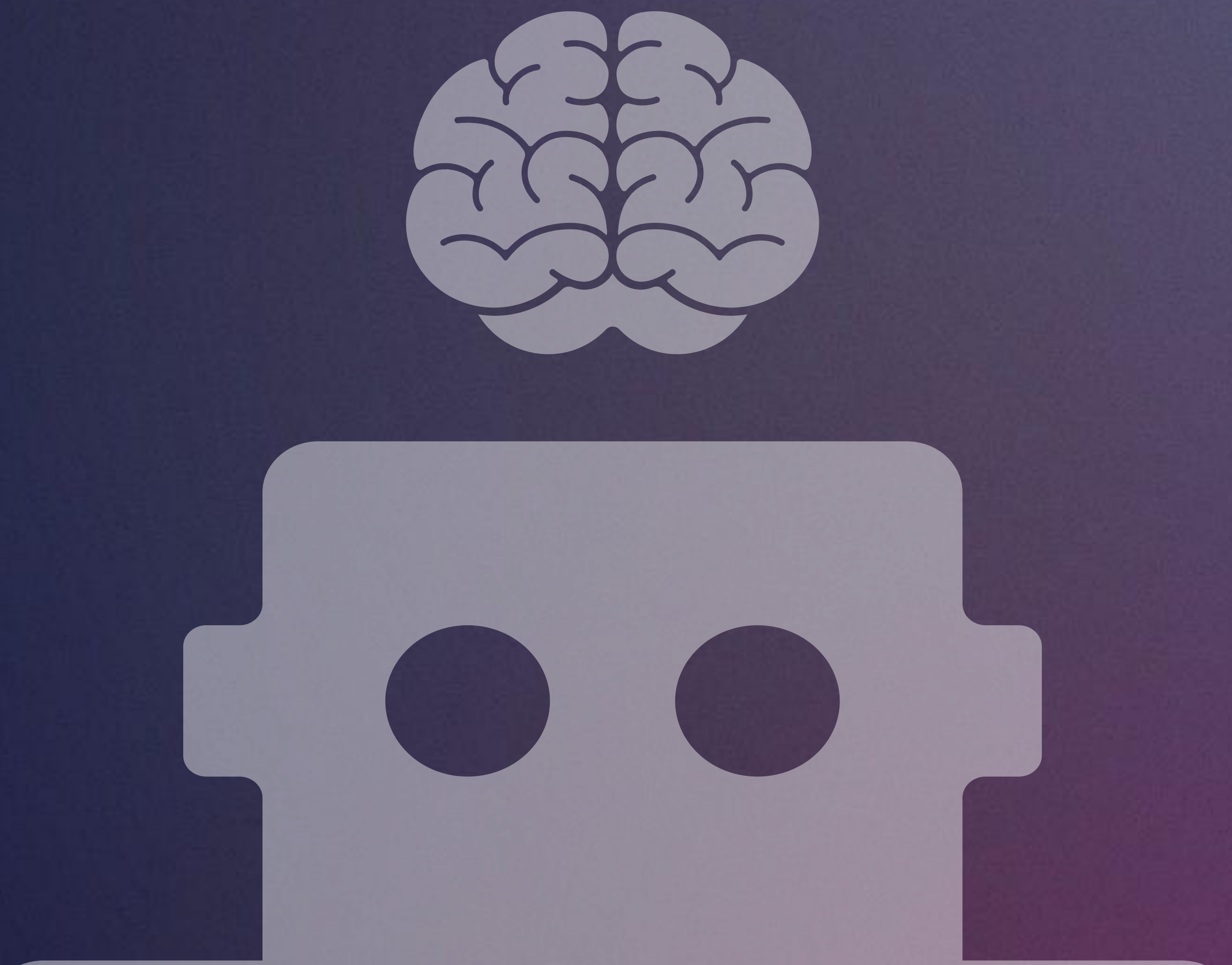
INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
Ceará

HUAWEI

Generative Adversarial Networks



Prof. Me. Saulo A. F. Oliveira
saulo.oliveira@ifce.edu.br

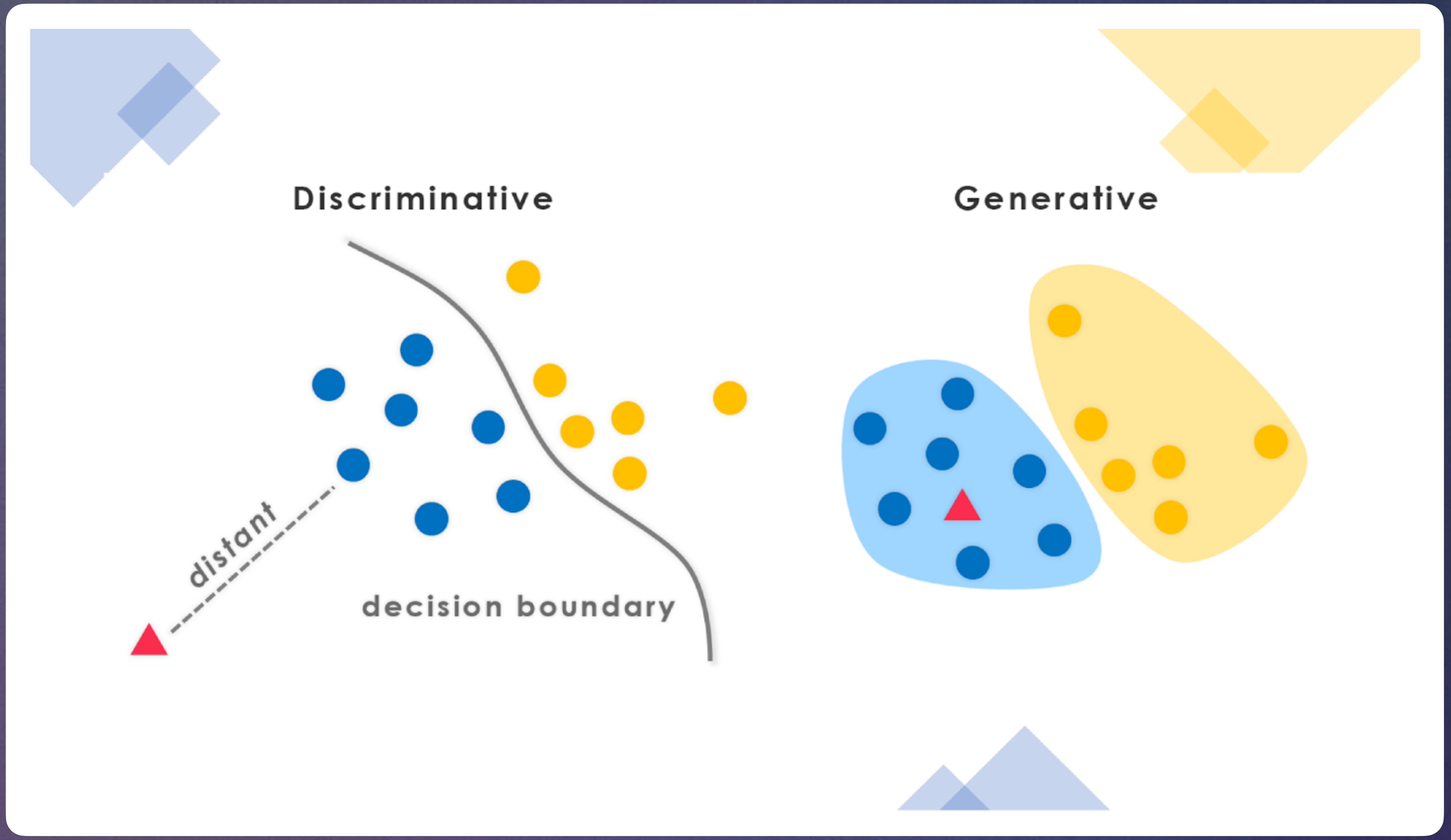
01

Main Concepts of GANs

Generative v.s. Discriminative models

- In general, a Discriminative model models the decision boundary between the classes. A Generative Model explicitly models the actual distribution of each class. In final, both of them are predicting the conditional probability $p(y|x)$. But Both models learn different probabilities and are generally used in **supervised learning problems**.

- A Generative model learns the **joint probability distribution** $p(x|y)$, predicting the conditional probability with the **Bayes Theorem's** help. In contrast, a Discriminative model learns the **conditional probability** distribution $p(y|x)$.



https://www.kdnuggets.com/2020/05/microsoft-research-three-efforts-advance-deep-generative-models.html

- **GENERATIVE CLASSIFIERS**
  - Naïve Bayes
  - Bayesian networks
  - Markov random fields
  - Hidden Markov Models (HMM)

- **DISCRIMINATIVE CLASSIFIERS**
  - Logistic regression
  - Scalar Vector Machine
  - Traditional neural networks
  - Nearest neighbor
  - Conditional Random Fields (CRF)s

https://medium.com/@mlengineer/generative-and-discriminative-models-af5637a66a3

Generative v.s. Discriminative models

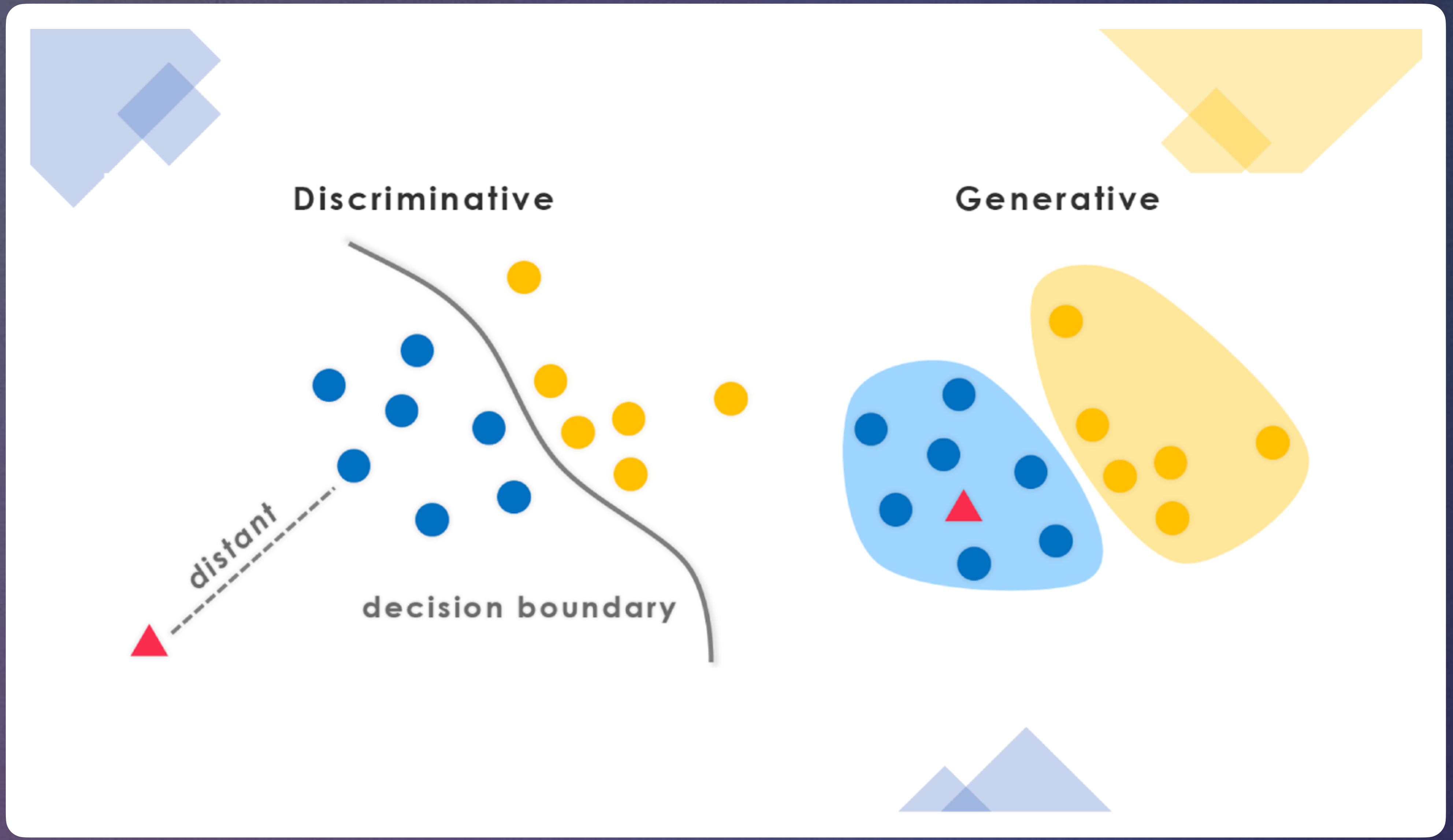
- In general, a Discriminative model models the decision boundary between the classes. A Generative Model explicitly models the actual distribution of each class. In final, both of them are predicting the conditional probability $p(y|x)$. But Both models learn different probabilities and are generally used in **supervised learning problems**.

Discriminative algorithms **map features to labels**, and they are concerned only with this correlation. One way of thinking about **generative algorithms** is that they **do the opposite**. Instead of predicting a label with certain features, **they try to predict the features with** a particular **label**.

- ... **ibution** ... **e Bayes** ... l learns

- ... SSIFIERS

  - ... ine
  - ... etworks
  - Hidden Markov Models (HMM)
  - Nearest neighbor
  - Conditional Random Fields (CRF)s



https://www.kdnuggets.com/2020/05/microsoft-research-three-efforts-advance-deep-generative-models.html

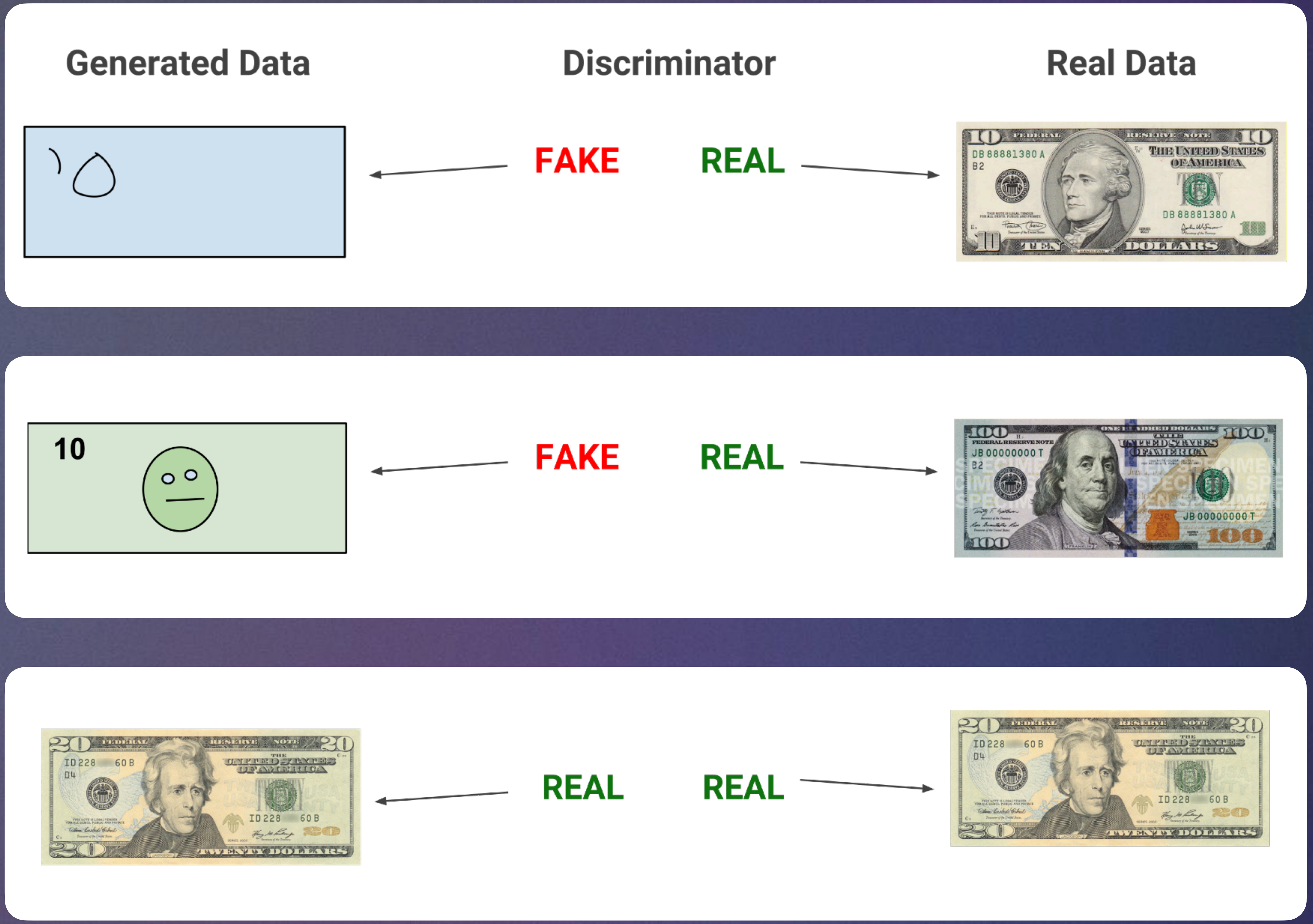
https://medium.com/@mlengineer/generative-and-discriminative-models-af5637a66a3

Overview of GAN Structure (1)

A GAN HAS TWO PARTS:

- The generator learns to generate plausible data. The generated instances become negative training examples for the discriminator.
- The discriminator learns to distinguish the generator's fake data from real data. The discriminator penalizes the generator for producing implausible results.

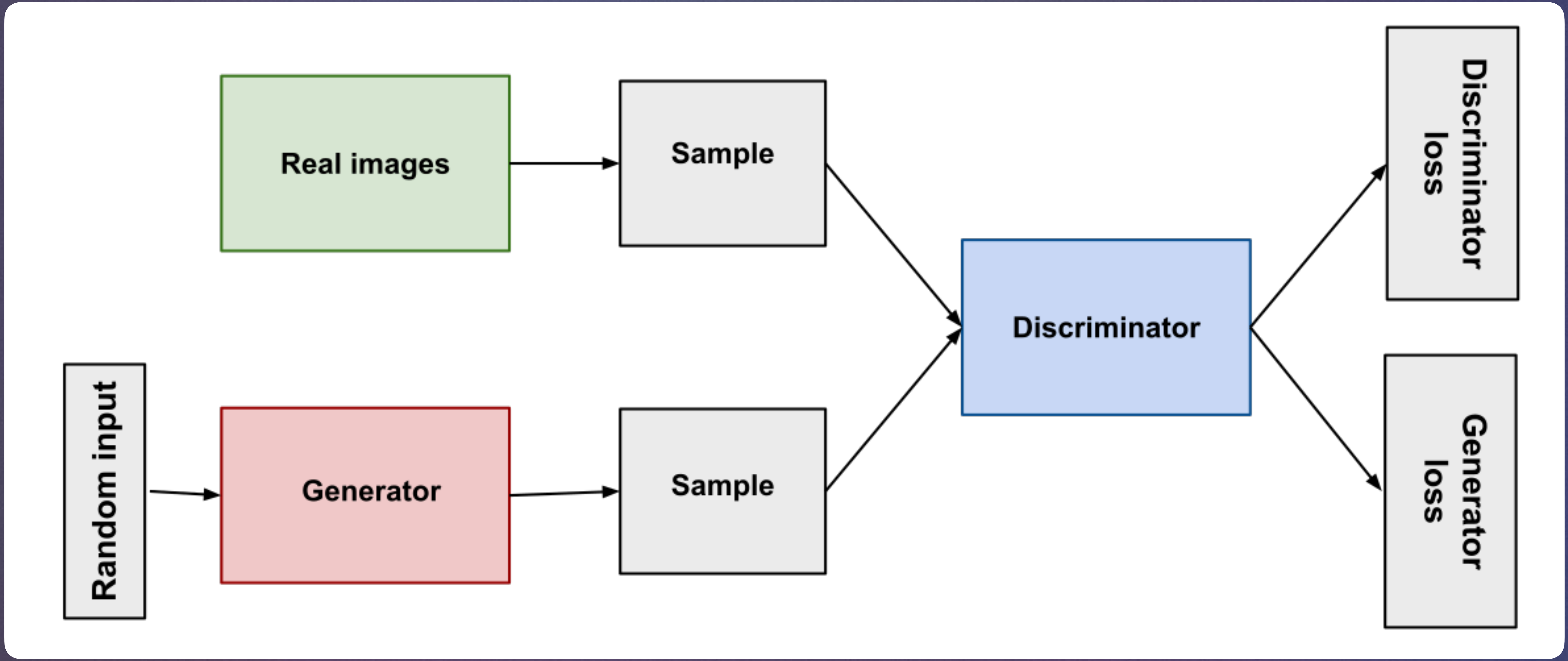
When training begins, the generator produces obviously fake data, and the discriminator quickly learns to tell that it's fake:



https://developers.google.com/machine-learning/gan/gan_structure

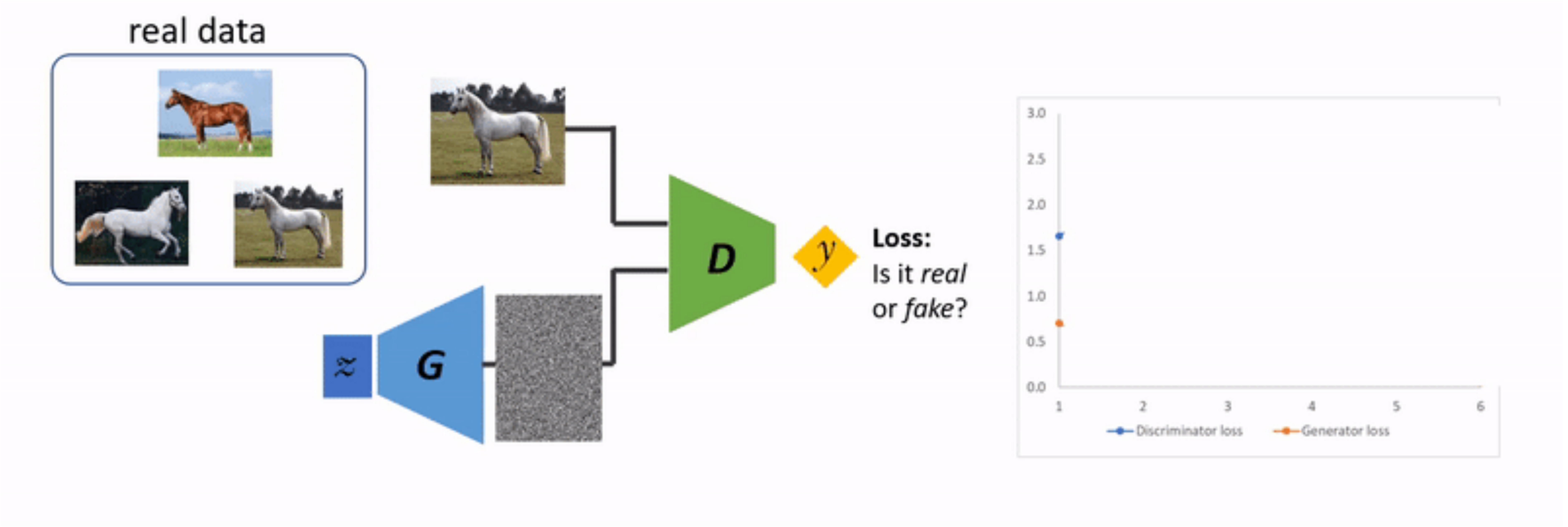
Overview of GAN Structure (2)

Both the generator and the discriminator are neural networks. The generator output is connected directly to the discriminator input. Through backpropagation, the discriminator's classification provides a signal that the generator uses to update its weights.



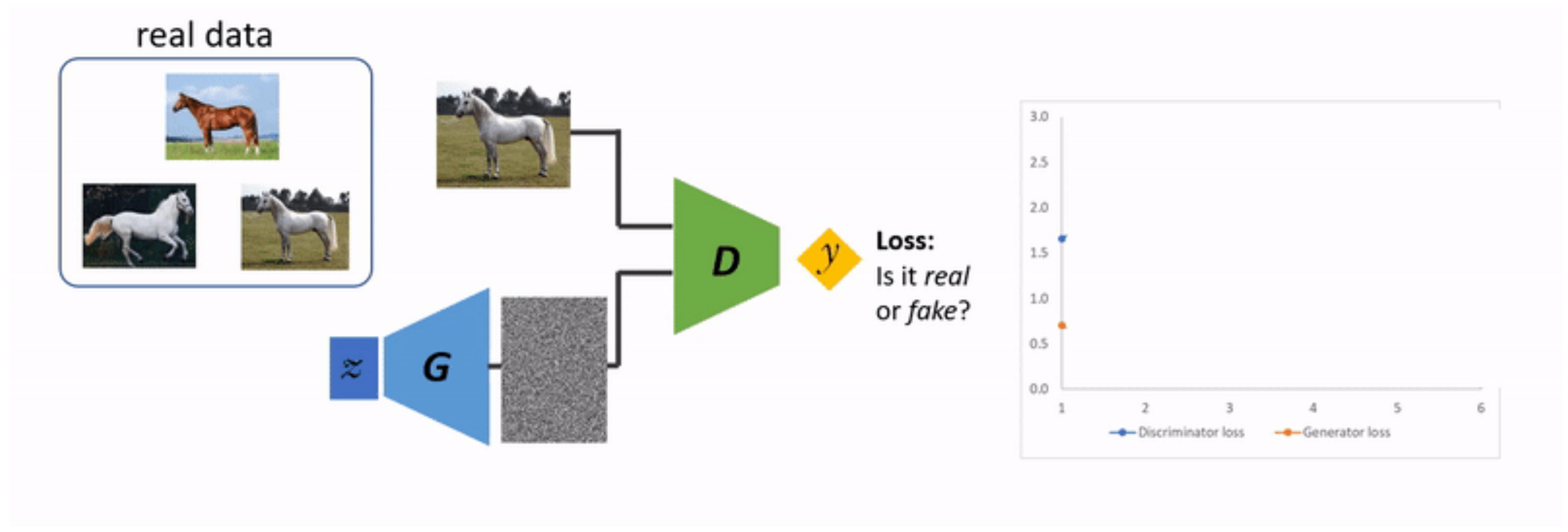
https://developers.google.com/machine-learning/gan/gan_structure

Overview of GAN Structure (3)



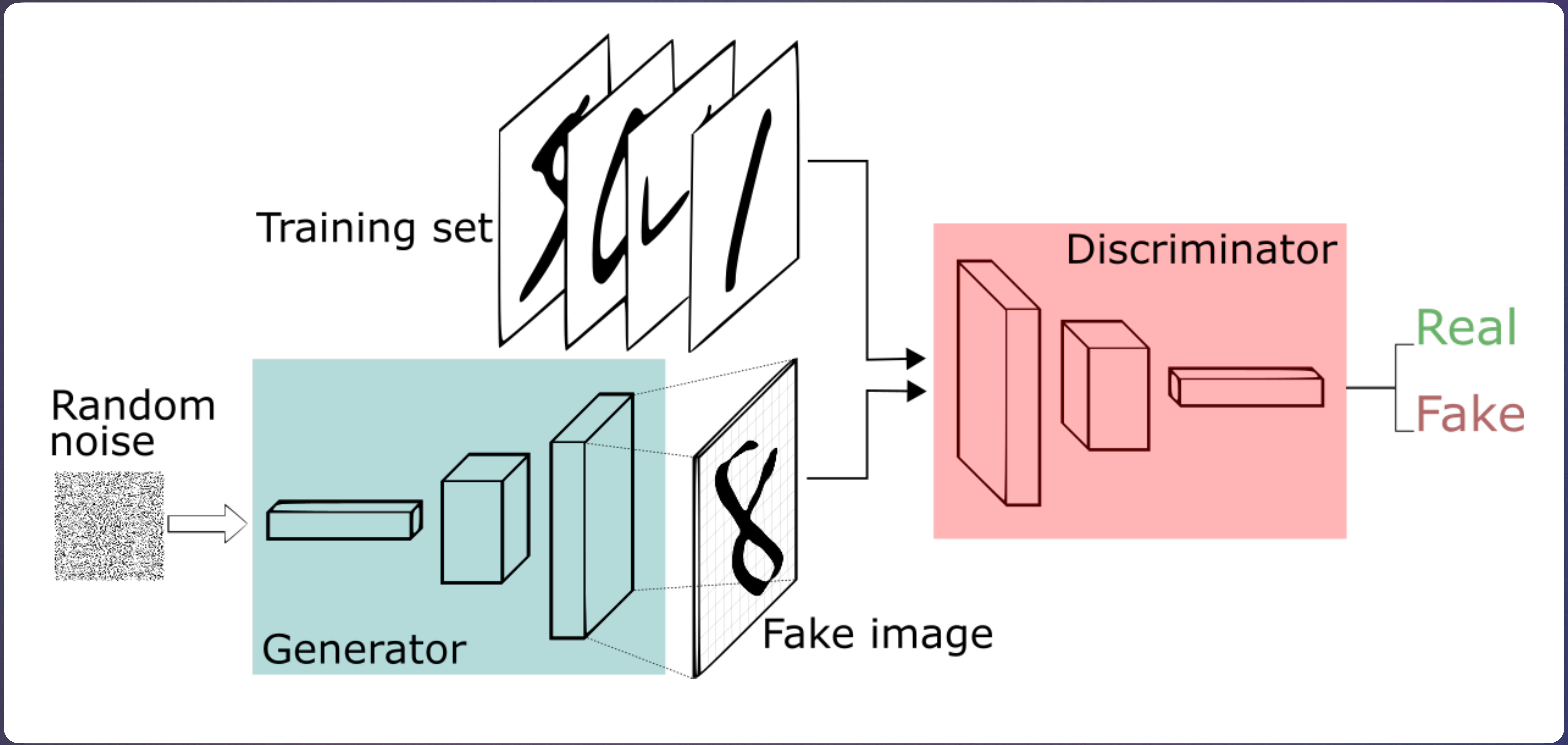
https://medium.com/@jamaltoutouh/lipizzaner-a-framework-for-co-evolutionary-distributed-gan-training-7a725fb17e49

Overview of GAN Structure (3)

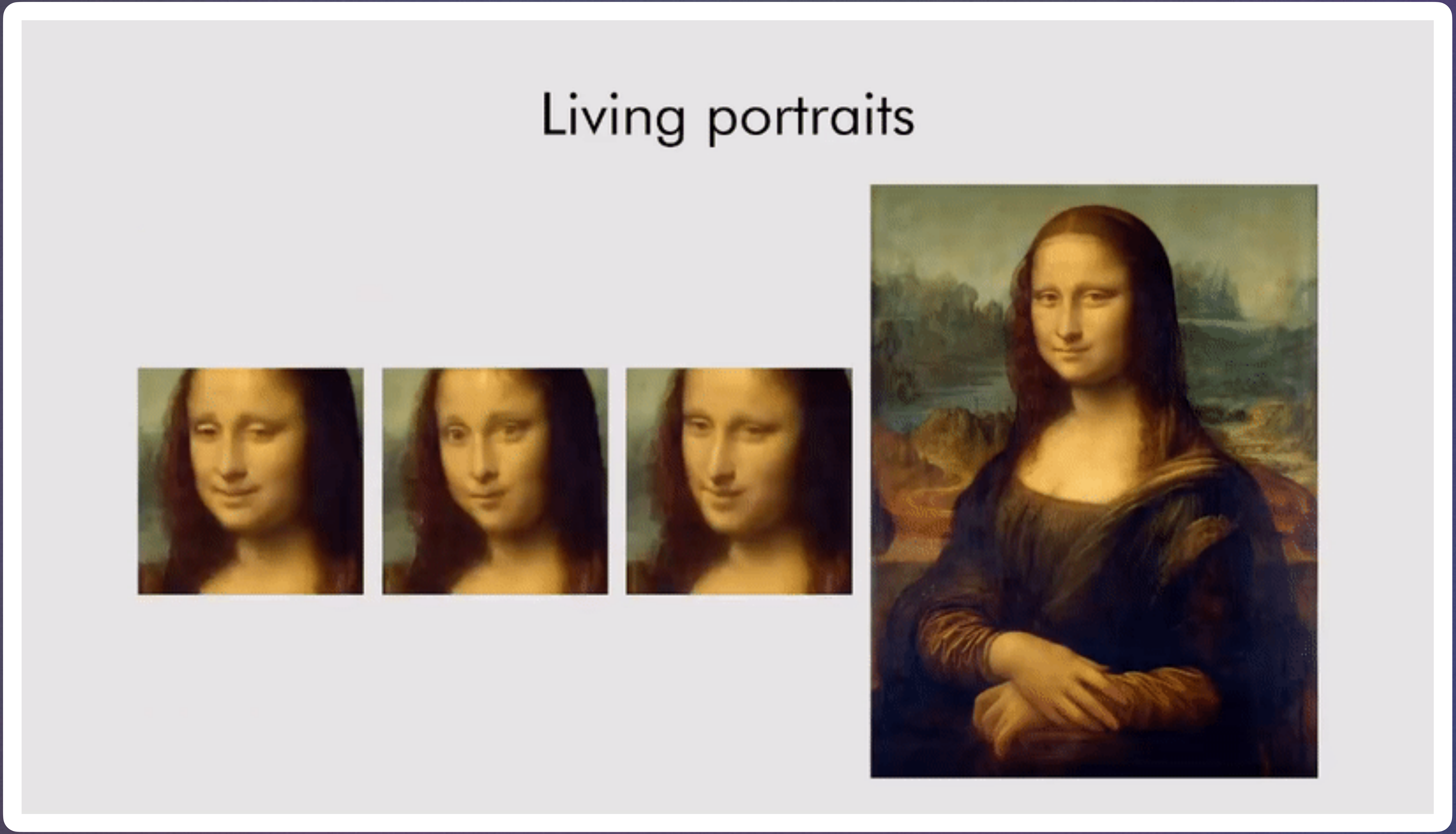


https://medium.com/@jamaltoutouh/lipizzaner-a-framework-for-co-evolutionary-distributed-gan-training-7a725fb17e49

Overview of GAN Structure (4)

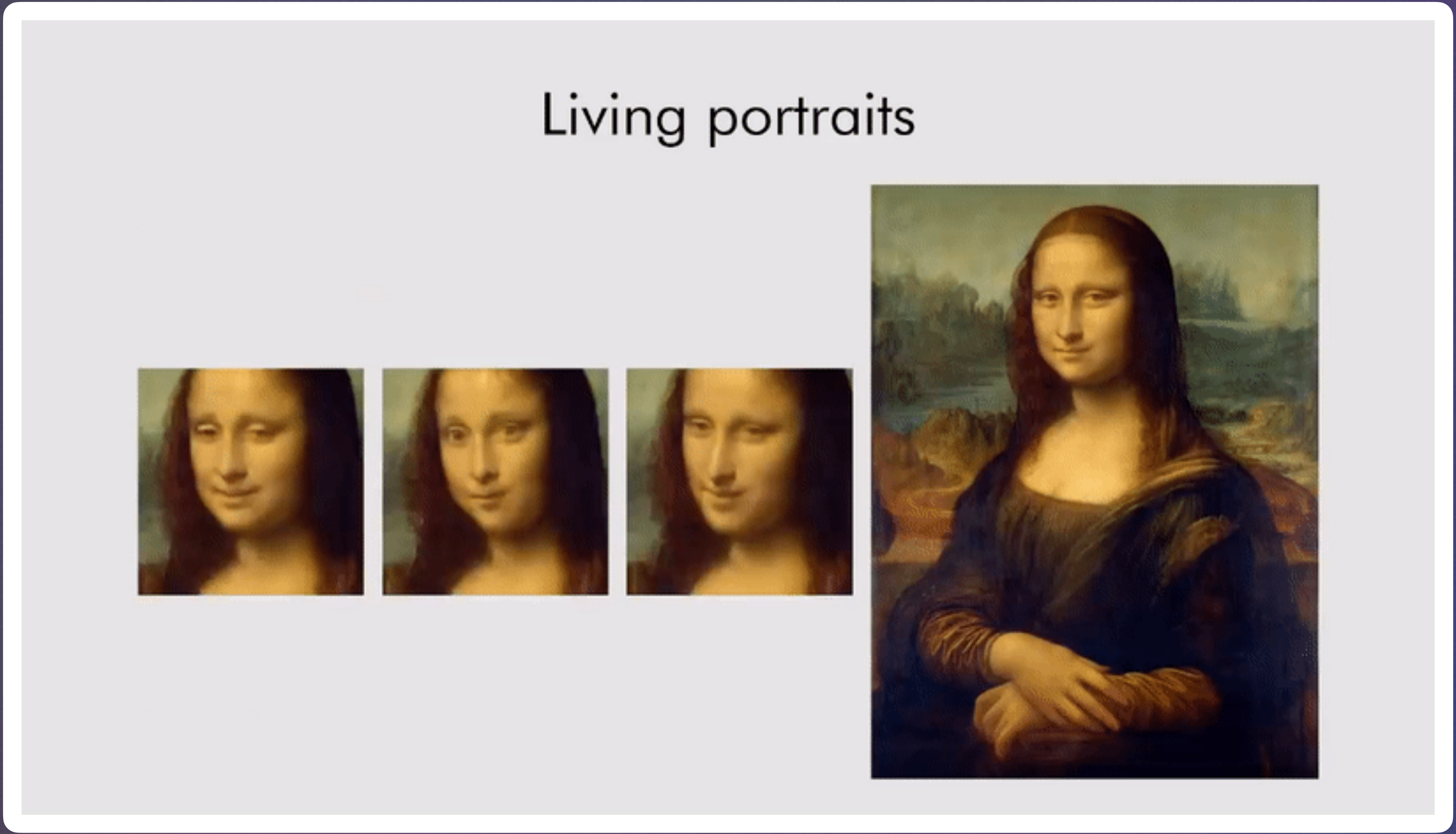


Overview of GAN Structure (5)



https://news.artnet.com/art-world/mona-lisa-deepfake-video-1561600

Overview of GAN Structure (5)



https://news.artnet.com/art-world/mona-lisa-deepfake-video-1561600

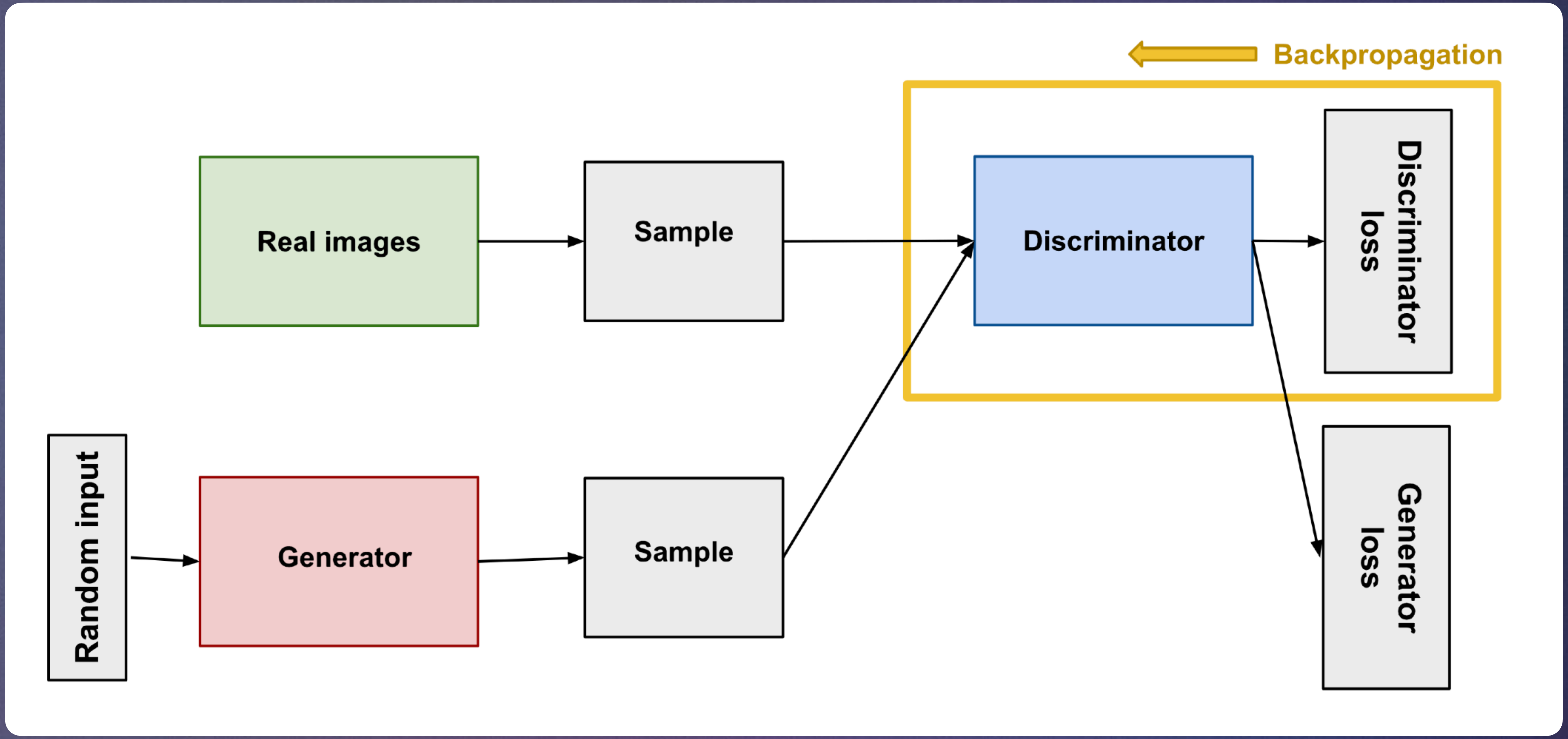
02

The Discriminator

Discriminator (1)

DISCRIMINATOR TRAINING DATA:

- **Real data** instances, such as real pictures of people. The discriminator employ these instances as positive examples during training.
- **Fake data** instances created by the generator. The discriminator uses such instances as negative examples during training.

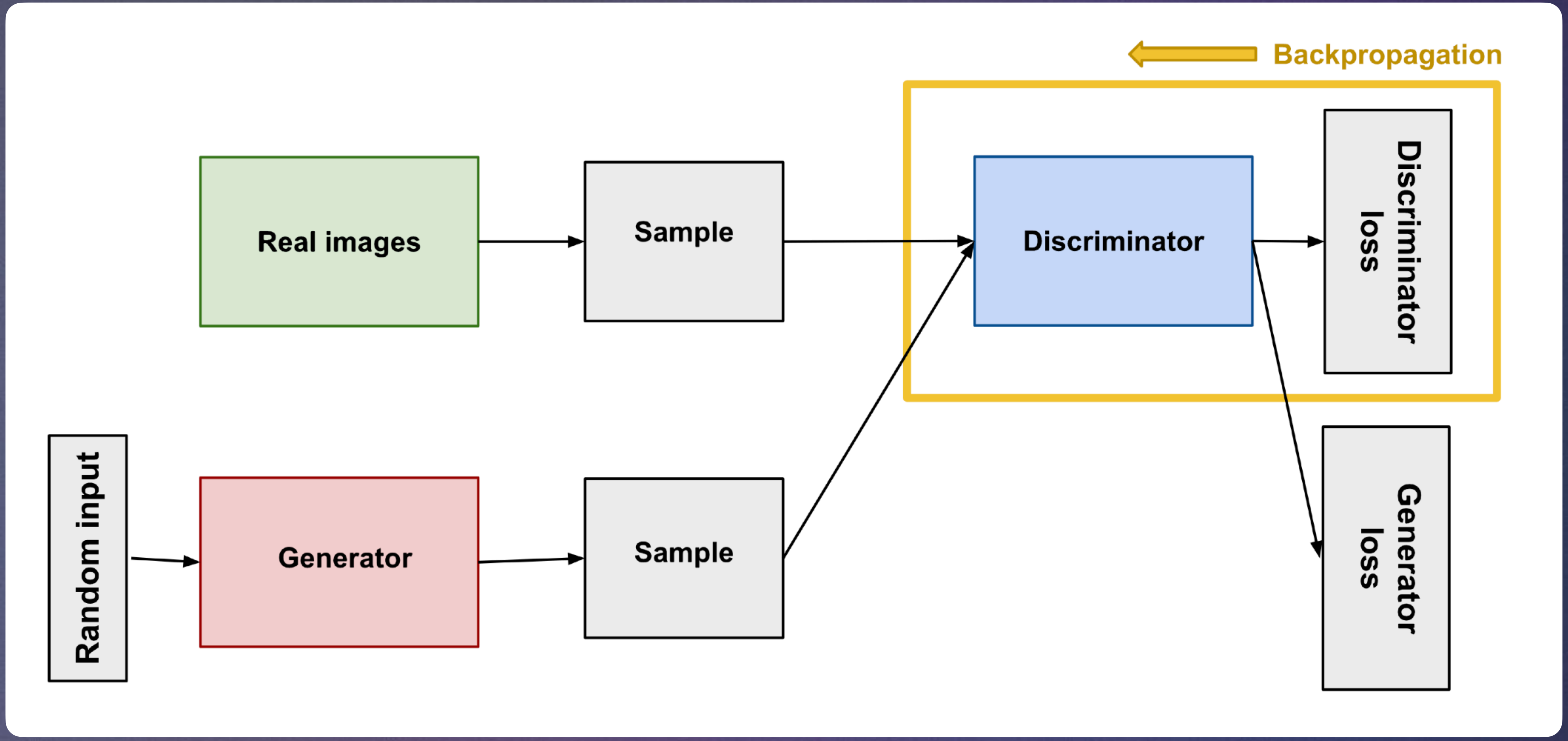


https://developers.google.com/machine-learning/gan/gan_structure

Discriminator (2)

DURING DISCRIMINATOR TRAINING:

- The discriminator classifies both the real data and the fake data from the generator.
- The discriminator loss penalizes the discriminator for misclassifying a real instance as fake or a fake instance as real.
- The **discriminator** updates its weights through backpropagation from the **discriminator loss** through the discriminator network.



https://developers.google.com/machine-learning/gan/gan_structure

03

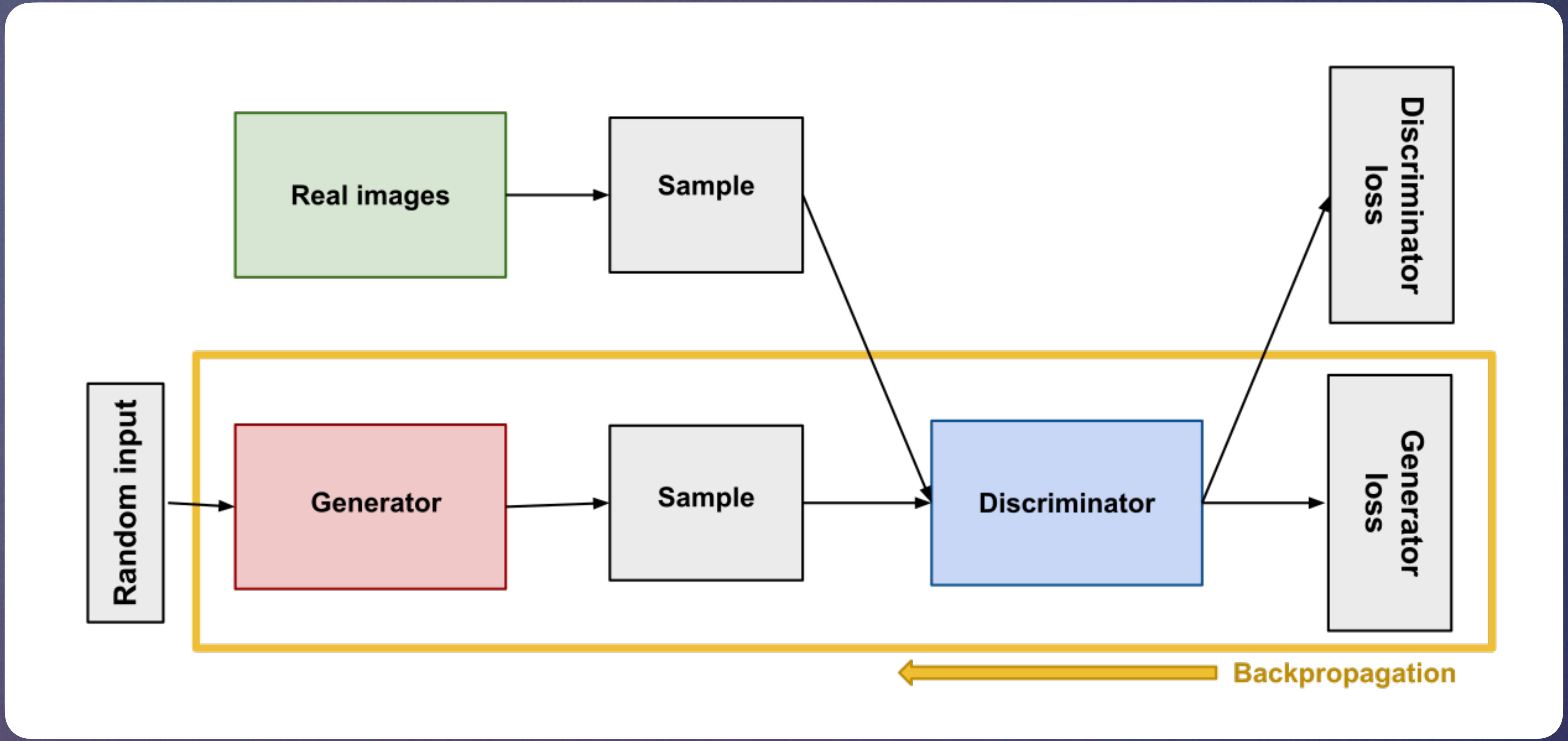
The Generator

Generator (1)

The generator part of a GAN learns to create fake data by incorporating feedback from the discriminator. It learns to make the discriminator classify its output as real. Generator training requires tighter integration between the generator and the discriminator. The portion of the GAN that trains the generator includes:

- random input;
- generator network, which transforms the random input into a data instance;
- discriminator network, which classifies the generated data;
- discriminator output;
- generator loss, which penalizes the generator for failing to fool the discriminator.

Experiments suggest that the distribution of the noise doesn't matter much, so we can choose something that's easy to sample from, like a uniform distribution.

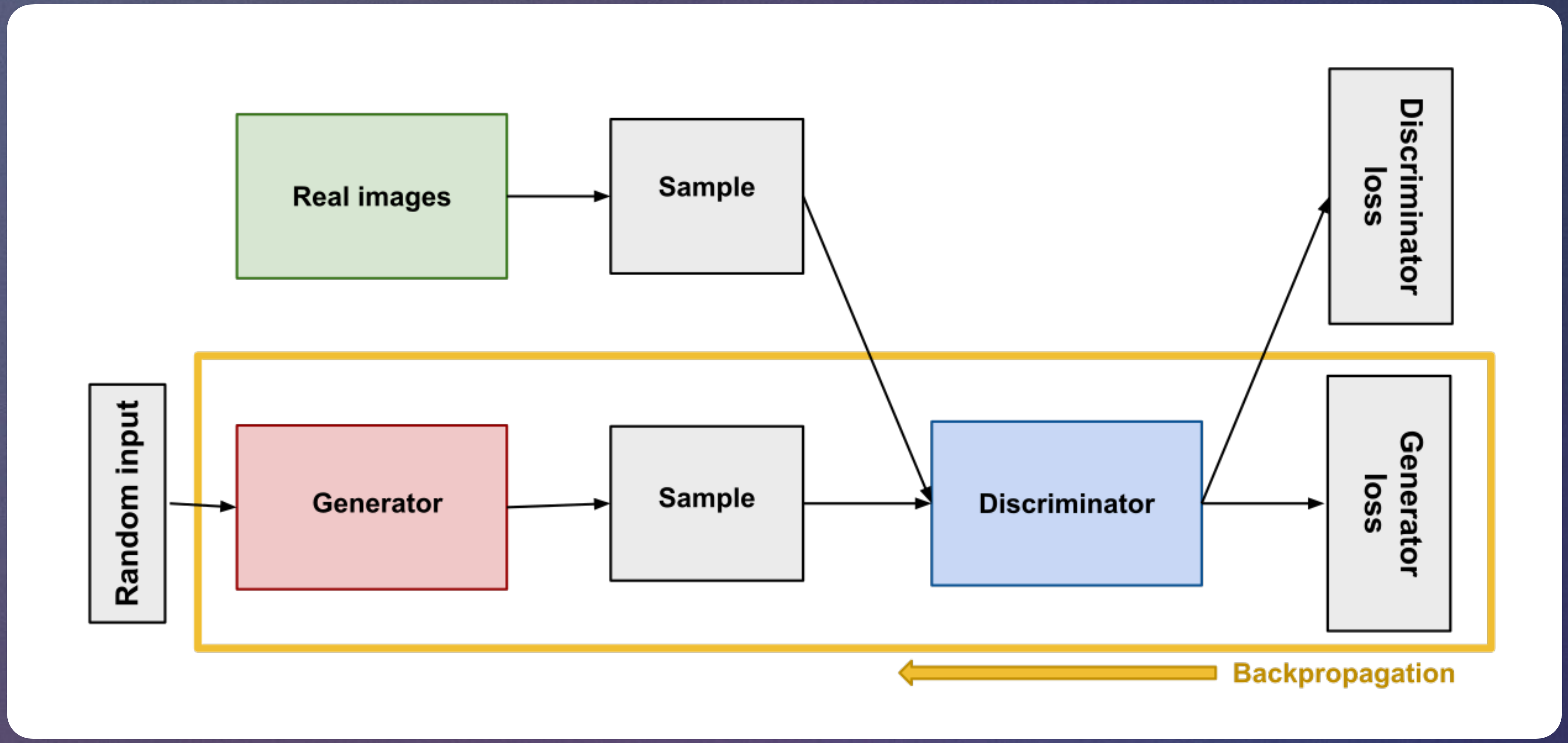


https://developers.google.com/machine-learning/gan/gan_structure

Generator (2)

To train a neural net, we alter the net's weights to reduce the error or loss of its output. In our GAN, however, the generator is not directly connected to the loss that we're trying to affect. The generator feeds into the discriminator net, and the discriminator produces the output we're trying to affect. The generator loss penalizes the generator for producing a sample that the discriminator network classifies as fake. At the same time, we don't want the discriminator to change during generator training. Trying to hit a moving target would make a hard problem even harder for the generator.

- So we train the generator with the following procedure:
- Sample random noise.
- Produce generator output from sampled random noise.
- Get discriminator "Real" or "Fake" classification for generator output.
- Calculate loss from discriminator classification.
- Backpropagate through both the discriminator and generator to obtain gradients.
- Use gradients to change only the generator weights.



https://developers.google.com/machine-learning/gan/gan_structure

04

GAN Training

GAN Training (1)

BECAUSE A GAN CONTAINS TWO SEPARATELY TRAINED NETWORKS, ITS TRAINING ALGORITHM MUST ADDRESS TWO COMPLICATIONS:

- GANs must juggle two different kinds of training (generator and discriminator).
- GAN convergence is hard to identify.

ALTERNATING TRAINING

The generator and the discriminator have different training processes. So how do we train the GAN as a whole?

GAN training proceeds in alternating periods:

- The discriminator trains for one or more epochs.
- The generator trains for one or more epochs.

Repeat steps 1 and 2 to continue to train the generator and discriminator networks.

We keep the generator constant during the discriminator training phase. As discriminator training tries to figure out how to distinguish real data from fake, it has to learn how to recognize the generator's flaws. That's a different problem for a thoroughly trained generator than it is for an untrained generator that produces random output.

Similarly, we keep the discriminator constant during the generator training phase. Otherwise the generator would be trying to hit a moving target and might never converge.

It's this back and forth that allows GANs to tackle otherwise intractable generative problems. We get a toehold in the difficult generative problem by starting with a much simpler classification problem.

Conversely, if you can't train a classifier to tell the difference between real and generated data even for the initial random generator output, you can't get the GAN training started.

https://developers.google.com/machine-learning/gan/gan_structure

GAN Training (2)

OPTIMIZATION OBJECTIVE:

- Value function:

$$\min_G \max_D (G, D) = \mathrm{E}_x \left[\log D(x)\right] + \mathrm{E}_z \left[\log(1 - D(G(z)))\right].$$

- In this function:
  - ➡ $D(x)$ is the discriminator's estimate of the probability that real data instance x is real.
  - ➡ $\mathrm{E}_x[\cdot]$ is the expected value over all real data instances.
  - ➡ $G(z)$ is the generator's output when given noise z.
  - ➡ $D(G(z))$ is the discriminator's estimate of the probability that a fake instance is real.
  - ➡ $\mathrm{E}_z[\cdot]$ is the expected value over all random inputs to the generator (in effect, the expected value over all generated fake instances $G(z)$).

- The formula derives from the cross-entropy between the real and generated distributions.
- The generator can't directly affect the $\log(D(x))$ term in the function, so, for the generator, minimizing the loss is equivalent to minimizing $\log(1 - D(G(z)))$.
- In the early training stage, when the outcome of $G(\cdot)$is very poor, $D(\cdot)$ determines that the generated sample is fake with high confidence, because the sample is obviously different from training data. In this case, $\log(1 - D(G(z)))$ is saturated (where the gradient is 0, and iteration cannot be performed). Therefore, we choose to train $G(\cdot)$ only by minimizing $[-\log(D(G(z))]$.

https://developers.google.com/machine-learning/gan/gan_structure

GAN Training Convergence

CONVERGENCE

As the generator improves with training, the discriminator performance gets worse because the discriminator can't easily tell the difference between real and fake. If the generator succeeds perfectly, then the discriminator has a 50% accuracy.

In effect, the discriminator flips a coin to make its prediction. This progression poses a problem for convergence of the GAN as a whole: the discriminator feedback gets less meaningful over time. If the GAN continues training past the point when the discriminator is giving completely random feedback, then the generator starts to train on junk feedback, and its own quality may collapse.

For a GAN, convergence is often a fleeting, rather than stable, state.

https://developers.google.com/machine-learning/gan/gan_structure

References

- Jesus Rodrigues. Microsoft Research Unveils Three Efforts to Advance Deep Generative Models. https://www.kdnuggets.com/2020/05/microsoft-research-three-efforts-advance-deep-generative-models.html. 2020, Accessed on Mar 2021.
- Prathap Manohar Joshi. Generative VS Discriminative Models. https://medium.com/@mlengineer/generative-and-discriminative-models-af5637a66a3. 2018, Accessed on Mar 2021.
- Thalles Silva. A Short Introduction to Generative Adversarial Networks. https://sthalles.github.io/intro-to-gans/. 2017, Accessed on Mar 2021.
- HUAWEI. Deep Learning Overview. 2020, Accessed on Mar 2021.
- Google. Overview of GAN Structure. https://developers.google.com/machine-learning/gan/gan_structure. 2021, Accessed on Mar 2021.
- Jamal Toutouh. Lipizzaner: a framework for co-evolutionary distributed GAN training. https://medium.com/@jamaltoutouh/lipizzaner-a-framework-for-co-evolutionary-distributed-gan-training-7a725fb17e49. 2020, Accessed on Mar 2021.
- Taylor Dafoe. Russian Researchers Used AI to Bring the Mona Lisa to Life and It Freaked Everyone Out. See the Video Here. https://news.artnet.com/art-world/mona-lisa-deepfake-video-1561600. 2019, Accessed on Mar 2021.

Thank you for your attention!

🙏 ☺️

Prof. Me. Saulo A. F. Oliveira
saulo.oliveira@ifce.edu.br