Generative Adversarial Networks

TIN0145 - TAA - Aprendizagem Profunda

2023.1 - Prof. Pedro Moura

Generative Adversarial Networks

- We have already seen the idea of deep learning models that can create novel and unique pieces of visual imagery:.
  - We might even be able to call *art*.

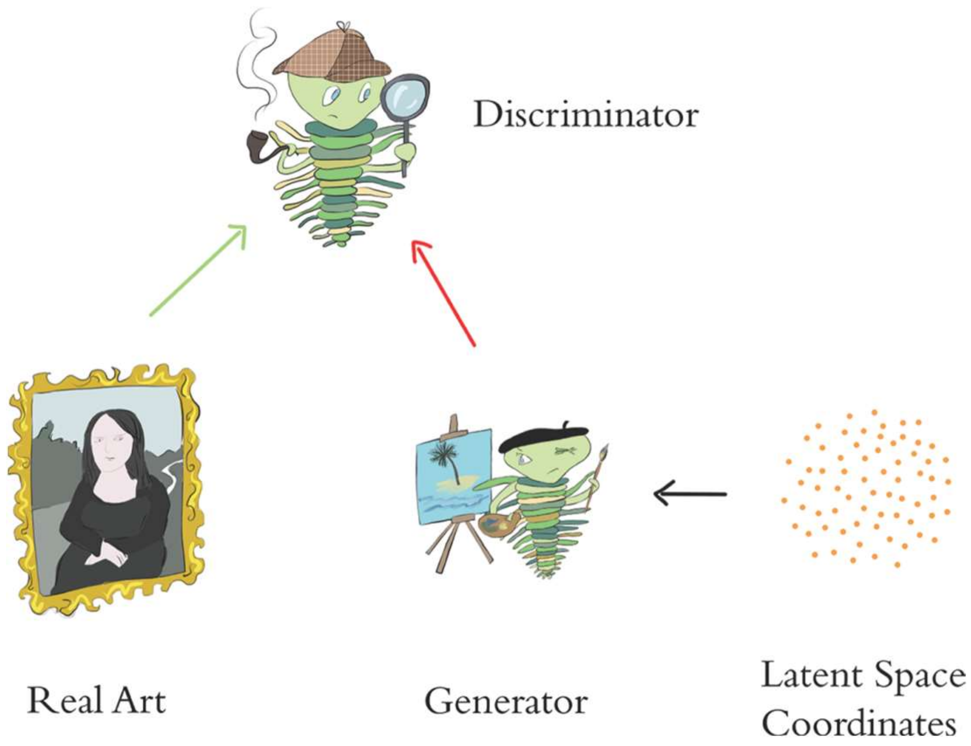
- In this class, we combine the high-level theory with the convolutional networks, the Keras Model class, and a couple of new layer types:
  - Enabling us to code up a **generative adversarial network (GAN)** that outputs images in the style of sketches hand drawn by humans.

Generative Adversarial Networks

- At highest level: two deep learning networks pitted against each other in an **adversarial** relationship.

- One network is a **generator** that produces forgeries of images X the other is a **discriminator** that attempts to distinguish the generator's fakes from the real things.

Generative Adversarial Networks



High-level schematic of a generative adversarial network (GAN). Real images, as well as forgeries produced by the generator, are provided to the discriminator, which is tasked with identifying which are the genuine articles. The orange cloud represents latent space "guidance" that is provided to the forger.

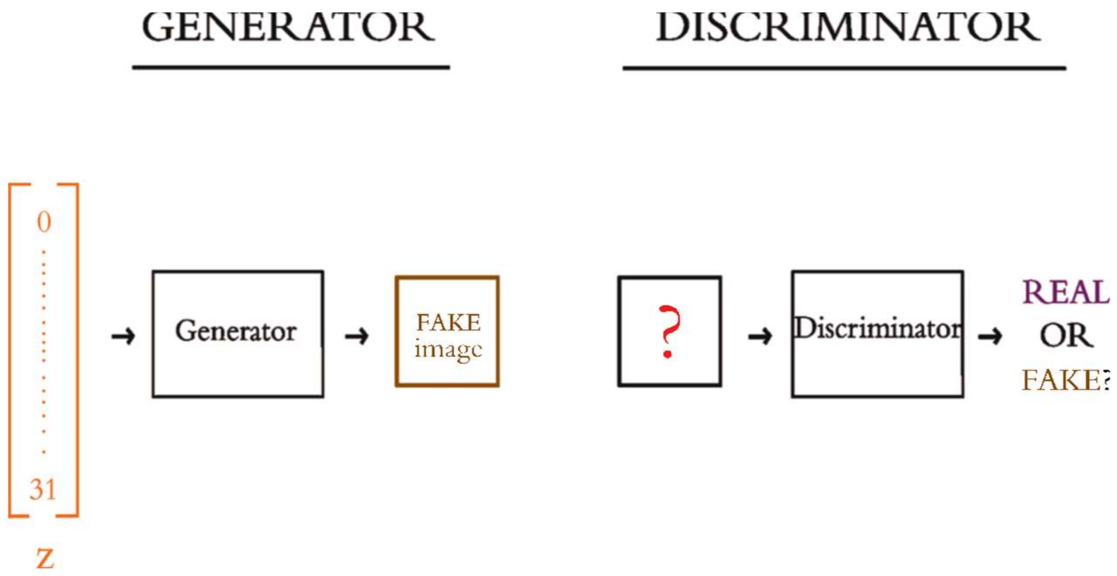
Generative Adversarial Networks

- More-technically:
  - Generator is tasked with receiving a random noise input and turning this into a fake image.
  - Discriminator is a binary classifier of real versus fake images.

- Over several rounds of training:
  - **Generator becomes better at producing more-convincing forgeries.**
  - **Discriminator improves it capacity for detecting the fakes.**

- As training continues, the two models battle it out, trying to outdo one another → **both models become more and more specialized at their respective tasks.**

Generative Adversarial Networks



Highly simplified schematic diagrams of the two models that make up a typical GAN: the generator (left) and the discriminator (right)

Generative Adversarial Networks

- Culminate: the generator producing fakes that are convincing not only to the discriminator network but also **to the human eye**.

- Training a GAN consists of two opposing (**adversarial**) processes:

1) **Discriminator training:** the generator produces fake images - that is, it performs inference only - while the **discriminator learns to tell the fake images from real ones**.

1) **Generator training:** the discriminator judges fake images produced by the generator. Here, it is the discriminator that performs inference only, whereas it's the generator that uses this information to learn - in this case, to learn **how to better fool the discriminator into classifying fake images as real ones**.

Generative Adversarial Networks

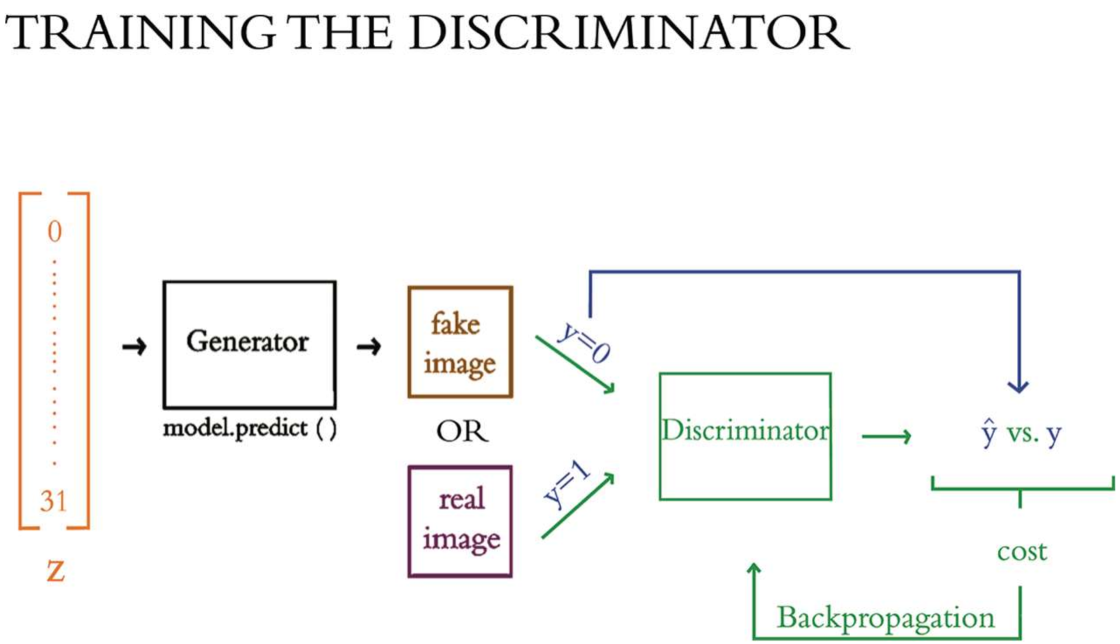
- In each of these two processes:
  - One of the models creates its output (either a fake image or a prediction of whether the image is fake) but is not trained.
  - And the other model uses that output to learn to perform its task better.

- During the overall process of training a GAN, **discriminator training** alternates with **generator training**.

Generative Adversarial Networks

- **Discriminator training:**
  - The generator produces fake images (by inference) that are mixed in with batches of real images and fed into the discriminator for training.
  - The discriminator outputs a prediction (**ŷ**) that corresponds to the likelihood that the image is real.
  - The cross-entropy cost is evaluated for the discriminator's **ŷ** predictions relative to the true **y** labels.
  - Via backpropagation tuning the discriminator's parameters, the cost is minimized in order to train the model to better distinguish real images from fake ones.

Generative Adversarial Networks



Outline of the discriminator training loop. Forward propagation through the generator produces fake images. These are mixed into batches with real images from the dataset and, together with their labels, are used to train the discriminator.

Generative Adversarial Networks

- **Note that during discriminator training, it is only the discriminator network that is learning.**
  - The generator network is not involved in the backpropagation, so it doesn't learn.
- Now let's turn our focus to the process that discriminator training alternates with: the training of the generator.

Generative Adversarial Networks

- **Generator training:**
  - The generator receives a random noise vector z as input and produces a fake image as an output.
    - This random noise vector z corresponds to the **latent-space** vector.
  - The fake images produced by the generator are fed directly into the discriminator as inputs. Crucially to this process, we lie to the discriminator and label all of these fake images as **real** (**y** = 1).
  - The discriminator outputs **ŷ** predictions as to whether a given input image is real or fake.
  - Cross-entropy cost here is used to tune the parameters of the generator network. More specifically, the generator learns how convincing its fake images are to the discriminator network.

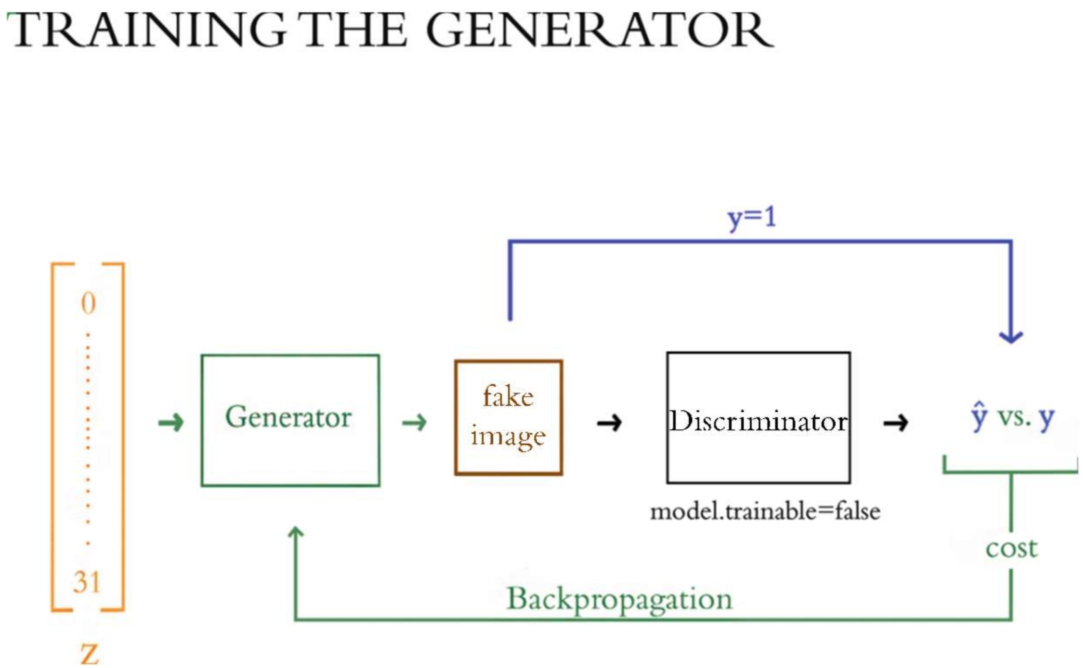
Generative Adversarial Networks

- By minimizing this cost, the generator will learn to produce forgeries that the discriminator mistakenly labels as real - forgeries that may even appear to be real to the human eye.

- **So, during generator training, it is only the generator network that is learning.**

- Later we will see how to freeze the discriminator's parameters so that backpropagation can tune the generator's parameters without influencing the discriminator in any way.

Generative Adversarial Networks



An outline of the generator training loop. Forward propagation through the generator produces fake images, and inference with the discriminator scores these images. The generator is improved through backpropagation.

Referências

1. Capítulo 12 do livro *“Deep Learning Illustrated: A Visual, Interactive Guide to Artificial Intelligence”* de Jon Krohn, Grant Beyleveld e Aglaé Bassens.