Density Based Clustering

이동빈

Index

1. Why - Density Based Clustering
2. DBSCAN
3. DBSCAN 의 장단점

KU-BIG

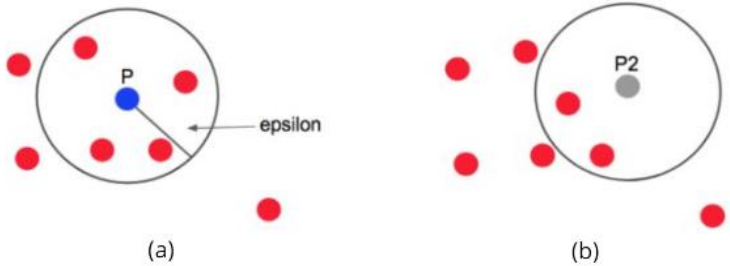
Why - Density Based Clustering

1. K-means Clustering
2. Hierarchical Clustering

→ Distance Based

3. Density Based Clustering

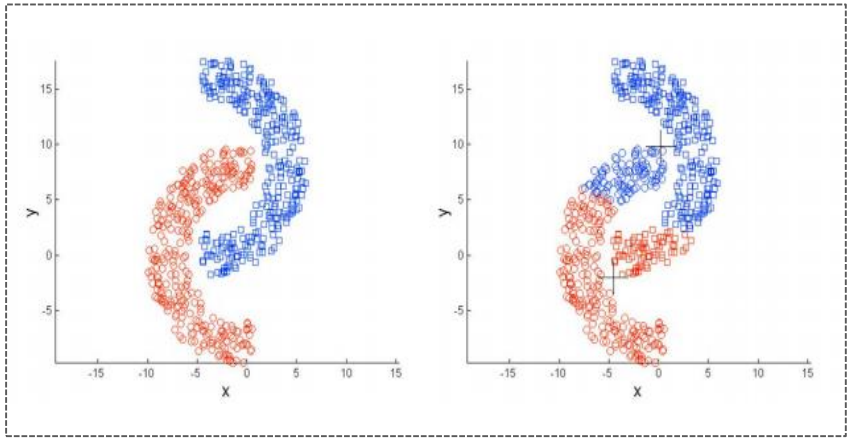
– 밀도(얼마나 빽빽한지) 를 기준으로 Clustering



KU-BIG

Why - Density Based Clustering

1. Distance Based Clustering 의 단점 보완



구 형태가 아닌 Clustering （ K = 2 ）

✚

〈 K-means Clustering 의 단점 보완 〉

Cluster의 수를 지정할 필요 X

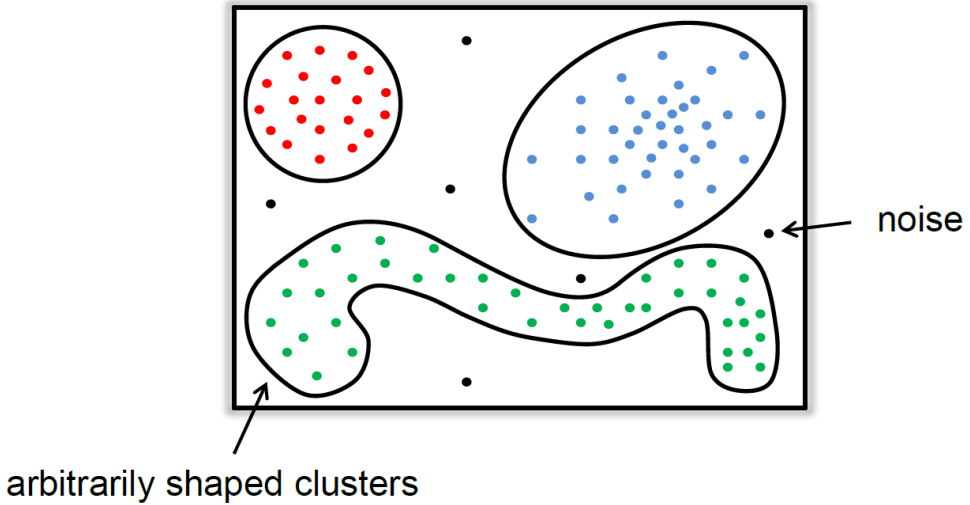
KU-BIG

Why - Density Based Clustering

2. Noise 데이터가 존재하는 경우

* Noise

: 어떤 Cluster 에도 속하지 않는 Data

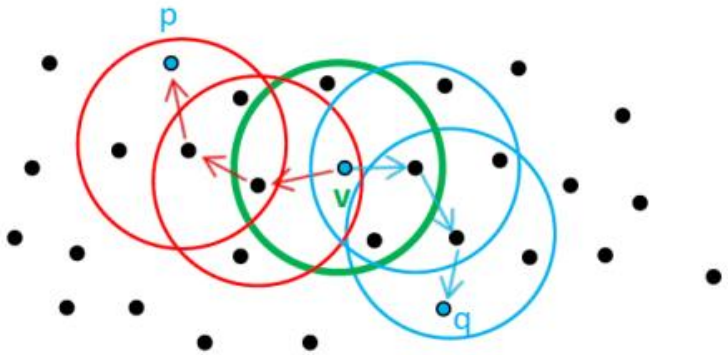


KU-BIG

DBSCAN

✓ DBSCAN (Density Based Spatial Clustering of Applications with Noise)

: 가장 대표적인 Density Based Clustering



– 밀도가 높은 (몰려 있는) 점들끼리 하나의 Cluster 를 이룬다

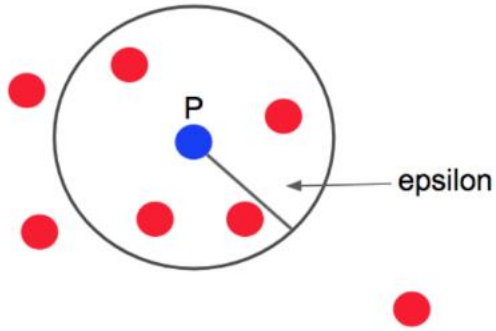
https://www.naftaliharris.com/blog/visualizing-dbscan-clustering/

KU-BIG

DBSCAN

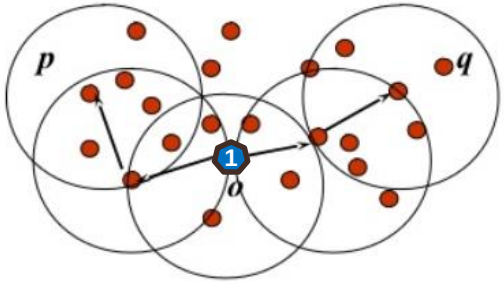
- 용어

1. Density : The number of points within a specified radius (Eps)
2. Core Point
   - **Epsilon** 반지름 내에 **MinPts** 이상의 점을 가지는 Point
3. Border Point
   - Core Point 의 radius 내부의 점
4. Noise
   - Core Point , Border Point 도 아닌 Point



DBSCAN

- DBSCAN 알고리즘



1. 모든 Point 들을 Core, Border Point로 분류
2. 모든 Noise Point 제거
3. 임의의 Core Point 에서 시작, Radius 내부의 모든 Core Point 들을 서로 연결, Cluster에 포함
4. 더 이상 연결될 Core Point가 없으면 새로운 Core Point 에서 3번을 반복
5. 포함된 모든 Core Point 들의 Border Point 를 포함

DBSCAN

- DBSCAN Parameter

1) Epsilon 2) MinPts

```
DBSCAN_multishapes <- dbscan(df_multishapes, eps = 0.15, minPts = 5)
```

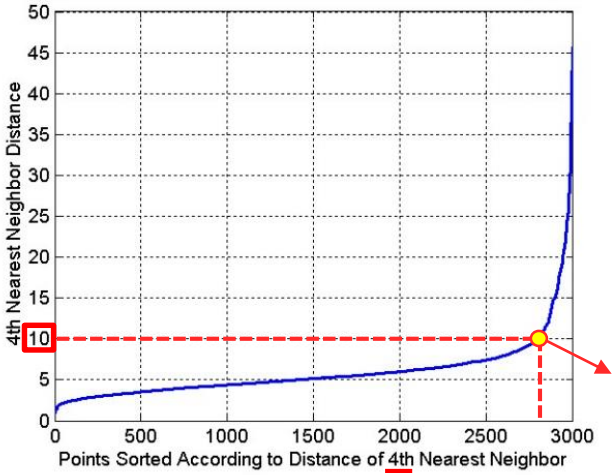
DBSCAN

- DBSCAN Parameter

  1) Epsilon 2) MinPts

```
DBSCAN_multishapes <- dbscan(df_multishapes, eps = 0.15, minPts = 5)
```

- K-th nearest neighbor



Elbow Point

MinPts : 4
Eps : 10

KU-BIG

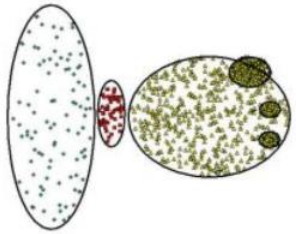
DBSCAN 의 장단점

▪ 장점

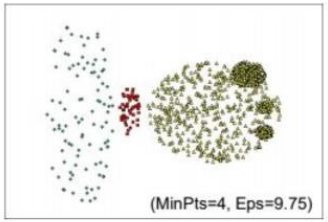
1. Cluster 의 개수를 지정할 필요 X
2. 구 형태가 아닌 다양한 형태,크기의 Cluster
3. Noise data 판별 가능
4. 데이터의 순서에 덜 Sensitive

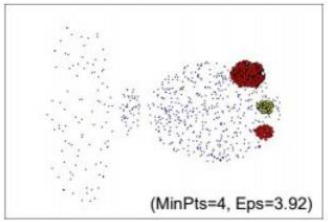
▪ 단점

1. 서로 다른 밀도를 가지는 Data에 제대로 작동 X



Original Points





2. High dimension data 에 적절하지 X

KU-BIG

EOD