

# Pipelines de Machine Learning com ZenML: Uma Abordagem Prática Aplicada ao Mercado Financeiro

**Vitor de Castro Neves Lobianco**

lobiancocnvitor@hotmail.com

IBMEC

**Cassius Marcellus do Carmo Figueiredo**

cassiusf@pobox.com

IBMEC

**Thiago Silva de Souza**

t.souza@ibmec.edu.br

IBMEC

**Resumo:** Este trabalho investiga como práticas de Machine Learning Operations (MLOps) podem facilitar a produtização de modelos de aprendizado de máquina, utilizando ZenML e MLflow para estruturar pipelines modulares, reprodutíveis e monitoráveis. A partir de uma revisão de literatura e de um estudo de caso no mercado financeiro, foi implementada uma solução de previsão de séries temporais com Random Forest, priorizando organização, rastreabilidade e automação. Os resultados demonstraram que a integração do ZenML, para construção de pipelines desacoplados, com o MLflow, para rastreamento de métricas e artefatos, contribui para a reprodutibilidade e escalabilidade em projetos de aprendizado de máquina. Conclui-se que a abordagem proposta se mostra eficiente para superar desafios comuns na etapa de produção de modelos, oferecendo uma infraestrutura robusta e confiável para uso em ambientes reais.

**Palavras Chave:** Machine Learning - MLOps - ZenML - MLFlow - Produtização

## 1. INTRODUÇÃO

O aprendizado de máquina (*machine learning*) está revolucionando a forma como as organizações lidam com dados e tomam decisões, estando cada vez mais presente em setores como a saúde e finanças. No entanto, ao mesmo tempo que trouxe novas soluções, a etapa de produtização de modelos de aprendizado de máquina continua sendo um dos maiores desafios enfrentados por profissionais da área (Liang et al., 2024).

Estudos apontam que a maioria dos projetos de aprendizado de máquina nunca chega a ser colocado em produção, sendo interrompidos ainda na fase de experimentação (Kreuzberger et al., 2022). E esse fato não surpreende, considerando que grande parte dos esforços concentram-se no desenvolvimento de modelos, e não nos componentes e infraestrutura necessários, que compõem um sistema complexo, para que esses modelos possam ser utilizados em casos reais.

Metodologias tradicionais de desenvolvimento de software, como o DevOps, não atendem às especificidades do *machine learning* (Zaharia et al., 2018). Isso gera desafios tanto na implantação desses modelos como no monitoramento e nos processos automatizados que vemos em softwares tradicionais, como o CI/CD. Como resposta a esses desafios, surge o conceito de *Machine Learning Operations* (MLOps).

O MLOps é uma metodologia que tem como objetivo facilitar a produtização de modelos de aprendizado de máquina, possibilitando seu uso confiável em aplicações de larga escala (Liang et al., 2024). Dessa forma, existem diversas ferramentas que auxiliam na implementação prática do MLOps, que abrangem desde o desenvolvimento de modelos até seu monitoramento quando em produção. Cada etapa do processo exige uma ferramenta específica, que ao se complementarem formam um ambiente propício ao MLOps.

Diante desse contexto, este trabalho investiga como o MLOps, apoiado pelas ferramentas ZenML e MLflow, pode ser utilizado para facilitar a produtização de modelos de aprendizado de máquina. Para isso, vamos discutir os fundamentos e atuais desafios do MLOps, seguido de um estudo de caso prático no mercado financeiro que evidencia a aplicação das ferramentas analisadas.

## 2. PROBLEMA DE PESQUISA E OBJETIVO

Diante da crescente relevância do aprendizado de máquina em setores como saúde, finanças e indústria, observa-se um desafio recorrente: muitos modelos desenvolvidos não chegam a ser utilizados em produção, em grande parte devido à ausência de processos estruturados e de infraestrutura adequada que garantam reproduzibilidade, escalabilidade e rastreabilidade. Nesse contexto, práticas de *Machine Learning Operations* (MLOps) surgem como uma abordagem promissora para superar essas barreiras, integrando princípios do DevOps ao ciclo de vida de projetos de aprendizado de máquina, com foco em automação de pipelines, monitoramento de modelos em produção e versionamento de dados e artefatos.

Dessa forma, surge a seguinte pergunta de pesquisa: **como as práticas de *Machine Learning Operations* (MLOps) podem ser aplicadas de forma efetiva para facilitar a produtização de modelos de aprendizado de máquina, garantindo pipelines modulares, reproduzíveis e monitoráveis?**

Para responder a essa questão, este trabalho tem como objetivo analisar a aplicabilidade de práticas de MLOps por meio de um estudo de caso prático, utilizando o ZenML, integrado ao MLflow, como ferramentas centrais na construção de pipelines de aprendizado de máquina. O estudo visa demonstrar, de forma estruturada, como tais ferramentas podem ser utilizadas em

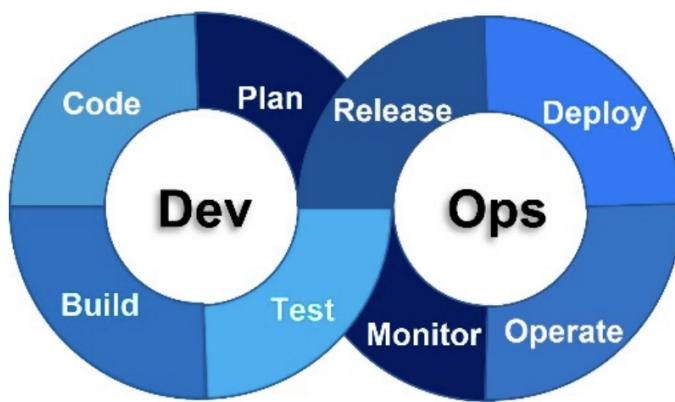
conjunto para organizar o fluxo de trabalho de projetos de *machine learning*, permitindo rastreamento de experimentos, versionamento de dados e modelos, e automação de etapas desde a preparação dos dados até a implantação do modelo em produção, alinhando-se aos princípios fundamentais de MLOps.

### 3. REFERENCIAL TEÓRICO

O conceito de MLOps surge como uma adaptação das práticas de DevOps aplicadas ao ciclo de vida do aprendizado de máquina. Ao integrar três disciplinas — aprendizado de máquina, engenharia de software e engenharia de dados —, o MLOps busca otimizar os processos de desenvolvimento, entrega e monitoramento dos modelos preditivos. Dessa forma, estabelece uma ponte entre o desenvolvimento (Dev) e a operação (Ops), facilitando a produtização de sistemas baseados em aprendizado de máquina (Kreuzberger et al., 2023).

Assim como o DevOps promoveu a automação e integração de etapas no desenvolvimento de software tradicional, o MLOps introduz uma série de práticas que visam garantir a reprodutibilidade, escalabilidade e robustez dos modelos de machine learning. Entre essas práticas estão: automação de pipelines de CI/CD; versionamento não apenas do código, mas também dos dados e modelos; treinamento e validação contínua; rastreamento de experimentos e metadados; além do monitoramento de modelos em tempo real (Ruf et al., 2021).

Por sua vez, o DevOps consolidou-se como um paradigma essencial na engenharia de software. Suas práticas promovem a colaboração entre as equipes de desenvolvimento e operação, resultando em ciclos de entrega mais curtos e produtos finais de maior qualidade. A figura 1 exemplifica um workflow típico do DevOps.



**Figura 1:** Workflow típico de DevOps.

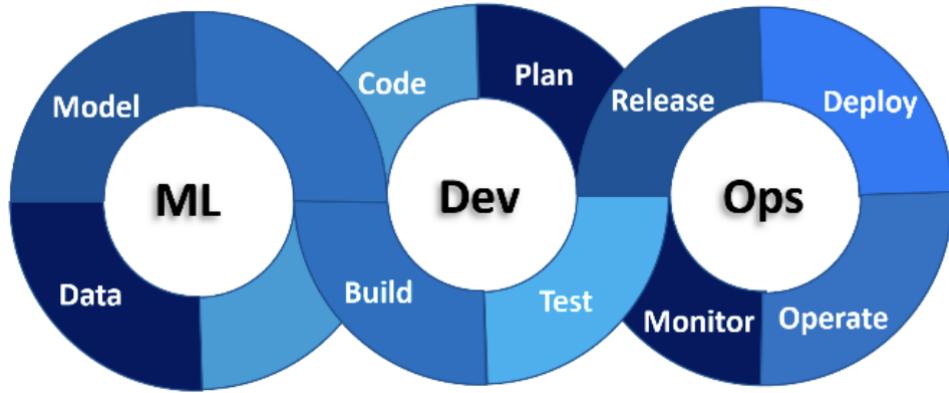
**Fonte:** Subramanya et al. (2022)

Os princípios criados a partir do DevOps mostraram-se úteis para projetos de aprendizado de máquina, uma vez que diversos ciclos iterativos são necessários para identificar modelos eficazes (Subramanya et al., 2022). O ciclo de vida de um modelo preditivo é composto por etapas que demandam revisão contínua, desde a coleta e preparação de dados até a implantação e monitoramento do modelo em produção.

Diferente de sistemas tradicionais, onde o código é o principal ativo, em projetos de aprendizado de máquina os dados assumem papel central, sendo frequentemente modificados e reavaliados (Wang et al., 2022). Dessa forma, a transição do DevOps para o MLOps representa uma adaptação de práticas tradicionais de desenvolvimento de software,

incorporando novas tecnologias e ferramentas específicas para o domínio de aprendizado de máquina.

Por isso, o MLOps enfatiza ferramentas e técnicas que asseguram o versionamento dos dados, rastreamento de experimentos, gestão de modelos e monitoramento de desempenho em tempo real. A figura 2 ilustra um workflow típico de MLOps, evidenciando as diferenças e complementaridades em relação ao DevOps.



**Figura 2:** Workflow típico de MLOps.

**Fonte:** Subramanya et al. (2022)

Cada etapa de um workflow de MLOps requer ferramentas especializadas. Não existe, portanto, uma ferramenta única que atenda a todo o ciclo de vida do modelo. Neste trabalho, destacam-se duas ferramentas de código aberto que têm ganhado espaço na comunidade: ZenML e MLflow.

### 3.1. ZENML

ZenML é um *framework* em Python orientado à construção de pipelines de *machine learning* reproduutíveis e modulares. Destaca-se por seu design agnóstico, o que permite integrações com diferentes plataformas e ferramentas, o que o torna versátil para projetos que demandam escalabilidade. Dentre suas funcionalidades estão: rastreamento automático de artefatos, versionamento de dados e modelos, e compatibilidade com ambientes como Kubernetes, Airflow e Kubeflow.

Um dos principais diferenciais do ZenML é sua abordagem declarativa, que permite padronizar pipelines de maneira intuitiva, promovendo reuso de componentes. Apesar de ainda em crescimento, a ferramenta apresenta potencial na organização de workflows de MLOps. A figura 3 apresenta a interface de visualização dos pipelines no *dashboard* do ZenML.



## Pipelines

Pipeline	Latest Run
<input type="checkbox"/> inference 04e20508	853e3082
<input type="checkbox"/> training c230fd08	1465047

**Figura 3:** Interface de visualização de pipelines do ZenML.

**Fonte:** Autores (2025).

### 3.2. MLFLOW

Desenvolvido pela Databricks, o MLflow é uma plataforma utilizada para gerenciar o ciclo de vida de modelos de aprendizado de máquina, com foco em rastreamento de experimentos, gerenciamento de modelos e reproduzibilidade. O MLflow fornece três componentes principais que podem ser utilizados em conjunto ou separadamente: Tracking, para registro de execuções e metadados; Projects, para empacotamento do código de maneira reproduzível; e Models, que facilita a implantação de modelos em diferentes ambientes.

É uma escolha popular para equipes que buscam rastreabilidade e comparabilidade entre experimentos de forma prática. Ainda, sua interface web permite a visualização de experimentos e modelos (Figura 4). Neste trabalho, atua como uma ferramenta complementar ao ZenML, compondo uma solução robusta para o MLOps.

Run Name	Created	Dataset	Duration	Source	Models
indecisive-bat-488	9 minutes ago	-	5.5s	run.py	sklearn
orderly-koi-949	1 month ago	-	5.4s	run.py	sklearn
auspicious-stoat-26	1 month ago	-	6.1s	run.py	sklearn
gifted-toad-257	1 month ago	-	10.1s	run.py	sklearn
able-foal-574	1 month ago	-	11ms	run.py	-
nebulous-ray-165	1 month ago	-	6.4s	run.py	sklearn
hilarious-quail-507	1 month ago	-	69ms	run.py	-
clumsy-turtle-85	1 month ago	-	11.5s	run.py	sklearn

**Figura 4:** Interface de rastreamento do MLFlow.

**Fonte:** Autores (2025).

## 4. METODOLOGIA

Foi realizada uma revisão da literatura sobre MLOps, abordando seus princípios e práticas usuais, como automação de pipelines e o versionamento e monitoramento de modelos. Também

foram analisadas ferramentas que viabilizam essas práticas, com ênfase no ZenML e sua integração com o MLflow.

A etapa prática consistiu na implementação de uma solução de aprendizado de máquina voltada para previsão de séries temporais financeiras, utilizando um modelo Random Forest. O foco principal foi demonstrar a organização dos pipelines, a automação das etapas e o rastreamento dos experimentos, e não a performance do modelo em si.

## 5. ANÁLISE DE RESULTADOS

### 5.1. ESTUDO DE CASO

Para demonstrar a aplicabilidade prática dos conceitos de MLOps, foi desenvolvido um estudo de caso voltado à previsão de séries temporais financeiras. O objetivo central não foi alcançar a maior performance preditiva possível, mas sim estruturar uma arquitetura modular e reproduzível, integrando o framework ZenML e o MLflow.

A tarefa preditiva envolveu a geração de sinais de compra e venda para o ouro, utilizando dados históricos. Para modelagem, adotou-se o algoritmo Random Forest, devido à sua robustez em contextos com múltiplas variáveis explicativas. A arquitetura adotada, no entanto, foi projetada de forma modular, permitindo substituição por outros algoritmos, como XGBoost ou redes neurais, evidenciando a flexibilidade do ZenML.

### 5.2. DESAFIOS TÉCNICOS E MOTIVAÇÕES DA ARQUITETURA

A construção do pipeline foi motivada por uma série de desafios comuns em projetos de aprendizado de máquina, como:

- Reproducibilidade: garantir que os dados e as transformações possam ser reproduzidas com consistência, utilizando versionamento do código, dados e dos modelos.
- Rastreabilidade: permitir a auditoria completa do processo, com logging de parâmetros, métricas e artefatos por meio do MLflow.
- Automação: automatizar desde o carregamento e transformação dos dados até a seleção e uso do melhor modelo para a inferência.

Esses aspectos foram incorporados à arquitetura para garantir rastreabilidade, escalabilidade e manutenção mais eficiente da aplicação.

### 5.3. ARQUITETURA DA SOLUÇÃO COM ZENML

A solução foi organizada em três pipelines principais:

- Pipeline de *Feature Engineering*
- Pipeline de Treinamento
- Pipeline de Inferência

Esses pipelines foram implementados utilizando os principais componentes do ZenML:

- Pipelines: representam a estrutura de alto nível que define a sequência de execução das etapas (*steps*), definidos como funções Python com a anotação `@pipeline`.
- Steps: etapas independentes e reutilizáveis, representando cada etapa do pipeline. São definidos como funções Python com a anotação `@step`.
- Artifacts: todos os outputs produzidos pelos steps são armazenados como artefatos pelo ZenML, possibilitando reuso e rastreabilidade.

#### 5.4. PIPELINE DE FEATURE ENGINEERING

Responsável por todo o processamento e preparação dos dados de forma padronizada. Compreende as seguintes etapas:

- data\_loader: carregamentos dos dados históricos do ativo;
- data\_preprocessor: limpeza, transformações e tratamento de valores ausentes;
- feature\_generation: criação de variáveis explicativas (como indicadores técnicos);
- data\_splitter: separação dos dados em conjuntos de treino e teste;
- data\_scaler: normalização dos dados, de forma consistente entre treino e teste.

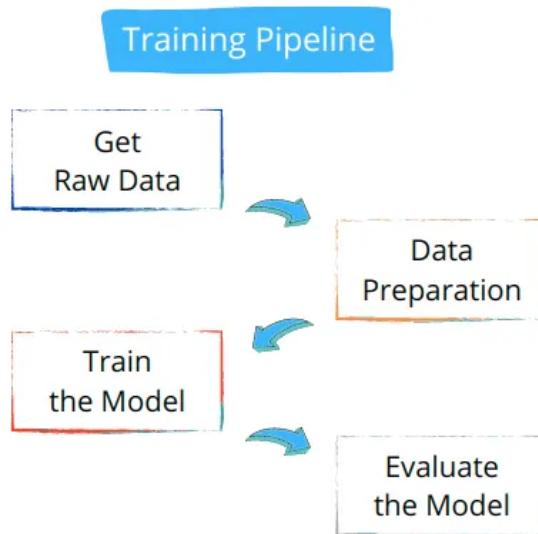
Cada etapa deste pipeline gera artefatos, como os dados processados, que são armazenados e reutilizados nos pipelines seguintes.

#### 5.5. PIPELINE DE TREINAMENTO

Neste pipeline, o modelo Random Forest é treinado utilizando os dados produzidos pelo pipeline de feature engineering. Ele é composto por:

- model\_trainer: treinamento do modelo com os dados de treino;
- model\_evaluator: avaliação do modelo com base em métricas como F1-score, com logging automático dos resultados no MLflow.

A Figura 5 representa um fluxo de treinamento.



**Figura 5:** Fluxo de uma pipeline de treinamento típica.

**Fonte:** Kattson Bastos, Medium (2022).

#### 5.6. PIPELINE DE INFERÊNCIA

Este pipeline executa a geração de previsões a partir dos dados de entrada. As etapas incluem:

- select\_best\_model: recuperação do melhor modelo salvo no MLflow com base na métrica de avaliação;
- inference\_predict: geração das previsões utilizando os dados já processados.

Essa arquitetura desacoplada permite atualizações frequentes do modelo, sem a necessidade de reconfigurar os dados ou reexecutar todo o pipeline.

## 5.7. INTEGRAÇÃO COM MLFLOW

O MLflow foi integrado à arquitetura com o propósito de oferecer rastreamento completo de experimentos. A ferramenta permitiu:

- Logging automático de métricas, parâmetros e artefatos;
- Versionamento dos modelos treinados;
- Comparação entre diferentes execuções de experimentos;
- Interface web para visualização e gerenciamento dos experimentos.

Essa abordagem integrada ilustra como o uso do ZenML em conjunto com o MLflow oferece uma base sólida para produtizar projetos de aprendizado de máquina com foco em confiabilidade, organização e escalabilidade.

## 6. CONCLUSÃO

Este trabalho demonstrou como o MLOps, apoiado pelas ferramentas ZenML e MLflow, facilita a produtização de modelos de aprendizado de máquina. A pesquisa partiu do reconhecimento de um problema recorrente: grande parte dos modelos desenvolvidos em projetos de *machine learning* não chega à etapa de produção, principalmente pela ausência de processos estruturados e infraestrutura adequada.

Dessa forma, o MLOps surge como um conjunto de práticas e ferramentas que visam preencher essa lacuna, adaptando os princípios do DevOps aos sistemas baseados em *machine learning*. O MLOps se propõe a tornar os processos de desenvolvimento, implantação e monitoramento mais eficientes, promovendo maior integração entre as áreas de ciência de dados, engenharia de dados e engenharia de software.

A aplicação em um caso real dos conceitos de MLOps evidenciou como o uso do ZenML e MLflow contribui para maior eficiência na implantação de modelos. Tais práticas resultam em processos mais confiáveis, escaláveis e sustentáveis, especialmente em contextos onde a robustez da infraestrutura é um fator crítico, como no setor financeiro.

Como contribuição, este trabalho oferece uma visão prática e atualizada sobre o uso de MLOps, demonstrando o uso de ferramentas modernas para resolver os desafios de um projeto de aprendizado de máquina. Acredita-se que os aprendizados aqui discutidos possam servir de base para outras aplicações em diferentes setores e fomentar a adoção de práticas modernas de MLOps nas organizações.

## 7. REFERÊNCIAS

**FAUBEL, L.; SCHMID, K.; EICHELBERGER, H.** Is MLOps different in Industry 4.0? General and Specific Challenges. In: Proceedings of the 3rd International Conference on Innovative Intelligent Industrial Production and Logistics, 2022, pp. 161–167.

**KREUZBERGER, Dominik; KÜHL, Niklas; HIRSCHL, Sebastian.** Machine Learning Operations (MLOps): Overview, Definition, and Architecture. IEEE Access, 2023.and Logistics, 2022, pp. 161–167.

**LIANG, Penghao et al.** Automating the Training and Deployment of Models in MLOps by Integrating Systems with Machine Learning, 2024.

**RUF, Philipp et al.** Demystifying MLOps and Presenting a Recipe for the Selection of Open-Source Tools. Applied Sciences, v. 11, n. 8861, p. 1–39, 2021.



**SUBRAMANYA, Rakshith; SIERLA, Seppo; VYATKIN, Valeriy.** From DevOps to MLOps: Overview and Application to Electricity Market Forecasting. *Applied Sciences*, v. 12, n. 9851, p. 1–25, 2022.

**WANG, X. et al.** End-to-End MLOps for Time-Series Applications, 2022.

**ZAHARIA, M. et al.** Accelerating the Machine Learning Lifecycle with MLflow. Databricks Inc., 2023.