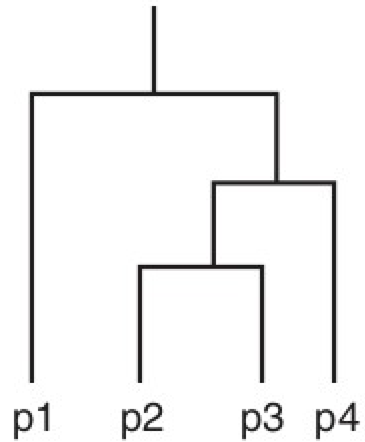
Minería de Datos

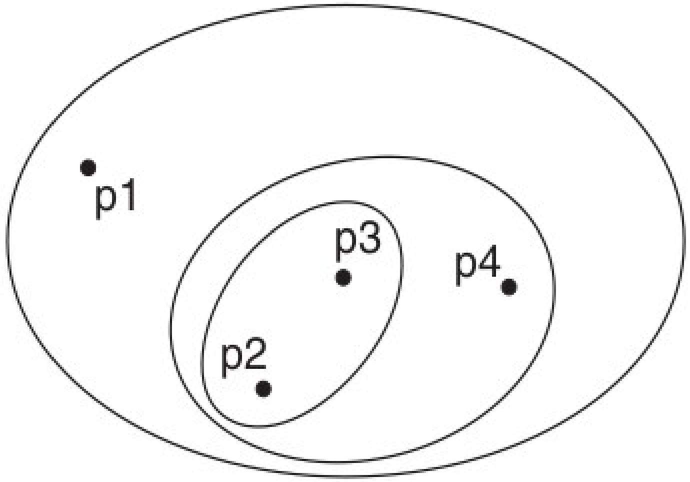
Clustering

Algoritmo	Parámetro	Escalabilidad	Caso de uso	Geometría
K-Means	Número de clusters	Escalable Mejora con modificación MiniBatch	Propósito general flat clustering K no muy grande	Distancia entre objetos
Affinity Propagation	Coeficiente de damping	No escalable	Non-flat clustering K grande	Grafo de distancias
Mean-shift	Ancho de banda	No escalable	Non-flat clustering K grande	Distancia entre objetos
Spectral clustering	Número de clusters	Escalabilidad media	Non-flat clustering K no muy grande	Grafo de distancias
Ward	Número de clusters	Escalable	K grande	Distancias entre objetos
Clustering aglomerativo	Número de clusters	Escalable	Distancias no Euclideanas K grande	Distancias entre objetos
DBSCAN	Tamaño del vecindario	Escalable	Clusters de tamaños distintos	Grafo de vecinos más cercanos
Mezcla de Gaussianas	Muchos	No escalable	Flat clustering Estimación de densidad	Distancias Mahalanobis a centroides

Clustering Jerárquico

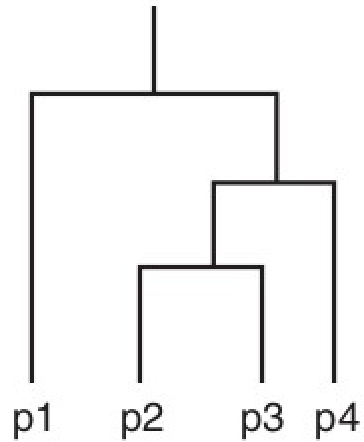
Idea:

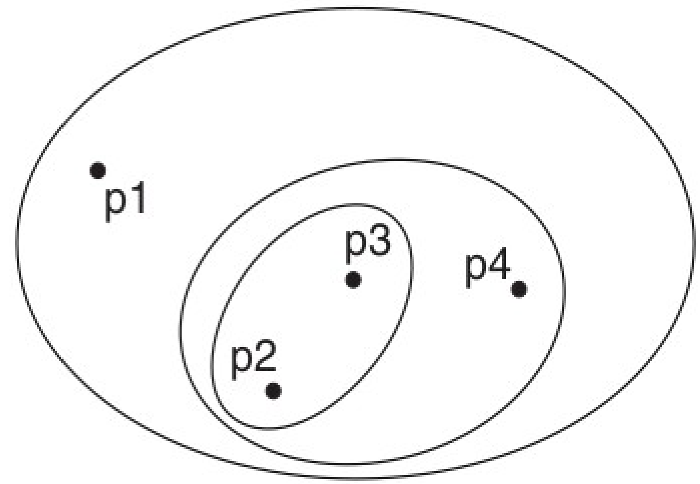




Clustering Jerárquico

Idea:



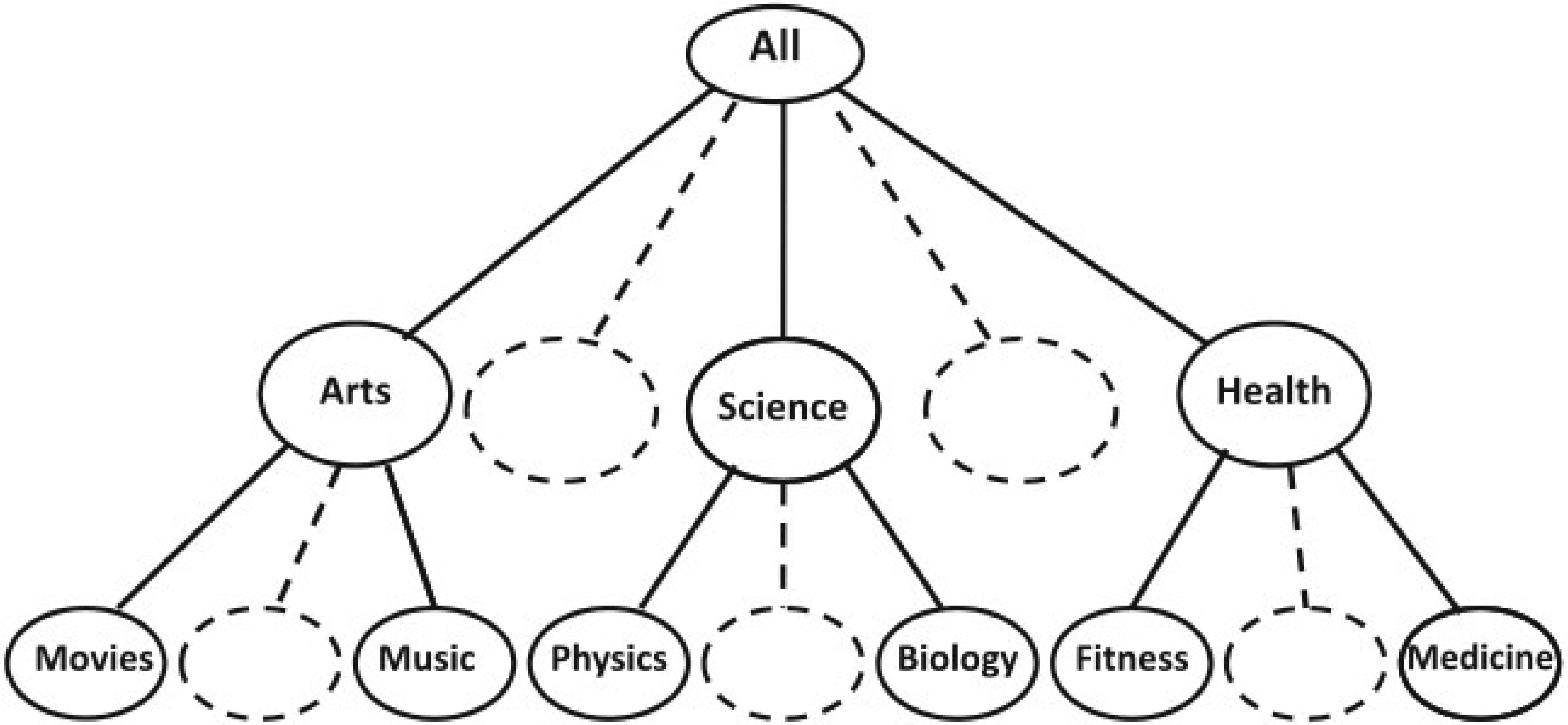


Algorithm Basic agglomerative hierarchical clustering algorithm.

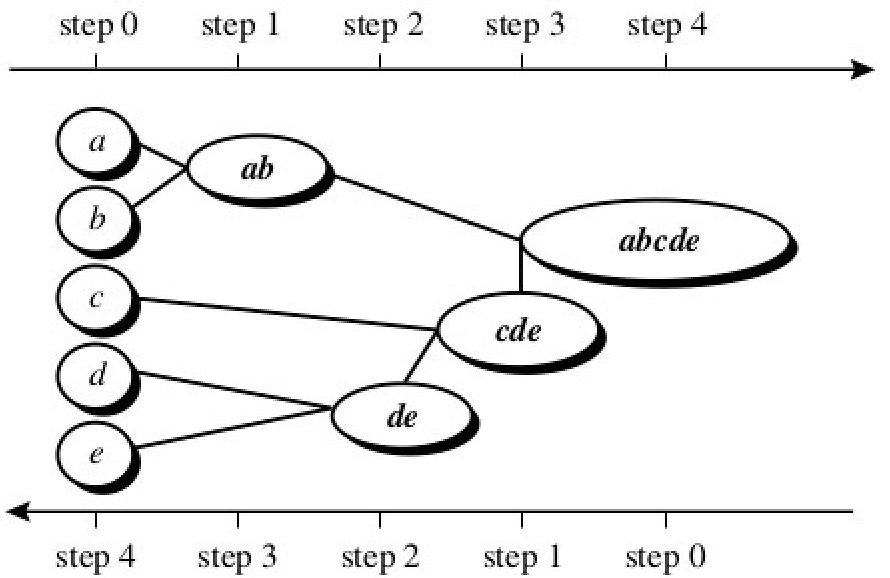
Compute the proximity matrix, if necessary.
repeat
    Merge the closest two clusters.
    Update the proximity matrix to reflect the proximity between the new cluster and the original clusters.
until Only one cluster remains.

Clustering Jerárquico

Estructura jerárquica de abstracción:

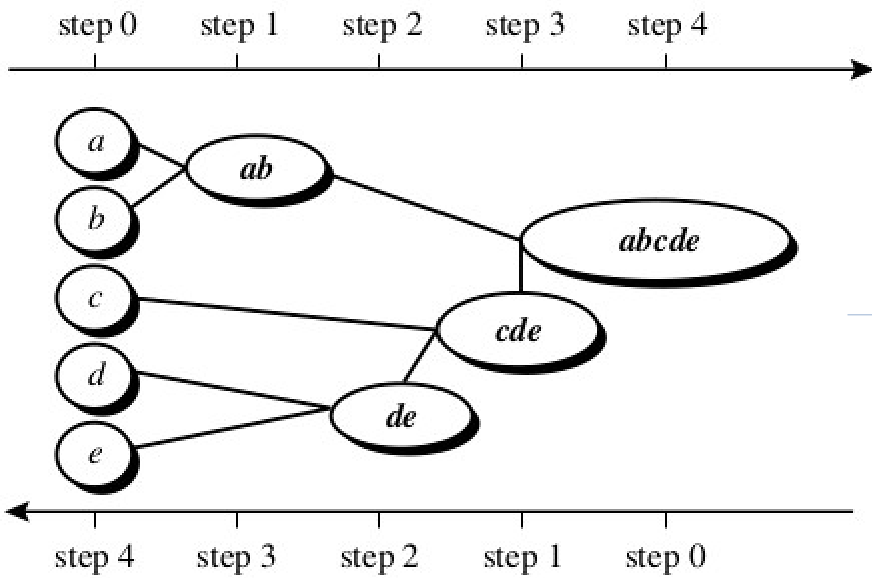


Clustering Jerárquico

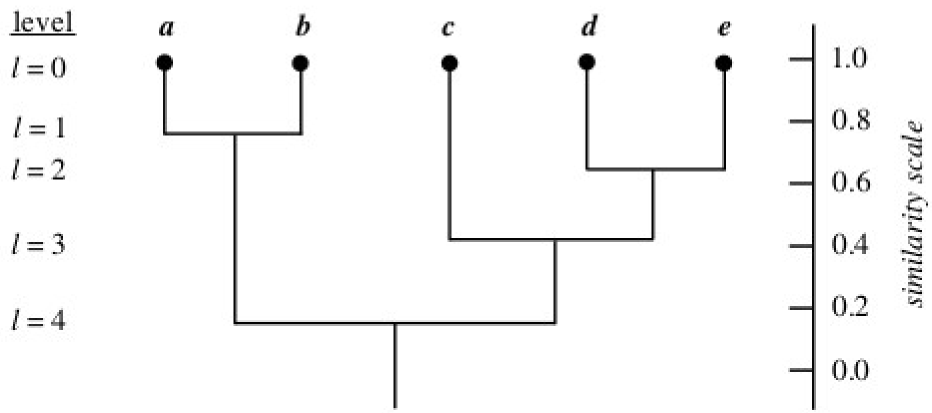


Clustering Jerárquico

aglomerativo

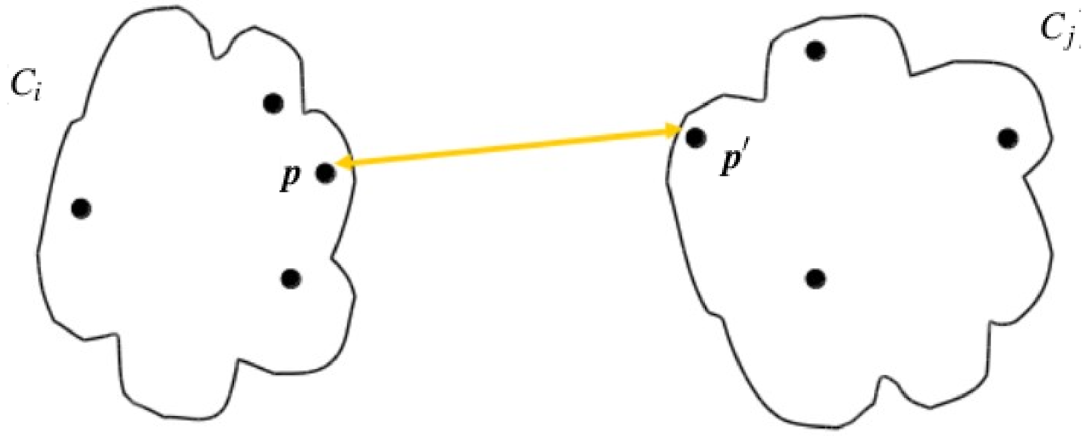


divisivo



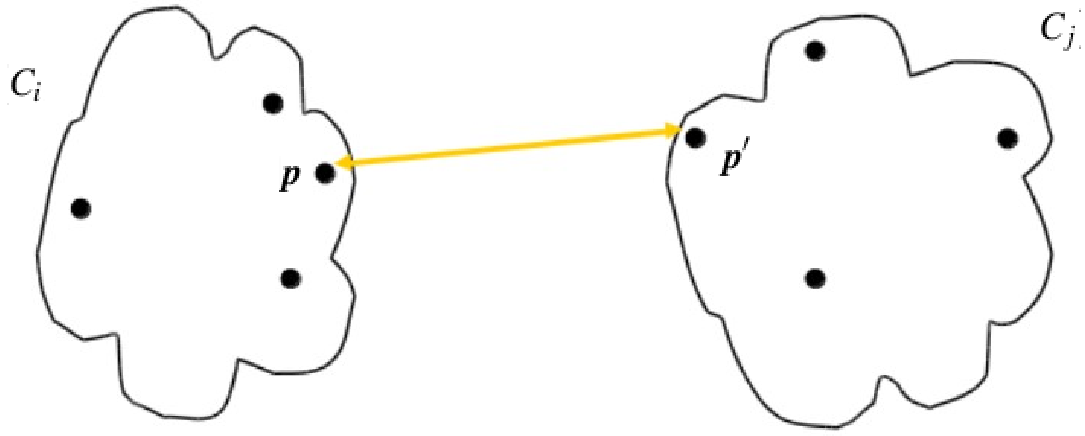
Clustering Jerárquico Aglomerativo

Single Link:



Clustering Jerárquico Aglomerativo

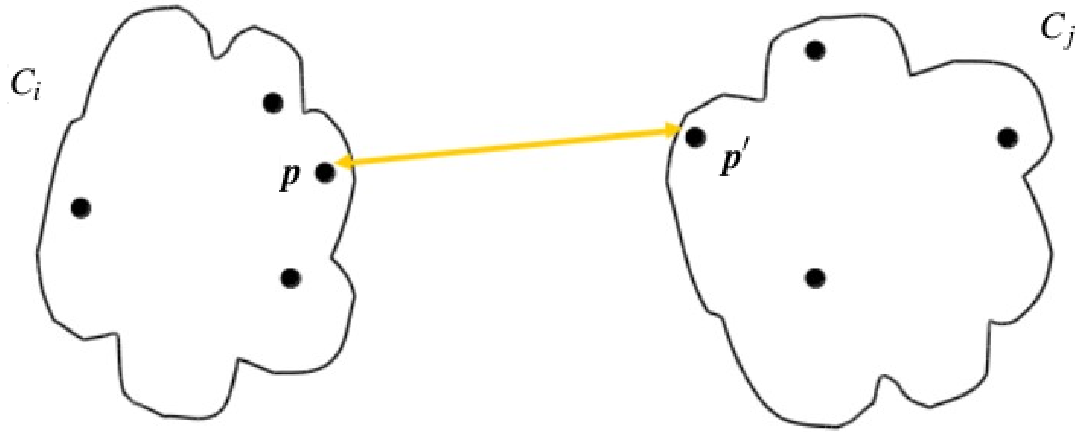
Single Link:



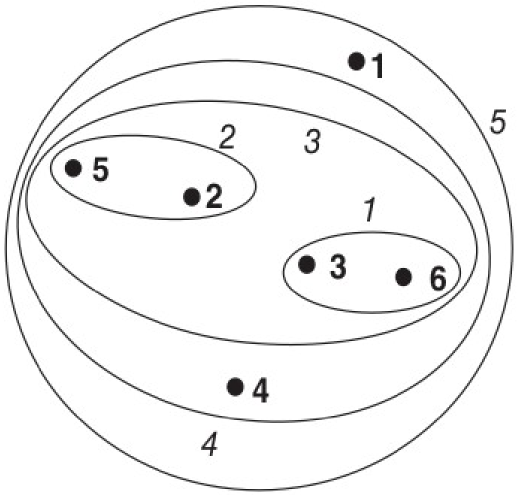
$$d_{min}(C_i, C_j) = \quad min_{\boldsymbol{p} \in C_i,\, \boldsymbol{p}' \in C_j} |\boldsymbol{p} - \boldsymbol{p}'|$$

Clustering Jerárquico Aglomerativo

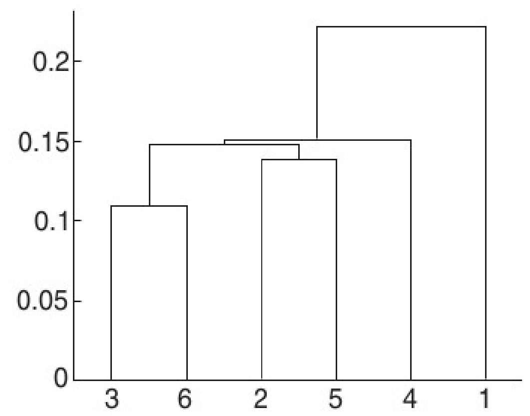
Single Link:



$$d_{min}(C_i, C_j) = \quad min_{p \in C_i,\, p' \in C_j} |p - p'|$$



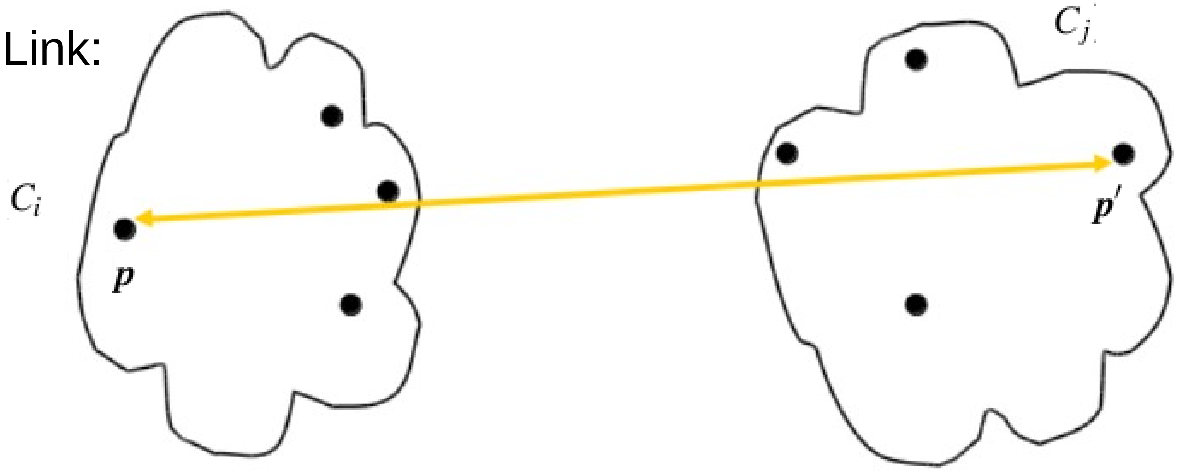
(a) Single link clustering.



(b) Single link dendrogram.

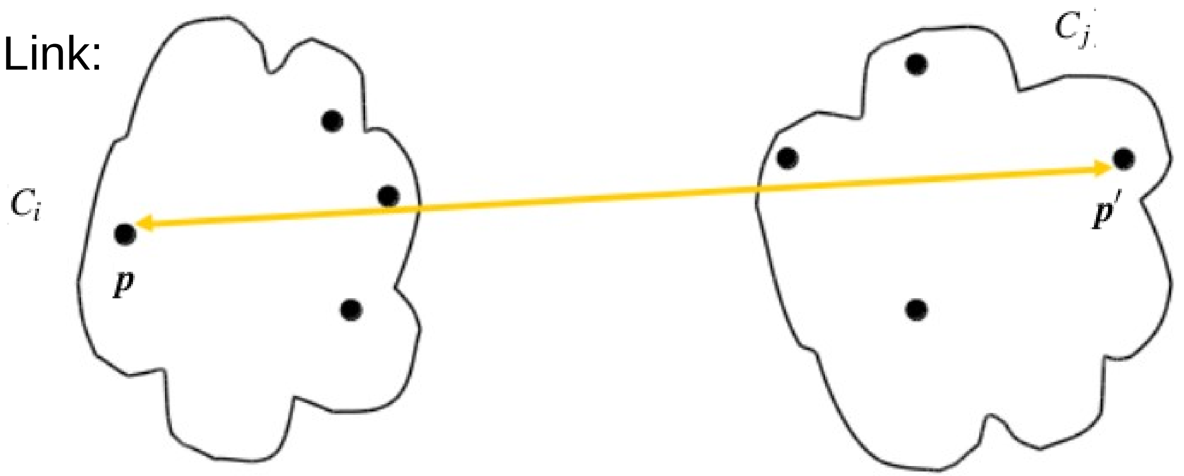
Clustering Jerárquico Aglomerativo

Complete Link:



Clustering Jerárquico Aglomerativo

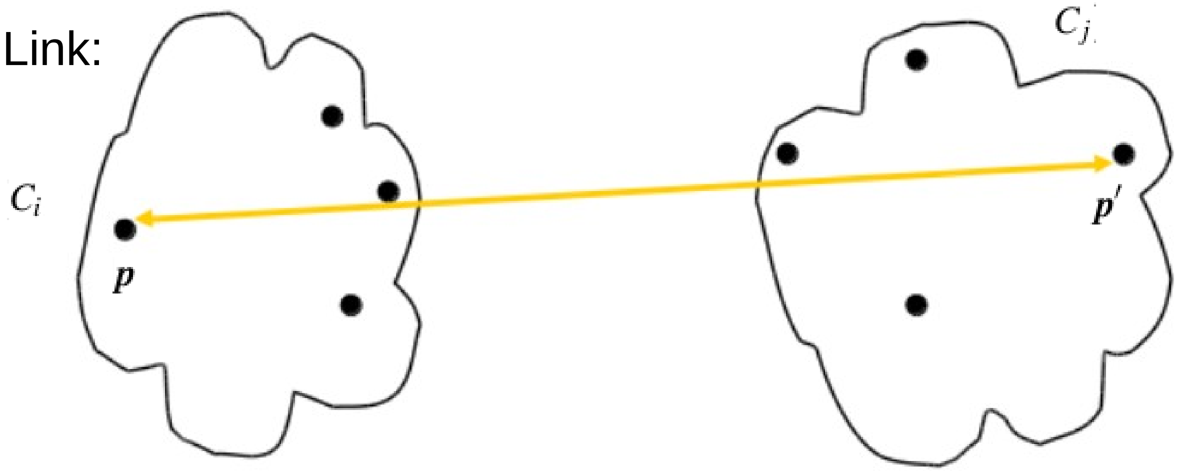
Complete Link:



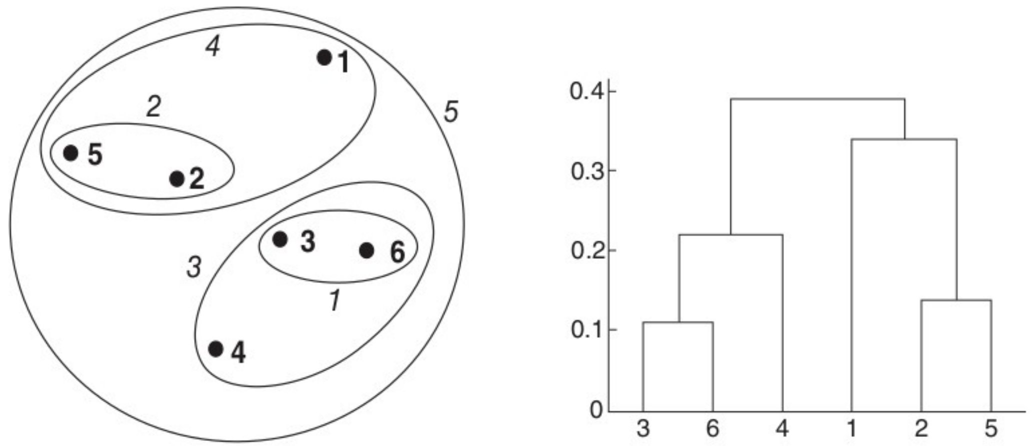
$$d_{max}(C_i, C_j) = \quad max_{p \in C_i, p' \in C_j} |p - p'|$$

Clustering Jerárquico Aglomerativo

Complete Link:



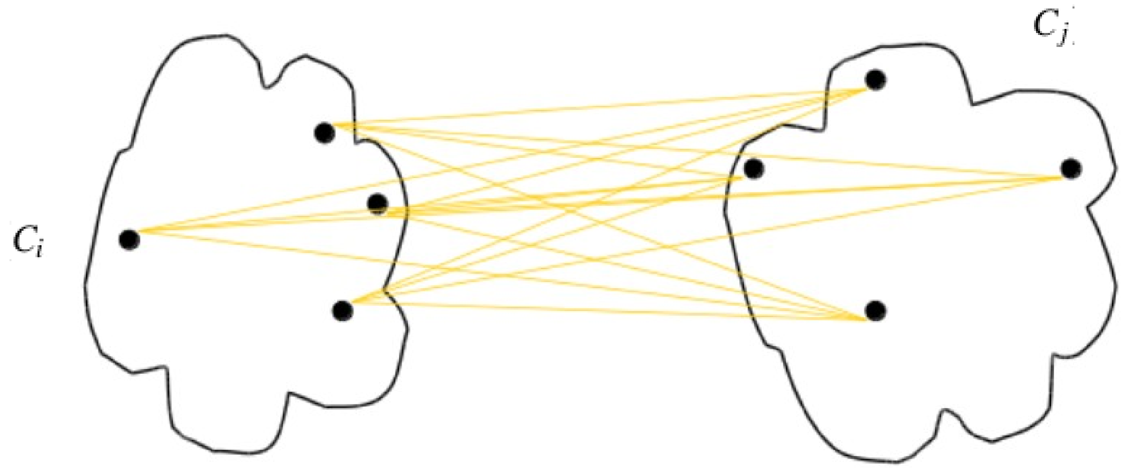
$$d_{max}(C_i, C_j) = \quad max_{p \in C_i, p' \in C_j} |p - p'|$$



(a) Complete link clustering. (b) Complete link dendrogram.

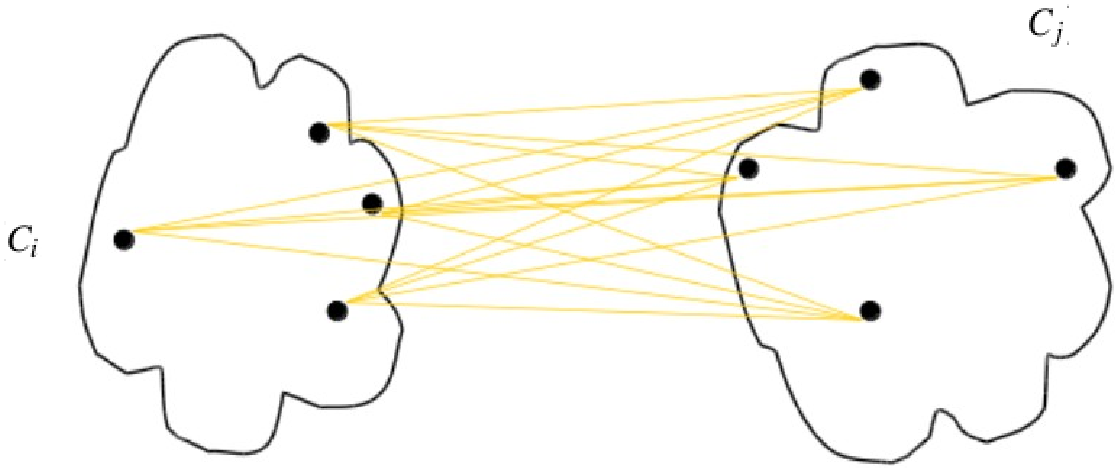
Clustering Jerárquico Aglomerativo

Average Link:



Clustering Jerárquico Aglomerativo

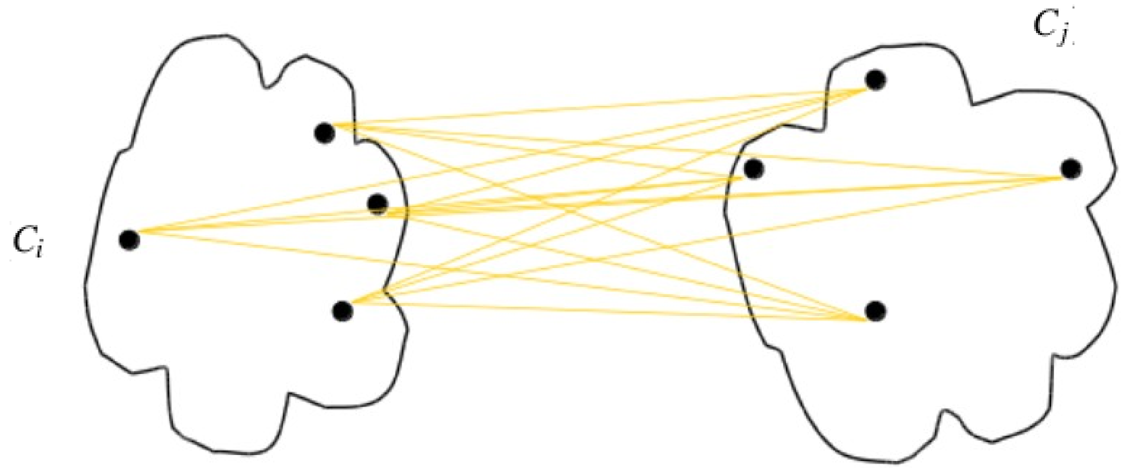
Average Link:



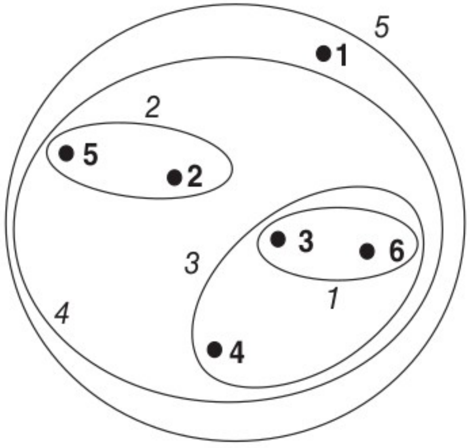
$$d_{avg}(C_i, C_j) = \frac{1}{n_i n_j} \sum_{\boldsymbol{p} \in C_i} \sum_{\boldsymbol{p}' \in C_j} |\boldsymbol{p} - \boldsymbol{p}'|$$

Clustering Jerárquico Aglomerativo

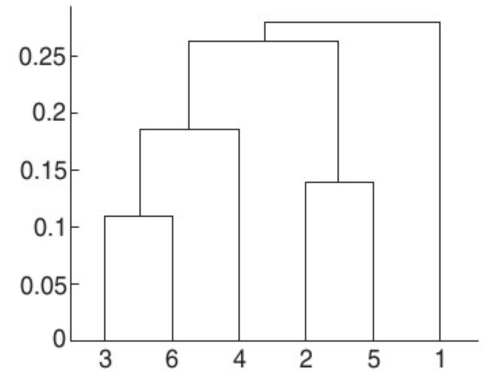
Average Link:



$$d_{avg}(C_i, C_j) = \frac{1}{n_i n_j} \sum_{p \in C_i} \sum_{p' \in C_j} |p - p'|$$



(a) Group average clustering.



(b) Group average dendrogram.

Clustering Jerárquico Aglomerativo

Método de Ward:

$$d_{ij} = d(\{X_i\}, \{X_j\}) = \|X_i - X_j\|^2.$$

Clustering Jerárquico Aglomerativo

Minimiza la varianza intra-cluster

Método de Ward:

$$d_{ij} = d(\{X_i\}, \{X_j\}) = \|X_i - X_j\|^2.$$

Clustering Jerárquico Aglomerativo

Minimiza la varianza intra-cluster

Método de Ward:

$$d_{ij} = d(\{X_i\}, \{X_j\}) = \|X_i - X_j\|^2.$$

$$\Delta(A,B) = \sum_{i \in A \bigcup B} ||\overrightarrow{x_i} - \overrightarrow{m}_{A \bigcup B}||^2 - \sum_{i \in A} ||\overrightarrow{x_i} - \overrightarrow{m}_A||^2 - \sum_{i \in B} ||\overrightarrow{x_i} - \overrightarrow{m}_B||^2 = \frac{n_A n_B}{n_A + n_B} ||\overrightarrow{m}_A - \overrightarrow{m}_B||^2$$

Clustering Jerárquico Aglomerativo

Minimiza la varianza intra-cluster

Método de Ward:

$$d_{ij} = d(\{X_i\}, \{X_j\}) = \|X_i - X_j\|^2.$$

#datos de cada cluster

$$\Delta(A,B) = \sum_{i \in A \bigcup B} ||\overrightarrow{x_i} - \overrightarrow{m}_{A \bigcup B}||^2 - \sum_{i \in A} ||\overrightarrow{x_i} - \overrightarrow{m}_A||^2 - \sum_{i \in B} ||\overrightarrow{x_i} - \overrightarrow{m}_B||^2 = \frac{n_A n_B}{n_A + n_B} ||\overrightarrow{m}_A - \overrightarrow{m}_B||^2$$

Centroide del nuevo cluster

Tarea: demostrar

Clustering Jerárquico Aglomerativo

Minimiza la varianza intra-cluster

Método de Ward:

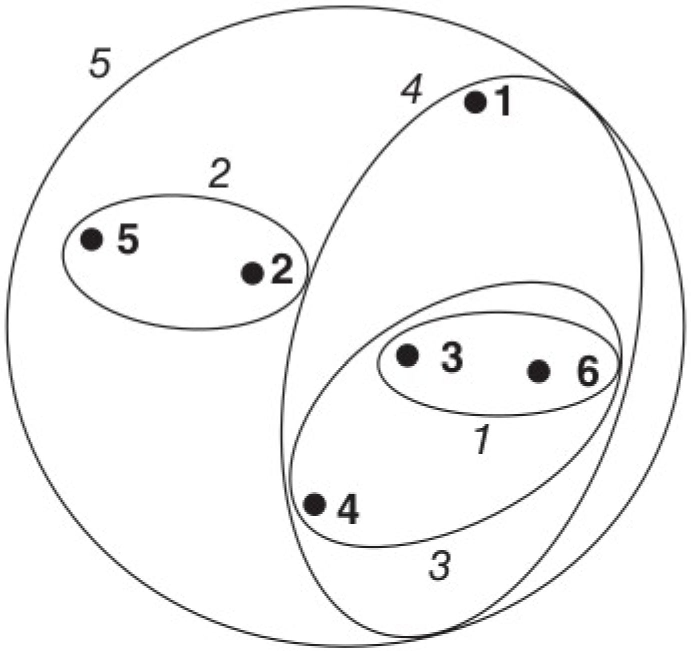
$$d_{ij} = d(\{X_i\}, \{X_j\}) = \|X_i - X_j\|^2.$$

#datos de cada cluster

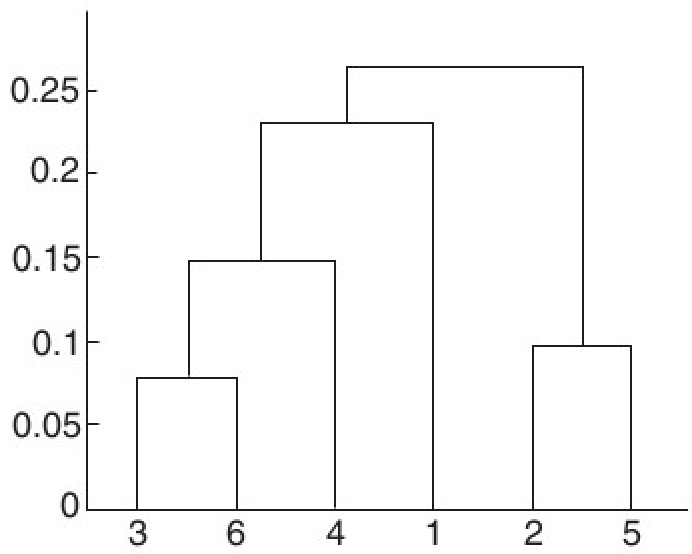
$$\Delta(A,B) = \sum_{i \in A \bigcup B} ||\overrightarrow{x_i} - \overrightarrow{m}_{A \bigcup B}||^2 - \sum_{i \in A} ||\overrightarrow{x_i} - \overrightarrow{m}_A||^2 - \sum_{i \in B} ||\overrightarrow{x_i} - \overrightarrow{m}_B||^2 = \frac{n_A n_B}{n_A + n_B} ||\overrightarrow{m}_A - \overrightarrow{m}_B||^2$$

Centroide del nuevo cluster

Tarea: demostrar

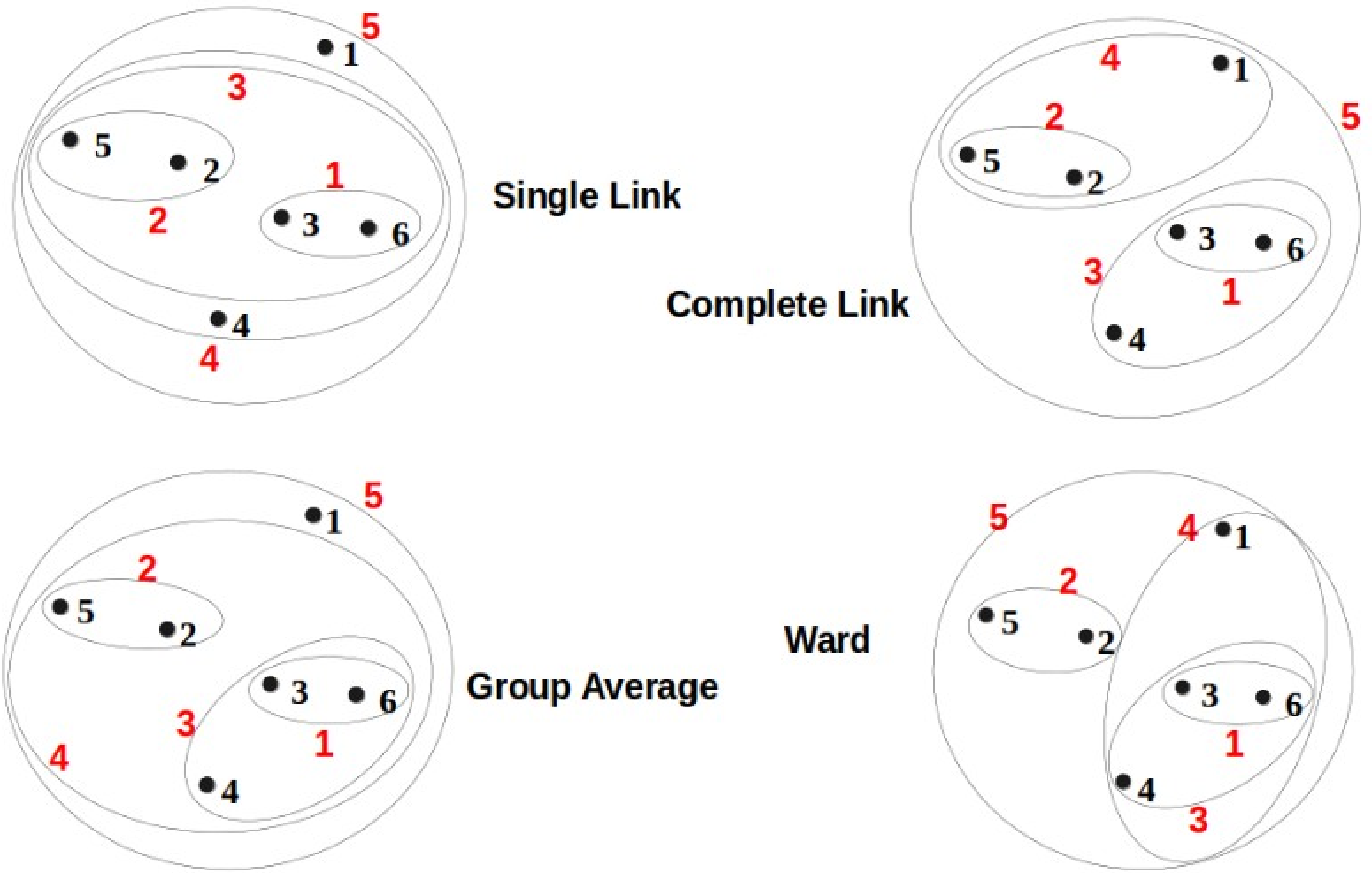


(a) Ward's clustering.

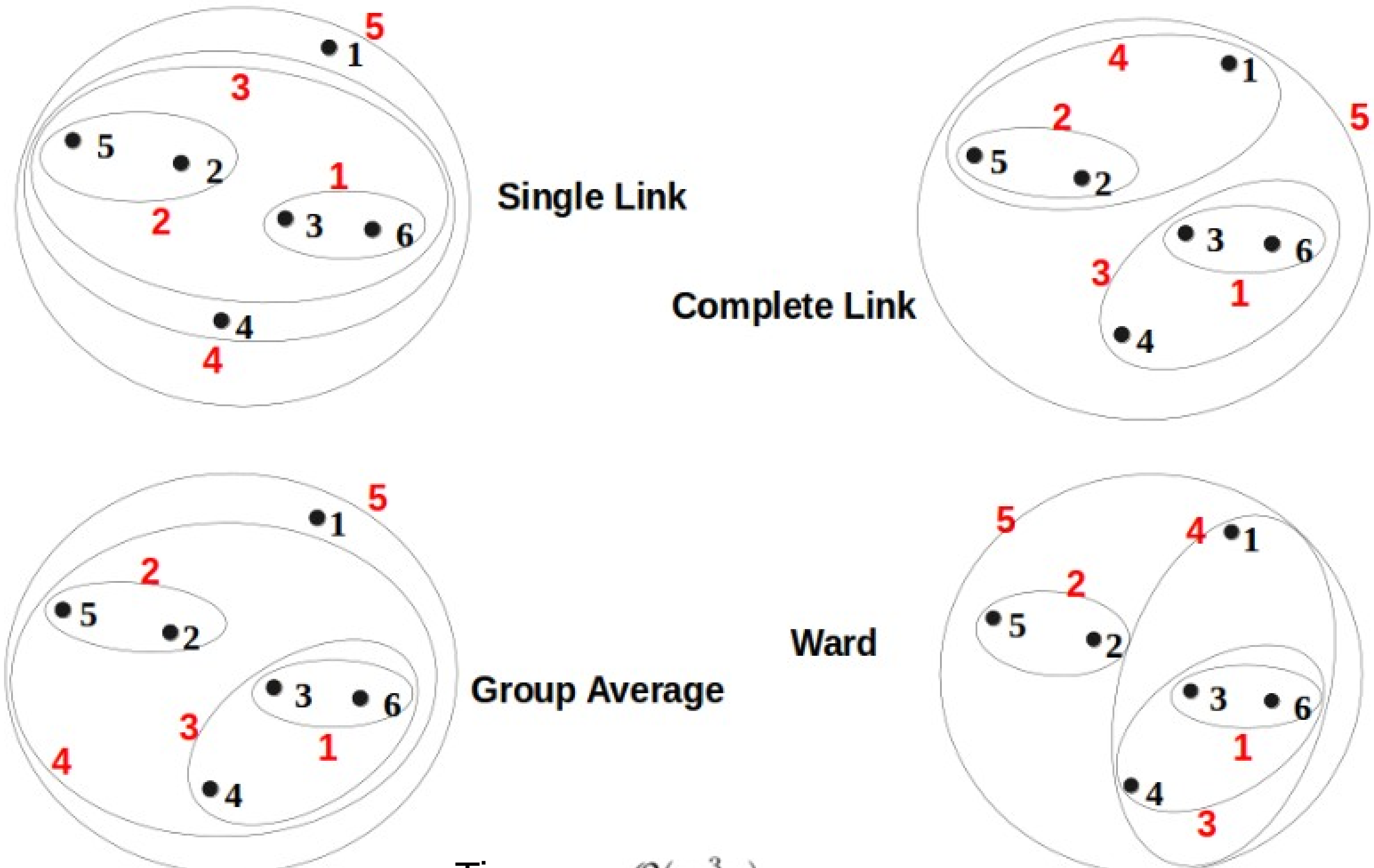


(b) Ward's dendrogram.

Clustering Jerárquico Aglomerativo



Clustering Jerárquico Aglomerativo



Tiempo: $\mathcal{O}(m^3 n)$

Espacio: $\Omega(m^2)$

m: #datos

n: dim

Clustering Jerárquico

Limitaciones

- No escala a grandes volúmenes de datos.
- No existen criterios claros para elegir la función de distancia y el criterio de mezcla.
- No trabaja con *missing values*.
- No es claro como trabajar con datos mezclados.

Clustering Jerárquico

Limitaciones

- No escala a grandes volúmenes de datos.
- No existen criterios claros para elegir la función de distancia y el criterio de mezcla.
- No trabaja con *missing values*.
- No es claro como trabajar con datos mezclados.

Cuando usarlo

- Data homogénea en espacio métrico.
- Volumen de datos a lo más de escala media (m ~ 10^4).
- Dendrogramas prestan mayor utilidad en *datasets* pequeños.

Clustering Jerárquico en sklearn

Setting:

```
> import numpy
> import sklearn
> import scipy
> import matplotlib
> from matplotlib import pyplot as plt
```

Dataset (digits):

```
> from sklearn import datasets
> digits = datasets.load_digits(n_class=10)
> X = digits.data
> y = digits.target
> n_samples, n_features = X.shape
```

Clustering Jerárquico en sklearn

Data embedding (en 2D):

```
> from sklearn import manifold
> X_red = manifold.SpectralEmbedding(n_components=2).fit_transform(X)
```

Normalización:

```
> x_min, x_max = numpy.min(X_red, axis=0), numpy.max(X_red, axis=0)
> X_red = (X_red - x_min)/(x_max - x_min)
```

Agglomerative Clustering:

```
> from sklearn.cluster import AgglomerativeClustering as hac
> clustering = hac(linkage="complete", n_clusters=10)
> clustering.fit(X_red)
```

Clustering Jerárquico en sklearn

Visualización (en 2D):

```
> for i in range(X_red.shape[0]):
...     plt.text(X_red[i,0], X_red[i,1], str(y[i]),
...     color=plt.cm.spectral(clustering.labels_[i]/10.),
...     fontdict={'weight': 'bold', 'size': 8})
...
> plt.show()
```