# Relatório de Atividade Prática: Lista #9

## Pré-processamento e Agrupamento em Detecção de Fraudes

**Curso:** Ciência da Computação
**Disciplina:** Inteligência Artificial
**Professora:** Cristiane Neri Nobre
**Aluno:** Pedro Henrique Félix Dos Santos

23 de novembro de 2025

## Sumário

# 1 Introdução

O objetivo deste trabalho é aplicar técnicas de pré-processamento de dados e algoritmos de aprendizado não supervisionado na base de dados *Credit Card Fraud Detection*. O desafio principal deste dataset reside no extremo desbalanceamento entre as classes.

Este relatório detalha as decisões tomadas durante a limpeza e tratamento dos dados e a avaliação dos algoritmos K-Means, DBSCAN e SOM.

# 2 Questão 1: Etapas de Pré-processamento

A preparação dos dados tomou como base os itens sugeridos no enunciado da lista, porém a ordem de execução foi adaptada para melhor se adequar ao fluxo de análise. O relatório a seguir descreve a sequência efetivamente utilizada no trabalho.

## 2.1 Visualização e Limpeza Inicial

A análise inicial do dataset revelou que ele contém 284.807 transações e 31 colunas.

- **Valores Ausentes:** A verificação de nulos retornou 0, indicando integridade na coleta dos dados.
- **Redundância:** Foram identificadas e removidas **1.081 linhas duplicadas**. A eliminação de duplicatas é crucial para evitar que o modelo dê peso excessivo a transações repetidas artificialmente, evitando *overfitting* no treinamento.
- **Desbalanceamento:** Conforme ilustrado na Figura 1, a classe de Fraudes é extremamente minoritária, representando **0.17%** dos dados.

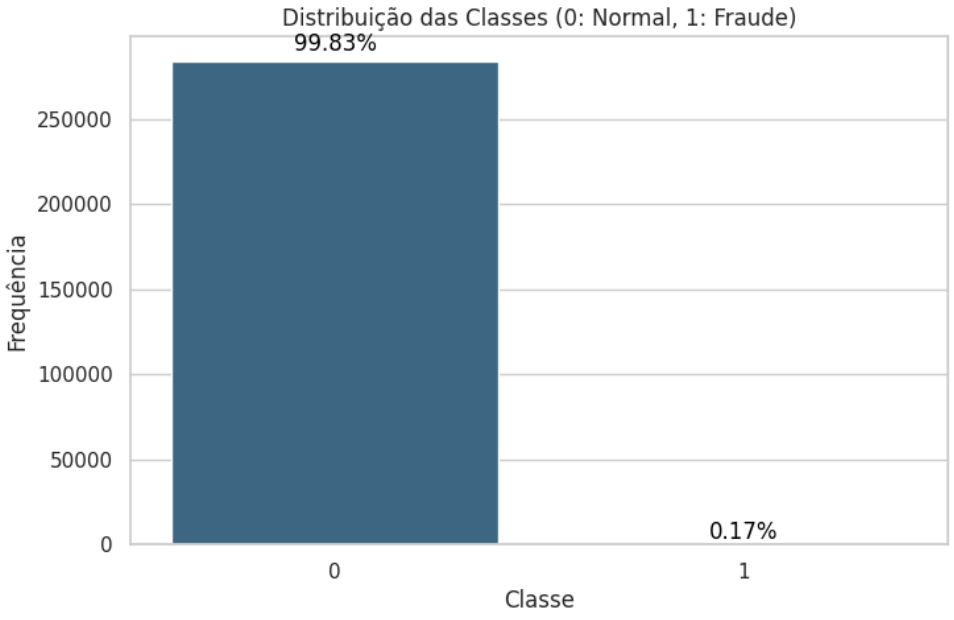



Figura 1: Distribuição das Classes (0: Normal, 1: Fraude).

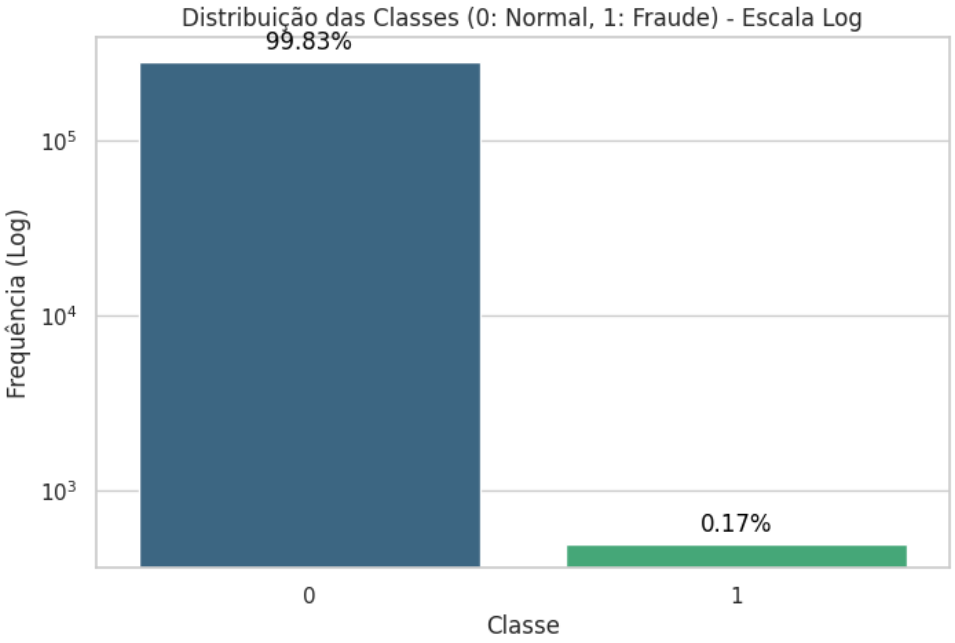



Figura 2: Distribuição das Classes em escala logarítmica para visualização das fraudes.

## 2.2 Detecção e Tratamento de Outliers

As colunas $V1$ a $V28$ são resultados de uma PCA (Análise de Componentes Principais) e já estão centralizadas. O foco da análise de outliers recaiu sobre a variável `Amount` (Valor da Transação).

**Análise do Gráfico:** O Boxplot original (Figura 3, esquerda) mostrava uma distribuição extremamente comprimida, com outliers atingindo valores muito altos que distorciam a escala.

**Decisão:** Optou-se por remover os registros acima do **percentil 99** da variável `Amount`. **Justificativa:** Embora fraudes possam ser de alto valor, valores extremos isolados prejudicam a convergência de algoritmos baseados em distância (como K-Means). O corte resultou na remoção de 2.838 registros, resultando em uma distribuição mais tratável (Figura 3, direita).

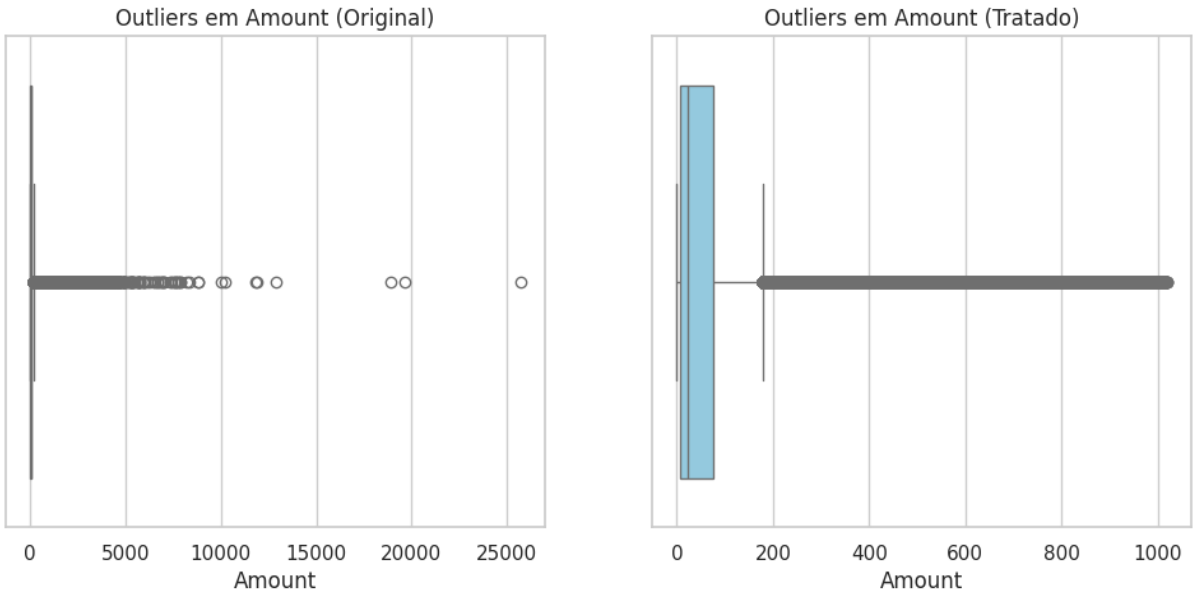



Figura 3: Comparação da distribuição de Amount antes e depois do tratamento de outliers.

## 2.3 Normalização e Correlação

- **Normalização:** Foi utilizado o `RobustScaler` para as colunas `Amount` e `Time`.
- **Justificativa:** Diferente do *StandardScaler* (que usa média e desvio padrão), o *RobustScaler* utiliza a mediana e o intervalo interquartil (IQR). Isso o torna mais resistente a outliers que sobreviveram ao corte inicial.
- **Correlação:** A matriz de correlação (Heatmap) confirmou que não há multicolinearidade significativa entre as variáveis V1-V28 (esperado devido à ortogonalidade do PCA), o que é benéfico para os modelos.

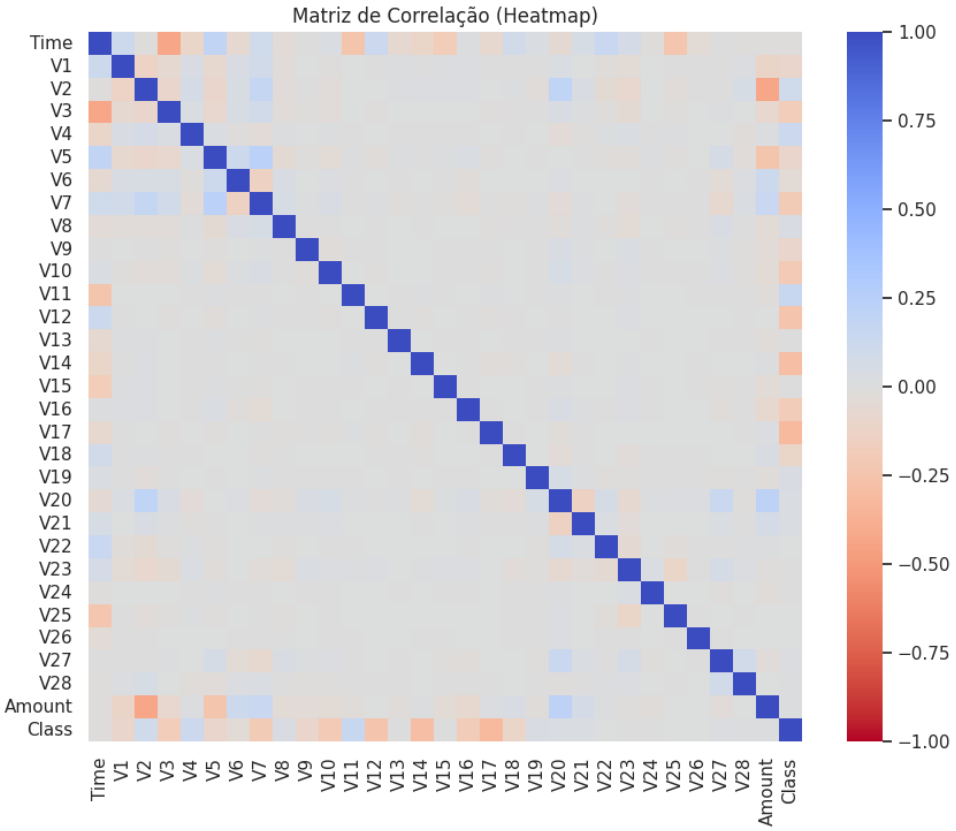



Figura 4: Matriz de correlação.

### 2.4 Balanceamento da Classe (SMOTE - Oversampling)

Para mitigar o desbalanceamento severo no conjunto de treino, aplicou-se a técnica **SMOTE** (Synthetic Minority Over-sampling Technique).

- **Resultado:** A técnica gerou exemplos sintéticos da classe de fraude através da interpolação de vizinhos próximos, igualando a quantidade de exemplos das duas classes no conjunto de treino. Isso é vital para que futuros classificadores não convirjam para a resposta trivial.

## 3 Questão 2: Algoritmos de Agrupamento

Nesta etapa, a variável alvo (`Class`) foi removida para testar a capacidade dos algoritmos não supervisionados de redescobrir os grupos de fraude e não-fraude baseando-se apenas na estrutura dos dados.

### 3.1 Resultados Quantitativos

A métrica utilizada para avaliação foi o **Silhouette Score** (Silhueta), que varia de -1 a 1.

| Algoritmo | Silhouette Score | Observação |
|---|---|---|
| K-Means ($k = 2$) | 0.332 | Separação linear forçada |
| DBSCAN ($\epsilon = 4.0$) | **0.370** | Melhor detecção de densidade |
| SOM (Híbrido) | $\approx 0.051$ | Alta sobreposição / Análise topológica |

Tabela 1: Comparativo de desempenho dos algoritmos de agrupamento.

**Análise dos Scores:** Os valores de silhueta em torno de 0.35 indicam que os grupos **não são bem separados**. Há uma sobreposição significativa entre transações normais e fraudulentas no espaço de características. Isso era esperado, pois fraudes são desenhadas para mimetizar transações legítimas.

## 3.2 Análise Visual e Comparativa

### 3.2.1 K-Means

O gráfico de dispersão do K-Means (Figura 5a) mostrou uma divisão geométrica clara dos dados. O algoritmo forçou a criação de dois grupos esféricos. No entanto, dada a complexidade das fraudes, essa fronteira rígida provavelmente classifica muitas transações legítimas como fraude e vice-versa.

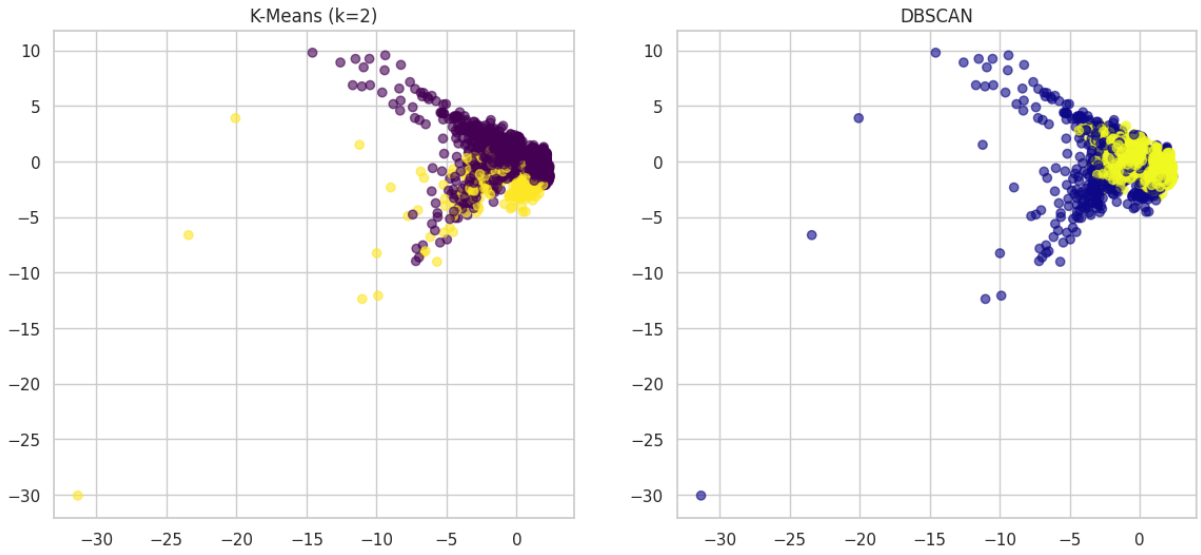
### 3.2.2 DBSCAN

O DBSCAN obteve o melhor score (0.370). **Análise:** Sua vantagem reside na capacidade de identificar ruídos e clusters de formas arbitrárias. O gráfico mostra que ele conseguiu agrupar a maior parte dos dados densos em um cluster principal, deixando os pontos dispersos (potenciais fraudes) como anomalias ou segundos grupos. A dificuldade foi ajustar o parâmetro $\epsilon$ (epsilon) devido à alta dimensionalidade.

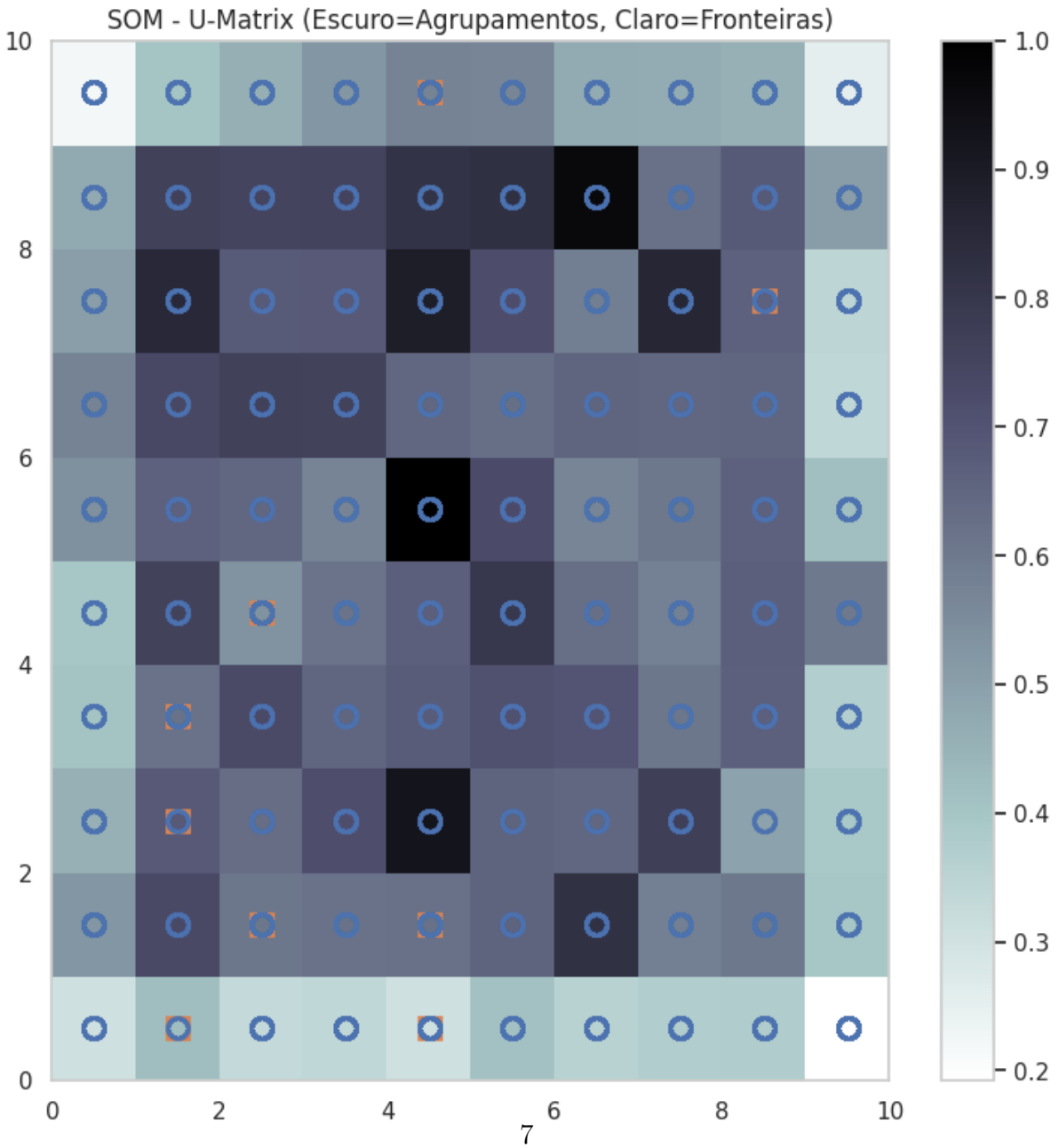
### 3.2.3 Self-Organizing Maps (SOM)

Utilizou-se uma rede SOM $10 \times 10$ para projetar a topologia dos dados.

- **U-Matrix (Matriz de Distâncias):** O gráfico gerado (Figura 5b) revelou áreas escuras (agrupamentos densos), mas a ausência de "paredes"claras (áreas brancas contínuas) separando dois grupos distintos.
- **Conclusão do SOM:** A visualização confirmou que a "fraude"não é um grupo isolado, mas sim um conjunto de pontos que ocorrem misturados ao grupo normal. O baixo valor de silhueta (0.051) confirma matematicamente essa sobreposição observada visualmente.




(a) Clusters: K-Means vs DBSCAN




(b) SOM: U-Matrix e Clusters

# 4 Conclusão

O pré-processamento foi a etapa mais crítica deste trabalho. A aplicação do `RobustScaler` e a remoção de duplicatas garantiram a estabilidade numérica dos algoritmos.

Na análise de agrupamento, conclui-se que métodos puramente não supervisionados têm dificuldade em separar perfeitamente fraudes de transações normais neste dataset, evidenciado pelos scores de silhueta baixos ($< 0.4$). O **DBSCAN** mostrou-se o mais promissor conceitualmente por tratar fraudes como anomalias de densidade, enquanto o **SOM** forneceu uma excelente ferramenta visual para confirmar a sobreposição das classes. Para um sistema de produção, recomenda-se o uso dos dados balanceados (SMOTE) em algoritmos supervisionados (como Random Forest ou Redes Neurais), que se beneficiariam deste pré-processamento.

# 5 Link para o Código

O código completo, executado no Google Colab, pode ser acessado no link abaixo:

`https://colab.research.google.com/drive/1pSUD6CeHM-4VmW9gUmvDXLgsn7yimCXt?usp=sharing`