K-MEANS CLUSTERING

Tags

Outline

Partitioning a database D of n objects into a set of **k** clusters

How does it works?

- 각 그룹의 중심과 그룹 내의 데이터 오브젝트와의 거리의 제곱합을 Cost function
- Minimize cost function → 각 데이터 오브젝트의 소속 그룹 업데이트

$$arg\ min\ \sum_{i=1}^{k} \sum_{x \in S_i} \parallel x - \mu_i \parallel^2$$

- 각 집합별 중심점~집합 내 오브젝트간 거리의 제곱합을 최소로 하는 집합 **S** 를 찾는 것

Algorithm

1. 초기 μ_i 설정

- 무작위 분할 - 각 데이터들을 임의의 클러스터에 배당한 후, 각 클러스터에 배당된 점들의 평균 값을 초기 μ_i 로 설정한다
- Forgy - 데이터 집합으로부터 임의의 k 개의 데이터를 선택하여 각 클러스터의 초기 μ_i 로 설정한다
- MacQueen - 데이터 집합으로부터 임의의 k개의 데이터를 선택하여 각 클러스터의 초기 μ_i로 설정한다. 이후 선택되지 않은 각 데이터들에 대해, 해당 점으로부터 가장 가까운 클러스터를 찾아 데이터를 배당한다. 모든 데이터들이 클러스터에 배당되고 나면 각 클러스터의 무게중심을 다시 계산하여 초기 μ_i로 다시 설정한다.
- Kaufman - 전체 데이터 집합 중 가장 중심에 위치한 데이터를 첫번째 μ_i로 설정한다. 이후 선택되지 않은 각 데이터들에 대해, 가장 가까운 무게중심 보다 선택되지 않은 데이터 집합에 더 근접하게 위치한 데이터를 또 다른 μ_i로 설정하는 것을 총 k개의 μ_i가 설정될 때까지 반복한다.

2. 클러스터 설정

- 데이터와 μ_i 사이의 거리를 계산하여 가까운 클러스터에 설정

$$S_i^{(t)} = \{x_p \ : \ |x_p - \mu_i^{(t)}|^2 \leq |x_p - \mu_j^{(t)}|^2, 1 \leq j \leq k\}$$

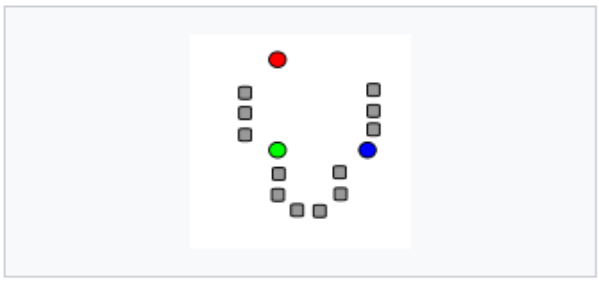
3. 클러스터 중심 재조정

- 각 클러스터에 있는 데이터들의 무게중심 값 재설정

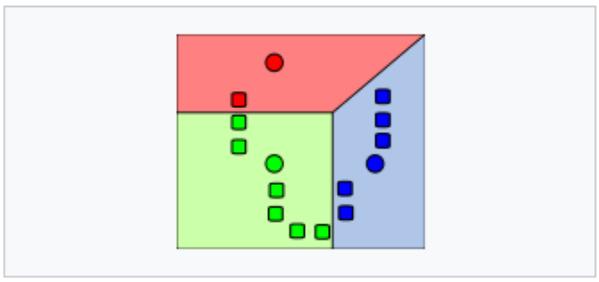
$$\mu_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{x_j \in S_i^{(t)}} x_j$$

4. 반복 2, 3 클러스터가 바뀌는 것이 없을 때까지

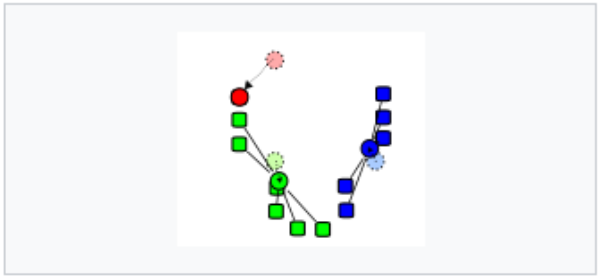
e.g.



1) 초기 *k* "평균값" (위의 경우 *k*=3) 은 데이터 오브젝트 중에서 무작위로 뽑힌다. (색칠된 동그라미로 표시됨).



2) *k* 각 데이터 오브젝트들은 가장 가까이 있는 평균값을 기준으로 묶인다. 평균값을 기준으로 분할된 영역은 보로노이 다이어그램으로 표시된다..



3) *k*개의 클러스터의 중심점을 기준으로 평균값이 재조정된다.

4) 수렴할 때까지 2), 3) 과정을 반복한다.

Disadvantages

- k값을 입력 파라미터로 지정
  - k값에 따라 결과 값이 다르다
- 이상값에 대하여 민감하다
- 구형이 아닌 클러스터를 찾는 데에 적절하지 않다.