

Autoencoders e Variational Autoencoders (VAE)

Almir Sérgio Ramos dos Santos Filho

NES - Redes Neurais

- Dataset: Fashion-MNIST (28×28 grayscale)
- Modelos: Autoencoder denso (latent=32) e VAE denso (latent=2)
- Objetivo: comparar reconstrução vs. geração e visualizar espaço latente

- Importações e configuração:
 - NumPy, Matplotlib, TensorFlow/Keras, SciPy.
 - Fixação de seed para reprodutibilidade.
- Preparação dos dados:
 - Carregamento do dataset Fashion-MNIST.
 - Normalização das imagens para o intervalo $[0,1]$.
 - Achatamento para vetores de 784 dimensões.
 - Separação dos tensores usados no AE e no VAE.

- Autoencoder:
 - Encoder denso: $784 \rightarrow 256 \rightarrow 32$.
 - Decoder simétrico: $32 \rightarrow 256 \rightarrow 784$.
 - Treinamento com `binary_crossentropy`.
- Variational Autoencoder (VAE):
 - Encoder produz `z_mean` e `z_log_var`.
 - Camada Sampling: truque da reparametrização.
 - Função de perda = reconstrução + KL divergence.
 - Classe VAE sobrescreve `train_step`.
- Visualizações produzidas:
 - Reconstruções AE e VAE.
 - Geração no grid do espaço latente.
 - Scatter 2D das classes no latente.

Reconstruções: Autoencoder (AE)

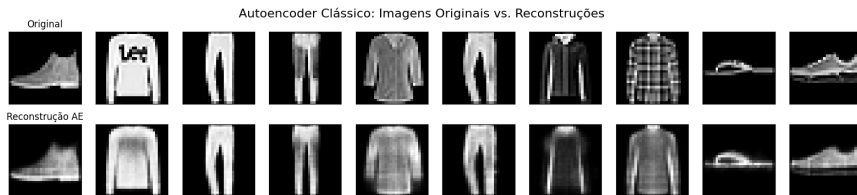


Figure: *

Original (linha superior) vs Reconstrução AE (linha inferior)

Reconstruções: Variational Autoencoder (VAE)

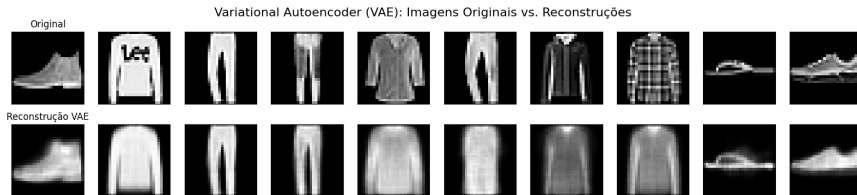


Figure: *

Original (linha superior) vs Reconstrução VAE (linha inferior)

Geração no Espaço Latente (VAE)

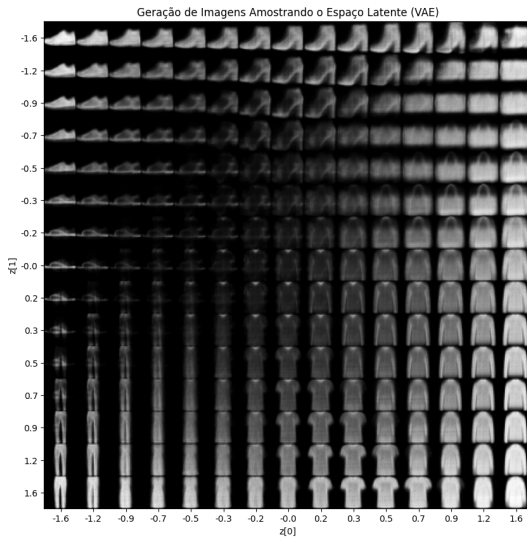


Figure: *

Comparação Visual AE vs VAE

Comparação Visual: Original vs. Reconstrução AE vs. Reconstrução VAE

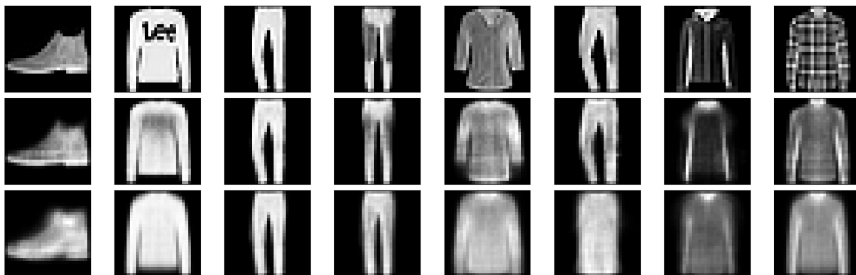


Figure: *

linhas: Original / AE / VAE

Visualização do Espaço Latente 2D

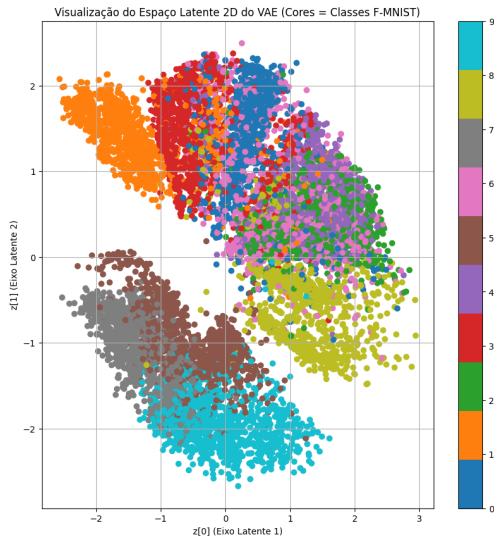


Figure: *

- AE: val_loss final ≈ 0.2787 (binary crossentropy). Convergência rápida.
- VAE: reconstruction_loss final ≈ 254.06 (escala por 784); KL ≈ 6.39 (estável).
- Interpretação: escala diferente — comparar qualitativamente ou normalizar métricas.

- AE: reconstruções mais nítidas; latente não regularizado (bom para reconstrução).
- VAE: reconstruções mais suaves; latente contínuo e semântico (bom para geração).
- O scatter 2D confirma clusters por classe, com sobreposições onde há similaridade visual.

- Uso de camadas densas reduz qualidade comparado a CNNs.
- Latente 2D limita diversidade; recomendado para visualização, não produção.
- Reconstruções VAE podem melhorar com arquitetura e treino mais robustos.

- Migrar para Conv-VAE (ConvEncoder + ConvTranspose decoder).
- Aumentar dimensionalidade do latente; testar β -VAE e KL-annealing.
- Avaliar métricas perceptuais (SSIM, PSNR) e gerativas (FID).
- Testar CVAE para geração condicionada e VAE-GAN para nitidez.

- AE reconstrói melhor; VAE organiza latente e gera imagens novas.
- Trade-off clássico: fidelidade \times capacidade gerativa.
- Com ajustes, este pipeline torna-se base para modelos geradores mais potentes.

Obrigado!

Perguntas?