

PROJETO CIENTISTA DE DADOS | MED.AI



MED AI - “Detecção de Fraudes em Planos de Saúde com Machine Learning”.

Aluna de Ciências de Dados Sarah Rodrigues Garcia

Estrutura do Projeto:

A entrega do projeto foi organizada em **três partes complementares**, que juntas apresentam tanto a fundamentação técnica quanto a comunicação estratégica dos resultados:

1. **Documento Técnico-Descritivo:**

Este documento apresenta uma visão geral das principais etapas do projeto: Coleta de Dados, Modelagem e Conclusões; Onde se é explicada a **criação e integração dos bancos de dados**, estruturação do **Data Lake**, **geração de dados sintéticos**, **seleção e parametrização dos modelos de Machine Learning**, além de outros aspectos relevantes da solução desenvolvida.


2. **Pipeline Técnica Detalhada (Markdown - GitHub):**

Contém toda a **pipeline de desenvolvimento do modelo de Machine Learning**, documentada em markdowns que descrevem, passo a passo, o processo de Coleta de Dados, Modelagem e Conclusões.

3. **Vídeo Narrado (Apresentação Executiva):**

Vídeo explicativo baseado em um **PowerPoint narrado por mim**, voltado para **stakeholders e tomadores de decisão**, com linguagem acessível e foco em demonstrar o funcionamento do modelo, suas etapas principais e o impacto da solução na identificação de fraudes.

Referências de Entrega:

 **Link da Pipeline (GitHub):** [\[link aqui\]](#)

 **Link do Vídeo Narrativo:** [\[link aqui\]](#)

Documento Técnico Descritivo - MED AI

Visão geral das principais etapas do projeto.

COLETA DE DADOS:

Contexto da Situação: O projeto tem como objetivo identificar **fraudes em transações médicas**. Essas fraudes estão distribuídas em **dois sistemas distintos**:

1. **Banco de Dados Financeiro do Plano de Saúde:** contém as informações de pagamentos, valores, datas, prestadores, métodos de repasse, etc.
2. **Banco de Dados Operacional Prestadores de Saúde:** contém os dados clínicos, autorizações, códigos de procedimentos, beneficiários e prestadores.

Para que o modelo de Machine Learning consiga **avaliar a transação como um todo**, é essencial unir os aspectos mais importantes desses dois conjuntos de dados em um novo Banco de Dados.

A. Etapas da Integração dos Bancos:

Breve explicação prática de como seria a integração dos dois bancos de dados necessários para a execução do MED AI.

1. Identificação da Chave de União

Primeiro, é necessário encontrar **chaves em comum** entre os dois bancos, como:

- id_transacao
- id_prestador
- id_beneficiario
- data_autorizacao OU data_pagamento

Essas chaves permitem realizar um **JOIN**, integrando as informações financeiras e clínicas necessárias para a detecção.

2. Criação de uma Camada de Dados Integrada (Data Lake ou Data Warehouse)

Após identificar as chaves, cria-se uma **camada intermediária**, chamada **camada integrada de dados** (ou *Data Lake / Data Warehouse*).

Ela pode ser atualizada em tempo real ou em intervalos curtos (por exemplo, a cada minuto ou segundo), dependendo da infraestrutura.

Essa camada faz:

- O **join automático** dos bancos de dados.
- A **limpeza e padronização dos dados** (por exemplo, formatos de datas, códigos de procedimentos).
- A **preparação dos dados de entrada** para o modelo de Machine Learning.

3. Processamento em Tempo Real

Para análises “ao vivo”, utiliza-se uma **pipeline de streaming de dados**, com tecnologias como:

- **Apache Kafka** ou **AWS Kinesis** → captura e transmite os eventos de transação em tempo real.

- **Spark Streaming** ou **Flink** → faz o *join* entre os bancos, limpa os dados e envia para o modelo.

Assim, quando uma nova transação é gerada:

1. Os dados financeiros e clínicos são capturados simultaneamente.
2. O sistema faz o *join* automático na camada de integração.
3. A transação é enviada ao **modelo de Machine Learning** já treinado.
4. O modelo classifica a transação como **fraudulenta ou legítima**, e retorna o resultado em segundos.

Todas essas etapas para a implementação do Modelo na vida Real.

Para o desenvolvimento deste projeto, foi realizada uma pesquisa por **bancos de dados operacionais e financeiros** que se aproximassem o máximo possível da realidade do setor de saúde. No entanto, devido às **restrições impostas pela Lei Geral de Proteção de Dados (LGPD)** e à **escassez de informações financeiras públicas sobre transações médicas**, optei pela **criação de um Banco de Dados Sintético**.

Esse banco foi estruturado de forma a **reproduzir de maneira fiel a dinâmica e a complexidade** dos dados que seriam integrados ao **Data Lake** em um ambiente real.

O Banco Sintético foi criado com as seguintes porcentagens:

n = 10000 # total de registros - 90% regular

fraude_pct = 0.10 - 10% de fraudes

phantom_pct = 0.0333 - 3 % de Phantom Billing

upcoding_pct = 0.0333 - 3% de Upcoding

duplicidade_pct = 0.0333 - 3% de Duplicidade de Pagamento

Aqui, teve-se de realizar um tipo de Engenharia reversa, visando causar essa injeção de fraudes dentro do banco de dados, essa injeção fraudulenta acontece pela indicação de outras variáveis como:

- Phantom Billing - Foi criado pela comparação de valor_pago > 0 e data_realizacao = NaT
- Upcoding - Foi criado a partir da comparação status_saude = -1 (saudável) + procedimentos_urgencia = 'sim' (Aqui também seria algo novo dentro das avaliações de saúde, a marcação do estado de saúde segundo o primeiro contato de atendimento médico)
- Duplicidade de cobrança - valor_pago muito alto.

Embora sintético, o banco foi construído de forma relacional, garantindo a coerência entre beneficiários, prestadores e autorizações, tal como ocorre em bases reais.

B. Tratamento do Banco de Dados:

Apesar do Banco de Dados ter sido criado sinteticamente, foi se feito o tratamento de valores nulos, padronização e normalização dos dados.

C. Definição de Limiares Fraudulentos:

Apesar de já serem conhecidos, justo pela a criação sintética, foi-se feito uma nova análise e nova abordagem em cima dos limiares fraudulentos já conhecidos para a marcação das targets dentro do Banco de Dados.

MODELAGEM:

(Adendo: Serão usados dois tipos de ML para a validação do algoritmo : XGBOOST & LIGTHGBM; Cada ML tem suas especificações, e cada um lida melhor com determinadas características de Bancos de Dados. Apesar de alguns não apresentarem tanta necessidade de padronização e balanceamento, aqui no projeto foram se feitas todas as etapas de tratamento e então a avaliação de qual banco tratado ou não tratado seria melhor para a escolha do melhor algoritmo para o MED AI).

A. Divisão de Treino e Teste:

A modelagem se inicia com a separação de Base de Treino e Teste.

Base de Treino: 70% da Base Inicial.

Base de Teste: 30% da Base Inicial.

B. Balanceamento da Variável Target na Base de Treino:

A Base Inicial possui valores altamente desbalanceados, 90% esta com valores regulares e somente 10% com valores irregulares.

Ao observar a Base de Treino, foi-se encontrado as seguintes proporções:

Para a criação da base de treinamento foi utilizado a técnica de SMOTE.

O **SMOTE** é uma técnica de **oversampling inteligente**. Ao invés de simplesmente duplicar os registros minoritários (o que causaria overfitting), ele **gera novos exemplos sintéticos** com base na **proximidade entre as amostras da classe minoritária**.

O uso do SMOTE permitiu avaliar o impacto do balanceamento artificial, comprovando que o XGBoost foi capaz de generalizar bem mesmo em bases naturalmente desbalanceadas

C. Escolha do ML:

Objetivo do projeto é a criação de um algoritmo rápido de detecção de Fraude; Os dois MLs para o processamento computacional precisam apresentar alta assertividade, velocidade e lidar eficientemente com Bancos de Dados de relação não Linear com a Variável Target. Tendo em mente isso, a escolha foi: XGBOOST e LIGTHGBM.

É importante a escolha de dois MLs para se realizar a comparação de resultados afim de escolher aquele que vá apresentar melhores resultados ao objetivo desejado.

D. Treinamento dos Modelos:

Uso de RandomSearch para a Busca dos melhores parâmetros dentre dos algoritmos.

É importante ressaltar que nesse ponto temos 3 tipos de Bancos de Dados para treinamento:

X, y - 70 % Base de Treino Bruta, somente com normalização e padronização dos Dados.

x_treino_balanceado.csv, y_treino_balanceado.csv - 70 % Base de Treino com padronização, normalização dos Dados e Balanceamento.

As Bases foram treinadas pelos dois algoritmos, XGBOOST e LIGTHGBM em quatro testes:

1) XGBOOST - Uso de DataFrame não Balanceado.

2) XGBOOST - Uso de DataFrame Balanceado.

3) LigthGBM - Uso de DataFrame não Balanceado.

4) LigthGBM - Uso de DataFrame Balanceado.

E. Teste dos Modelos:

Realização das Previsões com as Bases de Teste 30% do banco_dados_sintetico_operadora.csv

CONCLUSÕES:

O projeto **MedTrustAI** teve como objetivo principal a detecção automática de fraudes médicas sintéticas, abordando três tipos principais: *Phantom Billing*, *Upcoding* e *Duplicidade de Pagamento*. Após a aplicação e comparação de diferentes algoritmos de *Machine Learning*, o **XGBoost** destacou-se como o modelo com **melhor desempenho geral**, apresentando a **maior média macro (Macro Avg)** entre todas as abordagens testadas, mesmo **sem a aplicação de técnicas de balanceamento**.

Esse resultado evidencia a **capacidade do XGBoost de generalizar padrões complexos** de comportamento fraudulento a partir dos próprios dados, demonstrando **robustez e estabilidade** nas previsões. Além disso, sua estrutura baseada em *gradient boosting* mostrou-se eficiente na distinção entre registros regulares e fraudulentos, alcançando métricas equilibradas mesmo em classes minoritárias.

Para a interpretação dos resultados, foi empregada a técnica **SHAP (SHapley Additive Explanations)**, que permitiu compreender de forma transparente **quais variáveis mais influenciaram as decisões do modelo**. Por meio dos gráficos *Summary Plot*, *Waterfall* e *Dependence*, foi possível identificar os **atributos com maior impacto na classificação**, oferecendo uma visão detalhada dos fatores que contribuíram para a detecção de cada tipo de fraude.

A integração do **XGBoost com a análise SHAP** tornou o modelo não apenas preciso, mas também **explicável e auditável**, característica essencial em contextos médicos e regulatórios. Assim, o projeto alcançou um equilíbrio sólido entre **performance técnica e interpretabilidade**, consolidando o XGBoost como a melhor escolha para a detecção de fraudes médicas no contexto do **MedTrustAI**.

O MedTrustAI demonstrou que é possível aplicar técnicas de inteligência artificial de forma explicável e eficiente para o combate à fraude em saúde, servindo como base para futuras implementações em ambientes produtivos.