
🧠 Exercício: Construindo um VAE com Privacidade Diferencial usando Copilot

🎯 Objetivo

Desenvolver um modelo de Autoencoder Variacional (VAE) em Python utilizando PyTorch ou TensorFlow, e aplicar técnicas de **Privacidade Diferencial** para proteger os dados de treinamento.

📚 Parte 1: Introdução ao VAE

Tarefa: Com o auxílio do Copilot, gere um código que:

- Implemente um VAE simples com:
    - Encoder e decoder usando camadas lineares ou convolucionais
    - Função de perda que combine reconstrução + divergência KL
- Treine o modelo em um dataset público (por exemplo, MNIST ou Fashion-MNIST)

Dicas para os alunos:

- Pergunte ao Copilot: *"Como implementar um VAE em PyTorch usando MNIST?"*
- Explore a estrutura do encoder e decoder sugerida
- Analise a função de perda gerada e peça explicações ao Copilot

---

🔐 Parte 2: Adicionando Privacidade Diferencial

Tarefa: Modificar o treinamento do VAE para incluir privacidade diferencial usando a biblioteca Opacus (para PyTorch) ou TensorFlow Privacy.

Desafios:

- Integrar o DPOptimizer no loop de treinamento
- Ajustar o clipping de gradientes e o nível de ruído
- Monitorar o consumo de privacidade (epsilon)

Dicas para os alunos:

- Pergunte ao Copilot: *"Como usar Opacus para treinar um modelo com privacidade diferencial?"*
- Solicite ao Copilot que explique o impacto do parâmetro noise_multiplier
- Teste diferentes valores de epsilon e discuta os trade-offs entre privacidade e desempenho

---

🧪 Parte 3: Avaliação e Reflexão

Tarefa: Avaliar o desempenho do VAE com e sem privacidade diferencial.

- Compare reconstruções visuais
- Meça a perda de reconstrução
- Discuta como a privacidade afeta a qualidade do modelo

Perguntas para discussão:

- Como o ruído afeta a capacidade do VAE de aprender representações úteis?
- Quais aplicações reais exigiriam esse tipo de proteção?
- O que você aprendeu ao usar o Copilot como parceiro de codificação?

🛠️ Recursos úteis

Biblioteca	Link Oficial
PyTorch	https://pytorch.org
Opacus	https://opacus.ai
TensorFlow Privacy	https://github.com/tensorflow/privacy
