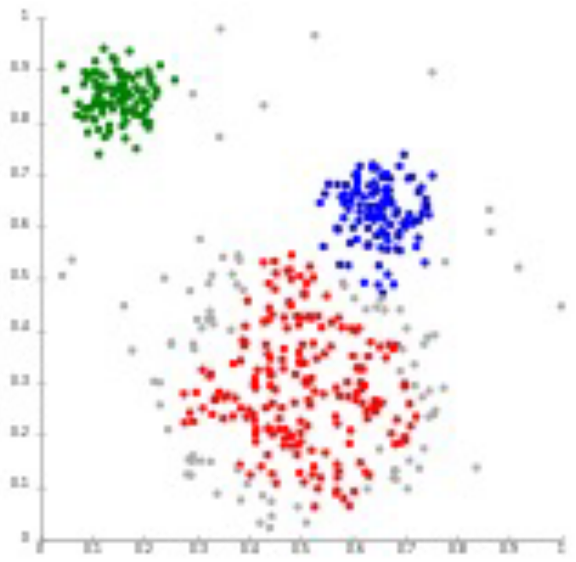
LEADING 2022
Unsupervised Learning

Shadi Rezapour, Ph.D.
Assistant Professor
Drexel University
sr3563@Drexel.edu
https://www.shadirezapour.com/

Supervised and Unsupervised ML

- Supervised learning (classification)
  - Supervision: The training data (observations, measurements, etc.) are accompanied by **labels** indicating the class of the observations
  - New data is classified based on the training set
- Unsupervised learning (clustering)
  - The class labels of training data is unknown
  - Given a set of measurements, observations, etc. with the aim of establishing the existence of classes or clusters in the data

What is Cluster Analysis?

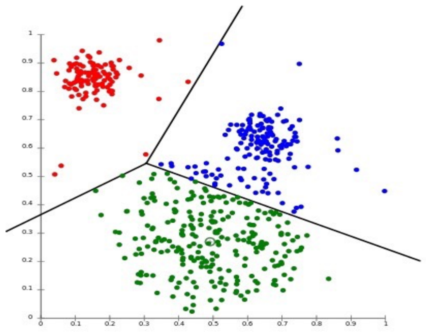


- Unsupervised learning: no predefined classes
- Cluster: a collection of data objects
  - Similar to one another within the same cluster
  - Dissimilar to the objects in other clusters
- Cluster analysis
  - Finding similarities between data according to the characteristics found in the data and grouping similar data objects into clusters
- Similarity measures (types of objects, similarity dissimilarity measures)
- Typical applications
  - As a stand-alone tool to get insight into data distribution
  - As a preprocessing step for other algorithms

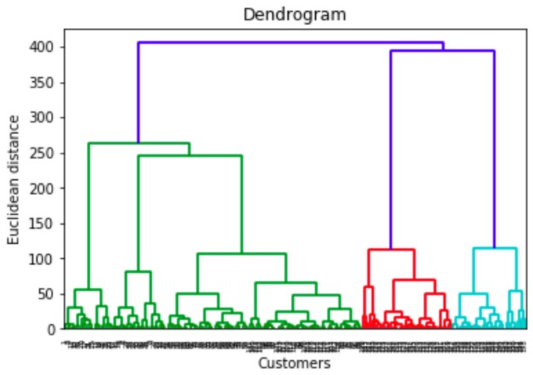
Clustering: Rich Applications and Multidisciplinary Efforts

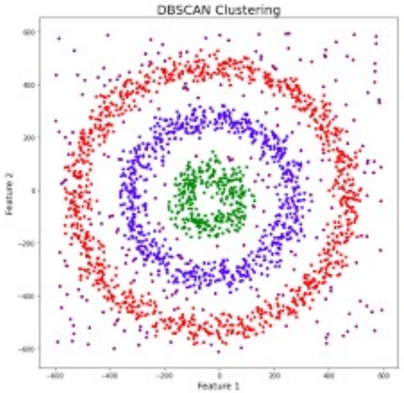
- Pattern Recognition
- Spatial Data Analysis
  - Create thematic maps in GIS by clustering feature spaces
  - Detect spatial clusters or for other spatial mining tasks
- Image Processing
- Economic Science (especially market research)
- WWW
  - Document classification
  - Cluster Weblog data to discover groups of similar access patterns

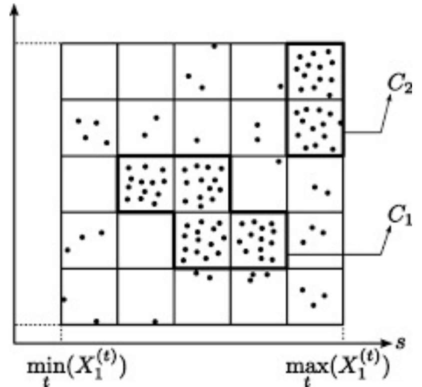
Major Clustering Approaches



- Partitioning approach:
    - Construct various partitions and then evaluate them by some criterion, e.g., minimizing the sum of square err
    - Typical methods: k-means, k-medoids, CLARANS
- Hierarchical approach:
    - Create a hierarchical decomposition of the set of data (or objects) using some criterion
    - Typical methods: Diana, Agnes, BIRCH, ROCK, CAMELEON
- Density-based approach:
    - Based on connectivity and density functions
    - Typical methods: DBSACN, OPTICS, DenClue
- Grid-based approach:
    - based on a multiple-level granularity structure
    - Typical methods: STING, WaveCluster, CLIQUE







The *K-Means* Clustering Method

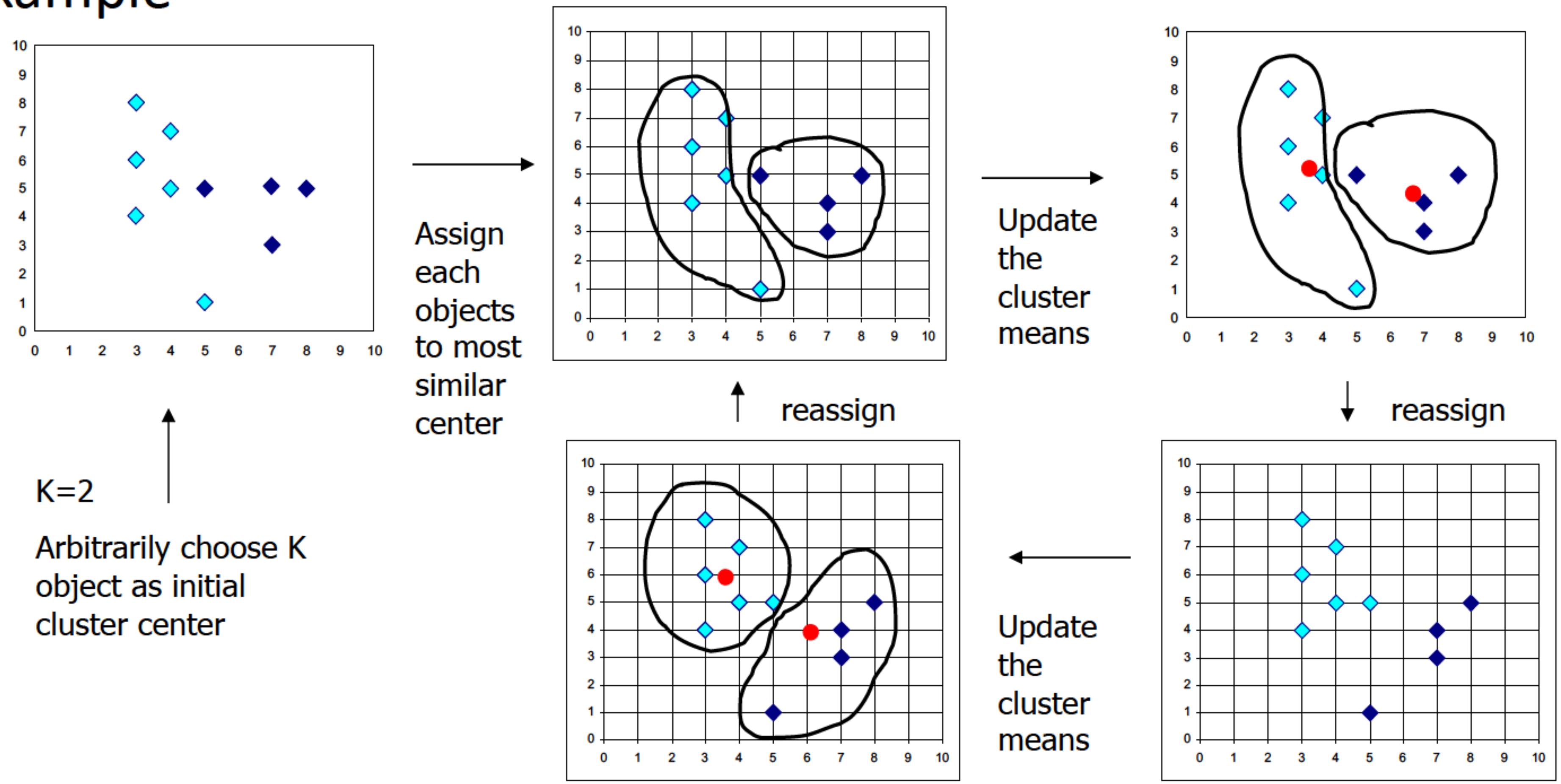
- Given *k*, the *k-means* algorithm is implemented in four steps:
  - Partition objects into *k* nonempty subsets
  - Compute seed points as the centroids of the clusters of the current partition (the centroid is the center, i.e., *mean point*, of the cluster)
    - Centroid: the "middle" of a cluster

$$C_m = \frac{\sum_{i=1}^{N}(t_{ip})}{N}$$

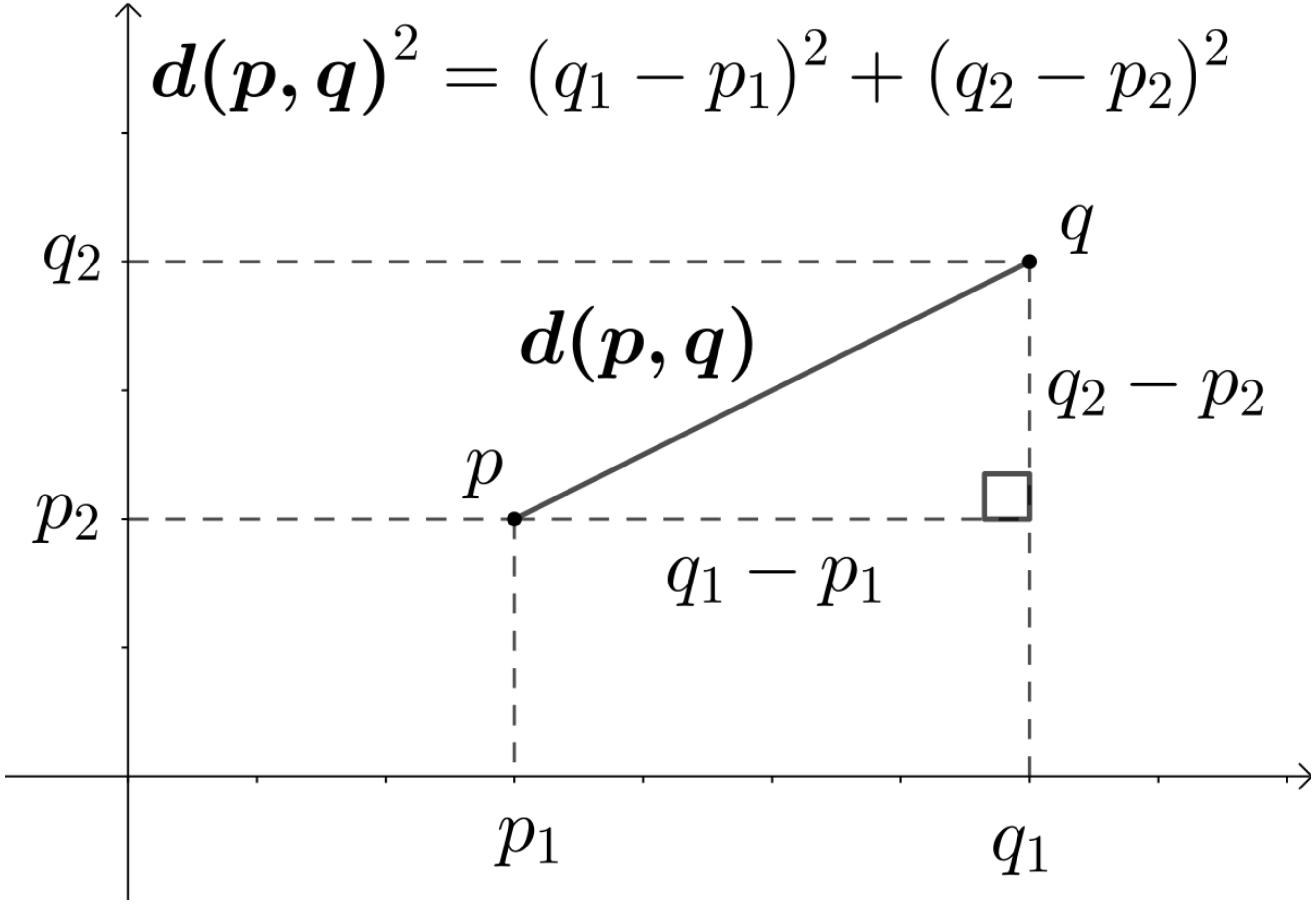
  - Assign each object to the cluster with the nearest seed point
  - Go back to Step 2, stop when no more new assignment

The *K-Means* Clustering Method

- Example



Euclidean Distance



Example of *K-Means* Method

	X1	X2
A	2	3
B	6	1
C	1	2
D	3	0

	X1	X2
AB	4	2
CD	2	1

Choose two centroids AB and CD,

AB = Average of A, B

CD = Average of C,D

Example of *K-Means* Method

	X1	X2
A	2	3
B	6	1
C	1	2
D	3	0

	X1	X2
AB	4	2
CD	2	1

	A	B	C	D
AB	5	5	9	5
CD	4	16	2	2

Choose two centroids AB and CD,

AB = Average of A, B

CD = Average of C,D

Calculate squared Euclidean distance between all data points to the centroids AB, CD.

For example; distance between A(2,3) and AB (4,2) can be given by $s^2 = (2-4)^2 + (3-2)^2$

Example of *K-Means* Method

	X1	X2
A	2	3
B	6	1
C	1	2
D	3	0

	X1	X2
AB	4	2
CD	2	1

	A	B	C	D
AB	5	5	9	5
CD	4	16	2	2

Choose two centroids AB and CD,

AB = Average of A, B

CD = Average of C,D

Calculate squared Euclidean distance between all data points to the centroids AB, CD.

For example; distance between A(2,3) and AB (4,2) can be given by $s^2 = (2-4)^2 + (3-2)^2$.

Example of *K-Means* Method

	X1	X2
A	2	3
B	6	1
C	1	2
D	3	0

	X1	X2
AB	4	2
CD	2	1

Choose two centroids AB and CD,
AB = Average of A, B
CD = Average of C,D

	A	B	C	D
AB	5	5	9	5
CD	4	16	2	2

Calculate squared Euclidean distance between all data points to the centroids AB, CD.
For example; distance between A(2,3) and AB (4,2) can be given by $s^2 = (2-4)^2 + (3-2)^2$.

	X1	X2
B	6	1
ACD	2	1.67

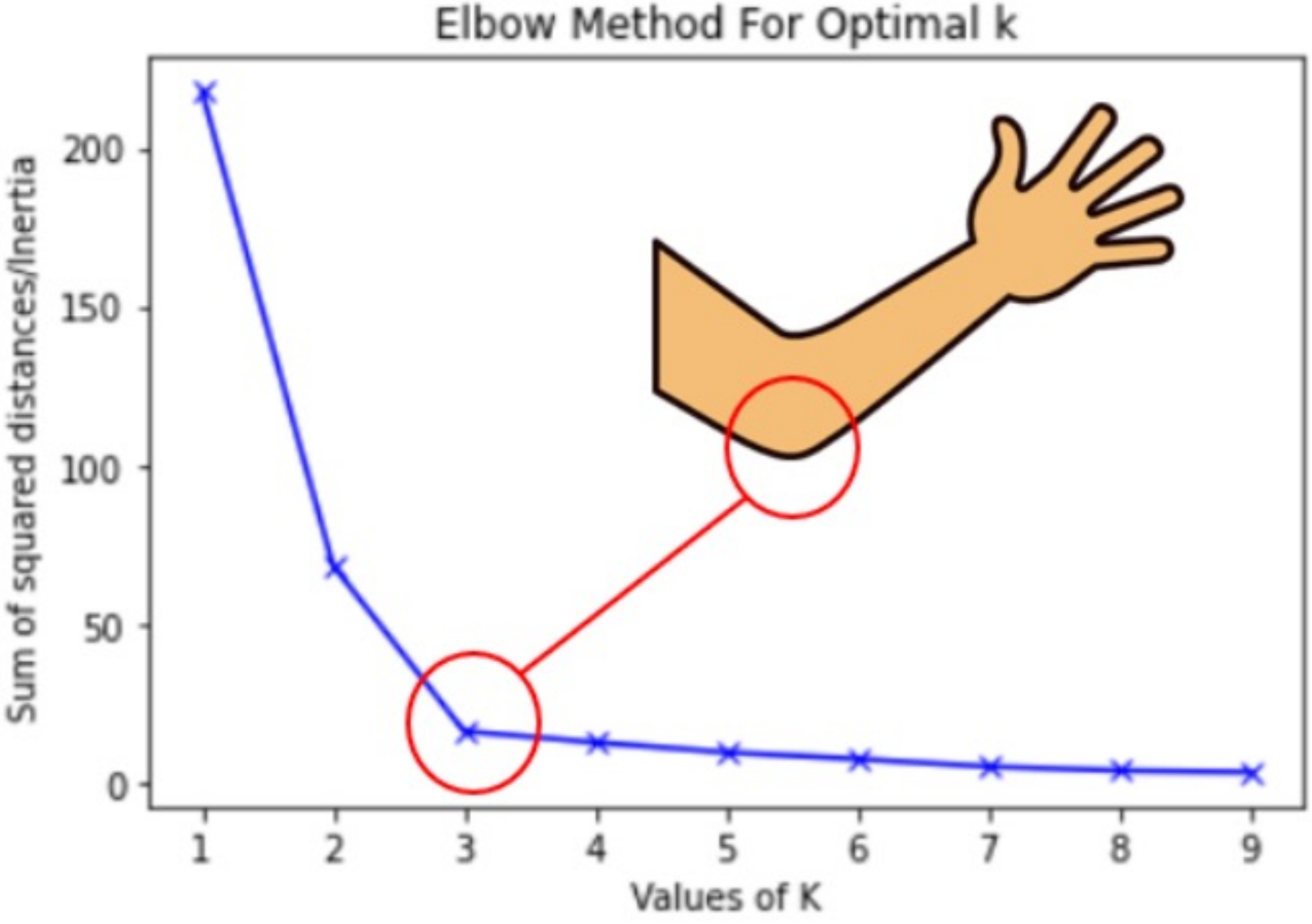
Choose new centroids
ACD = Average of A, C, D
B = B

	A	B	C	D
B	20	0	26	10
ACD	3.78	16.44	1.11	3.78

Comments on the *K-Means* Method

- <u>Strength:</u> *Relatively efficient*: $O(tkn)$, where n is # objects, k is # clusters, and t is # iterations. Normally, k, $t << n$.
- <u>Comment:</u> Often terminates at a *local optimum*. The *global optimum* may be found using techniques such as: *deterministic annealing* and *genetic algorithms*
- <u>Weakness</u>
  - Applicable only when *mean* is defined, then what about categorical data?
  - Need to specify *k,* the *number* of clusters, in advance

Selecting K: Elbow Method



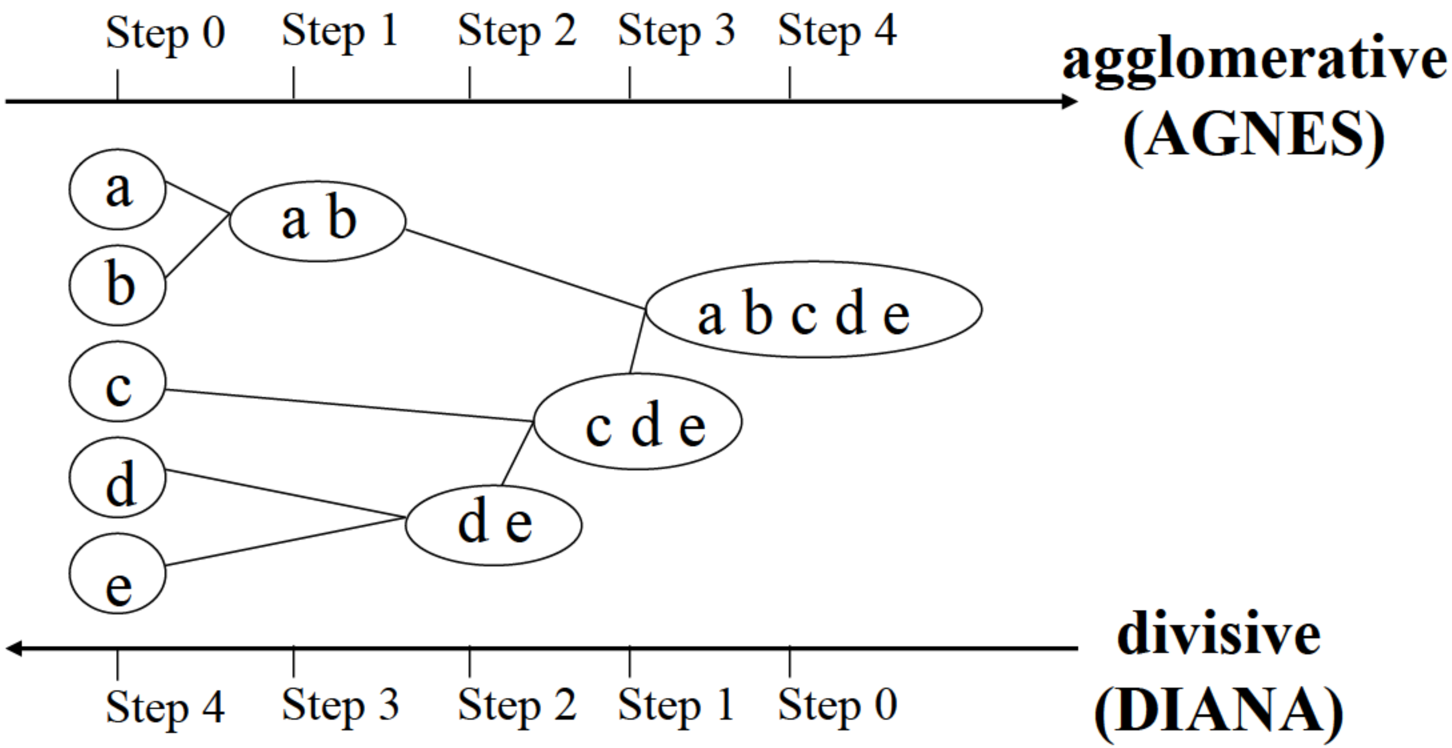
Line plot between K and inertia

Comments on the *K-Means* Method

- <u>Strength:</u> *Relatively efficient*: $O(tkn)$, where n is # objects, k is # clusters, and t is # iterations. Normally, k, $t << n$.
- <u>Comment:</u> Often terminates at a *local optimum*. The *global optimum* may be found using techniques such as: *deterministic annealing* and *genetic algorithms*
- <u>Weakness</u>
  - Applicable only when *mean* is defined, then what about categorical data?
  - Need to specify *k,* the *number* of clusters, in advance
  - Unable to handle noisy data and *outliers*
  - Not suitable to discover clusters with *non-convex shapes*

Hierarchical Clustering

- Use distance matrix as clustering criteria. This method does not require the number of clusters ***k*** as an input, but needs a termination condition



Bottom-up Clustering

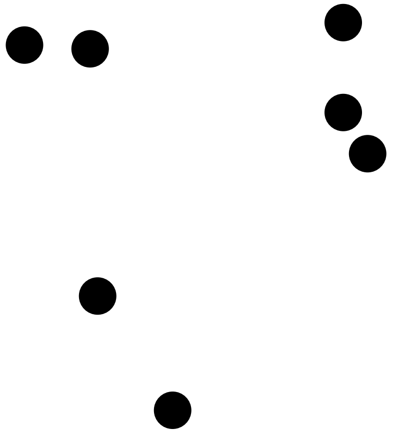
- Input: data
- Output: cluster hierarchy
- Algorithm:
  - Step 1: consider every data point as its own cluster
  - Step 2: compute the distance between all cluster pairs
  - Step 3: merge/combine the nearest two clusters into one
  - Step 4: repeat steps 2 and 3 until all data instances is in one cluster

Bottom-up Clustering

- Computing the distance between two clusters
  - **Single-Link:** the distance between the two nearest data points
  - **Complete-Link**: the distance between the two data points that are farthest apart
  - **Centroid-Link:** the distance between the centroids of two different clusters

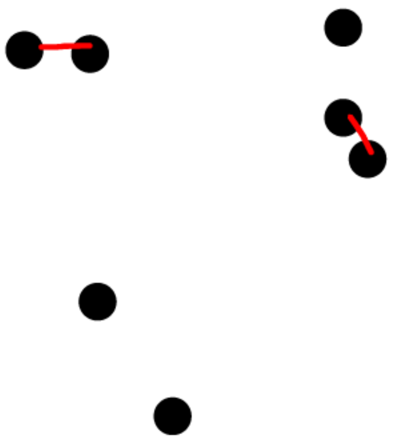
Bottom-up Clustering: Single Link

- Step 1: consider each data point its own cluster



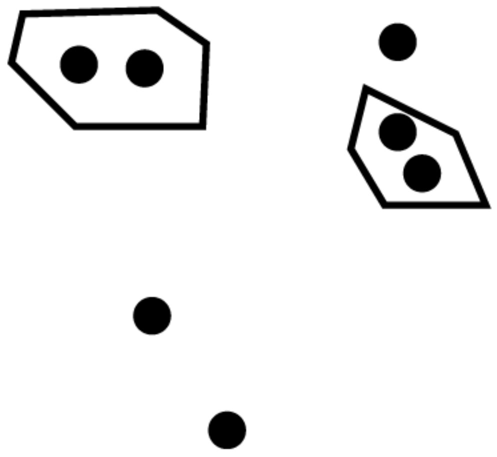
Bottom-up Clustering: Single Link

- Step 2: compute the distance between all cluster pairs
- Step 3: merge/combine the nearest two clusters into one



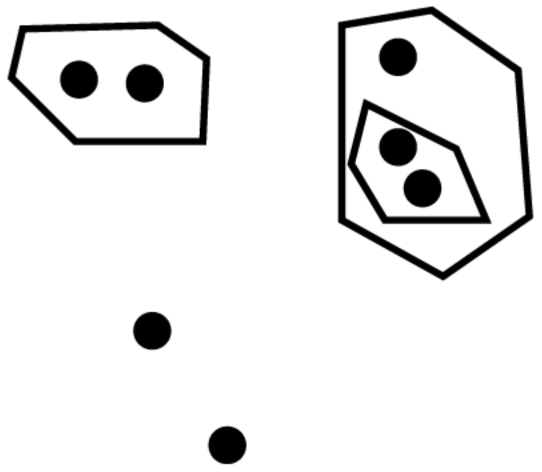
Bottom-up Clustering: Single Link

- Step 2: compute the distance between all cluster pairs
- Step 3: merge/combine the nearest two clusters into one



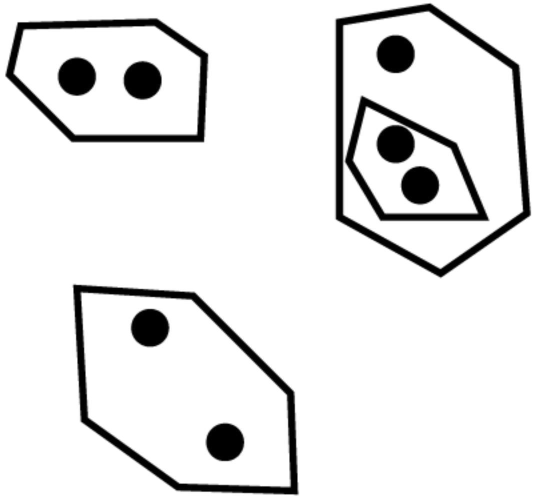
Bottom-up Clustering: Single Link

- Step 2: compute the distance between all cluster pairs
- Step 3: merge/combine the nearest two clusters into one



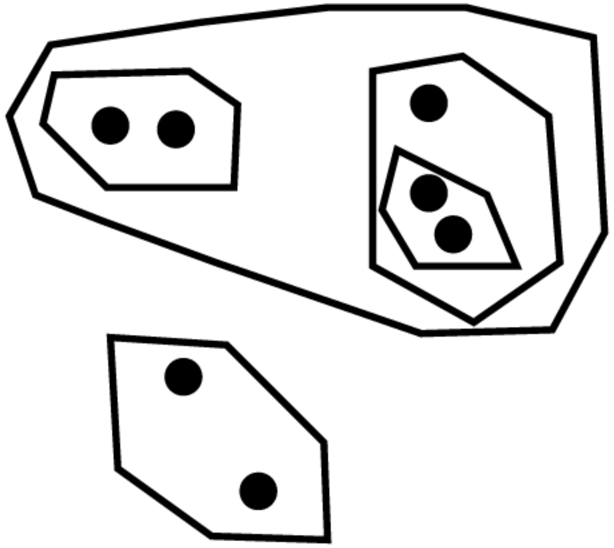
Bottom-up Clustering: Single Link

- Step 2: compute the distance between all cluster pairs
- Step 3: merge/combine the nearest two clusters into one



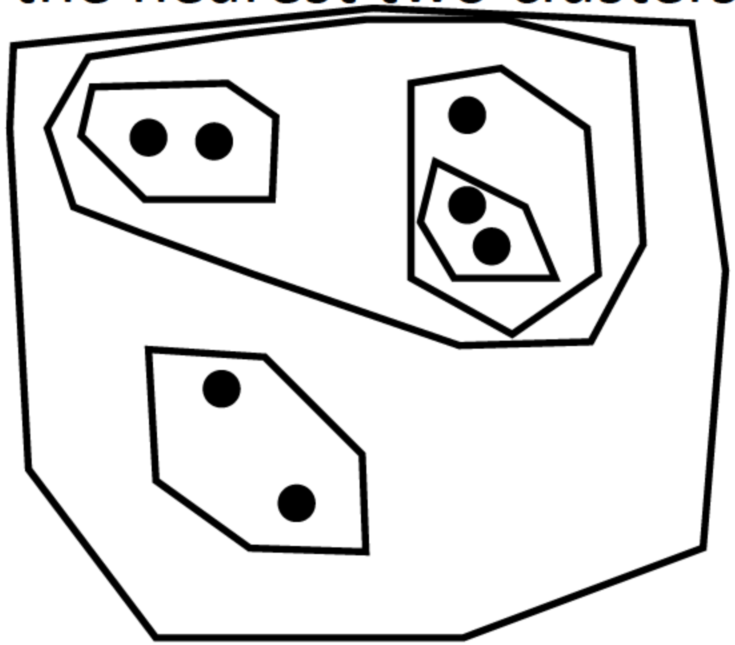
Bottom-up Clustering: Single Link

- Step 2: compute the distance between all cluster pairs
- Step 3: merge/combine the nearest two clusters into one



Bottom-up Clustering: Single Link

- Step 2: compute the distance between all cluster pairs
- Step 3: merge/combine the nearest two clusters into one



Coding time!