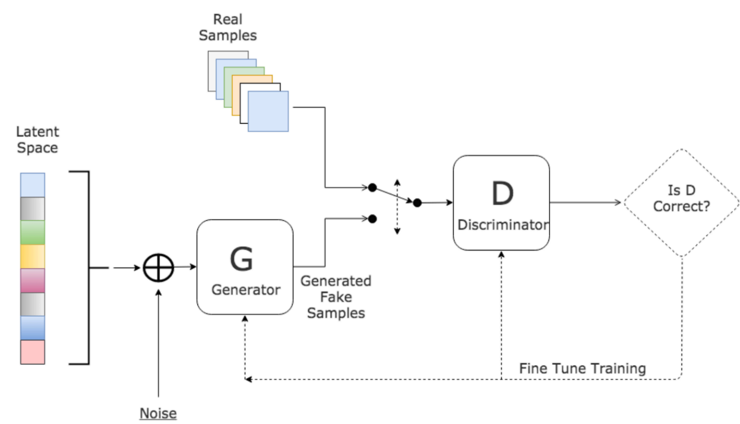
What are GANs ?

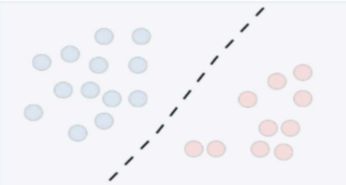
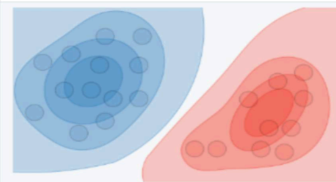
- Generative Adversarial Networks (GANs) are a type of deep learning architecture that uses two neural networks, a generator and a discriminator, to generate new data samples that are similar to a training set.
- The generator network creates new data samples while the discriminator network evaluates the generated samples and determines whether they are similar to the real training data.



Difficulties while training GANs

- **Mode Collapse:** It occurs when the generator network produces only a limited number of distinct outputs rather than a diverse set of outputs.
- **Instability:** GANs are highly sensitive to the initial conditions and the choice of hyperparameters.
- **Convergence:** GANs can be difficult to train to convergence, meaning that the generator and discriminator networks may never reach an optimal solution.

Discriminative vs Generative model

	Discriminative model	Generative model
Goal	Directly estimate $P(y\|x)$	Estimate $P(x\|y)$ to then deduce $P(y\|x)$
What's learned	Decision boundary	Probability distributions of the data
Illustration		
Examples	Regressions, SVMs	GDA, Naive Bayes

Discriminative Loss

- One common example of a discriminative loss function is binary cross-entropy loss, which is defined as:
  - $Loss \ = \ max_D \frac{1}{m} \sum [y \ \times log(D(x)) \ + \ (1 - y) \ \times log(1 - D(G(x))]$
  - Where y = 1 for real image and 0 for fake img, D(x) is the output of discriminator model for real img(x) while D(G(x)) is the output output of discriminator model for fake img generated by the generative model G(x)

Generative Loss

- The generative loss in GANs is a scalar value that represents the discrepancy between the generated samples and the real data.
- The goal of the generator is to minimize this loss so that the generated samples become as similar as possible to the real data.
  - $Loss \ = \ min_G \frac{1}{m} \sum log(1 - D(G(Z))$

GAN Pseudo Code

for number of training iterations **do**

- Sample minibatch of m noise samples $\{z^1, \ldots\ldots, z^m\}$ from noise prior p_g(z).
- Sample minibatch of m examples $\{x^1, \ldots\ldots, x^m\}$ from data generating distribution p_{data}(x).
- Update the discriminator by ascending its stochastic gradient:
  - $\nabla_{\Theta_d} \frac{1}{m} \sum_{i=1}^{m} [logD(x^i) + log(1 - D(G(z^i)))]$.
- Sample minibatch of m noise samples $\{z^1, \ldots\ldots, z^m\}$ from noise prior p_g(z).
- Update the generator by ascending its stochastic gradient:
  - $\nabla_{\Theta_g} \frac{1}{m} \sum_{i=1}^{m} log(D(G(Z^i)))$.

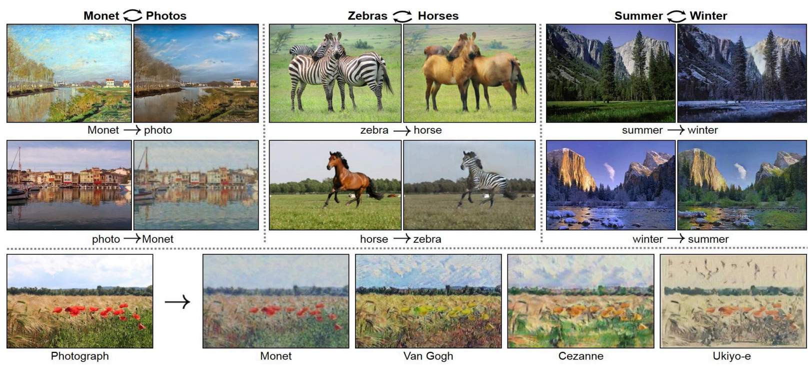
end for

Application of GAN

- Image Generation
- Image Translation
- Data Augmentation
- Text Generation
- Video Generation

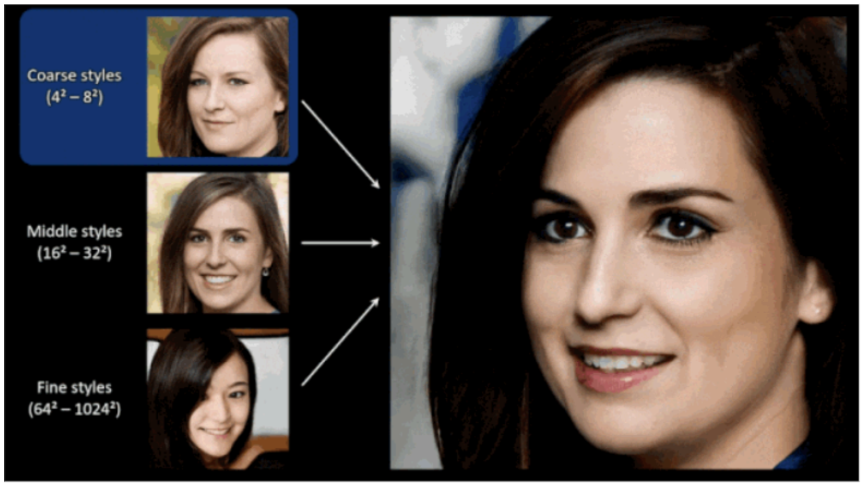
Different Types of GANs

1. CycleGAN: CycleGAN is a GAN that is used for image-to-image translation, where the goal is to transform an image from one domain to another, such as converting a horse into a zebra, or a photograph into a painting.



2. SRGAN is a Generative Adversarial Network used for single image super-resolution. It increases the resolution of an image, making it appear clearer and more detailed.

3. StyleGAN: StyleGAN is a Generative Adversarial Network (GAN) used for synthesizing new images of human faces that are highly realistic and diverse.


Coarse styles
(4² – 8²)
Middle styles
(16² – 32²)
Fine styles
(64² – 1024²)