Centro Universitário de Excelência
Sistemas de Informação

Pré-processamento de dados

Autor: Anna Beatriz S. Lima
Ian Salomão S. Carneiro
Valéria S. Santos

Feira de Santana, 2025

Agenda

Apresentar a implementação de técnicas de pré-processamento de dados em Python, sem bibliotecas externas.

1. Introdução:
A importância do Pré-Processamento na Mineração de Dados.

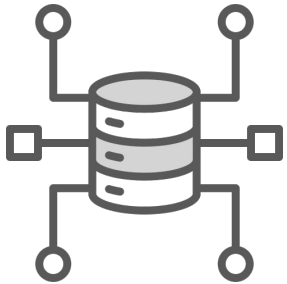
2. Implementação:
Classe Preprocessing e suas classes especializadas (*MissingValueProcessor, Scaler, Encoder*).

3. Técnicas Implementadas:
Tratamento de Ausentes, Transformação de Escalas e Codificação Categórica.

4. Resultados:
Demonstração da aplicabilidade e testes unitários.

5. Considerações Finais:
Demonstração da aplicabilidade e testes unitários.

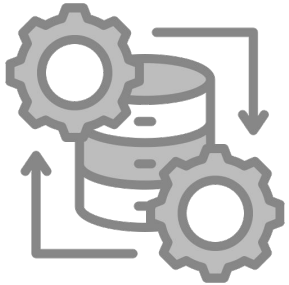
Introdução: A Importância do Pré-Processamento



Dados Brutos possuem:

- Inconsistências
- Erros
- Informações irrelevantes

Causando distorções na análise e insights falsos.



O Pré-Processamento:

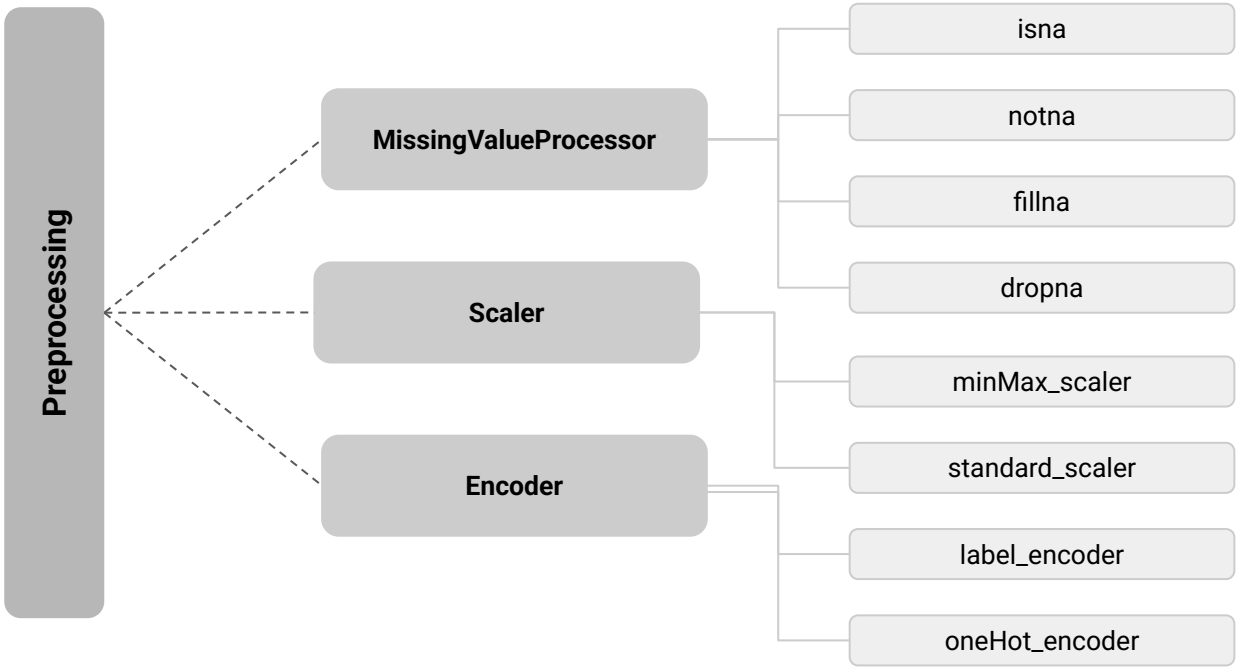
- Prepara
- Organiza
- Estrutura os dados brutos

Garantindo consistência dos dados

Um pré-processamento ineficiente poderia levar o sistema de recomendação do FoodDelivery a gerar previsões incorretas de preferências do usuário.

Pré-processamento de dados

Implementação



Pré-processamento de dados

Tratamento de Valores Ausentes

dropna()

Remove as linhas que contêm valores nulos (None) nas colunas especificadas.

```
CLASSE TratamentoDeValoresAusentes

    MÉTODO remover_linhas_ausentes(dataset)

        PARA CADA linha EM dataset:

            SE algum valor na linha ESTIVER AUSENTE ENTÃO

                REMOVER linha
```

Escalonamento de dados

minMax_scaler()

Transforma todos os valores de uma coluna para o intervalo [0,1]

```
função normalizar(valor, valor_min, valor_max):
   retorno (valor - valor_min) / (valor_max - valor_min)
```

Escalonamento de dados

```python
def minMax_scaler(self, columns: Set[str] = None):
    targetColumns = self._get_target_columns(columns)

    for col in targetColumns:
        data = self.dataset[col]

        valid_values = [x for x in data if x is not None]
        if valid_values:
            min_value = min(valid_values)
            max_value = max(valid_values)

            if max_value == min_value:
                self.dataset[col] = [0.0 if x is not None else None for x in data]
            else:
                self.dataset[col] = [(x - min_value) / (max_value - min_value) if x is not None else None for x in data]
```

Codificação de variáveis categóricas

label_encode()

Converte cada categoria em uma coluna em um número inteiro.

```
FUNÇÃO LABEL_ENCODE(COLUNA_CATEGORICA):
    CATEGORIAS_UNICAS, COLUNA_CODIFICADA = []
    MAPA_DE_ROTULOS = {}
    ROULO_ATUAL = 0

    PARA CADA VALOR em COLUNA_CATEGORICA:
        SE VALOR NÃO ESTÁ em CATEGORIAS_UNICAS:
            ADICIONAR VALOR a CATEGORIAS_UNICAS

    PARA CADA CATEGORIA em CATEGORIAS_UNICAS:
        MAPA_DE_ROTULOS[CATEGORIA] = ROULO_ATUAL
        ROULO_ATUAL = ROULO_ATUAL + 1

    PARA CADA VALOR_ORIGINAL em COLUNA_CATEGORICA:
        ROTULO = MAPA_DE_ROTULOS[VALOR_ORIGINAL]
        ADICIONAR ROTULO a COLUNA_CODIFICADA

    RETORNA COLUNA_CODIFICADA
```

Resultados

Todos os testes, incluindo casos de borda e extremos, tiveram resultados consistentes.

Considerações finais

O que pode ser melhorado

- Codificação de variáveis categóricas: Dependência do One-Hot Encoding pode causar explosão de dimensionalidade em datasets de alta cardinalidade.

Técnicas mais complexas

- Classe Scaler: Conceitos de normalização e padronização (MinMax e Z-Score) são mais difíceis de assimilar.
- Tratamento de valores vazios na Statistics: Foi necessário ajustar a lógica para lidar com listas vazias nos cálculos de métricas centrais.

Referências

- FreeImages. (s.d.) Ícone Visa. Disponível em: https://www.freeimages.com/illustrations/icon/visa. Acesso em: 25 set. 2025.
- Flaticon. (s.d.) Bug Icon. Disponível em: https://www.flaticon.com/free-icon/bug_1034634 flaticon.com
- Acesso em: 25 set. 2025.
- Reichert Junior, I. (2023) “Pré-processamento de Dados: o que é, por que fazer?”. Medium. Disponível em: https://medium.com/@ingoreichertjr/pr%C3%A9-processamento-de-dados-o-que-%C3%A9-por-que-fazer-dfc9fa3df8a3. Acesso em: 17 set. 2025.
- Gomes, P. C. T. (2019) “Pré-processamento de Dados”. DataGeeks. Disponível em: https://www.datageeks.com.br/pre-processamento-de-dados/. Acesso em: 17 set. 2025.