**Generative Adversarial Network Architecture**:

Generator:

1) Input Layer – Inputs images of size (28,28,1)

2) Conv2D – Filters: 32, Kernel size-3, activation: RELU

3) Max Pooling 2D Layer – pool size: 2

4) Conv2D – Filters: 64, Kernel size-3, activation: RELU

5) Max Pooling 2D Layer – pool size: 2

6) Flatten Layer

7) Output layer – Nodes: 28\*28

Discriminator:

1) Input Layer – Inputs images of size (28,28,1)

2) Conv2D – Filters: 128, Kernel size-3, activation: RELU

3) Max Pooling 2D Layer – pool size: 2

4) Conv2D – Filters: 64, Kernel size-3, activation: RELU

5) Max Pooling 2D Layer – pool size: 2

6) Flatten Layer

7) Output layer – Nodes: 1, activation: SOFTMAX

GAN Model is combination of Generator and Discriminator

**Variational Auto Encoder Architecture:**

Encoder:

1) Input Layer – Inputs images of size (28,28,1)

2) Dense Layer – Nodes: 28\*28, activation: RELU

3) Dense Layer – Nodes: 128, activation: RELU

4) Output layer – Nodes: 28, activation: SIGMOID

Decoder:

1) Input Layer – Inputs images of size (28,1)

2) Dense Layer – Nodes: 128, activation: RELU

3) Dense Layer – Nodes: 256, activation: RELU

4) Output layer – Nodes: 784, activation: SIGMOID

VAE is Combination of Encoder and Decoder.

**Accuracy:**

GAN has 0 % accuracy i.e. total failure while VAE has around 15% accuracy. GAN output complete white image while VAE shows some kind of blurry digit pattern in center of the image.

**Reason for Failure:**

1) GAN fails due to less epochs and smaller dataset used for training. Also, I might tweak the hyper parameters and change the architecture.

2) VAE fails due to simpler model architecture, I will use may be different type of architecture.