Unsupervised Learning

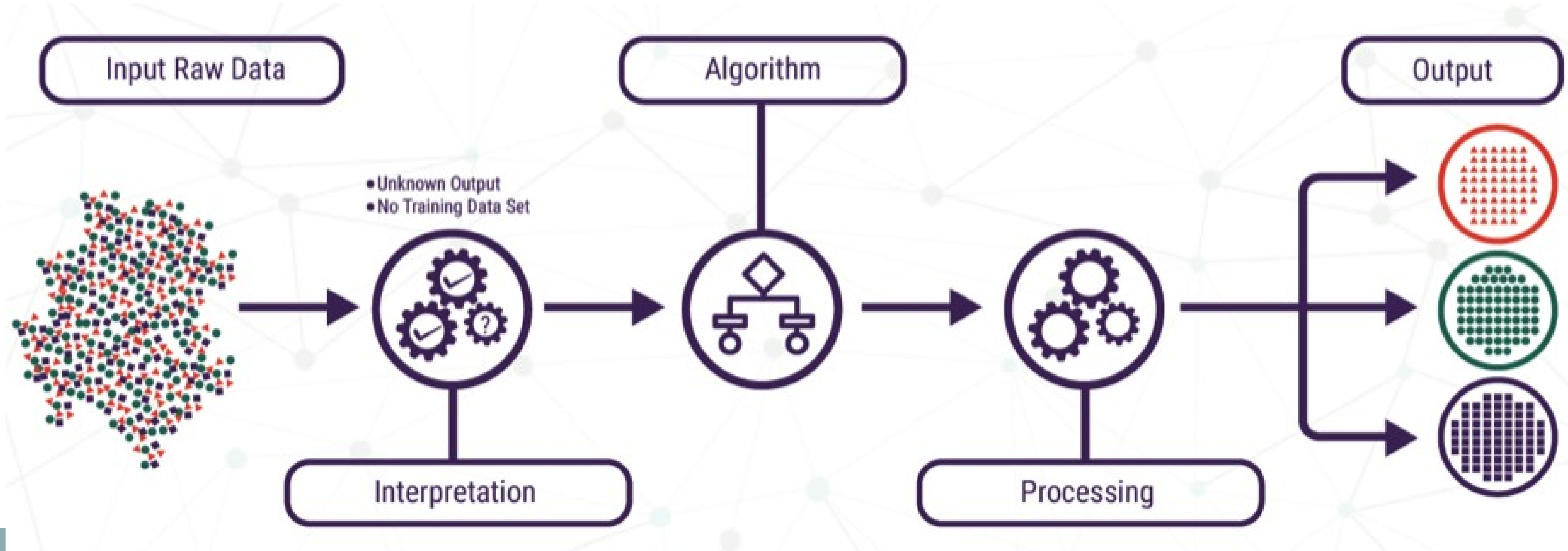
K-means Clustering

Faculty Trainer
Naw Varsha Pipada
Department of Computer Science & Engineering
Engineering College Bikaner

Contents

Quick Review

- Typically, unsupervised learning algorithms make inferences from datasets using only input vectors without referring to known, or labelled, outcomes.



Why Unsupervised Learning

- Issues
- Unsupervised Learning is harder as compared to Supervised Learning tasks..
- How do we know if results are meaningful since no answer labels are available?
- Let the expert look at the results (external evaluation)
- Define an objective function on clustering (internal evaluation)

- Still Needed
- Annotating large datasets is very costly and hence we can label only a few examples manually. Example: Speech Recognition
- There may be cases where we don't know how many/what classes is the data divided into. Example: Data Mining
- We may want to use clustering to gain some insight into the structure of the data before designing a classifier.

Unsupervised Learning Classification

Parametric Unsupervised Learning

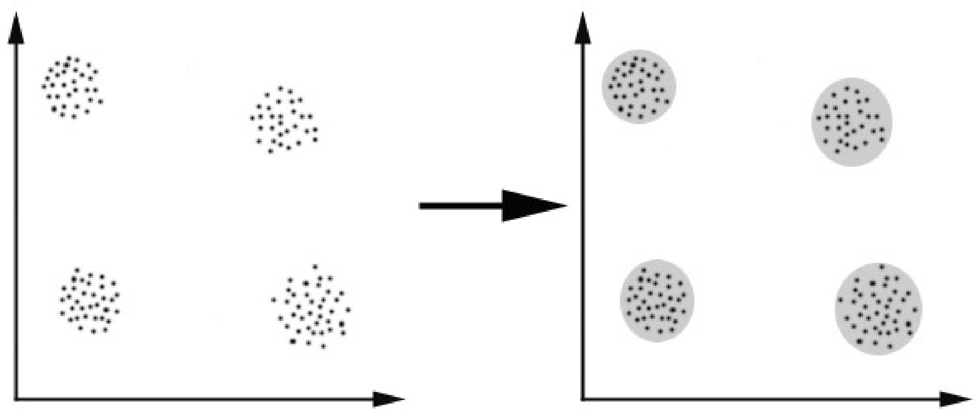
- It assumes that sample data comes from a population that follows a probability distribution based on a fixed set of parameters.
- Examples: Gaussian Mixture Models, Expectation-Maximization Algorithm, Probabilistic Clustering

Non-Parametric Unsupervised Learning

- Data is grouped into clusters, where each cluster(hopefully) says something about categories and classes present in the data
- Sometimes referred to as a distribution-free method
- Examples: K-means Clustering, Hierarchical Clustering, Association Rule Mining

Clustering

- It deals with finding a *structure* in a collection of unlabeled data.
- A loose definition of clustering could be “the process of organizing objects into groups whose members are similar in some way”.
- A *cluster* is therefore a collection of objects which are “similar” between them and are “dissimilar” to the objects belonging to other clusters.



Types of Clustering

- K-means Clustering
- Hierarchical Clustering
- Density Based Clustering
- Probabilistic Clustering

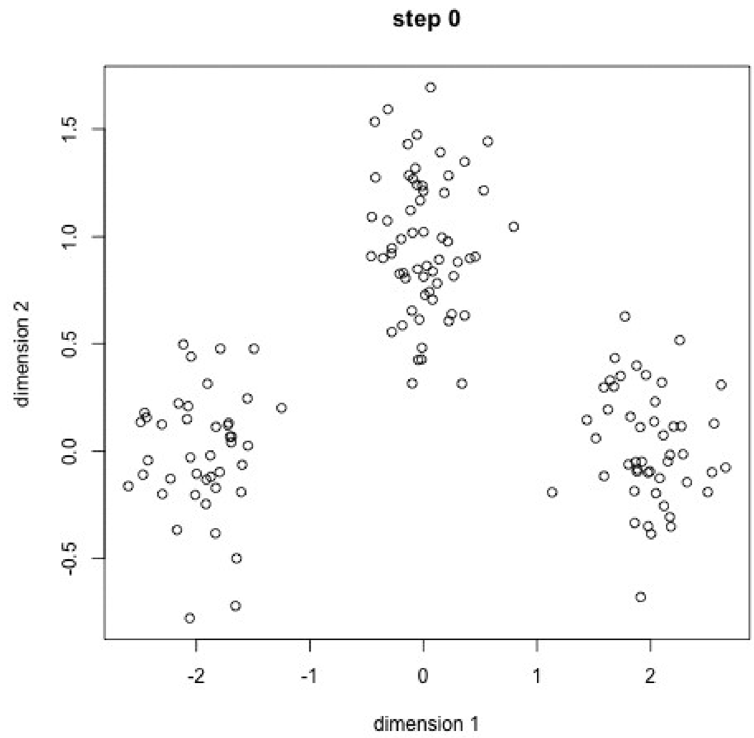
K-means Clustering

- The *'means'* in the K-means refers to averaging of the data; that is, finding the centroid.
- A centroid is the imaginary or real location representing the center of the cluster.
- *K* refers to the number of centroids you need in the dataset.

K-means Clustering – How it works

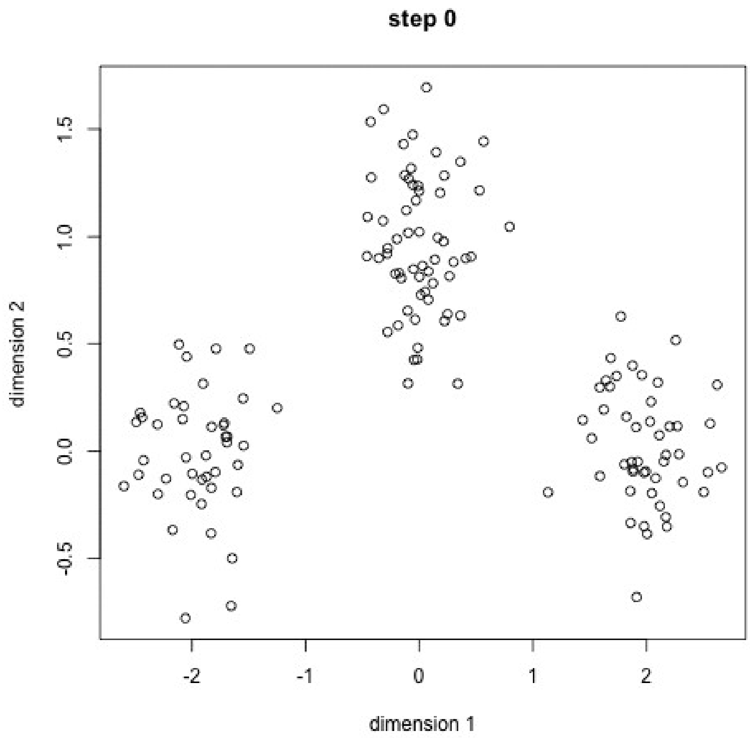
- Assume K=3, before starting to cluster.
- Algorithm:

1. Choose k (random) data points (seeds) to be the initial centroids
2. Assign each data point to the closest centroid, i.e., having minimum distance.



K-means Clustering - Illustration

4. Re-compute the centroids using the current cluster memberships, that is, by taking mean of all data points.
5. If convergence criteria is not met Go to step 2 else stop.



K-means Clustering: Convergence Criteria

- K-means performs iterative (repetitive) calculations to optimize the positions of the centroids
- It halts creating and optimizing clusters when either of the below criteria met:
  - The centroids have stabilized — there is no change in their values because the clustering has been successful.
  - The defined number of iterations has been achieved.

K-means Clustering: Convergence Criteria

- In other words, the iterations can be stopped if:

1. defined number of iterations has been achieved
2. no (or minimum) re-assignments of data points to different clusters, or
3. no (or minimum) change of centroids, or
4. minimum decrease in the sum of squared error

Distance (Dissimilarity) Measures

- Euclidean Distance

$$Dist_{xy} = \sqrt{\sum_{k=1}^{m} (x_{ik} - y_{ik})^2}$$

- Manhattan(City Block)

$$Dist_{xy} = \sum_{k=1}^{m} |x_{ik} - y_{ik}|$$

- Minkowski Distance

$$Dist_{xy} = \left(\sum_{k=1}^{d} |x_{ik} - x_{jk}|^{\frac{1}{p}} \right)^{p}$$

- Chebychev Distance

$$Dist_{xy} = max_k \; |x_{ik} - y_{ik}|$$

Example

- Cluster the following eight points (with (x, y) representing locations) into three clusters:
  - A1(2, 10), A2(2, 5), A3(8, 4), A4(5, 8), A5(7, 5), A6(6, 4), A7(1, 2), A8(4, 9)
- Initial cluster centers are: A1(2, 10), A4(5, 8) and A7(1, 2).
- The distance function between two points a = (x1, y1) and b = (x2, y2) is defined as-
  - $D(a, b) = |x2 - x1| + |y2 - y1|$
- Use K-Means Algorithm to find the three cluster centers after the second iteration.

Calculating Distance Between A1 and Clusters

- D(A1, C1) = |2 – 2| + |10 – 10|= 0
- D(A1, C2) = |5 – 2| + |8 – 10|= 0
- D(A1, C3) = |1 – 2| + |2 – 10|= 0

Similarly for all points

Given Points	Distance from center (2, 10) of Cluster-01	Distance from center (5, 8) of Cluster-02	Distance from center (1, 2) of Cluster-03	Point belongs to Cluster
A1(2, 10)	0	5	9	C1
A2(2, 5)	5	6	4	C3
A3(8, 4)	12	7	9	C2
A4(5, 8)	5	0	10	C2
A5(7, 5)	10	5	9	C2
A6(6, 4)	10	5	7	C2
A7(1, 2)	9	10	0	C3
A8(4, 9)	3	2	10	C2

Finding new cluster

- For Cluster-01:
  - We have only one point A1(2, 10) in Cluster-01.
  - So, cluster center remains the same.
- For Cluster-02:
  - Center of Cluster-02

    $= ((8 + 5 + 7 + 6 + 4)/5, (4 + 8 + 5 + 4 + 9)/5)$

    $= (6, 6)$
- For Cluster-03:
  - Center of Cluster-03

    $= ((2 + 1)/2, (5 + 2)/2)$

    $= (1.5, 3.5)$

Iteration 2

Given Points	Distance from center (2, 10) of Cluster-01	Distance from center (6, 6) of Cluster-02	Distance from center (1.5, 3.5) of Cluster-03	Point belongs to Cluster
A1(2, 10)	0	8	7	C1
A2(2, 5)	5	5	2	C3
A3(8, 4)	12	4	7	C2
A4(5, 8)	5	3	8	C2
A5(7, 5)	10	2	7	C2
A6(6, 4)	10	2	5	C2
A7(1, 2)	9	9	2	C3
A8(4, 9)	3	5	8	C1

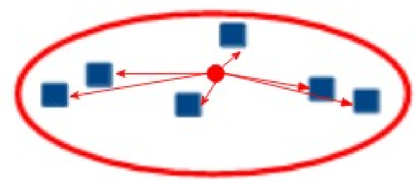
Finding new cluster

- For Cluster-01:
  - Center of Cluster-02
    = ((2 + 4)/2, (10+ 9)/2)
    = (3, 9.5)
- For Cluster-02:
  - Center of Cluster-02
    = ((8 + 5 + 7 + 6)/4, (4 + 8 + 5 + 4)/4)
    = (6.5, 5.25)
- For Cluster-03:
  - Center of Cluster-03
    = ((2 + 1)/2, (5 + 2)/2)
    = (1.5, 3.5)

- So the new clusters after iteration 2 are
- C1(3, 9.5)
- C2(6.5, 5.25)
- C3(1.5, 3.5)

Different Evaluation Metrics for Clustering

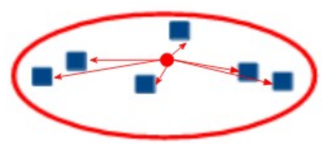
- Inertia
  - calculates the sum of distances of all the points within a cluster from the centroid of that cluster.
  - Also known as intracluster distance



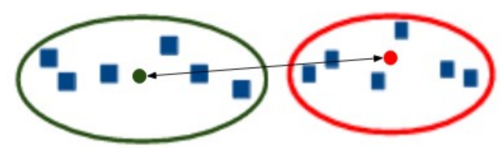
Intra cluster distance

- Dunn Index

$$\text{Dunn Index} = \frac{\min(\text{Inter cluster distance})}{\max(\text{Intra cluster distance})}$$

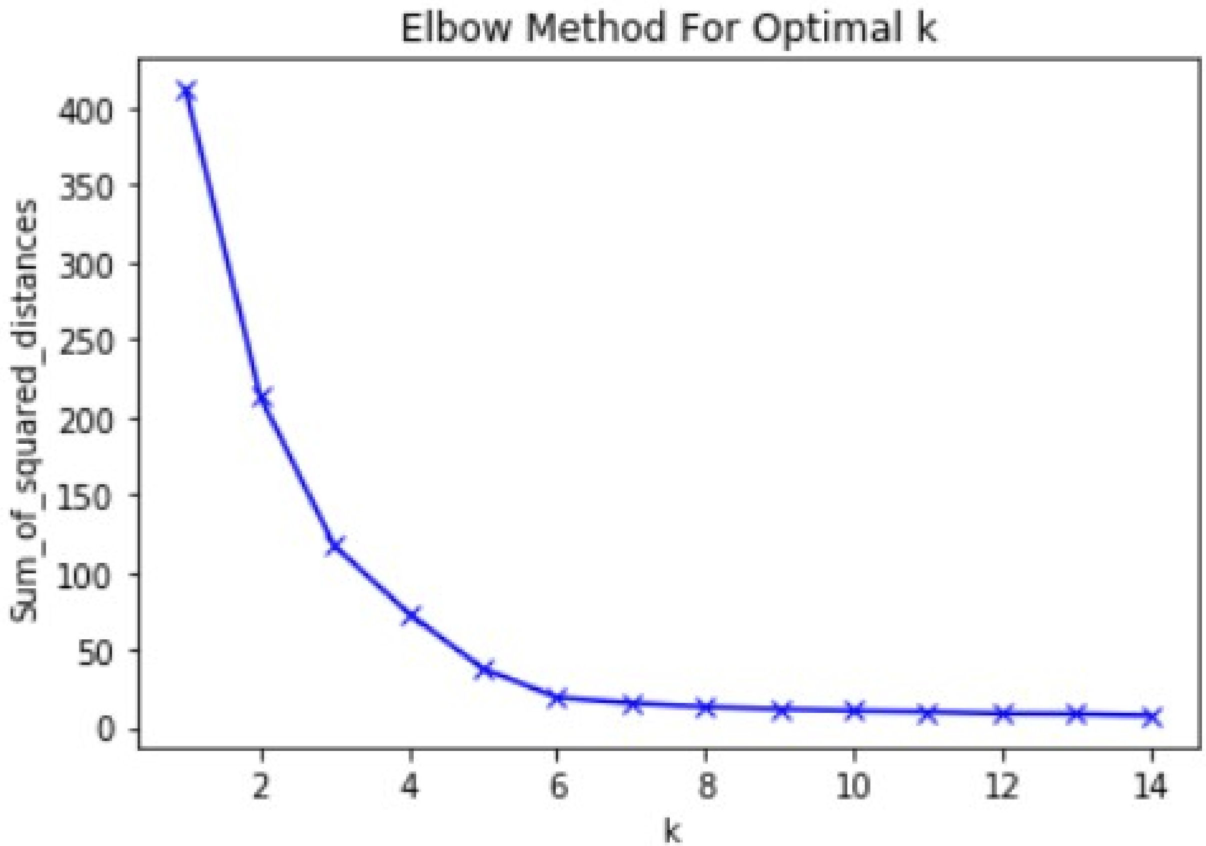


Intra cluster distance



Inter cluster distance

How to find optimum K : Elbow method



Pros

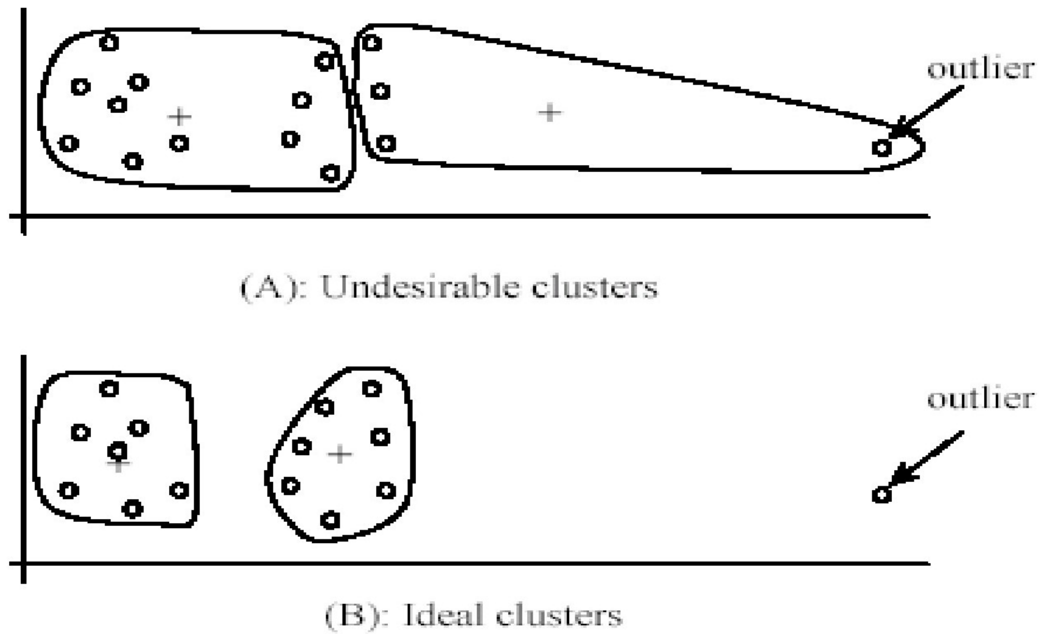
- Simple & easy to understand and to implement
- Efficient: Time complexity: O(tkn),
  - where n is the number of data points,
  - k is the number of clusters, and
  - t is the number of iterations.
- Since both k and t are small. k-means is considered a linear algorithm.
- K-means is the most popular clustering algorithm.

Cons

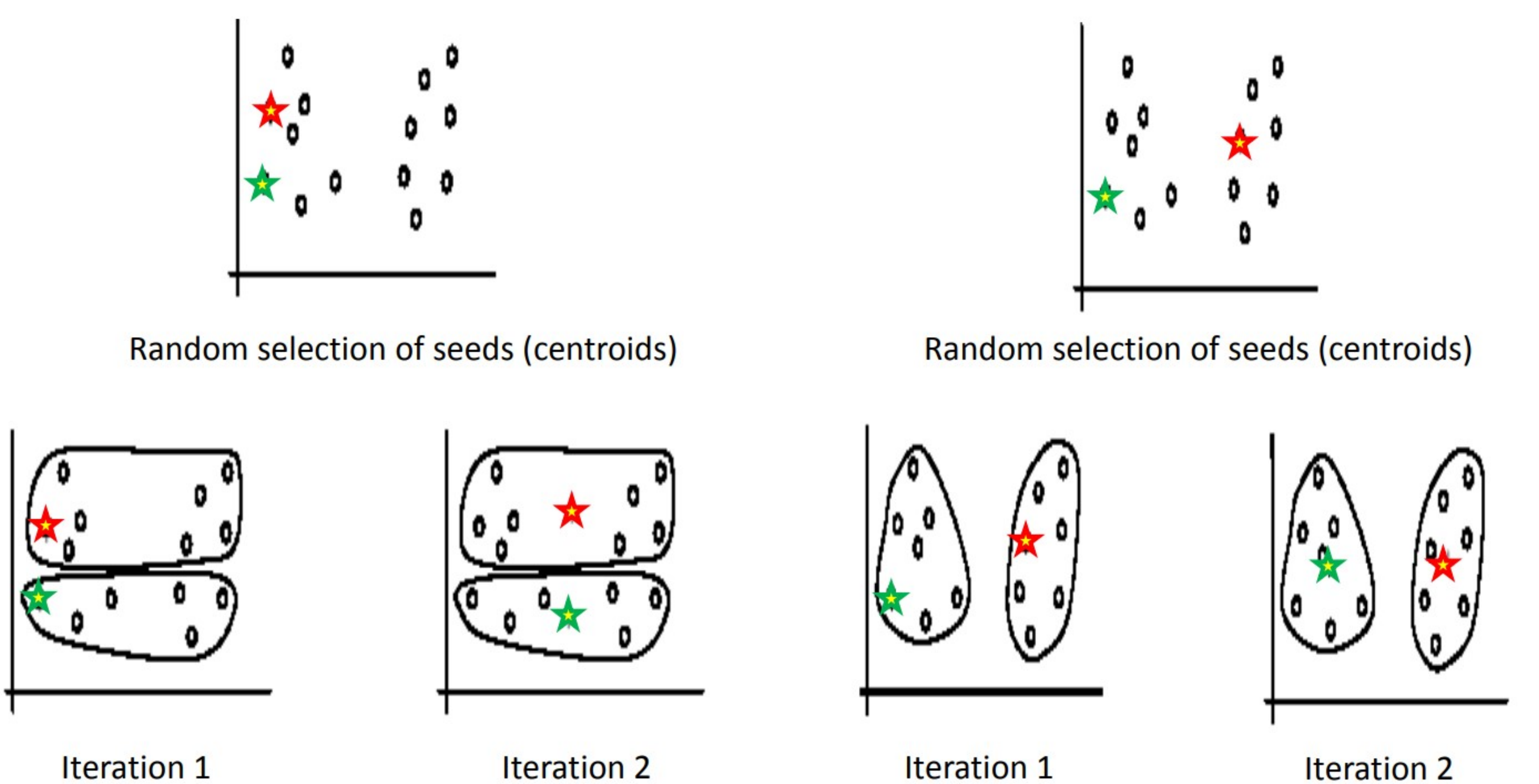
- The algorithm is only applicable if the mean is defined.
  - For categorical data, k-mode - the centroid is represented by most frequent values.
- The user needs to specify k.
- The algorithm is sensitive to outliers
  - Outliers are data points that are very far away from other data points.
  - Outliers could be errors in the data recording or some special data points with very different values.
- Sensitive to initially selected random centroids
- Not suitable for hyper-ellipsoids (or hyper-sphere)

Dealing with outliers

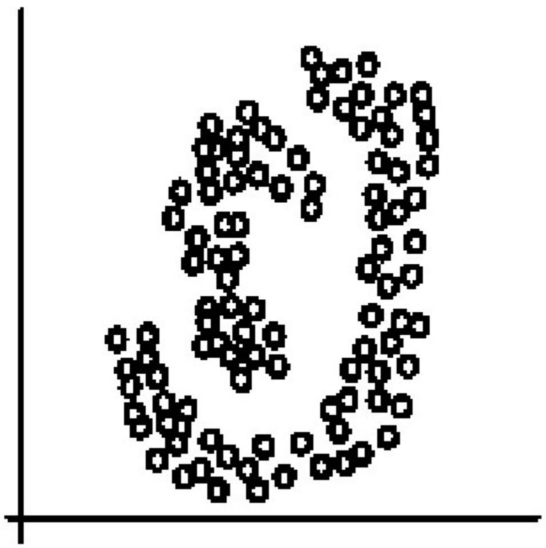
- Remove some data points that are much further away from the centroids than other data points
- Perform random sampling: by choosing a small subset of the data points, the chance of selecting an outlier is much smaller



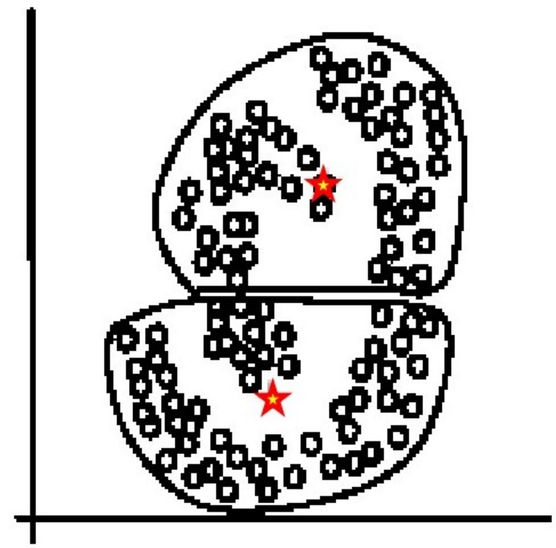
Sensitivity to initial seeds



Not suitable for hyper-ellipsoids(or sphere)



(A): Two natural clusters



(B): *k*-means clusters

THANK YOU!!!!