SUBJECT: Deep Learning and Computer Vision
SUBJECT CODE: ECS795P

Course Work 2: Unsupervised Learning by Generative Adversarial Network

NAME: Ganesh Kumaran Masilamani
STUDENT NO: 200434339

1. **What is the difference between supervised learning & unsupervised learning in image classification task? (10% of CW2)**

<u>**Supervised Learning in image classification task:**</u>

- In supervised learning, collection of images are given which has the labelled data. Classifier is then trained based upon the labelled images and an learning Algorithm is implemented to map the function for each inputs and its labels. Finally, for the new image without a label, our classifier predicts its label.
- Examples for supervised learning are Regression and Classification.

<u>**Unsupervised Learning in image classification task:**</u>

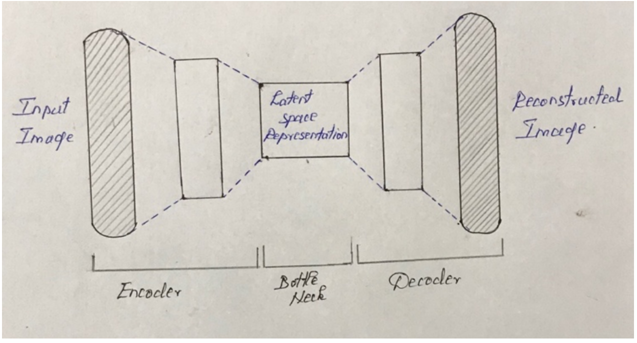
- In unsupervised learning, Training data will not be given. Hence the classifier has to learn the structure of dataset, then it has to choose an algorithm in which it can produce the better results and then it predict the Labels of the dataset.
- Examples of unsupervised learning are clustering, Generative Adversarial Networks(GAN).

2. **What is the difference between an auto-encoder and a generative adversarial network considering (1) model structure; (2) optimized objective function; (3) training procedure on different components. (10% of CW2)**

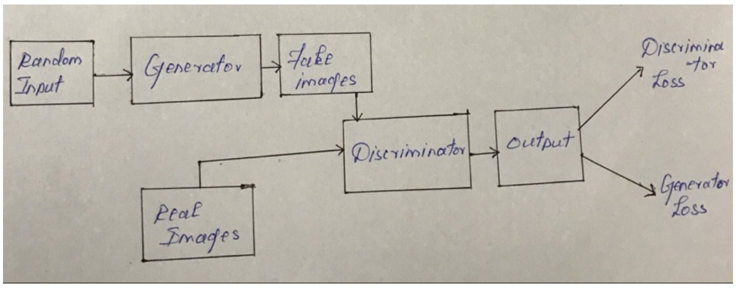
Both Auto-encoder and Generative adversarial network are considered as the deep generative models.

(1) <u>Model structure:</u>

- **Auto-encoders** consists of an encoder in the input end and a decoder in the output end.
- The encoder is used to squeeze the images into a vector and a decoder is used in reconstruction of the compressed image from a vector to the actual input.



- **Generative adversarial network(GAN**) consists of two major parts namely Generator and Discriminator. It uses random noise as input to generate the samples.
- Generator network is used to pass the real images into the model. Whereas, discriminator network is used to pass with fake images, discriminator is used to identify the produced images either Real or Fake.



(2) <u>Optimized objective function:</u>

- The optimized objective function of the **Autoencoder** is to minimize the reconstruction L2 loss to a minimum in which reconstructed image is as close to original input.

$$\mathcal{L}(\phi,\theta,x) = D_{KL}(q_\phi(h|x)||p_\theta(h)) - E_{q_\phi(h|x)}(\ln p_\theta(x|h))$$

- The optimized objective function of the **Generative Adversarial Network** is, the Generator network which generates sample of plausible data which minimize the two sample objective. and the Discriminator network which is used to distinguish samples from the original dataset with the generator fake data which is used to maximize the test objectives.

$$\min_G \max_D V(D,G) = E_{x\sim P_{data}}[\ln D(x)] + E_{x\sim P_z}[\ln(1-D(G(z)))]$$

(3) <u>Training procedure:</u>

- In **Autoencoders**, training is performed by tradeoff between the power of reconstruction generation and the mixed image generation .In this, training is performed between encoder and decoder.

- In **Generative adversarial network**, generator and discriminator are synchronized. The generative model is used to discriminate the generated samples from latent noise distribution inputs . In this, training is performed between generator and the discriminator.

3. **How is the distribution $p_g(x)$ learned by the generator compared to the real data distribution $p_{data}(x)$ when the discriminator cannot tell the difference between these two distributions? (15% of CW2)**

Firstly, The generator generates a latent random sample distribution Z and passed through the deep generative network of the generator **G(z),** and then through the Discriminator.

The main aim of the Generator is to achieve global optima where $\mathbf{P_g = P_{data}}$. To achieve global optima, generator has to keep updating its bias "*z*" and weights "theta". According to Goodfellow algorithm Generative Adversarial Networks use stochastic gradient ascent to update weights of discriminator and stochastic gradient descent to update weights of generator.

After the discriminator is trained, Each time when the discriminator gets the right output, i.e, **D(G(z))=0**, The loss is calculated and the weights gets updated since both D and G are differentiable functions. When weights gets updated, the deep network of Generator **G(z)** come close to $\mathbf{P_{data}(x)}$**.** Secondly, when $\mathbf{P_{data}(x)}$**= G(z), D(G(z))= 1,** the output of the discriminator will be 0.5 which means that, the discriminator either failed to calculate the difference or failed to identify the input which was passed from generator was fake.

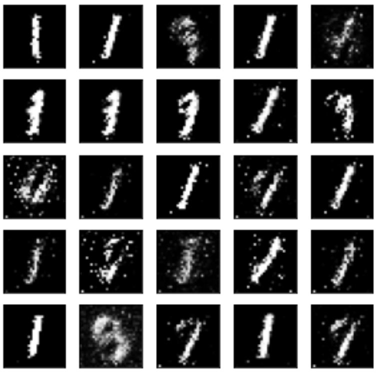
- To obtain to the final solution, we periodically update the loss function in which the target is to achieve the min(G) and max(D) as proposed by Goodfellow in the below equation,

$$\min_G \max_D V(D,G) = E_{x \sim P_{data}}[\ln D(x)] + E_{x \sim P_z}[\ln(1 - D(G(z)))]$$

Finally, we achieve $\mathbf{P_g = P_{data} = 0}$ When the output of the discriminator **D(x)** is 0.5, we conclude that Original data and the fake data are not distinguishable. $\mathbf{P_{data}(x)}$ by **G(z)** which tends to be much similar to every other data.

4. **Show the generated images at 10 epochs, 20 epochs,50 epochs,100 epochs by using the architecture required in Guidance. (15% of CW2)**

a.)Generated image at 10 epochs:

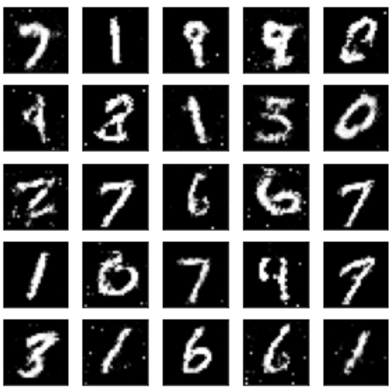


Epoch 10

b.)Generated image at 20 epochs:

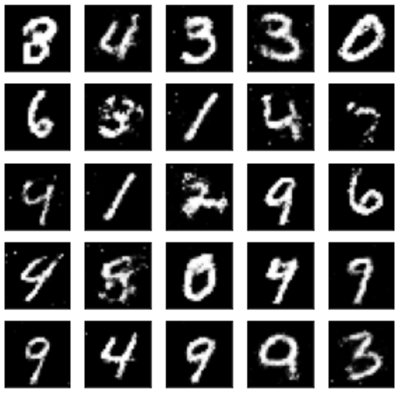
Epoch 20

c.)Generated image at 50 epochs:



Epoch 50

d.)Generated image at 100 epochs:



Epoch 100

e.)Plot for Generator and Discriminator loss: