SCC0276 - Aprendizado de Máquina

K-MEANS

Profa. Dra. Roseli Aparecida Francelin Romero
SCC - ICMC - USP

2019

Sumário

Agrupamento Não-supervisionado

- **Meta**: particionar automaticamente dados **não rotulados** em grupos de pontos de dados semelhantes.
- **Pergunta**: Quando e por que queremos fazer isso? Útil para:
- Organização automática de dados.
- Compreender a estrutura escondida nos dados (Representar dados de alta dimensão em um espaço de baixa dimensão (por exemplo, para fins de visualização).

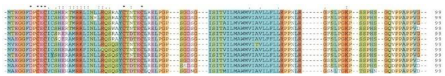
Aplicações - por toda parte

- Agrupar artigos de notícias ou páginas da web ou resultados de pesquisa por tópico.



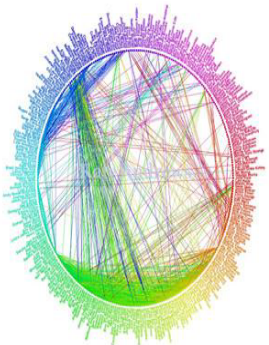

Aplicações - por toda parte

- Agrupar sequências de proteínas por função ou genes de acordo com o perfil de expressão

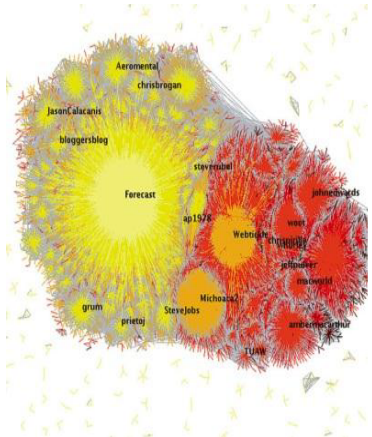


Aplicações - por toda parte

- Agrupar usuários de redes sociais por interesse (detecção de comunidade).



Facebook



Twitter

Aplicações - por toda parte

- Agrupar Clientes de acordo com o histórico de compras.
- Galáxias de aglomeração ou estrelas próximas (por exemplo, Sloan Digital Sky Survey)
- e mais e mais aplicações

Objetivo

- **Entrada**: Um conjunto S de n pontos, também uma medida de distância / dissimilaridade, especificando a distância d (x, y) entre pares (x, y). Por exemplo, # palavras-chave em comum, editar distância, wavelets, coef. Etc.
- **Meta**: saida é a partição dos dados.

Métodos

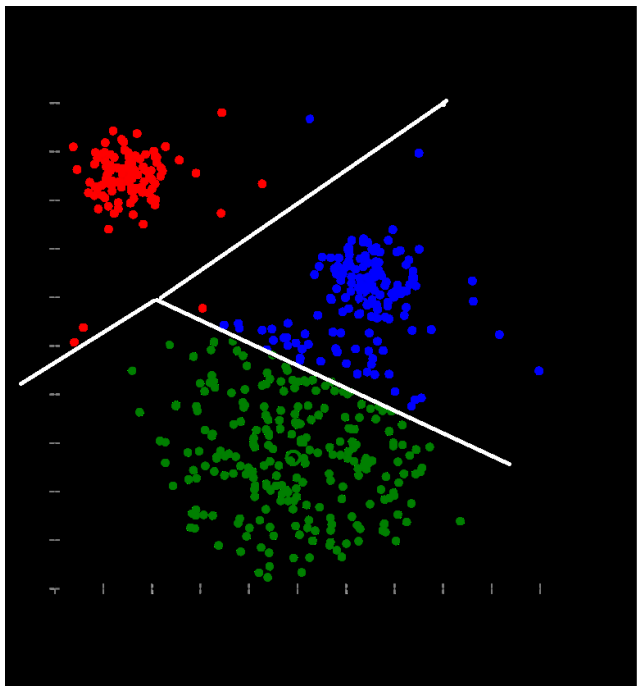
- **k-means**: encontrar pts centrais $c_1, c_2, ..., c_k$ para minimizar $\sum_{i=1}^{n} min_{j \in 1,\ldots,k}.d^2(x^i, c_j)$
- **k-média**: encontrar pts centrais $c_1, c_2, ..., c_k$ para minimizar $\sum_{i=1}^{n} min_{j \in 1,\ldots,k}.d(x^i, c_j)$
- **K-center**: encontrar partição para minimizar o raio máximo.

Euclidean K-means

- **Input**: Seja um conjunto de n pontos: $x_1, x_2, ..., x_n$ em R^d
- **Alvo**: Agrupamentos k
- **Output**: k representantes $c_1, c_2, \ldots, c_k$ em R^d
- **Objetivo**: escolher $c_1, c_2, \ldots, c_k \in \mathrm{R}^d$ para minimizar $\sum_{i=1}^{n} min_{j \in 1,\ldots,k}.d^2(x^i, c_j)$

Euclidean K-means

- Cada ponto é designado ao seu centro mais proximo e isto leva ao Diagrama de Voronoi



Complexidade

- **NP hard**: mesmo para $k = 2$ [Dagupta'08] ou $d = 2$ [Mahajan-Nimbhorkar-Varadarajan09]

- Existem alguns casos fáceis.

Heurística Comum: **Método de Lloyd** [a]

[a] [Least squares quantization in PCM, Lloyd, IEEE Transactions on Information Theory, 1982]

- **Entrada**: A set of n pontos $x_1, x_2, \ldots, x_n \in R^d$.
- Initializar centros $c_1, c_2, \ldots, c_k$ dos clusters $C_1, C_2, \ldots, C_k$ aleatoriamente.
- Repetir até não existir mudança nos centros:
  - Para cada **j**: $C_j \leftarrow \{x \in S$ cujo centro é $c_j\}$
  - Para cada **j**: c_j é a média C_j.

Heurística Comum: **Método de Lloyd**

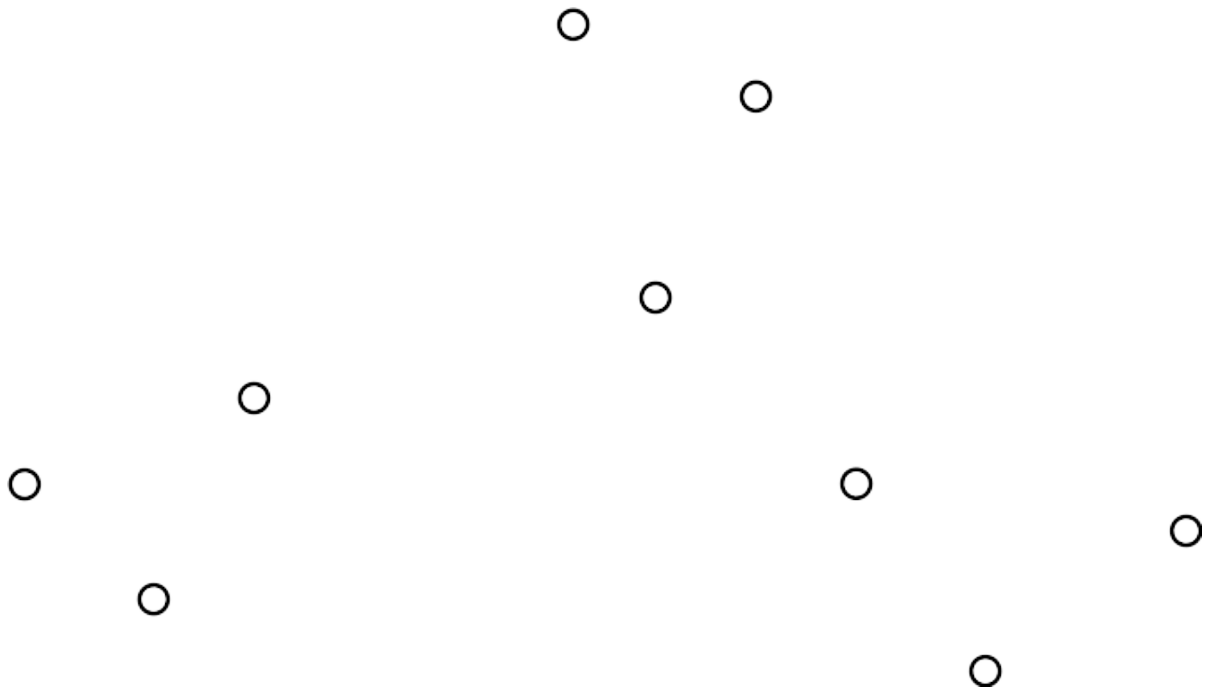
- Sempre converge.
- O custo sempre cai e
- Há apenas um número finito de partições de Voronoi (portanto, um número finito de valores que o custo pode assumir).

Heurística Comum: **Método de Lloyd**

- Initialização é crucial (quão rápido o método converge, qualidade da solução)
- Centros escolhidos a partir dos dados
- K-means ++ (funciona bem e tem prova de garantia)

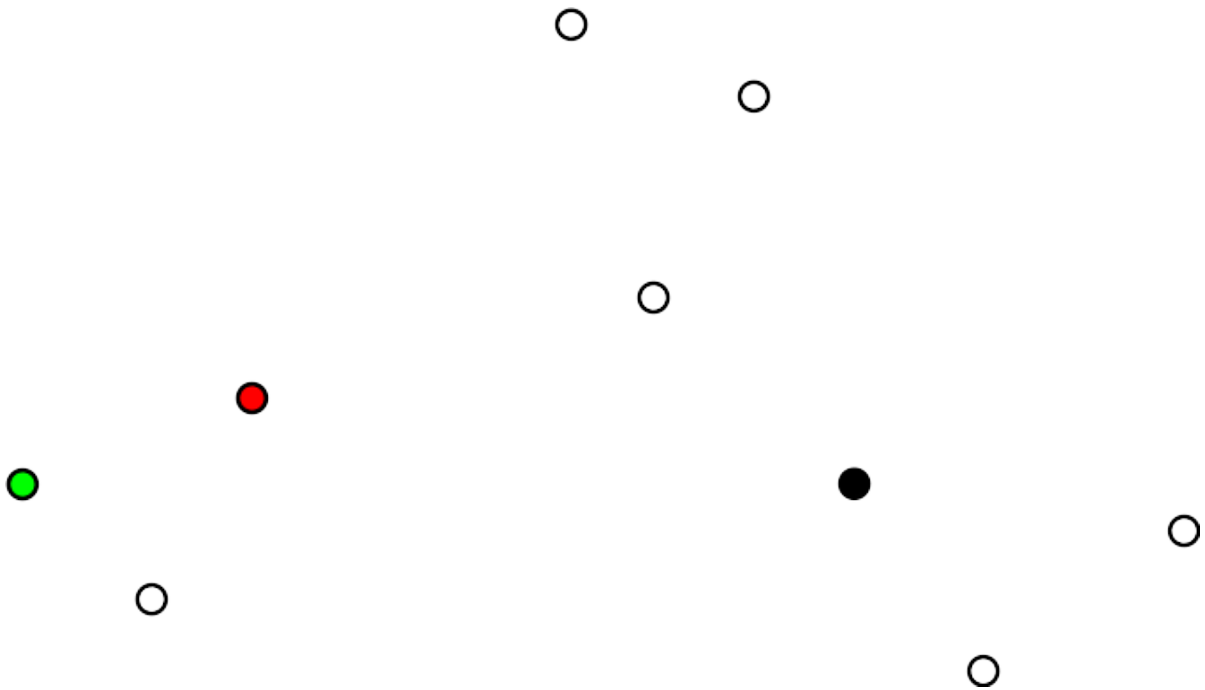
Método de Lloyd: Inicialização Aleatória

- Exemplo: Dado um conjunto de pontos:



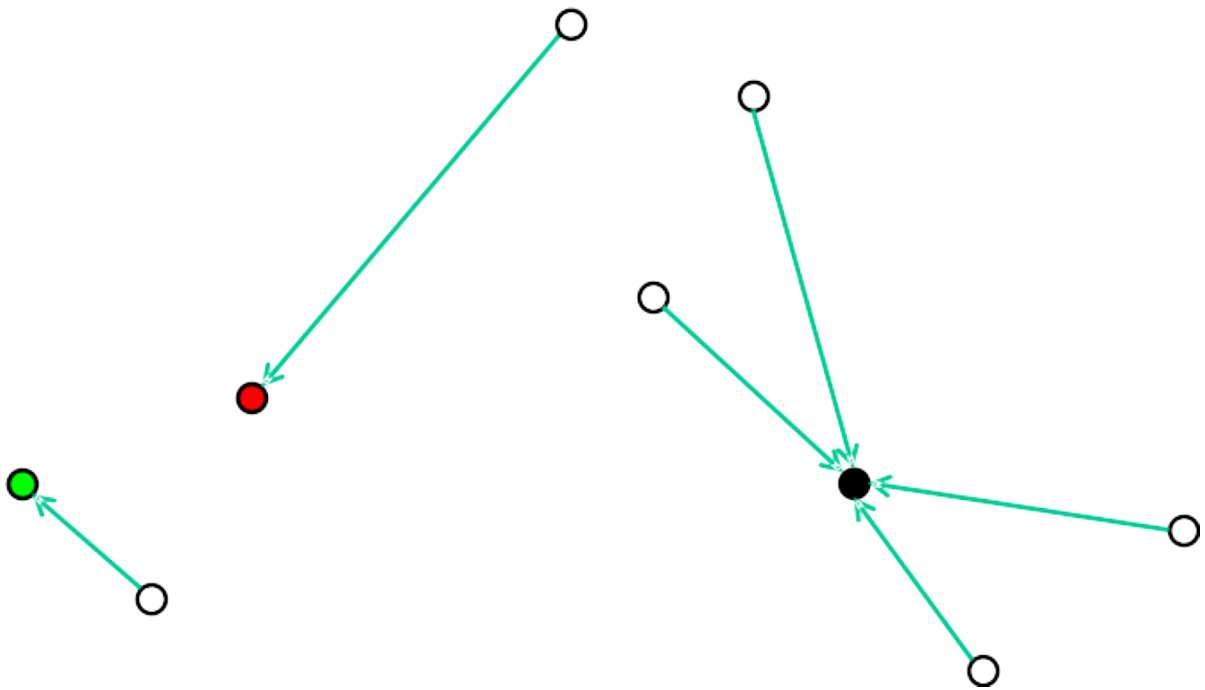
Método de Lloyd: Inicialização Aleatória

- Seleciona centros iniciais aleatoriamente:



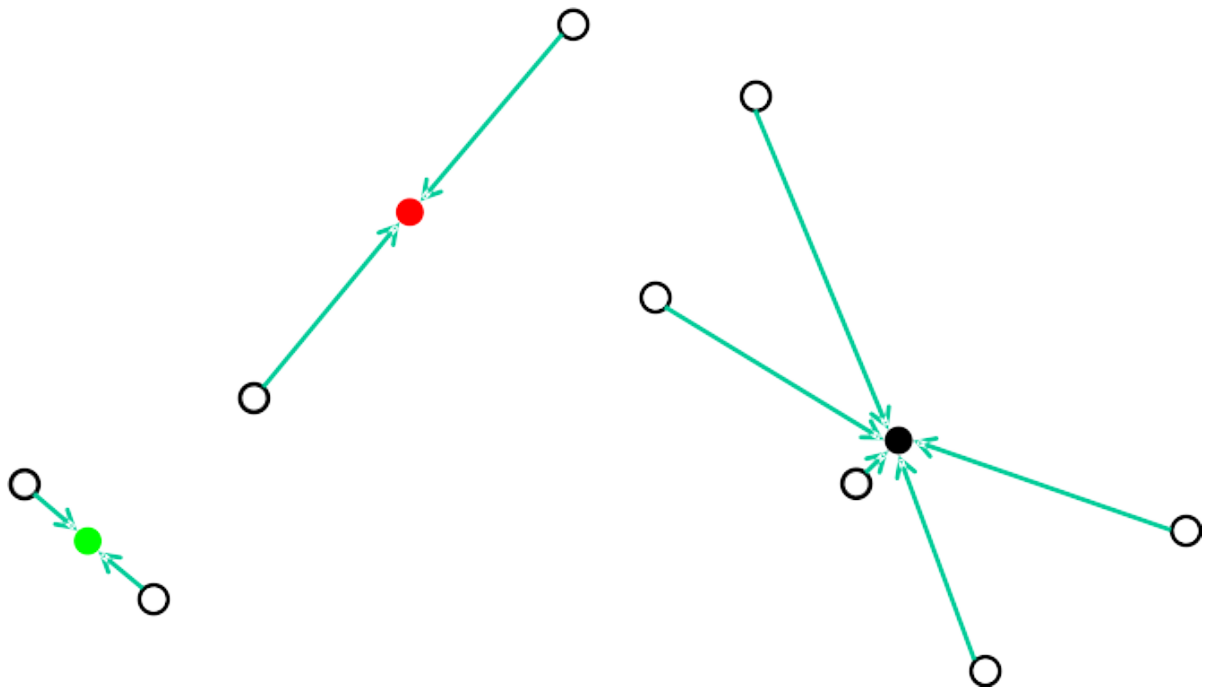
Método de Lloyd: Inicialização Aleatória

- Designa cada ponto ao seu centro mais próximo:



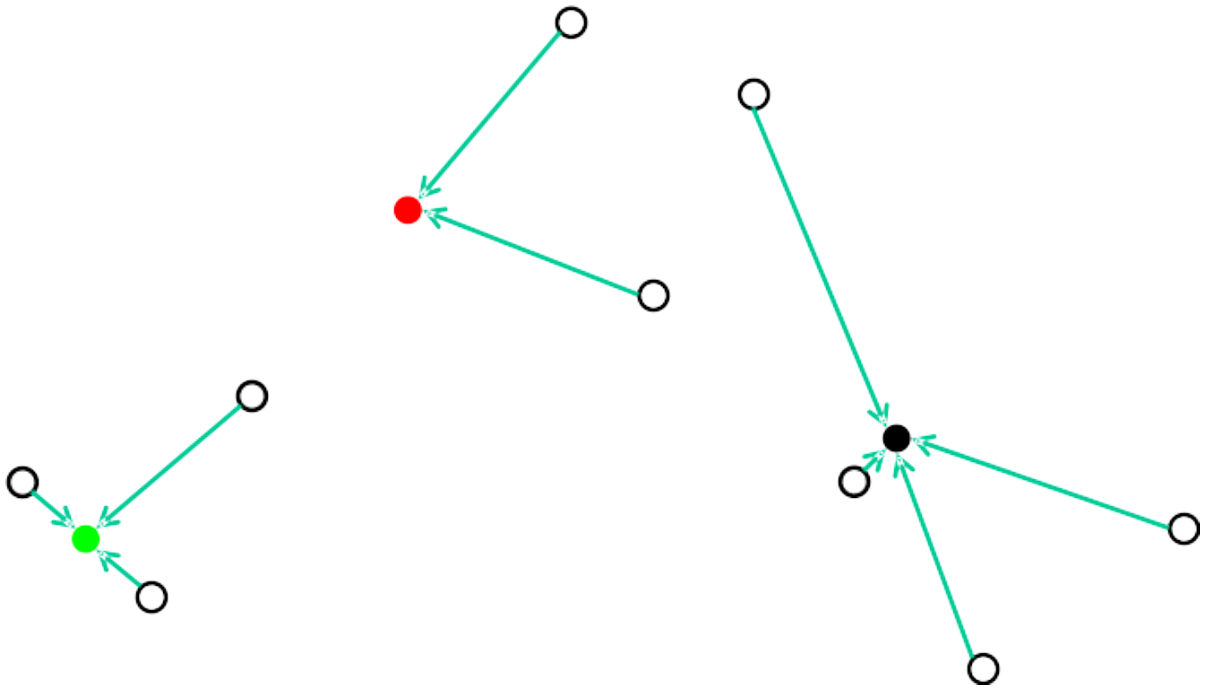
Método de Lloyd: Inicialização Aleatória

- Recomputa centros ótimos dados os agrupamentos fixados:



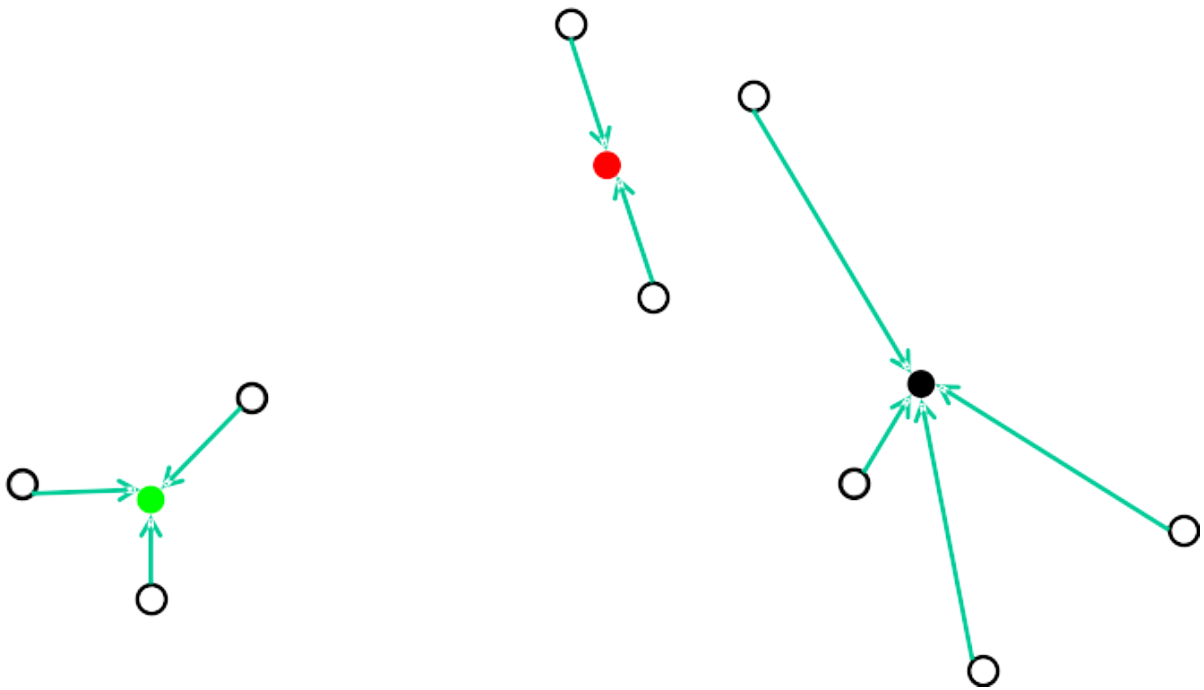
Método de Lloyd: Inicialização Aleatória

- Designa cada ponto ao seu centro mais próximo:



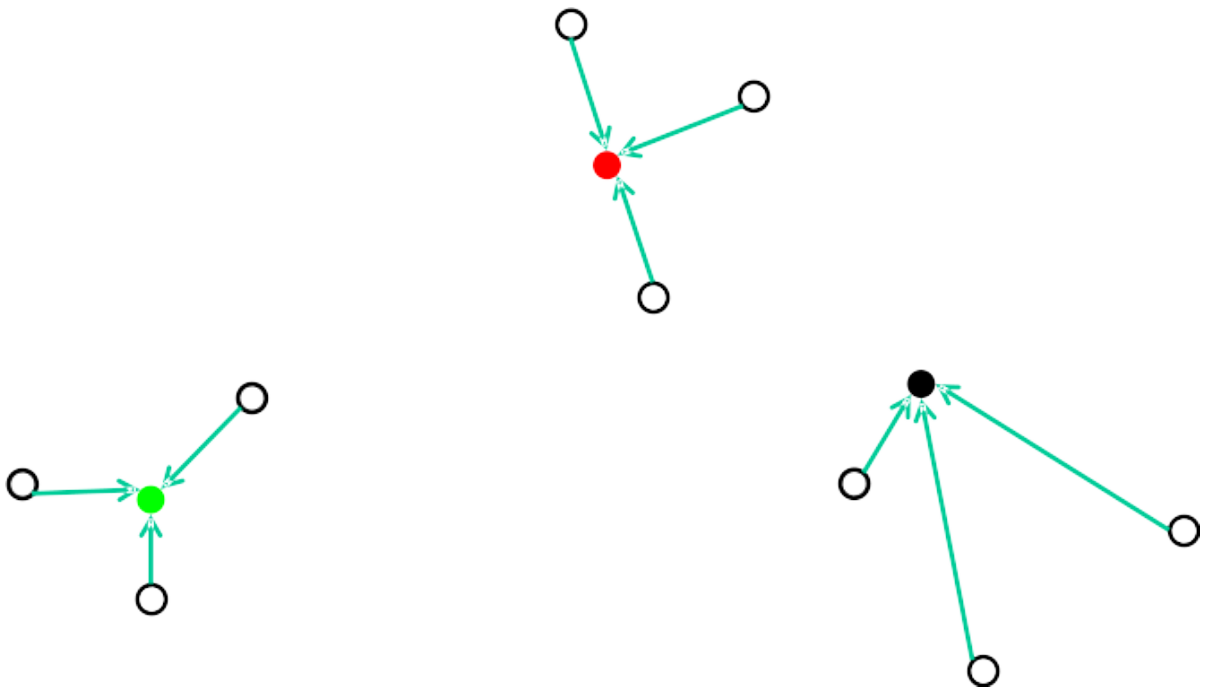
Método de Lloyd: Inicialização Aleatória

- Recomputa centros ótimos dados os agrupamentos fixados:



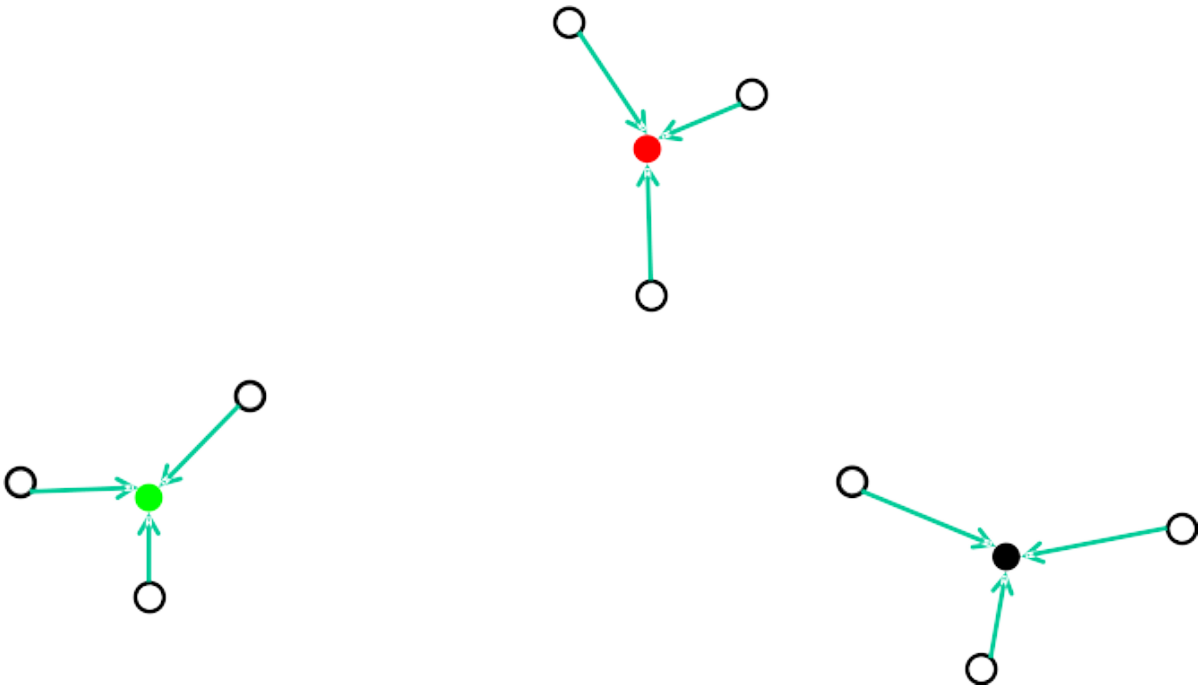
Método de Lloyd: Inicialização Aleatória

- Designa cada ponto ao seu centro mais próximo:



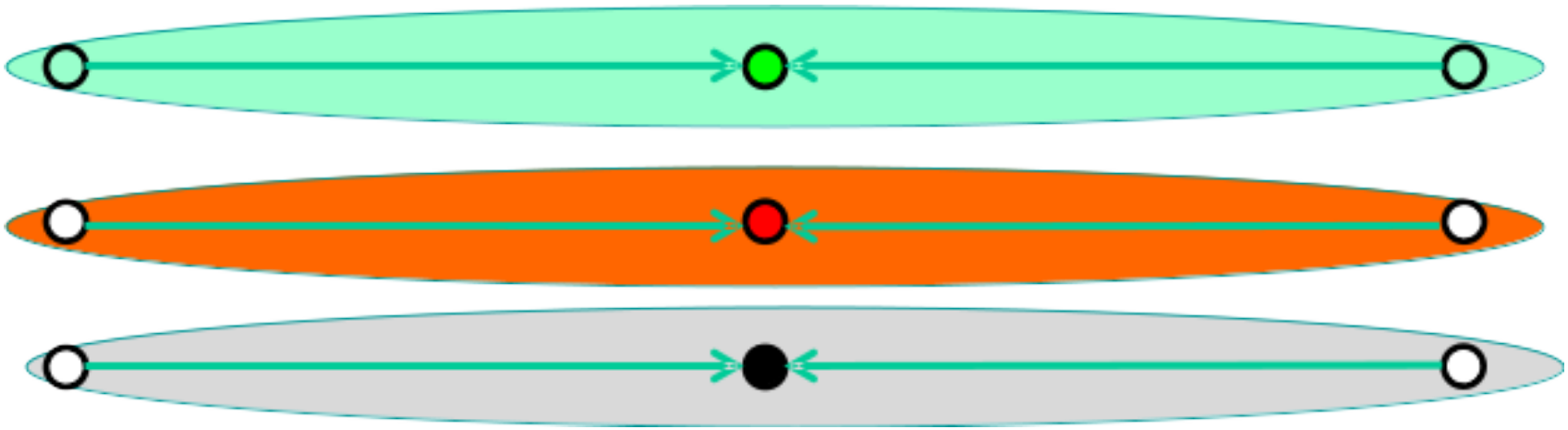
Método de Lloyd: Inicialização Aleatória

- Recomputa centros ótimos dados os agrupamentos fixados:



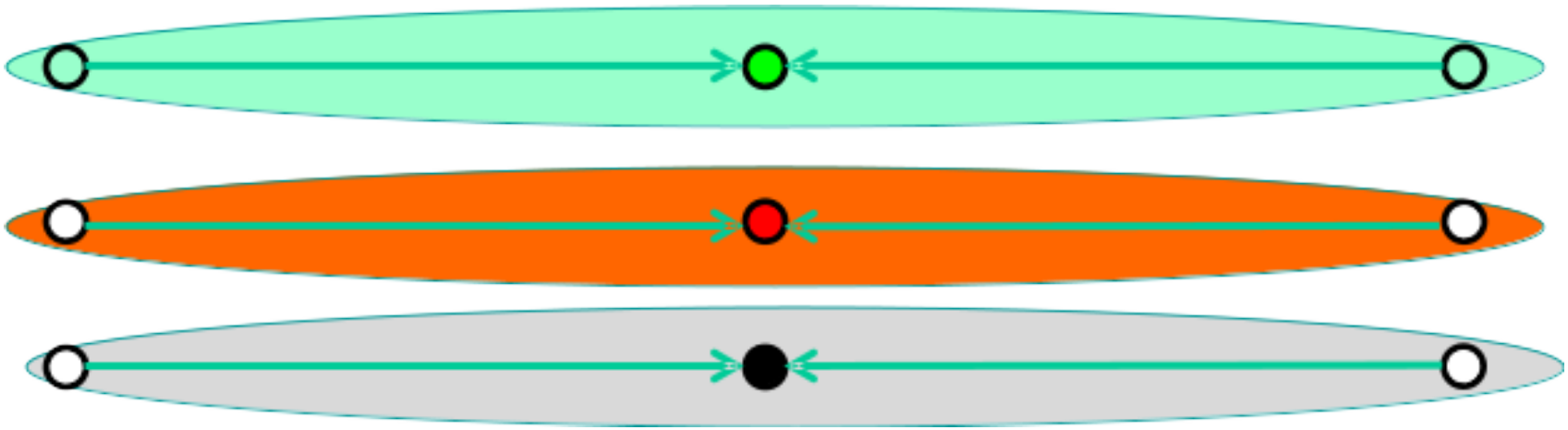
- Obteve uma boa solução neste exemplo.

Desempenho do Método de Lloyd



- O método sempre converge, mas ele pode convergir para um otimo local que é diferente de um mínimo global.

Desempenho do Método de Lloyd

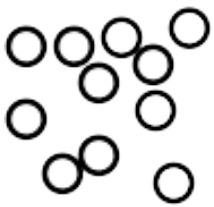


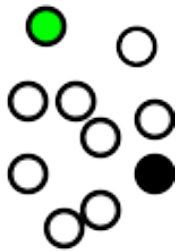
- Ótimo local: cada ponto é atribuído ao seu centro mais próximo e cada centro é o valor médio de seus pontos.

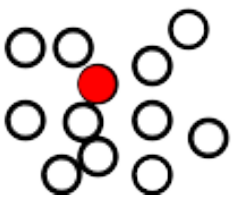
Desempenho do Método de Lloyd

- A má performance pode ocorrer mesmo com agrupamentos bem separados.

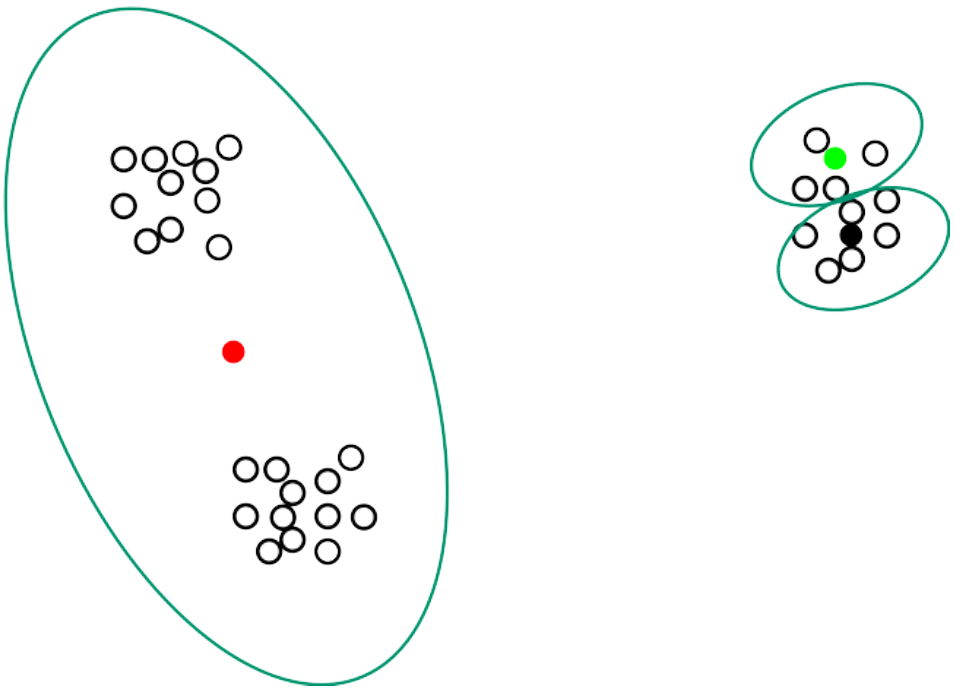
Desempenho do Método de Lloyd







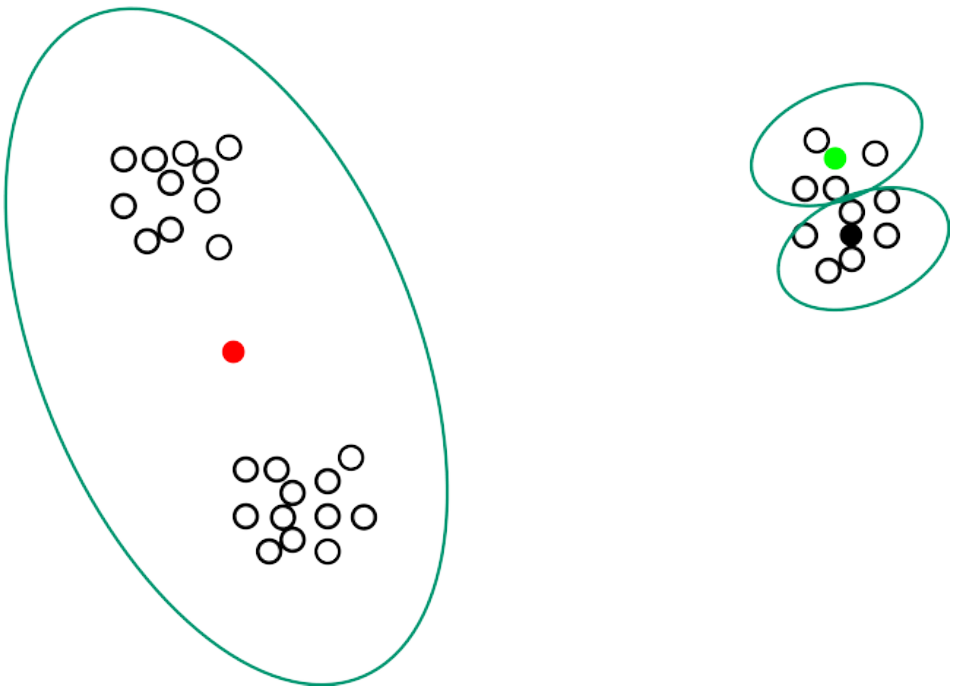
Desempenho do Método de Lloyd



Desempenho do Método de Lloyd

- Se fizermos uma inicialização aleatória, à medida que k aumenta, torna-se mais provável que não se tenha escolhido perfeitamente um centro para a nossa Gaussiana em nossa inicialização (portanto, o método de Lloyd produzirá uma solução ruim).
- Para k Gaussianas de tamanho igual, Pr [cada centro inicial está em um Gaussiano diferente] $\approx \frac{k!}{k^k} \approx \frac{1}{e^k}$
- Torna-se improvável quando k cresce muito.

Desempenho do Método de Lloyd



Outra Idéia de Inicialização - Heurística do Ponto Mais Distante

Escolha c_1 arbitrariamente

- For $j = 2, ..., k$

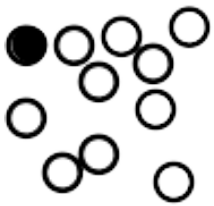
  Tome c_j entre os pontos: $x^1, x^2, ..., x^d$, que está mais longe dos centros $c_2, c_3, ..., c_{j-1}$ previamente escolhidos.

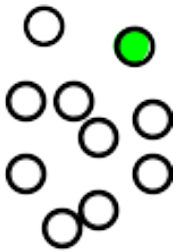
- **OBS:** Isto resolve o problema de encontrar as Gaussianas, mas pode ser descartado devido aos outliers.

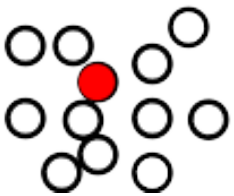
Heurística do Ponto Mais Distante

- Esta heuristica trabalha bem no exemplo anterior

Heurística do Ponto Mais Distante



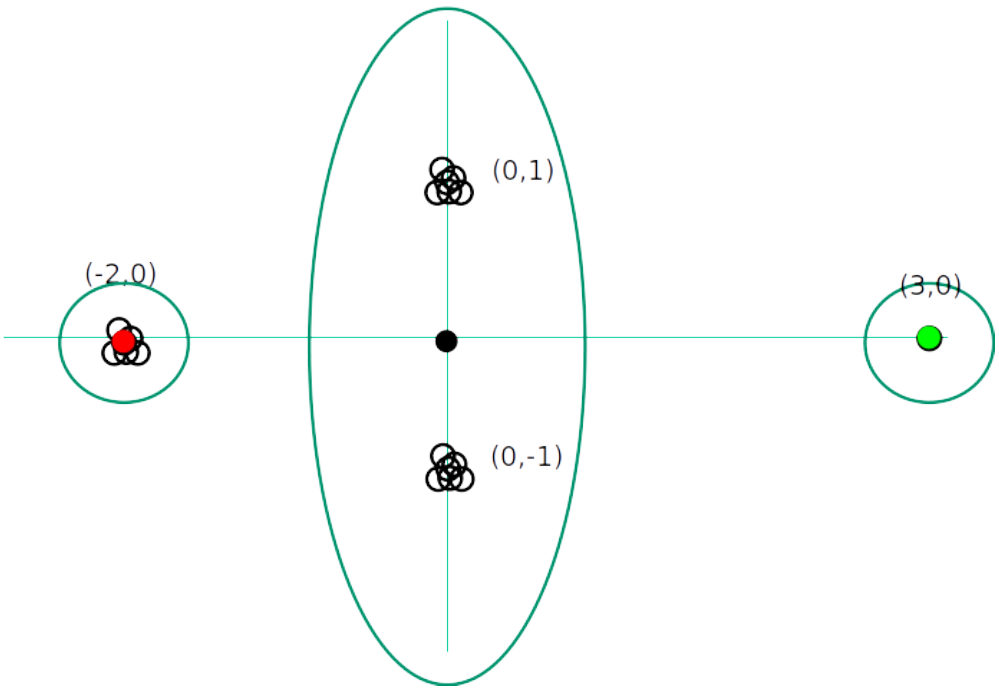




Heurística do Ponto Mais Distante

- Esta heuristica é sensivel a outliers.
- Assuma $k = 3$

Heurística do Ponto Mais Distante



Inicialização do K-MEANS++: amostragem D2

- Interpola entre o ponto aleatório e o mais distante.
- Seja D(x) ser a distancia entre o ponto x e o seu centro mais próximo. Escolher o proximo centro proporcional a D^2.
- Escolha c_1 arbitrariamente
  - For $j = 2, ..., k$

    Tome c_j entre os pontos: $x^1, x^2, ..., x^d$:

    $Pr(c_j = x^i) = min_{j'<j} ||x^i - c_{j'}||^2$

Ideia do K-means++ amostragem D2

Teorema: K-means++ sempre alcança uma aproximação **O(log k)** na obtenção da solução do K-MEANS.

Idéia do K-means++ amostragem D2

- Interpolar entre inicialização de pontos aleatórios e mais distantes
  - Seja D(x) a distância entre um ponto x e seu centro mais próximo. Escolha o próximo centro proporcional a D^{α}.
  - $\alpha = 0$ amostragem aleatória
  - $\alpha = \infty$, heurística do ponto mais distante
  - $\alpha = 2$, k-means++

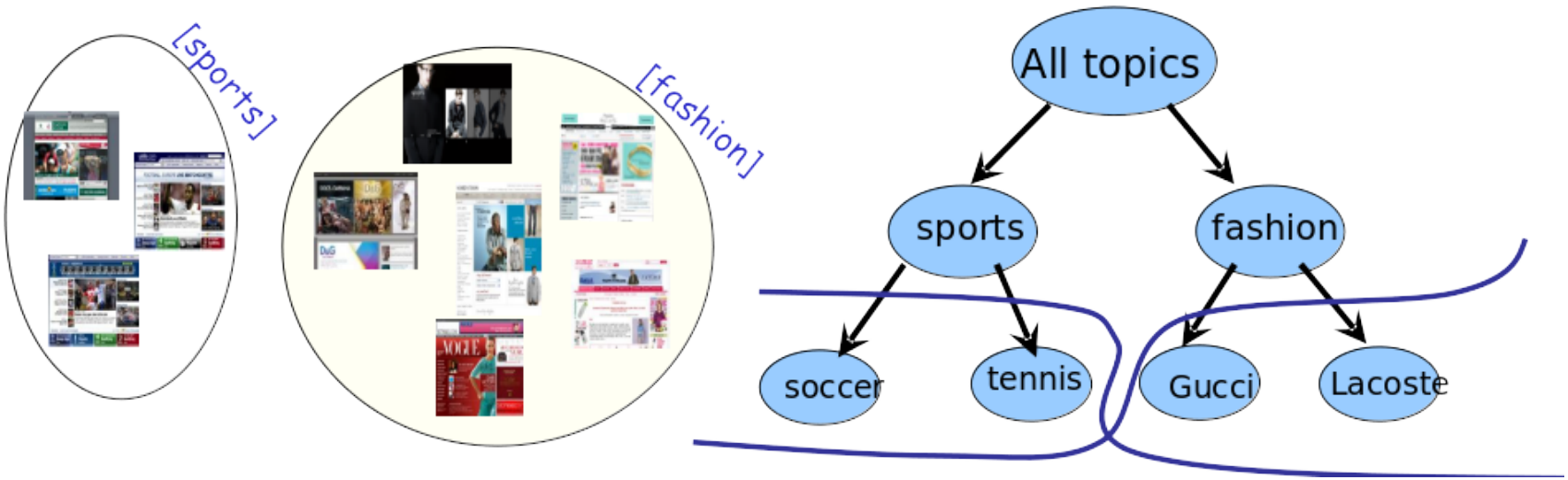
OBS: $\alpha = 1$, funciona bem para k-média

K-means++ Resolve o problema

Análise de Complexidade

- K-means ++ sempre alcança uma aproximação **O(log k)** à solução k-means ótima esperada.
- O método de Lloyd **O(nkd)** em cada iteração. O Lloyd só pode melhorar ainda mais a solução obtida.
- Exponential - no. de iterações no pior caso [AV07].

Agrupamento Hierárquico

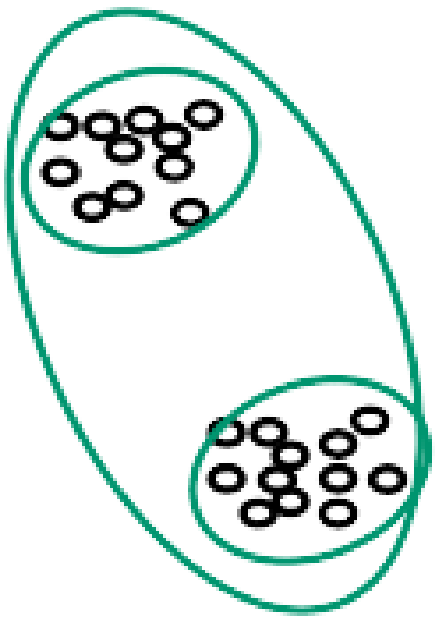


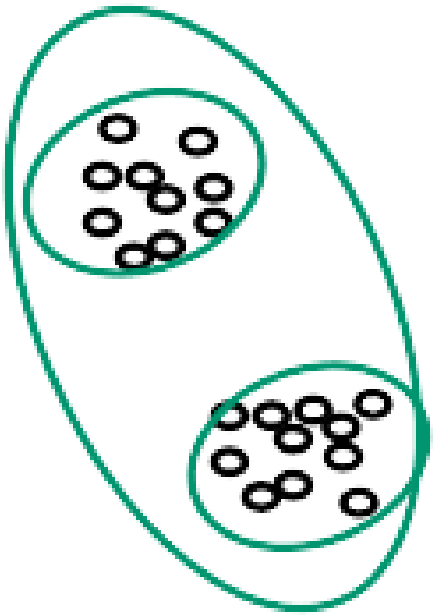
- Uma hierarquia pode ser mais natural.
- Diferentes usuários podem se importar com diferentes níveis de granularidade ou até mesmo podas.

Top-Down

Partição dos dados em 2 grupos (Divisivo)

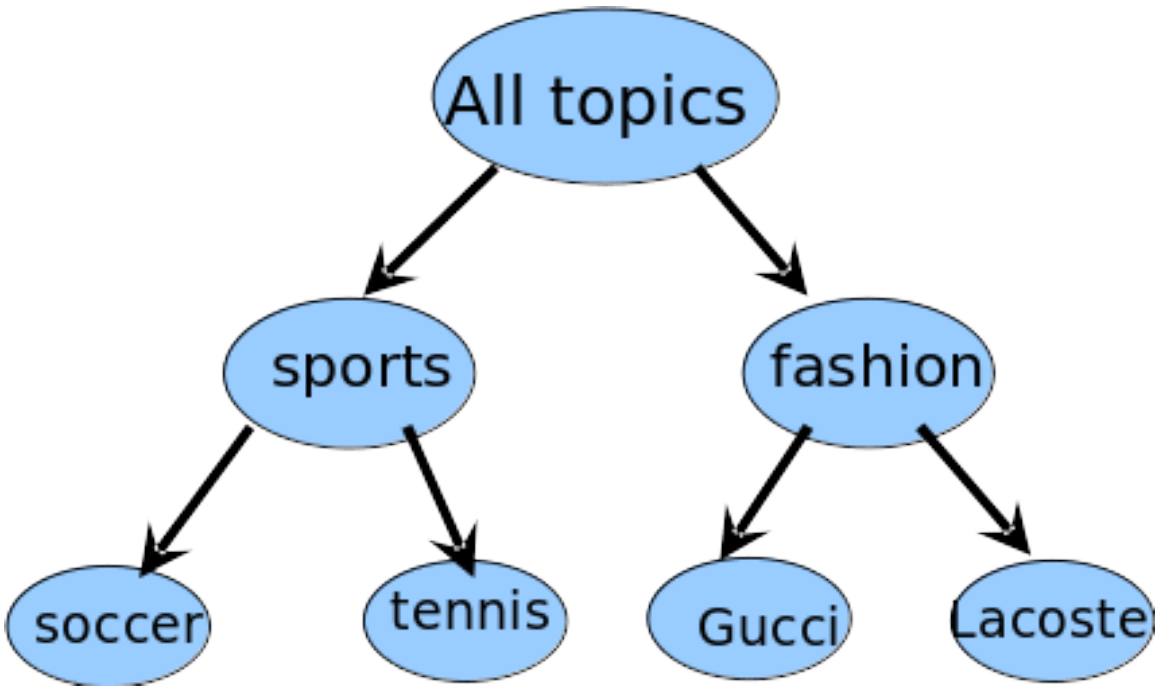
Agrupe recursivamente cada grupo.





Agrupamento Hierárquico

- Bottom-up - Aglomerativo
- Comece com cada ponto em seu próprio cluster.
- Mesclar repetidamente os dois grupos mais próximos.
- Diferentes defs de "mais próximo" fornecem diferentes algoritmos.



Agrupamento Hierárquico

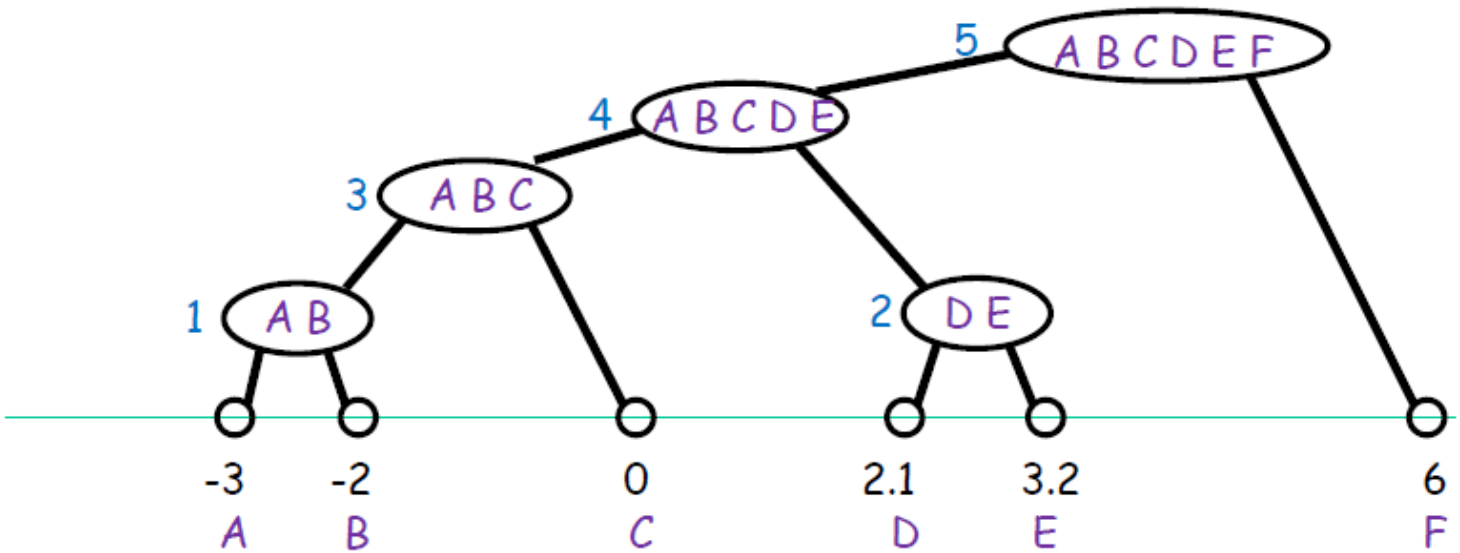
- d(x,y) – distancia entre dois objetos x and y
- **Single linkage**: $dist(A, B) = \min_{x \in A, x' \in B} dist(x, x)$
- **Complete linkage**:

  $dist(A, B) = \max_{x \in A, x' \in B} dist(x, x)$
- **Average linkage**:

  $dist(A, B) = avg_{x \in A, x' \in B} dist(x, x)$
- **Wards' method**

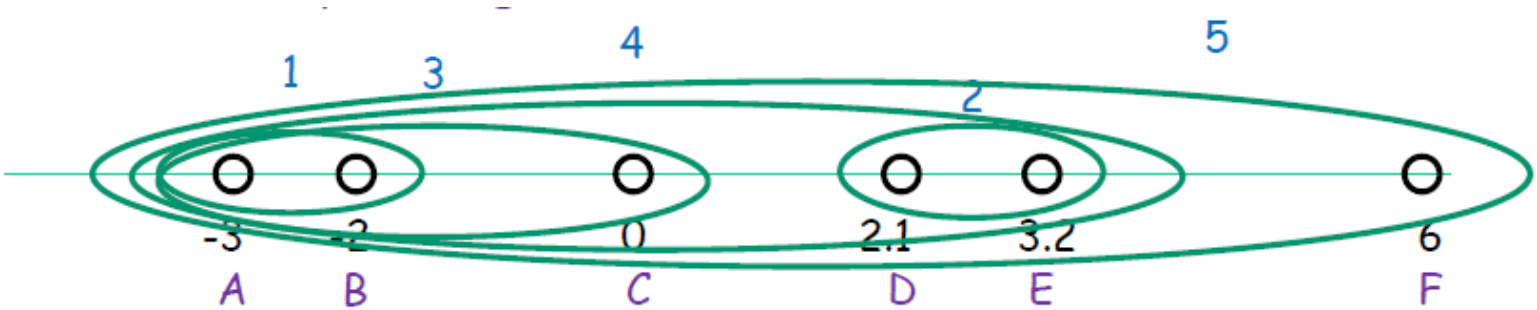
Método Single Linkage

- Dendogramas
- Comece com cada ponto em seu próprio cluster.
- Mesclar repetidamente os dois grupos mais próximos.



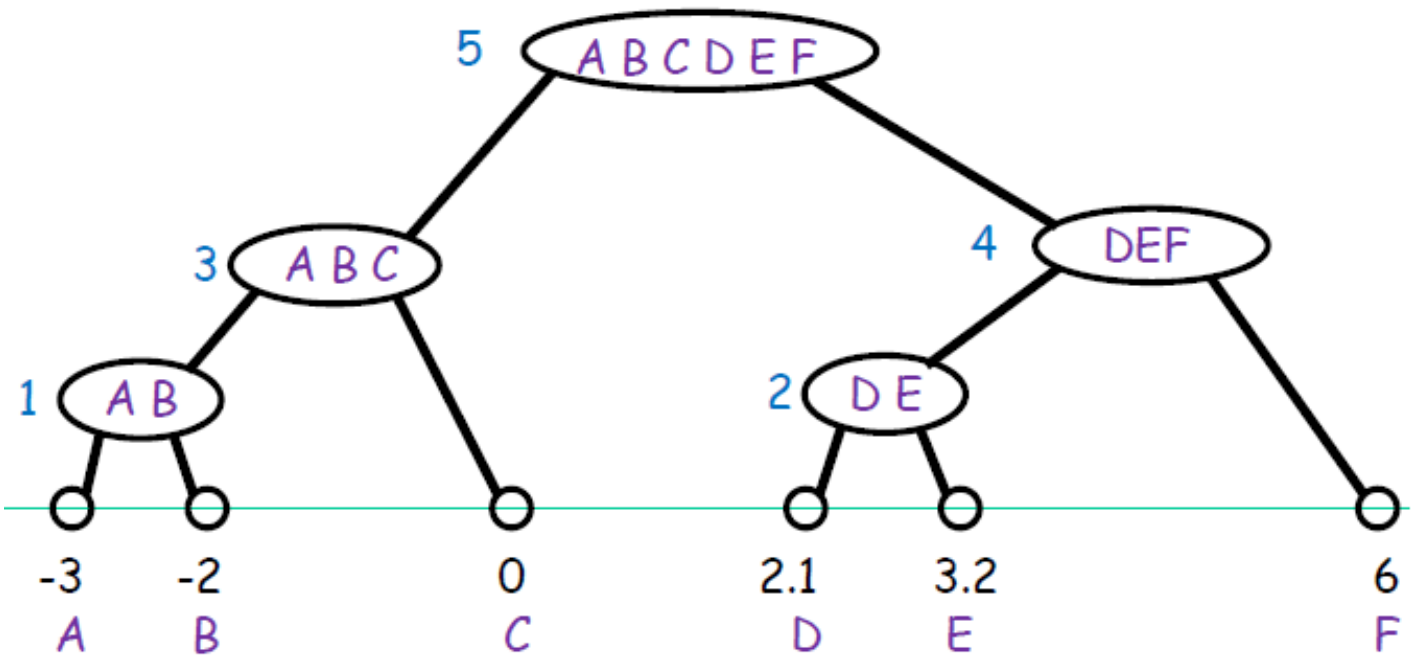
Método Single Linkage

- Grafos
- Comece com cada ponto em seu próprio cluster.
- Mesclar repetidamente os dois grupos mais próximos.



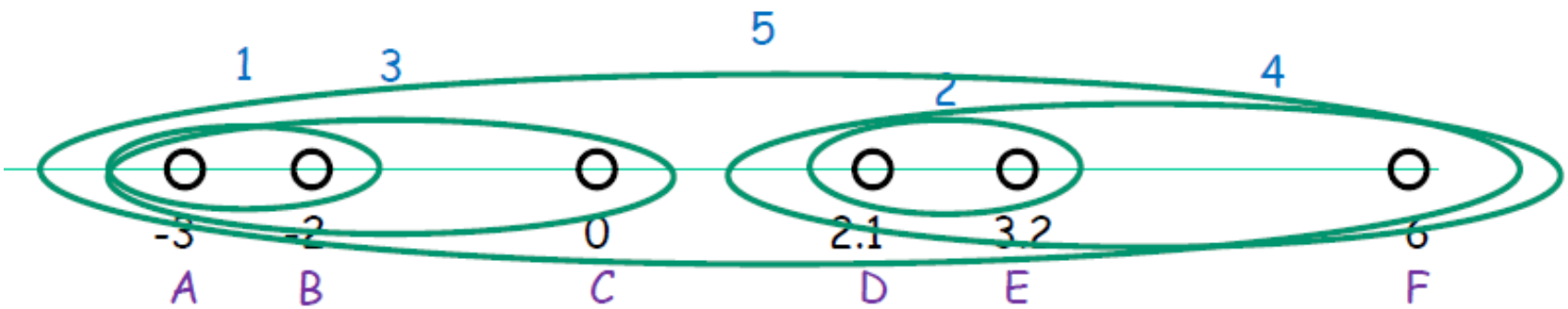
Método Completo Linkage

- Uma maneira de pensar nisso: mantenha o diâmetro máximo o menor possível



Método Completo Linkage

- Uma maneira de pensar nisso: mantenha o diâmetro máximo o menor possível



Método Ward

- Mesclar os dois clusters de modo que o aumento no custo de k-means seja o menor possível.
- Funciona bem na prática.

Ward's method: $\text{dist}(C, C') = \frac{|C| \cdot |C'|}{|C| + |C'|} \|\text{mean}(C) - \text{mean}(C')\|^2$