

Projeto 3 - Autoencoders Variacionais

Guilherme Pereira Campos RA:163787
Lucas Oliveira Nery de Araújo RA:158882
Universidade Federal de São Paulo

I. RESUMO

Este projeto teve como objetivo treinar modelos *Variational Autoencoders* (VAEs) em dois datasets rotulados, *MNIST* e *Fashion MNIST*, ajustando a topologia do modelo com base na função de custo. A exploração do espaço latente foi realizada para avaliar a formação de clusters, a separação dos rótulos e a variância explicada, além de investigar a possibilidade de enviar o espaço latente com os rótulos das amostras. O projeto demonstra a eficácia dos VAEs na reconstrução de imagens e na análise do espaço latente, com foco na visualização e interpretação dos padrões nos dados.

II. DATASETS

A. MNIST

O primeiro dataset utilizado é o *MNIST* (*Modified National Institute of Standards and Technology*), ele contém 70.000 imagens em escala de cinza, com tamanho de 28x28 pixels, representando dígitos manuscritos de 0 a 9. O dataset é dividido em 60.000 imagens de treinamento e 10.000 de teste.

B. Fashion MNIST

O segundo dataset utilizado é o *Fashion MNIST*. Ele contém 70.000 imagens em escala de cinza, com dimensões de 28x28 pixels, representando 10 categorias de roupas e acessórios.

C. Preparação dos dados

A seguinte preparação foi feita para ambos datasets:

- **Normalização das Imagens:**

- Os pixels originalmente variam de **0 a 255**.
- Para melhorar a estabilidade do modelo, os valores são convertidos para o **intervalo [0,1]**, usando divisão por 255.

- **Separação dos Conjuntos:**

- Para avaliar o desempenho do modelo antes do teste, **separamos 5.000 imagens do treino para validação**.
- Distribuição final dos dados:
 - * **Treino:** 55.000 imagens
 - * **Validação:** 5.000 imagens
 - * **Teste:** 10.000 imagens

- **Visualização dos Dados:**

- São exibidas **10 imagens do conjunto de treino** com seus respectivos rótulos.
- Isso permite verificar a qualidade e distribuição dos dados antes do treinamento.

Exemplos do MNIST

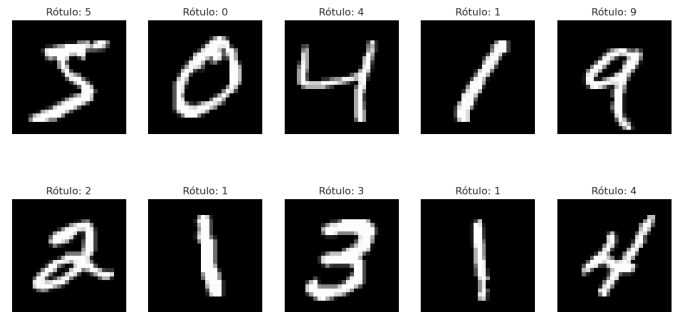


Fig. 1. Exemplos MNIST

Exemplos do Fashion MNIST

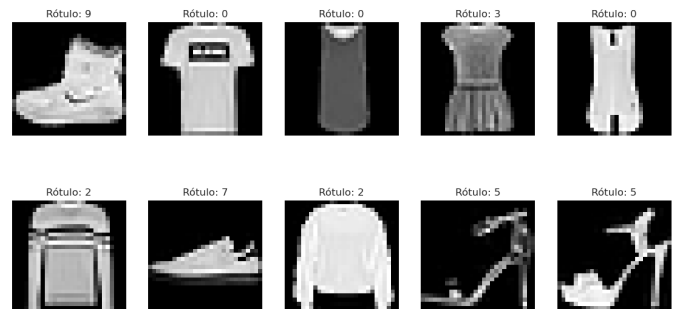


Fig. 2. Exemplos Fashion MNIST

III. AUTOENCODERS VARIACIONAIS (VAE)

A. Modelos Generativos

Os modelos generativos, como o Variational Autoencoder (VAE), geram novos dados a partir do conhecimento extraído de um conjunto inicial. Diferentemente dos modelos discriminativos, que modelam a probabilidade condicional $P(Y|X)$, os modelos generativos modelam a distribuição conjunta $P(X, Y)$, aprendendo a estrutura subjacente dos dados e utilizando essa informação para criar novos exemplos semelhantes.

O VAE impõe uma estrutura probabilística no espaço latente, garantindo que ele siga uma distribuição específica, como uma normal. Isso possibilita a geração de novos dados realistas a partir da amostragem nesse espaço latente. Aplicações incluem geração de imagens, síntese de fala, criação de novas moléculas e transferência de estilo, entre outras.

B. Autoencoders

Os autoencoders são redes neurais compostas por duas partes principais: o codificador (*encoder*) e o decodificador (*decoder*). Eles são modelos não supervisionados que aprendem uma representação compacta dos dados, reconstruindo as entradas como saída. O codificador mapeia a entrada x para um espaço latente z , enquanto o decodificador reconstrói x a partir de z .

A função de perda utilizada nos autoencoders tradicionais é dada por:

$$\text{loss} = \|x - d(e(x))\|^2 \quad (1)$$

onde x representa a entrada, $e(x)$ é a codificação gerada pelo codificador, e $d(e(x))$ é a reconstrução da entrada pelo decodificador. O objetivo é minimizar essa diferença para que a reconstrução se aproxime ao máximo dos dados originais.

C. Autoencoders Variacionais

O Variational Autoencoder (VAE) estende os autoencoders tradicionais ao introduzir uma distribuição latente $p(z|x)$. O encoder aprende uma distribuição de probabilidades para cada entrada, permitindo a geração de novos exemplos ao amostrar do espaço latente.

Para garantir um espaço latente contínuo e navegável, o VAE é regularizado, prevenindo sobreajuste (*overfitting*). Diferente dos autoencoders convencionais, que aprendem uma única representação z , o VAE aprende uma distribuição, garantindo uma interpolação suave entre os pontos no espaço latente.

A função de perda do VAE é composta por dois termos: o erro de reconstrução e a divergência KL (*Kullback-Leibler*) entre a distribuição latente e uma distribuição normal padrão $N(0, I)$:

$$\text{loss} = \|x - d(e(x))\|^2 + \text{KL}(N(\mu_x, \sigma_x), N(0, I)) \quad (2)$$

O objetivo é minimizar tanto o erro de reconstrução quanto a divergência KL, garantindo um espaço latente adequado para a geração de novos dados.

D. Divergência KL e Reparametrização

A divergência KL aproxima a distribuição latente $p(z|x)$ de uma normal padrão $N(0, I)$, garantindo um espaço latente contínuo. Para permitir a retropropagação durante o treinamento, o truque de reparametrização é aplicado, reescrevendo a amostragem como:

$$z = \mu_x + \sigma_x \cdot \epsilon, \quad \text{com } \epsilon \sim N(0, I) \quad (3)$$

Essa técnica permite o ajuste diferenciável dos parâmetros μ_x e σ_x , tornando o modelo treinável por gradiente descendente.

IV. MODELOS

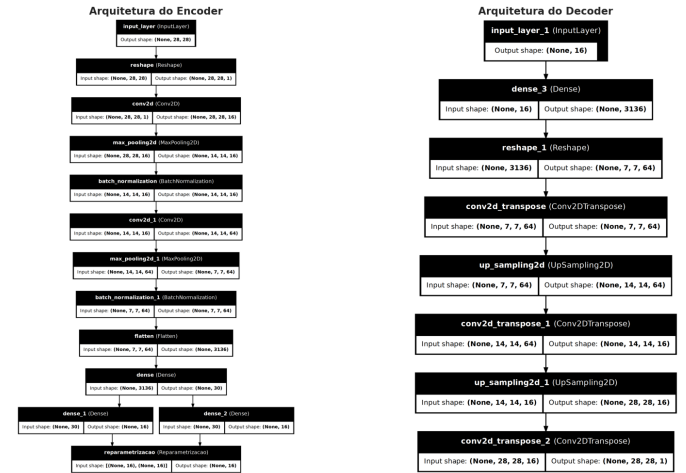
A biblioteca utilizada para implementar o VAE (Variational Autoencoder) neste projeto será o *TensorFlow* (<https://www.tensorflow.org>), um framework amplamente utilizado para aprendizado profundo.

A. Primeiro modelo VAE

O primeiro modelo *Variational Autoencoder* (VAE) convolucional foi configurado para processar imagens 28x28 e aprender uma representação latente compacta. O *encoder* utiliza duas camadas convolucionais com 16 e 64 filtros, *kernel* 3x3 e ativação ReLU, seguidas por operações de *MaxPooling2D* e *BatchNormalization* para estabilizar o treinamento. O espaço latente tem dimensão 16, onde os parâmetros da distribuição latente são ajustados por meio das camadas *codings_mean* e *codings_log_var*. A amostragem é realizada utilizando a técnica de reparametrização.

O *decoder* reconstrói as imagens a partir do vetor latente utilizando camadas densas e convolucionais transpostas (*Conv2DTranspose*), combinadas com operações de *UpSampling2D* para restaurar as dimensões espaciais. A camada final usa ativação sigmoial para garantir que os valores da imagem reconstruída permaneçam entre 0 e 1. Além disso, o modelo será treinado por 30 épocas.

A figura abaixo conta com o *plot_model*, que permite visualizar a arquitetura do modelo de forma gráfica, exibindo a conexão entre as camadas do *encoder* e do *decoder*, além das dimensões das saídas em cada etapa do processamento.



próxima à de treinamento, significa que o modelo generaliza bem, sem *overfitting*.



2) *Reconstrução VAE do primeiro modelo no primeiro dataset*: A reconstrução no VAE começa com o *encoder* comprimindo a imagem em um espaço latente, gerando média e log da variância de uma distribuição probabilística. A camada de reparametrização amostra um vetor latente a partir desses parâmetros, que é passado ao *decoder*. O *decoder* reconstrói a imagem aplicando camadas convolucionais transpostas para recuperar sua estrutura original. Durante o treinamento, a perda de reconstrução mede a diferença entre a imagem original e a reconstruída, enquanto a perda KL regula o espaço latente, garantindo uma boa representação.



Fig. 4. Reconstrução VAE do primeiro modelo no primeiro dataset

3) *Visualização do espaço latente do primeiro modelo no primeiro dataset*: O *t-SNE* (*t-Distributed Stochastic Neighbor Embedding*) é uma técnica de redução de dimensionalidade que transforma dados de alta dimensão em uma representação 2D ou 3D, preservando a proximidade dos pontos semelhantes. Ao aplicar *t-SNE* ao espaço latente de um modelo como o VAE, podemos visualizar como as amostras estão agrupadas no espaço aprendido, com base nas distâncias entre elas. Usando os rótulos das amostras, conseguimos observar a separação entre as classes. A adição das imagens no gráfico permite entender melhor como as amostras se distribuem no espaço latente.

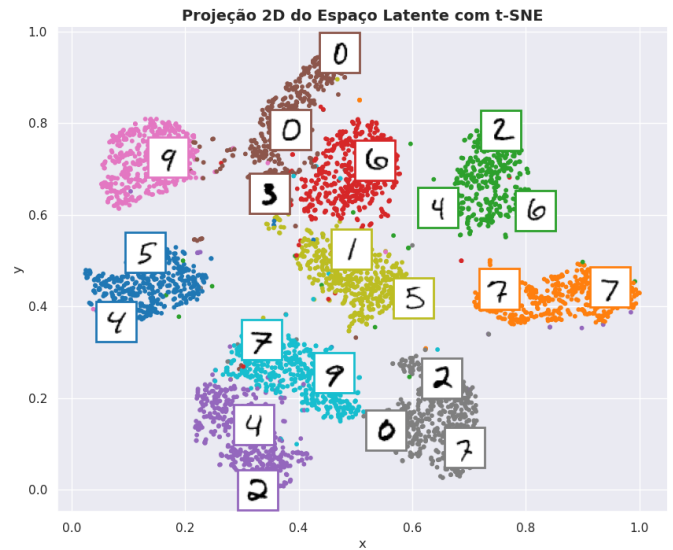


Fig. 5. Visualização do espaço latente do primeiro modelo no primeiro dataset

O primeiro modelo VAE configurado apresentou um desempenho satisfatório, com boas reconstruções das imagens, mas não de forma ótima. A qualidade das reconstruções ficou abaixo do esperado, com algumas perdas perceptíveis. O encoder conseguiu extrair informações essenciais, mas a configuração das camadas e o número de unidades na camada densa (30 unidades) podem ser insuficientes. O decoder gerou imagens razoáveis, mas poderia ser mais refinado. Além disso, na projeção 2D do espaço latente, alguns clusters não se separaram muito bem, indicando que o modelo não conseguiu capturar completamente as relações entre as classes. Ajustes nos hiperparâmetros e na arquitetura poderiam melhorar a fidelidade das reconstruções e a separação no espaço latente.

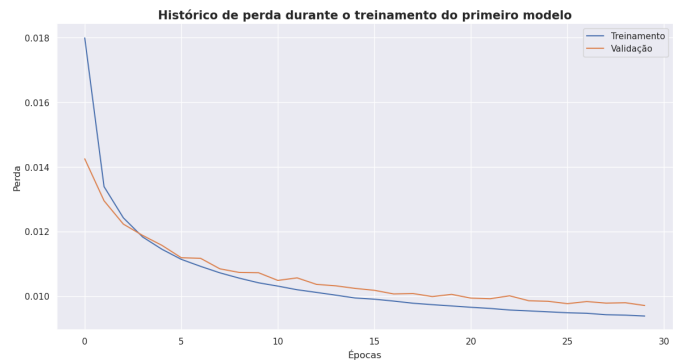


Fig. 6. Histórico de perda durante o treinamento do primeiro modelo no segundo dataset

O primeiro modelo no segundo dataset não apresentou um desempenho satisfatório. Dado que o Fashion MNIST é um dataset mais complexo, com imagens de roupas e acessórios, as reconstruções não foram muito boas e apresentaram algumas distorções perceptíveis. A separação dos clusters no espaço latente também não foi ideal, com as diferentes classes se sobrepondo de forma visível. Embora o encoder tenha sido



Fig. 7. Reconstrução VAE do primeiro modelo no segundo dataset

capaz de capturar algumas características das imagens, a arquitetura do modelo e o número de unidades na camada densa podem não ter sido adequados para a complexidade dos dados. A qualidade das reconstruções e a separação das classes no espaço latente indicam que ajustes nos hiperparâmetros e melhorias na arquitetura são necessários para alcançar um desempenho melhor e mais refinado.

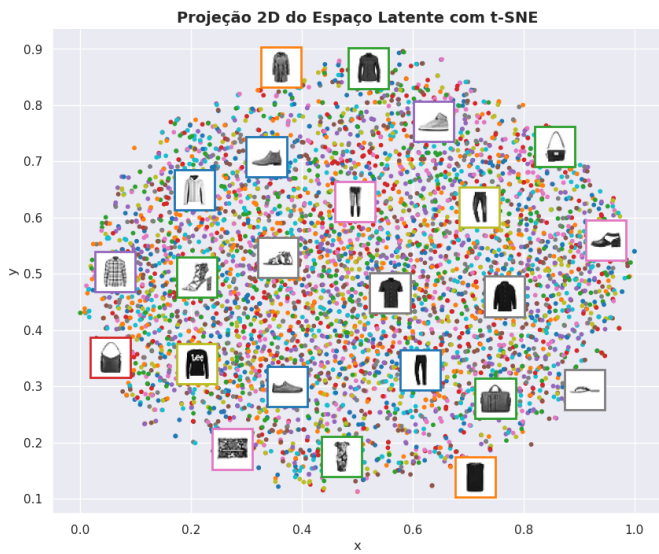


Fig. 8. Projeção do espaço latente do primeiro modelo no segundo dataset

B. Segundo modelo VAE

O segundo modelo VAE convolucional foi configurado para processar imagens de 28×28 pixels e aprender uma representação latente de dimensão 32, o dobro do primeiro modelo. O *encoder* possui quatro camadas convolucionais com filtros de 32 a 512, ativação *ReLU*, *MaxPooling2D* e *BatchNormalization*, ajustando a distribuição latente com *codings_mean* e *codings_log_var*. O *decoder* usa camadas densas, convolucionais transpostas e *UpSampling2D* para reconstruir as imagens. A camada final aplica uma ativação sigmoidal para garantir valores entre 0 e 1. O modelo foi treinado por 50 épocas para otimizar a reconstrução.

A figura abaixo conta com o *plot_model*, que permite visualizar a arquitetura do modelo de forma gráfica, exibindo a conexão entre as camadas do *encoder* e do *decoder*, além das dimensões das saídas em cada etapa do processamento.

O segundo modelo VAE apresentou um desempenho superior ao primeiro, com reconstruções mais refinadas e uma melhor separação do espaço latente. O *encoder*, com camadas

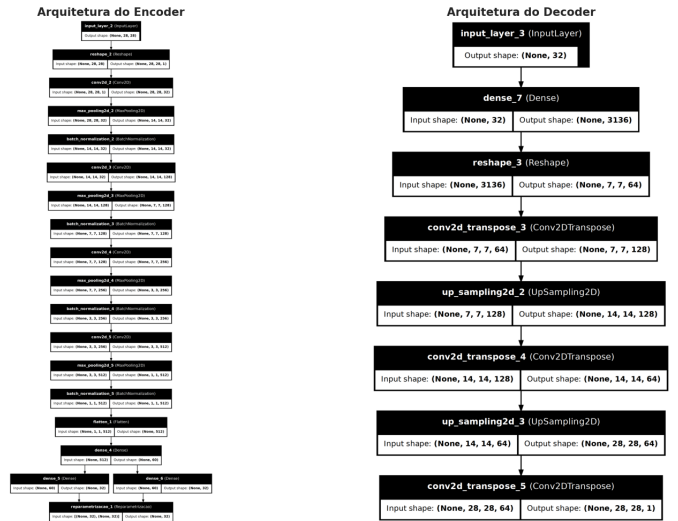


Fig. 9. Arquitetura do segundo modelo VAE



Fig. 10. Histórico de perda durante o treinamento do segundo modelo

convolucionais mais profundas e 60 unidades densas, extraiu informações mais ricas, enquanto o *decoder*, com mais filtros e camadas adicionais, gerou imagens de maior fidelidade. No *Fashion MNIST*, as reconstruções foram mais precisas e os clusters no espaço latente apresentaram uma organização mais clara, apesar de alguma sobreposição. O aumento da profundidade e do número de neurônios permitiu ao modelo aprender representações mais robustas e estruturadas.

V. CONCLUSÃO

Em conclusão, este projeto demonstrou a eficácia dos Variational Autoencoders (VAEs) na detecção de padrões e organização de dados em dois datasets rotulados: MNIST e Fashion MNIST. A aplicação dos VAEs permitiu explorar o

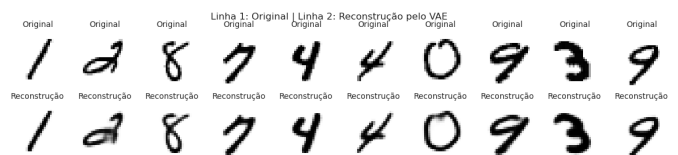


Fig. 11. Reconstrução do segundo modelo VAE

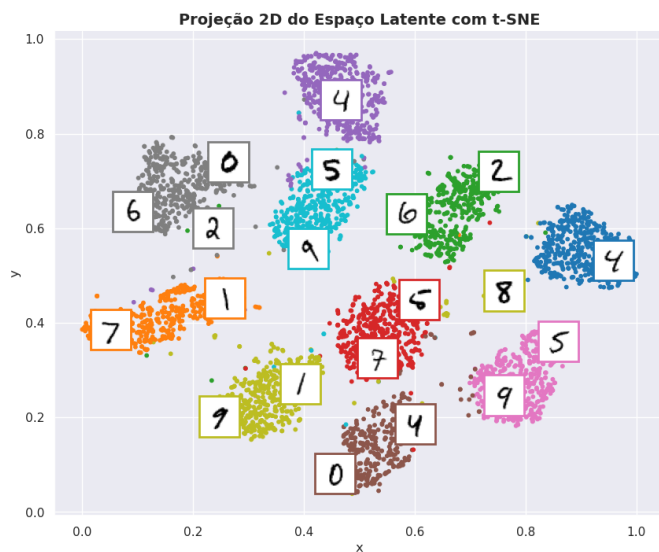


Fig. 12. Projeção do espaço latente do segundo modelo VAE



Fig. 13. Reconstrução do segundo modelo VAE

espaço latente, identificar clusters, analisar a separação dos rótulos e investigar a variância explicada. A análise também incluiu a possibilidade de enviesar o espaço latente com os rótulos, oferecendo uma visão mais detalhada sobre a relação entre as amostras. Os resultados indicaram que, mesmo com ajustes simples no modelo, o VAE foi eficaz na reconstrução das imagens e na exploração dos dados.