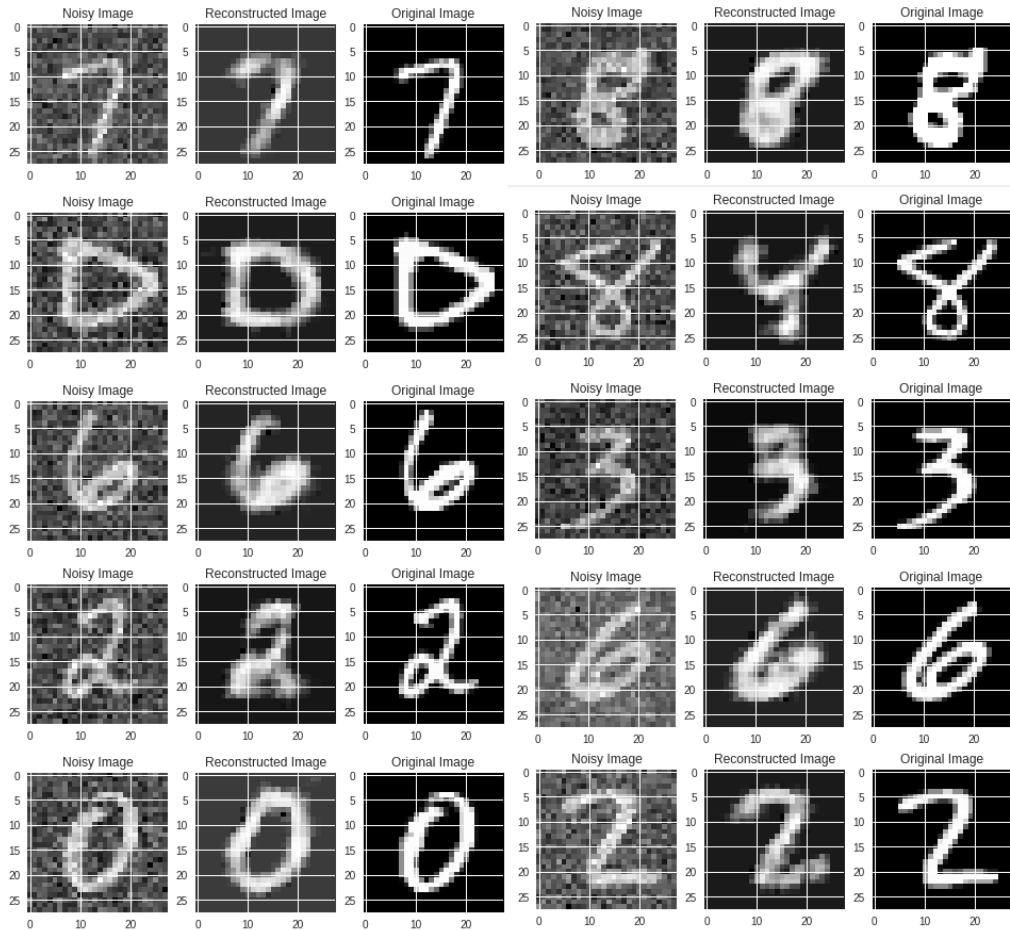


Talal Zahid – 01815

## Algorithms for Machine Learning Assignment #2 Report

Lr = 0.5 , Epochs = 10, batch size = 16, Encoder Layers = 2, Decoder Layers = 2

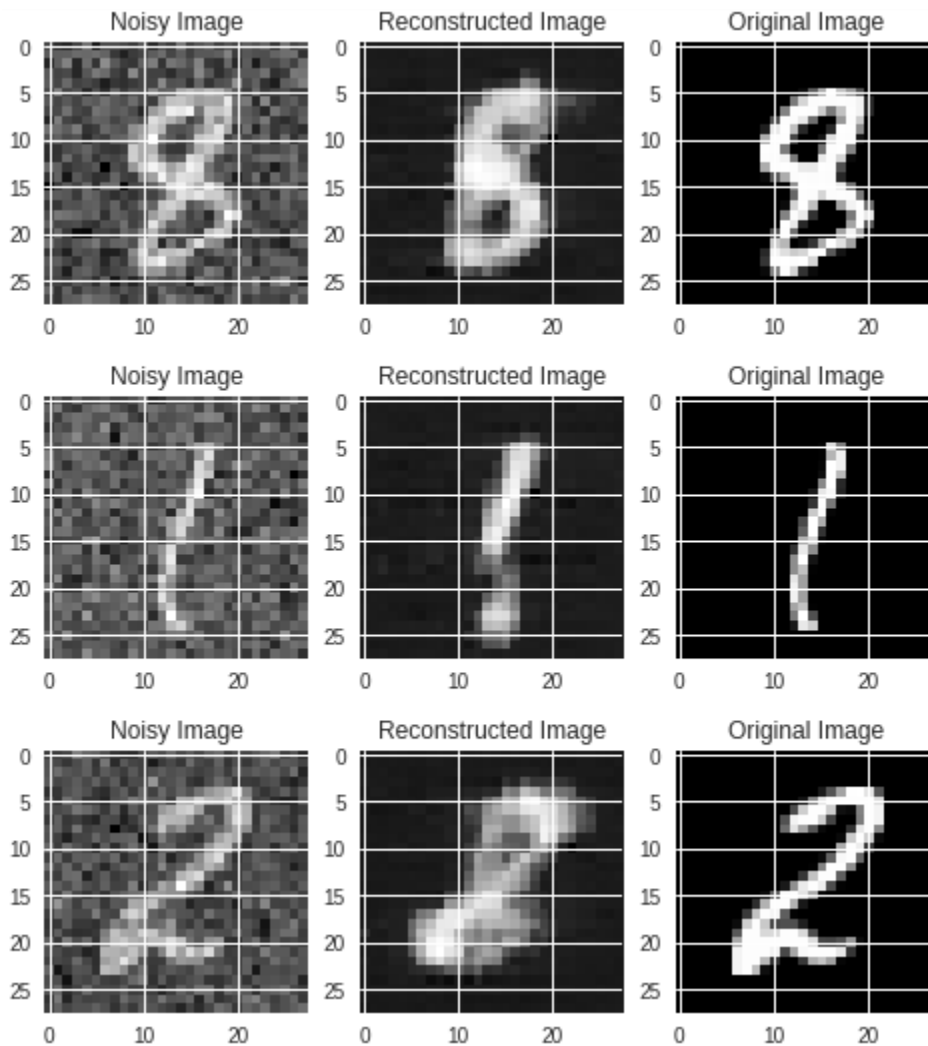


```
At Iteration : 1 ; Mean-Squared Error : 0.044658
At Iteration : 2 ; Mean-Squared Error : 0.034213
At Iteration : 3 ; Mean-Squared Error : 0.031628
At Iteration : 4 ; Mean-Squared Error : 0.030595
At Iteration : 5 ; Mean-Squared Error : 0.029990
At Iteration : 6 ; Mean-Squared Error : 0.029623
At Iteration : 7 ; Mean-Squared Error : 0.029423
At Iteration : 8 ; Mean-Squared Error : 0.029259
At Iteration : 9 ; Mean-Squared Error : 0.029131
At Iteration : 10 ; Mean-Squared Error : 0.029035
```

Where Iteration = epoch,

Mean-Squared Error = (epoch + 1, running loss/ (training set size/batch size))

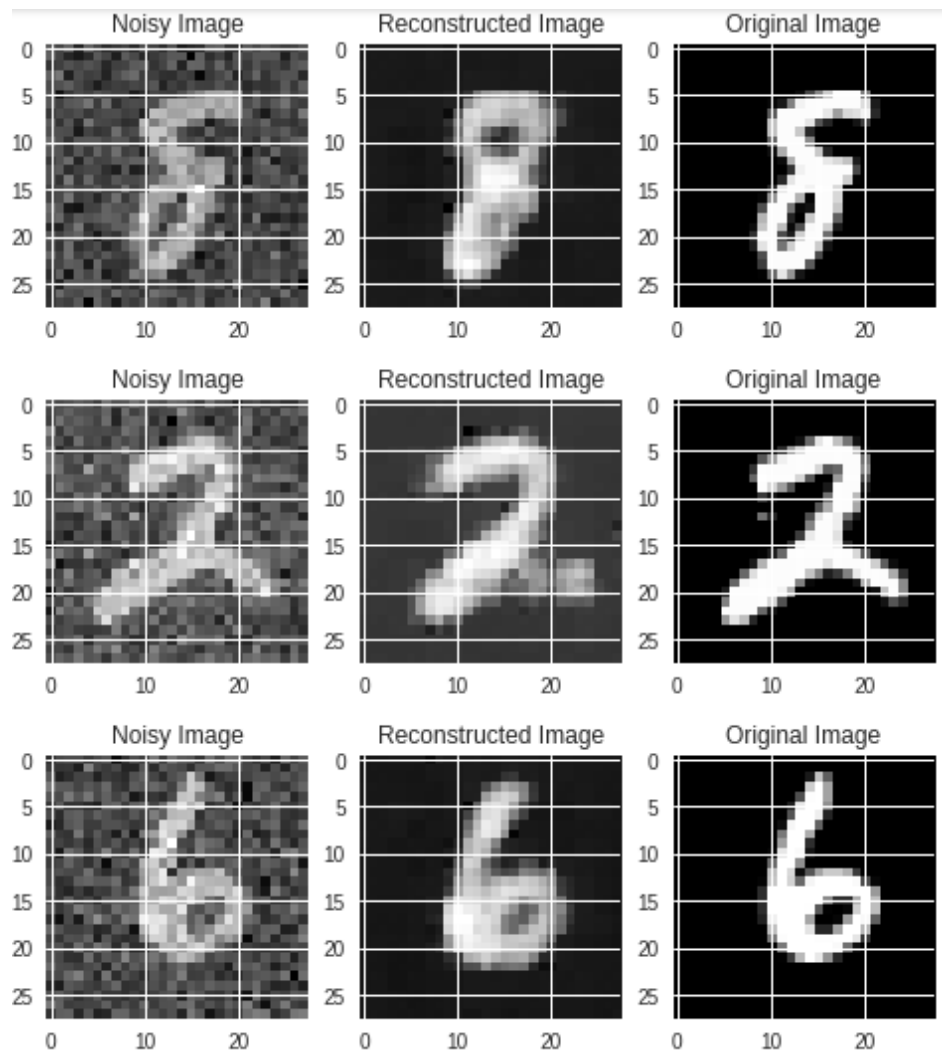
Lr = 0.5 , Epochs = 10, batch size = 128, Encoder Layers = 2, Decoder Layers = 2



```
At Iteration : 1 ; Mean-Squared Error : 0.057638
At Iteration : 2 ; Mean-Squared Error : 0.039783
At Iteration : 3 ; Mean-Squared Error : 0.037221
At Iteration : 4 ; Mean-Squared Error : 0.036242
At Iteration : 5 ; Mean-Squared Error : 0.035518
At Iteration : 6 ; Mean-Squared Error : 0.034800
At Iteration : 7 ; Mean-Squared Error : 0.034233
At Iteration : 8 ; Mean-Squared Error : 0.033617
At Iteration : 9 ; Mean-Squared Error : 0.033078
At Iteration : 10 ; Mean-Squared Error : 0.032721
```

Going from batch size 16 to 128, did not produce any significant improvement result (Mean Square Error). There seem to be underfitting happening. Or a saturation in the learning, with nothing new (no new features) to learn.

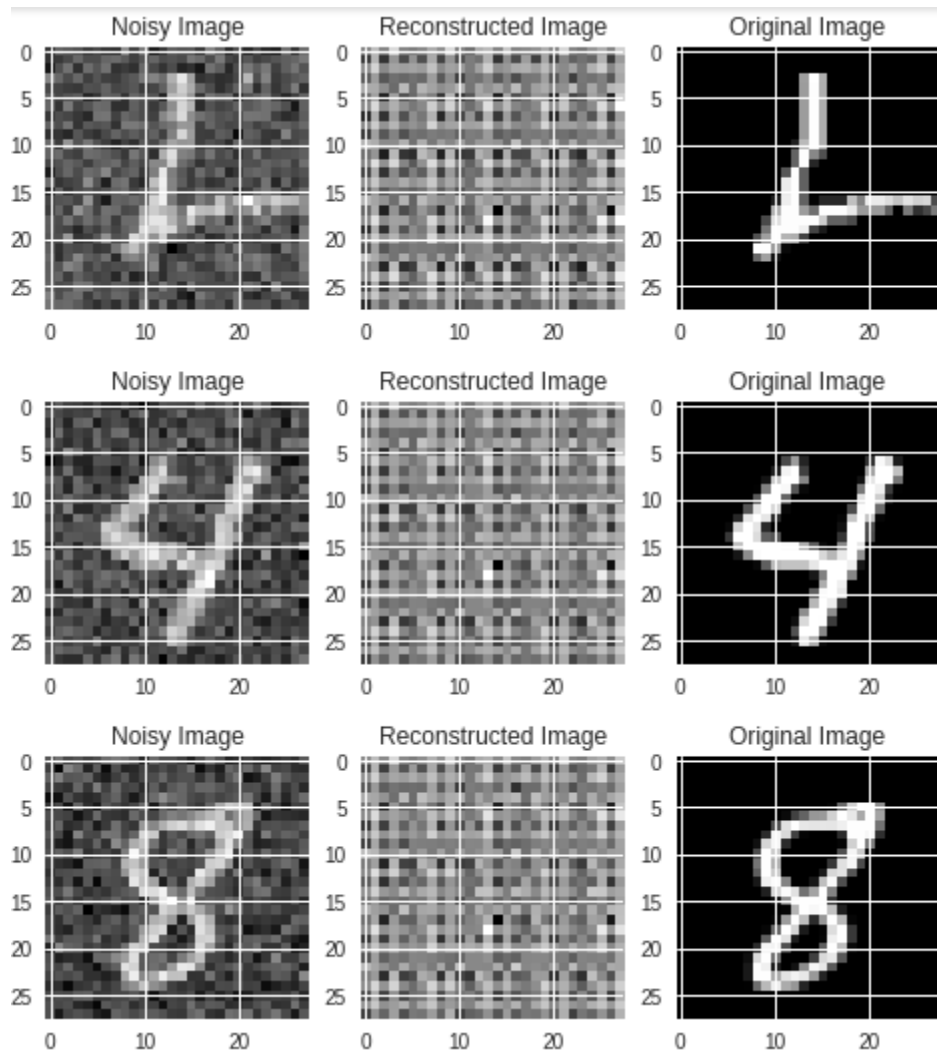
Lr = 0.5, Epochs = 100, batch size = 128, Encoder Layers = 2, Decoder Layers = 2



```
At Iteration : 1 ; Mean-Squared Error : 0.044658
At Iteration : 2 ; Mean-Squared Error : 0.034213
At Iteration : 3 ; Mean-Squared Error : 0.031628
At Iteration : 4 ; Mean-Squared Error : 0.030595
At Iteration : 5 ; Mean-Squared Error : 0.029990
...
At Iteration : 32 ; Mean-Squared Error : 0.023943
.....
At Iteration : 100 ; Mean-Squared Error : 0.023404
```

After Epoch 30, the Mean Squared Error improved by a very little margin. So, it was redundant to learn for that many epochs for no significant improvement.

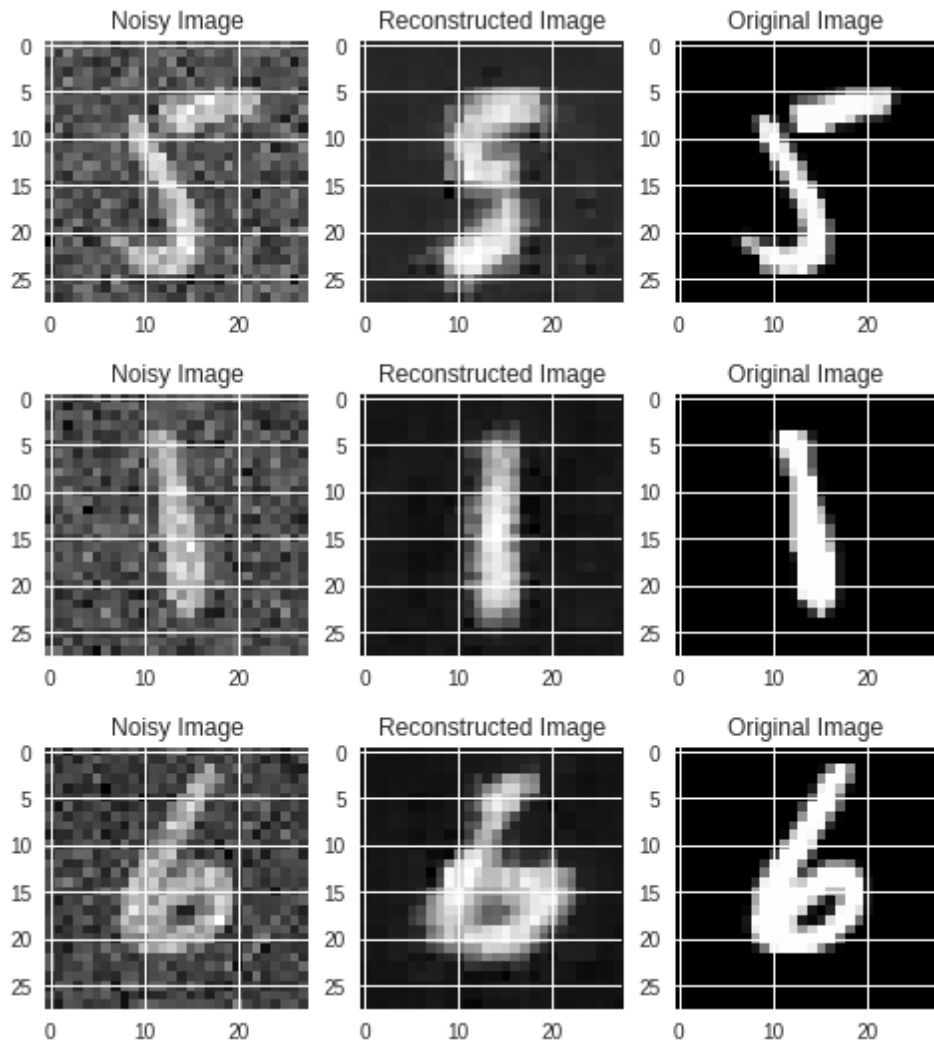
Lr = 0.00001, Epochs = 10, batch size = 128, Encoder Layers = 2, Decoder Layers = 2



```
At Iteration : 1 ; Mean-Squared Error : 0.099683
At Iteration : 2 ; Mean-Squared Error : 0.098232
At Iteration : 3 ; Mean-Squared Error : 0.097227
At Iteration : 4 ; Mean-Squared Error : 0.096530
At Iteration : 5 ; Mean-Squared Error : 0.096068
At Iteration : 6 ; Mean-Squared Error : 0.095730
At Iteration : 7 ; Mean-Squared Error : 0.095507
At Iteration : 8 ; Mean-Squared Error : 0.095351
At Iteration : 9 ; Mean-Squared Error : 0.095239
At Iteration : 10 ; Mean-Squared Error : 0.095166
```

Very Less learning rate, overfitting. Very poor results as not many features were learned. Learning rate should be higher than 0.00001. Similar results were experienced until I increased the learning rate till LR = 0.01.

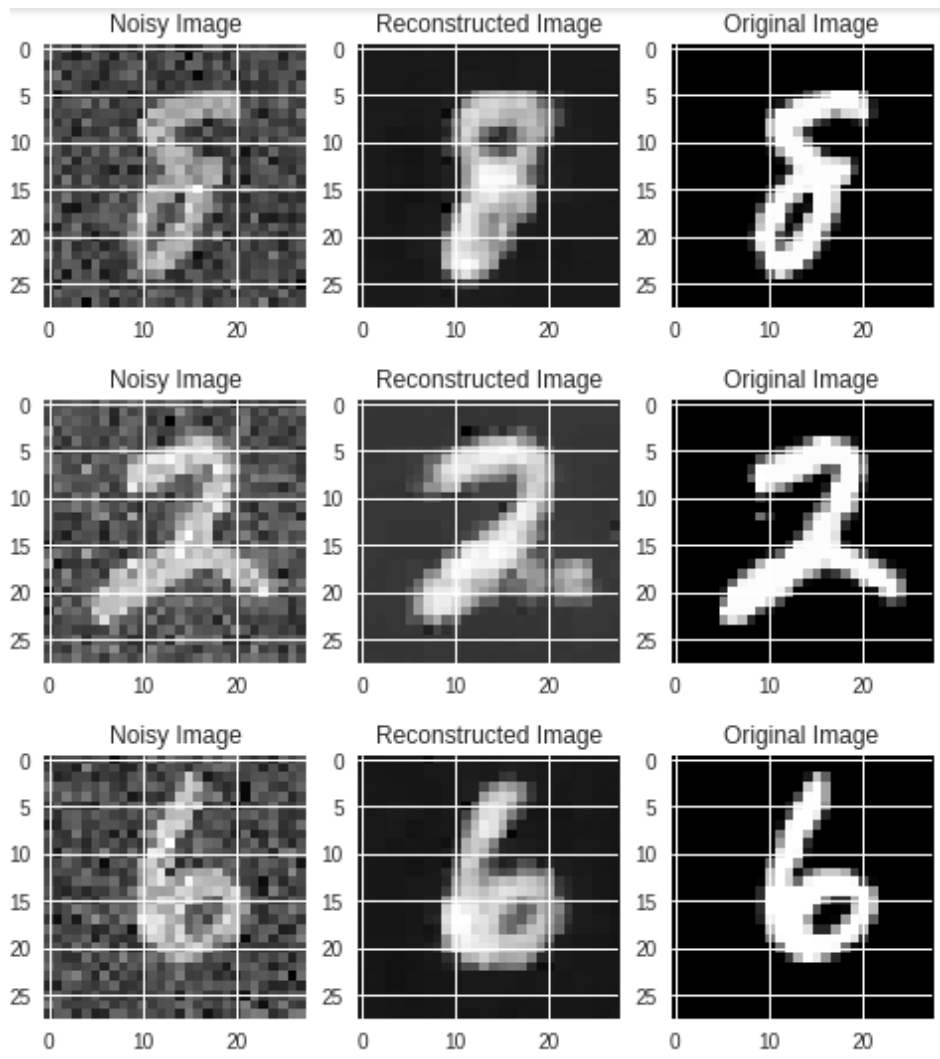
Lr = 0.1, Epochs = 10, batch size = 128, Encoder Layers = 2, Decoder Layers = 2



```
At Iteration : 1 ; Mean-Squared Error : 0.082504
At Iteration : 2 ; Mean-Squared Error : 0.057221
At Iteration : 3 ; Mean-Squared Error : 0.049953
At Iteration : 4 ; Mean-Squared Error : 0.045272
At Iteration : 5 ; Mean-Squared Error : 0.041762
At Iteration : 6 ; Mean-Squared Error : 0.039412
At Iteration : 7 ; Mean-Squared Error : 0.037914
At Iteration : 8 ; Mean-Squared Error : 0.036909
At Iteration : 9 ; Mean-Squared Error : 0.036236
At Iteration : 10 ; Mean-Squared Error : 0.035665
```

The MSQ certainly improved going from LR = 0.00001 to LR = 0.1. But did not produce results as good as LR = 0.5.

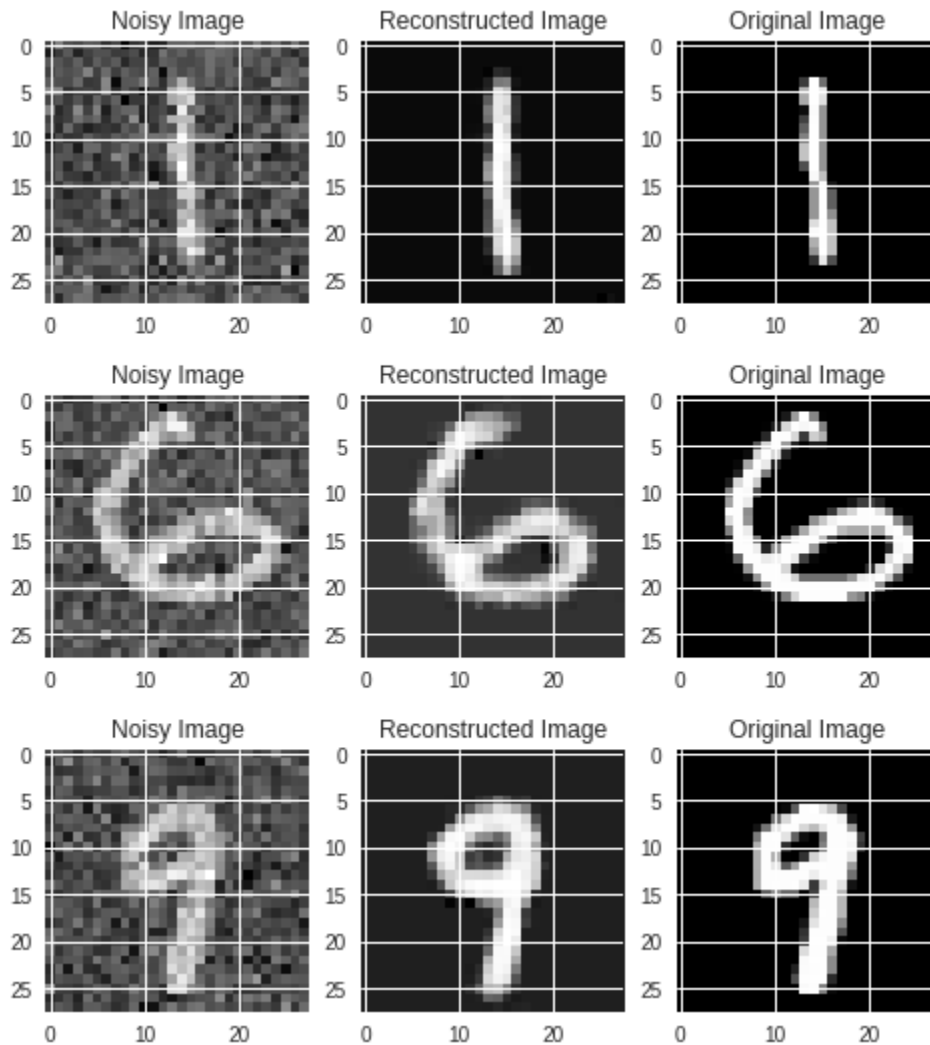
Lr = 0.5, Epochs = 10, batch size = 128, Encoder Layers = 3, Decoder Layers = 3



```
At Iteration : 1 ; Mean-Squared Error : 0.040490
At Iteration : 2 ; Mean-Squared Error : 0.031937
At Iteration : 3 ; Mean-Squared Error : 0.030775
At Iteration : 4 ; Mean-Squared Error : 0.030348
At Iteration : 5 ; Mean-Squared Error : 0.030013
At Iteration : 6 ; Mean-Squared Error : 0.029744
At Iteration : 7 ; Mean-Squared Error : 0.029609
At Iteration : 8 ; Mean-Squared Error : 0.029499
At Iteration : 9 ; Mean-Squared Error : 0.029392
At Iteration : 10 ; Mean-Squared Error : 0.029345
```

Increasing the encoder and decoder layer by 1 doesn't seem to have any improvement over the default 2 layers for each.

Lr = 0.5, Epochs = 25, batch size = 128, Encoder Layers = 2, Decoder Layers = 2



```
At Iteration : 1 ; Mean-Squared Error : 0.034589
At Iteration : 2 ; Mean-Squared Error : 0.024821
At Iteration : 3 ; Mean-Squared Error : 0.023310
At Iteration : 4 ; Mean-Squared Error : 0.022430
.....
At Iteration : 25 ; Mean-Squared Error: 0.019228
```

Best Results produced with minimum MSE. Details about the convolution layers is below.

#### Encoder Layer #1

```
nn.Conv2d(1, 32, 3, stride=3, padding=1), 32 X 10 X 10
nn.MaxPool2d(2, stride=2), 32 X 5 X 5
```

#### Encoder Layer #2

```
nn.Conv2d(32, 16, 3, stride=2, padding=1), 16 X 3 X 3
nn.MaxPool2d(2, stride=1) 16 X 2 X 2
```

)

**Decoder Layer #1**

```
nn.ConvTranspose2d(16, 32, 3, stride=2), b, 32 X 5 X 5
```

**Decoder Layer #2**

```
nn.ConvTranspose2d(32, 16, 5, stride=3, padding=1), 8 X 15 X 15
```

**Decoder Layer #3**

```
nn.ConvTranspose2d(16, 1, 2, stride=2, padding=1), 1, 28, 28
```