Github Repo: https://github.com/abe-hashan/mnist-classification-keras.git

1.
a)

Dataset has 28*28 sized images. 60000 Train data and 10000 test data.

- Started by adding 1 convolution layer with 32 kernel size of 3,3. The purpose was to learn spatial features of images.
- A max pool layer was added to reduce the size of features. It will help to reduce the computation while keeping the most prominent features leaned.
- Then the features were flattened to be inputs to a dense layer. At first, only one dense layer was used as the output layer.
- Later added another convolution layer and a max pool layer to increase the accuracy.
- Softmax was used as the activation function of last layer to enable multiclass classification
- Relu was used as activation function as it is a popular activation function recently because it does not cut off the positive value of a node value.

b)
Epoch 1/15 - accuracy: 0.8864 val_accuracy: 0.9790

Epoch 2/15 - accuracy: 0.9665 - val_accuracy: 0.9863

Epoch 3/15 - accuracy: 0.9750 - val_accuracy: 0.9868

Epoch 4/15 - accuracy: 0.9785 - val_accuracy: 0.9870

Epoch 5/15 - accuracy: 0.9814 - val_accuracy: 0.9895

Epoch 6/15 - accuracy: 0.9828 - val_accuracy: 0.9903

Epoch 7/15 - accuracy: 0.9840 - val_accuracy: 0.9905

Epoch 8/15 - accuracy: 0.9851 - val_accuracy: 0.9917

Epoch 9/15 - accuracy: 0.9862 - val_accuracy: 0.9908

Epoch 10/15 - accuracy: 0.9873 - val_accuracy: 0.9918

Epoch 11/15 - accuracy: 0.9879 - val_accuracy: 0.9915

Epoch 12/15 - accuracy: 0.9882 - val_accuracy: 0.9913

Epoch 13/15 - accuracy: 0.9888 - val_accuracy: 0.9922

Epoch 14/15 - accuracy: 0.9895 - val_accuracy: 0.9898

Epoch 15/15 - accuracy: 0.9894 - val_accuracy: 0.9922

2.

Test accuracy for noise factor 25: 0.9868999719619751

Test accuracy for noise factor 40: 0.9779999852180481

Test accuracy for noise factor 60: 0.9506999850273132

3.

For 0.25 noise factor, adding 2 more dense layers was able to increase the accuracy to 0.99 from 0.986

For 0.60 noise factor, adding 2 more dense layers was able to increase the accuracy to 0.95 from 0.961

Adding more than 2 dense layers reduced the accuracy.

Tries adding another convolution and max pool layer and it caused in a reduced accuracy as well.