Generative Adversarial Networks

Mahshid Alinoori

01
Introduction

02
GANs Variants

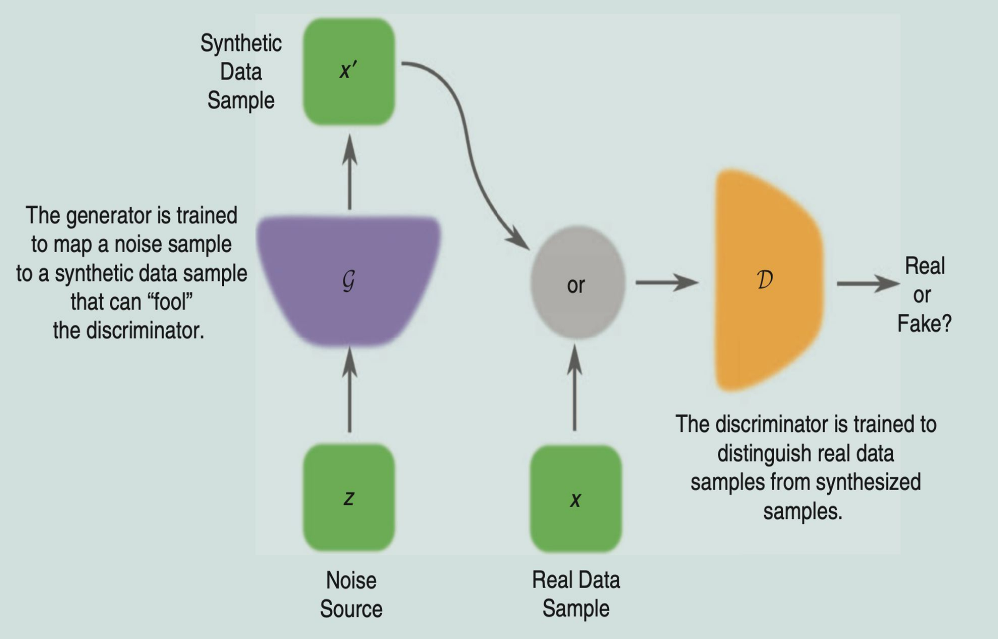
03
GANs Training

04
GANs Applications

05
Conclusion

Introduction

- GANs are networks that **implicitly** models the data distribution
- **Competition** between two networks: The Generator (the art forger) and the discriminator (the expert)
- The **discriminator** see the real data and is trained to **distinguish** between the real and the fake
- The generator gets feedback from the discriminator and tries to output synthesized sample (from latent space) that fools the discriminator optimally to a point it is totally confused

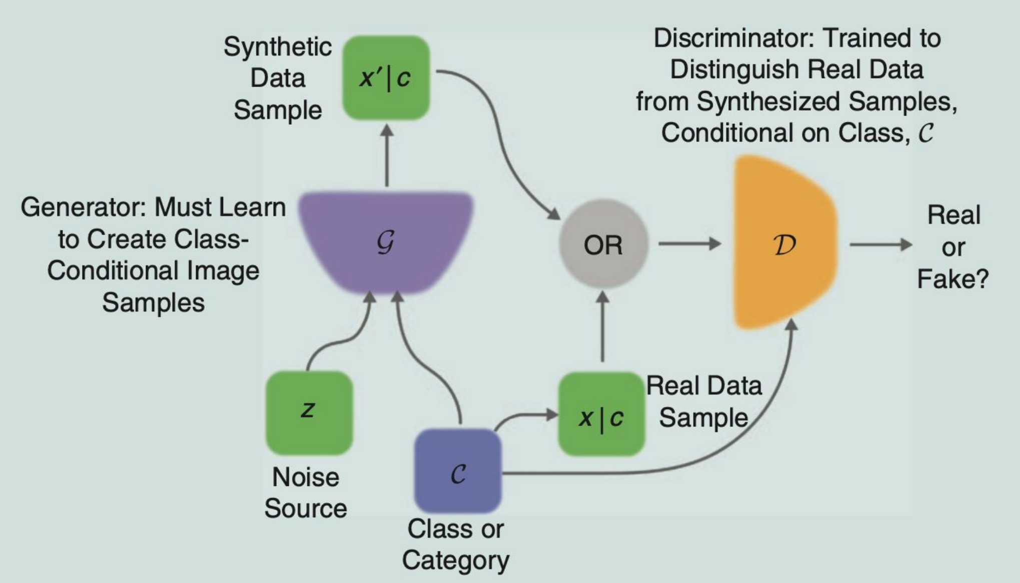


Fully Connected GANs & Convolutional GANs

- Earlier version of GANs were implemented using FC networks for both the generator and the discriminator
- The are more limited compared to other network architectures and were applied to simple datasets

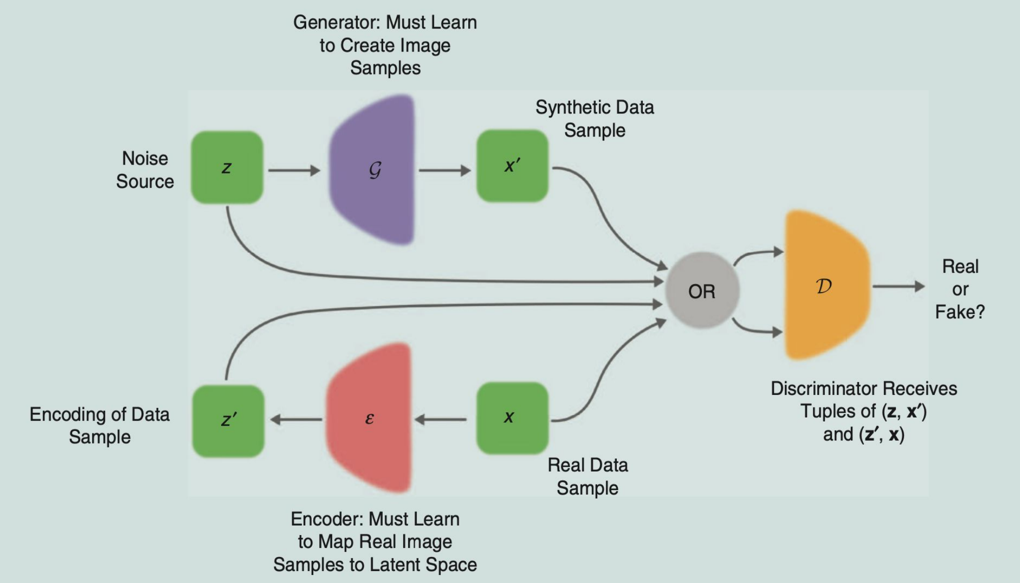
- Applying CNNs are more appropriate for image-related tasks but they are more difficult to train as the generator and the discriminator and solutions were offered as in:
  - **LAPGAN:** decomposing the generation process by decomposing the image to Laplacian pyramid
  - **DCGAN:** use of strided and fractionally strided convolutions to learn downsampling and upsampling operators

Conditional GANs



- Possibility of better representations for multimodal data generation
- Class-conditional generator and discrmінator
- A look into InfoGAN: Decomposing the data to an incompressible source and latent code that is used to discover object classes

GANs with Inference Models



- Inference mechanism was missing in the original model
- In ALI/BiGANs,
  - The generator consists of two networks: the encoder and the decoder
  - The discriminator has to judge the pairs (x, z): the pair of real data and its representation or the fake data and its latent space

Adversarial Autoencoders

Autoencoders reconstruct the data using an encoder and a decoder.

They learn linear mapping of data from data space to a latent space

Latent-space GAN were introduced as adversarial autoencoders with a loss function that measures the reconstruction error and the difference between the data distribution and encoder output

Training GANs

The cost of training:

$$\max_{\mathcal{D}} \min_{\mathcal{G}} V(\mathcal{G}, \mathcal{D}),$$

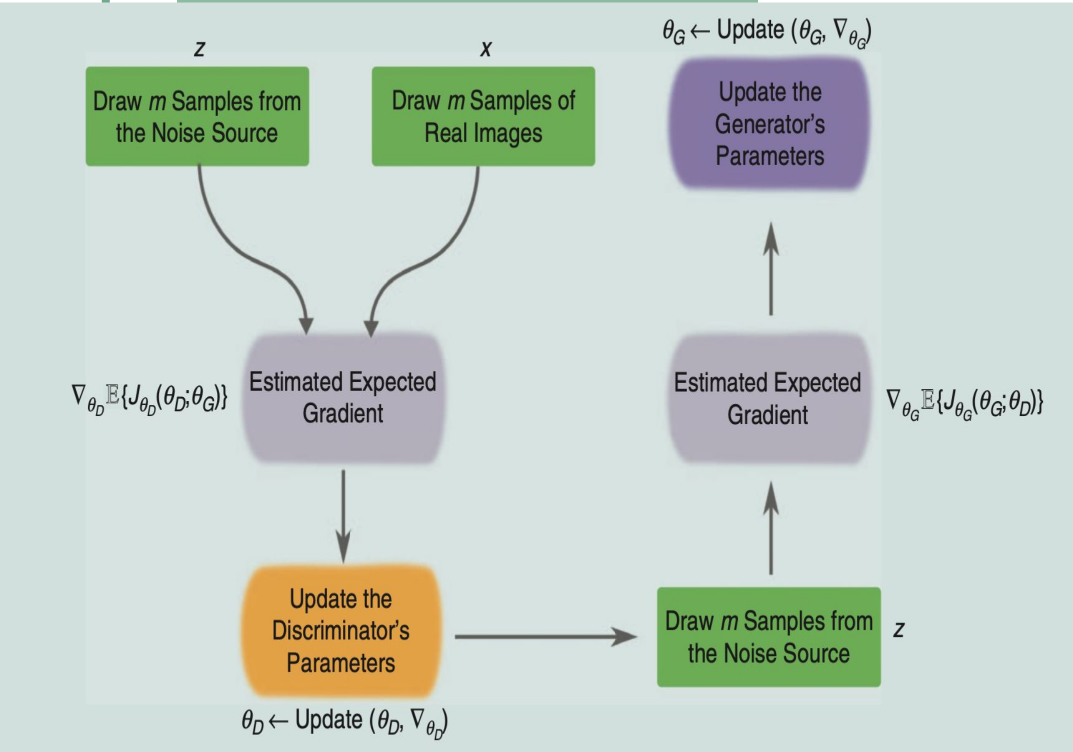
$$V(\mathcal{G}, \mathcal{D}) = \mathbb{E}_{p_{\text{data}}(\mathbf{x})} \log \mathcal{D}(\mathbf{x}) + \mathbb{E}_{p_g(\mathbf{x})} \log(1 - \mathcal{D}(\mathbf{x}))$$

Optimal Discriminator:

$$\mathcal{D}^*(\mathbf{x}) = p_{\text{data}}(\mathbf{x}) / (p_{\text{data}}(\mathbf{x}) + p_g(\mathbf{x})).$$

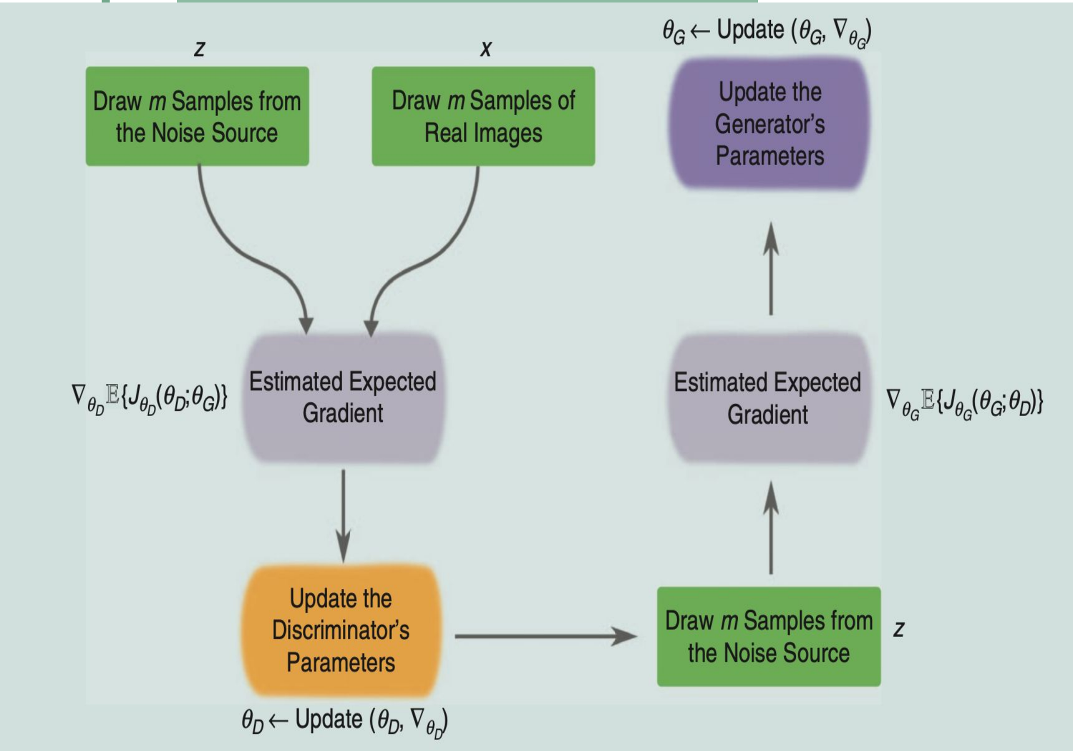
Optimal Generator:

$$p_g(\mathbf{x}) = p_{\text{data}}(\mathbf{x})$$



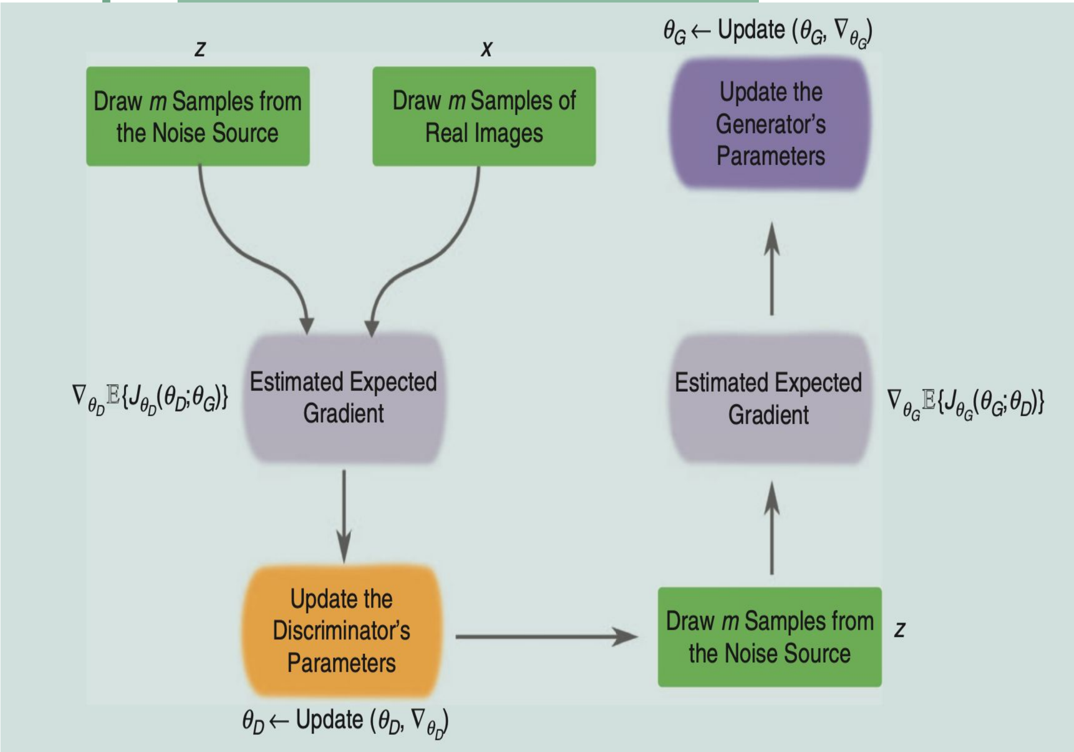
Training Issues

- Converging of the models
- Collapsing of the generator
- Zero loss of the discriminator prevents the updates of the generator



Training Tricks

- Strided and fractionally strided convolutions
- Batch normalization
- Leaky ReLU
- Minibatch discrimination
- Heuristic averaging
- One-sided label smoothing



GAN's Applications: Regression and Classification

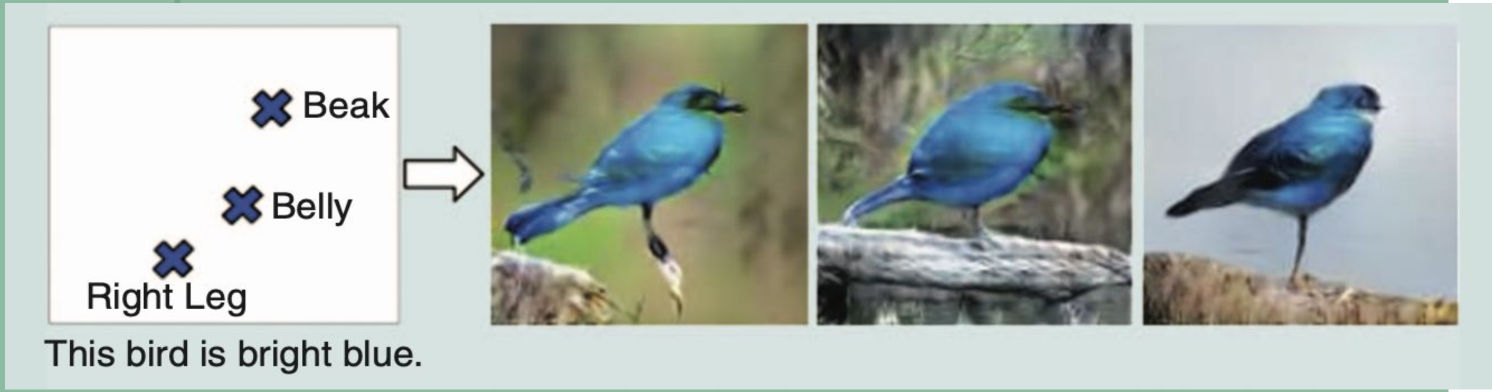
- The discriminator is used as a feature extractor and linear classification models can be applied to its output:
  - The representation in DCGAN and L2-SVM classifier
  - Adversarial training of ALI and applying L2-SVM to the representation
- Using adversarial training to synthesize training data
  - The problem of generalization
  - Adversarial training and multiple GANs were adopted to solve the problem

GANs' Applications: Image Synthesis

- Image synthesis is the core capability of GANs and conditioning cam be applied
  - LAPGAN: Conditioning by inputting D and G with some label information

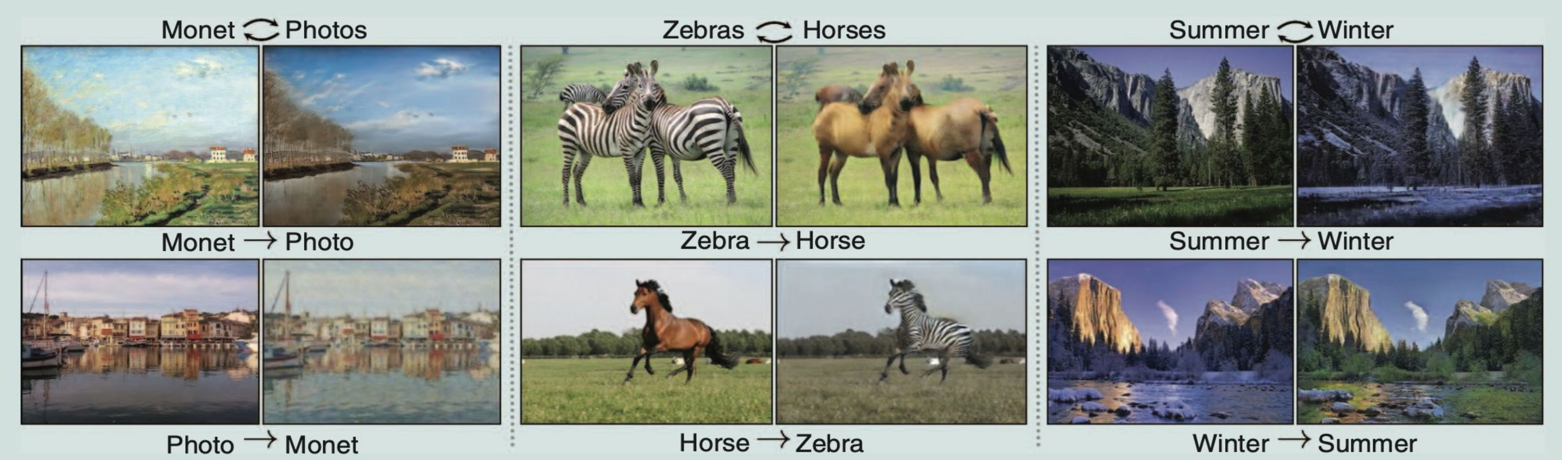
- Reverse Captioning using GAN conditioning
  - Conditioning on text description
  - GAWWN: conditions on image location and textual descriptions

- Image editing



GAN's Applications: Image-to-Image Translation

- Pix2Pix models were used for image-to-image translation
- GAN versions have been offered to do the same thing
  - CycleGAN: preserve the the original image after a cycle of translation and reverse the translation
  - Artistic style transfer



GAN's Applications: Superresolution

- Generating high-resolution images from low-resolution ones
- SRGAN: Inferring photo-realistic details from natural images
  - Adversarial loss, perceptual loss, and regularization loss
- Easy to get training data

Open Questions & Conclusion

GANs are generative models implemented as a min-max game between the discriminator and the generator

Different tricks are adopted to manage the issues while training the GANs

Multiple Variants that address different types of problems ranging from classification to image editing

Partial or complete collapse of the generator

Existing of saddle points rather than local minima as the optimization output

Picking a proper evaluation metric for generative models and GANs

Thanks!

CREDITS: This presentation template was created by Slidesgo, including icons by Flaticon, and infographics & images by Freepik.