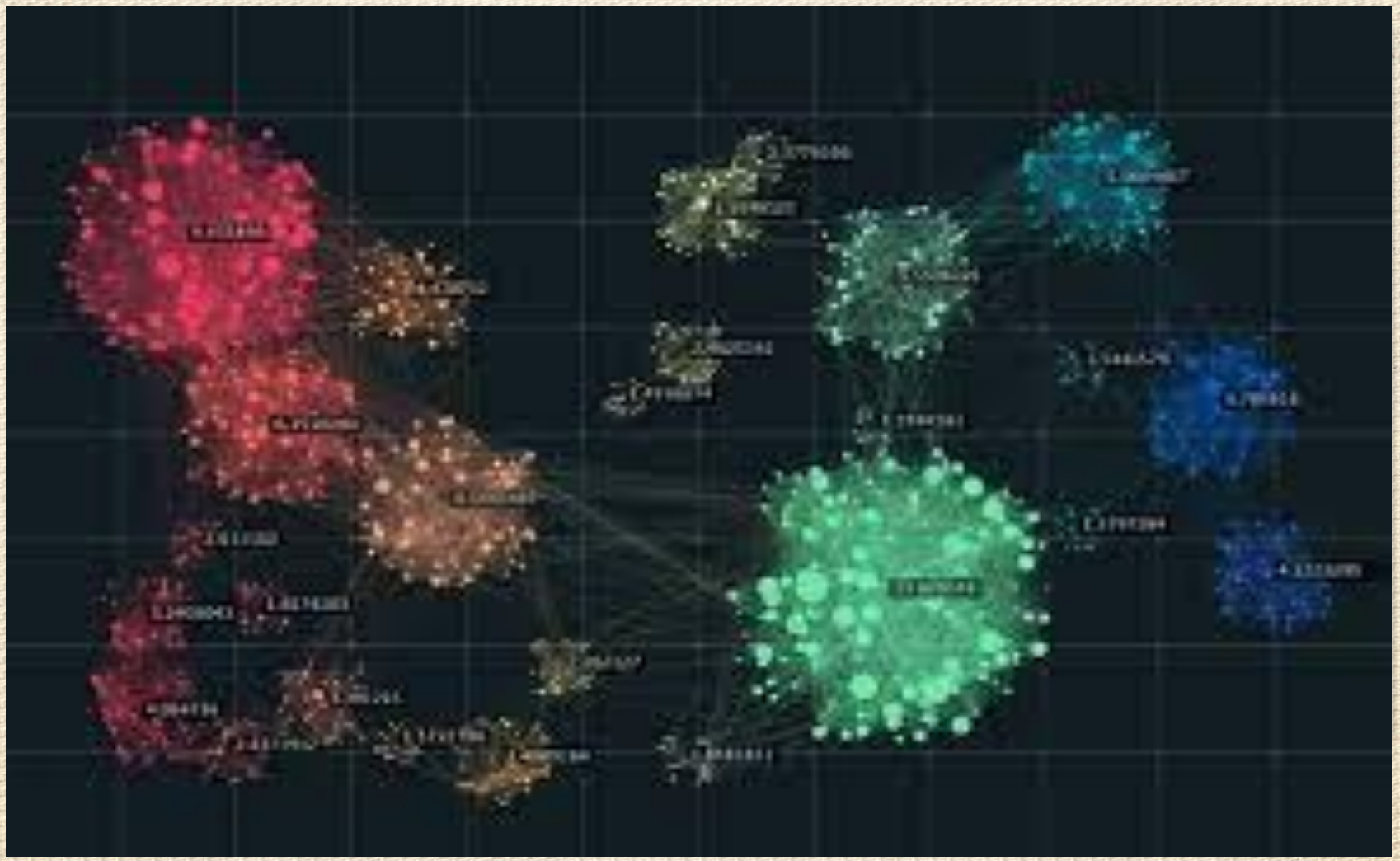
Module 5- CLUSTERING

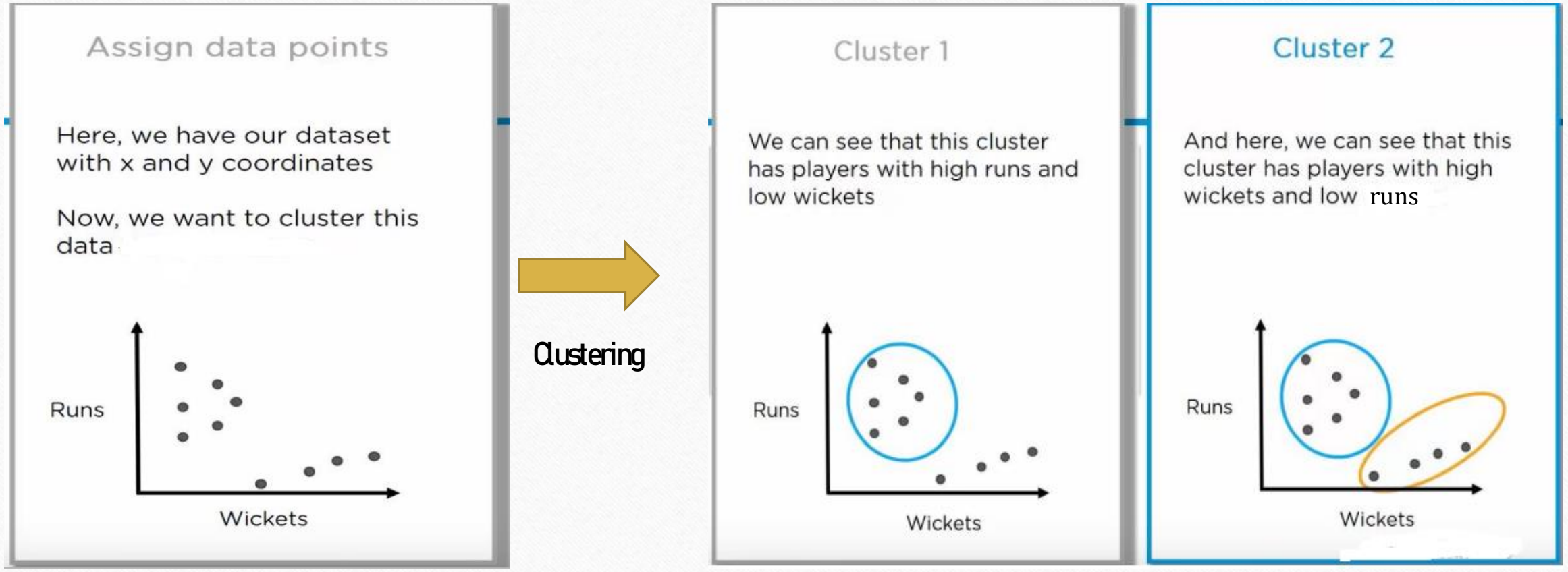


What's clustering?

Task: Identify bowlers and batsmen

- ❑ The data contains runs and wickets gained in the last 10 matches
- ❑ So, the bowler will have more wickets and the batsmen will have higher runs

Scores



Clustering is the process of grouping a set of objects into classes of similar objects.

Clustering

- Clustering is an unsupervised machine learning task that automatically divides the data into clusters of similar items. It does this without having been told what the groups should look like.
- Clustering is used for knowledge discovery rather than prediction. It provides an insight into the natural groupings found within data.

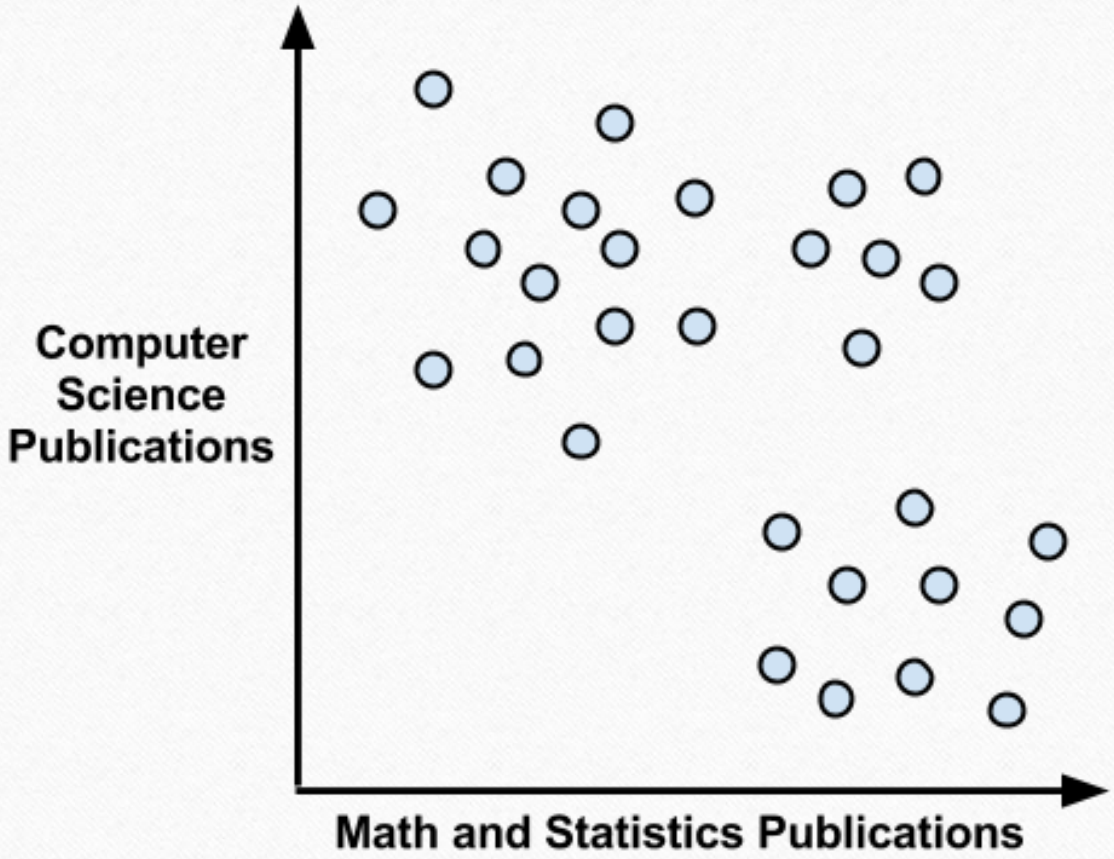
Applications of Clustering

- Clustering methods employed in applications such as:
- Segmenting customers into groups with similar demographics or buying patterns for targeted marketing campaigns and/or detailed analysis of purchasing behavior by subgroup.
- Detecting anomalous behavior, such as unauthorized intrusions into computer networks, by identifying patterns of use falling outside known clusters.
- Simplifying extremely large datasets by grouping a large number of features with similar values into a much smaller number of homogeneous categories.

- Clustering is somewhat different from the classification, numeric prediction, and pattern detection tasks. In each of these cases, the result is a model that **relates features to an outcome or features to other features;** the **model identifies patterns within data.**
- In contrast, clustering **creates new data.** Unlabelled examples are given a cluster label and inferred entirely from the relationships within the data. For this reason, the clustering task referred to as **unsupervised classification** because, this is classifying unlabelled examples.

Example

- Suppose you were organizing a conference on the topic of data science. To facilitate professional networking and collaboration, you planned to seat people in groups according to one of three research specialties: computer and/or database science, math and statistics, and machine learning.
- Unfortunately, after sending out the conference invitations, you realize that you had forgotten to include a survey asking the discipline the attendee would prefer to be seated with. In a stroke of brilliance, you realize that you might be able to infer each scholar's research specialty by examining his or her publication history. Toward this end,you begin collecting data on the number of articles each attendee published in computer science-related journals and the number of articles published in math or statistics-related journals. Using the data collected for several scholars, you create a scatterplot:

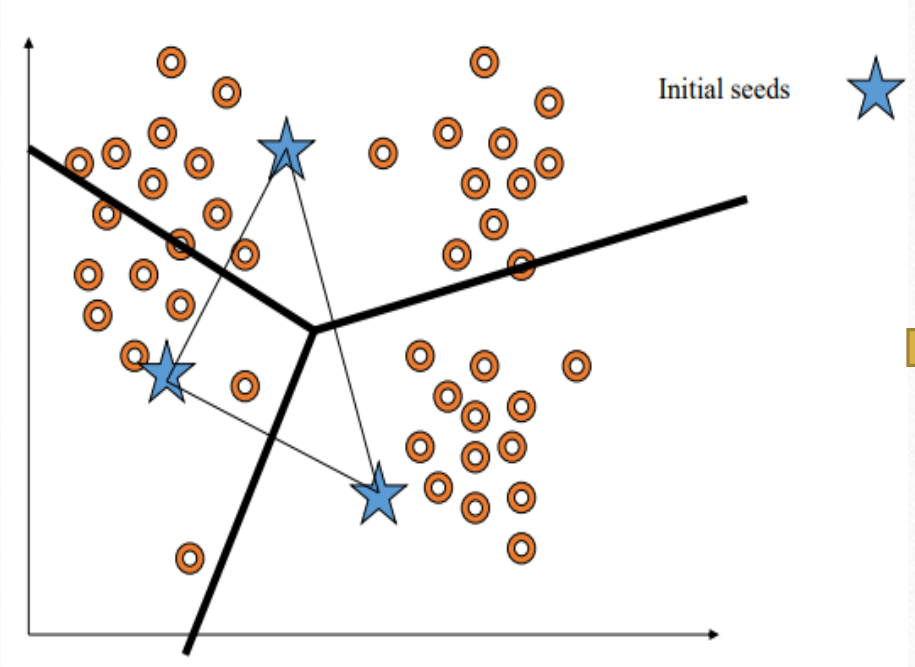

Computer Science Publications
Math and Statistics Publications

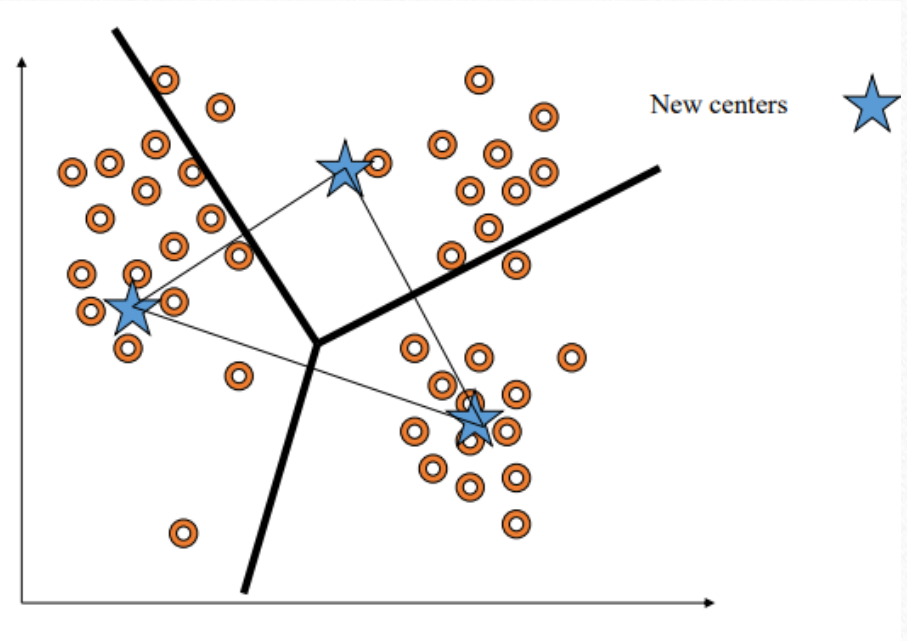
Clustering

- Clustering: the process of grouping a set of objects into classes of similar object.
  - Documents within a cluster should be similar.
  - Documents from different clusters should be dissimilar.
- Goal: maximize intra-cluster similarity and minimize inter-cluster similarity.
- Unsupervised learning = learning from raw data

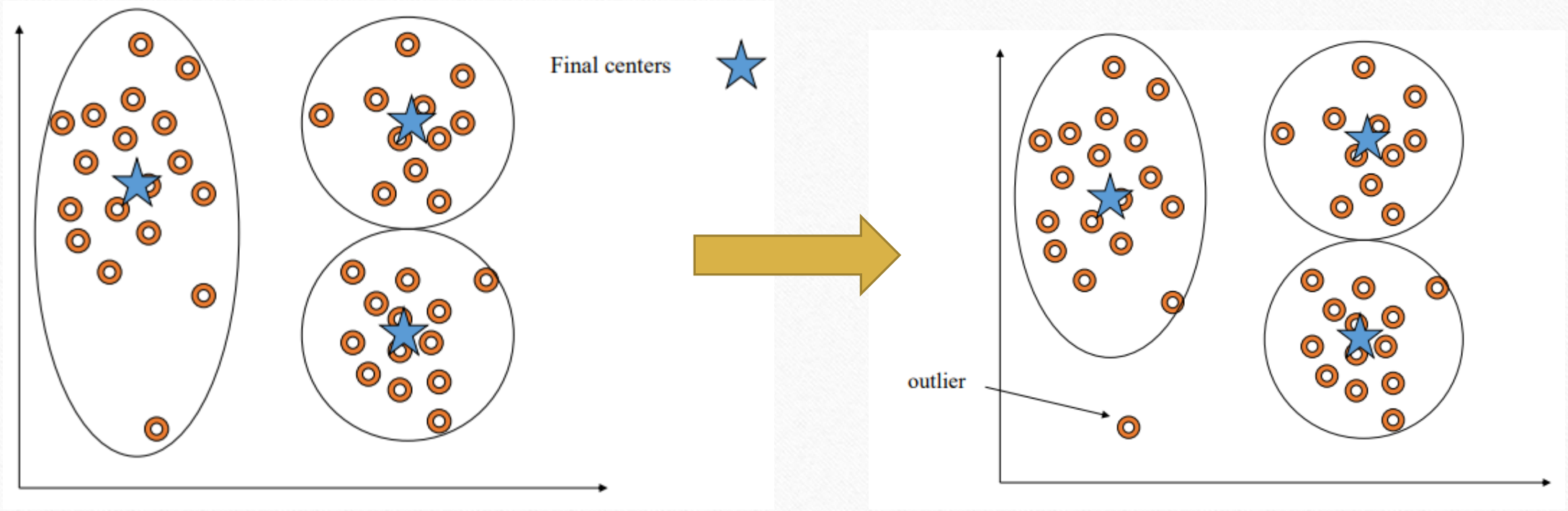
K-means Clustering

K = 3





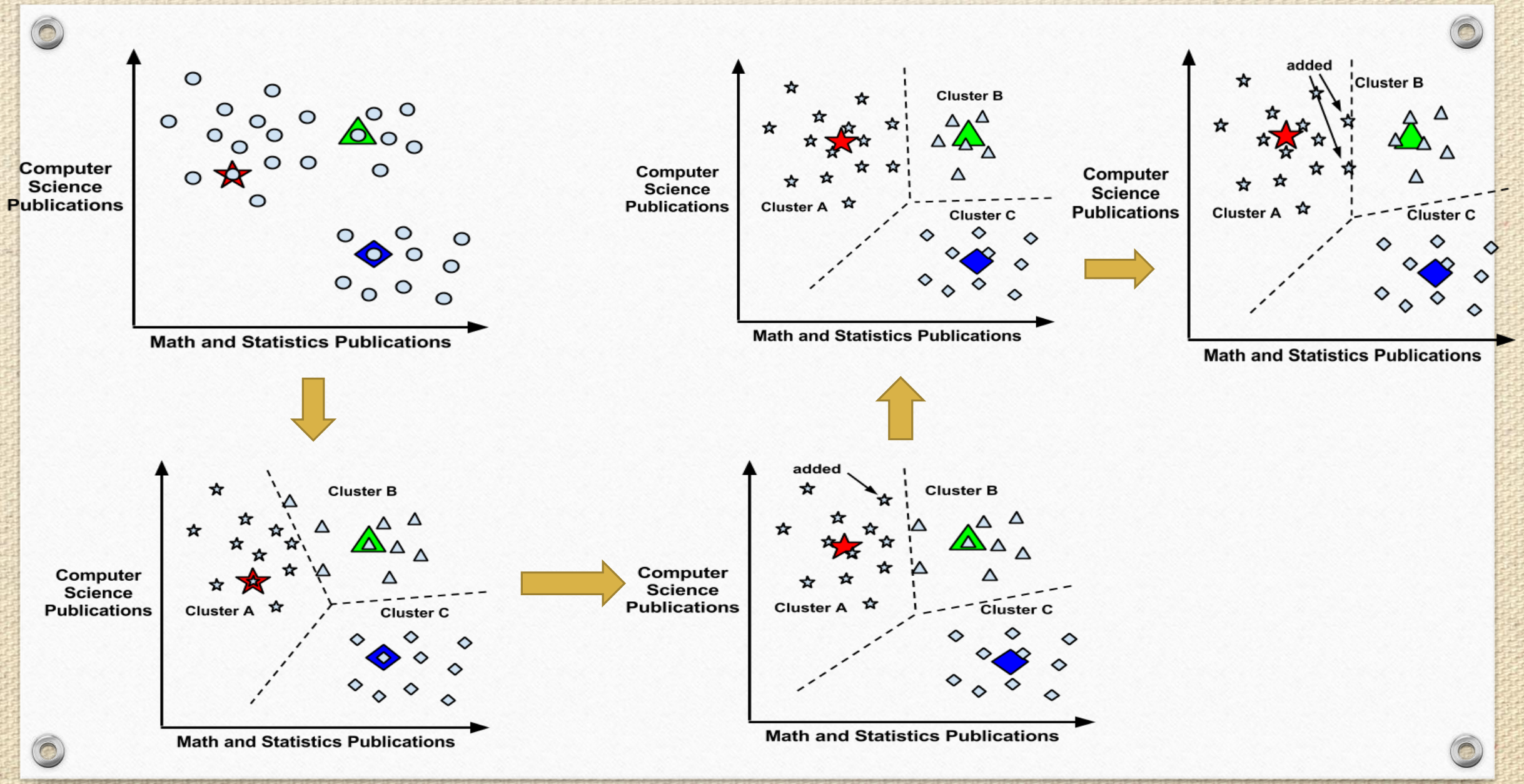
K-means Clustering



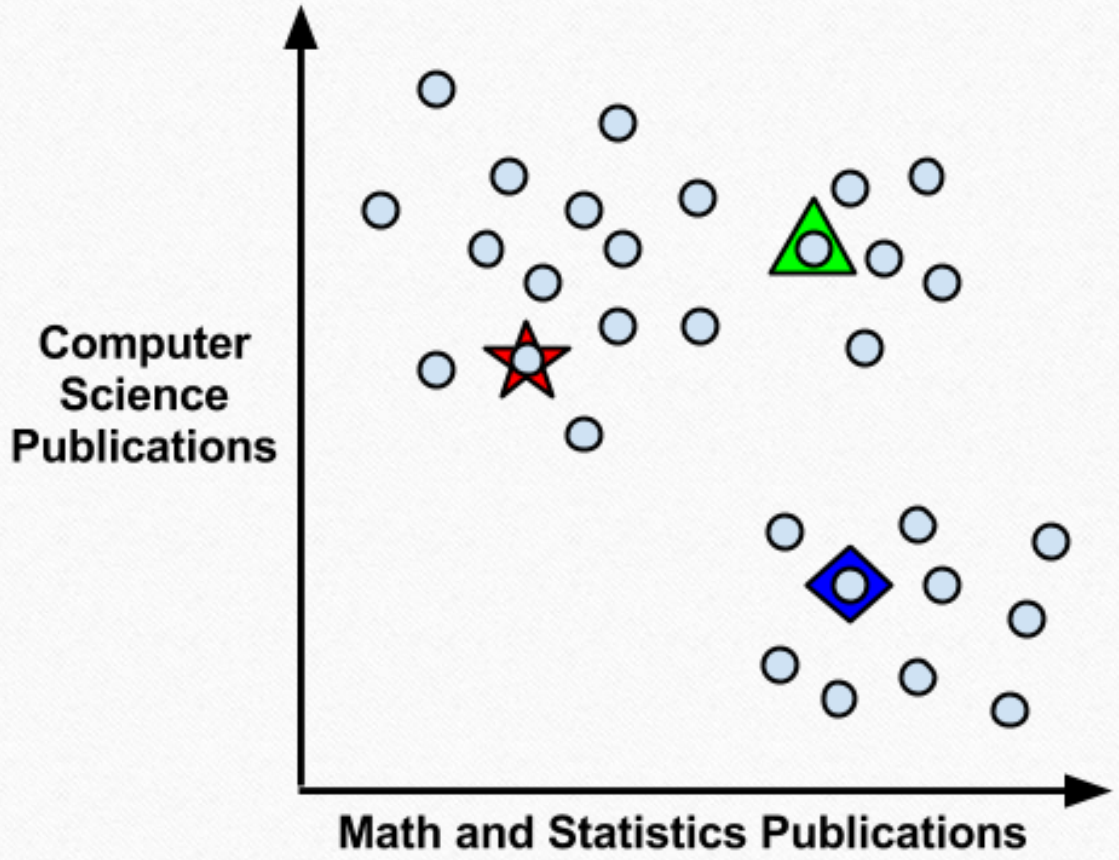
The k-means algorithm for clustering

- The k-means algorithm is perhaps the most often used clustering method.
- The k-means algorithm involves assigning each of the '*n*' examples to one of the k clusters (where k is a number that need to determine earlier).
- The goal is to minimize the differences within each cluster and maximize the differences between clusters.

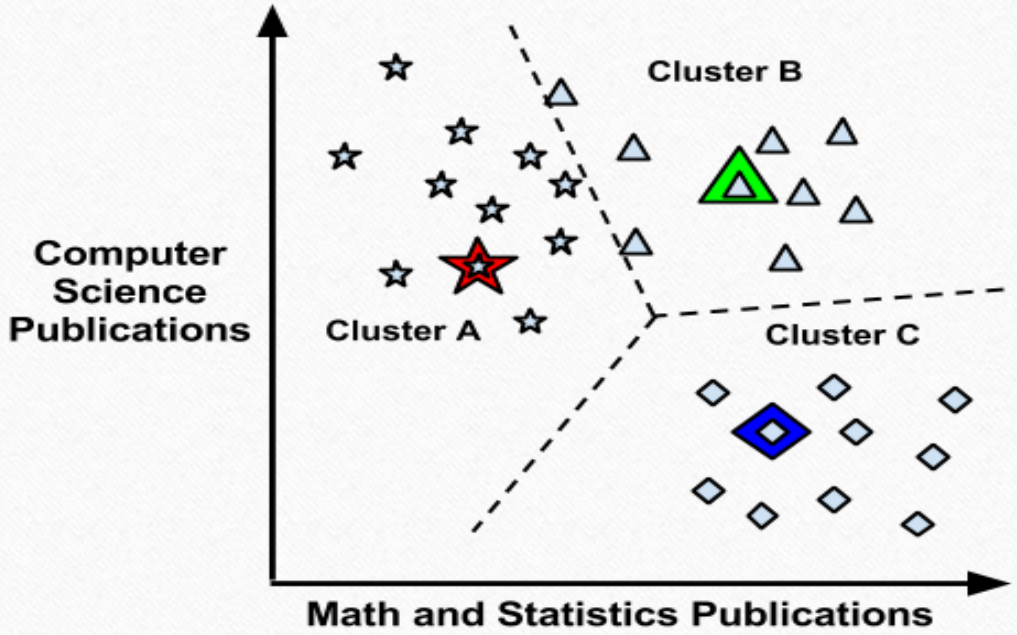
- The algorithm starts with an initial guess for the cluster assignments then modifies the assignments slightly to improve the homogeneity within the clusters.
- First, it assigns examples to an initial set of k clusters. Then, it updates the assignments by adjusting the cluster boundaries according to the examples that currently fall into the cluster.
- The process of updating and assigning occurs several times until making changes no longer improves the cluster fit. At this point, the process stops and the clusters are finalized.


Computer Science Publications
Math and Statistics Publications
Cluster A
Cluster B
Cluster C
added

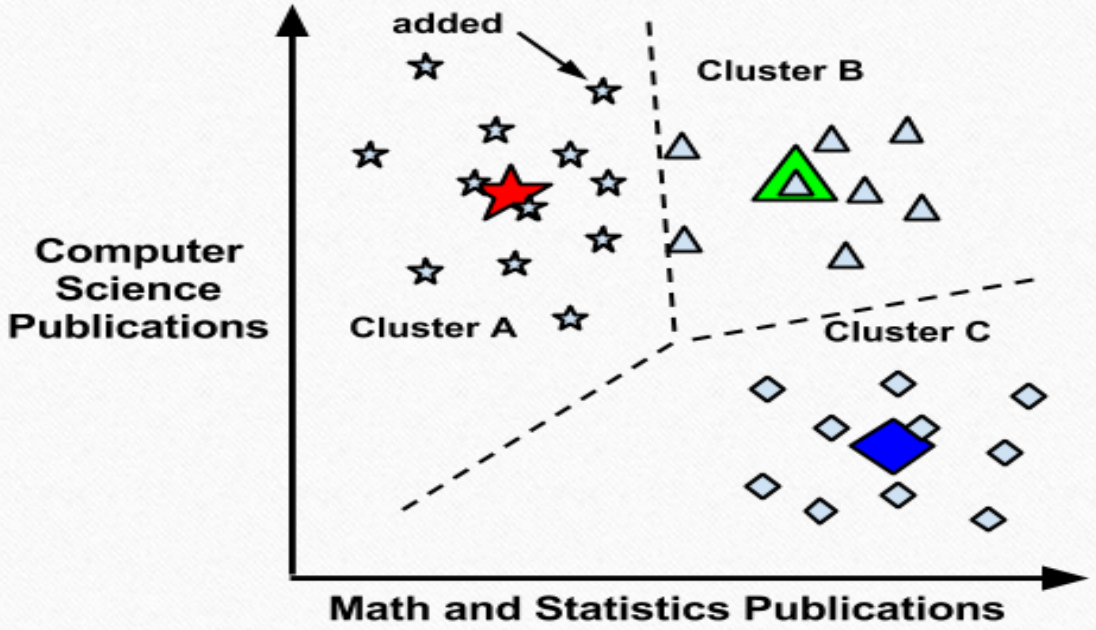
- As with kNN, k-means treats feature values as coordinates in a multidimensional feature space.
- The k-means algorithm begins by choosing k points in the feature space to serve as the cluster centers.
- Often, the points are chosen by selecting k random examples from the training dataset. Because we hope to identify three clusters, k = 3 points are selected. These points are indicated by the star, triangle, and diamond in the following figure:


Computer Science Publications
Math and Statistics Publications

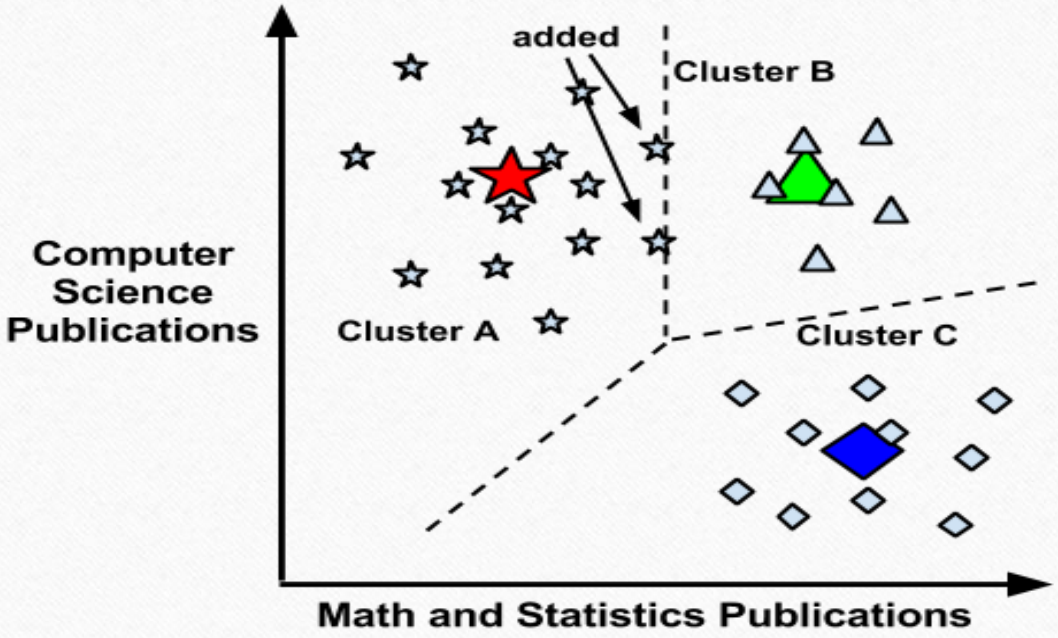
- After choosing the initial cluster centers, the other examples are assigned to the cluster center that is most similar or nearest according to the distance function. Traditionally, k-means uses Euclidean distance, but Manhattan distance or Minkowski distance are also sometimes used.
- As shown in the following figure, the three cluster centers partition the examples into three segments labeled Cluster A, B, and C.



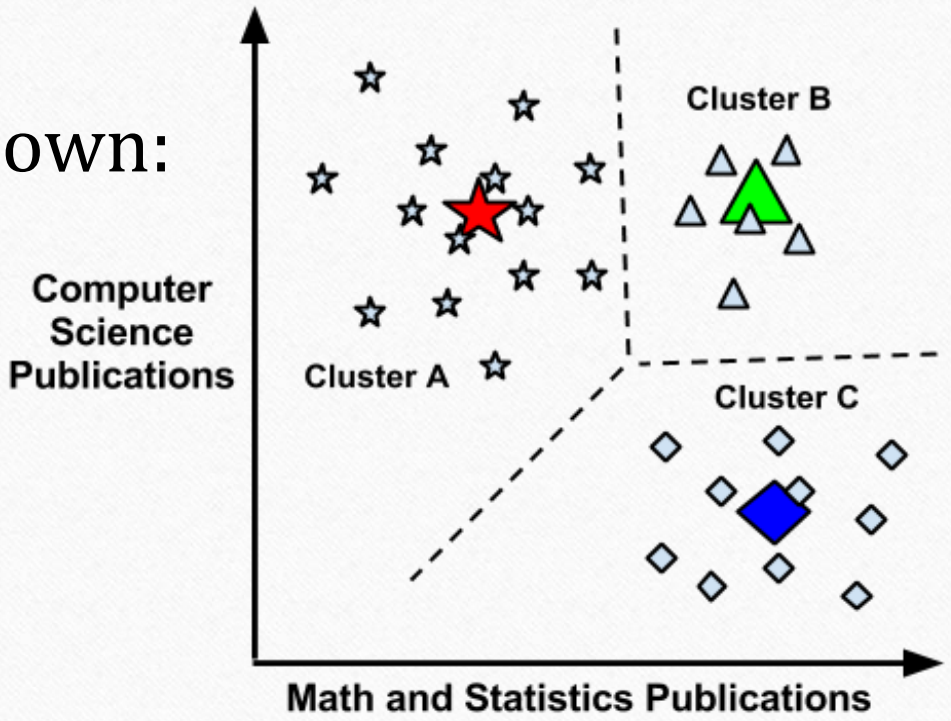
- The first step of updating the clusters involves shifting the initial centers to a new location, known as the centroid, which is calculated as the mean value of the points currently assigned to that cluster.
- The following figure illustrates how the cluster centers shift to the new centroids:



- Cluster A is able to claim an additional example from Cluster B (indicated by an arrow).
- Because of this reassignment, the k-means algorithm will continue through another update phase. After recalculating the centroids for the clusters, the figure looks like this:



- Two more points have been reassigned from Cluster B to Cluster A during this phase, as they are now closer to the centroid for A than B.
- This leads to another update as shown:



Algorithm: k-means

Algorithm: k-Means
Input: Dataset *D* consists of *n* samples with *p* features
Output: *k* number of clusters

Procedure:

Step1: Select randomly k number of initial means with dimension *p*

Step2: Calculate distance between each data point and cluster means

Step3: Assign the data point to the cluster mean whose distance from the cluster mean is minimum of all cluster centroids

Step4: Recalculate the new centroid

Step5: Recalculate the distance between each data point and new centroids

Step6: Repeat from step3 until no change of data points in every clusters

Algorithm: k-means

Algorithm: *k*-means. The *k*-means algorithm for partitioning, where each cluster's center is represented by the mean value of the objects in the cluster.

Input:

- *k*: the number of clusters,
- *D*: a data set containing *n* objects.

Output: A set of *k* clusters.

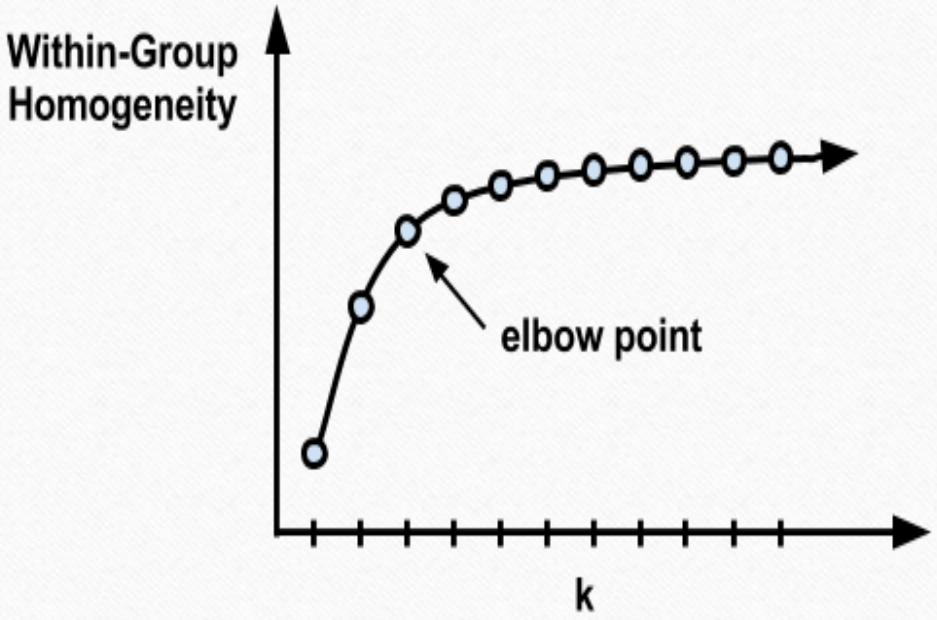
Method:

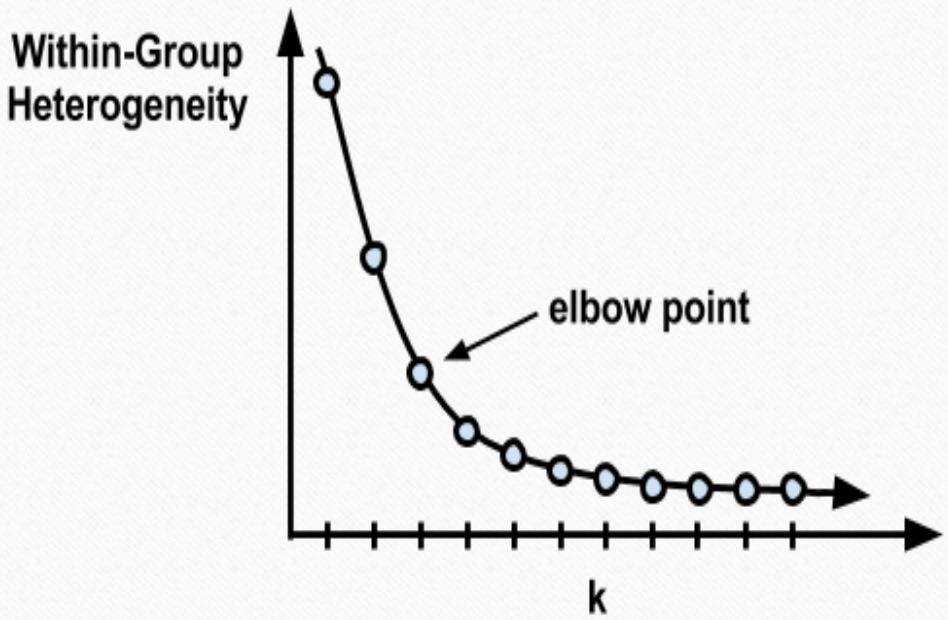
(1) arbitrarily choose *k* objects from *D* as the initial cluster centers;
(2) **repeat**
(3) (re)assign each object to the cluster to which the object is the most similar, based on the mean value of the objects in the cluster;
(4) update the cluster means, i.e., calculate the mean value of the objects for each cluster;
(5) **until** no change;

Choosing the appropriate number of clusters

- Ideally, we need to have some priori knowledge (that is, a prior belief) about the true groupings.
- In the data science conference seating problem, k might reflect the number of academic fields of study that were invited.
- In the case of no priori knowledge at all, one rule suggests setting k is equal to the square root of (n/2), where n is the number of examples in the dataset.

- A technique known as the **elbow** method attempts to gauge how the homogeneity or heterogeneity within the clusters changes for various values of k.
- As illustrated in the following figures, the homogeneity within clusters is expected to increase as additional clusters are added; similarly, heterogeneity will also continue to decrease with more clusters.
- We could continue to see improvements until each example is in its own cluster, the goal is not to maximize homogeneity or minimize heterogeneity, but rather to find k such that there are diminishing returns beyond that point. This value of k is known as the elbow point, because it looks like an elbow.


Within-Group
Homogeneity
elbow point
k


Within-Group
Heterogeneity
elbow point
k

The *k-Means Method….*

Example

- Problem: 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: $\rho(a, b) = |x2 - x1| + |y2 - y1|$.

The *k-Means Method….*

- Solution:
- The initial cluster centers – means, are (2, 10), (5, 8) and (1, 2), chosen randomly. Next, calculate the distance from the first point (2, 10) to each of the three means, by using the distance function:

point	mean1
$x1, y1$	$x2, y2$
(2, 10)	(2, 10)

$$\rho(a, b) = |x2 - x1| + |y2 - y1|$$

$$\begin{aligned}\rho(point, mean1) &= |x2 - x1| + |y2 - y1| \\ &= |2 - 2| + |10 - 10| \\ &= 0 + 0 \\ &= 0\end{aligned}$$

point	mean2
$x1, y1$	$x2, y2$
(2, 10)	(5, 8)

$$\rho(a, b) = |x2 - x1| + |y2 - y1|$$

$$\begin{aligned}\rho(point, mean2) &= |x2 - x1| + |y2 - y1| \\ &= |5 - 2| + |8 - 10| \\ &= 3 + 2 \\ &= 5\end{aligned}$$

point	mean3
$x1, y1$	$x2, y2$
(2, 10)	(1, 2)

$$\rho(a, b) = |x2 - x1| + |y2 - y1|$$

$$\begin{aligned}\rho(point, mean2) &= |x2 - x1| + |y2 - y1| \\ &= |1 - 2| + |2 - 10| \\ &= 1 + 8 \\ &= 9\end{aligned}$$

The *k-Means Method….*

- So, in this way, we fill the values as below:-
  - Iteration1:

		(2, 10)	(5, 8)	(1, 2)	
	Point	**Dist Mean 1**	**Dist Mean 2**	**Dist Mean 3**	**Cluster**
A1	(2, 10)	0	5	9	1
A2	(2, 5)	5	6	4	3
A3	(8, 4)	12	7	9	2
A4	(5, 8)	5	0	10	2
A5	(7, 5)	10	5	9	2
A6	(6, 4)	10	5	7	2
A7	(1, 2)	9	10	0	3
A8	(4, 9)	3	2	10	2

Cluster 1
(2, 10)

Cluster 2
(8, 4)
(5, 8)
(7, 5)
(6, 4)
(4, 9)

Cluster 3
(2, 5)
(1, 2)

The *k-Means Method….*

- Next, we need to re-compute the new cluster centers (means) by taking the mean of all points in each cluster.
- For Cluster 1, have only one point A1(2, 10), which was the old mean, so the cluster center remains the same.
- For Cluster 2, we have ((8+5+7+6+4)/5, (4+8+5+4+9)/5) = (6, 6)
- For Cluster 3, we have ((2+1)/2, (5+2)/2) = (1.5, 3.5)

The *k-Means Method….*

- Next, we go to Iteration2 (epoch2), Iteration3, and so on until the means do not change anymore.
- In Iteration2, we basically repeat the process from Iteration1 using the new means we computed.
- After Iteration2, the result will be,

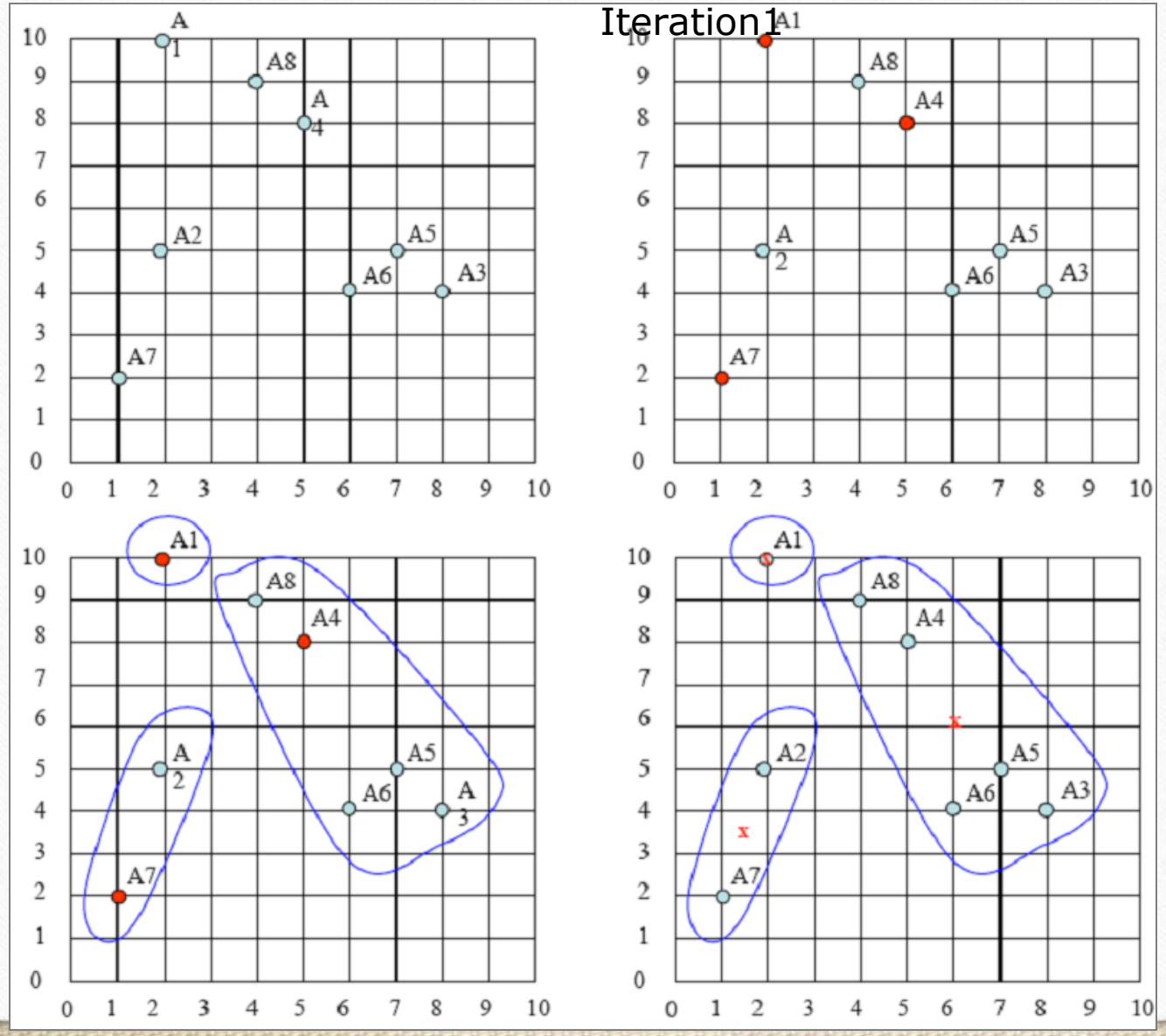
1: {A1, A8}, 2: {A3, A4, A5, A6}, 3: {A2, A7}
with centers C1=(3, 9.5), C2=(6.5, 5.25) and C3=(1.5, 3.5)

- After Iteration3, the result will be,

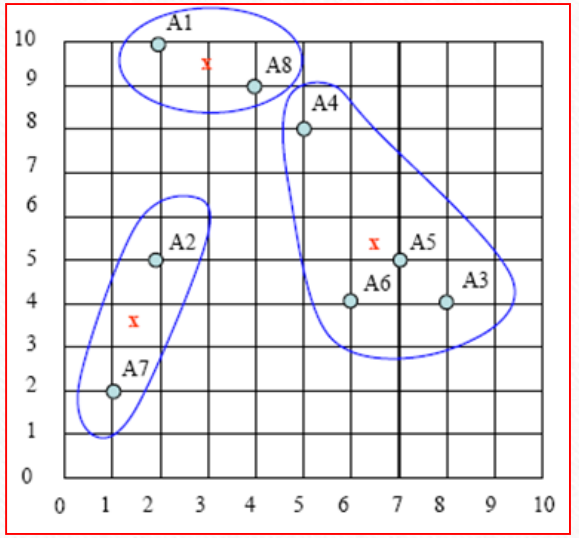
1: {A1, A4, A8}, 2: {A3, A5, A6}, 3: {A2, A7}
with centers C1=(3.66, 9), C2=(7, 4.33) and C3=(1.5, 3.5)

The *k-Means Method….*

Iteration1



Iteration2



Iteration3