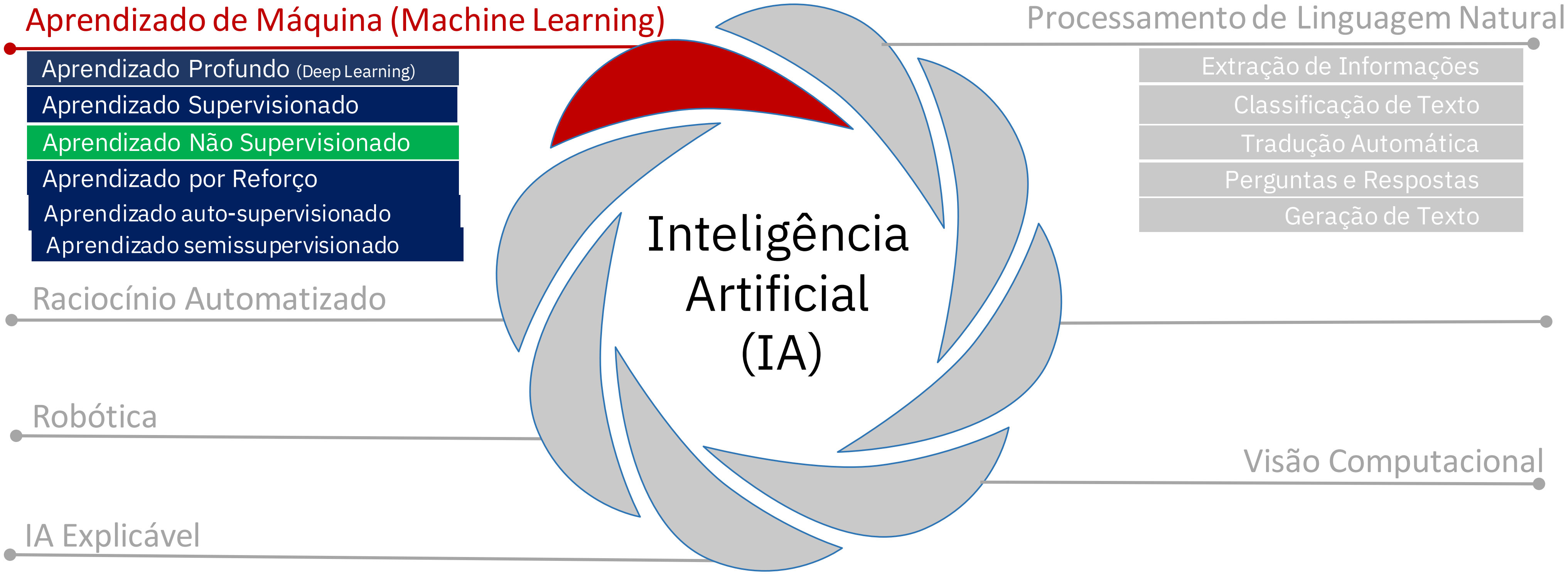
Aprendizado não supervisionado: parte 2

Profa Silvia Moraes

Subáreas da Inteligência Artificial



Aprendizado Não Supervisionado

- **Não exige** que os **dados** estejam **rotulados**
- Sem crítica, **usa regularidades e propriedades estatísticas** dos dados no processo de aprendizagem.

Entrada (x):
Imagem ou características

Dataset Sem Anotação

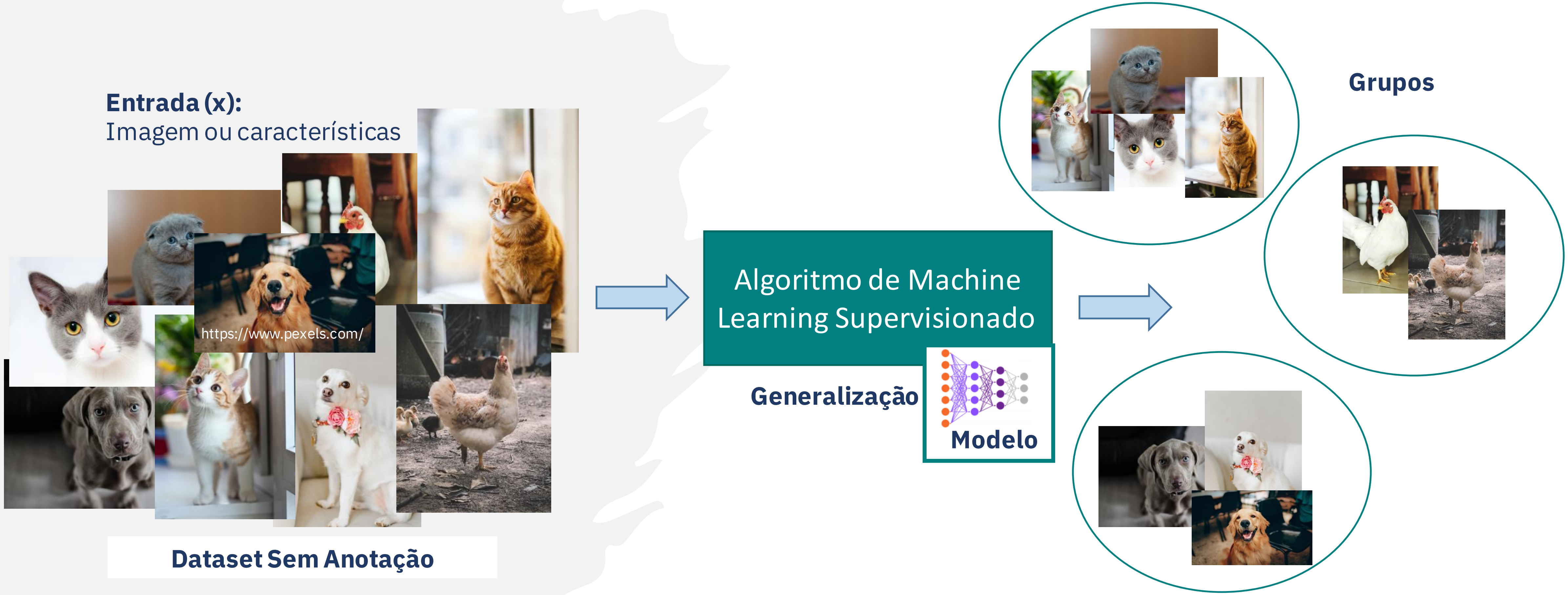
Aprendizado Não Supervisionado

Executa **tarefas descritivas**: explora ou descreve um conjunto de dados.

- **Agrupamento**: divisão em grupos baseada em alguma regularidade ou similaridade
- **Sumarização**: descrição simples e compacta
- **Associação**: relações frequentes entre dados

Agrupamento

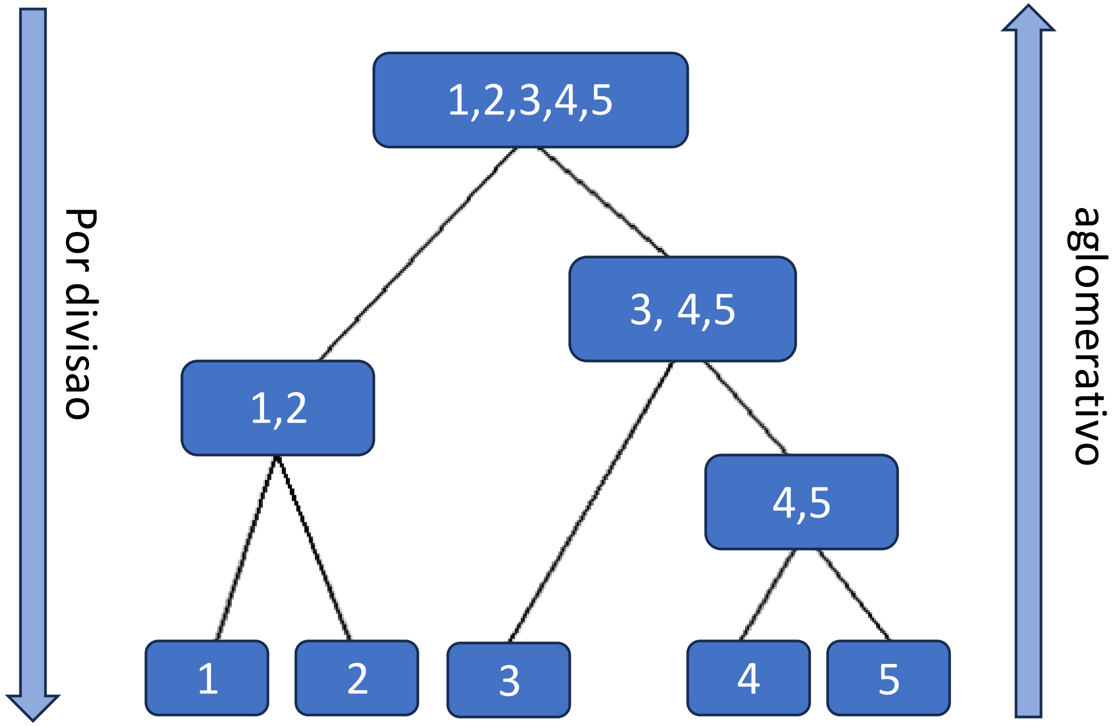
Organiza dados (não classificados, sem rótulos) em grupos de acordo com alguma medida de similaridade.



Agrupamento Hierárquico

Agrupamento Hierárquico

- Um algoritmo desse tipo gera a partir de uma matriz de proximidade, uma série de partições aninhadas.
- Pode ser:
  - Aglomerativo
  - Por divisão

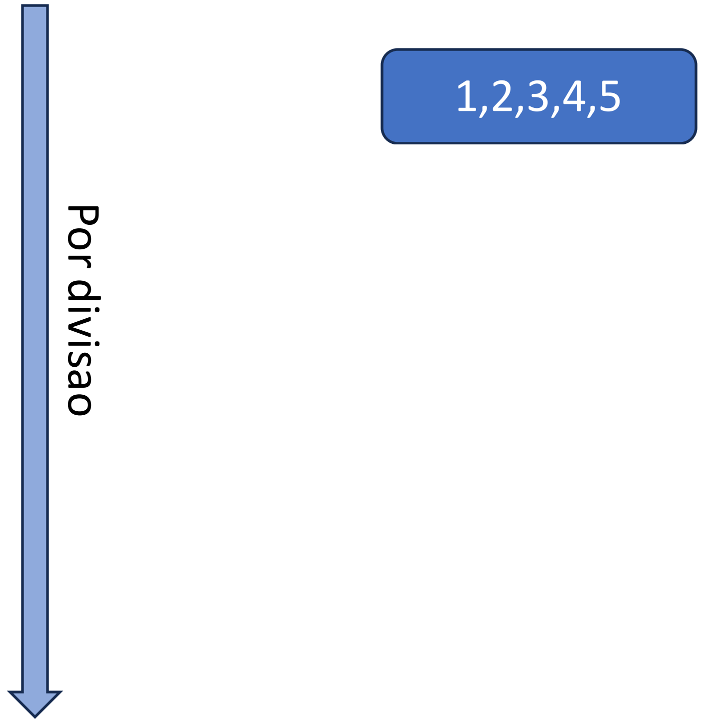


Agrupamento Hierárquico

- **Por Divisão:**
  - Inicia com todos os objetos representando um único grupo.
  - A cada passo, divide o grupo até que cada grupo contenha um objeto (ou k grupos) ou algum critério de parada seja atingido.
  - Para definir os grupos:
    - usa uma **matriz de similaridade ou distância**.

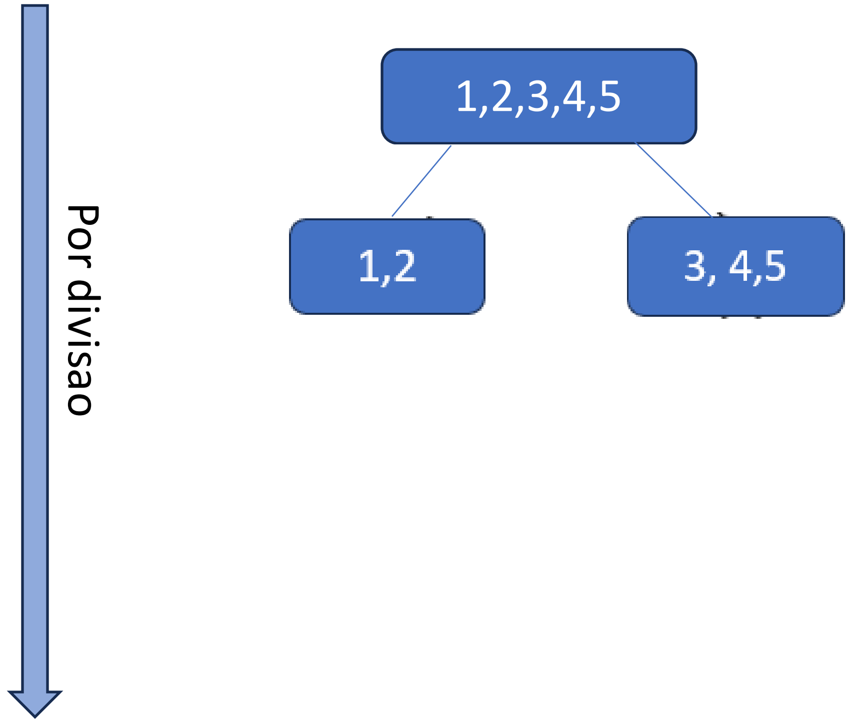
Agrupamento Hierárquico

- **Por divisão:**



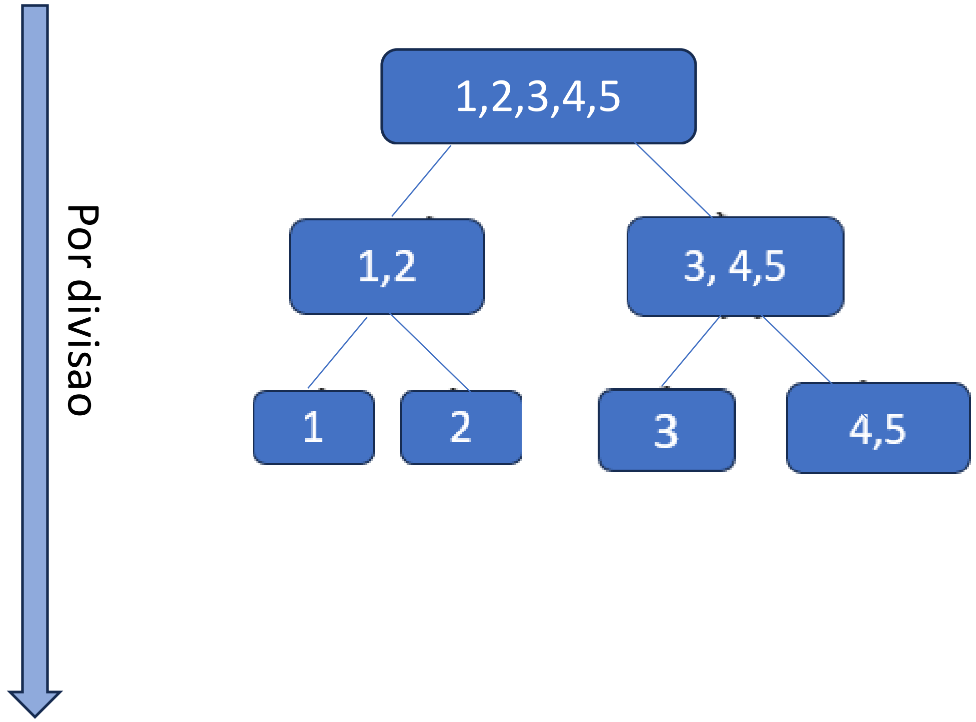
Agrupamento Hierárquico

- **Por divisão:**



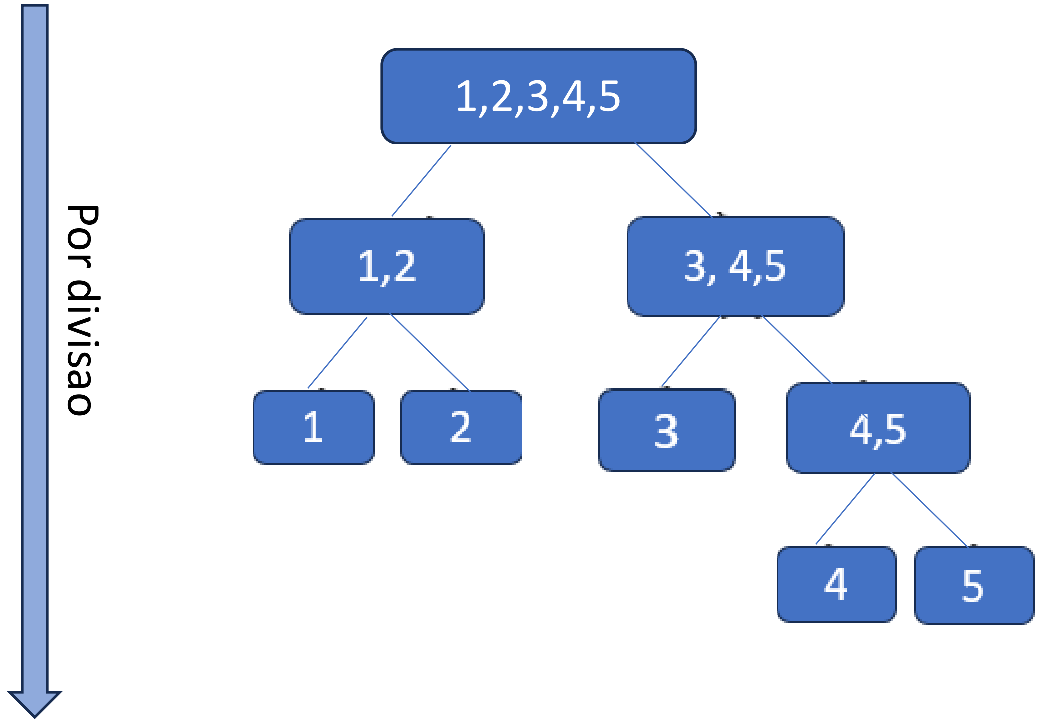
Agrupamento Hierárquico

- **Por divisão:**



Agrupamento Hierárquico

- **Por divisão:**



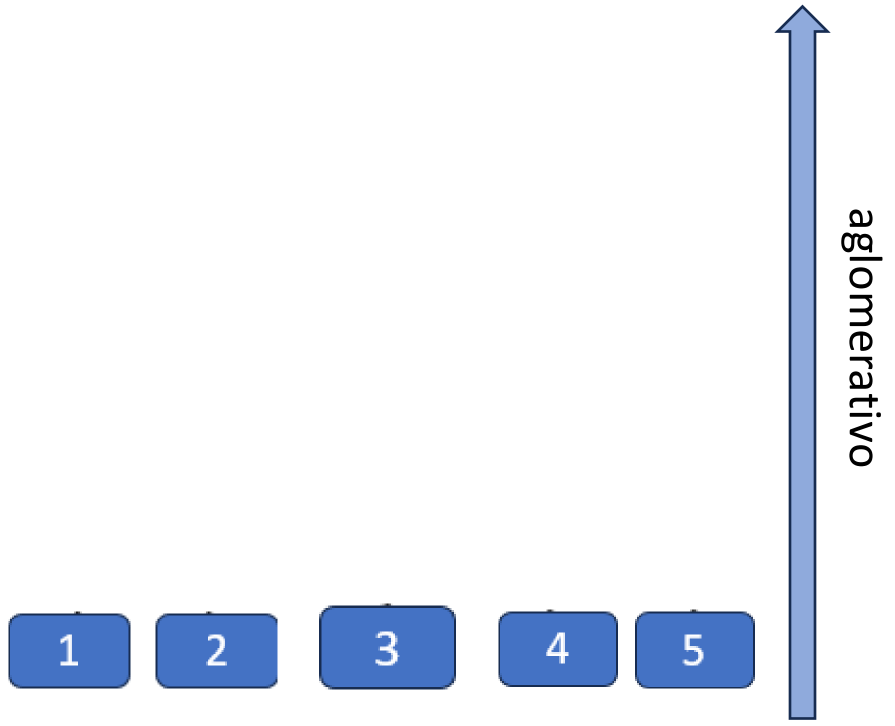
Agrupamento Hierárquico

- **Aglomerativo:**
  - Inicia com cada objeto representando um grupo individual.
  - A cada passo, combina o par mais próximo de grupos, até que somente um grupo (ou k grupos) reste ou algum critério de parada seja atingido.

- Para definir os grupos: algoritmos hierárquicos tradicionais usam a **matriz de similaridade ou distância**

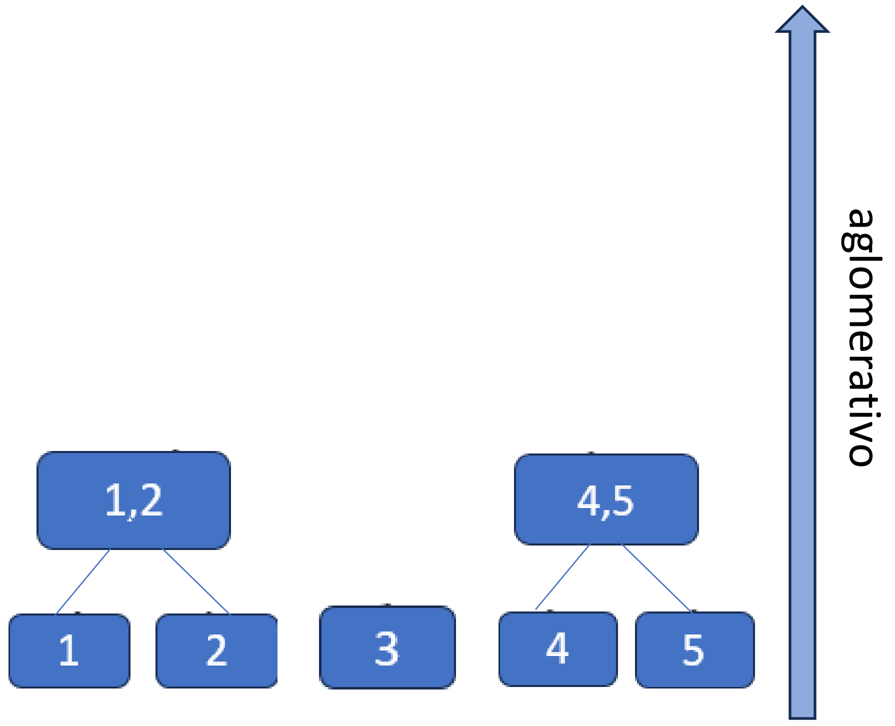
Agrupamento Hierárquico

- **Aglomerativo:**



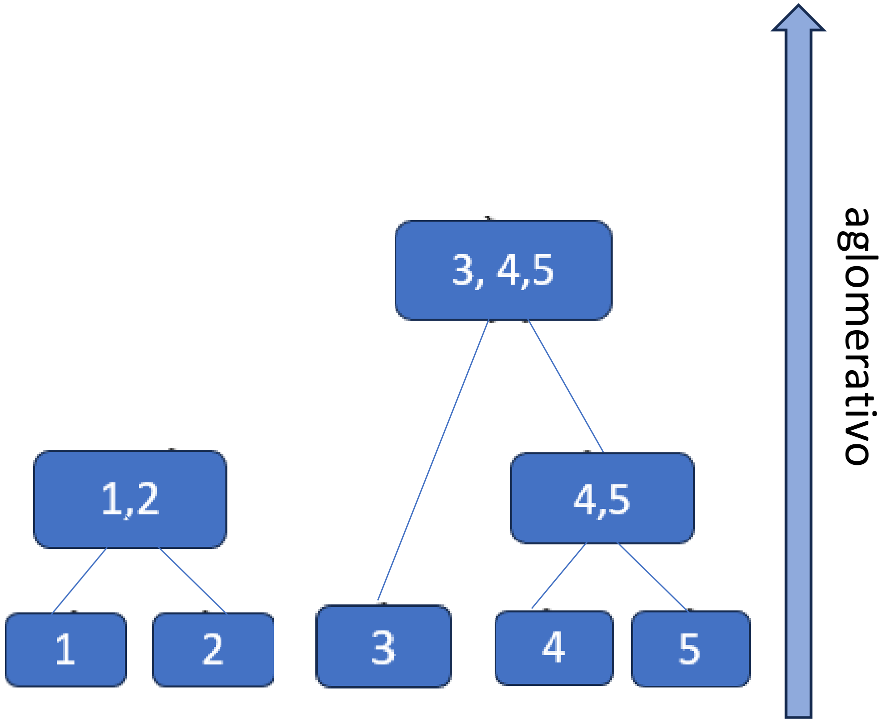
Agrupamento Hierárquico

- **Aglomerativo:**



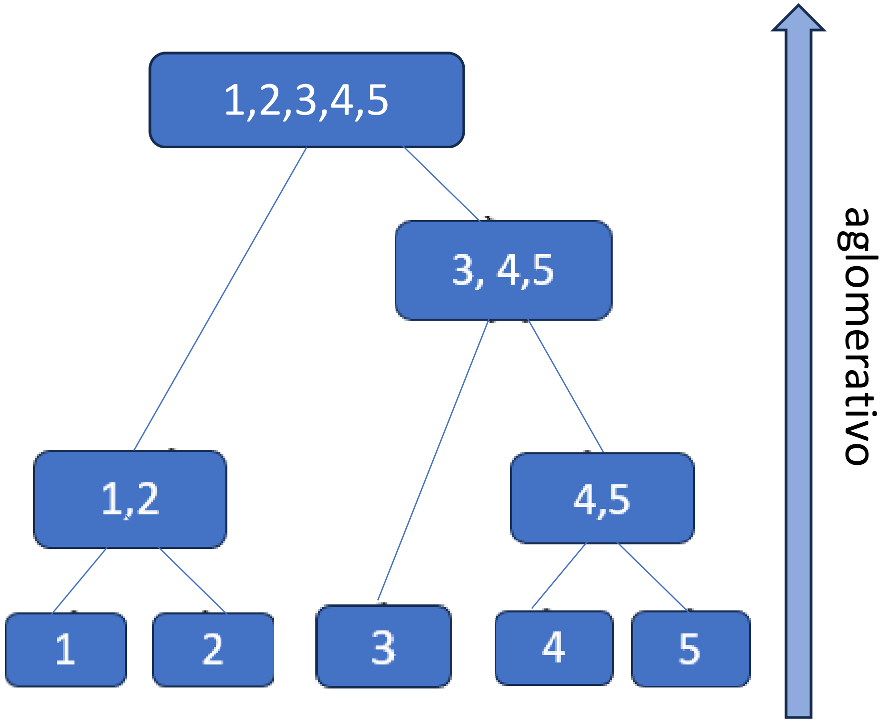
Agrupamento Hierárquico

- **Aglomerativo:**



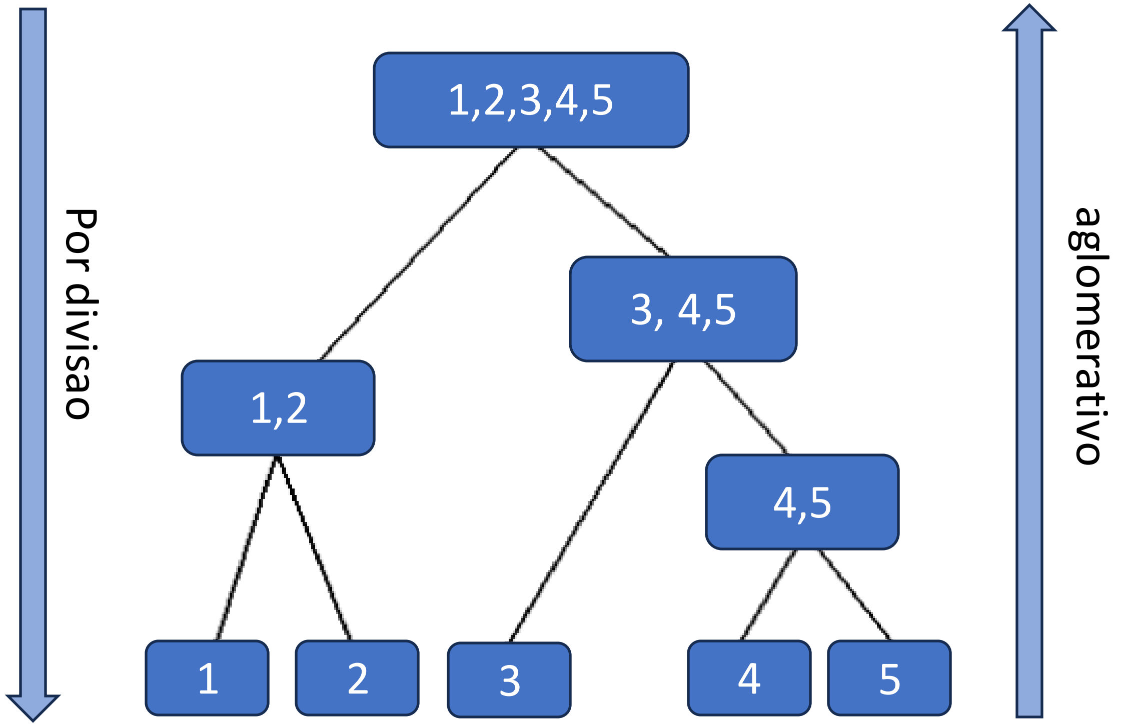
Agrupamento Hierárquico

- **Aglomerativo:**



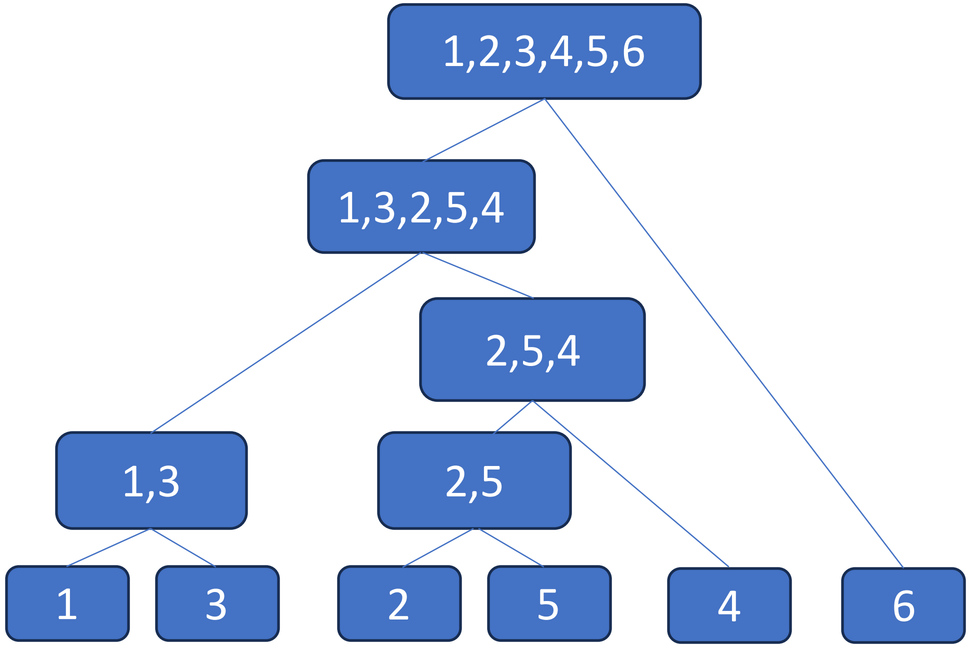
Agrupamento Hierárquico

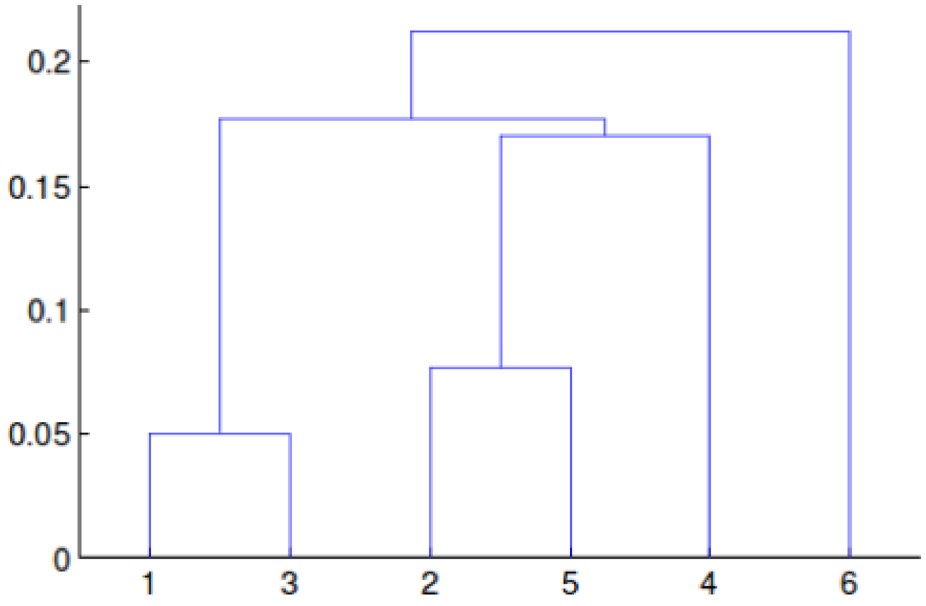
- Um algoritmo desse tipo gera a partir de uma matriz de proximidade, uma série de partições aninhadas.
- Pode ser:
  - Aglomerativo
  - Por divisão



Agrupamento Hierárquico

- Produz um conjunto de grupos aninhados, organizados como uma árvore, mostrando partições ou combinações.
- Pode ser visualizado por um dendograma..



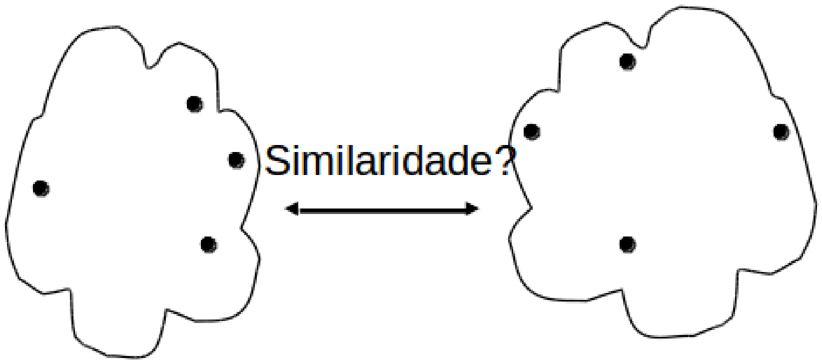


Agrupamento Hierárquico

- **Vantagens desse tipo de agrupamento:**
  - Não é necessário assumir um número particular de grupos
  - Qualquer número de grupos desejado pode ser obtido ao ‘cortar’ o dendograma no nível apropriado.
  - Podem corresponder a taxonomias úteis. Exemplos em ciências biológicas (e.g., reino animal, reconstrução filogênica, . . .).

Agrupamento Hierárquico

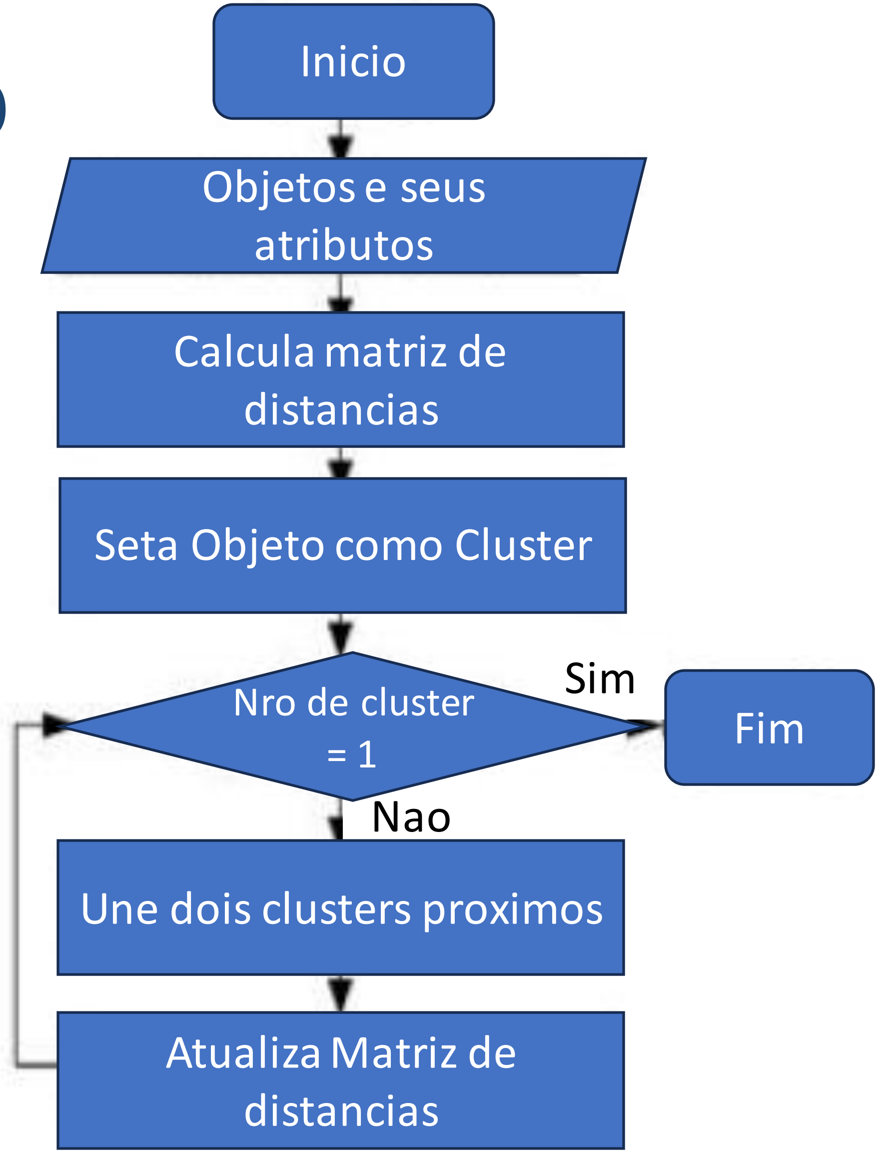
- Similaridade baseada nas distâncias entre os elementos ou centróides dos clusters.
- Em geral usam uma matriz de similaridade/distancia



	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						

Agrupamento Hierárquico

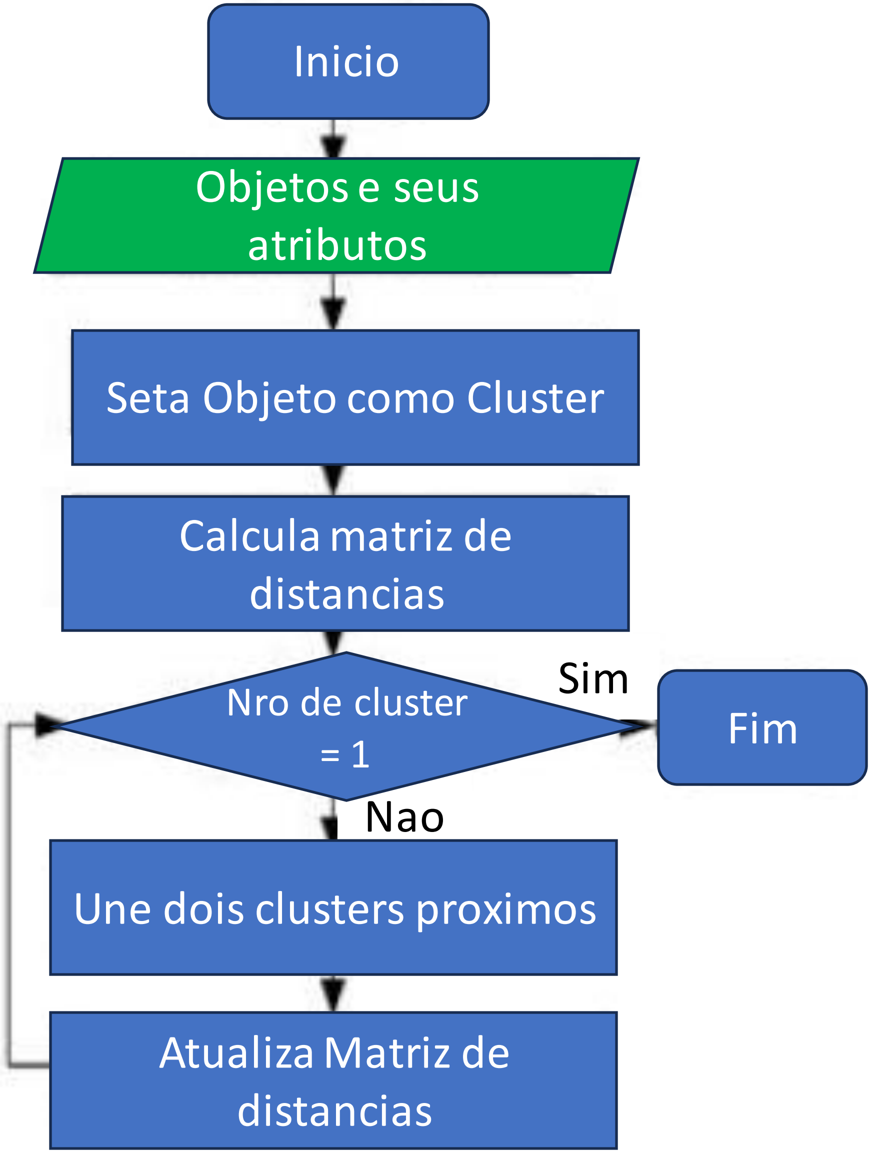
- Os algoritmos aglomerativos são mais usados.
- A operação-chave é a distância entre os dois clusters.
- A diferença entre os algoritmos que seguem essa abordagem está justamente no cálculo dessa distância



Agrupamento Hierárquico

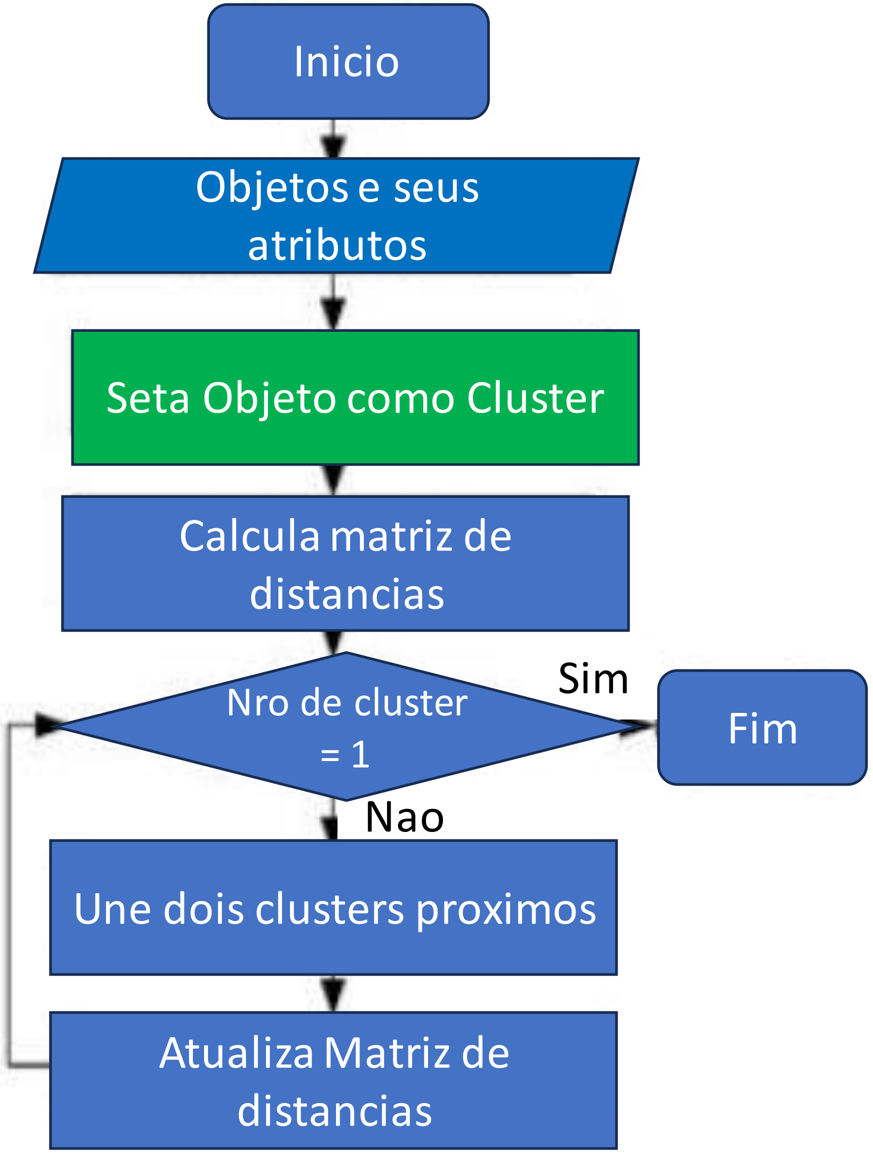
Exemplo:

- Considere os objetos (pontos):
  - A : (1,1)
  - B: (2,1)
  - C: (3,4)
  - D: (4,6)
  - E: (5,3)
  - F: (6,4)
  - G: (7,3)



Agrupamento Hierárquico

Exemplo:



A(1,1)	B(2,1)	C(3,4)	D(4,6)	E(5,3)	F(6,4)	G(7,3)
C1	C2	C3	C4	C5	C6	C7

Agrupamento Hierárquico

Exemplo:

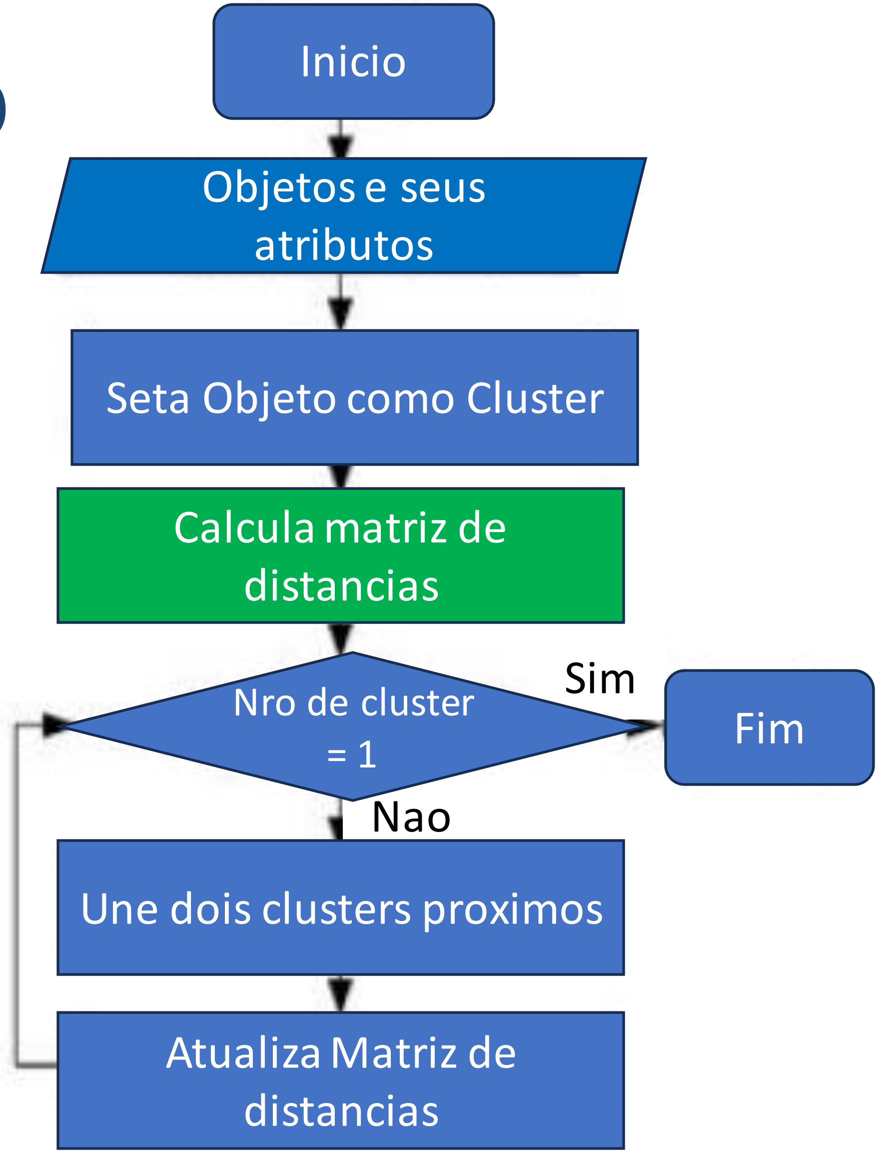
- Considere os objetos (pontos):

A : (1,1), B: (2,1), C: (3,4), D: (4,6), E: (5,3), F: (6,4), G: (7,3),

		A	B	C	D	E	F	G
C1	A	0	1	5	8	6	8	8
C2	B	1	0					
C3	C	5		0				
C4	D	8			0			
C5	E	6				0		
C6	F	8					0	
C7	G	8						0

Manhattan: $d(x_i, x_j) = \sum_{l=1}^{d} |x_i^l - x_j^l|$

dist(A,B) =|1-2|+|1-1|=1
dist(A,C) =|1-3|+|1-4|=5
dist(A,D) =|1-4|+|1-6|=8
dist(A,E) =|1-5|+|1-3|=6
dist(A,F) =|1-6|+|1-4|=8
dist(A,G) =|1-7|+|1-3|=8



Agrupamento Hierárquico

Exemplo:

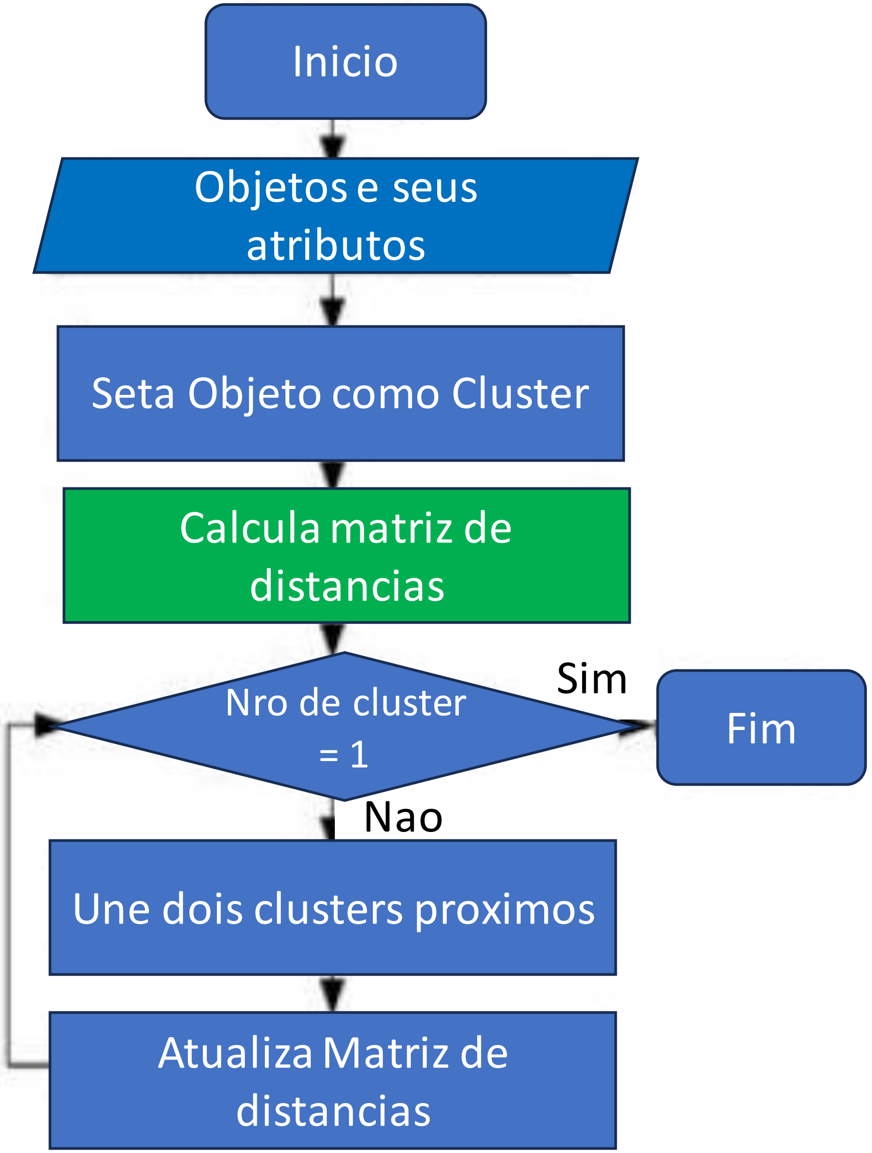
- Considere os objetos (pontos):

A : (1,1), B: (2,1), C: (3,4), D: (4,6), E: (5,3), F: (6,4), G: (7,3),

		A	B	C	D	E	F	G
C1	A	0	1	5	8	6	8	8
C2	B	1	0	4	7	5	7	7
C3	C	5	4	0				
C4	D	8	7		0			
C5	E	6	5			0		
C6	F	8	7				0	
C7	G	8	7					0

dist(B,C) =|2-3|+|1-4|=4
dist(B,D) =|2-4|+|1-6|=7
dist(B,E) =|2-5|+|1-3|=5
dist(B,F) =|2-6|+|1-4|=7
dist(B,G) =|2-7|+|1-3|=7

Manhattan: $d(x_i, x_j) = \sum_{l=1}^{d} |x_i^l - x_j^l|$



Agrupamento Hierárquico

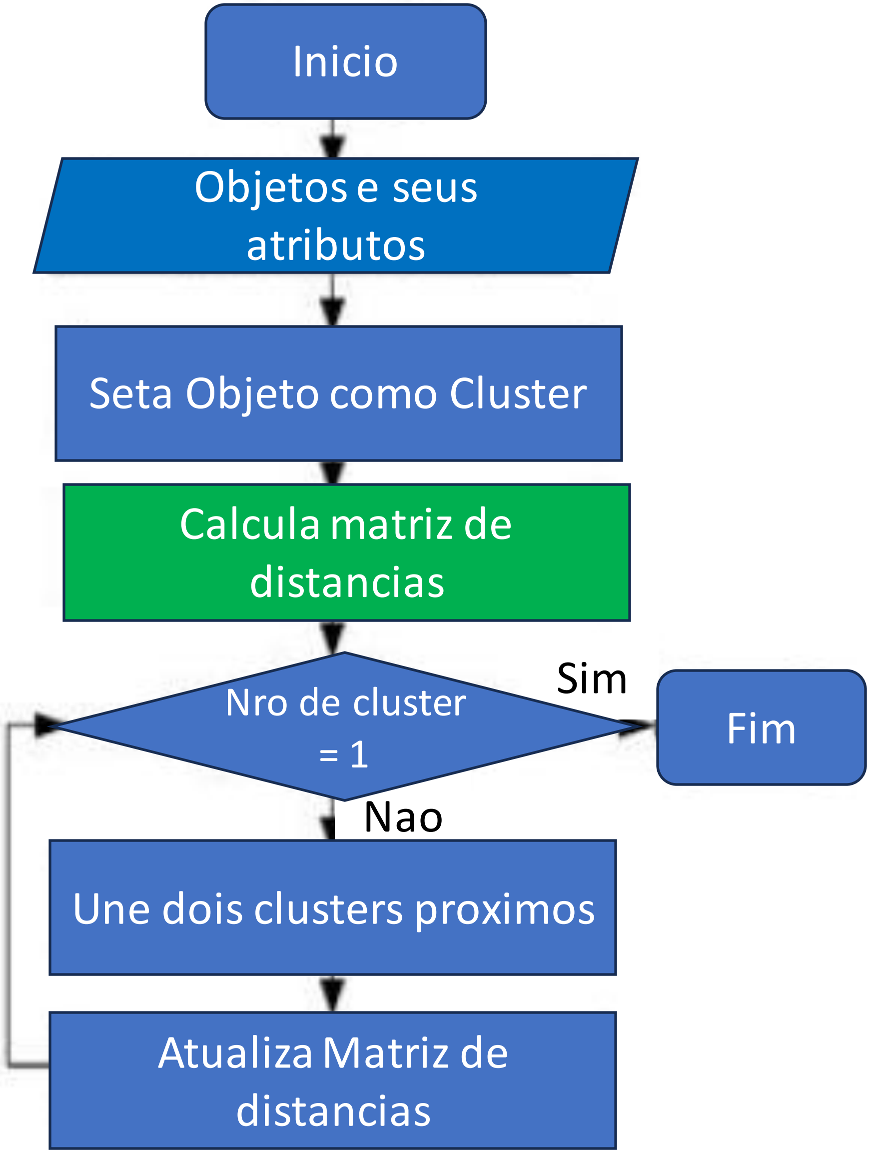
Exemplo:

- Considere os objetos (pontos):
  A : (1,1), B: (2,1), C: (3,4), D: (4,6), E: (5,3), F: (6,4), G: (7,3),

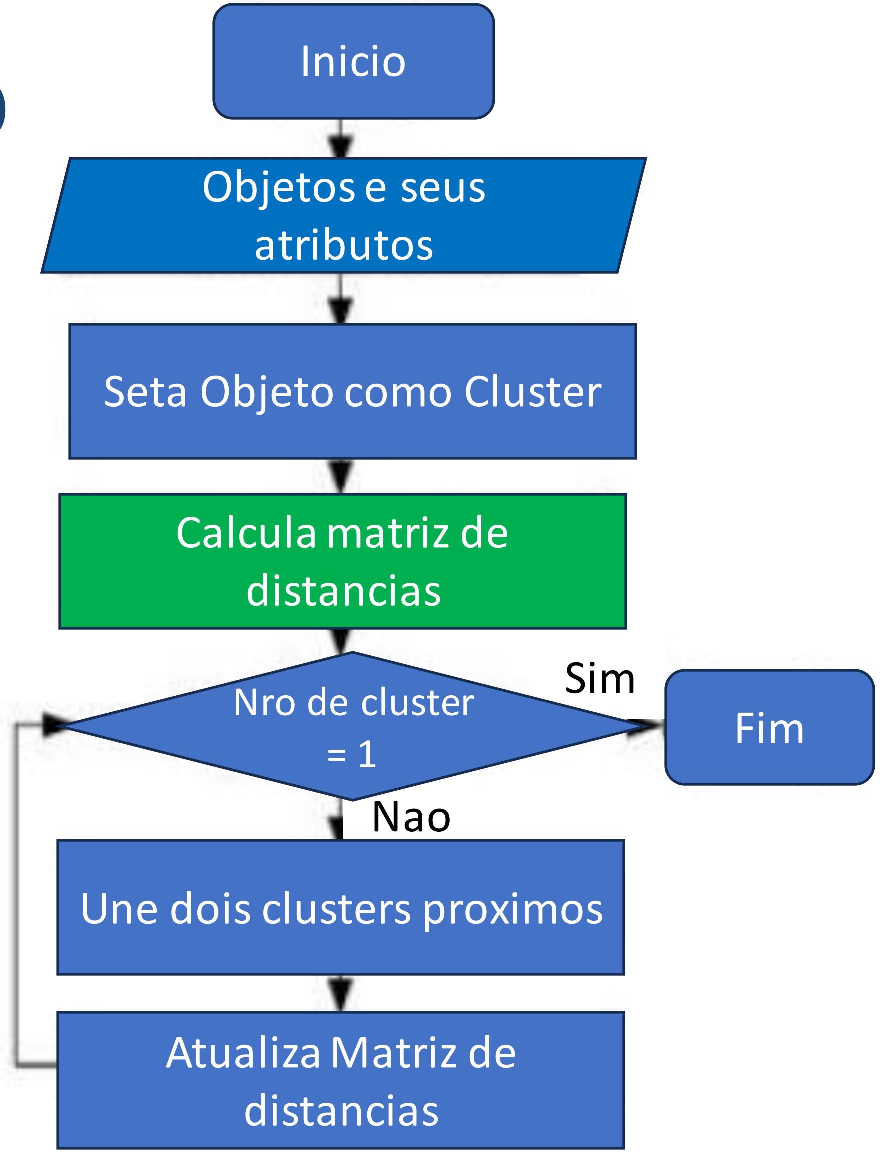
		A	B	C	D	E	F	G
C1	A	0	1	5	8	6	8	8
C2	B	1	0	4	7	5	7	7
C3	C	5	4	0	3	3	3	5
C4	D	8	7	3	0			
C5	E	6	5	3		0		
C6	F	8	7	3			0	
C7	G	8	7	5				0

dist(C,D) =|3-4|+|4-6|=3
dist(C,E) =|3-5|+|4-3|=3
dist(C,F) =|3-6|+|4-4|=3
dist(C,G) =|3-7|+|4-3|=5

Manhattan: $d(x_i, x_j) = \sum_{l=1}^{d} |x_i^l - x_j^l|$



Agrupamento Hierárquico



Exemplo:

- Considere os objetos (pontos):
  A : (1,1), B: (2,1), C: (3,4), D: (4,6), E: (5,3), F: (6,4), G: (7,3),

		A	B	C	D	E	F	G
C1	A	0	1	5	8	6	8	8
C2	B	1	0	4	7	5	7	7
C3	C	5	4	0	3	3	3	5
C4	D	8	7	3	0	4	4	6
C5	E	6	5	3	4	0		
C6	F	8	7	3	4		0	
C7	G	8	7	5	6			0

dist(D,E) =|4-5|+|6-3|=4
dist(D,F) =|4-6|+|6-4|=4
dist(D,G) =|4-7|+|6-3|=6

Manhattan: $d(x_i, x_j) = \sum_{l=1}^{a} |x_i^l - x_j^l|$

Agrupamento Hierárquico

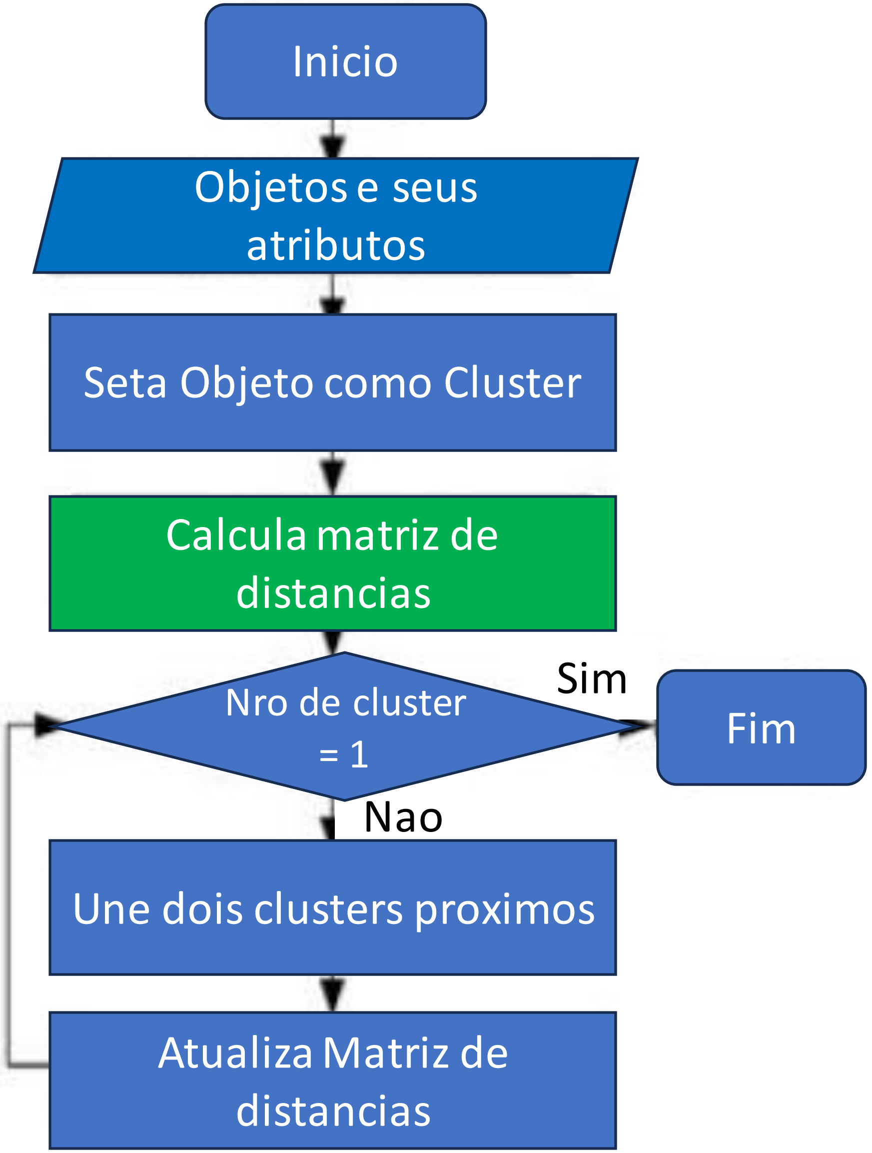
Exemplo:

- Considere os objetos (pontos):
  A : (1,1), B: (2,1), C: (3,4), D: (4,6), E: (5,3), F: (6,4), G: (7,3),

		A	B	C	D	E	F	G
C1	A	0	1	5	8	6	8	8
C2	B	1	0	4	7	5	7	7
C3	C	5	4	0	3	3	3	5
C4	D	8	7	3	0	4	4	6
C5	E	6	5	3	4	0	2	2
C6	F	8	7	3	4	2	0	
C7	G	8	7	5	6	2		0

dist(E,F) =|5-6|+|3-4|=2
dist(E,G) =|5-7|+|3-3|=2

Manhattan: $d(x_i, x_j) = \sum_{l=1}^{d} |x_i^l - x_j^l|$



Agrupamento Hierárquico

Exemplo:

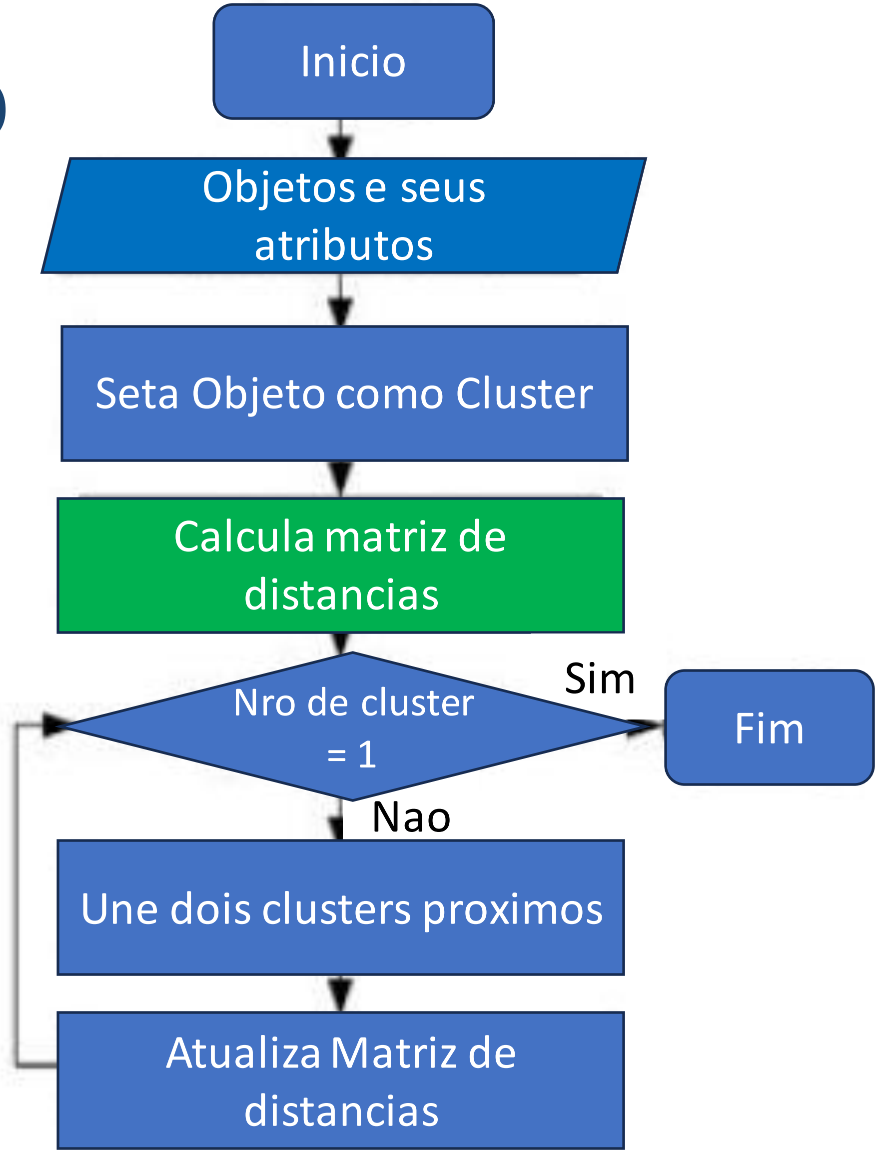
- Considere os objetos (pontos):

  A : (1,1), B: (2,1), C: (3,4), D: (4,6), E: (5,3), F: (6,4), G: (7,3),

	A	B	C	D	E	F	G
A	0	1	5	8	6	8	8
B	1	0	4	7	5	7	7
C	5	4	0	3	3	3	5
D	8	7	3	0	4	4	6
E	6	5	3	4	0	2	2
F	8	7	3	4	2	0	2
G	8	7	5	6	2	2	0

dist(F,G) =|6-7|+|4-3|=2

Manhattan: $d(x_i, x_j) = \sum_{l=1}^{d} |x_i^l - x_j^l|$



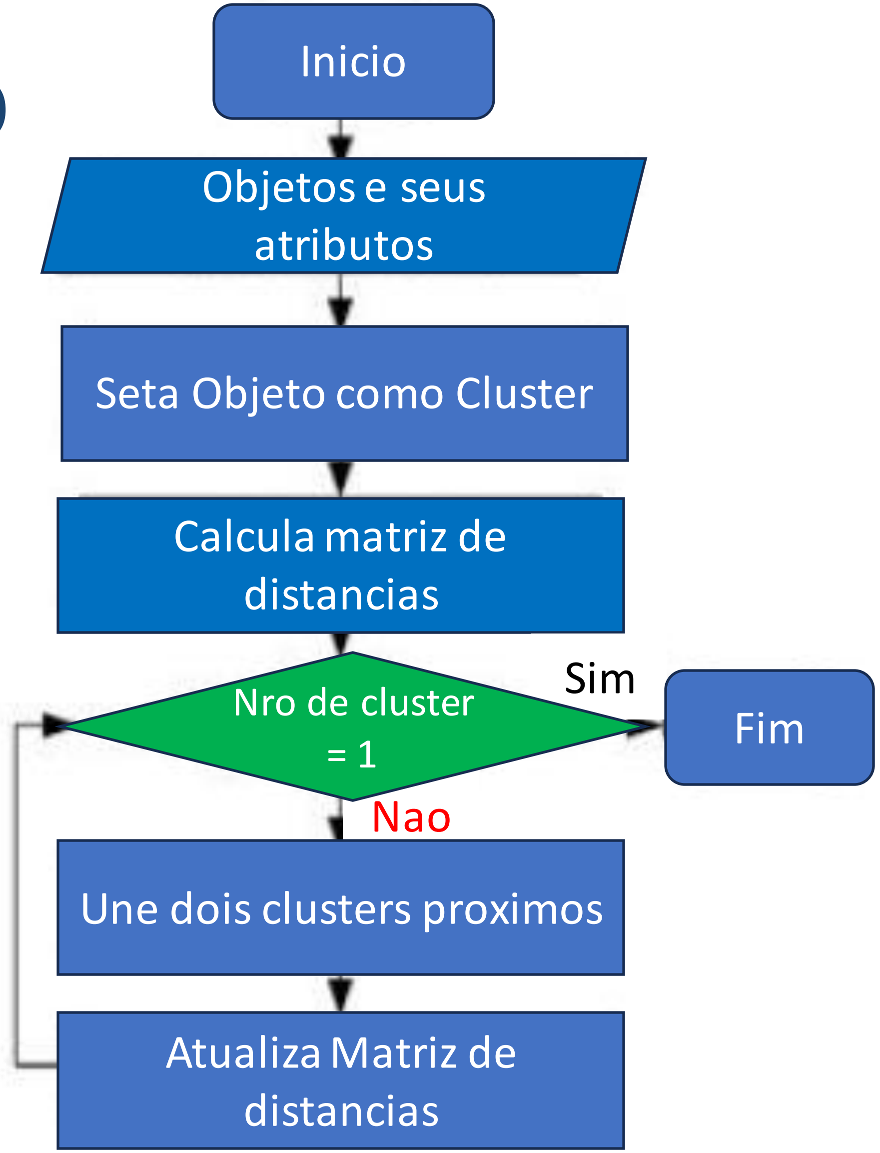
Agrupamento Hierárquico

Exemplo:

- Considere os objetos (pontos):
  A : (1,1), B: (2,1), C: (3,4), D: (4,6), E: (5,3), F: (6,4), G: (7,3),

	A	B	C	D	E	F	G
A	0	1	5	8	6	8	8
B	1	0	4	7	5	7	7
C	5	4	0	3	3	3	5
D	8	7	3	0	4	4	6
E	6	5	3	4	0	2	2
F	8	7	3	4	2	0	2
G	8	7	5	6	2	2	0

Manhattan: $d(x_i, x_j) = \sum_{l=1}^{d} |x_i^l - x_j^l|$



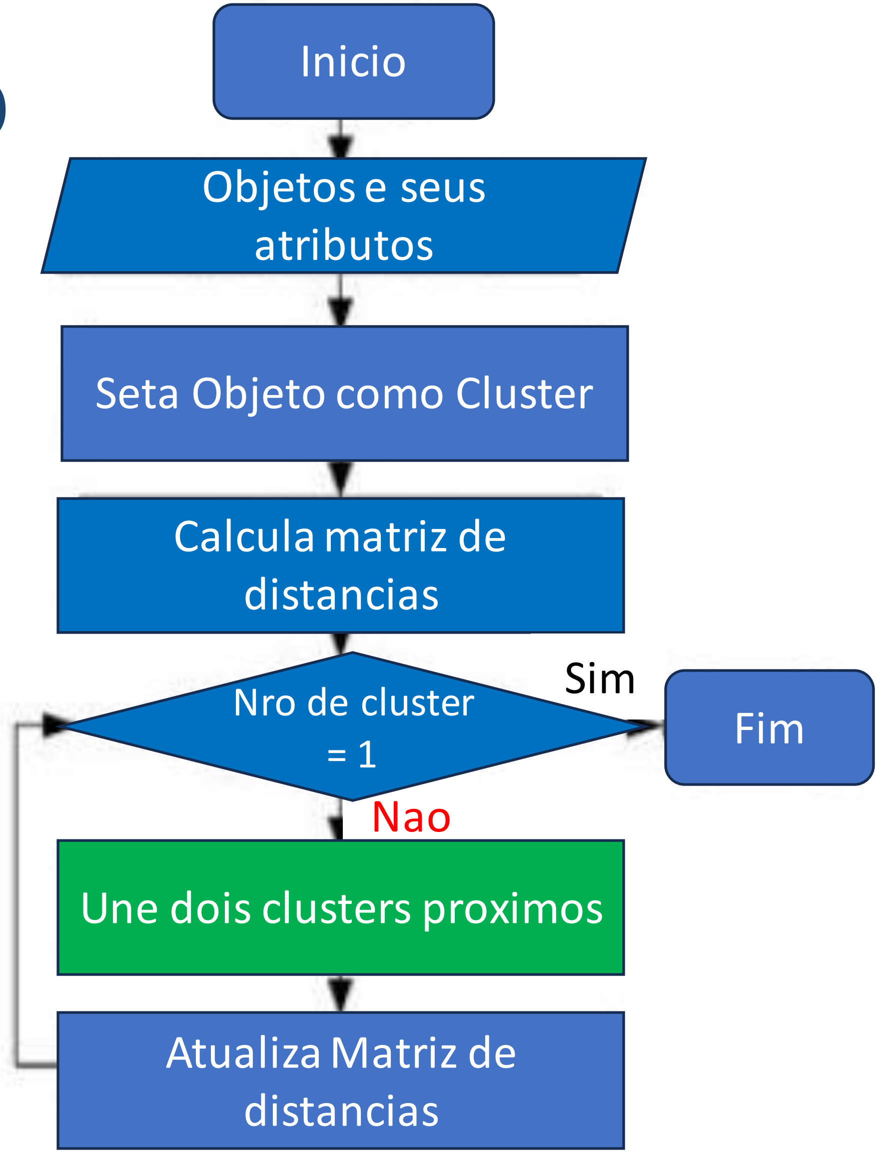
Agrupamento Hierárquico

Exemplo:

- Considere os objetos (pontos):
  A : (1,1), B: (2,1), C: (3,4), D: (4,6), E: (5,3), F: (6,4), G: (7,3),

	A	B	C	D	E	F	G
A	0	1	5	8	6	8	8
B	1	0	4	7	5	7	7
C	5	4	0	3	3	3	5
D	8	7	3	0	4	4	6
E	6	5	3	4	0	2	2
F	8	7	3	4	2	0	1
G	8	7	5	6	2	1	0

Manhattan: $d(x_i, x_j) = \sum_{l=1}^{d} |x_i^l - x_j^l|$



Agrupamento Hierárquico

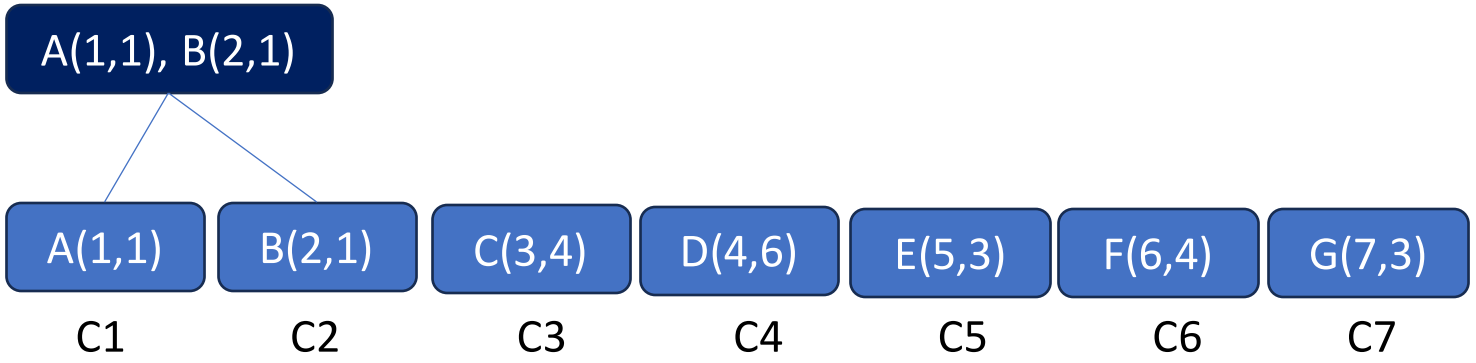
Exemplo:

	A	B	C	D	E	F	G
A	0	1	5	8	6	8	8
B	1	0	4	7	5	7	7
C	5	4	0	3	3	3	5
D	8	7	3	0	4	4	6
E	6	5	3	4	0	2	2
F	8	7	3	4	2	0	2
G	8	7	5	6	2	2	0

Une dois clusters proximos

A,B -> distancia =1

Metrica de integracao = menor distancia



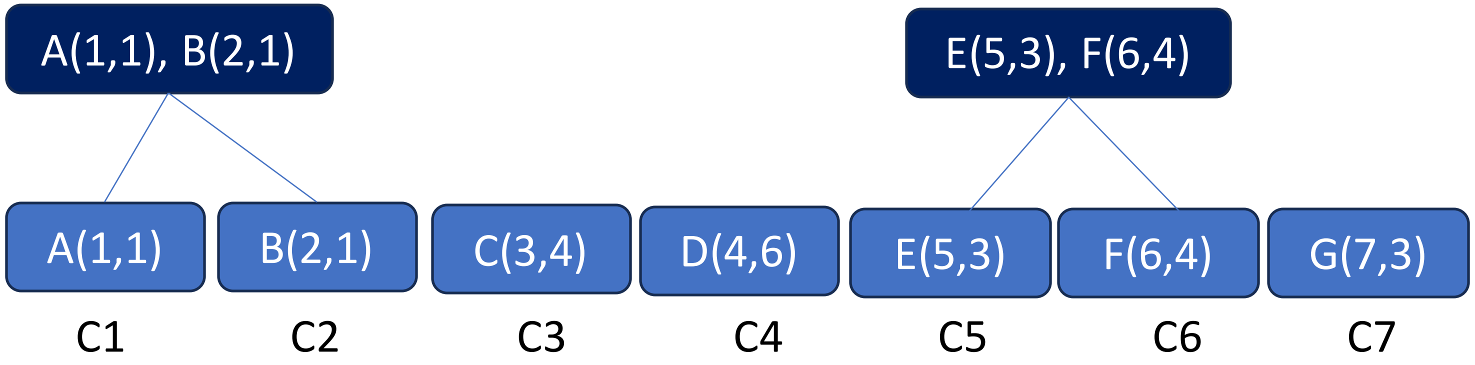
Agrupamento Hierárquico

Exemplo:

	A	B	C	D	E	F	G
A	0	1	5	8	6	8	8
B	1	0	4	7	5	7	7
C	5	4	0	3	3	3	5
D	8	7	3	0	4	4	6
E	6	5	3	4	0	2	2
F	8	7	3	4	2	0	2
G	8	7	5	6	2	2	0

Une dois clusters proximos

E,F -> distancia =2



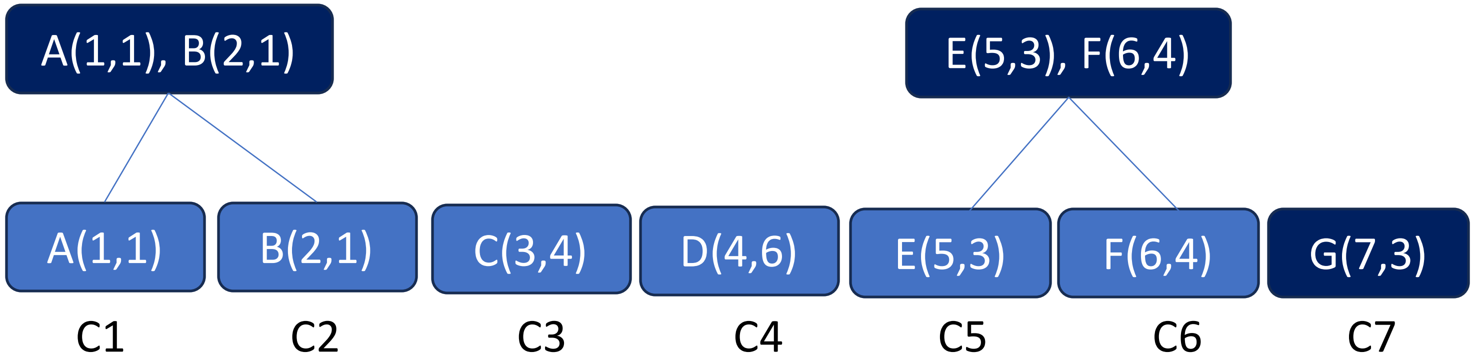
Agrupamento Hierárquico

Exemplo:

	A	B	C	D	E	F	G
A	0	1	5	8	6	8	8
B	1	0	4	7	5	7	7
C	5	4	0	3	3	3	5
D	8	7	3	0	4	4	6
E	6	5	3	4	0	2	2
F	8	7	3	4	2	0	2
G	8	7	5	6	2	2	0

Une dois clusters proximos

G



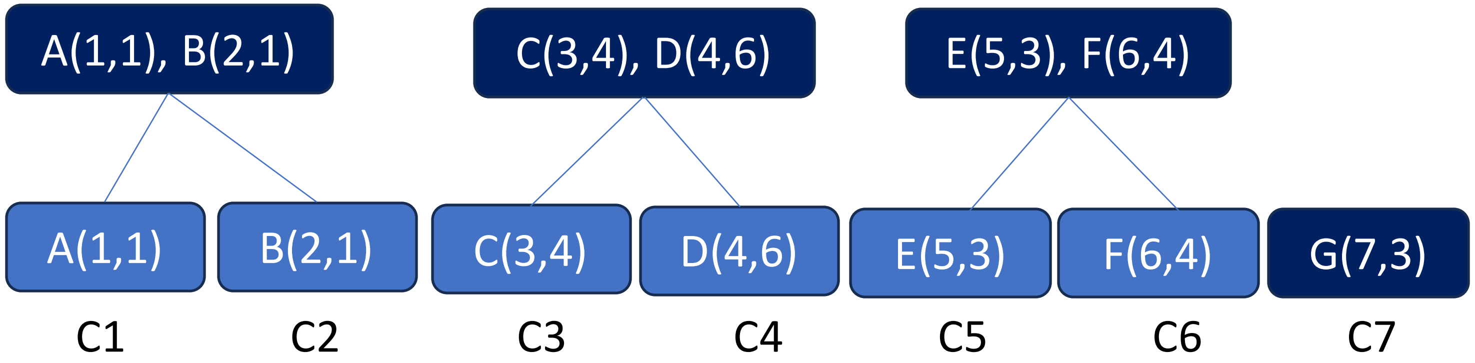
Agrupamento Hierárquico

Exemplo:

	A	B	C	D	E	F	G
A	0	1	5	8	6	8	8
B	1	0	4	7	5	7	7
C	5	4	0	3	3	3	5
D	8	7	3	0	4	4	6
E	6	5	3	4	0	2	2
F	8	7	3	4	2	0	2
G	8	7	5	6	2	2	0

Une dois clusters proximos

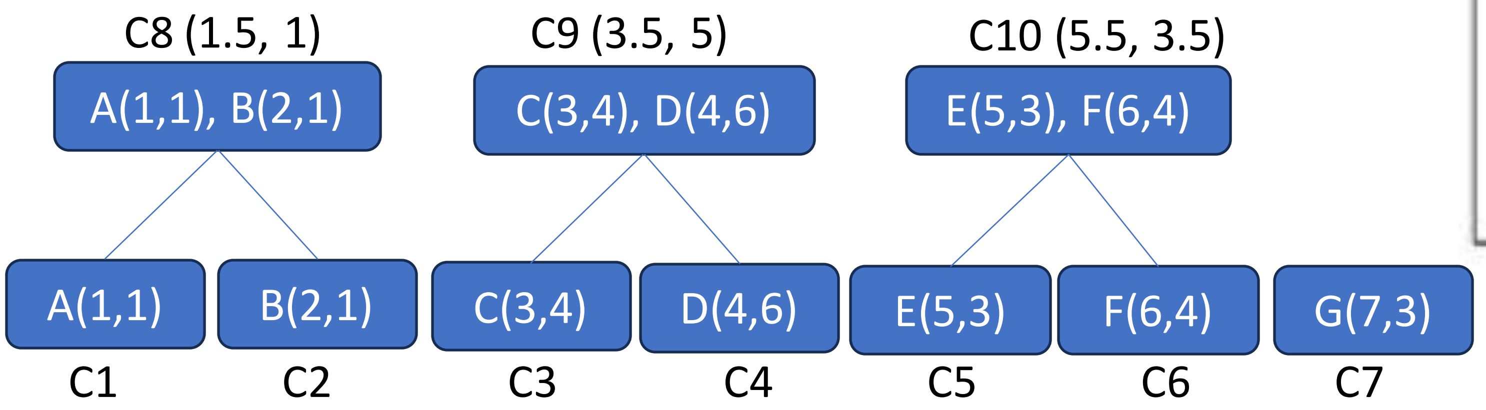
C,D -> distancia =3

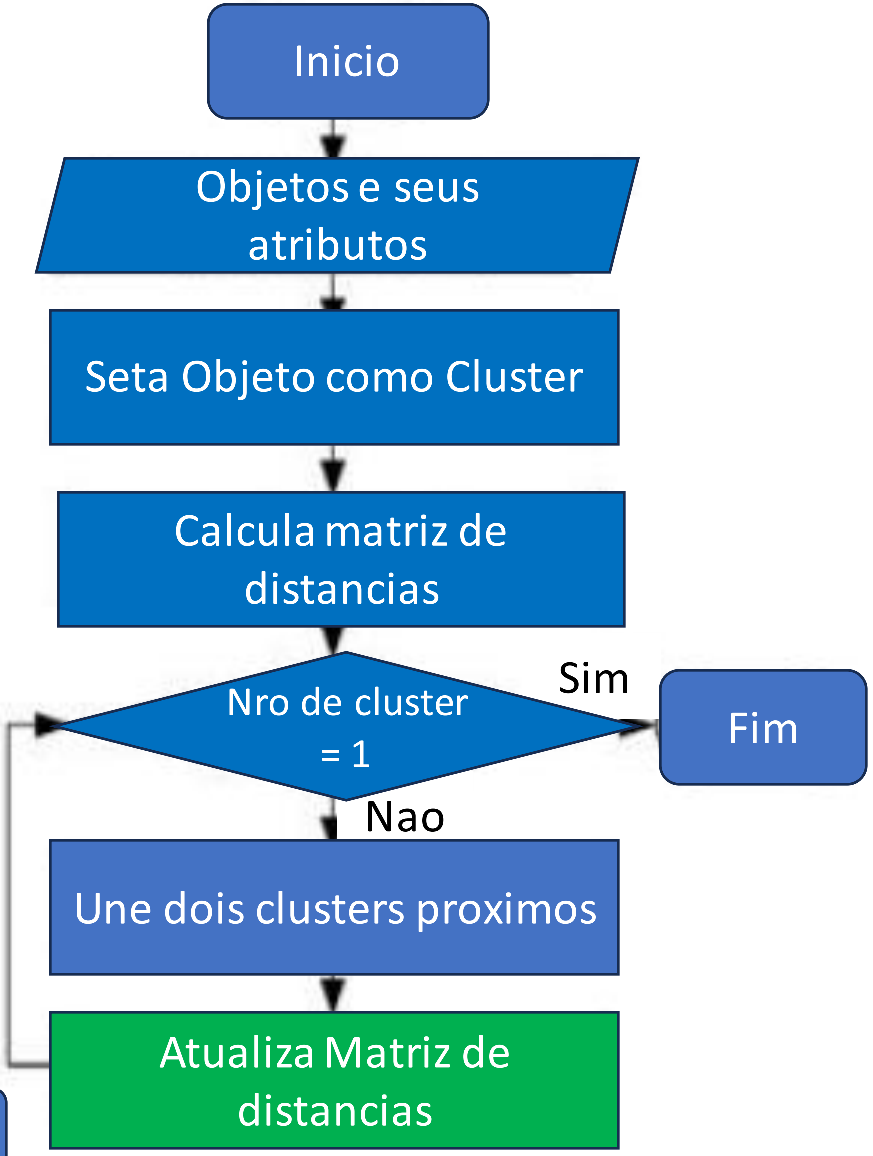


Agrupamento Hierárquico

Exemplo:

C8 = ((1+2)/2, (1+1)/2) = (1.5, 1)
C9 = ((3+4)/2, (4+6)/2) = (3.5, 5)
C10 = ((5+6)/2, (3+4)/2) = (5.5, 3.5)

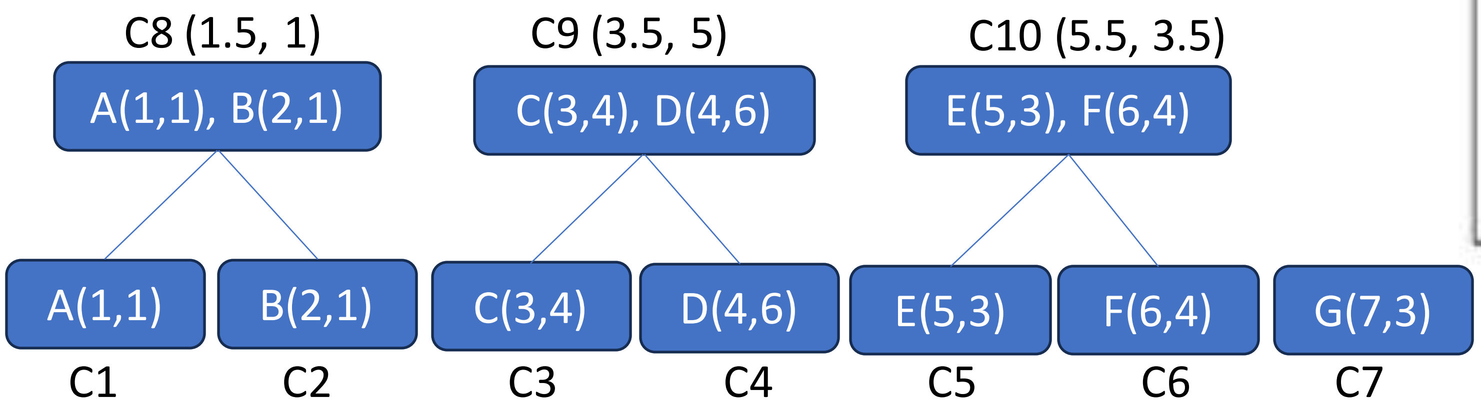


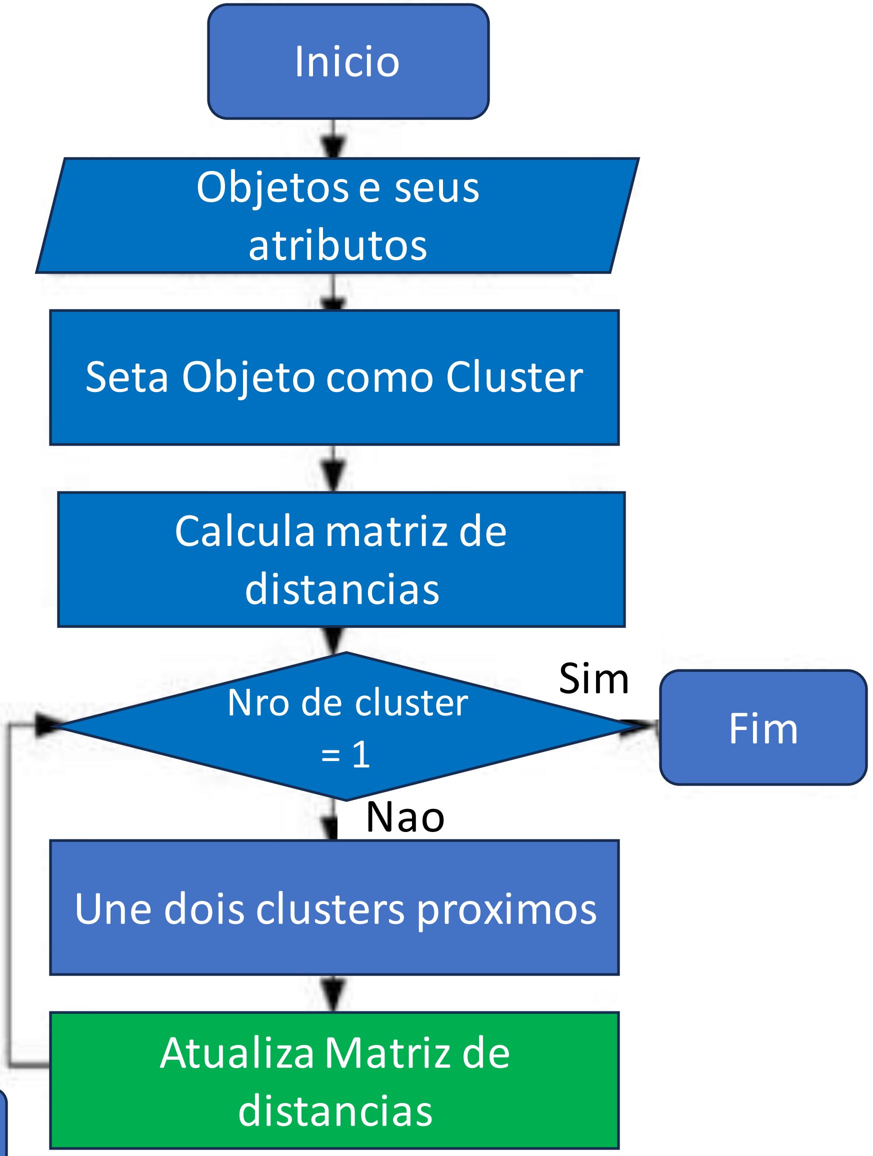


Agrupamento Hierárquico

Exemplo:

	C8 (1.5,1)	C9 (3.5, 5)	C10 (5.5,3.5)	C7 (7,3)
C8 (1.5,1)	0			
C9 (3.5, 5)		0		
C10 (5.5,3.5)			0	
C7 (7,3)				0





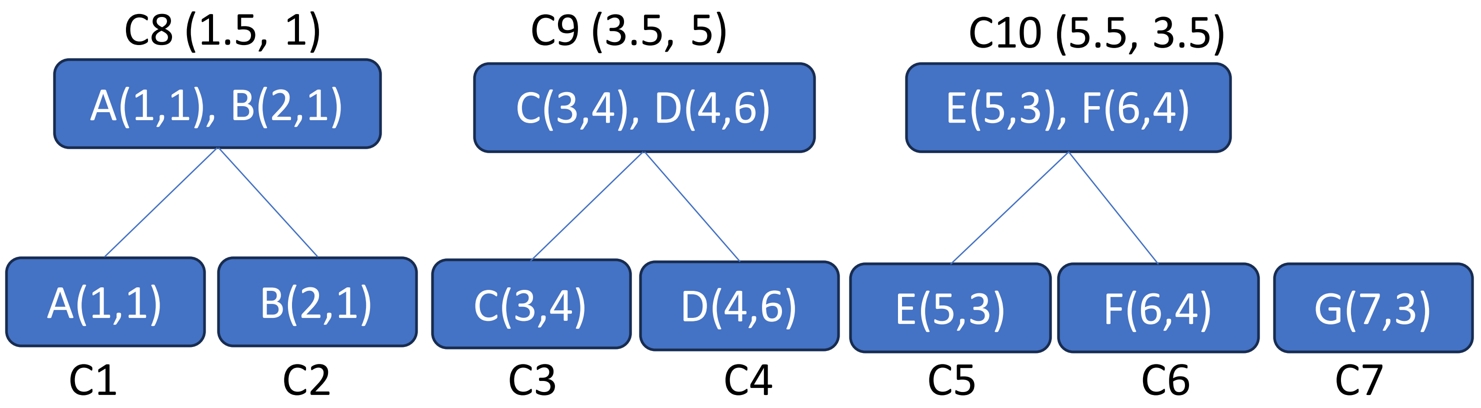
Agrupamento Hierárquico

Exemplo:

dist(C8,C9) = |1.5-3.5| + |1-5| = 6
dist(C8,C10) = |1.5-5.5| + |1-3.5|= 6.5
dist(C8,C7) = |1.5-7| + |1-3 | = 7.5

	C8 (1.5,1)	C9 (3.5, 5)	C10 (5.5,3.5)	C7 (7,3)
C8 (1.5,1)	0	6	6.5	7.5
C9 (3.5, 5)	6	0		
C10 (5.5,3.5)	6.5		0	
C7 (7,3)	7.5			0

Atualiza Matriz de distancias



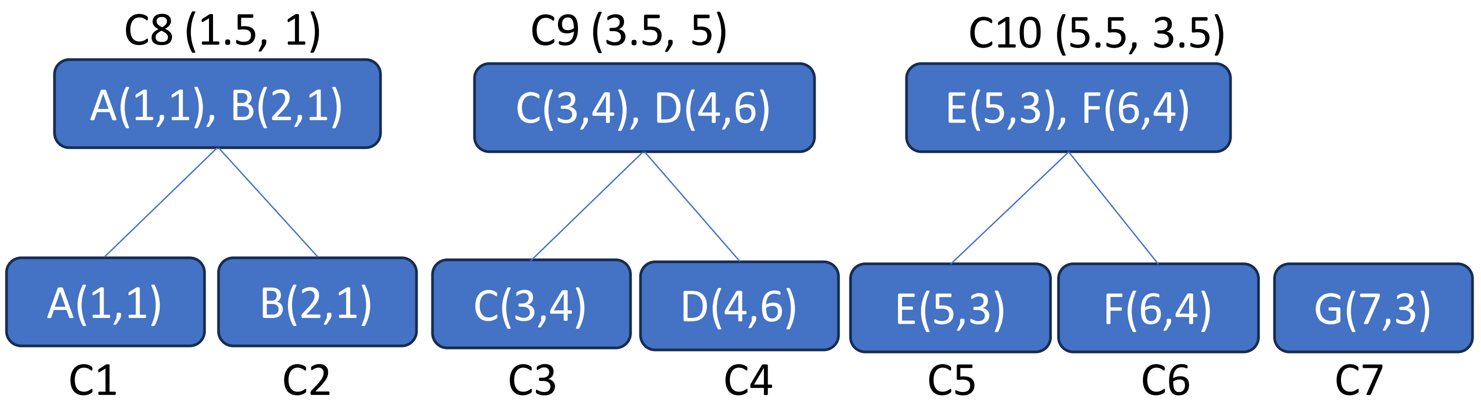
Agrupamento Hierárquico

Exemplo:

dist(C9,C10) = |3.5-5.5| + |5-3.5|= 3.5
dist(C9,C7) = |3.5-7| + |5-3 | = 5.5

	C8 (1.5,1)	C9 (3.5, 5)	C10 (5.5,3.5)	C7 (7,3)
C8 (1.5,1)	0	6	6.5	7.5
C9 (3.5, 5)	6	0	3.5	5.5
C10 (5.5,3.5)	6.5	3.5	0	
C7 (7,3)	7.5	5.5		0

Atualiza Matriz de distancias



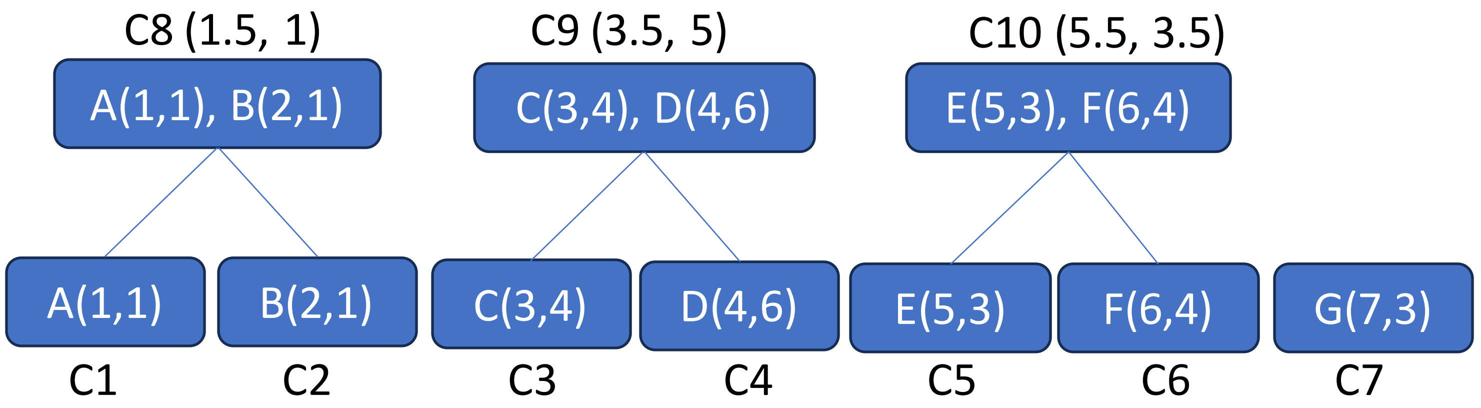
Agrupamento Hierárquico

Exemplo:

dist(C10,C7) = |5.5-7| + |3.5-3 | = 2

	C8 (1.5,1)	C9 (3.5, 5)	C10 (5.5,3.5)	C7 (7,3)
C8 (1.5,1)	0	6	6.5	7.5
C9 (3.5, 5)	6	0	3.5	5.5
C10 (5.5,3.5)	6.5	3.5	0	2
C7 (7,3)	7.5	5.5	2	0

Atualiza Matriz de distancias

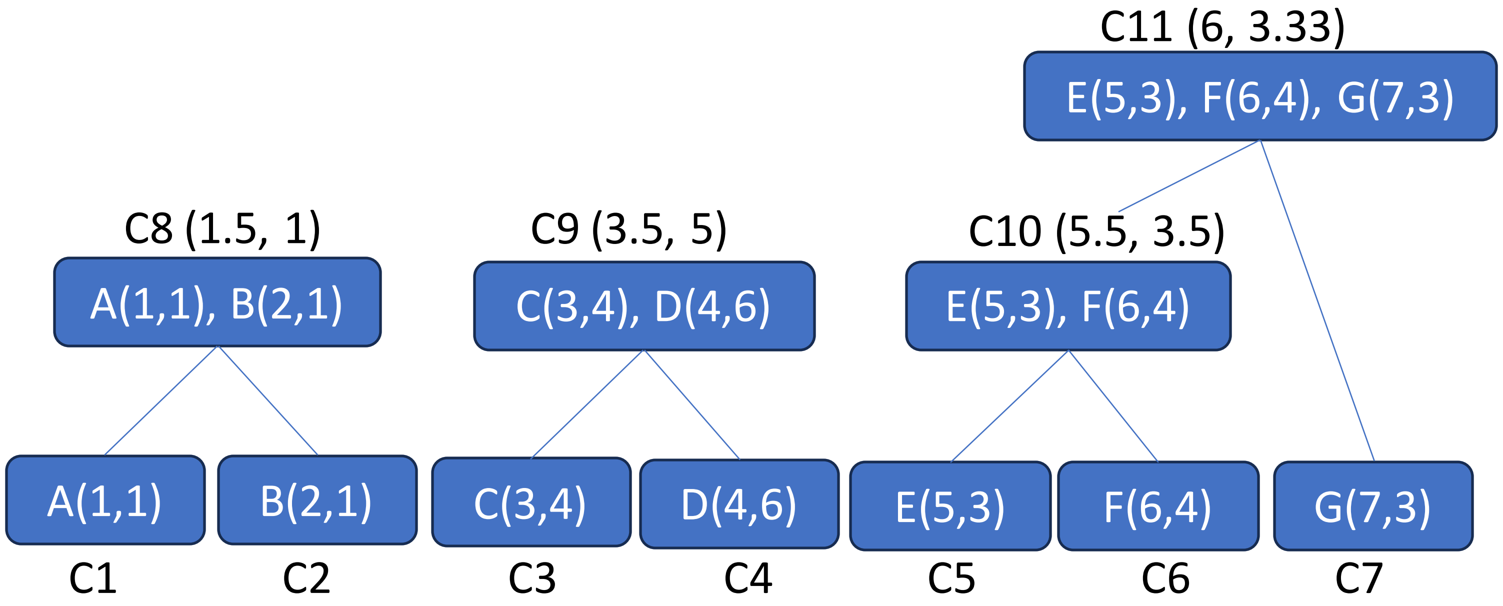


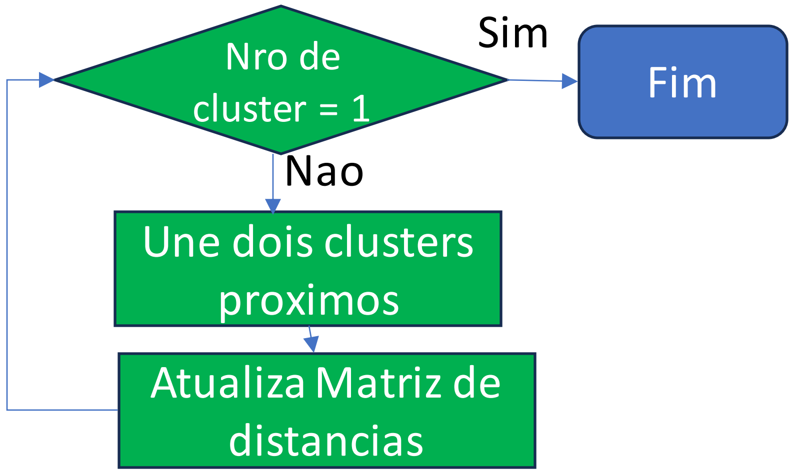
Agrupamento Hierárquico

Exemplo:

C11 = ((5+6+7)/3, (3+4+3)/3) = (6, 3,33)

	C8 (1.5,1)	C9 (3.5, 5)	C10 (5.5,3.5)	C7 (7,3)
C8 (1.5,1)	0	6	6.5	7.5
C9 (3.5, 5)	6	0	3.5	5.5
C10 (5.5,3.5)	6.5	3.5	0	2
C7 (7,3)	7.5	5.5	2	0



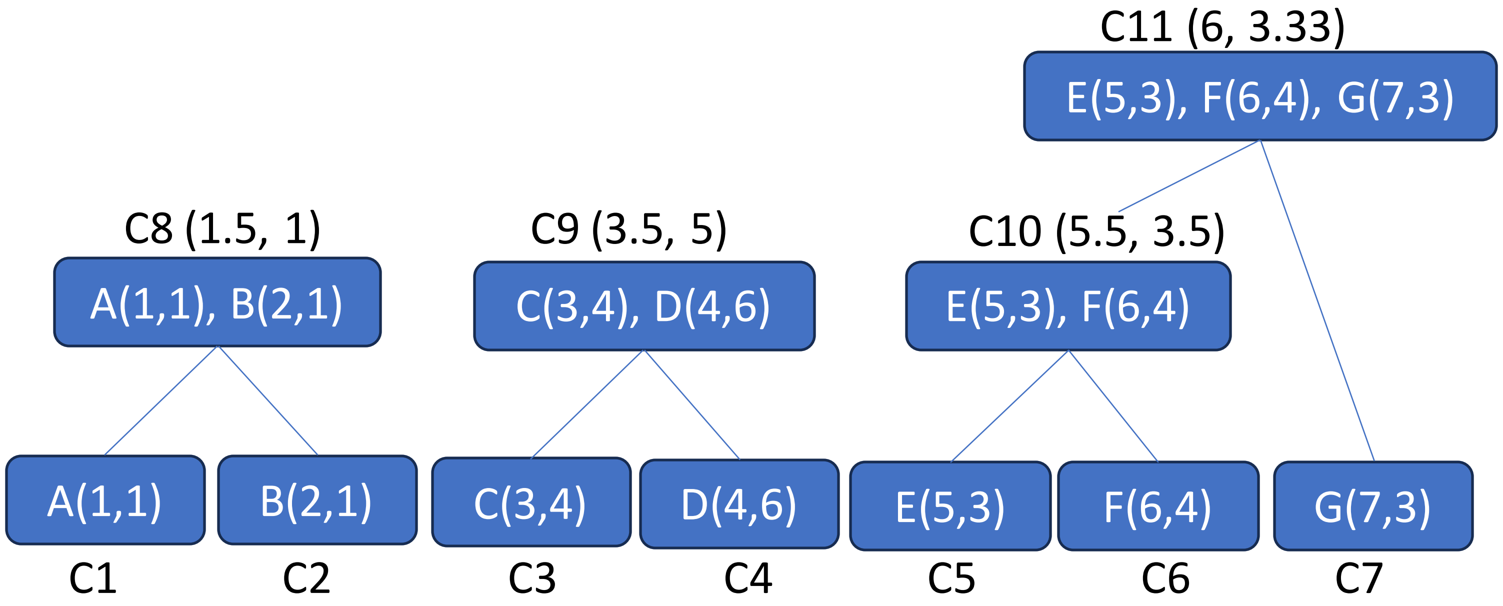


Agrupamento Hierárquico

Exemplo:

	C8 (1.5,1)	C9 (3.5, 5)	C11 (6,3.3)
C8 (1.5,1)	0		
C9 (3.5, 5)		0	
C11 (6,3.3)			0

Atualiza Matriz de distancias

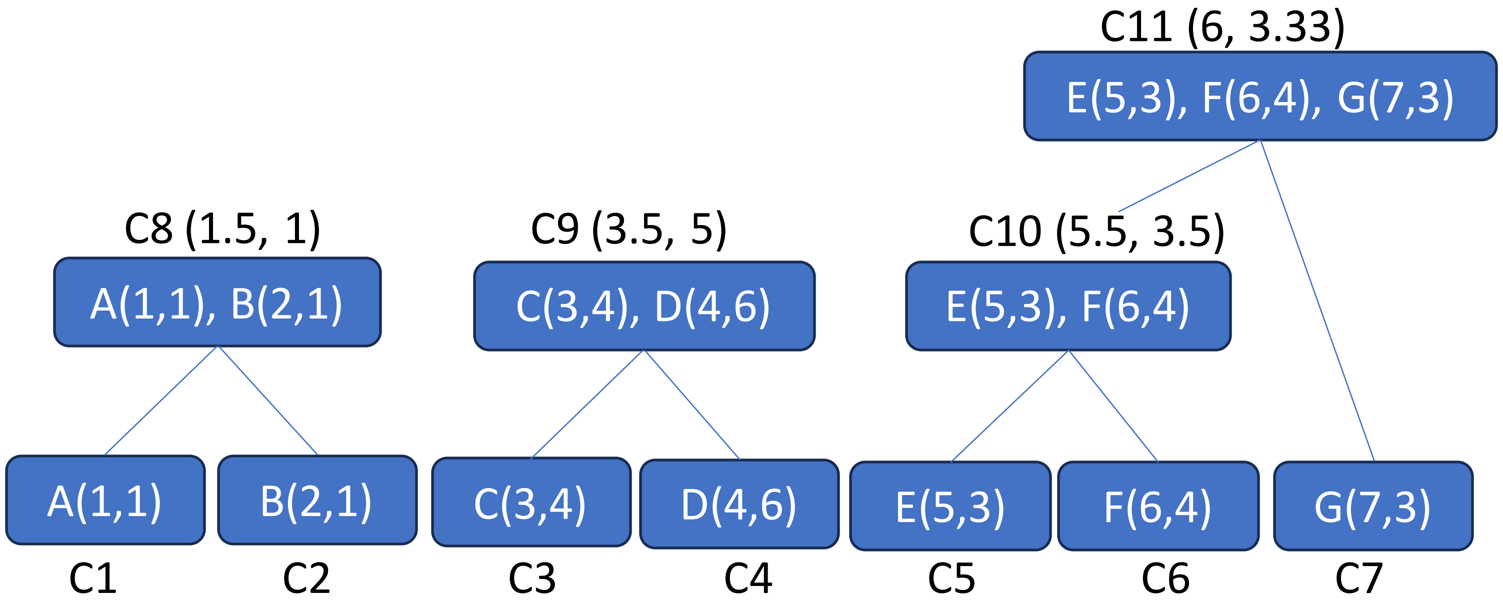


Agrupamento Hierárquico

Exemplo:

dist(C8,C9) = |1.5-3.5| + |1-5| = 6
dist(C8,C11)= |1.5-6| + |1-3.33| = 6.83

	C8 (1.5,1)	C9 (3.5, 5)	C11 (6,3.3)
C8 (1.5,1)	0	6	6.83
C9 (3.5, 5)	6	0	
C11 (6,3.3)	6.83		0



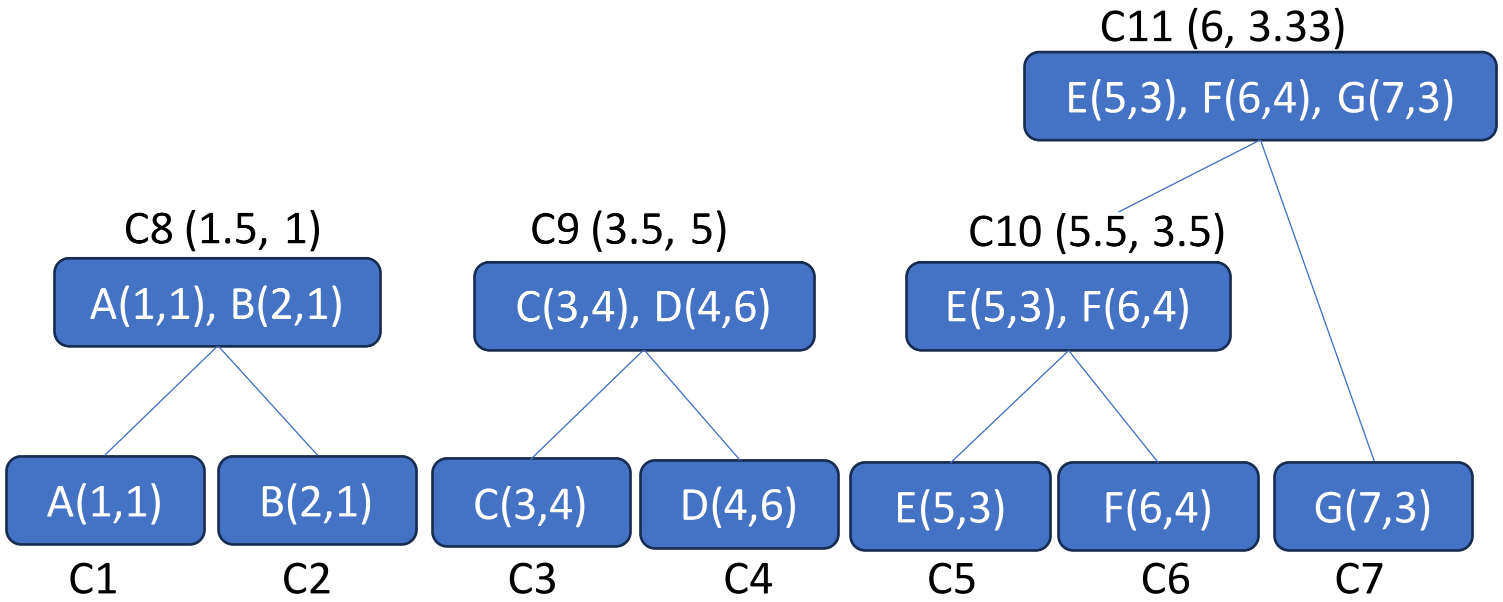
Atualiza Matriz de distancias

Agrupamento Hierárquico

Exemplo:

dist(C9,C11)= |3.5-6| + |5-3.33| = 4.17

	C8 (1.5,1)	C9 (3.5, 5)	C11 (6,3.3)
C8 (1.5,1)	0	6	6.83
C9 (3.5, 5)	6	0	4.17
C11 (6,3.3)	6.83	4.17	0



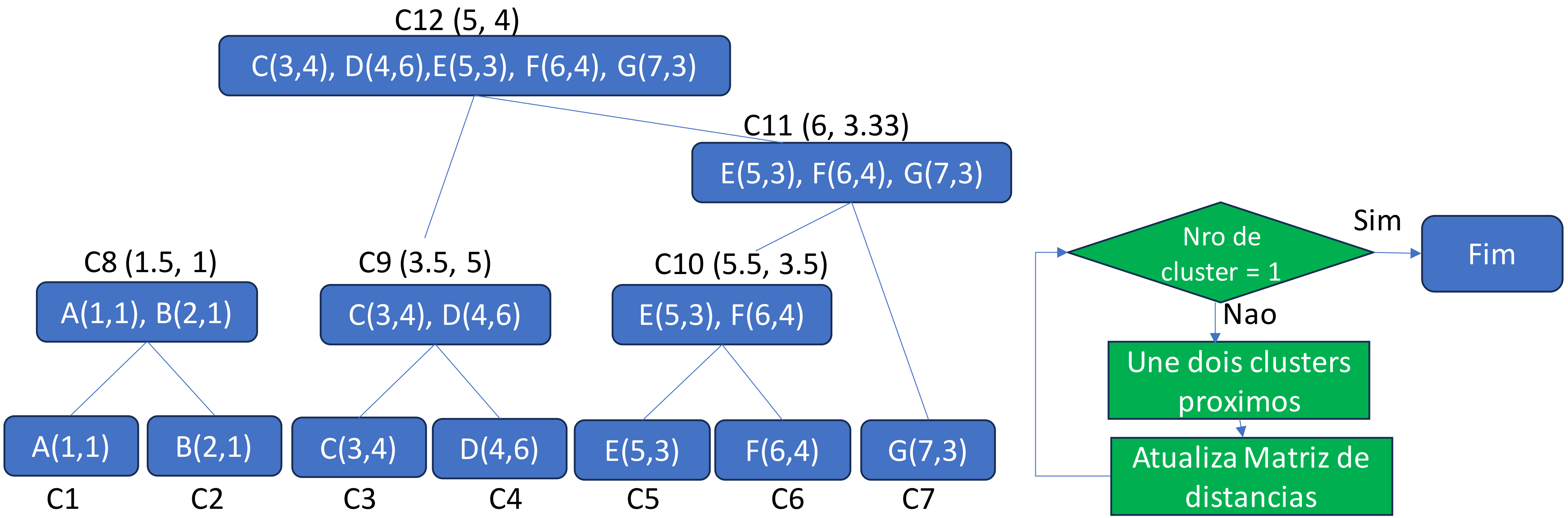
Atualiza Matriz de distancias

Agrupamento Hierárquico

Exemplo:

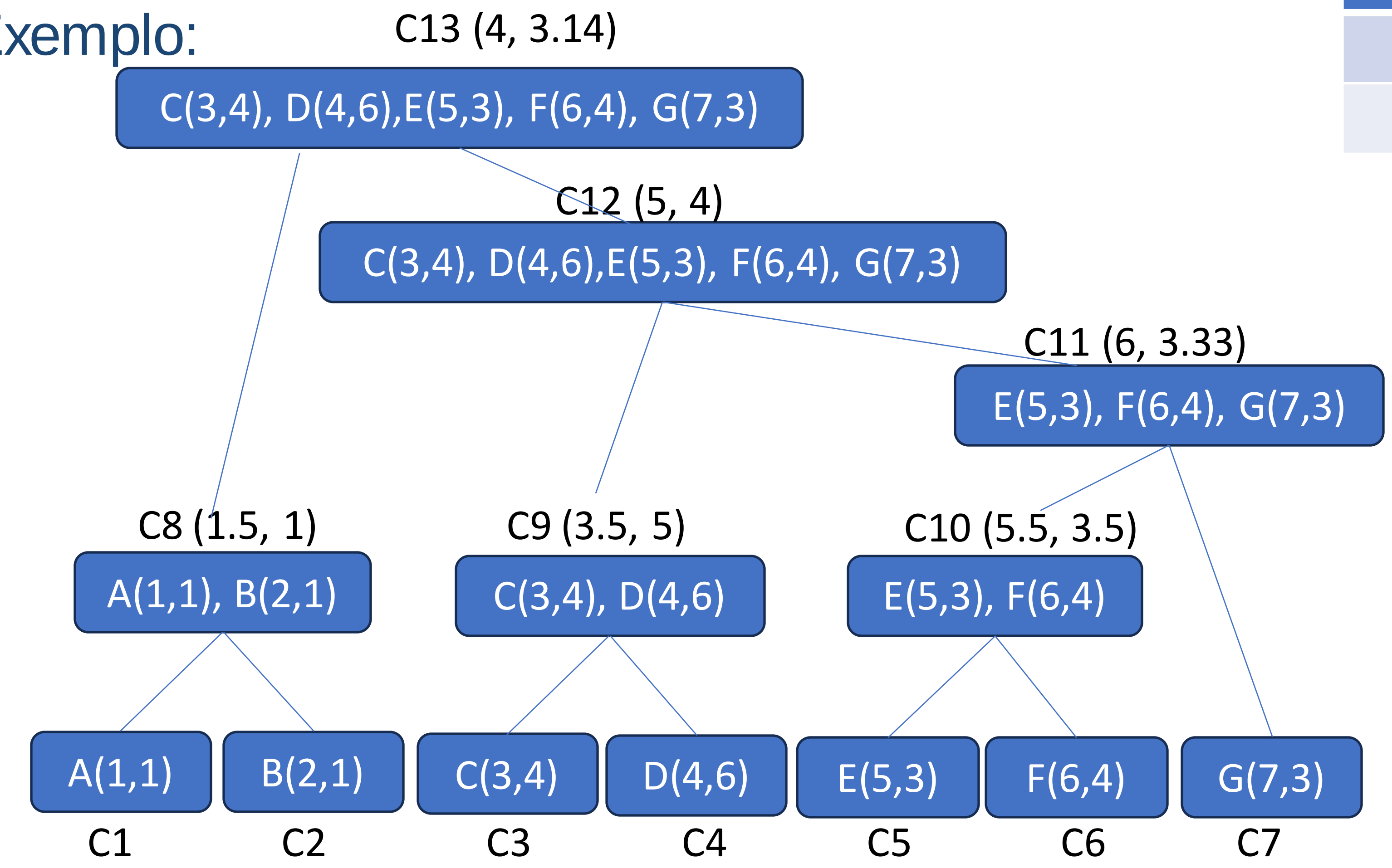
C12 = ((3+4+5+6+7)/5, (4+6+3+4+3)/5) = (5, 4)

	C8 (1.5,1)	C12 (5, 4)
C8 (1.5,1)		
C12 (5, 4)		



Agrupamento Hierárquico

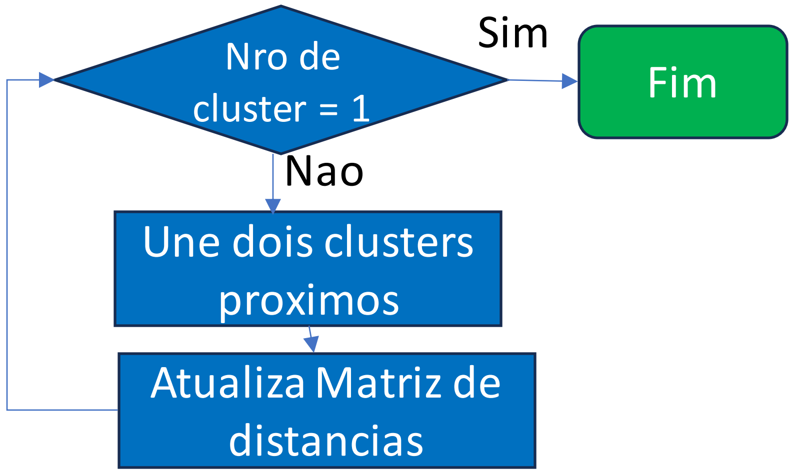
Exemplo:



	C8 (1.5,1)	C12 (5, 4)
C8 (1.5,1)	0	6.5
C12 (5, 4)	6.5	0

dist(C8,C12) = |1.5-5| + |1-4| =6.5

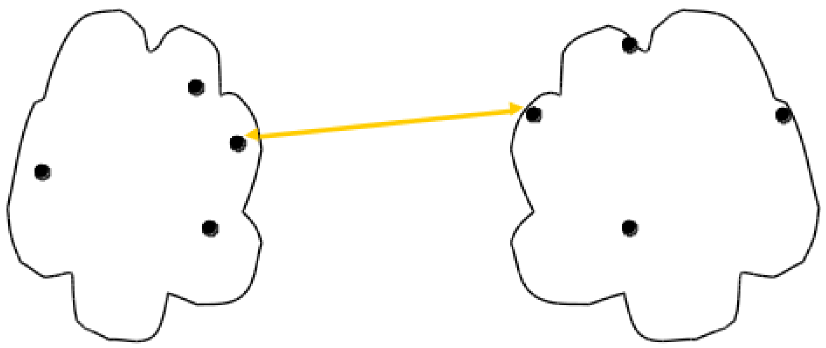
C13 = (28/7, 22/7) = (4, 3.14)



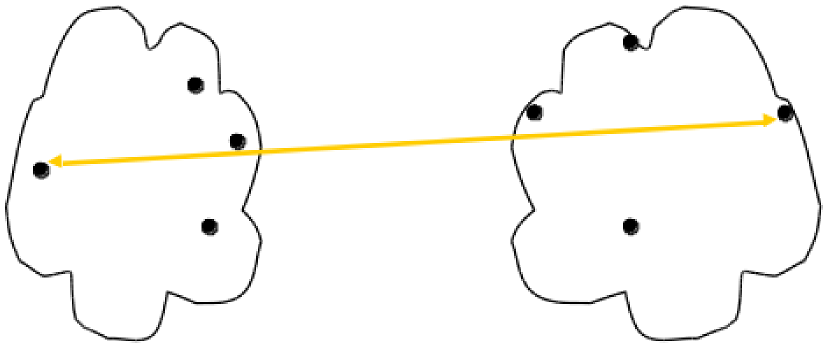
Agrupamento Hierárquico

Alguns algoritmos:

- **Single-link** : baseado na menor distância entre dois clusters C1 e C2 (usa os dados mais próximos, um de cada cluster)



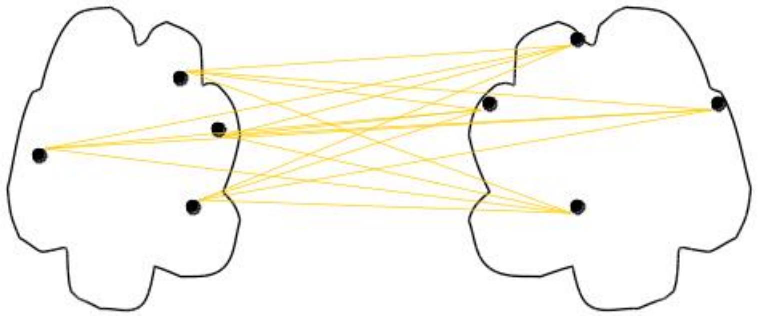
- **Complete-Link:** baseado na maior distância entre dois clusters C1 e C2 (usa os dados mais afastados, um de cada cluster).



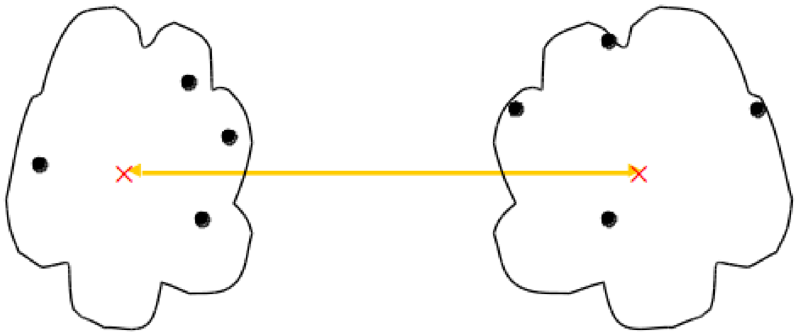
Agrupamento Hierárquico

Alguns algoritmos:

- **Average-link:** baseado na distância média entre dois clusters C1 e C2.



- **Distance Between Centroids**: baseado na distância entre os centróides de dois clusters C1 e C2.



Agrupamento Hierárquico

Em geral, algoritmos hierarquicos nao lidam bem com outliers e ruidos.

Analisando os algoritmos:

- **Single-link** :
  - Indicado para formas não elípticas
  - Sensível a ruídos e outliers
  - Favorece clusters finos e alongados.

- **Complete-Link:**
  - Menos suscetível a ruídos e outliers
  - Tende a quebrar clusters grandes
  - Tem problemas com formas convexas
  - Favorece clusters esféricos.

Agrupamento Hierárquico

- Scki-learn : https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html

`sklearn.cluster`.AgglomerativeClustering

class `sklearn.cluster.AgglomerativeClustering`(*n_clusters=2, *, affinity='deprecated', metric=None, memory=None, connectivity=None, compute_full_tree='auto', linkage='ward', distance_threshold=None, compute_distances=False*) [source]

Agglomerative Clustering.

Recursively merges pair of clusters of sample data; uses linkage distance.

Read more in the User Guide.

Parameters:

n_clusters : *int or None, default=2*
The number of clusters to find. It must be `None` if `distance_threshold` is not `None`.

affinity : *str or callable, default='euclidean'*
The metric to use when calculating distance between instances in a feature array. If metric is a string or callable, it must be one of the options allowed by `sklearn.metrics.pairwise_distances` for its metric parameter. If linkage is "ward", only "euclidean" is accepted. If "precomputed", a distance matrix (instead of a similarity matrix) is needed as input for the fit method.

Deprecated since version 1.2: `affinity` was deprecated in version 1.2 and will be renamed to `metric` in 1.4.

metric : *str or callable, default=None*
Metric used to compute the linkage. Can be "euclidean", "l1", "l2", "manhattan", "cosine", or "precomputed". If set to `None` then "euclidean" is used. If linkage is "ward", only "euclidean" is accepted. If "precomputed", a distance matrix is needed as input for the fit method.

Comentando alguns parâmetros:

- **n_cluster:** valor inteiro (int) ou None (nenhum). Número de clusters a encontrar. O valor padrão é 2.
- **affinity**: não usar mais
- **metric**: str ou callable. Corresponde a métrica para calcular a distância entre os clusters. Pode ser: "euclidian", "manhattan", "cosine" e outros. Quando está como None, a distância "euclidian" é usada como padrão.

Agrupamento Hierárquico

- Scki-learn : https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html

sklearn.cluster.AgglomerativeClustering

class `sklearn.cluster.AgglomerativeClustering`(*n_clusters=2, *, affinity='deprecated', metric=None, memory=None, connectivity=None, compute_full_tree='auto', linkage='ward', distance_threshold=None, compute_distances=False*) [source]

Agglomerative Clustering.

Recursively merges pair of clusters of sample data; uses linkage distance.

Read more in the User Guide.

Parameters:

linkage : *{'ward', 'complete', 'average', 'single'}, default='ward'*

Which linkage criterion to use. The linkage criterion determines which distance to use between sets of observation. The algorithm will merge the pairs of cluster that minimize this criterion.

- 'ward' minimizes the variance of the clusters being merged.
- 'average' uses the average of the distances of each observation of the two sets.
- 'complete' or 'maximum' linkage uses the maximum distances between all observations of the two sets.
- 'single' uses the minimum of the distances between all observations of the two sets.

New in version 0.20: Added the 'single' option

distance_threshold : *float, default=None*

The linkage distance threshold at or above which clusters will not be merged. If not `None`, `n_clusters` must be `None` and `compute_full_tree` must be `True`.

New in version 0.21.

Comentando alguns parâmetros:

- **linkage:** determina o critério para a união dos clusters.
  - **ward** :<u>minimiza a variância</u> dos clusters que estão sendo mesclados. (Padrão)
  - **average**: usa a <u>média</u> das distâncias de cada observação dos dois conjuntos
  - **complete**: utiliza as <u>distâncias máximas</u> entre todas as observações dos dois conjuntos
  - **single**: utiliza as <u>distâncias mínimas</u> entre todas as observações dos dois conjuntos

Agrupamento Hierárquico

- Scki-learn : https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html

`sklearn.cluster`.AgglomerativeClustering

class `sklearn.cluster.`**`AgglomerativeClustering`**(*n_clusters=2, *, affinity='deprecated', metric=None, memory=None, connectivity=None, compute_full_tree='auto', linkage='ward', distance_threshold=None, compute_distances=False*) [source]

Agglomerative Clustering.

Recursively merges pair of clusters of sample data; uses linkage distance.

Read more in the User Guide.

Parameters:

linkage : *{'ward', 'complete', 'average', 'single'}, default='ward'*

Which linkage criterion to use. The linkage criterion determines which distance to use between sets of observation. The algorithm will merge the pairs of cluster that minimize this criterion.

- 'ward' minimizes the variance of the clusters being merged.
- 'average' uses the average of the distances of each observation of the two sets.
- 'complete' or 'maximum' linkage uses the maximum distances between all observations of the two sets.
- 'single' uses the minimum of the distances between all observations of the two sets.

New in version 0.20: Added the 'single' option

distance_threshold : *float, default=None*

The linkage distance threshold at or above which clusters will not be merged. If not `None`, `n_clusters` must be `None` and `compute_full_tree` must be `True`.

New in version 0.21.

Comentando alguns parâmetros:

- **distance_threshold: float ou None** – valor limite definido para a distância entre o clusters. Acima desse valor, os clusters não serão unidos.

Agrupamento Hierárquico

- Scki-learn : https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html

sklearn.cluster.AgglomerativeClustering

class `sklearn.cluster.AgglomerativeClustering`(*n_clusters=2, *, affinity='deprecated', metric=None, memory=None, connectivity=None, compute_full_tree='auto', linkage='ward', distance_threshold=None, compute_distances=False*) [source]

Agglomerative Clustering.

Recursively merges pair of clusters of sample data; uses linkage distance.

Read more in the User Guide.

Attributes:

n_clusters_ : *int*
The number of clusters found by the algorithm. If `distance_threshold=None`, it will be equal to the given `n_clusters`.

labels_ : *ndarray of shape (n_samples)*
Cluster labels for each point.

n_leaves_ : *int*
Number of leaves in the hierarchical tree.

n_connected_components_ : *int*
The estimated number of connected components in the graph.

New in version 0.21: `n_connected_components_` was added to replace `n_components_`.

n_features_in_ : *int*
Number of features seen during fit.

New in version 0.24.

Comentando alguns atributos:

- **n_clusters: int** – número de clusters encontrados pelo algoritmo.
- **labels:** lista correspondente à entrada com os rótulos numéricos atribuídos aos dados.

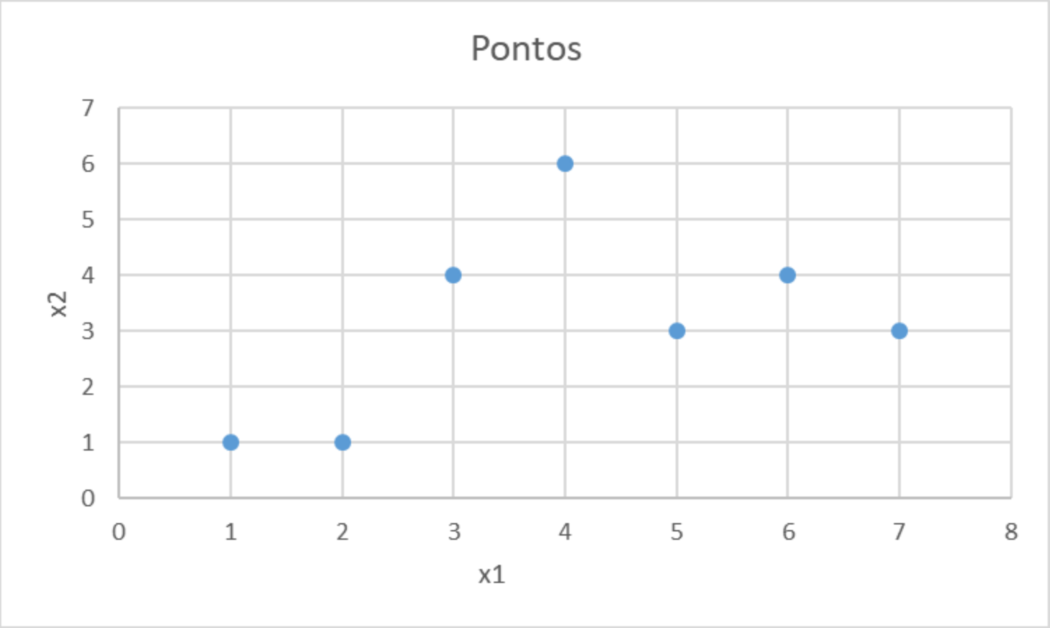
Agrupamento Hierárquico

```
[ ] import pandas as pd
    import numpy as np
    from matplotlib import pyplot as plt
    from sklearn.cluster import AgglomerativeClustering
    import scipy.cluster.hierarchy as sch

    # Usando pontos como exemplo
    x1 = [1,2,3,4,5,6,7]
    x2 = [1,1,4,6,3,4,3]
```

```
X = []
for i in range(0,len(x1)):
  X.append([x1[i],x2[i]])
print("Lista de pontos:\n", X , "\n")
```

```
Lista de pontos:
 [[1, 1], [2, 1], [3, 4], [4, 6], [5, 3], [6, 4], [7, 3]]
```

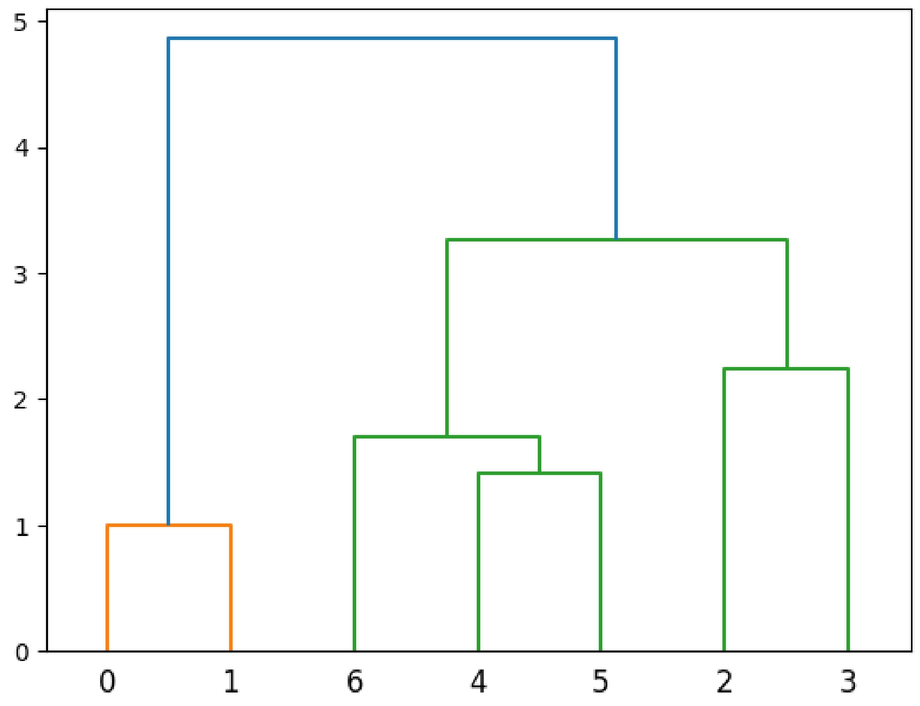


Agrupamento Hierárquico

```python
dendrogram = sch.dendrogram(sch.linkage(X, method='average'))
model = AgglomerativeClustering(n_clusters=5, metric='euclidean', linkage='average')

model.fit(X)
labels = model.labels_
numCluster = model.n_clusters
print("Cluster dos dados: ", labels)
print("Numero de clusters: ", numCluster)
```

```
Cluster dos dados:  [1 1 3 2 0 0 4]
Numero de clusters:  5
```



Agrupamento Hierárquico

```
[ ]  for iCluster in range(0,numCluster):
       print("Cluster: ", iCluster)
       for indice in range(0, len(labels)):
         if labels[indice]==iCluster: print(X[indice])

     plt.show()
```

```
Cluster:  0
[5, 3]
[6, 4]
Cluster:  1
[1, 1]
[2, 1]
Cluster:  2
[4, 6]
Cluster:  3
[3, 4]
Cluster:  4
[7, 3]
```

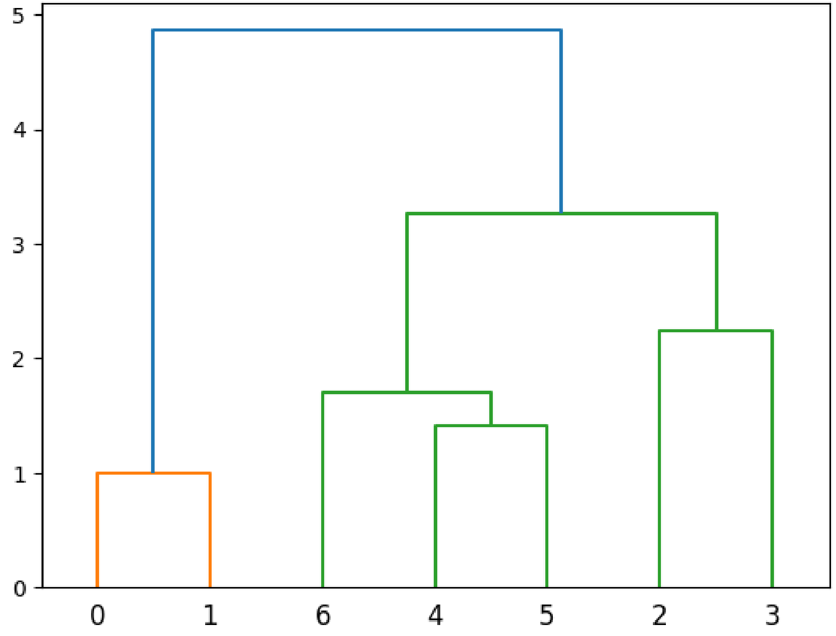
Agrupamento Hierárquico

```
dendrogram = sch.dendrogram(sch.linkage(X, method='average'))
model = AgglomerativeClustering(metric='euclidean', linkage='average')

model.fit(X)
labels = model.labels_
numCluster = model.n_clusters
print("Cluster dos dados: ", labels)
print("Numero de clusters: ", numCluster)
```

Omitindo a quantidade desejada de clusters

```
Cluster dos dados:  [1 1 0 0 0 0 0]
Numero de clusters:  2
```




```
[4] for iCluster in range(0,numCluster):
      print("Cluster: ", iCluster)
      for indice in range(0, len(labels)):
        if labels[indice]==iCluster: print(X[indice])

    plt.show()
```

```
Cluster:  0
[3, 4]
[4, 6]
[5, 3]
[6, 4]
[7, 3]
Cluster:  1
[1, 1]
[2, 1]
```

Dinâmica

- Testando e exemplificando o uso do agrupamento hierárquico aglomerativo.