

# Aula 10 - Exercício Prático Aprendizado Não Supervisionado

November 7, 2023

Vitor Albuquerque de Paula

**1 Execute os algoritmos k-means e single linkage nos datasets a seguir. Qual o melhor agrupamento que você obteve (pode plotar os grupos por cores para facilitar a identificação). Pode usar bibliotecas ou implementar seu próprio código.**

- [Aggregation Dataset](#)
- [D31 Dataset](#)
- [Pathbased Dataset](#)
- [Flame Dataset](#)

## 2 Análise de Cluster para Datasets Diversos

Neste notebook, realizaremos análises de clustering usando os algoritmos K-means e Single Linkage em quatro conjuntos de dados diferentes: Aggregation.txt, D31.txt, pathbased.txt, e flame.txt. Cada conjunto de dados será analisado individualmente para identificar padrões e agrupamentos.

```
[35]: import pandas as pd
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
from scipy.cluster.hierarchy import dendrogram, linkage
import warnings

# Filtrando avisos específicos
warnings.filterwarnings('ignore', category=FutureWarning, module='sklearn')
warnings.filterwarnings('ignore', message='KMeans is known to have a memory_
↳leak on Windows with MKL')

def perform_clustering(file_path, k_values=[3, 5, 7, 11]):
    # Carregando o dataset
    data = pd.read_csv(file_path, sep='\t', header=None)
    X = data.iloc[:, :2].values

    # Aplicando K-means
    plt.figure(figsize=(10, 5))
```

```

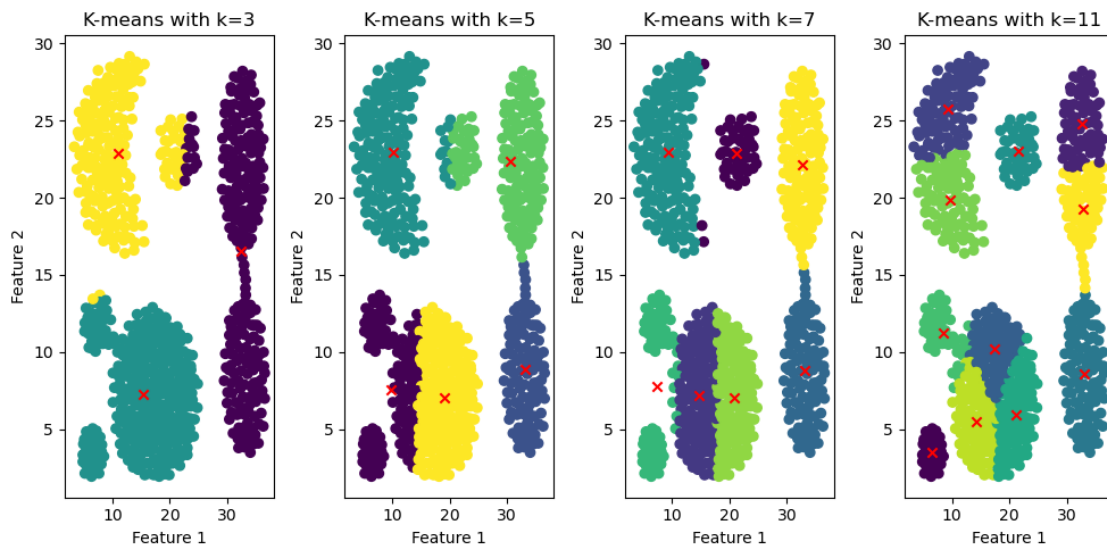
for i, k in enumerate(k_values, 1):
    # Ajustando n_init para evitar FutureWarning
    kmeans = KMeans(n_clusters=k, n_init=10, random_state=0).fit(X)
    labels = kmeans.labels_
    plt.subplot(1, len(k_values), i)
    plt.scatter(X[:, 0], X[:, 1], c=labels, cmap='viridis')
    plt.scatter(kmeans.cluster_centers_[0], kmeans.cluster_centers_[1],
↪1], c='red', marker='x')
    plt.title(f'K-means with k={k}')
    plt.xlabel('Feature 1')
    plt.ylabel('Feature 2')
plt.tight_layout()
plt.show()

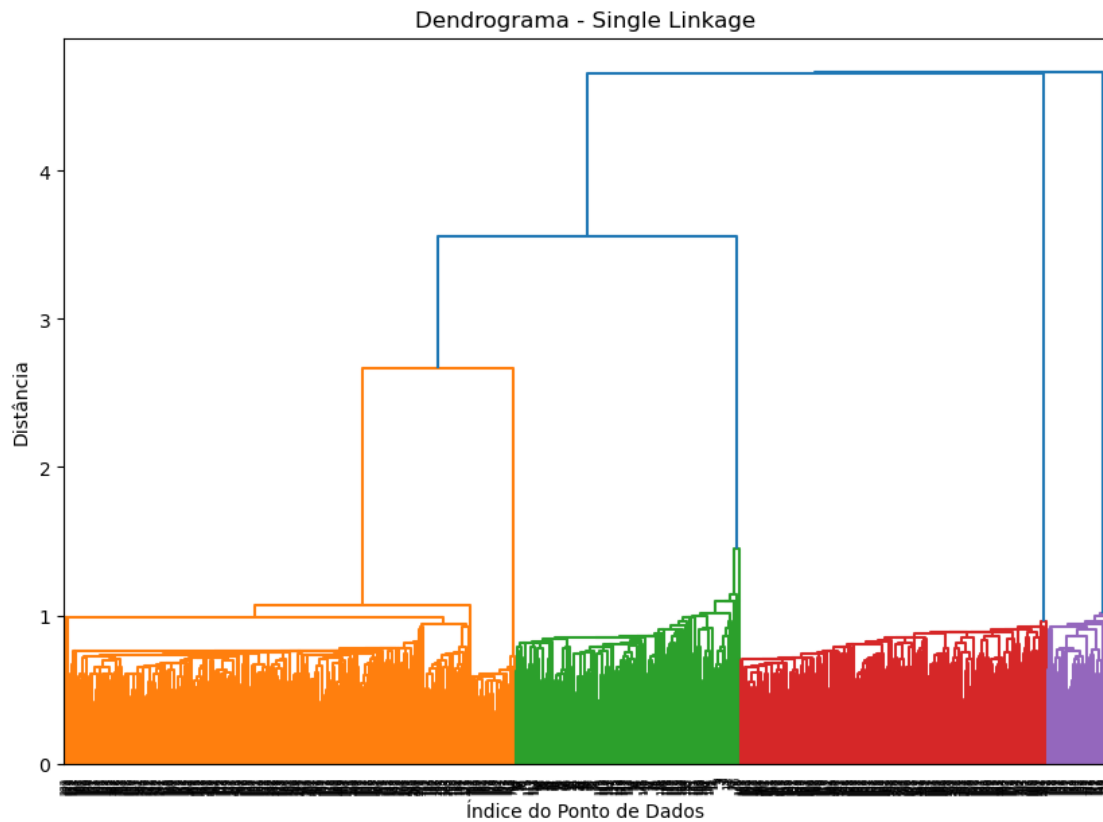
# Aplicando Single Linkage
linked = linkage(X, 'single')
plt.figure(figsize=(10, 7))
dendrogram(linked, orientation='top', distance_sort='descending',
↪show_leaf_counts=True)
plt.title('Dendrograma - Single Linkage')
plt.xlabel('Índice do Ponto de Dados')
plt.ylabel('Distância')
plt.show()

```

### 3 Análise do Dataset Aggregation.txt

[36]: perform\_clustering('Aggregation.txt')

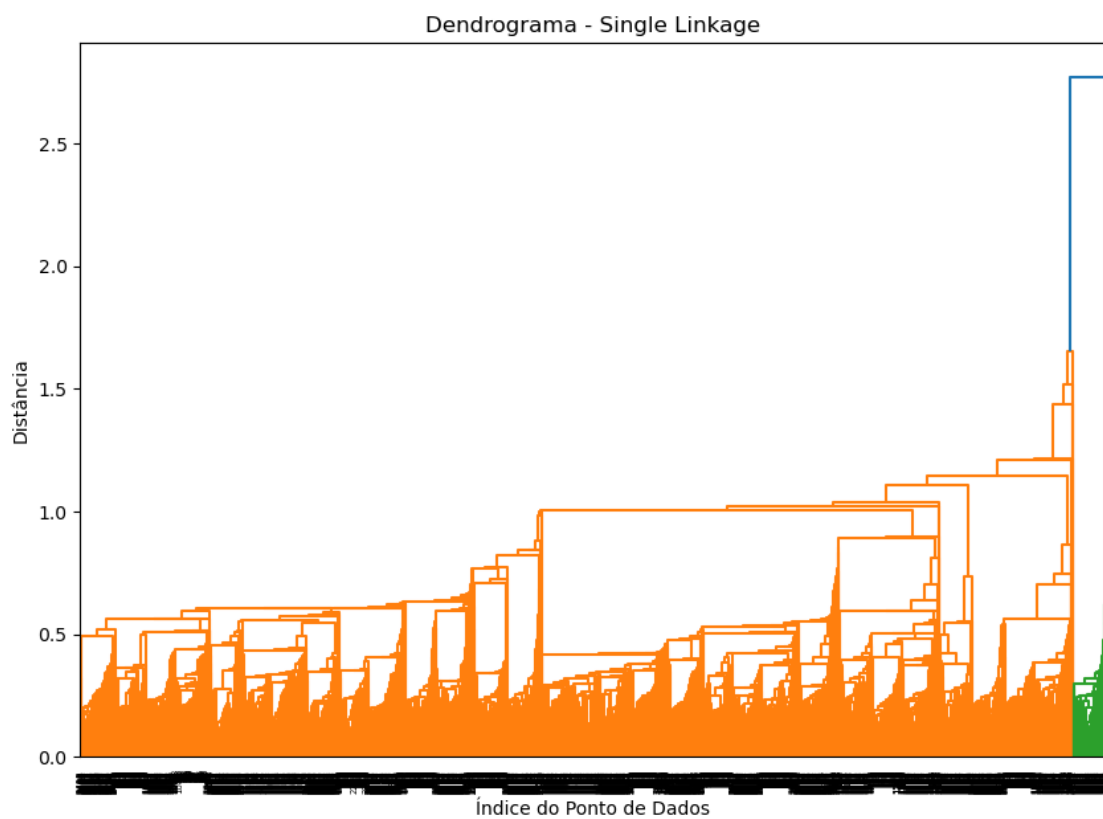
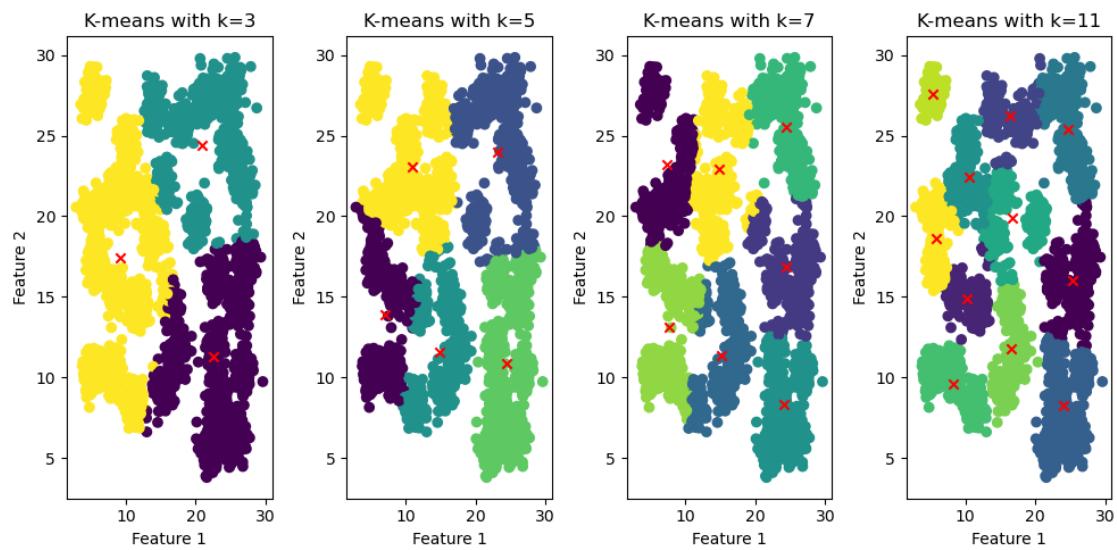




Para esse dataset, o melhor agrupamento foi o com  $K = 7$

## 4 Análise do Dataset D31.txt

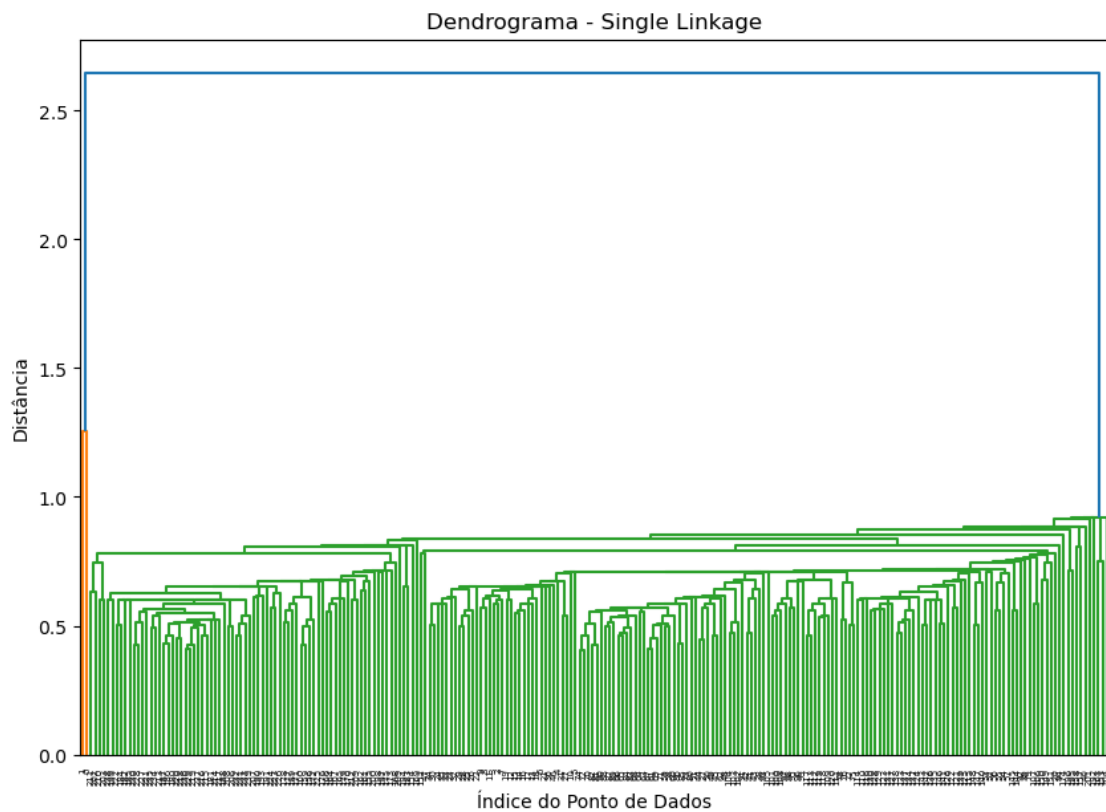
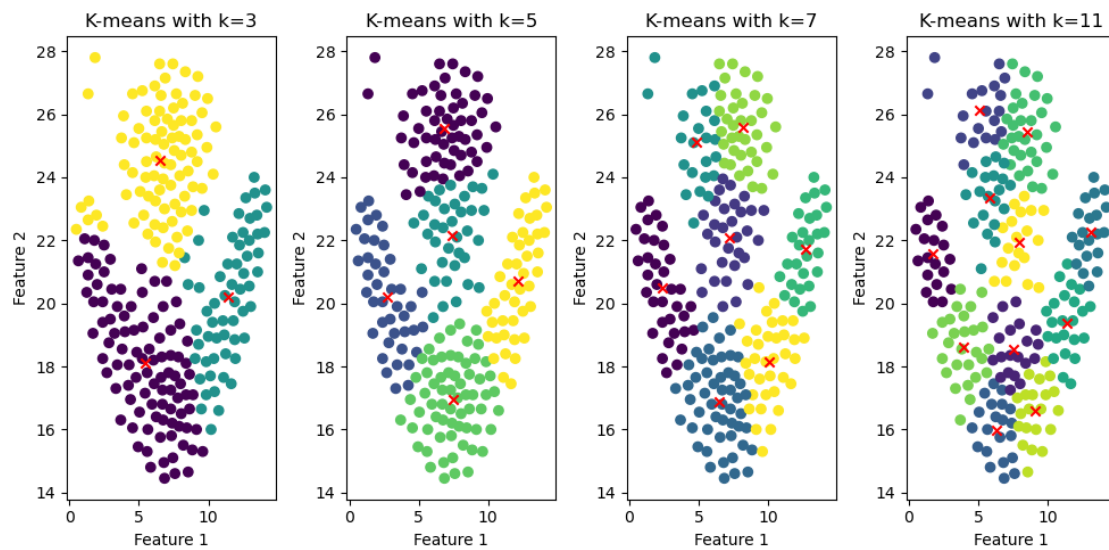
```
[37]: perform_clustering('D31.txt')
```



Para esse dataset, o melhor agrupamento foi o com  $K = 11$

## 5 Análise do Dataset flame.txt

```
[38]: perform_clustering('flame.txt')
```



Para esse dataset, o melhor agrupamento foi o com  $K = 5$

## 6 Análise do Dataset pathbased.txt

```
[39]: perform_clustering('pathbased.txt')
```

