Deep Generative Modeling
- part 2 -

Hana Sebia
hana.sebia@inria.fr

Decemeber, 2023

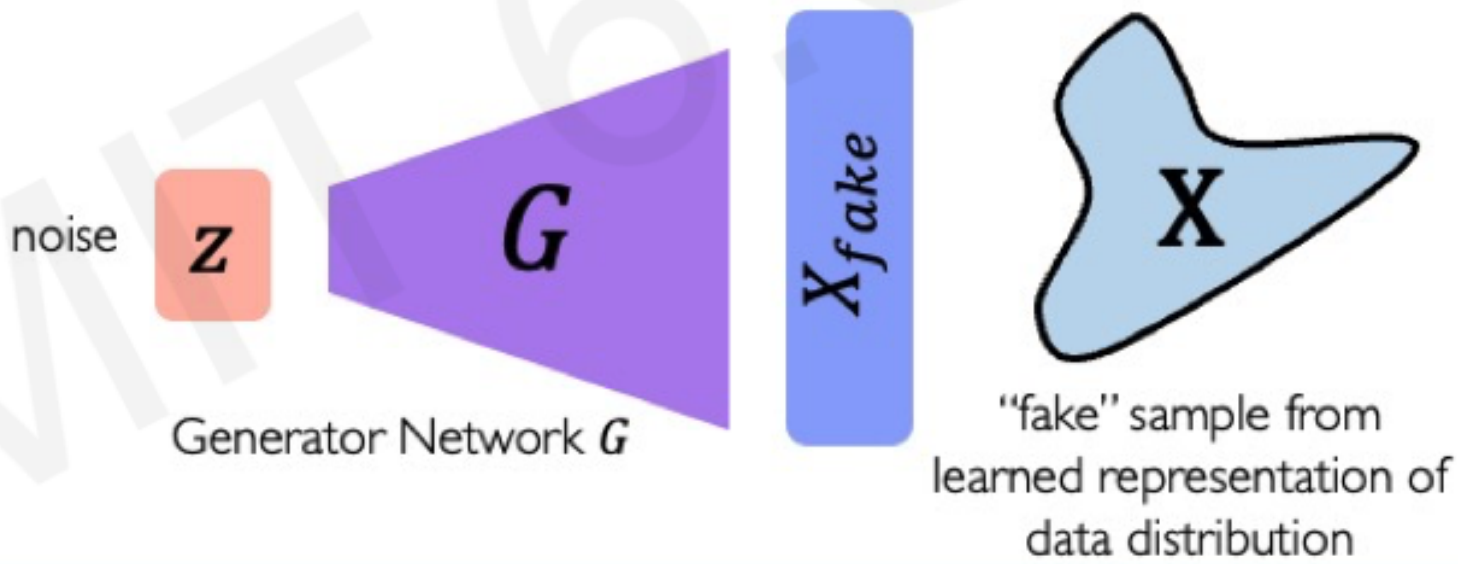
AIstroSight, Inria, Lyon Center

What if we just want to sample ?

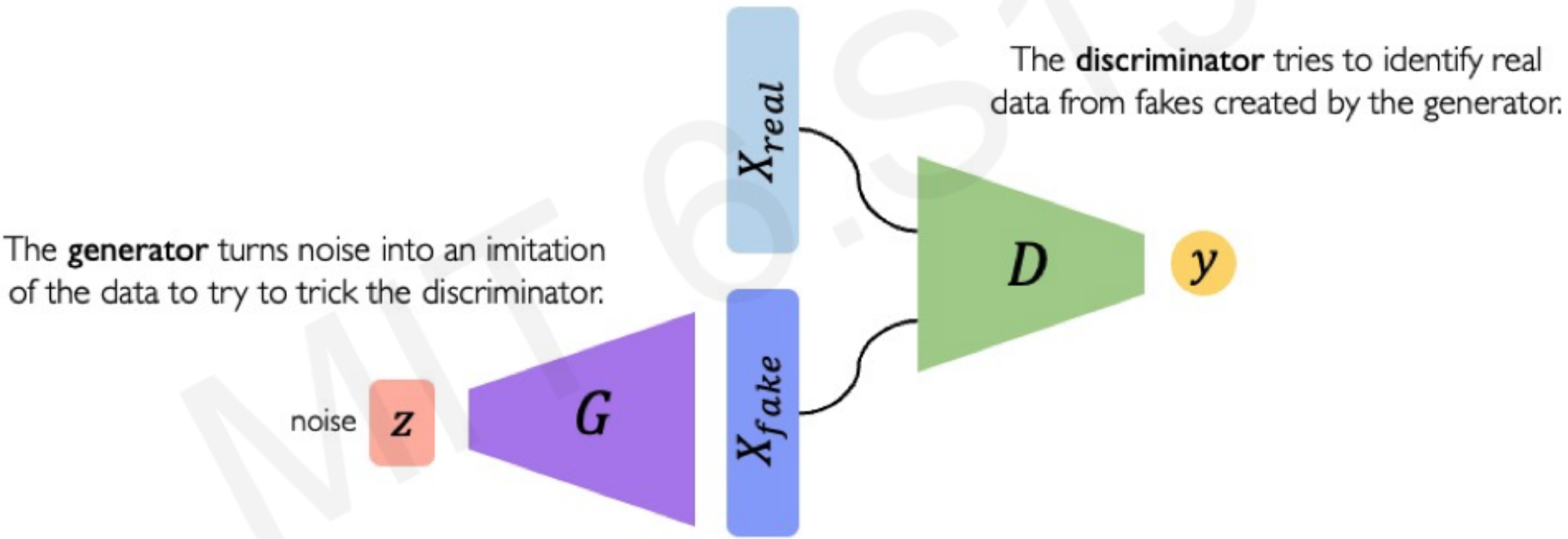
Idea : don't explicitly model density, and just sample to generate new instances

Problem : sample from complex distribution - can' dot this directly !

Solution : sample from something simple, learn a transformation to the data distribution.

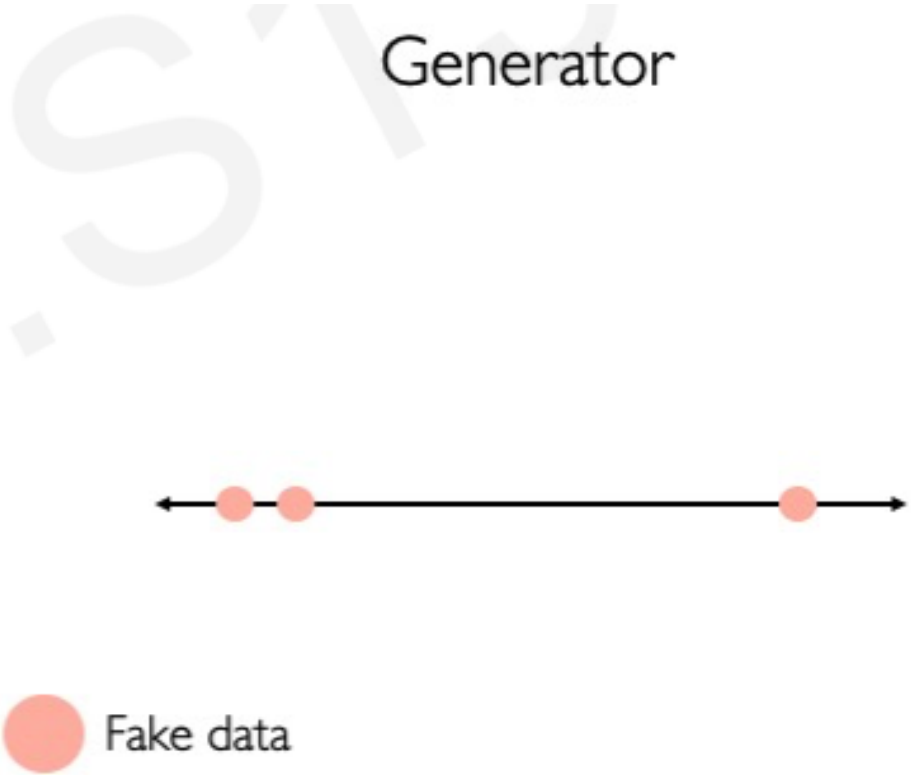


Generative Adversarial Networks (GANs) are a way to make generative model by having two neural networks compete with each other.



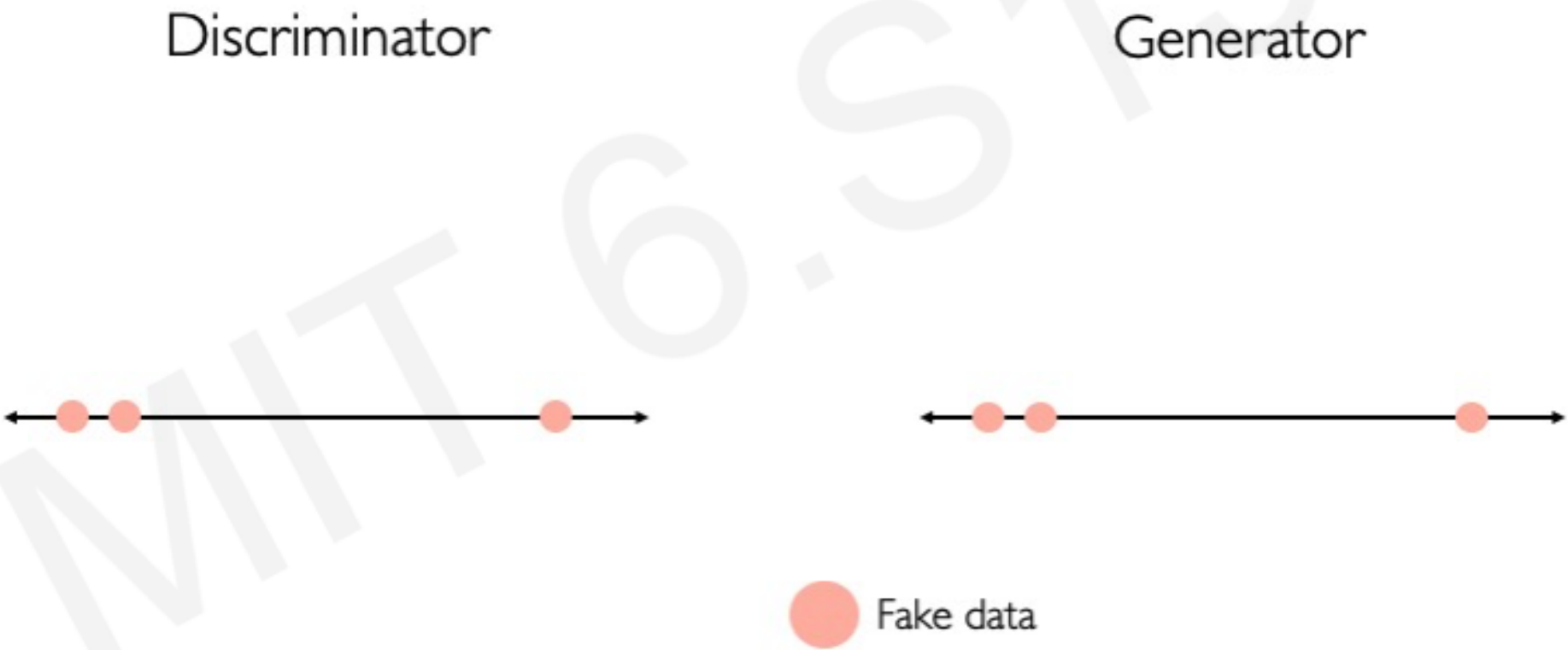
Intuition behind GANs

Generator starts from noise to try to create an imitation of the data



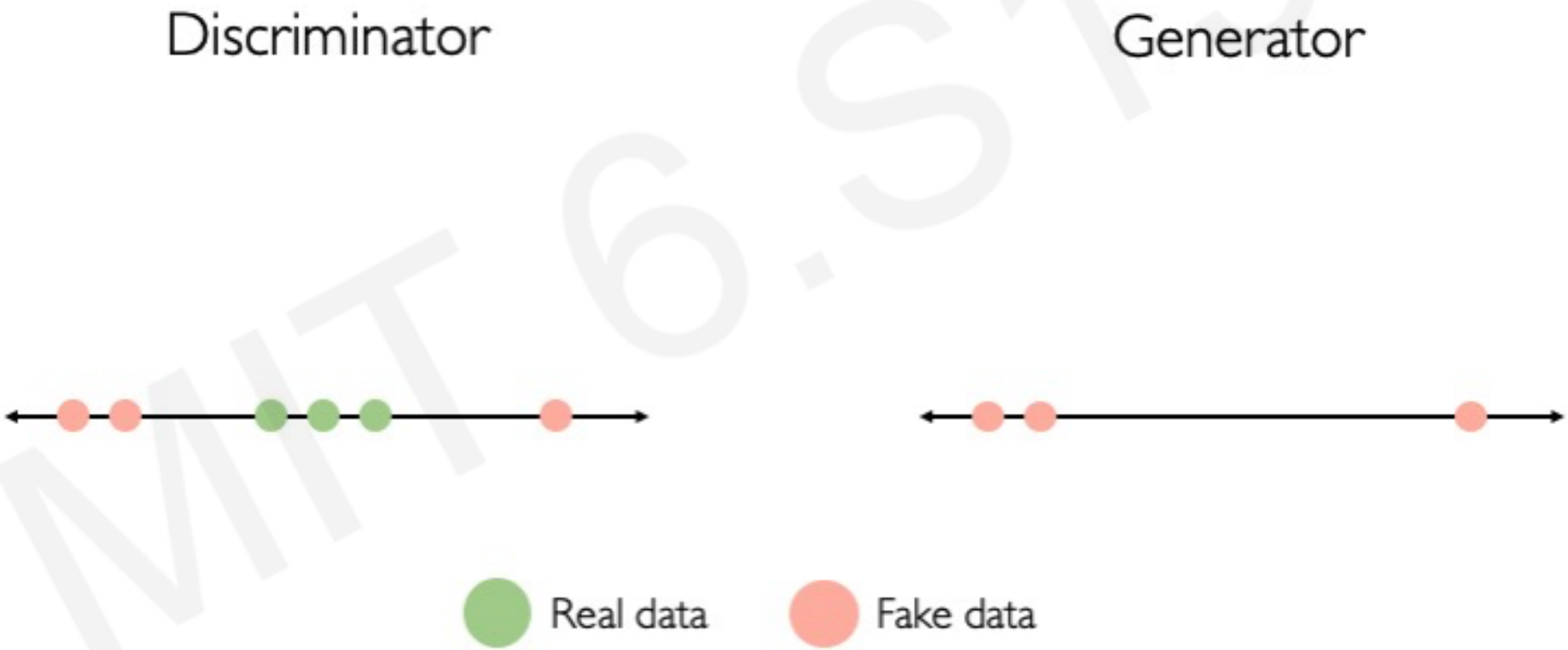
Intuition behind GANs

Discriminator looks at both real data and fake data created by the generator

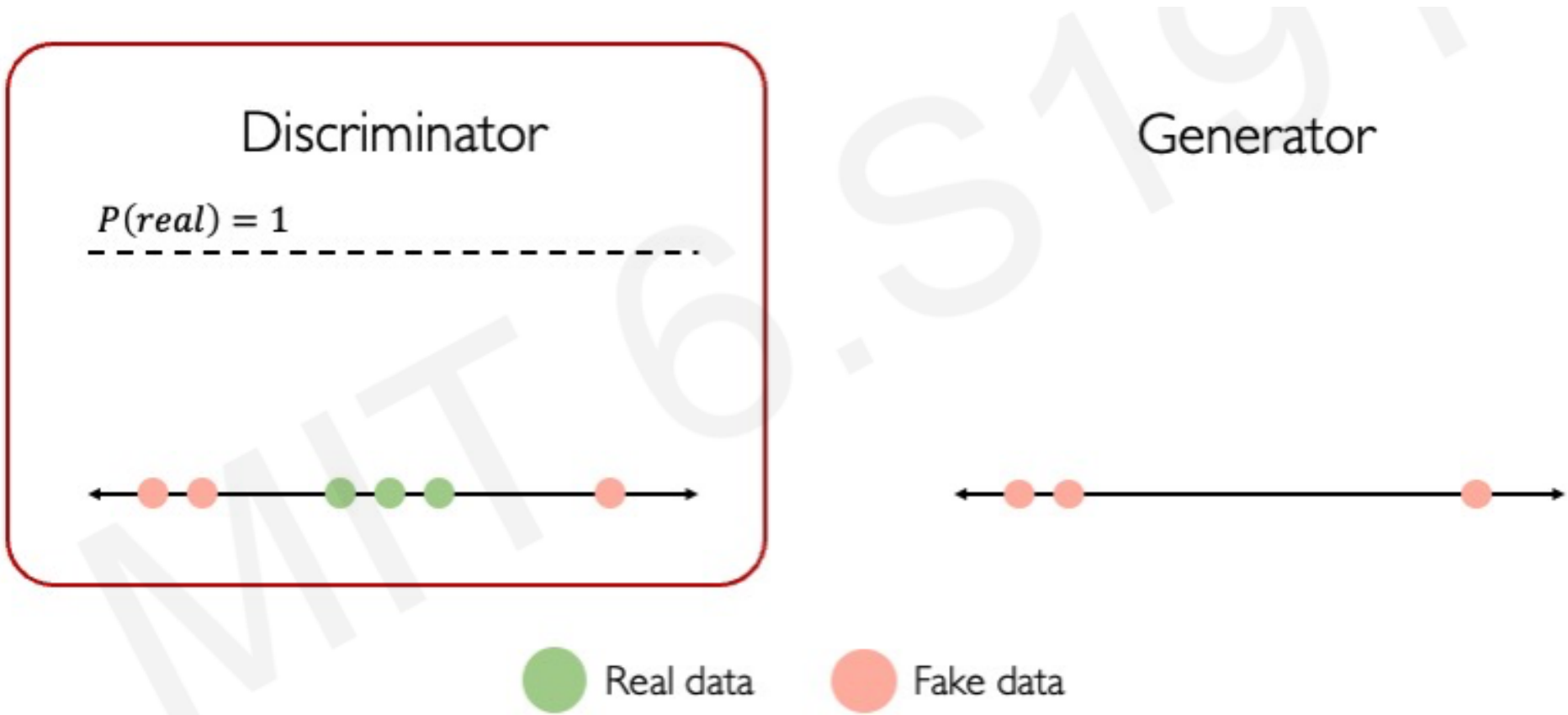


Intuition behind GANs

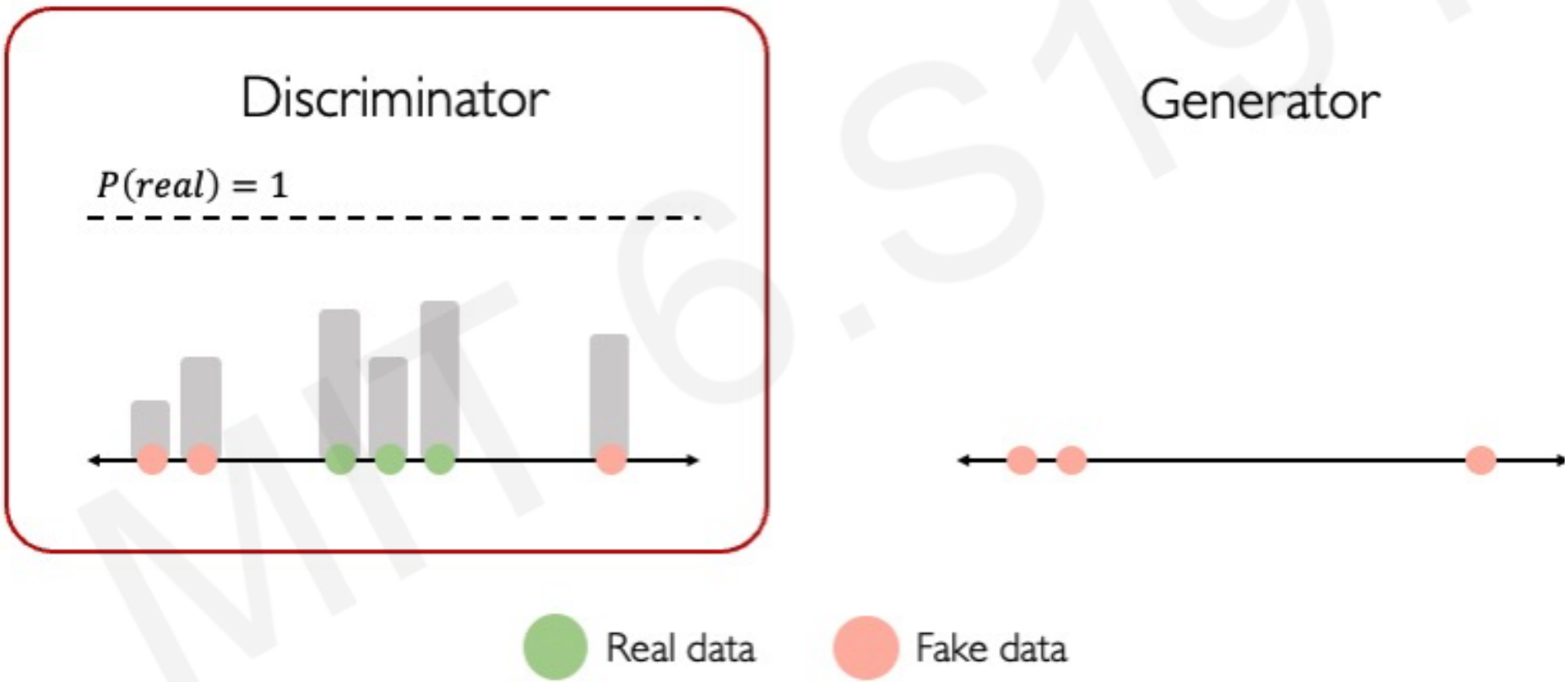
Discriminator tries to predicit what's real and what's fake



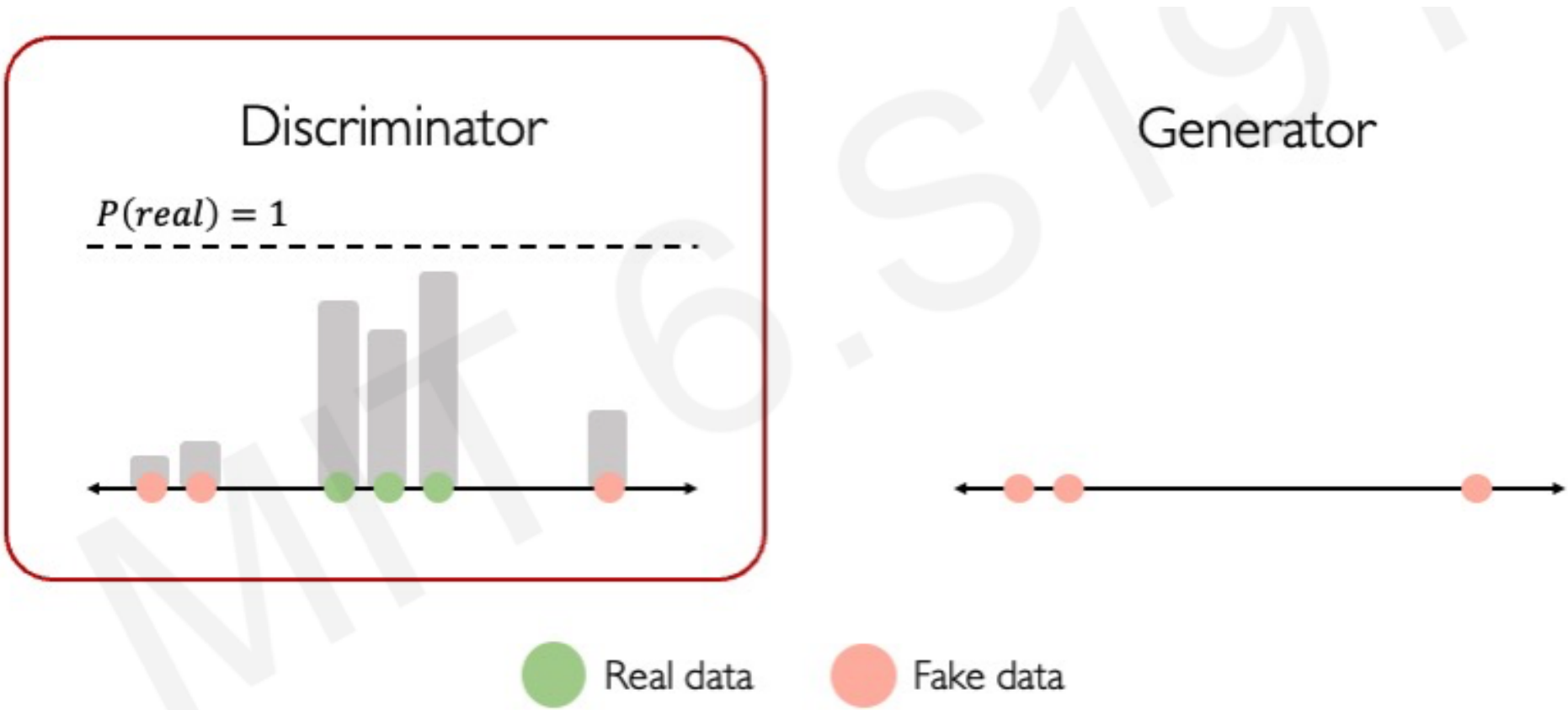
Discriminator tries to predicit what's real and what's fake



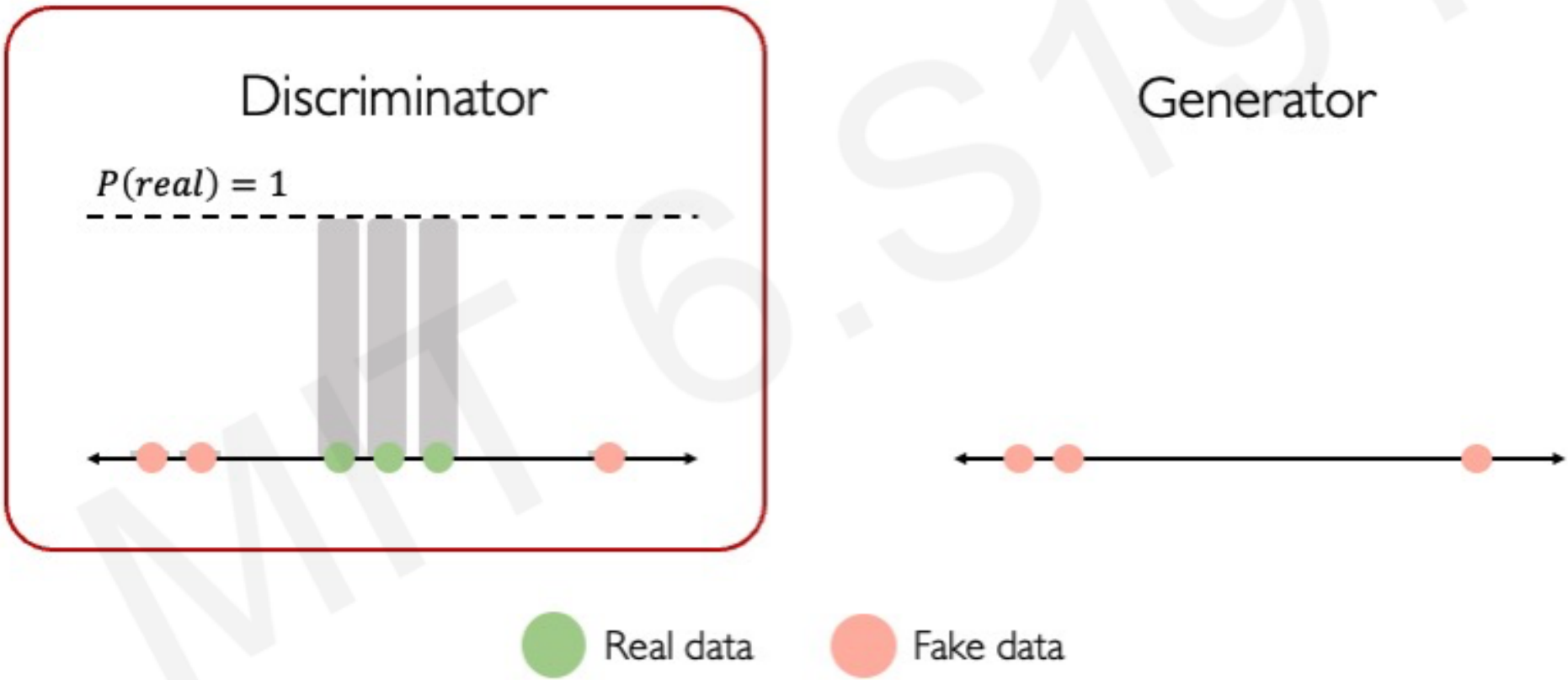
Discriminator tries to predicit what's real and what's fake



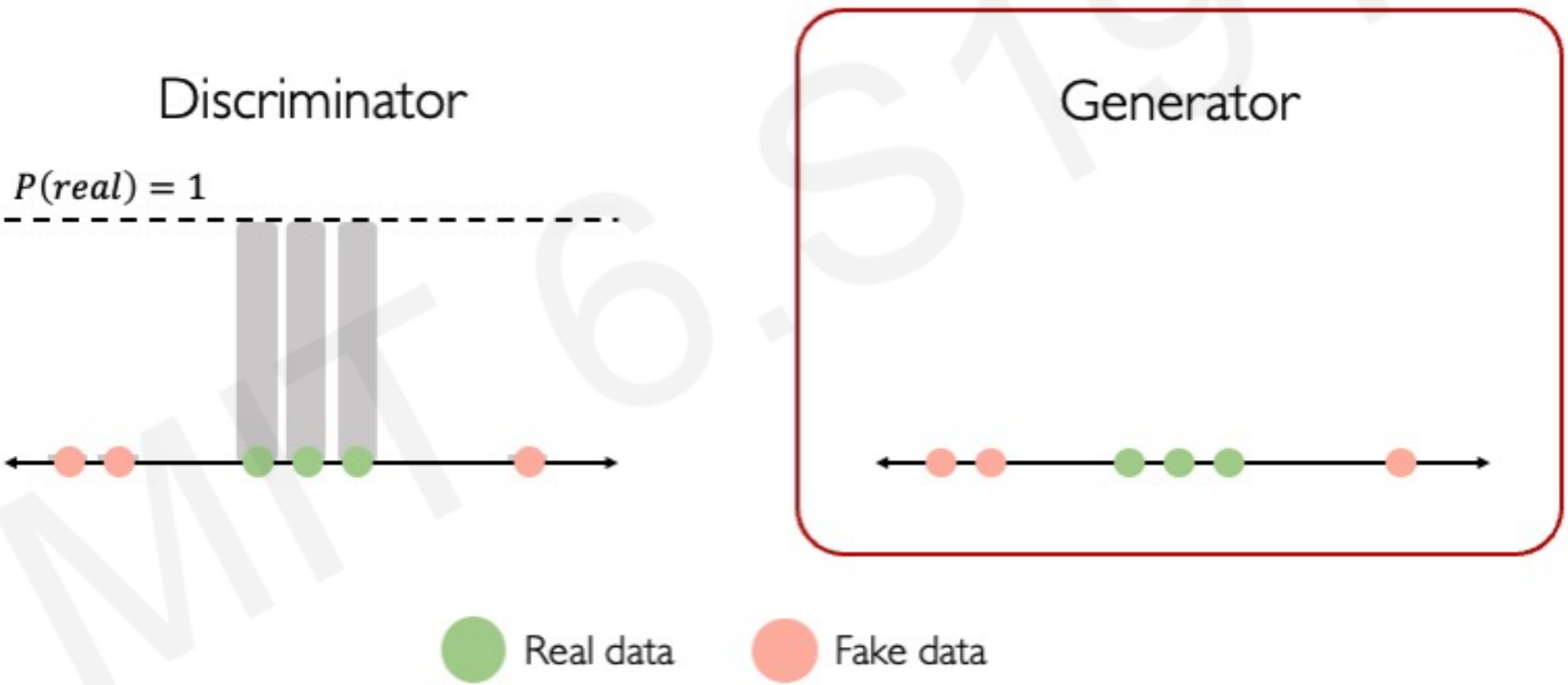
Discriminator tries to predicit what's real and what's fake



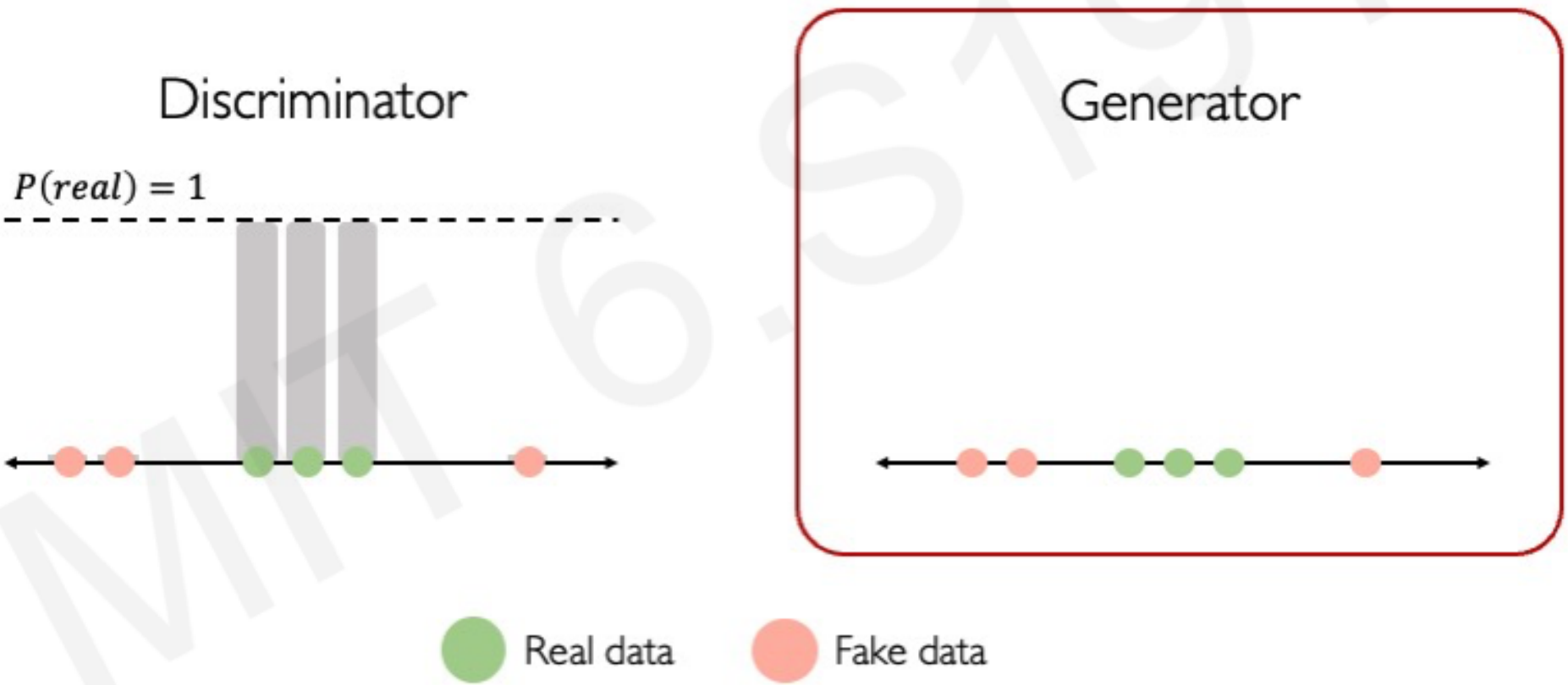
Discriminator tries to predicit what's real and what's fake



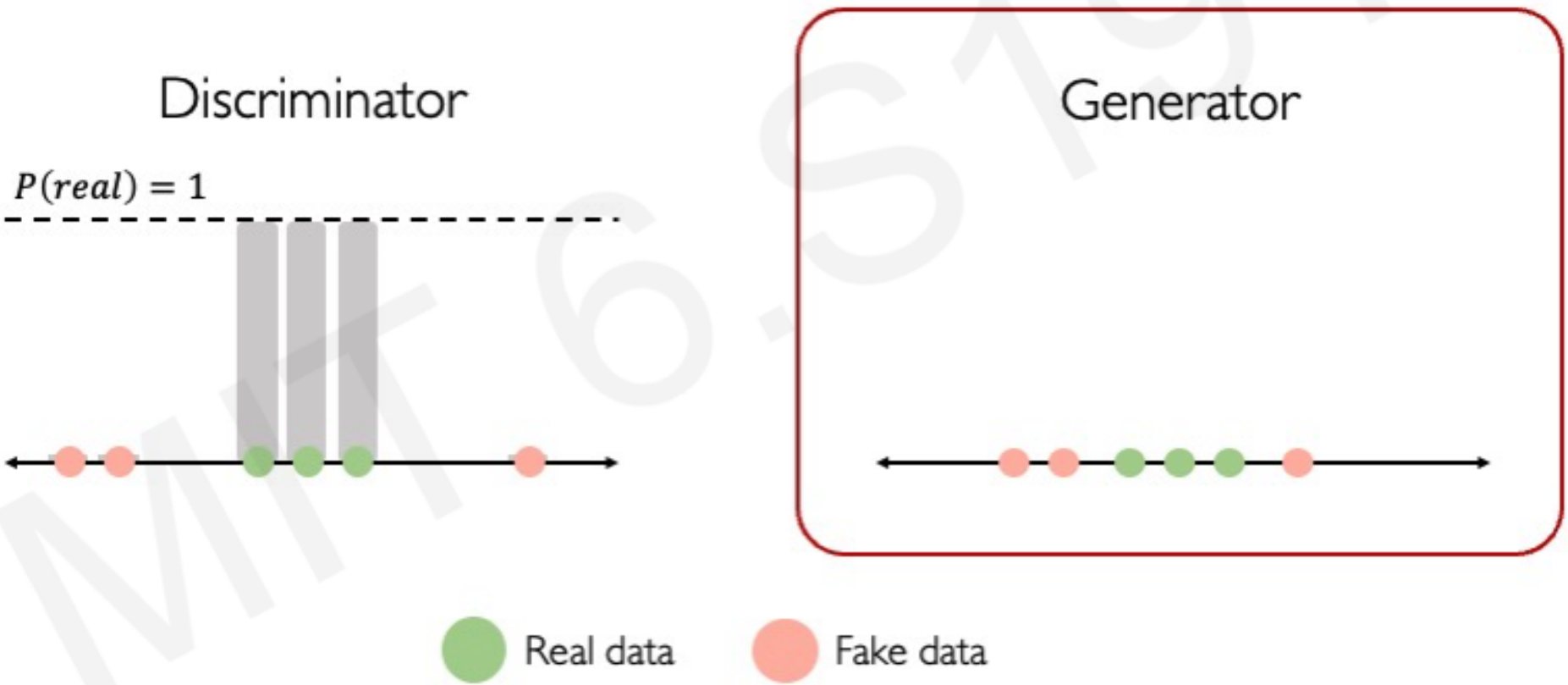
Generator tries to improve its imitation of the data



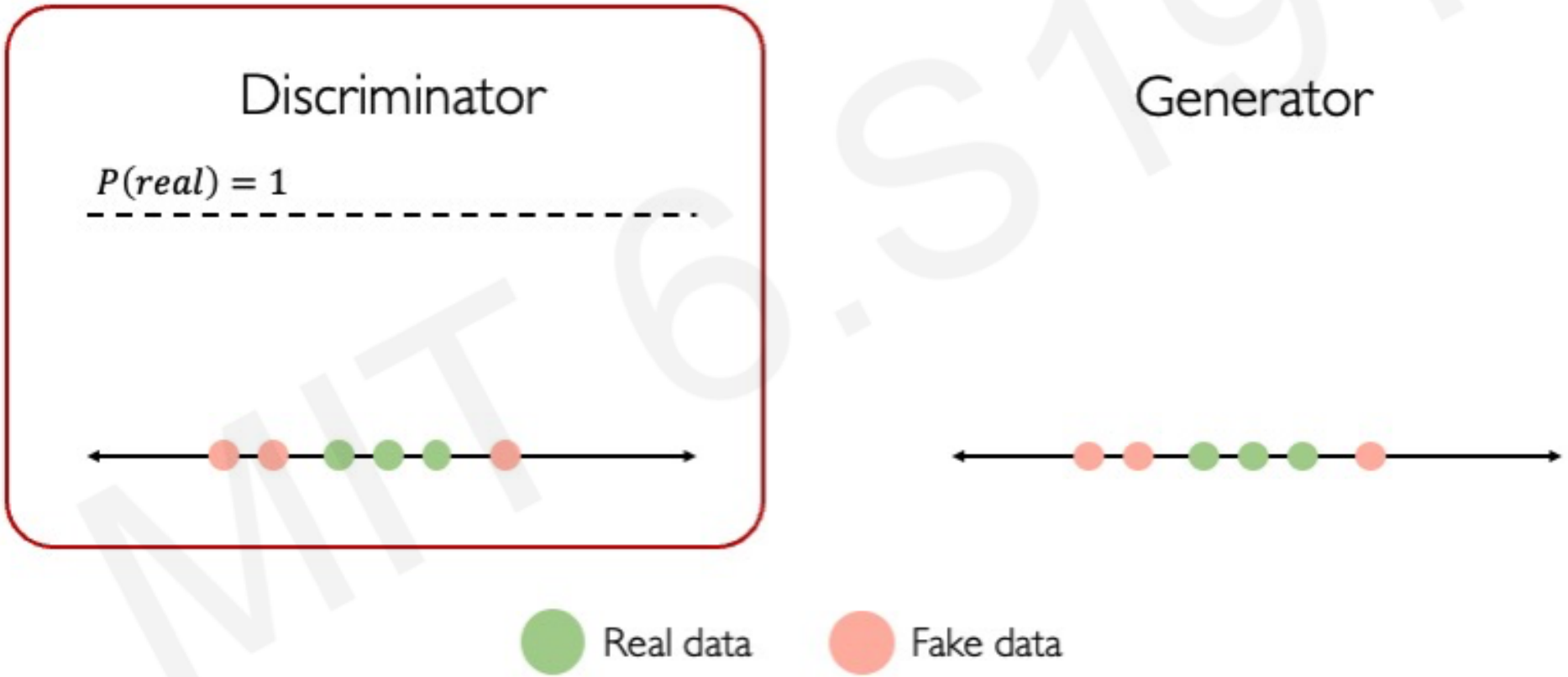
Generator tries to improve its imitation of the data



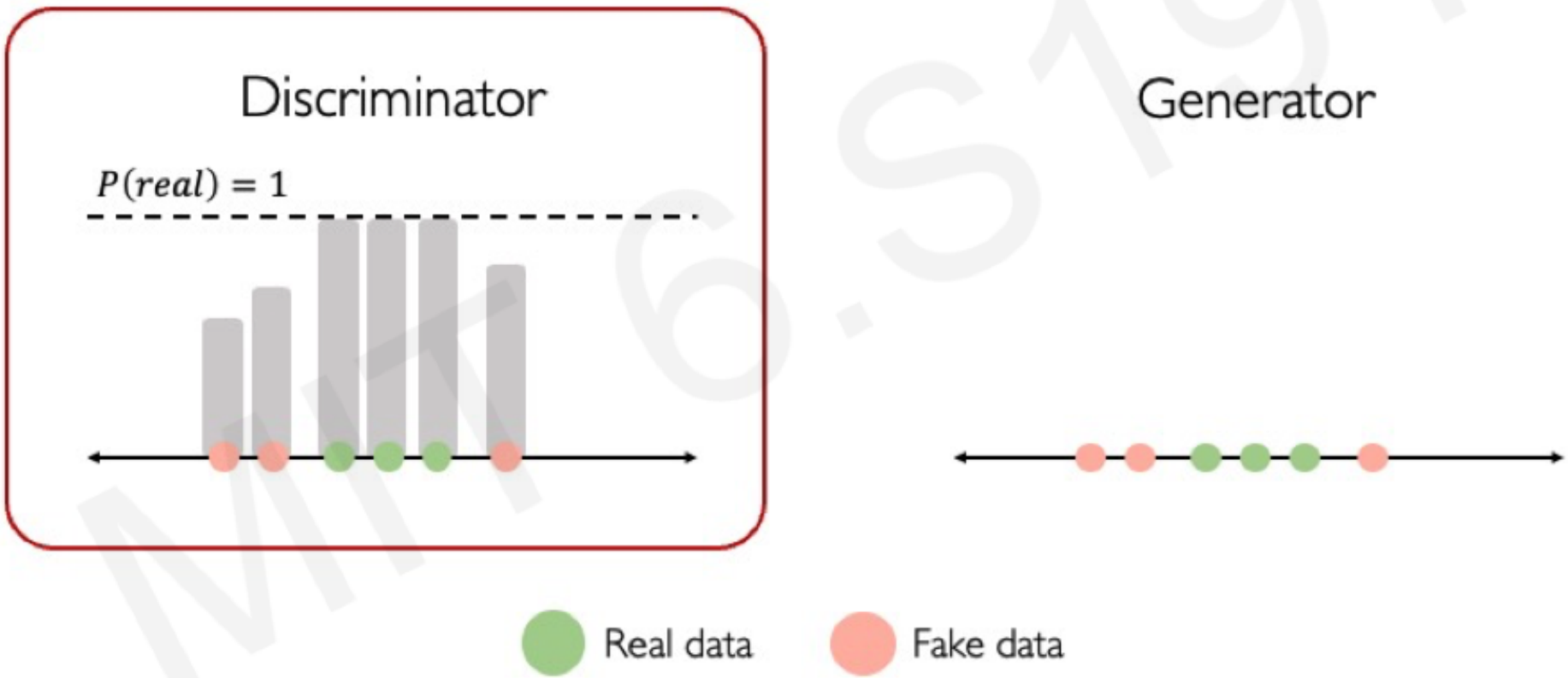
Generator tries to improve its imitation of the data



Discriminator tries to predicit what's real and what's fake

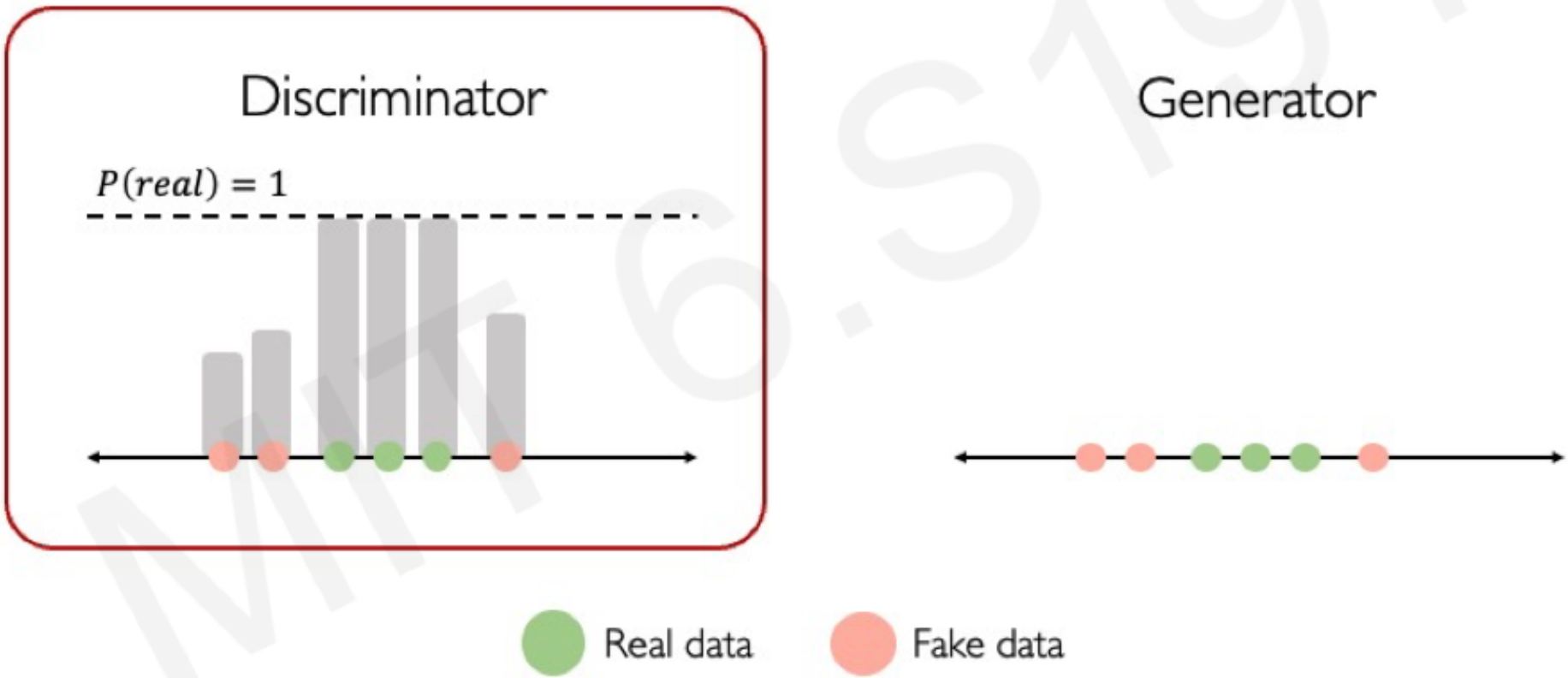


Discriminator tries to predicit what's real and what's fake

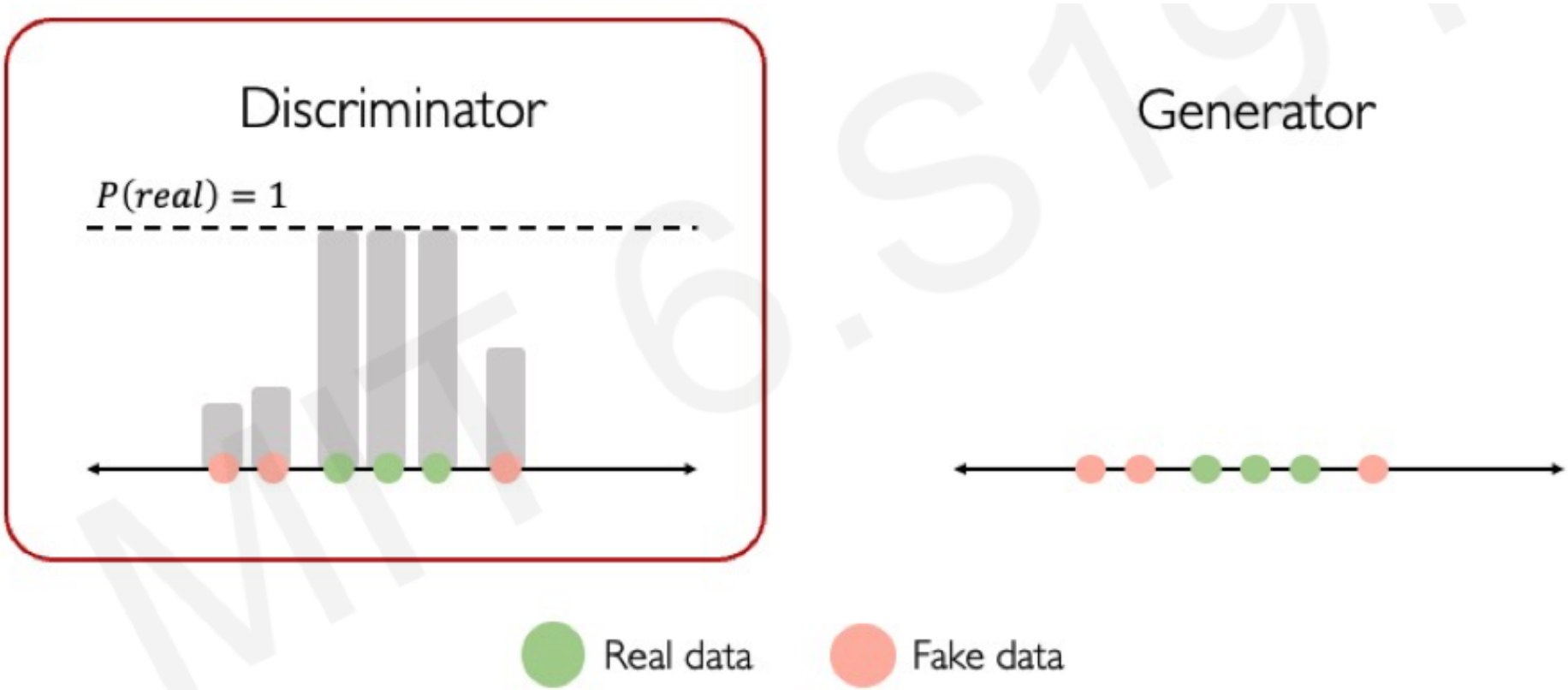


Intuition behind GANs

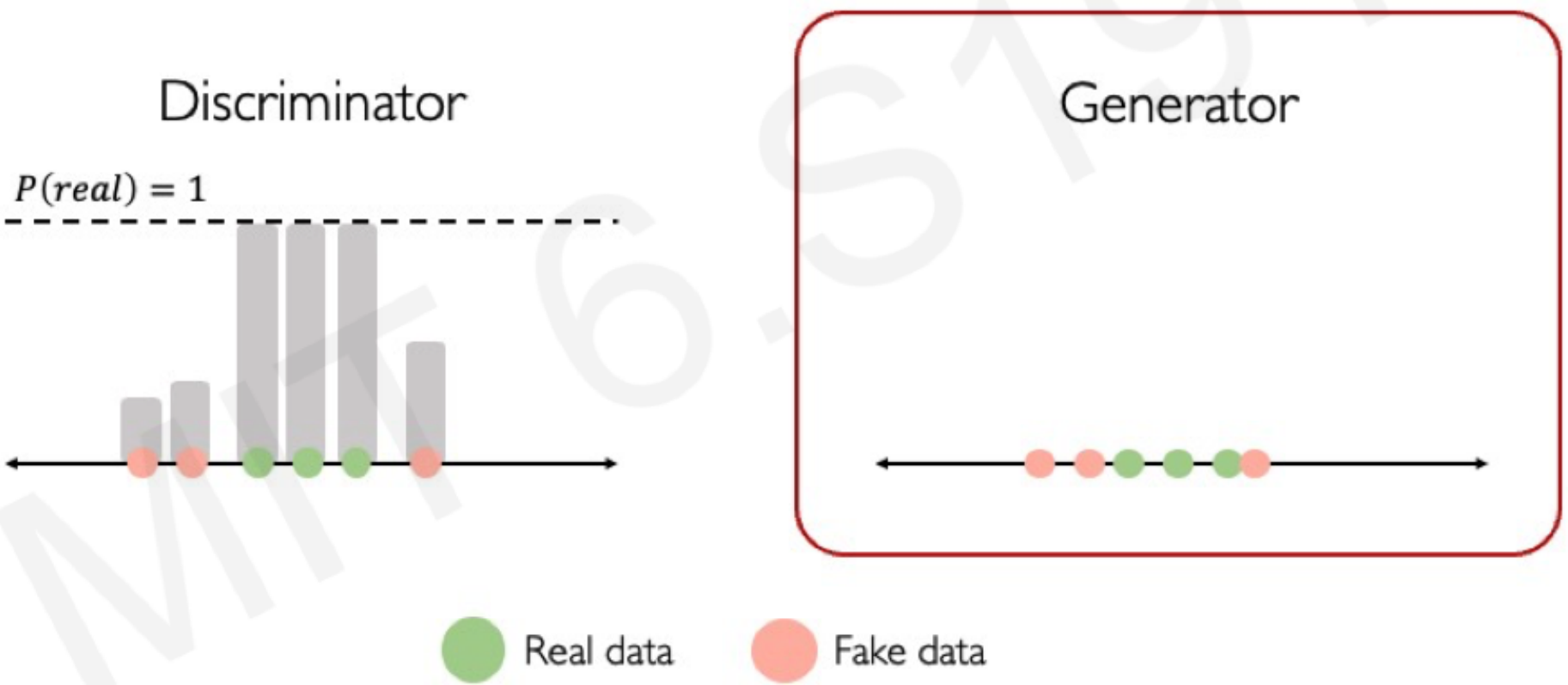
Discriminator tries to predicit what's real and what's fake



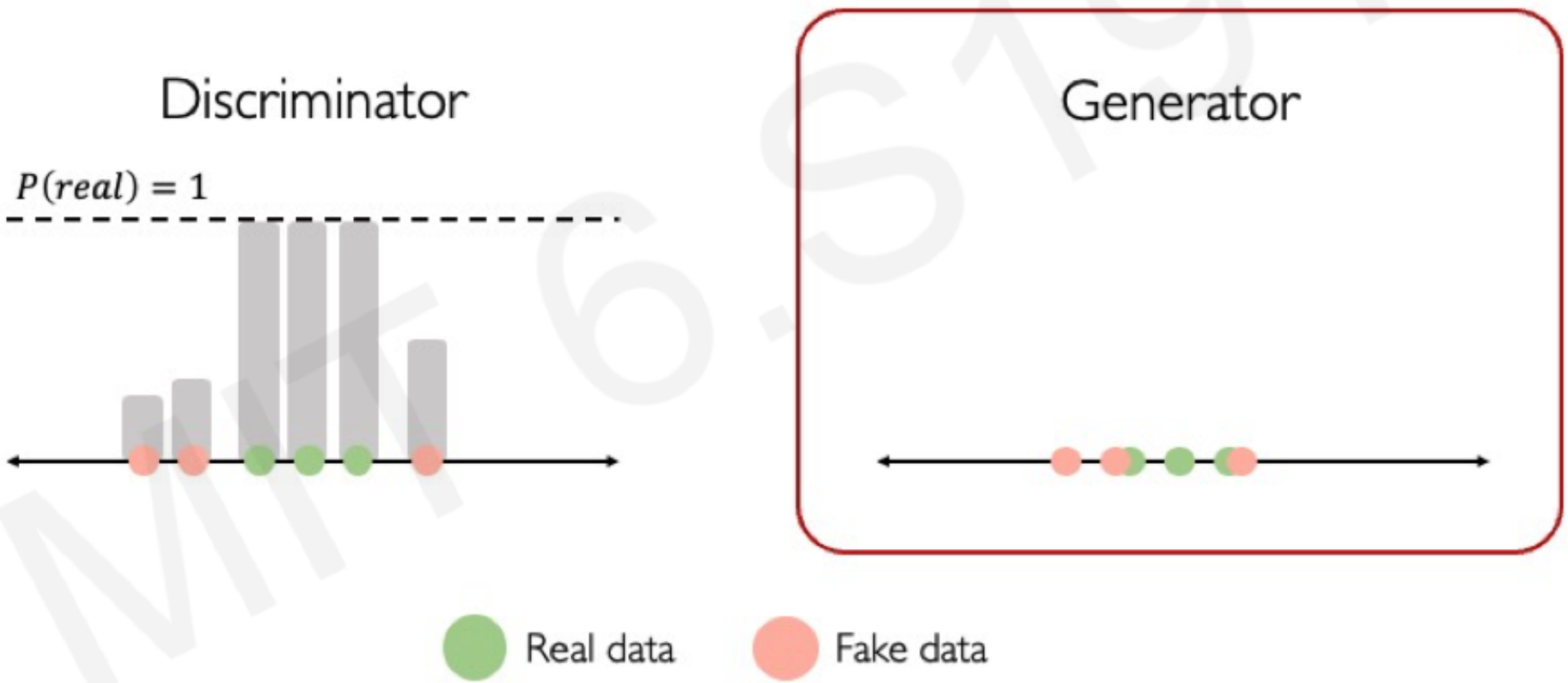
Discriminator tries to predicit what's real and what's fake



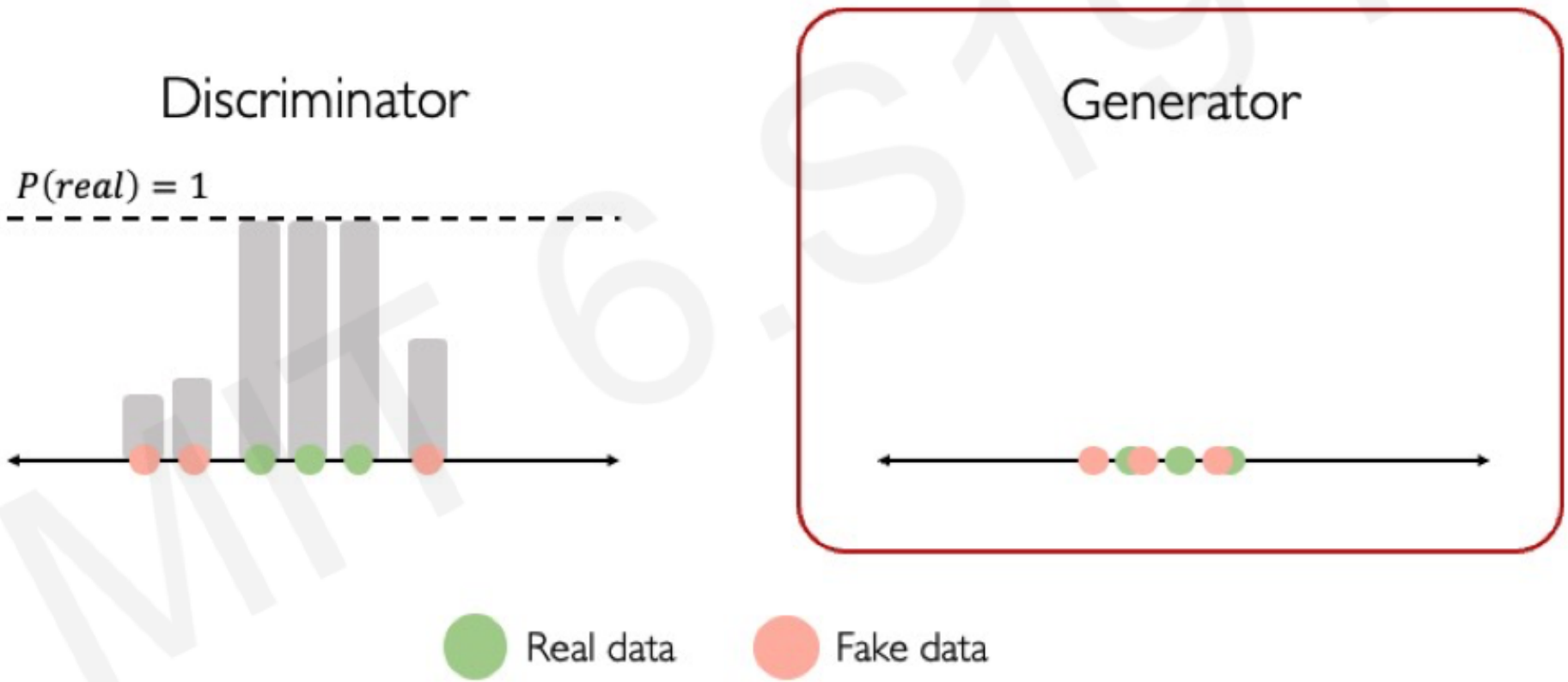
Generator tries to improve its imitation of the data



Generator tries to improve its imitation of the data

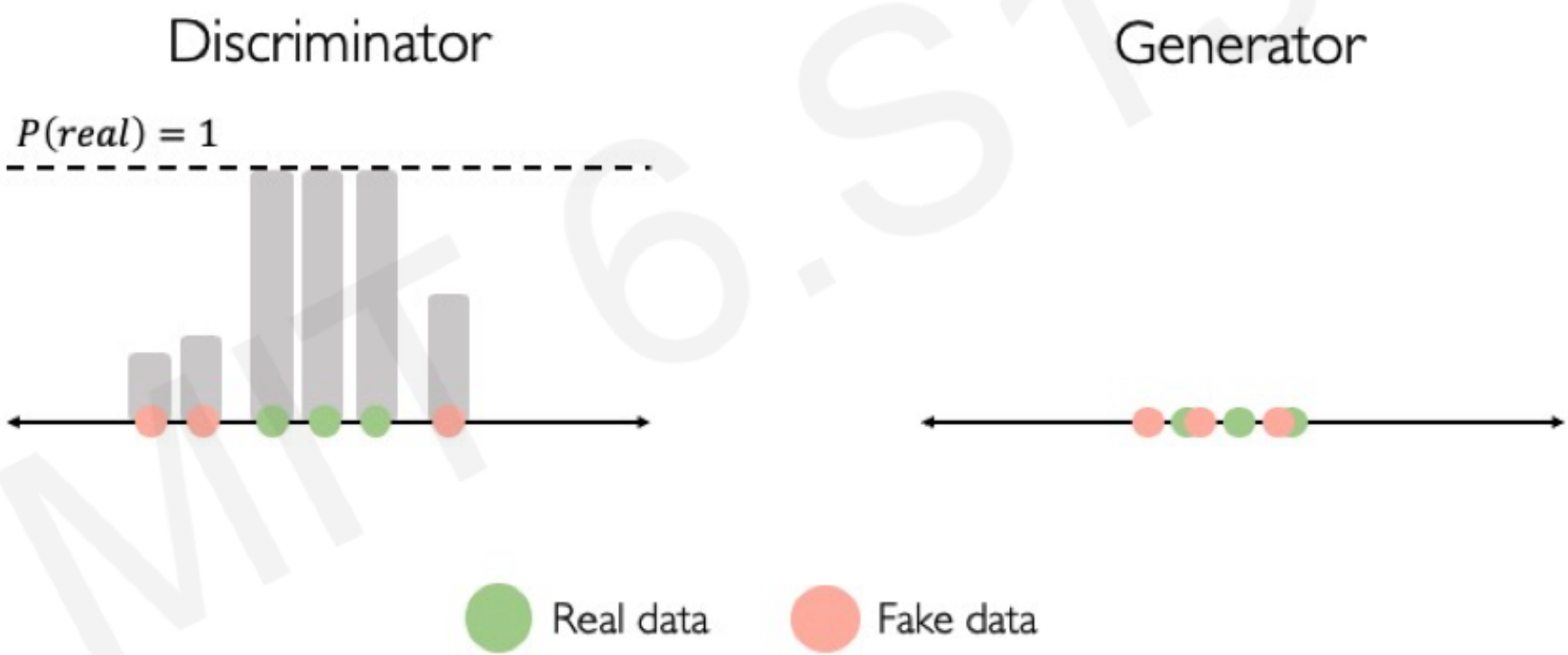


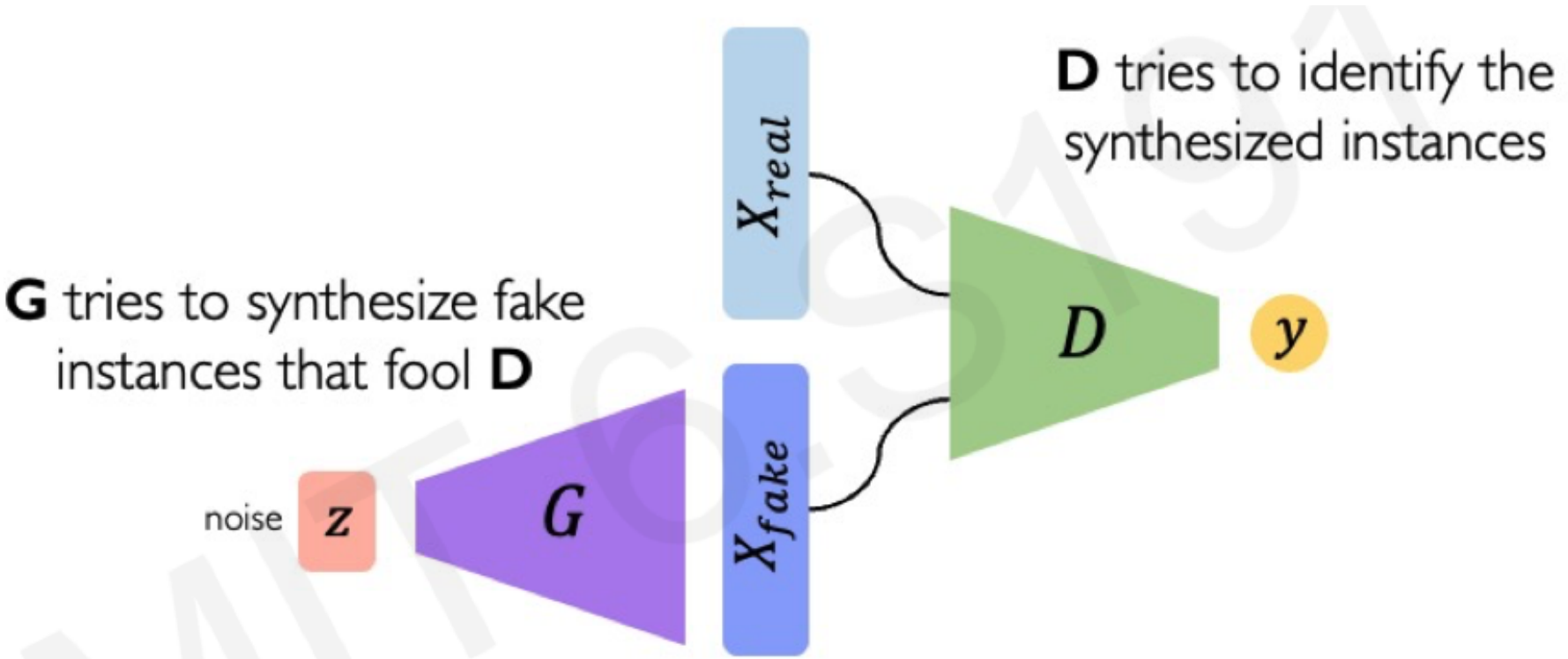
Generator tries to improve its imitation of the data



Intuition behind GANs

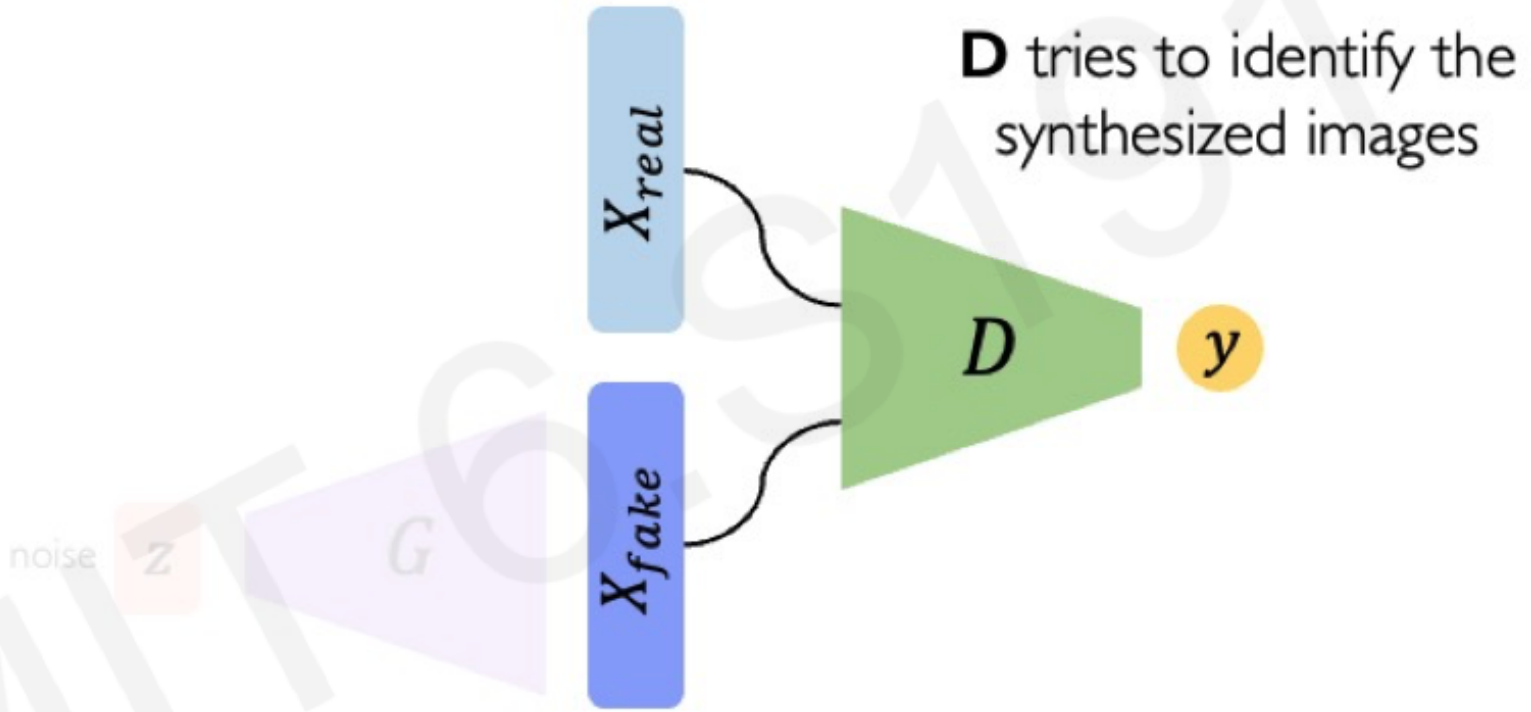
Discriminator tries to identify real data from fakes created by the generator
Generator tries to create imitations of the data to trick the discriminator





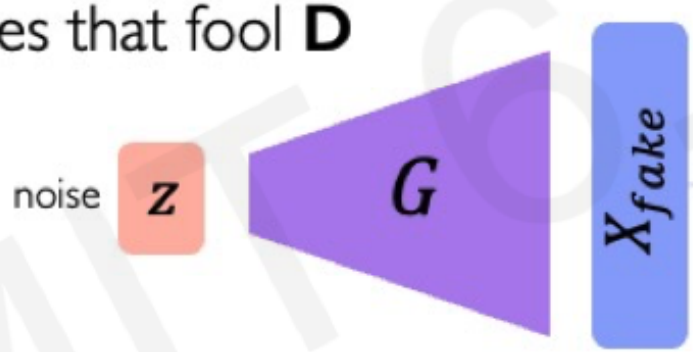
Training: adversarial objectives for **D** and **G**
Global Optimum: **G** reproduces the true data distribution

Training GANs : loss function



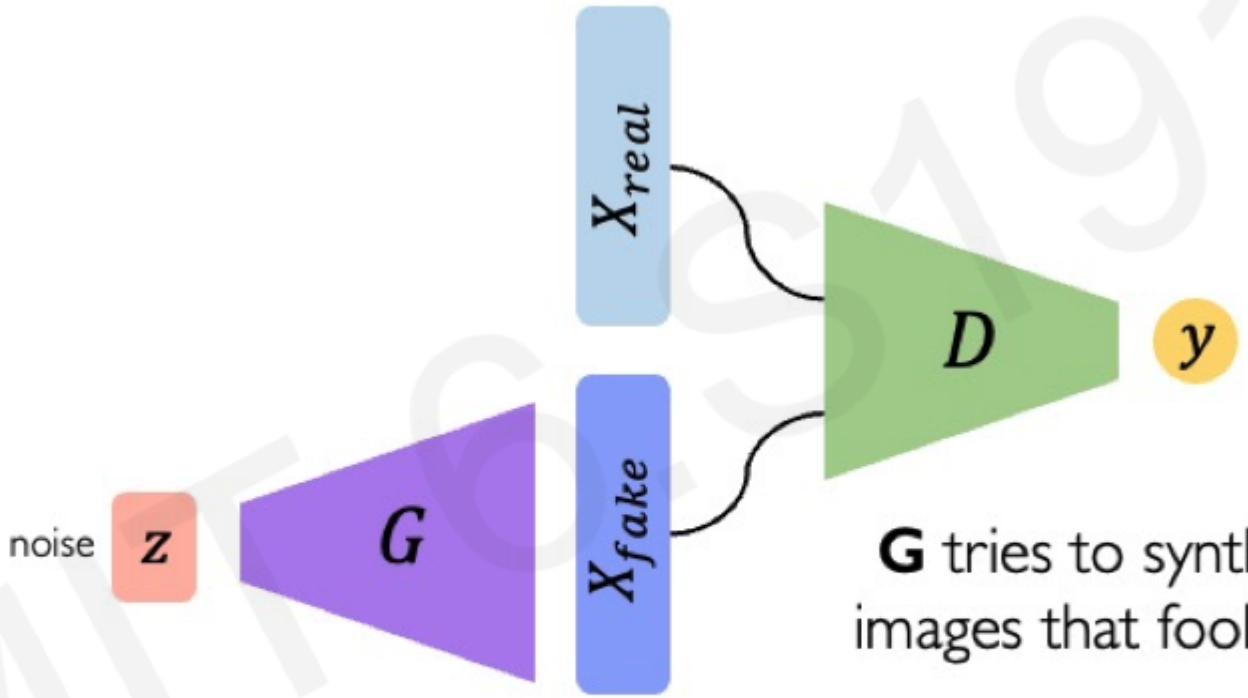
$$\arg\max_{D} \mathbb{E}_{\mathbf{z},\mathbf{x}}[\underbrace{\log D(G(\mathbf{z}))}_{\text{Fake}} + \underbrace{\log(1 - D(\mathbf{x}))}_{\text{Real}}]$$

G tries to synthesize fake images that fool **D**



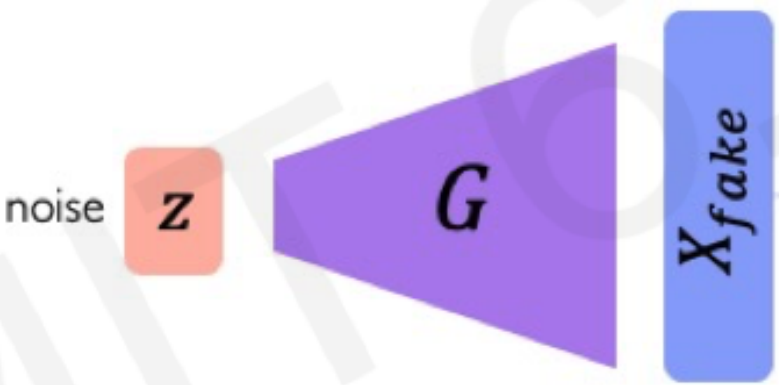
$$\arg\min_{G} \mathbb{E}_{\mathbf{z},\mathbf{x}}[\log D(G(\mathbf{z})) + \log(1 - D(\mathbf{x}))]$$

Training GANs : loss function



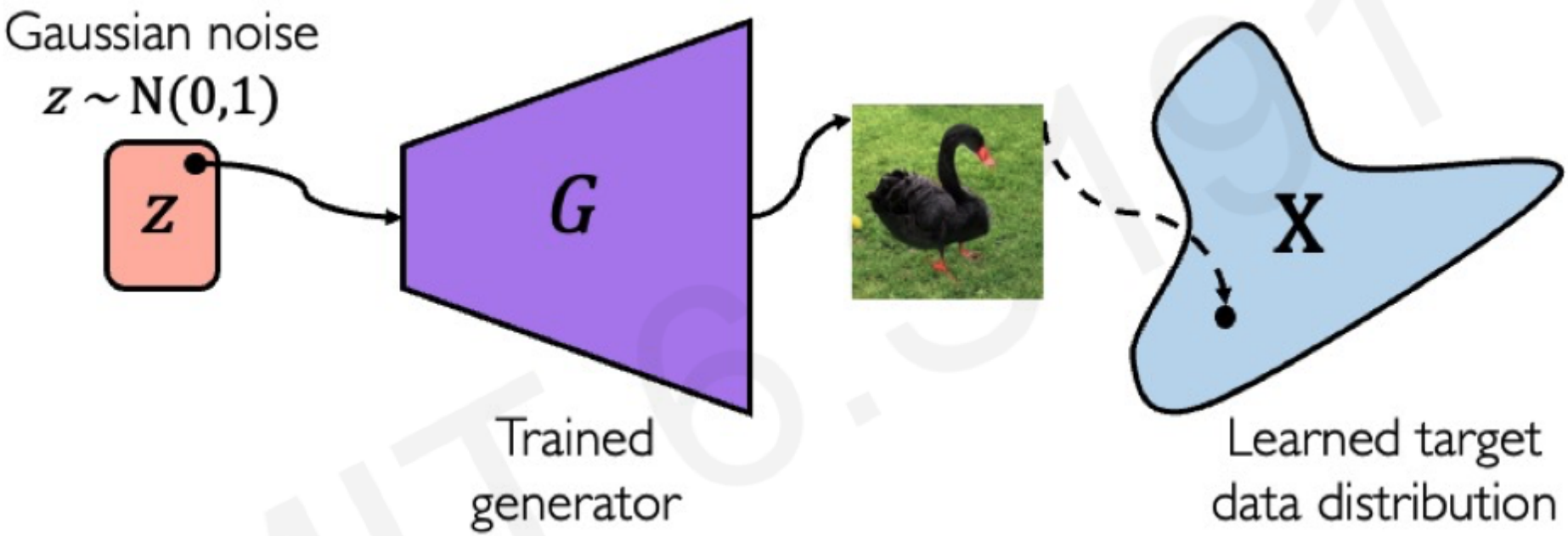
$$\arg\min_{G}\max_{D} \mathbb{E}_{\mathbf{z},\mathbf{x}}[\ \log D(G(\mathbf{z})) + \log\left(1 - D(\mathbf{x})\right)\ ]$$

Generating new data with GANs

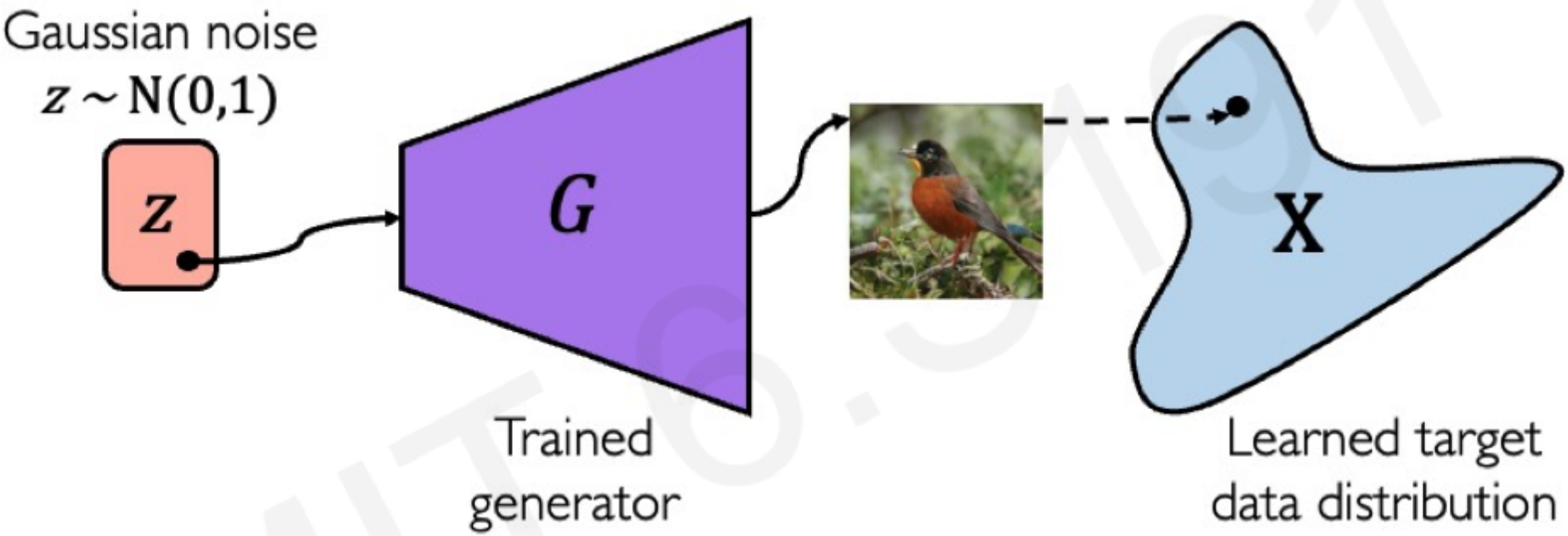


After training , use the generator network to create **new data** that's never been seen before

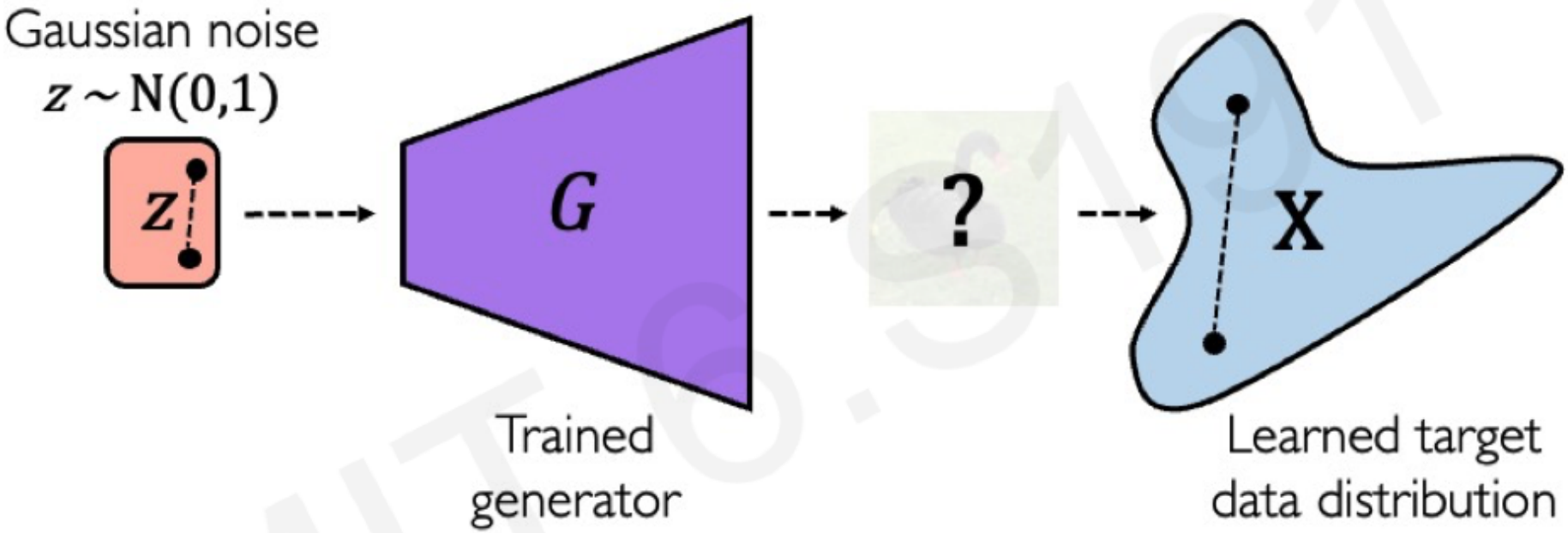
GANS are distribution transformers



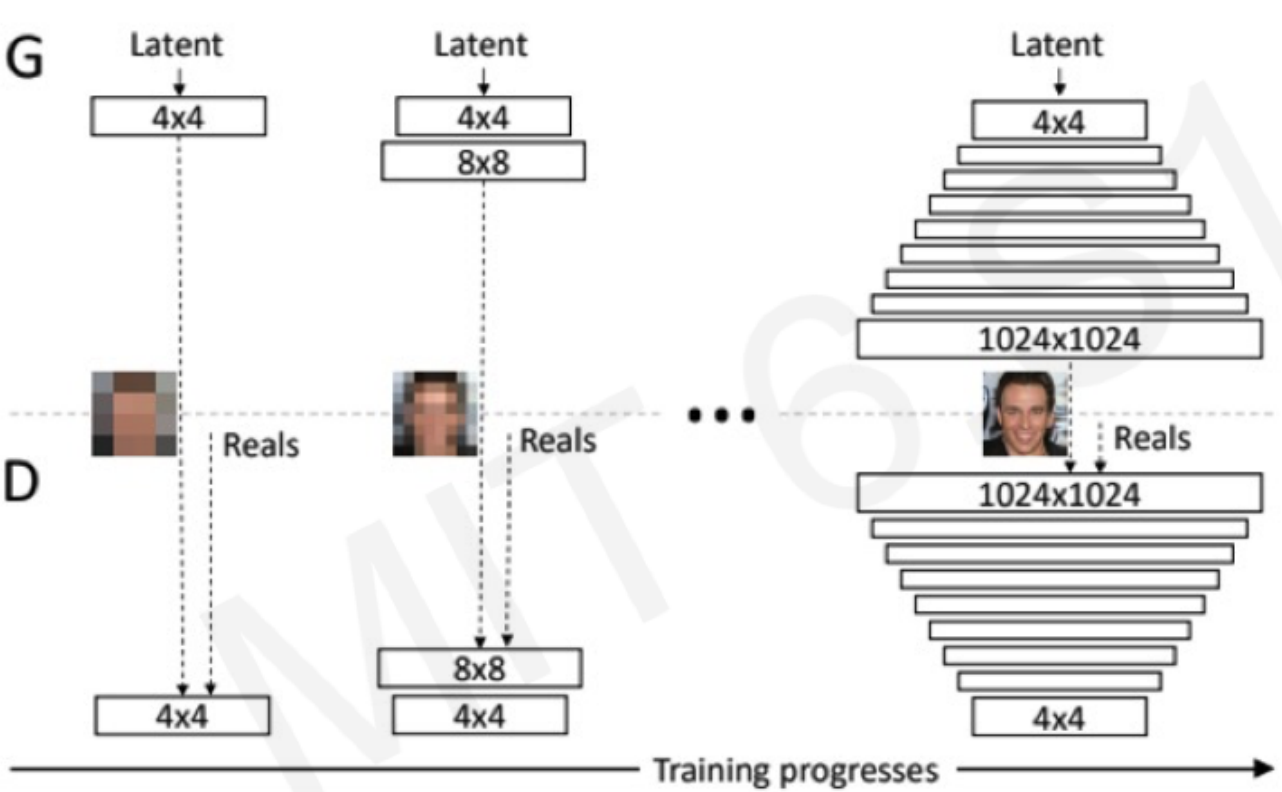
GANS are distribution transformers

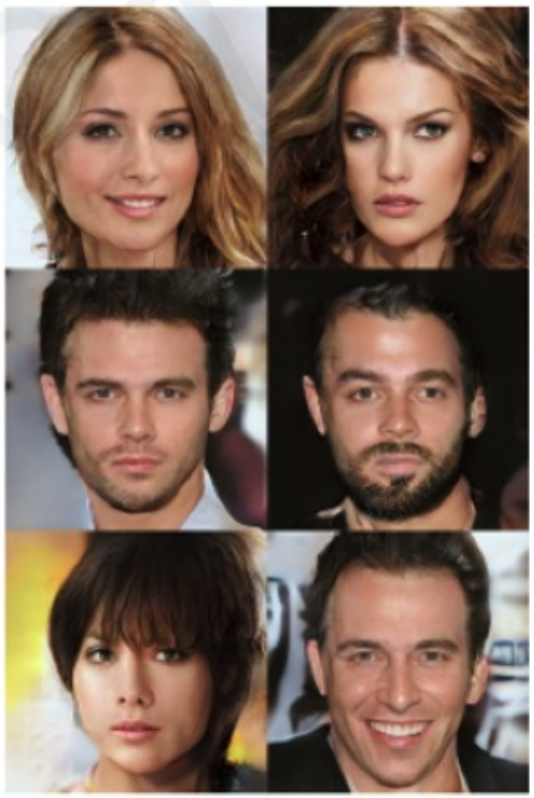


GANS are distribution transformers



Progressive growing of GANs

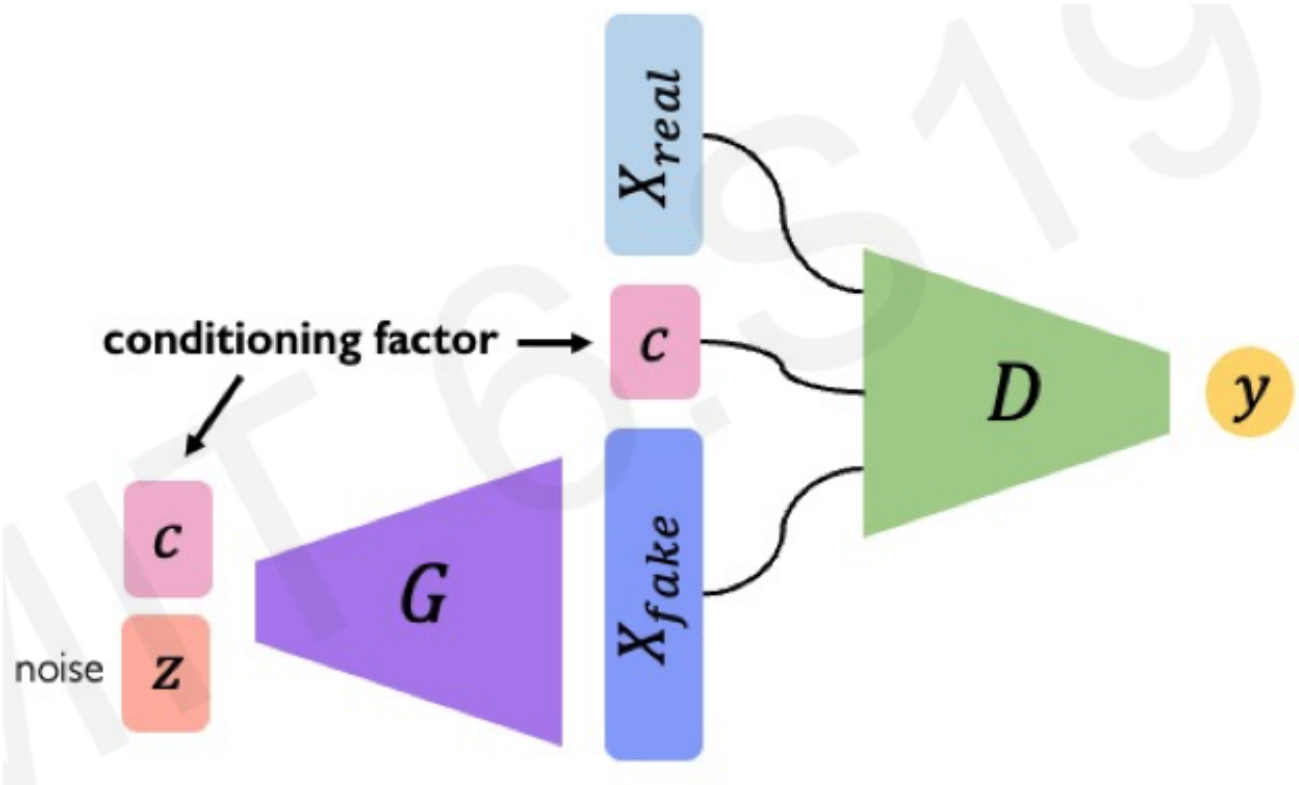




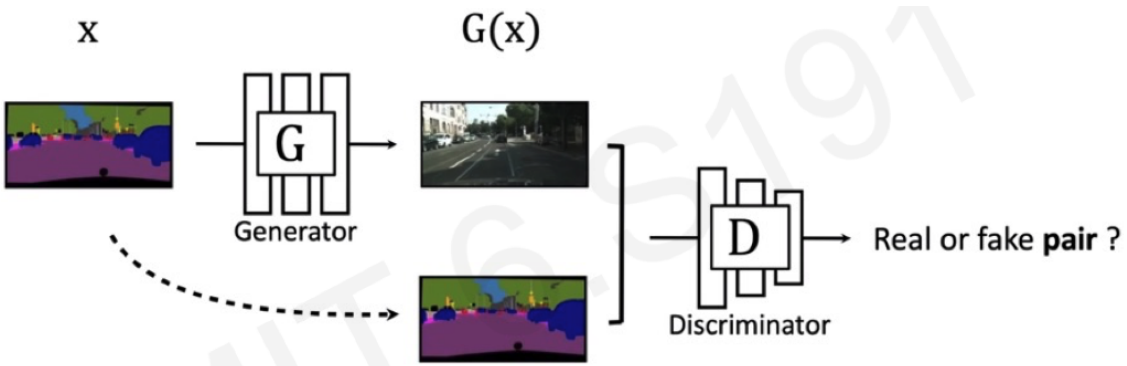
Progressive growing of GANs : results

Conditional GANs

What if we want to control the nature of the output, by conditioning on a label ?



Pix2pix: Paired translation

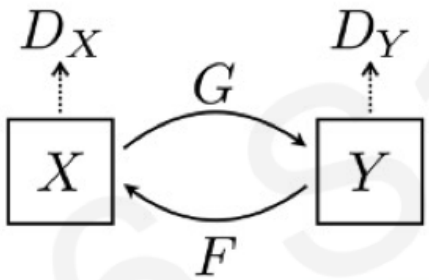


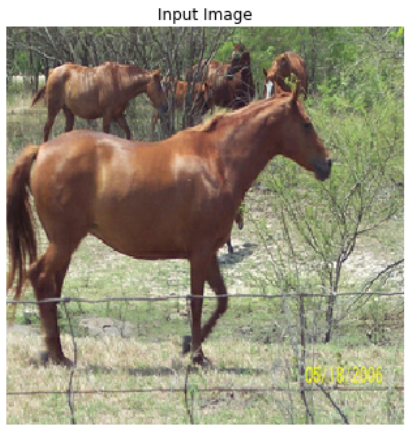
The discriminator, D, classifies between fake and real **pairs**.
The generator, G, learns to foo the discriminator.

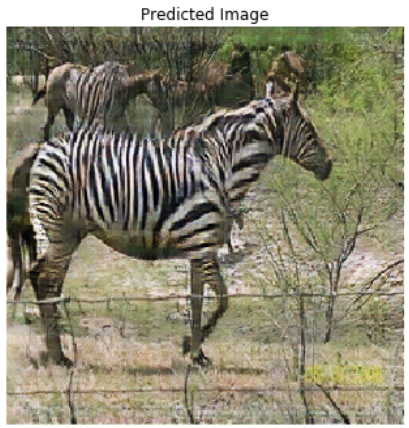
Paired Translation : results

CycleGAN: domain transformation

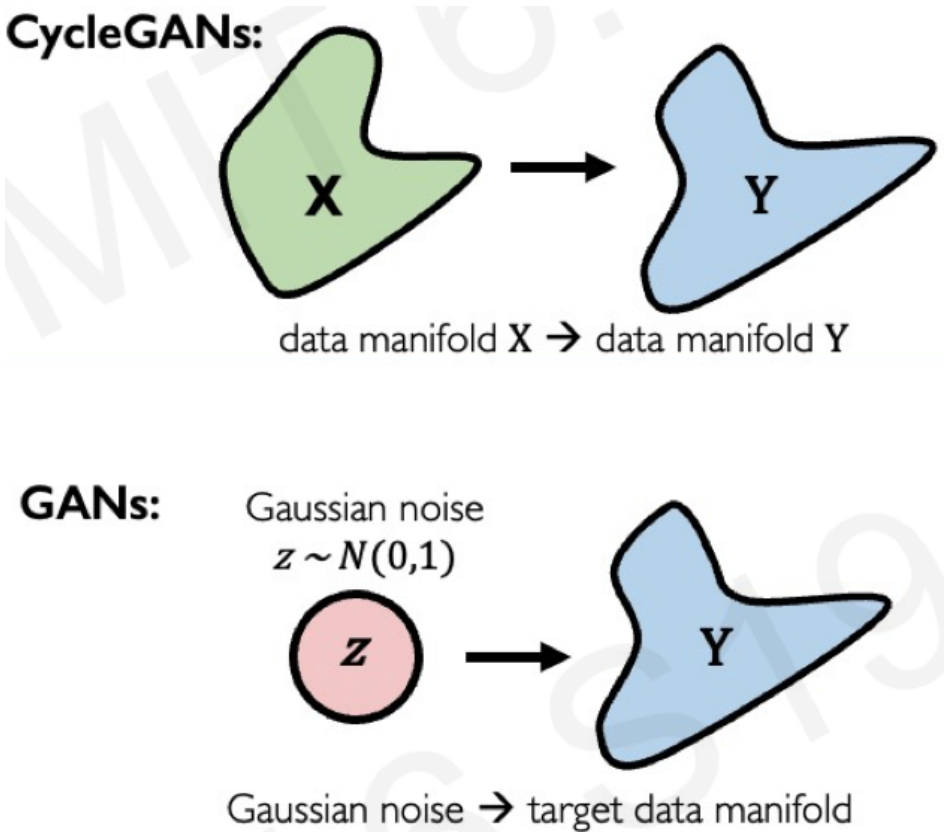
CycleGAN learns transformations across domains with unpaired data



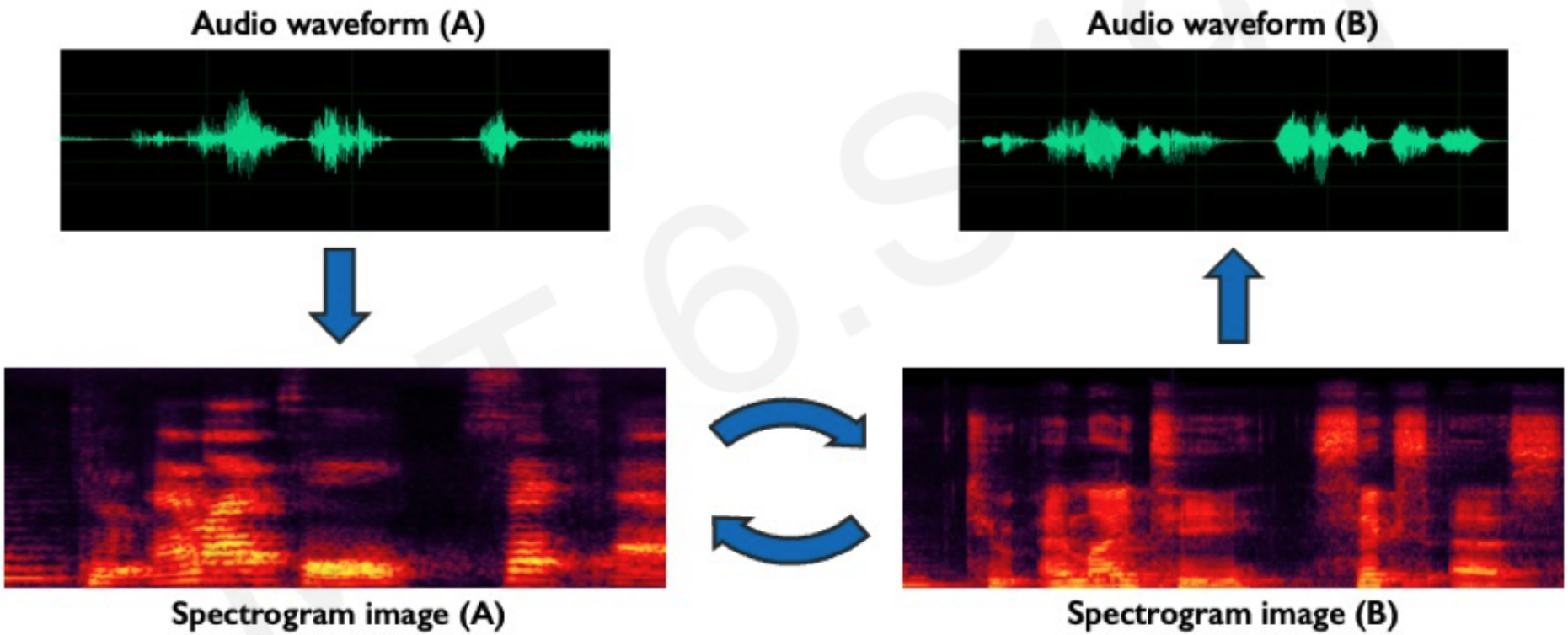




Distribution Transformation

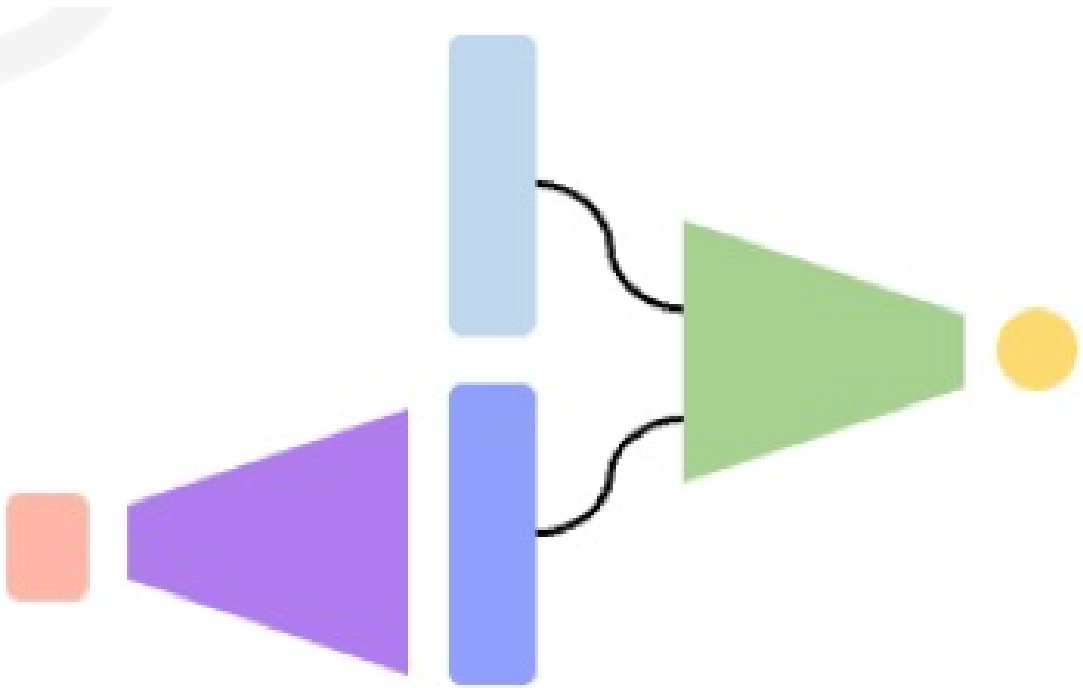


CycleGAN : Transforming Speech



GAN summary

1. Competing generator and discriminator networks
2. Both networks can be trained using only backpropagation
3. Unsupervised training (no need for labels!)
4. Generating data that looks similar to original data
5. Hard to train



Trilemma for GANs

- **High-fidelity samples** : sample quality is better (less blurry) than for VAE !
- **Low diversity samples** : also called **mode collapse**, this means the generator only learns to create a subset specialized in fooling discriminator.
- **Fast Sampling** : require just one pass of the generator network to turn noise into a realistic data sample