Módulo 2: Estudo dos algoritmos LIME e SHAP com dados tabulares

Introdução

Este relatório documenta as atividades, descobertas e direcionamentos estratégicos da segunda sprint e modulo do projeto. Esta fase foi concebida como um período de imersão profunda, crucial para estabelecer o alicerce teórico e prático sobre o qual todo o trabalho experimental subsequente será construído. O foco residiu em três pilares: aprofundar o conhecimento sobre as técnicas de explicabilidade (XAI), construir uma bancada de testes de software funcional e mapear os desafios inerentes à preparação do complexo dataset médico MIMIC-IV-ED. O progresso alcançado aqui não apenas mitigou riscos técnicos significativos, mas também forneceu um roteiro claro para as próximas etapas de modelagem e análise.

Progresso

1. Missão e Objetivos Estratégicos

Esta sprint foi definida não por meras tarefas, mas por missões estratégicas para direcionar a pesquisa e garantir sua viabilidade:

- **Estabelecer o Alicerce Teórico em XAI:** Ir além de uma compreensão superficial de LIME e SHAP, aprofundando nos seus fundamentos matemáticos, pressupostos e limitações. O objetivo era formar uma base de conhecimento crítica para poder, posteriormente, analisar e justificar as diferenças nos resultados de explicabilidade de forma rigorosa.
- **Construir uma Bancada de Testes Funcional:** Desenvolver um esqueleto de código (*scaffold*) que não apenas valide a funcionalidade das bibliotecas, mas que sirva como um ambiente de testes robusto (*test bench*). Este ambiente permitiria iterar rapidamente sobre o pipeline de dados, a modelagem e a integração das ferramentas de XAI, validando a viabilidade da abordagem de ponta a ponta.
- **Mapear o Território dos Dados (MIMIC-IV-ED):** Realizar uma análise exploratória focada nos desafios práticos de se trabalhar com dados médicos do mundo real. A missão era transformar a complexidade de múltiplas tabelas díspares em um plano de ação concreto para unificação, sanitização e engenharia de características, preparando o terreno para a modelagem.

2. Execução das Atividades e Descobertas

2.1. Imersão nos Algoritmos de Explicabilidade: LIME vs. SHAP

A primeira fase da sprint foi dedicada a um estudo comparativo aprofundado das duas técnicas de XAI selecionadas.

- **Análise Teórica e Implicações:** A revisão dos artigos seminais de Ribeiro, Singh & Guestrin (2016) e Lundberg & Lee (2017) revelou uma distinção filosófica fundamental.
    - **LIME:** Foi compreendido como uma heurística brilhante e intuitiva. Sua abordagem de perturbar uma única instância para criar uma "vizinhança" e, em seguida, ajustar um modelo linear simples para explicar o comportamento local do modelo "caixa-preta" é poderosa em sua simplicidade. No entanto, o estudo revelou sua principal vulnerabilidade: a instabilidade. A definição da "vizinhança" (através do tamanho do kernel e do número de perturbações) é um hiperparâmetro crítico e, por vezes, arbitrário, o que pode levar a explicações que variam entre execuções.
    - **SHAP:** Em contraste, o SHAP foi identificado como uma abordagem com rigor matemático. Sua base na teoria dos jogos cooperativos e nos valores de Shapley garante propriedades teóricas desejáveis, como a consistência (uma característica que se torna mais importante no modelo nunca terá sua importância na explicação diminuída) e a aditividade (a soma das contribuições das características iguala a saída do modelo). Esta robustez teórica posiciona o SHAP como uma ferramenta potencialmente mais confiável para cenários de alto risco, como o diagnóstico médico, onde a confiabilidade da explicação é primordial.
- **Análise Prática e Experiência com as Bibliotecas:** A teoria foi complementada com a exploração prática. Foram executados notebooks de exemplo das bibliotecas lime e shap. A experiência revelou que, enquanto LIME gera um gráfico de barras simples e de fácil interpretação imediata, SHAP oferece um arsenal de visualizações mais rico (force plots, summary plots, dependence plots), permitindo uma análise muito mais profunda, tanto local quanto global. Foi investigada a diferença entre KernelExplainer (lento, mas verdadeiramente modelo-agnóstico) e TreeExplainer (altamente otimizado e rápido para modelos baseados em árvores), uma distinção crucial para o planejamento da fase de modelagem.

2.2. Desenvolvimento da Bancada de Testes

Foi estruturado um notebook Jupyter modular, projetado para facilitar a experimentação.

- **Arquitetura do Código:** O esqueleto do código foi organizado em funções distintas para (1) carregamento e junção de dados, (2) pré-processamento e

engenharia de características, (3) treinamento e avaliação do modelo, e (4) geração de explicações. Esta modularidade permitirá testar e modificar cada componente de forma independente.
- **Escolha do Modelo Base:** O RandomForestClassifier de scikit-learn foi deliberadamente escolhido como o modelo "caixa-preta" inicial por dois motivos estratégicos: primeiro, é um modelo robusto e de alta performance para dados tabulares; segundo, por ser baseado em árvores, ele permite o uso do TreeExplainer do SHAP, que é computacionalmente muito mais eficiente do que o KernelExplainer, acelerando significativamente o ciclo de desenvolvimento e análise.

2.3. Análise e Estruturação de Dados do MIMIC-IV-ED

Esta foi a atividade mais intensiva da sprint. A exploração inicial dos dados brutos revelou a complexidade inerente aos registros de saúde do mundo real e a necessidade de um plano de ação meticuloso.

- **Desafio da Unificação - Construindo a Matriz de Características:** A análise confirmou que os dados são altamente relacionados e distribuídos. edstays.csv foi confirmado como a espinha dorsal do projeto. A estratégia de unificação foi solidificada: iniciar com a tabela edstays e executar uma série de left joins com as tabelas triage e diagnosis usando a chave stay_id. A escolha do left join é crítica para garantir que nenhuma visita à emergência seja perdida, mesmo que faltem dados em outras tabelas. O objetivo final é materializar uma única e ampla **matriz de características**, onde cada linha representa uma visita única e as colunas representam todas as informações estáticas e demográficas relevantes para essa visita.
- **Desafio da Agregação - "Achatando" os Dados Temporais:** A tabela vitalsign apresentou o desafio mais complexo devido à sua natureza longitudinal (múltiplas medições por visita). A solução delineada foi um processo de engenharia de características para "achatar" (flatten) essa dimensão temporal. Para cada sinal vital (frequência cardíaca, pressão arterial, etc.) e para cada stay_id, serão calculadas múltiplas estatísticas agregadas:
    - **Média e Mediana:** Para capturar a tendência central do sinal vital durante a estadia.
    - **Desvio Padrão:** Para capturar a variabilidade ou instabilidade do paciente, uma característica potencialmente muito preditiva.
    - **Mínimo e Máximo:** Para capturar eventos extremos (picos ou quedas) que podem ter ocorrido. Este processo transforma uma rica série temporal em um conjunto de características estáticas informativas, prontas para serem usadas por modelos de MIL clássicos.

- **Desafio da Sanitização - O Dilema dos Dados Clínicos:**
    - **Valores Ausentes:** Foi identificado um dilema crucial: um valor ausente significa que a medição não foi feita ou que o valor era normal e, portanto, não foi registrado? A estratégia inicial será a imputação pela mediana (que é mais robusta a outliers do que a média), mas com a consciência de que uma análise de sensibilidade a diferentes estratégias de imputação pode ser necessária no futuro.
    - **Variáveis Categóricas:** A conversão de variáveis como gender e race via One-Hot Encoding é direta. No entanto, para variáveis com muitas categorias (como chiefcomplaint), essa abordagem pode levar a uma explosão de dimensionalidade. A decisão foi iniciar com as variáveis categóricas de baixa cardinalidade e tratar as de alta cardinalidade posteriormente, talvez com técnicas de embedding ou agrupamento.
    - **Variável Alvo:** A coluna disposition foi selecionada para criar a variável alvo. Foi definido um problema de classificação binária: prever se um paciente será admitido no hospital (disposition == 'ADMITTED'). Esta é uma questão clínica de alto valor (gestão de leitos, alocação de recursos) e espera-se que seja um alvo razoavelmente balanceado para iniciar a modelagem.

3. Resultados e Direcionamentos Estratégicos

Esta sprint de imersão não produziu apenas código e análise, mas um roteiro estratégico claro, informando diretamente as próximas etapas críticas do projeto.

1. **Finalizar o Pipeline de Pré-processamento como um Ativo Reutilizável:** A prioridade máxima é implementar o script de unificação, agregação e sanitização de forma modular e bem documentada. O objetivo é criar uma função que transforme os arquivos CSV brutos em um único DataFrame limpo com um único comando, tornando o processo de preparação de dados reproduzível e eficiente.
2. **Executar uma Análise Exploratória de Dados (EDA) Guiada por Hipóteses:** Com o dataset unificado, a EDA não será apenas descritiva. Ela será usada para validar hipóteses geradas nesta sprint, como: "A variabilidade (desvio padrão) da frequência cardíaca é mais preditiva do que a média?" ou "O nível de acuidade na triagem (acuity) está entre as 5 características mais correlacionadas com a admissão?".
3. **Refinar a Definição da Tarefa de Predição:** Antes do treinamento, é crucial analisar o balanceamento de classes da variável alvo (ADMITTED vs. não admitido) e planejar o uso de técnicas de amostragem (ex: SMOTE), se necessário. Mais importante, será necessário realizar uma verificação cuidadosa

de vazamento de dados (*data leakage*), garantindo que nenhuma característica utilizada para treinar o modelo contenha informações sobre o futuro que não estariam disponíveis no momento da predição real.

4. **Estabelecer uma Linha de Base (Baseline) Robusta:** O treinamento da primeira versão do RandomForestClassifier não tem como objetivo alcançar a performance máxima, mas sim estabelecer uma linha de base de desempenho sólida e bem avaliada (com Acurácia, Precisão, Recall, F1-Score, AUC-ROC). Este baseline será o ponto de referência contra o qual todas as futuras melhorias no modelo e na engenharia de características serão rigorosamente mensuradas.