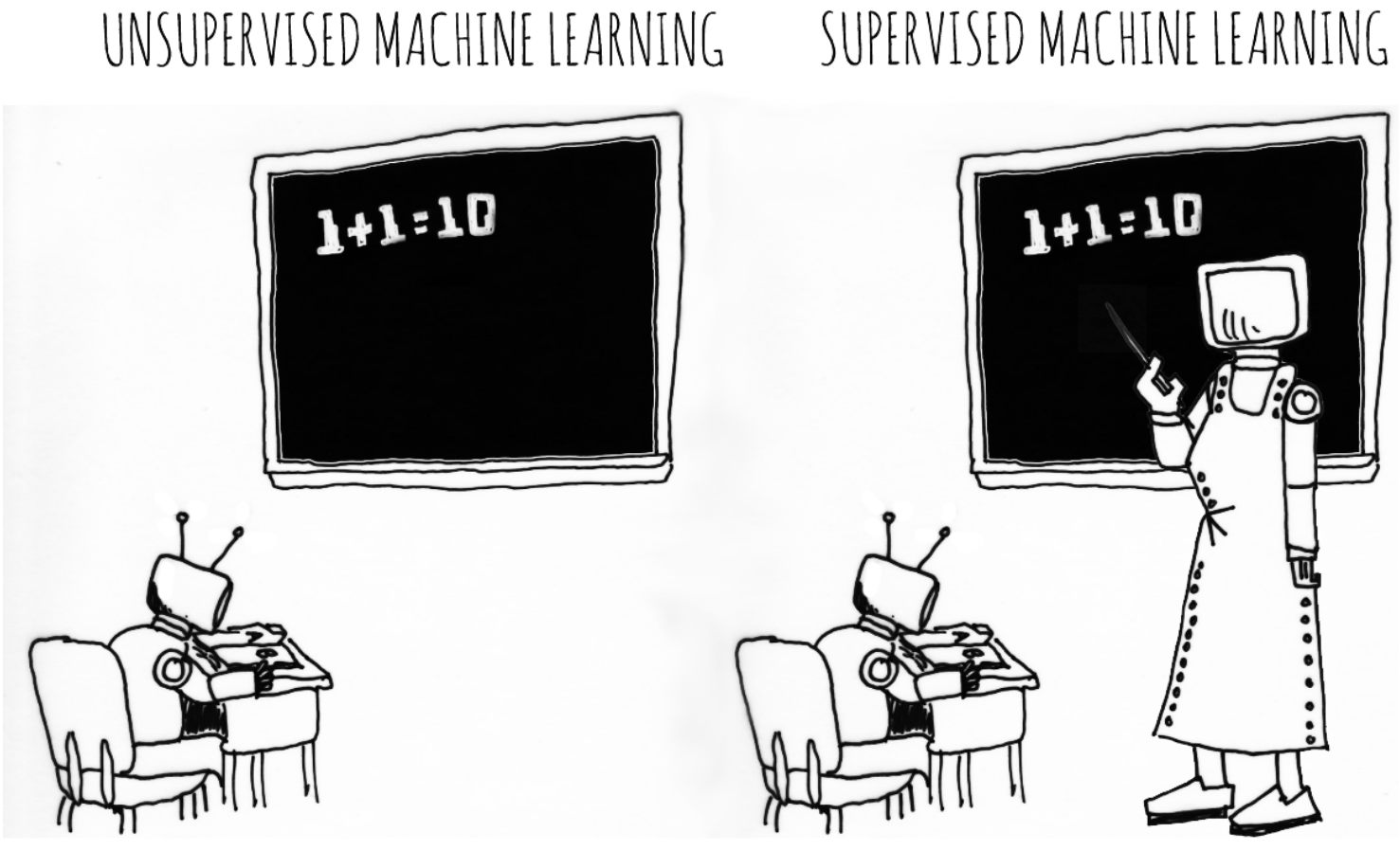
Universidade Federal do Paraná

Laboratório de Estatística e Geoinformação - LEG

Aprendizado não supervisionado

Eduardo Vargas Ferreira

Supervisionado *vs* não supervisionado



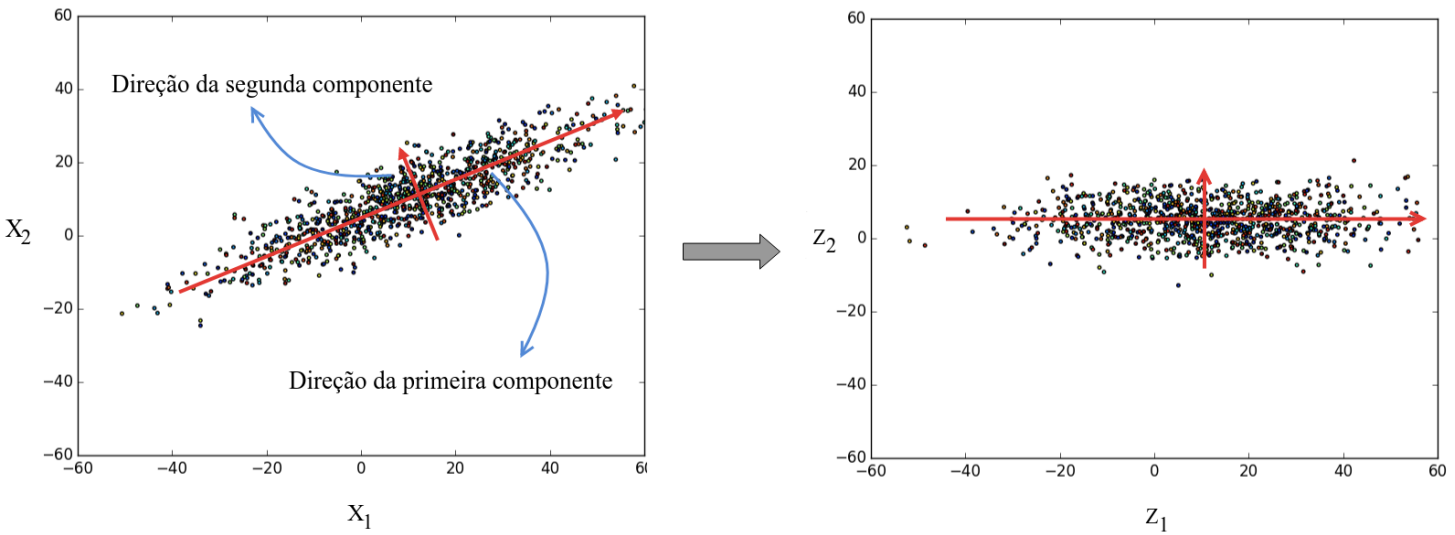
Fonte: Proofreader's Whimsy

Análise de Componentes Principais (ACP)

- A **primeira componente principal** de um conjunto de características $X_1, X_2, \ldots, X_p$ é a combinação linear normalizada ($\sum_{j=1}^{p} \phi_{j1}^2 = 1$)

$$\begin{aligned} Z_1 &= \phi_{11}X_1 + \phi_{21}X_2 + \ldots + \phi_{p1}X_p \\ &= \boldsymbol{\phi}_1^t \boldsymbol{X} \end{aligned}$$

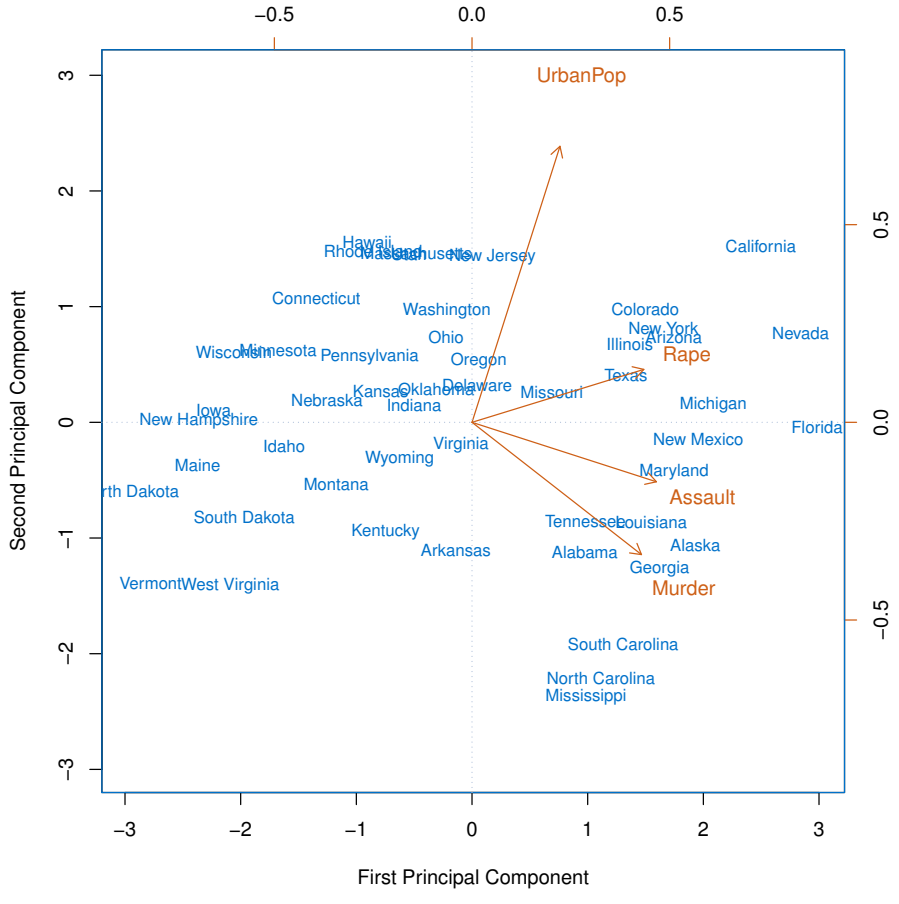
- Que maximiza a $Var(Z_1) = \phi_1^t \Sigma \phi_1$, com Σ = matriz de covariância de $\boldsymbol{X}$;



- A **segunda componente principal** $\phi_2^t \boldsymbol{X}$ maximiza $Var(\phi_2^t \boldsymbol{X})$, sujeito a restrição $\phi_2^t \phi_2 = 1$ e $Cov(\phi_1^t \boldsymbol{X}, \phi_2^t \boldsymbol{X}) = 0$.

Exemplo: `USAarrests` data

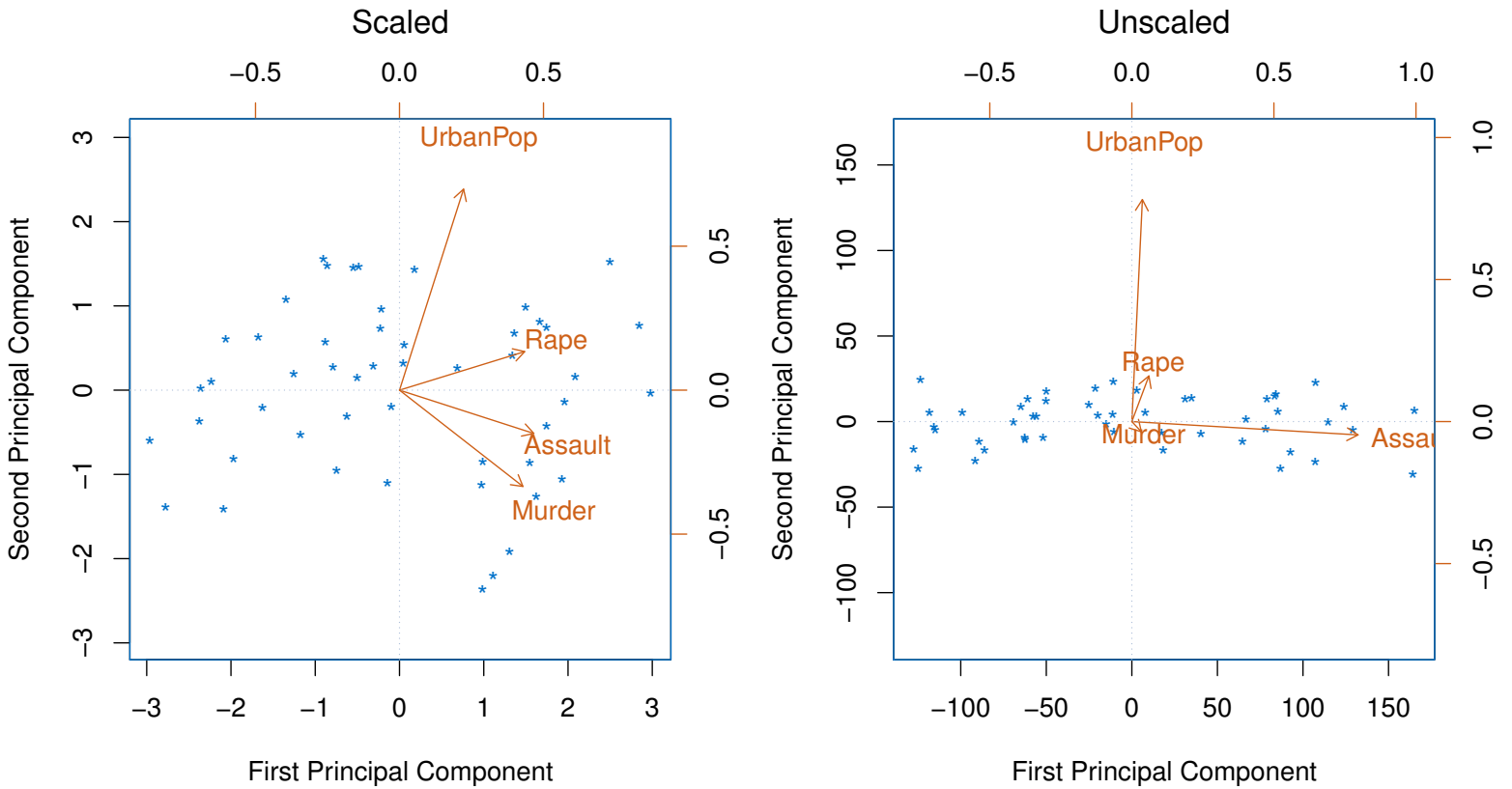
- Os dados contém o número de prisões por 100.000 residentes nos 50 estados dos Estados Unidos;



- $Z_1 = \phi_{1i}\mathbf{UrbanPop} + \phi_{2i}\mathbf{Rape} + \phi_{3i}\mathbf{Assault} + \phi_{4i}\mathbf{Murder}$

Escalando as variáveis

- Se as variáveis estão em diferentes unidades é recomendável escalar cada uma para se ter um desvio padrão igual a 1.



Cálculo das componentes principais

- A *k*-ésima **componente principal** de um conjunto de características $X_1, X_2, \ldots, X_p$ é a combinação linear

$$Z_k = \phi_{1k} X_1 + \phi_{2k} X_2 + \ldots + \phi_{pk} X_p,$$

- Que maximiza a $Var(\phi_k^t \boldsymbol{X})$, i.e.:

$$\underset{\phi_{1k},\ldots,\phi_{pk}}{argmax} \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{p} \phi_{jk} x_{ij} \right)^2, \quad sujeito\ a \begin{cases} \phi_k^t \phi_k = 1 \\ Cov(\phi_g^t \boldsymbol{X}, \phi_k^t \boldsymbol{X}) = 0, \forall g < k \end{cases}$$

- Lembrando que $Var(Z) = E(Z^2) - [E(Z)]^2$ e $E(X_i) = 0, \forall i \in \{1, \ldots, p\}$.

Proporção da variância explicada

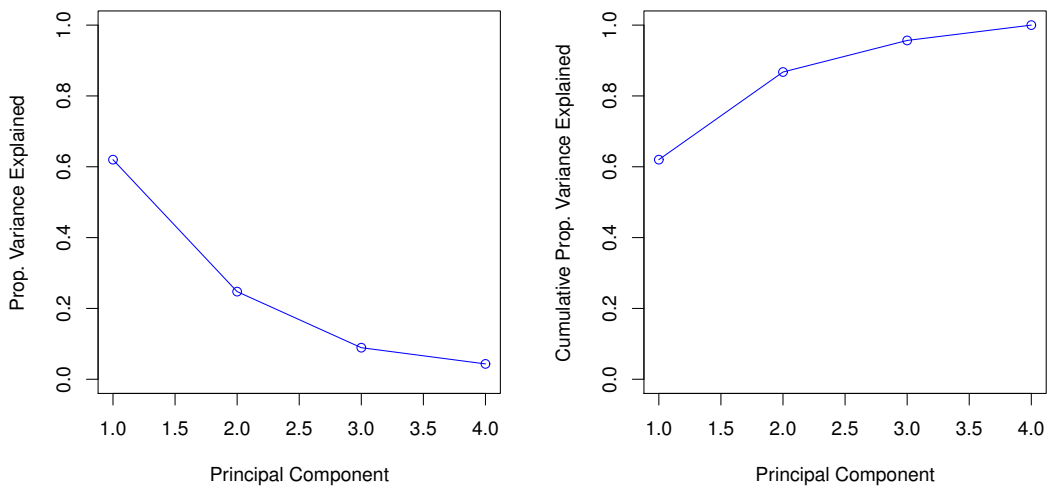
- A variância total presente nos dados é definida como

$$\sum_{j=1}^{p} Var(Z_j) = \lambda_1 + \lambda_2 + \ldots + \lambda_p$$

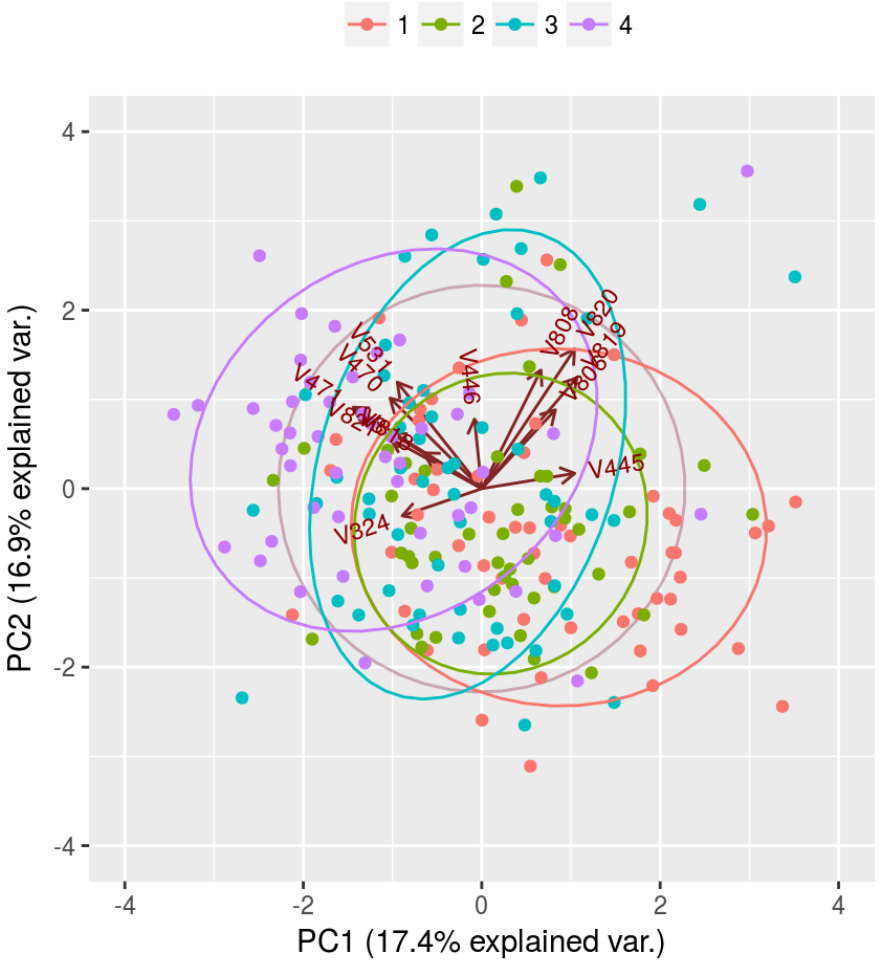
- Assim, a proporção da variância explicada pela j-ésima componente principal é dada por

$$\frac{\lambda_j}{\lambda_1 + \lambda_2 + \ldots + \lambda_p}$$

- Abaixo, a proporção da variância explicada nos dados `USAarrests`;



Exemplo: clientes em atraso

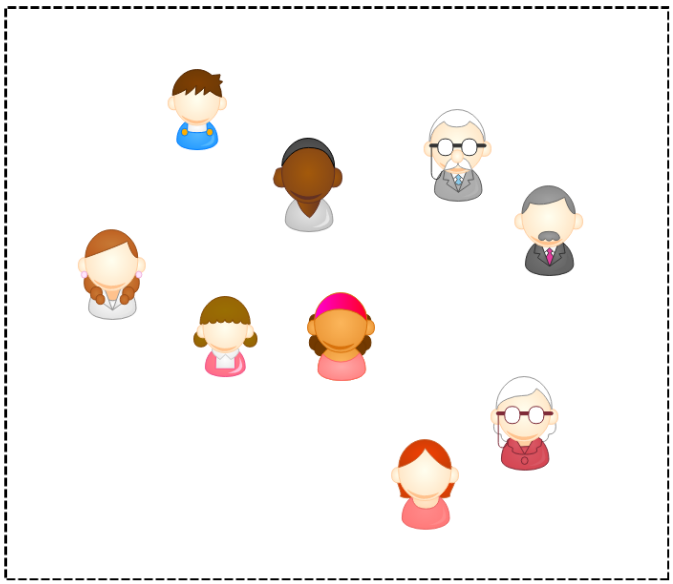


Clustering

Clustering

- **Clustering** refere-se ao conjunto de técnicas para encontrar subgrupos (ou *clusters*) a partir dos dados;

- Buscamos partições em grupos distintos, tal que observações dentro de cada grupo sejam similares entre si e diferentes dos demais;



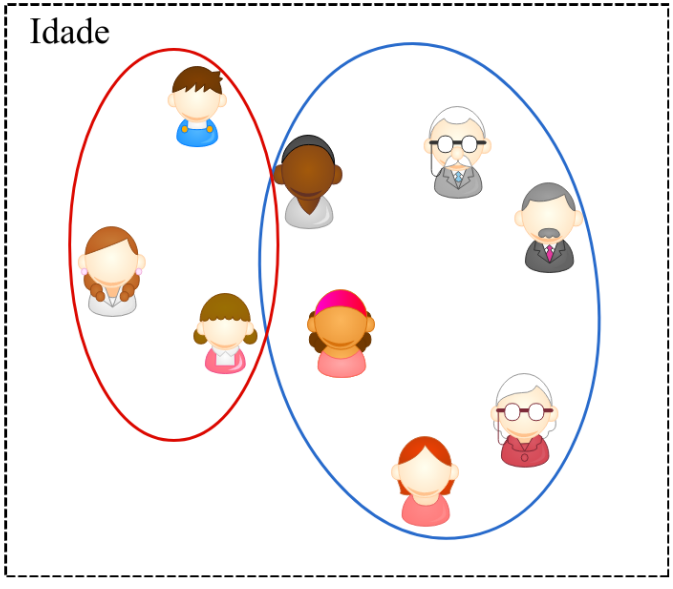
- Veremos três métodos:

1. ***K*-means clustering**;
2. **Hierarchical clustering**;
3. **DBSCAN**.

Clustering

- **Clustering** refere-se ao conjunto de técnicas para encontrar subgrupos (ou *clusters*) a partir dos dados;

- Buscamos partições em grupos distintos, tal que observações dentro de cada grupo sejam similares entre si e diferentes dos demais;



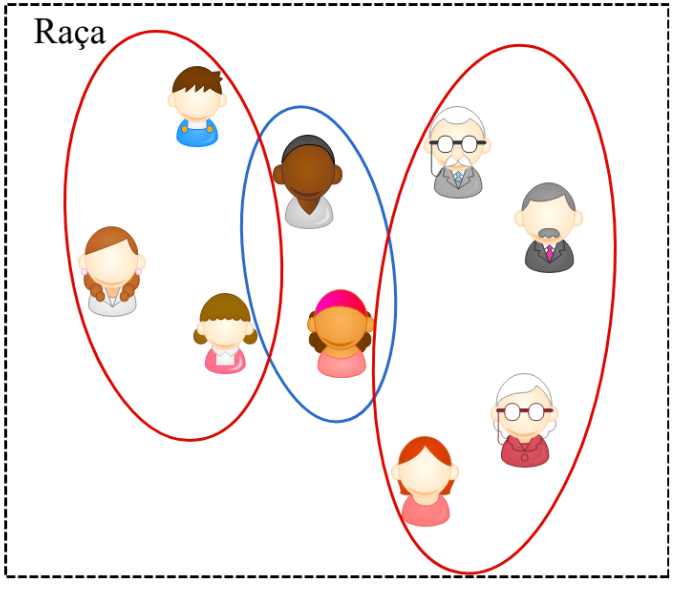
- Veremos três métodos:

1. ***K*-means clustering**;
2. **Hierarchical clustering**;
3. **DBSCAN**.

Clustering

- **Clustering** refere-se ao conjunto de técnicas para encontrar subgrupos (ou *clusters*) a partir dos dados;

- Buscamos partições em grupos distintos, tal que observações dentro de cada grupo sejam similares entre si e diferentes dos demais;



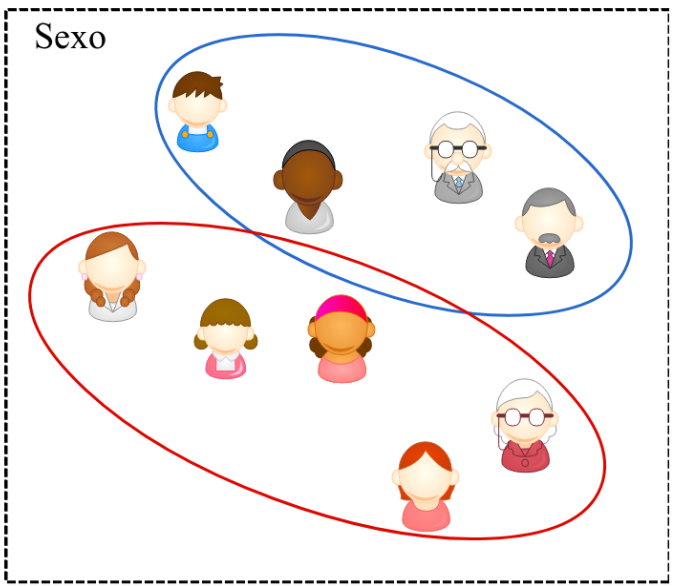
- Veremos três métodos:

1. ***K*-means clustering**;
2. **Hierarchical clustering**;
3. **DBSCAN**.

Clustering

- **Clustering** refere-se ao conjunto de técnicas para encontrar subgrupos (ou *clusters*) a partir dos dados;

- Buscamos partições em grupos distintos, tal que observações dentro de cada grupo sejam similares entre si e diferentes dos demais;



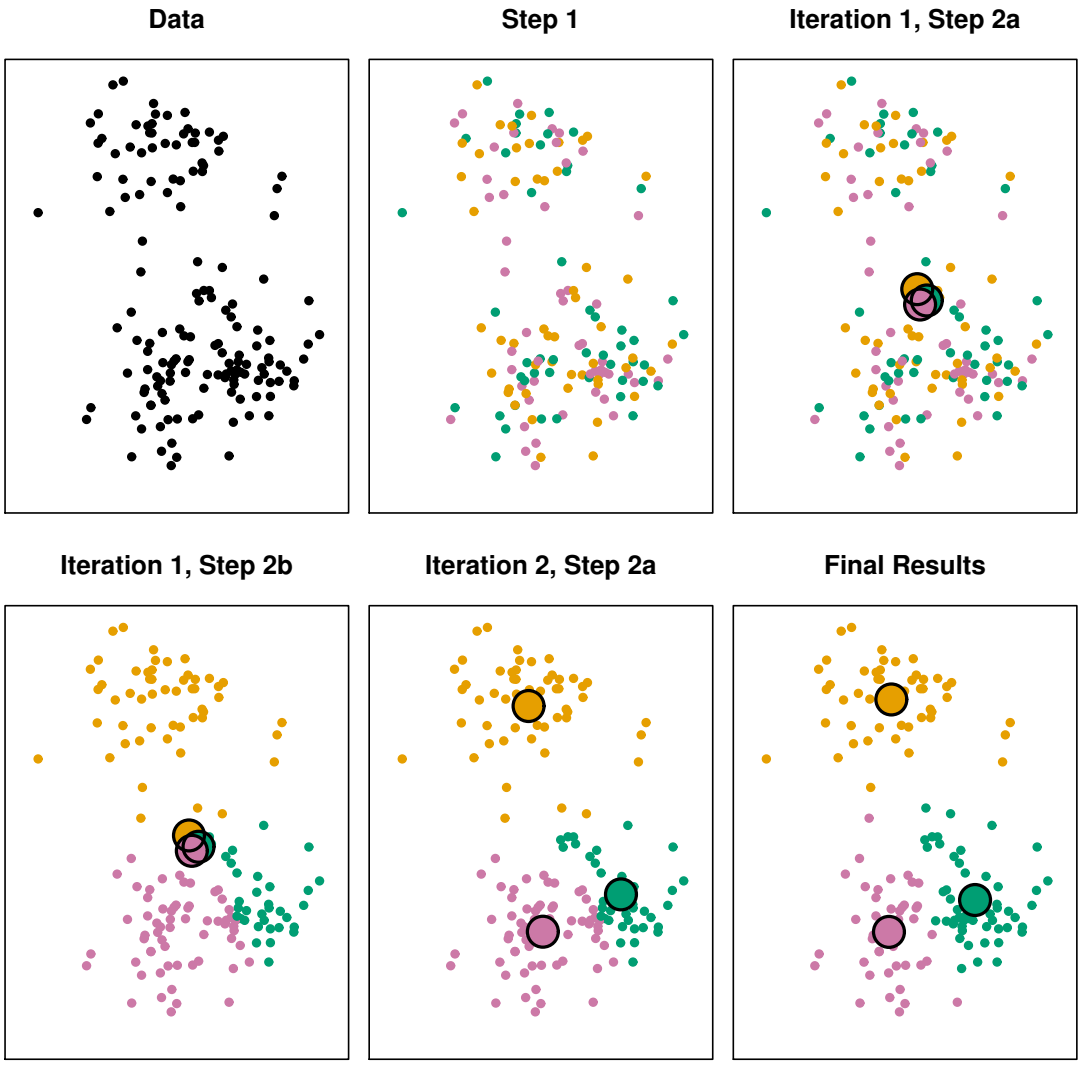
- Veremos três métodos:

1. ***K*-means clustering**;
2. **Hierarchical clustering**;
3. **DBSCAN**.

Clustering

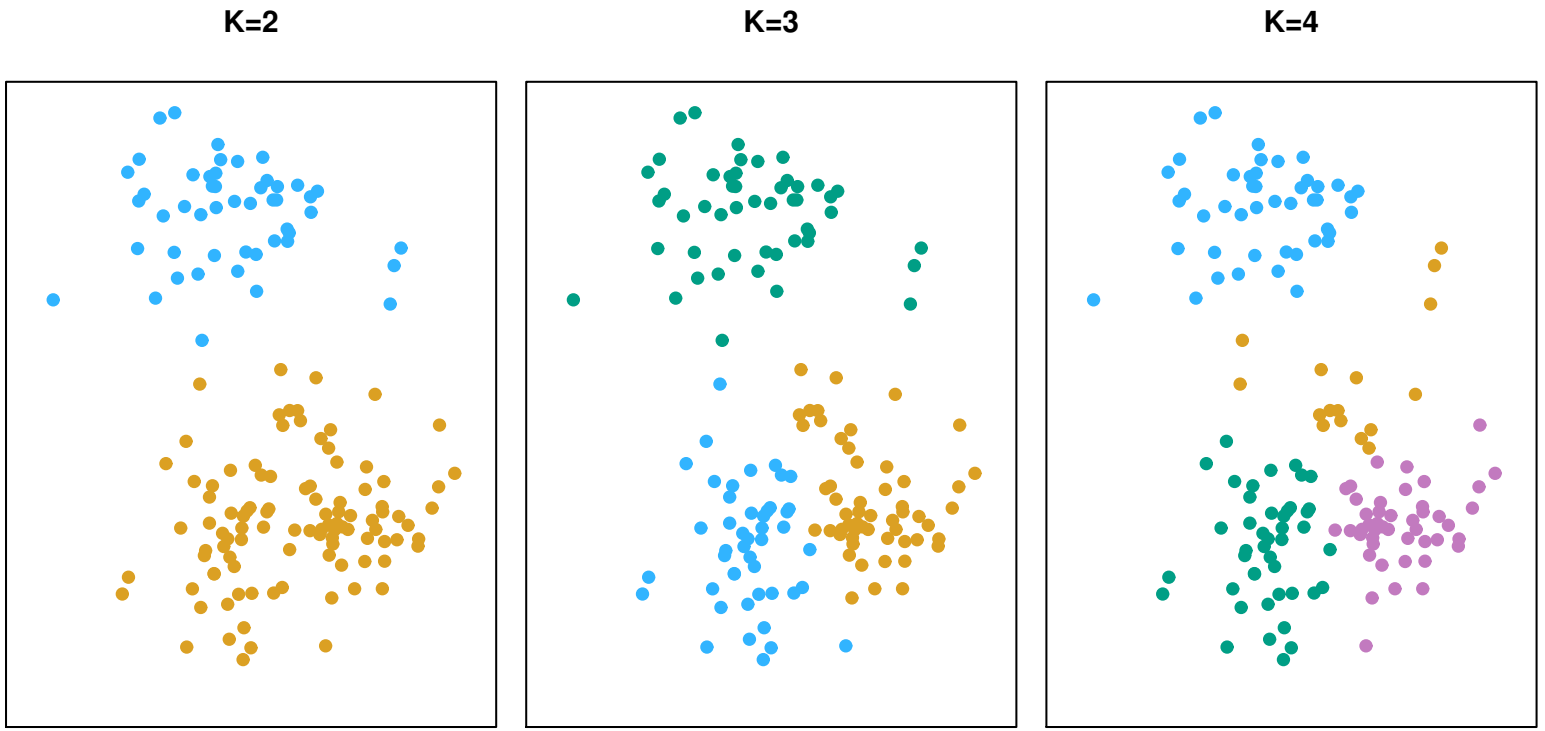
K-means

K-means


Data
Step 1
Iteration 1, Step 2a
Iteration 1, Step 2b
Iteration 2, Step 2a
Final Results

Exemplo simulado

- Os dados simulados consistem em 150 observações. Os painéis representam os resultados de K-means para diferentes K's;



Detalhes do algoritmo

- Seja $C_1, C_2, \ldots, C_K$ respectivos grupos contendo os índices das observações, satisfazendo as seguintes propriedades:

  - $C_1 \cup C_2 \cup \ldots \cup C_K = \{1, \ldots, n\}$. Em outras palavras, cada observação pertence à pelo menos um grupo;

  - $C_k \cap C_{k'} = \emptyset$. Ou seja, as observações não pertencem a mais de um grupo ao mesmo tempo.

- A variação dentro do cluster C_k (*within-cluster variation*) é medida por

$$WCV(C_k) = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2,$$

em que $|C_k|$ denota o número de observações no k-ésimo cluster.

Detalhes do algoritmo

- Sendo assim, queremos resolver o seguinte problema

$$\underset{C_1,\ldots,C_K}{\textit{argmin}} \left\{ \sum_{k=1}^{K} \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 \right\} = \underset{C_1,\ldots,C_K}{\textit{argmin}} \left\{ \sum_{k=1}^{K} 2 \sum_{i \in C_k} \sum_{j=1}^{p} (x_{ij} - \bar{x}_{kj})^2 \right\},$$

em que $\bar{x}_{kj} = \frac{1}{|C_k|} \sum_{i \in C_k} x_{ij}$ é a média da característica j no cluster C_k.

Algoritmo

- **Step 1:** Atribua, aleatoriamente, cada observação em um dos K clusters (este é o chute inicial);
- **Step 2:** Itere até que os clusters se estabilizem:
  (a) Para cada K cluster, calcule seu centroide;
  (b) Atribua cada observação ao cluster mais próximo (menor distância Euclideana).

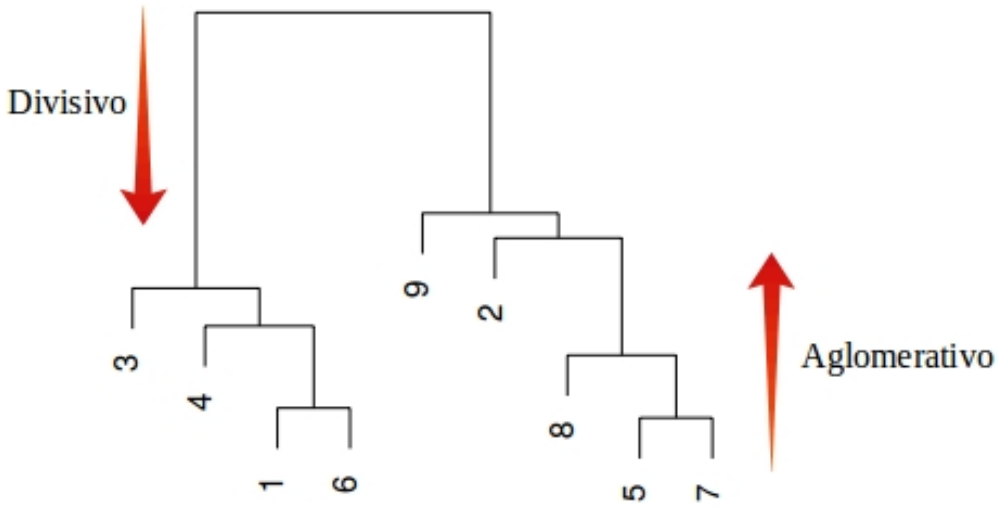
Atenção para o chute inicial

Clustering

Hierarchical

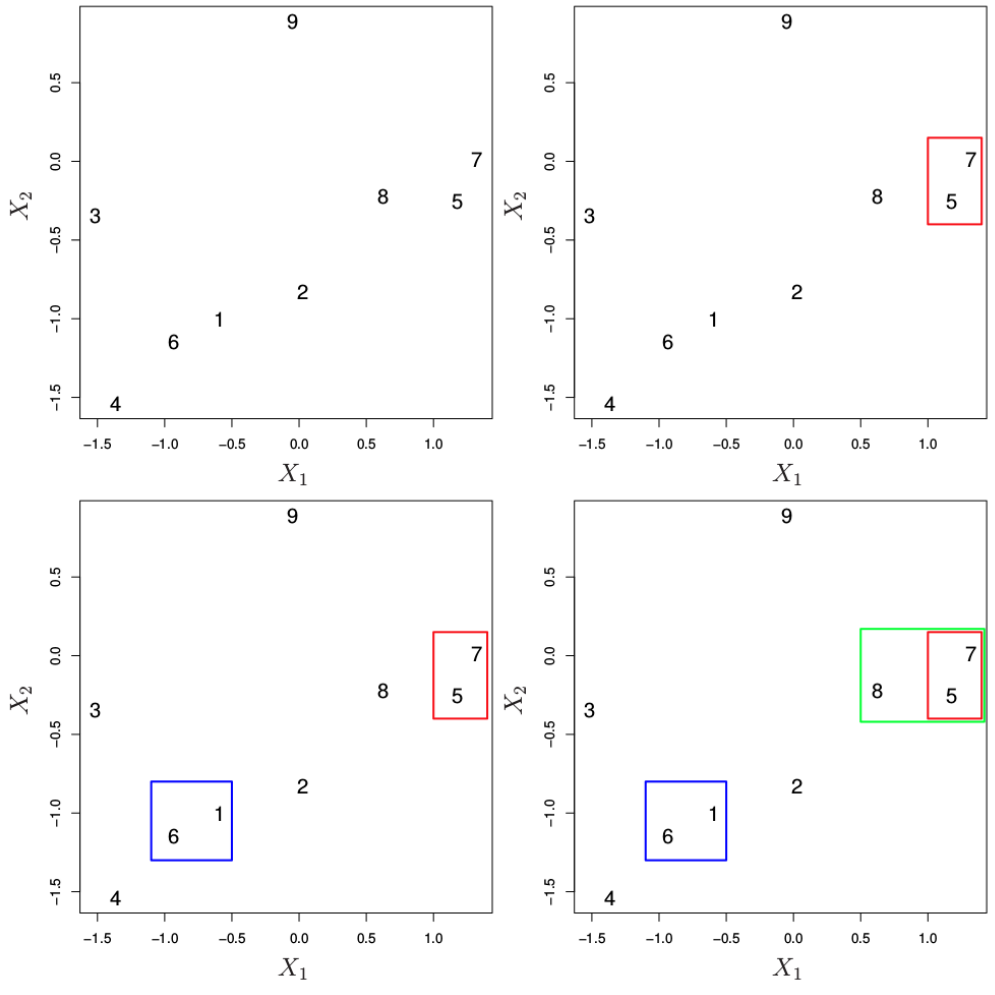
Hierarchical Clustering

- **Hierarchical clustering** é uma abordagem alternativa, que não exige comprometimento com a escolha de K;

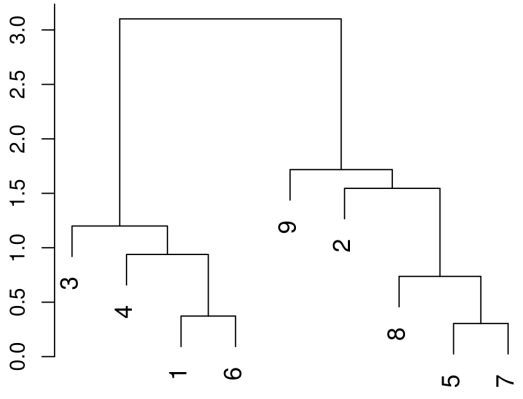


- A ideia do **cluster aglomerativo** é construir um dendrograma com folhas que se agrupam até chegar ao tronco.

Ideia do algoritmo

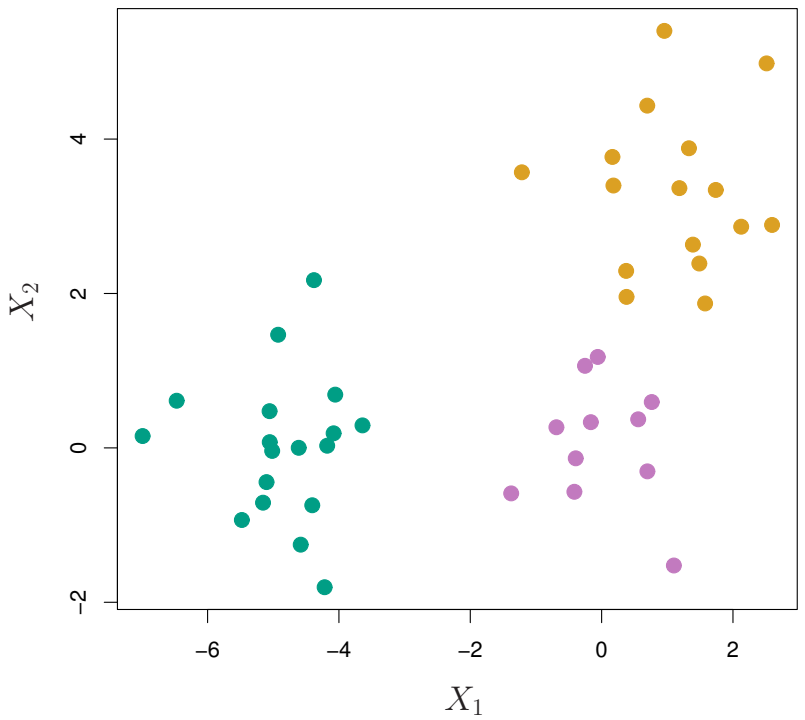


- Iniciamos com cada ponto sendo seu próprio cluster;
- Identificamos os dois clusters mais próximos e os agrupamos;
- Repetimos este processo até restar um cluster.



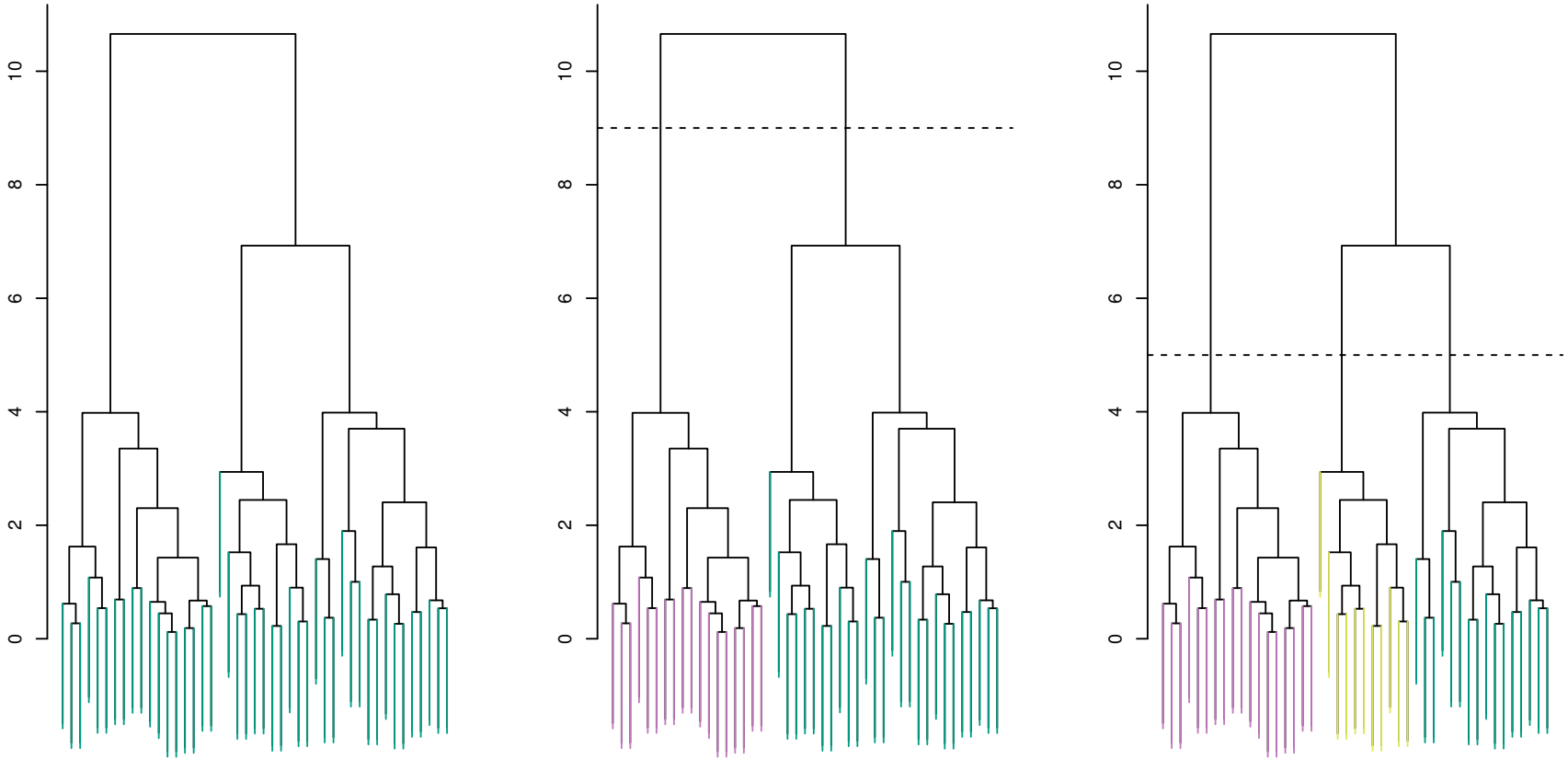
Exemplo

- Temos 45 observações, e 3 classes distintas (separadas por cores);

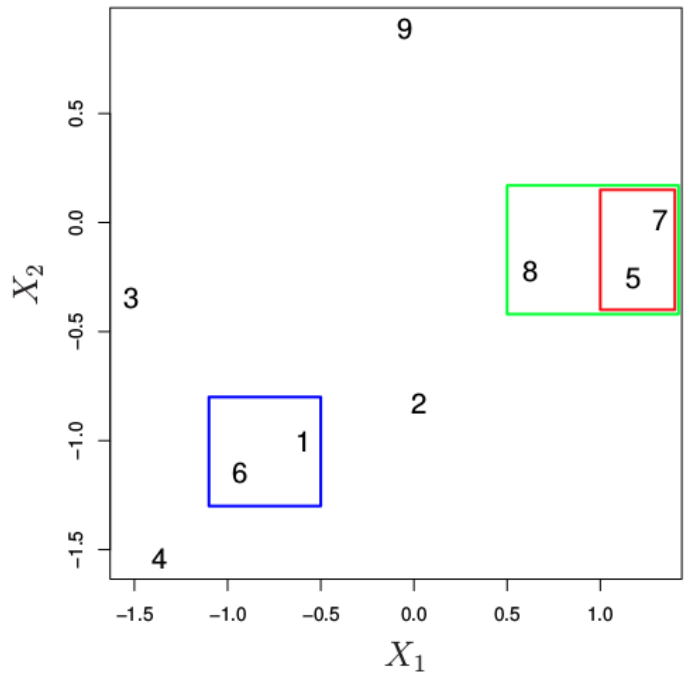


Exemplo

- Abaixo, três dendrogramas com diferentes alturas de corte (que resulta em clusters distintos);



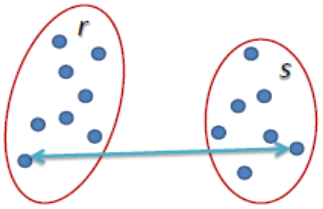
Tipo de Linkage



Tipo de Linkage

Complete

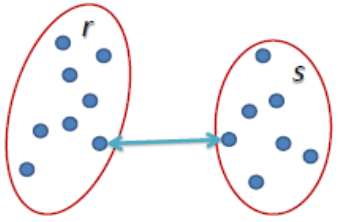
- Calculamos a máxima dissimilaridade entre os clusters.



$$L(r,s) = \max(D(x_{ri}, x_{sj}))$$

Single

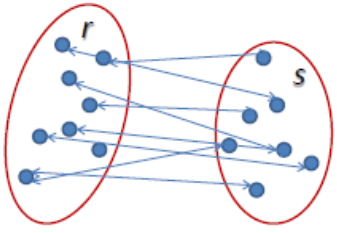
- Calculamos a mínima dissimilaridade entre os clusters.



$$L(r,s) = \min(D(x_{ri}, x_{sj}))$$

Average

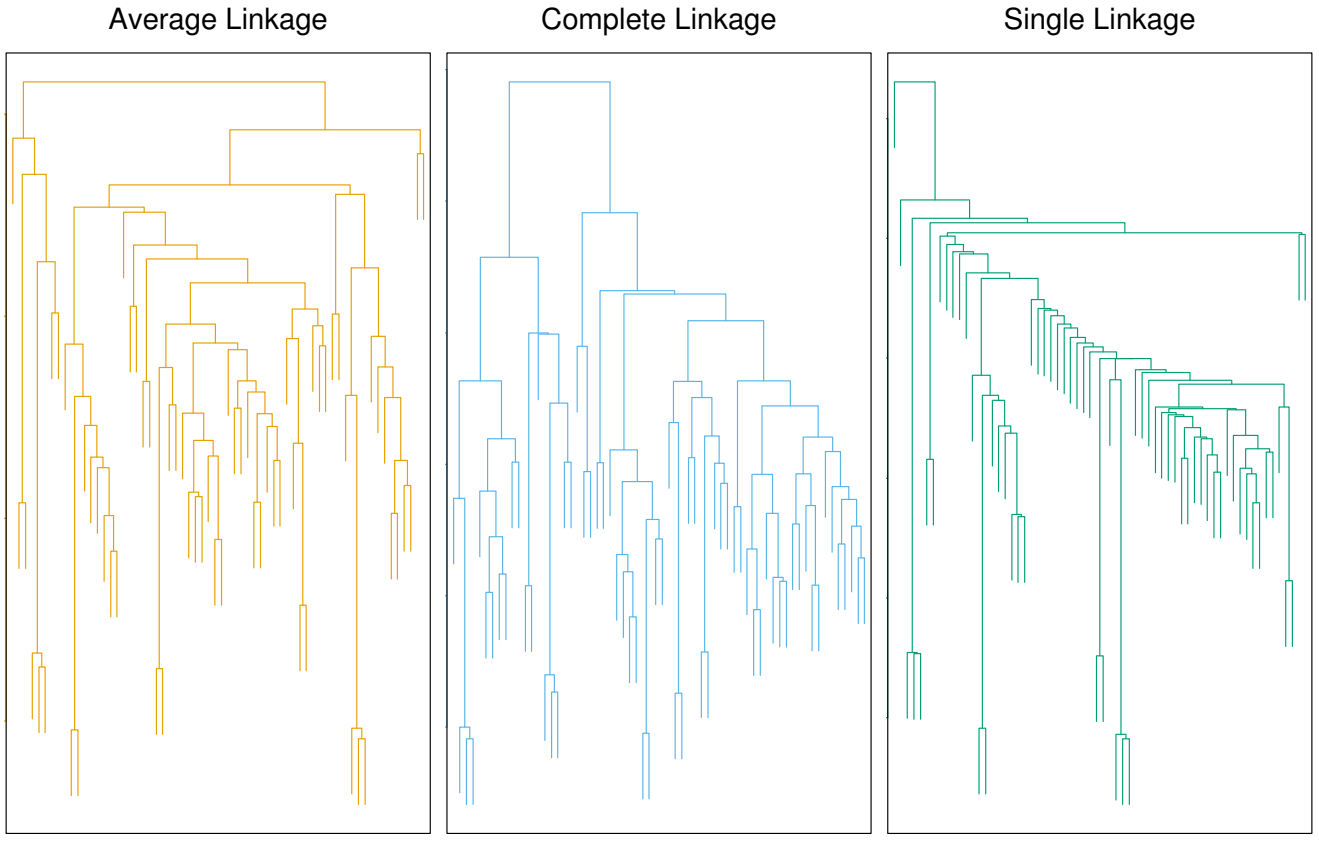
- Calculamos a dissimilaridade média entre os clusters.



$$L(r,s) = \frac{1}{n_r n_s} \sum_{i=1}^{n_r} \sum_{j=1}^{n_s} D(x_{ri}, x_{sj})$$

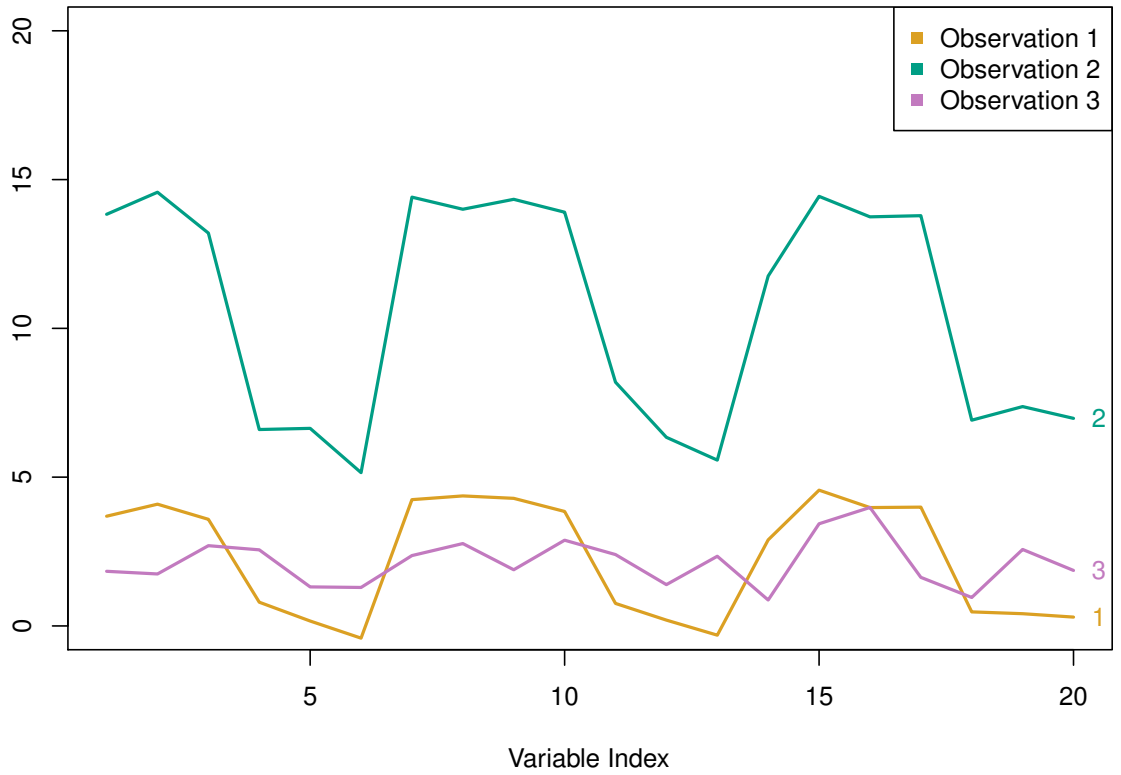
Exemplo

- Em geral, **average** e **complete** linkage tendem a produzir agrupamentos mais equilibrados.



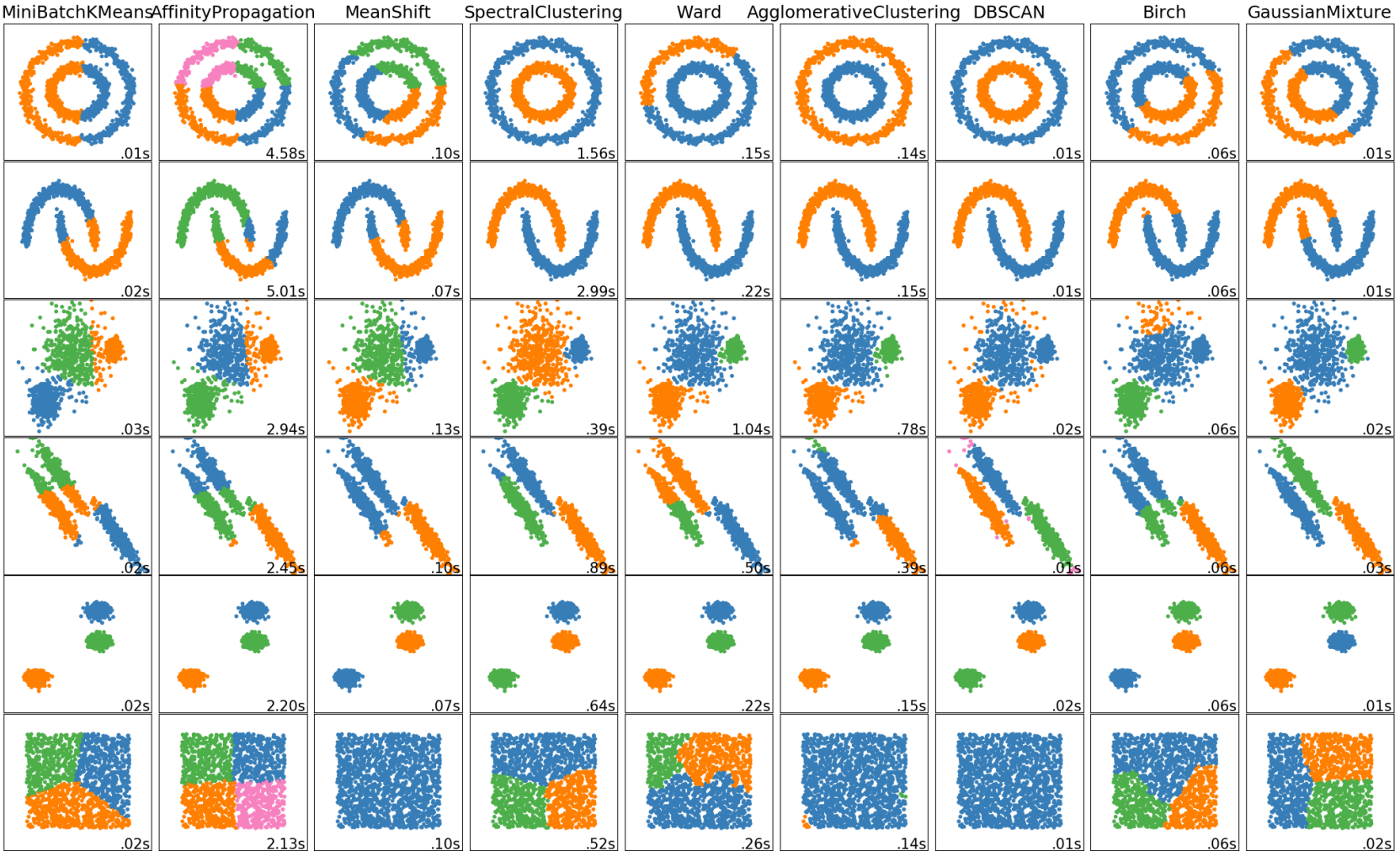
Correlation-based distance

- Trata-se de alternativa para a distância Euclideana.



- Ela considera duas observações similares se suas características são altamente correlacionadas.

Outros tipos de clusters

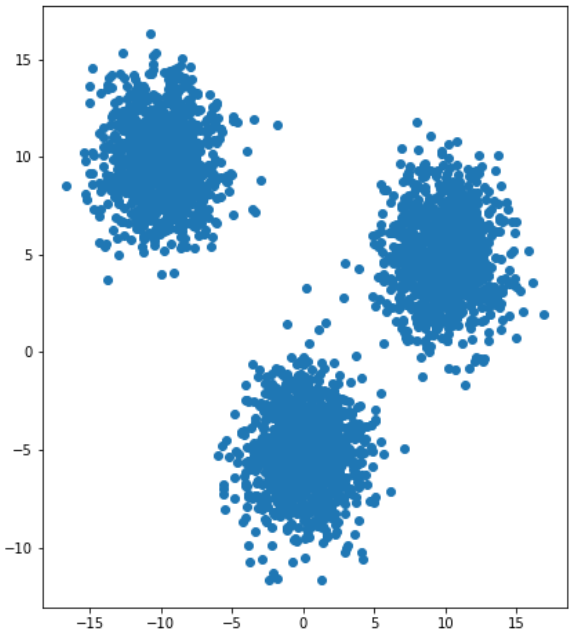


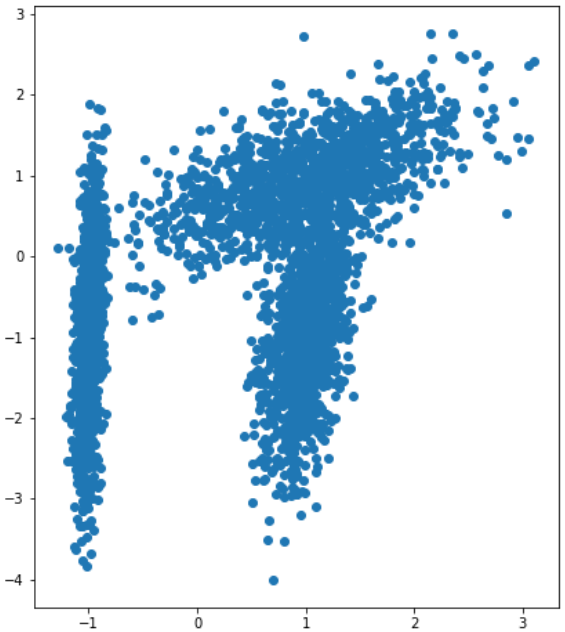
Clustering

DBSCAN

DBSCAN

- Os métodos que vimos anteriormente são adequados para encontrar agrupamentos esféricos, em regiões bem definidas e ausentes de outliers.

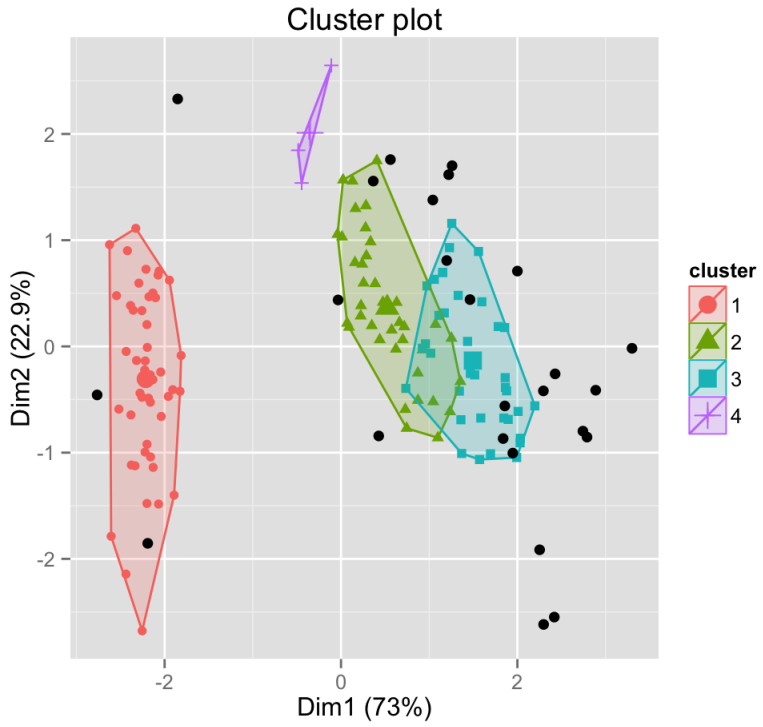




- Entretanto, no mundo real, os clusters podem ter formas arbitrárias (oval, em forma de "S" etc.), e virem com outliers e ruídos.

DBSCAN

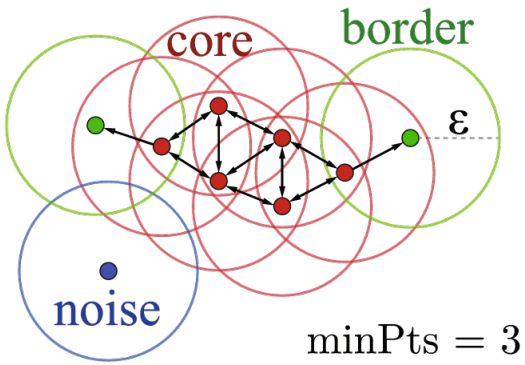
- DBSCAN (*Density-Based Spatial Clustering and Application with Noise*) é um algoritmo de cluster baseado em densidade;



Clusters são regiões densas, separadas por regiões de menor densidade.

Como o método funciona

- Temos dois parâmetros de *tuning*:
  - ⋆ `eps`: que define o raio, ϵ, em torno do ponto x;
  - ⋆ `MinPts`: número mínimo de vizinhos dentro do raio ϵ.



- Qualquer ponto x, com uma quantidade de vizinhos maior ou igual a `MinPts` é considerado *core*;
- O ponto pertence a fronteira se o número de vizinhos $<$ `MinPts`, mas está contido no raio de algum *core*;
- Finalmente, se o ponto não é interior nem de fronteira, ele é considerado como ruído ou outlier.

Prós e contras do método

Vantagens

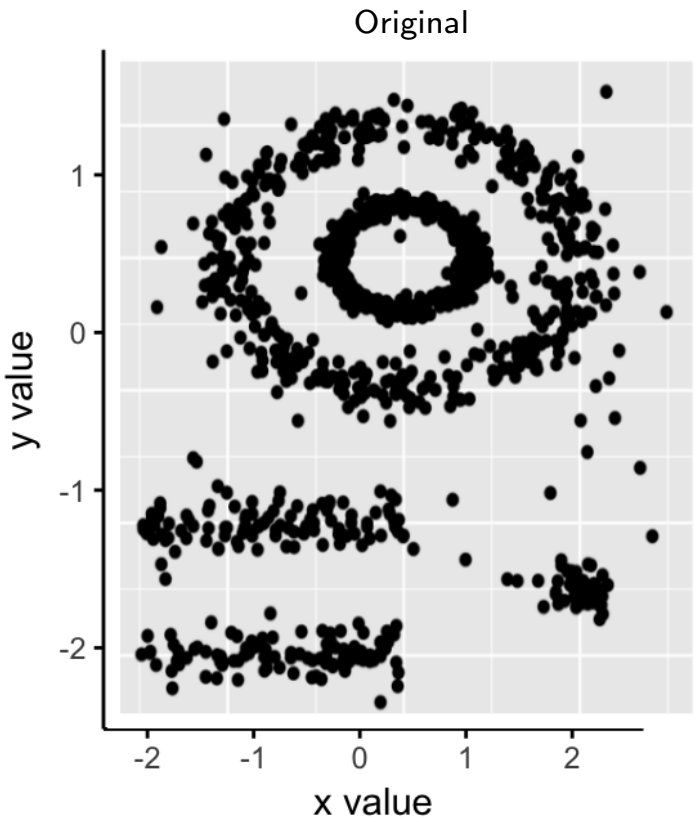
- Não requer um número predefinido de *clusters*;
- Podem ser de qualquer forma, incluindo não esféricos;
- A técnica é capaz de identificar dados de ruído (outliers).

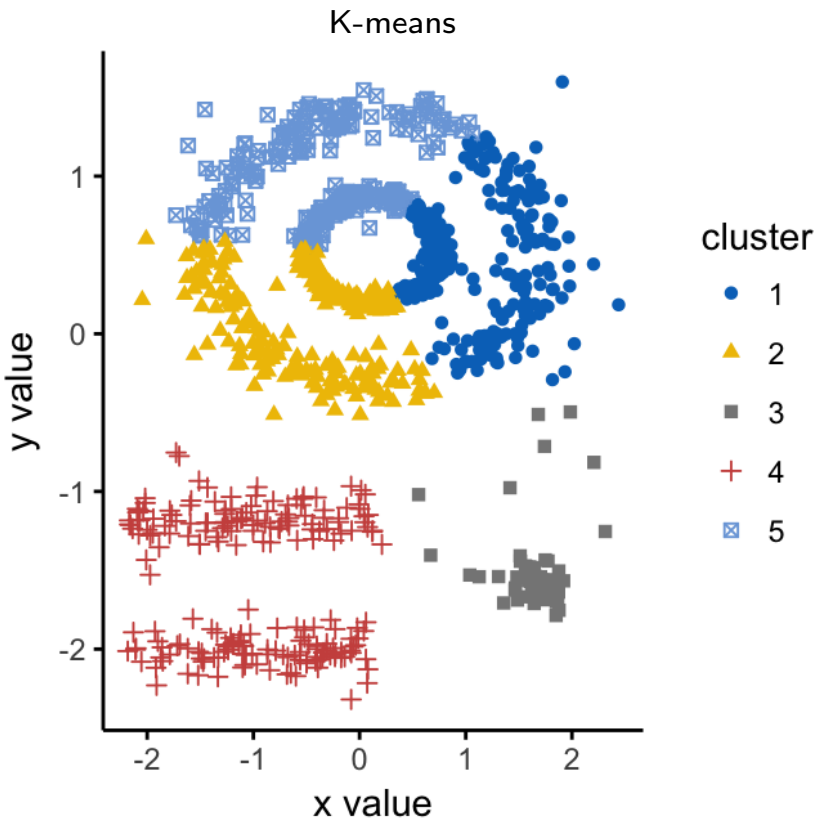
Desvantagem

- Pode falhar se não houver queda de densidade entre clusters;
- É sensível aos parâmetros que definem a densidade de *tuning*;
- A configuração adequada pode exigir conhecimento e domínio.

K-means *vs* DBSCAN

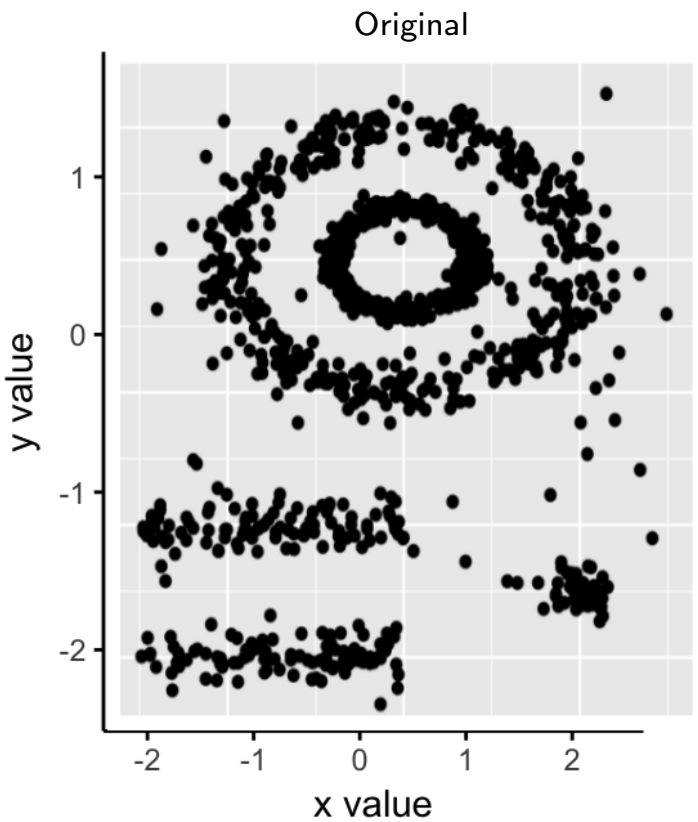
- No exemplo abaixo, comparamos DBSCAN com o *k*-means, através de um conjunto de dados simulados.

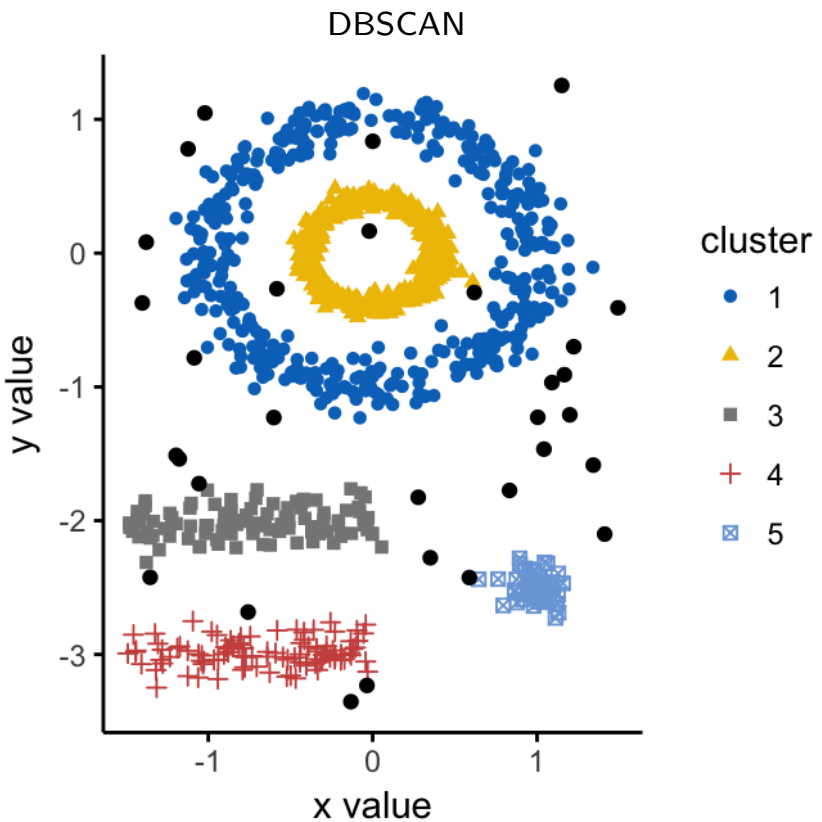




K-means *vs* DBSCAN

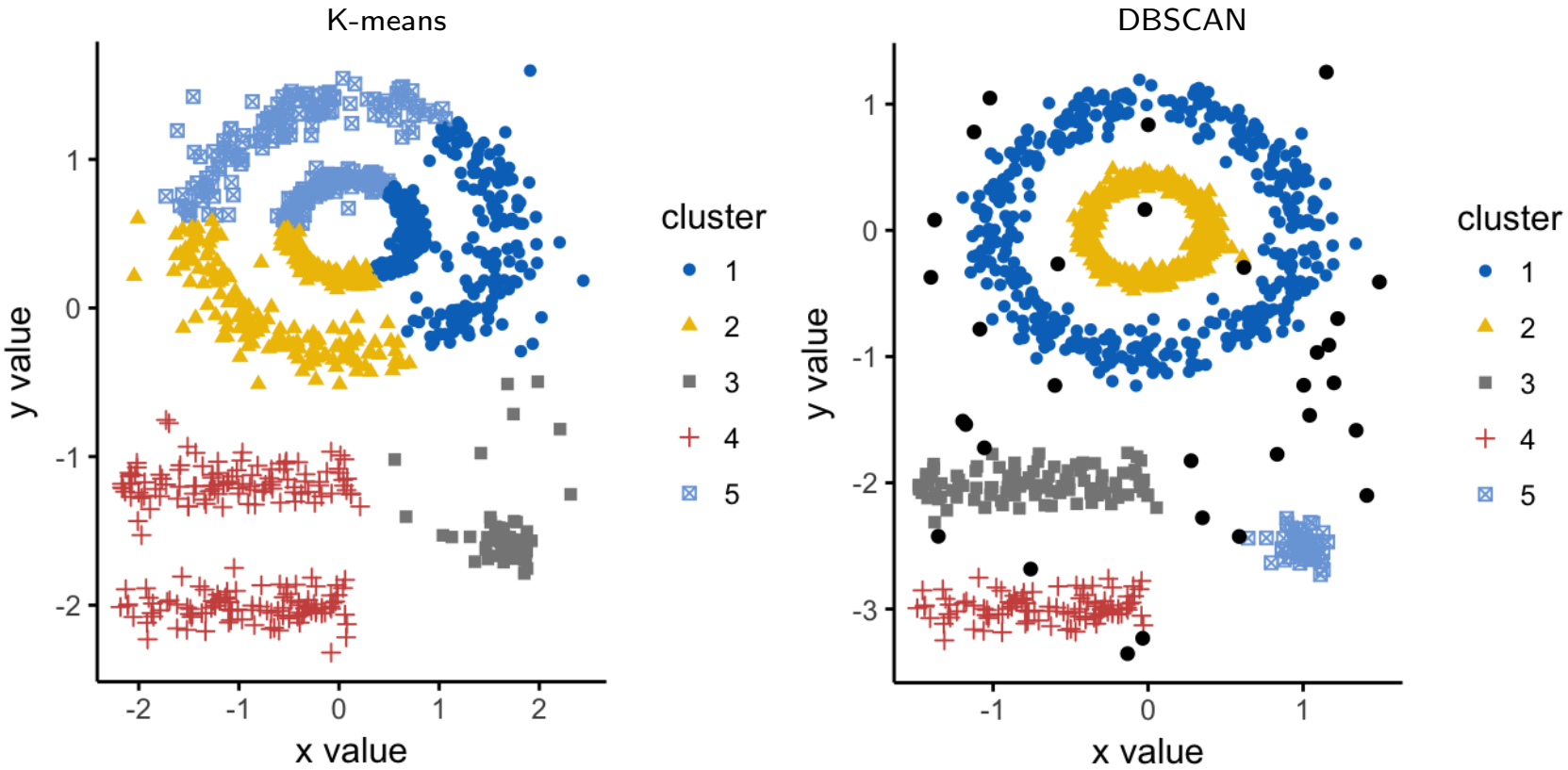
- No exemplo abaixo, comparamos DBSCAN com o *k*-means, através de um conjunto de dados simulados.



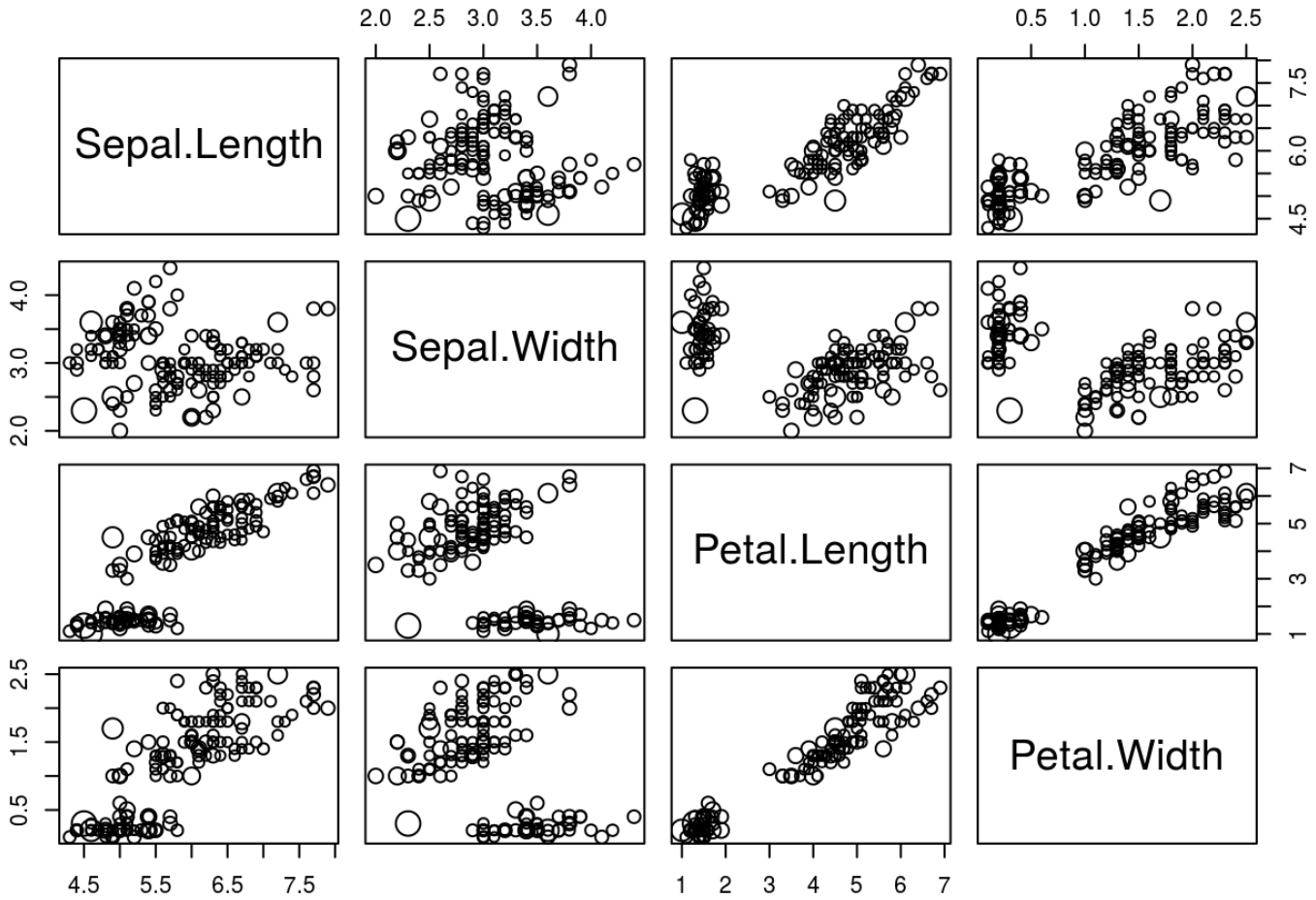


K-means *vs* DBSCAN

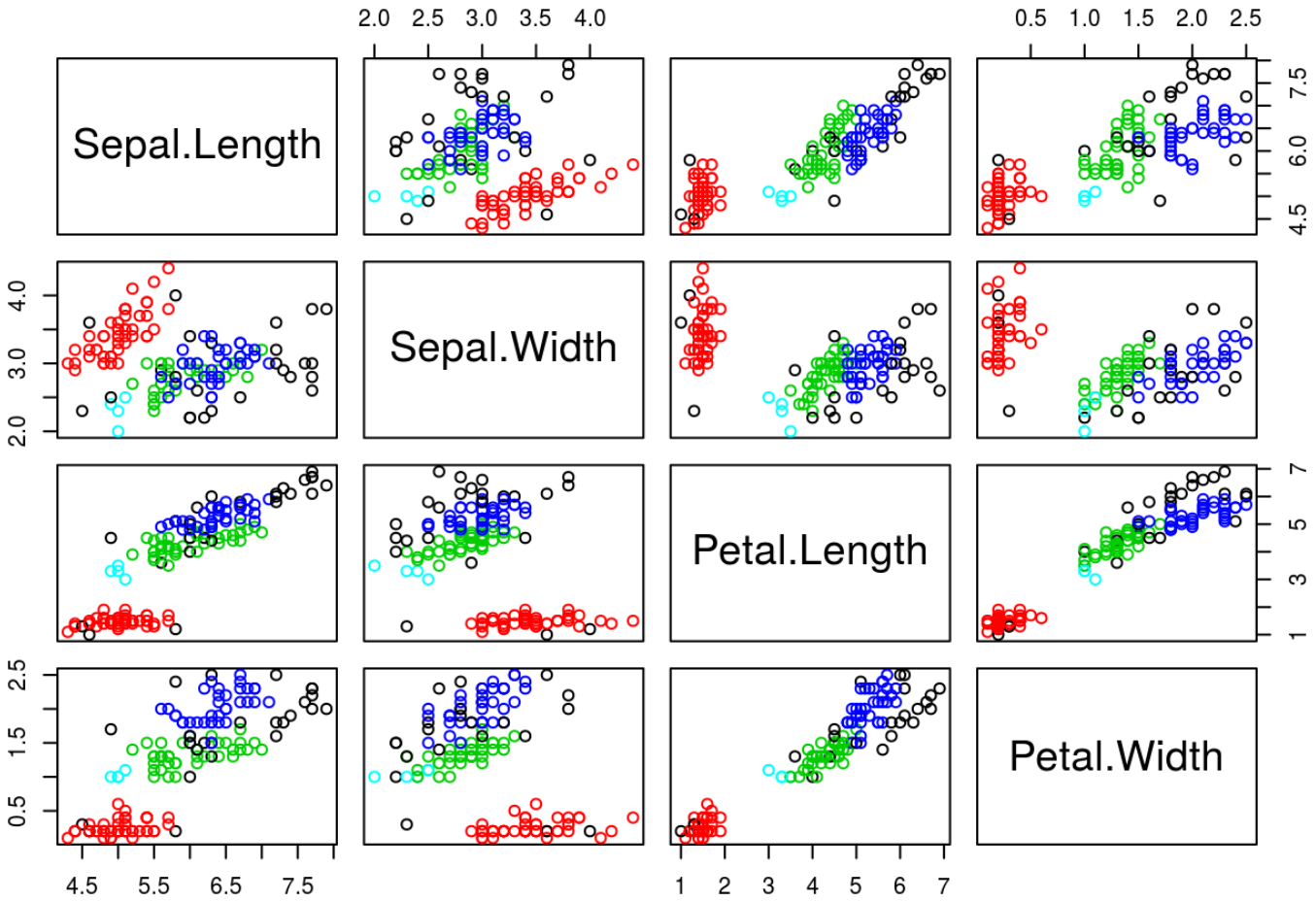
- No exemplo abaixo, comparamos DBSCAN com o *k*-means, através de um conjunto de dados simulados.



Exemplo `Iris`



Exemplo `Iris`



Referências

- James, G., Witten, D., Hastie, T. e Tibshirani, An Introduction to Statistical Learning, 2013;
- Hastie, T., Tibshirani, R. e Friedman, J., The Elements of Statistical Learning, 2009;
- Lantz, B., Machine Learning with R, Packt Publishing, 2013;
- Tan, Steinbach, and Kumar, Introduction to Data Mining, Addison-Wesley, 2005;
- Some of the figures in this presentation are taken from "An Introduction to Statistical Learning, with applications in R"(Springer, 2013) with permission from the authors: G. James, D. Witten, T. Hastie and R. Tibshirani