Autoencoders: da motivação às variantes modernas

CPE 727 - Aprendizado de Profundo

Felipe Fink Grael, Rafael Tadeu Cardoso dos Santos, Thalles Nonato Leal Santos e Jefferson Osowsky

24 de novembro de 2025

Table of Contents

Introdução

1 Motivação

- **Definição:** Algoritmos cujo propósito principal é copiar sua entrada na saída [1]. São tipicamente construídos como redes neurais artificiais treinadas de forma não supervisionada.
- **Arrquitetura Básica:**
  - Encoder: transforma entrada em representação latente
  - Representação: espaço latente de menor, maior ou igual dimensão
  - Decoder: reconstrói a entrada original
- **Objetivo:** Aprender a função identidade $f(x) \approx x$ através de um espaço latente comprimido

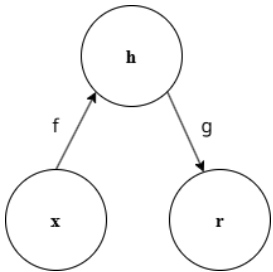

Figura: Estrutura básica de Autoencoders.

Componentes Fundamentais

1 Motivação

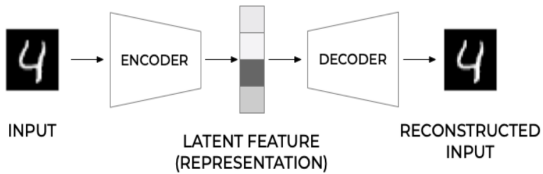

Figura: Estrutura de um autoencoder com representação latente[1].

- **Encoder:** $f_\theta : \mathcal{X} \to \mathcal{H}$ onde $h = f_\theta(x)$
- **Decoder:** $g_\phi : \mathcal{H} \to \mathcal{X}$ onde $x' = g_\phi(h)$
- **Reconstrução:** $x' = g_\phi(f_\theta(x))$
- **Espaço latente $\mathcal{H}$:** Representação comprimida dos dados ($\dim(\mathcal{H}) < \dim(\mathcal{X})$)

[1] Figura de Umberto Michelucci [2]

Por que usar Autoencoders?

1 Motivação

Aprendizado de Representações:

- Extrair características relevantes automaticamente dos dados
- Redução de dimensionalidade não-linear (superior ao PCA para dados complexos)
- Aprendizado não supervisionado - não requer labels

Vantagens sobre métodos tradicionais:

- PCA: apenas transformações lineares
- Autoencoders: capturam relações não-lineares complexas
- Profundidade permite representações hierárquicas [3]

Table of Contents

2 Formulação Matemática

Definição Formal

2 Formulação Matemática

Seja μ_{ref} uma distribuição de probabilidade de referência em $\mathcal{X}$ e $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ uma função de distância.

Função de custo do autoencoder:

$$L(\theta, \phi) = \mathbb{E}_{x \sim \mu_{ref}} \left[d\left(x, g_\phi(f_\theta(x))\right) \right]$$

Objetivo de treinamento:

$$(\theta^*, \phi^*) = \arg\min_{\theta, \phi} L(\theta, \phi)$$

Exemplo: Autoencoder Linear de Uma Camada

2 Formulação Matemática

Encoder:

$$h = f_{W,b}(x) = \sigma(Wx + b)$$

Decoder:

$$x' = g_{W',b'}(h) = \sigma(W'h + b')$$

Função Custo:

$$L(W, b, W', b') = \frac{1}{N} \sum_{i=1}^{N} \|x_i - g_{W',b'}(f_{W,b}(x_i))\|_2^2$$

Parâmetros a otimizar: $\theta = W, b$, $\phi = W', b'$

Exemplo: Undercomplete Autoencoder Linear (Caso Especial)

2 Formulação Matemática

Autoencoder linear: sem função de ativação $\sigma(z) = z$

Teorema: O autoencoder linear ótimo projeta os dados no subespaço gerado pelos primeiros k autovetores da matriz de covariância Σ_{XX} [4].

Erro mínimo:

$$\Sigma(A, B) = Tr(\Sigma) - \sum_{i=1}^{k} \lambda_i = \sum_{i=k+1}^{n} \lambda_i$$

Onde $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$ são os autovalores de Σ_{XX}.
Conexão com PCA: Autoencoders lineares aprendem o mesmo subespaço que a Análise de Componentes Principais.

Table of Contents

Undercomplete Autoenconder

3 Undercomplete Autoencoders com redes neurais

- Autoencoders são treinados para copiar a entrada para a saída, mas o objetivo real é aprender representações úteis dos dados.
- Para isso, impõe-se que a dimensão do *espaço latente* satisfaça $\dim(\boldsymbol{h}) < \dim(\boldsymbol{x})$, formando um *autoencoder undercomplete*, o que força o modelo a capturar as características mais relevantes da distribuição dos dados.
- O treinamento consiste em minimizar a função de perda:

$$L(\boldsymbol{x}, g(f(\boldsymbol{x}))),$$

onde L mede quão diferente $g(f(\boldsymbol{x}))$ está de $\boldsymbol{x}$ (por exemplo, usando MSE).

Undercomplete Autoenconders: Correspondência com PCA/SVD

PCA e Autoencoders: Quando o decoder é linear e L é o erro quadrático médio (MSE), um autoencoder undercomplete aprende a abranger o mesmo subespaço que o PCA.

Isso ocorre porque, sob linearidade e codificação com dimensão reduzida, o autoencoder busca a melhor reconstrução no sentido dos mínimos quadrados, exatamente como o PCA.

Formulação do Problema

3 Undercomplete Autoencoders com redes neurais

O objetivo de um autoencoder linear é resolver:

$$\min_{D,E} \frac{1}{2}\|X - D(E(X))\|_F^2.$$

Se escolhemos D e E como matrizes (mapeamentos lineares), o problema é:

$$\min_{D,E} \frac{1}{2}\|X - DEX\|_F^2.$$

Introduzimos uma variável auxiliar:

- $A = DE$
- $\min_{A,D,E} \frac{1}{2}\|X - AX\|_F^2$ sujeito a $A = DE$.
- A restrição $A = DE$ é o que torna o problema não convexo.

Relaxação: Ignorando a Restrição

Ignorando por um momento a restrição $A = DE$:

- O problema vira mínimos quadrados clássico.
- Uma solução analítica está disponível.

A solução é:

$$A^\star = (XX^\top)^{-1}(XX^\top).$$

Se $(XX^\top)$ é inversível:

$$A^\star = I,$$

ou seja, o mapeamento ótimo (sem a restrição) seria a identidade.

Usando a SVD de $XX^\top$

Como $XX^\top$ é quadrada, sua decomposição SVD é:

$$XX^\top = USU^\top,$$

onde U é ortogonal e S contém os autovalores.
Substituindo na solução:

$$A^\star = (USU^\top)^{-1}(USU^\top)$$

$$= (U^\top)^{-1}S^{-1}U^{-1}\ USU^\top$$

$$= UU^\top.$$

Solução Analítica do Autoencoder Linear

Reintroduzindo a restrição $A = DE$ obtemos:

$$A^\star = D^\star E^\star = UU^\top.$$

Portanto:

$$D^\star = U, \qquad E^\star = U^\top.$$

- Se X tem posto n, somente as primeiras n colunas de U são necessárias.
- Assim, o autoencoder linear aprende o mesmo subespaço do PCA.

Interpretação: Relação Explícita com PCA

3 Undercomplete Autoencoders com redes neurais

A matriz U contém os **autovetores** dos dados — exatamente as **direções principais** do PCA.

Portanto, a solução ótima do autoencoder linear é:

$$E^\star = U^\top \quad \text{(projeção para o espaço PCA)}$$

$$D^\star = U \quad \text{(reconstrução a partir das componentes principais)}$$

- O **encoder** recupera as coordenadas PCA.
- O **decoder** reconstrói os dados a partir dessas coordenadas.

Table of Contents

4 Considerações sobre Arquitetura

Autoencoders neurais: Número de camadas

4 Considerações sobre Arquitetura

Teorema do Aproximador Universal: Redes neurais com uma única camada oculta podem aproximar qualquer função contínua.

Autoencoders profundos: Na prática, encoder e decoders possuem pelo menos uma camada oculta cada

- Maior capacidade de modelagem
- Podem aprender representações hierárquicas
- Podem ser treinados camada a camada (greedy layer-wise pretraining)

Dimensão do Espaço Latente

4 Considerações sobre Arquitetura

Subcompletos (undercomplete): Espaço latente tem dimensão menor que a entrada ($\dim(\mathcal{H}) < \dim(\mathcal{X})$)

- Forçam compactação dos dados (com perdas)
- Encoders e decoders não podem ser bons demais

Sobrecompletos (overcomplete): Espaço latente pode ter dimensão maior que a entrada ($\dim(\mathcal{H}) > \dim(\mathcal{X})$)

- Risco de aprender a função identidade
- Usam **regularização** para conferir características desejáveis

Table of Contents

5 Autoencoders com regularização

Autoencoders com regularização

5 Autoencoders com regularização

Técnicas que adicionam **termos de regularização** ou modificam o processo de treinamento para melhorar a qualidade das representações aprendidas. São frequentemente **sobrecompletos**.

- **Autoencoders Esparsos:** Força esparsidade na representação latente
- **Denoising Autoencoders:** Perturba a entrada com ruído e reconstrói a entrada limpa
- **Contractive Autoencoders:** Penaliza o jacobiano do encoder em relação à entrada
- **Variational Autoencoders:** Espaço latente se torna uma distribuição probabilística

Sparse Autoencoders

5 Autoencoders com regularização

Objetivo: Forçar a representação latente a ser esparsa, ou seja, a maioria dos neurônios na camada latente deve estar inativa (valores próximos de zero).

Função de Custo Modificada:

$$L(\theta, \phi) = \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2^2 + \alpha \|\boldsymbol{h}\|_1$$

Table of Contents

Referências Bibliográficas

6 Referências Bibliográficas

[1] Rumelhart, E. David, M. James, and L. James, *Parallel distributed processing: explorations in the microstructure of cognition. Volume 1. Foundations*. 01 1986.

[2] U. Michelucci, "An introduction to autoencoders," *CoRR*, vol. abs/2201.03898, 2022.

[3] M. Tschannen, O. Bachem, and M. Lucic, "Recent advances in autoencoder-based representation learning," *CoRR*, vol. abs/1812.05069, 2018.

[4] E. Oja, "Simplified neuron model as a principal component analyzer," *Journal of Mathematical Biology*, vol. 15, pp. 267–273, 1982.

Autoencoders: da motivação às variantes modernas

Obrigado pela Atenção!
Alguma Pergunta?

Natanael Moura Junior

natmourajr@poli.ufrj.br, natmourajr@lps.ufrj.br