Relatório de Aprendizado de Máquina:
Previsão de Inadimplência de Crédito

Mateus de Sena Reis El-Yachar
mateuselyachar@poli.ufrj.br
DRE: [121144292]

19 de julho de 2025

Sumário

1 Introdução

O presente trabalho tem como objetivo desenvolver um modelo de classificação para prever a probabilidade de inadimplência em solicitações de crédito, como parte da avaliação da disciplina de Introdução ao Aprendizado de Máquina. O desafio consiste em utilizar um conjunto de dados históricos de 20.000 solicitantes para treinar um modelo preditivo, cuja performance final é avaliada em um conjunto de teste de 5.000 amostras através da plataforma de competição Kaggle.

Este relatório descreve o processo metodológico completo, abrangendo: (1) a análise exploratória e o pré-processamento dos dados; (2) a engenharia de novas variáveis (feature engineering); (3) o treinamento e a avaliação comparativa de múltiplos algoritmos de classificação; e (4) a otimização iterativa de hiperparâmetros para maximizar o desempenho.

2 Análise e Preparação dos Dados

A preparação dos dados é uma etapa fundamental para o sucesso de qualquer modelo. As seguintes sub-etapas foram executadas:

2.1 Limpeza Inicial e Análise da Variável Alvo

O dataset inicial foi inspecionado, e colunas consideradas redundantes ou pouco informativas foram removidas. A análise da variável alvo, `inadimplente`, revelou um dataset perfeitamente balanceado, com 10.000 amostras para a classe '0' (bom pagador) e 10.000 para a classe '1' (inadimplente). Este cenário ideal permitiu o uso da acurácia como uma métrica de avaliação confiável.

2.2 Tratamento de Dados Faltantes

A estratégia de tratamento de valores nulos foi crucial, especialmente para as colunas `profissao_companheiro` (57.6% nulos) e `grau_instrucao_companheiro` (64.3% nulos). Ao invés de descartá-las, optou-se por imputar os valores nulos com `-1`. Esta abordagem cria uma nova categoria que representa a ausência de informação (ex: solicitante solteiro), preservando o potencial preditivo que essa condição pode ter. Para as demais colunas com nulos, foram utilizadas a moda (para variáveis categóricas) e a mediana (para numéricas).

2.3 Engenharia de Atributos

Para enriquecer o dataset e potencialmente melhorar o poder preditivo, foram criadas novas variáveis, como `renda_total`, `proporcao_renda_extra`, `numero_de_cartoes` e `faixa_etaria`. Esta etapa se mostrou fundamental para extrair sinais mais complexos dos dados originais.

2.4 Seleção de Atributos Baseada em Correlação

Uma matriz de correlação foi gerada para visualizar as relações lineares entre as variáveis numéricas (Figura 1). Essa análise guiou a remoção de variáveis com correlação muito baixa com a variável alvo, além de variáveis altamente correlacionadas entre si (multicolinearidade), como `local_onde_trabalha`, que era redundante com `local_onde_reside`. As features originais usadas na engenharia de atributos também foram removidas para evitar redundância.

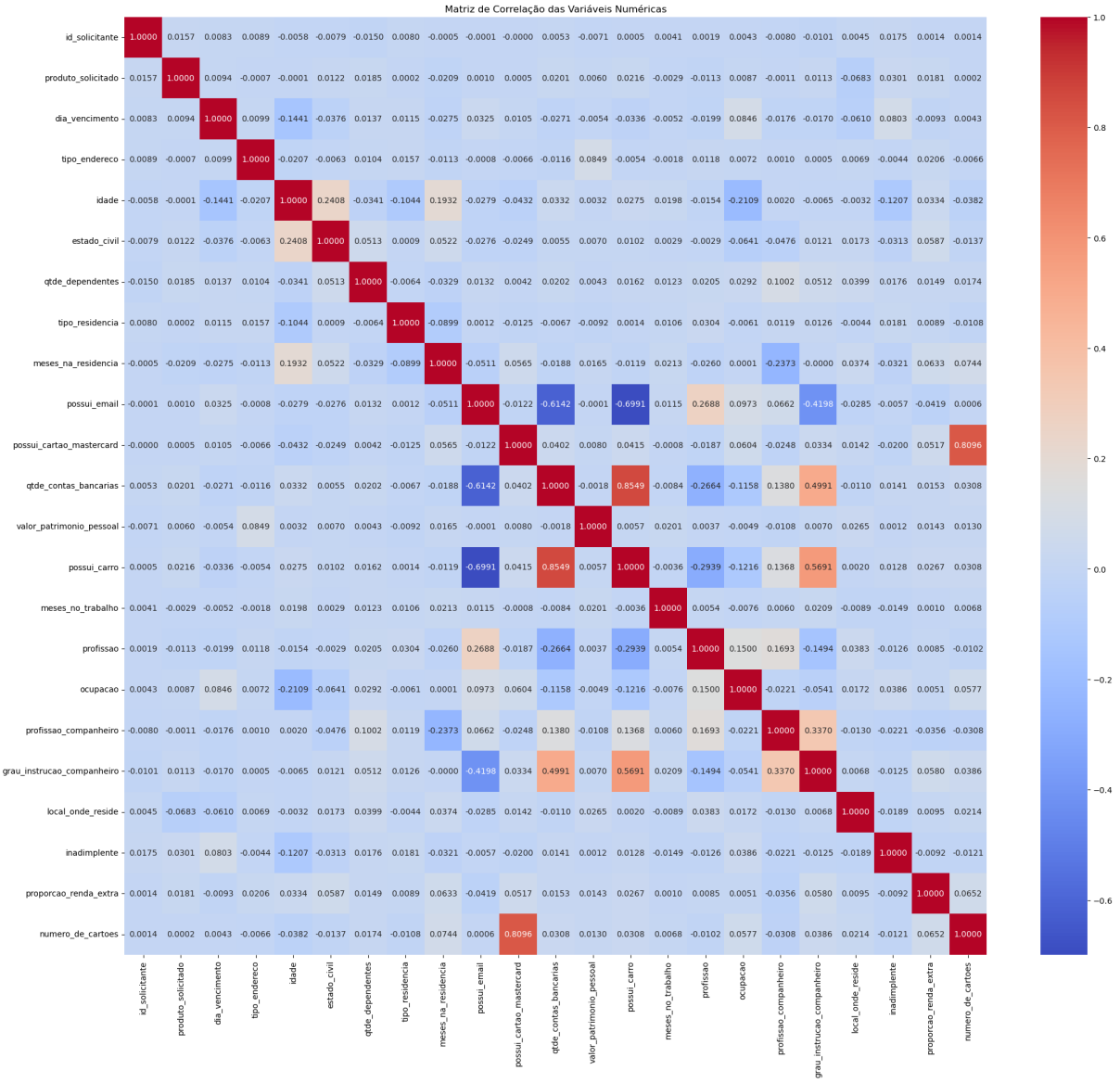

Figura 1: Matriz de correlação final utilizada para a seleção de atributos. A análise permitiu simplificar o modelo removendo ruído e redundância.

2.5 Codificação Final

Por fim, o processo de One-Hot Encoding foi aplicado para converter todas as features categóricas restantes em formato numérico, expandindo o dataset para 270 colunas e tornando-o pronto para a modelagem.

3 Modelagem e Otimização Iterativa

A modelagem seguiu uma abordagem iterativa, partindo de modelos simples e progredindo para otimizações mais complexas.

3.1 Modelos Baseline e Validação Cruzada

Inicialmente, foram treinados modelos de Regressão Logística e Random Forest com seus parâmetros padrão. Para obter uma medida de desempenho mais robusta, foi aplicada a Validação Cruzada com 5 folds. A Regressão Logística obteve uma acurácia média de **0.5914**, enquanto o Random Forest alcançou **0.5800**, estabelecendo o modelo linear como um forte baseline inicial.

3.2 Otimização com GridSearchCV

A primeira rodada de otimização foi realizada com `GridSearchCV`. Para o RandomForest, foram testadas diferentes combinações de hiperparâmetros. A robustez da validação foi aumentada de `cv=3` para `cv=10`, o que refinou a escolha dos melhores parâmetros e resultou em uma acurácia média de **0.5950**. Para a Regressão Logística, a busca encontrou uma acurácia de **0.5929**, confirmando que uma forte regularização (`C=0.01`) era benéfica.

3.3 Otimização Avançada com RandomizedSearchCV

Para explorar um espaço de hiperparâmetros ainda maior de forma eficiente, foi utilizado o `RandomizedSearchCV` no modelo RandomForest. Testando 50 combinações aleatórias de parâmetros em uma grade mais ampla e com `cv=5`, foi possível encontrar uma nova configuração que elevou a acurácia média para **0.5969**, superando o resultado do `GridSearchCV`.

4 Resultados Finais e Conclusão

4.1 Comparativo de Desempenho

A Tabela 1 consolida os resultados de todos os modelos experimentados, destacando a evolução da performance com a otimização.

Tabela 1: Tabela Comparativa Final de Desempenho dos Modelos

Modelo e Estratégia	**Acurácia (CV)**	**Comentários**
RandomForest (RandomizedSearch)	**0.5969**	**Modelo Campeão.** A busca aleatória em uma grade ampla encontrou a melhor combinação de hiperparâmetros.
RandomForest (GridSearch)	0.5950	Otimização inicial que já superou os modelos lineares.
Regressão Logística (Otimizada)	0.5929	Melhor modelo linear, demonstrou a importância da regularização.
LinearSVC (Otimizado)	0.5927	Performance similar à Regressão Logística.
SVM com Kernel (Padrão)	∼0.5783	Custo computacional elevado sem ganho de performance.
Gaussian Naive Bayes	∼0.5079	Inadequado para os dados devido à violação de suas premissas.

A Figura 2 apresenta a matriz de confusão do modelo campeão, detalhando sua performance na distinção entre as classes.

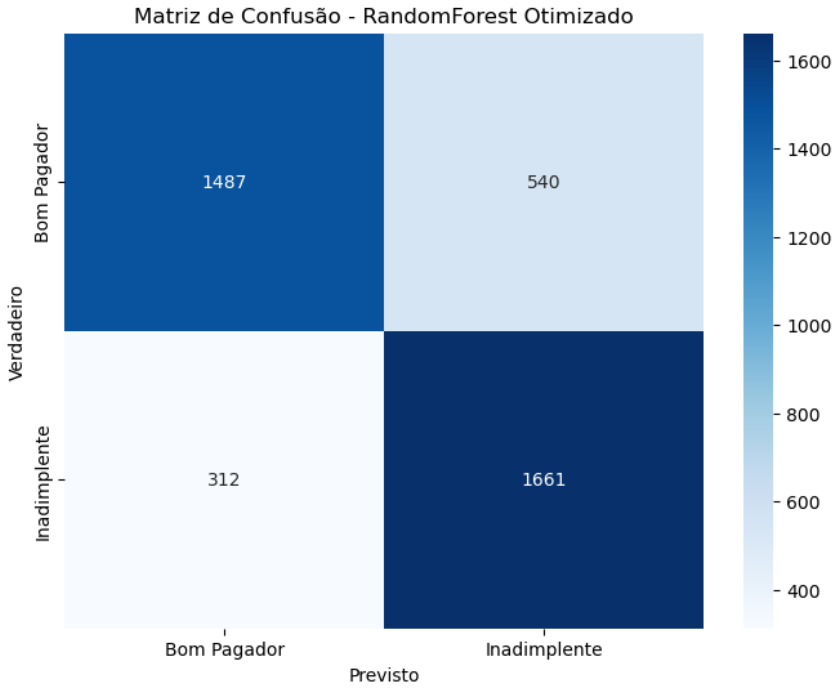

Figura 2: Matriz de Confusão do modelo RandomForest final, otimizado com RandomizedSearchCV.

4.2 Conclusão Final

O modelo **RandomForest otimizado via ‘RandomizedSearchCV‘** foi selecionado como o modelo final devido à sua performance superior na validação cruzada. Após ser treinado com o conjunto completo de dados de treino, ele foi utilizado para gerar as previsões para o conjunto de teste. A submissão na plataforma Kaggle alcançou um **score público de 0.5916**.

Este projeto demonstra a eficácia de um processo iterativo em aprendizado de máquina. A combinação de um pré-processamento cuidadoso, engenharia de atributos e, principalmente, a otimização sistemática de hiperparâmetros, foi fundamental para extrair o máximo de poder preditivo dos dados e alcançar um resultado competitivo.