

| | |
|---|---|
| Curso ANÁLISE E DESENVOLVIMENTO DE SISTEMAS | Disciplina INTELIGÊNCIA ARTIFICIAL E MACHINE LEARNING |
| Professor LUCY MARI | Data 10/11/2025 |

PROJETO INTERDISCIPLINAR – CANNOLI

Integrantes:

- Anderson Yavi Fernandez - RA: 24025678
- Gabriel Gonçalves Pires - RA: 24026518
- Isabela Nunes Zeferino - RA: 24026460
- Kaique Neres de Oliveira - RA: 24026134
- Felipe Galdino de Carvalho - RA: 2402656

1. Introdução e Objetivo

O principal desafio de marketing da Cannoli é a falta de personalização, pois a base de clientes não é segmentada. Isso leva a campanhas de marketing genéricas e pouco eficientes.

O objetivo deste trabalho foi aplicar um modelo de Machine Learning para resolver esse problema, agrupando os clientes em perfis distintos com base no seu comportamento de compra. A ideia é criar segmentos que permitam à Cannoli desenvolver estratégias de marketing mais inteligentes e direcionadas.

2. Aplicação do Modelo

O processo foi dividido em três etapas principais, executadas em um notebook Colab.

2.1. Etapa 1: Preparação dos Dados (RFM)

O primeiro passo foi preparar os dados para o modelo. Carregamos o arquivo de pedidos (Order_semicolon.csv), limpamos as datas e, a partir daí, calculamos três métricas essenciais para cada cliente, um processo conhecido como **Análise RFM**:

- **Recência (R)**: Há quantos dias foi a última compra.
- **Frequência (F)**: O número total de vezes que o cliente já comprou.
- **Monetário (M)**: O valor total que o cliente já gastou.

```
import pandas as pd
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler
import matplotlib.pyplot as plt

# 1. Carregar e Limpar os Dados
print("Carregando e limpando os dados de pedidos...")
df_order = pd.read_csv('Order_semicolon.csv', sep=';')
df_order['createdAt'] = pd.to_datetime(df_order['createdAt'],
dayfirst=True, errors='coerce')
df_order = df_order.dropna(subset=['createdAt'])
print("...dados carregados e limpos.")

# 2. Calcular e Normalizar as Métricas RFM
print("\nCalculando e normalizando as métricas RFM...")
data_maxima = df_order['createdAt'].max() + pd.to_timedelta('1D')

rfm_df = df_order.groupby('customer').agg(
    Recencia=('createdAt', lambda date: (data_maxima -
date.max()).days),
    Frequencia=('id', 'count'),
    Monetario=('totalAmount', 'sum')
).reset_index()

rfm_features = rfm_df[['Recencia', 'Frequencia', 'Monetario']]
```

```
scaler = StandardScaler()
rfm_scaled = scaler.fit_transform(rfm_features)
print("...métricas RFM prontas para o modelo.")
```

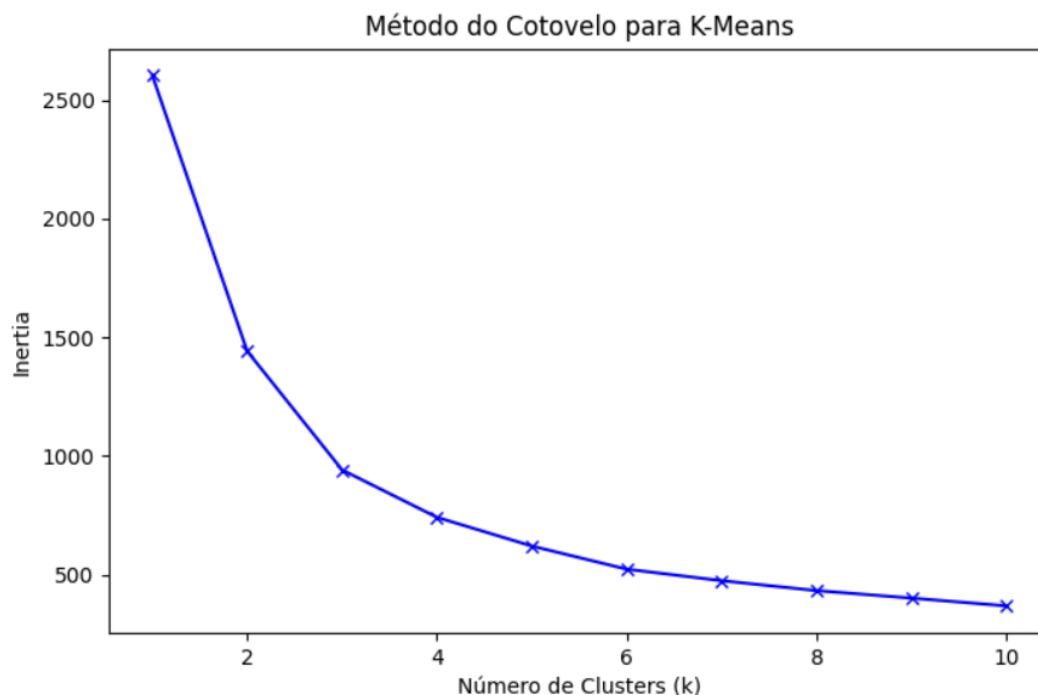
2.2. Etapa 2: Definição e Aplicação do Modelo (K-Means)

Com os dados preparados, usamos a técnica "Método do Cotovelo" para decidir em quantos grupos (clusters) iríamos dividir os clientes. O gráfico resultante indicou que **4** era um bom número. Em seguida, aplicamos o algoritmo de Machine Learning **K-Means** para agrupar os clientes.

```
# 3. Rodando o Método do Cotovelo (Elbow Method) para achar o
melhor numero de grupos
print("\nRodando o Método do Cotovelo para achar 'k'...")
inertia = []
K = range(1, 11)
for k in K:
    kmeans = KMeans(n_clusters=k, random_state=42, n_init=10)
    kmeans.fit(rfm_scaled)
    inertia.append(kmeans.inertia_)

# Plotando o grafico para a gente visualizar o "cotovelo"
plt.figure(figsize=(8, 5))
plt.plot(K, inertia, 'bx-')
plt.xlabel('Número de Clusters (k)')
plt.ylabel('Inertia')
plt.title('Método do Cotovelo para K-Means')
plt.show()
```

Resultado do Método do Cotovelo:



2.3. Etapa 3: Análise e Interpretação dos Resultados

Esta é a etapa final, onde pegamos os grupos que o modelo criou e os traduzimos para o negócio, dando um "apelido" (persona) para cada um deles.

```
# 4. Aplicando o K-Means e nomeando os segmentos
print("\nAplicando o modelo K-Means com k=4...")
kmeans_final = KMeans(n_clusters=4, random_state=42, n_init=10)
kmeans_final.fit(rfm_scaled)
rfm_df['Cluster'] = kmeans_final.labels_
print("...modelo aplicado! Clientes foram agrupados.")

# Interpretando e dando nome aos grupos
print("\nAnalisando e nomeando os segmentos encontrados...")
analise_clusters = rfm_df.groupby('Cluster')[['Recencia',
'Frequencia', 'Monetario']].mean().reset_index()

# Usando um sistema de ranking pra nomear os grupos automaticamente
analise_clusters['rank_recencia'] =
analise_clusters['Recencia'].rank(ascending=True)
analise_clusters['rank_frequencia'] =
analise_clusters['Frequencia'].rank(ascending=False)
analise_clusters['rank_monetario'] =
analise_clusters['Monetario'].rank(ascending=False)
analise_clusters['pontuacao_geral'] =
analise_clusters['rank_recencia'] +
analise_clusters['rank_frequencia'] +
analise_clusters['rank_monetario']
analise_clusters = analise_clusters.sort_values('pontuacao_geral')

nomes_segmentos = ['Clientes Campeões', 'Clientes Leais', 'Clientes
Novos/Promissores', 'Clientes em Risco']
analise_clusters['Segmento'] =
nomes_segmentos[:len(analise_clusters)]
print("...segmentos nomeados com sucesso!")

# Exibindo o resultado final de negocio
print("\n\n--- RESULTADO FINAL: Resumo dos Segmentos de Clientes ---")
display(analise_clusters[['Segmento', 'Recencia', 'Frequencia',
'Monetario']])
```

Tabela de Resumo dos Segmentos:

--- RESULTADO FINAL: Resumo dos Segmentos de Clientes ---

| | Segmento | Recencia | Frequencia | Monetario |
|---|----------------------------|------------|------------|------------|
| 3 | Clientes Campeões | 54.304348 | 4.673913 | 314.200000 |
| 0 | Clientes Leais | 89.773585 | 3.041509 | 182.892113 |
| 2 | Clientes Novos/Promissores | 71.854610 | 1.542553 | 82.622872 |
| 1 | Clientes em Risco | 257.921397 | 1.436681 | 82.419432 |

3. Conclusão

A aplicação do modelo K-Means foi bem-sucedida em segmentar a base de clientes em quatro grupos acionáveis: **Clientes Campeões**, **Clientes Leais**, **Clientes Novos/Promissores** e **Clientes em Risco**.

O projeto cumpre o objetivo de aplicar um modelo de ML a um problema de negócio real, transformando dados brutos em inteligência para o direcionamento de campanhas de marketing e otimização de recursos. Com essa segmentação, a Cannoli pode agora criar campanhas muito mais eficientes, como uma oferta de reativação para os "Clientes em Risco".