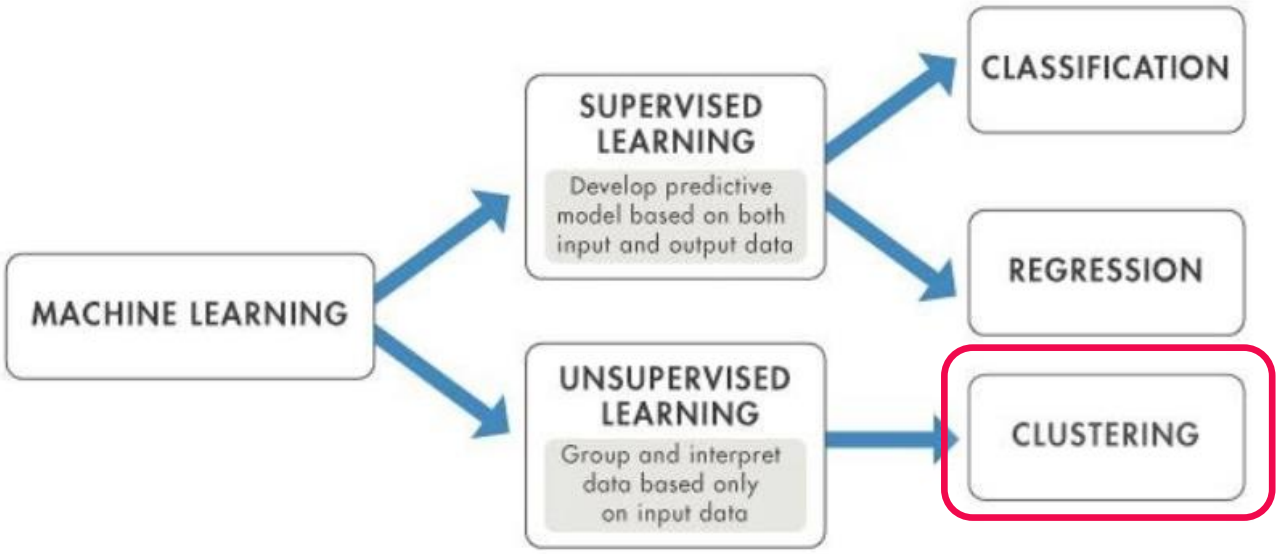
AI & CHATBOT

Aula 17 – Aprendizado de Máquina
Não Supervisionado:
Algoritmos de Argrupamento

Prof. Érick T. Yamamoto

Aprendizado não supervisionado

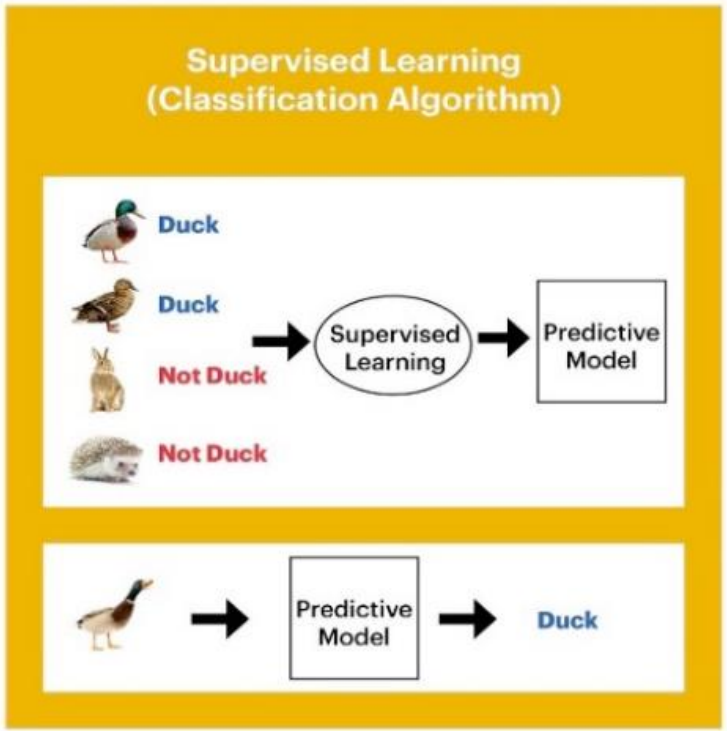
- No aprendizado não supervisionado **não sabemos *a priori* os rótulos das classes**.
- Com essas técnicas gostaríamos de resolver problemas de:
  - **Agrupamento** (ou clusterização)
  - **Regras de Associação**
  - **Redução de Dimensionalidade**
  - **Detecção de Outlier**

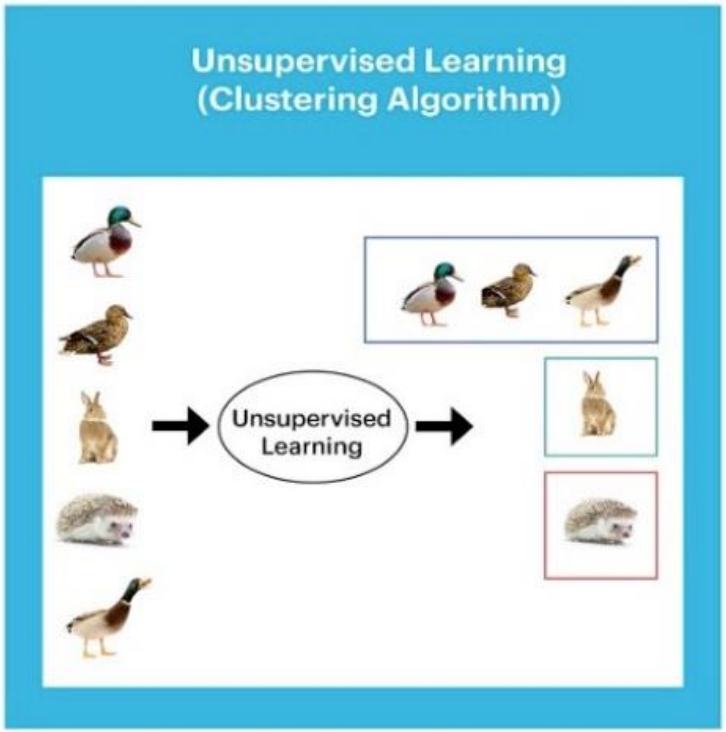


https://www.researchgate.net/profile/Rory-Bunker/publication/319937079/figure/fig1/AS:542578279366656@1506372150206/Supervised-Learning-versus-Unsupervised-Learning-Mathworks-nd.png

Aprendizado não supervisionado

- Na **classificação** nosso atributo alvo é um objeto (string) previamente conhecido (dado rotulado);
- Na **clusterização** nosso atributo alvo é um número (int) previamente desconhecido (dado não rotulado);

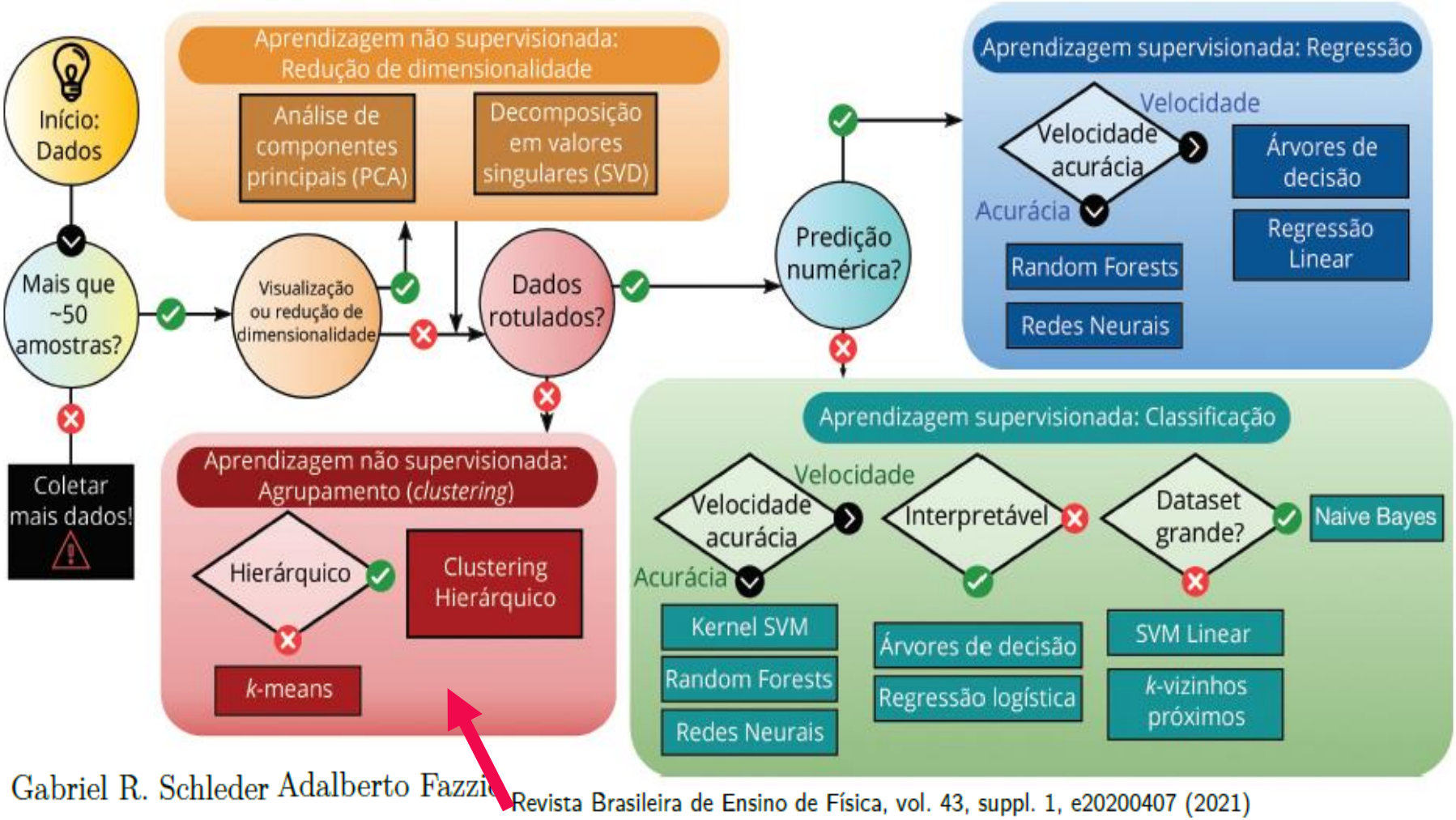




https://miro.medium.com/max/1000/0*4q_X_xQxevOQY_0u

Western Digital.

Aprendizado não supervisionado



Gabriel R. Schleder Adalberto Fazzio Revista Brasileira de Ensino de Física, vol. 43, suppl. 1, e20200407 (2021)

Agrupamento

Agrupamento é uma técnica não supervisionada!

$$f(x_1, x_2, \ldots, x_N)$$

Atributos descritivos

Grupo alvo desconhecido

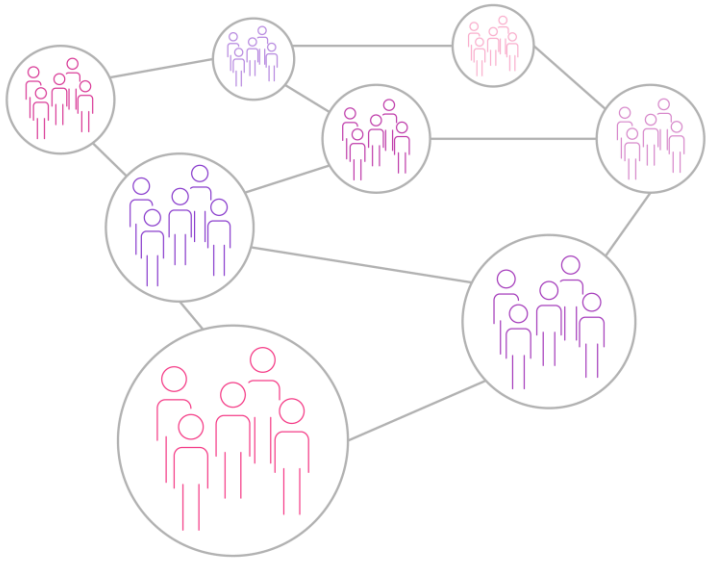
Índice da linha	x1	x2	...	xn	$\hat{y}$
1	548.4	-9789		0.4875	?
2	689.4	-10235		-0.358	?
3	3154.8	-1031858	...	-0.1458	?
...	...			...	
k	803.54	-20000		1.054	?

Queremos criar essa coluna!

Agrupamento

GERAÇÃO DE GRUPOS OU CLUSTERS

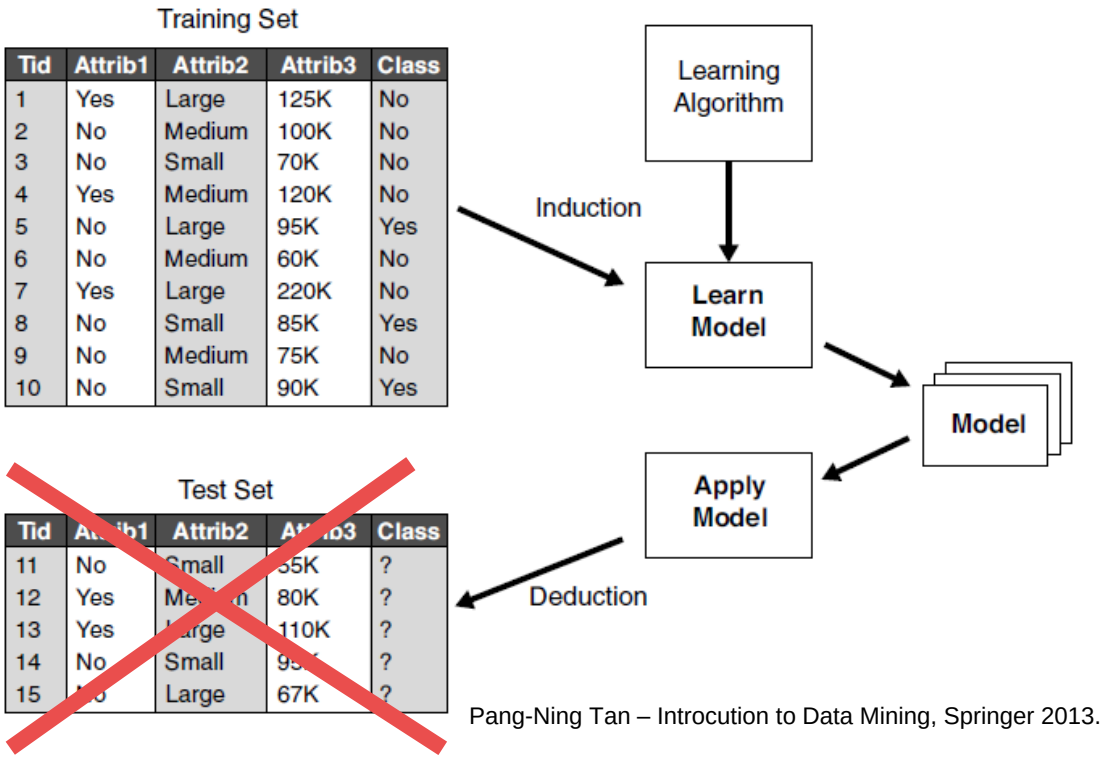
- Os grupos são formados de maneira a maximizar a similaridade entre os elementos de um grupo (similaridade intragrupo) e minimizar a similaridade entre elementos de grupos diferentes (similaridade intergrupos).
- Aprendizado não supervisionado.



Agrupamento

Em algoritmos de agrupamento não temos como testar com as respostas esperadas, já que não sabemos qual é exatamente essa resposta.

A etapa de **Avaliação de Desempenho** para algoritmos de Agrupamento é diferente do que a usada nos algoritmos de Aprendizado Supervisionado.



Pang-Ning Tan – Introcution to Data Mining, Springer 2013.

ALGORITMOS DE CLUSTERIZAÇÃO

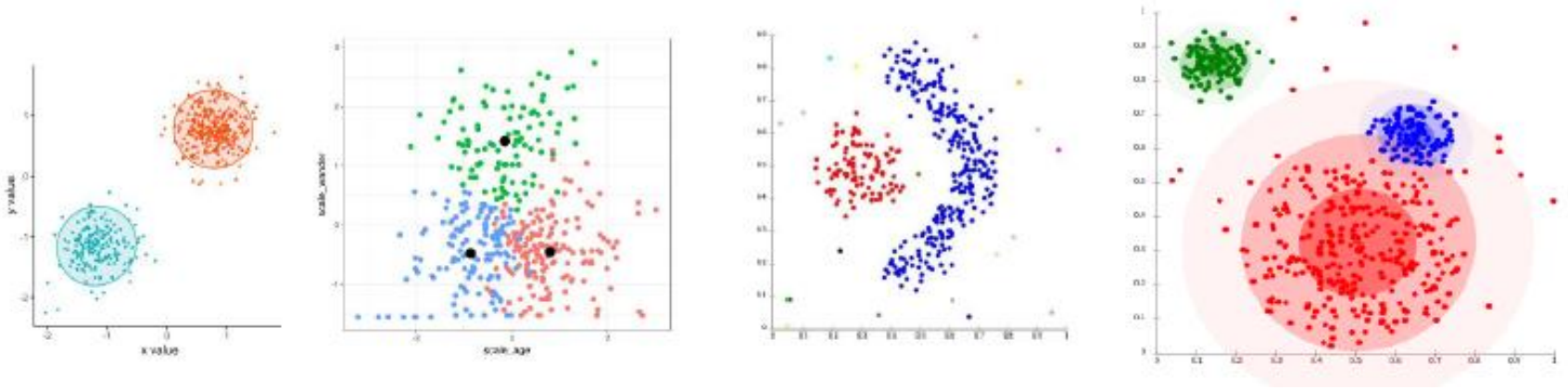
Agrupamento

- Matematicamente:
  - Homogeneidade (coesão interna – similaridade)
  - Heterogeneidade (separação entre grupos – dissimilaridade);
  - Queremos **maximizar a similaridade intra-grupo** e **minimizar a similaridade inter-grupos**;
- Um grupo é um conjunto de entidades semelhantes, que pode ser definido com uma aglomeração de pontos no espaço.

I. Zhang, Density-Based Clustering Algorithms, *cs.hku.hk/~dbgroup/seminar/*

Agrupamento

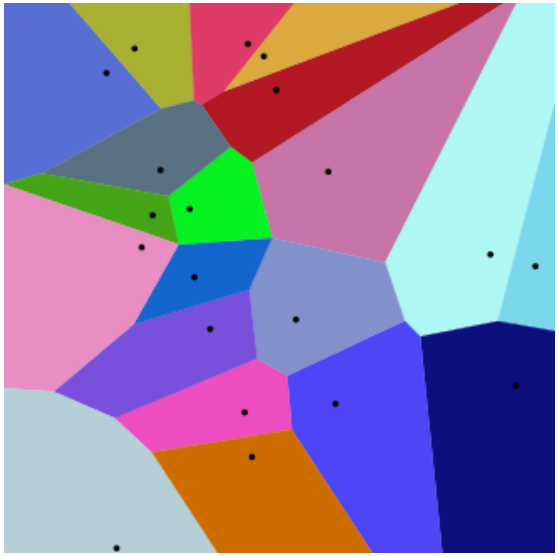
- Grupos podem:
  - Ter diferentes tamanhos, formas e densidades;
  - Formar uma hierarquia;
  - Ter sobreposição ou serem disjuntos



- Existem muitos algoritmos diferentes para fazer agrupamento. Alguns deles são baseados em:
  - **Ligações** como **Agrupamento Hierárquico**;
  - **Densidade** como o **DBSCAN**;
  - **Partições** como o **K-Means**;
  - **Grid** como o STING e o WaveCluster
  - **Modelos** como o SOM, redes neurais e **Mistura Gaussiana**;

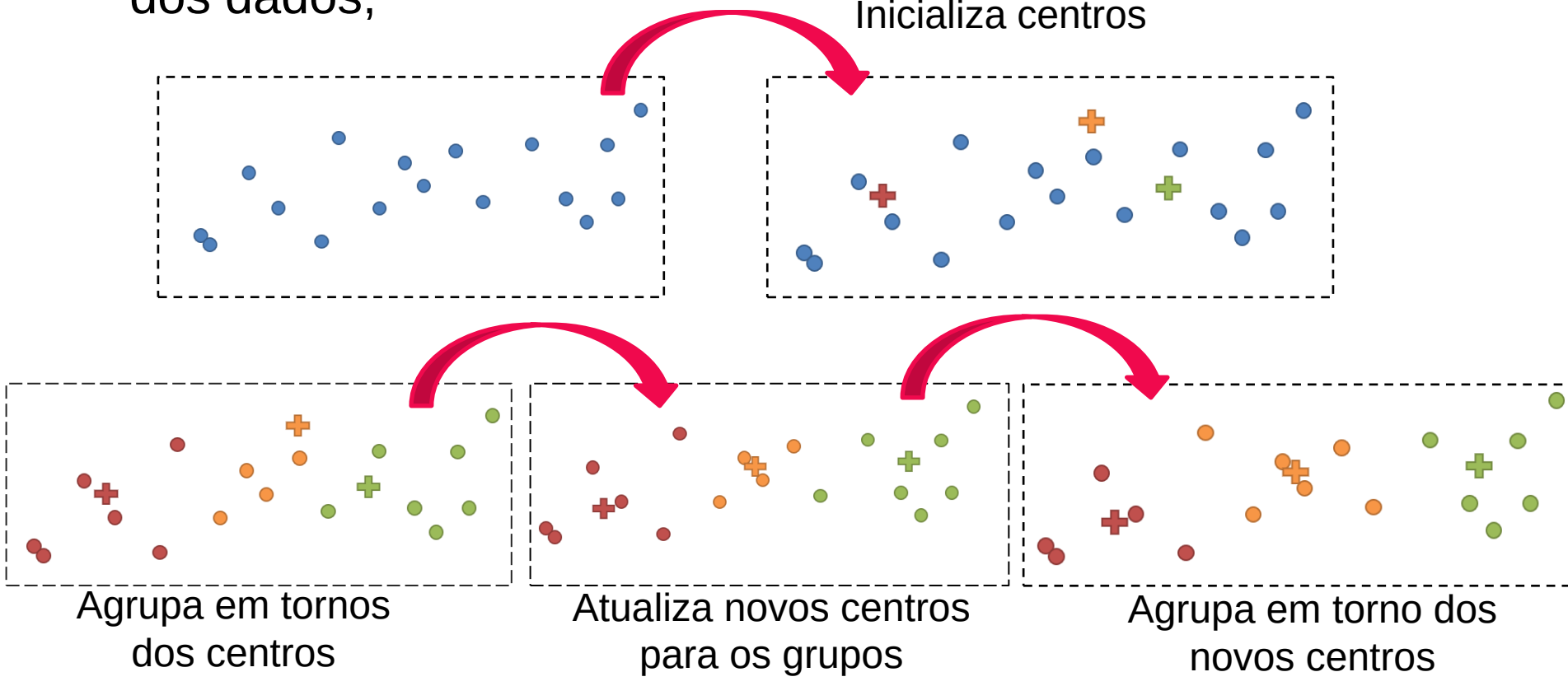
Baseado em Partição – k-Means

- O k-Means é um dos métodos mais antigos (referências originais datam de 1956, 1965 e 1967) e mais utilizados;
- Stuart P. Lloyd propôs em 1957, enquanto trabalhava nos Laboratórios Bell, um algoritmo bastante semelhante ao moderno k-means para fazer modulação PCM. O algoritmo é conhecido como Algoritmo de Lloyd ou Relaxação de Voronoi (devido ao diagrama de Voronoi ou Mosaico de Dirichelet);
- O k-means é simples e intuitivo, baseado na ideia de se quebrar o espaço multidimensional em partições a partir do centróide dos dados;



Baseado em Partição – k-Means

- O k-Means é um dos métodos mais antigos (referências originais datam de 1956, 1965 e 1967) e mais utilizados;
- Ele é simples e intuitivo, baseado na ideia de se quebrar o espaço multidimensional em partições a partir do centróide dos dados;



k-Means – Algoritmo

O pseudocódigo do k-Means pode ser sumarizado como:

1. Escolher aleatoriamente k centros para os clusters;
2. Atribuir cada objeto para o cluster de centro mais próximo segundo alguma métrica de distância (ex: euclidiana);
3. Mover cada centro para a média (centróide) dos objetos do cluster correspondente;
4. Repetir os passos 2 e 3 até que algum critério de convergência seja atendido (ex: número máximo de interação, limiar mínimo de mudança nos centróides).

k-Means – Algoritmo / Métrica

- Precisamos usar uma métrica de distância entre os centróides e os pontos de dados;
- Podemos usar diferentes métricas. A mais comum é a distância Euclidiana:

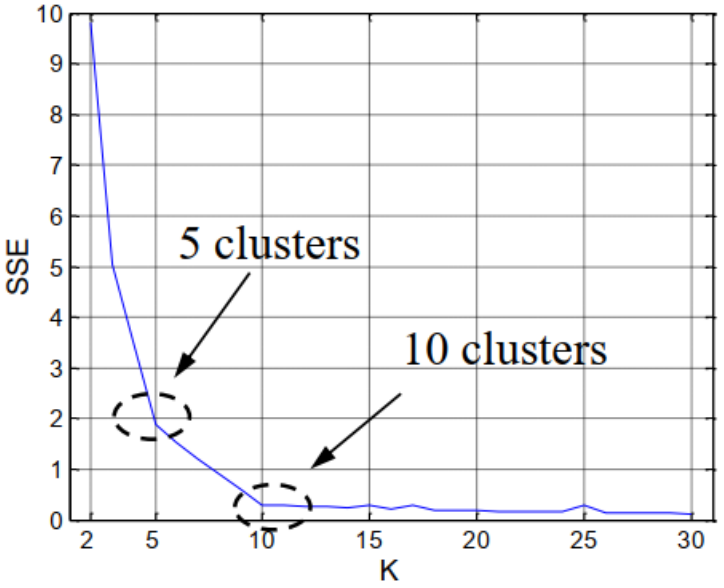
$$d(A,B) = \sqrt{\sum_{i=1}^{n} \left(A_{x_i} - B_{x_i}\right)^2}$$

identifier	class name	args	distance function
"euclidean"	EuclideanDistance	•	`sqrt(sum((x - y)^2))`
"manhattan"	ManhattanDistance	•	`sum(|x - y|)`

Outras métricas:
https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.DistanceMetric.html

k-Means – Algoritmo / Hiperparâmetro

- O k-Means tem o hiperparâmetro k que é o número de grupos;
- Como saber qual é o melhor número de k?
- Podemos usar a Soma dos Erros Quadráticos (SSE) em relação ao centróide para encontrar o “joelho” da curva de otimização:



$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} dist(c_i, x)^2$$

- $dist$ é a distância euclidiana;
- c_i é o centro do i-ésimo agrupamento;
- x são os dados pertencentes ao i-ésimo agrupamento.

k-Means – Vantagens e Desvantagens

FIAP

Vantagens:

- Implementação simplificada.
- Facilidade em lidar com qualquer medida de similaridade e por consequência, qualquer tipo de atributo.

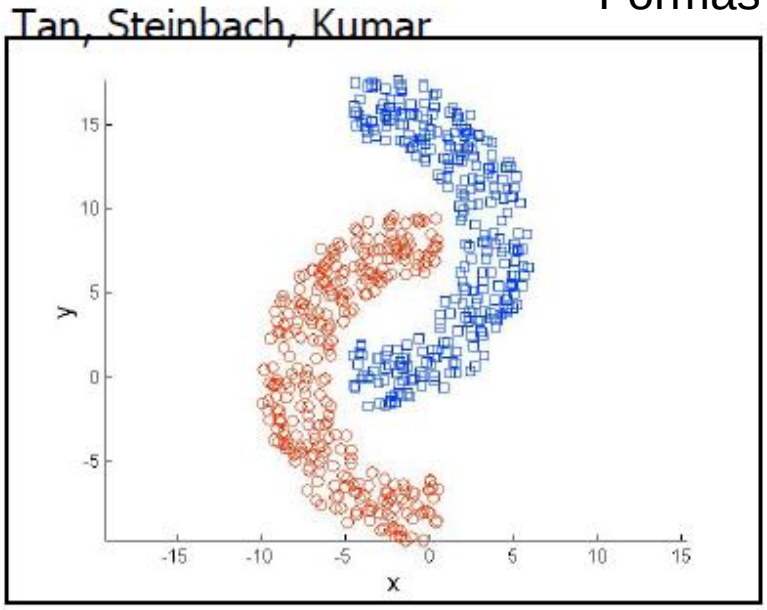
Desvantagens:

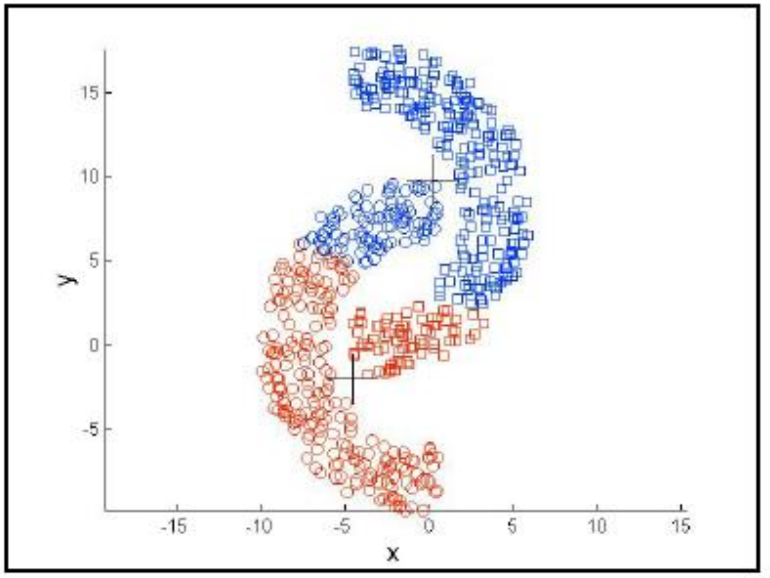
- Dificuldade na definição do valor de “k”.
- Suscetível a outliers e a ausência de normalização.

k-Means - Problemas

K-means é mais suscetível a **problemas** quando os clusters são de **diferentes tamanhos, densidades ou de formas não-globulares** (ex: dados do grupo distribuídos ao longo um retângulo muito comprido e muito fino)

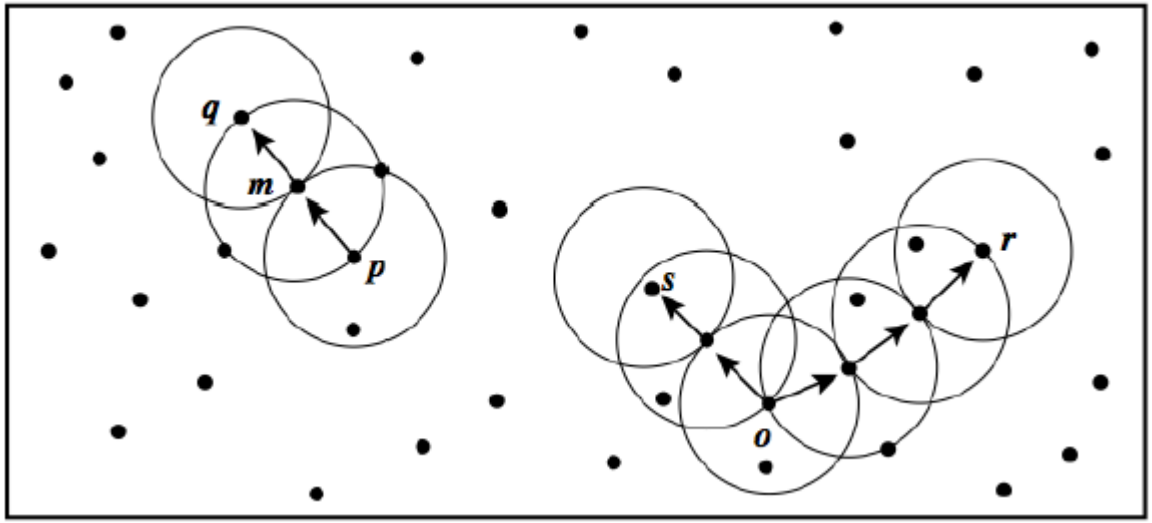
Formas não-globulares





Baseado em Densidade - DBSCAN

- Em um algoritmo **baseado em densidade** os clusters são determinados por **regiões com alta concentração** de objetos (pontos) e a separação se dá por regiões com baixa concentração de pontos;
- **DBSCAN** (Density-Based Spatial Clustering and Application with Noise) é algoritmo mais conhecido de agrupamento baseado na densidade dos pontos;



DBSCAN - Hiperparâmetros

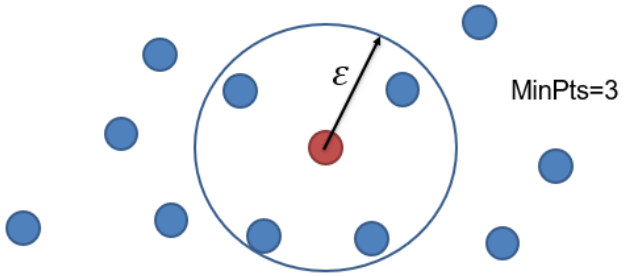
FIAP

- O DBSCAN tem dos hiperparâmetros principais
  - eps ou ε: raio ao redor de um dado ponto;
  - MinPts: mínimo de pontos dentro do raio para que seja definido um agrupamento;
- Com esses parâmetros o algoritmo irá classificar cada ponto de dado em **core**, **border** e **noise**;

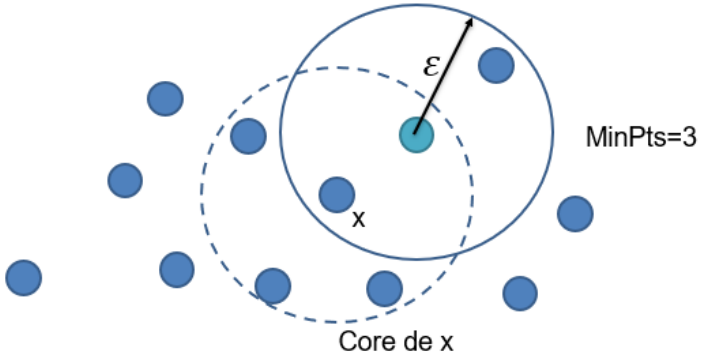
epsilon	*MinPnt*	Resultado
Alto	Alto	Poucos clusters, grandes e densos.
Baixo	Alto	Mais clusters, pequenos e densos
Alto	Baixo	Menos clusters, grandes e pouco densos
Baixo	Baixo	Muitos clusters, pequenos e pouco densos

DBSCAN - Algoritmo

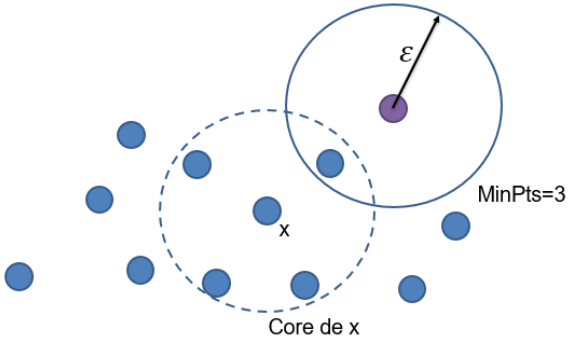
Core: qualquer ponto com uma quantidade de vizinhos maior ou igual a MinPts;



Border (ponto de fronteira): se o número de vizinhos é menor que MinPts, mas está contido em um core;



Noise (ruído): se não é nem um **core** nem um **border**;



DBSCAN - Algoritmo

O pseudocódigo do DBSCAN pode ser sumarizado como:

1. Percorra os dados e rotule os objetos como **core, border ou noise**;
2. Elimine os objetos rotulados como **noise**;
3. Insira uma aresta entre cada par de objetos core vizinhos (2 objetos são vizinhos se um estiver dentro do raio eps do outro);
4. Faça cada componente conexo resultante ser um **cluster**;
5. Atribua cada **border** ao **cluster** de um de seus **cores** associados (resolva empates se houver objetos core associados de diferentes clusters).

DBSCAN - Algoritmo

FIAP

Pontos Originais

Clusters

DBSCAN – Vantagens e Desvantagens

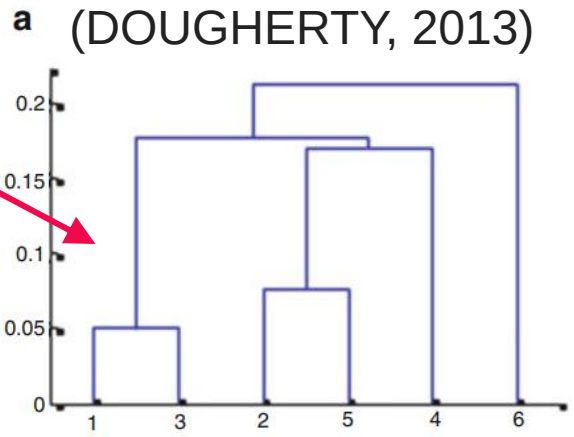
- Vantagens:
  - Resistência ao ruído e capacidade de trabalhar com outliers;
  - Trabalha com grandes volumes de dados (linhas).

- Desvantagens:
  - Grupos com diferentes densidades;
  - Dados em alta-dimensão;
  - Dificuldade na clusterização de clusters concêntricos (um cluster dentro do outro);
  - Conjunto de dados cuja distância média entre as amostras seja muito distinta entre clusters (clusters mais densos que outros); o DBSCAN pode encontrar dificuldades em função do raio de vizinhança (eps);
  - Dados com alta dimensionalidade (colunas).

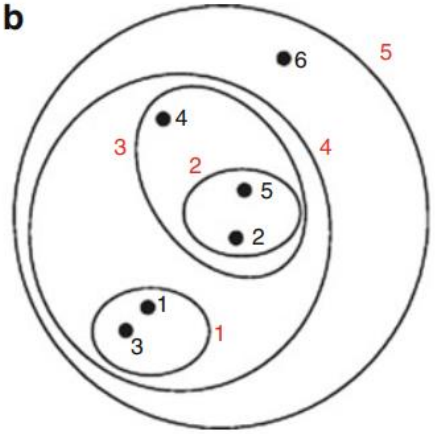
Baseado em Ligações – Agrupamento Hierárquico

- No agrupamento hierárquico as partições tem ligações entre si, através de regras mais gerais. Pode-se pensar em conjuntos e subconjuntos, ou ainda em uma organização em árvore.

Dendograma

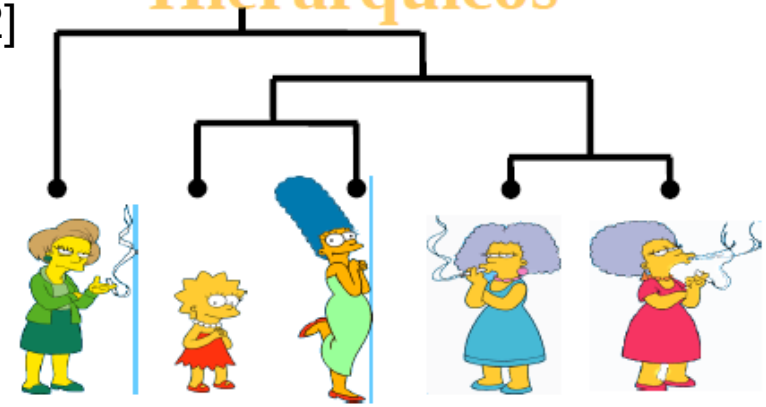


(DOUGHERTY, 2013)

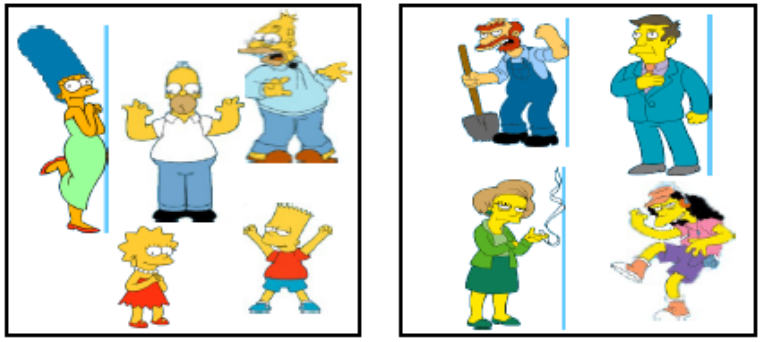


[2]

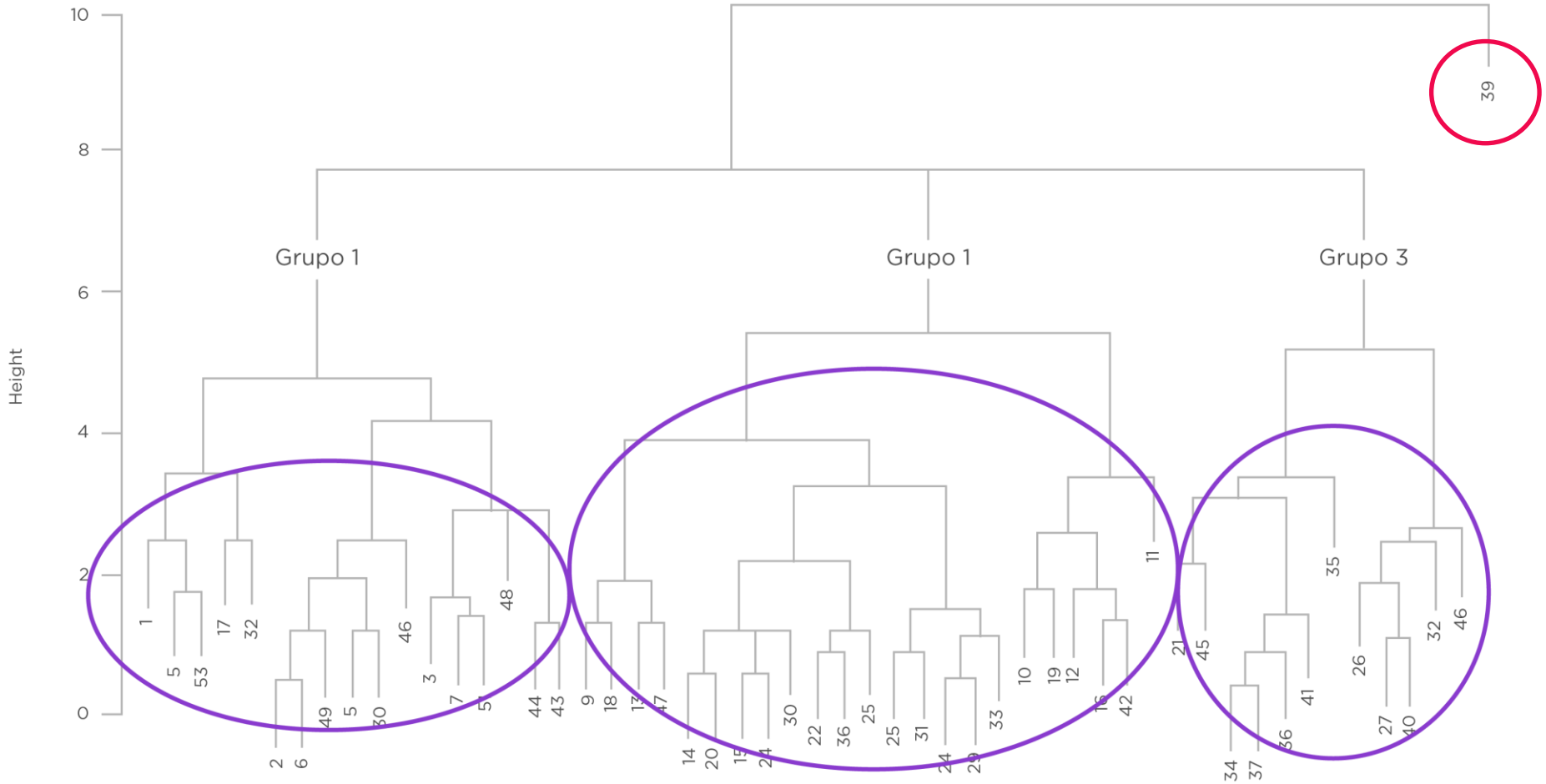
Hierárquicos



Particionais

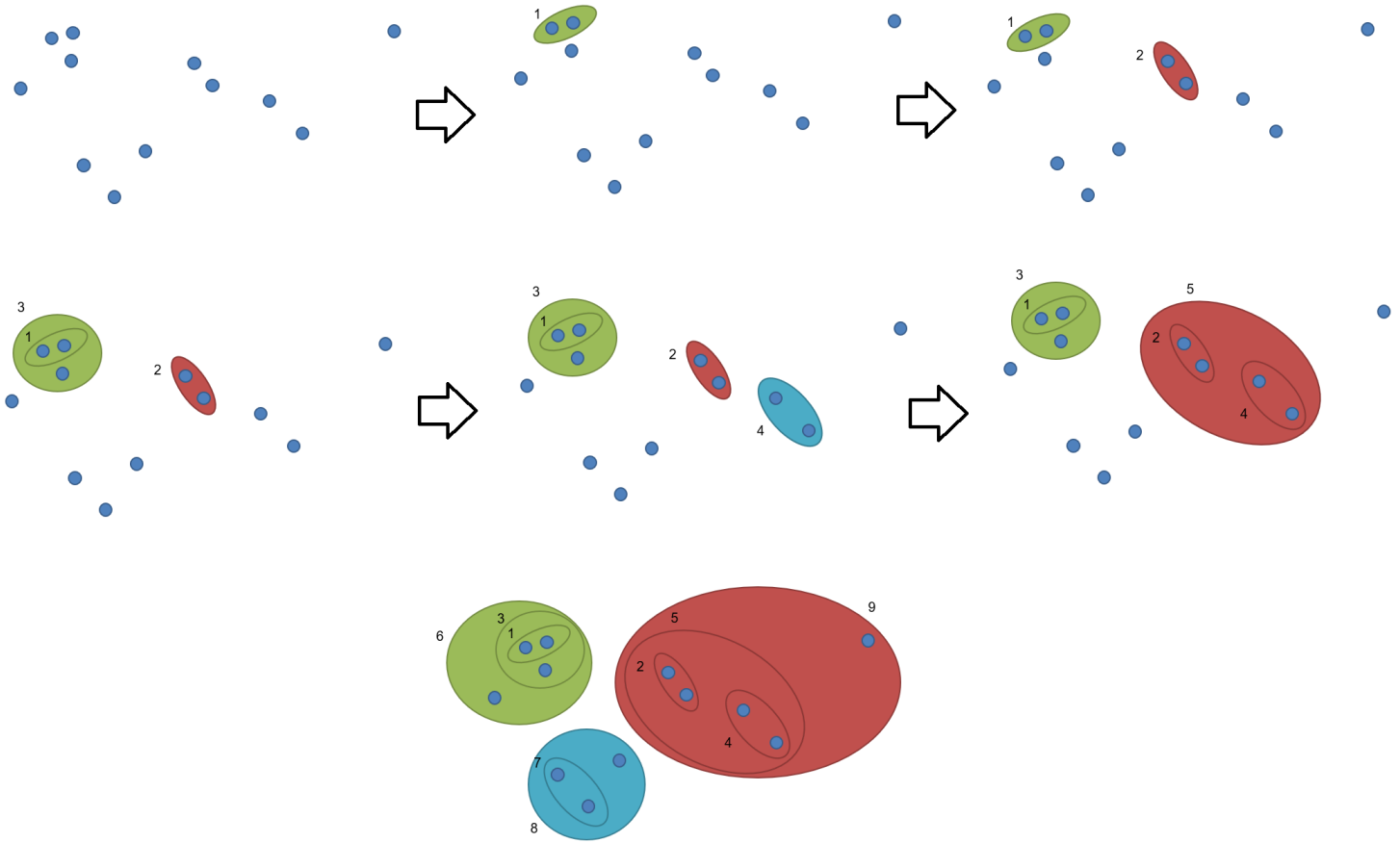


Baseado em Ligações – Agrupamento Hierárquico



Agrupamento Hierárquico – Algoritmo

- Podemos agrupar hierarquicamente indo de baixo para cima (**aglomerativos** bottom-up) ou de cima para baixo (**divisivos** top-down); Vejamos a sequência de um algoritmo aglomerativo:



Agrupamento Hierárquico - Hiperparâmetros

Quando estamos fazendo a união dos agrupamentos uma pergunta pode surgir: qual o critério que devemos adotar?

- Na biblioteca do scikit-learn temos:
  - **Ward**: Minimiza a variância dos agrupamentos que serão unidos
  - **Average**: Usa a média das distâncias entre cada dado dos dois clusters.
  - **Complete** or Maximum linkage: Usa a distância máxima entre os dados dos dois agrupamentos;
  - **Single**: usa o mínimo das distâncias entre todas as observações dos dois conjuntos;

Hierárquico – Vantagens e Desvantagens

- Vantagens:
  - Implementação simplificada;
  - Facilidade em lidar com qualquer medida de similaridade e por consequência, qualquer tipo de atributo.

- Desvantagens:
  - Dificuldade na definição de qual nível da árvore (dendrograma) melhor representa a clusterização;
  - Suscetível a outliers e ausência de normalização.

MÉTRICAS DE DESEMPENHO DE AGRUPAMENTO

Métricas de Desempenho

Se temos dois algoritmos de **Aprendizado de Máquina não Supervisionado de Agrupamento**, qual deles é “melhor” para resolver um determinado problema/dataset?

Em termos de “assertividade” (lembre-se, não temos os rótulos verdadeiros dos dados), precisamos de uma métrica de desempenho para poder comparar. Temos métricas de índice interno e de índice externo:

Interno: quão bom foi meu agrupamento “por si só”?

Externo: quão semelhantes foram duas técnicas de agrupamento?

INTERNO

- Silhouette
- Dunn
- Davies-Bouldin

EXTERNO

- Rand Index
- Adjusted Rand Index
- Jaccard
- Folkes e Mallows

Rand Index

O Rand Index é utilizado para comparar coincidência de dois métodos distintos de agrupamento ou métodos com critérios distintos;

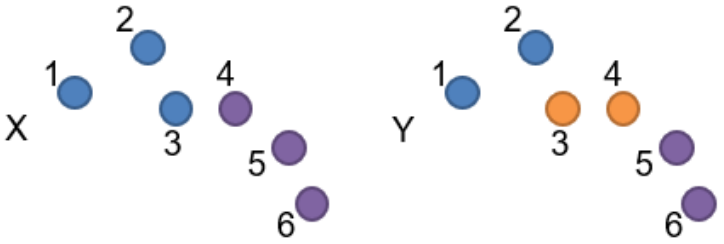
O Rand Index utiliza os seguintes conceitos:

a: número de **pares de elementos** nos mesmos clusters para método X e método Y

b: número de pares de elementos em diferentes clusters para método X e método Y

c: número de pares de elementos no mesmo cluster para o método X em diferentes clusters para o método Y

d: número de pares de elementos em diferentes clusters para o método X e nos mesmos clusters para o método Y



labels_X = [0, 0, 0, 1, 1, 1]
labels_Y = [0, 0, 1, 1, 2, 2]

a = [(1,2);(5,6)] = 2 pares

b = [(1,4),(1,5),(1,6),(2,4),(2,5)(2,6); (3,5),(3,6)] = 8 pares

c = [(1,3);(2;3);(4,5);(4,6)] = 4 pares

d = [(3,4)] = 1 par

Rand Index

O Rand Index é definido como:

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

No nosso exemplo:

$$R = \frac{2+8}{2+8+4+1} = 0{,}667$$

Método 1:

Método 2:

```
>>> from sklearn import metrics
>>> labels_true = [0, 0, 0, 1, 1, 1]
>>> labels_pred = [0, 0, 1, 1, 2, 2]
>>> metrics.rand_score(labels_true, labels_pred)
0.66...
```

Adjusted Rand Index

Um dos problemas desse método é que caso sejam fornecidos dois datasets com labels gerados aleatoriamente, o *Rand Index* não nos garante um resultado igual a zero.

Para isso, surge o *Adjusted Rand index:*

$$AR = \frac{\binom{n}{2}(a+d) - [(a+b)(a+c)+(c+d)(b+d)]}{\binom{n}{2}^2[(a+b)(a+c)+(c+d)(b+d)]}$$

Onde:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

```
>>> metrics.rand_score(labels_true, labels_pred)
0.66...
>>> metrics.adjusted_rand_score(labels_true, labels_pred)
0.24...
```

Adjusted Rand Index

FIAP

Labels que estão muito diferentes terão valores negativos ou próximos de 0 com o **Adjusted Rand Index**, enquanto que com o **Rand Index** o valor pode não ser tão baixo assim:

```
>>> labels_true = [0, 0, 0, 0, 0, 0, 1, 1]
>>> labels_pred = [0, 1, 2, 3, 4, 5, 5, 6]
>>> metrics.rand_score(labels_true, labels_pred)
0.39...
>>> metrics.adjusted_rand_score(labels_true, labels_pred)
-0.07...
```

Labels idênticos tem resultado igual a 1:

```
>>> labels_pred = labels_true[:]
>>> metrics.rand_score(labels_true, labels_pred)
1.0
>>> metrics.adjusted_rand_score(labels_true, labels_pred)
1.0
```

Silhouette

O Silhouette é uma métrica que avalia o “formato” dos clusters obtidos.

Ele é obtido calculando a distância média entre um dado de um agrupamento com todos os outros dados do mesmo cluster (a) e com a média desse mesmo dado com todos os dados do agrupamento mais próximo.

Essa métrica é definida como:

$$Silhouette(i) = \frac{b_i - a_i}{\max(a_i, b_i)}$$

b: distância entre i e o agrupamento mais próximo

a: distância média entre i e todos os pontos de seu agrupamento

Silhouette

A média do Silhouette de todos os dados nos define o quão bom é o nosso agrupamento:

$$-1 \leq Silhouette \leq 1$$

Dados foram atribuídos aos grupos de forma errada

Agrupamentos com problemas

Agrupamentos bem separados e é possível distingui-los

Agrupamentos ok

Referências

[1] Ronaldo Prati, Mineração de Dados, UFABC 2021.

[2] Sarajane Peres e Clodoaldo Lima, Técnicas de Agrupamento, USP 2015.

[3] Andriy Burkov, Machien Learning Engineering, 2020.

[4] Aurélien Géron, Handos-on Machine Learning with Scikit-Learn & TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, O'Reilly, 2017.

FIAP GRADUAÇÃO