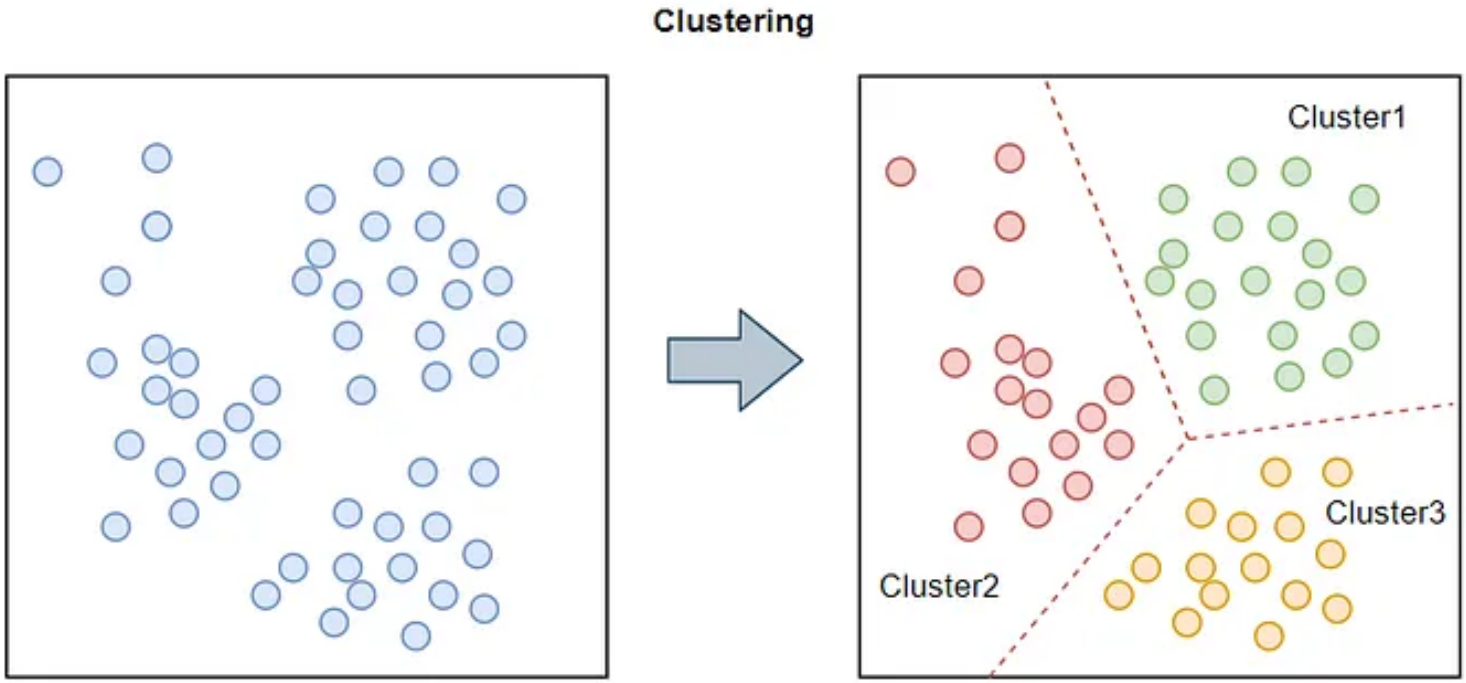
Clusterização

Cesar de Oliveira F. Silva

Setembro / 2026

Roteiro

Clusterização - Análise de Agrupamentos



O que é análise de agrupamentos?

- **Cluster**: coleção de objetos
  - semelhantes entre si dentro do mesmo grupo;
  - dissimilares em relação aos objetos de outros grupos.
- **Clusterização**: processo de agrupar dados em clusters.
- É uma forma de **classificação não supervisionada**:
  - não há rótulos ou classes pré-definidas;
  - o próprio algoritmo descobre a estrutura dos dados.

O que *não* é clusterização

- **Classificação supervisionada**
  - existe um atributo meta (classe) conhecido;
  - o objetivo é aprender a mapear entrada $\rightarrow$ classe.
- **Segmentação simples**
  - ex.: separar alunos por ordem alfabética;
  - grupos não refletem *similaridade* de características.
- **Resultado de consulta SQL**
  - grupos são impostos por filtros externos (`WHERE`, `GROUP BY`).
- **Particionamento de grafos**
  - pode ter sentido estrutural, mas nem sempre reflete proximidade em espaço de atributos.

Aplicações de clusterização

- **Reconhecimento de padrões** em geral.
- **Dados espaciais**:
  - criação de mapas temáticos em SIG;
  - agrupamento de regiões com características semelhantes.
- **Saúde**: agrupamento de pacientes com sintomas parecidos.
- **Marketing**: segmentação de mercado.
- **Web**:
  - classificação de documentos;
  - análise de logs para detectar padrões de navegação.
- **Agrometeorologia**: identificação de áreas pluviometricamente homogêneas.

O que é uma boa clusterização?

- Clusters com:
  - **alta similaridade interna** (objetos de um cluster muito parecidos);
  - **baixa similaridade externa** (clusters bem separados).
- A qualidade depende de:
  - medida de similaridade;
  - algoritmo e implementação;
  - conhecimento do domínio.
- Muitas vezes a avaliação é parcialmente **subjetiva**.

Tipos de dados em clusterização

- **Numéricos**: reais ou inteiros (temperatura, latitude, altura).
- **Binários**: dois estados (0/1, sim/não, presença/ausência).
- **Nominais**: categorias sem ordem natural (cor, tipo de solo, classe de uso da terra).
- **Mistos**: combinação de vários tipos (frequente em dados ambientais).

Normalização de variáveis numéricas

- É importante que variáveis fiquem na **mesma escala**:
  - evita que atributos com unidades grandes dominem a distância.
- **Min–Max** para um atributo x:

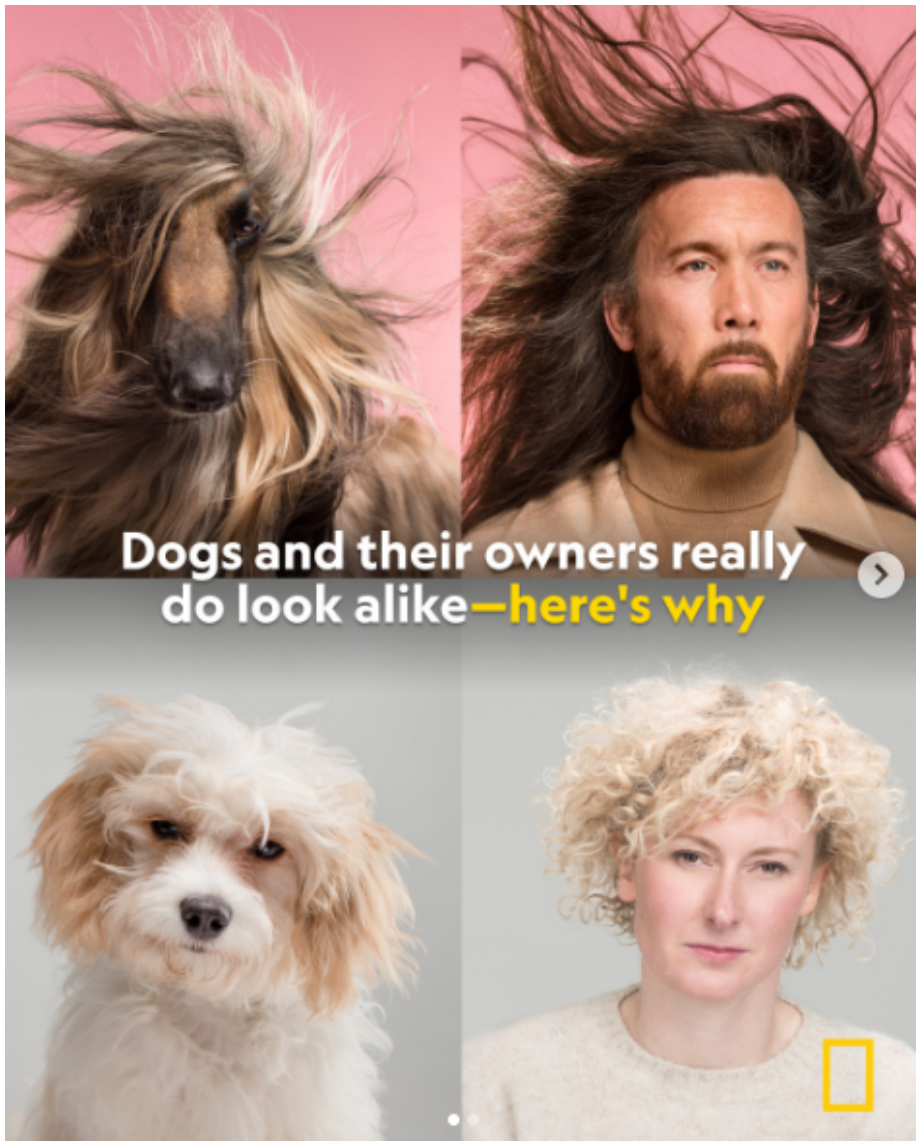
$$x' = \frac{x - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}}$$

- **Z-score**:

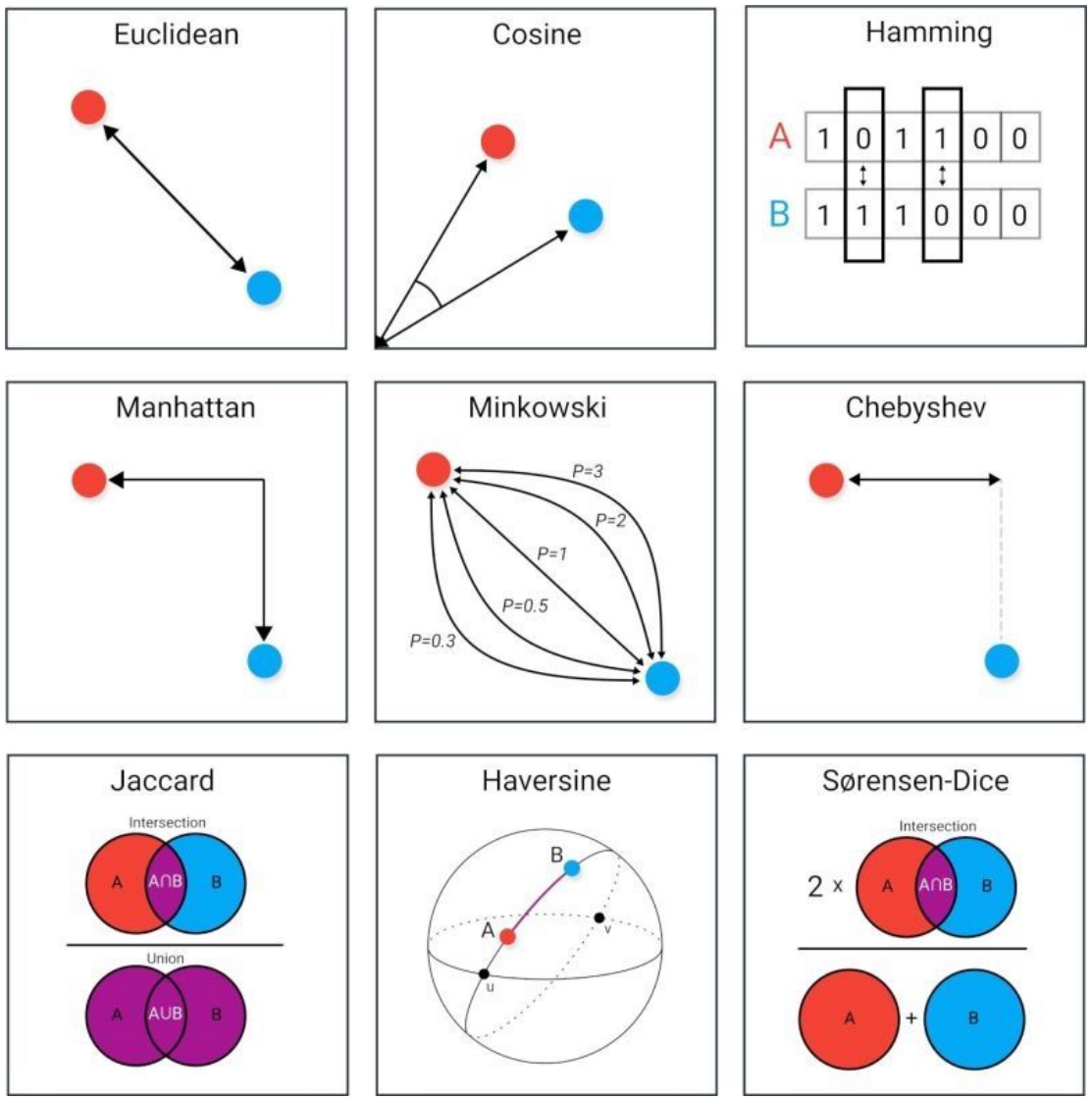
$$x' = \frac{x - \mu}{\sigma}$$

- Pode-se usar também o **desvio absoluto médio** como alternativa robusta.

Similaridade



Distâncias


Euclidean
Cosine
Hamming
A 1 0 1 1 0 0
B 1 1 1 0 0 0
Manhattan
Minkowski
P=3
P=2
P=1
P=0.5
P=0.3
Chebyshev
Jaccard
Intersection
A
A∩B
B
Union
A
AUB
B
Haversine
B
A
v
u
Sørensen-Dice
Intersection
2 x
A
A∩B
B
A
+
B

Usos das Distâncias

- **Euclidiano/Manhattan**: para características numéricas normalizadas.
- **Cosseno/Jaccard**: para características de texto e categóricas.
- **Mahalanobis**: para dados gaussianos multivariados.
- **Hamming**: para comparação em nível de bit.
- **Bray-Curtis**: para análise de composição ecológica.

Similaridade entre variáveis numéricas

- A similaridade é geralmente medida por **distância**.
- Distância de **Minkowski** (geral):

$$d_q(i,j) = \left(\sum_{f=1}^{p} |x_{if} - x_{jf}|^q \right)^{1/q}$$

- Casos especiais:
  - $q = 1$: **distância de Manhattan**

$$d_1(i,j) = \sum_{f=1}^{p} |x_{if} - x_{jf}|$$

  - $q = 2$: **distância Euclidiana**

$$d_2(i,j) = \sqrt{\sum_{f=1}^{p} (x_{if} - x_{jf})^2}$$

Exercício rápido

Pontos: $A(7, 5)$ e $B(4, 3)$

- Distância de Manhattan:

$$d_1(A, B) = |7 - 4| + |5 - 3| = 3 + 2 = 5$$

- Distância Euclidiana:

$$d_2(A, B) = \sqrt{(7 - 4)^2 + (5 - 3)^2} = \sqrt{3^2 + 2^2} = \sqrt{13}$$

Similaridade entre variáveis binárias

Para duas variáveis binárias, usamos a tabela de contingência:

	$y = 1$	$y = 0$
$x = 1$	a	b
$x = 0$	c	d

- **Simétrica** (ex.: sexo): coeficiente de *simple matching*

$$s = \frac{a + d}{a + b + c + d}$$

- **Assimétrica** (ex.: presença de doença, ocorrência de geada): coeficiente de Jaccard

$$s_J = \frac{a}{a + b + c}$$

(ignora concordâncias 0–0).

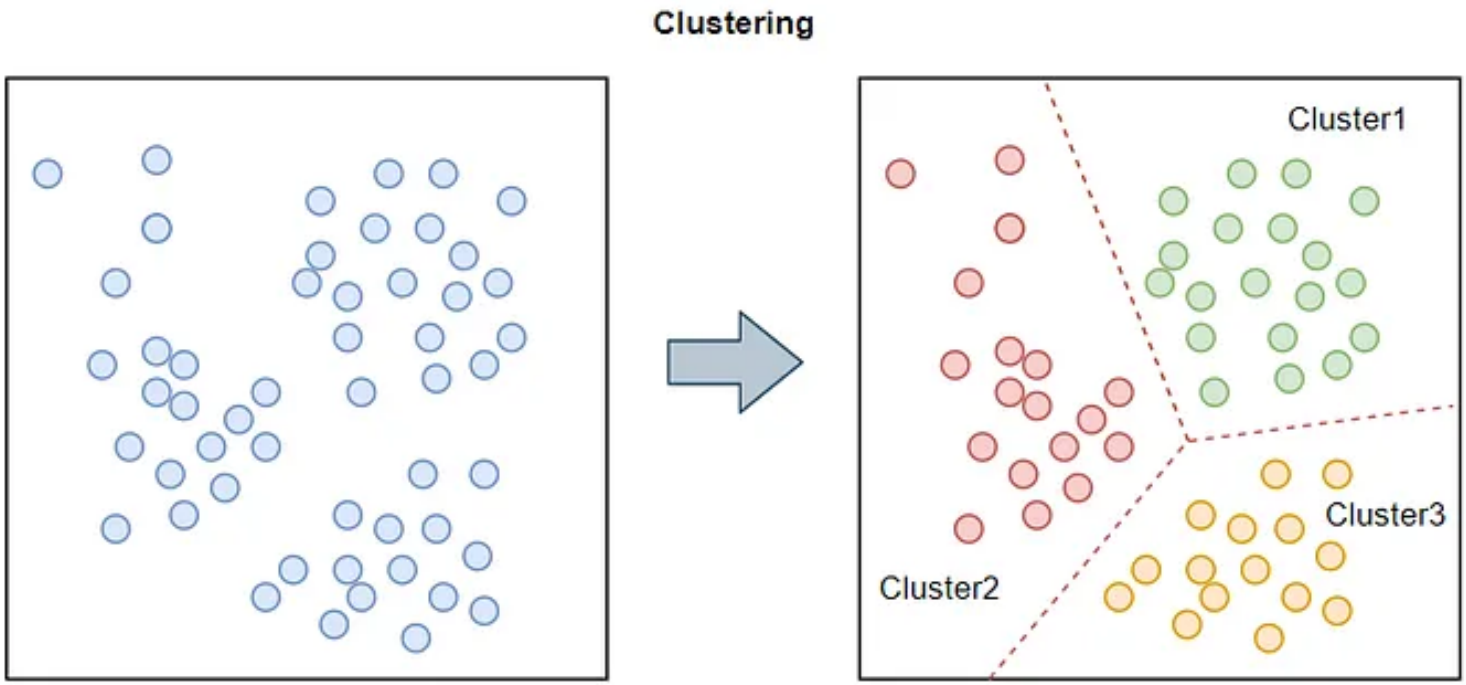
Similaridade para variáveis nominais e mistas

- **Nominais** (múltiplas categorias):
  - *Simple matching*: fração de atributos com mesmo valor.
  - Ou codificação em várias variáveis binárias (one-hot).
- **Mistas**:
  - combinar contribuições de cada atributo:

$$d(i,j) = \frac{\sum_f w_f \, d_{ij}^{(f)}}{\sum_f w_f}$$

  - onde $d_{ij}^{(f)}$ é a distância normalizada no atributo f, e w_f é o peso (importância) daquele atributo.

Métodos de clusterização



Grandes famílias de métodos

- **Particionamento**
  - constrói diretamente *k* grupos (ex.: k-means, k-medoids).
- **Hierárquicos**
  - constroem árvore de aglomeração/divisão (dendrograma).
- **Baseados em densidade**
  - usam regiões densas no espaço de atributos (ex.: DBSCAN).

Função objetivo em métodos de particionamento

- Dado um conjunto de n objetos e um número k de clusters, queremos uma partição $C_1, \ldots, C_k$ que minimize:

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} \|p - m_i\|^2$$

  onde m_i é o **centroide** do cluster C_i.
- Intuição: pontos de cada cluster devem ficar próximos ao seu centro.

Algoritmo k-means

Entrada: número de clusters k, conjunto de dados D. **Saída:** centroides e clusters associados.

1. Escolher aleatoriamente k pontos como centroides iniciais.
2. Atribuir cada ponto ao cluster do centroide mais próximo.
3. Atualizar cada centroide como a média dos pontos do cluster.
4. Repetir (2)–(3) até não haver mais mudanças significativas.

k-means: pontos positivos e negativos

Pontos positivos

- Relativamente eficiente e escalável:

$$\mathcal{O}(tkn), \quad t = \text{iterações}, \; k = \text{clusters}, \; n = \text{pontos}$$

- Simples de implementar e interpretar.

Limitações

- Precisa de k **a priori**.
- Sensível a **outliers** e à escolha dos centroides iniciais.
- Assume clusters aproximadamente esféricos (não pega formas arbitrárias).
- Versão básica é inadequada para muitos atributos categóricos.

Variações do k-means

- **k-medoids** (PAM):
  - cada cluster é representado por um objeto real (medoide);
  - mais robusto a outliers.
- **k-modes**:
  - substitui médias por modas para atributos categóricos;
  - usa medidas de dissimilaridade apropriadas.
- **EM / Gaussian Mixture**:
  - generaliza k-means com modelos probabilísticos;
  - atribui a cada ponto uma probabilidade de pertencer a cada cluster.

Escolhendo o número de clusters k

- Não existe regra perfeita; é em parte **subjetivo**.
- Regra prática: $k \approx \sqrt{n/2}$ (onde n é o número de pontos).
- **Método do cotovelo**:
  - calcular a soma da variância intra-cluster para vários valores de k;
  - escolher o k a partir do qual a redução adicional é pequena (*"cotovelo"* da curva).

Ideia geral do algoritmo EM

- Problema clássico: mistura de distribuições (por exemplo, duas moedas com probabilidades diferentes).
- EM começa com uma estimativa inicial dos parâmetros (médias, covariâncias, pesos).
- Itera dois passos:

  E-step: calcula, para cada ponto, a probabilidade de pertencer a cada componente.

  M-step: atualiza os parâmetros das distribuições usando essas probabilidades.
- Converge para um máximo local da verossimilhança.

Clusterização hierárquica

- **Aglomerativa** (bottom-up):
  - cada ponto começa como um cluster;
  - clusters mais próximos vão sendo fundidos;
  - exemplo: AGNES (Agglomerative Nesting).
- **Divisiva** (top-down):
  - começa com um único cluster contendo todos os pontos;
  - vai dividindo em clusters menores;
  - exemplo: DIANA (Divisive Analysis).
- Resultado: **dendrograma**, que pode ser cortado em diferentes níveis de k.

Hierárquicos: pontos fortes e fracos

Pontos fortes

- Não precisam de k a priori (basta escolher o corte no dendrograma).
- Podem ser combinados com métodos não hierárquicos.
- Capturam estruturas em múltiplas escalas.

Pontos fracos

- Complexidade típica: $\mathcal{O}(n^2)$ (não escalável para n muito grande).
- Uma vez feito um merge/split, não há “undo”.

DBSCAN – visão geral

- **DBSCAN** é um método baseado em densidade.
- Parâmetros:
  - ε (Eps): raio de vizinhança;
  - `MinPts`: número mínimo de pontos dentro de ε.
- Tipos de pontos:
  - **core point**: possui pelo menos `MinPts` vizinhos em ε;
  - **border point**: está perto de um core, mas não tem densidade própria;
  - **noise point**: não é core nem border.
- Clusters são conjuntos máximos de pontos densamente conectados.

DBSCAN – algoritmo básico

1. Escolher um ponto ainda não visitado p.
2. Encontrar todos os pontos na vizinhança ε de p.
3. Se p for core, criar um novo cluster e expandi-lo com todos os pontos densamente conectados.
4. Se p for border ou noise, marcar e seguir para o próximo ponto.
5. Repetir até que todos os pontos tenham sido processados.

- Encontra clusters de **forma arbitrária**, robusto a ruídos/outliers.

Por que validar clusters?

- Ao contrário da classificação, não temos rótulos para checar acurácia.
- Precisamos de medidas para:
  - evitar interpretar *ruído* como padrão;
  - comparar algoritmos de clusterização;
  - escolher parâmetros (por exemplo, k).
- Em muitos casos, a decisão final envolve também **especialistas do domínio**.

Medidas internas: coesão e separação

- **Coesão**: quão próximos estão os objetos de um mesmo cluster.
  - exemplo: soma dos erros quadráticos dentro dos clusters

$$\text{WSS} = \sum_{i=1}^{k} \sum_{p \in C_i} \|p - m_i\|^2$$

- **Separação**: quão distante está um cluster dos outros.
  - exemplo: soma de quadrados entre clusters

$$\text{BSS} = \sum_{i=1}^{k} |C_i| \, \|m_i - m\|^2$$

  onde m é o centro global.
- Um bom particionamento deve ter **alta coesão** e **boa separação**.

Resumo e próximos passos

- Clusterização agrupa objetos semelhantes *sem* rótulos prévios.
- Escolhas críticas:
  - tipo de dado, normalização, medida de similaridade;
  - algoritmo (k-means, hierárquicos, DBSCAN, EM, etc.);
  - número de clusters ou parâmetros de densidade.
- Validação de clusters combina:
  - medidas internas (coesão/separação);
  - conhecimento de especialistas.
- Próximo passo: aplicar esses métodos a **dados ambientais reais** (chuva, uso da terra, qualidade da água, etc.).