EE6132: Deep Learning for Image Processing

Assignment 2: Convolutional Neural Networks

Adarsh B (MM14B001)

Contents

1 Question 1: MNIST classification using CNN

1.1 Baseline Model

1.1.1 Overall architecture

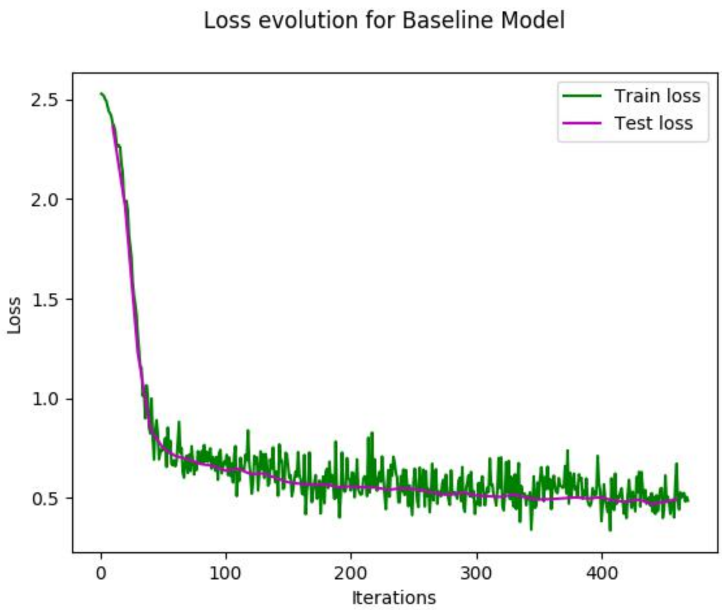
Input --→ Conv (32 3x3 filters, Stride of 1, Zero padding of 1) --→ 2x2 Maxpool (Stride of 2) --→ Fully connected (10 outputs) --→ Softmax classifier

Batch size	128
Regularization	L2 (for weights only)
Regularization parameter (λ)	0.01
No. of training epochs	5
Update algorithm	SGD with Momentum acceleration

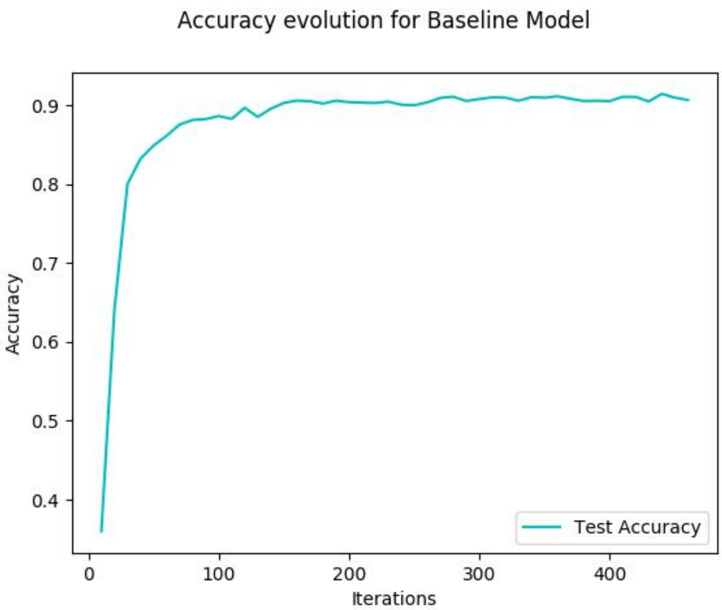
1.1.2 Results

The final test accuracy obtained for the model is **92.65%.**

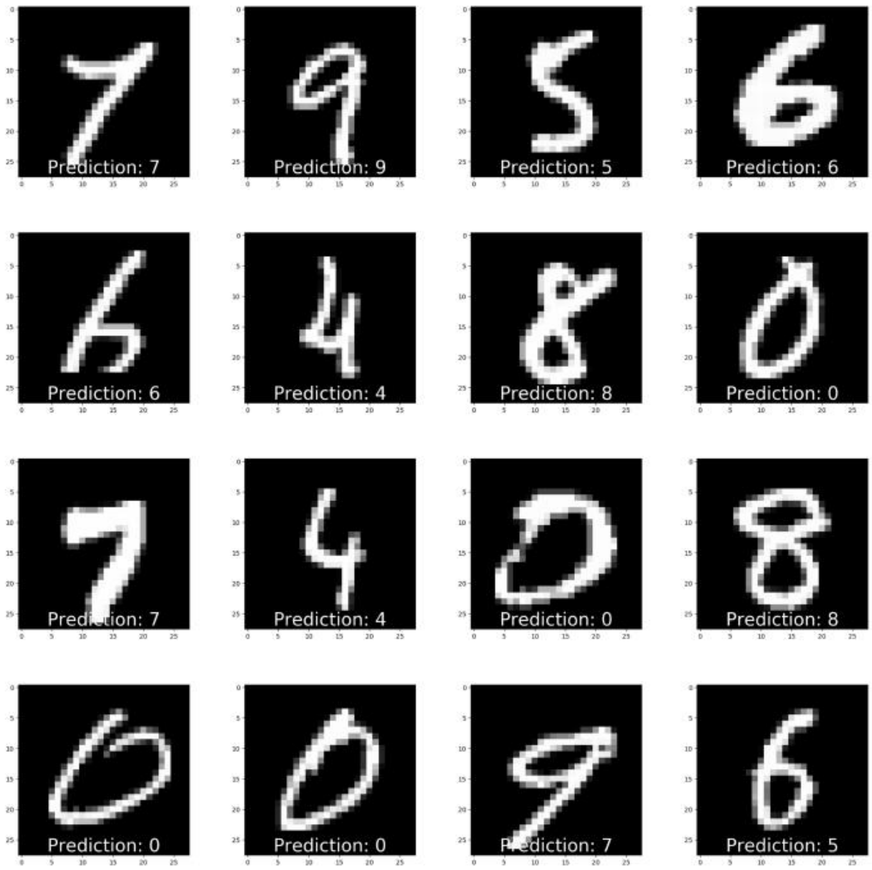
Following is the plot showing training and validation loss evolution over iterations:



Following is the plot showing validation accuracy evolution over iterations:



1.1.3 Sample predictions



As can be seen from the above predictions, baseline model could predict **14 out of 16 images** correctly.

1.2 2 convolutional layers

1.2.1 Overall architecture

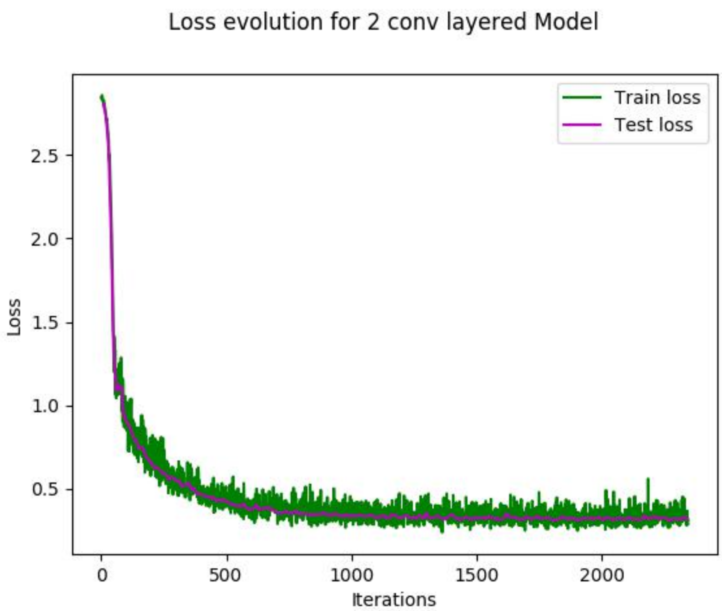
Input --→ Conv1 (32 3x3 filters, Stride of 1, Zero padding of 1) --→ 2x2 Maxpool (Stride of 2) --→ Conv2 (32 3x3 filters, Stride of 1, Zero padding of 1) --→ 2x2 Maxpool (Stride of 2) --→ Fully connected (10 outputs) --→ Softmax classifier

Batch size	128
Regularization	L2 (for weights only)
Regularization parameter (λ)	0.01
No. of training epochs	5
Update algorithm	SGD with Momentum acceleration

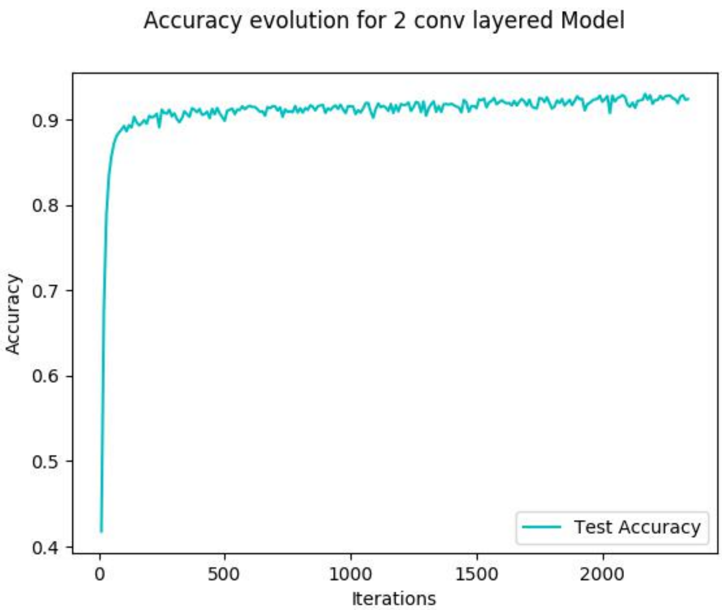
1.2.2 Results

The final test accuracy obtained for the model is **96.38%.**

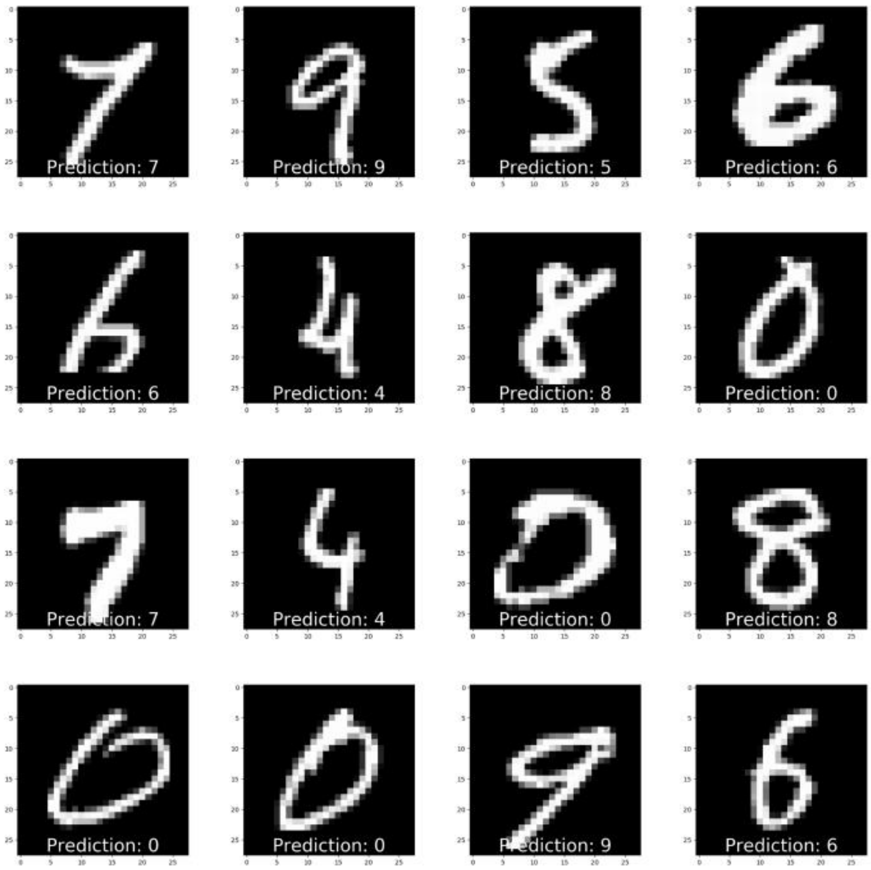
Following is the plot showing training and validation loss evolution over iterations:



Following is the plot showing validation accuracy evolution over iterations:



1.2.3 Sample predictions



As can be seen from the above predictions, baseline model could predict **16 out of 16 images** correctly.

1.3 2 convolutional layers + 1 hidden fully connected layer

1.3.1 Overall architecture

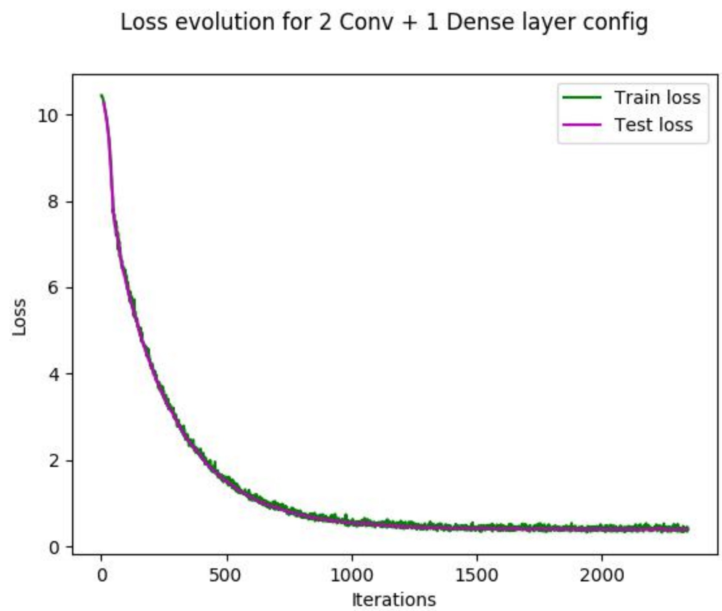
Input --→ Conv1 (32 3x3 filters, Stride of 1, Zero padding of 1) --→ 2x2 Maxpool (Stride of 2) --→ Conv2 (32 3x3 filters, Stride of 1, Zero padding of 1) --→ 2x2 Maxpool (Stride of 2) --→ Fully connected (500 outputs) --→ Fully connected (10 outputs) --→ Softmax classifier

Batch size	128
Regularization	L2 (for weights only)
Regularization parameter (λ)	0.01
No. of training epochs	5
Update algorithm	SGD with Momentum acceleration

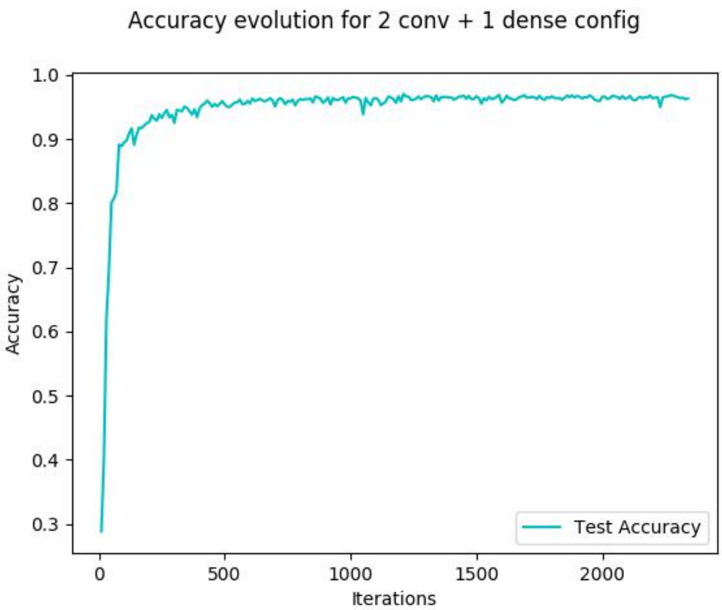
1.3.2 Results

The final test accuracy obtained for the model is **96.65%.**

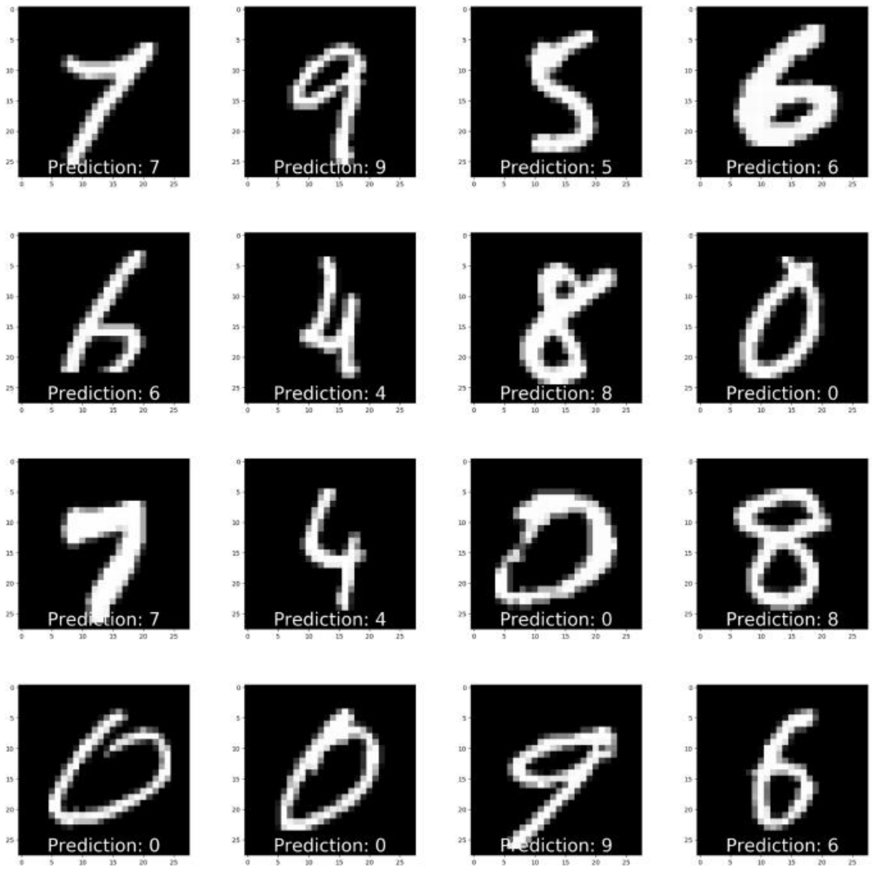
Following is the plot showing training and validation loss evolution over iterations:



Following is the plot showing validation accuracy evolution over iterations:



1.3.3 Sample predictions



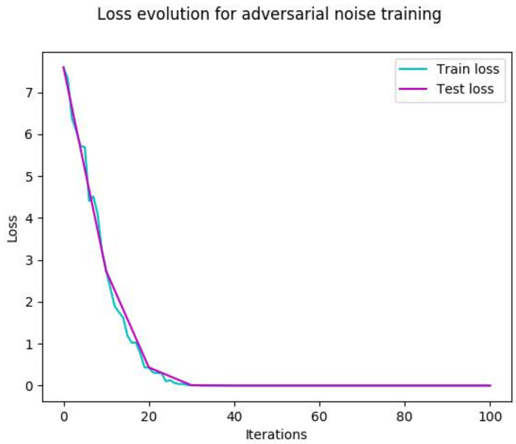
As can be seen from the above predictions, baseline model could predict **16 out of 16 images** correctly.

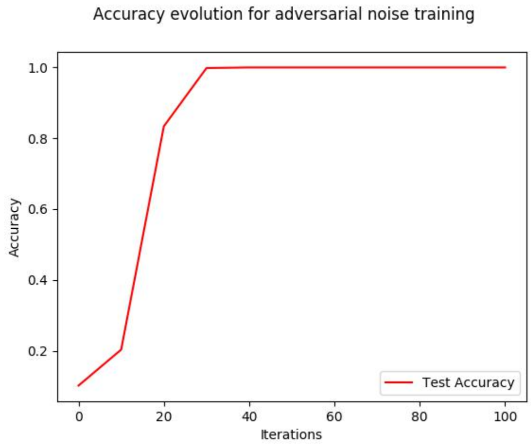
2 Generating Adversarial Examples

The best model obtained from the above examples is the model with **2 convolutional + 1 hidden dense** configuration. Hence we use a trained model of that particular configuration for the following experiments. For training the noise, we carry out 100 batch iterations because the loss seems to converge within 100 iterations itself. And in each of the 10 cases, prediction accuracy reaches ~**100%**.

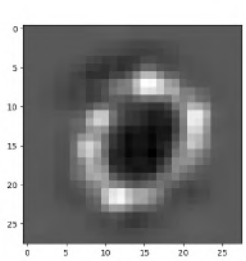
2.1 Adversarial Examples for label 0

Following are plots of loss evolution and accuracy evolution for adversarial noise training.

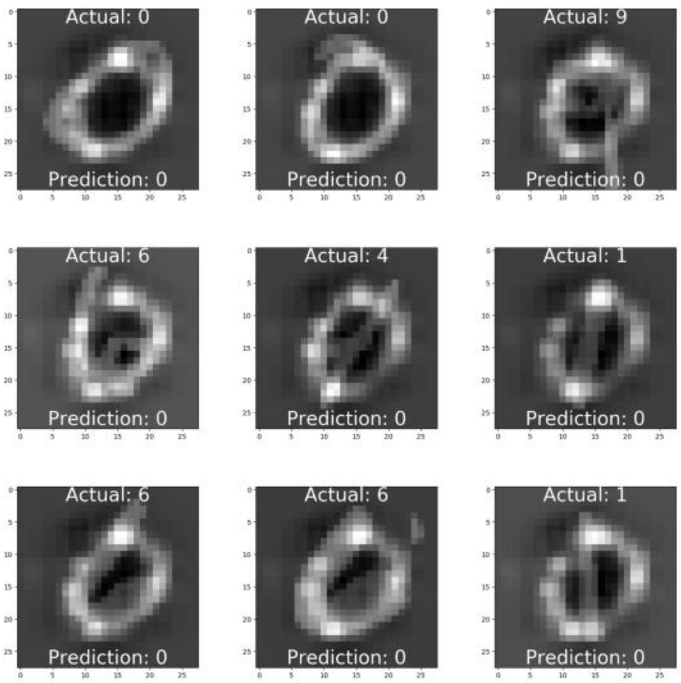




Following is the generated adversarial noise for 0:

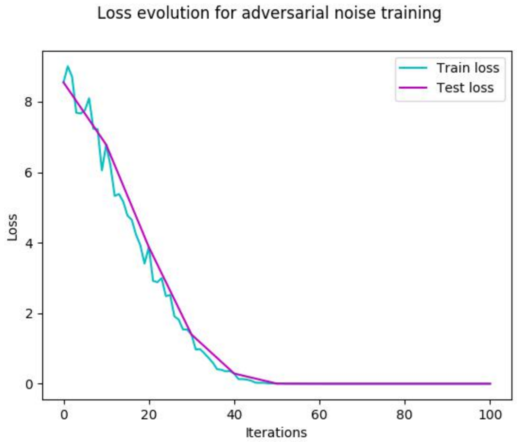


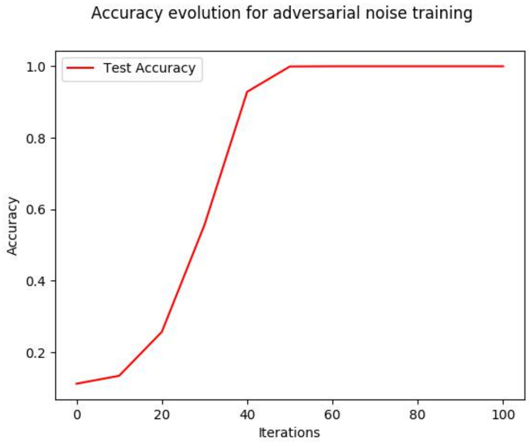
Following are the predictions for noise infused images:



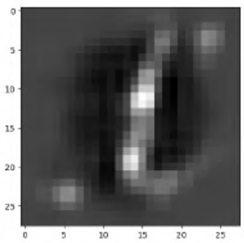
2.2 Adversarial Examples for label 1

Following are plots of loss evolution and accuracy evolution for adversarial noise training.

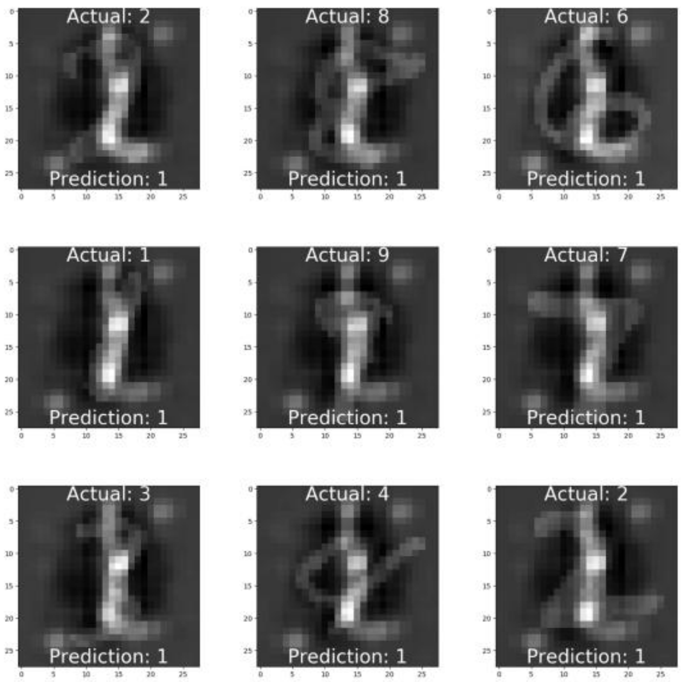




Following is the generated adversarial noise for 1:

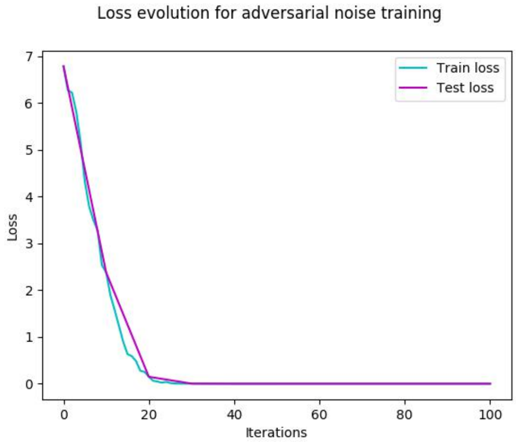


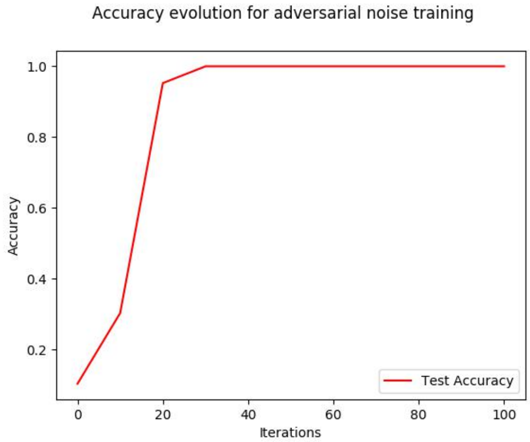
Following are the predictions for noise infused images:



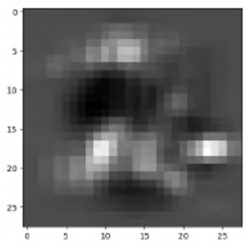
2.3 Adversarial Examples for label 2

Following are plots of loss evolution and accuracy evolution for adversarial noise training.

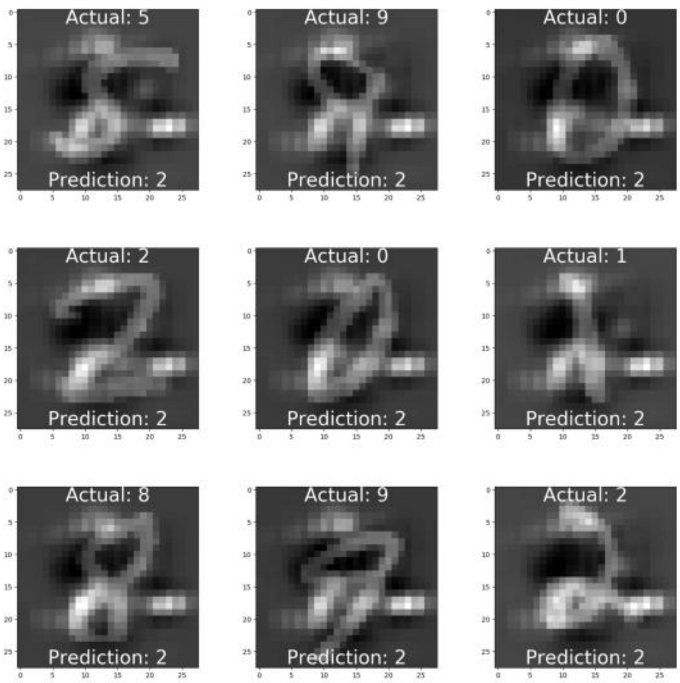




Following is the generated adversarial noise for 2:

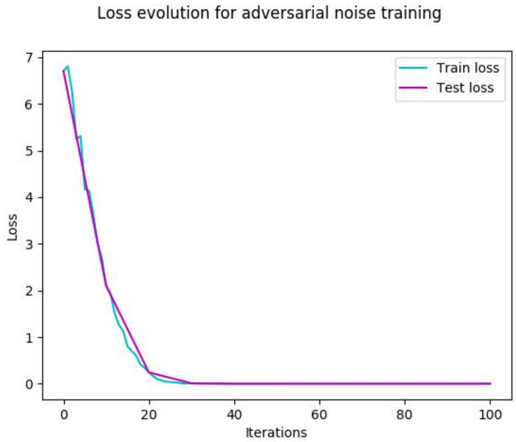


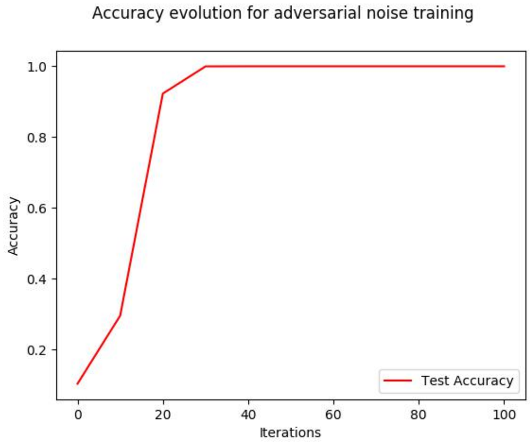
Following are the predictions for noise infused images:



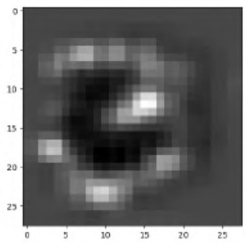
2.4 Adversarial Examples for label 3

Following are plots of loss evolution and accuracy evolution for adversarial noise training.

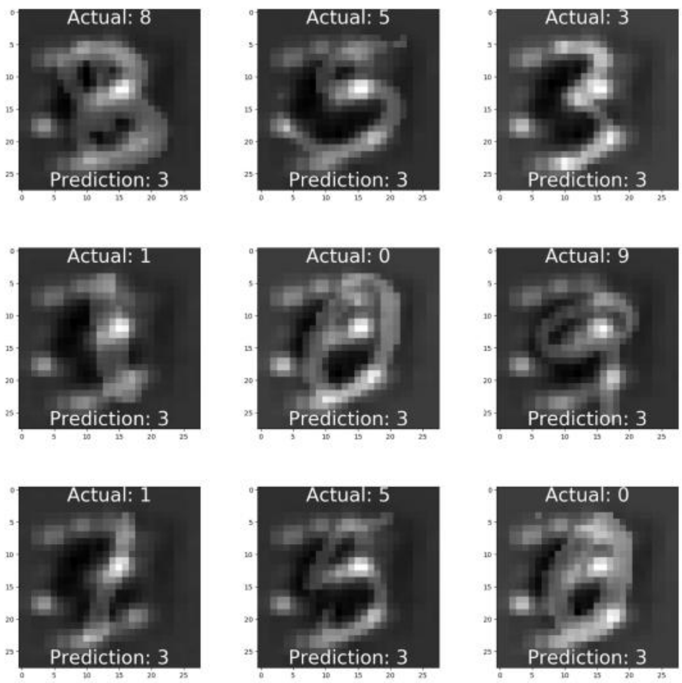




Following is the generated adversarial noise for 2:

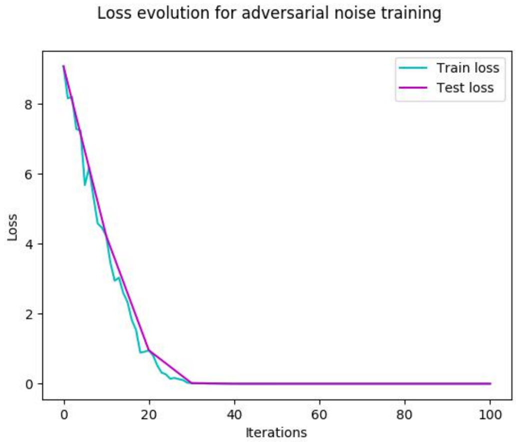


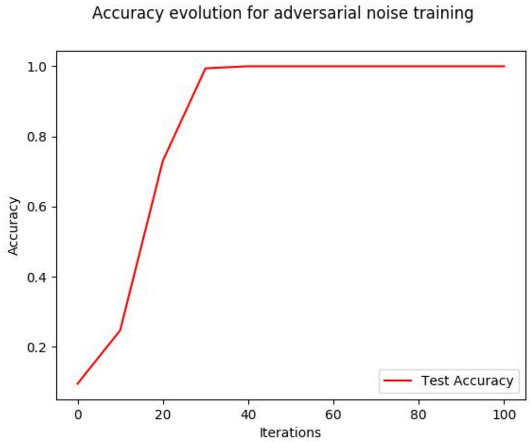
Following are the predictions for noise infused images:



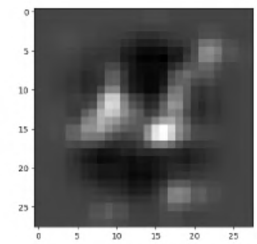
2.5 Adversarial Examples for label 4

Following are plots of loss evolution and accuracy evolution for adversarial noise training.

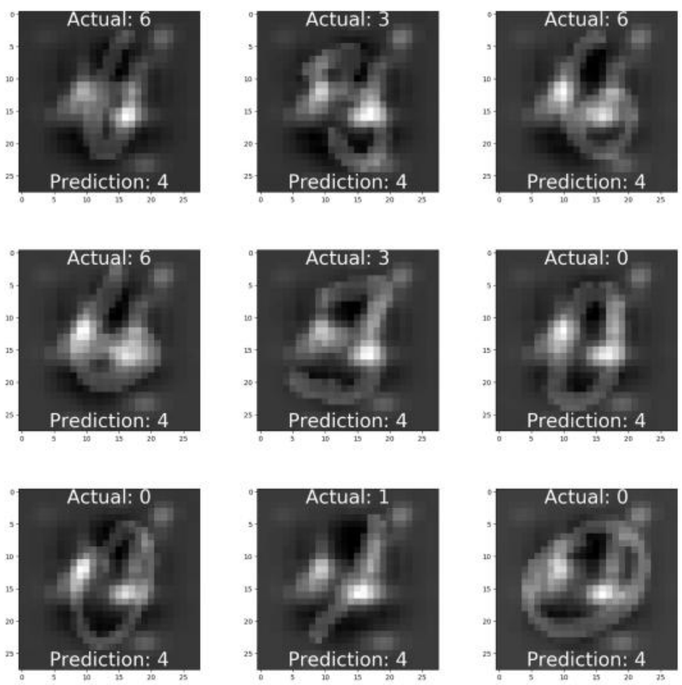




Following is the generated adversarial noise for 4:

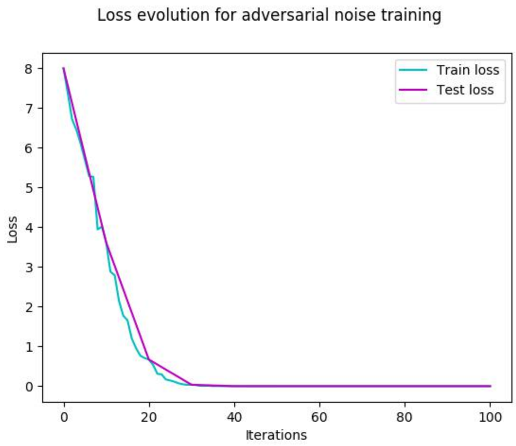


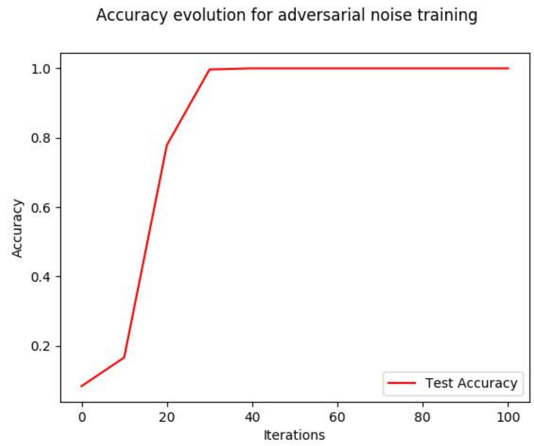
Following are the predictions for noise infused images:



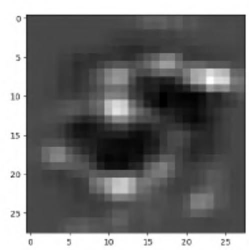
2.6 Adversarial Examples for label 5

Following are plots of loss evolution and accuracy evolution for adversarial noise training.

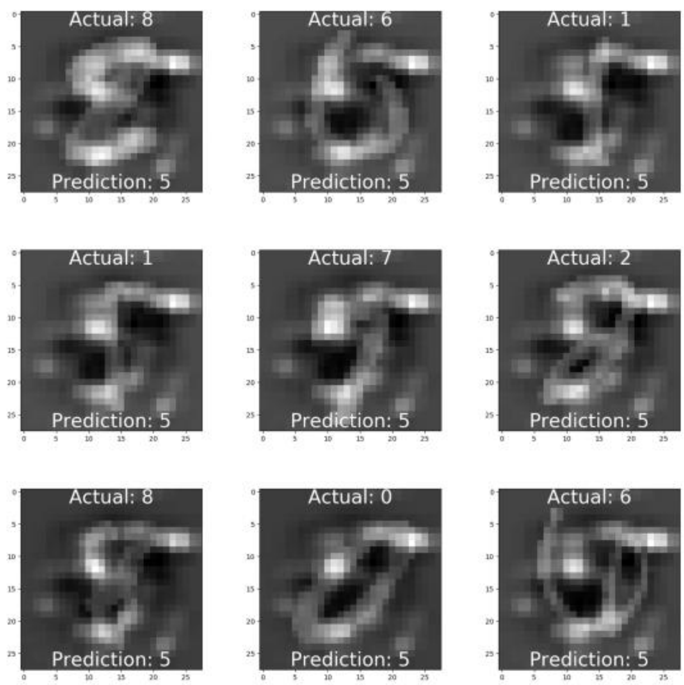




Following is the generated adversarial noise for 5:

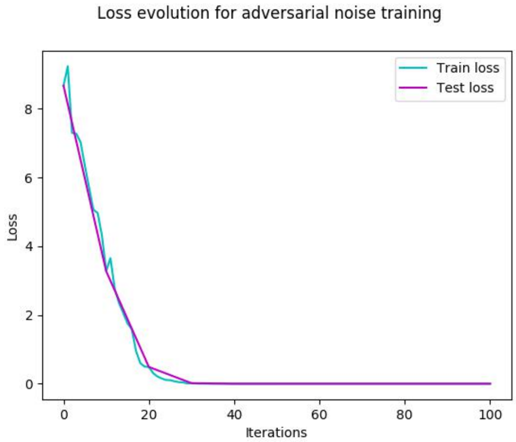


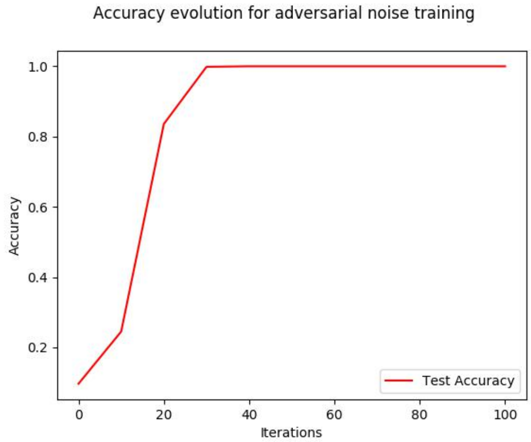
Following are the predictions for noise infused images:



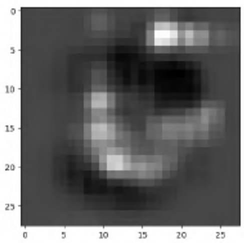
2.7 Adversarial Examples for label 6

Following are plots of loss evolution and accuracy evolution for adversarial noise training.

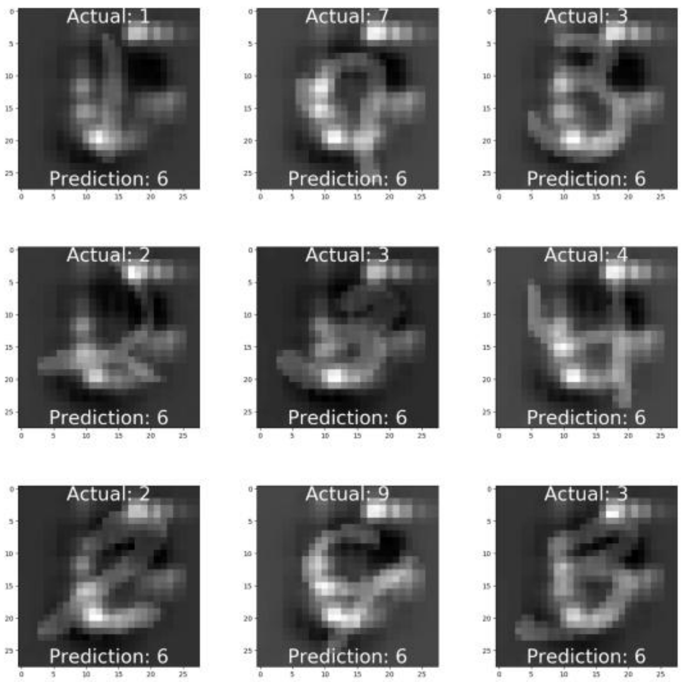




Following is the generated adversarial noise for 6:

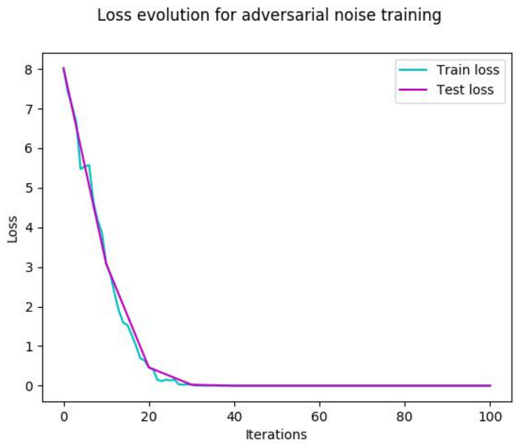


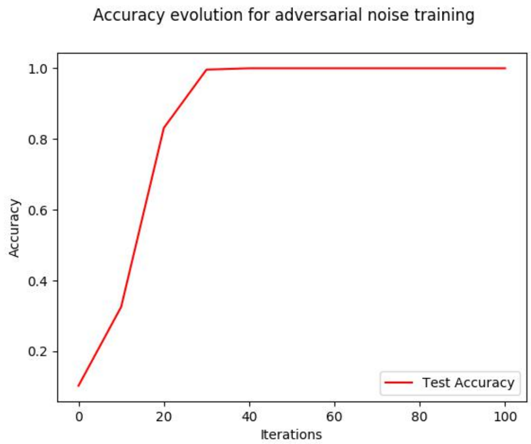
Following are the predictions for noise infused images:



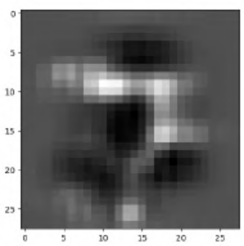
2.8 Adversarial Examples for label 7

Following are plots of loss evolution and accuracy evolution for adversarial noise training.

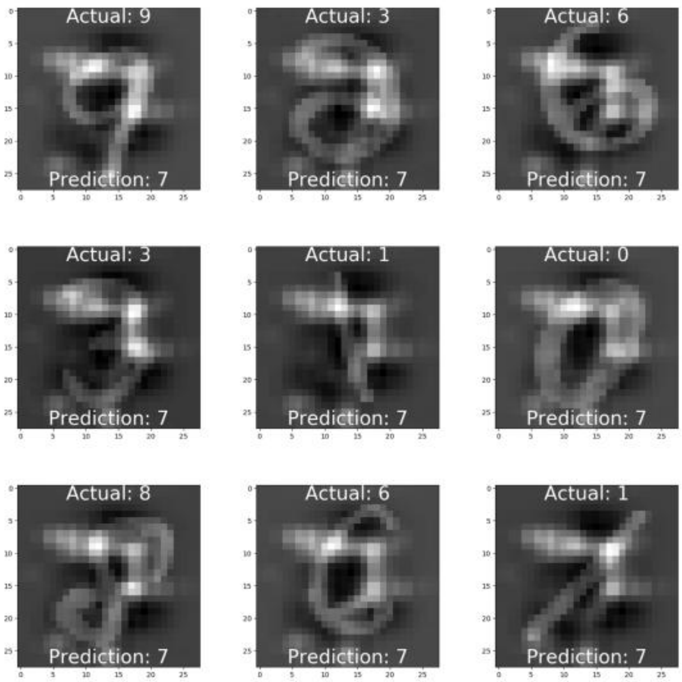




Following is the generated adversarial noise for 7:

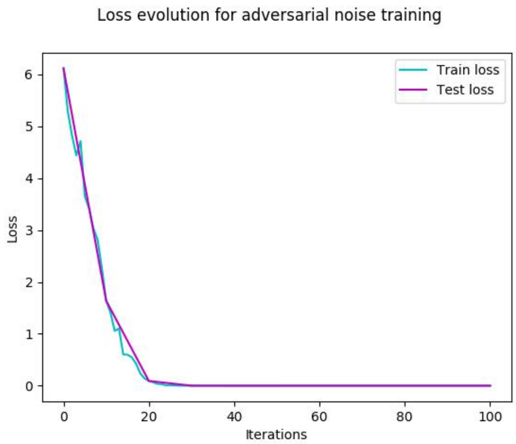


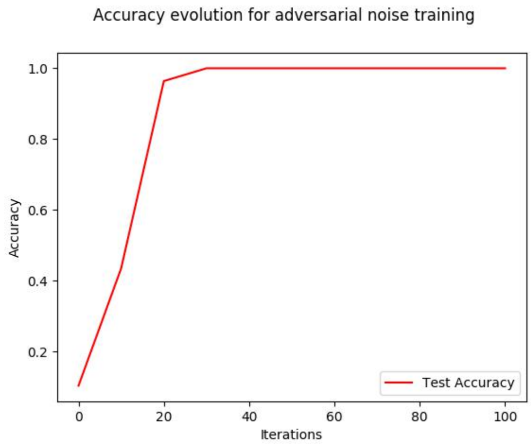
Following are the predictions for noise infused images:



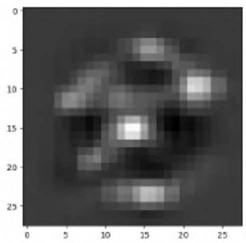
2.9 Adversarial Examples for label 8

Following are plots of loss evolution and accuracy evolution for adversarial noise training.

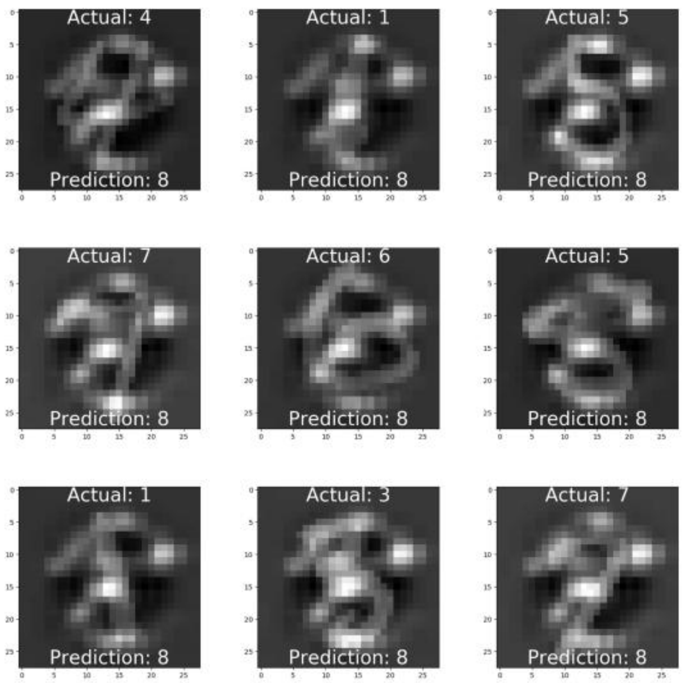




Following is the generated adversarial noise for 8:

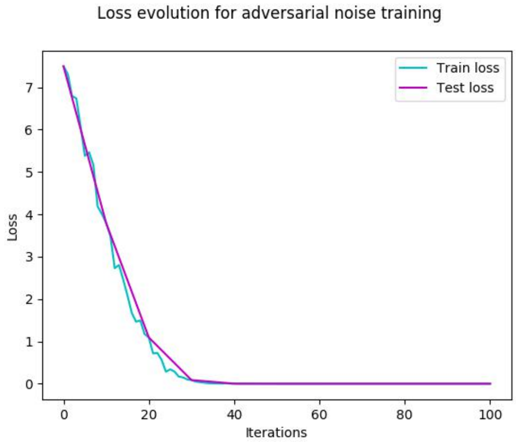


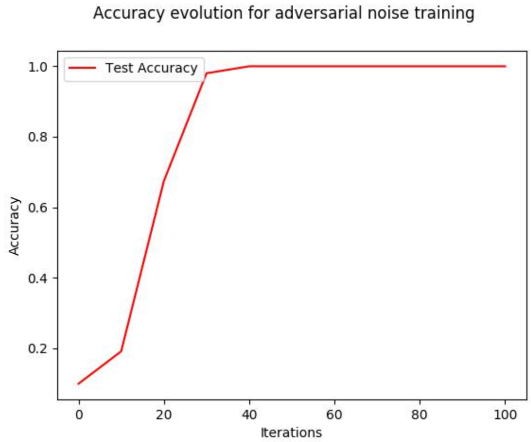
Following are the predictions for noise infused images:



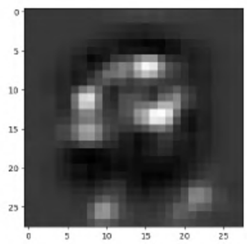
2.10 Adversarial Examples for label 9

Following are plots of loss evolution and accuracy evolution for adversarial noise training.

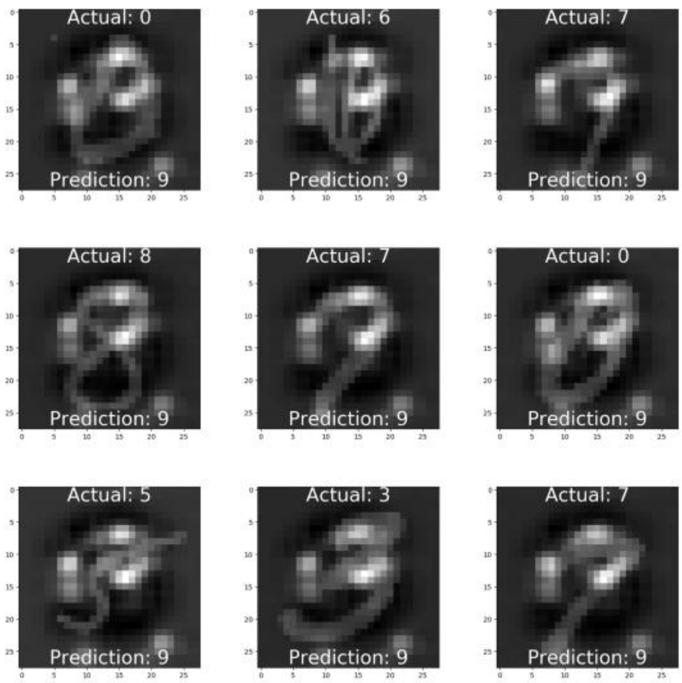




Following is the generated adversarial noise for 9:



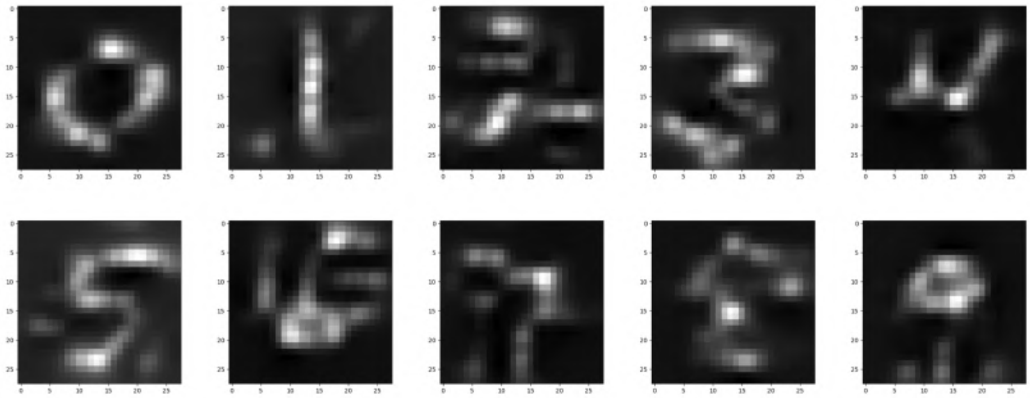
Following are the predictions for noise infused images:



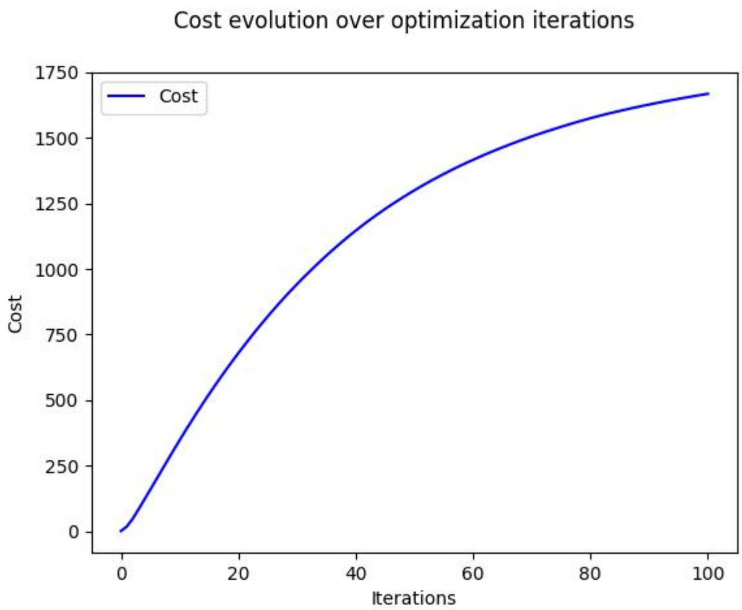
3 Visualizing Convolutional Neural Networks

3.1 Final layer neurons as cost

The noise matrix visualizations for optimizing over final neurons are as follows:



There is a common trend observed in the increase in cost function over optimization iterations:



3.2 Max pooling centre neurons as cost

The noise matrix visualizations for optimizing over maxpool filter centre neurons are as follows: