Carnegie Mellon University

GENERATIVE ADVERSARIAL NETWORKS - PART 1

11785- Introduction to Deep Learning

AKSHAT GUPTA

Slides Inspired by Benjamin Striner

"This (GANS), and the variations that are now being proposed is the most interesting idea in the last 10 years in ML, in my opinion"

–Yann LeCun

Carnegie Mellon University

Video: https://www.youtube.com/watch?v=QiiSAvKJIHo

CONTENTS

Carnegie Mellon University

- Motivation
- Generative vs Discriminative Models
- GANs vs VAEs
- GANs Introduction
- GANs Theory
- GANs Evaluation

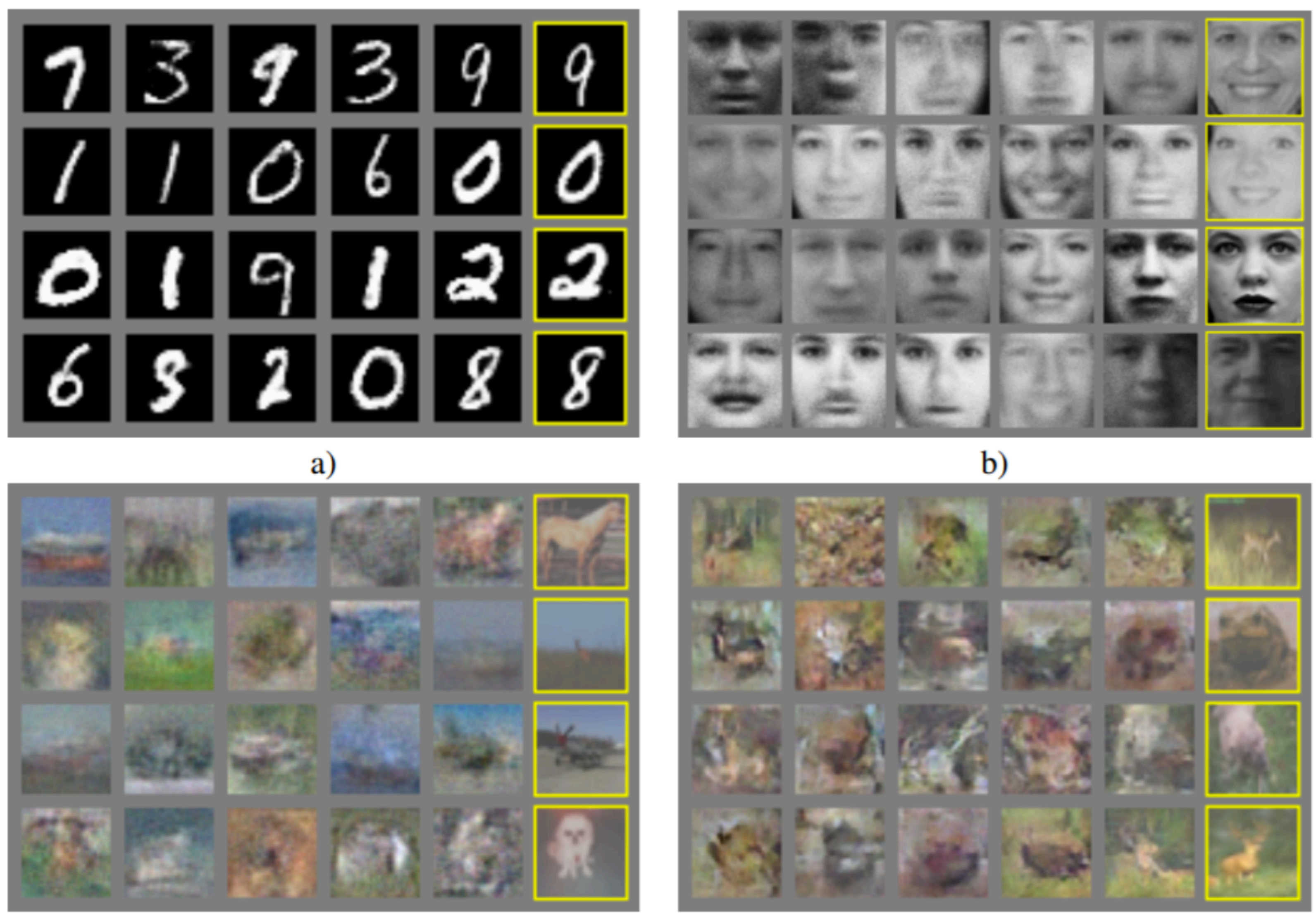
CONTENTS

Carnegie Mellon University

- MOTIVATION
- Generative vs Discriminative Models
- GANs vs VAEs
- GANs Introduction
- GANs Theory
- GANs Evaluation

ORIGINAL PAPER (GANS, 2014)

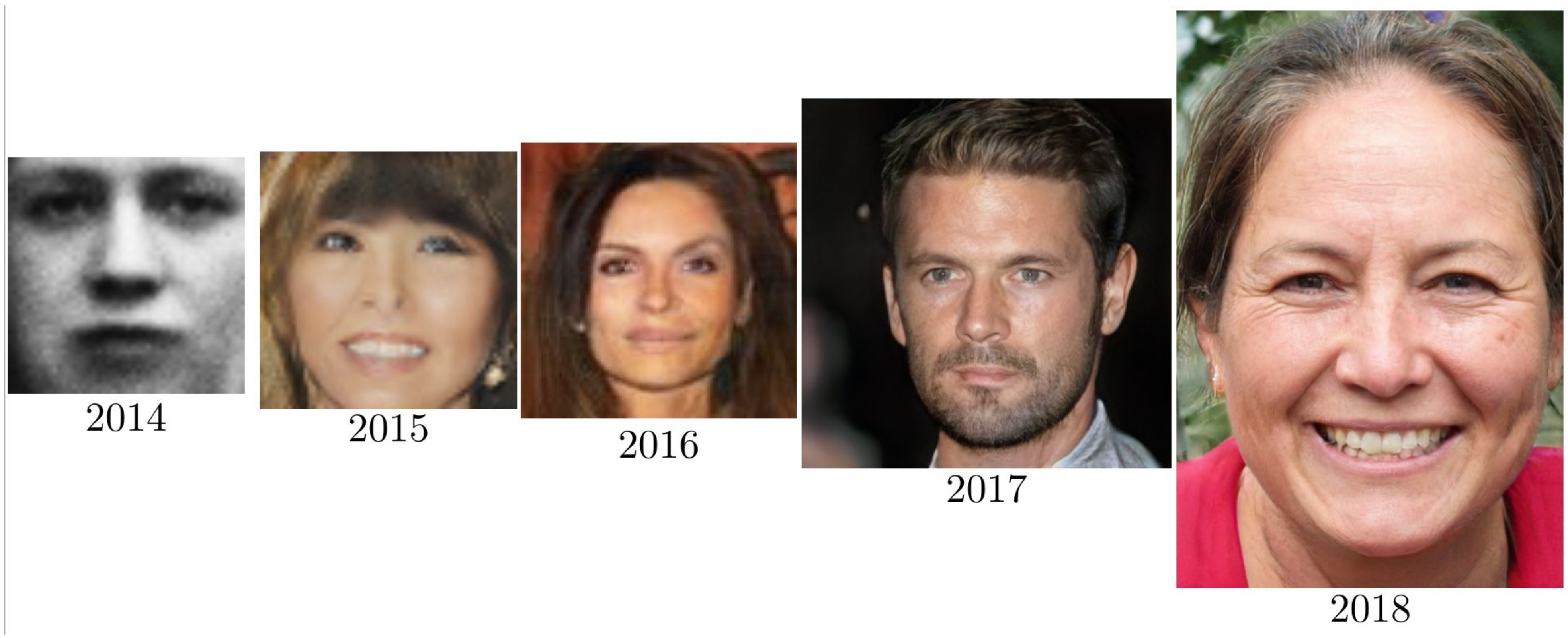
Output of original GAN paper, 2014 [GPM+14]



a) b) c) d)

GANS PROGRESSION

- Better quality
- High Resolution



https://twitter.com/goodfellow_ian/status/1084973596236144640?lang=en

STARGAN(2018)

Manipulating Celebrity Faces [CCK+17]

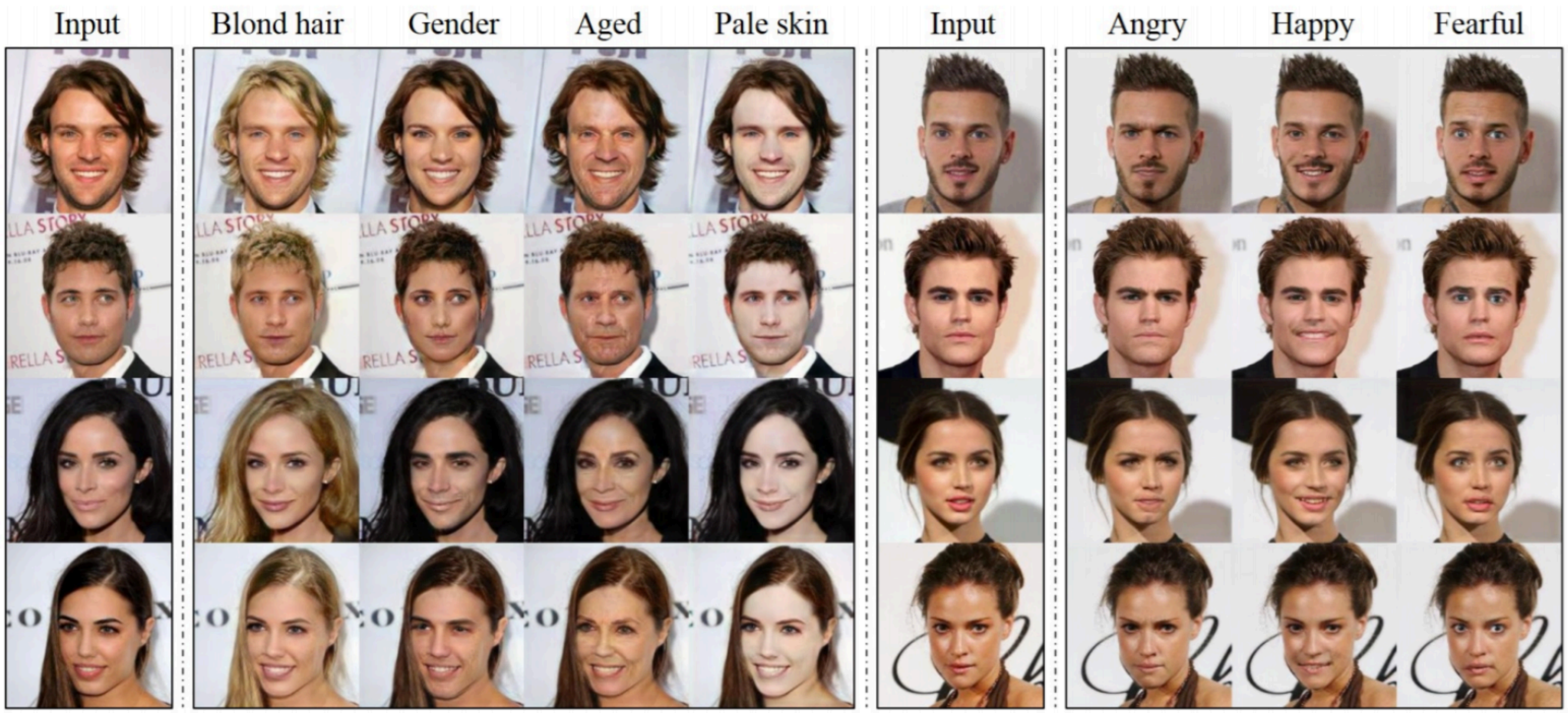


Figure 1. Multi-domain image-to-image translation results on the CelebA dataset via transferring knowledge learned from the RaFD dataset. The first and sixth columns show input images while the remaining columns are images generated by StarGAN. Note that the images are generated by a single generator network, and facial expression labels such as angry, happy, and fearful are from RaFD, not CelebA.

PROGRESSIVE GROWING OF GANS (2018)

Figure 5: 1024×1024 images generated using the CELEBA-HQ dataset. See Appendix F for a larger set of results, and the accompanying video for latent space interpolations.

HIGH FIDELITY NATURAL IMAGES (2019)

Generating High-Quality Images [BDS18]

CONTENTS

- Motivation
- DISCRIMINATIVE vs GENERATIVE MODELS
- GANs vs VAEs
- GANs Introduction
- GANs Theory
- GANs Evaluation

DISCRIMINATIVE vs GENERATIVE MODELS

Carnegie Mellon University

Given a distribution of inputs X and labels Y.

DISCRIMINATIVE MODELS

- Discriminative models learn conditional distribution P(Y | X)

GENERATIVE MODELS

- Generative models learn the joint distribution P(Y, X)

DISCRIMINATIVE vs GENERATIVE MODELS

Given a distribution of inputs X and labels Y.

DISCRIMINATIVE MODELS

- Discriminative models learn conditional distribution P(Y | X)
- Learns decision boundary between classes.
- Limited scope. Can only be used for classification tasks.

GENERATIVE MODELS

- Generative models learn the Generative models learn the joint distribution P(Y, X)
- Learns actual probability distribution of data.
- Can do both generative and discriminative tasks.

DISCRIMINATIVE vs GENERATIVE MODELS

Carnegie Mellon University

Given a distribution of inputs X and labels Y.

DISCRIMINATIVE MODELS

- Discriminative models learn conditional distribution P(Y | X)
- Learns decision boundary between classes.
- Limited scope. Can only be used for classification tasks.

GENERATIVE MODELS

- Generative models learn the Generative models learn the joint distribution P(Y, X)
- Learns actual probability distribution of data.
- Can do both generative and discriminative tasks. distribution P(Y, X)
- Harder problem. Requires a deeper understanding of the distribution than discriminative models.

EXPLICIT VS IMPLICIT DISTRIBUTION MODELLING

Carnegie Mellon University

EXPLICIT DISTRIBUTION MODELS

- Calculates P(x ~ X) for all x

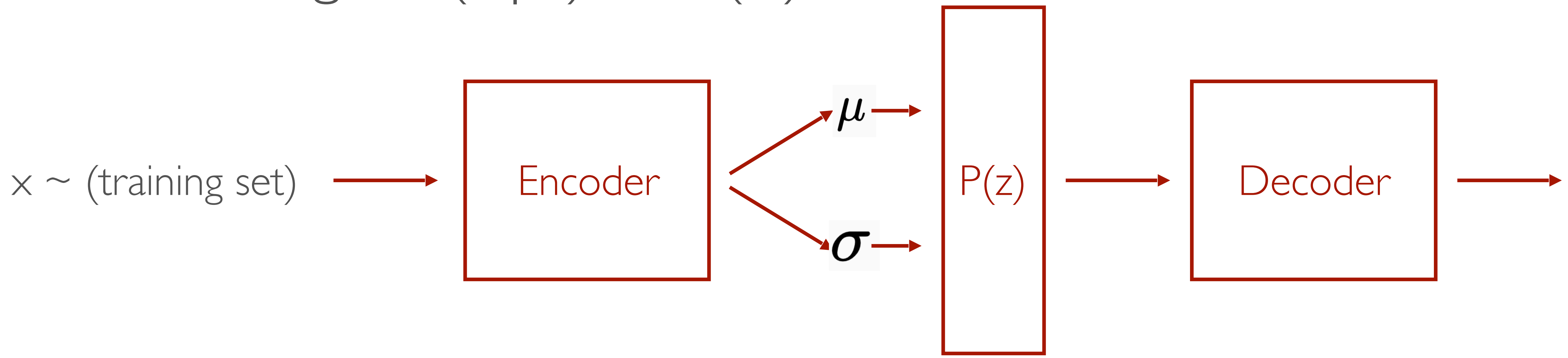
IMPLICIT DISTRIBUTION MODELS

- Generate x ~ X

CONTENTS

VARIATIONAL AUTOENCODERS (VAE)

- Encoder models P(Z|X)
- Decoder models P(X|Z)
- Loss encourages P(Z|X) ~ Q(Z)



VAEs vs GANs

Carnegie Mellon University

VAEs

- Minimizing the KL-divergence
- Minimize a <u>bound</u> on the divergence between generated distribution and target distribution
- Simpler optimization. Trains faster and more reliably
- Results are blurry

GANs

- Minimizing the Jenson-Shannon Divergence
- Minimize divergence between generated distribution and target distribution
- Noisy and difficult optimization
- Sharper results

CONTENTS

Carnegie Mellon University

- Motivation
- Discriminative vs Generative Models
- GANs vs VAEs
- GANs INTRODUCTION
- GANs Theory
- GANs Evaluation

WHAT ARE GANS?

Generative Adversarial Networks

WHAT ARE GANS?

Carnegie Mellon University

Generative Adversarial Networks

Generative Models

We try to learn the underlying the distribution from which our dataset comes from.

Eg: Variational AutoEncoders (VAE)

WHAT ARE GANS?

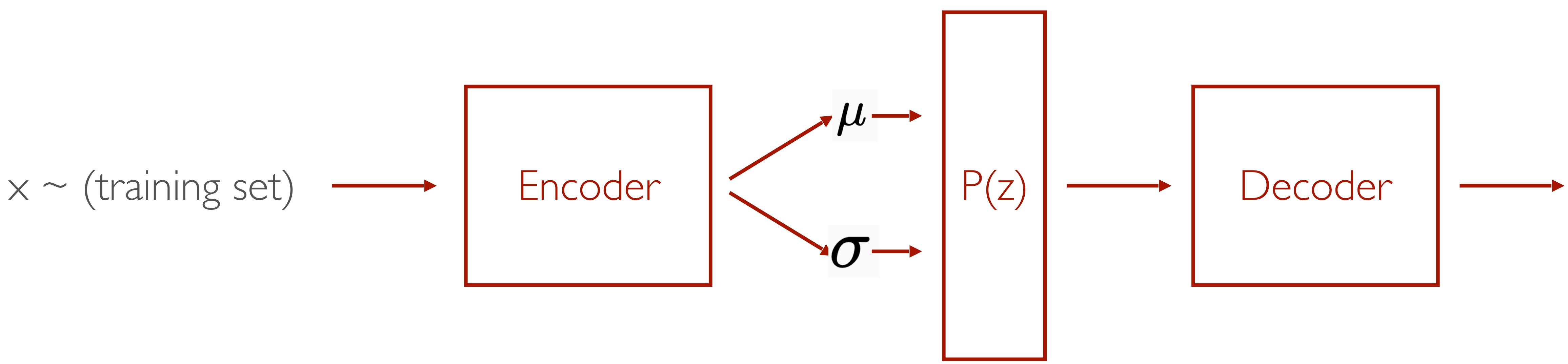
Carnegie Mellon University

Generative Adversarial Networks

Generative Models

We try to learn the underlying the distribution from which our dataset comes from.

Eg: Variational AutoEncoders (VAE)



WHAT ARE GANS?

Carnegie Mellon University

Generative Adversarial Networks

Generative Models

We try to learn the underlying the distribution from which our dataset comes from.
Eg: Variational AutoEncoders (VAE)

Adversarial Training

GANS are made up of two competing networks (adversaries) that are trying beat each other.

WHAT ARE GANS?

Generative Adversarial Networks

Generative Models

We try to learn the underlying the distribution from which our dataset comes from.
Eg: Variational AutoEncoders (VAE)

Adversarial Training

GANS are made up of two competing networks (adversaries) that are trying beat each other.

Neural Networks

WHAT ARE GANS?

Carnegie Mellon University

Generative Adversarial Networks

Generative Models

We try to learn the underlying the distribution from which our dataset comes from.
Eg: Variational AutoEncoders (VAE)

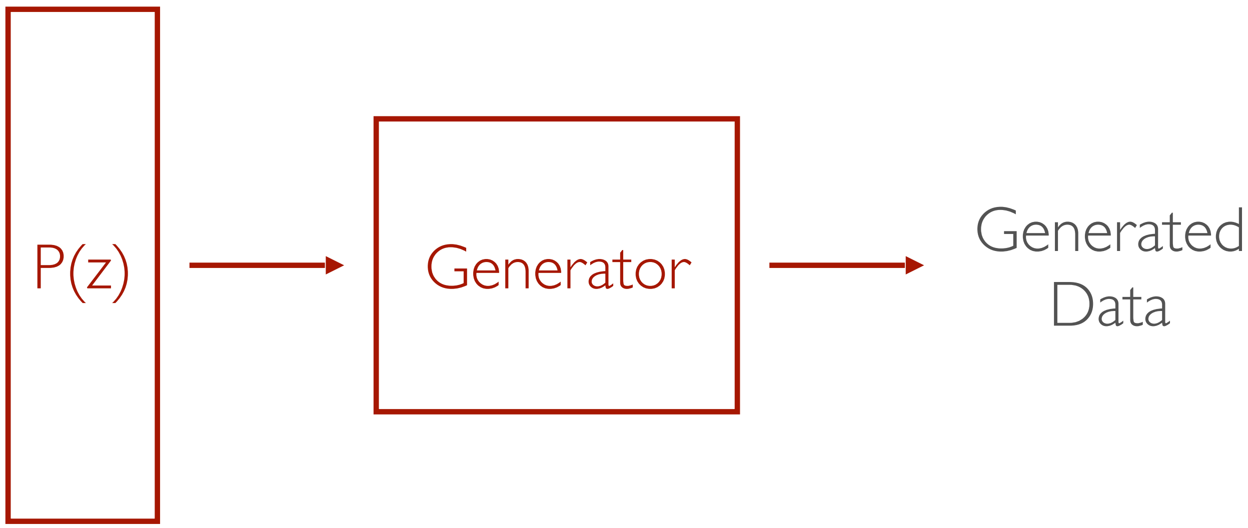
Adversarial Training

GANS are made up of two competing networks (adversaries) that are trying beat each other.

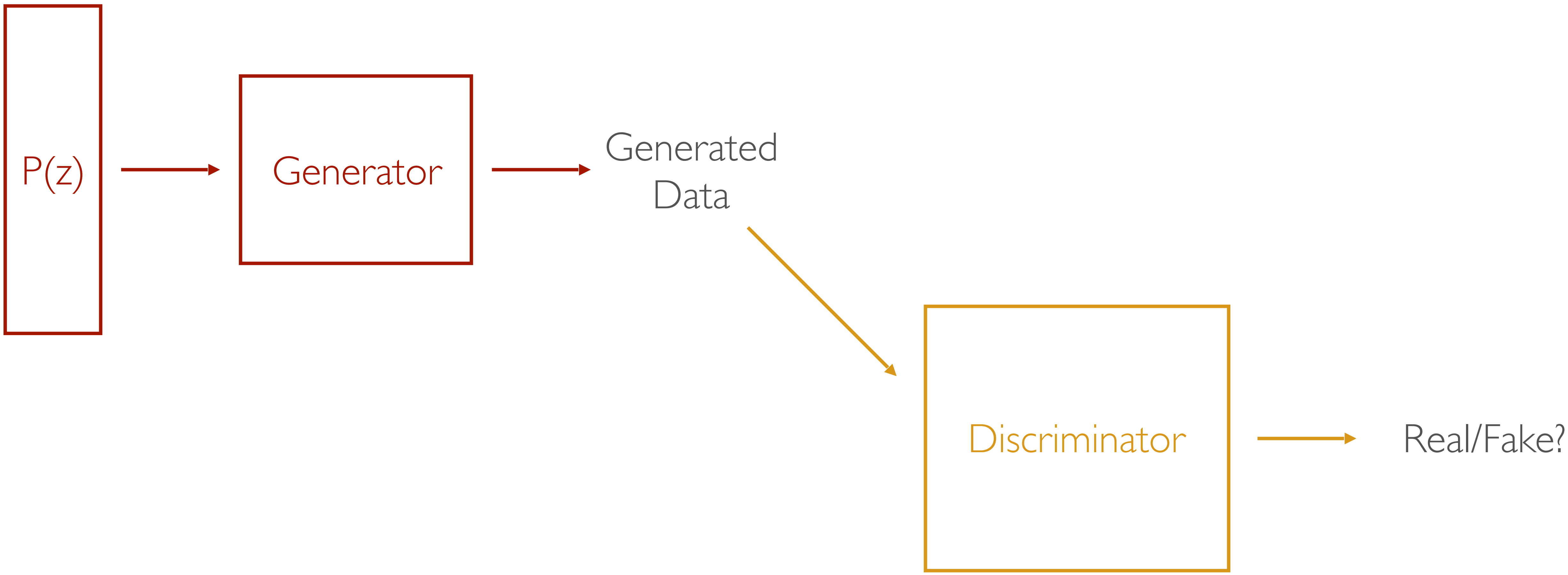
Neural Networks

GOAL: Generate data from an unlabelled distribution.

WHAT ARE GANS?

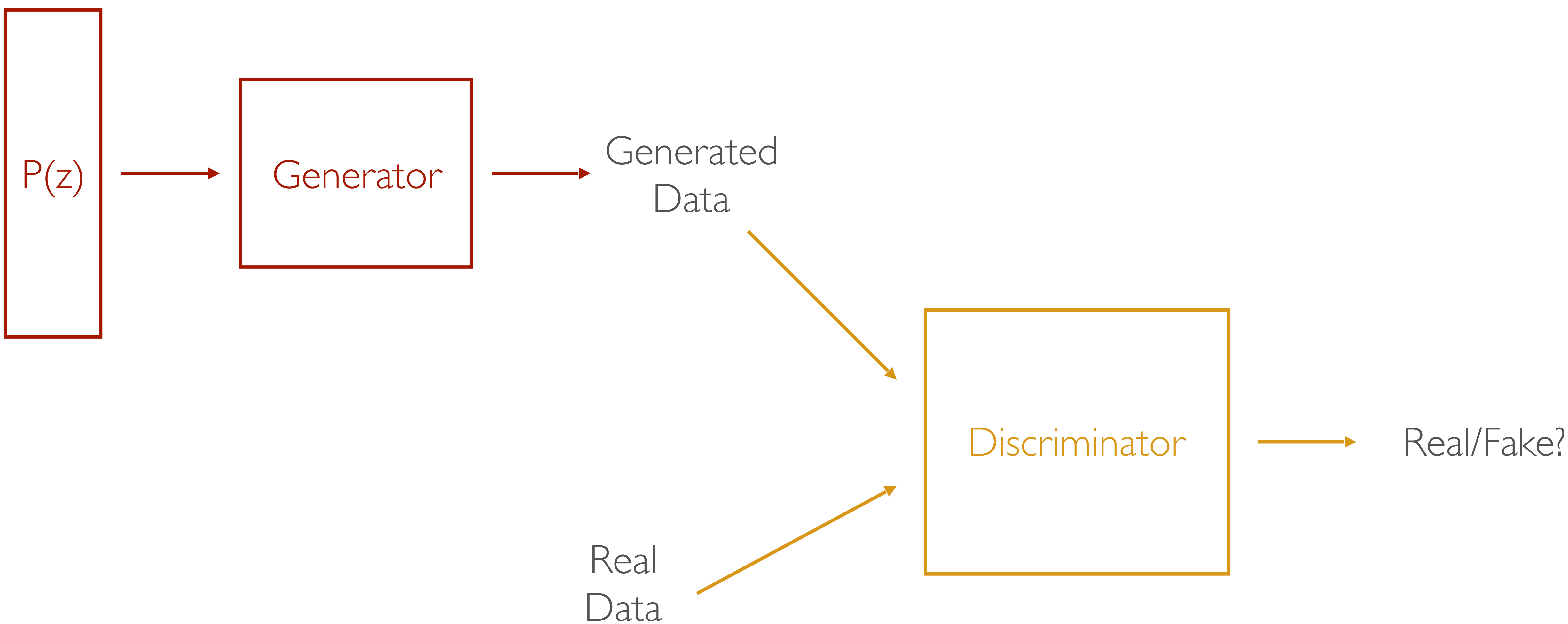


WHAT ARE GANS?



WHAT ARE GANS?

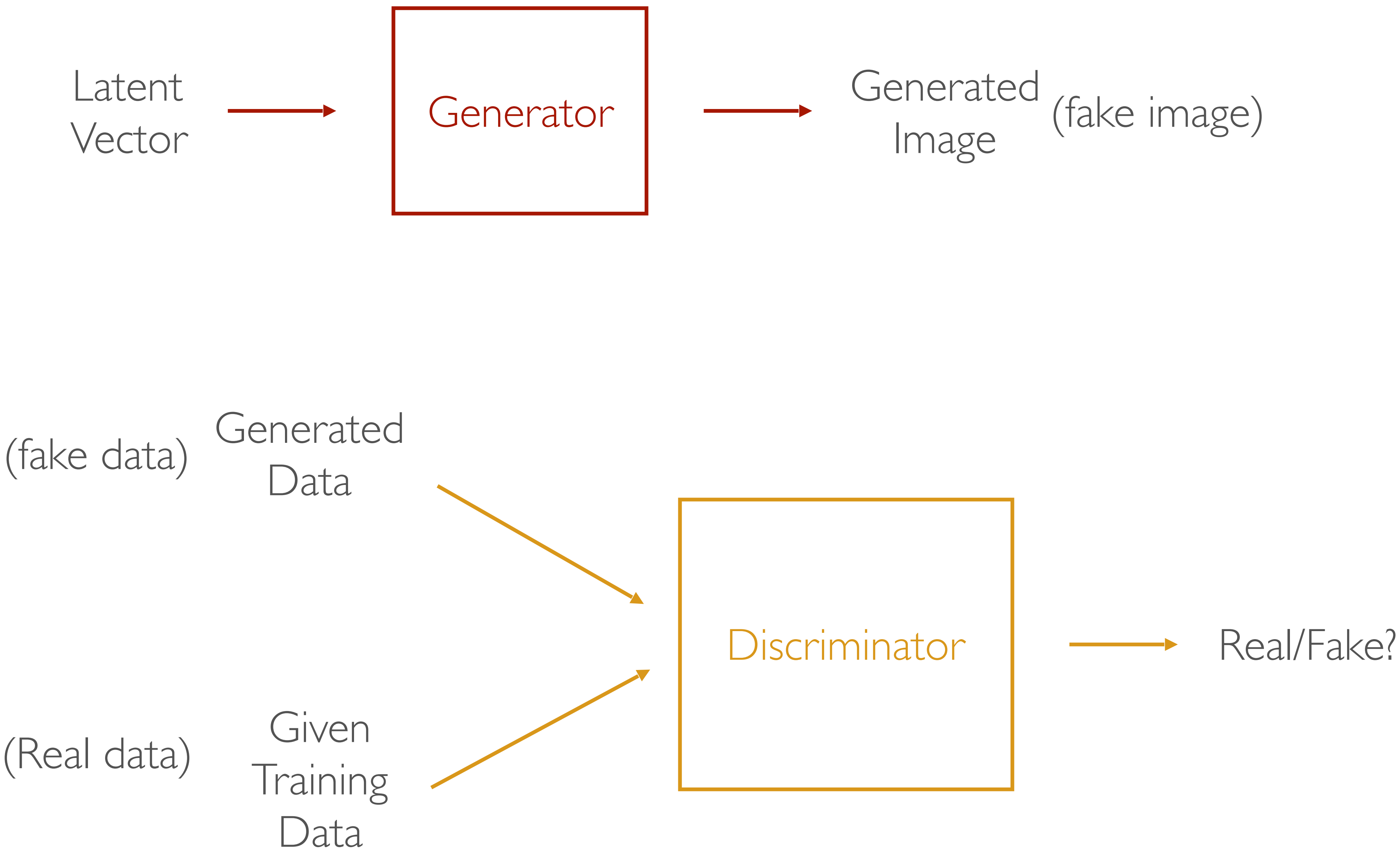
Carnegie Mellon University



HOW TO TRAIN A GAN?

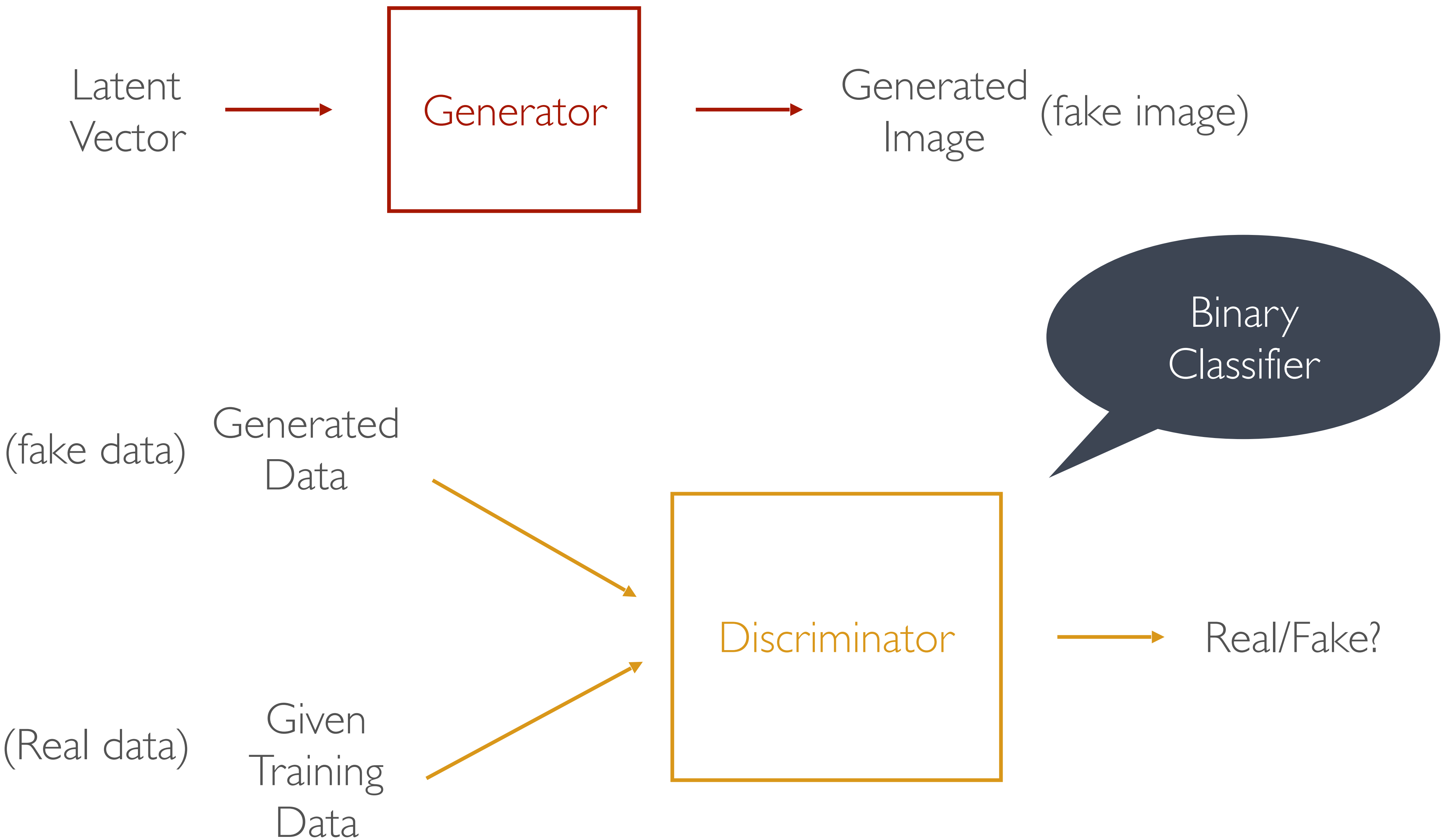
HOW TO TRAIN A GAN?

At t = 0,



HOW TO TRAIN A GAN?

At t = 0,



HOW TO TRAIN A GAN?

Which network should I train first?

HOW TO TRAIN A GAN?

Which network should I train first?

Discriminator!

HOW TO TRAIN A GAN?

Carnegie Mellon University

Which network should I train first?

Discriminator!

But with what training data?

Carnegie Mellon University

HOW TO TRAIN A GAN?

Which network should I train first?

Discriminator!

But with what training data?

The Discriminator is a Binary classifier.
The Discriminator has two class - Real and Fake.
The data for Real class if already given: THE TRAINING DATASET
The data for Fake class? -> generate from the Generator

HOW TO TRAIN A GAN?

What's next? -> Train the Generator

But how? What's our training objective?

HOW TO TRAIN A GAN?

What's next? -> Train the Generator

But how? What's our training objective?

Generate images from the Generator such that they are classified incorrectly by the Discriminator!

HOW TO TRAIN A GAN?

Carnegie Mellon University

Discriminator

Step 1:
Train the Discriminator
<u>using the current ability</u>
of the Generator.

HOW TO TRAIN A GAN?

Carnegie Mellon University

Discriminator

Generator

Step 1:

Train the Discriminator
<u>using the current ability</u>
of the Generator.

Step 2:

Train the Generator
<u>to beat</u>
the Discriminator.

Carnegie Mellon University

HOW TO TRAIN A GAN?

Discriminator

Generator

Step 1:
Train the Discriminator
<u>using the current ability</u>
of the Generator.

Step 2:
Train the Generator
<u>to beat</u>
the Discriminator.

Generate images from the Generator
such that they are classified incorrectly by the Discriminator!

Carnegie Mellon University

HOW TO TRAIN A GAN?

Discriminator

Generator

Step 1:

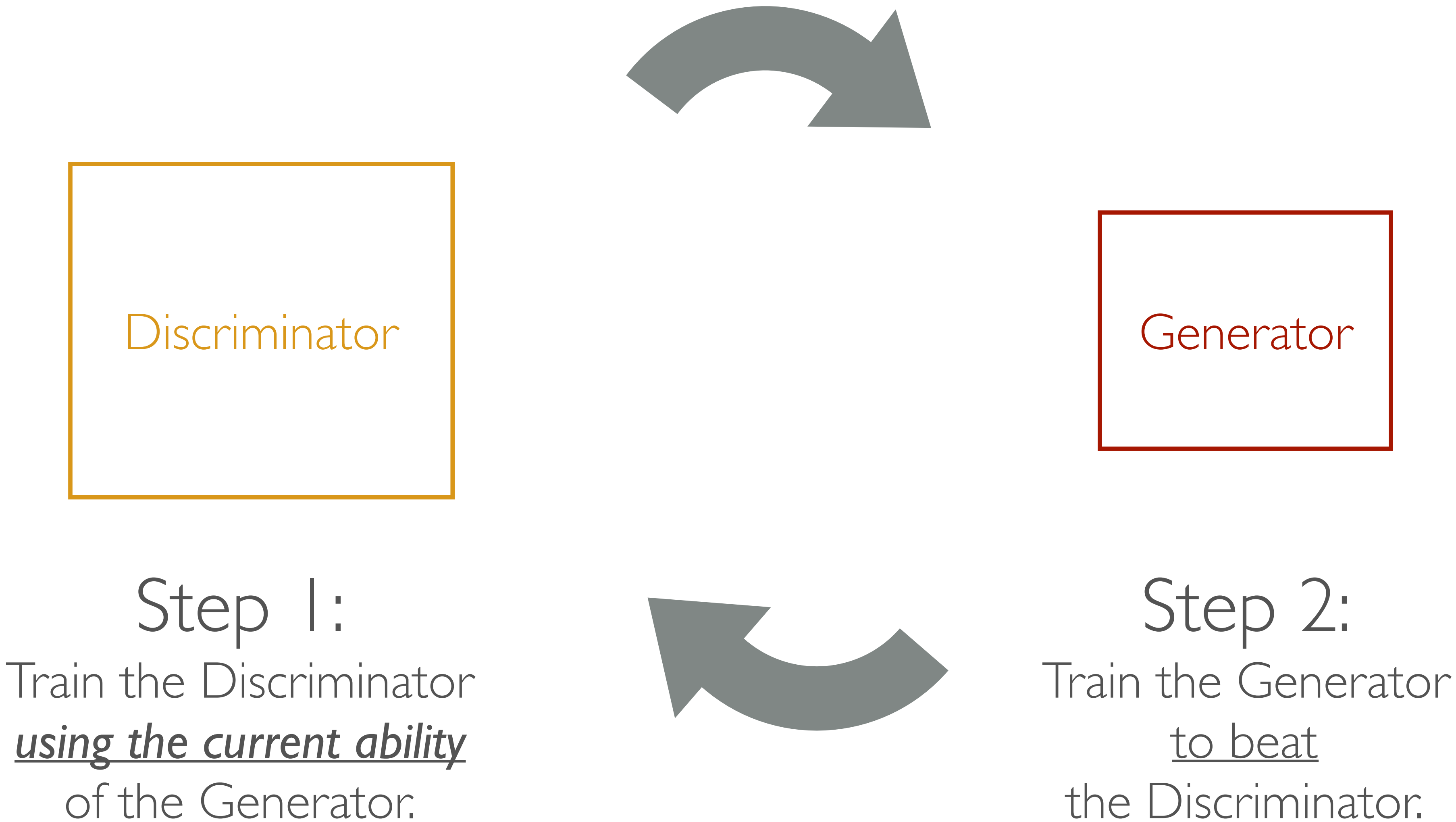
Train the Discriminator
<u>using the current ability</u>
of the Generator.

Step 2:

Train the Generator
<u>to beat</u>
the Discriminator.

Carnegie Mellon University

HOW TO TRAIN A GAN?



CONTENTS

Carnegie Mellon University

GENERATIVE ADVERSARIAL NETWORKS

Carnegie Mellon University

- Introduced in 2014
- Goal is to model P(X), the distribution of training data
- Model can generate samples from P(X)
- Trained using a pair of "adversaries"

THE GENERATOR

- The generator learns P(X|Z) : Produces realistic looking data X from a latent vector Z
- Z is sampled from a known prior, such as a Gaussian
- Maps a simple known distribution to a complicated data distribution
- GOAL : Generated distribution, G(z), matches the true data distribution P(X)

THE DISCRIMINATOR

Carnegie Mellon University

- Trained to tell the difference between real and generated (fake) data
- Backpropagates its expectations to the generator
- "Thrown away" after generator is trained

Carnegie Mellon University

ORIGINAL GAN FORMULATION

The original GAN formulation is the following min-max game

$$\min_G \max_D V(D, G) = \mathbb{E}_X \log D(X) + \mathbb{E}_Z \log(1 - D(G(Z)))$$

- D wants $D(X) = 1$ and $D(G(Z)) = 0$
- G wants $D(G(Z)) = 1$

THE OPTIMAL DISCRIMINATOR

P_D = actual data distribution
P_G = generated data distribution

$D(X)$ = discriminator output

Objective: $\min_G \max_D V(D, G) = \mathbb{E}_X \log D(X) + \mathbb{E}_Z \log(1 - D(G(Z)))$

What is the optimal discriminator?

$$f := \mathbb{E}_{X \sim P_D} \log D(X) + \mathbb{E}_{X \sim P_G} \log(1 - D(X))$$
$$= \int_X [P_D(X) \log D(X) + P_G(X) \log(1 - D(X))] \, dX$$
$$\frac{\partial f}{\partial D(X)} = \frac{P_D(X)}{D(X)} - \frac{P_G(X)}{1 - D(X)} = 0$$
$$\frac{P_D(X)}{D(X)} = \frac{P_G(X)}{1 - D(X)}$$
$$(1 - D(X))P_D(X) = D(X)P_G(X)$$
$$D(X) = \frac{P_D(X)}{P_G(X) + P_D(X)}$$

THE OPTIMAL DISCRIMINATOR

P_D = actual data distribution
P_G = generated data distribution

D(X) = discriminator output

$$D(X) = \frac{P_D(X)}{P_G(X) + P_D(X)}$$

CASE -1 : BAD GENERATOR

"There's no way the input X = G(z) looks like my data"

Carnegie Mellon University

THE OPTIMAL DISCRIMINATOR

P_D = actual data distribution
P_G = generated data distribution

$D(X)$ = discriminator output

$$D(X) = \frac{P_D(X)}{P_G(X) + P_D(X)}$$

CASE -1 : BAD GENERATOR

"There's no way the input $X = G(z)$ looks like my data"

$$P_D(X) = 0, P_G(X) = 1$$
$$D(X) = 0$$

THE OPTIMAL DISCRIMINATOR

P_D = actual data distribution
P_G = generated data distribution

D(X) = discriminator output

$$D(X) = \frac{P_D(X)}{P_G(X) + P_D(X)}$$

CASE - II : GOOD GENERATOR

"I cannot tell the difference between X = G(z) and my data"

Carnegie Mellon University

THE OPTIMAL DISCRIMINATOR

P_D = actual data distribution
P_G = generated data distribution

$D(X)$ = discriminator output

$$D(X) = \frac{P_D(X)}{P_G(X) + P_D(X)}$$

CASE - II : GOOD GENERATOR

"I cannot tell the difference between $X = G(z)$ and my data"

$$P_D(X) = 1, P_G(X) = 1$$
$$D(X) = 0.5$$

Carnegie Mellon University

THE OPTIMAL GENERATOR

P_D = actual data distribution

P_G = generated data distribution

D(X) = discriminator output

G(Z) = generator output

Objective: $\min_G \max_D V(D, G) = \mathbb{E}_X \log D(X) + \mathbb{E}_Z \log(1 - D(G(Z)))$

$$
\begin{aligned}
f &:= \mathbb{E}_{X \sim P_D} \log D(X) + \mathbb{E}_{X \sim P_G} \log(1 - D(X)) \\
&= \mathbb{E}_{P_D} \log \frac{P_D(X)}{P_G(X) + P_G(X)} + \mathbb{E}_{P_G} \log \frac{P_G(X)}{P_G(X) + P_G(X)} \\
&= JSD(P_D | P_G) - \log 4
\end{aligned}
$$

Carnegie Mellon University

THE OPTIMAL GENERATOR

P_D = actual data distribution
P_G = generated data distribution

D(X) = discriminator output
G(Z) = generator output

Objective: $\min_G \max_D V(D, G) = \mathbb{E}_X \log D(X) + \mathbb{E}_Z \log(1 - D(G(Z)))$

Generator wants to minimize this!

$$
\begin{aligned}
f &:= \mathbb{E}_{X \sim P_D} \log D(X) + \mathbb{E}_{X \sim P_G} \log(1 - D(X)) \\
&= \mathbb{E}_{P_D} \log \frac{P_D(X)}{P_G(X) + P_G(X)} + \mathbb{E}_{P_G} \log \frac{P_G(X)}{P_G(X) + P_G(X)} \\
&= JSD(P_D | P_G) - \log 4
\end{aligned}
$$

Carnegie Mellon University

THE OPTIMAL GENERATOR

$$f := \mathbb{E}_{X\sim P_D} \log D(X) + \mathbb{E}_{X\sim P_G} \log(1 - D(X))$$

$$= \mathbb{E}_{P_D} \log \frac{P_D(X)}{P_G(X) + P_G(X)} + \mathbb{E}_{P_G} \log \frac{P_G(X)}{P_G(X) + P_G(X)}$$

$$= JSD(P_D|P_G) - \log 4$$

Jenson-Shanon Divergence

$$m(X) = \frac{P_D + P_G}{2}$$

$$JS(P_D\|P_G) = \frac{1}{2}KL(P_D\|m) + \frac{1}{2}KL(P_G|m)$$

THE OPTIMAL GENERATOR

What is the optimal generator?

$$\min_G JSD(P_D \| P_G) - \log 4$$

Minimize the Jensen-Shannon divergence between the real and generated distributions (make the distributions similar)

MIN-MAX STATIONARY POINT

- There exists a stationary point:
  - If the generated data exactly matches the real data, the discriminator outputs 0.5 for all inputs
  - If discriminator outputs 0.5, the gradients for the generator is flat, so generator does not learn

MIN-MAX STATIONARY POINT

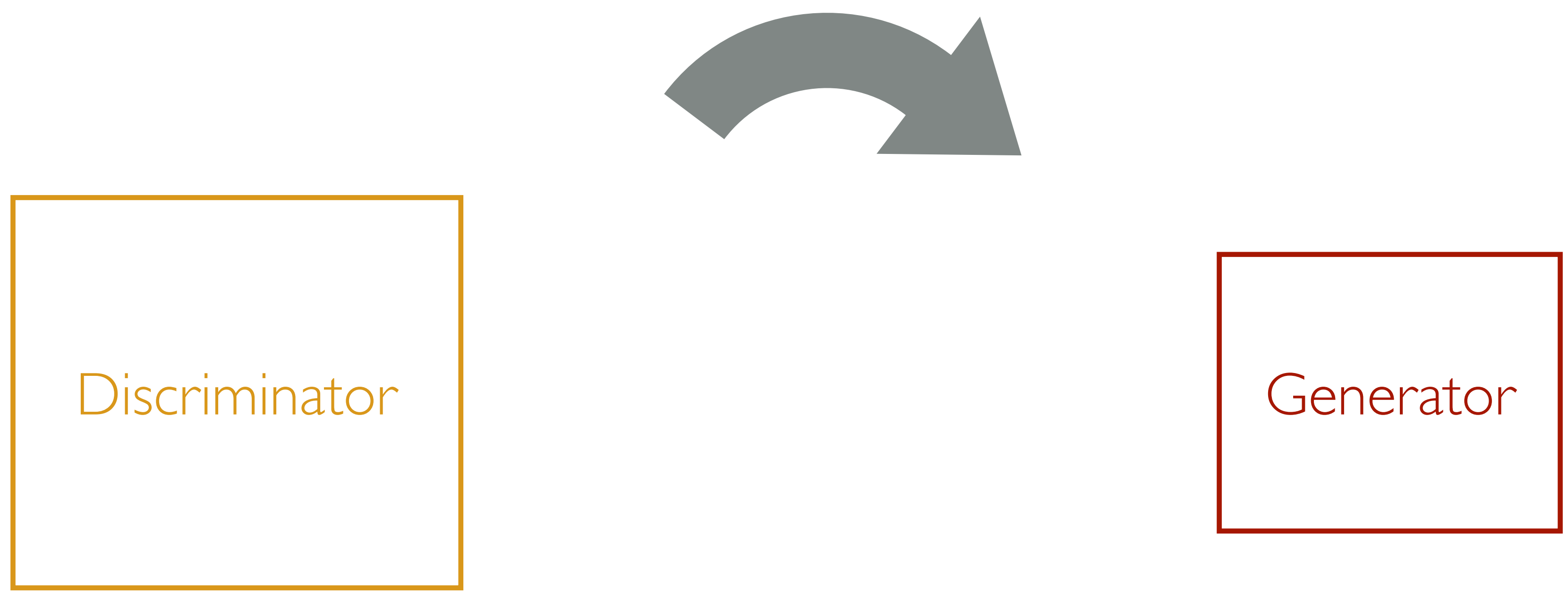
- Stationary points need not be stable (depends on the exact GANs formulation and other factors)

MIN-MAX OPTIMIZATION

- Both generator and the discriminator need to be trained simultaneously
- If discriminator is undertrained, it provides sub-optimal feedback to the generator
- If the discriminator is overtrained, there is no local feedback for marginal improvements
- Discriminator and generator needs to be trained together

Carnegie Mellon University

HOW TO TRAIN A GAN?



Step 1:
Train the Discriminator
using the current ability
of the Generator.

Step 2:
Train the Generator
to beat
the Discriminator.

Objective: $\min_G \max_D V(D, G) = \mathbb{E}_X \log D(X) + \mathbb{E}_Z \log(1 - D(G(Z)))$

CONTENTS

Carnegie Mellon University

- Motivation
- Discriminative vs Generative Models
- GANs vs VAEs
- GANs Introduction
- GANs Theory
- GANS EVALUATION

GANS EVALUATION

- Human Evaluation
- Approximate Test Set likelihood
- Evaluate with Discriminative Network

GANS EVALUATION : INCEPTION SCORE

- Use a discriminative network (originally based on Inception v3 Architecture) to classify generated images
  - Inception should produce a variety of labels
  - Each label should have high confidence (low entropy)

QUESTIONS?