 and training data

	precision	recall	f1-score	support
T-Shirt/Top	0.86	0.95	0.90	1009
Trouser	0.99	0.99	0.99	972
Pullover	0.82	0.89	0.85	951
Dress	0.94	0.93	0.93	1012
Coat	0.90	0.86	0.88	979
Sandal	1.00	0.92	0.96	1008
Shirt	0.85	0.76	0.80	1022
Sneaker	0.94	0.97	0.96	1037
Bag	0.99	0.96	0.98	974
Ankle Boot	0.95	0.98	0.96	1036
avg / total	0.92	0.92	0.92	10000

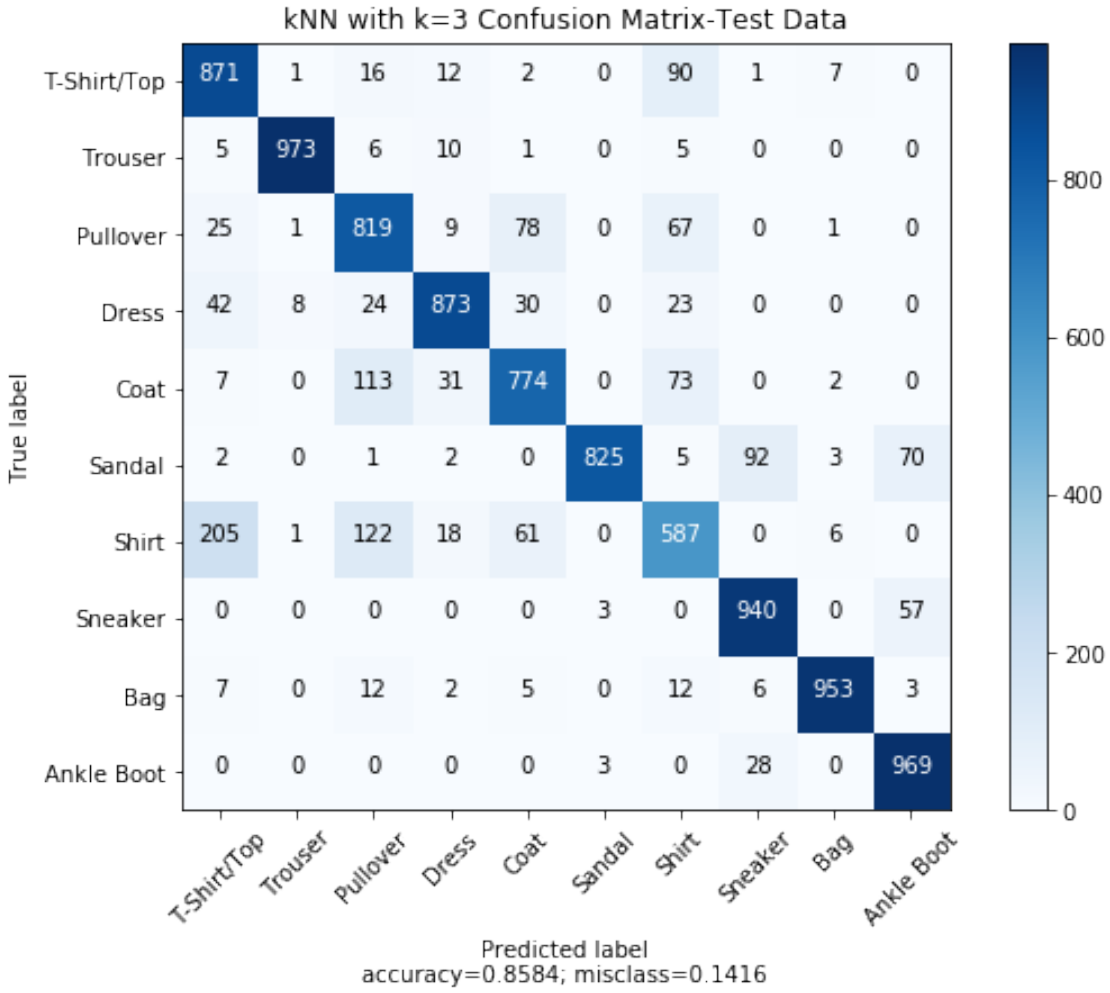
1.2.3 Precision, Recall and F1-score at k = 3 and test data

	precision	recall	f1-score	support
T-Shirt/Top	0.75	0.87	0.80	1000
Trouser	0.99	0.97	0.98	1000
Pullover	0.74	0.82	0.78	1000
Dress	0.91	0.87	0.89	1000
Coat	0.81	0.77	0.79	1000
Sandal	0.99	0.82	0.90	1000
Shirt	0.68	0.59	0.63	1000
Sneaker	0.88	0.94	0.91	1000
Bag	0.98	0.95	0.97	1000
Ankle Boot	0.88	0.97	0.92	1000
avg / total	0.86	0.86	0.86	10000

1.2.4 Confusion Matrix of training data @ k= 3



1.2.5 Confusion Matrix of test data @ k=3



1.3 Results of kNN Classifier with k = 5

1.3.1 Accuracy when k = 5

```
Accuracy of the system on train dataset (10000 samples) when k=5 is 0.904
Accuracy of the system on test dataset  when k=5 is 0.8589
```

1.3.2 Precision, Recall and F1-score at k = 5 and training data

```
Train Data Classification Report with k = 5

              precision    recall  f1-score   support

T-Shirt/Top       0.84      0.93      0.88      1009
```

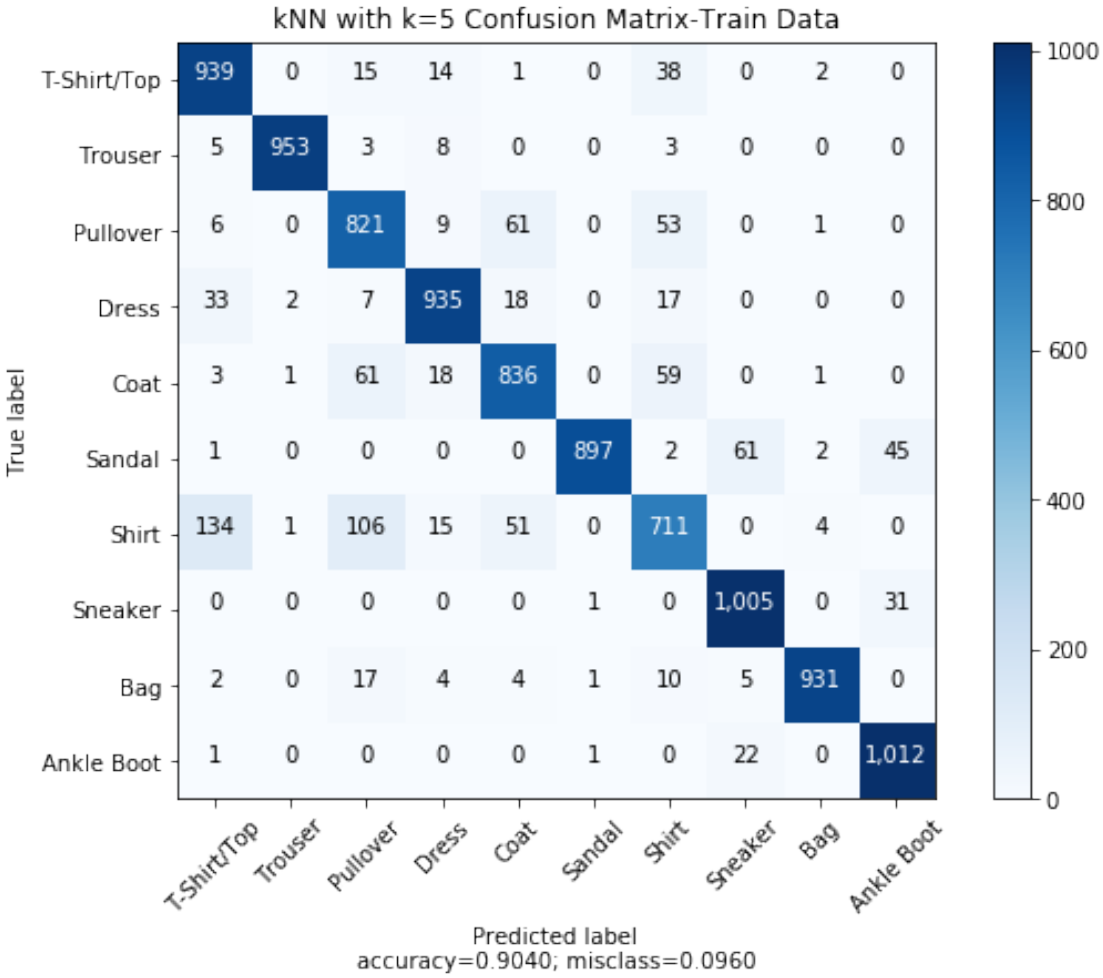
Trouser	1.00	0.98	0.99	972
Pullover	0.80	0.86	0.83	951
Dress	0.93	0.92	0.93	1012
Coat	0.86	0.85	0.86	979
Sandal	1.00	0.89	0.94	1008
Shirt	0.80	0.70	0.74	1022
Sneaker	0.92	0.97	0.94	1037
Bag	0.99	0.96	0.97	974
Ankle Boot	0.93	0.98	0.95	1036
avg / total	0.91	0.90	0.90	10000

1.3.3 Precision, Recall and F1-score at k = 5 and test data

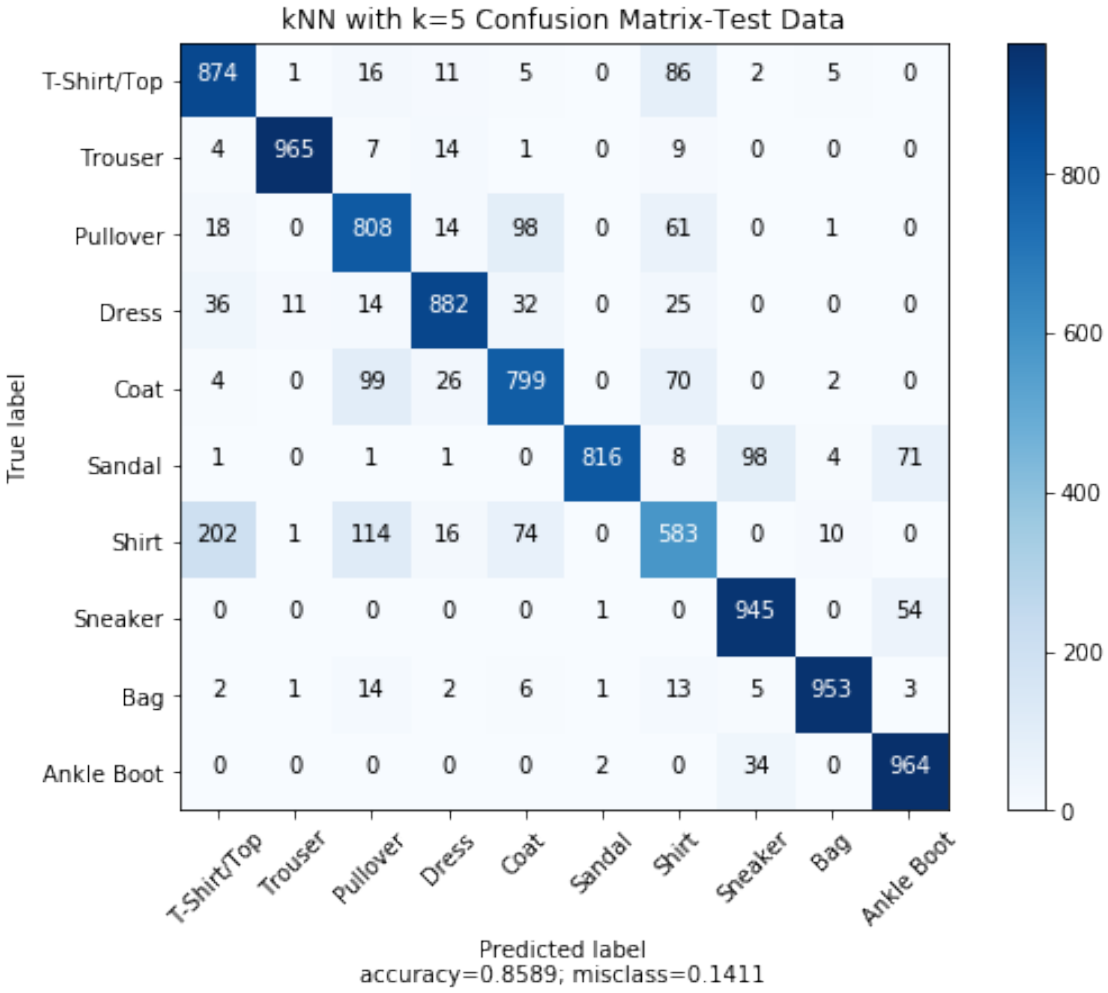
Test Data Classification Report with k = 5

	precision	recall	f1-score	support
T-Shirt/Top	0.77	0.87	0.82	1000
Trouser	0.99	0.96	0.98	1000
Pullover	0.75	0.81	0.78	1000
Dress	0.91	0.88	0.90	1000
Coat	0.79	0.80	0.79	1000
Sandal	1.00	0.82	0.90	1000
Shirt	0.68	0.58	0.63	1000
Sneaker	0.87	0.94	0.91	1000
Bag	0.98	0.95	0.97	1000
Ankle Boot	0.88	0.96	0.92	1000
avg / total	0.86	0.86	0.86	10000

1.3.4 Confusion Matrix of train data @ k=5



1.3.5 Confusion Matrix of test data @ k=5



1.4 Results of kNN Classifier with k = 7

1.4.1 Accuracy when k = 7

```
Accuracy of the system on train dataset (10000 samples) when k=7 is 0.8912
Accuracy of the system on test dataset when k=7 is 0.8558
```

1.4.2 Precision, Recall and F1-score at k = 7 and train data

```
Train Data Classification Report with k = 7

              precision    recall  f1-score   support

T-Shirt/Top       0.82      0.92      0.86      1009
```

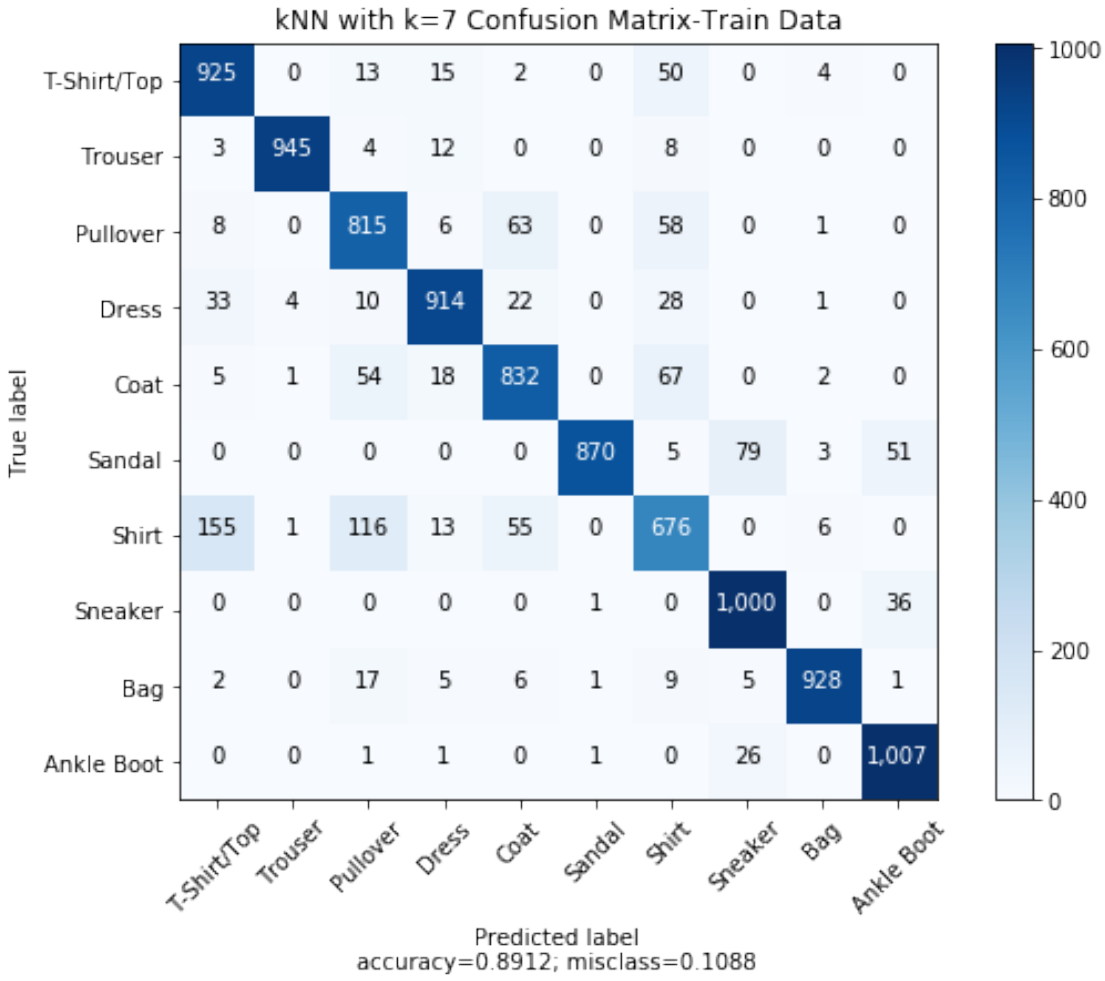
Trouser	0.99	0.97	0.98	972
Pullover	0.79	0.86	0.82	951
Dress	0.93	0.90	0.92	1012
Coat	0.85	0.85	0.85	979
Sandal	1.00	0.86	0.93	1008
Shirt	0.75	0.66	0.70	1022
Sneaker	0.90	0.96	0.93	1037
Bag	0.98	0.95	0.97	974
Ankle Boot	0.92	0.97	0.95	1036
avg / total	0.89	0.89	0.89	10000

1.4.3 Precision, Recall and F1-score at k =7 and test data

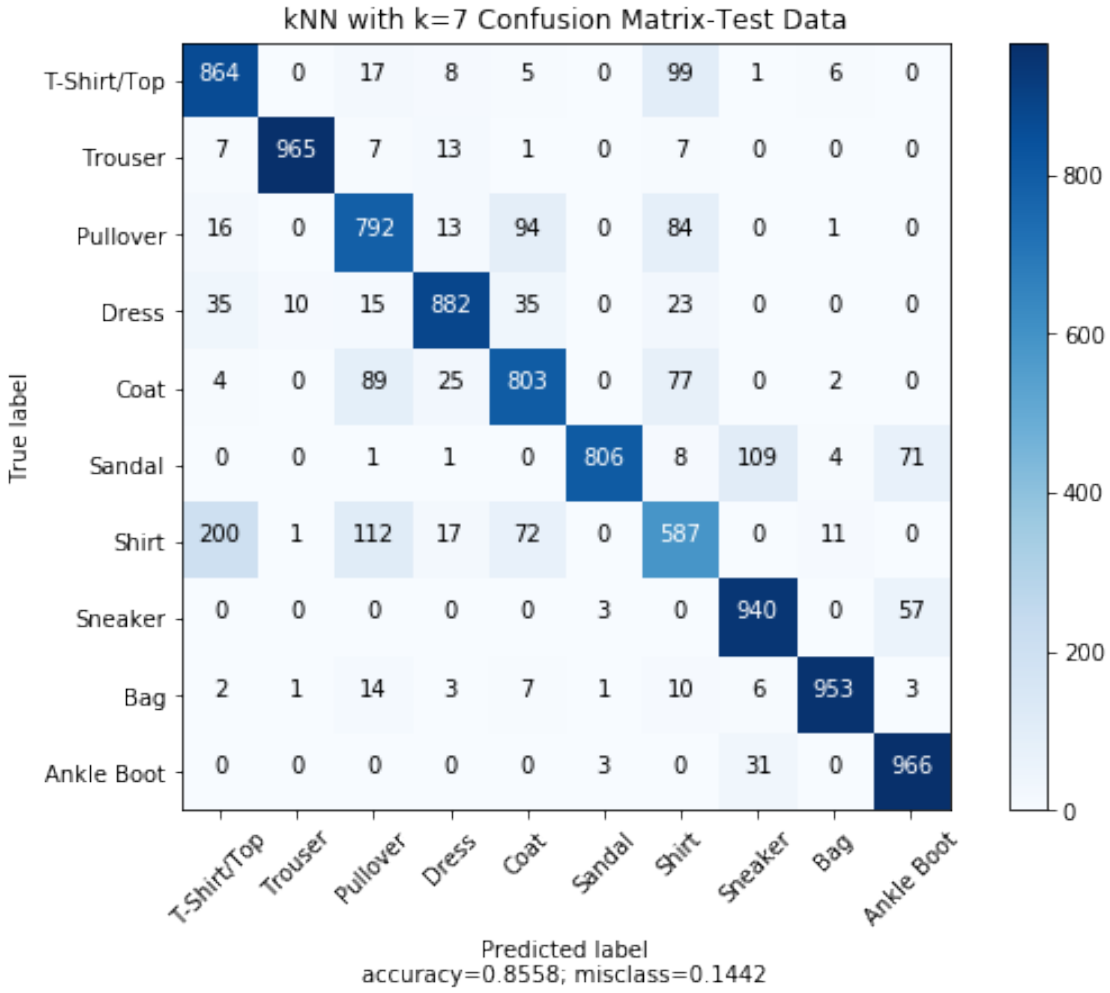
Test Data Classification Report with k = 7

	precision	recall	f1-score	support
T-Shirt/Top	0.77	0.86	0.81	1000
Trouser	0.99	0.96	0.98	1000
Pullover	0.76	0.79	0.77	1000
Dress	0.92	0.88	0.90	1000
Coat	0.79	0.80	0.80	1000
Sandal	0.99	0.81	0.89	1000
Shirt	0.66	0.59	0.62	1000
Sneaker	0.86	0.94	0.90	1000
Bag	0.98	0.95	0.96	1000
Ankle Boot	0.88	0.97	0.92	1000
avg / total	0.86	0.86	0.86	10000

1.4.4 Confusion Matrix of training data @ k=7



1.4.5 Confusion Matrix of test data @ k=7



1.5 Analysis for K Nearest Neighbour Classifier

- Accuracy
  - Test accuracy for all three K values are similar which falls around 86% approximately.
  - Training accuracy (92%) for K=3 is highest than other K values. One reason could be as in measuring the top K values, the sample itself contributes as one vote and makes decision biased towards it.
  - Training accuracy is measured on 10,000 trainig samples out of 60,000 samples.
    - The reason is that kNN algorithm measures similarity/distance with all other samples. For example, distance/similarity of 1 training sample measured against 60,000 samples and top k will predict the label of that training sample.
    - 784 features in each sample means comparison of 784 values with each of the 60,000 training samples and that makes it very computation expensive.

    - We tried with different sample size of training dataset and the accuracy falls around the same values.
  - Test accuracy is measured against all the 10,000 samples.

- F-1, Precsion and Recall measures
  - Precision(P), Recall(R) and F1 measure are other methods to measure the efficiency of the classifier.
  - Tables show seperate P, R and F1 values for each label.
  - Looking at the F1 measure we can see that most 5 items have F1 score more than 0.9.
  - "Shirt" object has the least F1 score which falls under 70% in test cases while for training samples its F1 score is around 77%
    - The reason we see is that "Shirt" would be very similar to some other object in the vector space model. kNN based model cannot identify the difference between shirt and other objects and classifies it wrongly or classifies other object as "shirt"

- Confusion Matrix
  - Confusion Matrix is another way to show the relation between the PREDICTED CLASS and the TRUE CLASS
  - Value in the diagonals are "true positive" values.
  - False Negative means TRUE CLASS predicted as WRONG CLASS. We can measure it using CONFUSION MATRIX.
    - Add all the values in a row.
    - Subtract "true positive" value from it.
    - Result would be "false negative" and is used to measure Recall
  - False Postive means WRONG CLASS predicted as TRUE CLASS. We can measure it using CONFUSION MATRIX.
    - Add all the values in a column.
    - Subtract "true positive" value from it.
    - Result would be "false positive" and is used to measure Precision.
  - The more values in the diagonal, the better is our model.
  - Confusion matrix displays that object "shirt" is the most object that was predicted as wrong class. It is mostly confused with "T-shirt" or "Pullover" and we can visualize it in Confusion Matrix.

- Limitations observed in kNN
  - kNN was very time consuming and memory intensive. RAM and Hard Disk utilization touched 100%. Accuracy of 10,000 training samples was tested instead of 60,000.
  - kNN accuracy on unseen images was around 86% for different values of k. It works in vector space model and finds similarities before using the top K results. It computes similarites with all other samples every time it is executed.
  - It cannot find the hidden patterns in the images as it works on vector space model.
  - Therefore in next sections we will use better and stronger models for image classification.

2 Single Layer Neural Network

During the training phase, the 20 epochs each of training were run using different combinations of optimizer and the loss function to see which one performed best. The same optimizer/loss function was chosen for the other classifiers as well.

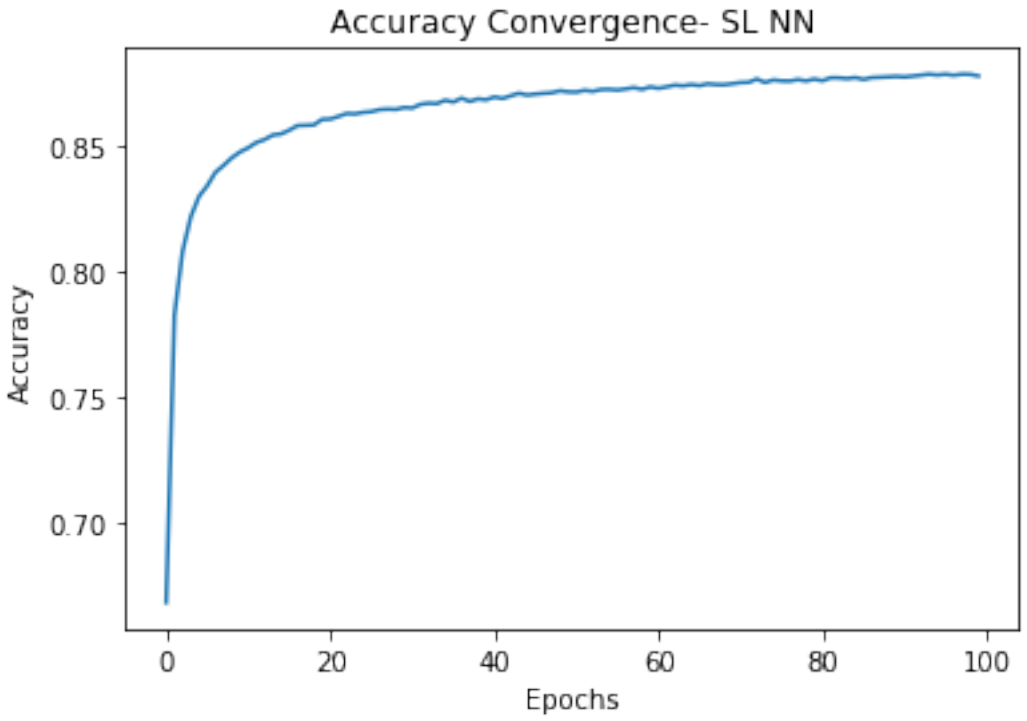
2.1 Hyper Parameters

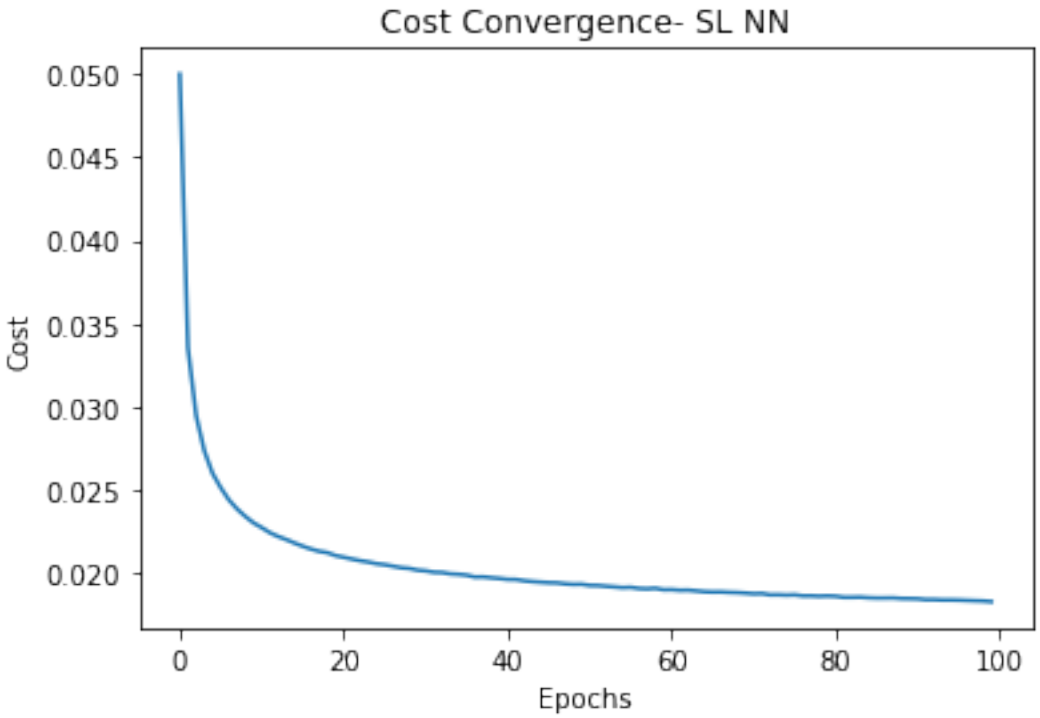
- No of Layers: 1
- Details of Layers
  - Layer 1
    - No of Neurons: 10
    - Activation: Softmax
- Epochs: 100
- Optimizer: RMS Prop
- Loss Function: MSE

2.2 Results

The results of training and subsequent analysis are as follows:

2.2.1 Accuracy and Loss Convergence






```
Loss:  0.020196859022974967
Accuracy:  0.8652
```

2.2.2 Training Data Classification Report

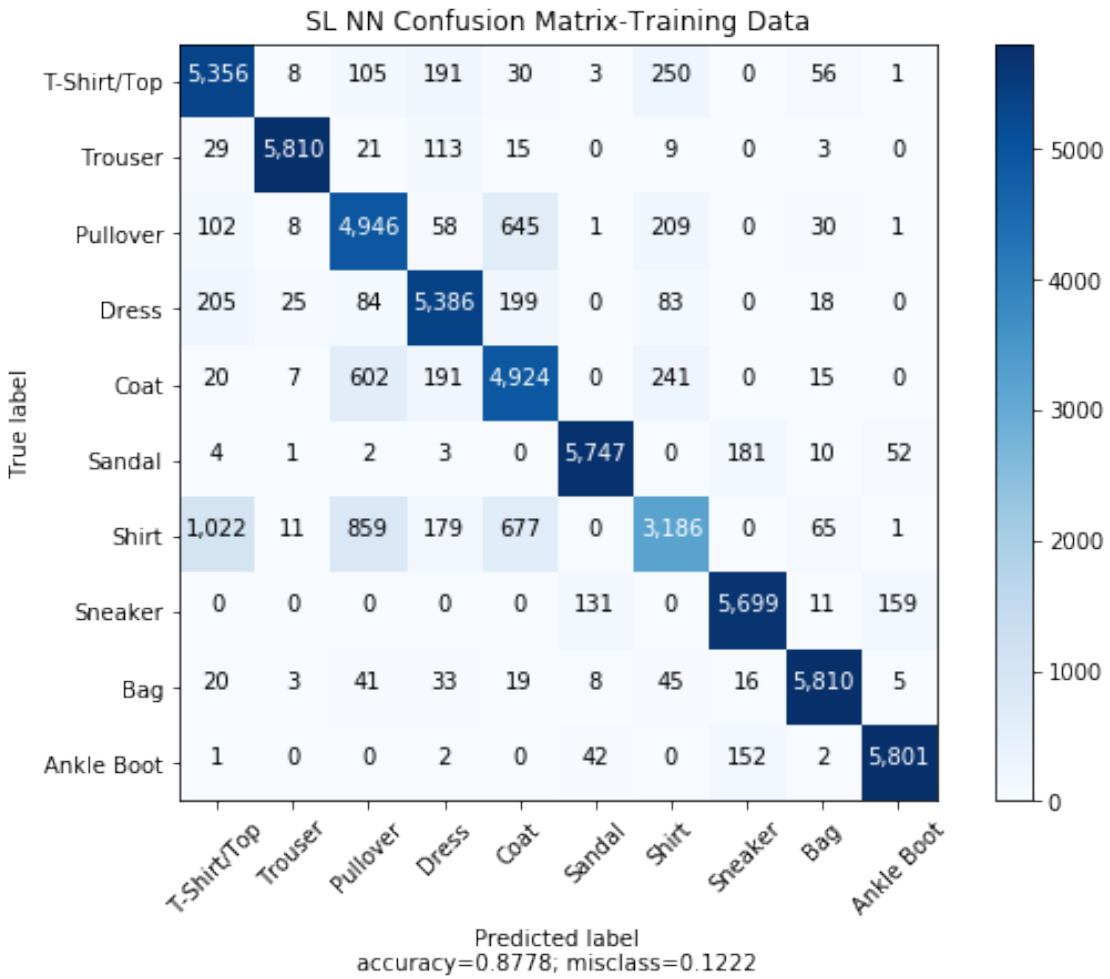
	precision	recall	f1-score	support
T-Shirt/Top	0.79	0.89	0.84	6000
Trouser	0.99	0.97	0.98	6000
Pullover	0.74	0.82	0.78	6000
Dress	0.87	0.90	0.89	6000
Coat	0.76	0.82	0.79	6000
Sandal	0.97	0.96	0.96	6000
Shirt	0.79	0.53	0.64	6000
Sneaker	0.94	0.95	0.95	6000
Bag	0.97	0.97	0.97	6000
Ankle Boot	0.96	0.97	0.97	6000
avg / total	0.88	0.88	0.88	60000

2.2.3 Test Data Classification Report

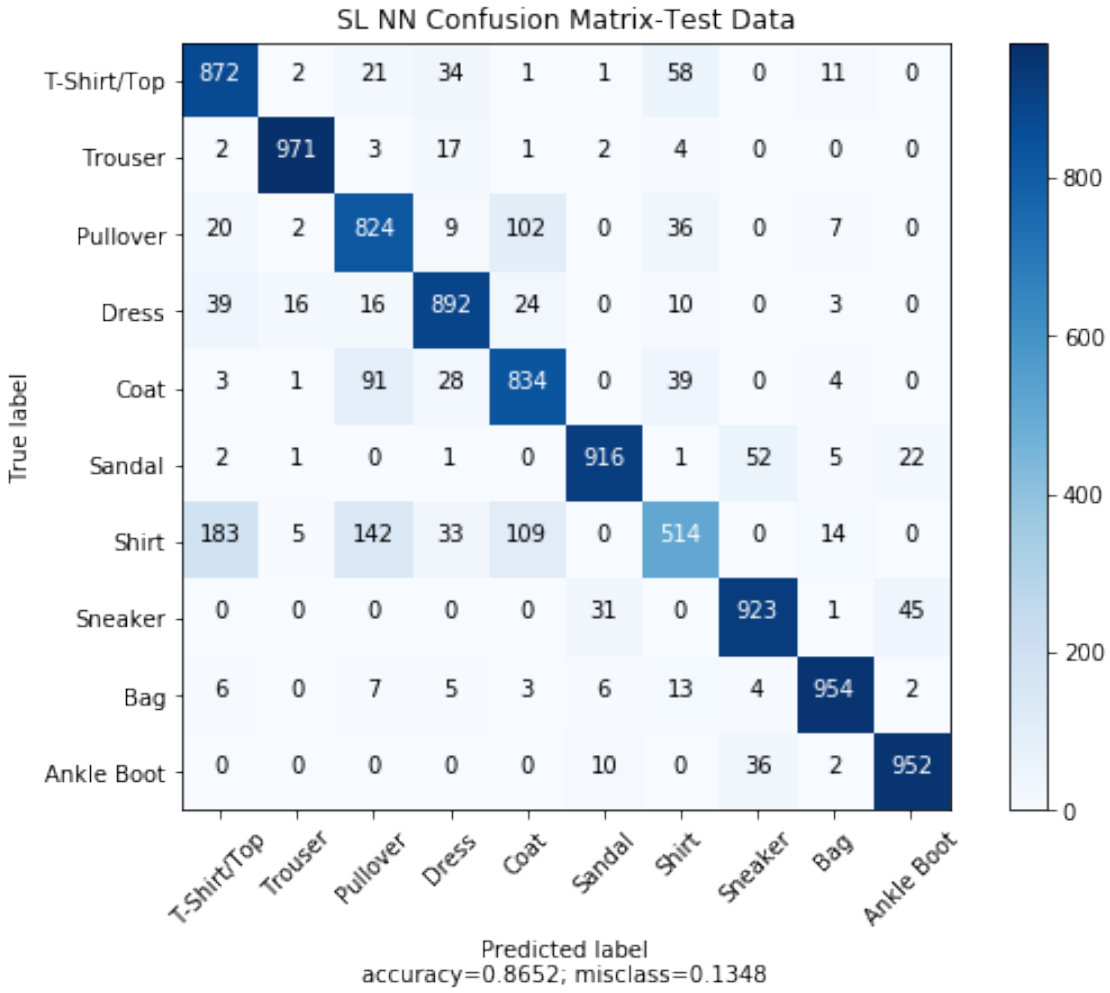
	precision	recall	f1-score	support

T-Shirt/Top	0.77	0.87	0.82	1000
Trouser	0.97	0.97	0.97	1000
Pullover	0.75	0.82	0.78	1000
Dress	0.88	0.89	0.88	1000
Coat	0.78	0.83	0.80	1000
Sandal	0.95	0.92	0.93	1000
Shirt	0.76	0.51	0.61	1000
Sneaker	0.91	0.92	0.92	1000
Bag	0.95	0.95	0.95	1000
Ankle Boot	0.93	0.95	0.94	1000
avg / total	0.86	0.87	0.86	10000

2.2.4 Training Data Confusion Matrix



2.2.5 Test Data Confusion Matrix



2.3 Analysis of Results

An initial run showed a very poor accuracy of close to 30% on the dataset. However the accuracy rose to 88% (on the training set) when normalization was applied to each pixel value by divising each of them by their max value (255). The results above show that both the loss and accuracy are converging nicely and accuracy on the test set is also a very healthy 87% which shows that the model has been trained adequately and there is no overfitting. A closer look at the confusion matrix shows that although the overall classification is good, there are some misclassifications between the shirt/t-shirt, shirt/pullover, shirt/coat and coat/pullover pairs. This according to our intuition is because a Single Layer Network is essetially a Logistic Regression applied to each class separaltely and then the result combined to output the class with the maximum probability. A shallow neural network such as this is not able to distinguish between more complex features and since the general shape of the above mentioned cloth pairs is the same (similar pixel values occur at similar locations in the image), the classifier confuses one class for another similar class.

3 Multilayer Perceptron

The no. of layers, the number of neurons within them were found by hot and trial, searching on the internet for similar solutions, etc, trying them out and finding the best combination possible within the limited time available.

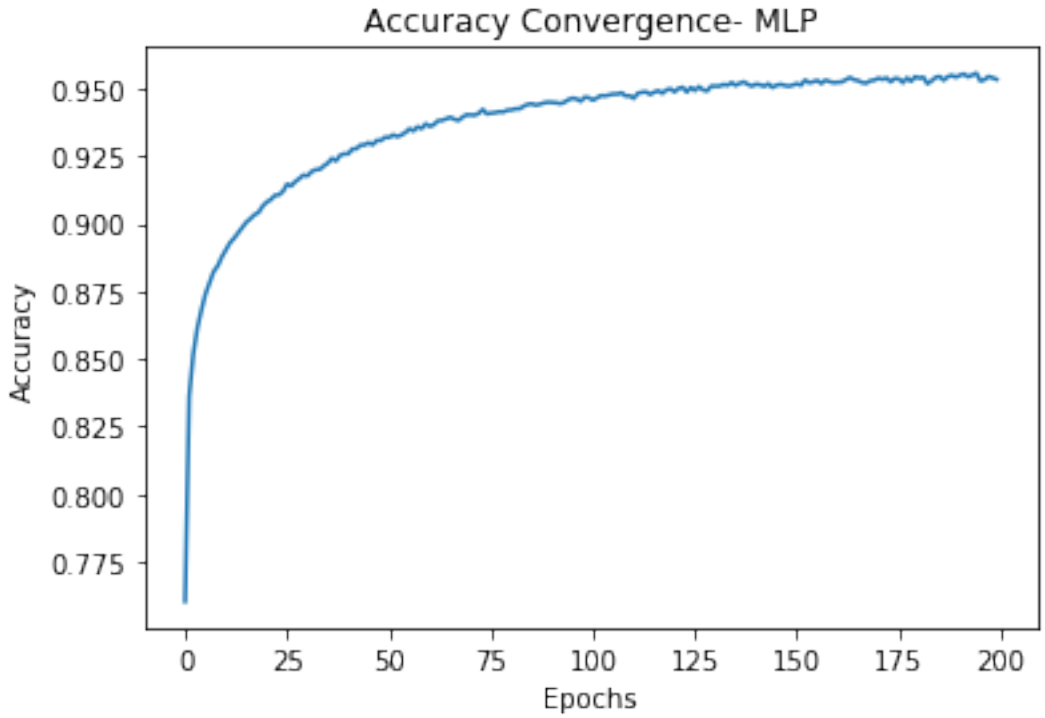
3.1 Hyper Parameters

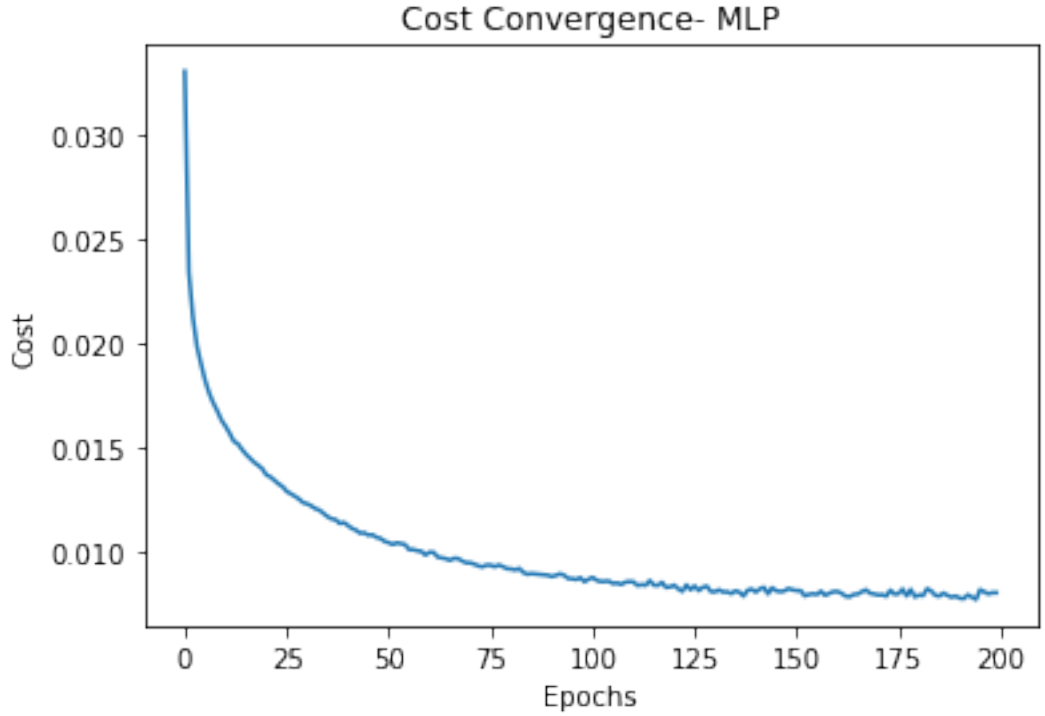
- No of Layers: 5
- Details of Layers
  - Layer 1
    - Layer Type: Dense
    - No of Neurons: 32
    - Activation: ReLU
  - Layer 2
    - Layer Type: Dense
    - No of Neurons: 48
    - Activation: ReLU
  - Layer 3
    - Layer Type: Dense
    - No of Neurons: 72
    - Activation: ReLU
  - Layer 4
    - Layer Type: Dense
    - No of Neurons: 108
    - Activation: ReLU
  - Layer 5
    - Layer Type: Dense
    - No of Neurons: 10
    - Activation: Softmax
- Epochs: 100
- Optimizer: RMS Prop
- Loss Function: MSE

3.2 Results

The results of training and subsequent analysis are as follows:

3.2.1 Accuracy and Loss Convergence






```
Loss:  0.0228513350635767
```

```
Accuracy:  0.876
```

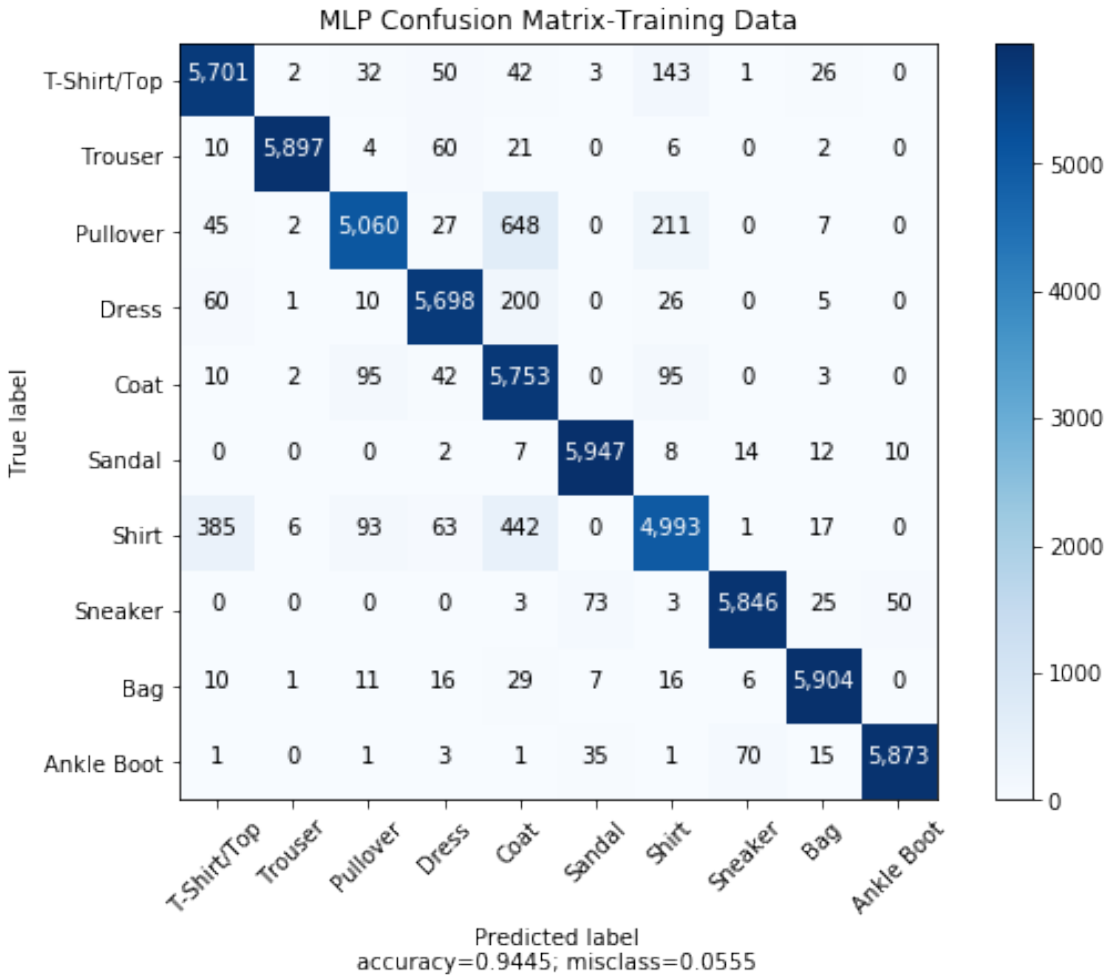
3.2.2 Training Data Classification Report

	precision	recall	f1-score	support
T-Shirt/Top	0.92	0.95	0.93	6000
Trouser	1.00	0.98	0.99	6000
Pullover	0.95	0.84	0.90	6000
Dress	0.96	0.95	0.95	6000
Coat	0.81	0.96	0.88	6000
Sandal	0.98	0.99	0.99	6000
Shirt	0.91	0.83	0.87	6000
Sneaker	0.98	0.97	0.98	6000
Bag	0.98	0.98	0.98	6000
Ankle Boot	0.99	0.98	0.98	6000
avg / total	0.95	0.94	0.94	60000

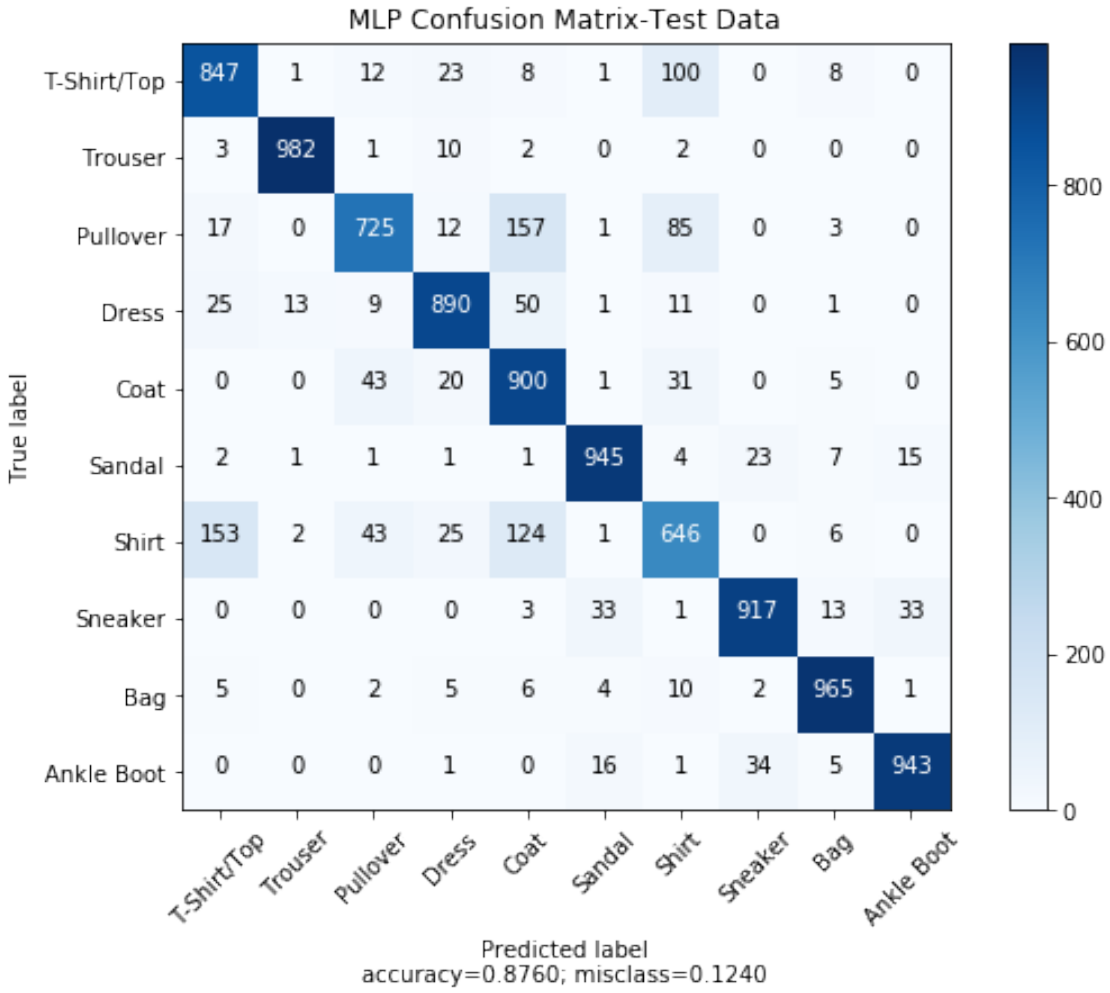
3.2.3 Test Data Classification Report

	precision	recall	f1-score	support
T-Shirt/Top	0.81	0.85	0.83	1000
Trouser	0.98	0.98	0.98	1000
Pullover	0.87	0.72	0.79	1000
Dress	0.90	0.89	0.90	1000
Coat	0.72	0.90	0.80	1000
Sandal	0.94	0.94	0.94	1000
Shirt	0.73	0.65	0.68	1000
Sneaker	0.94	0.92	0.93	1000
Bag	0.95	0.96	0.96	1000
Ankle Boot	0.95	0.94	0.95	1000
avg / total	0.88	0.88	0.88	10000

3.2.4 Training Data Confusion Matrix



3.2.5 Test Data Confusion Matrix



3.3 Analysis of Results

The results above show that both the loss and accuracy are converging nicely and accuracy on the test set is also a very healthy 88% which shows that the model has been trained adequately and there is no overfitting. This is slightly better than the 86% accuracy from the Single Layer Neural Network. Our intuition was that a MLP can better classify the image set because it can infer more complex relationships between the input images and the class labels. However it seems it has not done so. A closer look at the confusion matrix shows that similar to the Single Layer NN, there are some misclassifications between the shirt/t-shirt, shirt/pullover, shirt/coat and coat/pullover pairs, but are generally slightly better. Again it seems the classifier confuses one class for another similar class as it is only considering values at their flattened pixel positions and not any pattern or artifact within them.

4 Convolutional Neural Network

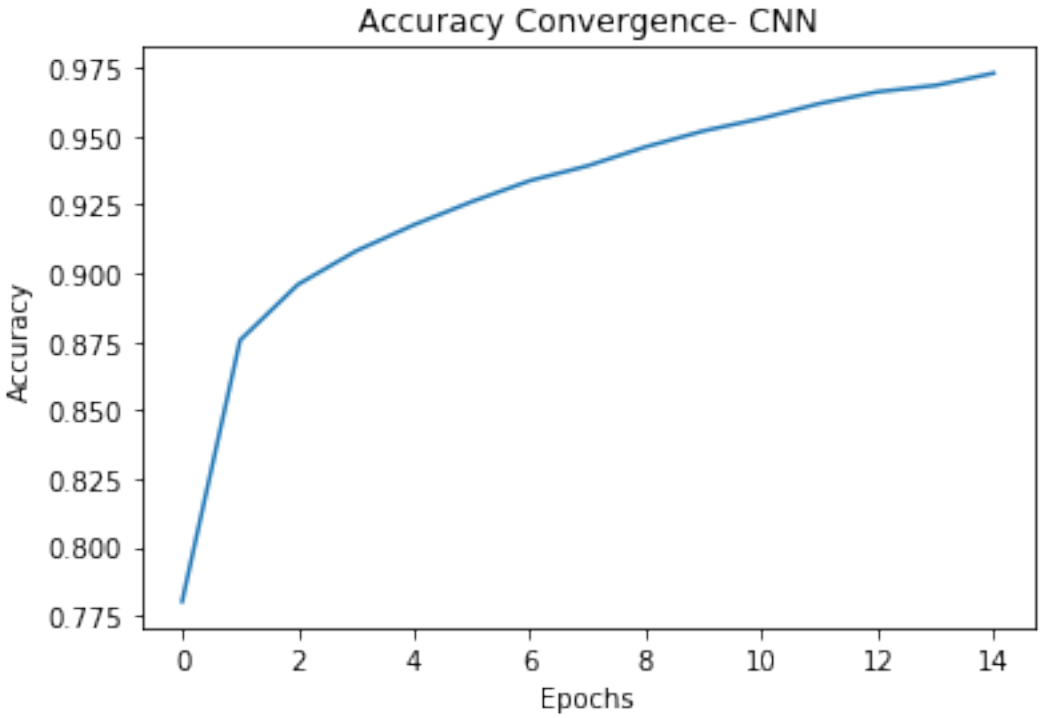
4.1 Hyper Parameters

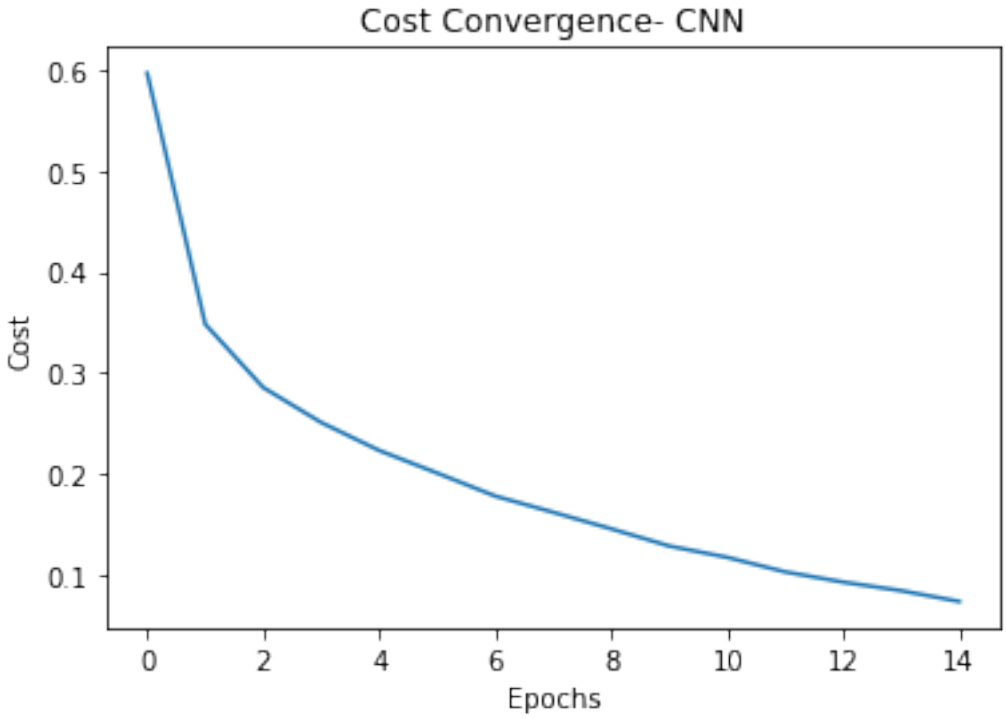
- No of Layers: 6
- Details of Layers
  - Layer 1
    - Layer Type: Conv2D
    - Filter Size: 5x5
    - Activation: ReLU
  - Layer 2
    - Layer Type: MaxPooling
    - Filter Size: 2x2
  - Layer 3
    - Layer Type: Conv2D
    - Filter Size: 5x5
    - Activation: ReLU
  - Layer 4
    - Layer Type: MaxPooling
    - Filter Size: 2x2
  - Layer 5
    - Layer Type: Dense
    - No of Neurons: 1000
    - Activation: ReLU
  - Layer 6
    - Layer Type: Dense
    - No of Neurons: 10
    - Activation: Softmax
- Epochs: 15
- Optimizer: RMS Prop
- Loss Function: MSE

4.2 Results

The results of training and subsequent analysis are as follows:

4.2.1 Accuracy and Loss Convergence





```
Loss:  0.3328172839641571
Accuracy:  0.9194
```

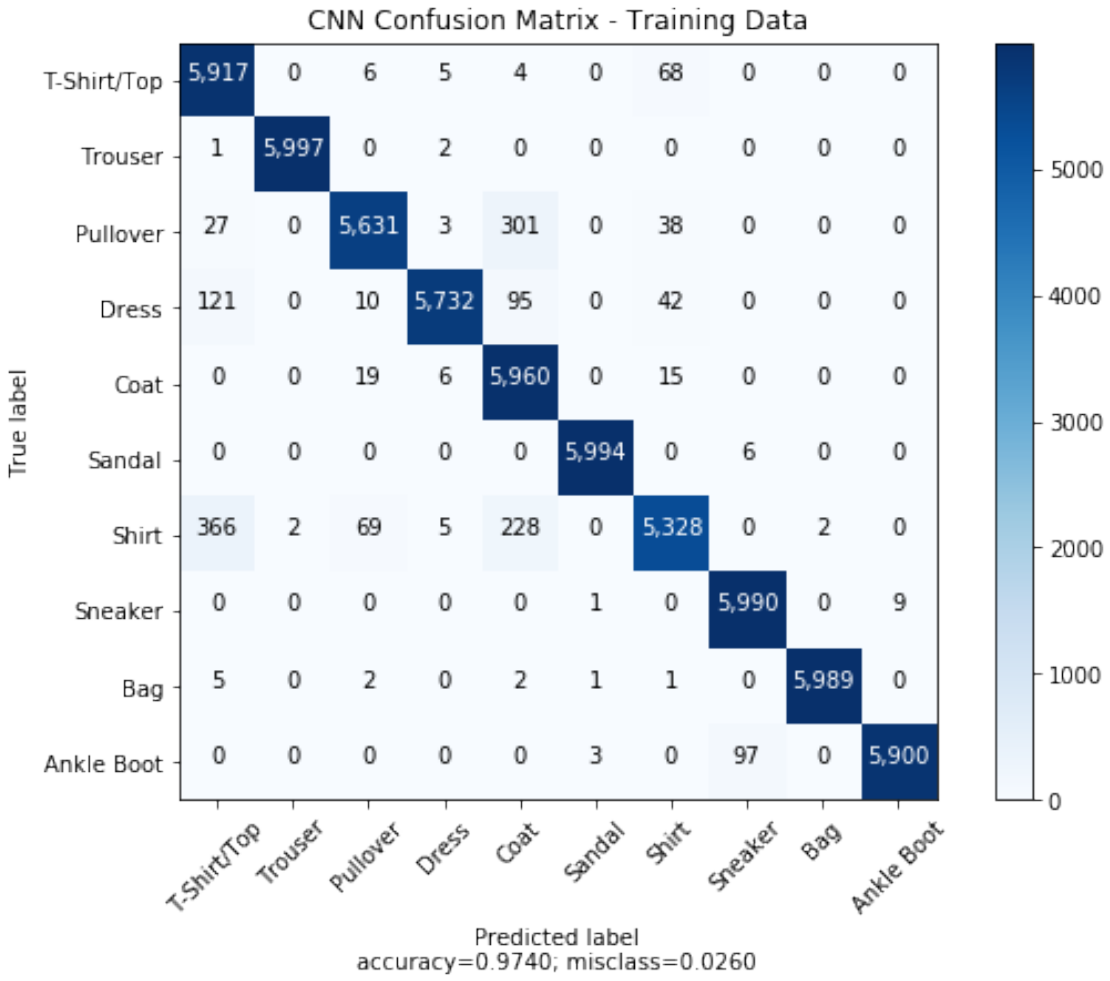
4.2.2 Training Data Classification Report

	precision	recall	f1-score	support
T-Shirt/Top	0.92	0.99	0.95	6000
Trouser	1.00	1.00	1.00	6000
Pullover	0.98	0.94	0.96	6000
Dress	1.00	0.96	0.98	6000
Coat	0.90	0.99	0.95	6000
Sandal	1.00	1.00	1.00	6000
Shirt	0.97	0.89	0.93	6000
Sneaker	0.98	1.00	0.99	6000
Bag	1.00	1.00	1.00	6000
Ankle Boot	1.00	0.98	0.99	6000
avg / total	0.98	0.97	0.97	60000

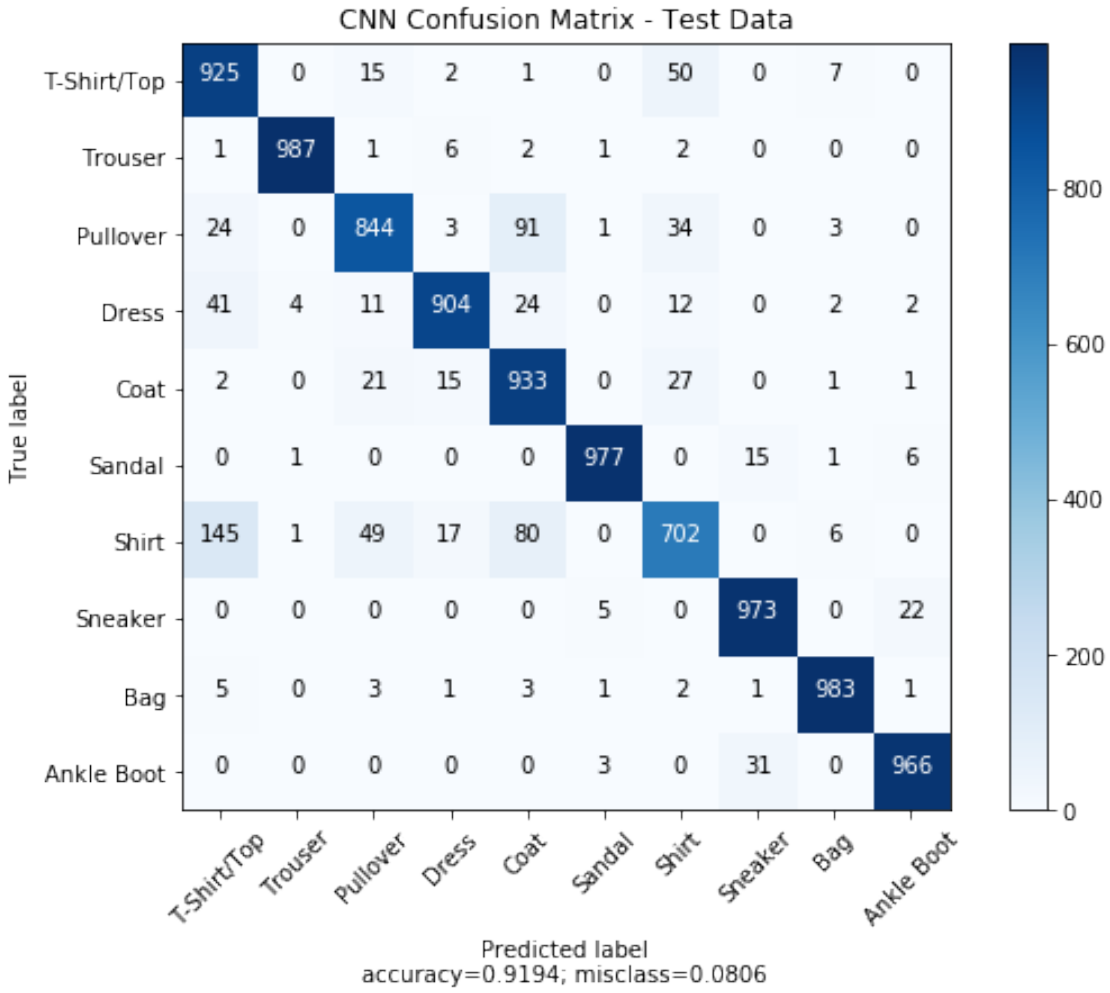
4.2.3 Test Data Classification Report

	precision	recall	f1-score	support
T-Shirt/Top	0.81	0.93	0.86	1000
Trouser	0.99	0.99	0.99	1000
Pullover	0.89	0.84	0.87	1000
Dress	0.95	0.90	0.93	1000
Coat	0.82	0.93	0.87	1000
Sandal	0.99	0.98	0.98	1000
Shirt	0.85	0.70	0.77	1000
Sneaker	0.95	0.97	0.96	1000
Bag	0.98	0.98	0.98	1000
Ankle Boot	0.97	0.97	0.97	1000
avg / total	0.92	0.92	0.92	10000

4.2.4 Training Data Confusion Matrix



4.2.5 Test Data Confusion Matrix



4.3 Analysis of Results

The results above show that both the loss and accuracy are converging nicely and accuracy on the test set is the best of all (92%). Also the model has generalized adequately and there is no overfitting. The results are much better than all the others because unlike the SLNN and MLP, CNN is adept at capturing small shapes and artifacts from within an image. In a deep CNN, it does so incrementally with the initial layers learning and detecting basic features/shapes and the later layers learning/detecting more complex artifacts made up of those basic features/shapes. A look at the confusion matrix shows that there are still some misclassifications between the shirt/t-shirt pair but misclassifications between shirt/pullover, shirt/coat and coat/pullover pairs are reduced.

5 COMPARISON OF CLASSIFIERS

Comparison of Classifiers for the Fashion-MNIST Dataset						
	kNN(k=3)	kNN(k=5)	kNN(k=7)	SLNN	MLP	Conv
Accuracy/ Precision/ Recall/ F1-Score (Training Data)	92.14%	90.40%	89.12%	87.78%	94.45%	97.4%
Accuracy/ Precision/ Recall/ F1-Score (Test Data)	85.84%	85.89%	85.58%	86.52%	87.60%	91.94%

As the table shows the results of convolution neural network surpasses other classifiers. CNN is a very powerful classifier that can find hidden patterns in the image. The advantage can be seen however we also see that if we talk about individual classes then CNN could not perform well on "shirts" class.