SNS COLLEGE OF TECHNOLOGY
(An Autonomous Institution)
Coimbatore – 35.

DEPARTMENT OF BIOMEDICAL ENGINEERING

UNIT 2

Generative Adversarial Networks (GAN)

GAN(Generative Adversarial Network) represents a cutting-edge approach to generative modeling within deep learning, often leveraging architectures like **convolutional neural networks**. The goal of generative modeling is to autonomously identify patterns in input data, enabling the model to produce new examples that feasibly resemble the original dataset.

GAN(Generative Adversarial Network) represents a cutting-edge approach to generative modeling within deep learning, often leveraging architectures like **convolutional neural networks**. The goal of generative modeling is to autonomously identify patterns in input data, enabling the model to produce new examples that feasibly resemble the original dataset.

Generator Loss

The objective of the generator in a GAN is to produce synthetic samples that are realistic enough to fool the discriminator. The generator achieves this by minimizing its loss function J_G. The loss is minimized when the log probability is maximized, i.e., when the discriminator is highly likely to classify the generated samples as real. The following equation is given below:

$$J_G = -\frac{1}{m}\Sigma_{i=1}^{m} log D(G(z_i))$$

Where,

- J_G measure how well the generator is fooling the discriminator.
- log $D(G(z_i))$represents log probability of the discriminator being correct for generated samples.
- The generator aims to minimize this loss, encouraging the production of samples that the discriminator classifies as real $(log D(G(z_i))$, close to 1.

Discriminator Model

An artificial neural network called a discriminator model is used in Generative Adversarial Networks (GANs) to differentiate between generated and actual input. By evaluating input samples and allocating probability of authenticity, the discriminator functions as a binary classifier.

Over time, the discriminator learns to differentiate between genuine data from the dataset and artificial samples created by the generator. This allows it to progressively hone its parameters and increase its level of proficiency.

Convolutional layers or pertinent structures for other modalities are usually used in its architecture when dealing with picture data. Maximizing the discriminator's capacity to accurately identify generated samples as fraudulent and real samples as authentic is the aim of the adversarial training procedure. The discriminator grows increasingly discriminating as a result of the generator and discriminator's interaction, which helps the GAN produce extremely realistic-looking synthetic data overall.

Discriminator Loss

The discriminator reduces the negative log likelihood of correctly classifying both produced and real samples. This loss incentivizes the discriminator to accurately categorize generated samples as fake and real samples with the following equation:

$J_D = -\frac{1}{m}\Sigma_{i=1}^{m} log\ D(x_i) - \frac{1}{m}\Sigma_{i=1}^{m} log(1 - D(G(z_i))$

- J_D assesses the discriminator's ability to discern between produced and actual samples.
- The log likelihood that the discriminator will accurately categorize real data is represented by $logD(x_i)$.
- The log chance that the discriminator would correctly categorize generated samples as fake is represented by $log(1 - D(G(z_i)))$.
- **The discriminator aims to reduce this loss by accurately identifying artificial and real samples.**

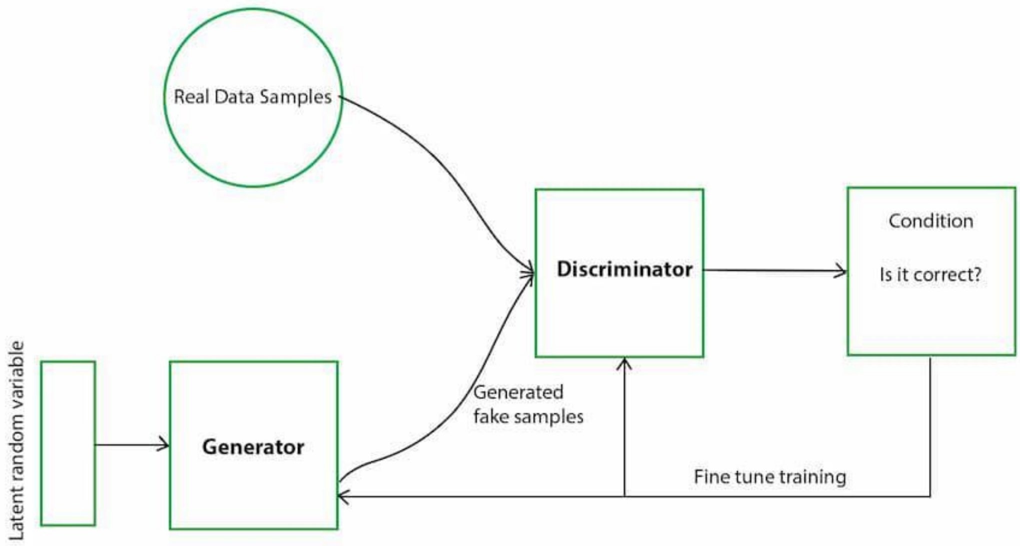
MinMax Loss

In a Generative Adversarial Network (GAN), the minimax loss formula is provided by:

$min_G \; max_D(G, D) = [\mathbb{E}_{x \sim p_{data}}[log \; D(x)] + \mathbb{E}_{z \sim p_z(z)}[log(1 - D(g(z)))]$

Where,

- G is generator network and is D is the discriminator network
- Actual data samples obtained from the true data distribution $p_{data}(x)$ are represented by x.
- Random noise sampled from a previous distribution $p_z(z)$(usually a normal or uniform distribution) is represented by z.
- D(x) represents the discriminator's likelihood of correctly identifying actual data as real.
- D(G(z)) is the likelihood that the discriminator will identify generated data coming from the generator as authentic.



How does a GAN work?

The steps involved in how a GAN works:

1. **Initialization:** Two neural networks are created: a Generator (G) and a Discriminator (D).
   - G is tasked with creating new data, like images or text, that closely resembles real data.
   - D acts as a critic, trying to distinguish between real data (from a training dataset) and the data generated by G.

2. **Generator's First Move:** G takes a random noise vector as input. This noise vector contains random values and acts as the starting point for G's creation process. Using its internal layers and learned patterns, G transforms the noise vector into a new data sample, like a generated image.
3. **Discriminator's Turn:** D receives two kinds of inputs:
   - Real data samples from the training dataset.
   - The data samples generated by G in the previous step. D's job is to analyze each input and determine whether it's real data or something G cooked up. It outputs a probability score between 0 and 1. A score of 1 indicates the data is likely real, and 0 suggests it's fake.
4. **The Learning Process:** Now, the adversarial part comes in:
   - If D correctly identifies real data as real (score close to 1) and generated data as fake (score close to 0), both G and D are rewarded to a small degree. This is because they're both doing their jobs well.
   - However, the key is to continuously improve. If D consistently identifies everything correctly, it won't learn much. So, the goal is for G to eventually trick D.
5. **Generator's Improvement:**
   - When D mistakenly labels G's creation as real (score close to 1), it's a sign that G is on the right track. In this case, G receives a significant positive update, while D receives a penalty for being fooled.
   - This feedback helps G improve its generation process to create more realistic data.
6. **Discriminator's Adaptation:**
   - Conversely, if D correctly identifies G's fake data (score close to 0), but G receives no reward, D is further strengthened in its discrimination abilities.
   - This ongoing duel between G and D refines both networks over time.

As training progresses, G gets better at generating realistic data, making it harder for D to tell the difference. Ideally, G becomes so adept that D can't reliably distinguish real from fake data. At this point, G is considered well-trained and can be used to generate new, realistic data samples.

Reference:

https://www.geeksforgeeks.org/generative-adversarial-network-gan/