

Mini-Projeto

Aplicação de Machine Learning em Problemas Bancários

Contexto

Ao longo do curso, até agora, vimos técnicas fundamentais de Machine Learning supervisionado, passando por estatística, análise exploratória, pré-processamento, modelos clássicos e métricas de avaliação.

Neste mini-projeto, o objetivo é conectar esses conceitos a um problema real do ambiente bancário, construindo uma prova de conceito (PoC) baseada em dados.

O foco não é apenas o modelo, mas a capacidade de identificar um gap real e propor uma solução viável com ML.

Organização

- Trabalho em **grupos de até 3 pessoas**
- Tempo estimado de trabalho: **~10 horas**
- Apresentação final: **20 minutos por grupo (15 apresentação + 5 arguição)**

Desafio

Cada grupo deverá:

Identificar um problema relevante do contexto bancário que possa ser endereçado com Machine Learning, explicitar o gap existente e apresentar uma solução baseada em dados.

O problema não precisa existir hoje exatamente da forma descrita, mas deve ser plausível, realista e aderente à operação de um banco.

Categorias de problema (escolher 1)

O problema escolhido deve se enquadrar em uma das categorias abaixo:

1. Risco

- Inadimplência
- Atraso de pagamento
- Probabilidade de default

2. Fraude

- Detecção de transações suspeitas
- Comportamento anômalo
- Redução de falsos positivos em alertas

3. Clientes

- Churn
- Propensão à contratação de produtos
- Segmentação de clientes

4. Operacional

- Classificação de demandas
- Previsão de volume
- Priorização de atendimentos ou processos

Caso o grupo queira explorar outra categoria, mais aderente com a sua área de trabalho ou pesquisa, o mesmo deve apresentar a ideia antecipadamente para validação.

Dados

Os dados podem ser:

- Públicos (ex.: Kaggle, UCI, datasets abertos);
- Sintéticos / adaptados, desde que façam sentido para o problema;
- Reais, desde que completamente anonimizado e com permissão do gestor para utilização em atividade acadêmica.

Requisito: O dataset deve ser tabular e permitir aprendizado supervisionado (classificação ou regressão).

Considere a plausibilidade na obtenção dos dados, caso opte-se por dados sintéticos.

Entrega

A apresentação deve ser clara, objetiva e focada na **tomada de decisão e no PoC**, não apenas no código.

1. O problema e o gap

- Qual é o problema?
- Qual decisão hoje é mal feita, lenta ou inexistente?
- Onde está o **gap** que justifica o uso de ML?

2. Formulação do problema em ML

- Classificação ou regressão?
- Qual é a variável alvo?
- O que o modelo está tentando prever?

3. Análise dos dados

- Breve análise exploratória
- Correlação (uni e multivariada)
- Missings e possíveis vieses

4. Preparação dos dados

- Normalização ou padronização
- Tratamento de desbalanceamento (se aplicável)
- Justificativa das escolhas

5. Modelagem

- Pelo menos **2 modelos** entre:
 - Regressão logística
 - Árvore de decisão
 - Random Forest

É permitido (mas não obrigatório) utilizar outras técnicas não vistas em aula, desde que:

- Sejam explicadas
- Façam sentido para o problema

6. Avaliação

- Métrica adequada ao problema (accuracy, recall, precision, AUC, etc.)
- Comparação entre modelos
- Trade-offs relevantes (ex.: falso positivo vs falso negativo)

7. Conclusão executiva

- O modelo resolve o gap identificado?
- Onde ele falha?
- O que seria necessário para levar isso para produção?
- Há potencial de geração de valor (ex.: redução de custo, aumento de eficiência, mitigação de risco)? Caso positivo, explore os resultados sob uma ótica qualitativa ou estimada, considerando impacto financeiro, ROI ou payback.

8. Riscos e limitações

- Riscos técnicos
- Riscos de viés
- Riscos regulatórios ou operacionais

9. Apresentação do PoC

Demonstração da solução concebida em funcionamento, preferencialmente como:

- Uma aplicação completa (ex.: Streamlit), e/ou
- Um notebook ou programa em Python executável via linha de comando.

Avaliação

A avaliação será realizada de forma comparativa e conceitual, posicionando o trabalho apresentado em relação aos demais trabalhos da turma.

Cada grupo receberá feedback escrito individualizado, incluindo sua colocação relativa e comentários qualitativos sobre pontos fortes e oportunidades de melhoria.

A avaliação vai considerar:

- Clareza na definição do problema
- Coerência entre problema, dados e modelo
- Qualidade da análise e das decisões técnicas
- Capacidade de comunicação e argumentação
- Maturidade na discussão de riscos

Observações finais

- O foco não é complexidade, mas consistência
- Não é esperado um modelo perfeito
- O importante é mostrar raciocínio de Data Science aplicado a um problema real
- Considere entregar a prova de conceito como um programa em Python ou mesmo uma aplicação construída com Streamlit, caso faça sentido para o gap que se deseja preencher.

Entrega Final e Apresentação

- Dia 27/2 em tempo de aula, sendo que a avaliação formal sobre a apresentação é entregue no dia 2/3.
- No momento da apresentação o grupo deve entregar arquivo compactado com a apresentação além de qualquer arquivo acessório (como código

Python, notebooks ou datasets) necessário para uma avaliação crítica do trabalho, desde que inexistam restrições na publicidade desse material.