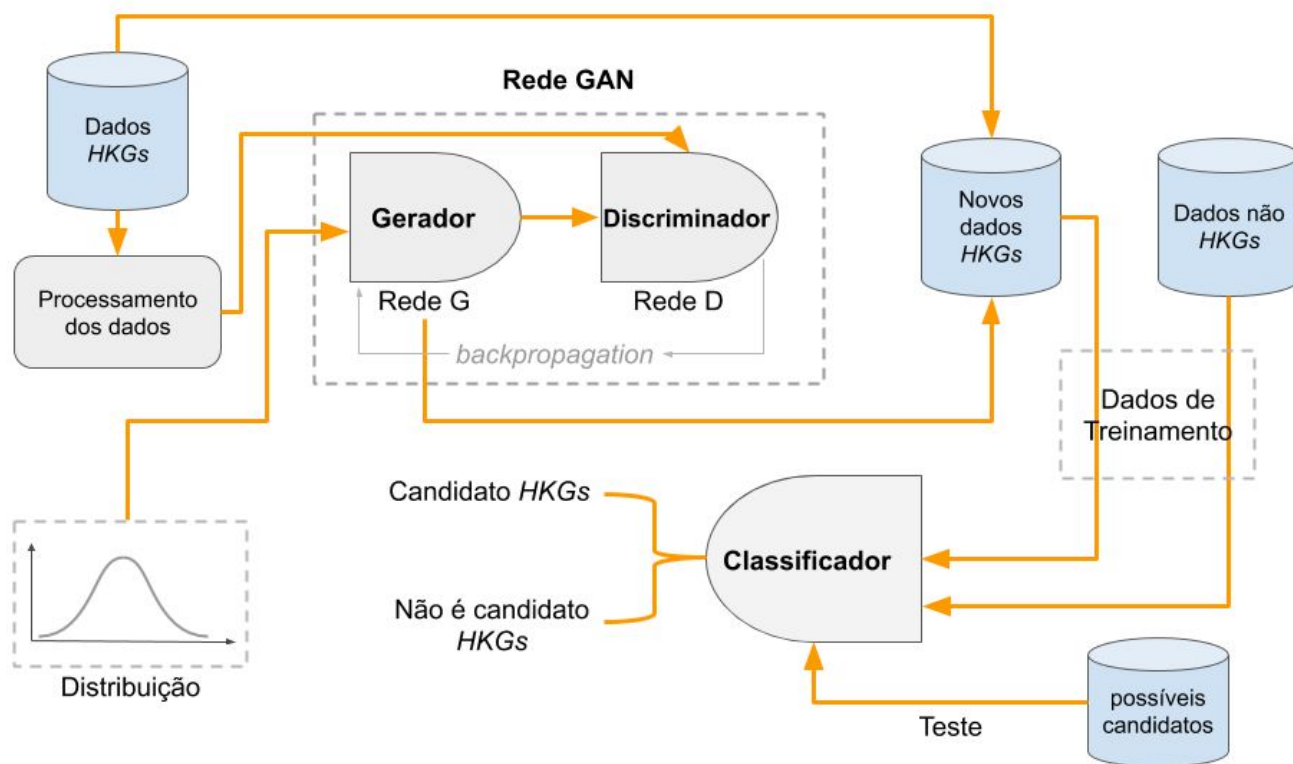

Generative Adversarial Nets para identificação de genes *housekeeping* mediante classificação

— Edwin Jahir Rueda Rojas —

Metodologia

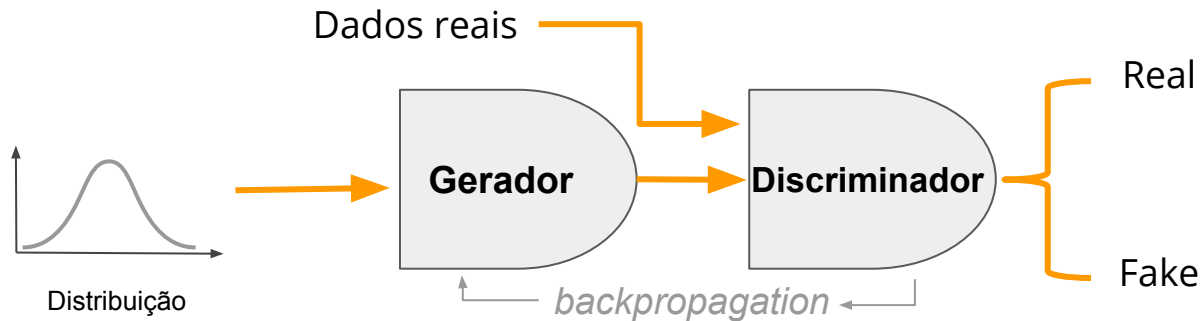


Rede GAN

É um modelo max-min onde a rede *Geradora* é treinada para maximizar o erro da rede *Discriminadora*.

$$\min_G \max_D V(G, D) = \mathbb{E}_{x \sim p_{data}(x)} [\log(D)] + \mathbb{E}_{z \sim p_z(x)} [\log(1 - D(G(z)))]$$

$$D(x) = \frac{p_{data}(x)}{p_{data}(x) + p_g(x)} = \frac{1}{2}$$



Implementação da GAN

for número de epochs **do**

for batches in data train **do**

$$real = \{x_1, x_2, \dots, x_m\}$$

$$fake = \{z_1, z_2, \dots, z_m\}$$

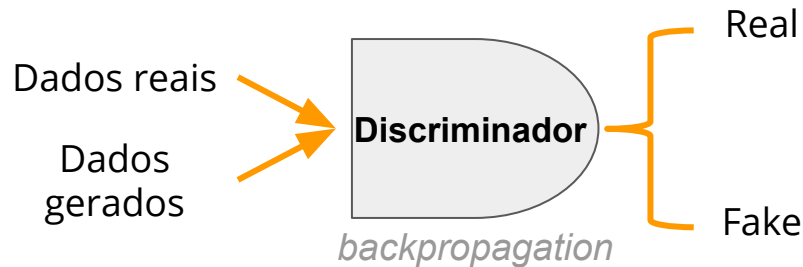
- *treinamento do **Discriminador*** $\{real, fake\}$

$$D_G = \{w_1, w_2, \dots, w_n\}$$

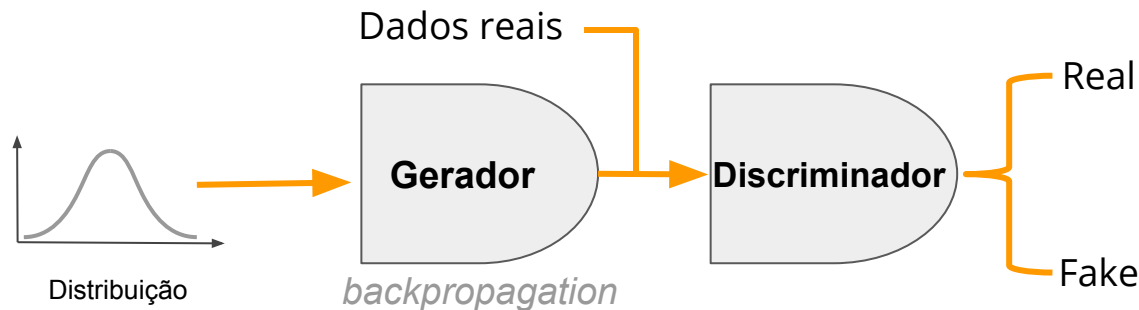
- *treinamento da GAN* $\{D_G\}$ o discriminador não é treinado



Implementação da GAN



O primeiro passo é treinar a rede **D** com os dados reais e os dados gerados pela rede **G**. O algoritmo de otimização é o *backpropagation*.

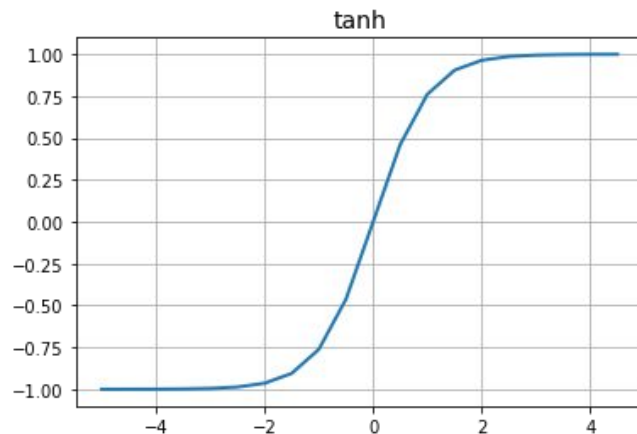


O segundo passo é treinar a rede **G** só com os dados de certa distribuição, mas só podem ser atualizados os pesos da rede **G**.

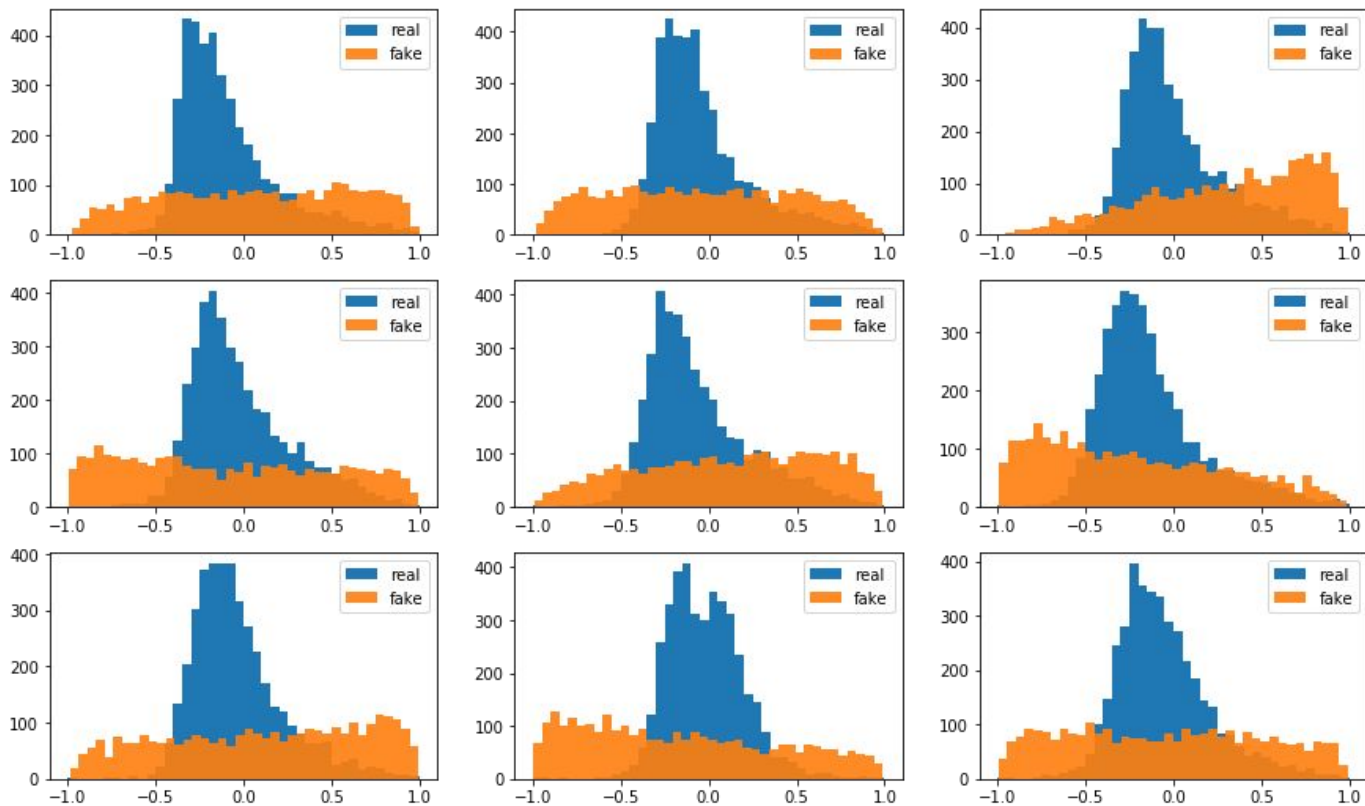
Processamento

$$x = \frac{x - \min(x)}{\max(x) - \min(x)} \bullet (\max - \min) + \min$$

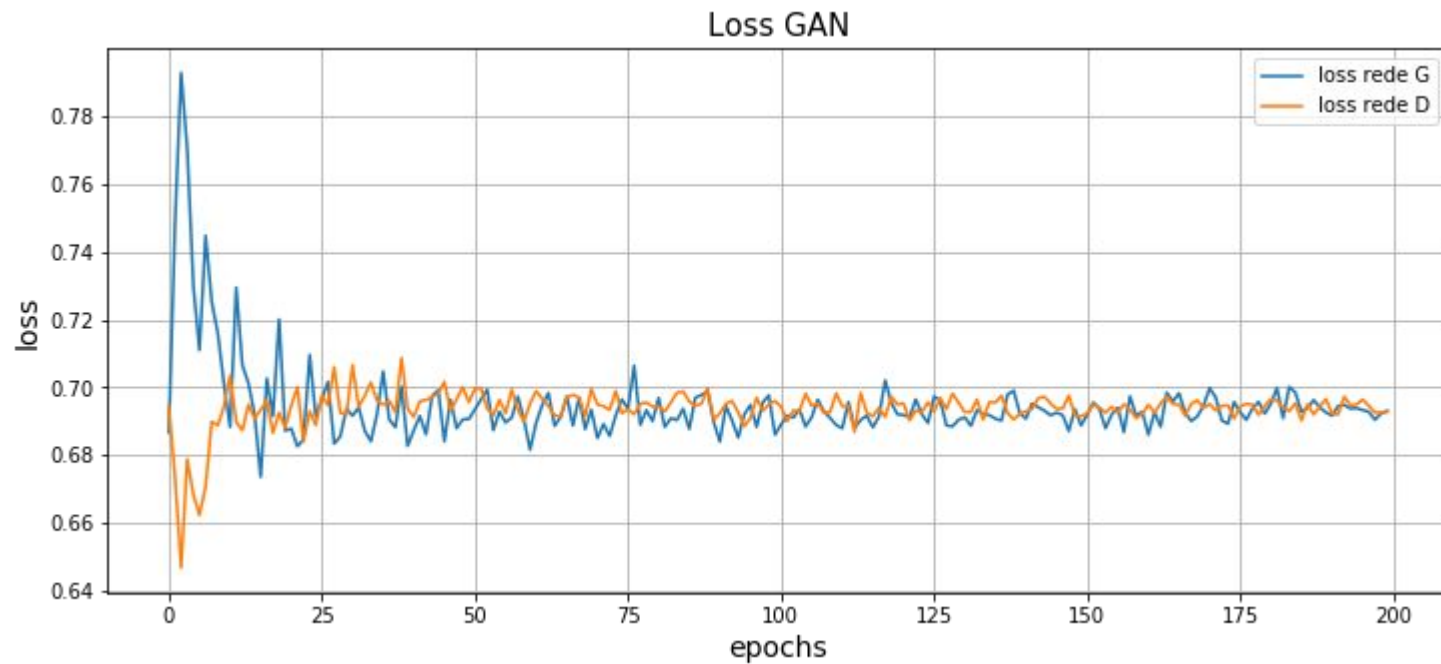
$$\min = -1 \ ; \ \max = 1$$



Geração de genes pela rede sim treinar

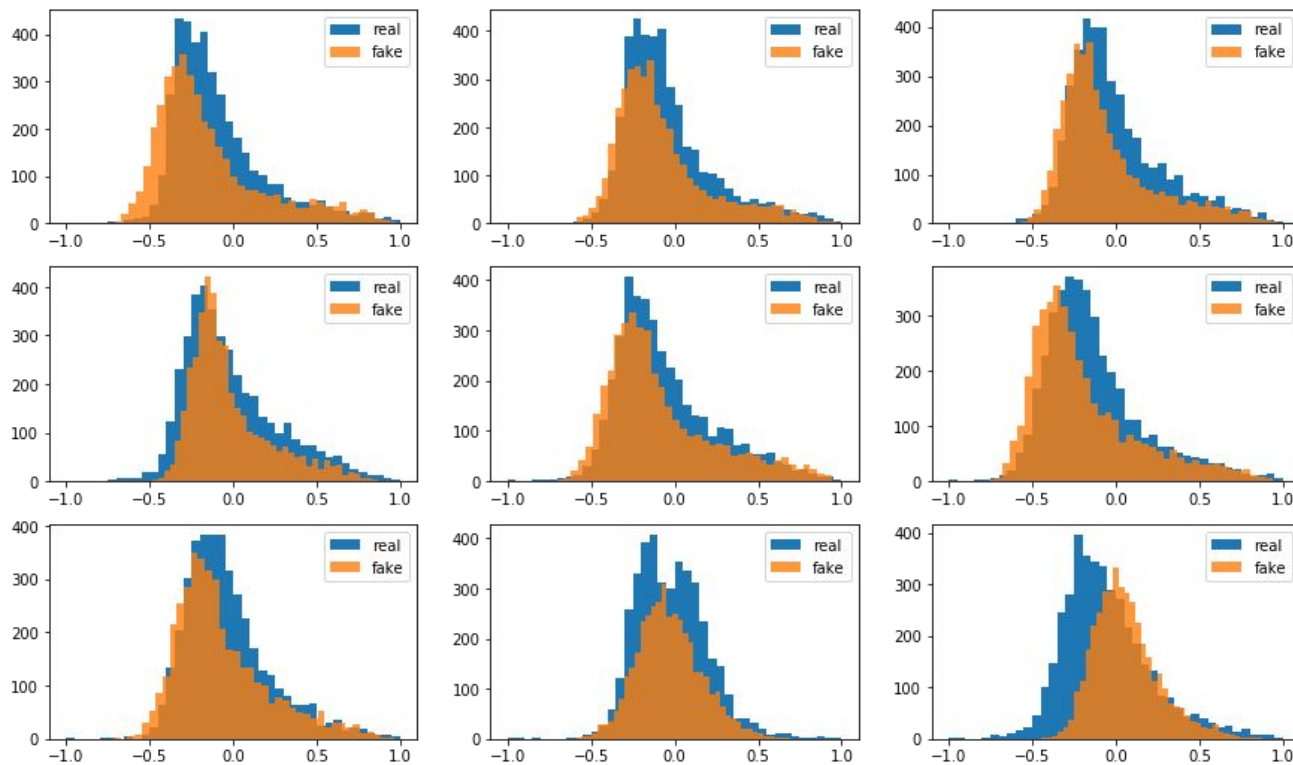


Erro da GAN



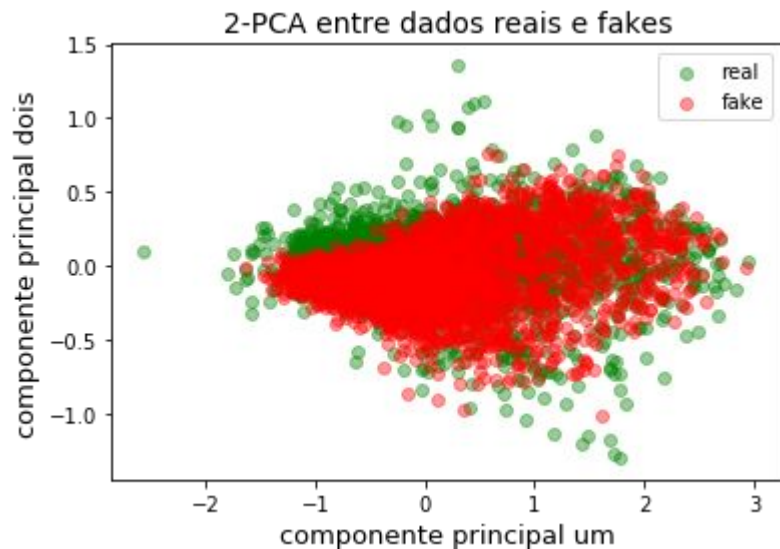
loss = 0.69

Geração de genes pela rede treinada

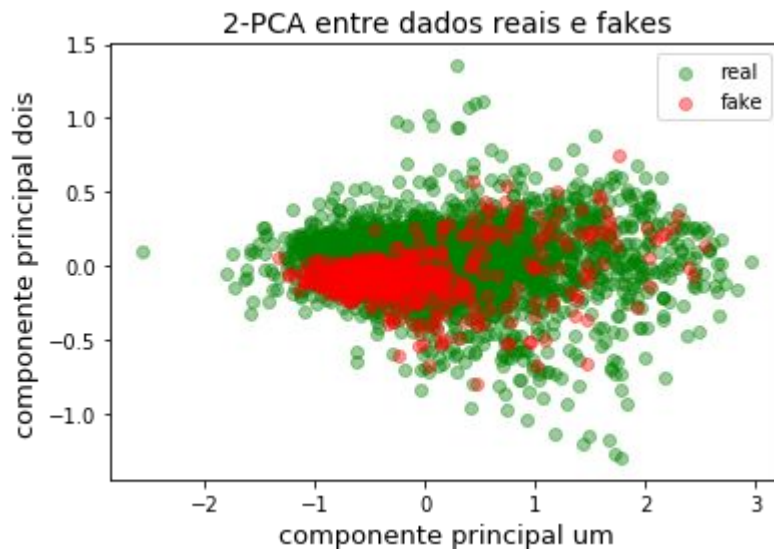


Geração de genes pela rede treinada

Os pontos vermelhos são os dados que a rede Geradora(**G**) é capaz de gerar depois de ser treinada.



3000 novos dados



500 novos dados

Medições

- Dada uma entrada *fake* do tamanho dos dados treinados o **Discriminador** acredita reais **62.65%** e o **37.35%** restante diz que não são reais.
- Dada a entrada de só dados reais no **Discriminador**, este diz que o **56.5%** dos dados são reais e o outro **43.5%** são dados *fakes*.