Generative Adversarial Networks (GANs) são poderosos modelos para geração de dados reais fazendo a competiçao de duas redes neurais, o gerador(**generator)** e o descriminador(**discriminator)**, em um processo conhecido como “adversarial training”. O Processo funcional consiste em:

**Gerador**

* **Proposito**: criar dados sintéticos parecidos com o dataset real;
* **Entrada**: Vetor randômico geralmente distribuições normais ou uniformes;
* **Saida**: um dataset gerado, seja ele uma imagem, texto ou outro tipo de dado;
* **Objetivo**: Criar dado que o descriminador não consiga distinguir entre real e falso.
* **Arquitetura**: Normalmente composto por camadas de upsampling (ex.: convoluções transpostas) para transformar o vetor de ruído de baixa dimensão em uma saída estruturada de alta dimensão.

**Descriminador**

* **Propósito:** Avaliar se um exemplo de dado é real (do conjunto de dados) ou falso (gerado pelo gerador).
* **Entrada:** Um exemplo de dado real ou gerado.
* **Saída:** Uma pontuação de probabilidade indicando a chance de o dado ser real.
* **Objetivo:** Classificar corretamente os exemplos reais e gerados, atuando como um crítico para o gerador.
* **Arquitetura:** Geralmente composto por camadas de downsampling (ex.: convoluções e pooling) para extrair características e fazer previsões binárias.

**Processo Adversarial**

O gerador e o discriminador competem em um **jogo de soma zero**, onde: o gerador tenta minimizar a capacidade do discriminador de distinguir entre real e falso e o discriminador tenta maximizar sua capacidade de distinguir entre real e falso.

**Dinâmica de Treinamento**

* **Otimização Alternada:** O discriminador é treinado para maximizar sua precisão de classificação, e o gerador é treinado para enganar o discriminador.
* **Backpropagation:** Os gradientes fluem por ambas as redes (gerador e discriminador) para atualizar seus parâmetros.
* **Desafios:** O treinamento de GANs pode ser instável devido a problemas como colapso de modo, gradientes desaparecendo ou desequilíbrio no treinamento. Para mitigar isso, estratégias como:
  + **Feature matching** (combinação de características).
  + Uso da **perda de Wasserstein** (GANs-Wasserstein).
  + Penalidades de gradiente.