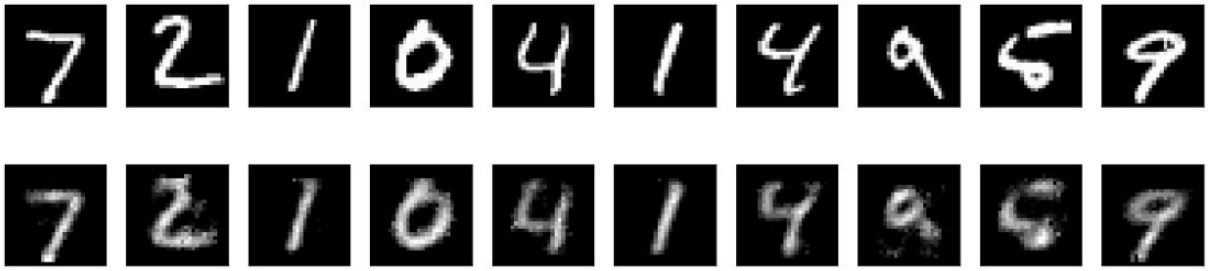
Hammad Ansari 2018450002

Code:

1) Input and Output images:



2) Hyperparameters set for training: (Code size, number of layers, number of nodes per layer, loss function)

    Code size: 32
    Number of layers: 3 convolutional layers
    Number of nodes per layer:
        First layer 1 to 16
        Second layer 16 to 32
        Third layer 32 to 64
    Loss function: binary_crossentropy

3) Number of Epochs considered.

    100 Epochs to get better results

4) Observation

- First of all, we are going to import all the required libraries for the Convolutional Operation on the MNIST dataset.
- Now we can create encoder and decoder since the convolutional layer consists of an encoder, decoder, and code.
- Setting up the format of the image 32x32 for the encoding part and while creating the model we will use Activation Function ‘relu’ and also using ‘relu’ as Activation Function in the decoder.
- Setting up the models, Encoder with image dimensions and encoded dense layer. Same with the decoder.
- Now, compiling the model wih optimizer adam and loss function binary cross entropy.
- Now, we need to pull the dataset from csv or anywhere and store it into variables. Scaling the dataset is very important so we’ll normalize and reshape it.
- Now is the time to train our model on the training dataset with epochs=100, batch size 256, true shuffling and validation test with test dataset.

- After the training we can view the dataset and output images through test array and decoded array.

5) Conclusion
   Traditional method of compressing an image or reducing dimensionality is Pricipal Component Analysis which provides a really lengthy process and takes a lot of time whereas Convolutional Autoencoders are very simple and takes very less time to compress images. Users can generate back the original images through compressed images.

6) Practical applications
   a) Image coloring
   b) Feature variations
   c) Dimensionality reduction
   d) Denoising images
   e) Watermark removal
   f) High resolution images
   g) Sharp and clear images