Generative Networks

Lecture 13

Types of Learning

- Supervised learning
  - Learning from a “teacher”
  - Training data includes desired outputs

- Reinforcement learning
  - Learning to act under evaluative feedback (rewards)

- Unsupervised learning
  - Discover structure in data
  - Training data does not include desired outputs

Taxonomy of Generative Models

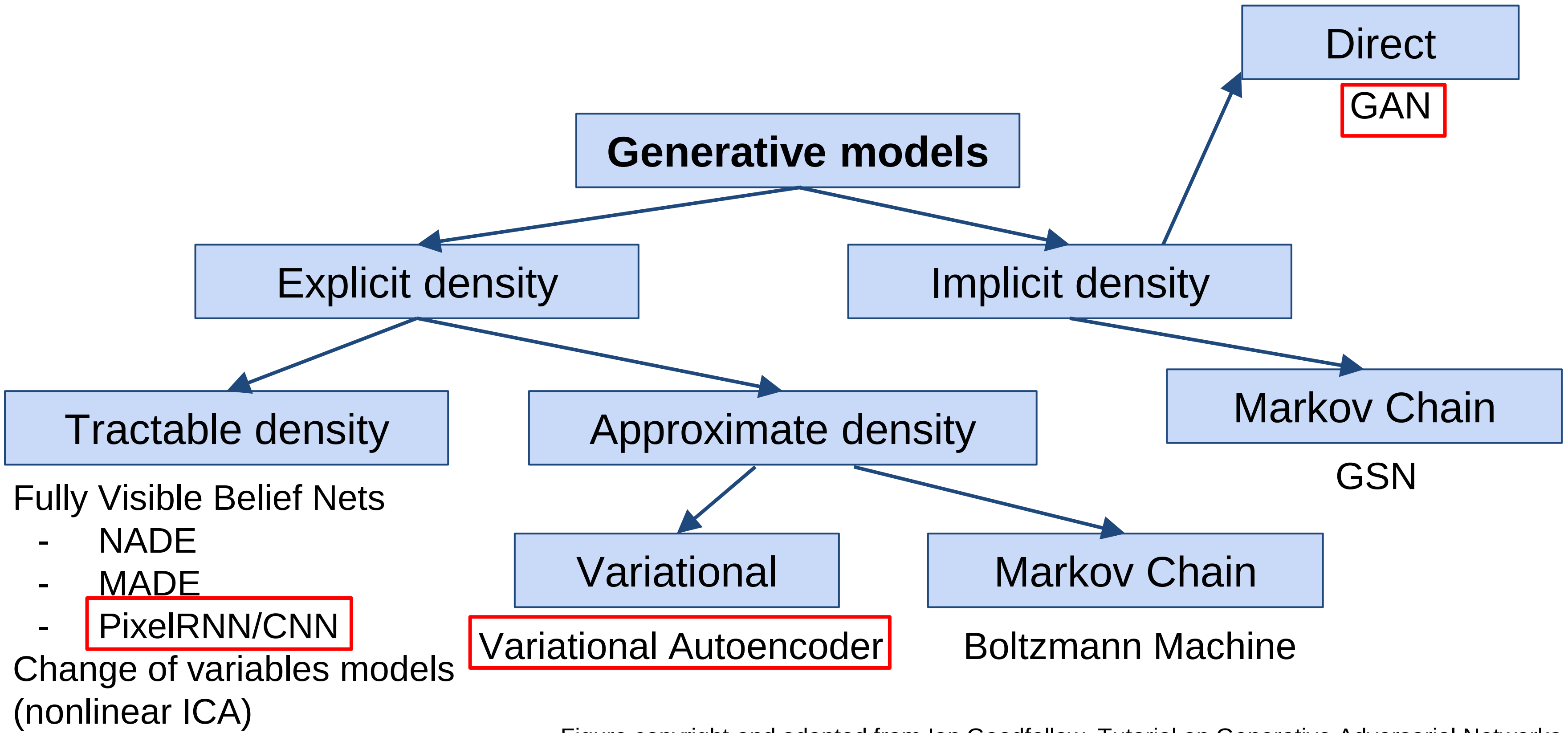
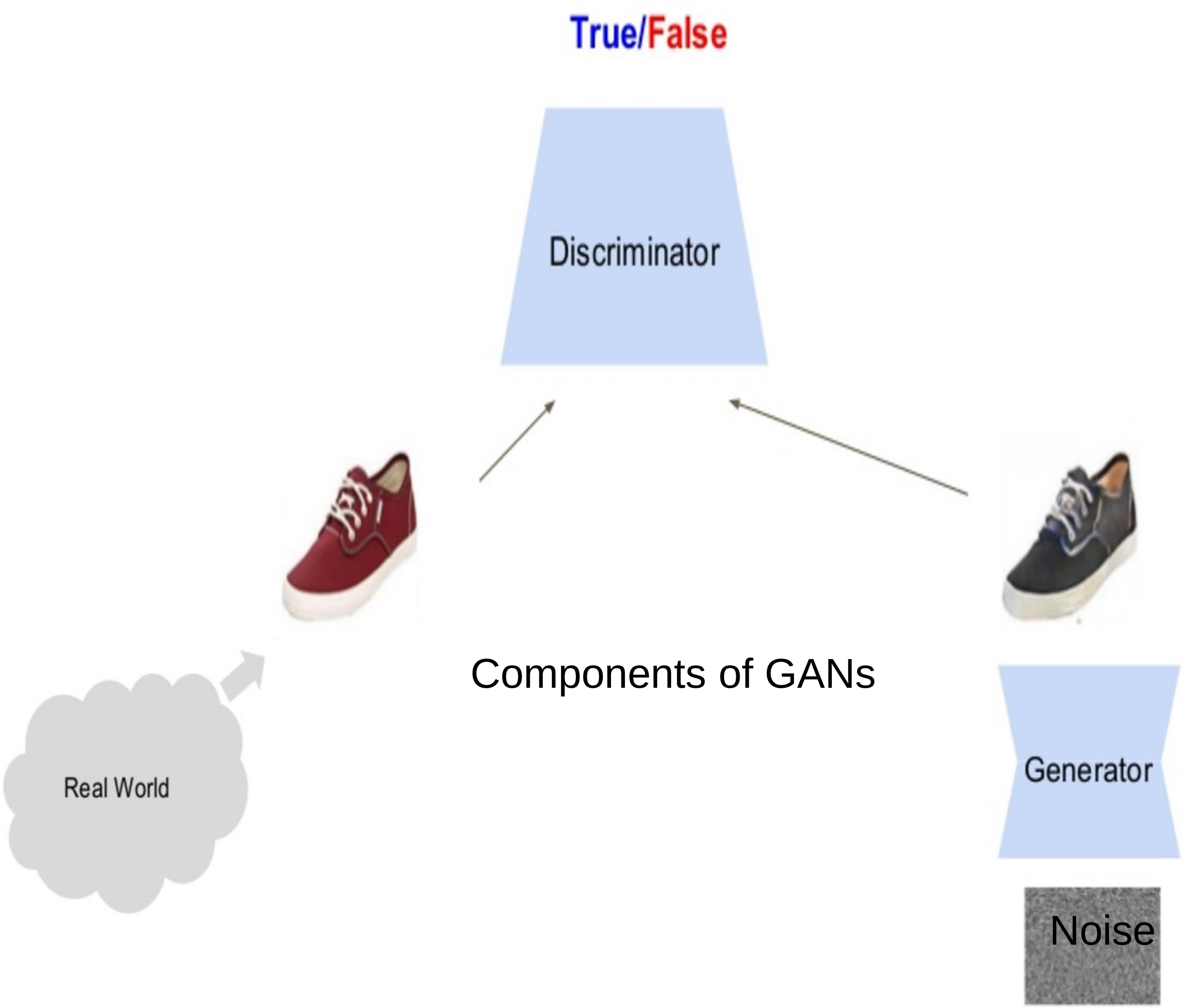


Figure copyright and adapted from Ian Goodfellow, Tutorial on Generative Adversarial Networks, 2017.

What are GANs?

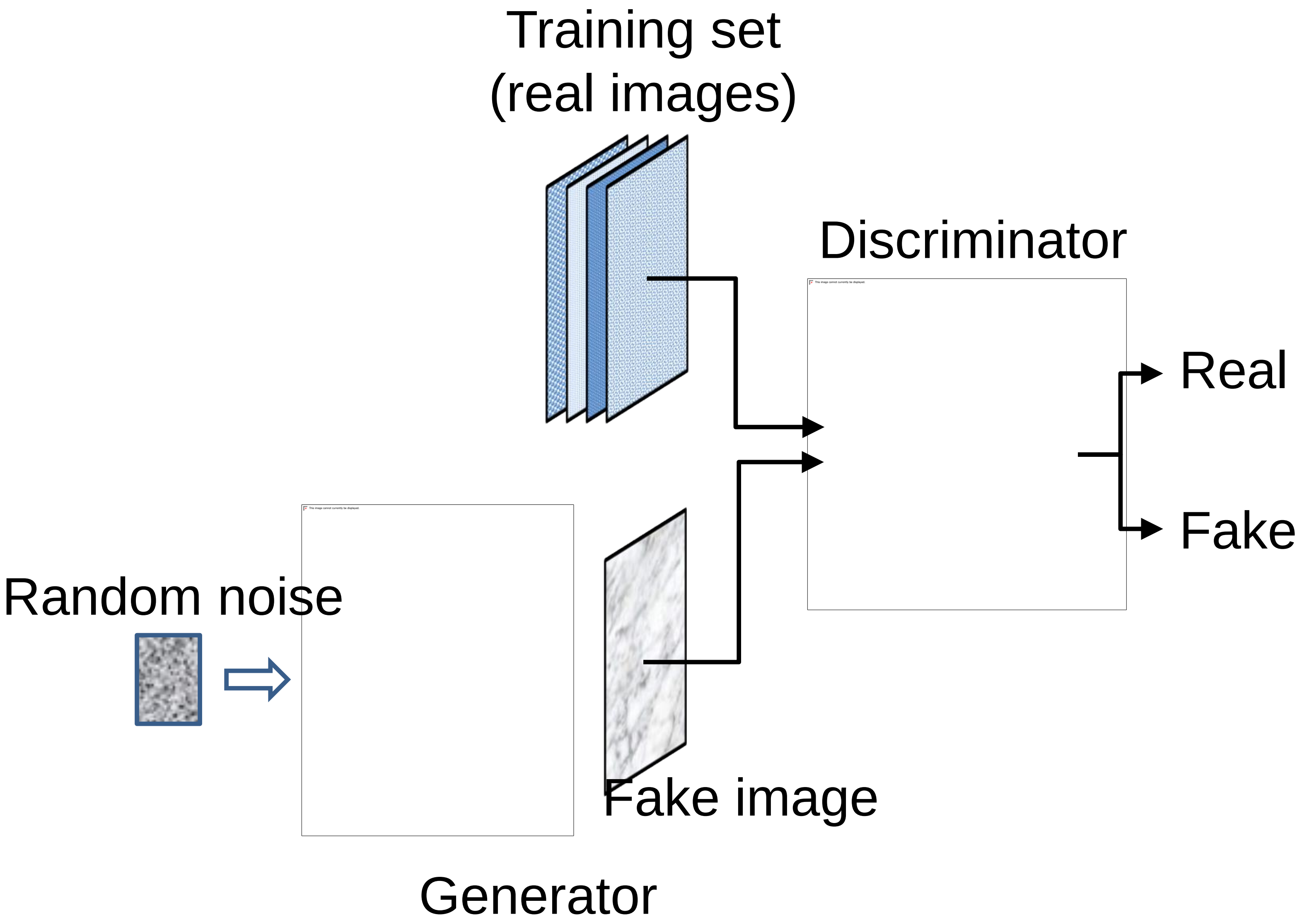
- Two ANNs: the Generator and the Discriminator
  - Generator produces ‘fake’ images from noise
  - Discriminator checks ‘fake’ images and compares with real images
- The Generator is rewarded if it fools the discriminator
- The Discriminator is rewarded if it detects a fake

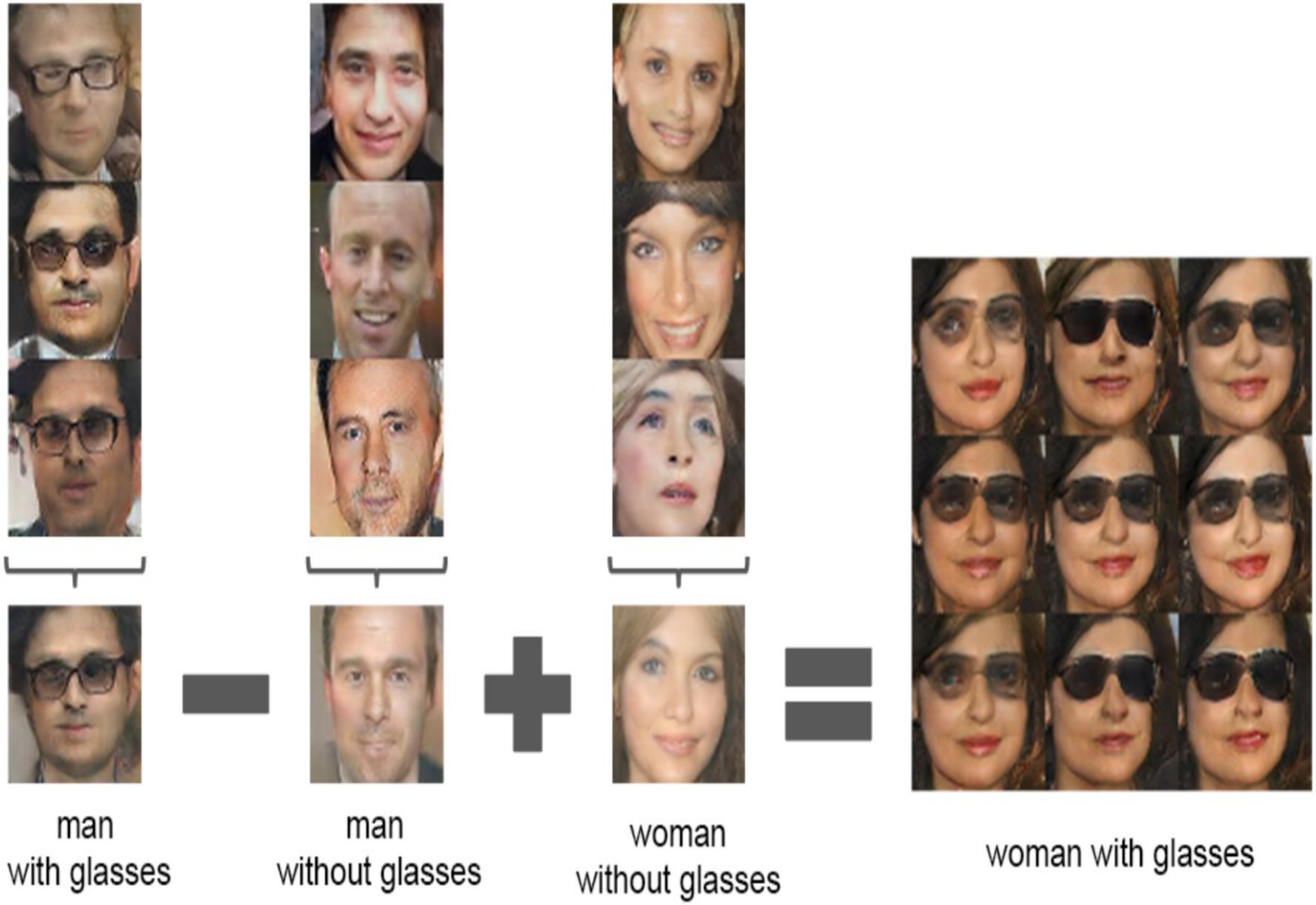
Goofellow et al., 2014
https://arxiv.org/pdf/1406.2661.pdf



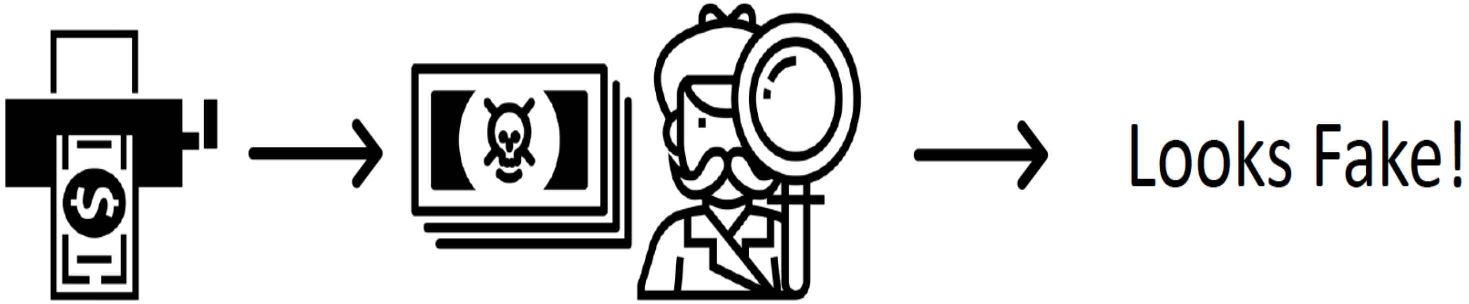
Components of GANs

GAN architecture



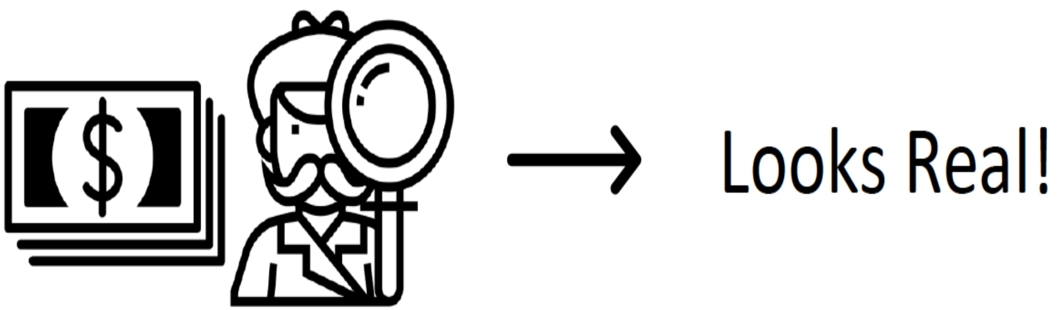

man
with glasses
man
without glasses
woman
without glasses
woman with glasses

Generative Adversarial Networks: Idea

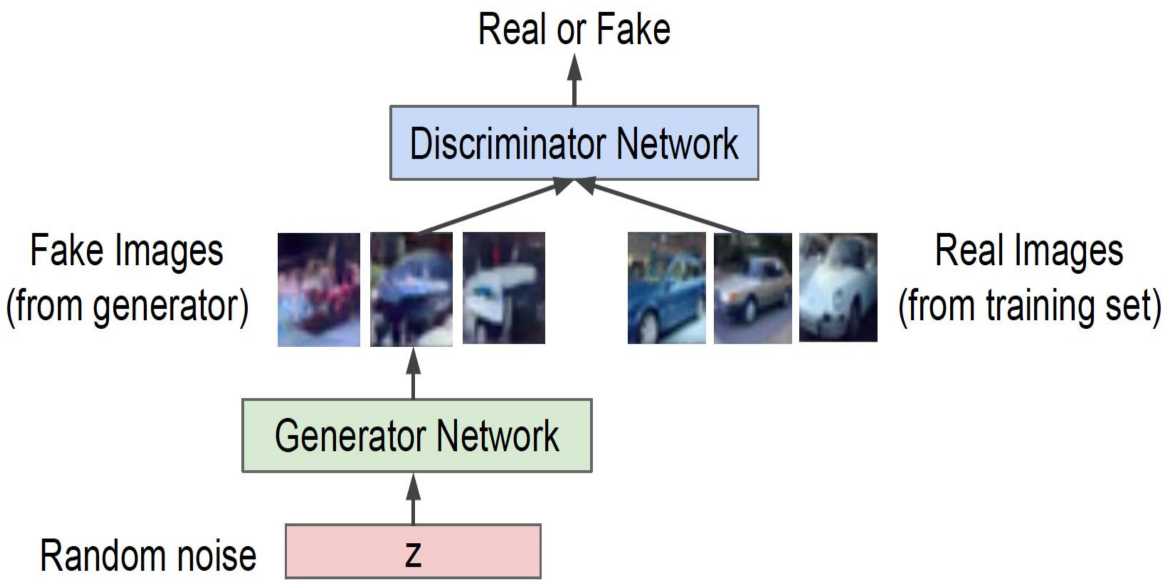


Generator (Counterfeiter): Creates fake data from random input

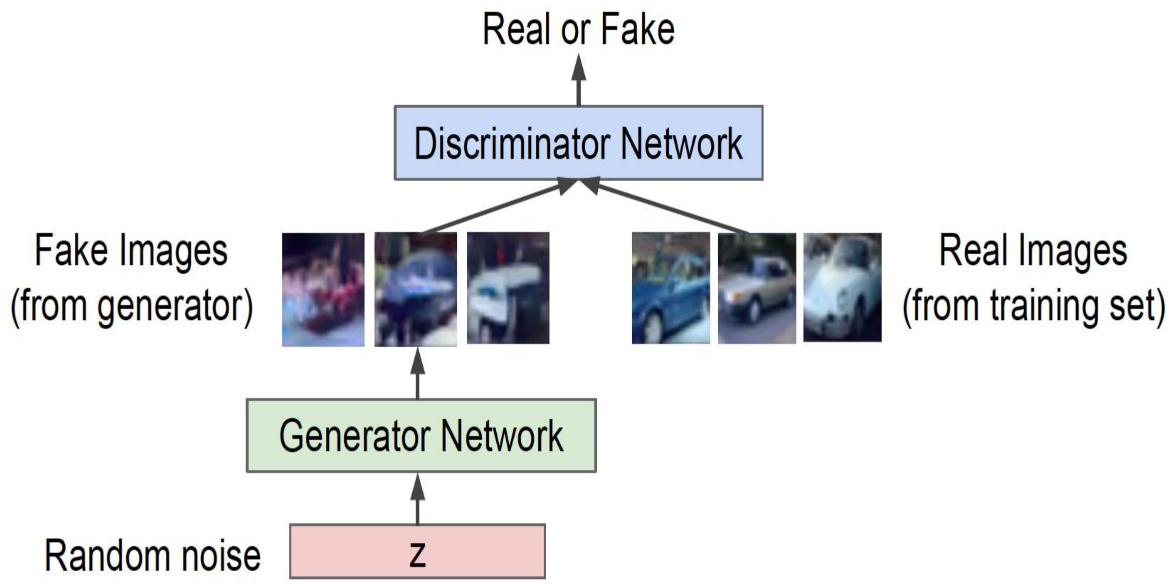
Discriminator (Detective): Distinguish real data from fake data



Generative Adversarial Networks



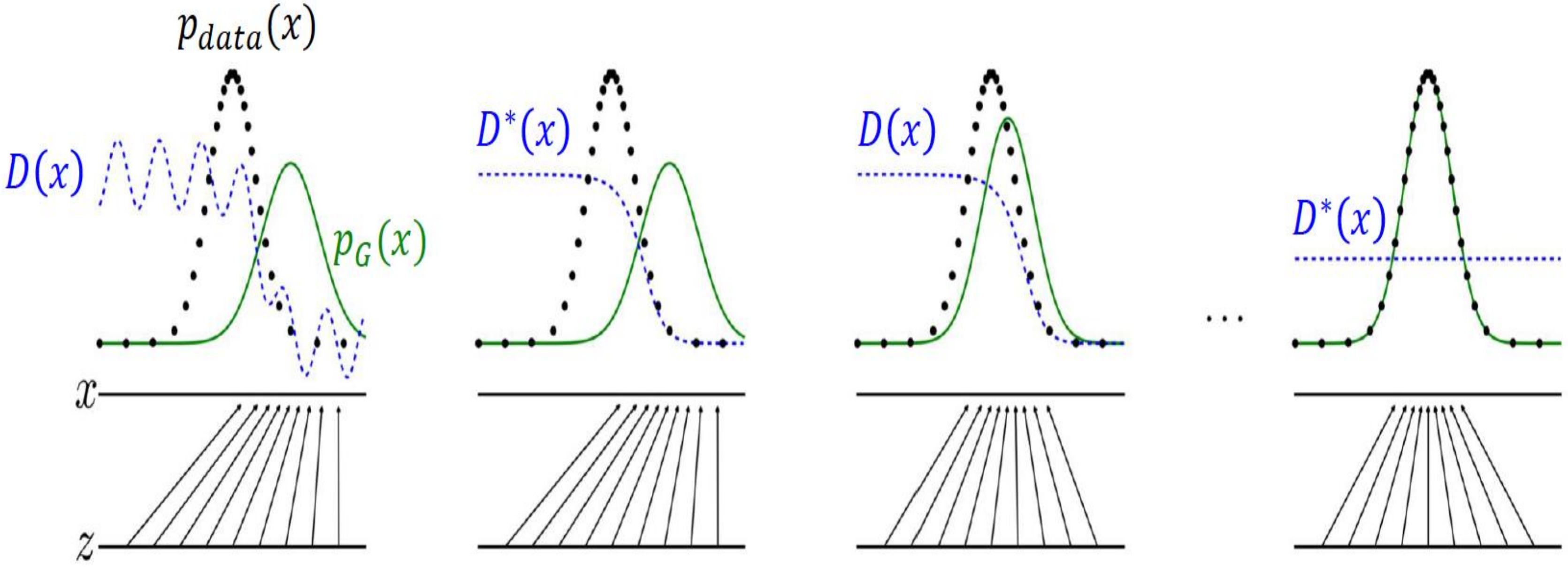
Generative Adversarial Networks



Minimax objective function:

$$\min_{\theta_g}\max_{\theta_d}\left[\mathbb{E}_{x\sim p_{data}}\log \underbrace{D_{\theta_d}(x)}_{\substack{\text{Discriminator output}\\ \text{for real data x}}} + \mathbb{E}_{z\sim p(z)}\log(1-\underbrace{D_{\theta_d}(G_{\theta_g}(z))}_{\substack{\text{Discriminator output for}\\ \text{generated fake data G(z)}}})\right]$$

Distributions during training



Objective function: minimax form

$$\min_{\theta_g} \max_{\theta_d} \left[\mathbb{E}_{x\sim p_{data}} \log \underbrace{D_{\theta_d}(x)}_{\text{Discriminator output for real data x}} + \mathbb{E}_{z\sim p(z)} \log(1 - \underbrace{D_{\theta_d}(G_{\theta_g}(z))}_{\text{Discriminator output for generated fake data G(z)}}) \right]$$

- Discriminator D tries to **maximize** this objective function, perform gradient *ascent* on the function:

$$\max_{\theta_d} \left[\mathbb{E}_{x\sim p_{data}} \log D_{\theta_d}(x) + \mathbb{E}_{z\sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z))) \right]$$

- Generator G tries to **minimize** this objective function, perform gradient *descent* on the function:

$$\min_{\theta_g} \mathbb{E}_{z\sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z)))$$

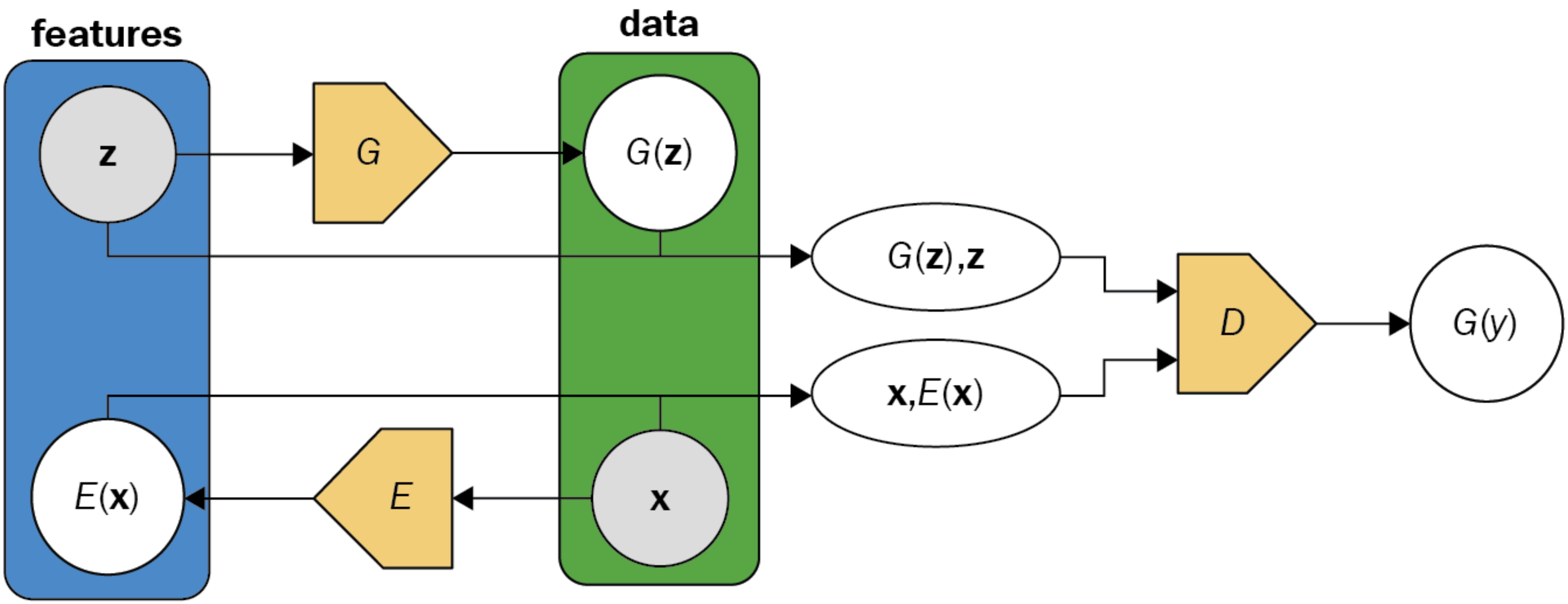
- For G, apparently better to perform gradient *ascent* on the function:

$$\max_{\theta_g} \mathbb{E}_{z\sim p(z)} \log(D_{\theta_d}(G_{\theta_g}(z)))$$

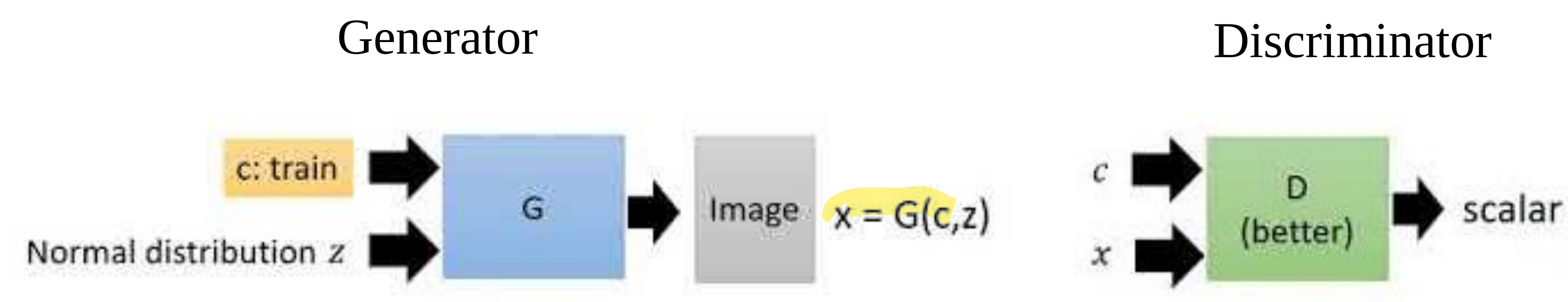
i.e. ***maximise*** probability that discriminator is ***wrong*** (rather than ***minimising*** probability that it is ***right***)

Group Work

- Work in a group of 3 and write explanation about the following GAN variants:



Group Work



Generator
Discriminator
c: train
Normal distribution z
G
Image
x = G(c,z)
c
x
D
(better)
scalar