Relatório Técnico: Detecção de Fraudes e Agrupamento

Breno Moreira

21 de novembro de 2025

Parte 1: Pré-processamento e Classificação

1 Introdução e Objetivo

Este relatório descreve o processo de pré-processamento e modelagem realizado na base de dados *Credit Card Fraud Detection*. O objetivo principal foi tratar os dados para permitir a criação de um modelo preditivo capaz de identificar transações fraudulentas.

A base de dados apresenta desafios significativos, sendo o principal deles o extremo desbalanceamento entre as classes (fraudes representam apenas 0,17% do total).

2 Etapas de Pré-processamento

2.1 Tratamento de Dados Iniciais

A verificação inicial não apontou valores ausentes. No entanto, foi identificada a presença de redundância.

- **Ação:** Foram removidas **1.081 linhas duplicadas** para evitar viés no modelo.

2.2 Análise do Desbalanceamento

A visualização da variável alvo (*Class*) confirmou o desbalanceamento severo.

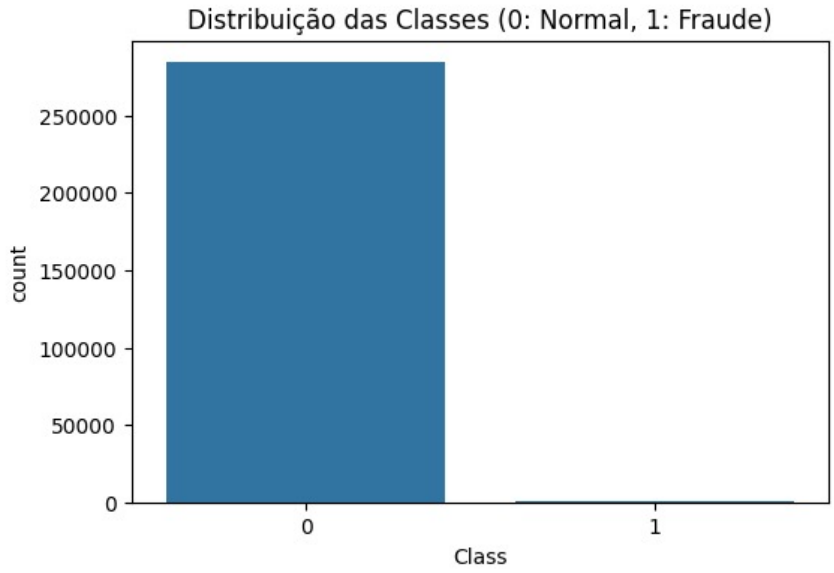

Figura 1: Distribuição das Classes. A classe 1 (Fraude) é quase invisível.

2.3 Normalização e Outliers

Como as variáveis V1-V28 já estavam padronizadas pelo PCA, o foco foi nas variáveis *Time* e *Amount*. Utilizou-se o **RobustScaler**, pois ele utiliza a mediana e ignora outliers extremos, algo comum em valores de fraudes.

2.4 Análise de Correlação

A matriz de correlação evidenciou que não existe multicolinearidade entre as variáveis V1-V28 (ortogonais devido ao PCA).

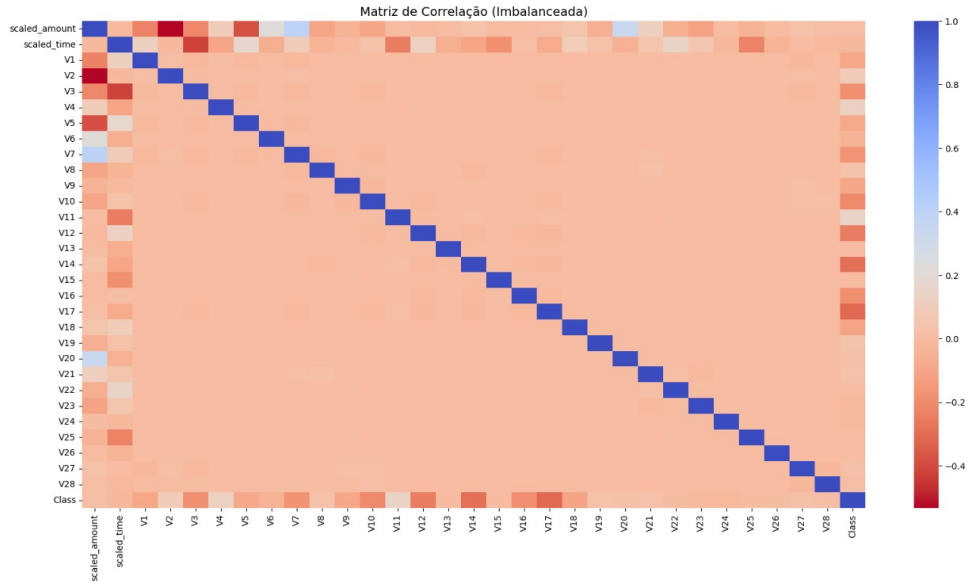

Figura 2: Matriz de Correlação das variáveis.

3 Resultados da Classificação (Antes x Depois)

Foi utilizado o método **SMOTE** (apenas nos dados de treino) para balancear as classes.

- **Sem SMOTE:** Recall de 0.58 (detectava apenas 58% das fraudes).
- **Com SMOTE:** Recall subiu para **0.87** (detecta 87% das fraudes).

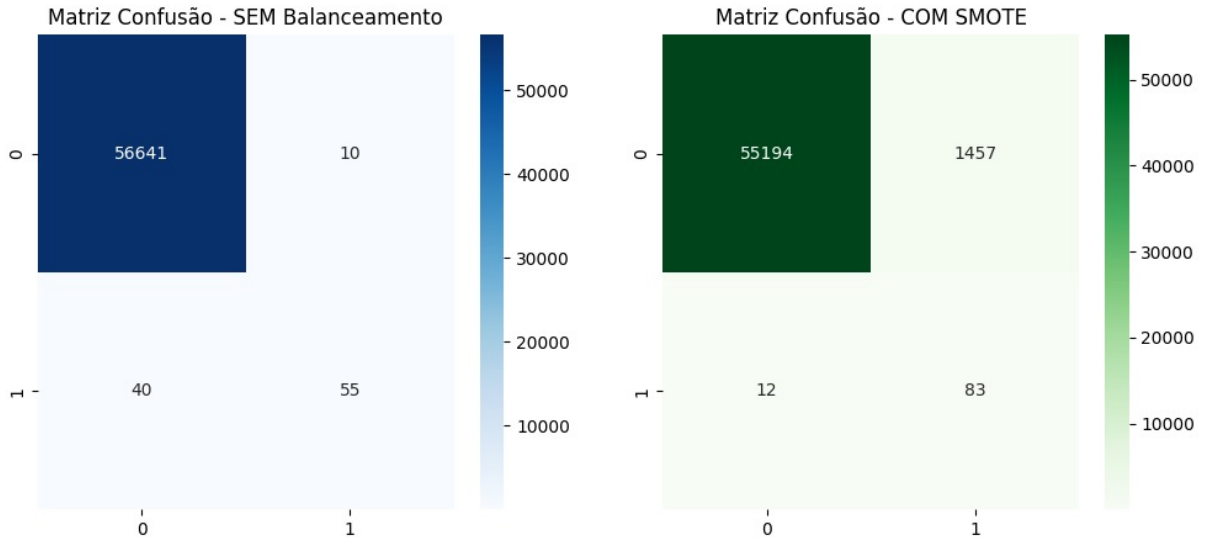

Figura 3: Matriz de Confusão: Sem Balanceamento vs Com SMOTE.

Parte 2: Algoritmos de Agrupamento

1 Metodologia

O objetivo desta etapa foi aplicar algoritmos não supervisionados (K-Means, DBSCAN e SOM) para verificar se a base possui dois grupos naturais (Fraude vs Normal).

- A variável *Class* foi removida para o teste ser "cego".
- Foi utilizada uma amostra de **10.000 instâncias** para viabilizar o cálculo da métrica *Silhouette Score*.

2 Resultados dos Agrupamentos

A tabela abaixo resume o desempenho dos algoritmos na tentativa de separar os dados em dois grupos.

Algoritmo	**Silhueta**	**Grupos Encontrados (Contagem)**	**Avaliação**
K-Means (k=2)	0.6180	G0: 9712 — G1: 288	Enganoso
DBSCAN	-0.0527	G0: 7263 — Ruído: 1779	Realista
SOM (2x1)	0.4940	G0: 9353 — G1: 647	Forçado

Tabela 1: Comparação das métricas de agrupamento.

3 Análise Crítica e Conclusão

3.1 Análise dos Algoritmos

1. **K-Means:** Obteve a melhor silhueta (0.61), mas o agrupamento é enganoso. O grupo menor (288 elementos) é muito maior que a quantidade real de fraudes na amostra (aprox. 17). O K-Means provavelmente separou *outliers* de valor alto, e não fraudes.
2. **DBSCAN:** Foi o algoritmo decisivo. Com silhueta negativa e alto ruído, mostrou que **não existem dois clusters densos** e separados. As fraudes estão "misturadas"e dispersas entre as transações normais.
3. **SOM:** Mesmo forçando a rede a ter apenas 2 neurônios, a silhueta mediana (0.49) confirma que a separação não é natural.

3.2 Conclusão Final

Os algoritmos de agrupamento demonstraram que a base de dados **não possui dois grupos naturais bem definidos** no espaço vetorial. As fraudes não formam um cluster isolado, o que justifica a necessidade de métodos supervisionados (como o SMOTE usado na Parte 1) para ensiná-lo explicitamente o que é uma fraude.

Acesso ao Código Fonte

O código completo desenvolvido para ambas as partes deste trabalho, contendo os scripts de pré-processamento, visualização, balanceamento e agrupamento, está disponível no repositório abaixo:

`https://github.com/brnSalamandra/etapas-de-pr--processamento-`