

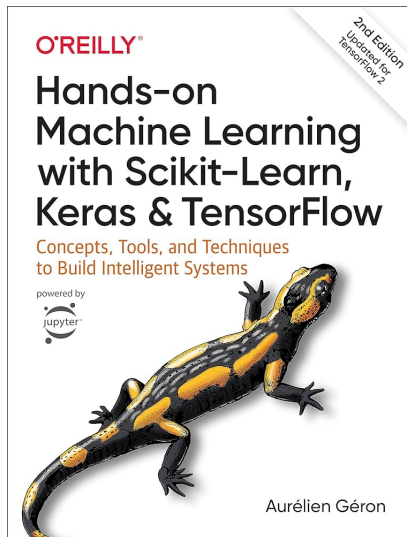
Autoencoders

Eduardo de Medeiros da Silveira

Universidade Federal de Santa Maria

Capítulo Escolhido

- ▶ *Autoencoders, GANs, and Diffusion Models.*



Representação Eficiente de Dados

Em 1970, William Chase and Herbert Simon fizeram um experimento com jogadores profissionais de xadrez, para estudar a relação entre memória, percepção e reconhecimento de padrões.

- ▶ Capazes de memorizar o tabuleiro em poucos segundos.
- ▶ Somente quando as peças estavam em posições naturais.
- ▶ O reconhecimento de padrões ajuda na memorização.

Representação Eficiente de Dados



Figura: Etapas do experimento da memória no xadrez.

Manifold Hypothesis

- Conjuntos de dados com várias dimensões existem em *manifolds* de menor dimensão.

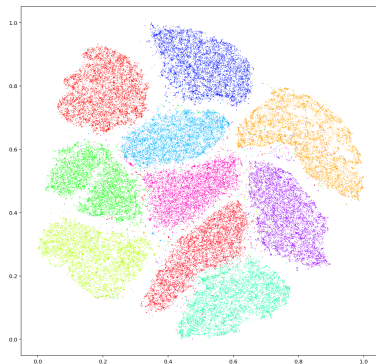


Figura: Aplicação da técnica T-SNE ao MNIST.

Autoencoder

Um *autoencoder* é uma rede neural que **tenta** aprender a função identidade.

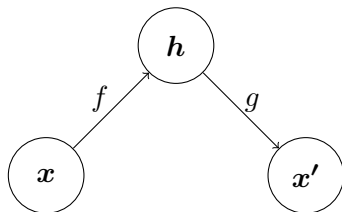


Figura: Esquema geral de um *autoencoder*, que mapeia uma entrada x para uma saída x' , através de uma representação interna h . O *autoencoder* é composto por um codificador f e um decodificador g .

Autoencoder

Algumas características:

- ▶ Aprendizado não-supervisionado ou auto-supervisionado.
- ▶ A saída não importa.
- ▶ A representação latente \mathbf{h} importa.
- ▶ Restrições.

Autoencoder

Algumas características:

- ▶ Mesmo número de neurônios na entrada e na saída.
- ▶ Geralmente é simétrico.
- ▶ *Stacked* ou *deep autoencoders*.
- ▶ Nem sempre \mathbf{h} vai capturar informações importantes.

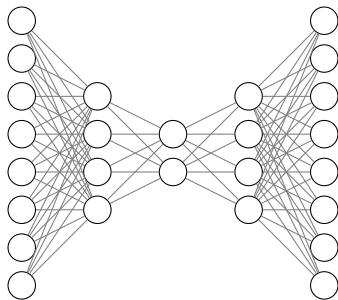


Figura: Exemplo de um autoencoder.

Undercomplete Autoencoder

- ▶ A dimensão de \mathbf{h} é menor do que a dimensão de \mathbf{x} .
- ▶ Minimiza-se L , que calcula a dissimilaridade de \mathbf{x} e \mathbf{x}' .

$$L(\mathbf{x}, \mathbf{x}') = L(\mathbf{x}, g(f(\mathbf{x})))$$

Convolutional Autoencoders

- ▶ Camadas convolucionais em vez de camadas densas.
- ▶ Principalmente para imagens.
- ▶ O *encoder* faz a compressão da imagem, enquanto o *decoder* faz a decompressão (camadas transpostas).

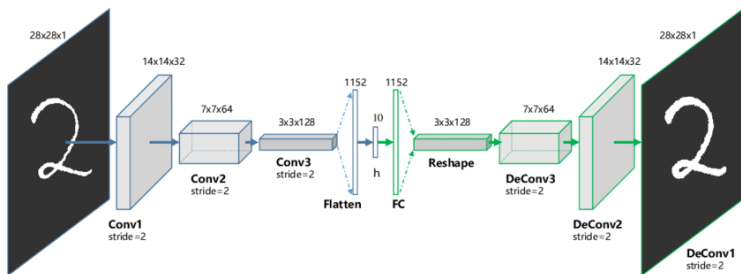


Figura: Exemplo de um autoencoder convolucional.

Overcomplete Autoencoders

- A representação latente \mathbf{h} tem dimensão maior ou igual à da entrada.

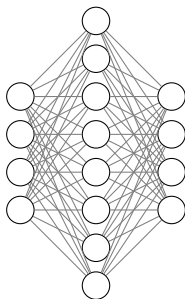


Figura: Exemplo de um autoencoder overcomplete.

Denoising Autoencoders

- ▶ Adição de ruído na entrada (Gaussiano ou Dropout).
- ▶ Treinamento para recuperar os dados originais.

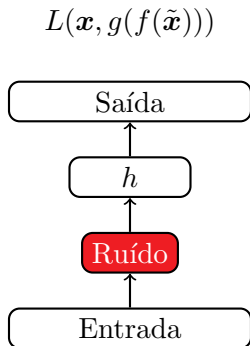


Figura: Esquema de um *denoising autoencoder*. O ruído pode representar a adição de ruído Gaussiano aos dados ou uma camada de Dropout.

Sparse Autoencoders

- ▶ A representação latente é penalizada por sua esparsidade.
- ▶ Escolhemos uma quantidade de neurônios que queremos ativados.
- ▶ Cada neurônio acaba representando uma característica específica.

$$L(\mathbf{x}, \mathbf{x}') + \Omega(\mathbf{h}) = L(\mathbf{x}, g(f(\mathbf{x}))) + \Omega(\mathbf{h})$$

Pré-treino não-supervisionado

- ▶ Queremos treinar um modelo supervisionado.
- ▶ Temos poucas observações rotuladas.
- ▶ Podemos treinar um *autoencoder* e reutilizar o *encoder*.
- ▶ Congelamento dos parâmetros.

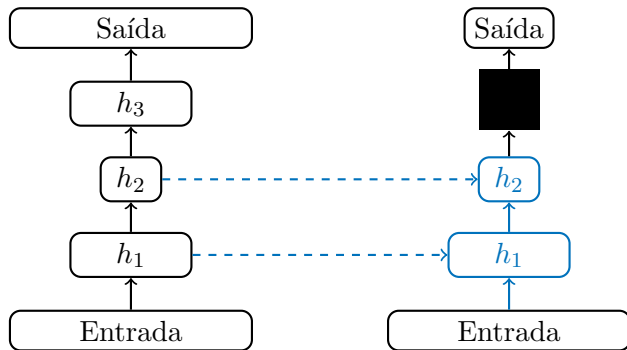


Figura: Aproveitamento da função de *encoding* em outra rede neural.

Pré-treino não-supervisionado

- ▶ Aumenta a generalização do modelo (remove ruído).
- ▶ Acelera o treinamento.
- ▶ Pode ocorrer perda de informação importante.

Enlace de parâmetros

- ▶ Se o *autoencoder* for simétrico, podemos criar um enlace entre os parâmetros do *encoder* e *decoder*.
- ▶ Reduzimos pela metade o número de parâmetros.
- ▶ O vetor de viés é mantido.

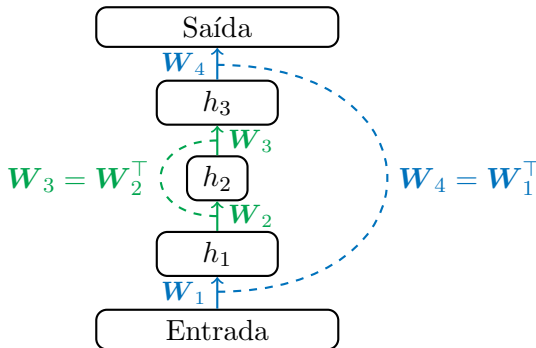


Figura: Enlace dos parâmetros de um *encoder* e um *decoder*.

Treinamento por Camadas

Em vez de treinar todo o *autoencoder*, podemos dividir o treino por camadas.

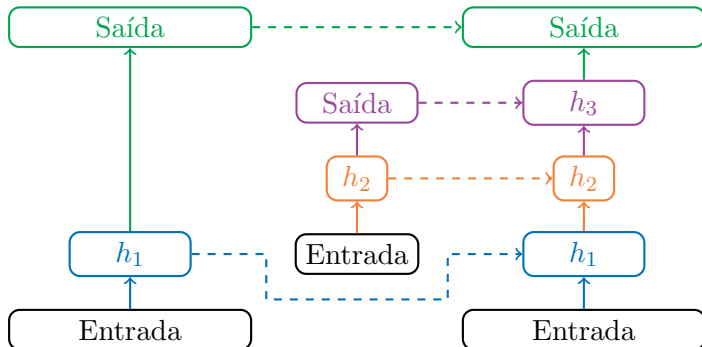


Figura: Treinamento por camadas de um *autoencoder*. Primeiramente, treinamos as camadas h_1 e Saída. Depois, treinamos h_2 e h_3 .

Detecção de Anomalias

Podemos usar *autoencoders* para detectar anomalias.

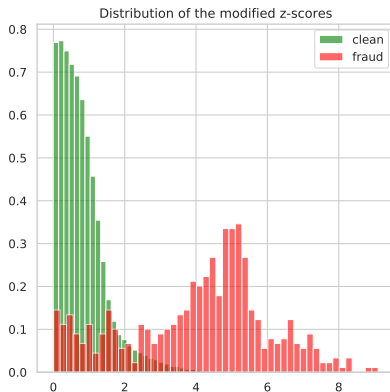


Figura: Distribuição do erro de reconstrução de dados normais e anômalos.

Remoção de Ruído

- Podemos usar um *Denoising Autoencoder* para remover ruído de imagens.

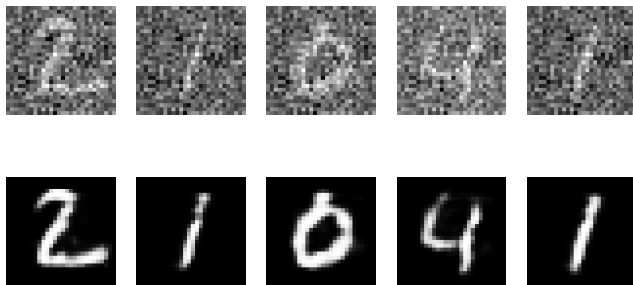


Figura: Remoção de ruído de dígitos do MNIST.

Colorização de Imagens

- Também podemos colorizar imagens com *autoencoders*.



Figura: Colorização de um imagem com um *autoencoder*

Variational Autoencoders

- Modelo generativo e probabilístico.

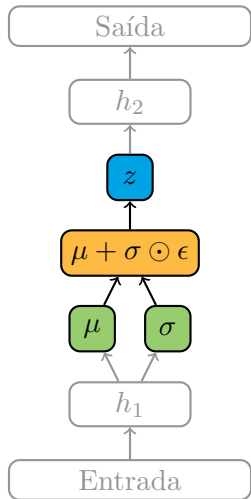


Figura: Esquema de um *variational autoencoder*, onde $\epsilon \sim \mathcal{N}(0, 1)$.

Variational Autoencoders - Sampling

- ▶ Kullback-Leibler divergence.