Universidade Federal de Pernambuco
Inteligência Artificial

Matheus Miranda Cabral de Menezes

ENCODING E SCALING

Segunda Atividade de Ciência de Dados (CIN0208)
Profa. Juscimara Gomes

Dataset Escolhido: Adult
Link do Notebook Jupyter Python

Recife - PE
2025

1. INTRODUÇÃO

Objetivo da Atividade: O objetivo deste trabalho é explorar e analisar o impacto de diferentes técnicas de codificação de variáveis categóricas (Encoding) e de normalização de features numéricas (Scaling) no desempenho do algoritmo K-Nearest Neighbors (KNN).

Características do Dataset: O dataset utilizado foi o "Adult", obtido da plataforma OpenML. Este dataset contém dados demográficos de censos, com o objetivo de prever se um indivíduo tem uma renda anual superior ou inferior a 50 mil dólares. O dataset processado sofreu limpeza de dados ausentes.

Tipo de Problema: O problema consiste em **classificação** binária. Isso é confirmado pela implementação, que utiliza um KNeighborsClassifier, e pela avaliação com base nas métricas de Acurácia e F1-Score, adequadas para classificação. A variável-alvo “class” foi codificada usando LabelEncoder.

2. ENCODING E SCALING

2.1 Metodologia e Implementação

O processo experimental foi conduzido usando bibliotecas Python, principalmente scikit-learn, pandas e category_encoders.

- **Preparação dos Dados:**
  - O dataset "Adult" foi carregado via fetch_openml.
  - Foi realizado um tratamento de valores ausentes: entradas marcadas como "?" foram substituídas por np.nan e, em seguida, todas as linhas com valores nulos foram removidas (dropna()).
  - As features foram separadas automaticamente em numeric_features e categorical_features com base no dtype das colunas.
- **Divisão dos Dados:**
  - Os dados foram divididos em 70% para treino e 30% para teste, conforme especificado na atividade.
  - Foi utilizado “stratify = y” para garantir que a proporção das classes de renda fosse mantida em ambos os conjuntos.
- **Pipeline de Experimentos:**
  - Um pipeline da scikit-learn foi utilizado para encadear o pré-processamento e o classificador.
  - O ColumnTransformer foi o componente-chave, aplicando os scalers apenas às numeric_features e os encoders apenas às categorical_features.
  - Implementação dos Encoders:
    - One-Hot;
    - Dummy;
    - Effect.

- Implementação dos Scalers:
  - As técnicas StandardScaler, MinMaxScaler, MaxAbsScaler, RobustScaler, e QuantileTransformer, essa última com distribuição uniforme e normal, foram importadas e usadas, conforme solicitado.

  - O cenário "No Scaling" foi implementado passando a string "passthrough" para o ColumnTransformer.

- **Modelo:** O KNeighborsClassifier foi instanciado com o valor padrão "n_neighbors = 5", como exigido pela atividade.

- **Avaliação:**
  - Conforme a tarefa é de classificação, as métricas accuracy_score e f1_score (com average = "weighted") foram calculadas para cada combinação.

  - As métricas MAE e RMSE, no entanto, são aplicadas exclusivamente a problemas de regressão. Elas não são aplicáveis a esta tarefa de classificação, pois não é possível medir um "erro numérico" entre uma previsão de categoria e um rótulo real. Por isso, foram ignoradas.

➢ **Análise e Visualização:**
- Os resultados foram armazenados em um DataFrame.

- As funções de análise e plotagem foram usadas para gerar as tabelas de pivot e os gráficos de barras comparativos.

2.2 Tabelas de Resultados

Os resultados detalhados de Acurácia e F1-Score para cada combinação de encoder e scaler estão apresentados abaixo:

Tabela de Resultados: Acurácia

Encoder	Max-Abs	Min-Max	No Scaling	Quantile	Quantile-Normal	Robust	Standard
Dummy	0.82	0.82	0.76	0.82	0.83	0.85	0.82
Effect	0.82	0.82	0.76	0.82	0.82	0.85	0.82
One-Hot	0.82	0.82	0.76	0.82	0.83	0.85	0.82

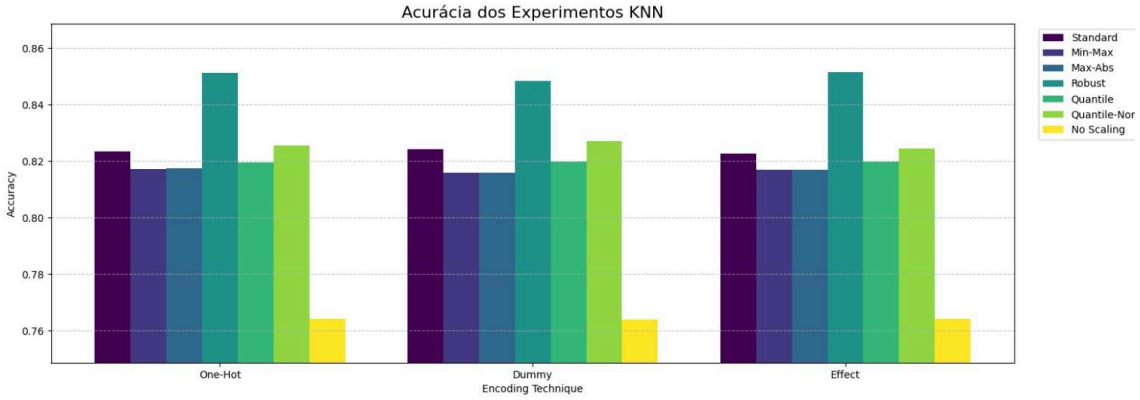
Tabela de Resultados: F1-Score

Encoder	Max-Abs	Min-Max	No Scaling	Quantile	Quantile-Normal	Robust	Standard
Dummy	0.81	0.81	0.74	0.82	0.83	0.85	0.82
Effect	0.81	0.81	0.74	0.82	0.82	0.85	0.82
One-Hot	0.81	0.81	0.74	0.82	0.82	0.85	0.82

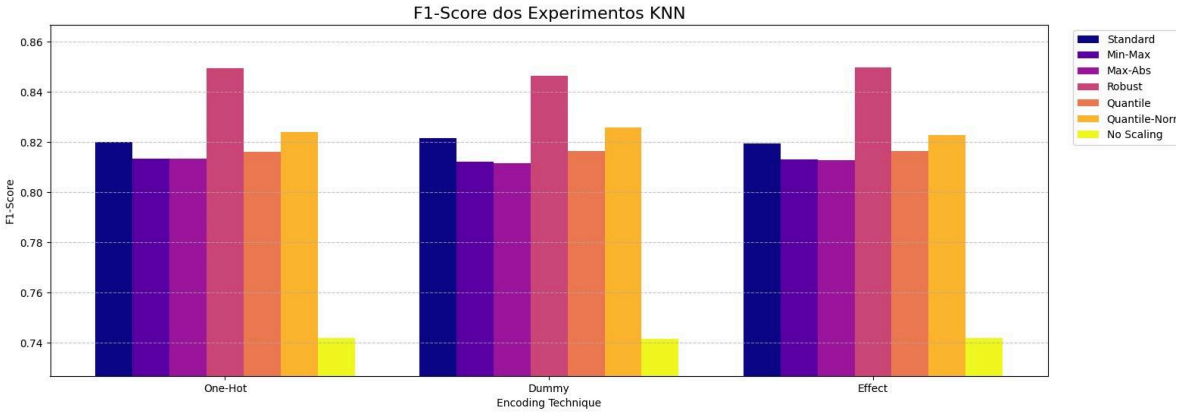
2.3 Gráficos Comparativos

Os gráficos gerados de Acurácia e F1-Score estão apresentados abaixo:

➢ Gráfico da Acurácia dos Experimentos KNN:



➢ Gráfico do F1-Score dos Experimentos KNN:



2.4 Discussão e Observações

Analisando as tabelas e os gráficos, várias observações importantes podem ser feitas:

1. **O Impacto Crítico do Scaling:** A observação mais clara é a diferença drástica de desempenho entre os dados com scaling e o baseline "No Scaling". Conforme visto nos gráficos, a barra "No Scaling" (amarela) é significativamente menor em todos os casos, com uma Acurácia de 0.764 e F1-Score de 0.741. Isso confirma que o KNN, sendo um algoritmo baseado em distância, é extremamente sensível a features em escalas diferentes.

2. **Melhor Desempenho:** O Robust Scaler foi o vencedor incontestável, apresentando a melhor Acurácia e F1-Score em todas as três técnicas de encoding. O desempenho máximo foi alcançado com a combinação Effect Encoding + Robust Scaler, que obteve Acurácia de 0.8516 e F1-Score de 0.8497.

3. **Hipótese para o Sucesso do Robust Scaler:** O Robust Scaler utiliza estatísticas que são robustas a outliers. O seu desempenho superior sugere fortemente que o dataset "Adult" possui outliers em suas features numéricas. Técnicas como o StandardScaler ou Min-Max Scaler são negativamente influenciadas por esses valores extremos. O Robust Scaler lidou melhor com essa característica dos dados, gerando um espaço de features mais adequado para o KNN.

4. **Impacto do Encoding:** Em contraste com o scaling, a escolha da técnica de encoding teve um impacto mínimo, quase insignificante, nos resultados. Como visto nos gráficos, para um mesmo scaler, as barras de Acurácia e F1-Score são quase idênticas entre os três grupos de encoding. Por exemplo, com o Robust Scaler, a Acurácia variou apenas entre 0.8484 e 0.8516. Isso indica que, para este problema, a normalização das features numéricas foi muito mais determinante para o sucesso do modelo.

3. CONCLUSÃO

- **Principais Achados:**
  - Foi comprovado que a etapa de pré-processamento de scaling é fundamental para o bom desempenho do algoritmo KNN. A ausência de scaling degradou a Acurácia do modelo em quase 9%.
  - O Robust Scaler foi a técnica de normalização mais eficaz para o dataset em estudo, sugerindo a presença de outliers nos dados.
  - A escolha entre One-Hot, Dummy ou Effect encoding não apresentou diferença significativa de performance para este problema de classificação.

- **Lições Aprendidas:**
  - A principal lição é a confirmação prática da sensibilidade do KNN à escala das features.
  - A análise exploratória dos dados, especificamente a verificação de outliers é crucial para a escolha da técnica de scaling mais adequada.
  - O uso de Pipelines e ColumnTransformer no scikit-learn é uma prática de implementação robusta e eficiente para gerenciar pré-processamentos complexos e evitar vazamento de dados entre treino e teste.

4. ANEXOS

Seguem links para melhor visualização das tabelas e dos gráficos:

- Link para gráfico da acurácia;
- Link para gráfico do F1-Score;
- Link para Resultados do KNN e Link para Tabela desses Resultados;
- Link para o código no Drive.