BITS F464: Machine Learning

18 Clustering K-Means

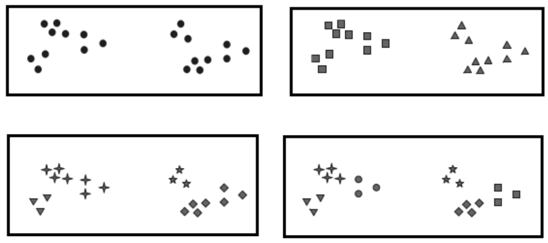
Dr. Kamlesh Tiwari
Assistant Professor, Department of CSIS,
BITS Pilani, Pilani Campus, Rajasthan-333031 INDIA

Feb 26, 2021 ONLINE (Campus @ BITS-Pilani Jan-May 2021)

http://ktiwari.in/ml

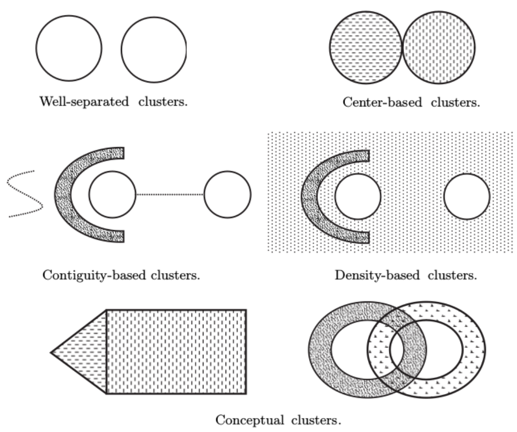
Clustering

Grouping data based on their homogeneity (similarity or closeness).



Objects within a group are similar (or related) and are different from the objects in other groups. When it is better?

Clustering Approaches



K-means Algorithm

Number of clusters *i.e.* the value of K is provided by the user

Algorithm 1: K-means

```
Randomly select K points as centroids
repeat
    foreach datum point d_i do
        Assign d_i to one of the closest centroids
        (thereby forming K clusters)
    Recompute centroid (mean) for each cluster
until The centroids converge;
```

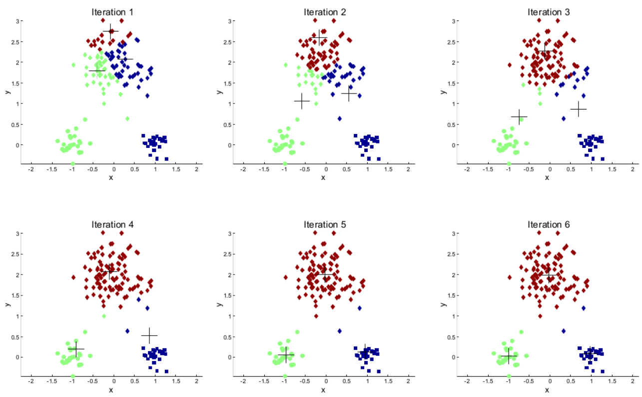
Closeness is measured by **Euclidean distance**, cosine similarity, correlation, Bregman divergence *etc*

Clustering

- **Unsupervised** in nature (i.e. right answers are not known)
- Clustering is useful to 1) Summarization, 2) Compression, and 3) Efficiently Finding Nearest Neighbors
- **Type:**
  - Hierarchical (nested) versus Partitional
  - Exclusive versus Overlapping versus Fuzzy
  - Complete versus Partial
- **K-means:** This is a prototype-based[1], partitional clustering technique that attempts to find a user-specified number of clusters (K), which are represented by their centroids.

[1] object is closer (more similar) to a prototype

K-means in Action



Evaluation of K-means[2]

For a given data set $\{x_1, x_2, ..., x_n\}$, let K-means partitions it in $\{S_1, S_2, ..., S_K\}$ then the objective is to minimize

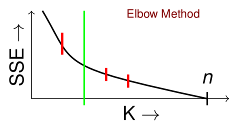
$$\underset{S}{\text{argmin}} \sum_{i=1}^{K} \sum_{x \in S_i} dist^2(x, \mu_i)$$

where μ_i corresponds to i^{th} centroid. $\mu_i = \frac{1}{|S_i|} \sum_{x \in S_i} x$

- Typical choice for *dist* function is Euclidean Distance

How to proceed?

- Choose a K (How?)
  - Run K-means algorithm multiple times
  - Choose clusters corresponding to the one that minimized sum of squared error (SSE)
- If $K == n$, no error.
- Good clustering has smaller K

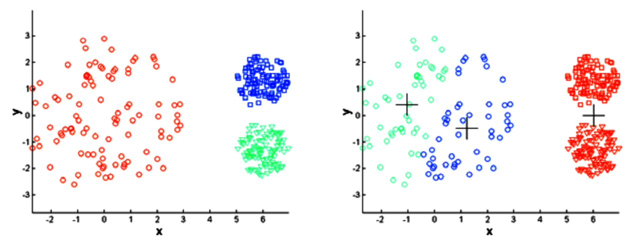


[2] Hamerly, Greg and Elkan, Charles, "Learning the k in k-means", pp 281–288, NIPS-2003

Evaluation of K-means

- **Choosing K:** 1) Domain Knowledge, 2) Preprocessing with another algorithm, 3) Iteration on K
- **Initialization of Centers:** 1) Random point in space, 2) Random point of data, 3) look for dense region, 4) Space uniformly in feature space, 5) K-Means++ (high probable if at far)
- **Cluster Quality:** 1) Diameter of cluster verses Inter-cluster distance, 2) Distance between members of a cluster and the cluster center, 3) Diameter of smallest sphere, 4) Ability to discover hidden patterns
- **Efficiently:** mini-batch K-Means

Limitations of K-means



- Has problem when data has
  - Different size clusters
  - Different densities
  - Non-globular shape
- Handling Empty Clusters
- When there are outliers
- Updating Centroids Incrementally

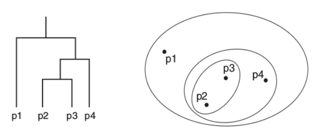
Important Note:

- K-Means and K-NN are different (K nearest neighbors)

K-NN is a **supervised** approach for **classification**

Other Approaches

- **Mini Batch K-Means** less computation and faster convergence
- **K-Medoids:** chooses data point as center and minimizes a sum of pairwise dissimilarities. Resistance to noise and/or outliers
- **Agglomerative Hierarchical Clustering:** repeatedly merging the two closest clusters until a single (Single Link)

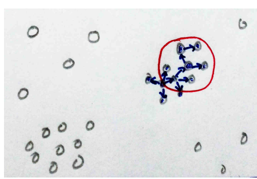


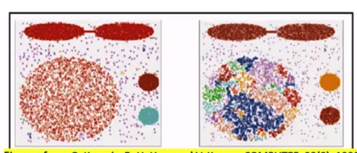
- **DBSCAN:** density-based clustering algorithm that produces a partitional clustering, in which the number of clusters is automatically determined by the algorithm.
- **More variations:** Affinity propagation, Mean Shift, Spectral Clustering, Ward hierarchical, Optics, Gaussian Mixture, Birch

DBSCAN

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a spatial clustering algorithm of KDD96

- Parameters (Eps/MinPts) and points (core/border/noise)
- Uses DFS

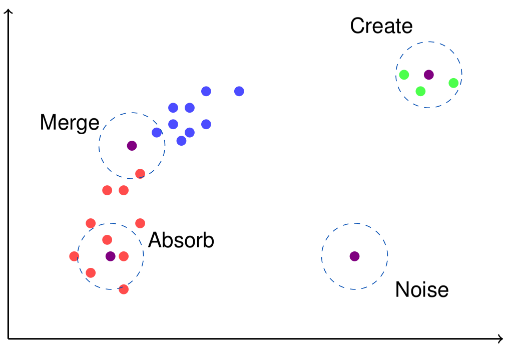




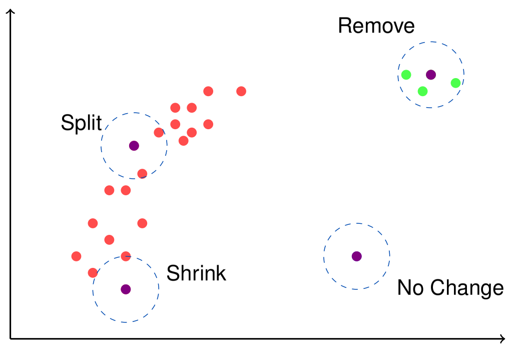
Figures from G. Karypis, E.-H. Han, and V. Kumar, *COMPUTER*, 32(8), 1999

- Disadvantage: Sensitive to parameters
- Advantage: 1) clusters of arbitrary shape, 2) Can handle dynamic databases

Incremental DBSCAN (Addition)



Incremental DBSCAN (Deletion)



Thank You!

Thank you very much for your attention! (Reference[3])

Queries ?

[3] [1] Book - *Machine Learning*, ch-3, Tom M. Mitchell. [2] Decision Tree 1: how it works https://www.youtube.com/watch?v=eKD5gxPPeY0, [2] An efficient k-means clustering algorithm: Analysis and implementation, T. Kanungo, D. M. Mount, N. Netanyahu, C. Piatko, R. Silverman, and A. Y. Wu, IEEE Transaction on Pattern Analysis and Machine Intelligence, pp 881–892, 24 (2002) [3] https://www-users.cs.umn.edu/∼kumar/dmbook/ch8.pdf