(5) Redes neurais auto-associativas e geradoras

Redes Neurais e Arquiteturas Profundas

Moacir Ponti
CeMEAI/ICMC, Universidade de São Paulo
MBA em Ciência de Dados

www.icmc.usp.br/~moacir — moacir@icmc.usp.br

São Carlos-SP/Brasil – 2020

Agenda

Autoencoders

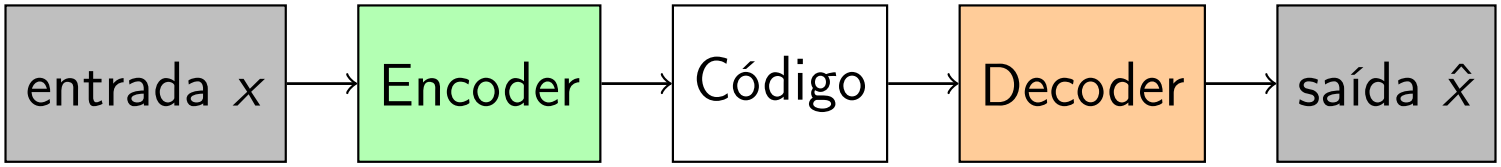
- Tipo "undercomplete"
- Tipo "overcomplete"
- Denoising autoencoders

Redes Geradoras

- Modelos geradores
- Autoencoders variacionais (VAEs)
- Redes adversárias geradoras (GANs)

Autoencoders

Redes neurais auto-associativas, ou auto-encoders, são métodos não supervisionados para aprendizado de representações.



Autoencoders: encoder e decoder

Encoder

Aprende um código, também chamado de representação latente ou feature embedding

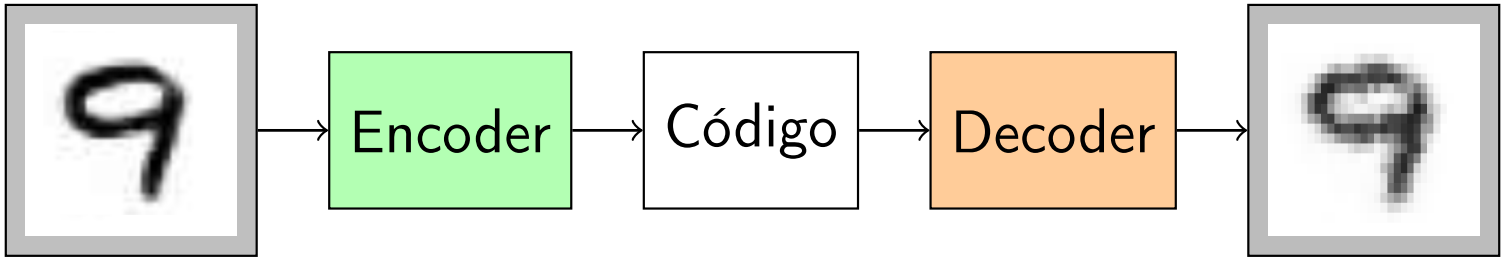
$$\mathrm{h} = s(\mathrm{W}\mathrm{x} + \mathrm{b}) = f(\mathrm{x})$$

Decoder

Aprende uma reconstrução da entrada

$$\hat{\mathrm{x}} = s(\mathrm{W}'\mathrm{h} + \mathrm{b}') = g(\mathrm{h})$$

Arquitetura de um autoencoder



Autoencoders: função de perda

A partir da saída $\hat{x} = g(f(x))$
minimizamos o erro/perda relativa à reconstrução da entrada

$$\mathcal{L}(x, g(f(x)) = \hat{x})$$

Mean squared error (perda quadrática) muito utilizada

$$\mathcal{L}(x, \hat{x}) = ||x - \hat{x}||^2$$

Autoencoders: tipos

Seja um código h $\in R^m$

Undercomplete

- m possui menos dimensões que x
- A camada do código é chamada de gargalo ou "bottleneck" por ser restrita

Overcomplete

- m possui dimensões maiores ou iguais às de x
- Há diferentes versões desse tipo para compensar a falta de restrição no código

Agenda

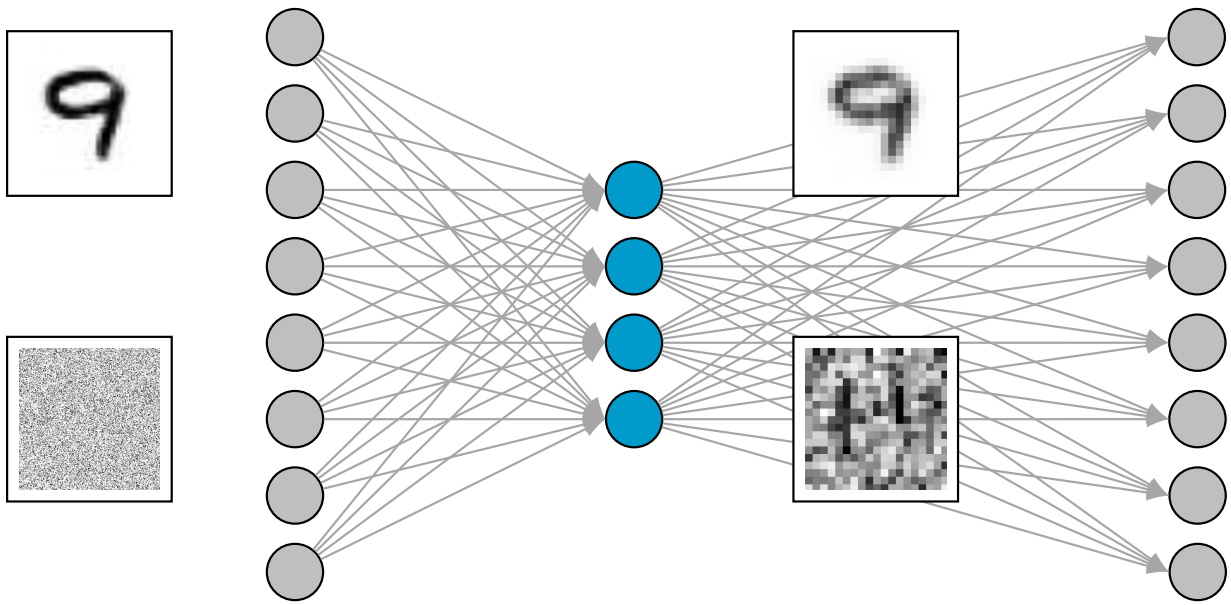
Autoencoders
- Tipo "undercomplete"
- Tipo "overcomplete"
- Denoising autoencoders

Redes Geradoras
- Modelos geradores
- Autoencoders variacionais (VAEs)
- Redes adversárias geradoras (GANs)

Undercomplete

Código é uma compressão com perdas da entrada

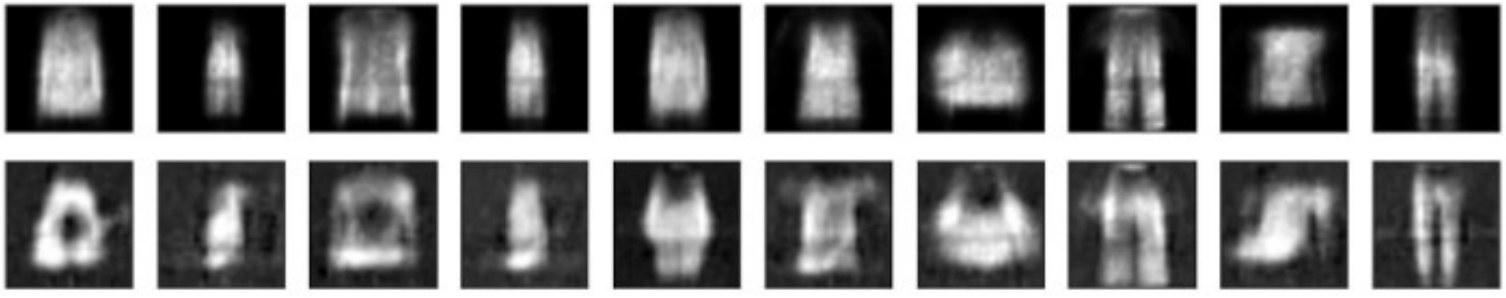
- camada do código é chamada de "bottleneck"
- o código produz boa representação para os dados de treinamento, em particular para reconstrução



Undercomplete

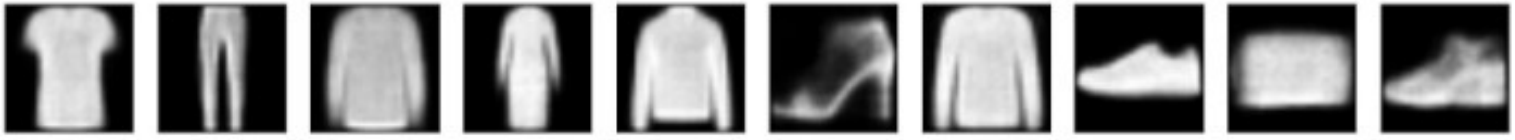
Auto-encoder treinado na MNIST

Auto-encoders treinados na Fashion



Undercomplete

Auto-encoder treinado na Fashion



Auto-encoders treinados na MNIST

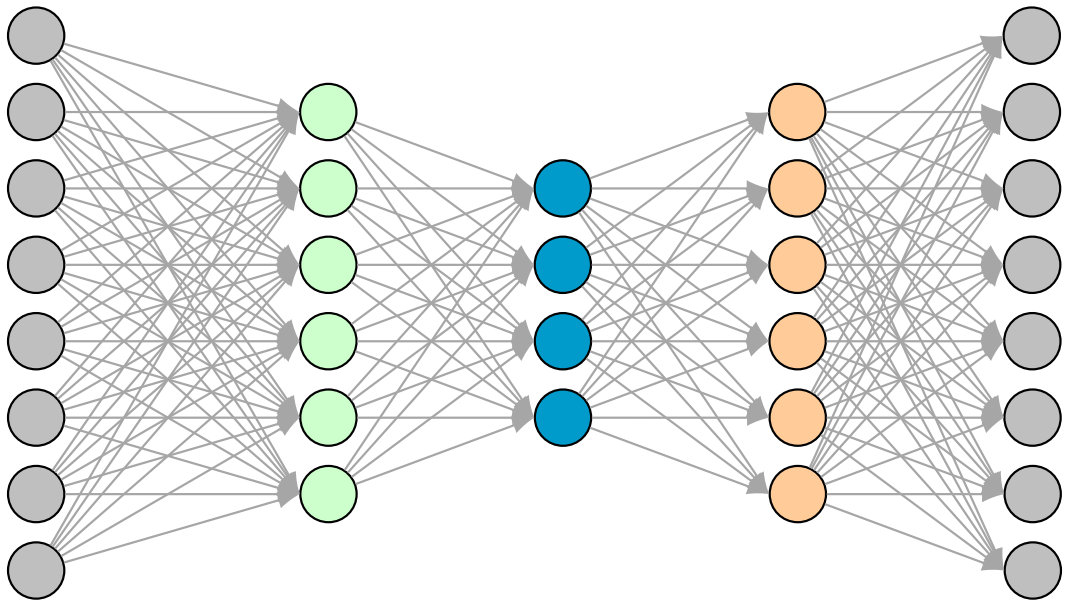
Undercomplete

- Pode ser usada para aprender uma **redução de dimensionalidade**
- Um autoencoder denso com uma única camada encoder/decoder tem relações com o método Principal Component Analysis (PCA)
- Se a variedade (manifold) dos dados é linear, o AE tende a convergir para uma projeção nos m principais componentes

Deep Undercomplete Autoencoders

Autoencoders profundos:

- Camadas não densas, ex: convolucionais, pooling, etc.
- Camada do código é comumente densa para permitir projeção dos dados



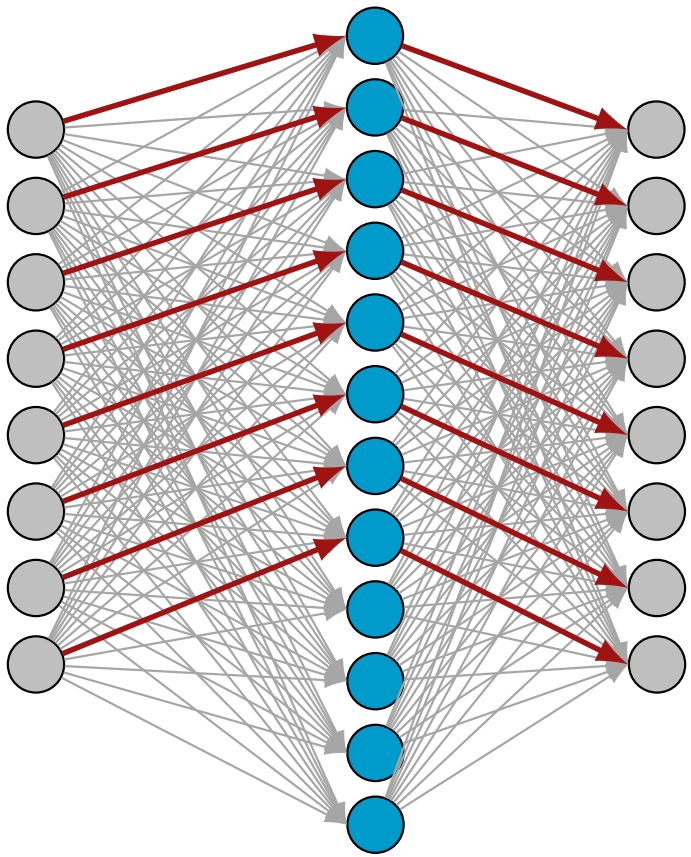
Modelos com alta capacidade podem mapear cada conceito de entrada em um único neurônio da camada do código

Agenda

Overcomplete

Camada intermediária com alta dimensionalidade

- uma implementação simples permitira a cópia simples (e perfeita) dos dados de forma que $x = \hat{x}$



Overcomplete regularized AEs

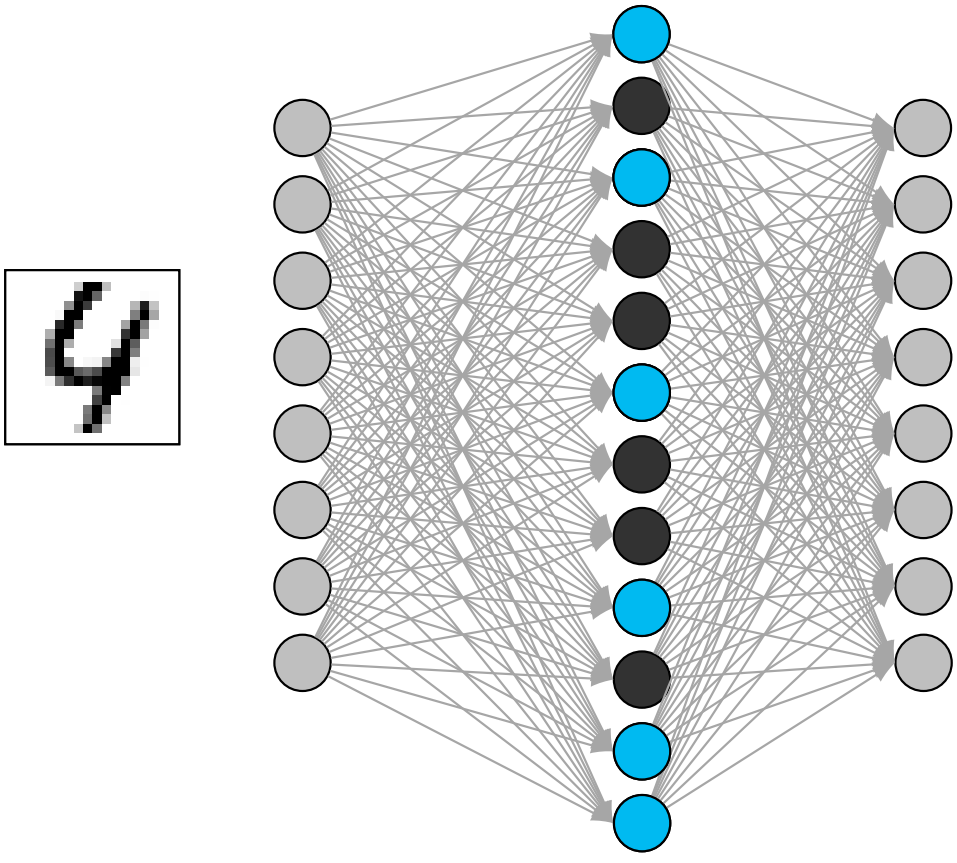
Uma maneira de impedir a cópia é **regularização** com alguma função $R(.)$, ex. regularização L1

$$\mathcal{L}(x, g(f(x))) + R(f(x))$$
$$\mathcal{L}(x, g(f(x))) + \lambda \sum_i |h_i|,$$

- a função de custo tenta manter um baixo número de ativações por entrada
- dropout também pode ser usado, nesse caso imediatamente antes da camada do código

Overcomplete regularized AEs

Regularização com restrição de esparsidade do código



Agenda

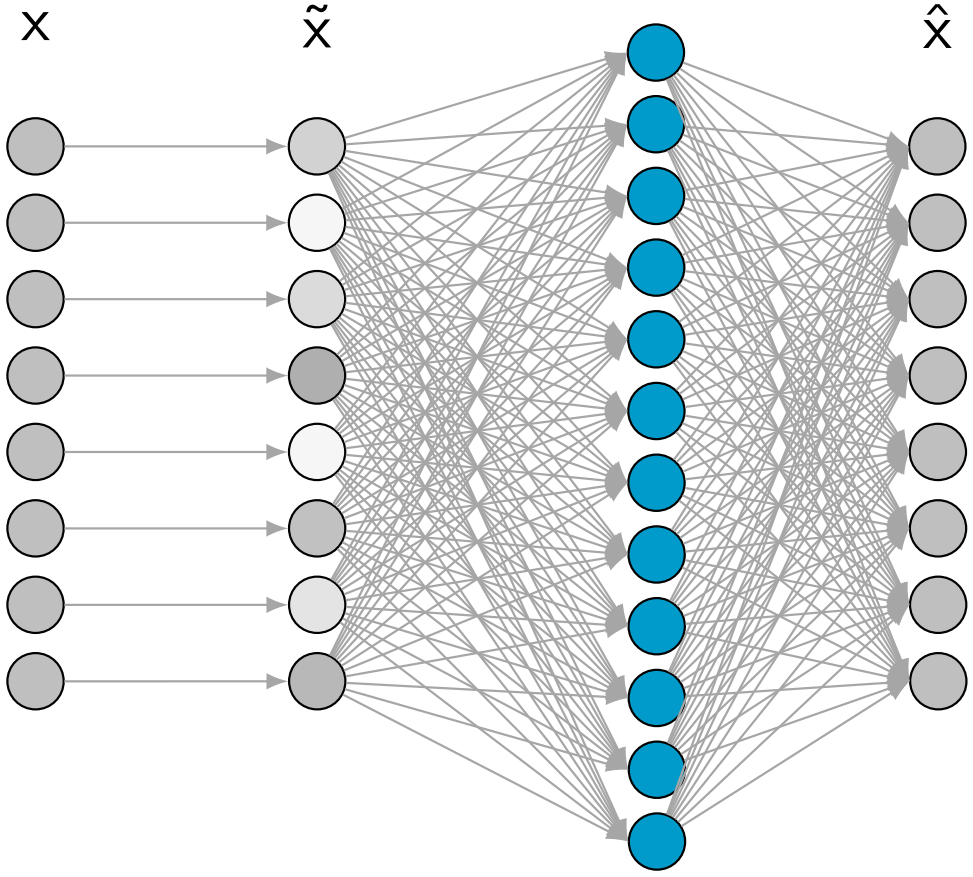
Denoising AEs (DAEs)

A regularização é atingida adicionando ruído à entrada, $\tilde{x} = \mathcal{N}(x)$

- a perda é computada usando a entrada não ruidosa x
- AE aprende a reconstruir x a partir de $\tilde{x}$
- o encoder deve aprender a remover o ruído, mantendo apenas as informações essenciais no código, permitindo que o decoder reconstrua a entrada

Denoising AEs (DAEs)

- DAEs aprendem uma representação robusta como efeito colateral de aprender a remover o ruído da entrada



Denoising AEs (DAEs)

Como adicionar ruído? Prática comum

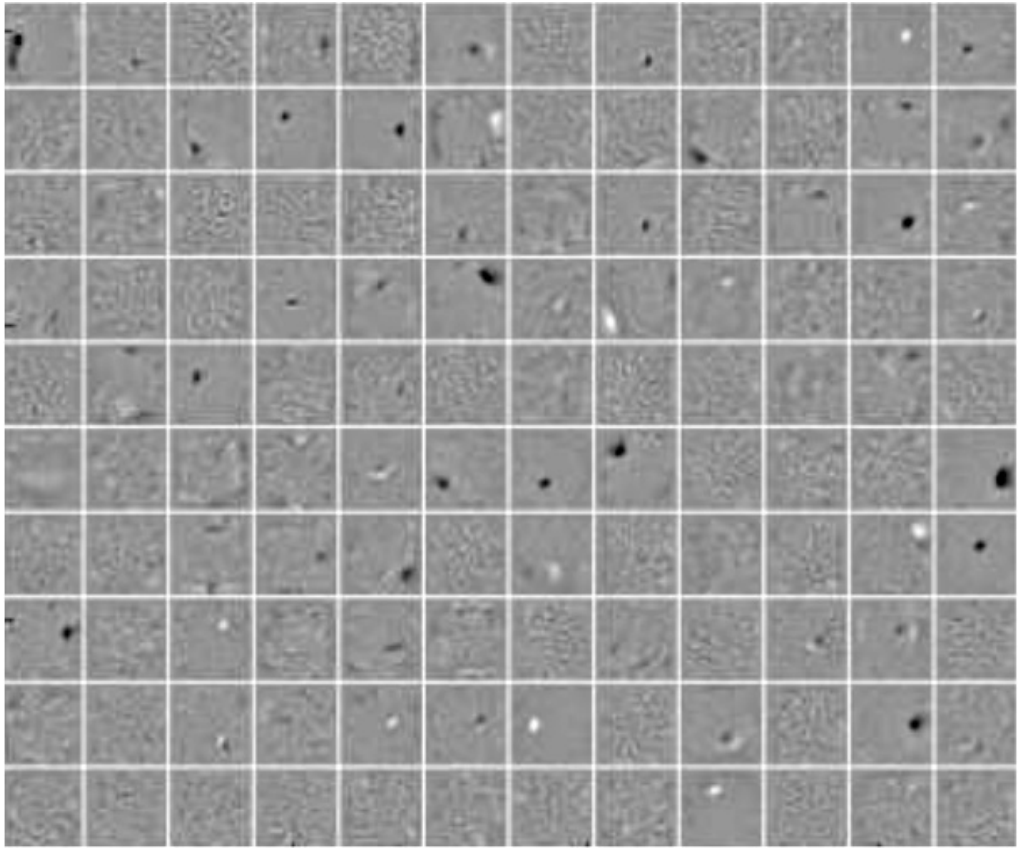
- Comumente ruído Gaussiano/Normal com $\mu = 0$, $\sigma \geq 1$;
- Ruído impulsivo: atribuir zero a uma porcentagem da entrada, com probabilidade p (dropout na entrada).

Intuição

- Aprende a projetar os dados ao longo de uma variedade/manifold relativo aos dados de entrada originais
- apenas entradas fora da distribuição original gerarão alto erro de reconstrução

Denoising AEs (DAEs): exemplo

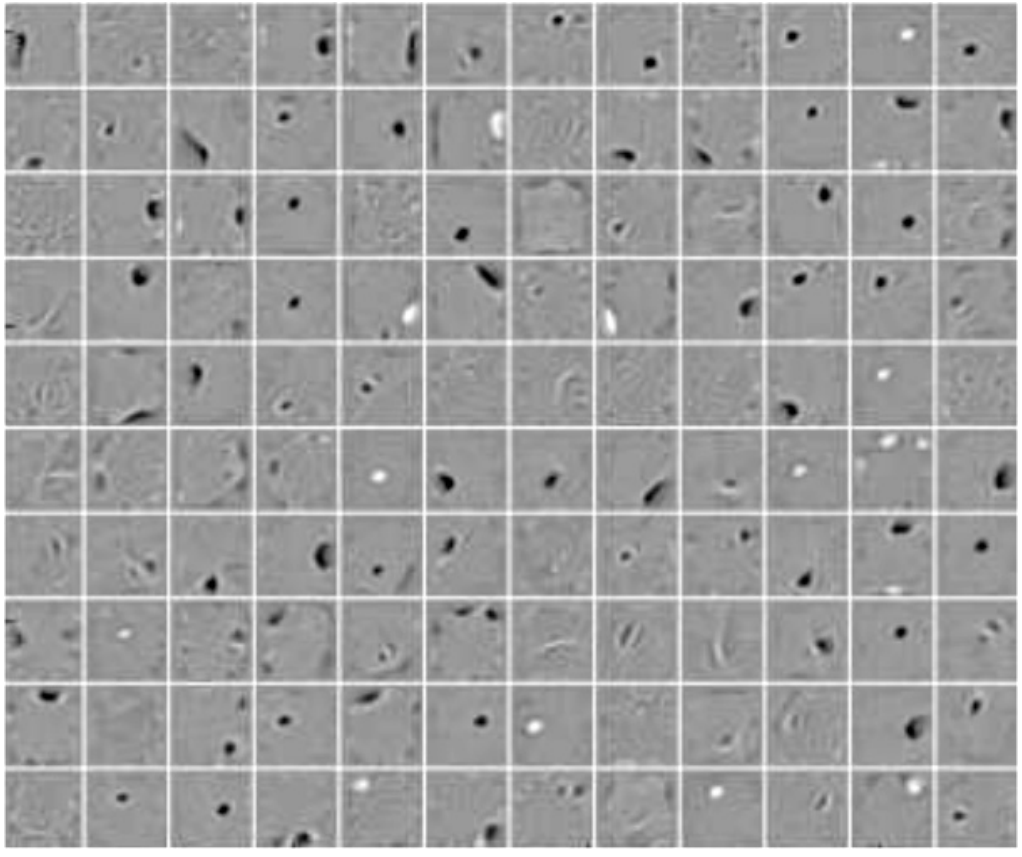
Padrões de ativação do dataset MNIST com Autoencoder convencional



Vincent, Pascal, et al. "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion." Journal of Machine Learning Research, 2010: 3371-3408.

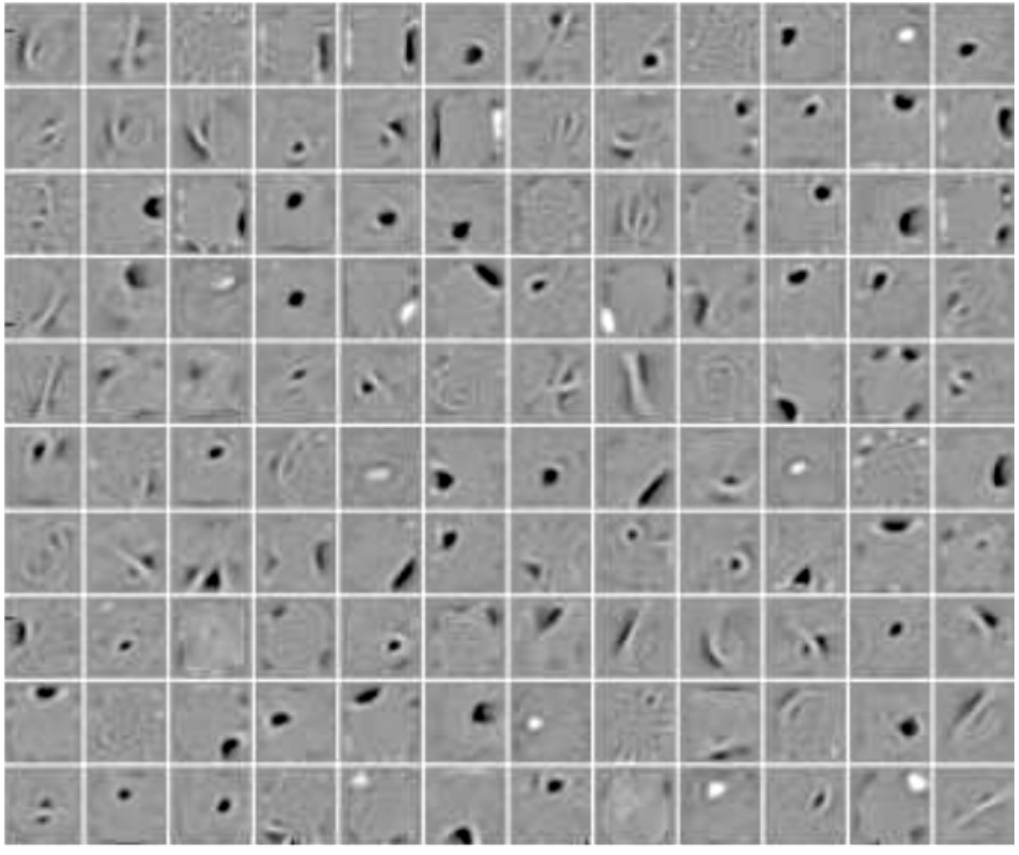
Denoising AEs (DAEs): exemplo

Padrões de ativação do dataset MNIST, zerando entradas com taxa 25%



Denoising AEs (DAEs): exemplo

Padrões de ativação do dataset MNIST, zerando entradas com taxa 50%



Técnicas de regularização

- Podem ser usadas também nos AEs undercomplete

Considerações

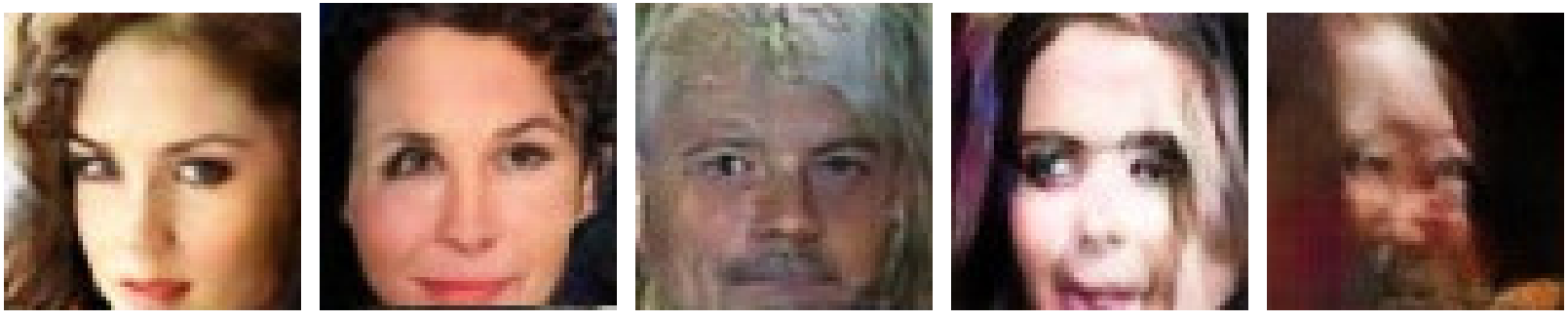
- AEs podem ser boa escolha com dados **não supervisionados** para aprendizado de manifolds e agrupamento;
- Representam uma nova tarefa: **reconstrução** que pode ser acoplada a outras arquiteturas

Agenda

Autoencoders
- Tipo "undercomplete"
- Tipo "overcomplete"
- Denoising autoencoders

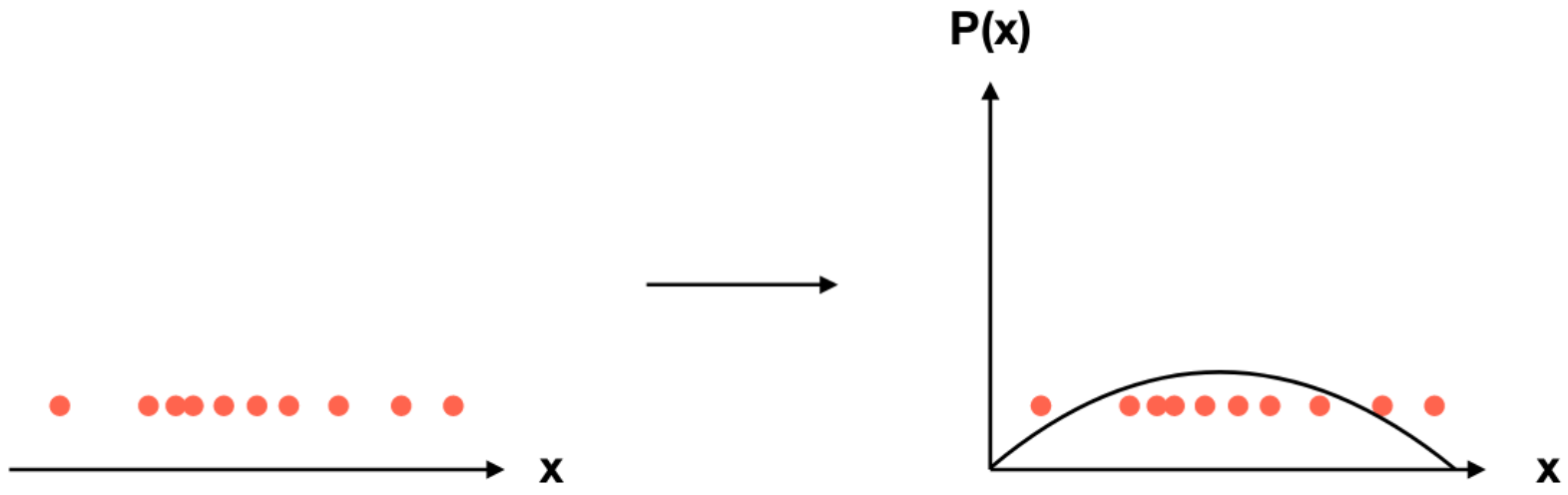
Redes Geradoras
- Modelos geradores
- Autoencoders variacionais (VAEs)
- Redes adversárias geradoras (GANs)

Geração de dados



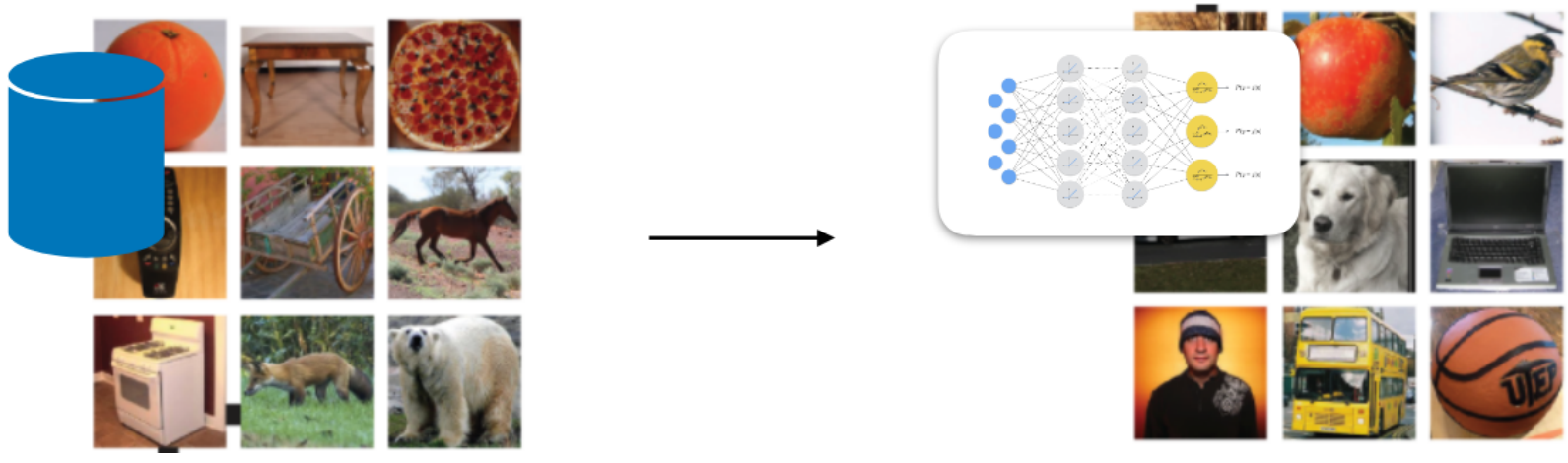
Interpretação: aprender a distribuição

Aprender a distribuição que "gera" os dados permite amostrar a partir dela



Interpretação: aprender a gerar dados

Mas podemos querer apenas dados, não uma distribuição...



Tipos de métodos

Funções densidade explícitas

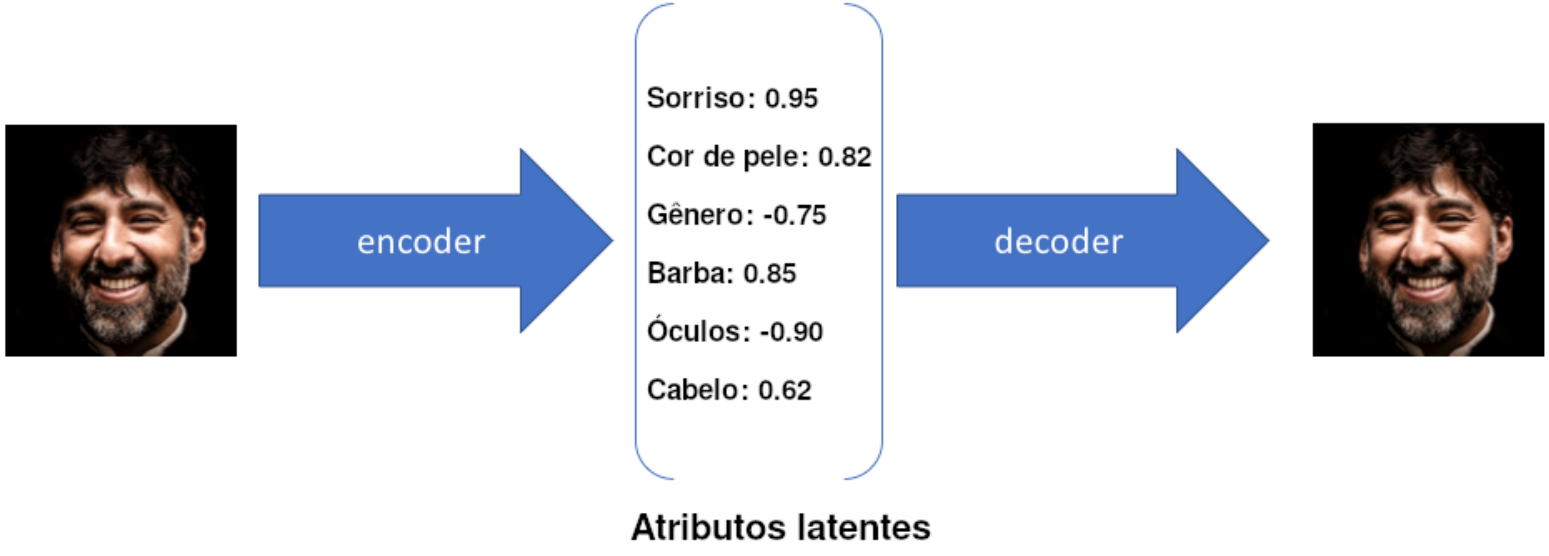
- Fully Visible Belief Nets
- Boltzmann Machines
- Variational Autoencoders

Funções densidade implícitas

- Métodos de Monte Carlo
- Likelihood-free inference via classification
- Generative Adversarial Networks

Autoencoders

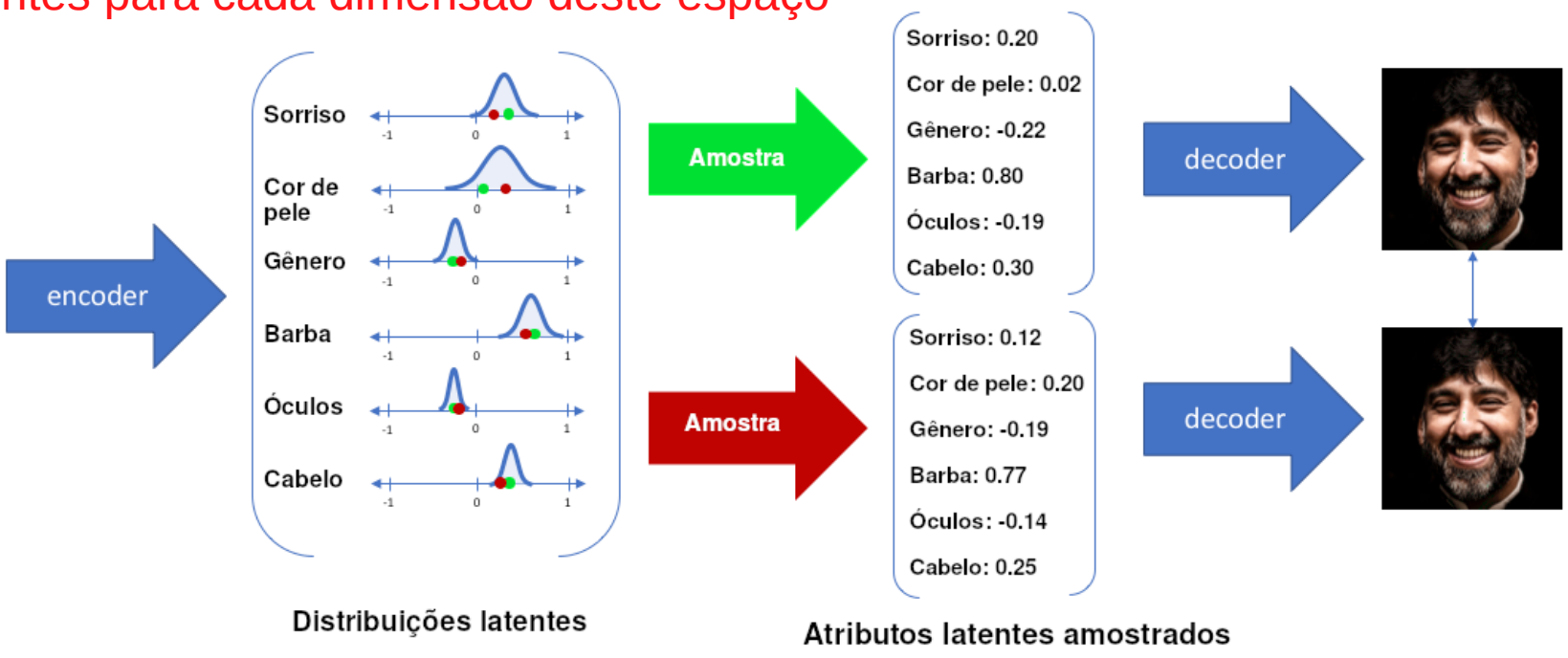
Autoencoders convencionais tentam codificar atributos de forma discreta



Autoencoders variacionais (VAEs)

Autoencoders variacionais aprendem distribuições (seus parâmetros) de cada variável, a partir do qual se amostram valores

o espaço latente não é composto por variáveis latentes, mas sim distribuições latentes para cada dimensão deste espaço



Redes adversárias geradoras (GANs)

Não se assume uma distribuição explícita

- **Adversária** pois há dois componentes que "disputam"
- **Geradora** pois o objetivo central é aprender a gerar dados

Redes adversárias geradoras (GANs)

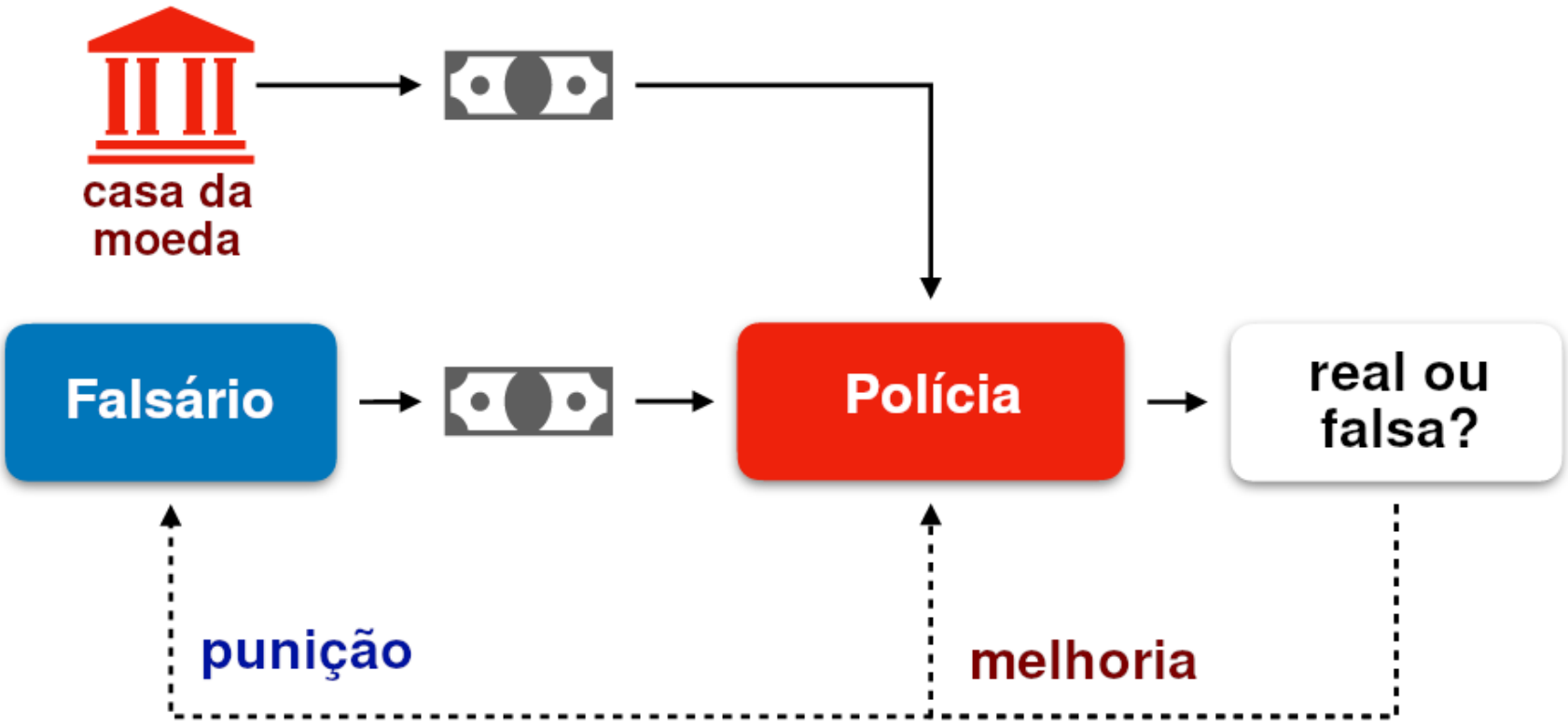
Gerador $\mathcal{G}$

- recebe um exemplo z' obtido de uma distribuição, i.e. $z' \sim q(z)$
- gera x por meio de uma função $x = \mathcal{G}_\Theta(z')$

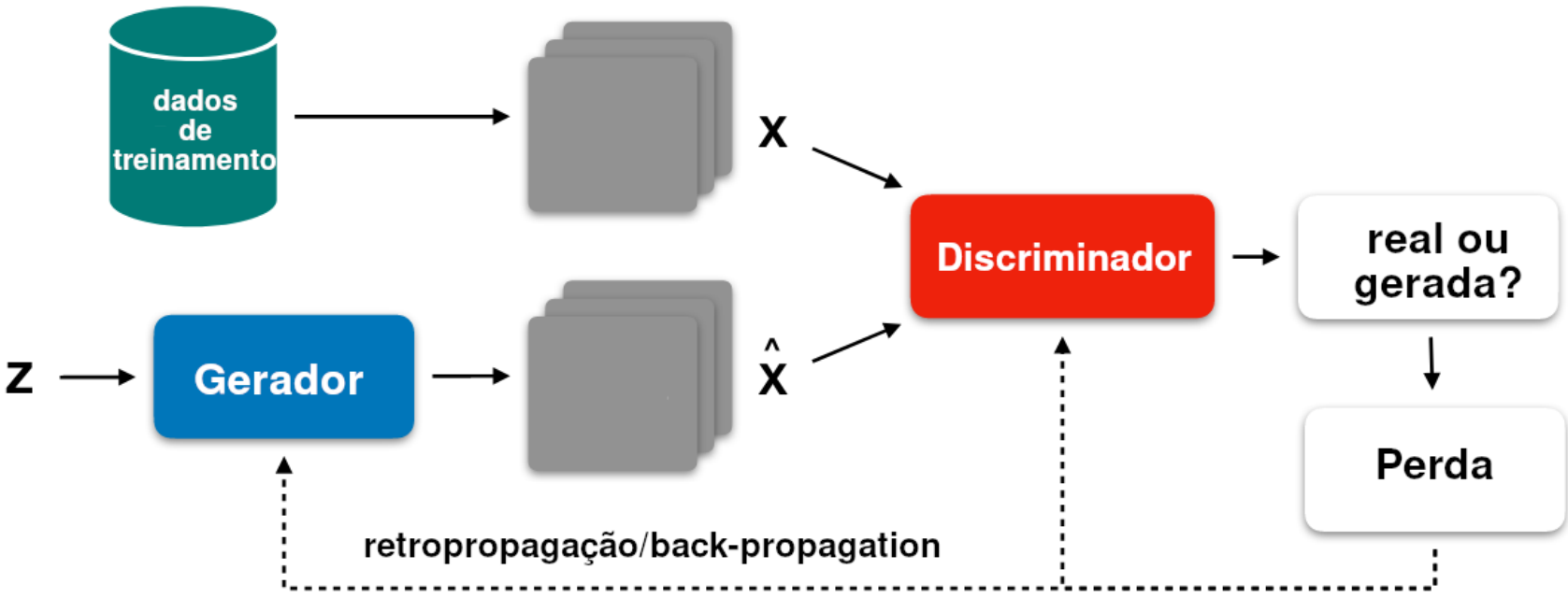
Discriminador $\mathcal{D}$

- recebe x e classifica se esse foi produzido pela distribuição original ou pela aproximação do gerador

Redes adversárias geradoras (GANs)



Redes adversárias geradoras (GANs)



Redes adversárias geradoras (GANs)

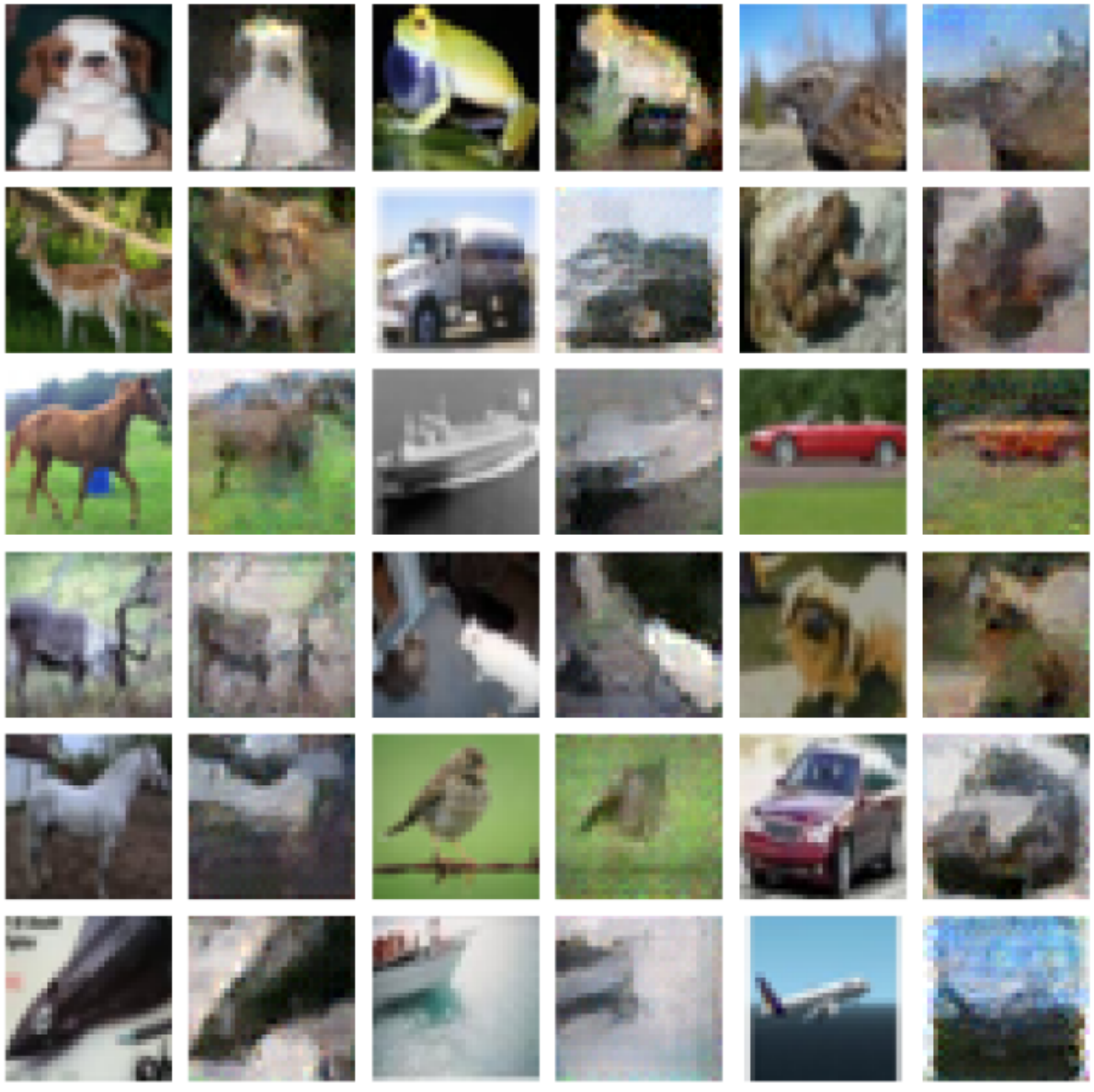
Formulação

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

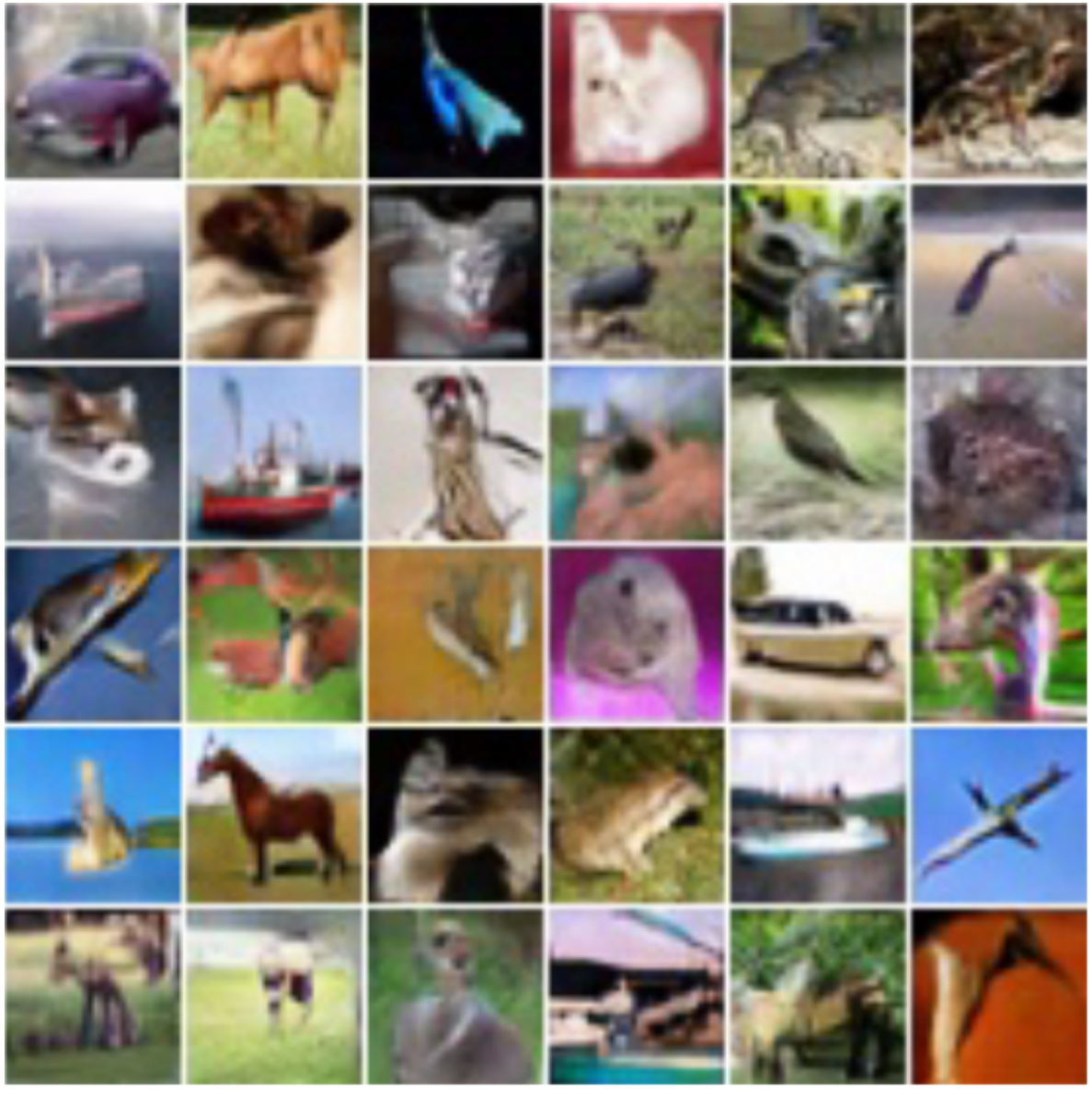
Função original

$$-(1/2)E_{x \sim p_{data}(x)}[\log D(x)] - (1/2)E_z[\log 1 - D(G(z))]$$

Exemplos gerados por uma GAN de 2017



Exemplos gerados por uma GAN de 2020



Redes adversárias geradoras (GANs)