Case Data Science

Análise e Modelagem de Classificação de Clientes de Alto Valor

João Guilherme Marcondes

28 de julho de 2025

Resumo

Este relatório detalha o desenvolvimento de um modelo de classificação de clientes para o Banco Inter. O objetivo principal é identificar perfis de clientes de alto valor e clientes padrão, otimizando a oferta de produtos e serviços. O documento cobre desde a análise exploratória dos dados e o pré-processamento até a experimentação com diferentes algoritmos de Machine Learning, culminando na definição e implementação do modelo mais adequado.

Sumário

1 Análise Exploratória de Dados (EDA)

A fase inicial do projeto consistiu em uma análise exploratória para entender a distribuição e as características dos dados. As principais observações foram:

- **Forte desbalanceamento do dataset:** O conjunto de dados apresentou um forte desbalanceamento entre as classes.
- **Perfil esperado de cliente de alto valor:** Pessoas de 30 a 60 anos, independentemente do sexo, que já atingiram uma maturidade profissional e financeira, mesmo que não possuam uma grande quantia investida.

2 Passo 1: Definição do Problema

O objetivo do projeto foi definido como a criação de um modelo de classificação que segmente os clientes em duas categorias distintas: "Alto valor"e "Padrão". O propósito de negócio é permitir a realização de ofertas diretas e personalizadas para os clientes com maior propensão a gastos com cartão de crédito.

3 Passo 2: Pré-processamento dos Dados

A preparação dos dados foi uma etapa crucial para garantir a performance e a estabilidade dos modelos. As seguintes ações foram executadas:

- Codificação de variáveis categóricas com *One-Hot Encoder*.
- Escalonamento de variáveis numéricas utilizando *StandardScaler*.
- Divisão do dataset em conjuntos de treino e teste na proporção de 80/20.
- Normalização geral das variáveis.

4 Passo 3: Primeiro Modelo - Regressão Logística

O primeiro modelo testado foi uma Regressão Logística, escolhida por sua simplicidade e interpretabilidade.

- **Resultado:** A acurácia obtida foi de aproximadamente 75%, um resultado ilusório.
- **Diagnóstico:** O modelo demonstrou um viés significativo, tendendo a classificar quase todas as instâncias como pertencentes à classe majoritária ("Padrão"), tornando-o ineficaz para o objetivo de negócio.

5 Passo 4: Segundo Modelo - Random Forest

Para buscar um desempenho superior, um modelo mais complexo, o *Random Forest*, foi implementado.

- **Resultado:** A acurácia caiu, mas o desempenho foi superior ao do modelo anterior, pois conseguiu classificar pelo menos um cliente de alto valor.
- **Feature Importance:** A análise de importância das features revelou que o **dinheiro investido** e a **idade** do cliente eram as variáveis mais preditivas.

Por apresentar desempenho superior, ainda que pequeno, optou-se por seguir com o modelo Random Forest.

6 Passo 5: Otimização do Modelo e Manipulação de Classes

Com o *Random Forest* como base, foram implementadas estratégias para refinar sua performance:

- **Ajuste de Pesos (Class Weight):** Aplicou-se a técnica de balanceamento de pesos para dar mais importância à classe minoritária.
- **Ajuste de Hiperparâmetros:** Utilizou-se *GridSearchCV* para uma busca exaustiva dos melhores hiperparâmetros.
- **Validação Cruzada:** Foi utilizada validação cruzada para melhorar o treinamento e evitar que o modelo seja ajustado apenas aos dados de treinamento.

Nenhuma das estratégias trouxe melhorias significativas. O desempenho permaneceu estável, indicando que o gargalo poderia estar nos dados de entrada.

7 Passo 6: Manipulação dos Dados de Entrada

Ao receber o dataset, identifiquei que o problema principal era o número limitado de dados disponíveis. Decidi então aplicar a técnica SMOTE para lidar com o desbalanceamento.

- **Resultado:** A acurácia diminuiu, mas o modelo se tornou mais eficaz.
- **Considerações:** Este foi o modelo que melhor classificou clientes de alto valor até o momento. A acurácia caiu porque o modelo reconheceu alguns clientes padrão como de alto valor, o que, para o case, é mais vantajoso do que deixar de identificar clientes de alto valor.
- **Decisão:** Como o desempenho ainda era insatisfatório, decidi testar modelos mais adequados ao problema.

8 Passo 7: Terceiro Modelo - Gradient Boosting (LightGBM)

Em busca de um modelo mais robusto, utilizei um algoritmo de *Gradient Boosting*, especificamente o *LightGBM*.

- **Resultado:** O desempenho melhorou significativamente. O modelo identificou 50% dos clientes de alto valor, mantendo um bom controle sobre os erros (falsos positivos). O recall subiu de 10% para 50%, e a precisão de 20% para 36%.

9 Passo 8: Otimização do Modelo

Apesar dos bons resultados com o *LightGBM*, era necessário melhorar o desempenho para atender às exigências de uma campanha de marketing. Foram testadas as seguintes técnicas:

- **Engenharia de features:** Criação de variáveis mais informativas.
- **Seleção de features:** Utilização do Random Forest para seleção antes do treinamento final.

As variáveis `razao_gasto_investimento` e `total_movimentacao` foram essenciais para o bom desempenho do modelo.

A criação dessas features levou o modelo a atingir 90% de recall e 100% de precisão. Entretanto, no contexto do projeto, um recall ainda maior era mais desejável, mesmo que à custa de um pouco da precisão.

Assim, testei:

- Otimização do GridSearch voltada para recall
- Uso de pesos de classe na busca de hiperparâmetros
- Ajuste do limiar de decisão priorizando recall

Mesmo com as alterações, o modelo permaneceu com 90% de recall e 100% de precisão — bons números para o projeto.

10 Passo 9: Revisão do Modelo

Na validação final, identifiquei que a engenharia de features causou vazamento de dados no treinamento, comprometendo a validade do modelo.

Para corrigir isso, retornei ao modelo anterior à engenharia e recomecei com mais cuidado.

11 Passo 10: Reestruturação e Otimização

Adicionei uma etapa de validação, além do treinamento e teste, e criei novas variáveis como proporção entre gastos e investimento e um score combinando idade e investimento, evitando vazamento de dados.

Os resultados melhoraram significativamente, embora tenha ocorrido overfitting. Ajustei o grid de hiperparâmetros, o que resolveu o problema e trouxe métricas sólidas:

- Precisão: 90%
- Recall: 82%
- F1-Score: 0,86
- AUC: entre 0,96 e 0,985

As métricas foram consistentes entre validação e teste, indicando boa generalização.

12 Passo 11: Resultados do Novo GridSearchCV e Avaliação Final

Após a reestruturação e para garantir a máxima performance, foi executado um novo e mais refinado processo de otimização de hiperparâmetros com GridSearchCV. Os resultados obtidos consolidam a robustez do modelo final.

- **Melhores Hiperparâmetros Encontrados:** A busca em grade identificou a seguinte combinação como ótima, com um score AUC de **0.9943** na validação cruzada:

```
{'colsample_bytree': 0.9, 'learning_rate': 0.1,
 'min_child_samples': 30, 'n_estimators': 100,
 'num_leaves': 7, 'reg_alpha': 0.01,
 'reg_lambda': 0.1, 'subsample': 0.8}
```

Avaliação nos Conjuntos de Validação e Teste

O modelo foi então avaliado nos conjuntos de dados separados para validar sua capacidade de generalização.

- **Desempenho na Validação:**
  - **Métricas:** Para a classe *alto_valor*, obteve-se precisão, recall e F1-score de **0.90**. A acurácia geral foi de **0.95**.
  - **AUC-ROC:** O valor foi de **0.9900**, confirmando a alta performance.
- **Desempenho Final no Teste:**
  - **Métricas:** No conjunto de teste, a performance para a classe *alto_valor* foi ainda mais sólida, com **precisão de 1.00**, **recall de 0.91** e **F1-score de 0.95**. A acurácia do modelo foi de **0.97**.
  - **AUC-ROC:** O AUC no teste foi de **0.9812**.

Diagnóstico de Overfitting

Uma análise final das métricas de AUC nos diferentes conjuntos de dados foi realizada para garantir que o modelo não estivesse sobreajustado.

- **AUC Treino:** 1.0000
- **AUC Validação:** 0.9900
- **AUC Teste:** 0.9812

A diferença absoluta entre o AUC de treino e validação foi de apenas **0.0100**. A performance se manteve alta e estável no conjunto de teste, indicando que o **modelo está bem generalizado** e não sofre de overfitting.

13 Conclusão

O projeto teve como objetivo principal desenvolver um modelo de classificação capaz de identificar clientes de alto valor para o Banco Inter, visando otimizar ações de marketing direcionadas. Diversas abordagens foram testadas, desde modelos simples, como Regressão Logística, até algoritmos mais sofisticados, como o LightGBM, com diferentes estratégias de pré-processamento, balanceamento de classes e engenharia de features.

O principal desafio enfrentado foi o desbalanceamento da base e a limitação de informações úteis. Após a correção de um vazamento de dados e múltiplos ciclos de otimização, o modelo final, baseado em LightGBM, apresentou resultados robustos e consistentes:

- **Precisão (teste):** 1.00
- **Recall (teste):** 0.91
- **F1-Score (teste):** 0.95
- **AUC (teste):** 0.9812

O diagnóstico final confirmou que o modelo possui excelente capacidade de generalização, sem sinais de overfitting.

Dessa forma, o modelo está apto a ser usado como ferramenta de apoio à tomada de decisão, permitindo ao banco direcionar campanhas promocionais com mais assertividade e retorno potencial.