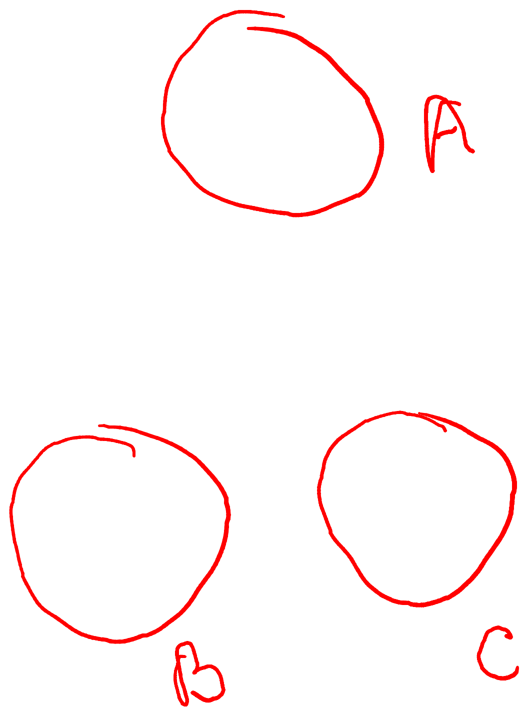
Importance of Choosing Intial Centroids

- Depending on the choice of initial centroids, B and C may get merged or remain separate



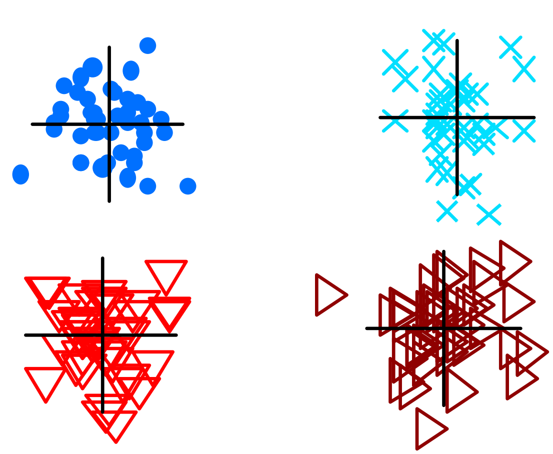
Problems with Selecting Initial Points

- If there are K 'real' clusters then the chance of selecting one centroid from each cluster is small.
  - Chance is relatively small when K is large
  - If clusters are the same size, n, then

$$P = \frac{\text{number of ways to select one centroid from each cluster}}{\text{number of ways to select } K \text{ centroids}} = \frac{K! n^K}{(Kn)^K} = \frac{K!}{K^K}$$

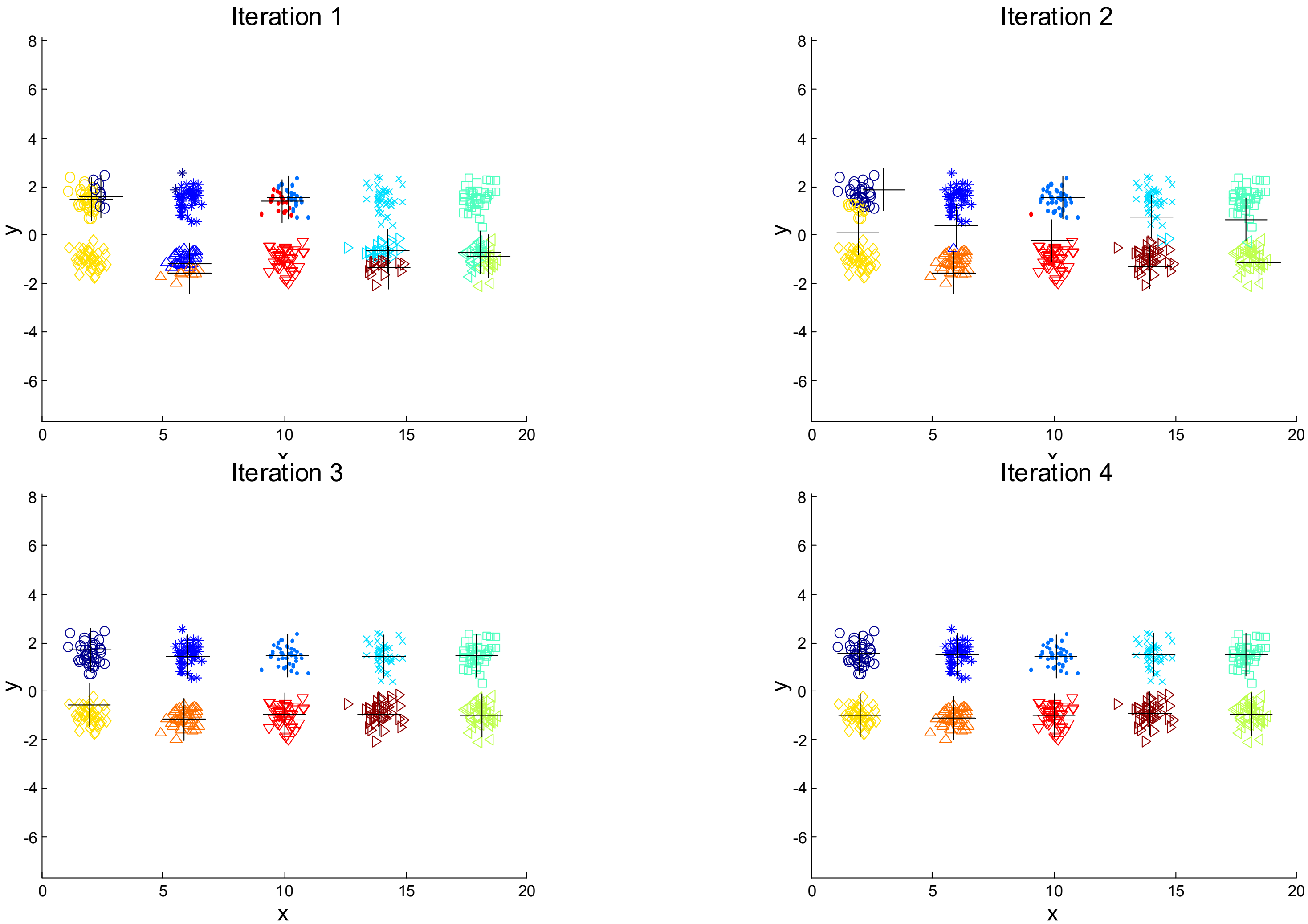
  - For example, if K = 10, then probability = $10!/10^{10}$ = 0.00036
  - Sometimes the initial centroids will readjust themselves in 'right' way, and sometimes they don't
  - Consider an example of five pairs of clusters

10 Clusters Example



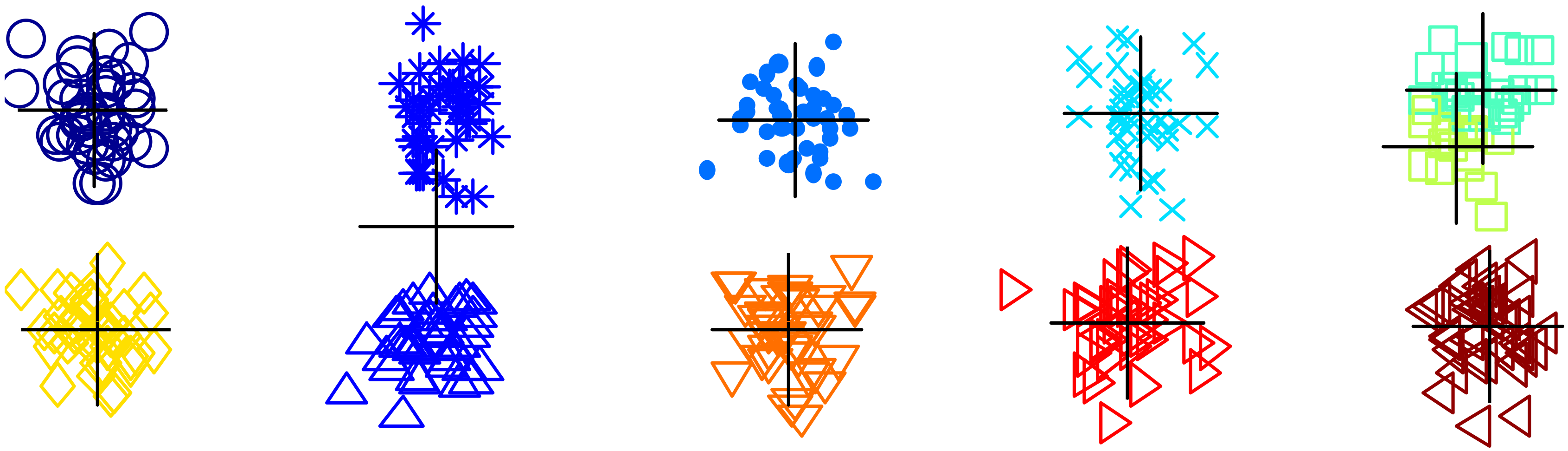
Starting with two initial centroids in one cluster of each pair of clusters

10 Clusters Example



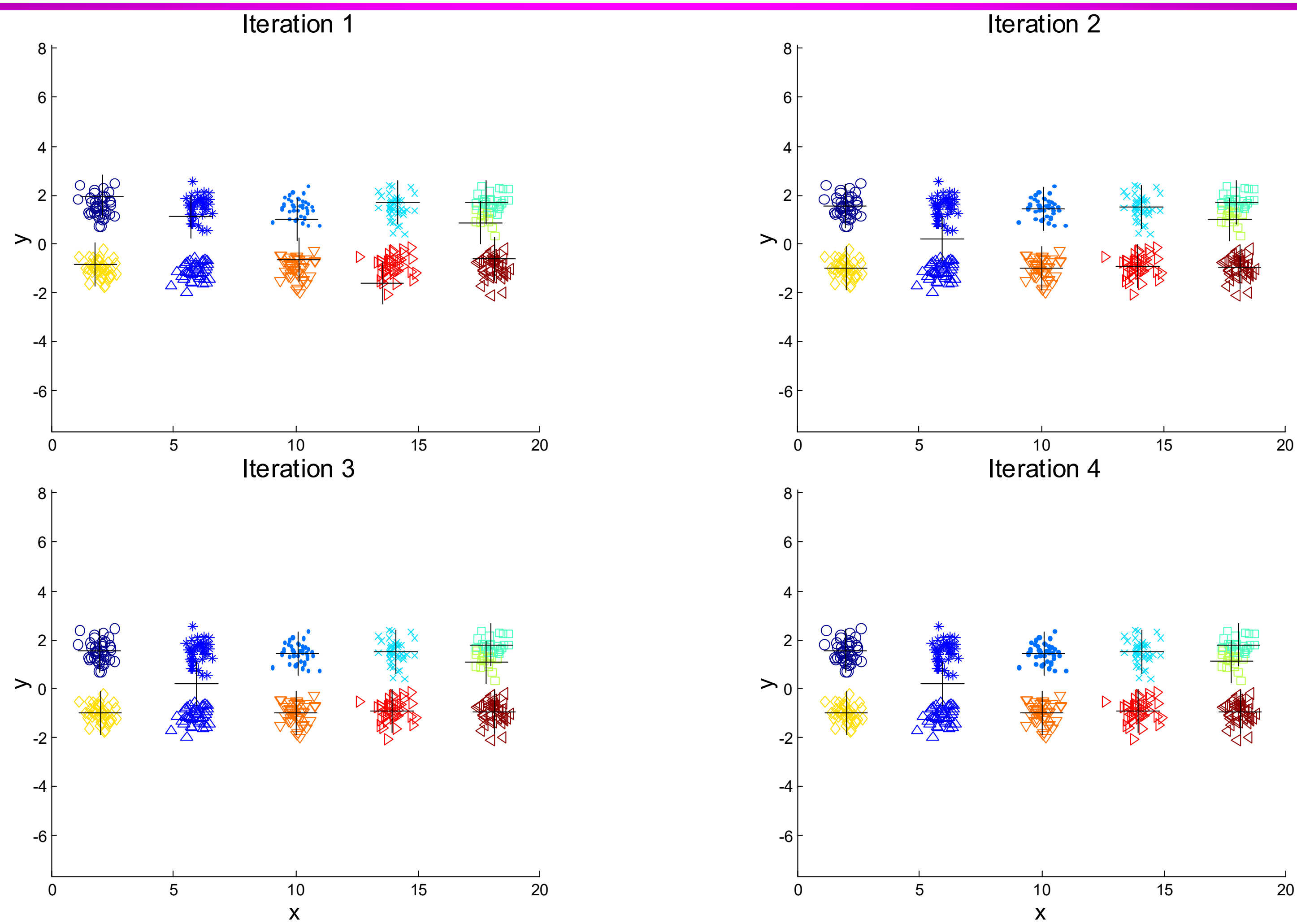
Starting with two initial centroids in one cluster of each pair of clusters

10 Clusters Example



Starting with some pairs of clusters having three initial centroids, while other have only one.

10 Clusters Example



Starting with some pairs of clusters having three initial centroids, while other have only one.

Solutions to Initial Centroids Problem

- Multiple runs
  - Helps, but probability is not on your side
- Use some strategy to select the k initial centroids and then select among these initial centroids
  - Select most widely separated
    - K-means++ is a robust way of doing this selection
  - Use hierarchical clustering to determine initial centroids
- Bisecting K-means
  - Not as susceptible to initialization issues

دو نیم کردن K
به مسائل اولیه سازی حساس نیست

K-means++

- This approach can be slower than random initialization, but very consistently produces better results in terms of SSE
  - The k-means++ algorithm guarantees an approximation ratio O(log k) in expectation, where k is the number of centers
- To select a set of initial centroids, *C*, perform the following

---

Algorithm 5.2 K-means++ initialization algorithm.

1: For the first centroid, pick one of the points at random.
2: **for** $i = 1$ to *number of trials* **do**
3: Compute the distance, $d(x)$, of each point to its closest centroid.
4: Assign each point a probability proportional to each point's $\boxed{d(x)^2}$.
5: Pick new centroid from the remaining points using the weighted probabilities.
6: **end for**

Bisecting K-means

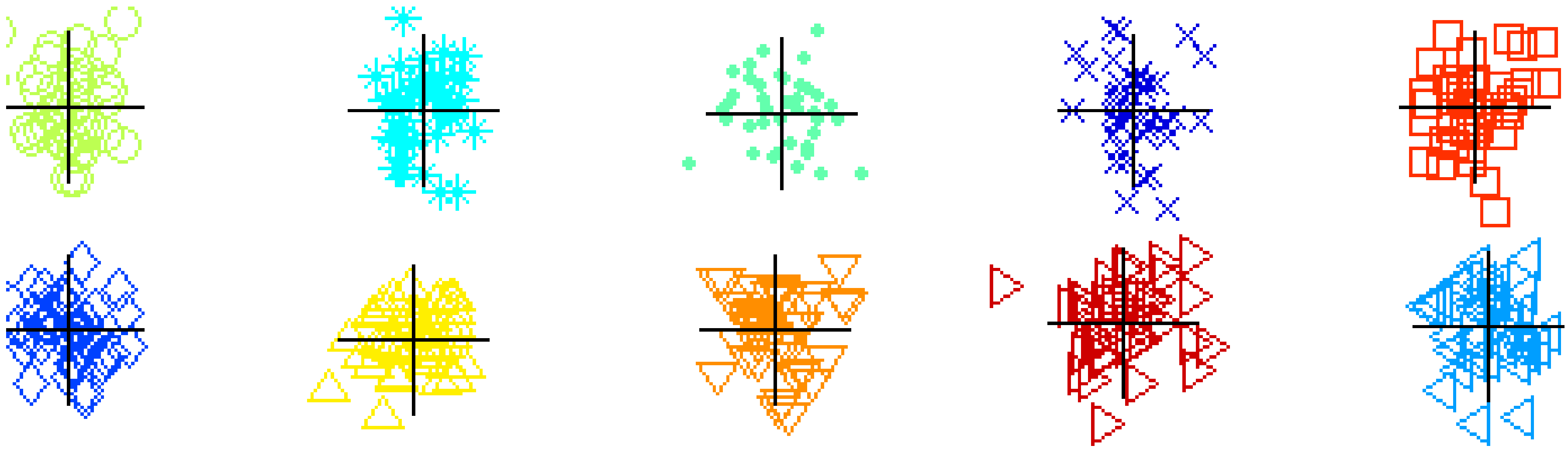
- Bisecting K-means algorithm
  - Variant of K-means that can produce a partitional or a hierarchical clustering

---

Initialize the list of clusters to contain the cluster containing all points.
repeat
  Select a cluster from the list of clusters
  for $i = 1$ to *number_of_iterations* **do**
    Bisect the selected cluster using basic K-means
  end for
  Add the two clusters from the bisection with the lowest SSE to the list of clusters.
until Until the list of clusters contains K clusters

CLUTO: http://glaros.dtc.umn.edu/gkhome/cluto/cluto/overview

Bisecting K-means Example



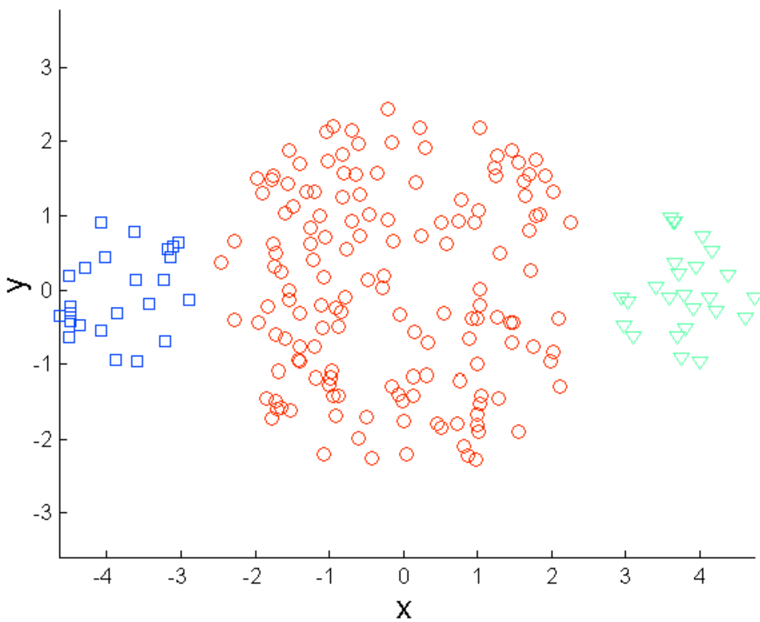
Limitations of K-means

- K-means has problems when clusters are of differing
  - Sizes
  - Densities
  - Non-globular shapes

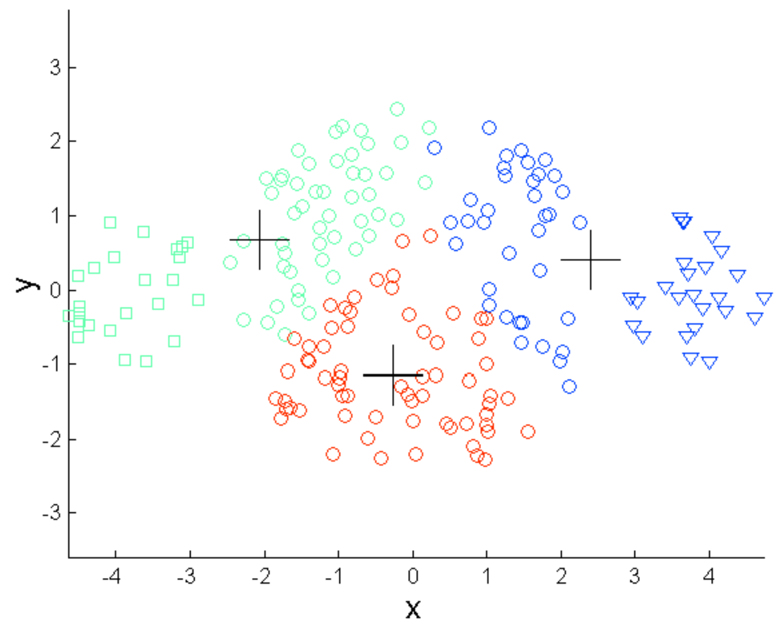
محدودیت های K-means
K-means زمانی که خوشه ها متفاوت هستند مشکل دارد
- اندازه ها
- تراکم ها
- اشکال غیر کروی
هنگامی که داده ها حاوی مقادیر پرت باشد، K-means مشکل دارد.
- یک راه حل ممکن حذف نقاط پرت قبل از خوشه بندی است

- K-means has problems when the data contains outliers.
  - One possible solution is to remove outliers before clustering

Limitations of K-means: Differing Sizes



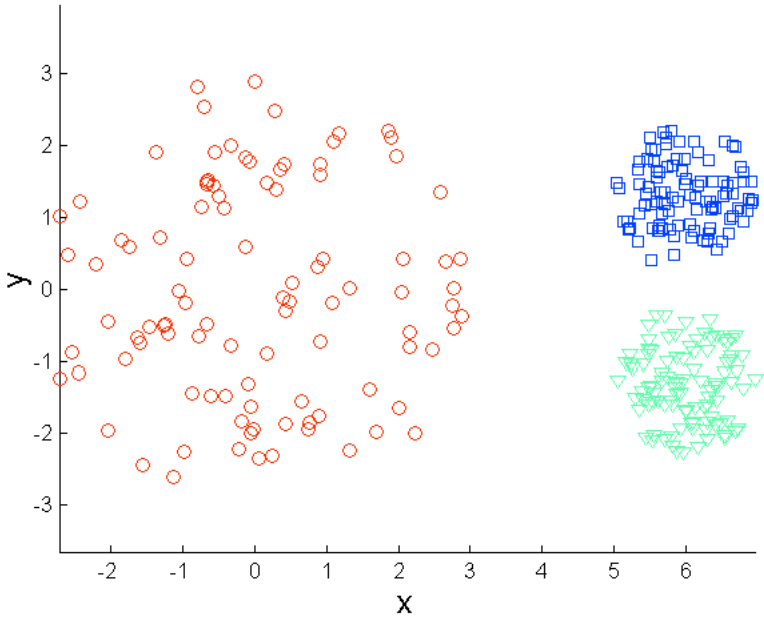
Original Points



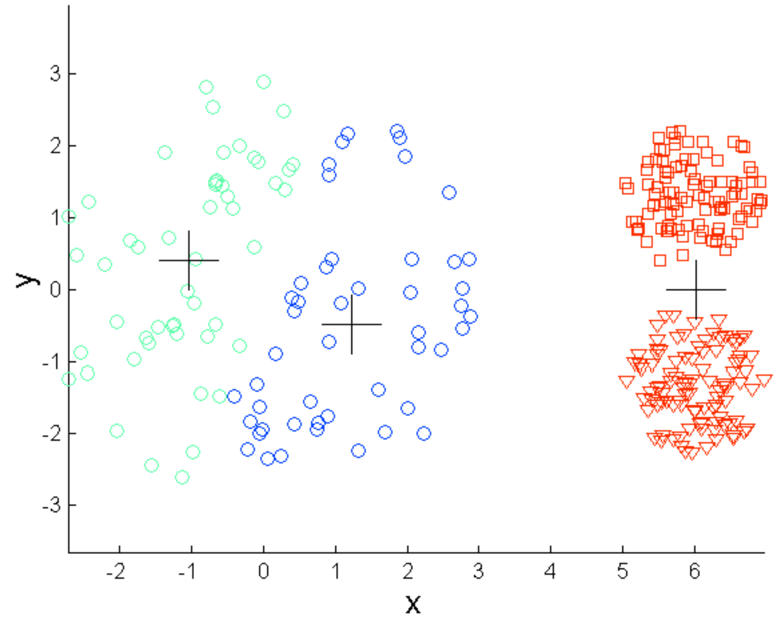
K-means (3 Clusters)

Limitations of K-means: Differing Density

چگالی: تجمع نقاط حول میانگین



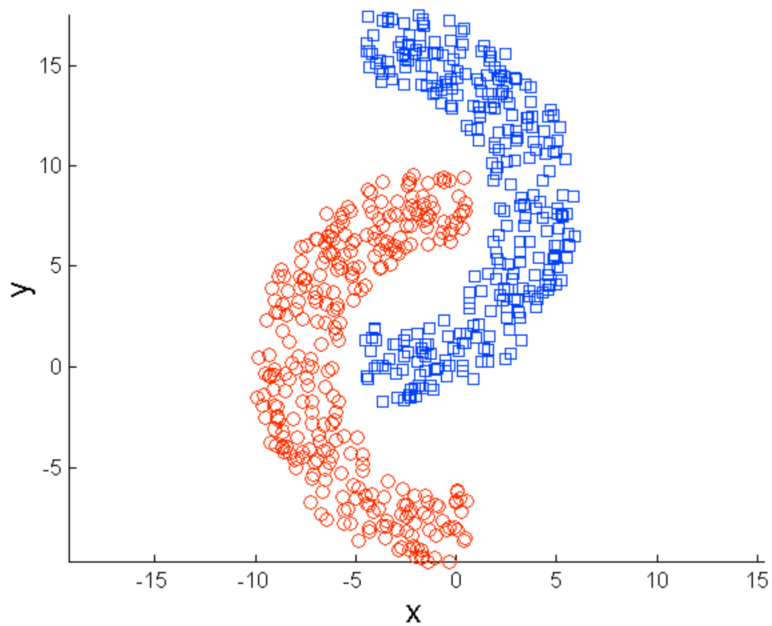
Original Points



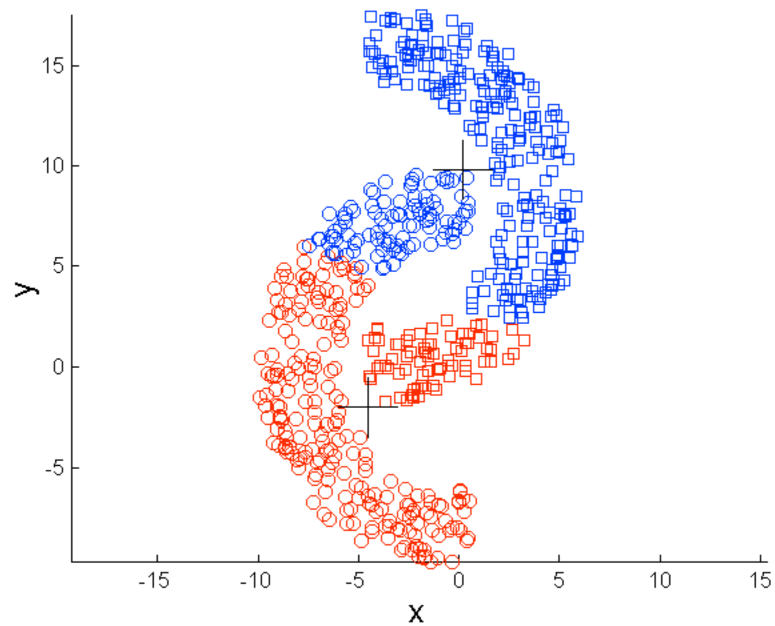
K-means (3 Clusters)

محدودیت های K-means: اشکال غیر کروی

Limitations of K-means: Non-globular Shapes

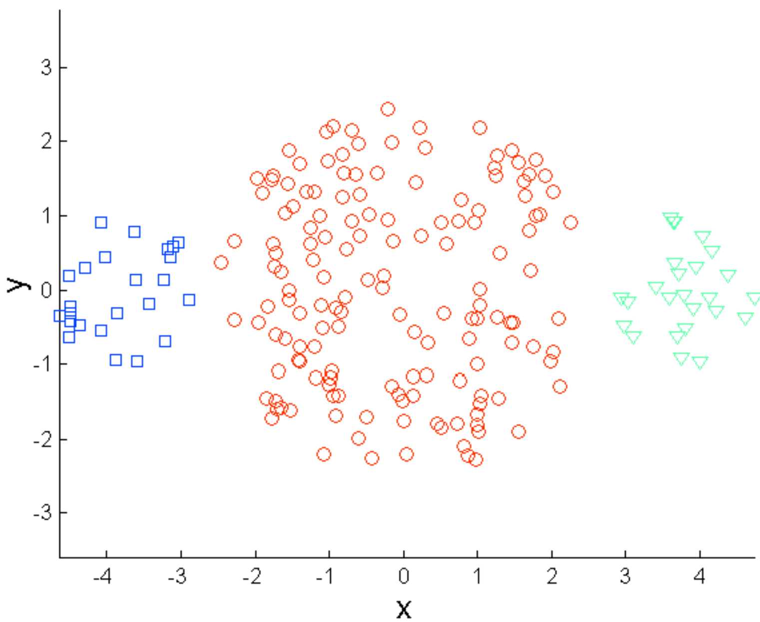


Original Points

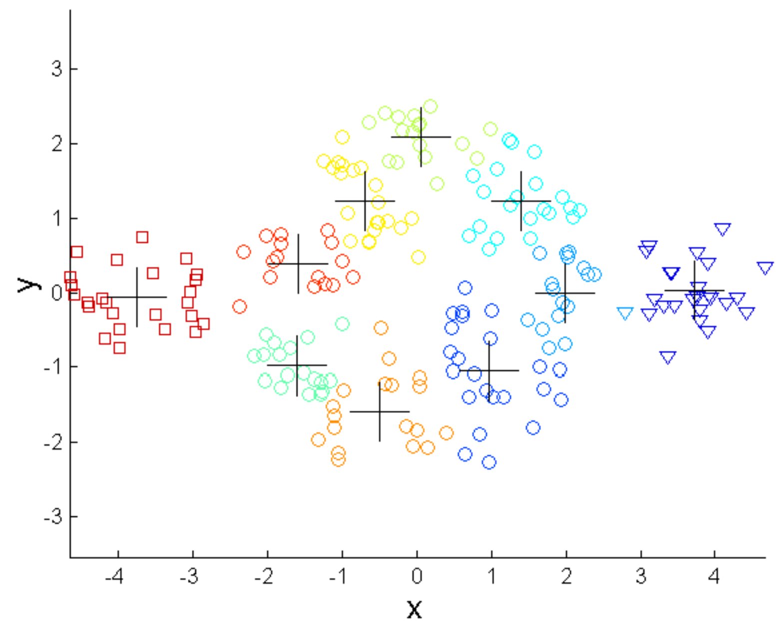


K-means (2 Clusters)

Overcoming K-means Limitations



Original Points

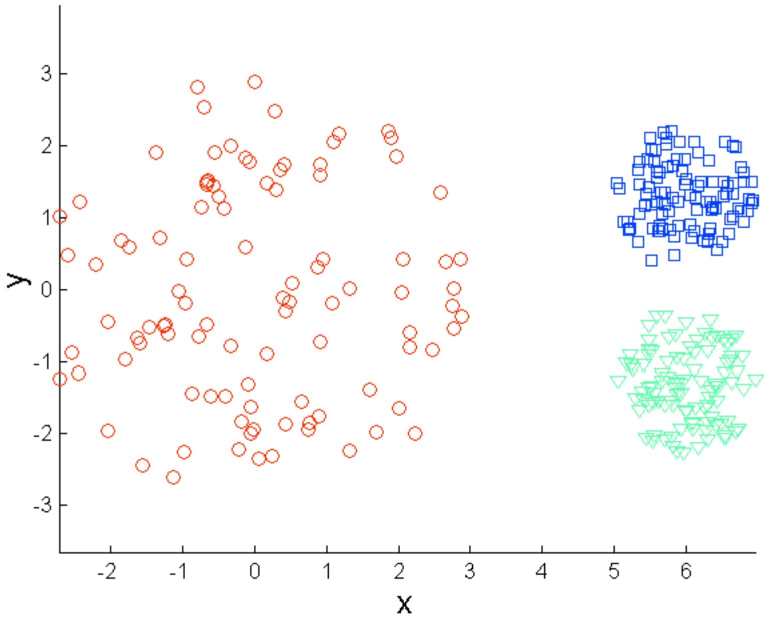


K-means Clusters

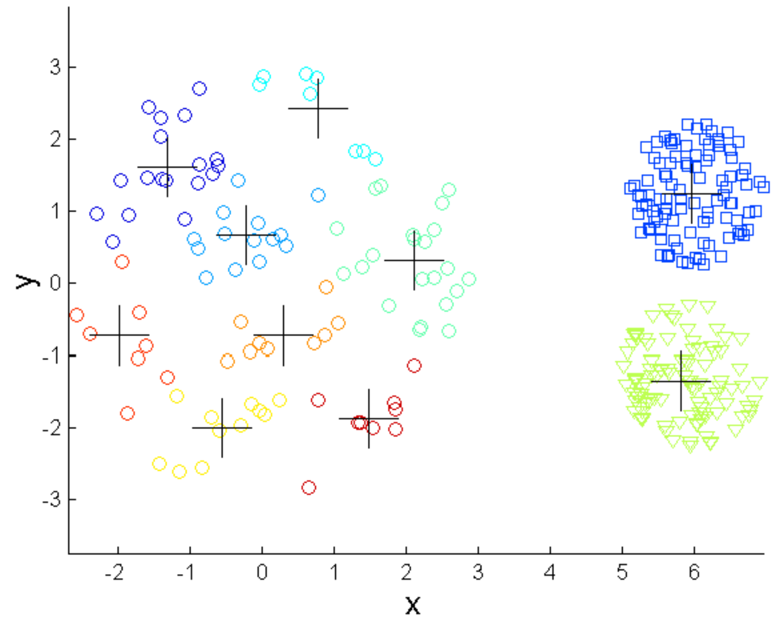
One solution is to find a large number of clusters such that each of them represents a part of a natural cluster. But these small clusters need to be put together in a post-processing step.

یک راه حل این است که تعداد زیادی خوشه پیدا کنیم به طوری که هر یک از آنها بخشی از یک خوشه طبیعی را نشان دهد. اما این خوشه های کوچک باید در یک مرحله پس از پردازش کنار هم قرار گیرند.

Overcoming K-means Limitations



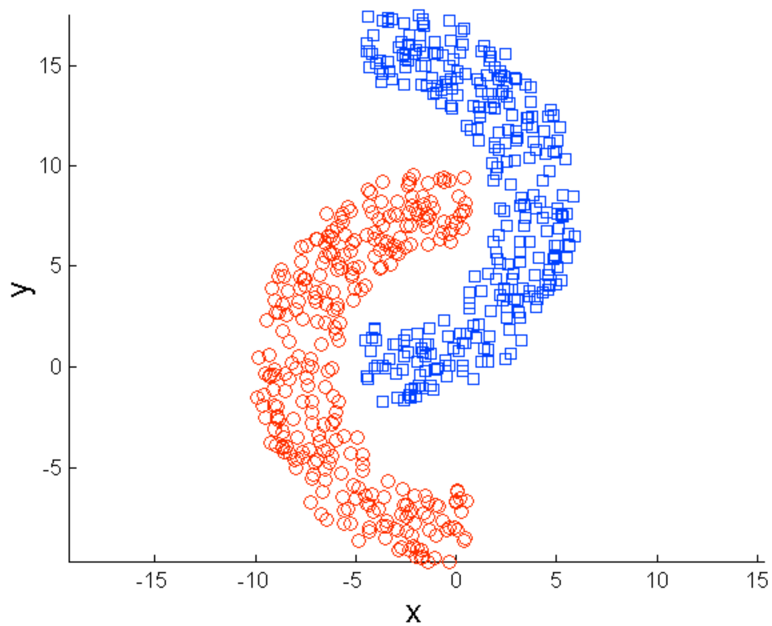
Original Points



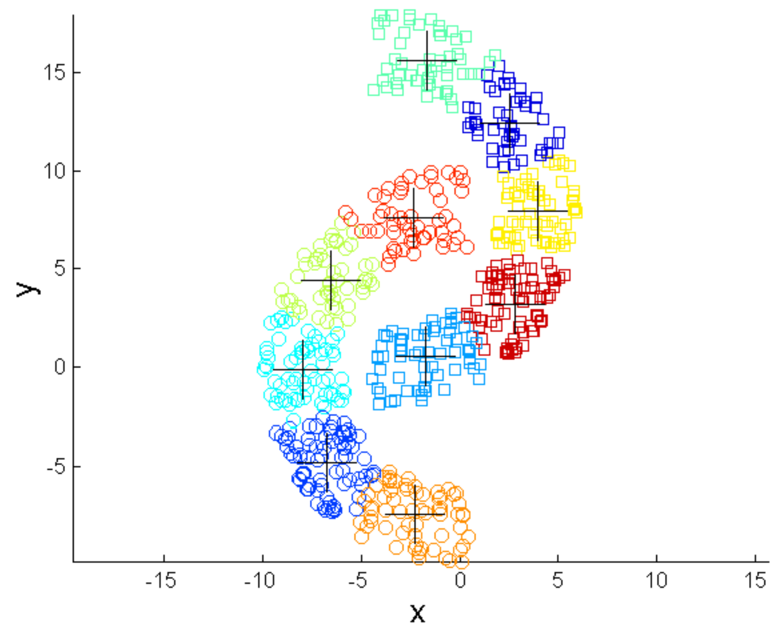
K-means Clusters

One solution is to find a large number of clusters such that each of them represents a part of a natural cluster. But these small clusters need to be put together in a post-processing step.

Overcoming K-means Limitations



Original Points



K-means Clusters

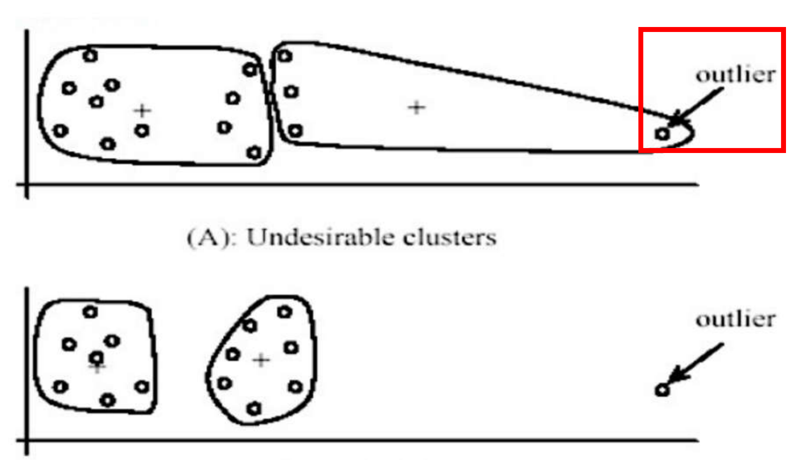
One solution is to find a large number of clusters such that each of them represents a part of a natural cluster. But these small clusters need to be put together in a post-processing step.

K-means and outlier!

- K-Medoids: Instead of taking the mean value of the object in a cluster as a reference point, medoids can be used, which is the most centrally located object in a cluster

K-Medoids: به جای در نظر گرفتن مقدار میانگین شی در یک خوشه به عنوان نقطه مرجع، می توان از medoids استفاده کرد که مرکزی ترین شی در یک خوشه است.

https://www.cese.nsw.gov.au/effective-practices/unit-4-outliers



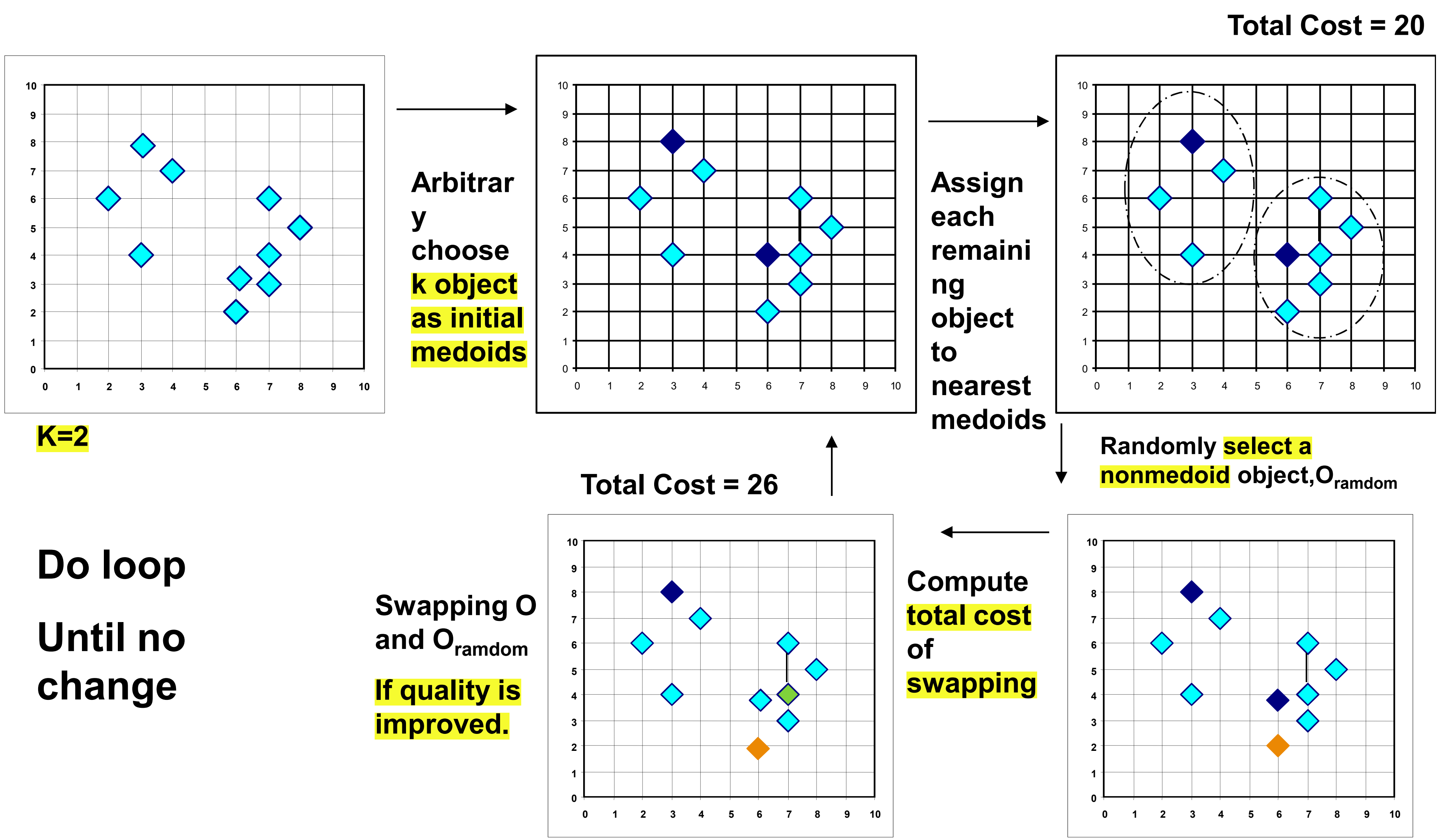
(B): Ideal clusters

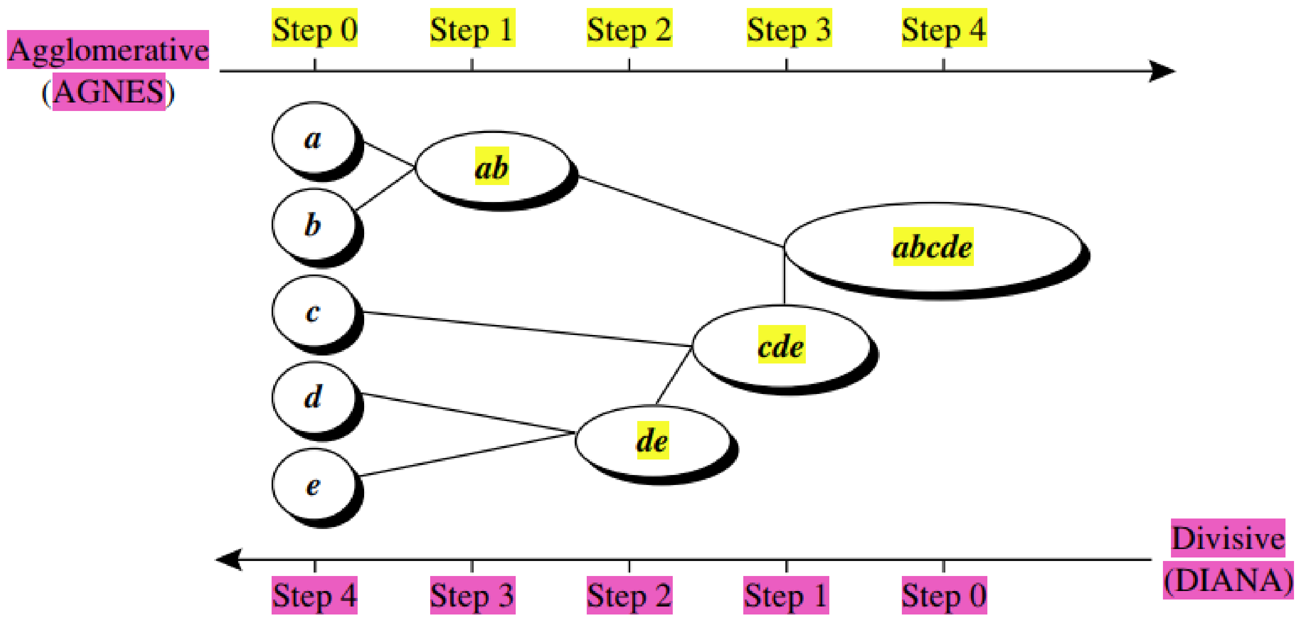
https://www.slideshare.net/anilyadav5055/15857-cse422-unsupervisedlearning

The K-Medoid Clustering Method

- *K-Medoids* Clustering: Find *representative* objects (medoids) in clusters
  - *PAM* (Partitioning Around Medoids, Kaufmann & Rousseeuw 1987)
    - Starts from an initial set of medoids and iteratively replaces one of the medoids by one of the non-medoids if it improves the total distance of the resulting clustering
    - *PAM* works effectively for small data sets, but does not scale well for large data sets (due to the computational complexity)
- Efficiency improvement on PAM
  - *CLARA* (Kaufmann & Rousseeuw, 1990): PAM on samples
  - *CLARANS* (Ng & Han, 1994): Randomized re-sampling

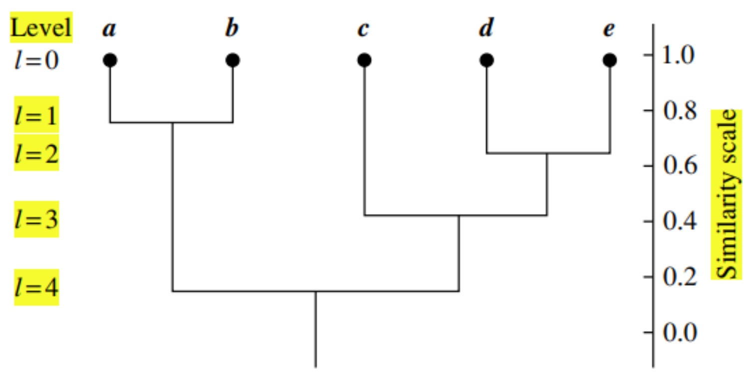
PAM: A Typical K-Medoids Algorithm





HIERARCHICAL CLUSTERING

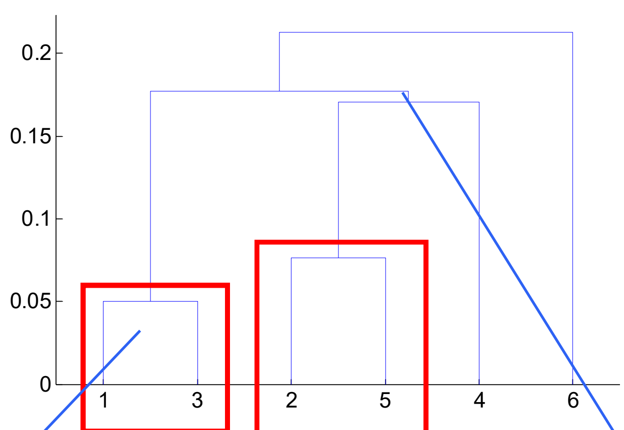
Agglomerative and divisive hierarchical clustering on data objects $\{a, b, c, d, e\}$.

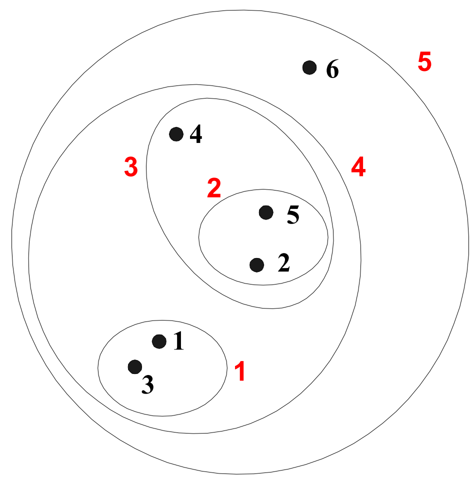


Hierarchical Clustering

- Produces a set of nested clusters organized as a hierarchical tree
- Can be visualized as a dendrogram
  - A tree like diagram that records the sequences of merges or splits

مجموعه ای از خوشه های تو در تو را
که به صورت درخت سلسله مراتبی
سازماندهی شده اند تولید می کند.
یک نمودار مانند درخت که دنباله های
ادغام یا تقسیم را ثبت می کند





فاصله ی نمونه ی
۱و۳ برابر 0.05

خود نمونه ها روی محور x
فاصله نمونه ها از هم روی محور y

فاصله ی نمونه ی ۴
با نمونه های ۲و۵
برابر 0.16

Strengths of Hierarchical Clustering

- Do not have to assume any particular number of clusters
  - Any desired number of clusters can be obtained by ‘cutting’ the dendrogram at the proper level

- They may correspond to meaningful taxonomies
  - Example in biological sciences (e.g., animal kingdom, phylogeny reconstruction, …)

نقاط قوت خوشه بندی سلسله مراتبی
مجبور نیستید تعداد خاصی از خوشه ها را فرض کنید
هر تعداد خوشه دلخواه را می توان با "برش" دندروگرام در سطح مناسب به دست آورد.
آنها ممکن است با طبقه بندی های معنی دار مطابقت داشته باشند
مثال در علوم زیستی (مانند پادشاهی حیوانات، بازسازی فیلوژنی، ...)

Hierarchical Clustering

۲تا روش برای ایجاد
خوشه ها داریم:
۱. از بالا به پایین
۲. از پایین به بالا

- Two main types of hierarchical clustering
  - Agglomerative:
    - Start with the points as individual clusters
    - At each step, merge the closest pair of clusters until only one cluster (or k clusters) left
  - Divisive:
    - Start with one, all-inclusive cluster
    - At each step, split a cluster until each cluster contains an individual point (or there are k clusters)

دو نوع اصلی خوشه بندی سلسله مراتبی
- تجمعی:
با نقاط به عنوان خوشه های فردی شروع کنید
در هر مرحله، نزدیکترین جفت خوشه ها را
ادغام کنید تا تنها یک خوشه (یا k خوشه) باقی
بماند

-. تفرقه افکن:
با یک خوشه فراگیر شروع کنید
در هر مرحله، یک خوشه را تا زمانی تقسیم کنید که هر خوشه دارای
یک نقطه جداگانه باشد (یا k خوشه وجود داشته باشد)

- Traditional hierarchical algorithms use a similarity or distance matrix
  - Merge or split one cluster at a time

الگوریتم های سلسله مراتبی سنتی از یک شباهت یا ماتریس فاصله استفاده می کنند
- ادغام یا تقسیم یک خوشه در یک زمان

Agglomerative Clustering Algorithm

Key Idea: Successively merge closest clusters

- Basic algorithm
  1. Compute the proximity matrix
  2. Let each data point be a cluster
  3. **Repeat**
  4. Merge the two closest clusters
  5. Update the proximity matrix
  6. **Until** only a single cluster remains

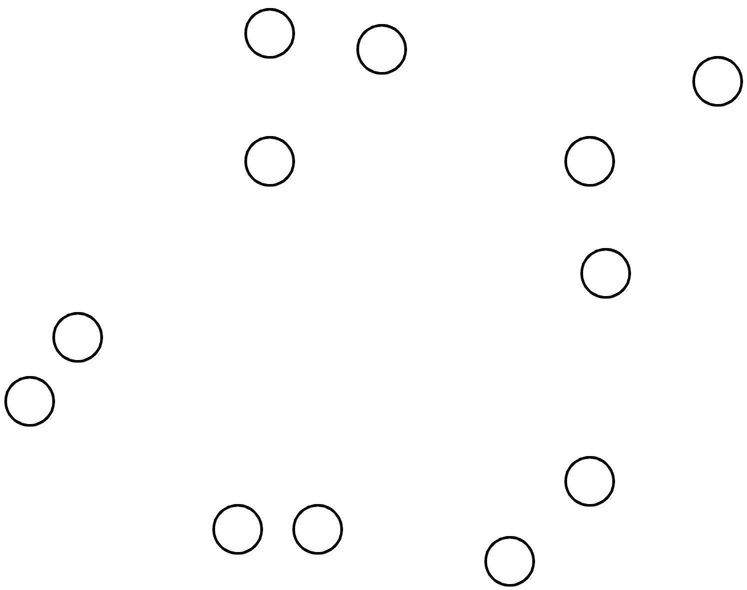
Key operation is the computation of the proximity of two clusters

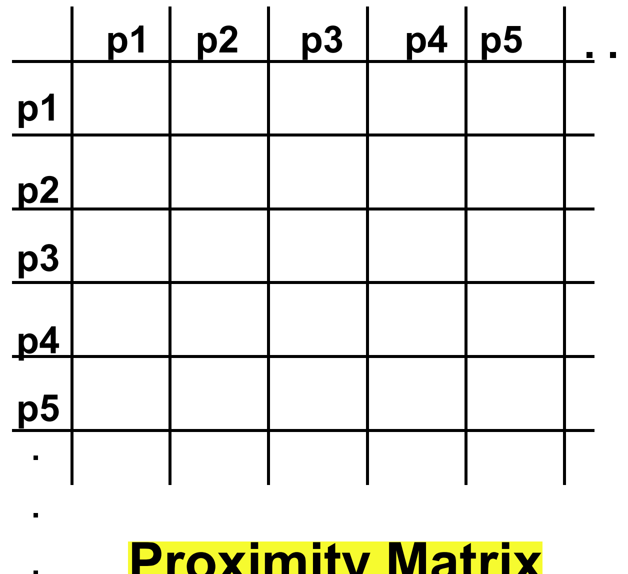
- Different approaches to defining the distance between clusters distinguish the different algorithms

نقاطی که نزدیک به هم هستند را مرج میکنیم توی یک خوشه پس باید فاصله ی بین هردونقطه را داشته باشیم.

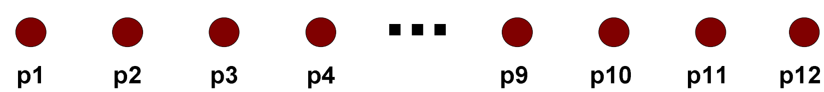
Steps 1 and 2

- Start with clusters of individual points and a proximity matrix



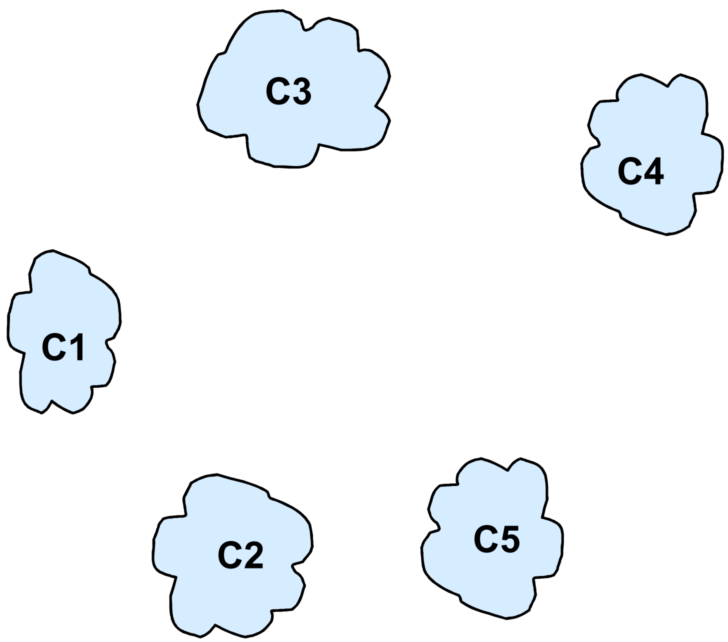


Proximity Matrix



Intermediate Situation

- After some merging steps, we have some clusters

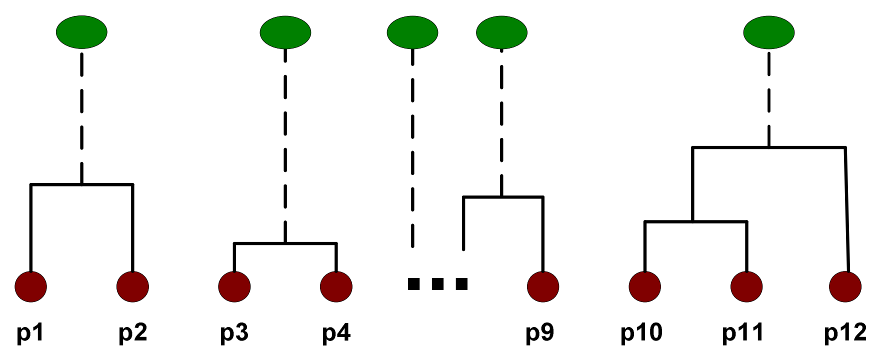


فاصله ی خوشه های
بدست اومده از هم

	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

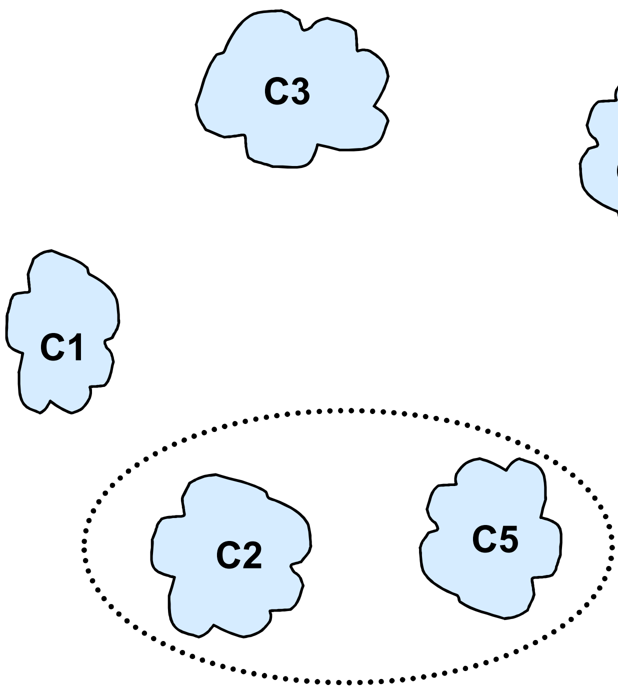
Proximity Matrix

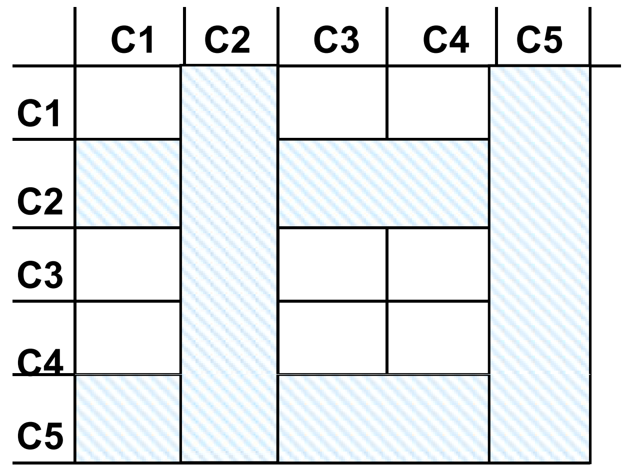
ماتریس مجاورت



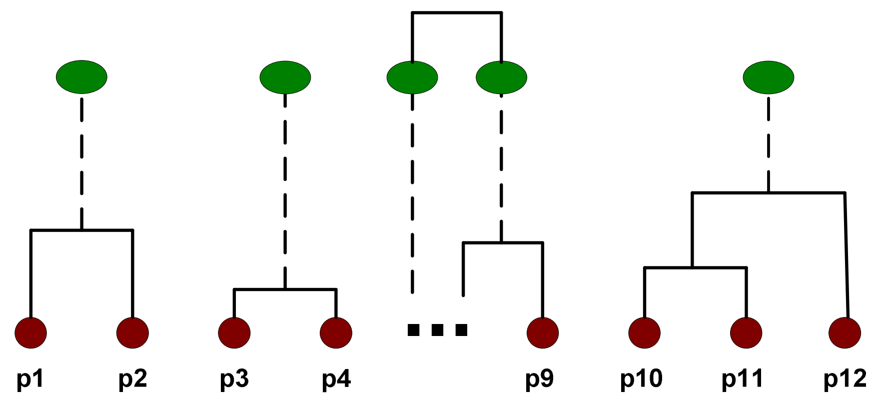
Step 4

- We want to merge the two closest clusters (C2 and C5) and update the proximity matrix.





Proximity Matrix

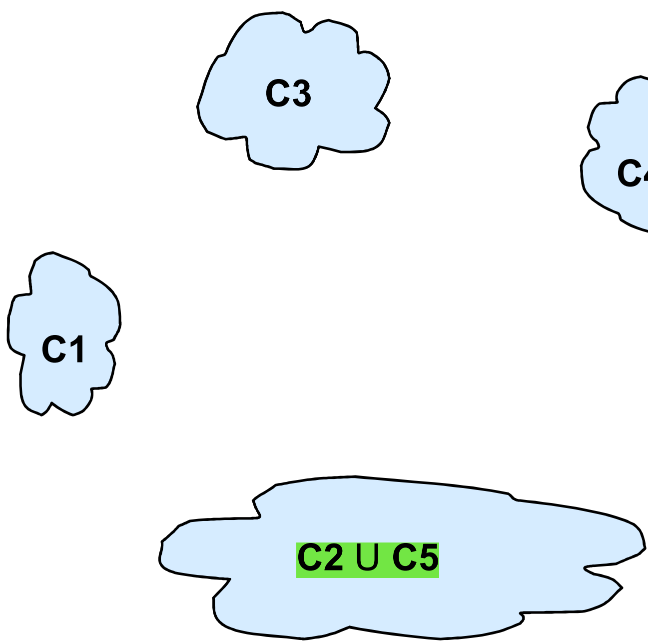


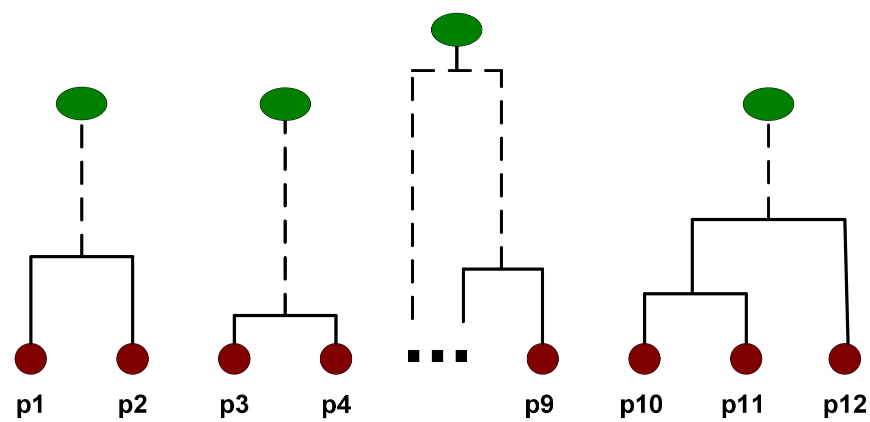
Step 5

- The question is "How do we update the proximity matrix?"

	C1	C2 U C5	C3	C4
C1		?		
C2 U C5	?	?	?	?
C3		?		
C4		?		

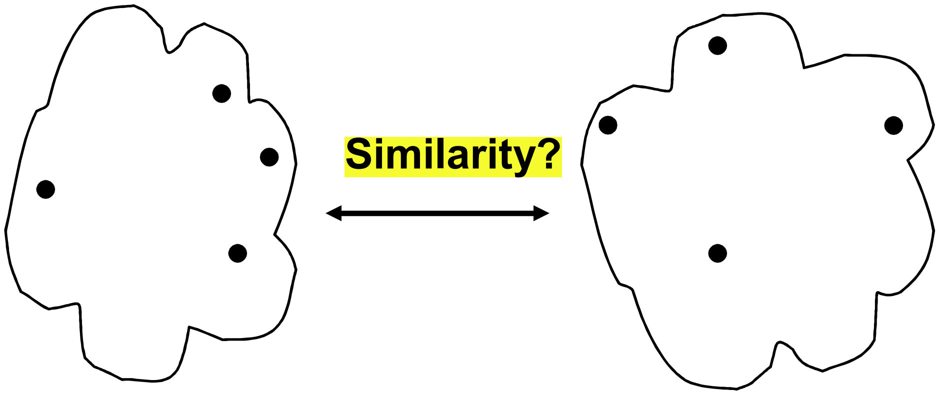
Proximity Matrix





نحوه تعریف فاصله بین خوشه ای

How to Define Inter-Cluster Distance

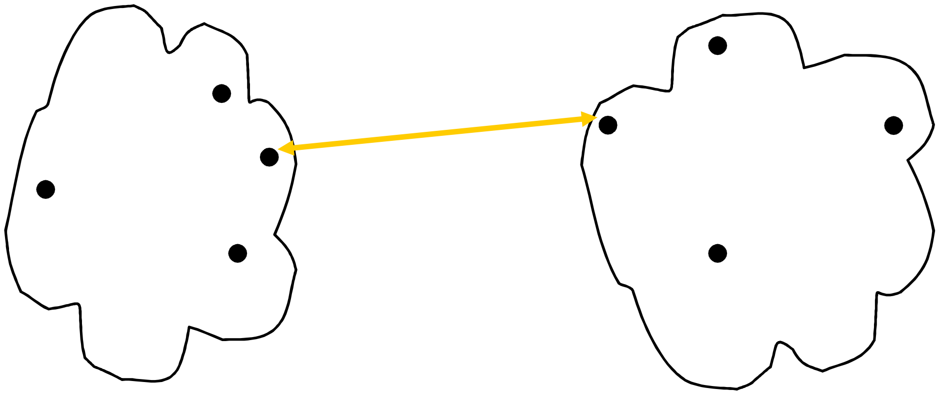


سوال: چطوری فاصله ی بین خوشه ها را اپدیت کنیم؟

	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

How to Define Inter-Cluster Similarity

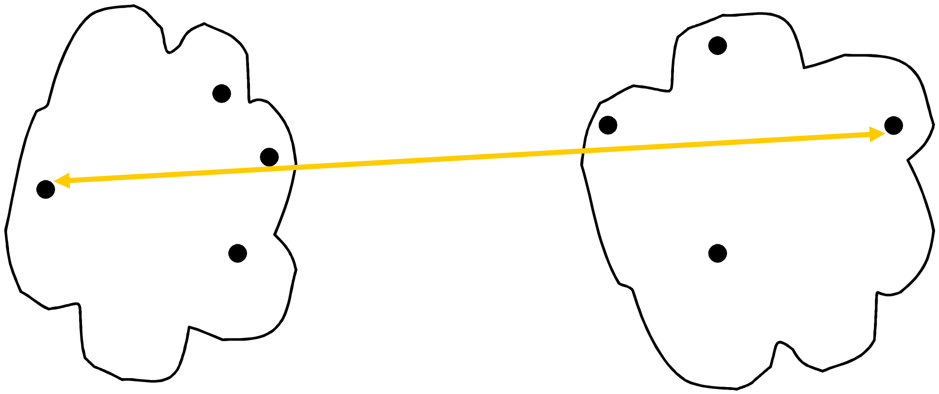


- MIN

	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

How to Define Inter-Cluster Similarity

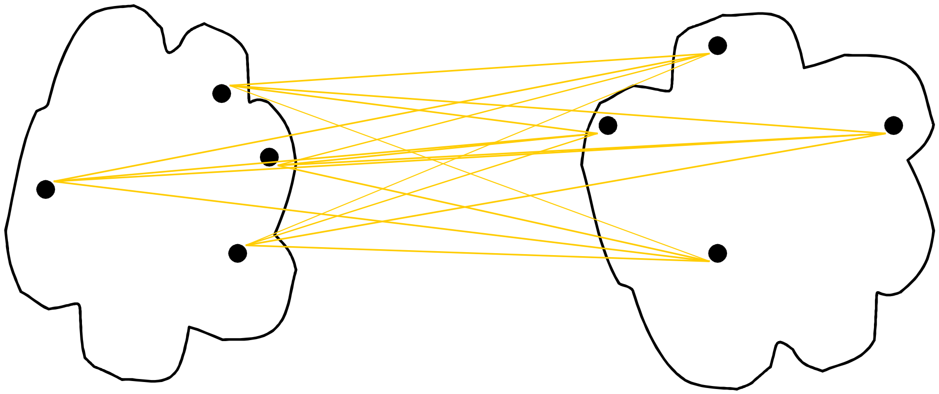


	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

- MIN
- MAX

How to Define Inter-Cluster Similarity

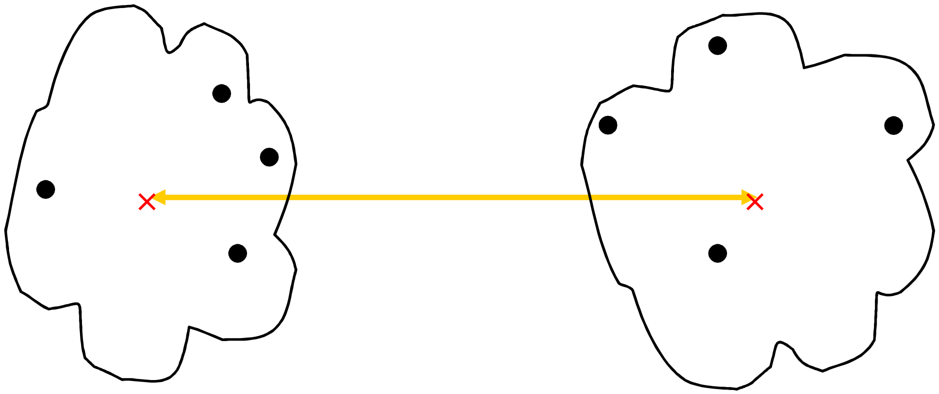


	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

- MIN
- MAX
- Group Average

How to Define Inter-Cluster Similarity



	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

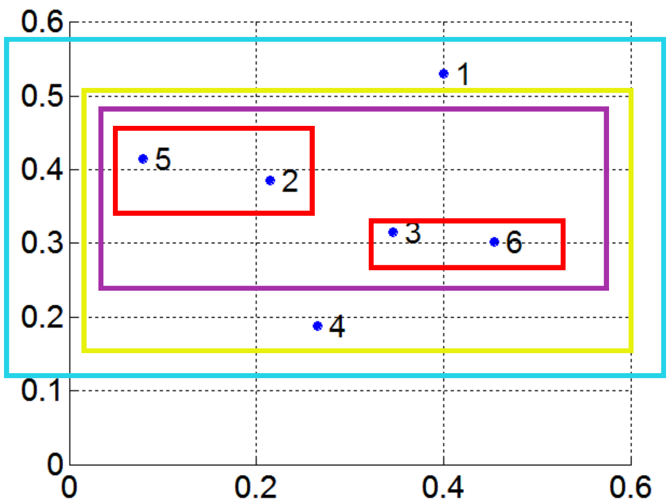
Proximity Matrix

- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward's Method uses squared error

MIN or Single Link

- Proximity of two clusters is based on the two closest points in the different clusters
  - Determined by one pair of points, i.e., by one link in the proximity graph
- Example:

مجاورت دو خوشه بر اساس دو نزدیکترین نقطه در خوشه های مختلف است.
- با یک جفت نقطه، به عنوان مثال، توسط یک پیوند در نمودار یا گراف مجاورت تعیین می شود

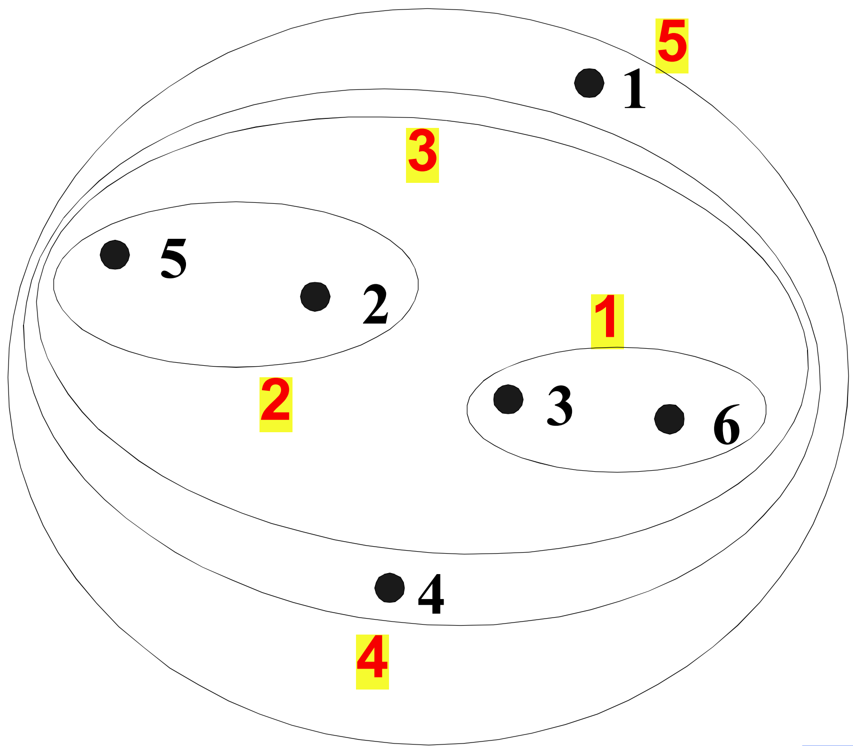


نقاط در یک فضای دو بعدی نمایش داده شدند

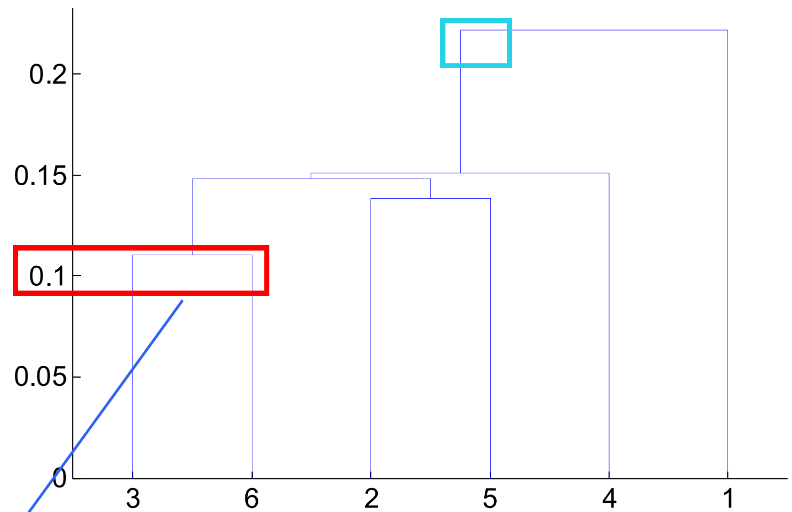
Distance Matrix:

	p1	p2	p3	p4	p5	p6
p1	0.00	0.24	0.22	0.37	0.34	0.23
p2	0.24	0.00	0.15	0.20	0.14	0.25
p3	0.22	0.15	0.00	0.15	0.28	0.11
p4	0.37	0.20	0.15	0.00	0.29	0.22
p5	0.34	0.14	0.28	0.29	0.00	0.39
p6	0.23	0.25	0.11	0.22	0.39	0.00

Hierarchical Clustering: MIN



فاصله ی نقطه ی ۱ از کلاستر
بدست امده 0.22 است.



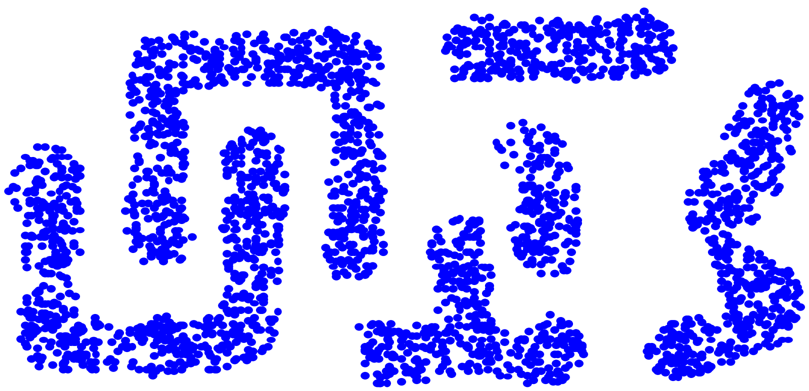
فاصله ی دونقطه ی ۳و۶
برابر 0.11 است.

Nested Clusters

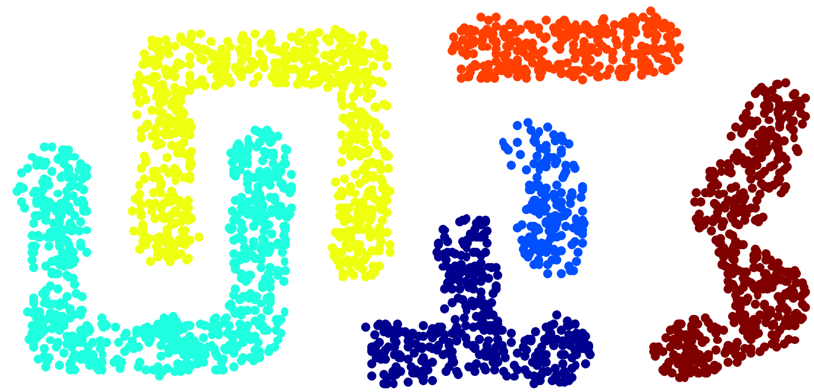
Dendrogram

دندروگرام:
یک نمودار درختی، به ویژه نموداری که
روابط طبقه بندی را نشان می دهد.

Strength of MIN



Original Points

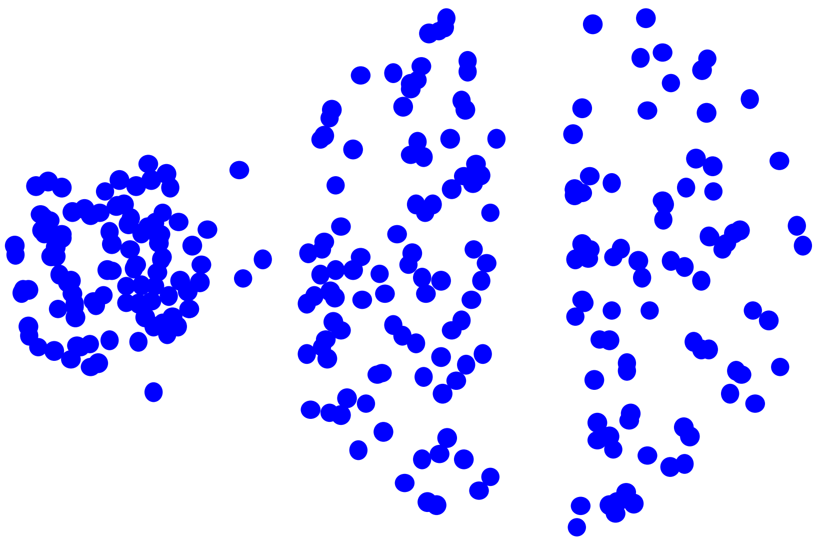


Six Clusters

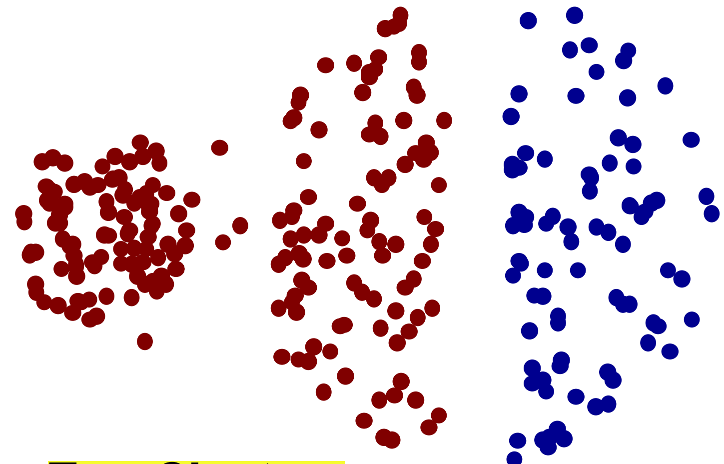
- Can handle non-elliptical shapes

• می تواند اشکال غیر بیضوی را اداره کند

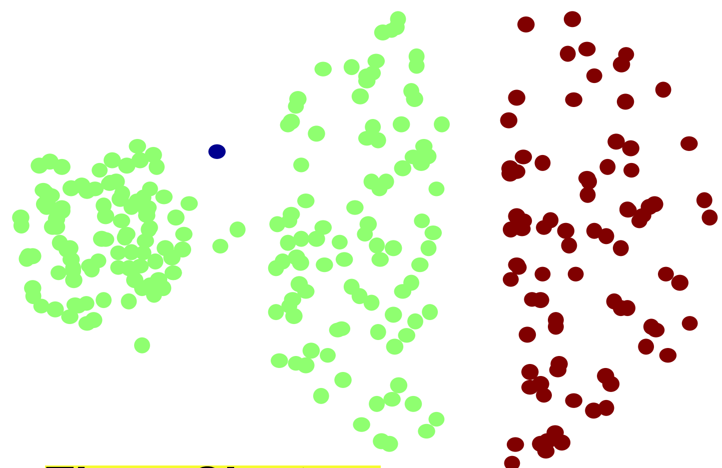
Limitations of MIN



Original Points



Two Clusters



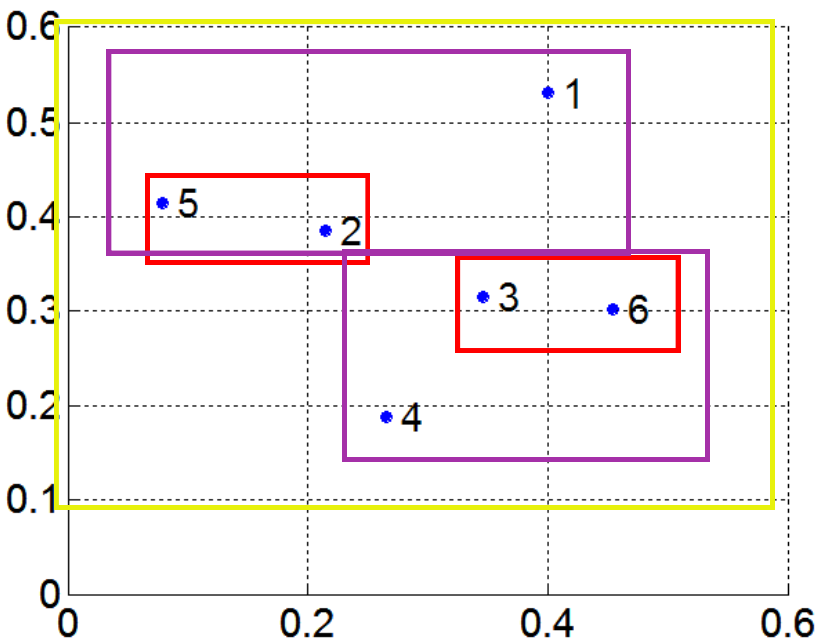
Three Clusters

- Sensitive to noise

MAX or Complete Linkage

مجاورت دو خوشه بر اساس دو دورترین نقطه در خوشه های مختلف است.
- توسط تمام جفت نقاط در دو خوشه تعیین می شود

- Proximity of two clusters is based on the two most distant points in the different clusters
  - Determined by all pairs of points in the two clusters



معیارمون برای پیداکردن نزدیک ترین خوشه ها، ماکس فاصله ی نقاط در دو خوشه است.

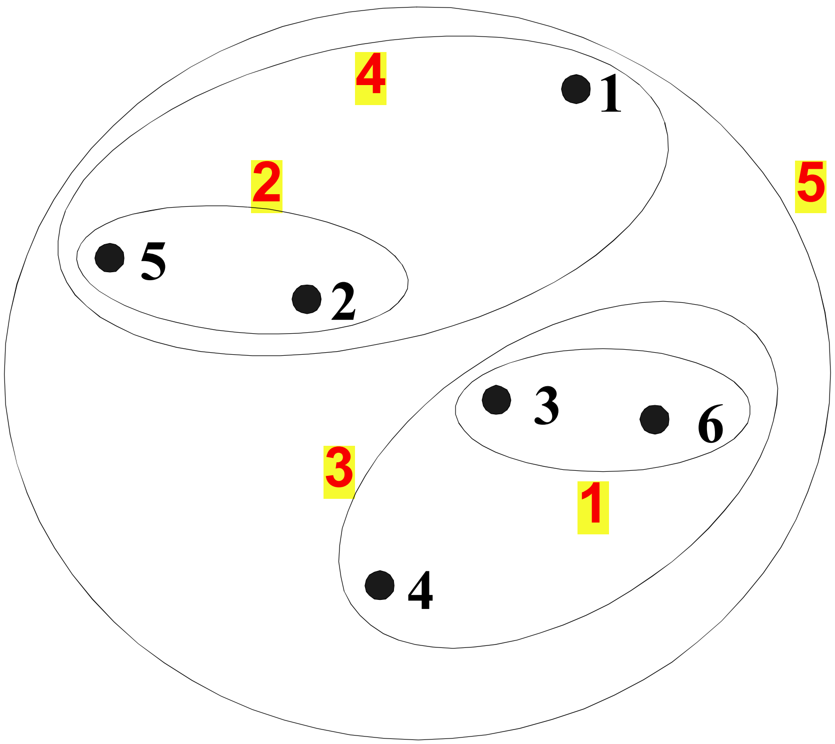
Distance Matrix:

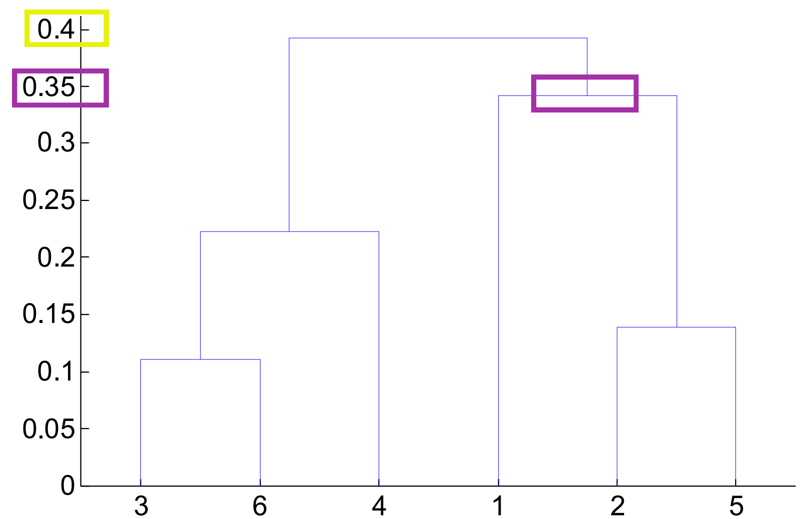
	p1	p2	p3	p4	p5	p6
p1	0.00	0.24	0.22	0.37	0.34	0.23
p2	0.24	0.00	0.15	0.20	0.14	0.25
p3	0.22	0.15	0.00	0.15	0.28	0.11
p4	0.37	0.20	0.15	0.00	0.29	0.22
p5	0.34	0.14	0.28	0.29	0.00	0.39
p6	0.23	0.25	0.11	0.22	0.39	0.00

ین دوتا خوشه ی بنفشی که درست شد، دو نقطه ی p5,p6 بیشترین فاصله را از هم دارند یعنی 0.39

در ابتدا که هیچ خوشه ای نداریم، نقاطی که کمترین فاصله از هم دارند را توی یک خوشه قرار میدهیم چون p2,p5 کمترین فاصله را از هم دارند پس یک خوشه میشوند.

Hierarchical Clustering: MAX

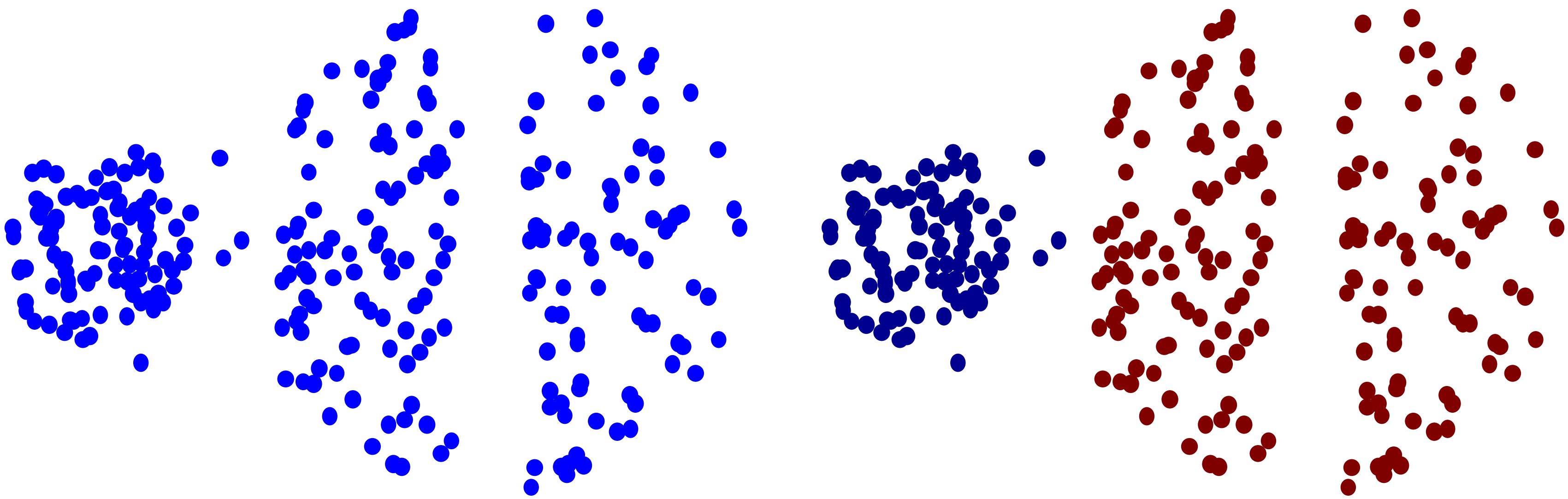




Nested Clusters

Dendrogram

Strength of MAX

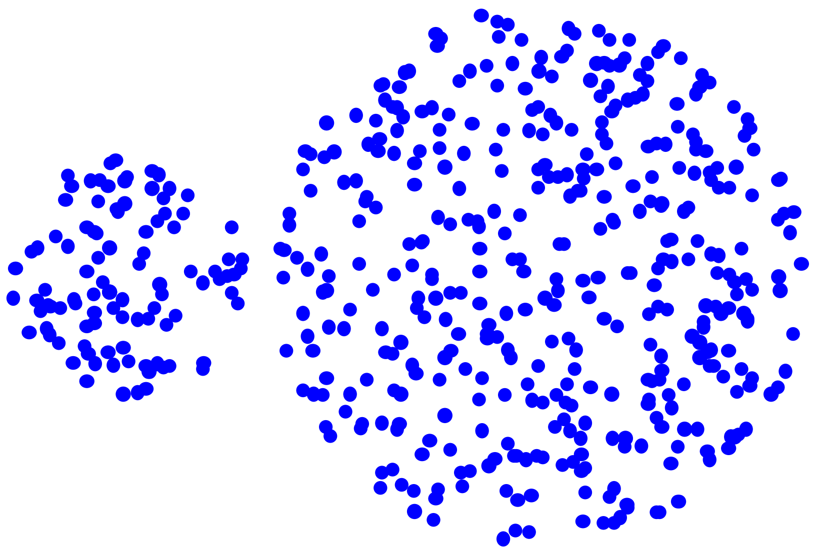


Original Points　　　　Two Clusters

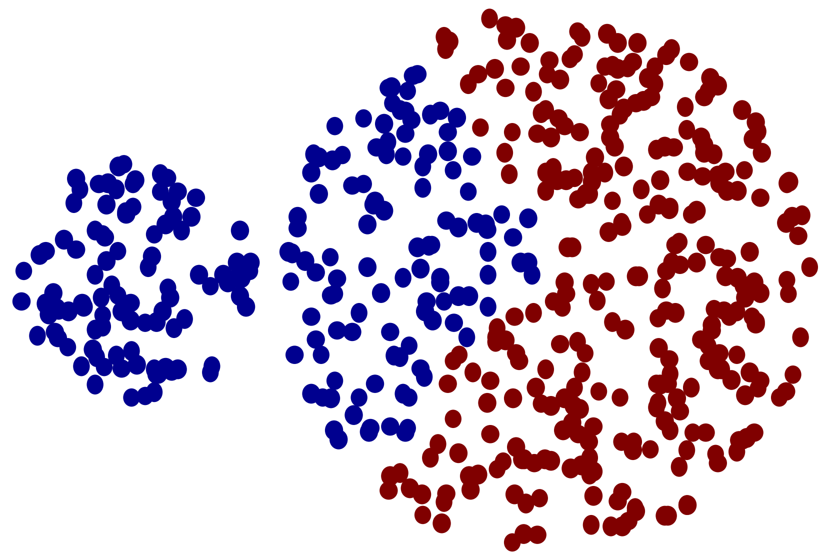
- **Less susceptible to noise**

کمتر مستعد نویز است

Limitations of MAX



Original Points



Two Clusters

- Tends to **break large clusters**
- **Biased** towards **globular clusters**

• تمایل به شکستن خوشه های بزرگ دارد
• گرایش به خوشه های کروی

Group Average

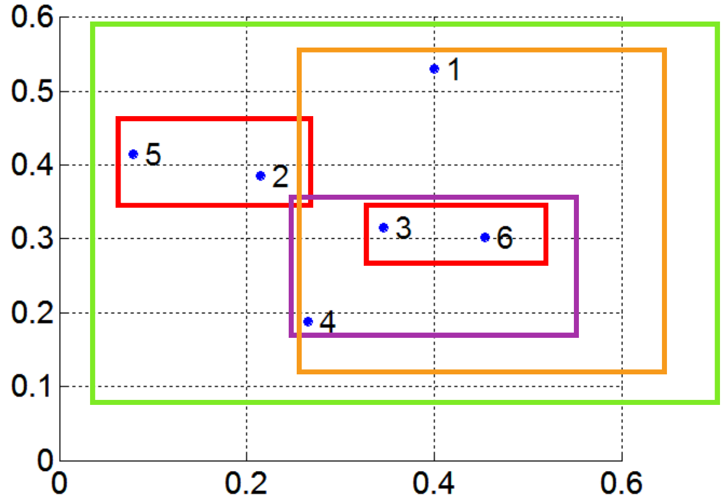
- Proximity of two clusters is the average of pairwise proximity between points in the two clusters.

مجموع فاصله ی همه ی نقاط در دو خوشه

مجاورت دو خوشه میانگین مجاورت زوجی بین نقاط دو خوشه است.

$$\text{proximity}(\text{Cluster}_i, \text{Cluster}_j) = \frac{\sum\limits_{\substack{p_i \in \text{Cluster}_i \\ p_j \in \text{Cluster}_j}} \text{proximity}(p_i, p_j)}{|\text{Cluster}_i| \times |\text{Cluster}_j|}$$

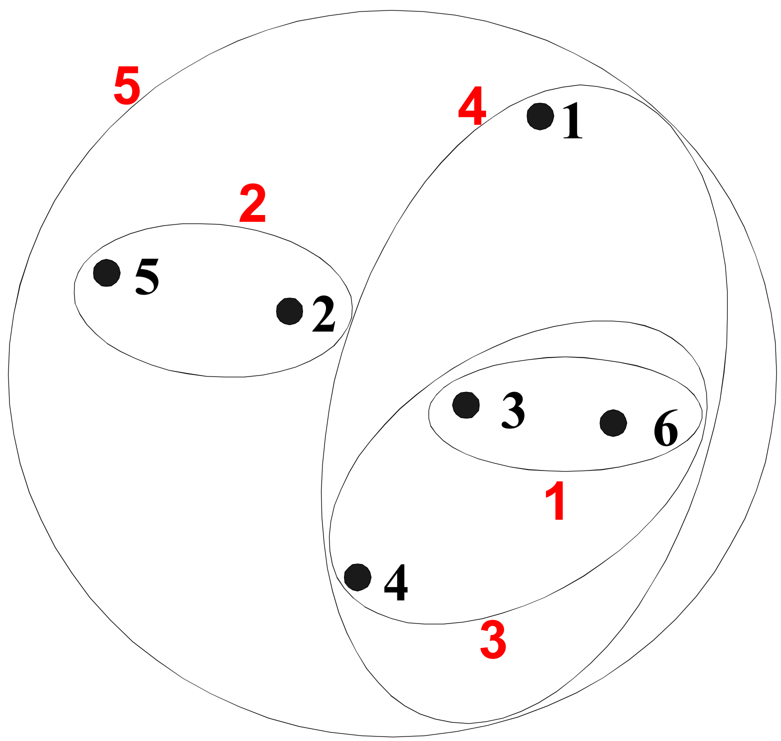
ضرب تعداد نقاط در دو خوشه



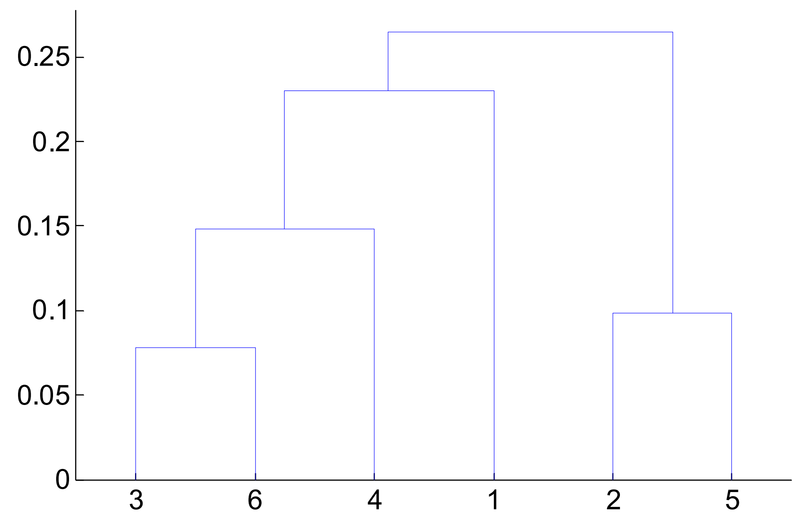
Distance Matrix:

	p1	p2	p3	p4	p5	p6
p1	0.00	0.24	0.22	0.37	0.34	0.23
p2	0.24	0.00	0.15	0.20	0.14	0.25
p3	0.22	0.15	0.00	0.15	0.28	0.11
p4	0.37	0.20	0.15	0.00	0.29	0.22
p5	0.34	0.14	0.28	0.29	0.00	0.39
p6	0.23	0.25	0.11	0.22	0.39	0.00

Hierarchical Clustering: Group Average



Nested Clusters



Dendrogram

Hierarchical Clustering: Group Average

- Compromise between Single and Complete Link

- Strengths
  - Less susceptible to noise

- Limitations
  - Biased towards globular clusters

سازش بین پیوند واحد و کامل
نقاط قوت
- کمتر مستعد نویز است
محدودیت ها
- گرایش به خوشه های کروی

Cluster Similarity: Ward's Method

- Similarity of two clusters is based on the

increase in squared error when two clusters are merged

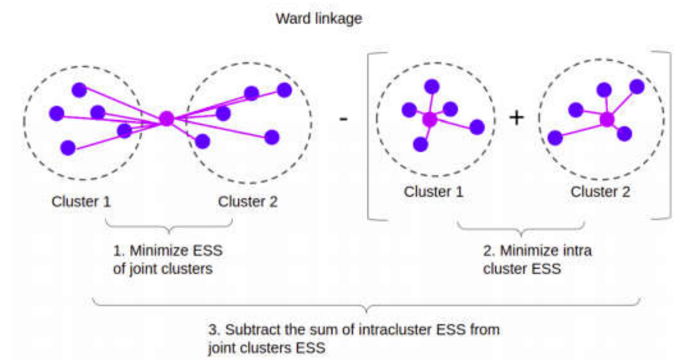
  – Similar to group average if distance between points is distance squared

- Less susceptible to noise

- Biased towards globular clusters

- Hierarchical analogue of K-means

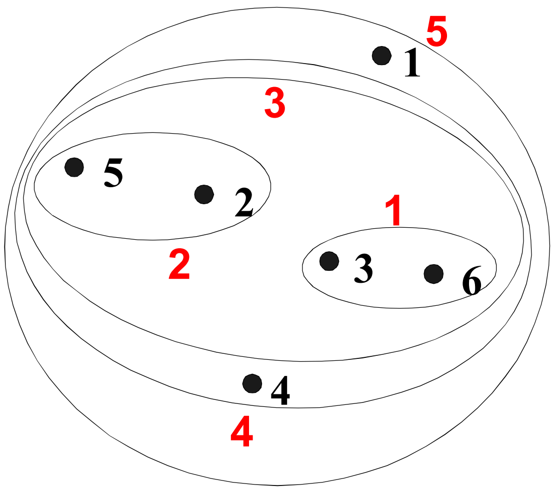
  – Can be used to initialize K-means



https://stackabuse.com/hierarchical-clustering-with-python-and-scikit-learn/

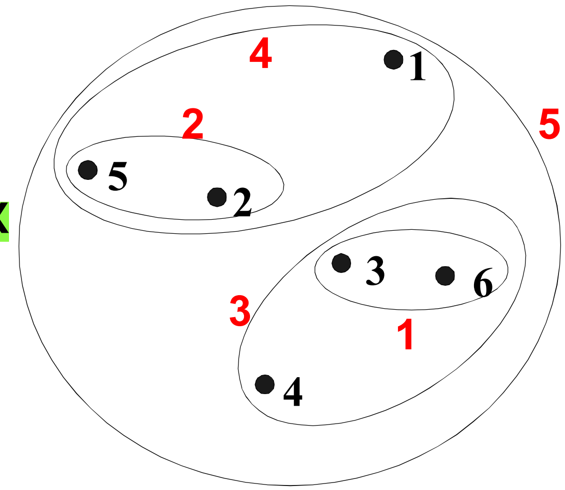
تشابه دو خوشه بر اساس افزایش مربعات خطا هنگام ادغام دو خوشه است.
- مشابه میانگین گروهی اگر فاصله بین نقاط مجذور فاصله باشد.
کمتر مستعد نویز است
گرایش به خوشه های کروی
آنالوگ سلسله مراتبی K-means- می تواند برای مقداردهی اولیه K-means استفاده شود

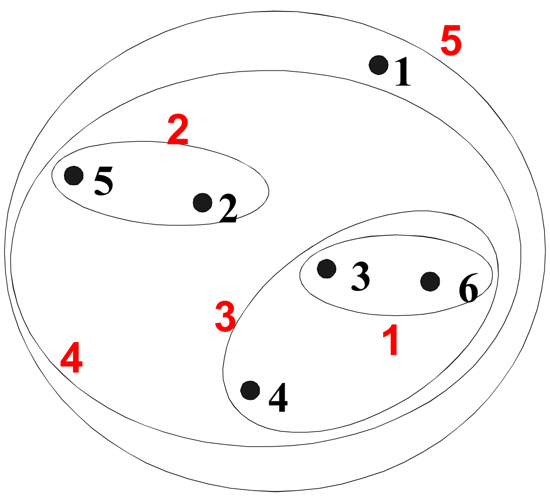
Hierarchical Clustering: Comparison



MIN

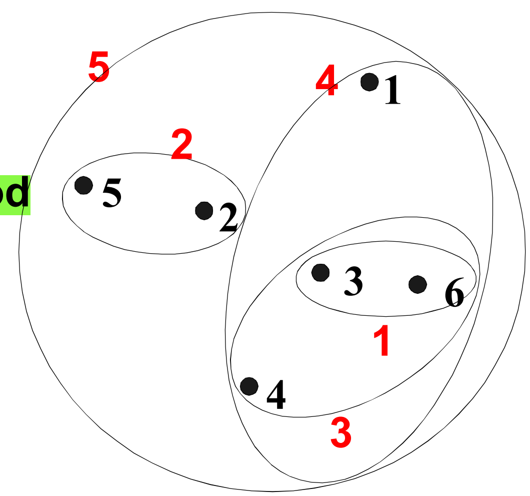
MAX





Group Average

Ward's Method



Hierarchical Clustering: Time and Space requirements

- $O(N^2)$ space since it uses the proximity matrix.
  - N is the number of points.

- $O(N^3)$ time in many cases
  - There are N steps and at each step the size, N^2, proximity matrix must be updated and searched
  - Complexity can be reduced to $O(N^2 \log(N))$ time with some cleverness

Hierarchical Clustering: Problems and Limitations

- Once a decision is made to combine two clusters, it cannot be undone

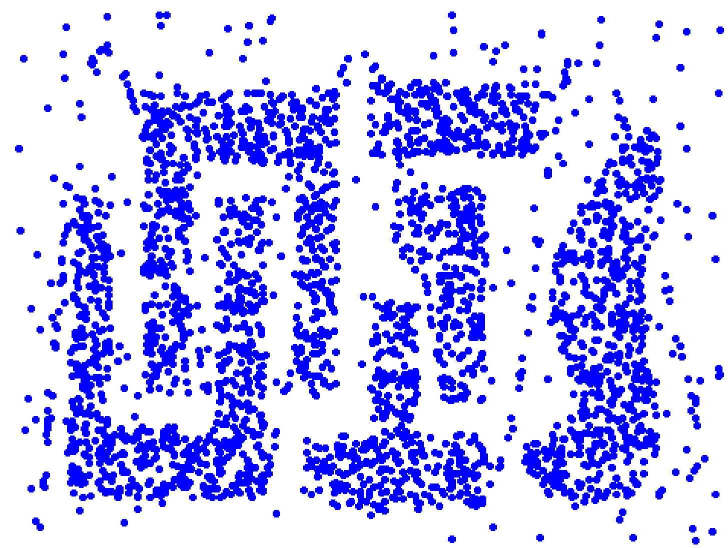
- No global objective function is directly minimized

- Different schemes have problems with one or more of the following:
  - Sensitivity to noise
  - Difficulty handling clusters of different sizes and non-globular shapes

DENSITY BASED CLUSTERING

Density Based Clustering

- Clusters are regions of high density that are separated from one another by regions on low density.
  - MinPts
  - Eps

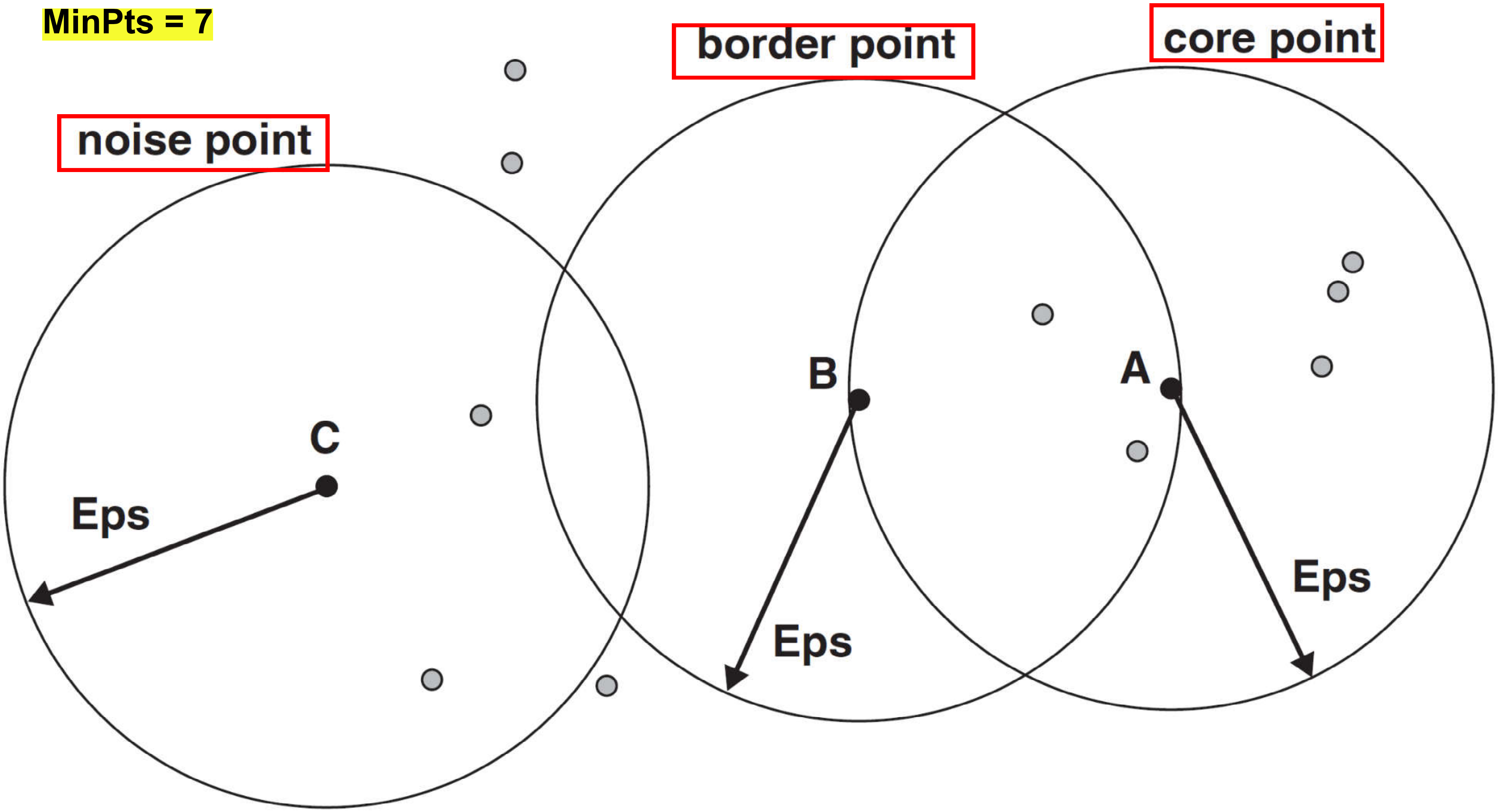


DBSCAN

- DBSCAN is a density-based algorithm.
  - Density = number of points within a specified radius (Eps)
  - A point is a core point if it has at least a specified number of points (MinPts) within Eps
    - These are points that are at the interior of a cluster
    - Counts the point itself
  - A border point is not a core point, but is in the neighborhood of a core point
  - A noise point is any point that is not a core point or a border point

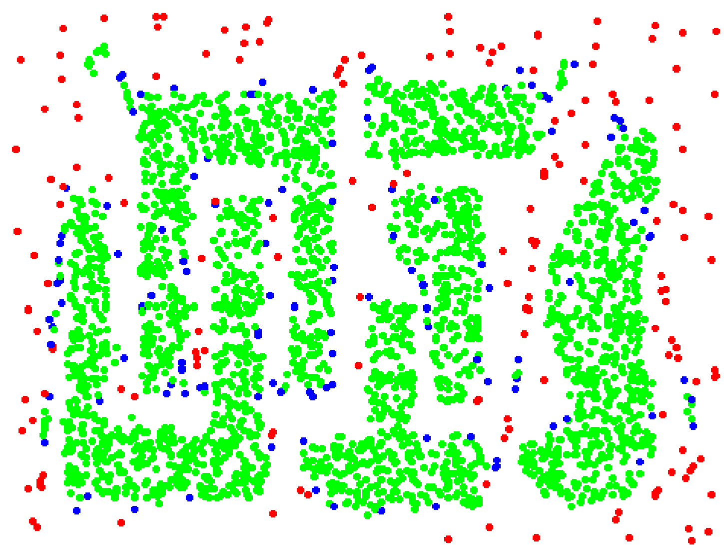
DBSCAN: Core, Border, and Noise Points

MinPts = 7



DBSCAN: Core, Border and Noise Points

Original Points



Point types: core, border and noise

Eps = 10, MinPts = 4

DBSCAN Algorithm

- Form clusters using core points, and assign border points to one of its neighboring clusters

1: Label all points as core, border, or noise points.

2: Eliminate noise points.

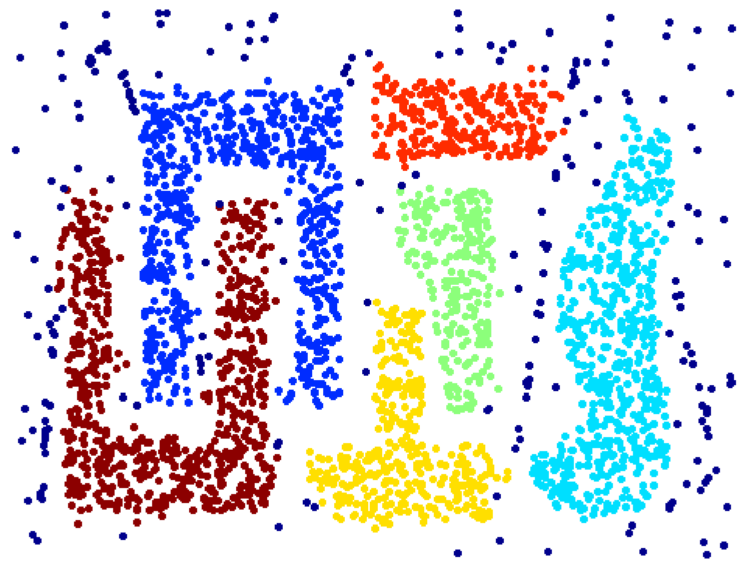
3: Put an edge between all core points within a distance *Eps* of each other.

4: Make each group of connected core points into a separate cluster.

5: Assign each border point to one of the clusters of its associated core points

When DBSCAN Works Well

Original Points

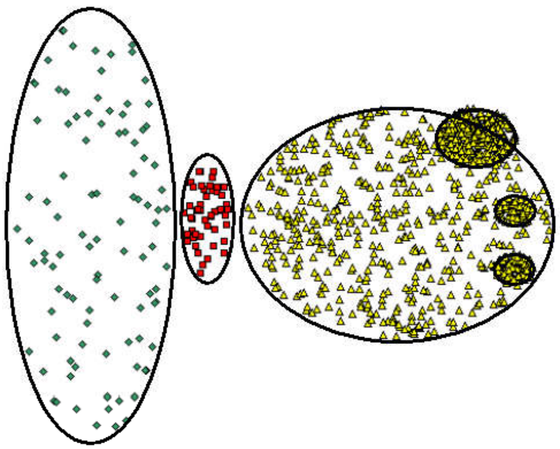


Clusters (dark blue points indicate noise)

- Can handle clusters of different shapes and sizes
- Resistant to noise

K is not required apriori

When DBSCAN Does NOT Work Well

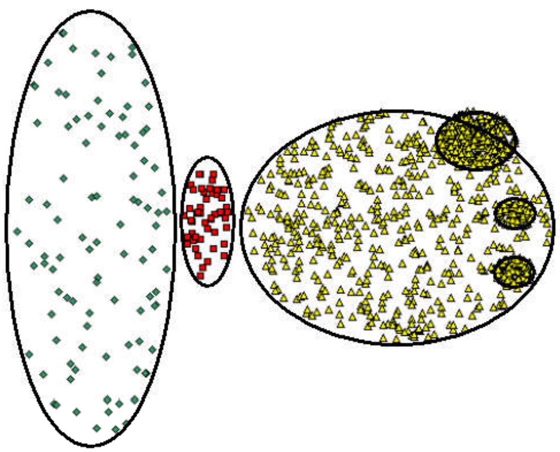


Original Points

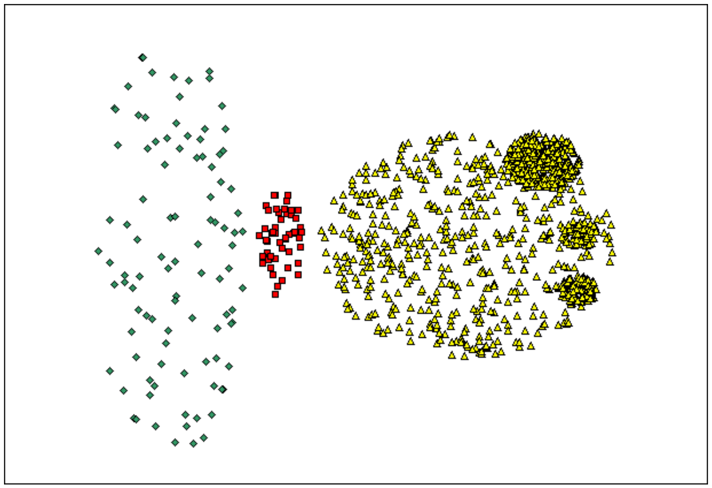
معایب:
DBScan با چگالی های مختلف به خوبی کار نمی کند (به زیر مراجعه کنید)
.نقاط مرزی خودسرانه به خوشه های همسایه اختصاص داده می شوند
اگر داده ها و مقیاس به خوبی درک نشده باشند، انتخاب یک آستانه فاصله معنی دار ε (Eps) می تواند دشوار باشد.
داده های بعد بالا محاسبات اقلیدسی را تنزل می دهد

Disadvantages:
DBScan does not work well with varying densities (see below).
Border points are arbitrarily assigned to neighboring clusters.
If the data and scale are not well understood, choosing a meaningful distance threshold ε (Eps) can be difficult.
HIgh dimension data degrades the Euclidean calculations

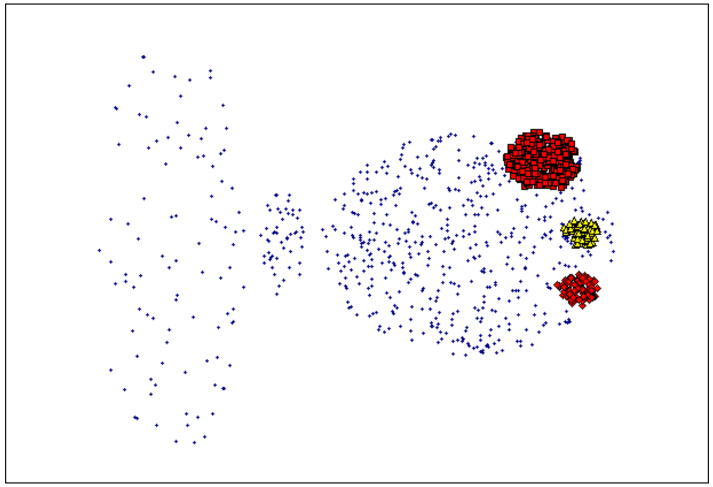
When DBSCAN Does NOT Work Well



Original Points



(MinPts=4, Eps=9.92).

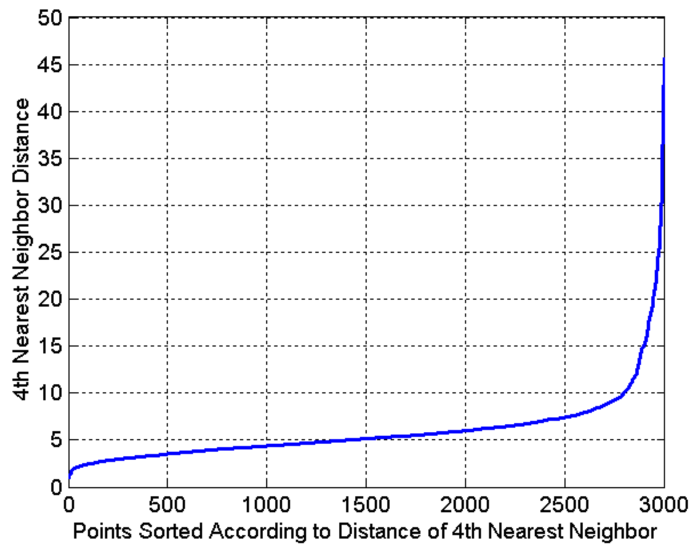


(MinPts=4, Eps=9.75)

- **Varying densities**
- **High-dimensional data**

DBSCAN: Determining EPS and MinPts

- Idea is that for points in a cluster, their k^{th} nearest neighbors are at close distance
- Noise points have the k^{th} nearest neighbor at farther distance
- So, plot sorted distance of every point to its k^{th} nearest neighbor



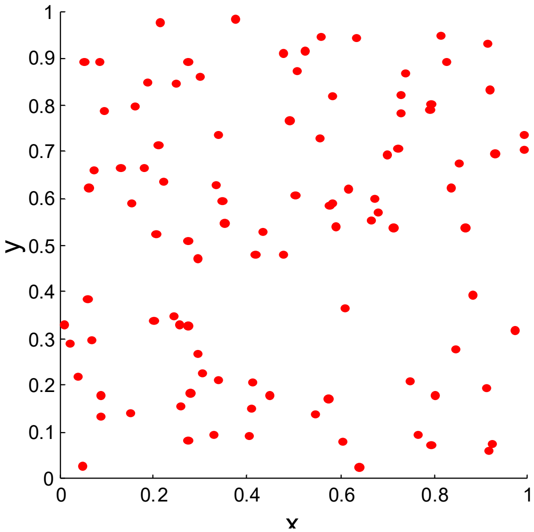
CLUSTER EVALUATION

Cluster Validity

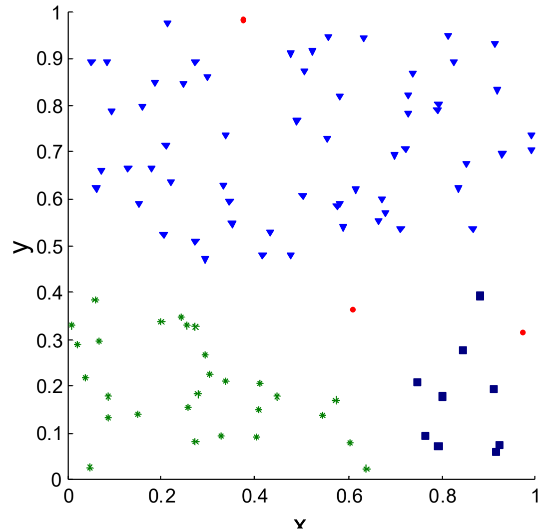
- For supervised classification we have a variety of measures to evaluate how good our model is
  - Accuracy, precision, recall
- For cluster analysis, the analogous question is how to evaluate the "goodness" of the resulting clusters?
- But "clusters are in the eye of the beholder"!
  - In practice the clusters we find are defined by the clustering algorithm
- Then why do we want to evaluate them?
  - To avoid finding patterns in noise
  - To compare clustering algorithms
  - To compare two sets of clusters
  - To compare two clusters

Clusters found in Random Data

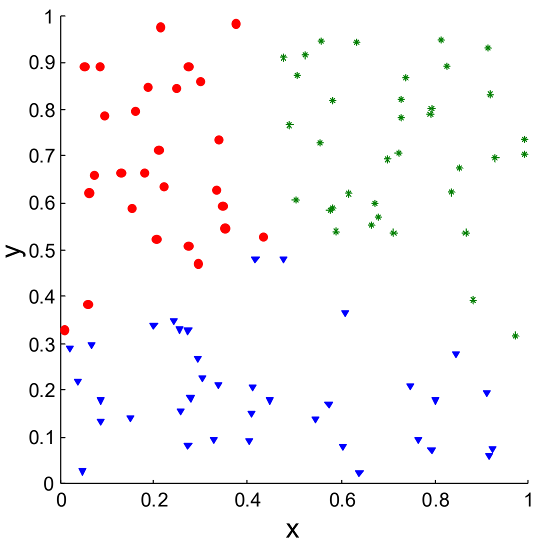
Random Points



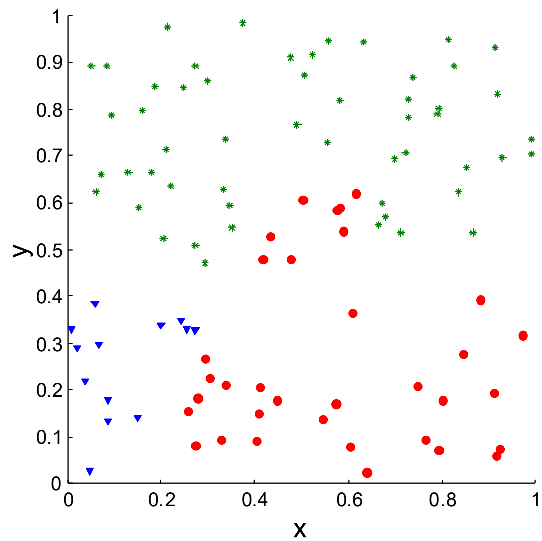
DBSCAN



K-means



Complete Link



Measures of Cluster Validity

- Numerical measures that are applied to judge various aspects of cluster validity, are classified into the following two types.
  - Supervised: Used to measure the extent to which cluster labels match externally supplied class labels.
    - Entropy
    - Often called *external indices* because they use information external to the data
  - Unsupervised: Used to measure the goodness of a clustering structure *without* respect to external information.
    - Sum of Squared Error (SSE)
    - Often called *internal indices* because they only use information in the data

- You can use supervised or unsupervised measures to compare clusters or clusterings

Unsupervised Measures: Cohesion and Separation

- Cluster Cohesion: Measures how closely related are objects in a cluster
  - Example: SSE

میزان یکپارچه بودن اعضای یک خوشه را بررسی میکنه

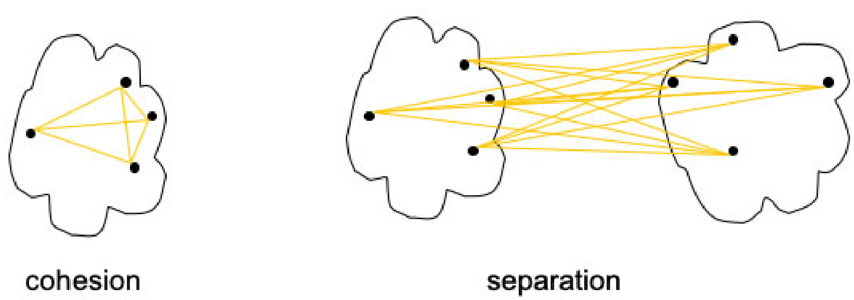
- Cluster Separation: Measure how distinct or well-separated a cluster is from other clusters
- Example: Squared Error
  - Cohesion is measured by the within cluster sum of squares (SSE)

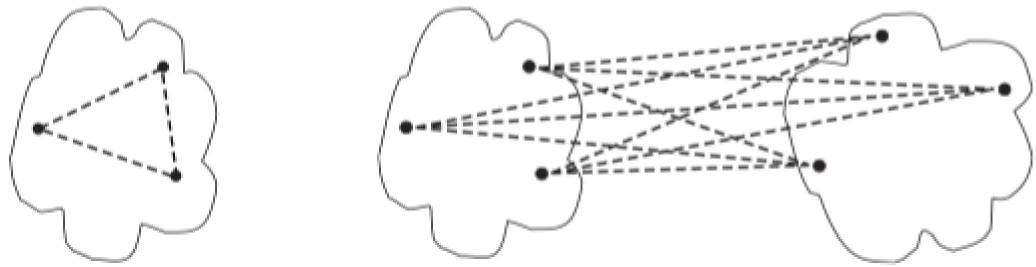
$$SSE = \sum_i \sum_{x \in C_i} (x - m_i)^2$$

  - Separation is measured by the between cluster sum of squares

$$SSB = \sum_i |C_i| (m - m_i)^2$$

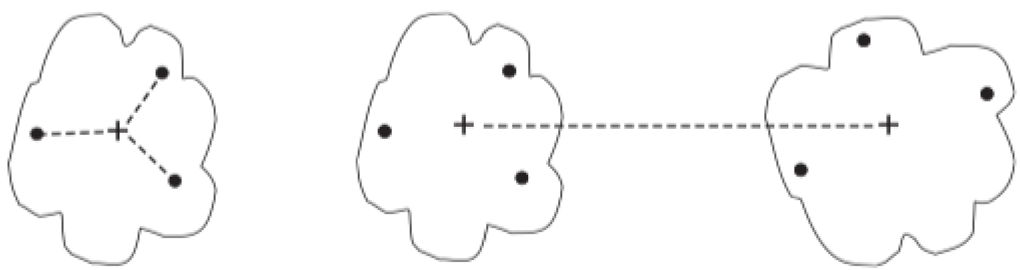
Where $|C_i|$ is the size of cluster *i*





(a) Cohesion. (b) Separation.

Figure 8.27. Graph-based view of cluster cohesion and separation.



(a) Cohesion. (b) Separation.

Figure 8.28. Prototype-based view of cluster cohesion and separation.

Cohesion and Separation

- **Cohesion** is measured by the within cluster sum of squares

$$WSS = \sum_i \sum_{x \in C_i} (x - m_i)^2$$

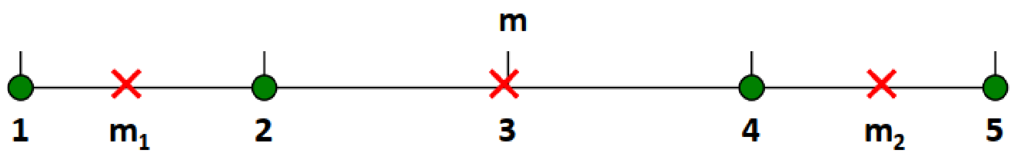
- **Separation** is measured by the between cluster sum of squares

$$BSS = \sum_i |C_i| (m - m_i)^2$$

where $|C_i|$ is the size of cluster *i*, m is the centroid of the whole data set

- BSS + WSS = constant
- WSS (Cohesion) measure is called Sum of Squared Error (SSE)—a commonly used measure
- A larger number of clusters tend to result in smaller SSE

Example



K=1 :

$$WSS = (1-3)^2 + (2-3)^2 + (4-3)^2 + (5-3)^2 = 10$$

$$BSS = 4 \times (3-3)^2 = 0$$

$$Total = 10 + 0 = 10$$

K=2 :

$$WSS = (1-1.5)^2 + (2-1.5)^2 + (4-4.5)^2 + (5-4.5)^2 = 1$$

$$BSS = 2 \times (3-1.5)^2 + 2 \times (4.5-3)^2 = 9$$

$$Total = 1 + 9 = 10$$

K=4:

$$WSS = (1-1)^2 + (2-2)^2 + (4-4)^2 + (5-5)^2 = 0$$

$$BSS = 1 \times (1-3)^2 + 1 \times (2-3)^2 + 1 \times (4-3)^2 + 1 \times (5-3)^2 = 10$$

$$Total = 0 + 10 = 10$$