Projeto 5: Investigando Diabetes em Mulheres Indígenas Pima

Contextualização do Dataset

O conjunto de dados **"Pima Indians Diabetes"** é um clássico na área de ciência de dados e saúde, originalmente coletado pelo Instituto Nacional de Diabetes e Doenças Digestivas e Renais dos EUA. O objetivo é prever o diagnóstico de diabetes em mulheres indígenas Pima com base em variáveis clínicas e demográficas.

Características Principais:

- **Variáveis Preditoras**:
    - Pregnancies: Número de gestações.
    - Glucose: Concentração de glicose no plasma.
    - BloodPressure: Pressão arterial diastólica (mm Hg).
    - SkinThickness: Espessura da dobra cutânea do tríceps (mm).
    - Insulin: Nível de insulina sérica (mu U/ml).
    - BMI: Índice de massa corporal (peso em kg / altura em m^2).
    - DiabetesPedigreeFunction: Escore genético associado ao histórico familiar de diabetes.
    - Age: Idade da paciente.
- **Variável Alvo**:
    - Outcome: Diagnóstico de diabetes (0 = negativo, 1 = positivo).

Desafio Implícito:

- O dataset contém **valores ausentes mascarados como zeros** (ex: Glucose = 0), o que inviabiliza análises precisas. Além disso, é necessário integrar múltiplas etapas de processamento em um fluxo reprodutível.

Situação Problema

Você foi contratado(a) por uma equipe de saúde pública para preparar um dataset confiável que será usado em modelos preditivos de diabetes. O dataset original foi criticado por conter inconsistências, como valores clínicos impossíveis (ex: pressão arterial zero) e dados ausentes não tratados.

Objetivos:

1. **Importar e Integrar Dados**:
    - Carregar o dataset diabetes.csv e um arquivo Excel (descricao_funcoes.xlsx) com descrições das funções de limpeza.
2. **Tratar Dados Ausentes**:
    - Identificar valores zero que representam ausência de informação e convertê-los para NaN.
    - Decidir estratégias para imputação ou remoção de dados ausentes.
3. **Garantir Reprodutibilidade**:
    - Particionar o projeto em etapas, salvando resultados intermediários (ex: dataset_parte2.csv).
4. **Realizar Análise Exploratória (EDA)**:
    - Identificar padrões e outliers que possam afetar modelos futuros.

Passos para a Solução

Passo 1: Importação e Inspeção Inicial

1. Carregue o dataset diabetes.csv usando pandas, definindo valores ausentes como NaN onde zeros são inválidos (ex: Glucose, BloodPressure).
2. Leia o arquivo descricao_funcoes.xlsx para entender as funções de limpeza disponíveis no módulo limpeza_dados.py.

Passo 2: Tratamento de Dados Ausentes

1. **Identificação**:
    - Use dataset.apply(lambda x: (x == 0).sum()) para detectar zeros em colunas numéricas.
2. **Substituição**:
    - Converta zeros inválidos (ex: Glucose = 0) para NaN usando np.nan.
3. **Análise de Impacto**:
    - Calcule a porcentagem de dados ausentes por coluna com relatorio_valores_ausentes_por_coluna(dataset).
    - Decida se usará imputação (ex: mediana) ou remoção de linhas.

Passo 3: Particionamento do Projeto

1. Salve o dataset após cada etapa (ex: dataset_parte2.csv) para garantir continuidade.
2. Documente decisões técnicas (ex: por que escolheu imputar SkinThickness com a mediana?).

Passo 4: Análise Exploratória (EDA)

1. **Estatísticas Descritivas**:
    - Use dataset.describe() para identificar médias, desvios padrão e outliers.
2. **Visualizações**:
    - Plote histogramas para distribuição de variáveis como BMI e Age.
    - Use boxplots para detectar outliers em Insulin ou DiabetesPedigreeFunction.
3. **Correlações**:
    - Calcule a matriz de correlação para entender relações entre variáveis e o diagnóstico de diabetes.

Passo 5: Exportação Final

- Salve o dataset processado em formato CSV e compartilhe um relatório resumindo as alterações e insights.

Entregáveis Esperados

1. **Notebook (.ipynb)** organizado em partes, contendo:
    - Código para tratamento de dados ausentes e justificativas.
    - Visualizações e análise crítica dos resultados.
    - Links entre as partes (ex: carregar dataset_parte2.csv na Parte 3).
2. **Dataset Processado** em CSV, pronto para modelagem.

Dica: Utilize funções do módulo limpeza_dados.py para automatizar tarefas repetitivas. Documente cada etapa para facilitar a revisão pela equipe de saúde!

Bom trabalho! 🩺📈