Clustering

Diplomatura en Ciencia de Datos,
Aprendizaje Automático y sus Aplicaciones
FaMAF-UNC
agosto 2019

Mapa de ruta

1. Embeddings
2. **Clustering**, y revisitar todos los conceptos que vimos hasta ahora
3. Reglas de Asociación
4. K-nn y recomendación
5. Grafos
6. Aprendizaje Semi-supervisado

Entregables:

- Trabajos con sus mentores
- para ello van a tener a disposición notebooks para rehacer las figuras de la clase

Mapa de ruta

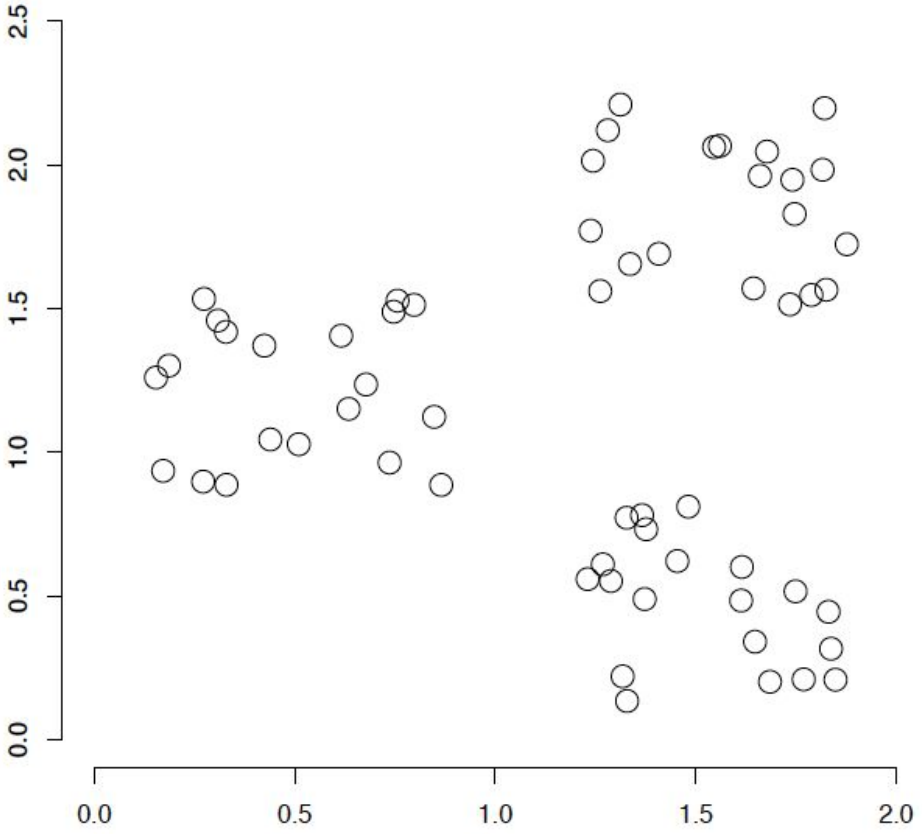
1. Cómo funciona
2. Evaluación
3. Qué puedo esperar
4. Metodología iterativa
5. Ejemplos con notebooks

Cómo funciona clustering

Agrupar objetos semejantes

- Entrada: objetos en un espacio n-dimensional
- Salida: una **solución** con grupos (**clusters**) de objetos semejantes ➜ cercanos en el espacio
  - Se minimiza la distancia entre los objetos de un mismo grupo
  - Se maximiza la distancia entre los objetos de distintos clusters
- Los centros de cada cluster son los **centroides**

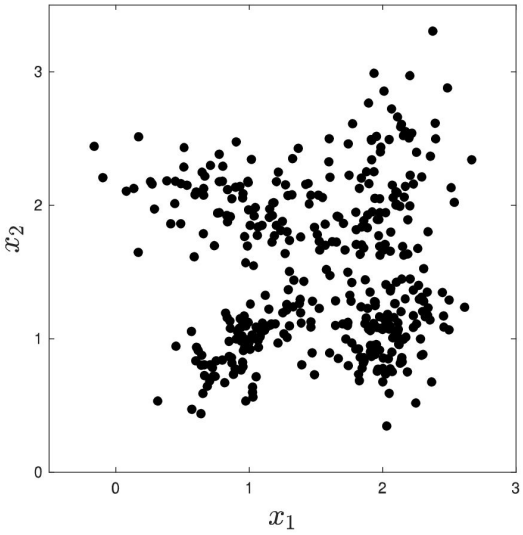
Dataset con clara estructura de clusters



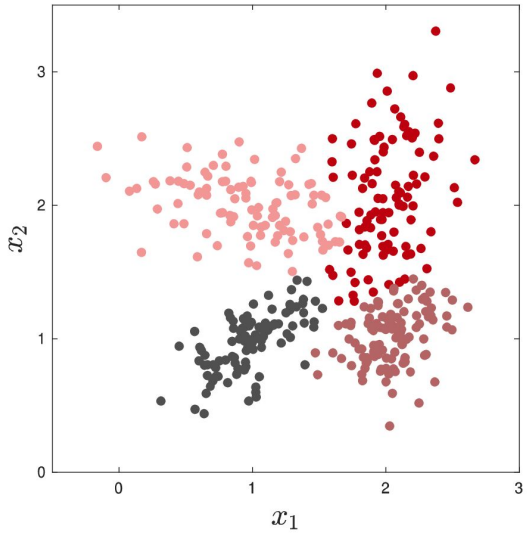
¿Cómo sería un algoritmo para encontrar clusters en este espacio?

Dataset con no tan clara estructura de clusters

¿Cómo sería un algoritmo para encontrar clusters en este espacio?



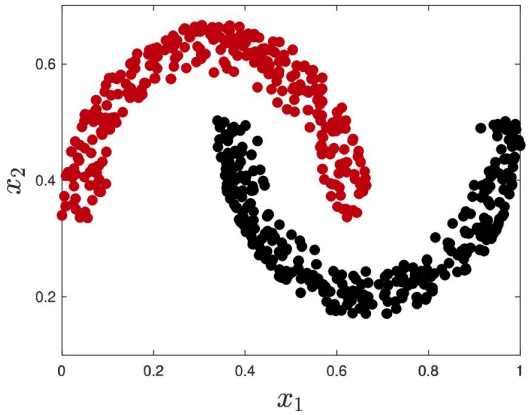
(a)



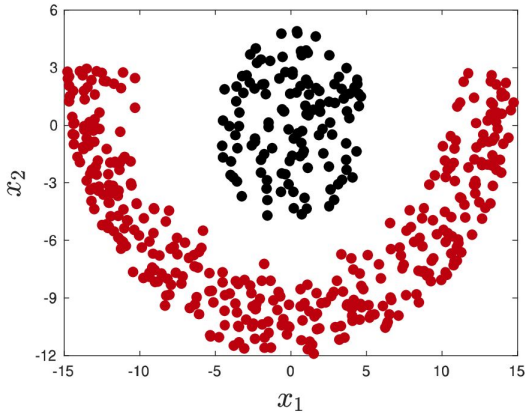
(b)

Dataset con clara estructura de clusters

¿Cómo sería un algoritmo para encontrar clusters en este espacio?



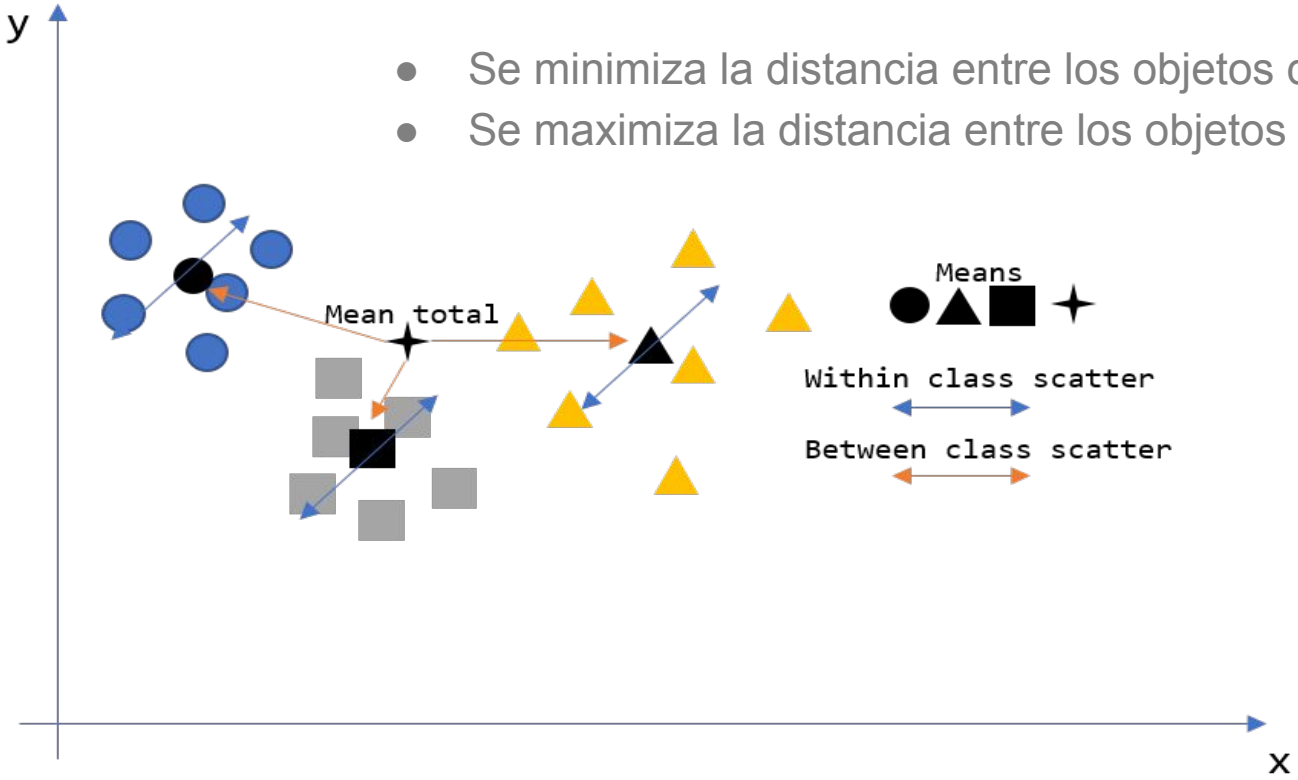
(a)



(b)

Cómo funciona clustering

- Se minimiza la distancia entre los objetos de un mismo grupo
- Se maximiza la distancia entre los objetos de distintos clusters¶



Cuestiones cruciales

- ¿Cómo es el espacio? ¿Cómo represento mis problemas?
- ¿Cómo se calcula la distancia (semejanza) en este espacio?
- ¿Cuántos clusters quiero distinguir?
- ¿Qué distribución tienen estos clusters? ¿Gaussiana? ¿En serie?
- ¿Busco una estructura jerárquica o plana?
- ¿Cómo veo qué hay en cada cluster?
- ¿Cómo evalúo la bondad de cada solución?

Cuestiones cruciales

- **¿Cómo es el espacio? ¿Cómo represento mis problemas?**

  - Es multi dimensional?
  - Mis datos son naturalmente categóricos? ordinales? contínuos?
  - Tengo informacion que me permita decir que debería encontrar grupos compactos?
  - No se nada y quiero usar clustering en forma exploratoria

-

Cuestiones cruciales

- **¿Cómo se calcula la distancias entre objetos en este espacio?**
  - Es un espacio Euclideo? Métrica usual anda bien? Conviene usar ángulos en vez de distancias?
  - No es un espacio Euclídeo? Similaridades ? Matriz de afinidad?

- Entender mi espacio me ayuda a elegir un método más razonable.
- Si mi método más razonable no me da nada, quizás sea porque no hay nada para ver...
-

Cuestiones cruciales

- **¿¿Cuántos clusters quiero distinguir??**
  - Tengo información? Hice varios experimentos? tengo varias databases de días diferentes y locales diferentes?
  - Clustering exploratorio, debo estudiar los distintos agrupamientos para distintos números de clusters.
  - Distintas técnicas para encontrar el mejor modelo de agrupamiento.

Cuestiones cruciales

- **¿Busco una estructura jerárquica o plana?**

  - Si mis clusters estan anidados, tengo una estructura muy fuerte que explica los datos
  - Si mis clusters son estructuras cercanas, las une sin remedio

Cuestiones cruciales

- **¿¿Cómo veo qué hay en cada cluster?**
  - Visualización es la pesadilla. Rapido de correr, lento de analizar!!!
  - Proyecciones en espacios de menor dimensión ayudan a visualizar los resultados.
  - Principal component analysis (PCA),
  - t-distributed Stochastic Neighbor Embedding (t-SNE)

Cuestiones cruciales

- **¿Cómo evalúo la bondad de cada solución?**
  - En clustering hay dos conceptos de medida, una para generar la partición y otra para evaluar la partición.
  - La primera es en el espacio de los datos y la segunda en el espacio de los posibles clusterings.
- Empíricas
  - Silhouette Score
  - Elbow method
- Matemáticas
  - Rand measure
  - Mutual Information score

Semejanza (Distancia)

- La semejanza debería acercarse a las causas latentes
  - Entre documentos: semántica
  - Entre clientes: motivación para las compras
  - Entre imágenes: objetos físicos que representan
  - Entre propiedades inmobiliarias: elementos que otorgan valor
- Idealmente, debería calcularse de forma independiente para cada dimensión

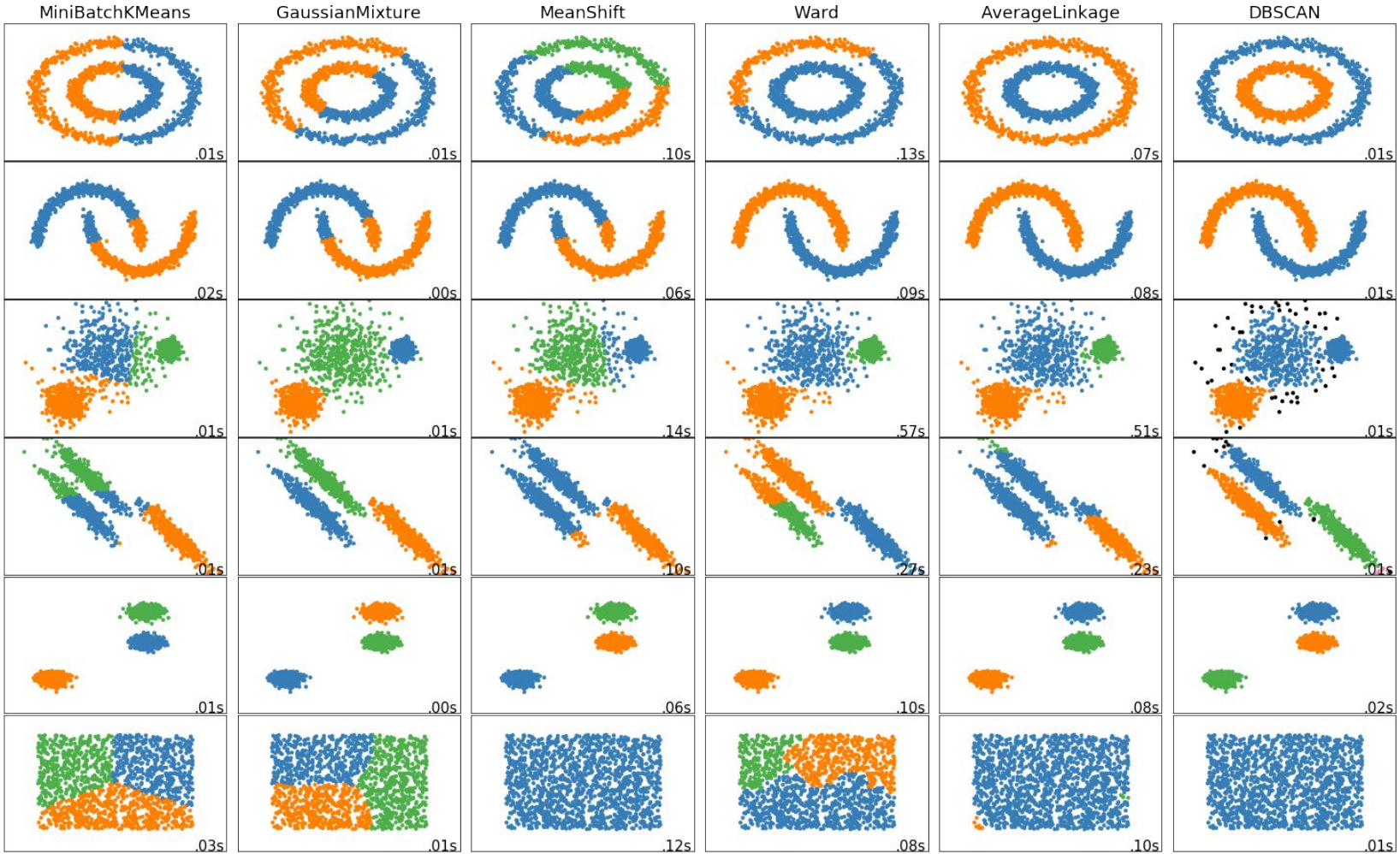
Distancias (semejanzas)

- Euclídea
- Coseno ➜ normalizado por longitud, producto punto ➜ correlación!
- Distancia de Manhattan
- Distancia de Edición (Levenshtein)

- Divergencia de Kullback-Leibler

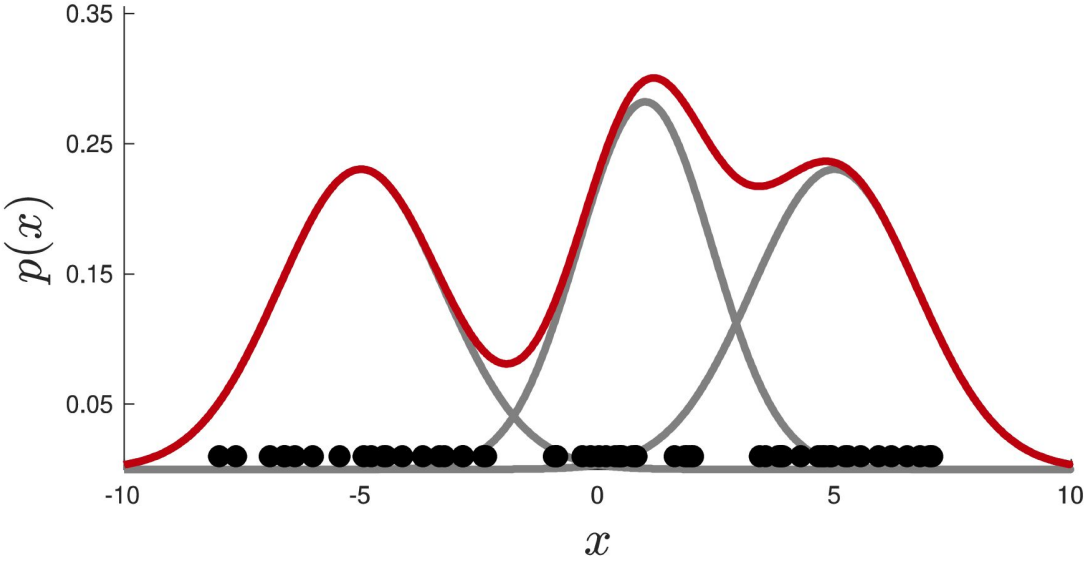
Familias de algoritmos de clustering

- Métodos generativos
  - Mezcla de gaussianas, MeanShift
- Agrupamiento por particiones
  - k-Means, PAM/CLARA/CLARANS
- Métodos basados en densidad
  - DBSCAN, Optics, DenClue
- Clustering jerárquico
  - Ward, Diana/Agnes, BIRCH, CURE, Chameleon, ROCK


MiniBatchKMeans
GaussianMixture
MeanShift
Ward
AverageLinkage
DBSCAN
.01s
.01s
.10s
.13s
.07s
.01s
.02s
.00s
.06s
.09s
.08s
.01s
.01s
.01s
.14s
.57s
.51s
.01s
.01s
.01s
.10s
.27s
.23s
.01s
.01s
.00s
.06s
.10s
.08s
.02s
.03s
.01s
.12s
.08s
.10s
.01s

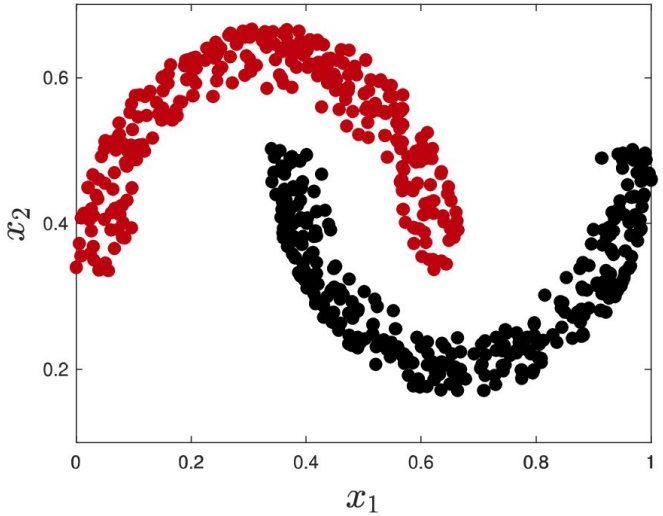
Mezcla de Gaussianas

- Supongamos tener alguna información
  - Consideremos que estos datos son reales,
  - puedo trabajar con la distancia Euclídea.
  - datos producidos por una densidad mezcla de Gaussianas,

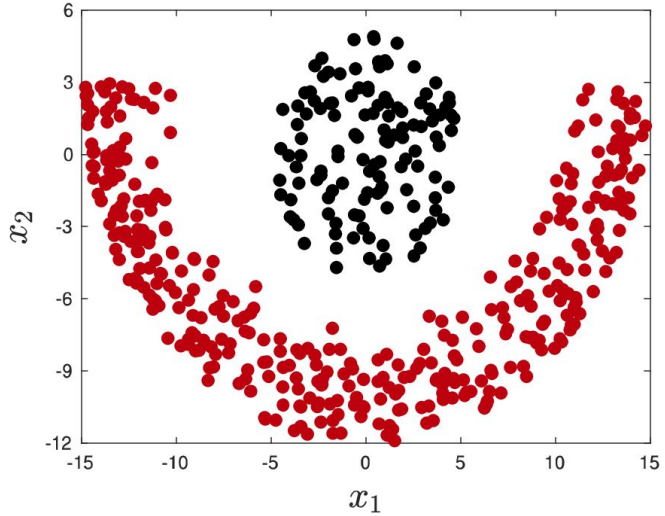


Mezcla de Gaussianas

- Cualquier dato puede ser modelado con una mezcla de gaussianas?



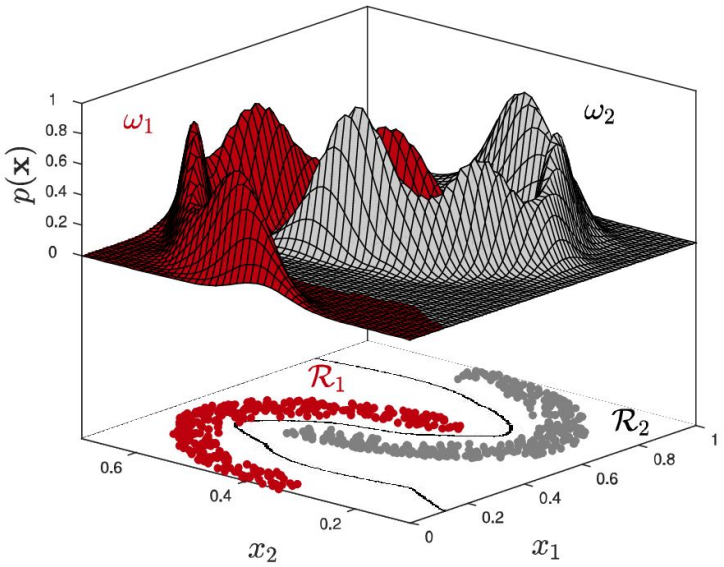
(a)



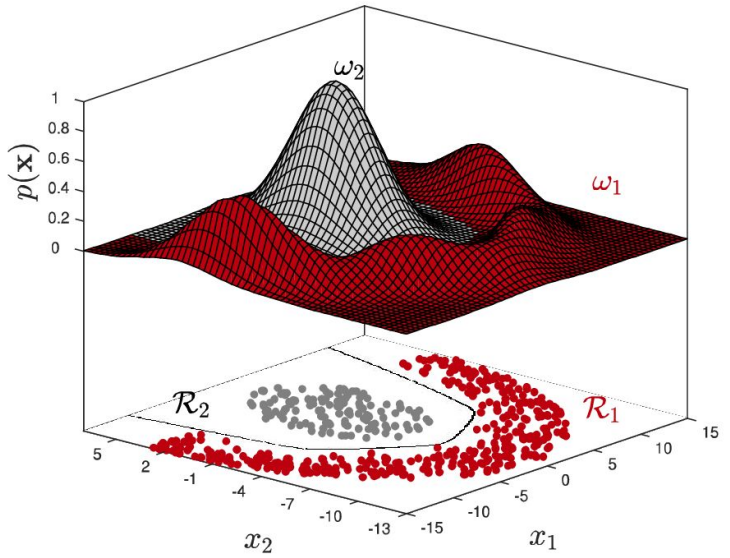
(b)

Mezcla de Gaussianas

- No todos, pero muchos si se pueden modelar, si uno conoce la cantidad de gaussianas que forman la mezcla
-



(a)



(b)

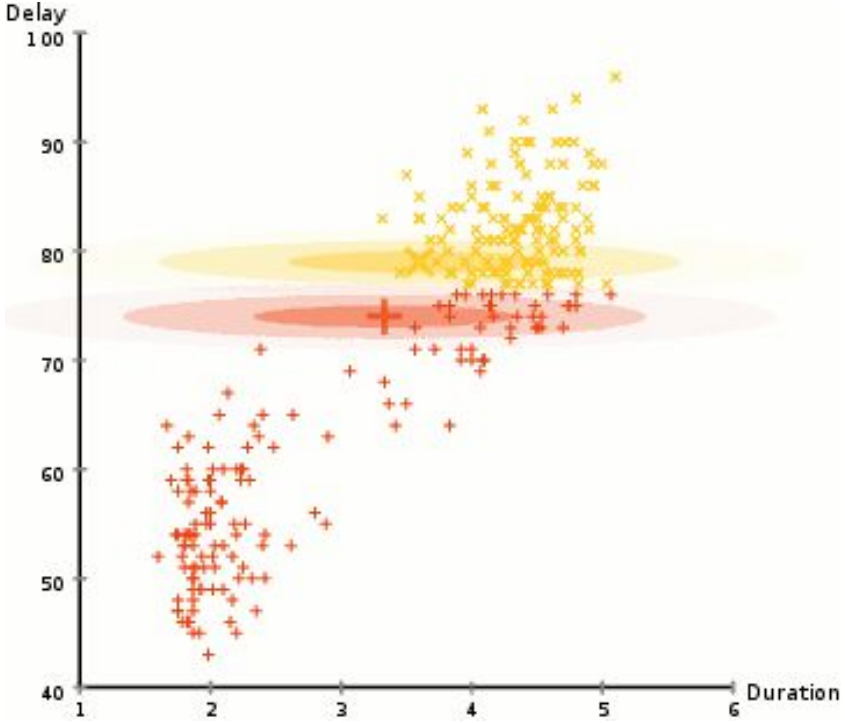
Como funciona el GMM?

- Si uno fija la cantidad de gaussianas que uno considera que hay en la mezcla,
  - se estiman los parámetros de cada gaussiana y los parámetros de representación
  - se imputa cada dato como proveniente de la una de las componentes de la mezcla.
  - La estimación se realiza mediante el algoritmo Expectation Maximization.

-

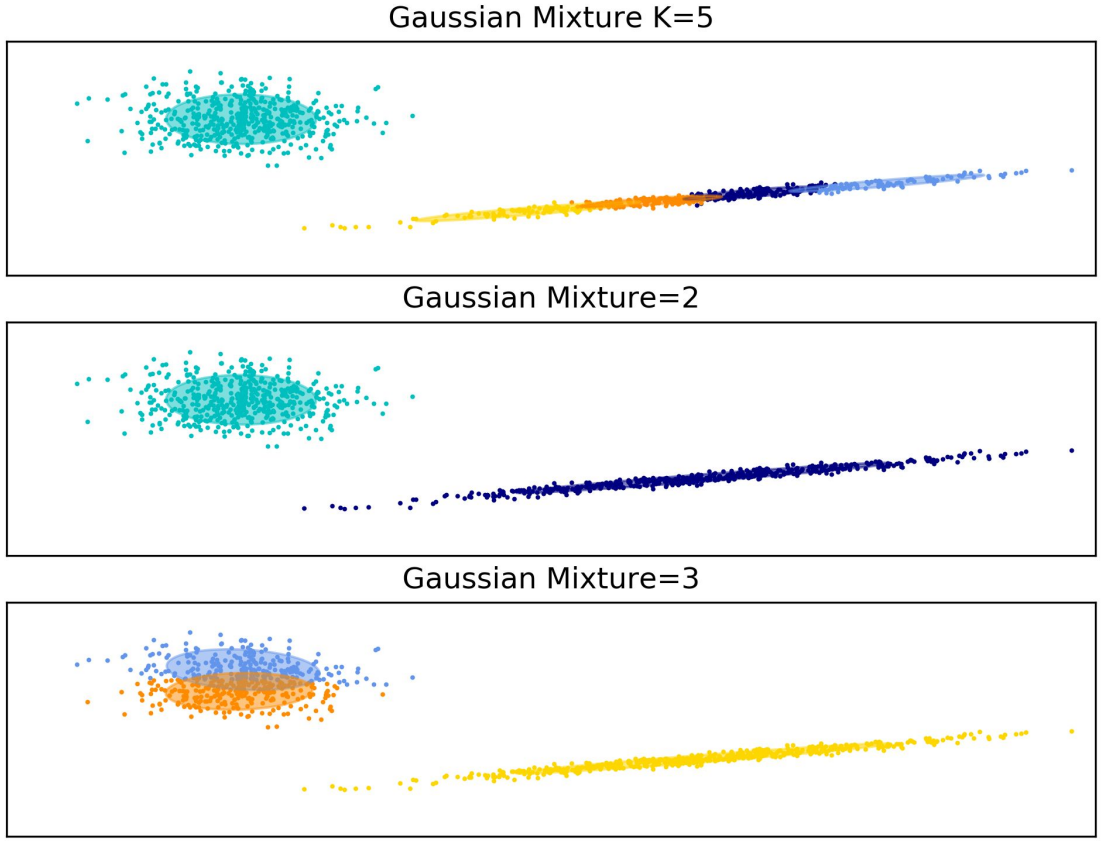
Como funciona el GMM?

- Comenzamos con una partición aleatoria de la cual se sacan los parámetros de inicio y desde allí se itera



Parámetros

- Gran problema de GMM es la determinación del número de componentes de la mezcla
- Si no se elige un buen número, el modelo particiona de forma aglutinada pero los clusters pueden no tener sentido.
- La otra característica que puede ser forzada de inicio es el tipo de matriz de varianza covarianza.
- Este ejemplo (Notebook1) ha sido realizado modelando matrices de covarianza full usando el módulo sklearn.


Gaussian Mixture K=5
Gaussian Mixture=2
Gaussian Mixture=3

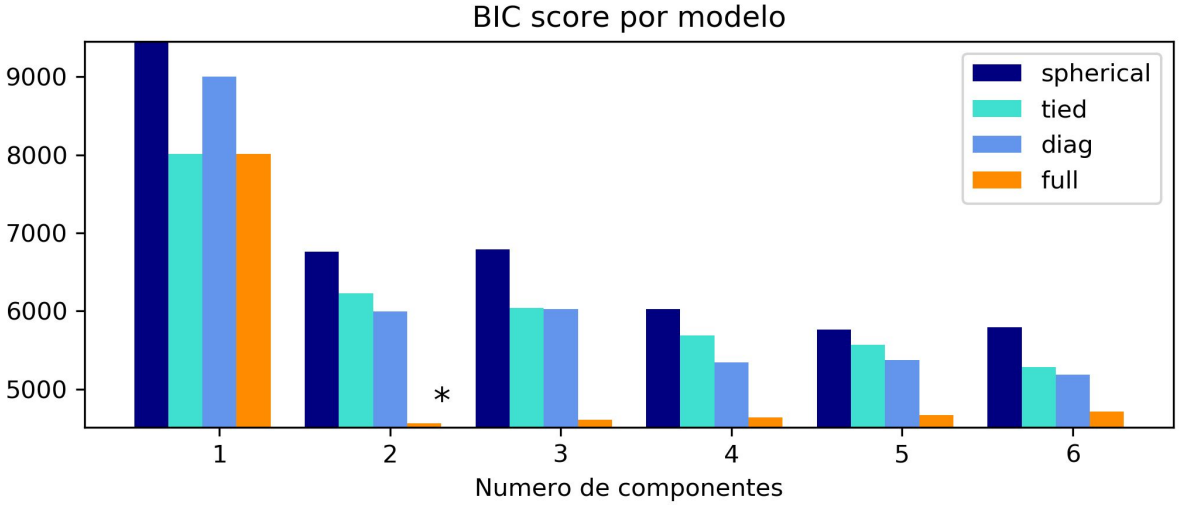
Detección automática de k

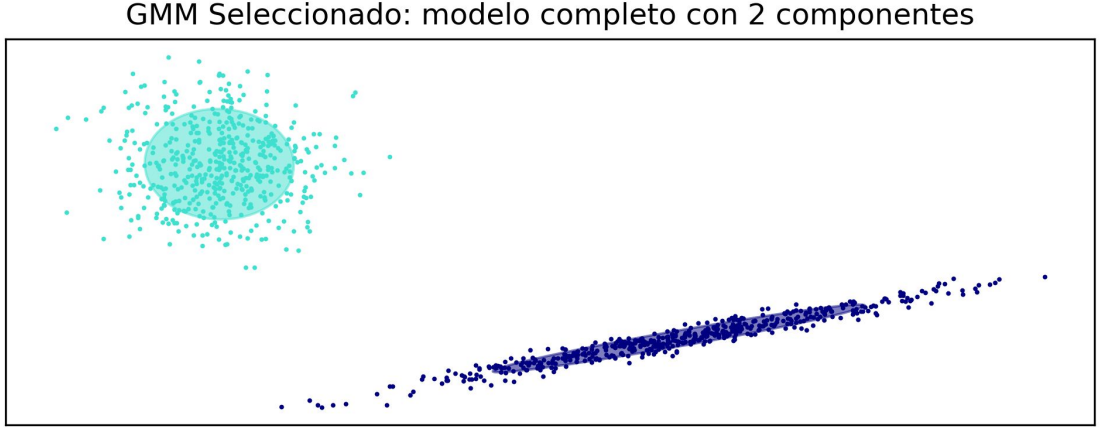
Bayesian Information Criterium (BIC) da un score al modelo con m parámetros.

$BIC = -2 * log(L(\Theta)) + log(n)m$

Puede usarse otro índice llamado Akaike Information Criterium (AIC) $AIC = -2 * log(L(\Theta)) + 2m$

donde L(Θ) es la verosimilitud, n el número de datos, y m el número de parámetros estimados, (k, el número de componentes, más las medias y entradas de la matriz de varianza covarianza.)

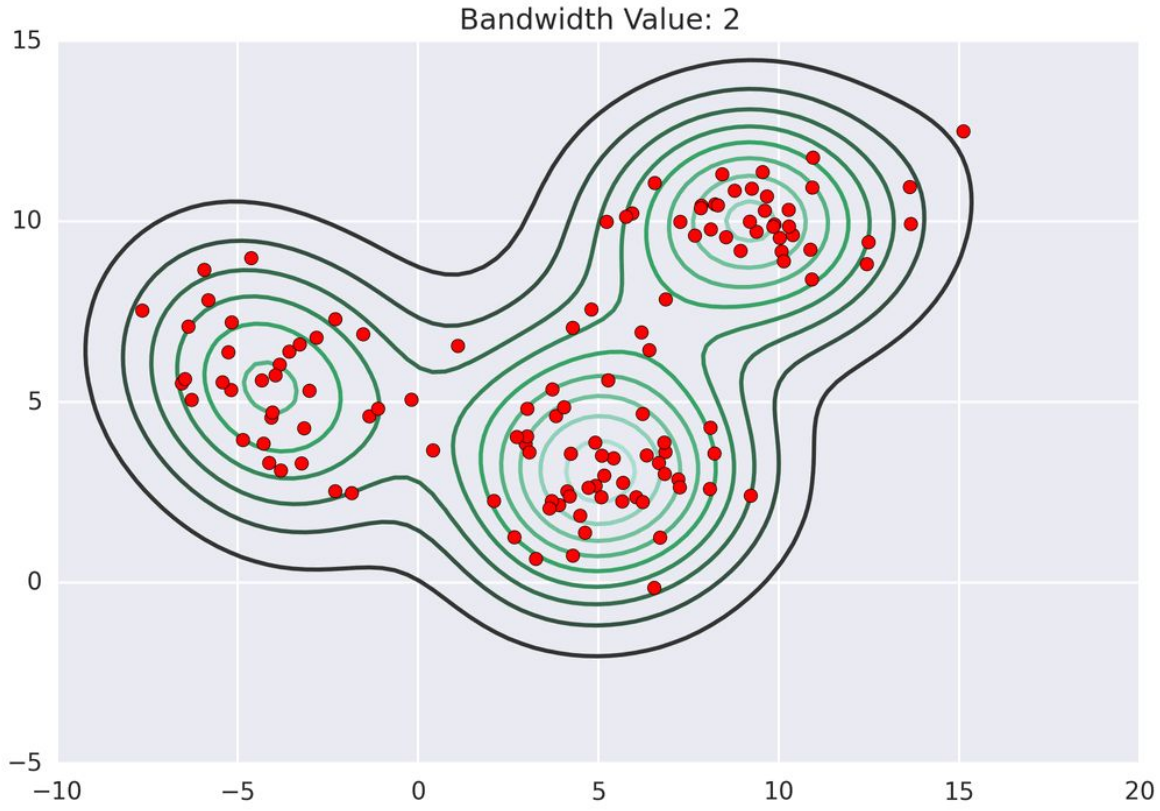

BIC score por modelo
9000
8000
7000
6000
5000
spherical
tied
diag
full
*
1
2
3
4
5
6
Numero de componentes


GMM Seleccionado: modelo completo con 2 componentes

Mean Shift Algorithm

- Mean shift se basa en el concepto de kernel density estimation (KDE)
- Si los datos se suponen muestreados de una distribución de probabilidad, KDE es un estimador no paramétrico de la densidad asociada a dicha distribución.
- KDE aplica un kernel, esto es, una función de peso, en una ventana alrededor del punto con un ancho de banda (bandwidth) determinado. Sumando todos las estimaciones individuales se obtiene el estimador de la densidad.
- Para generar la partición, el algoritmo Mean-Shift Clustering va deslizando la ventana y computando el promedio de los datos pesados por el kernel, para localizar las áreas de alta densidad.
- Cada moda de la densidad va a ser considerada un centroide, y los puntos de la partición van a ser asignados al centroide más próximo

Mean Shift Algorithm



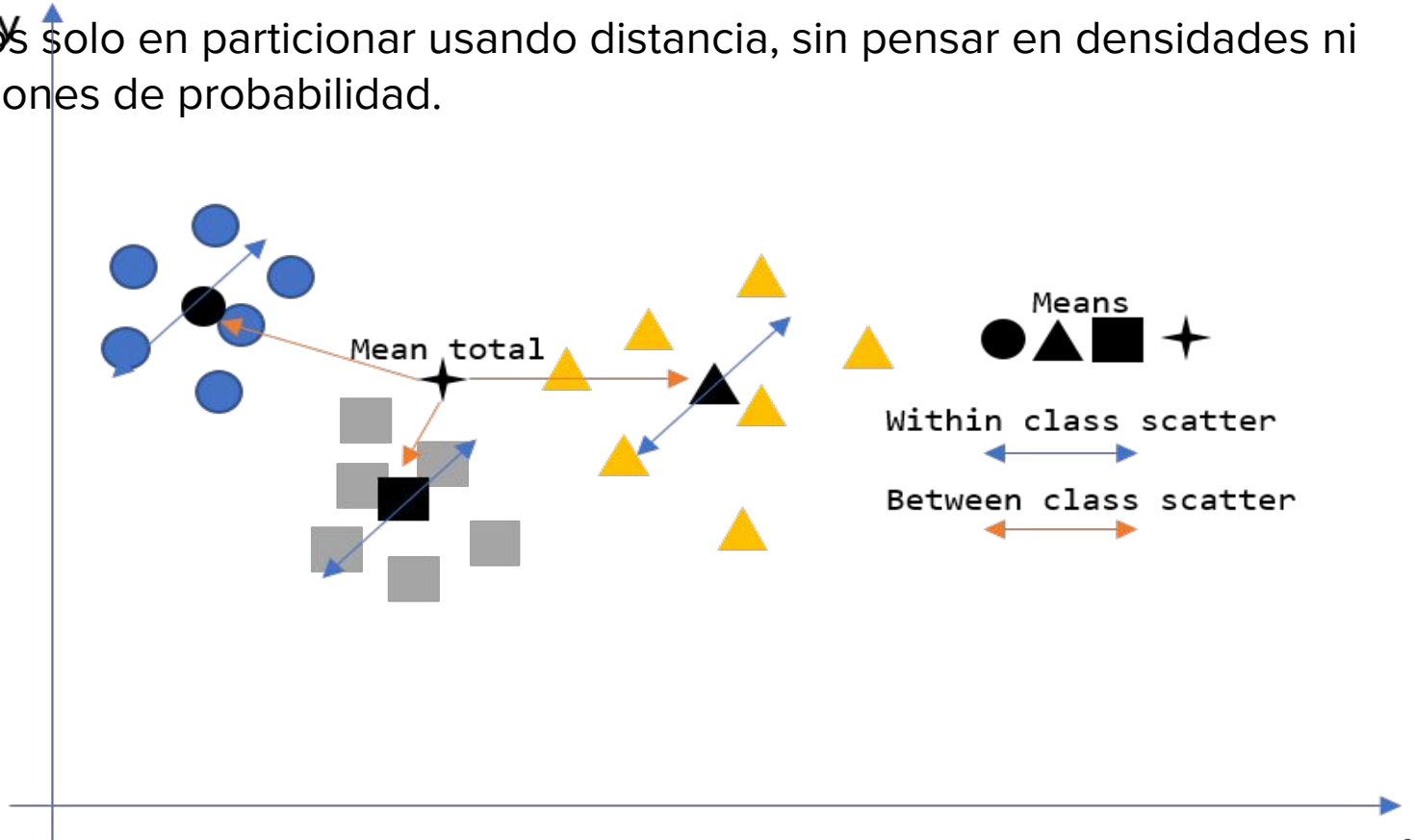
Mean Shift Algorithm

Mean Shift Algorithm

- Parámetro bandwidth puede ser estimado utilizando la teoría no paramétrica, dependiendo de que kernel se use.
- Notebook2 tiene un ejemplo de estimación usando Sklearn.

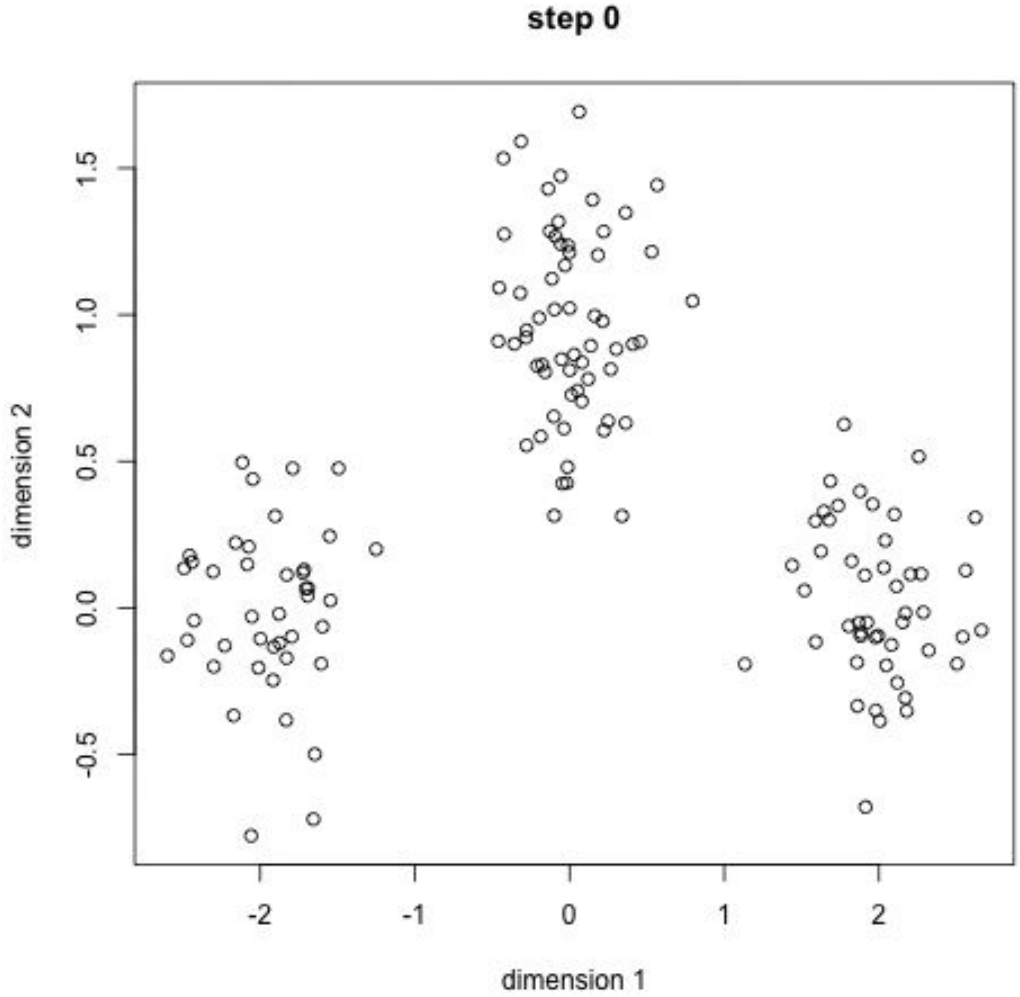
K-means

- Pensemos solo en particionar usando distancia, sin pensar en densidades ni distribuciones de probabilidad.



Kmedias

- K medias es el algoritmo más usado para aglomerar datos.
  - K medias comienza por elegir k centros aleatorios.
  - Después, todos los puntos son asignados al centro más cercano basado en la distancia euclídea, lo cual genera una partición del espacio.
  - Luego los centros son re calculados usando la nueva partición y el ciclo comienza nuevamente.
  - Este proceso continúa hasta que no haya más cambios en la partición entre iteraciones.
- Este algoritmo genera una partición similar a la de la mezcla de Gaussianas Esféricas, esto es, con una matriz de varianza Covarianza múltiplo de la Identidad.


step 0
dimension 2
1.5
1.0
0.5
0.0
-0.5
-2
-1
0
1
2
dimension 1

K-means

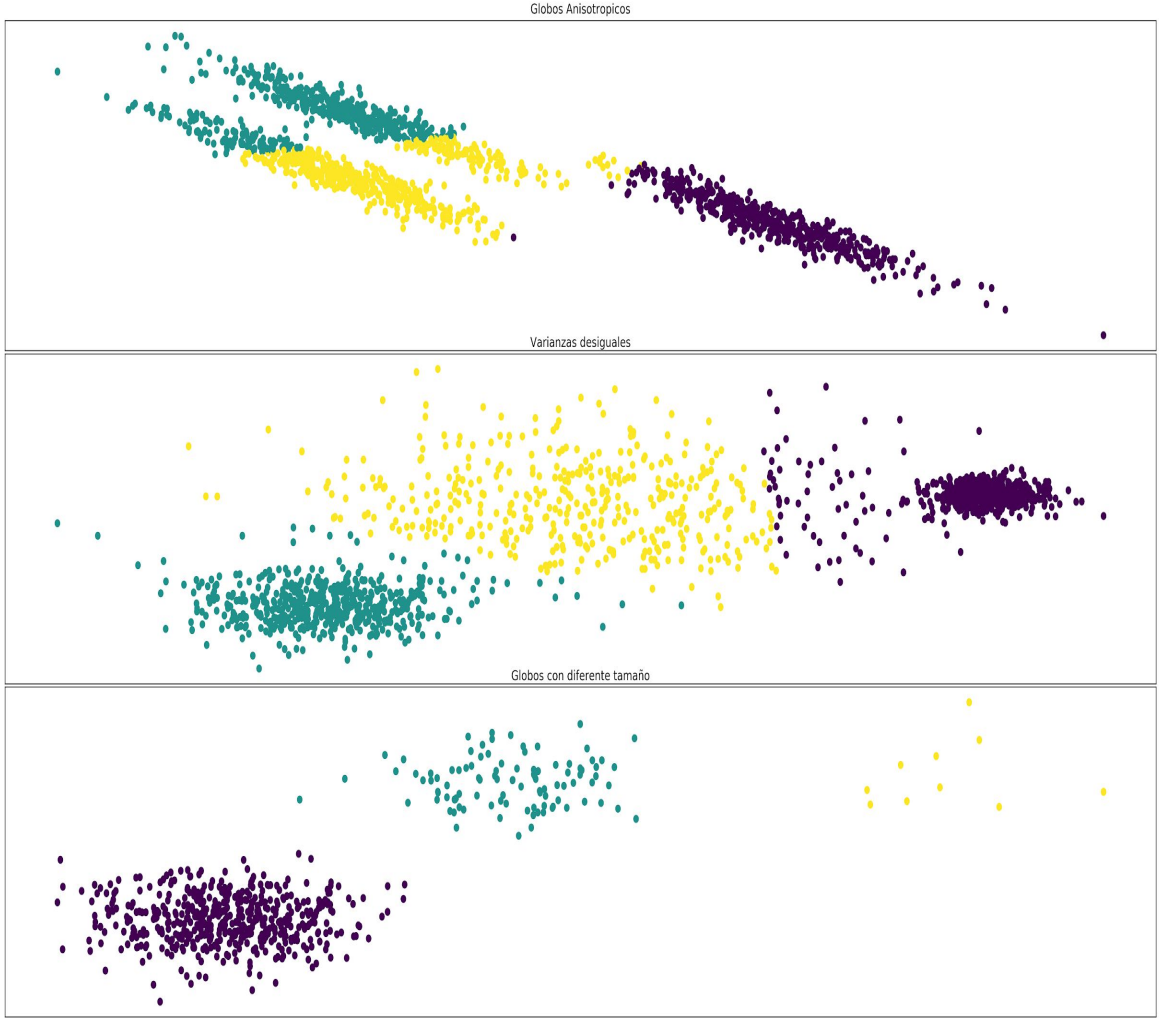
Problemas

- Inestabilidad
- Mínimos locales (mucha sensibilidad a las semillas)
- Soluciones globales ➔ sensibles a outliers
- El número de clusters k suele ser desconocido

Parámetros

- Inicialización
- número de veces que se vuelven a tirar las semillas
- cuántas iteraciones hasta que termina la búsqueda

K-means


Globos Anisotropicos
Varianzas desiguales
Globos con diferente tamaño

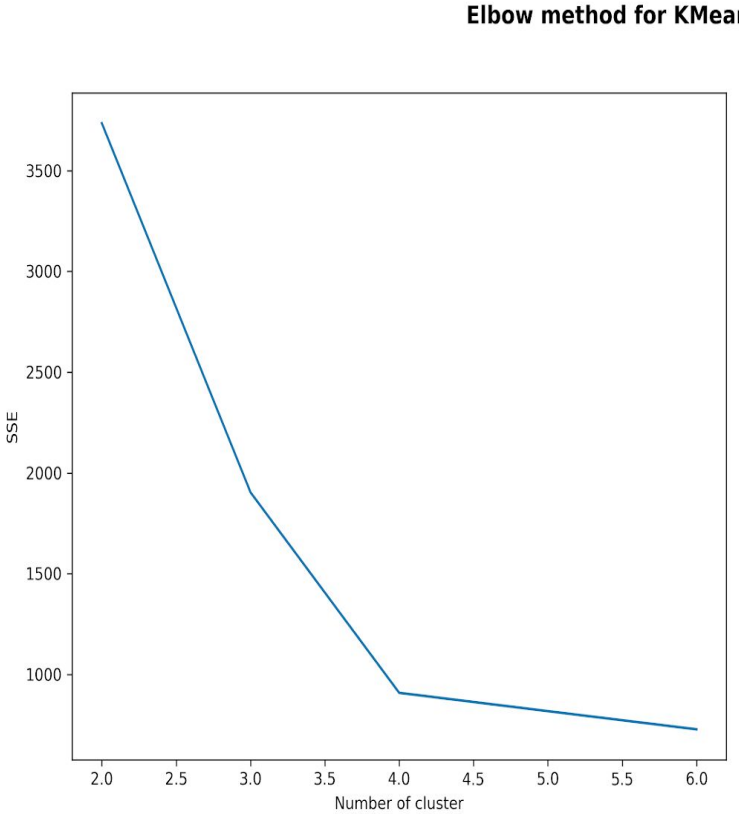
K-means

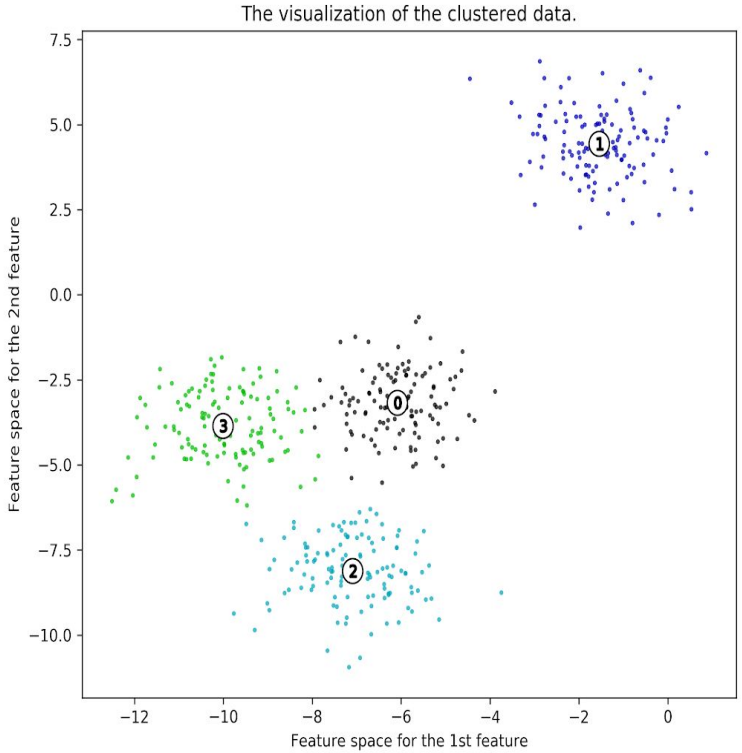
Cómo encontramos el k ?

- No podemos usar BIC, o MDL o AIC porque no usamos un modelo de verosimilitud para ajustar.
- Pero si podemos comparar entre diferentes modelos en función de k el valor de la inercia del modelo, esto es, la suma de distancias cuadradas dentro de cada cluster de la partición final.
- La inercia se considera una medida de cuán coherentes los clusters son. La notebook3 muestra cómo computar el coeficiente silhouette y la inercia para elegir el k mas apropiado.

-

K-means


Elbow method for KMeans clustering on sample data
SSE
Number of cluster
3500
3000
2500
2000
1500
1000
2.0
2.5
3.0
3.5
4.0
4.5
5.0
5.5
6.0


The visualization of the clustered data.
Feature space for the 2nd feature
Feature space for the 1st feature
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
−12
−10
−8
−6
−4
−2
0
0
1
2
3

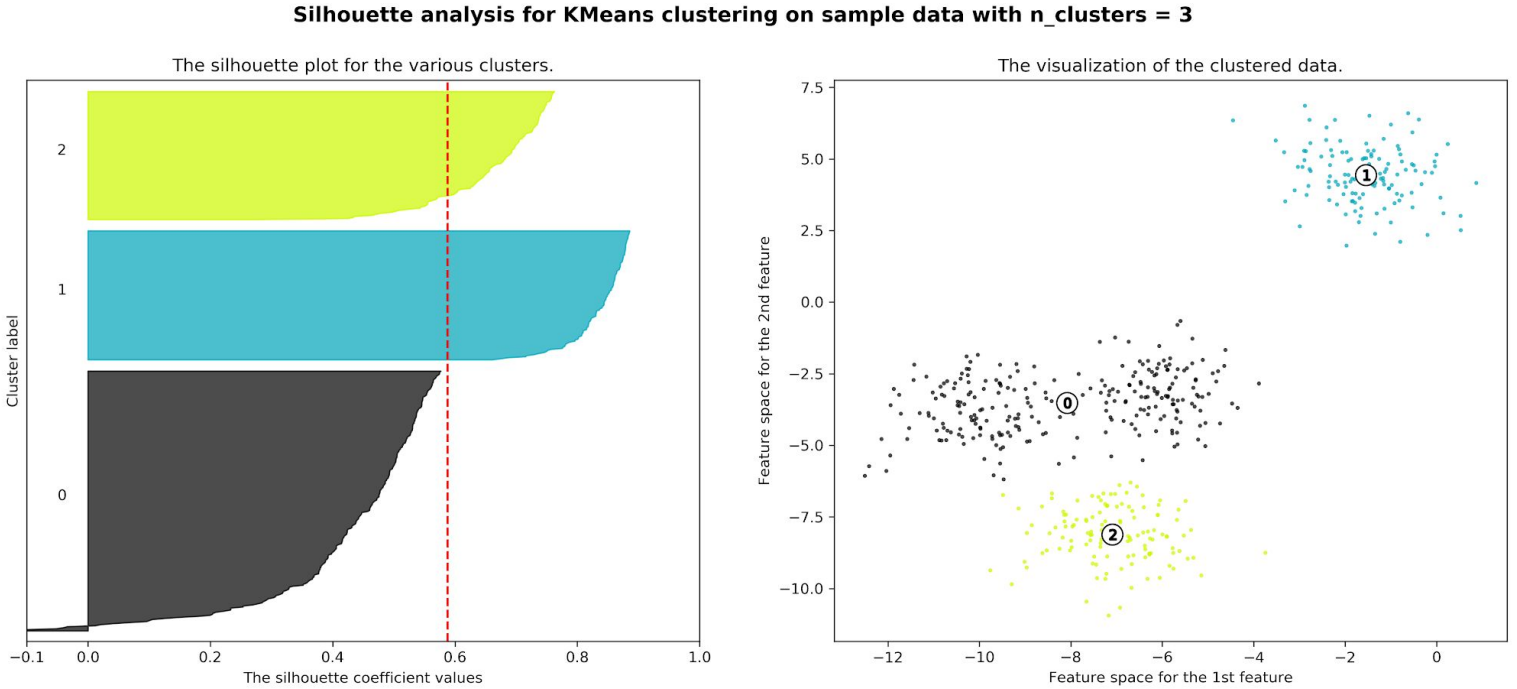
K-means: Análisis de siluetas

- Puede ser usado para estudiar la separación entre los clusters de la partición.
- El gráfico de silhoutte muestra una medida de cercanía entre los puntos de un cluster a los puntos de sus clusters vecinos.
- La medida s(i) tiene valor entre -1 y 1.
- Los coeficientes s(i) cercanos a +1 indican que la muestra está lejos de los clusters vecinos.
- El valor 0 indica que la muestra está muy cerca del borde de decisión entre los clusters.
- Un valor negativo indica que esos puntos deben haber sido asignados al cluster equivocado.

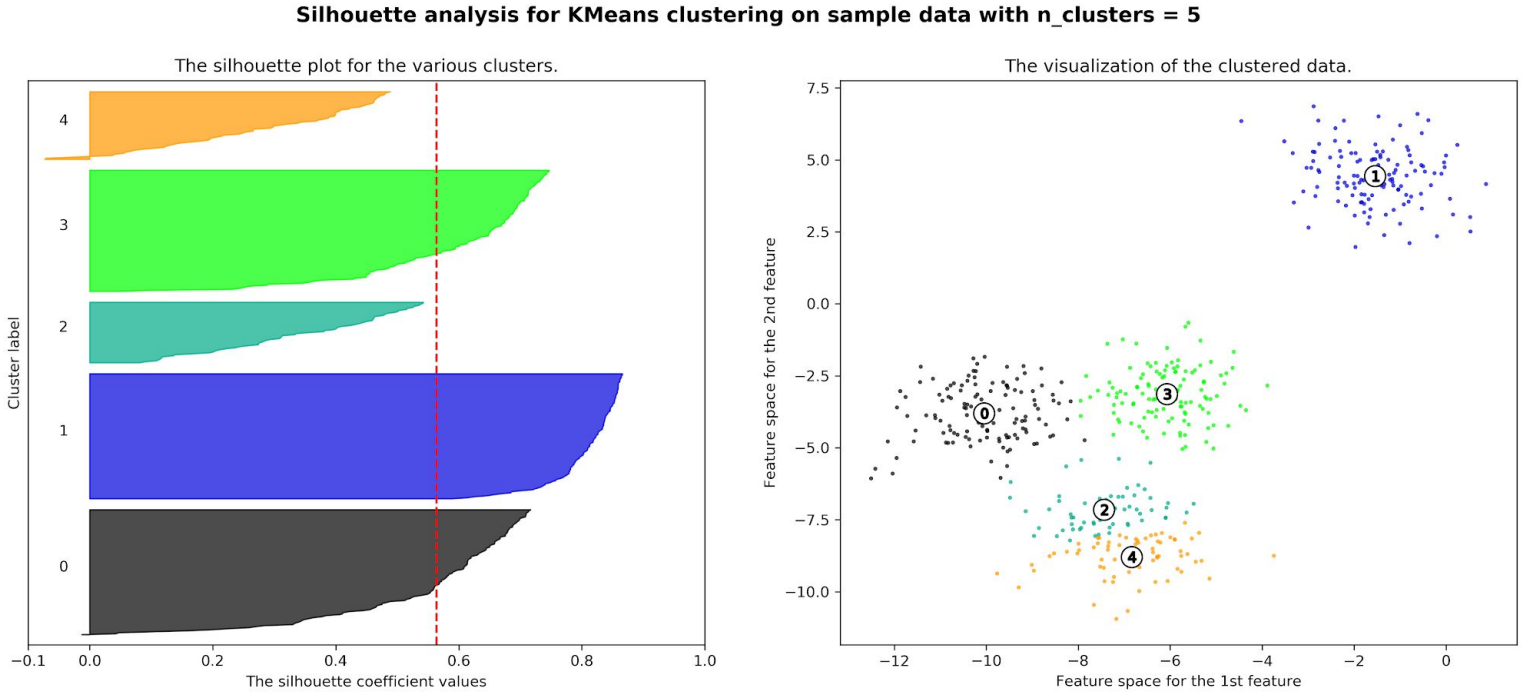
K-means: Ejemplo

- Se simula un grupo de datos con cuatro gaussianas.
- Se calcula el gráfico de silueta para particiones de k medias con K=2,3,4,5,y 6.
- El gráfico de silueta para los valores de k =3, 5 and 6 muestran que esos k son una mala elección, dado que hay clusters por debajo del valor de silueta promedio y clusters con valores negativos.

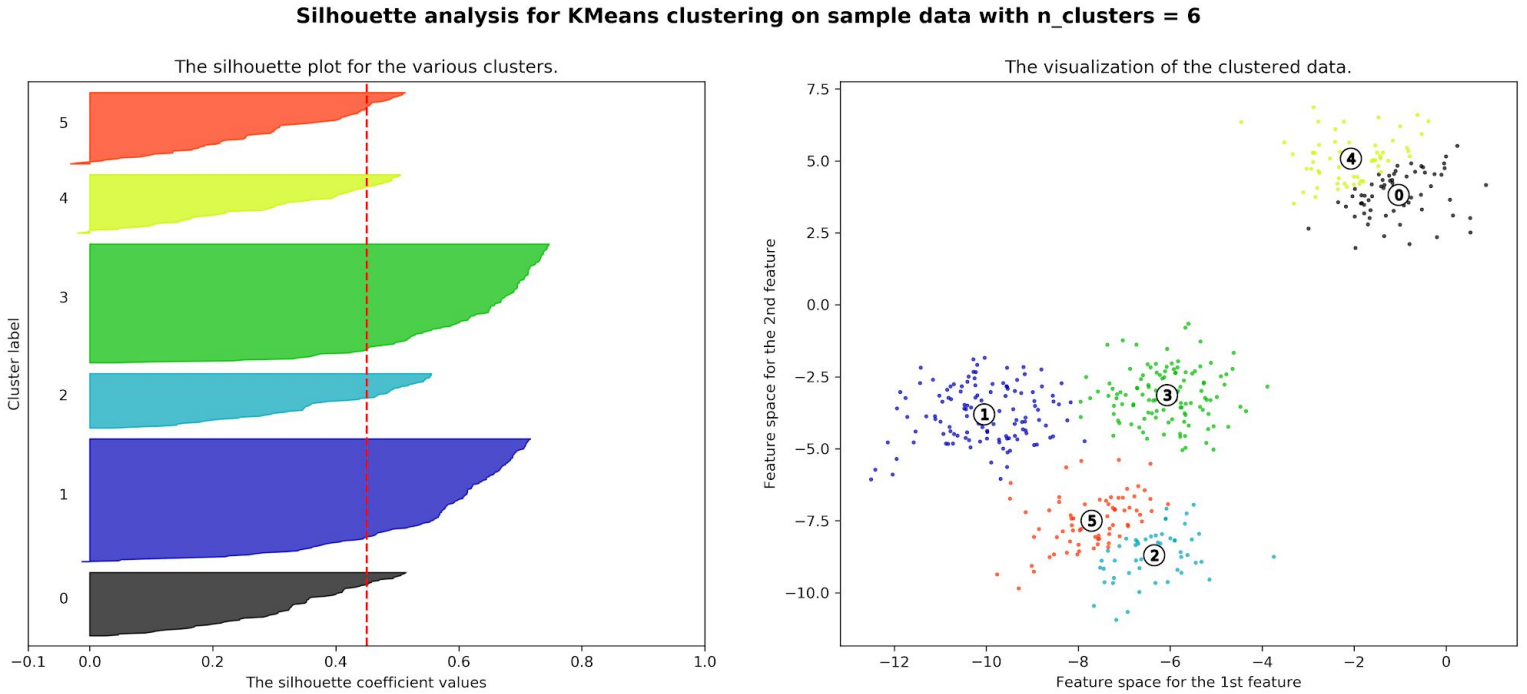
K-means


Silhouette analysis for KMeans clustering on sample data with n_clusters = 3
The silhouette plot for the various clusters.
Cluster label
2
1
0
−0.1
0.0
0.2
0.4
0.6
0.8
1.0
The silhouette coefficient values
The visualization of the clustered data.
Feature space for the 2nd feature
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
−12
−10
−8
−6
−4
−2
0
Feature space for the 1st feature
0
1
2

K-means


Silhouette analysis for KMeans clustering on sample data with n_clusters = 5
The silhouette plot for the various clusters.
Cluster label
The silhouette coefficient values
The visualization of the clustered data.
Feature space for the 2nd feature
Feature space for the 1st feature

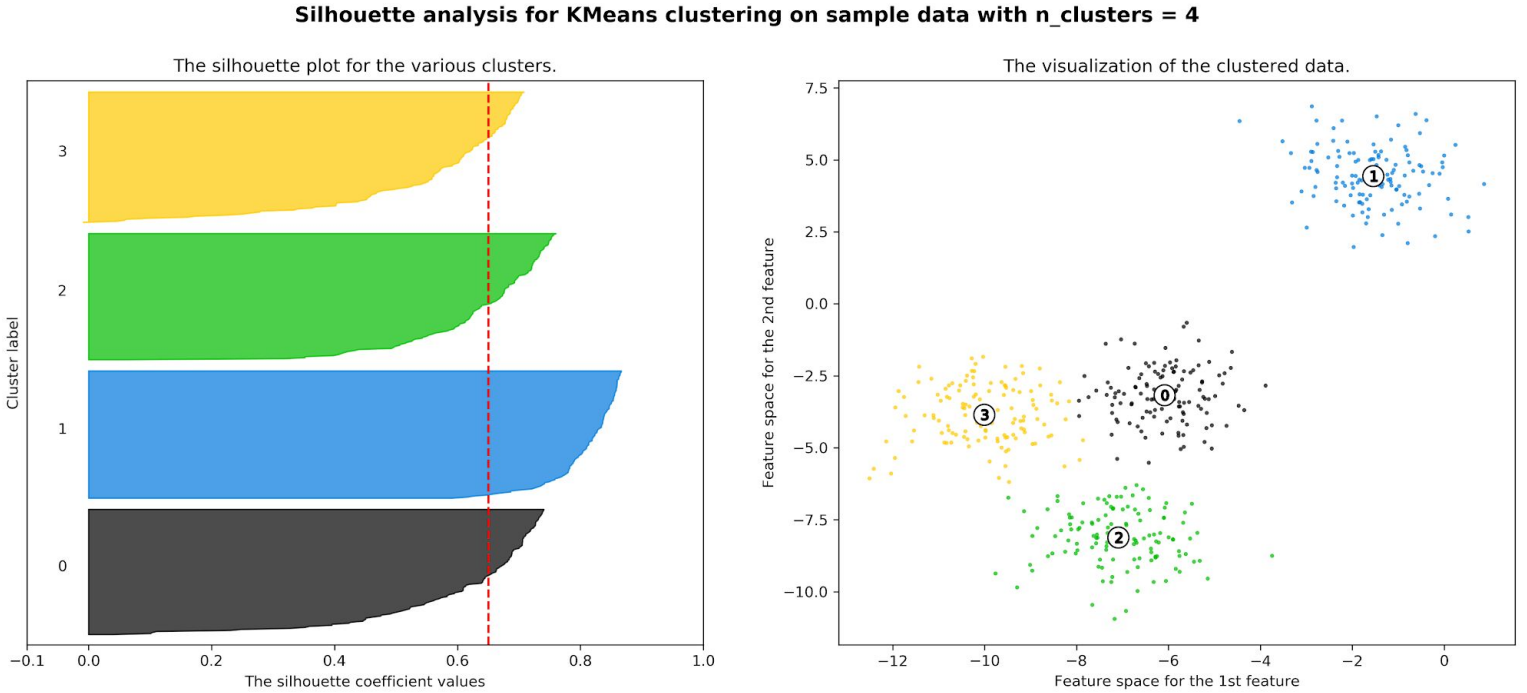
K-means


Silhouette analysis for KMeans clustering on sample data with n_clusters = 6
The silhouette plot for the various clusters.
Cluster label
The silhouette coefficient values
The visualization of the clustered data.
Feature space for the 2nd feature
Feature space for the 1st feature

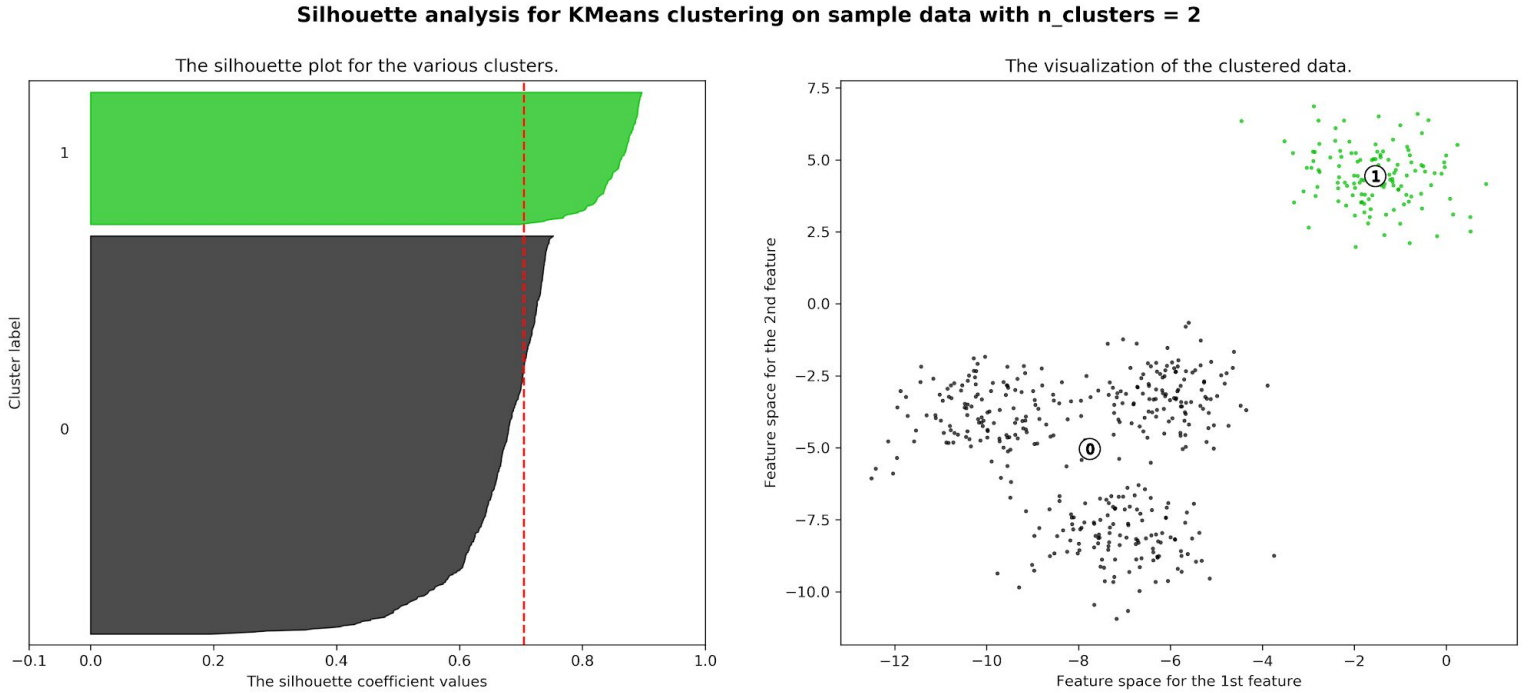
K-means: Ejemplo

- Estas características no ocurren el el caso de k=2 y 4
- Si se estudia el grosor de gráfico de silueta se ve que k=2 produce una partición muy desbalanceada, dado que uno de los clusters absorbe tres clusters diferentes.
- Cuando k=4 las siluetas están balanceadas, por lo cual este es el mejor k

K-means


Silhouette analysis for KMeans clustering on sample data with n_clusters = 4
The silhouette plot for the various clusters.
Cluster label
3
2
1
0
−0.1
0.0
0.2
0.4
0.6
0.8
1.0
The silhouette coefficient values
The visualization of the clustered data.
Feature space for the 2nd feature
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
−12
−10
−8
−6
−4
−2
0
Feature space for the 1st feature
1
0
3
2

K-means


Silhouette analysis for KMeans clustering on sample data with n_clusters = 2
The silhouette plot for the various clusters.
Cluster label
1
0
−0.1
0.0
0.2
0.4
0.6
0.8
1.0
The silhouette coefficient values
The visualization of the clustered data.
Feature space for the 2nd feature
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
−12
−10
−8
−6
−4
−2
0
Feature space for the 1st feature

Dbscan

- El algoritmo DBSCAN define clusters como áreas de alta densidad separadas por áreas de baja densidad, que pueden ser de cualquier forma y tamaño.¶
- El concepto central del algoritmo es la clasificación y detección de puntos de núcleo (core samples), que son los puntos ubicados en zonas de alta densidad.
- Un cluster es un conjuntos de puntos de núcleo cercanos unos con otros, junto a un conjunto de puntos no núcleo que estan cercanos a algun punto de nucleo.
- Hay dos parámetros en este algoritmo: min_samples y eps, donde min_samples son los datos mínimos requeridos en un entorno de radio eps, los cuales definen la noción de zona densa.

Dbscan

- El algoritmo empieza con una muestra aleatoria y encuentra todos los puntos en el entorno de radio eps. Si el número de puntos es mayor a min_number se etiqueta ese punto como un punto de núcleo, si no es un punto outlier.
- Todos los puntos del entorno se etiquetan como puntos no núcleo de cluster. Se realiza el mismo procedimiento para cada uno de ellos, cambiando a punto core su etiqueta y agregando nuevos puntos, o marcando outliers.
- Si no hay más puntos en un entorno eps de cada punto del cluster, se salta a otro punto aleatoriamente y se continúa hasta que todo punto es bien un punto de cluster o un outlier.¶

Dbscan

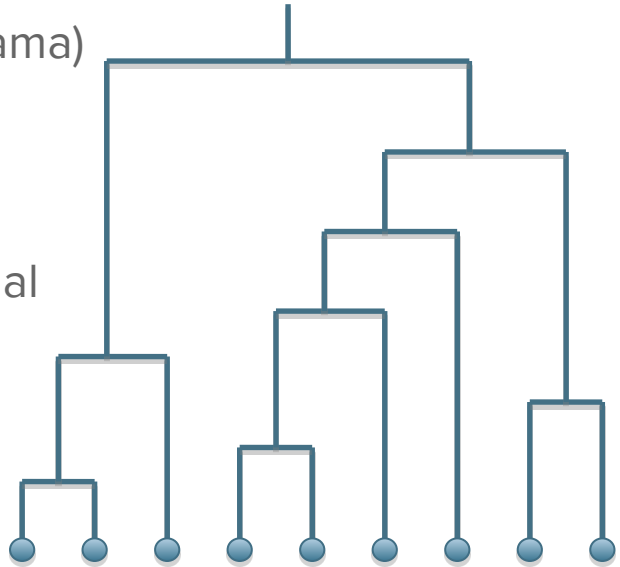
Clustering jerárquico

Si no queremos especificar k...

Algoritmos jerárquicos que generan una

taxonomía jerárquica de clusters (dendrograma)

- Interpretación más rica
- Más difícil de interpretar
- El corte del árbol tiene que ser ortogonal



Clustering jerárquico aglomerativo

Bottom-up

- Cada objeto es su propio cluster
- Se unen en un solo cluster el par de clusters más semejantes
- La historia de uniones forma un árbol binario (jerarquía)

Semejanza entre clusters

Single-link

1. Para cada par de clusters A y B, el par de objetos a, b más cercanos tal que a pertenece a A y b pertenece a B
2. Se unen los clusters con el par de objetos más semejante

Complete-link

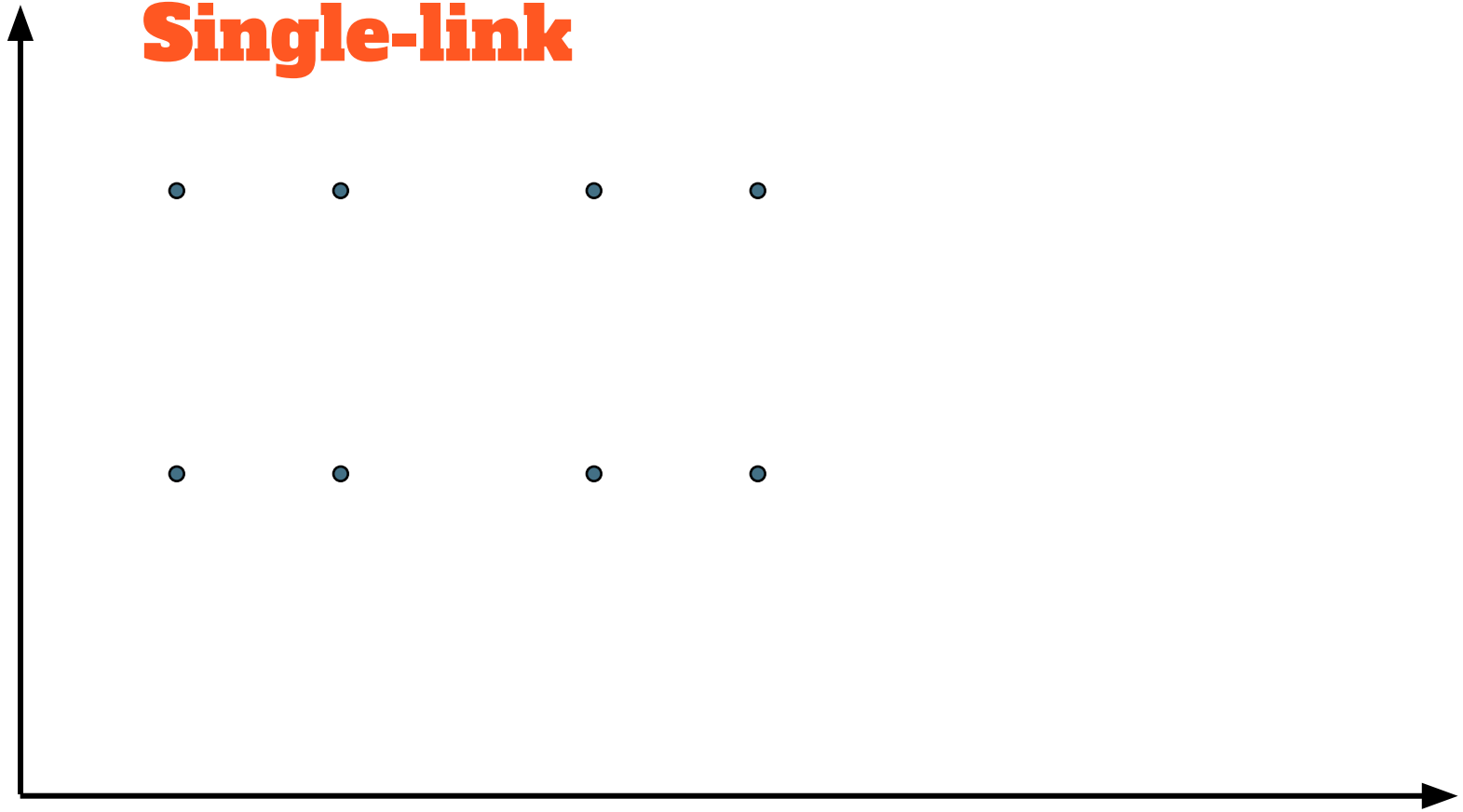
1. Para cada par de clusters A y B, el par de objetos a, b más distantes tal que a pertenece a A y b pertenece a B
2. Se unen los clusters con el par de objetos más semejante

Average-link

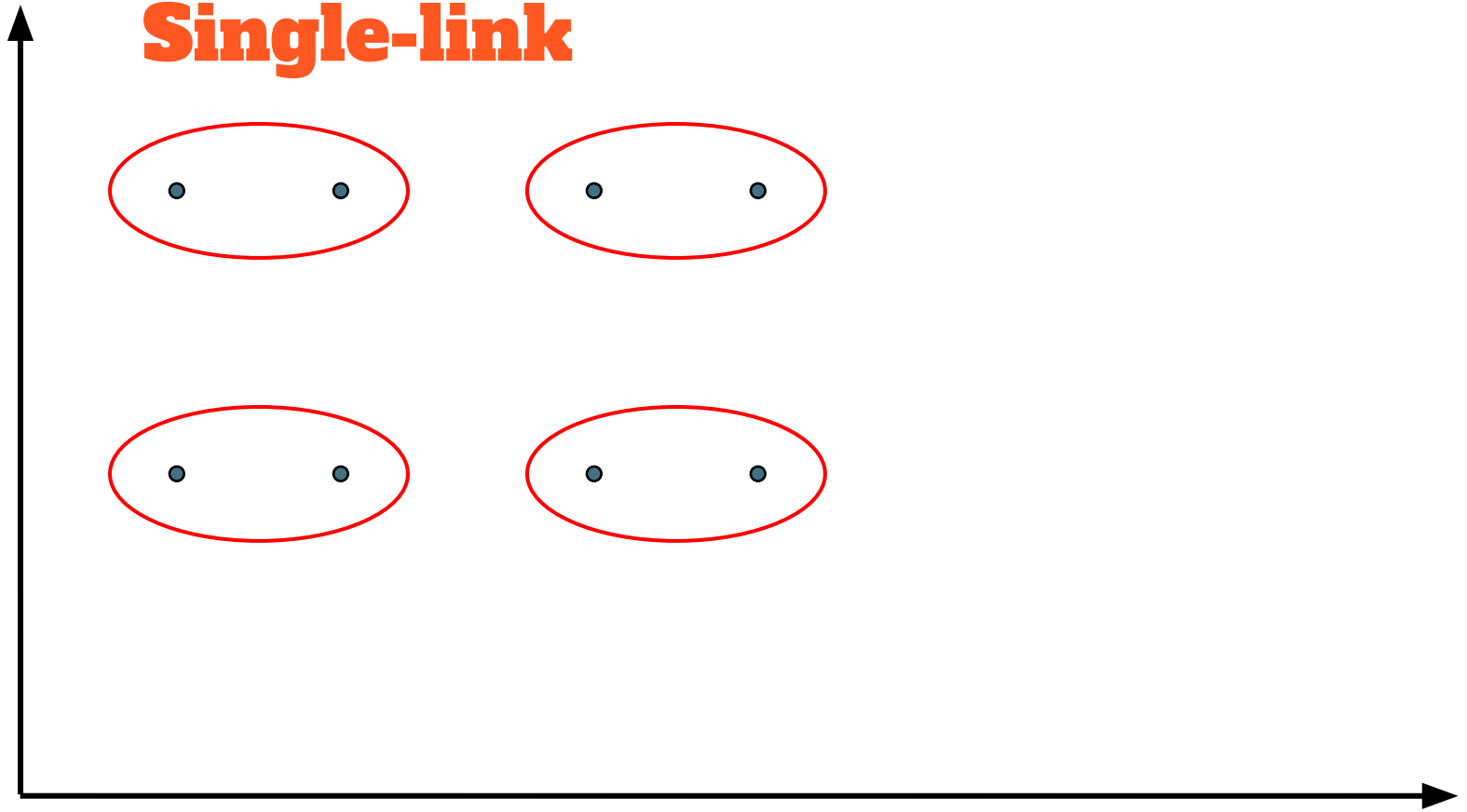
1. Para cada par de clusters A y B, se calcula la distancia entre todo par de objetos a, b tal que a pertenece a A y b pertenece a B
2. Se unen los clusters con el promedio de distancia más bajo

Centroid: Se unen los clusters con los centroides más cercanos

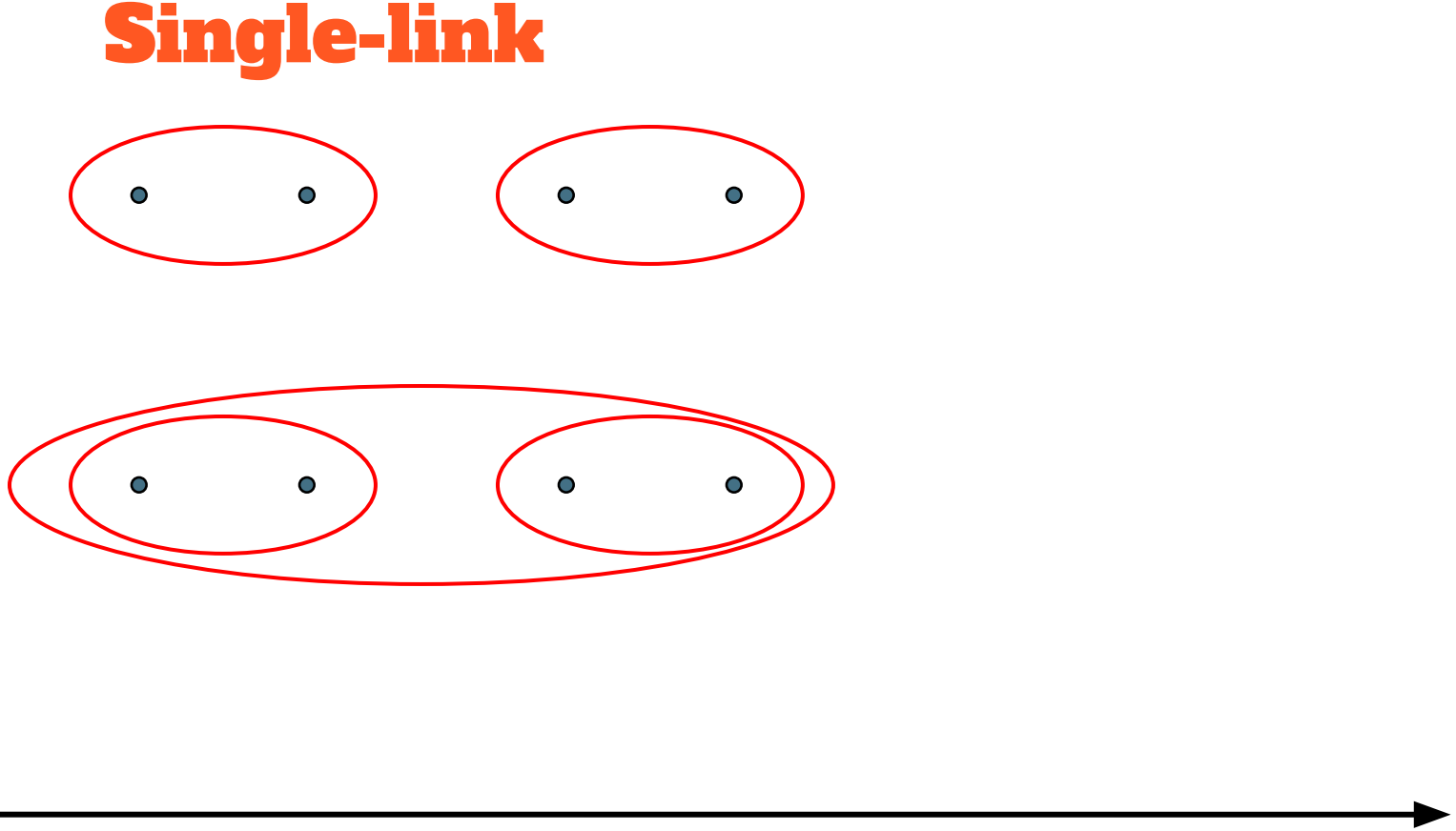
Single-link



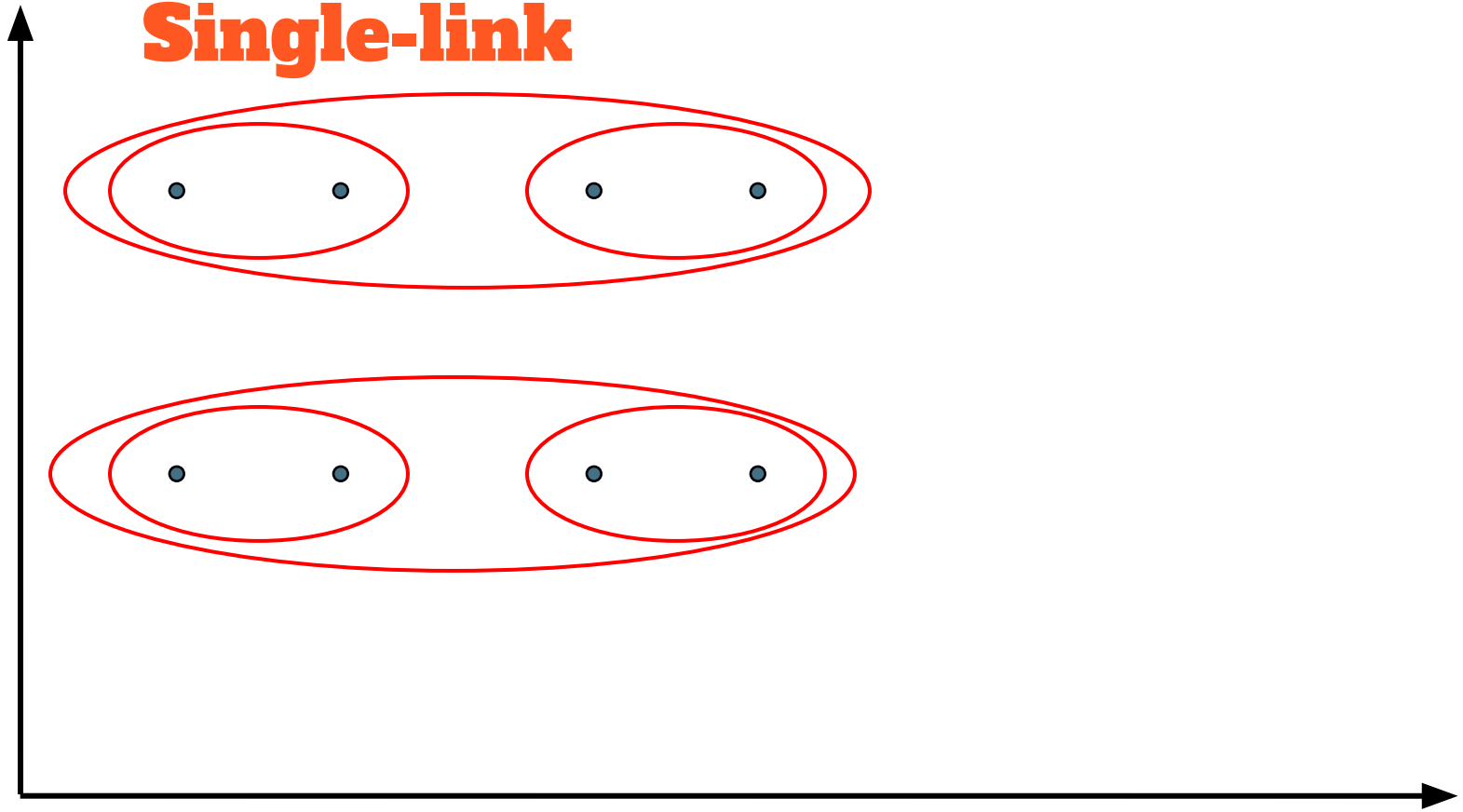
Single-link

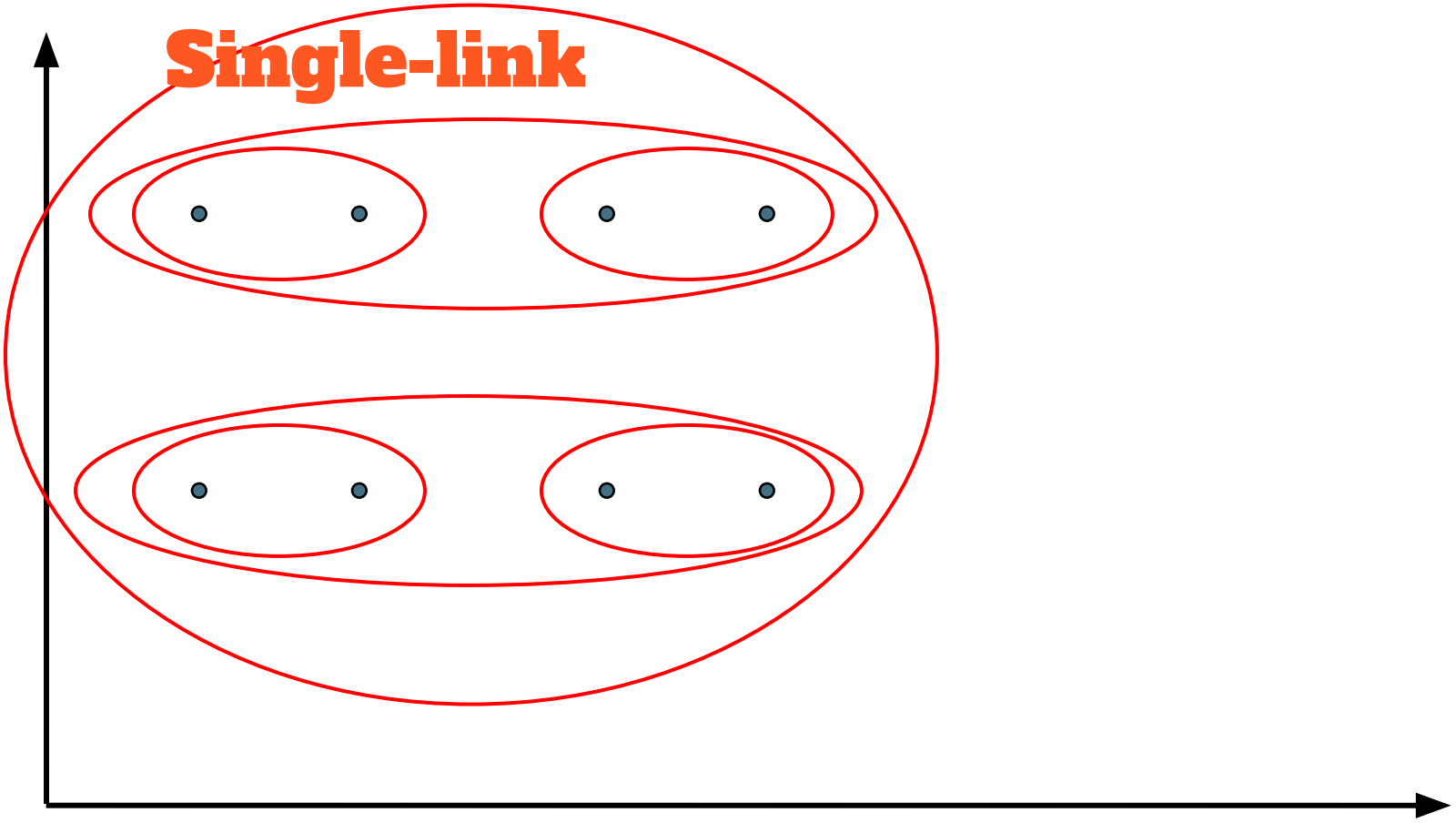


Single-link

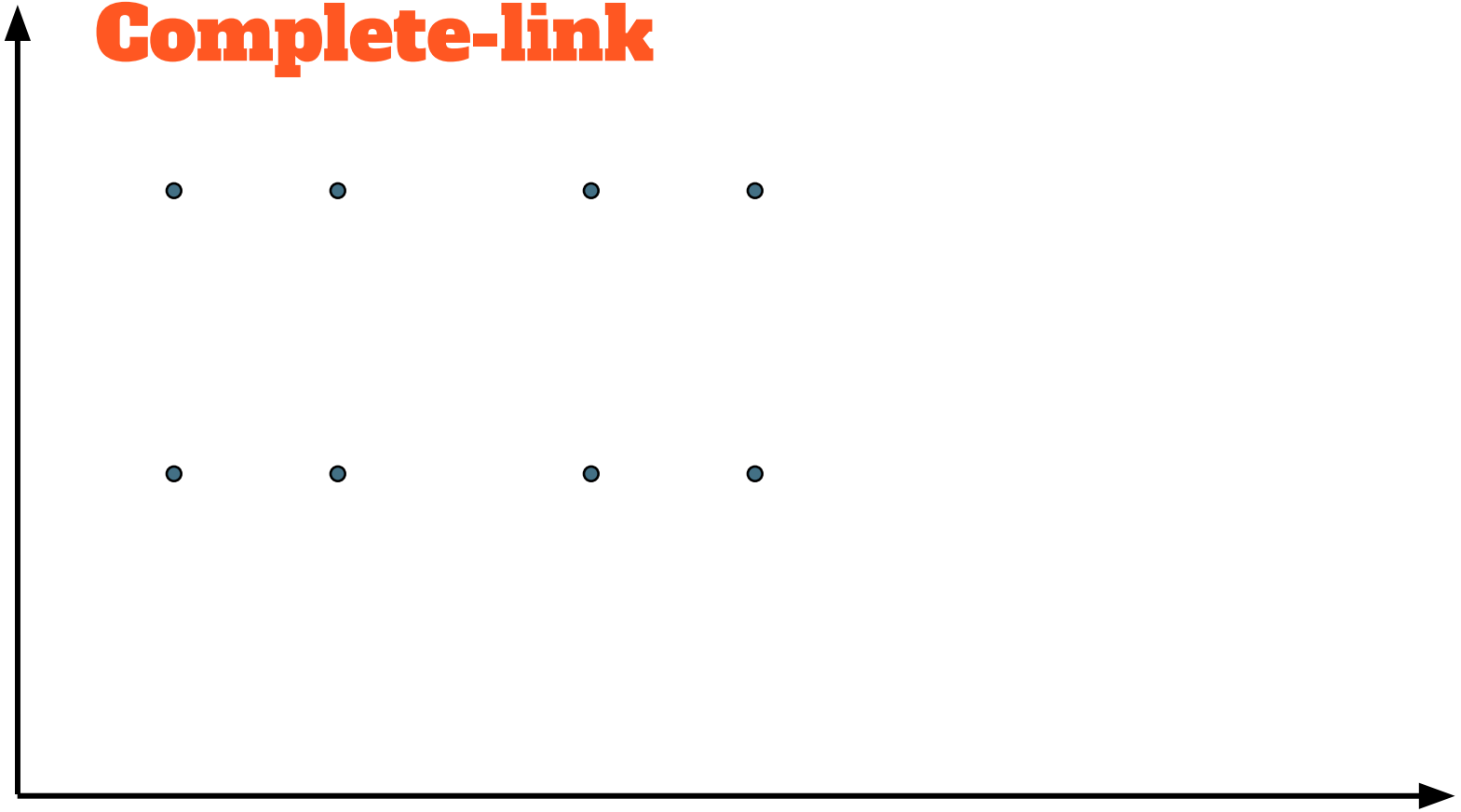


Single-link

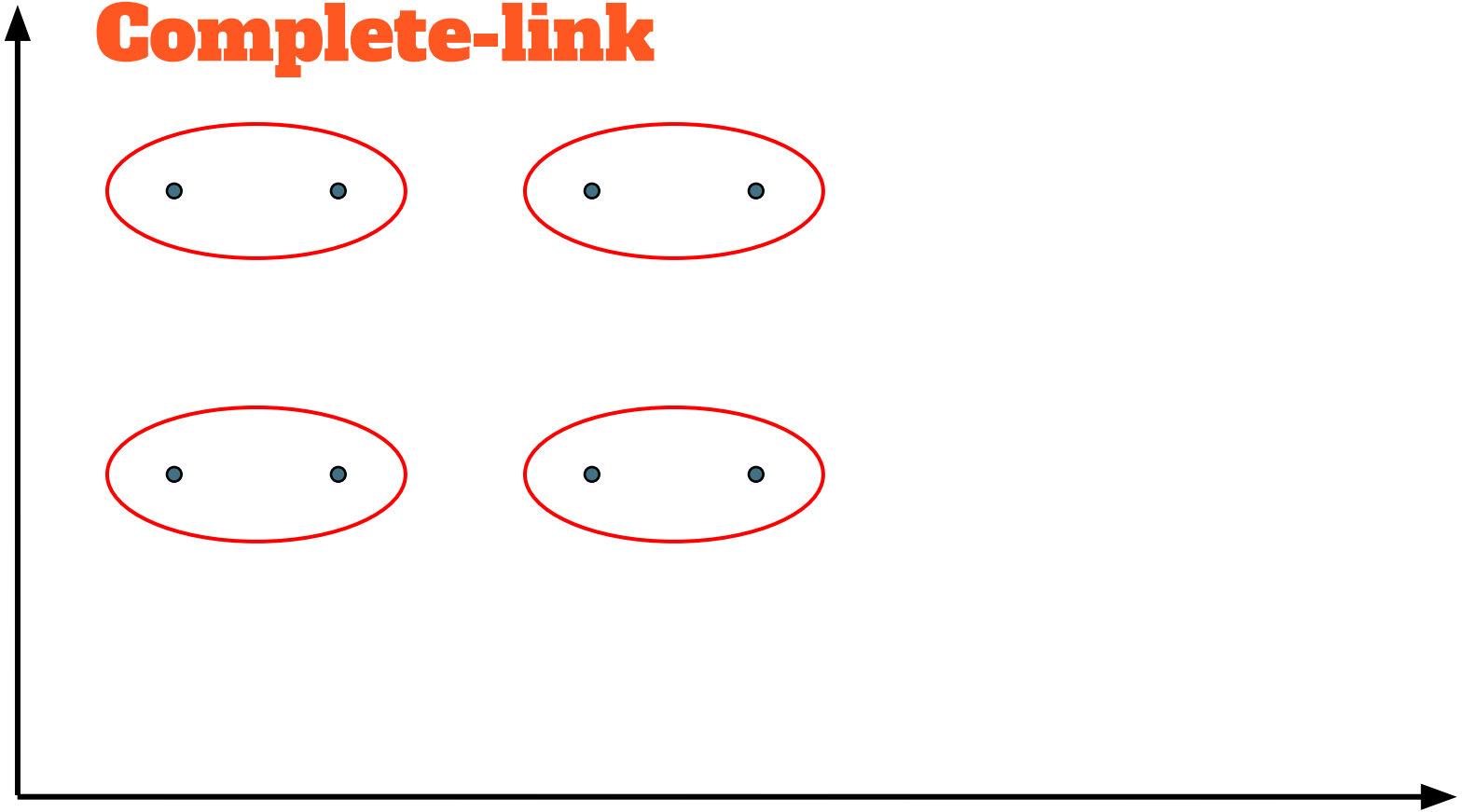



Single-link

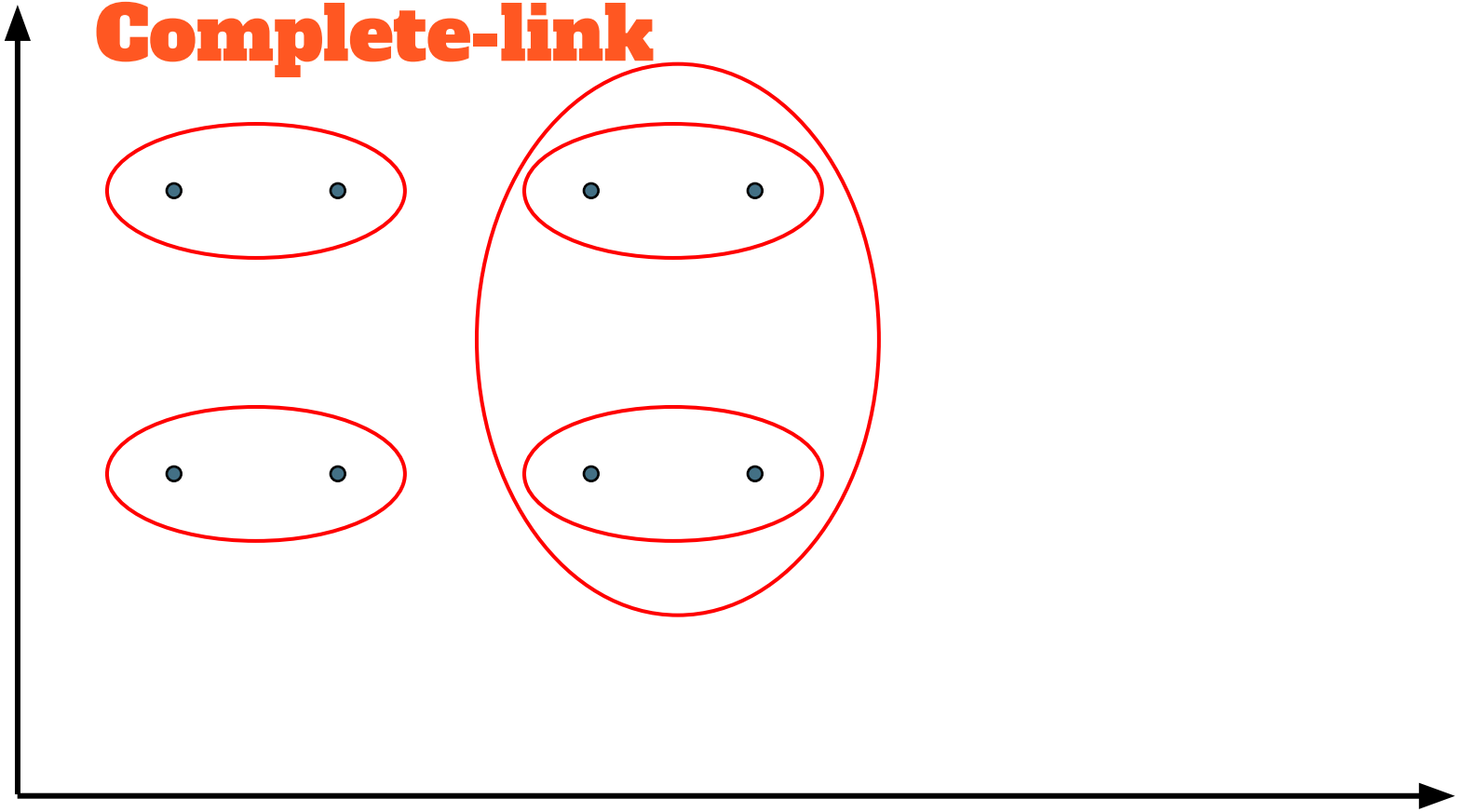
Complete-link



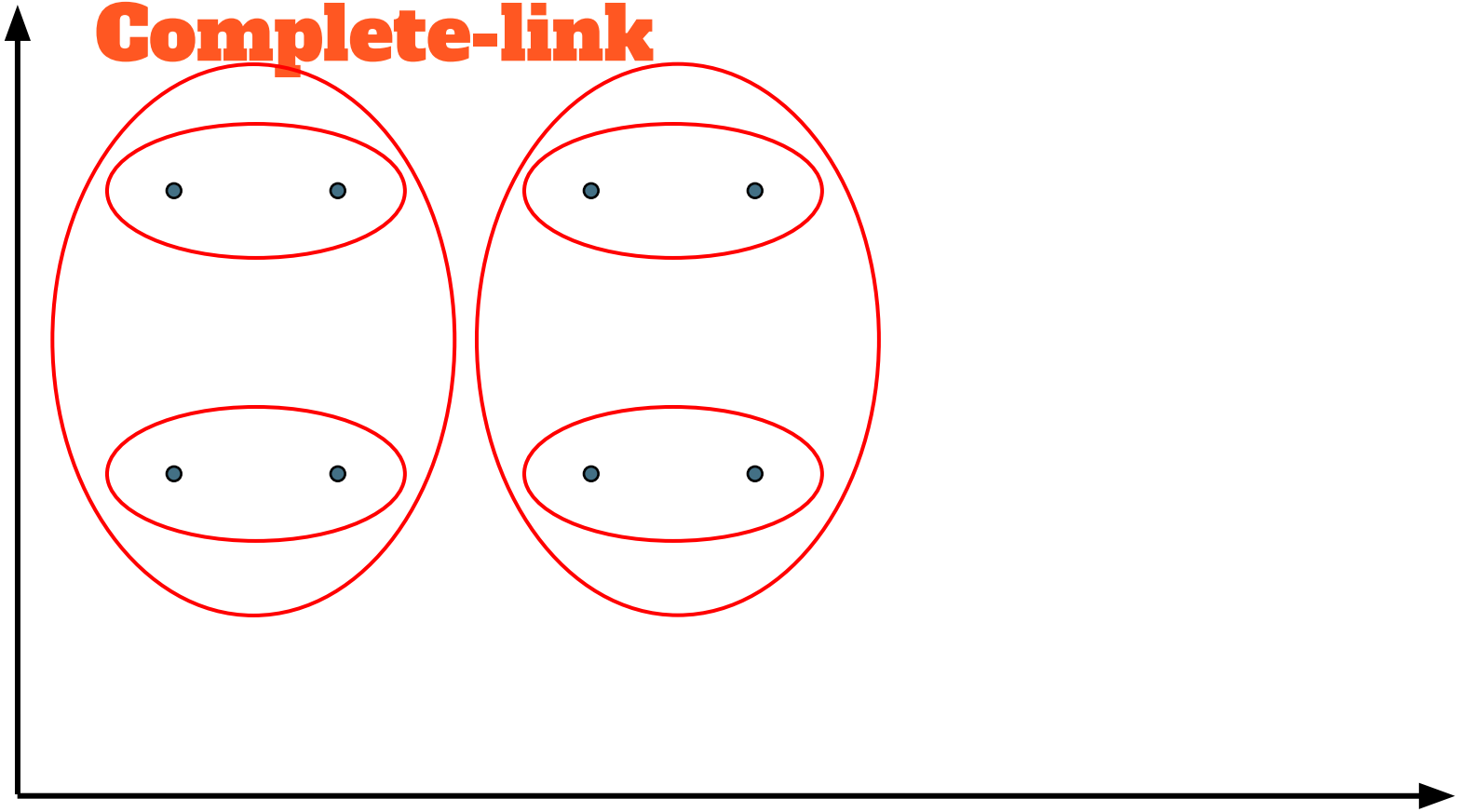
Complete-link

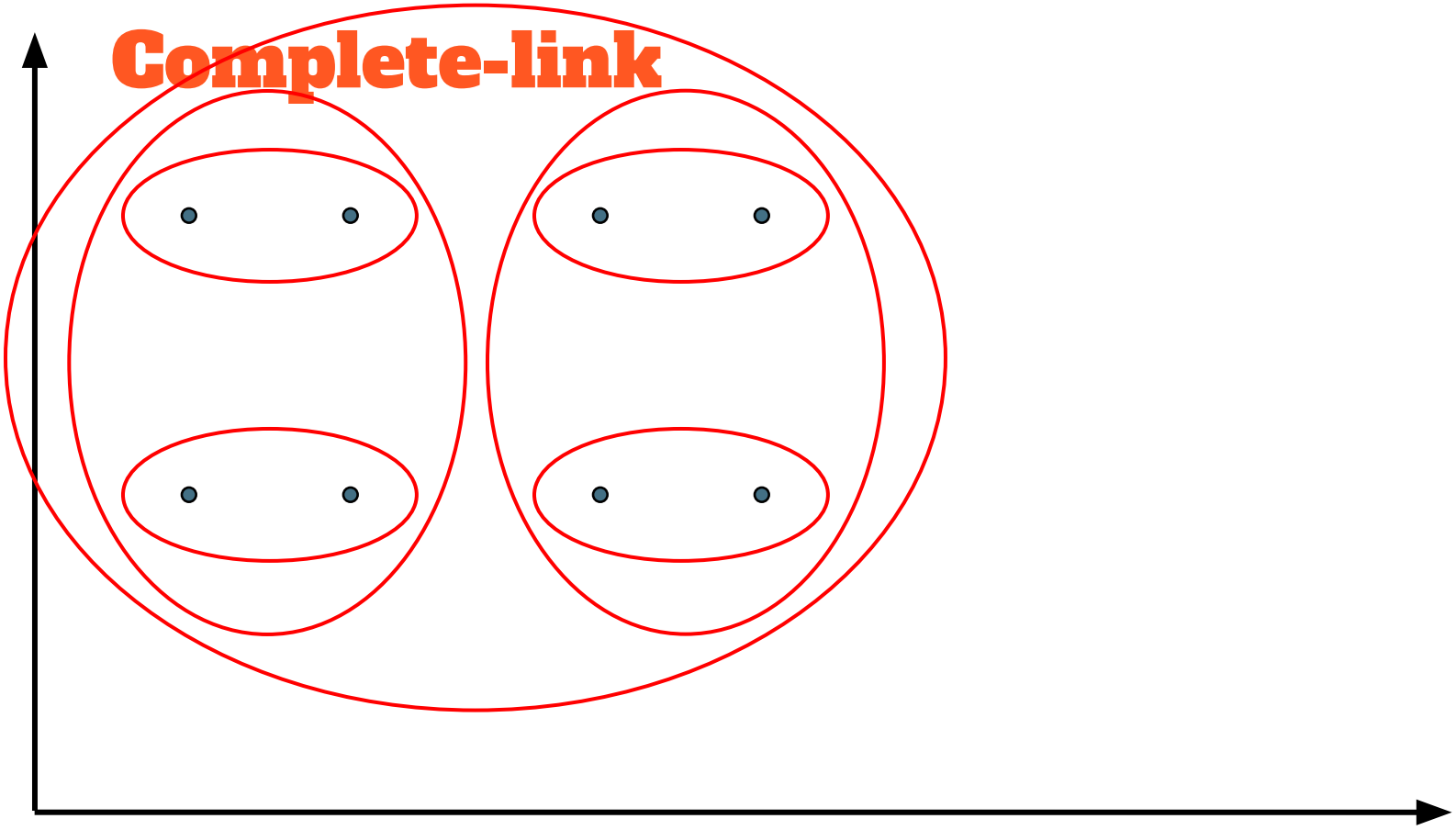


Complete-link



Complete-link




Complete-link

Clustering jerárquico partitivo

Top-down

Aplicar recursivamente un algoritmo de clustering partitivo

- Rápido
- Fácil de implementar
- Fácil de manipular
- Voraz (mínimos locales)

Evaluación

Un experto de dominio **interpreta** los clusters y encuentra información valiosa

¿Cómo mostrar el contenido de los clusters?

- Centroides (medoides)
- Resumen de características
- Características más distintivas de cada cluster
- Aplicar un algoritmo de aprendizaje automático interpretable (Decision Tree)

Evaluación intrínseca

Coeficiente Silhouette

Mide la semejanza de cada objeto al cluster al que se asigna (cohesión), comparada con otros clusters (separación).

Si el valor es bajo o negativo, el número de clusters puede ser inadecuado

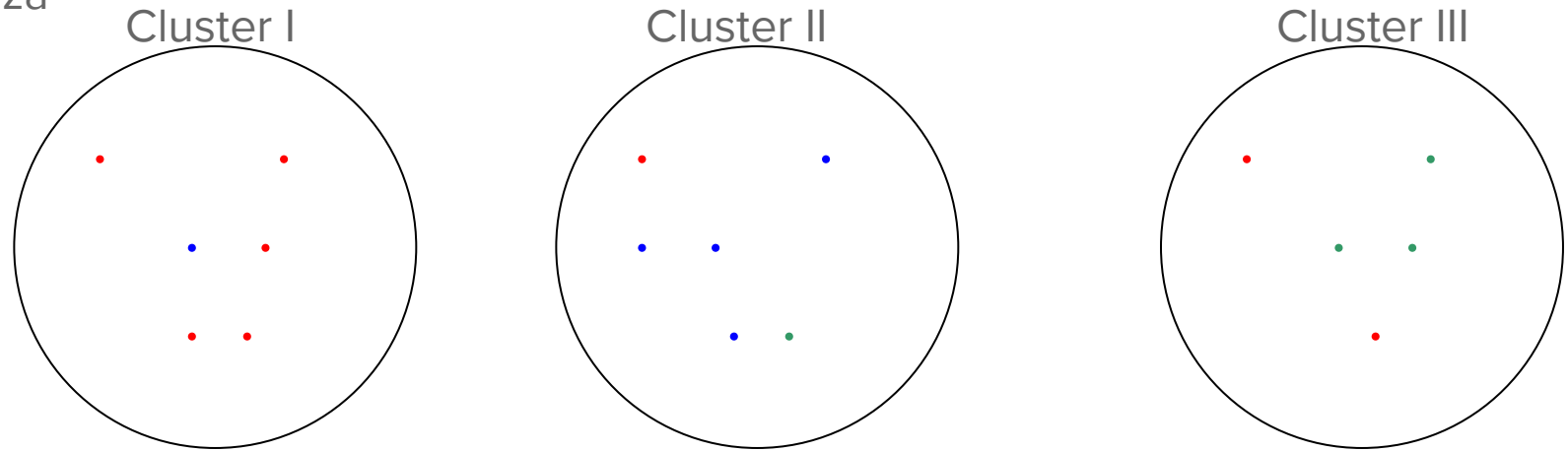
Evaluación con clases

Si los objetos tienen alguna etiqueta, observamos su distribución en los clusters

- Homogeneidad: cada cluster contiene sólo miembros de una clase
- Completitud: todos los miembros de una clase están en el mismo cluster
- V-measure: media harmónica de los anteriores
- Adjusted Rand index: semejanza entre las etiquetas originales y las asignadas
- Información Mútua entre etiquetas originales y asignadas
- Matriz de confusión!

Evaluación con clases

Pureza



Cluster I: Pureza = 1/6 (max(5, 1, 0)) = 5/6

Cluster II: Pureza = 1/6 (max(1, 4, 1)) = 4/6

Cluster III: Pureza = 1/5 (max(2, 0, 3)) = 3/5

Evaluación con testigos

1. Se seleccionan aleatoriamente pares de objetos del dataset
2. Un experto del dominio decide si tienen que estar en el mismo cluster o en diferentes clusters
3. Observamos el grado de acuerdo entre cada solución y los testigos

4. Se seleccionan aleatoriamente objetos del dataset
5. Se los etiqueta
6. Se observa cómo se distribuyen en el dataset

Indicadores de malas soluciones

En general, las malas soluciones se deben a malas características

- Una clase muy grande y el resto mucho más chicas ➔ la mayoría de objetos son no diferenciables con esas características o distancia
- Clases con uno o pocos elementos ➔ el número de clases es demasiado grande para el dataset
- Clusters con las mismas características, poco distinguibles
- Soluciones muy diferentes con diferentes inicializaciones, número de clusters

Clustering no es clasificación

No vamos a obtener clases bien diferenciadas, sino más bien mucho ruido

Es fuertemente sensible a las características de los objetos, a los parámetros, a los outliers

La mayor parte de aproximaciones son muy inestables

La primera aproximación suele ser inservible, hay que refinar características e iterar

Aplicaciones

- Segmentación de clientes, usuarios... para marketing personalizado
- Encontrar temas ➜ topic detection
- Imágenes de los mismos objetos ➜ gatitos, tumores (imágenes médicas), tormentas (imágenes satelitales), plagas (imágenes de cultivos)
- Agrupamiento de productos
- Detección de anomalías
- Taxonomías de plantas y otros organismos
- Detección de clases con significados semejantes