

GABRIEL MILANEZ CAVALHERI

Mogi Guaçu, SP | (19) 98222-1223 | gmcavalheri@gmail.com
[linkedin.com/in/gabriel-milanez-cavalheri](https://www.linkedin.com/in/gabriel-milanez-cavalheri) | github.com/GMCavalheri

RESUMO

Físico (UFSCar) com MBA em Inteligência Artificial e Big Data (USP, em andamento) e formação prática em ciência e engenharia de dados. Experiência em projetos autorais de pipelines de ETL, integração entre Data Lake e Data Warehouse, modelagem de Machine Learning e automação com Docker. Busca posição como Engenheiro(a), Cientista ou Analista de Dados para aplicar conhecimento técnico em soluções orientadas a dados.

FORMAÇÃO ACADÊMICA

- MBA em Inteligência Artificial e Big Data** — Universidade de São Paulo (USP) | 07/2025 – 10/2026 (em andamento)
- Bacharelado em Física** — Universidade Federal de São Carlos (UFSCar) | Concluído em 03/2025
- Iniciação Científica em Acústica Aplicada** — UFSCar | 2023 – 2025

PROJETOS DE DADOS

Integração de Data Warehouse e Data Lake com Pipeline de ETL

<https://github.com/GMCavalheri/Complete-BI-and-ETL-with-Data-Warehouse-and-Datalake>

- Stack: Python, PostgreSQL, PyArrow, DuckDB, Polars, Docker, SQL
- Construção de pipeline de ETL que integra um Data Lake a um Data Warehouse relacional, com consultas SQL do nível básico ao avançado; todo o ambiente containerizado com Docker para reprodutibilidade por outros usuários.

Pipeline de Cotações Cambiais — Duas Arquiteturas: Airflow vs. AWS Serverless

<https://github.com/GMCavalheri/currency-rates-aws-serverless>

<https://github.com/GMCavalheri/Complete-ETL-Pipeline-with-Airflow>

- Stack: AWS Lambda, S3, EventBridge, Glue, Athena, Terraform (v2) · Python, Apache Airflow, PostgreSQL, Docker (v1)
- Mesmo problema (API de câmbio → dados limpos e consultáveis) resolvido com dois paradigmas de arquitetura: pipeline serverless orientado a eventos na AWS, 100% dentro do free-tier (Lambda + S3 + EventBridge + Glue + Athena, infraestrutura em Terraform), e pipeline batch self-hosted com Airflow, Docker e PostgreSQL — demonstrando domínio de cloud gerenciada e orquestração própria.

Detecção de Fraudes em Transações com Spark

<https://github.com/GMCavalheri/fraud-detection-spark>

- Stack: Apache Spark (cluster distribuído), Spark MLlib, MinIO, PostgreSQL, FastAPI, Streamlit, Docker Compose
- Pipeline distribuído de detecção de fraude: dados sintéticos de ~1,5M transações, vindos de 3 sistemas com schemas e problemas de qualidade distintos, são limpos e enriquecidos em um cluster Spark real (1 master + 2 workers); um modelo GBTCClassifier (MLlib) é treinado com validação temporal e servido via API (FastAPI) com scoring em tempo real, visualizado em um dashboard interativo (Streamlit). Suíte com 57 testes automatizados e CI configurado.

Smart Pokédex — Aplicação Web com Django e LLM

<https://github.com/GMCavalheri/Smart-Pokedex>

- Stack: Python, Django, PokéAPI, OpenRouter (LLM), MySQL, Redis, Docker Compose
- Aplicação web em Django que exibe dados de Pokémon via PokéAPI e permite consultas em linguagem natural a um LLM (via OpenRouter). Cache em Redis reduziu o tempo de carregamento da listagem de ~18s para dezenas de milissegundos; backend com suíte de testes automatizados (pytest) com 100% de cobertura.

Outros projetos aplicados

- Predição de diabetes** (Random Forest) e **diagnóstico de falhas em engrenagens mecânicas** a partir de vibração (Redes Neurais Profundas).
- Detecção de tumores cerebrais** em ressonância magnética (Redes Neurais Convolucionais).
- Análise de sentimento** em avaliações de produtos e **reconhecimento de entidades legais** em documentos jurídicos (NLP).
- Sistema de recomendação de filmes** (algoritmo APRIORI) e **análise geoespacial** de distribuição de desenvolvedores de software.

HABILIDADES TÉCNICAS

- Dados & Engenharia:** SQL (PostgreSQL, MySQL, Spark SQL), NoSQL (MongoDB), Apache Spark, Apache Airflow, ETL, PyArrow, DuckDB, Polars
- Linguagens:** Python (Pandas, NumPy, SciPy, Scikit-Learn, TensorFlow, Keras, Streamlit), R (tidyverse, dplyr, ggplot2)
- Machine Learning & IA:** Aprendizado Supervisionado, Não-Supervisionado e por Reforço; Redes Neurais (MLP, CNN, RNN, Transformers, Generativas); NLP (NLTK, spaCy)
- Cloud & DevOps:** AWS (Lambda, S3, EventBridge, Glue, Athena), Terraform, Docker, Kubernetes (básico), Git/GitHub
- Análise & Visualização:** Microsoft Excel, Power BI, Seaborn, estatística aplicada (inferência, testes de hipótese, distribuições)
- IA Generativa & Automação:** Claude, ChatGPT, Gemini, Ollama e modelos open-source locais; n8n, Make
- Metodologia:** Ágil, Kanban, Clean Code, Storytelling com dados

IDIOMAS

Português — Nativo | Inglês — Avançado (C1 Advanced) | Espanhol — Básico