Introducción a Machine Learning

Sesión 2.2: Aprendizaje No Supervisado

Ronald Cárdenas Acosta

Agosto, 2016

Aprendizaje No Supervisado

- Data de entrenamiento: $x^1, x^2, ..., x^N$
- Objetivo: encontrar agrupaciones o estructuras abstractas en la data
- Forma probabilística: $p(x|parametro)$
- Aplicaciones
  - Clustering
  - Aprendizaje de Hiperplanos (Manifold Learning)
  - Descomposición de señales
  - Reducción de dimensionalidad
  - Detección de outliers
  - entre otros

Outline

Tipos de Clustering

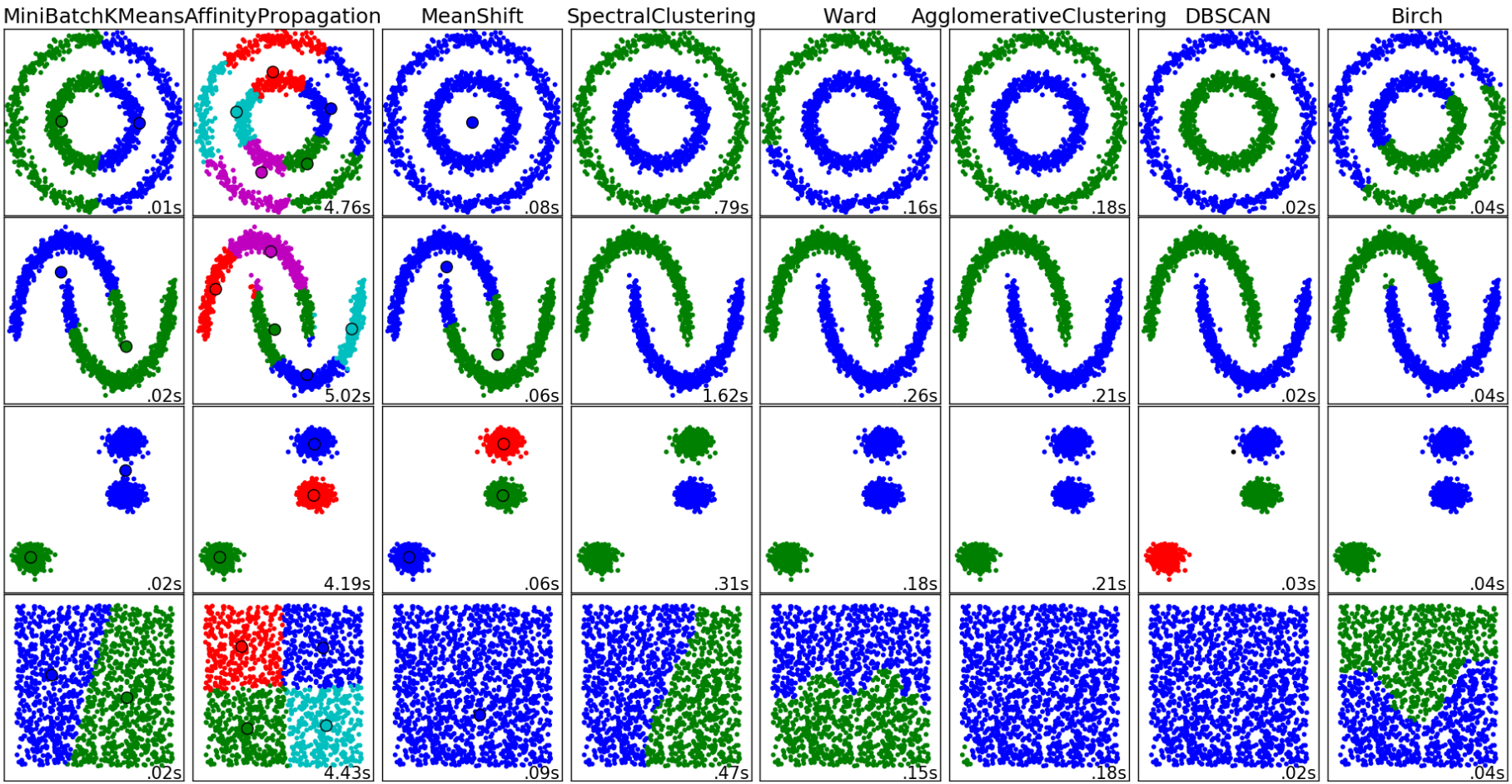

Figure: Tipos de clustering

Outline

Métricas de evaluación: [Adjusted] Rand Index

Sea C la asignación conocida de grupos y K la del clustering

$$RI = \frac{a + b}{C_2^N}$$

Donde:

- a: numero de pares que estan en el mismo cluster en C y en K
- b: numero de pares que estan en diferentes clusters en C y en K

Normalizando por chance:

$$ARI = \frac{RI - E[RI]}{max(RI) - E[RI]}$$

Métricas basadas en Información Mutua

Miden la similitud entre los dos grupos de asignaciones de cluster (real y estimado) mediante Información Mutua.

$$MI(C,K) = \sum_{i=1}^{|C|}\sum_{j=1}^{|K|} P(i,j) log(\frac{P(i,j)}{P(i).P(j)})$$

Donde:

- $P(i) = |C|/N$
- $P(j) = |K|/N$
- $P(i,j) = |C \cap K|/N$

Métricas basadas en pertenencia

- Homogeneidad: grado en el que cada cluster contiene solo miembros de una clase

$$h = 1 - \frac{H(C|K)}{HC)}$$

Donde:

  - $H(C|K) = -\sum_{c=1}^{|C|}\sum_{k=1}^{|K|}\frac{n_{c,k}}{N} log(\frac{n_{c,k}}{N})$
  - $H(C) = -\sum_{c=1}^{|C|}\frac{n_c}{N} log(\frac{n_c}{N})$

- Completividad: grado en el que todos los miembros de una clase son asignados a un mismo cluster

$$c = 1 - \frac{H(K|C)}{H(K)} \quad (1)$$

- V-measure: media armónica entre h y c:

$$v = 2\frac{h.c}{h+c} \quad (2)$$

Outline

Algoritmo KMeans

- Separa la data en K grupos disjuntos de igual varianza
- Minimiza criterio de *Inercia* o *Suma de Cuadrados dentro del cluster*
- Cada cluster esta descrito por su centroide μ_j, de la forma:

$$\sum_{i=0}^{N} \min_{\mu_j \in C} (\|x_i - \mu_j)$$

- La inercia asume que los clusters son convexos e isotrópicos, lo cual no siempre es el caso

Algoritmo KMeans: Pseudo-codigo

- Incializar aleatoriamente los K centroides $\mu_1, \mu_2, ...\mu_K \in \Re^M$
- Iterar por *num iteraciones*
  - for $i = 1 -> N$
    c^i = index (de 1 a K) del centroide mas cercano a x^i
  - for $k = 1 -> K$
    μ_k = promedio de puntos asignados a cluster k