



Introduction To Data Mining

Isfahan University of Technology (IUT)
Esfand1401

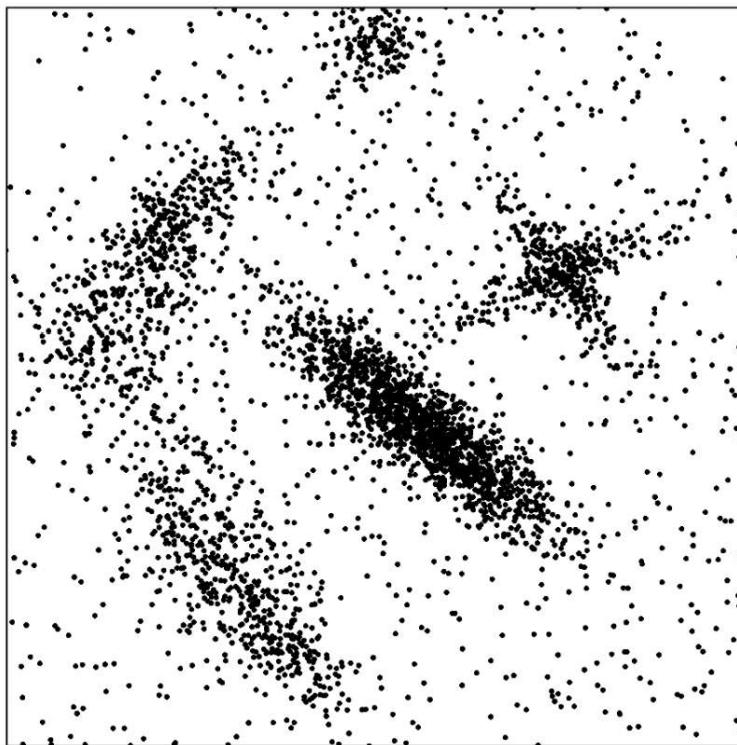


Clustering

Dr. Hamidreza Hakim
hamid.hakim.u@gmail.com

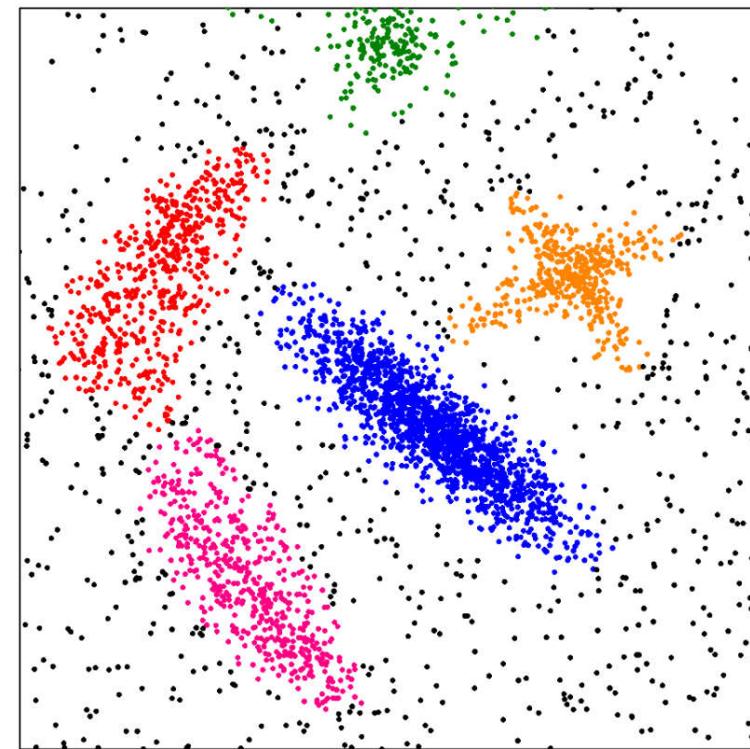
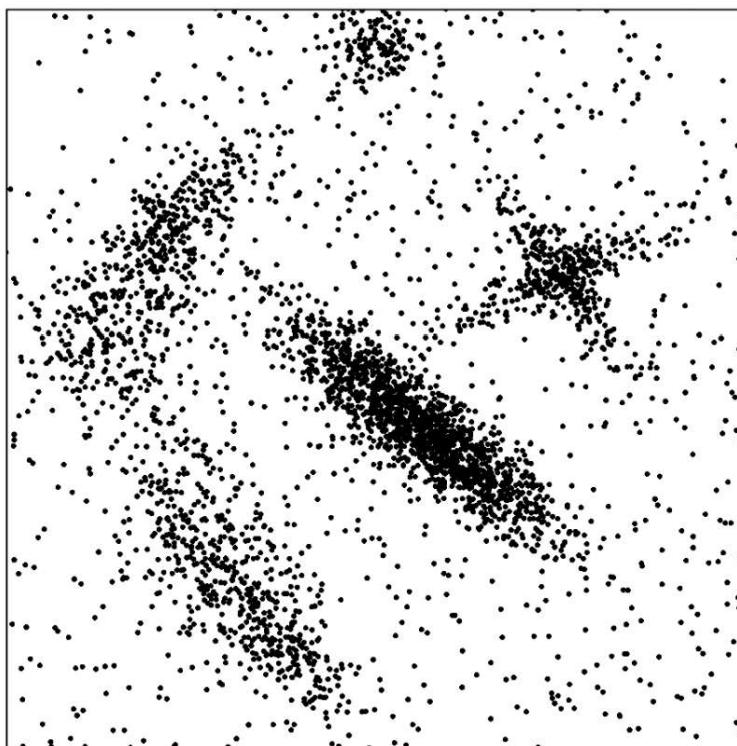
بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِيْمِ

What is Cluster Analysis?



تحلیل خوش‌ای چیست؟

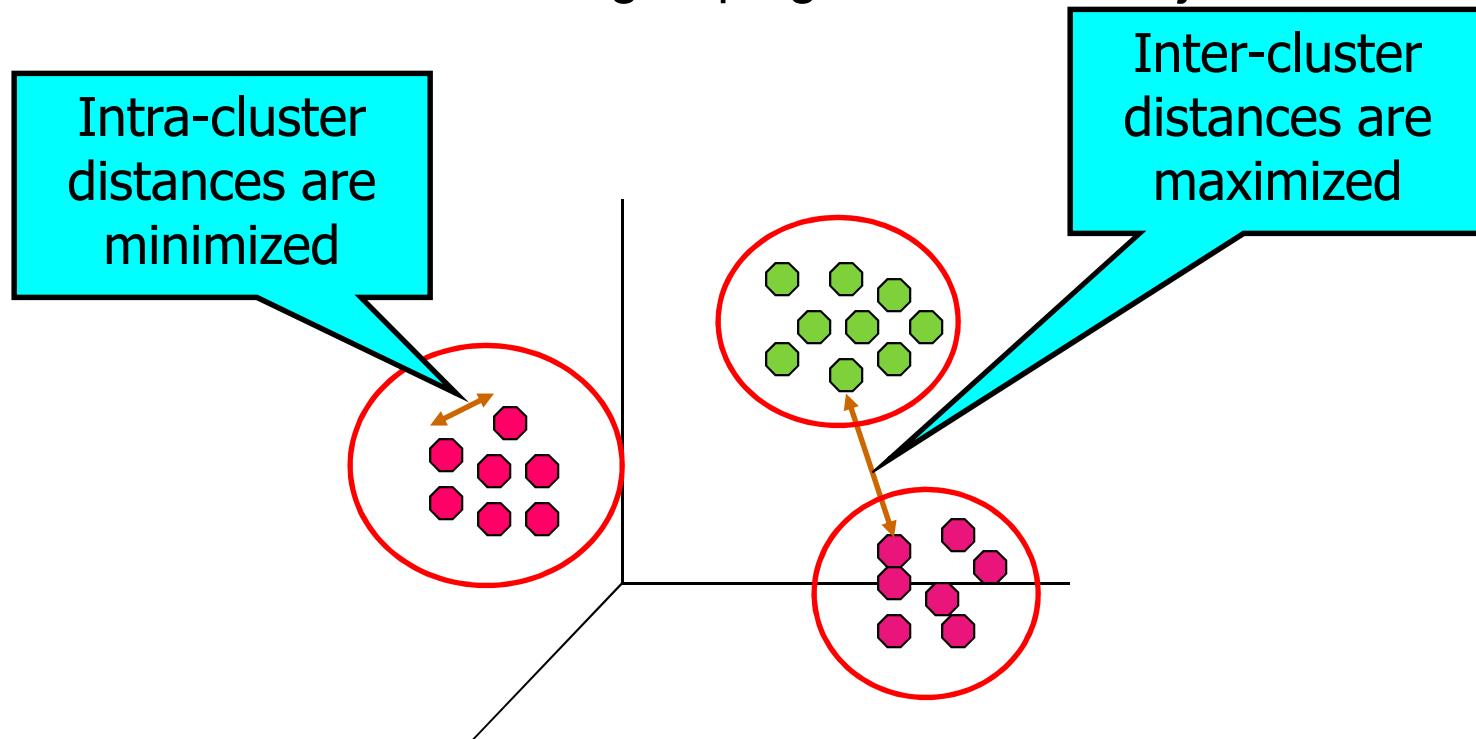
What is Cluster Analysis?



الگوریتم خوشه بندی سعی میکنه توده های نقاط رو استخراج بکنه و به ما بگه که این دیتای ما تشکیل شده از 6 دسته گروه نویز: بک گرانده میشه

What is Cluster Analysis?

- **Cluster**: A collection of data objects
 - similar (or related) to one another within the same group
 - dissimilar (or unrelated) to the objects in other groups
- **Cluster analysis** (or *clustering, data segmentation, ...*)
 - Finding similarities between data according to the characteristics found in the data and grouping similar data objects into clusters



-
مجموعه ابجکت هایی هستیم که شبیه هم هستند توی یک خوش و به شدت با نقاط خوش دیگه متفاوت هستن

عملیات خوش بندی چندین کاربر داره:
پیدا کردن ابجکت های شبیه به هم

....

دوتا اصطلاح:

فاصله درون کلاسی: توی خوش بندی دنبال این هستیم که ابجکت هایی رو درون یک خوش بذاریم
که فاصله اون ابجکت تا نمونه های هم خوشه ایش حداقل باشه

فاصله بیرون کلاسی: فاصله بین خوشه ای که این حداکثر باشه --> از اون نمونه به نمونه خوشه
مجاور فاصله حداقل باشه

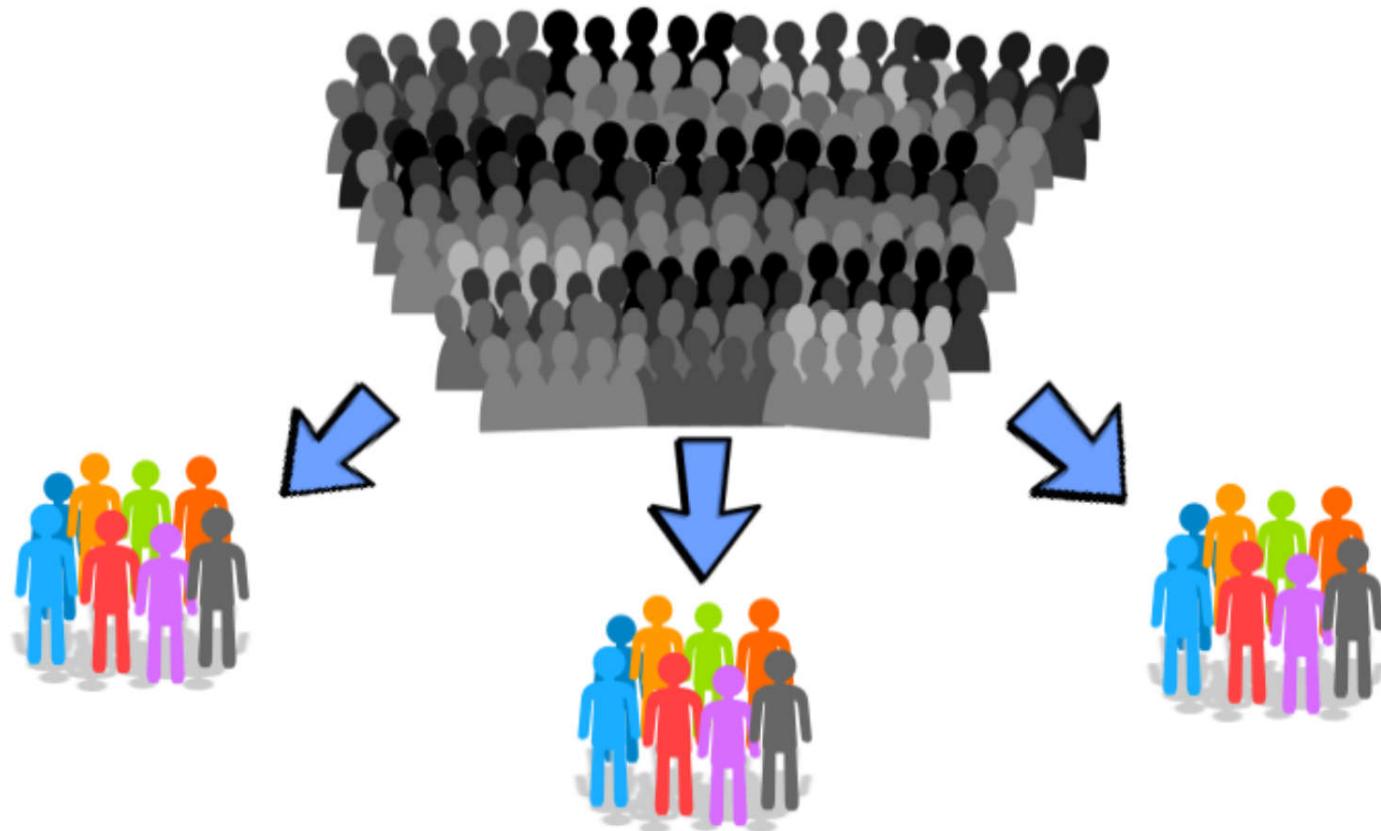
What is Cluster Analysis?

- **Unsupervised learning**: no predefined classes (i.e., *learning by observations* vs. learning by examples: supervised)
- Typical applications
 - As a **stand-alone tool** to get insight into data distribution
 - As a **preprocessing step** for other algorithms

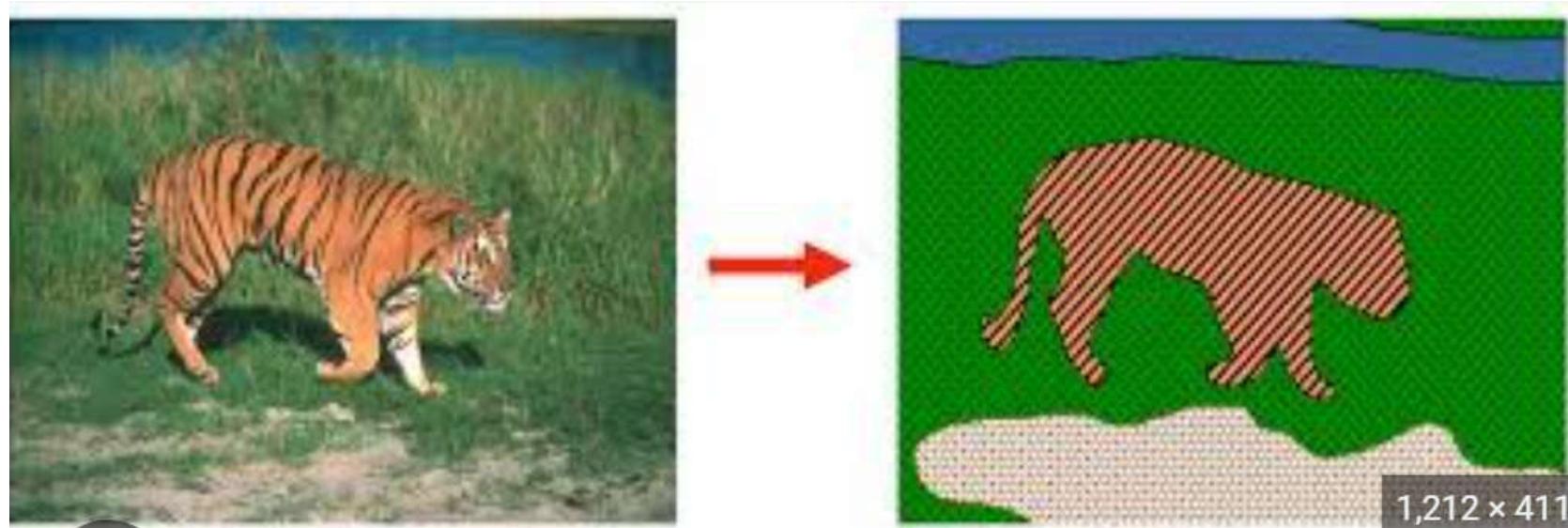
- اینجا برچسب نداریم
کاربرد خوش بندی:

- 1- ماهیت خوش بندی --> بهمون یک دیتایی میدن و بهمون میگن اینو خوش بندی بکن ینی این دیتا رو چند گروهش بکن
- 2- پیش پردازش --> یک حجم زیادی دیتا داریم و می خوایم کلاسیفایر رو روش ترین بکنیم و خیلی دیتا زیاد است و اصلا نمی توانیم ترین بکنیم با این حجم دیتا پس میایم اینارو تقسیم بندی می کنیم ینی به جای 1 میلیون به مدل 100 هزار داده بدیم

Example: customer segmentation



Example: image segmentation



Example: information retrieval

Google jaguar

All Images News Videos Books More Tools

car cat f type f pace drawing wallpaper leopard GU

W Wikipedia
Jaguar - Wikipedia

N National Geographic
Jaguar, facts and photos

E Encyclopedia Britannica
Jaguar | Habitat, Diet, & Fact...

S St. Louis Zoo
Saint Louis Zoo | Jaguar

W ia.wikipedia.org
Jaguar - Wikipedia, le encyclopedia libere

C CarWale
Jaguar XF Price - Images, Colours ...

W www.jaguar.co.uk
Luxury Sports Cars, Executive Saloons ...

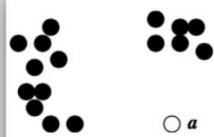
Example: outlier detection

- What are outliers?
- The set of objects are considerably dissimilar from the remainder of the data
- Outlier detection is useful in fraud detection applications such as creditcard fraud detection.
- It is also useful as a preprocessing tool.
- One way for outlier detection is using clustering:
 - Objects that do not belong to any cluster
 - Objects far from other objects in the same cluster
 - Clusters of very small cardinality

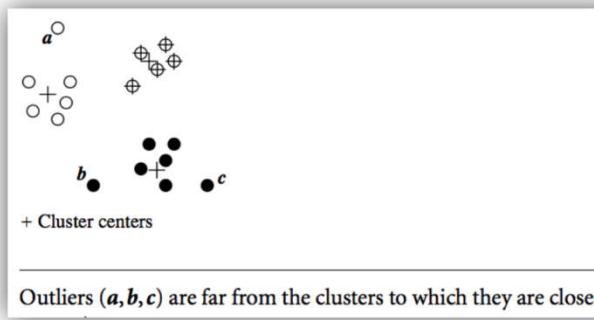
:outlier detection

وقتی با دیتاهای مجاورش و با بقیه دیتاهای خیلی متفاوت باشه
می تونیم با روش های خوش بندی داده هایی که پر ت هستن رو یه جور ای شناسایی بکنیم و یا
فضای جستجو که به داده ها مشکوک بشیم کم بکنیم و به یک گروه خاصی مشکوک بشیم

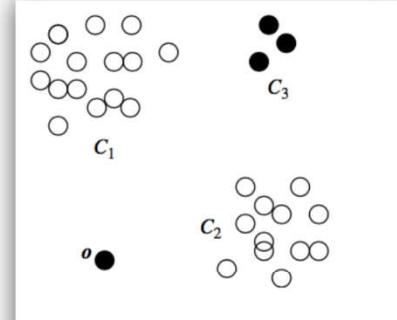
Example: outlier detection



Object a is an outlier because it does not belong to any cluster.



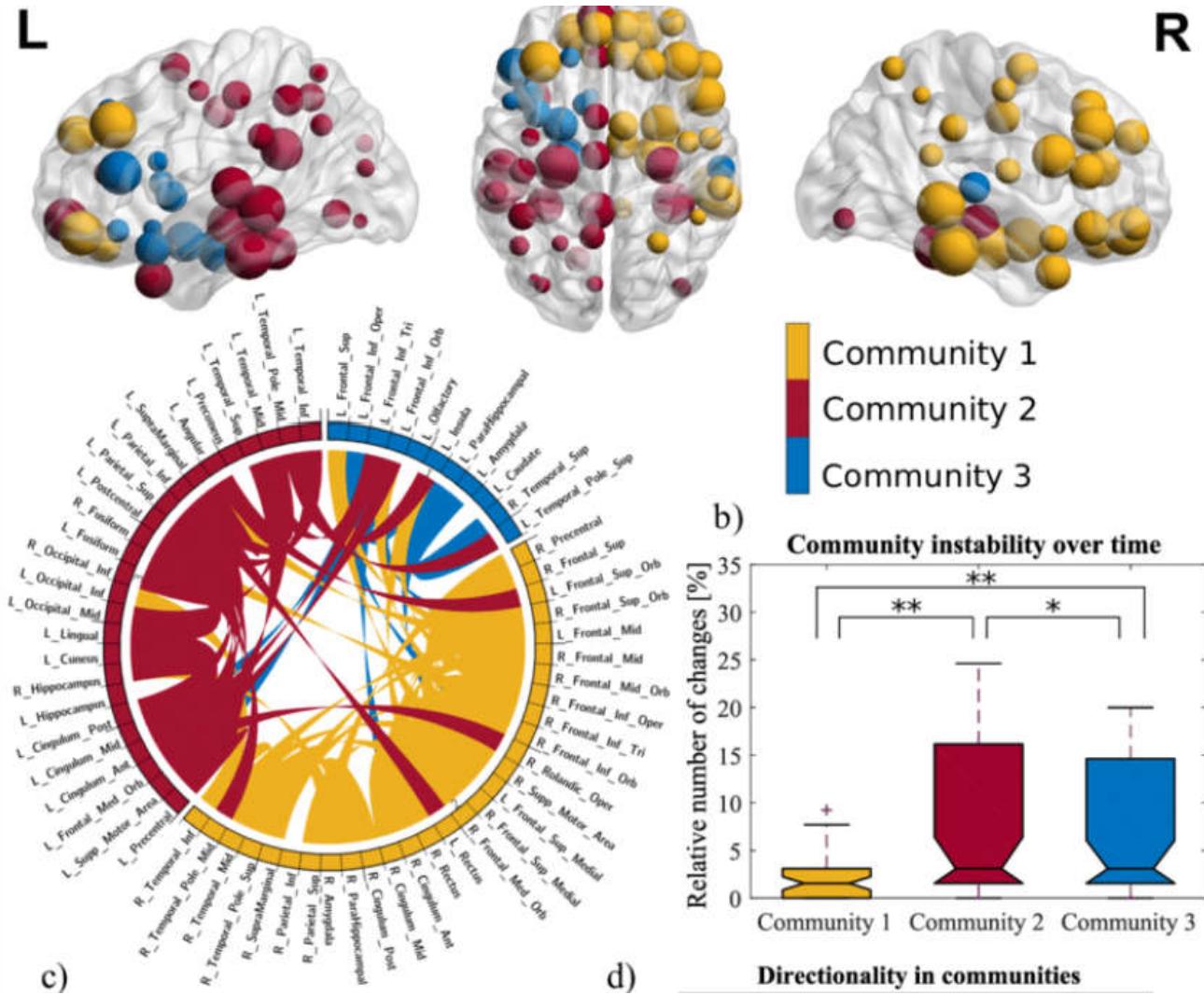
Outliers (a, b, c) are far from the clusters to which they are closest.



Outliers in small clusters.

Refer to chapter 12 of the third edition of “Data Mining: Concepts and Techniques” for more information.

Example: community detection



: community detection

یک مسئله ای داریم توی بحث انالیز داده های گرافی --> وقتی یک شبکه ای اجتماعی داریم که فعالیت می کنن و این نقاط مثل هم رفتار می کنن مثلا وقتی می خوایم دست رو حرکت بدیم چندین نقطه باید فعال بشه که این دست حرکت بکنه

یک مسئله ای توی گراف داریم ک وقته یک شبکه رو دیدیم برآمون مهم است که ببینیم چه نودهایی هم خانواده با هم هستن (یک اجتماع رو تشکیل میدن) --> با روابطشون می خوایم به این مسئله بررسیم و توی community detection سراغ این مسئله می رن
یکی از کاربردهای خوشه بندی community detection است

Clustering for Data Understanding and Applications

- Biology: taxonomy of living things: kingdom, phylum, class, order, family, genus and species
- Information retrieval: document clustering
- Land use: Identification of areas of similar land use in an earth observation database
- Marketing: Help marketers discover distinct groups in their customer bases, and then use this knowledge to develop targeted marketing programs
- City-planning: Identifying groups of houses according to their house type, value, and geographical location
- Earthquake studies: Observed earth quake epicenters should be clustered along continent faults
- Climate: understanding earth climate, find patterns of atmospheric and ocean
- Economic Science: market research

خوشه بندی برای درک داده ها و
برنامه های کاربردی

زیست شناسی: طبقه بندی موجودات زنده: پادشاهی، شاخه، طبقه، نظم، خانواده، جنس و گونه
بازیابی اطلاعات: خوشه بندی اسناد

کاربری اراضی: شناسایی مناطق با کاربری مشابه در پایگاه داده رصد زمین

بازاریابی: به بازاریابان کمک کنید تا گروه های متمایز را در پایگاه مشتریان خود کشف کنند و سپس از این دانش برای توسعه برنامه های بازاریابی هدفمند استفاده کنند.

برنامه ریزی شهری: شناسایی گروه خانه ها بر اساس نوع خانه، ارزش و موقعیت جغرافیایی.

مطالعات زمین لرزه: کانون های زمین لرزه مشاهده شده باید در امتداد گسل های قاره قرار گیرند.

آب و هوا: درک آب و هوای زمین، یافتن الگوهای جوی و اقیانوسی

علوم اقتصادی: تحقیقات بازار

Clustering as a Preprocessing Tool (Utility)

- Summarization:
 - Preprocessing for regression, PCA, classification, and association analysis
- Compression:
 - Image processing: vector quantization
- Finding K-nearest Neighbors
 - Localizing search to one or a small number of clusters
- Outlier detection
 - Outliers are often viewed as those “far away” from any cluster

-
این پیش پردازش رو کجا نیاز داریم؟

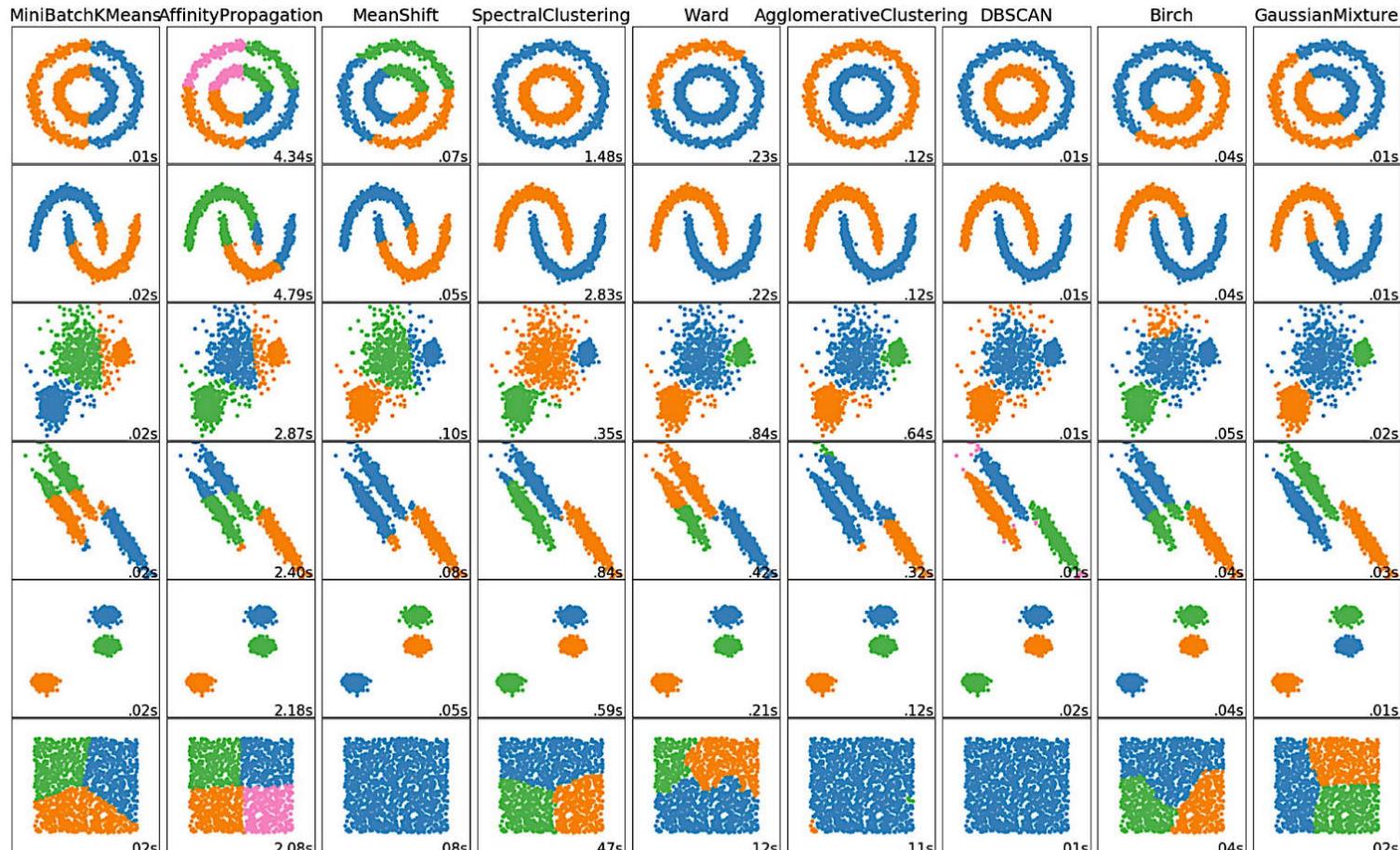
حجم داده ها زیاده و میخوایم کم بکنیم یعنی گروه گروه بررسی کنیم

می خوایم از اون Outlier حذف بکنیم

می خوایم خلاصه بکنیم داده ها رو

KNN توی بحث

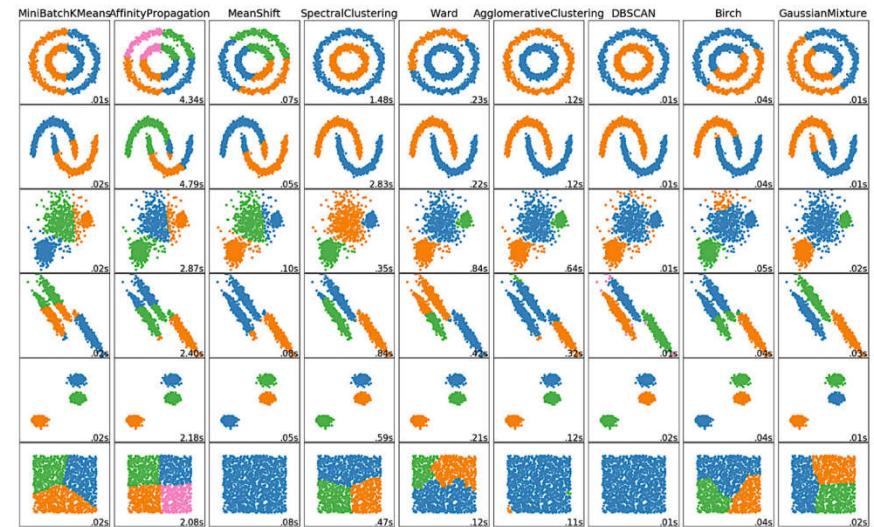
Clustering Toy Data



http://scikit-learn.org/stable/auto_examples/cluster/plot_cluster_comparison.html

Clustering Toy Data

- These are toy 2D datasets.
- The last dataset is an example of a ‘null’ situation for clustering.
- With the exception of the last dataset, the parameters of each of these dataset algorithm pairs has been tuned to produce good clustering results.
- Note that the intuitions might not apply to very high dimensional data.



http://scikit-learn.org/stable/auto_examples/cluster/plot_cluster_comparison.html

اینها مجموعه داده های دو بعدی اسباب بازی هستند.

آخرین مجموعه داده نمونه ای از وضعیت «تهی» «برای خوشبندی است.

به استثنای آخرین مجموعه داده، پارامترهای هر یک از این جفت‌های الگوریتم مجموعه داده برای تولید نتایج خوشبندی خوب تنظیم شده‌اند.

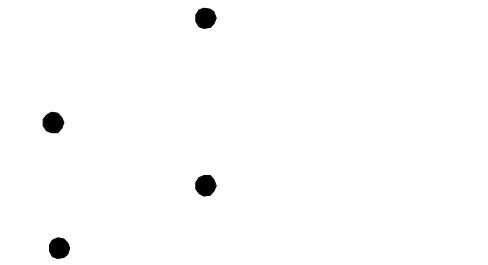
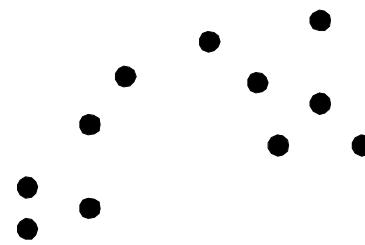
توجه داشته باشید که شهود ممکن است برای داده‌های ابعادی بسیار بالا اعمال نشود

Types of Clusterings

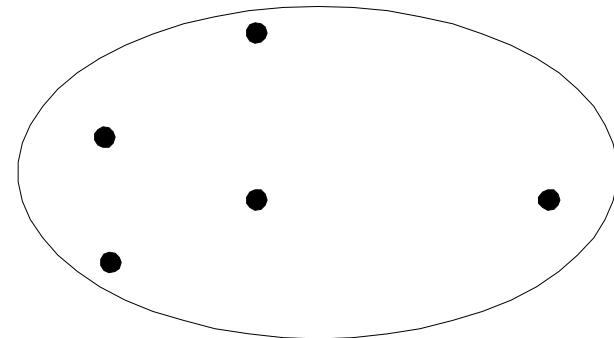
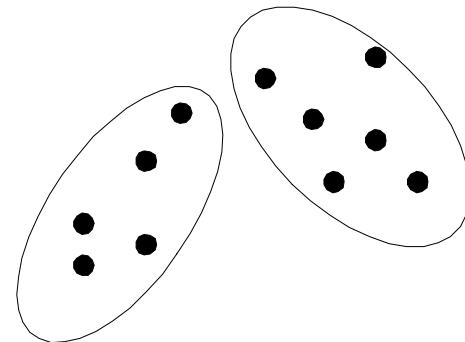
- A **clustering** is a set of clusters
- Important distinction between **hierarchical** and **partitional** sets of clusters
 - **Partitional Clustering**
 - ◆ A division of data objects into non-overlapping subsets (clusters)
 - **Hierarchical clustering**
 - ◆ A set of nested clusters organized as a hierarchical tree

- خوشه های ما چه رابطه ای با هم دارند --> معروف به خوشه بندی سلسله مراتبی و **partitional**

Partitional Clustering



Original Points

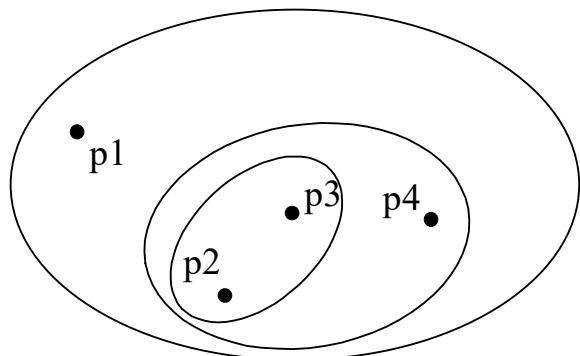


A Partitional Clustering

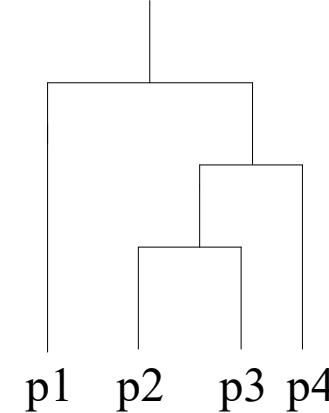
کاری که خوش بندی **partitional** میکنه:

یک خوش به هر توده از داده ها میده و خوش ها جدا جدا هستن و اشتراکی با هم ندارند

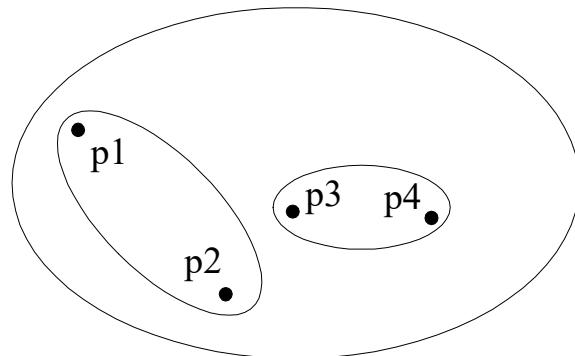
Hierarchical Clustering



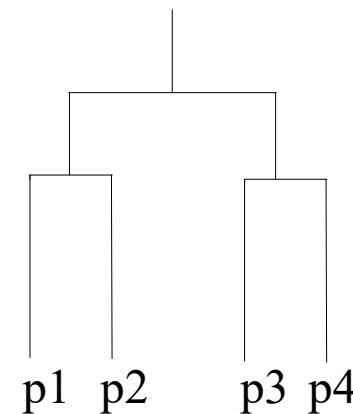
Traditional Hierarchical Clustering



Traditional Dendrogram



Non-traditional Hierarchical Clustering



Non-traditional Dendrogram

توی خوشه بندی سلسله مراتبی:

توی این روش خوشه ها لزوما اینطوری نیستن که بگیم یک خوشه با یک خوشه دیگه کامل جداست بلکه می تونیم یک خوشه ای داشته باشیم که داخل اون خوشه، خوشه کوچکتری است

Other Distinctions Between Sets of Clusters

- Exclusive versus non-exclusive
 - In non-exclusive clusterings, points may belong to multiple clusters.
 - ◆ Can belong to multiple classes or could be ‘border’ points
 - Fuzzy clustering (one type of non-exclusive)
 - ◆ In fuzzy clustering, a point belongs to every cluster with some weight between 0 and 1
 - ◆ Weights must sum to 1
 - ◆ Probabilistic clustering has similar characteristics
- Partial versus complete
 - In some cases, we only want to cluster some of the data

-
بحث هایی که توی خوشه بندی داریم:

اون خوشه هایی که داریم در نظر می گیریم ایا نمونه هایی که توی هر خوشه هستن حتما مال اون خوشن یا میشه به صورت فازی به اینا نگاه کنیم؟

بعضی از روش های خوشه بندی میان نمونه ها رو قطعا به یک دسته متعلق می دونن ولی بعضی از خوشه ها هستن که حرف قطعی نمی زنن که به این میگن فازی یعنی اینقدر نمونه مال این خوشه است و اینقدر نمونه مال اون خوشه --> اینجا الگوریتم ها قطعی هستن (توی این درس)

اون الگوریتم خوشه بندی میاد روی کل دیتا کار رو انجام میده یا روی یک بخش خاصی -->
الگوریتم هایی که اینجا میگیم کل دیتا رو در نظر می گیره

Types of Clusters

- Well-separated clusters
- Prototype-based clusters
- Contiguity-based clusters
- Density-based clusters
- Described by an Objective Function

- این عبارت مهمه خارجیشون:

اول تیپی که خوشه ها می تونن نسبت به هم می تونن پیدا بکنن:

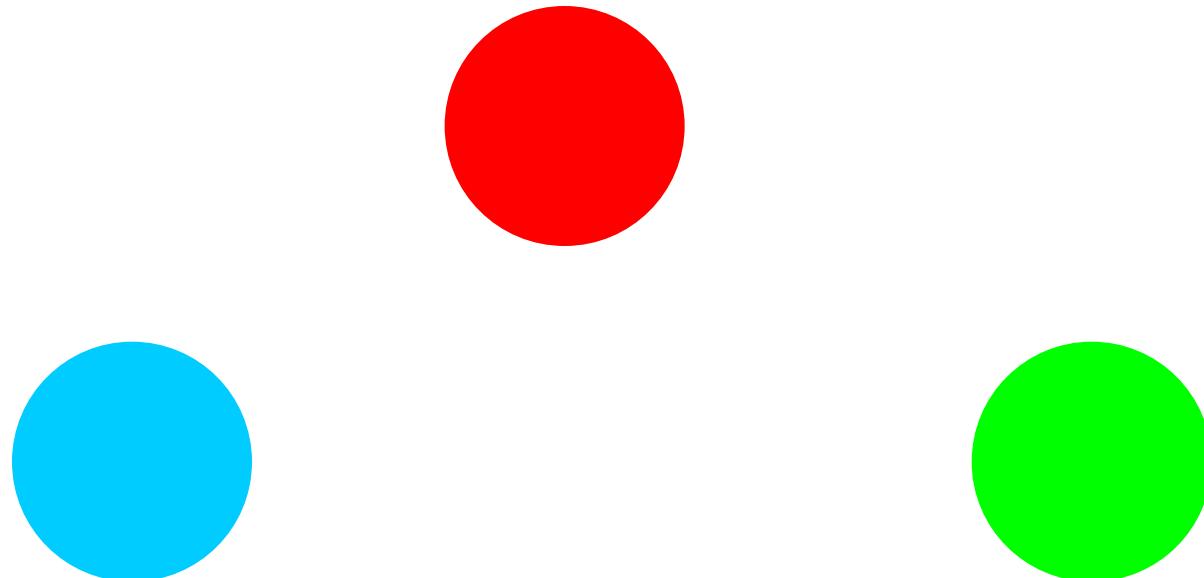
1- حالت هایی که خوشه ها کاملا از هم جدا هستن --> هیچ ربطی ندارن نمونه های توی خوشه به نمونه های توی خوشه دیگه

2- یه وقت هایی اون خوشه هایی که توی داده ها شکل می گیرن به صورت Prototype-based هستن --> ممکنه داده ها خیلی داده ها به هم نزدیک باشن ولی ما ذاتا این ها رو دوتا خوشه در نظر میگیریم چرا به این ذهنیت می رسیم؟ چون میگیم این داده ها حول یک مرکزی قرار دارند و بهتره اینا رو یک خوشه بگیریم و اونایی که حول یک مرکز دیگه هستن رو یک خوشه دیگه میگیریم

3- خوشه های مجاور: مثلا یک دیتاای شبیه شکل

Types of Clusters: Well-Separated

- Well-Separated Clusters:
 - A cluster is a set of points such that **any point in a cluster is closer** (or more similar) to **every other point** in the cluster than to any point not in the cluster.



3 well-separated clusters

انواع خوشه ها: به خوبی از هم جدا شده اند
خوشه های به خوبی جدا شده:

- خوشه مجموعه ای از نقاط است به طوری که هر نقطه در یک خوشه به هر نقطه دیگر در خوشه نزدیک تر (یا شبیه تر) باشد تا به هر نقطه ای که در خوشه نیست.

Types of Clusters: Prototype-Based

- Prototype-based

- A cluster is a set of objects such that an object in a cluster is closer (more similar) **to the prototype** or “center” of a cluster, than to the center of any other cluster
- The center of a cluster is often a **centroid**, the average of all the points in the cluster, or a **medoid**, the most “representative” point of a cluster



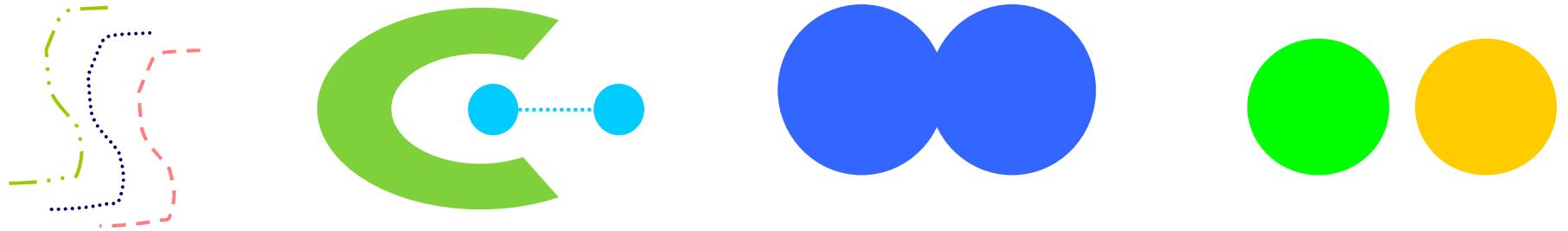
4 center-based clusters

أنواع خوشة ها: مبتنى بر نمونه اوليه
مبتنى بر نمونه اوليه

- خوشه مجموعه اي از اشیاء است به طوری که يك شى در يك خوشه به نمونه اوليه يا "مرکز" يك خوشه نزدیکتر (شبیه تر) از مرکز هر خوشه دیگری است.
- مرکز يك خوشه اغلب يك مرکز، میانگین تمام نقاط خوشه، يا يك مدويد، «نمایندہترین» نقطه يك خوشه است.

Types of Clusters: Contiguity-Based

- Contiguous Cluster
(Nearest neighbor or Transitive)
 - A cluster is a set of points such that a point in a cluster is closer (or more similar) to one or more other points in the cluster than to any point not in the cluster.
 - Used when the clusters are irregular or intertwined(non Noise)



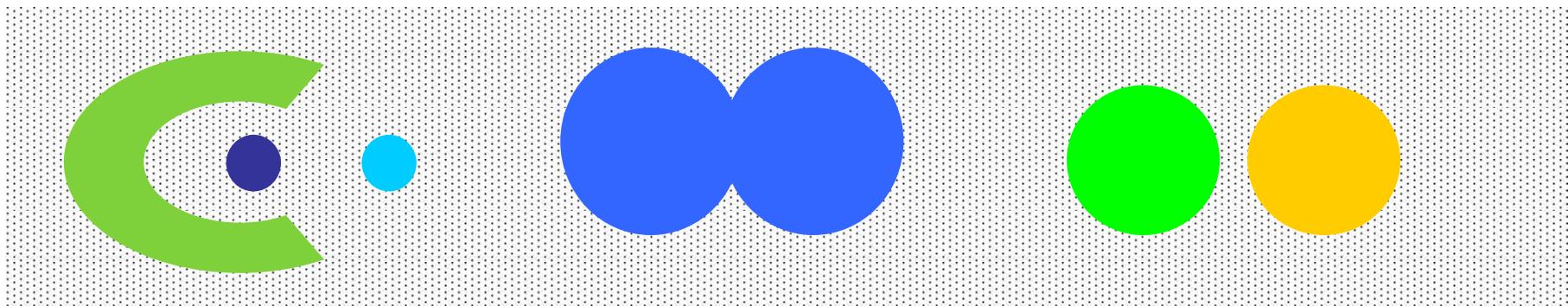
8 contiguous clusters

-
ویژگ این خوشه:

نقاطی که همسایه هم هستن حتما توی یک خوشه اند

Types of Clusters: Density-Based

- Density-based
 - A cluster is a dense region of points, which is separated by low-density regions, from other regions of high density.
 - Used when the clusters are irregular or intertwined, and when noise and outliers are present.



6 density-based clusters

- خوشه هایی که مبتنی بر تراکم داده ها شکل گرفته: بنی براساس تراکم داده ها میایم میگیم این یک خوشه میشه

Clustering Algorithms

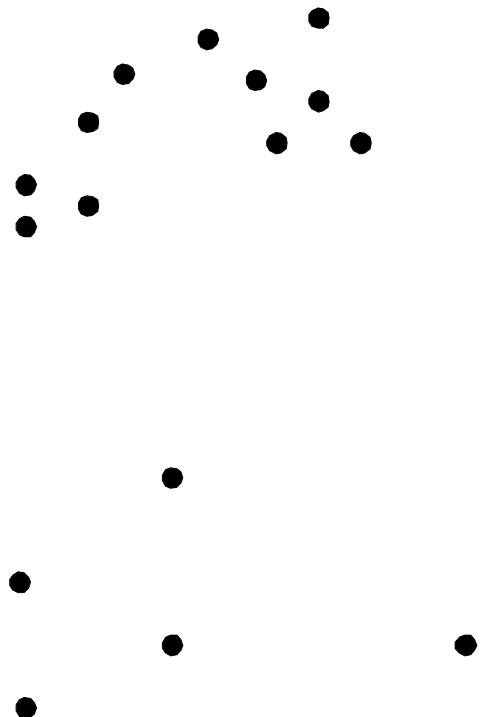
- K-means and its variants
- Hierarchical clustering
- Density-based clustering

الگوریتم های خوش بندی
K-means و انواع آن
خوش بندی سلسله مراتبی
خوش بندی مبنی بر چگالی

K-MEANS CLUSTERING

Partitioning Algorithms: Basic Concept

- Partitioning method: Partitioning a database D of n objects into a set of k clusters,



- : K-MEANS

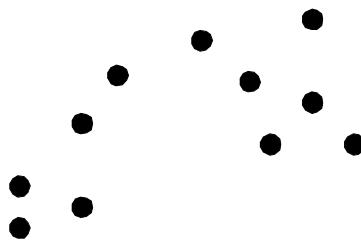
این روش سلسله مراتبی نیست

خوشه هایی که از این الگوریتم به دست میاد کاملاً جدا هستند

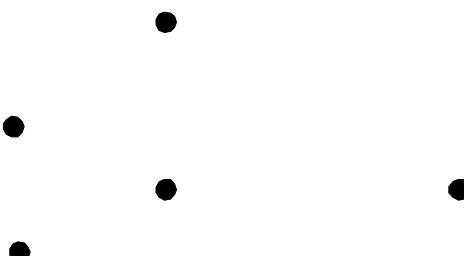
می خوایم این دیتا رو به k تا خوشه تقسیم بکنیم --> برای این که این الگوریتم این کار رو انجام بد
میاد یک تابع هزینه در نظر میگیره و این تابع هزینه به این الگوریتم کمک میکنه که تا کجا ادامه
بده و براساس این تابع هزینه می فهمه به سمت خوبی می ره یا بهتره مسیر رو تغییر بد

Partitioning Algorithms: Basic Concept

- Partitioning method: Partitioning a database D of n objects into a set of k clusters,
such that the sum of squared distances is minimized (where c_i is the centroid or medoid of cluster C_i)



$$E = \sum_{i=1}^k \sum_{p \in C_i} (p - c_i)^2$$



- روش K-MEANS مبتنی بر یک سری مرکز خوش است

روش K-MEANS با اون نماینده کار میکنه ینی در نهایت می خوایم برای خوش یک نماینده

معرفی بکنیم --> سعی میکنیم یک نماینده خوب معرفی بکنیم و براساس اون نماینده خوش بندی رو

انجام بدیم تا هر مرحله باید این نماینده رو انتخاب بکنیم این نماینده میشه این p

این تابع هزینه با این نماینده کار میکنه ینی هر موقع نماینده معرفی کردی ما میایم فاصله های نمونه

های درون خوش رو از نماینده اش می سنجیم --> این فاصله ها رو اگر جمع بزنیم یک عددی

میشه این عدد میشه معرف این روش خوش بندی پنی چقدر نماینده هایی که معرفی کردی از نمونه

های درون خوششون فاصله دارن که این میشه یک معیار برای ارزیابی کیفیت کار

روش K-MEANS راه و رسم نماینده پیدا کردن رو بهمن یاد میده

Partitioning Algorithms: Basic Concept

- Given k , find a partition of k clusters that optimizes the chosen partitioning criterion
 - Global optimal: exhaustively enumerate all partitions
 - Heuristic methods: k -means and k -medoids algorithms
 - k -means (MacQueen'67, Lloyd'57/'82):
Each cluster is represented by the center of the cluster
 - k -medoids or PAM (Partition around medoids) (Kaufman & Rousseeuw'87):
Each cluster is represented by one of the objects in the cluster

$$E = \sum_{i=1}^k \sum_{p \in C_i} (p - c_i)^2$$

-
توی K-MEANS میانگین رو به عنوان نماینده در نظر می گیریم --> لزوما به جواب بهینه نمی رسه و داره حریصانه عمل می کنه
توی k-medoids میانه ها رو به عنوان نماینده در نظر می گیریم

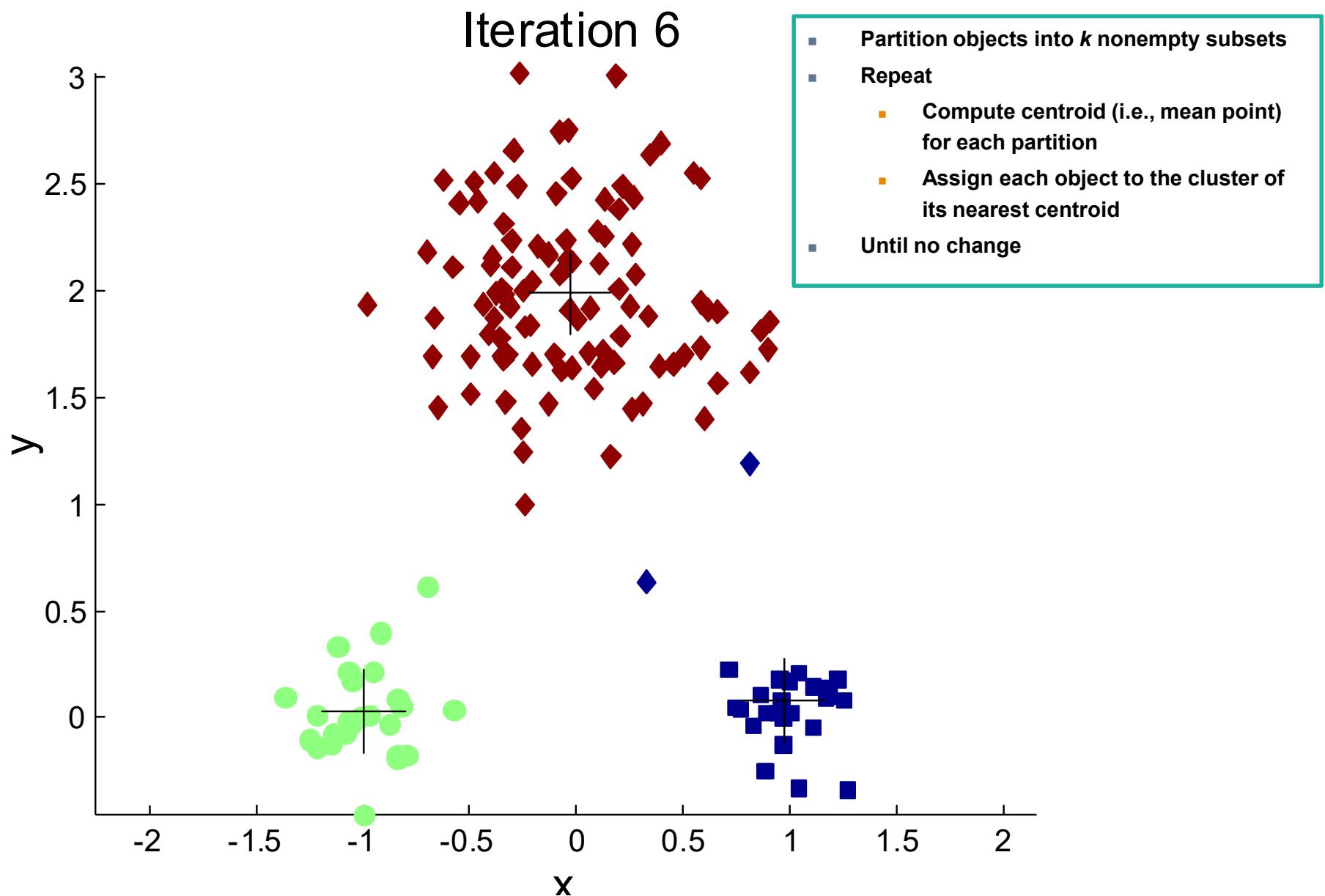
The *K-Means* Clustering Method

- 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 partitioning (the centroid is the center, i.e., *mean point*, of the cluster)
 - Assign each object to the cluster with the nearest seed point
 - Go back to Step 2, stop when the assignment does not change

-

مسئله ای که راجع به K-Means وجود داره حالت های اولیه است ینی حالت های اولیه رو کجا بذاریم که بعضا اینارو به صورت تصادفی انتخاب می کنن که حساس نباشه به یک سری مسائلی

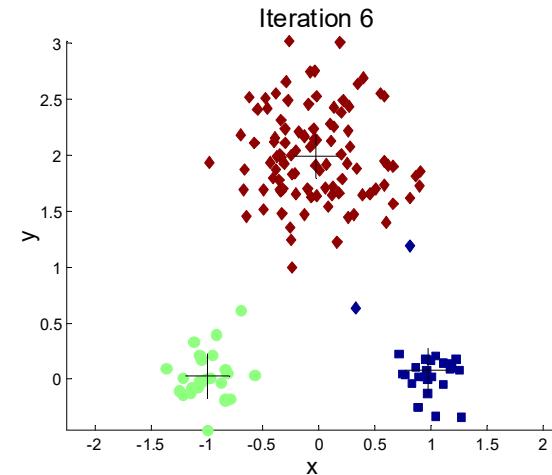
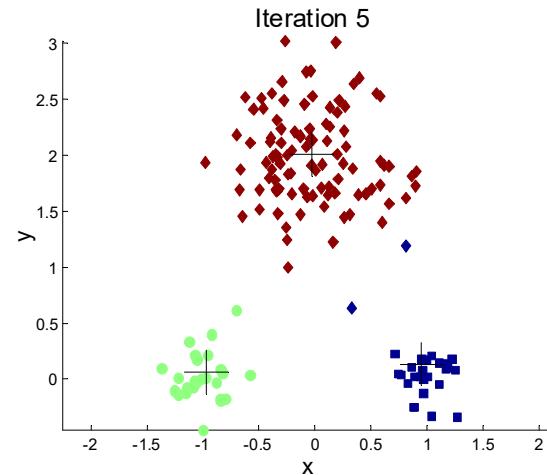
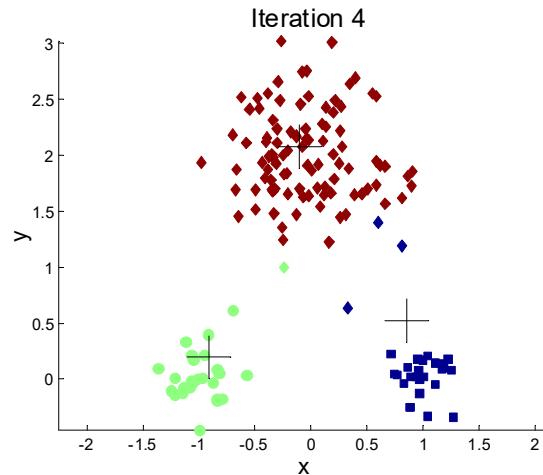
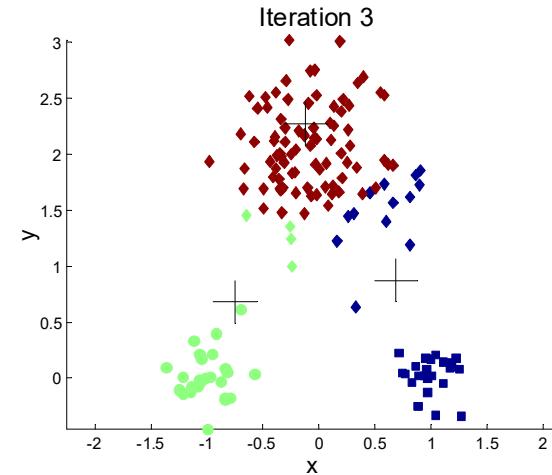
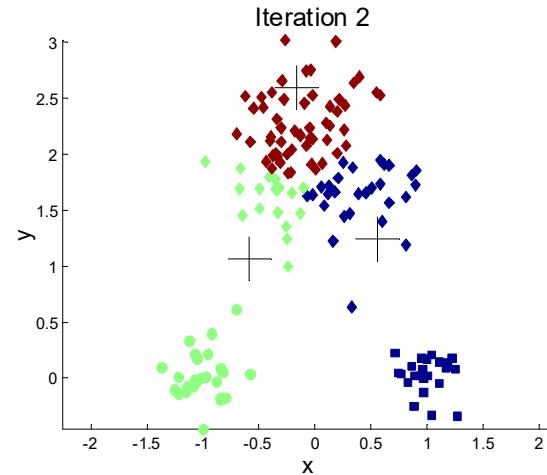
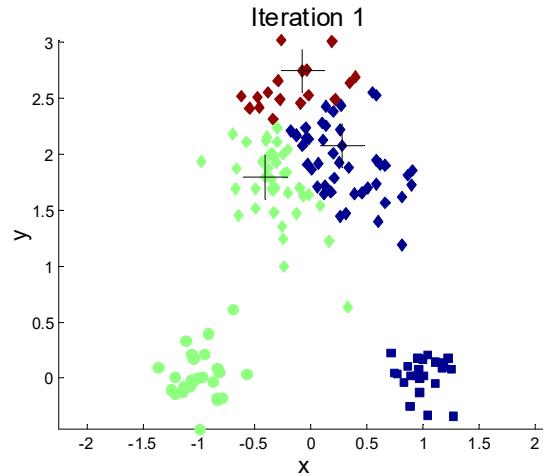
Example of K-means Clustering



مثال:

اولین بار سه تا از نمونه ها رو به عنوان نماینده در نظر بگیر به صورت تصادفی
توی این روش باید از قبل بدونیم چندتا خوشه است و بعد نماینده رو انتخاب بکنیم
و بعد کاری که میکنه براساس این که این فاصله های نمونه ها از این نماینده ها چقدر است این
نمونه ها رو رنگ میکنه --> اون داده هایی که به هر کدام از اون خوشه ها نزدیک تر باشن ما می
تونیم بگیم اون داده ها متعلق به اون خوشه هستن
و این عملیات رو مرتب تکرار می کنیم

Example of K-means Clustering



اولین بار یه تعداد نمونه رو انتخاب می کنیم به عنوان نماینده (به صورت تصادفی) و بعد میاد فاصله بقیه نمونه ها رو از اون ها حساب می کنه

وقتی نمونه ها رو متعلق دوستیم به یک حوشه جدید بعد باید ببایم یک نماینده جدیدی انتخاب بکنیم و دوباره رنگ بندی رو انجام بدیم و دوباره این مراحل تکرار میشه

K-MEANS نماینده رو به عنوان میانگین در نظر میگیره

تا یه جایی پیش می ریم که هرچی الگوریتم اجرا میشه دیگه نماینده تغییر نمی کنه --> توی این گام می گیم K-MEANS همگرا شده

الگوریتم K-MEANS یک الگوریتم گام به گام است و به صورت حریصانه عمل می کنه ینی برنمیگردد به عقب اگه اشتباهی کرد و همیشه در این جهت حرکت میکنه که اون تابع هزینه رو کمینه بکنه

K-means Clustering – Details

- Simple **iterative** algorithm.
 - Choose initial centroids;
 - repeat {assign each point to a nearest centroid; re-compute cluster centroids}
 - until centroids stop changing.
- **Initial centroids** are often chosen **randomly**.
 - Clusters produced can vary from one run to another
- The centroid is (typically) the mean of the points in the cluster, but other definitions are possible (see Table 5.2).

K-means Clustering – Details

- K-means will **converge** for common **proximity measures** with appropriately defined centroid (see Table 5.2)
- Most of the convergence happens in the first **few iterations**.
 - Often the stopping condition is changed to ‘Until relatively few points change clusters’
- **Complexity is $O(n * K * I * d)$**
 - n = number of points, K = number of clusters,
 I = number of iterations, d = number of attributes

K-means Objective Function

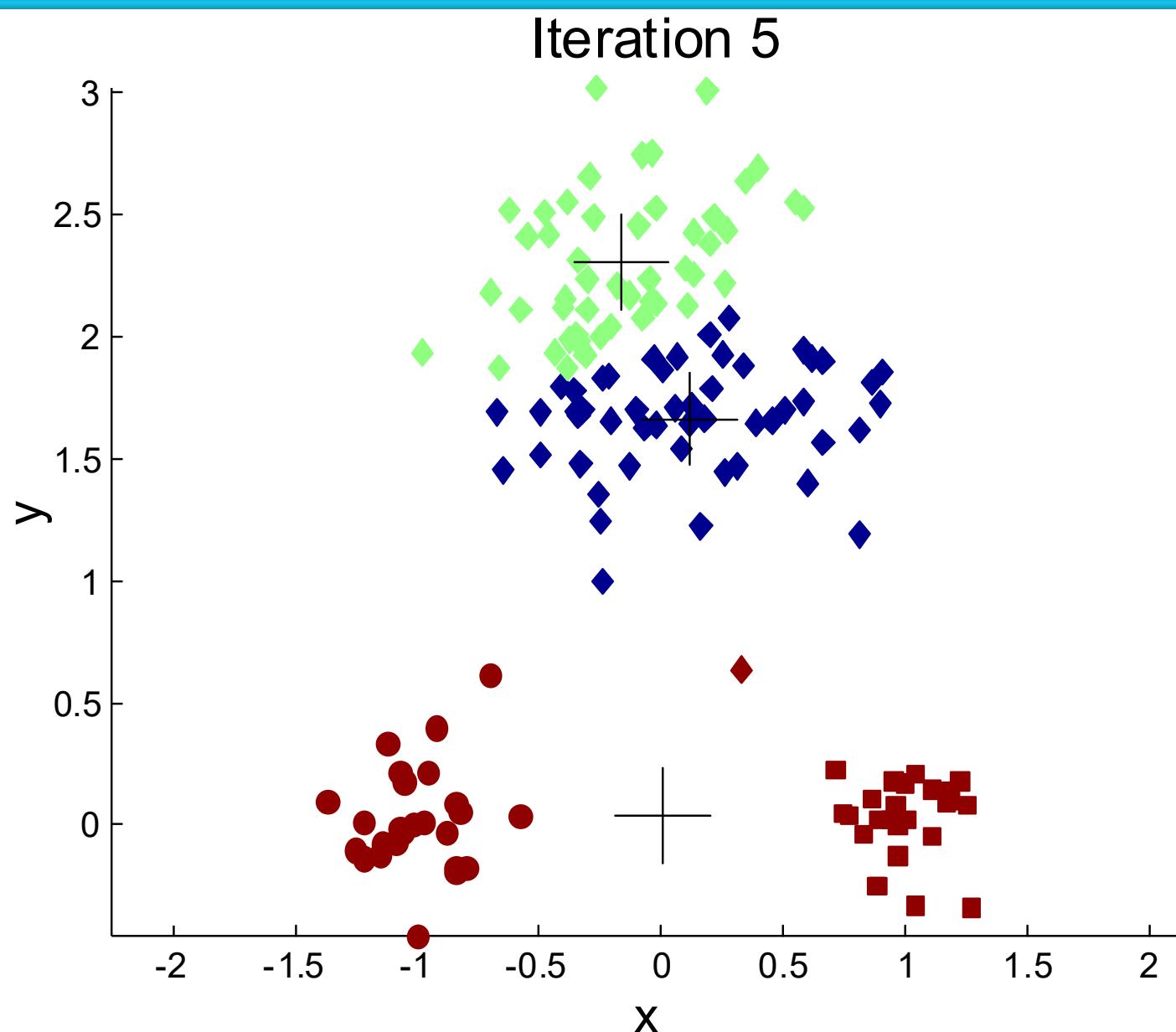
- A common objective function (used with Euclidean distance measure) is Sum of Squared Error (SSE)
 - For each point, the error is the distance to the nearest cluster center
 - To get SSE, we square these errors and sum them.

$$SSE = \sum_{i=1}^K \sum_{x \in C_i} dist^2(m_i, x)$$

- x is a data point in cluster C_i and m_i is the centroid (mean) for cluster C_i
- SSE improves in each iteration of K-means until it reaches a local or global minima.

توی هر Iteration این SSE یک مقدار میده تا اونجایی که دیگه این نمونه ها تغییر نمی کن و این SSE همگرا میشه پس این SSE می تونه یک الگویی داشته باشه

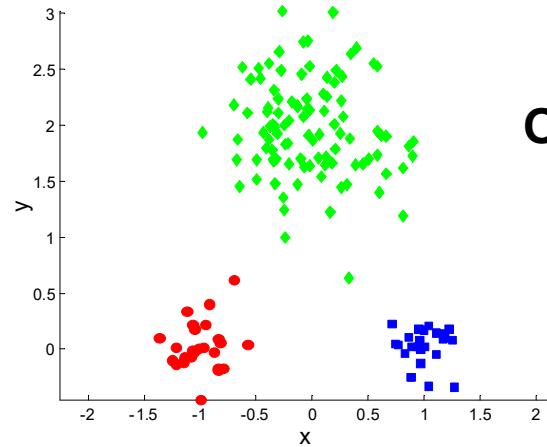
Importance of Choosing Initial Centroids ...



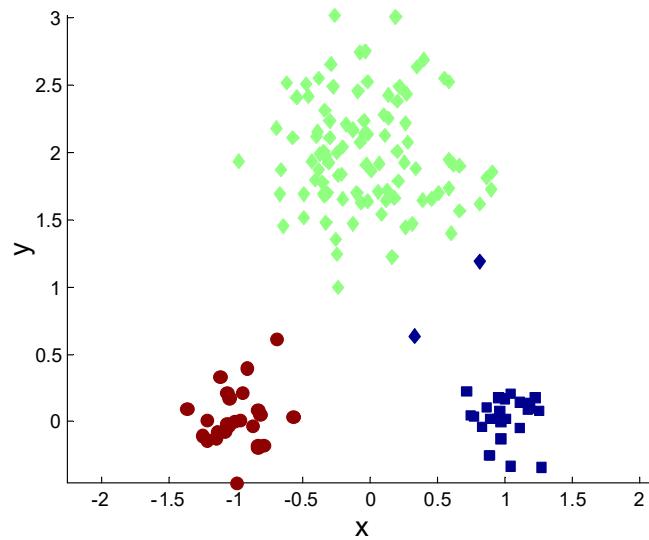
مقادیر اولیه:

مثلا برای این داده ها مقادیر اولیه رو اینطوری انتخاب کردیم:
و براساس فاصله رنگ بندی شکل گرفته

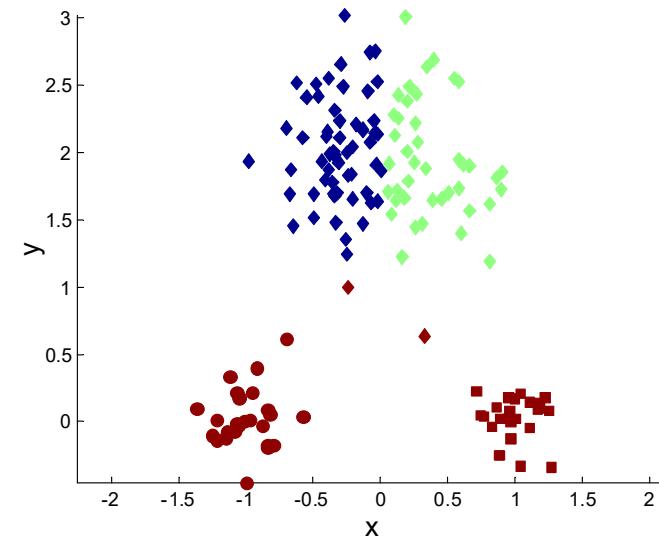
Two different K-means Clusterings



Original Points

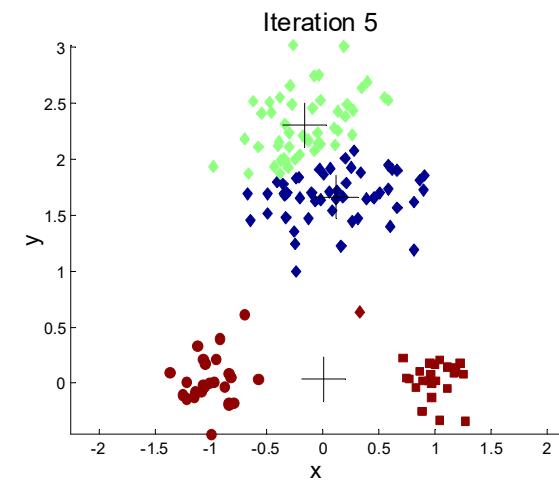
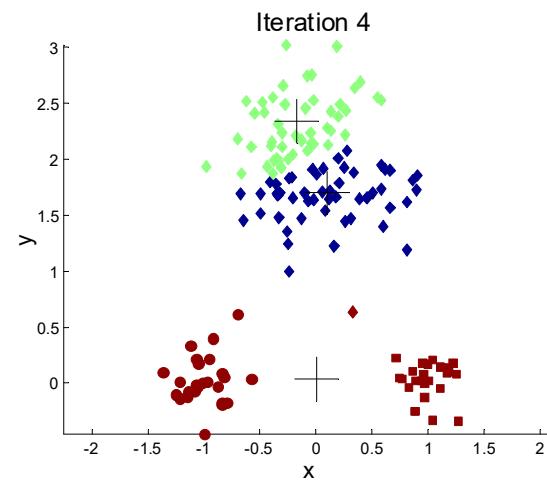
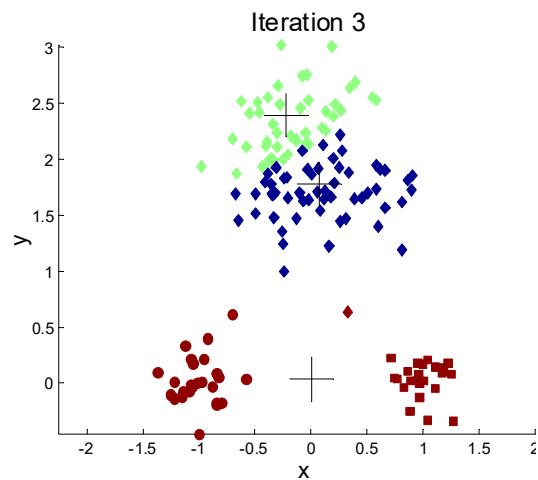
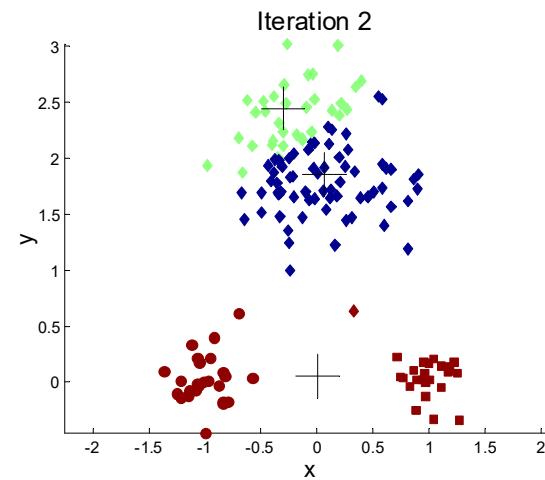
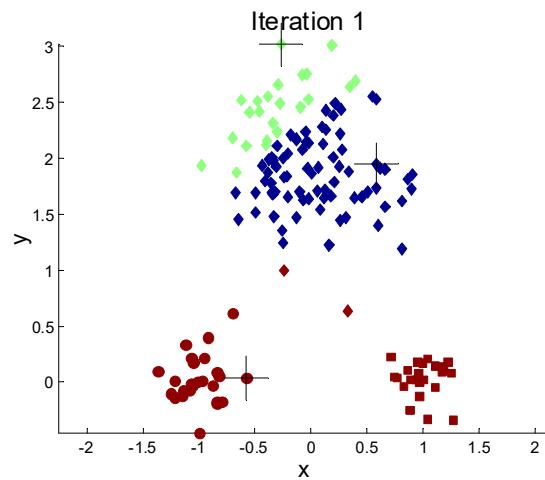


Optimal Clustering



Sub-optimal Clustering

Importance of Choosing Initial Centroids ...

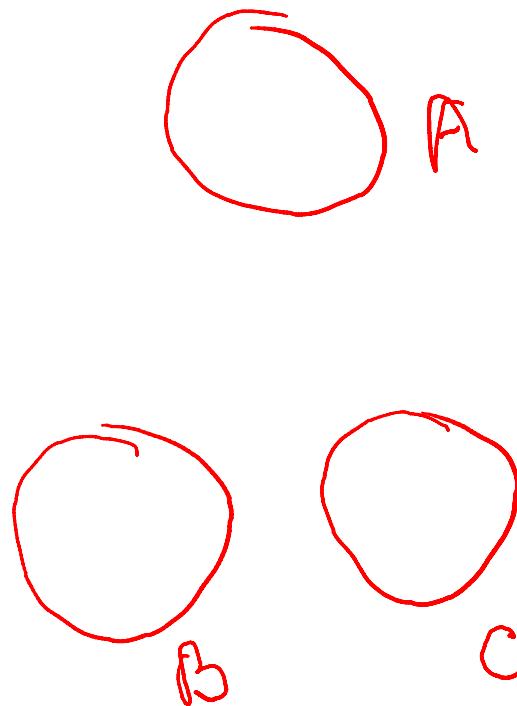


بعد از 5 Iteration دیگه اتفاقی نمی افته و هر جوری حساب بکنیم دیگه همینا می مونه و این خروجی درست نیست و این الان او مده دوتا خوشه رو به عنوان یک خوشه بهمون معرفی کرده: قرمز

و این چرا این اتفاق افتاد؟ بخاطر اون مقدار اولیه است و مقدار اولیه رو غلط در نظر گرفتیم و اگر درست در نظر می گرفتیم خوشه ها به یک سمت دیگه ای می رفتن
برای این کار باید برای نحوه انتخاب برای نمونه ها اولیه یک ساز و کار داشته باشیم که بهمون کمک می کنه که نیوفته توی این شرایط

Importance of Choosing Initial Centroids

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



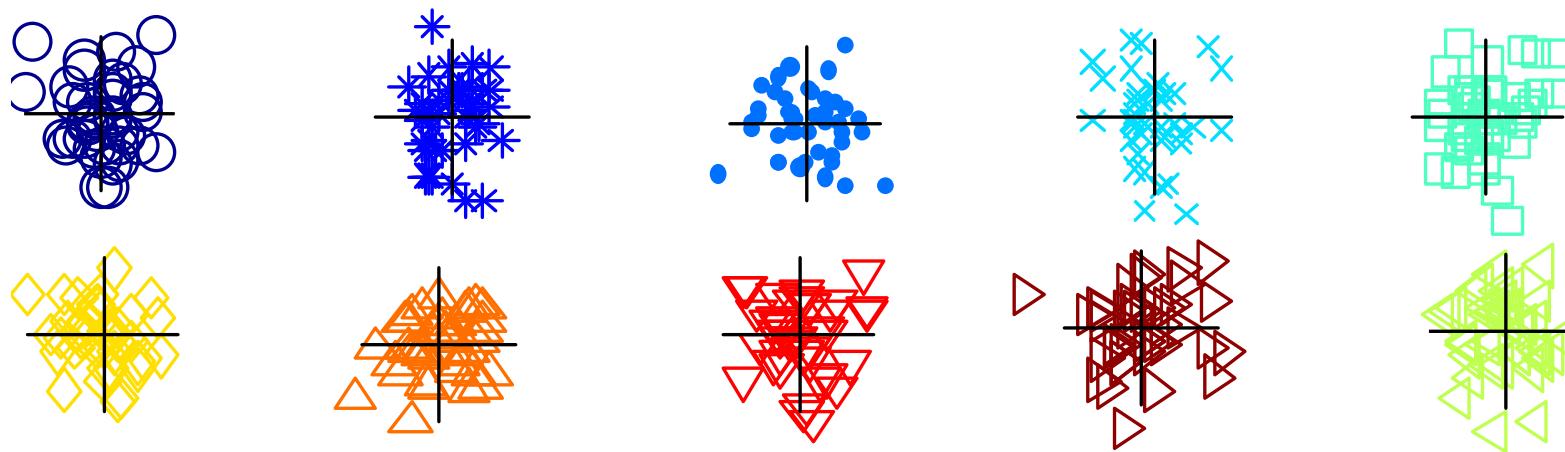
Problems with Selecting Initial Points

- 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

یک مثال دیگه:

ایا الگوریتم K-means با این مرکز خوشه ها موفقیت امیز است ؟

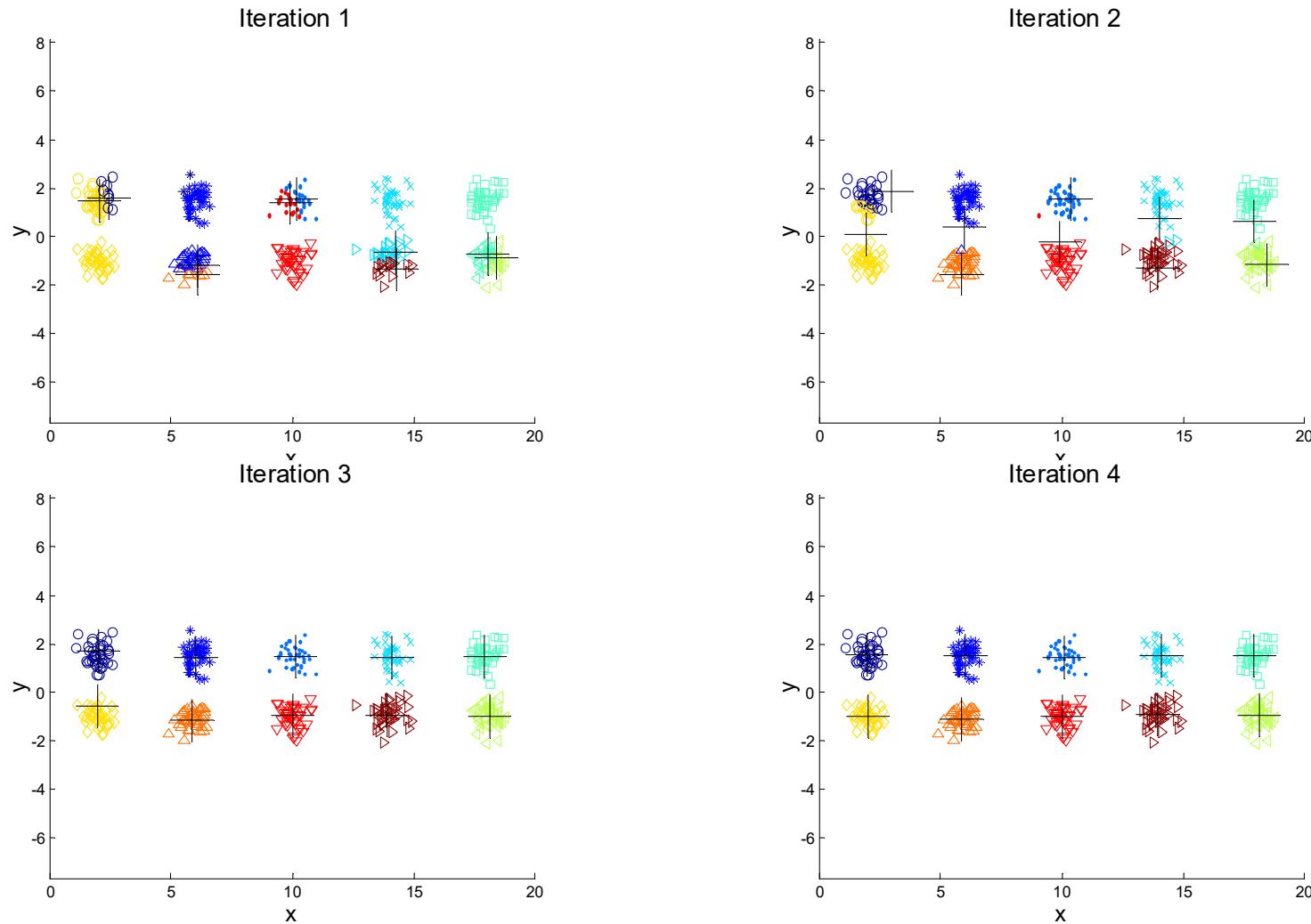
آخرش چیشد؟؟؟؟

فک کنم خوب شد

توی هر کدام از این جفت خوشه ها دو تا نقطه اولیه گذاشته بودیم و این مشکل رو حل کرد؟؟؟

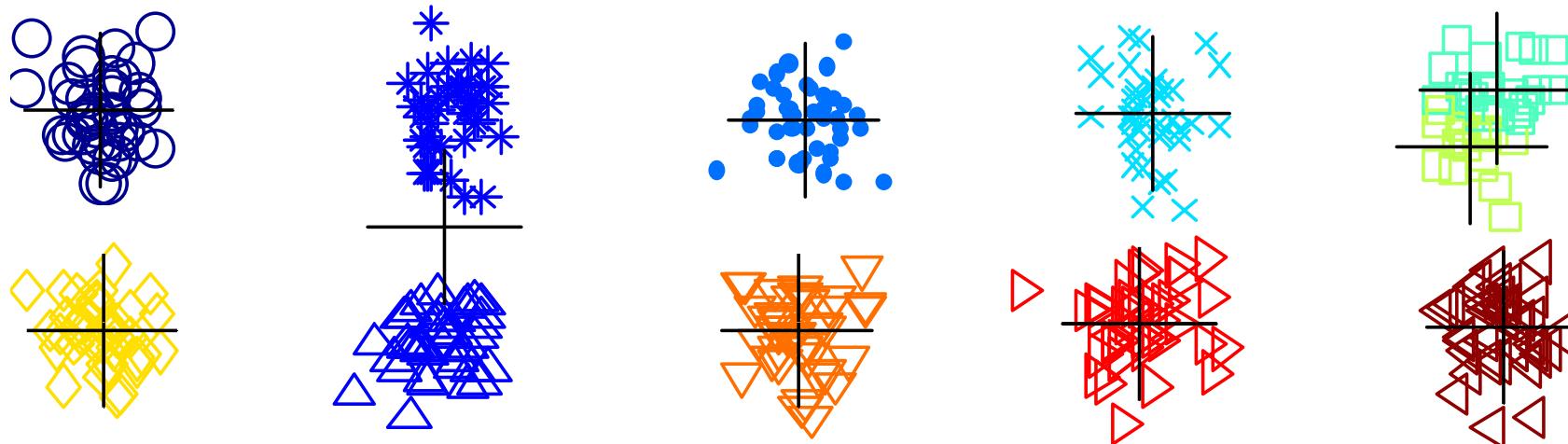
شروع با دو مرکز اولیه در یک خوشه از هر جفت خوشه

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.

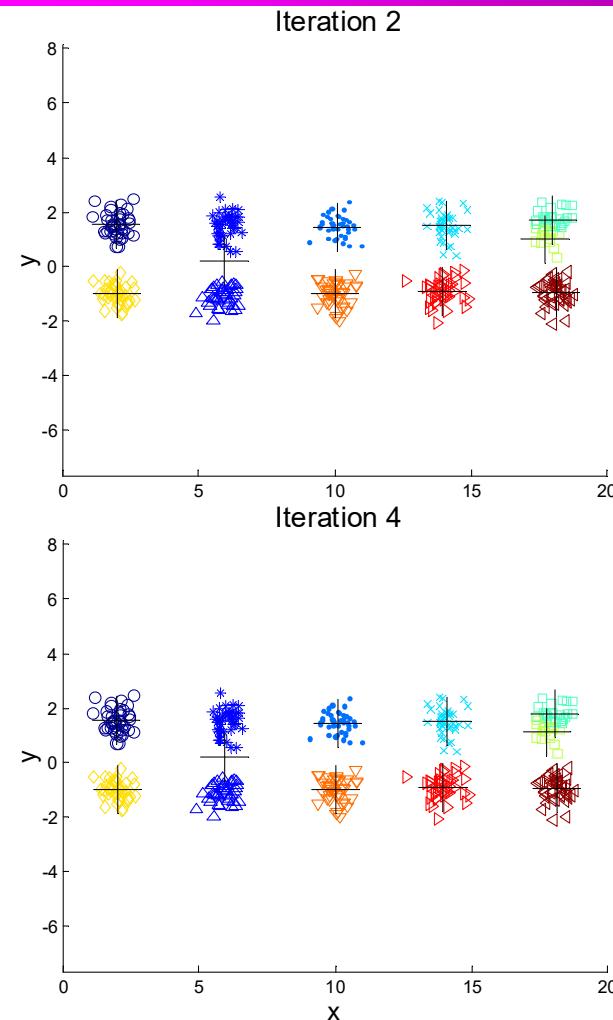
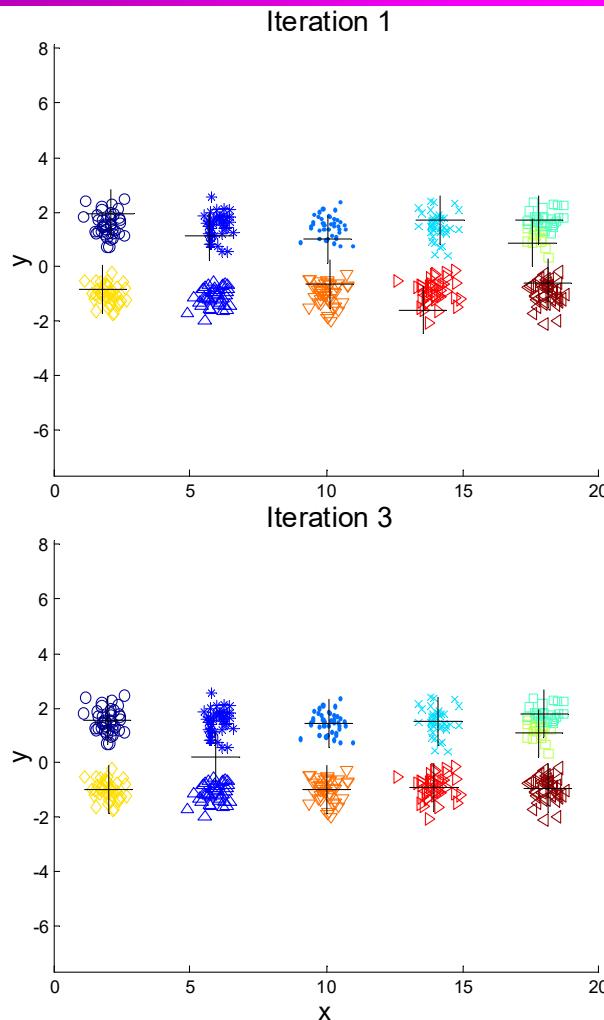
یک مثال دیگه:

توی این شرایط دیگه K-means خوب عمل نمی کنه

نکته: هم تعداد نقاط اولیه مهمه و هم موقعیت قرار گیری نقاط اولیه مهمه

با برخی از جفتهای خوش شروع میشود که دارای سه مرکز اولیه هستند، در حالی که برخی دیگر فقط یک مرکز دارند.

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

برای رفع این مشکل:

1- الگوریتم رو چندین بار با نقاط اولیه متفاوت ران بگیر و ببین میزان خطا یا SSE چقدر میشه و هر کدوم SSE کمتری داشت ینی به نتیجه بهتری رسیدیم ولی این راه حل خوبی نیست چون زمان بر است

2- روش K-means^{++} یک نسخه توسعه یافته K-means است
نقاط اولیه که داری در نظر میگیری یه جور انتخاب بکن که بهم نزدیک نباشن ینی همون اول کار که یک نقطه ای اومد اجازه نمی ده یک نقطه مجاورش دوباره نقطه اولیه بشه و برو یه جای دوری این نقطه اولیه رو قرار بده ینی یه کاری بکن که این نقاط اولیه فاصلشون رو از هم حفظ بکن و بهم نزدیک نباشن

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 $d(x)^2$.
 - 5: Pick new centroid from the remaining points using the weighted probabilities.
 - 6: **end for**
-

: ++K-means

میاد یک احتمالی در نظر میگیره بینی نمونه های که می خواهد به عنوان نقطه اولیه انتخاب بشه با یک احتمالی مرتبط میشه که اون احتمال فاصله اون نمونه است از نقطه اولیه

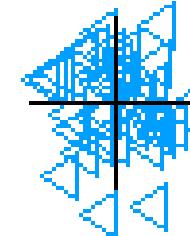
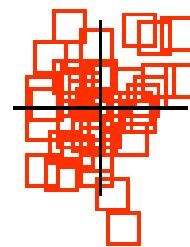
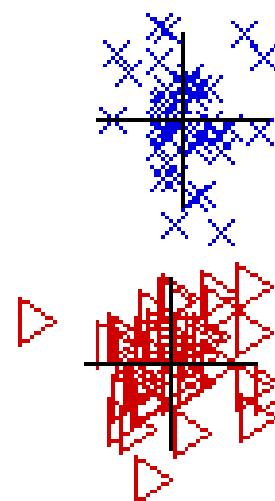
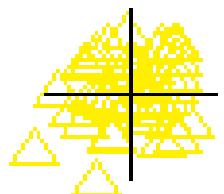
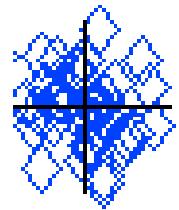
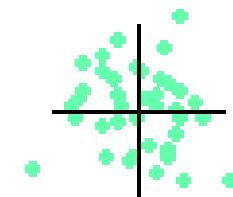
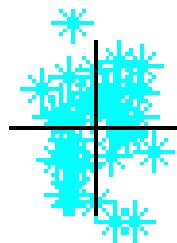
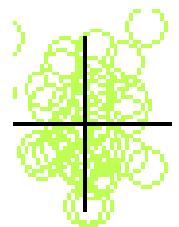
Bisecting K-means

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

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

CLUTO: <http://gilaros.dtc.umn.edu/gkhome/cluto/cluto/overview>

Bisecting K-means Example



Limitations of K-means

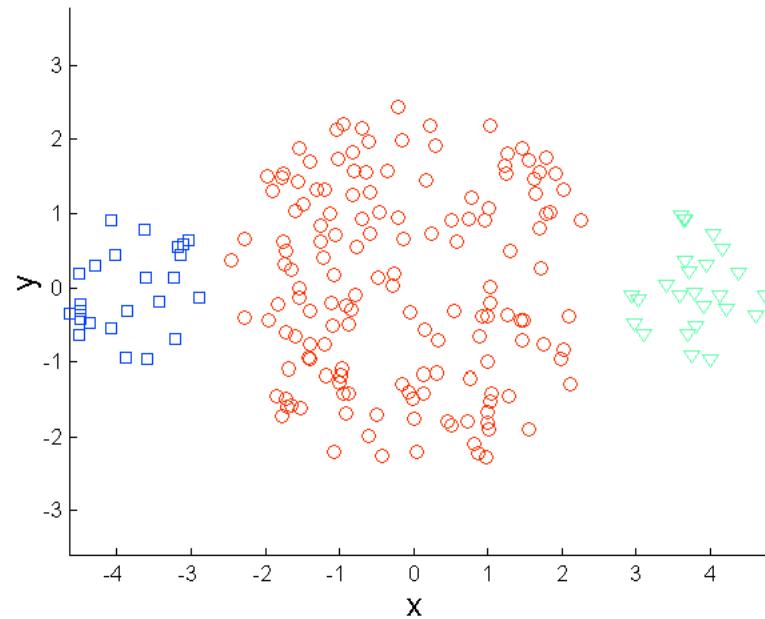
- K-means has problems when clusters are of differing
 - Sizes
 - Densities
 - Non-globular shapes
- K-means has problems when the data contains outliers.
 - One possible solution is to remove outliers before clustering

مشکلاتی که K-means داره:

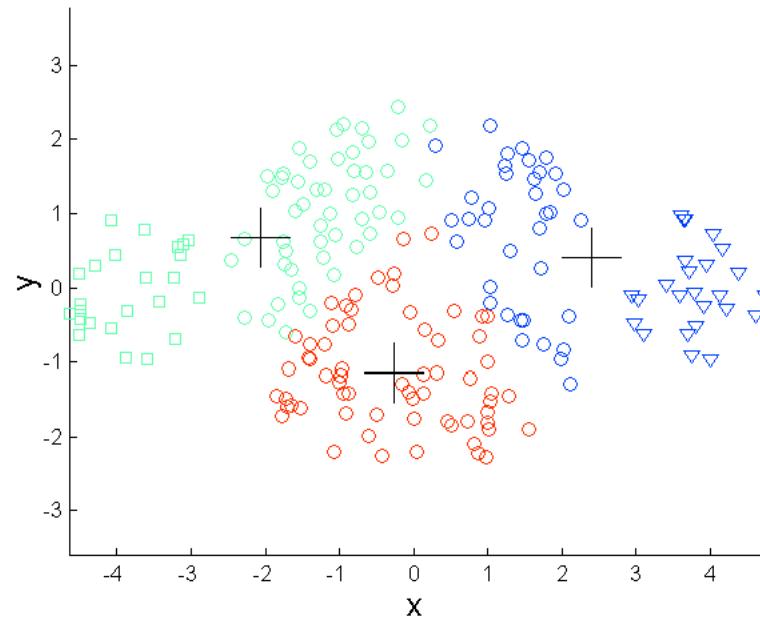
K-means وقتایی که خوشه ها از لحاظ سایز با هم متفاوت باشن یا چگالی هاشون با هم فرق بکنه یا شکل خوشه ها کروی نباشه مشکل پیدا میکنه --> توی همچنین موقعی نباید بریم سراغ K-means

اگر توی دیتا outliers داشتیم نباید اون K-means رو در نظر بگیریم چون K-means میکنه

Limitations of K-means: Differing Sizes



Original Points

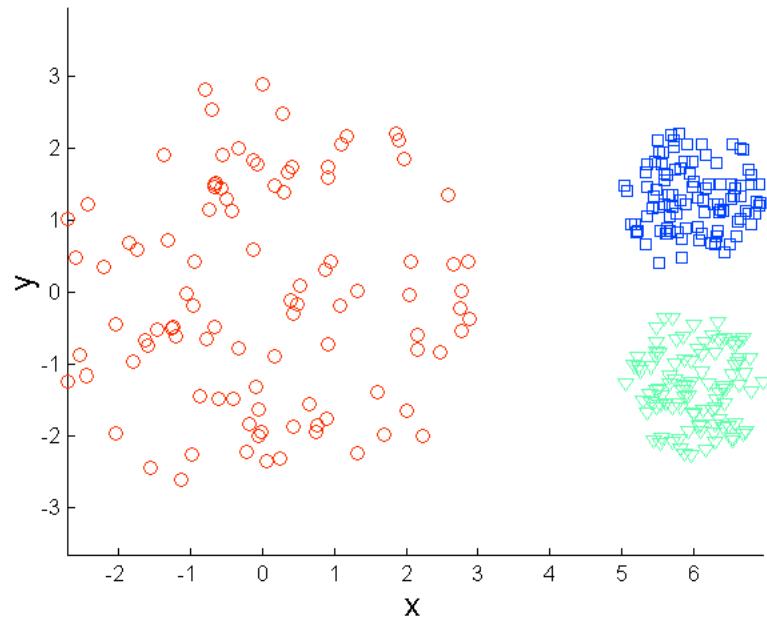


K-means (3 Clusters)

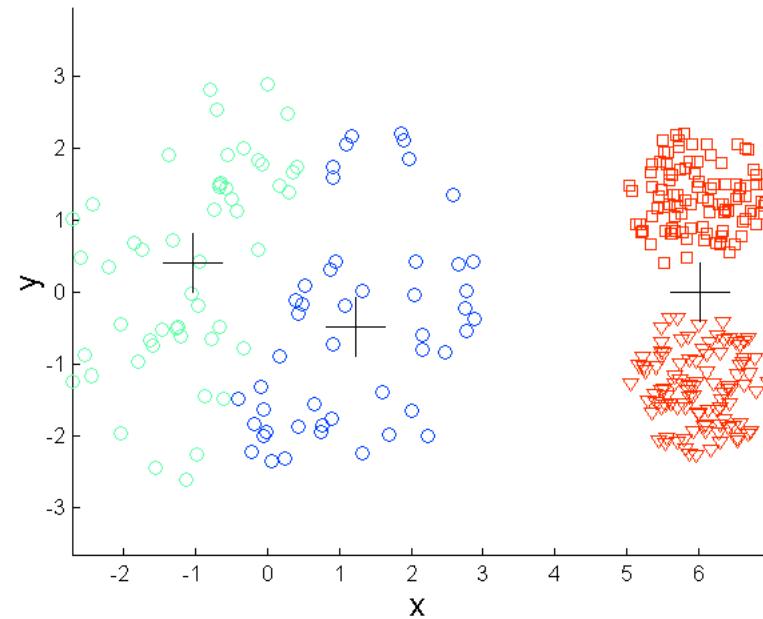
فرض کنیم همچین دیتایی داریم:

اگر K-means روی این دیتا اجرا بکنیم یک چیزی خیلی محتمل است که داخلش رخ بده ینی نقاط اولیه جوری قرار می گیرن که تعداد نمونه هایی که توی هر دسته هست با هم برابر باشه و این مشکل چرا ایجاد شد؟ چون یک خوشه ای داشتیم که 100 تا نقطه داشت و دو تا خوشه دیگه 20 تا نقطه داشت و چون تعداد اون یکی خوشه ای زیاد بود احتمال اینکه همه نقاط اولیه توی اون خوشه بزرگه باشن زیاده و خوشه ها هم روی اینا انتخاب بشه و این مشکل به وجود بیاد اگر خوشه ها از لحاظ تعداد نمونه داخلیش با هم برابر بود احتمال اینکه این اتفاق بیوفته خیلی کمتر بود

Limitations of K-means: Differing Density



Original Points

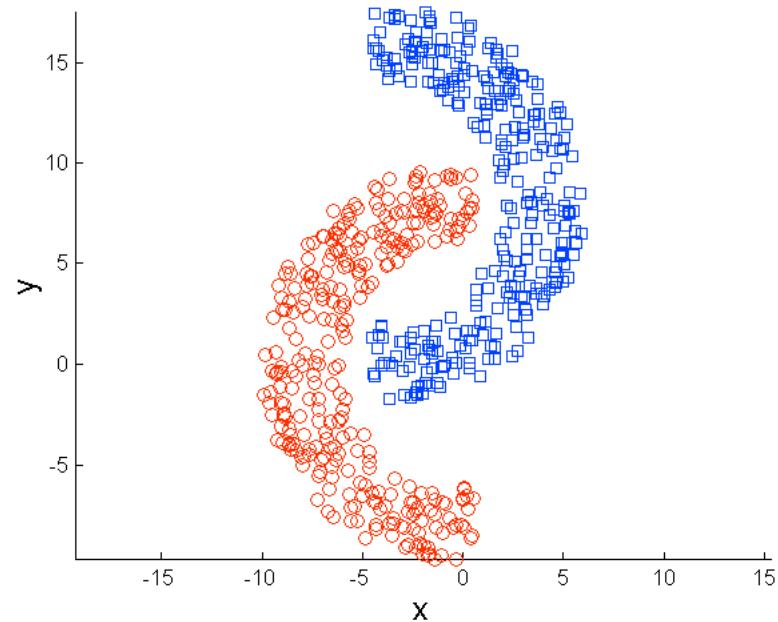


K-means (3 Clusters)

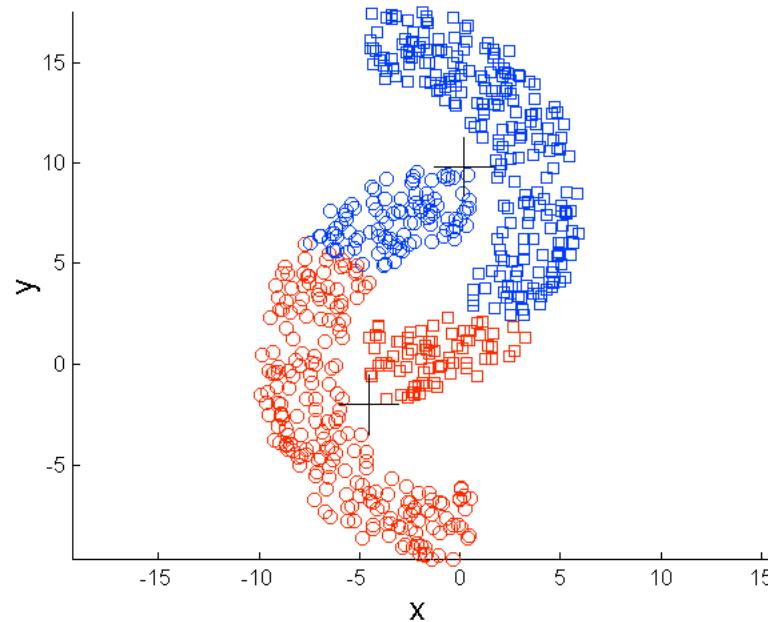
یک مشکل دیگه هم اینه که چگالی اینا با هم متفاوت باشه ???

توی این حالت هم K-means رو اجرا بکنیم باعث میشه که چگالی ها نقطه اولیه رو سمت خودشون ببرن و حول یک نقطه گیر می کنه نقطه اولیشون و میانگین رو می کشن به سمت خودشون و در نهایت بخاطر تراکمی که اینجا داریم این دو تا میشن یک خوشه کافیه که یک نقطه اولیه بیوپته توی اون نقطه پرتراکم در این حالت زود برچسب رو میگیره و نمونه ها متعلق به اون خوشه میشه

Limitations of K-means: Non-globular Shapes



Original Points



K-means (2 Clusters)

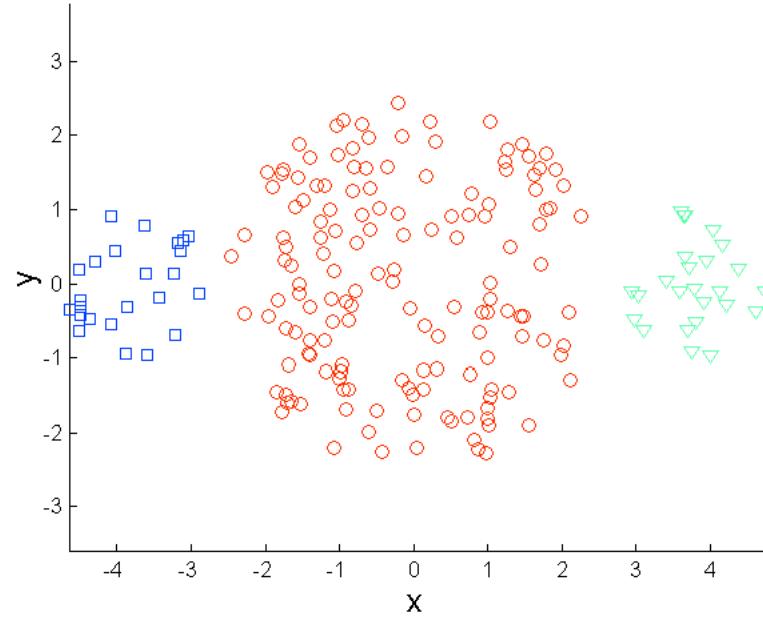
مشکل بعدی:

این از وسط نصف میشه چرا؟ چون میانگین می خواد بگیره
اینجا خوش به صورت کروی نیست و خم است

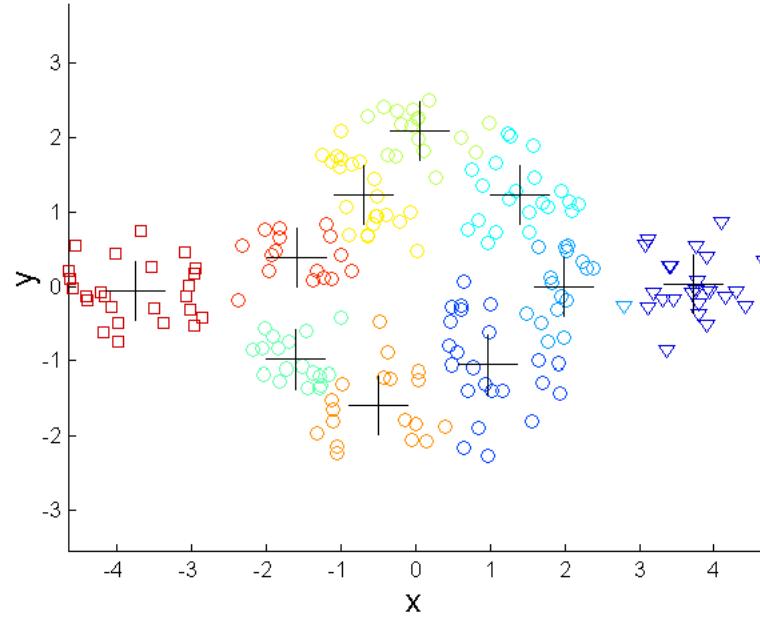
راهکارها:

یک بخشیش میشه همون centroid یا نقطه مرکزی یا نقطه اولیه پیدا کردن
یک بخش دیگه هم ما بباییم خوش به ها رو زیاد در نظر بگیریم

Overcoming K-means Limitations



Original Points



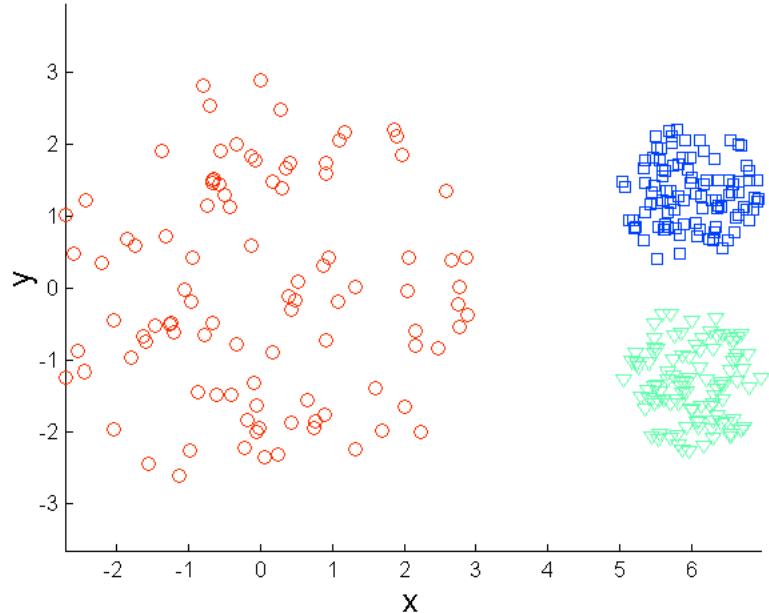
K-means Clusters

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.

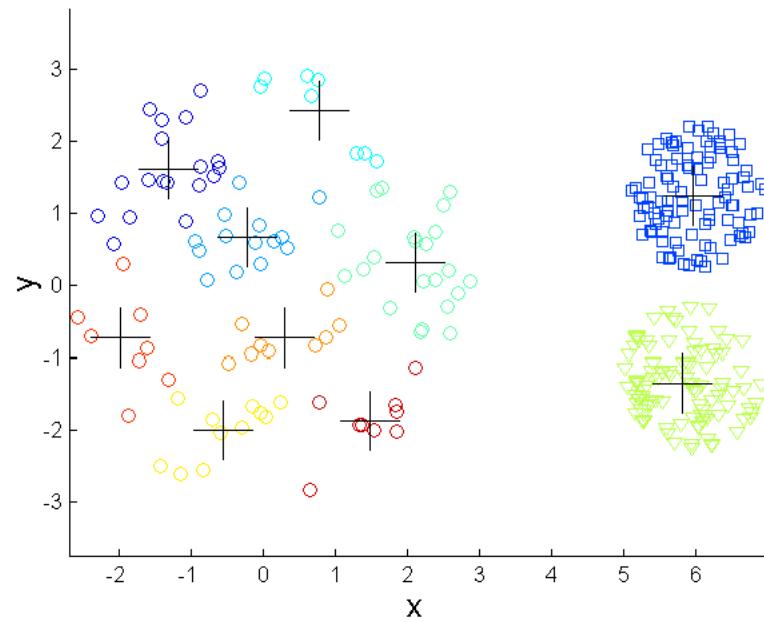
یک راه حل این است که تعداد زیادی خوشه پیدا کنیم به طوری که هر یک از آنها بخشی از یک خوشه طبیعی را نشان دهد. اما این خوشه های کوچک باید در یک مرحله پس از پردازش کنار هم قرار گیرند

تعداد خوشه ها زیاد شده : راهکار

Overcoming K-means Limitations



Original Points

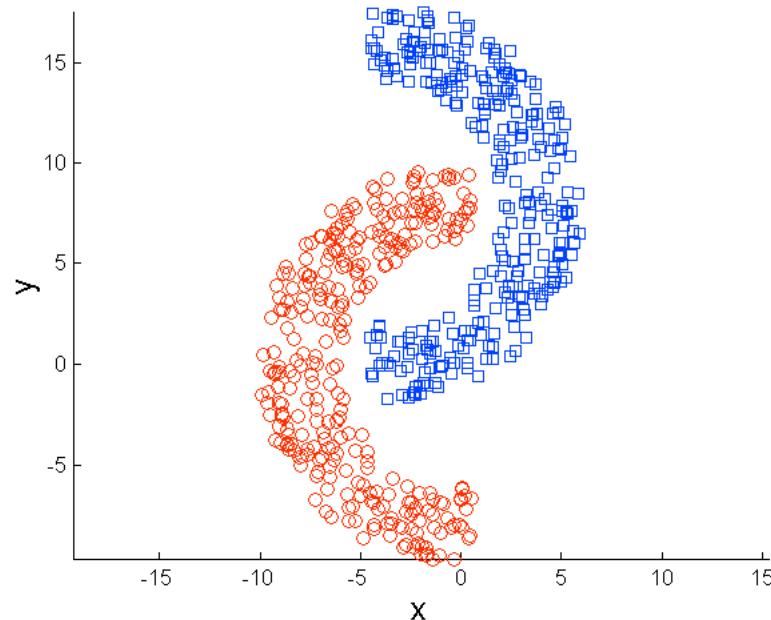


K-means Clusters

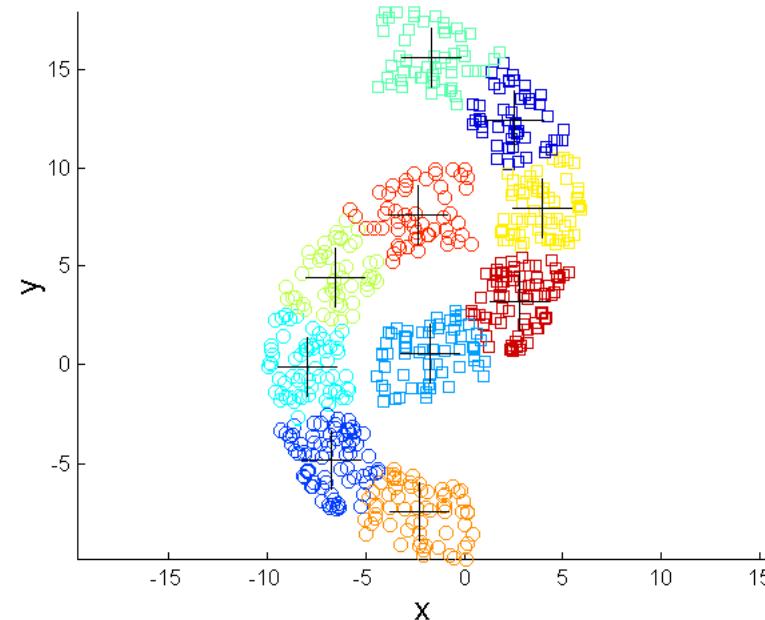
One solution is to find **a large number of clusters** such that each of them represents a part of a natural cluster. But these small clusters need to be **put together in a post-processing step**.

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



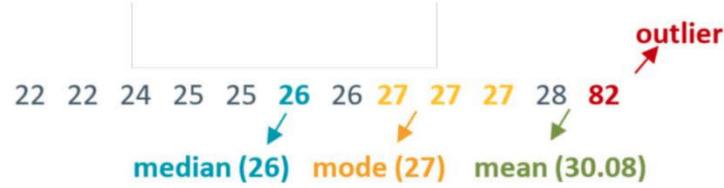
K-means Clusters

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.

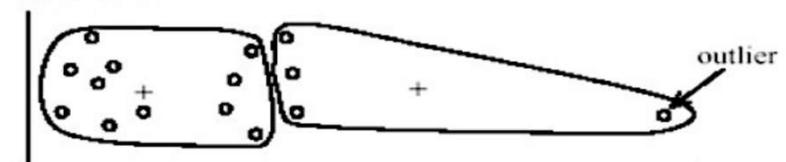
یک راه حل این است که تعداد زیادی خوشه پیدا کنیم به طوری که هر یک از آنها بخشی از یک خوشه طبیعی را نشان دهد. اما این خوشه های کوچک باید در یک مرحله پس از پردازش کنار هم قرار گیرند

K-means and outlier!

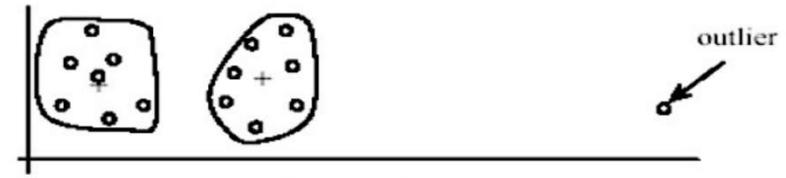
- K-Medoids: Instead of taking the mean value of the object in a cluster as a reference point, medoids can be used, which is the most centrally located object in a cluster



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



(A): Undesirable clusters



(B): Ideal clusters

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

چه بلایی سر K-means میاره؟ outlier

با خاطر حضور این outlier مرکز این خوشه می‌افته توی یک فضای دوری ینی فضایی که اصلاً تراکم داده نداریم و همین کار رو خراب می‌کنه

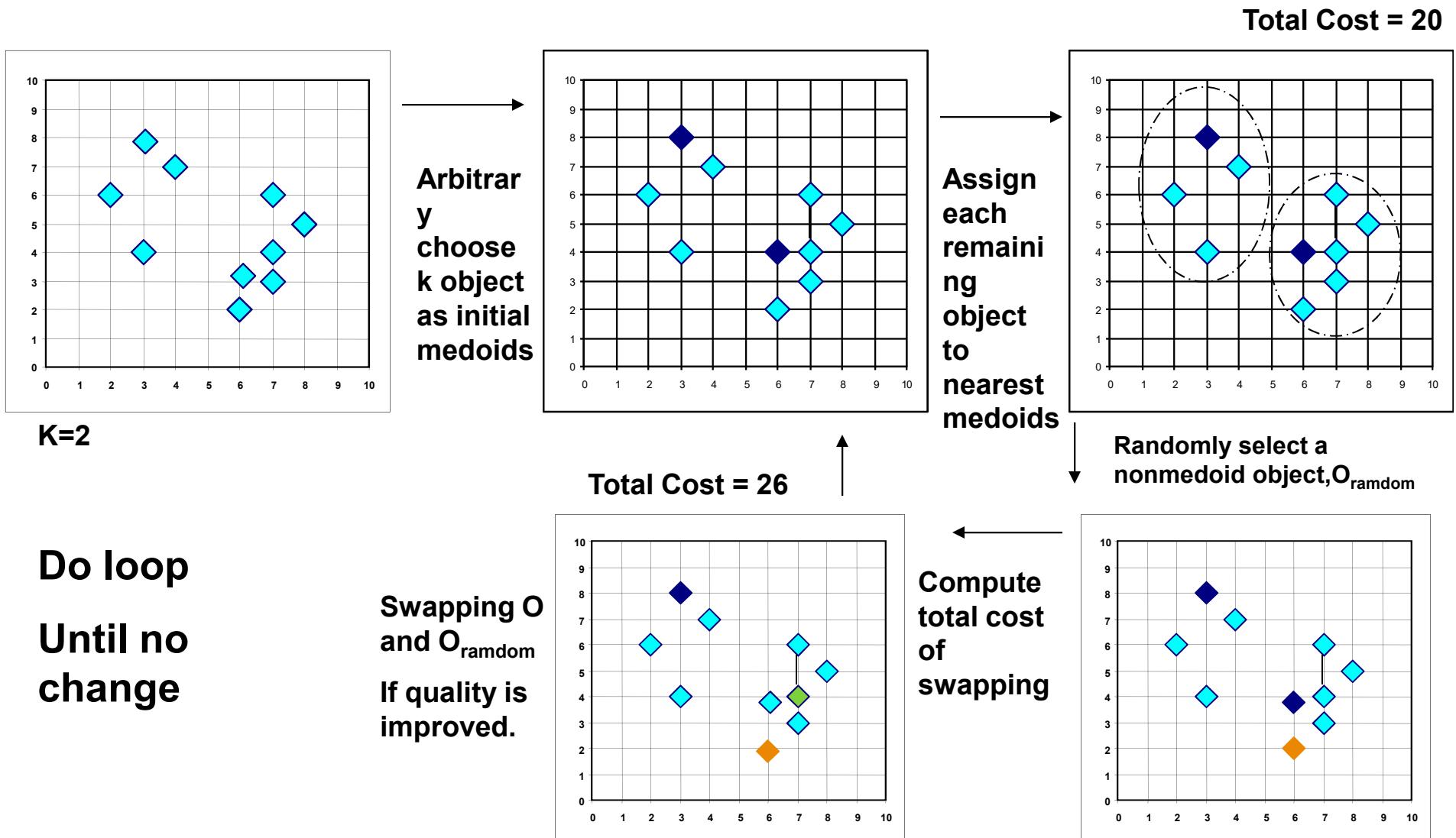
توی روش‌های خوشه بندی مخصوصاً K-means نگاه بکنیم اون مرکزی که پیدا شده اصلاً حولش نمونه است اگر حولش نمونه نبود ما چار یک شرایط outlier شدیم و برای اینکه این مشکل پیش نیاد نسخه K-Medoids گفته می‌شده استفاده بشه

توی روش K-Medoids به جای میانگین میان میانه رو می‌گیریم و میانه نسبت به این نویز خیلی مقاوم‌تر است و این باعث می‌شده نماینده بهتری داشته باشیم و اون Outlier رو در نظر نگیریم

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

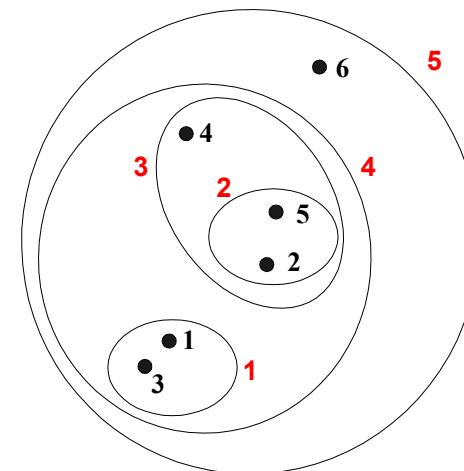
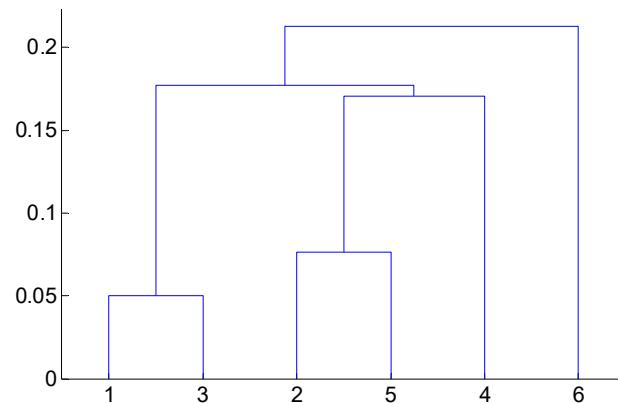
PAM: A Typical K-Medoids Algorithm



HIERARCHICAL CLUSTERING

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



الگوریتم سلسله مراتبی:

اینجا خوشه ها بهم ربط پیدا می کنن

کنار این روش خوشه بندی سلسله مراتبی همیشه یک نموداری می دن به اسم dendrogram این نمودار dendrogram نماینده این روش خوشه بندی است و باهاش همه کاری میشه کرد نمودار :dendrogram

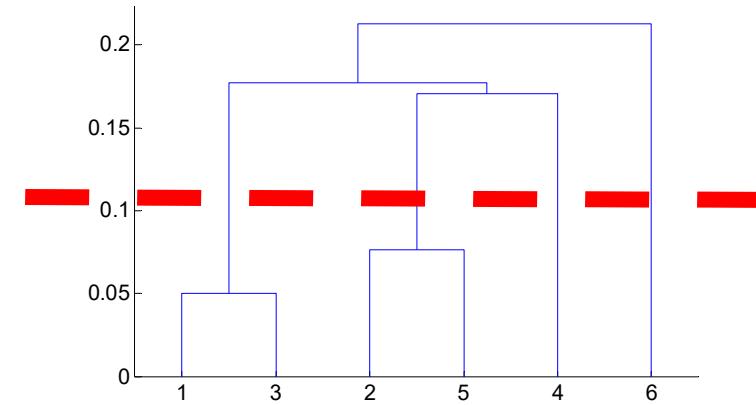
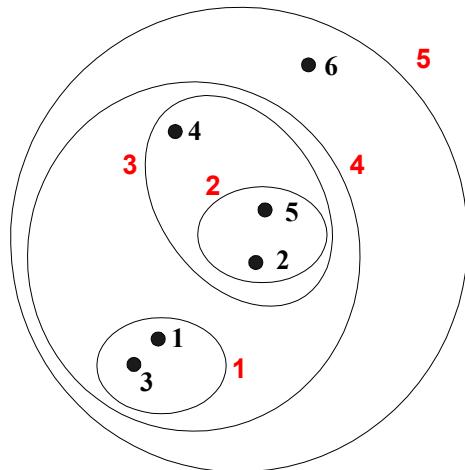
توی محور X شماره نمونه ها است

و توی محور y فاصله این نمونه ها از هم

مثلًا الان نمونه 1 و 3 توی یک خوشه قرار گرفته اند و مثلًا 2 و 5 با 4 یک خوشه شده است

Strengths of Hierarchical Clustering

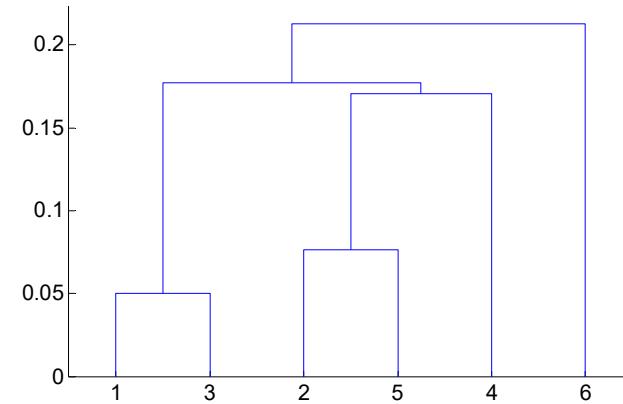
- Do not have to assume any particular number of clusters
 - Any desired number of clusters can be obtained by ‘cutting’ the dendrogram at the proper level



این خط توی 4 نقطه نمودار dendrogram رو قطع کرده چون ما 4 تا خوشه می خواستیم و می خواستیم بدونیم کدو ما میشه در واقع: الان اینجا 1 و 3 توی یک خوشه و 2 و 5 توی یک خوشه دیگه و 4 یک خوشه و 6 هم یک خوشه که در کل میشه 4 تا خوشه که با این خط فهمیدیم می تونیم با بالا و پایین کردن این خط تعداد خوشه ها رو تغییر بدیم یک چیز جالب ما می تونیم به تعداد نمونه ها خوشه داشته باشیم ینی لول پایینش میگه به ازای تعداد نمونه ها می تونیم خوشه داشته باشیم

Strengths of Hierarchical Clustering

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



یک ویژگی که راجع به نمودار dendrogram وجود دارد: خیلی به مفهوم سلسله مراتب دانشی هم نزدیک تر است مثلاً میخوایم یک تعداد میوه رو دسته بندی بکنیم میاد میوه های شبیه هم رو توی یک خوش می ذاره و ...

Hierarchical Clustering

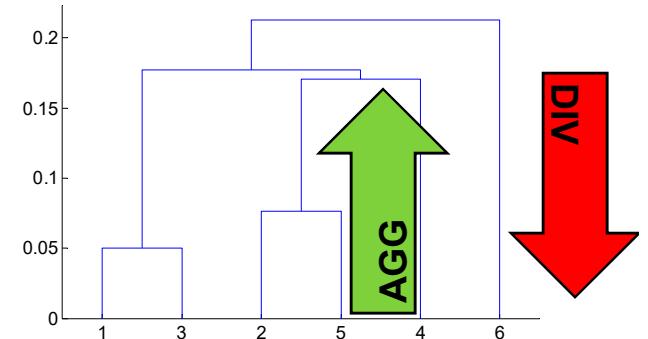
- Two main types of hierarchical clustering

- **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)



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

ایجاد نمودار:

دوتا مسیر هست برای تولید نمودار: وقتی نمودار رو می سازیم (نمودار ابی) اینکه نمودار از پایین به بالا باشه یا بر عکس این دو روش میشه روش های Agglomerative

نمودار از پایین رشد میکنه یعنی هر نمونه میشه به عنوان یک نماینده از یک خوشه و ما سعی میکنیم اینجا نمونه های که بهم نزدیک هستن رو توی یک خوشه تجمعی بکنیم و همینطوری بریم بالا و توی ارتفاق درخت یک خوشه داریم

روش های Divisive

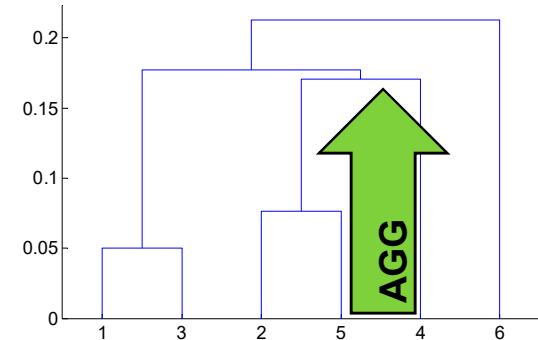
اینجا بر عکسه یعنی یک خوشه داریم که شامل همه نمونه ها هست و الگوریتم به ما میگه کجا این خوشه رو تقسیم بکنیم

توی کلاس بیشتر روش Agglomerative میگیم

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

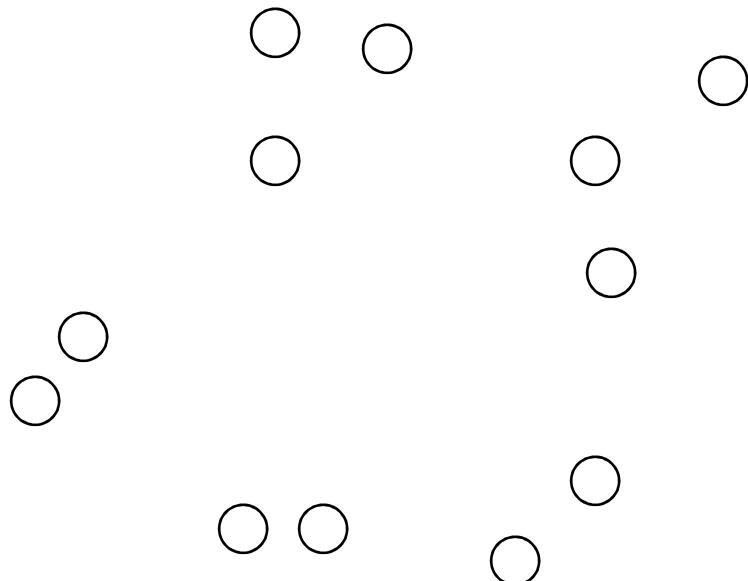
الگوريتم Agglomerative

يک حلقه داره و يک روند تكراريه و اين حلقه اينقدر تكرار ميشه که برسيم به يک خوش
اول به تعداد نمونه ها يک خوش داريم و گام به گام خوش هايی که بهم نزديک هستن تركيب ميكنيم
و خوش ها با هم تركيب مي شه کم کم تا برسيم به يک خوش
چطوری خوش ها با هم تركيب بكنيم؟

ترکیب کردن خوش ها براساس فاصله بین نمونه های اون خوش ها هست
فاصله نمونه ها رو با يک ماتريسي می سنجيديم به اسم ماتريس فاصله يا شباهت --> که اندازه می
کرد هر نمونه با نمونه ديگه چقدر فاصله داره
با اين ماتريس خيلي کار داريم و براساس اين همه کارها صورت ميگيره و براساس اين ماتريس
تصميم می گيريم که کدام خوش ها با هم مرج بشن

Steps 1 and 2

- Start with clusters of individual points and a proximity matrix

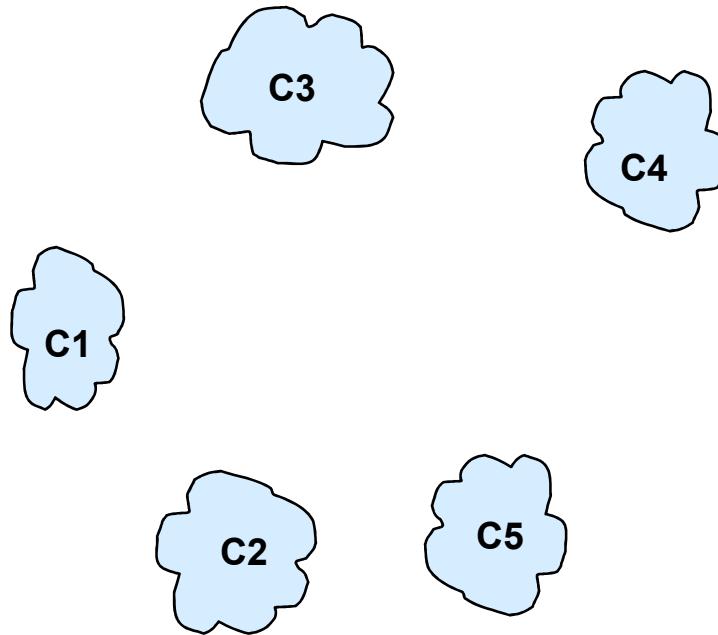


	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
.						
.						
Proximity Matrix						

p1 p2 p3 p4 ... p9 p10 p11 p12

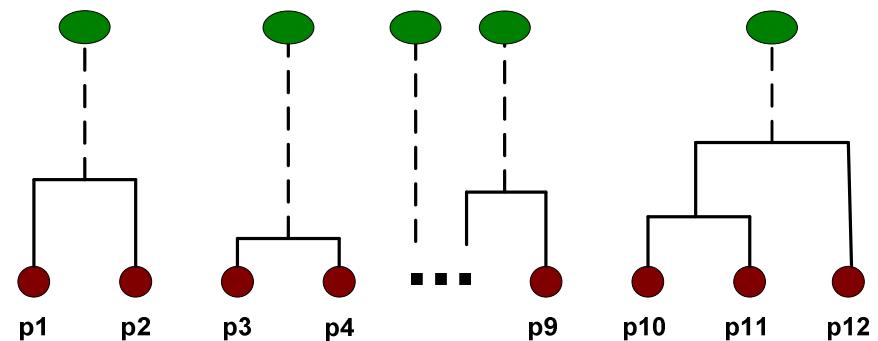
Intermediate Situation

- After some merging steps, we have some clusters



	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

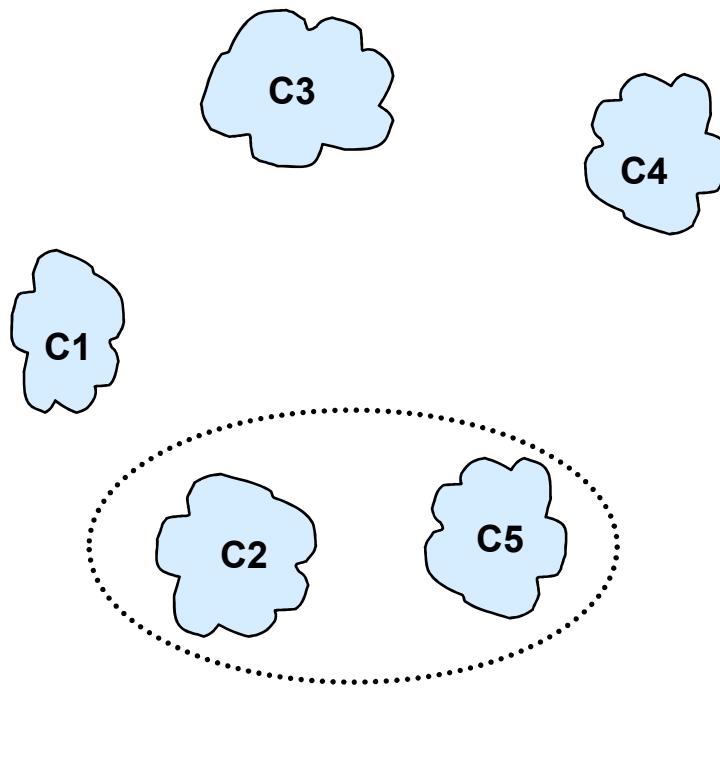
Proximity Matrix



این می خواهد فاصله هر نمونه با نمونه های دیگر را حساب بکنه و ماتریس رو پر بکنه
الان کدام یکی از این خوشه ها را با هم ترکیب بکنیم؟
کدام نمونه نزدیک تر است به نمونه دیگر نسبت به بقیه و اینارو میخوایم یک خوشه می کنیم
مثلًا توانی این مثال شد ۵ تا خوشه

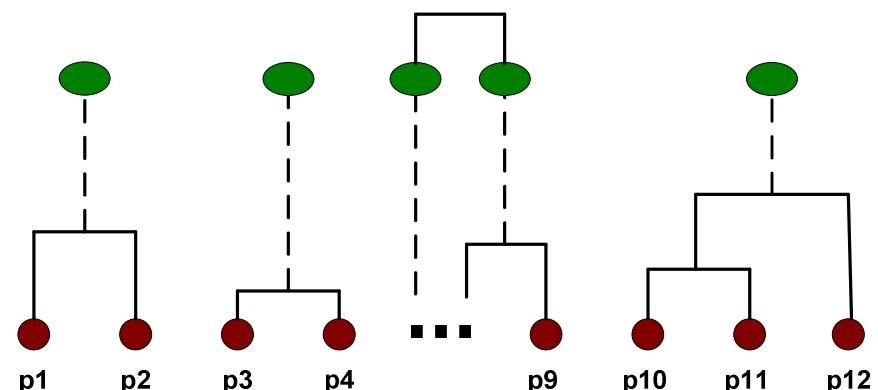
Step 4

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



	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

Proximity Matrix

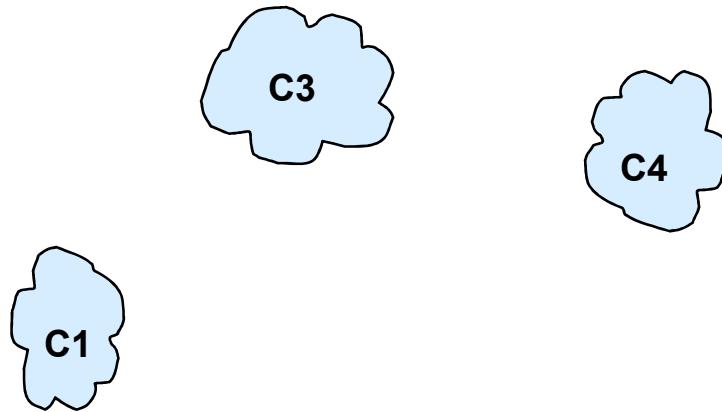


الان ماتریس باید راجع به این 5 تا خوشه بحرفه
الان براساس فاصله بین خوشه ها باید تصمیم بگیریم
ماتریس فاصله بین خوشه ها رو چطوری حساب بکنیم ؟
اینجا میخوایم خوشه رو با خوشه مقایسه بکنیم
معیار هایی داریم اینجا که جلوتر میگه

براساس ماتریس تصمیم گرفتیم خوشه 2 و 5 رو با هم ترکیب بکنیم و این ماتریس مجاورت قرار
نیست همه المان هاش تغییر بکنه فقط دوتا از خوشه ها دچار تغییر شدن پس سطر و ستون های 2 و
5 تغییر می کنه و باید اینارو اصلاح بکنیم

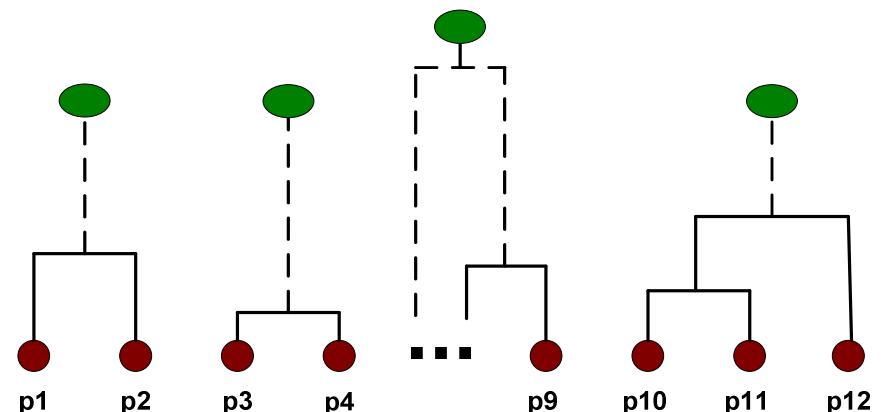
Step 5

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



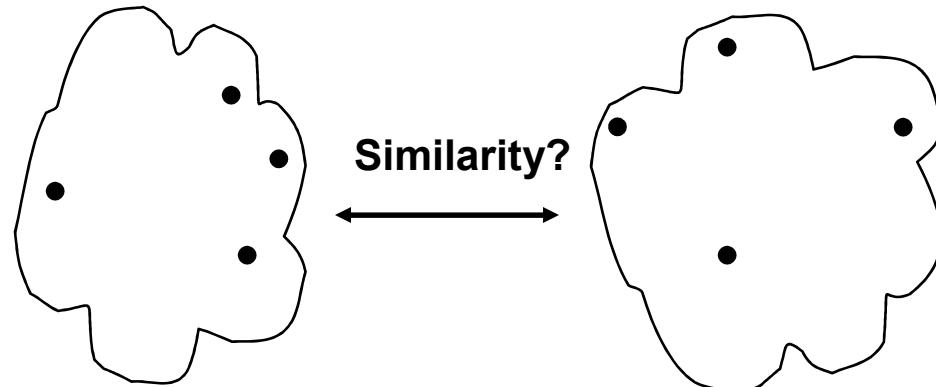
		C2 U C5	C1	C5	C3	C4
		C1	?			
C2 U C5		?	?	?	?	?
		C3	?			
		C4	?			

Proximity Matrix



ماتریس مجاورت هی داره کوچیک کوچیک تر میشه
اولین بار به تعداد نمونه ها سطر و ستون داره و بعد هی کم کم میشه

How to Define Inter-Cluster Distance

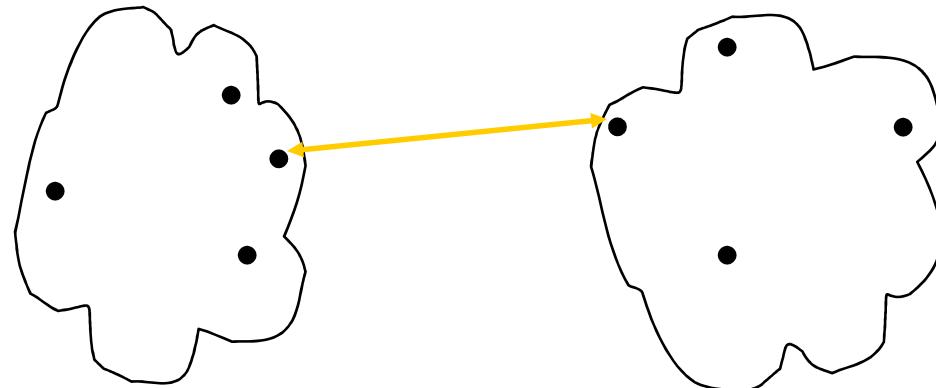


	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
.						

- Proximity Matrix

فاصله بین دو خوش چجوری؟

How to Define Inter-Cluster Similarity



- MIN or Single Link

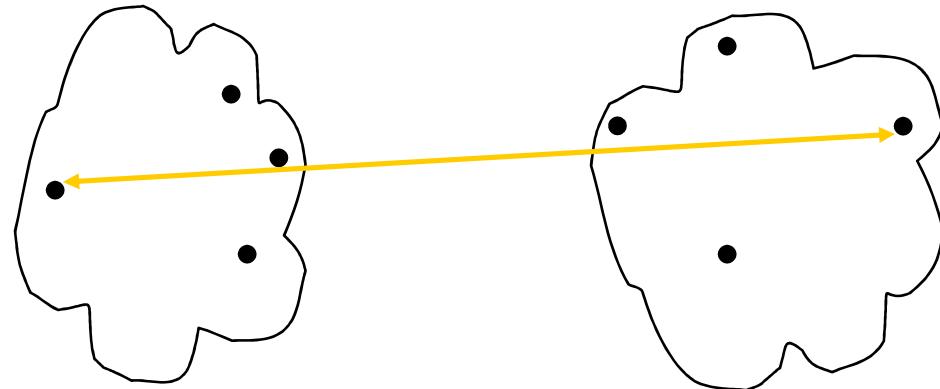
	p1	p2	p3	p4	p5	...
p1						
.						

- Proximity Matrix

:MIN or Single Link روش

وقتی فاصله بین دو تا خوش رو داریم حساب میکنیم میگه بیا بین نزدیکترین نمونه ها توی این خوش چیه و این میشه فاصله بین دو تا خوش

How to Define Inter-Cluster Similarity



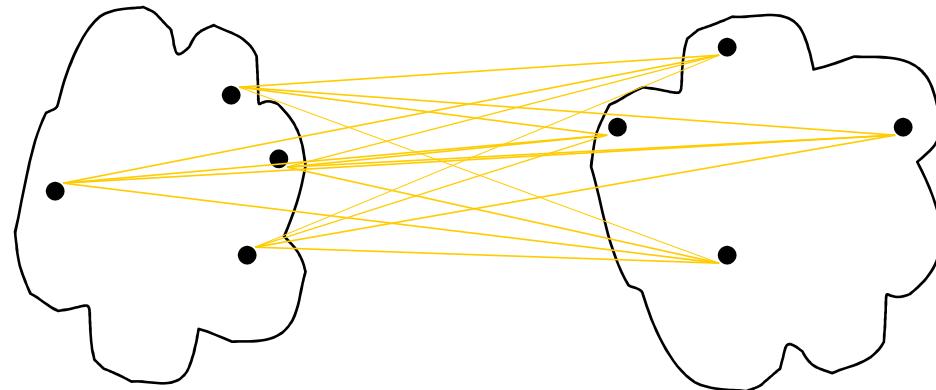
- MIN or Single Link
- MAX or Complete Linkage

	p1	p2	p3	p4	p5	...
p1						
.

▪ Proximity Matrix

روش :MAX or Complete Linkage
دورترین نمونه توی خوشه ها میشه معرف فاصله بین دوتا خوشه

How to Define Inter-Cluster Similarity



- MIN or Single Link
- MAX or Complete Linkage
- Group Average

	p1	p2	p3	p4	p5	...
p1						
.

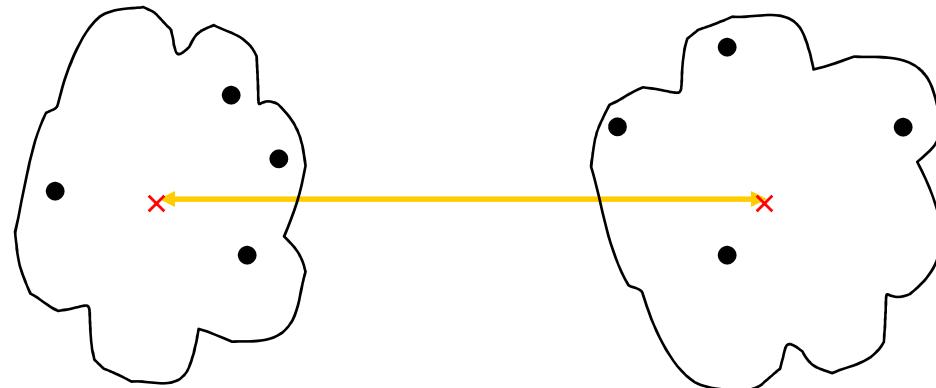
· Proximity Matrix

روش Group Average

فاصله بین نظیر به نظیر می سنجیم --> میشه 16 تا فاصله و بعد بیا میانگین بگیر

این چه فرقی داره با فاصله بین میانگین ها؟ ایا یک جواب میده؟
اینجا داریم فاصله بین نمونه به نمونه می سنجیم و بعد میایم میانگین می گیریم

How to Define Inter-Cluster Similarity



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

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

· Proximity Matrix

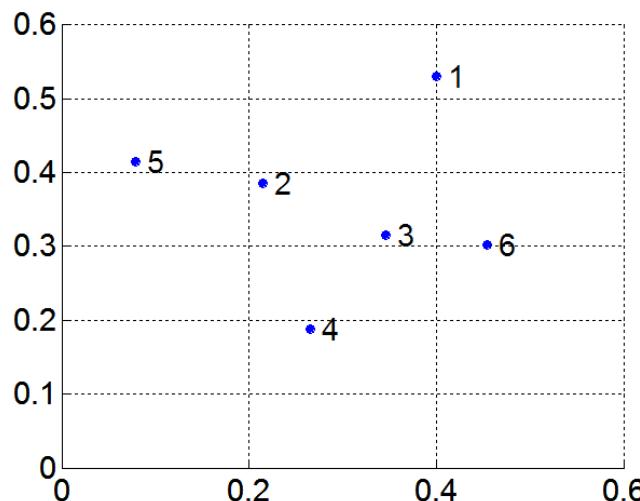
روش :Distance Between Centroids

یک معیار دیگه اینه که نقطه مرکزی در نظر بگیریم ینی با یک تعریفی مرکز رو مشخص بکنیم که این مرکز ممکنه خودش نمونه باشه یا میانگین باشه یا یک معیار دیگه و براساس اون فاصله رو مشخص بکنیم

یک روش دیگه هم هست که با یک objective کار می کنن:
ینی نیا لزوما با یک نقطه کار بکنه و همه نقاط رو بده به دست یک تابعی و این تابع براساس کار خودش میاد یک مقدار میده و بعد با اون عدد بیا کار بکن

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:

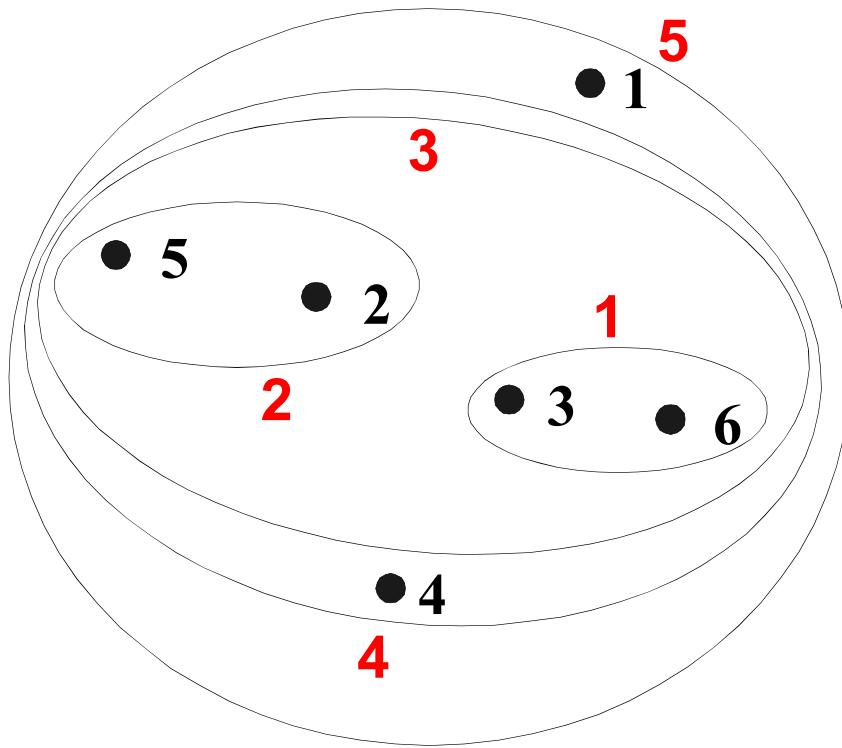


Distance Matrix:

	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

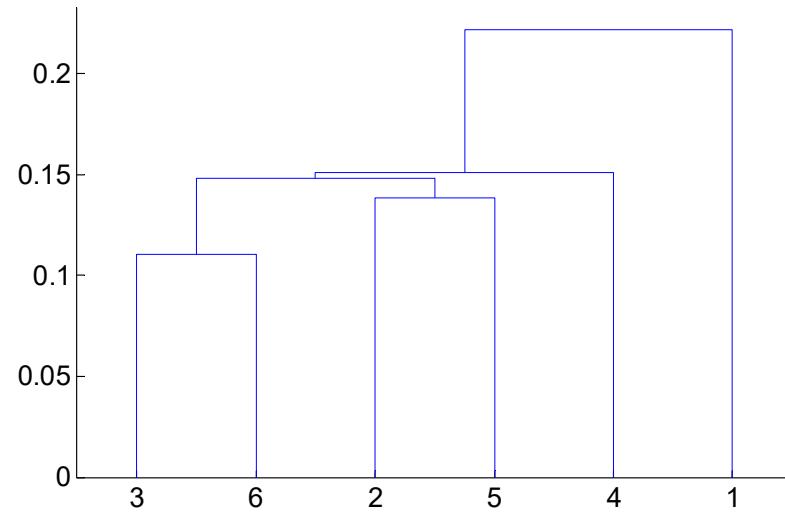
الان 3 و 6 باید اپدیت بشه --> کمترین مقدار بین تمامی ستون ها و سطر ها غیر از دو ستون بالا دوباره میایم کمترین مقدار رو توى ستون ها پیدا میکنیم که این هست 0.14 بین نمونه ها میشه این البته و حالت بعدی هم اینه که بیایم نمونه ها رو با خوشه درست شده از مرحله قبل بسنجیم --> حالت دومش ینی بیایم 3 رو با بقیه بینیم که هست 0.15 کمترینش پس همون 0.14 انتخاب میشه

Hierarchical Clustering: MIN



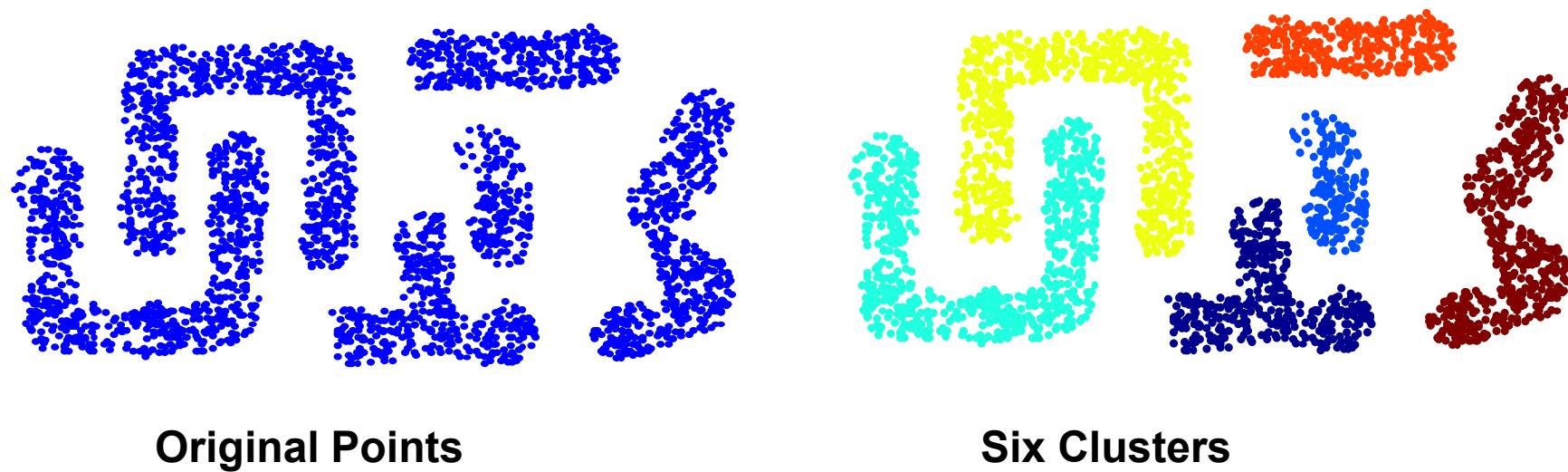
Nested Clusters

	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



Dendrogram

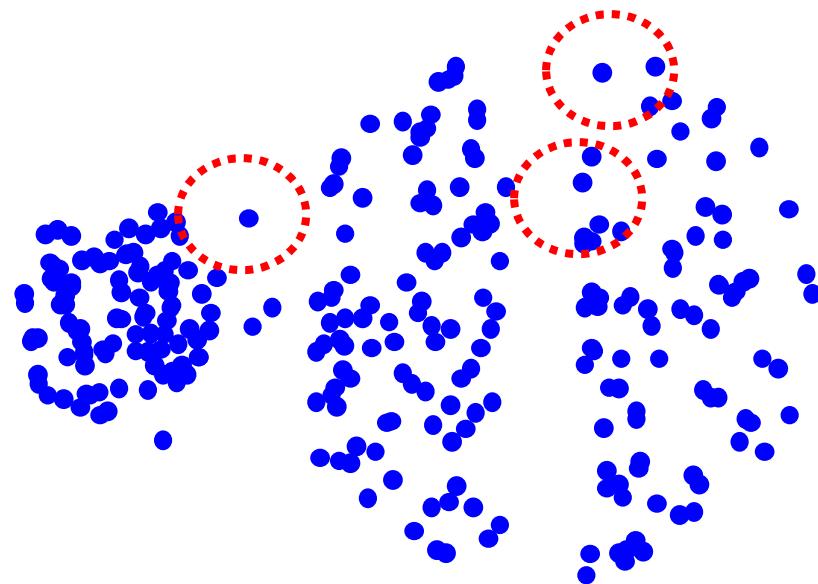
Strength of MIN



- Can handle non-elliptical shapes

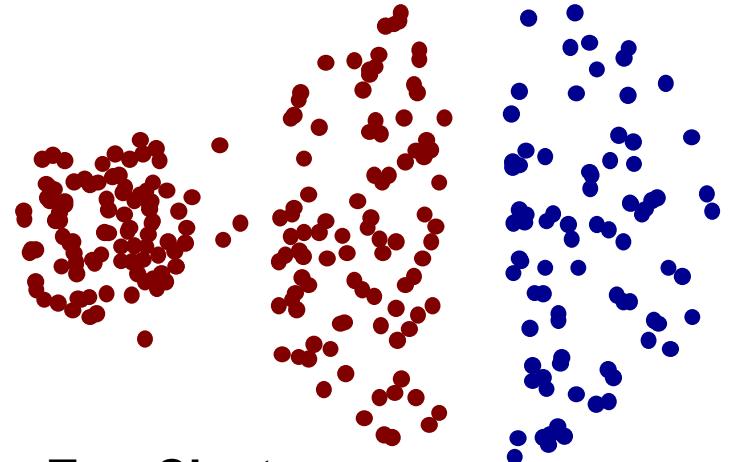
• می تواند اشکال غیر بیضوی را اداره کند

Limitations of MIN



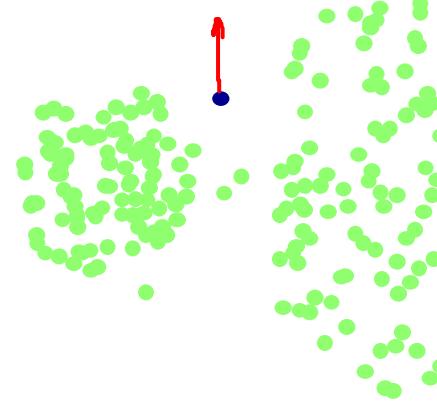
Original Points

- Sensitive to noise



Two Clusters

اين اخرا سر مياد مرج ميشه واسه همين تک افتاده

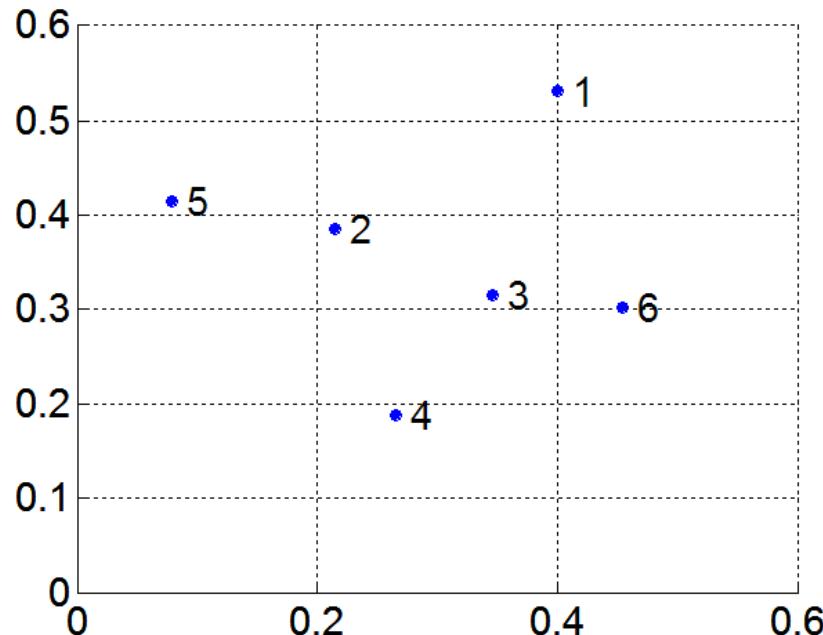


Three Clusters

حساس به نویز

MAX or Complete Linkage

- 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

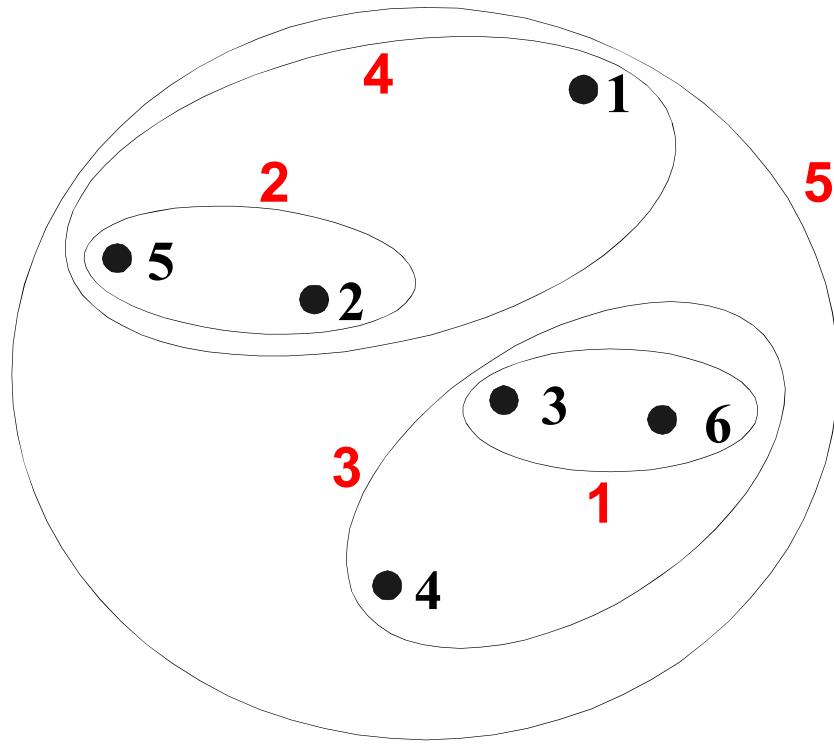


Distance Matrix:

	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

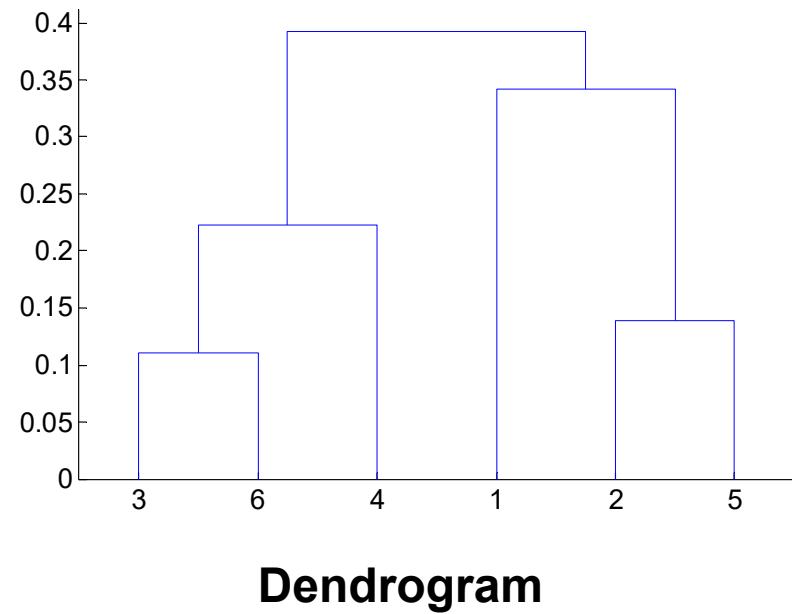
توی همه اینا همشون اولشون یک خوشه است --> اینو بپرس مطمئن بشی؟

Hierarchical Clustering: MAX



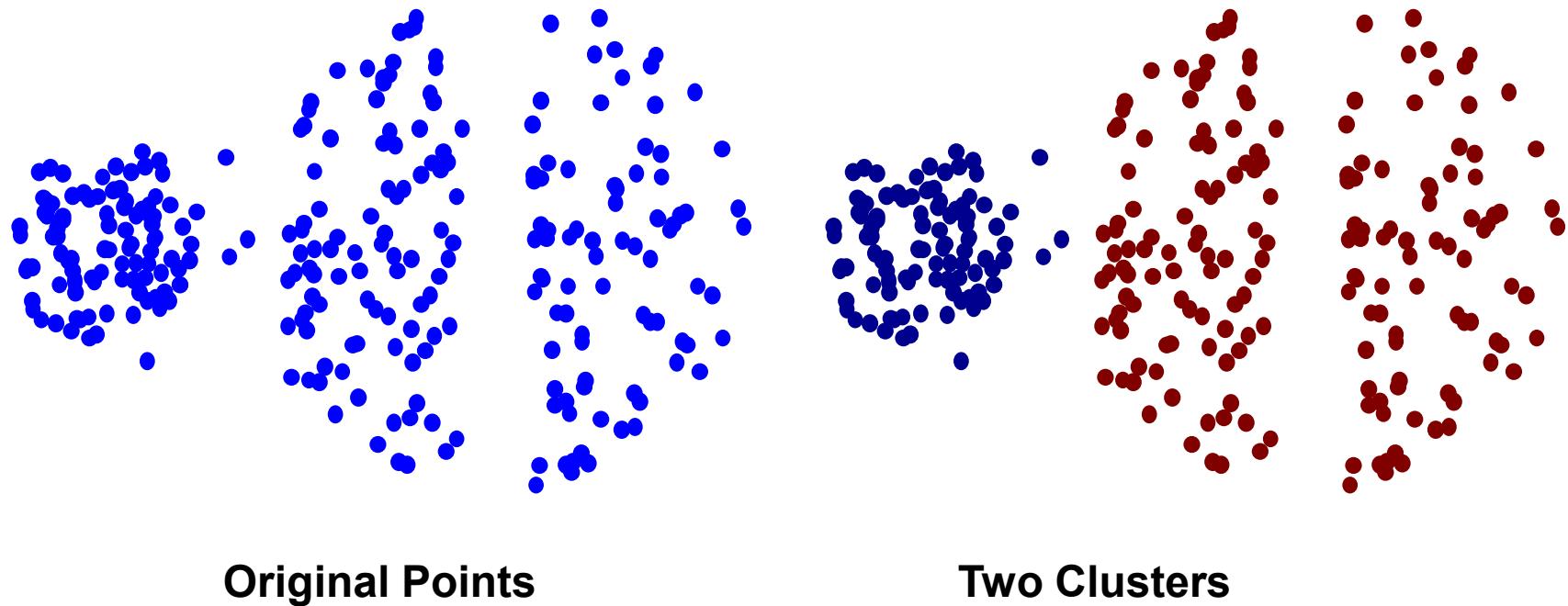
Nested Clusters

	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



Dendrogram

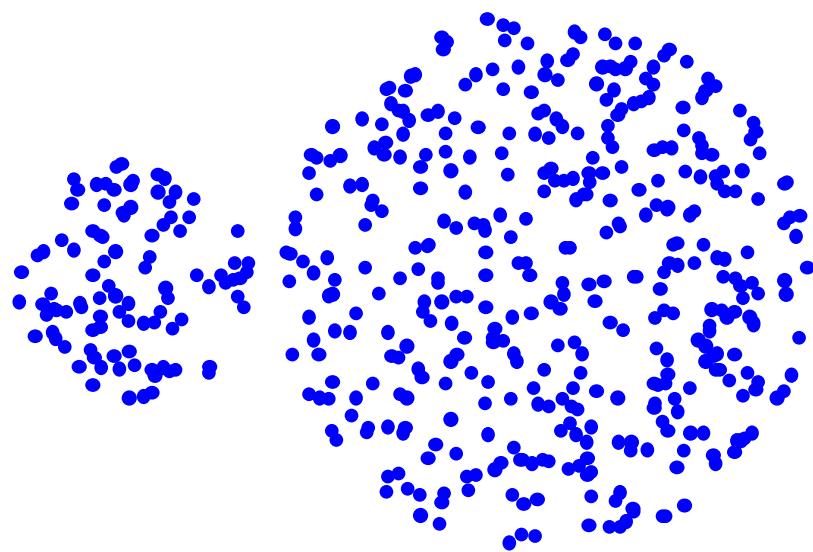
Strength of MAX



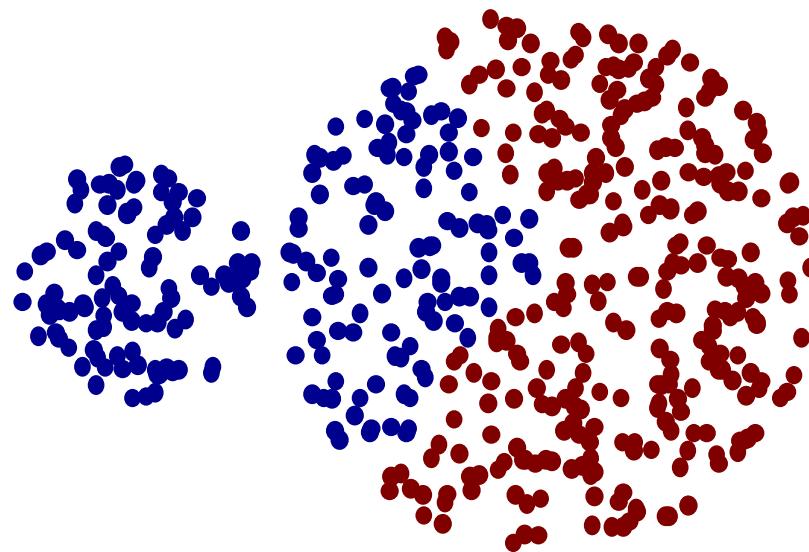
- Less susceptible to noise

قدرت MAX
کمتر مستعد نویز است

Limitations of MAX



Original Points



Two Clusters

- Tends to break large clusters
- Biased towards globular clusters

محدودیت های MAX

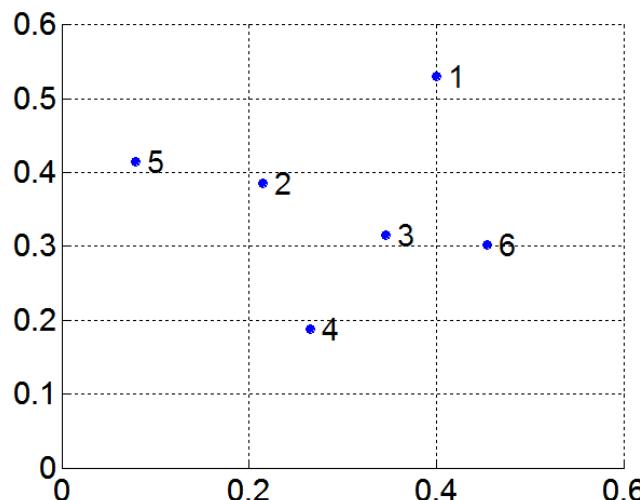
تمایل به شکستن خوشه های بزرگ دارد

- گرایش به خوشه های کروی

Group Average

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

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



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

میانگین بین دو خوش:

فاصله بین 2 تا 3

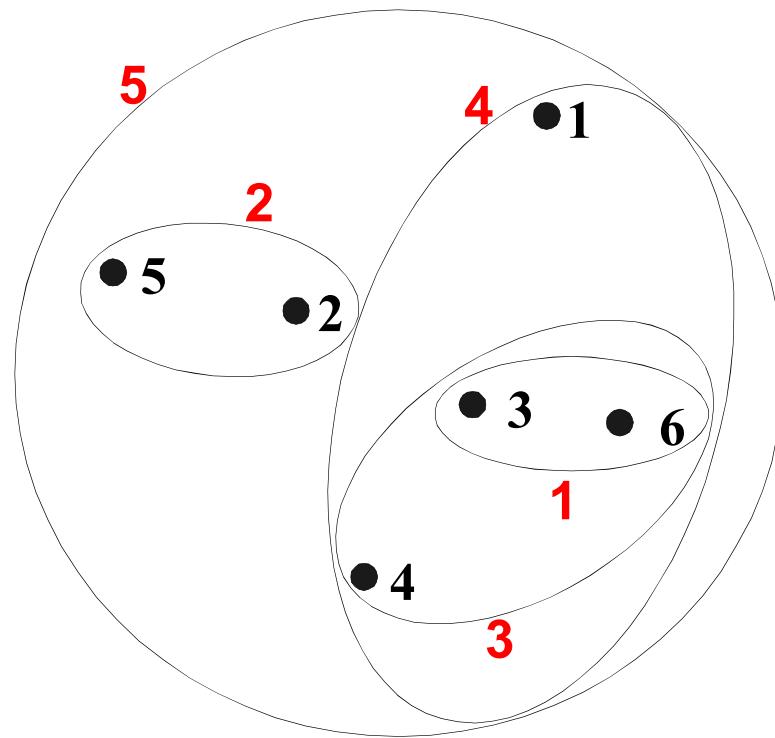
فاصله بین 2 تا 6

فاصله بین 5 تا 3

فاصله بین 5 تا 6

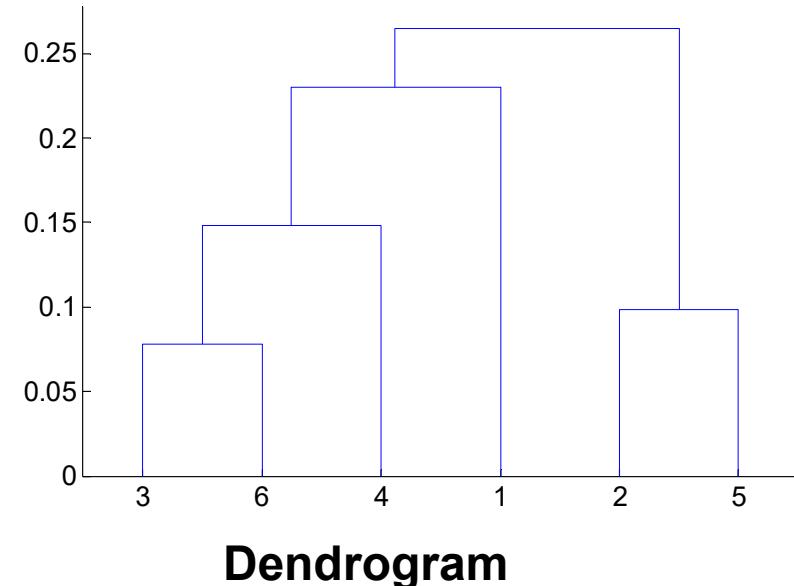
میانگین این 4 تا رو می گیریم و این میشه فاصله بین دو تا خوش

Hierarchical Clustering: Group Average



Nested Clusters

	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

- Compromise between Single and Complete Link
- Strengths
 - Less susceptible to noise
- Limitations
 - Biased towards globular clusters

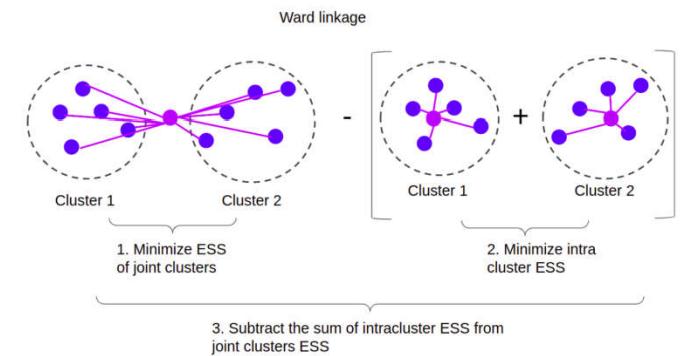
توی این Group Average چون داریم رفتار توده ای نمونه ها رو نگاه میکنیم انگار داریم هم به نمونه هایی که لب مرز توی خوش هستن توجه می کنیم و هم اونایی که اوون عقب هستن انگار یه چیزی بین Compromise between Single and Complete Link است و حساسیت کمتری به نویز داره

با این حال چون داره توده ای عمل می کنه جنس فاصله ها کروی میشه چون کل نمونه ها رو دارن اثر می ذارن و یه جورایی رشد خوش ها به صورت کروی است

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/>

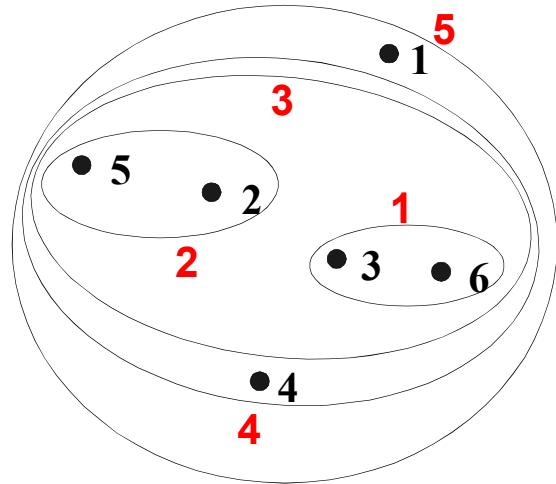
یک روش توی سلسله مراتبی ها Ward است
ما میخوایم خوشه های مختلف رو باهم ترکیب کنیم
تا الان داشتیم به نمونه هاش توجه می کردیم

این روش Ward میاد از مفاهیم خود خوشه بندی کمک می کیره ینی میگه چرا خوشه ها رو داریم
با هم ترکیب می کنیم و چی باعث این کار شده

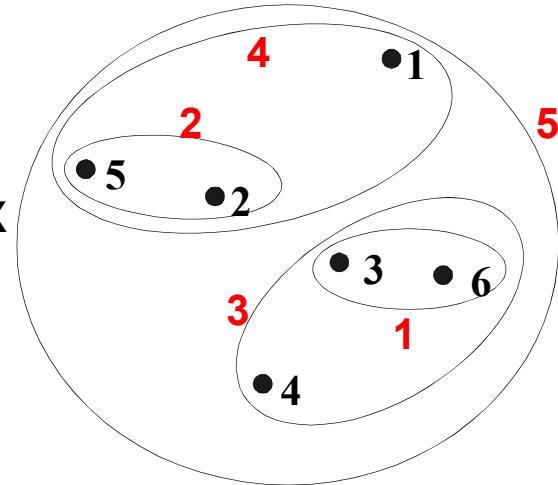
و ادعا این است که ما اگر این دوتا خوشه رو با هم ترکیب کردیم و یک خوشه جدید ساختیم اون خوشه جدید تجمع نقاطشون نسبت به مرکزشون بهتر است ینی برو خوشه هایی رو انتخاب بکن برای ترکیب کردن که اون خوشه ها نتیجه ادغام اون خوشه ها ما رو ببره به یک فضایی که بازم یک تجمع نقاط حول یک مرکزی داشته باشیم --> میاد اینو کمی می کنه ینی میگه وقتی می خواهی بینی خوشه های مطلوب برای ادغام کدوم هستن بیا این متريک رو حساب بکن ینی به ازای هر کدوم خوشه ای که میخواستی مرج بکنی اینو حساب بکن --> فاصله مرکز اون خوشه از بقیه نمونه ها که از اجتماع اینا حساب شده چقدر شده و ایا واقعاً بهتر است و اونی رو انتخاب بکن که فاصله از مرکزیتش کمتر است ینی مربع خطاهاش بعد از اینکه ما این ها رو مرج کردیم بهتر شد

ویژگی:
حساسیتش نسبت به نویز کمتر است
اینجا هم اون الگوها به صورت کروی رخ میدن

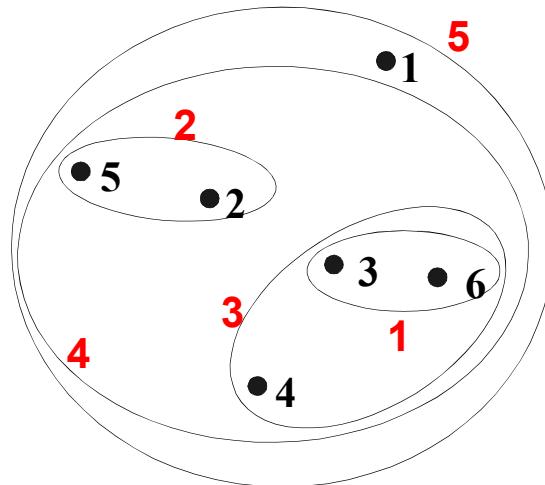
Hierarchical Clustering: Comparison



MIN

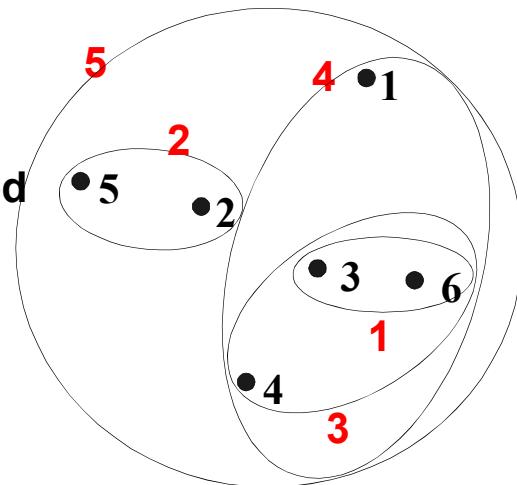


MAX



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

پیچیدگی زمانی و فضایی این روش ها:

از لحاظ پیچیدگی فضا ما باید این proximity matrix رو ذخیره بکنیم و این توی بدترین حالت به تعداد نمونه ها میشه ینی n به توان 2

پیچیدگی زمانی: یکی محاسبه کردن خود این ماتریس است که میشه n به توان 2 یک n برای خود کل درخت است ینی چقدر طول می کشه که درخت را تولید بکنیم

بدترین حالتی که این روش های خوشه بندی سلسله مراتبی خواهند داشت حالتی است که در هر گام فقط یک نمونه به یک خوشه اضافه بشه و می خوایم ببینم کدام یکی از این دو تا نم^{؟؟}/نم^{؟؟} نفهمیدم

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

-
توی این روش وقتی یک نمونه وارد خوشه شد دیگه ما برنمی گردیم و بگیم کاش اینو وارد این خوشه نمی کردیم
برای حساب کردن فاصله ها یک تابع کلی دیگه نداریم و باید بیایم نمونه به نمونه فاصله اینارو از هم داشته باشیم و بحث اجرا سازی خیلی کند میکنه
اینجا **objective function** به خصوصی نداریم
حساسیت به نویز و پیچیدگی پیاده سازی جز مسائلی است که خوشه بندی سلسله مراتبی ها دارن

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 **MinPts** (at least a specified number of points)within **Eps** area
 - ◆ 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

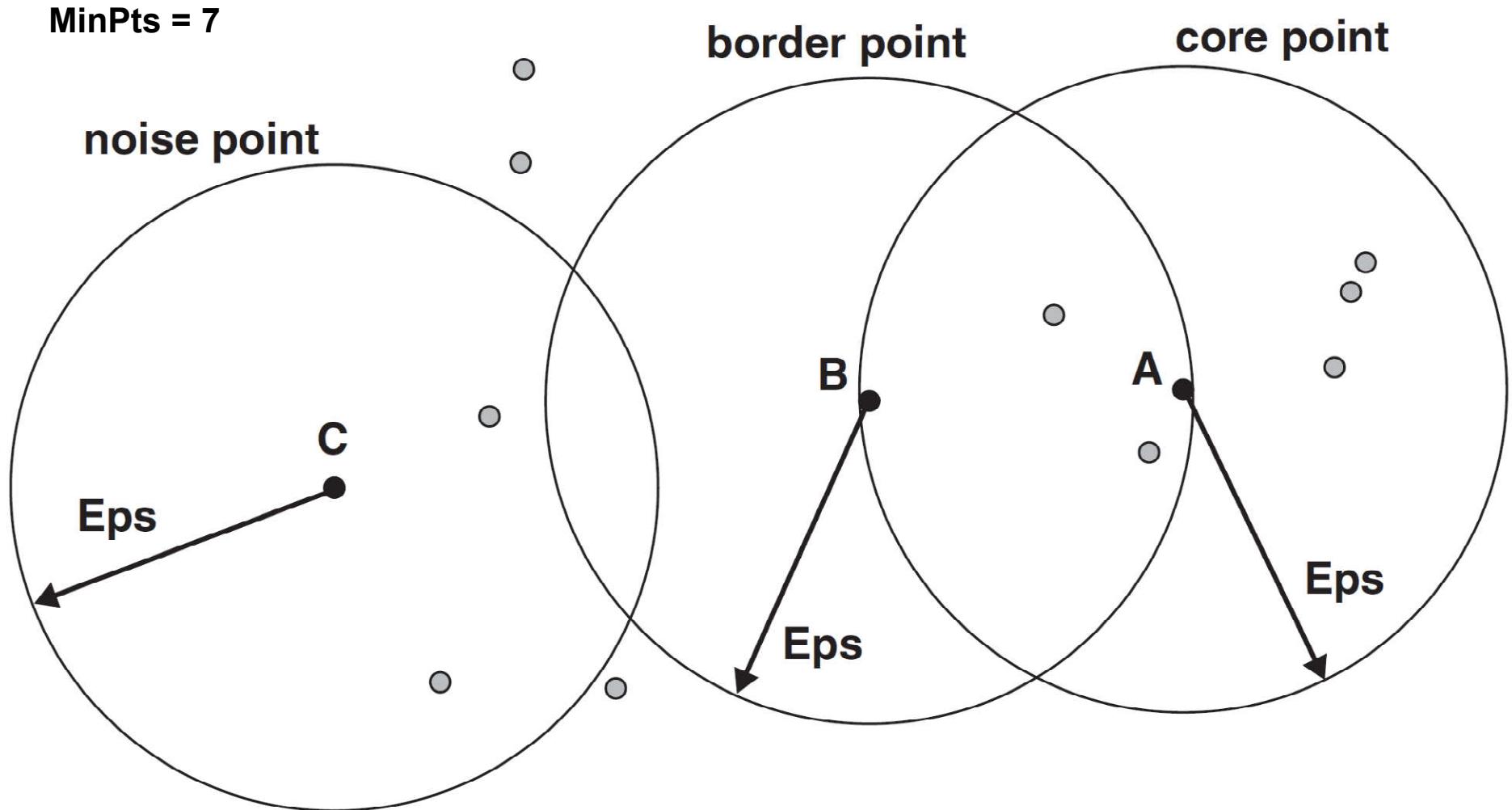
:DBSCAN

در این روش سعی میشے به چگالی نمونه یا تراکم نمونه ها توی یک فضا دقت میکنه و وقتی با یک توده ای از نقاط رو به رو میشن اون توده رو تبدیل به یک خوشه بکنن

:MinPts

Eps

DBSCAN: Core, Border, and Noise Points



توی DBSCAN میان نقاط رو به سه دسته تقسیم می کنن --> نقاطی که توی داده ها داریم سه تا
برچسب پیدا می کنن یا نقاط نویزن یا border ینی توی همسایگی یک ناحیه چگال هستن یا
ینی نقاطی که اطرافشون تعداد زیادی نقطه وجود داره و توی محدوده پر چگال هستن

فضای پر تراکم رو اینطوری تعریف کردیم:

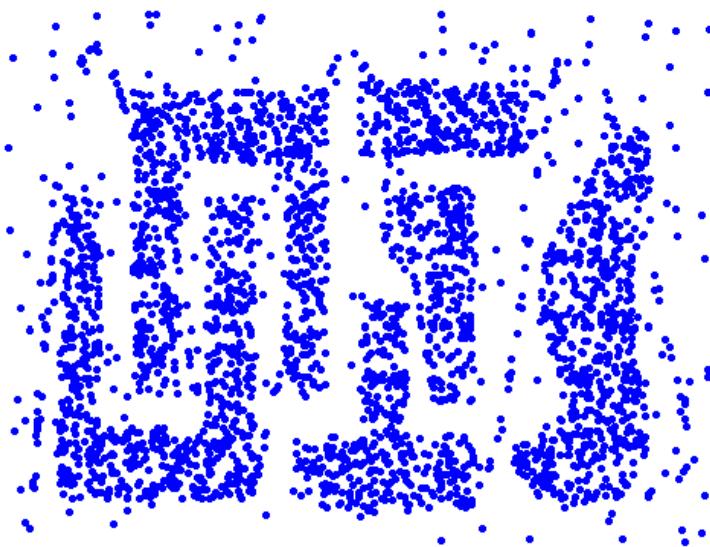
فضای پر تراکم فضایی است که توی یک شعاعی از اپسیلون ما 7 تا نقطه بینیم --> 7 تا نقطه رو با
مشخص میکنیم و شعاع اون فضا رو با eps minpts هر نقطه ای می ره توی اون سه دسته صفحه قبل:

نقطه a : نقطه core است چون توی همسایگی a رو نگاه کنیم می بینیم حداقل 7 تا نقطه وجود
داره و چون نقطه a یک نقطه ای بود که توی همسایگیش اون حداقلی که برای تعریف چگال داشتیم
ما میگیم این نقطه a یک نقطه core است

نقطه b: توی همسایگیش 3 تا هست و این 3 اون حداقل برای این فضا نیست ولی توی همسایگی
این نقطه حداقل یک نقطه وجود داره که اون نقطه میشه نقطه core

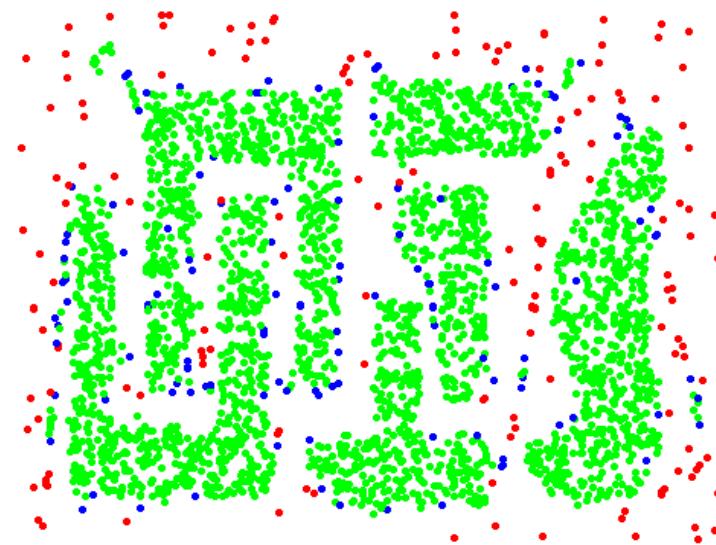
نقطه c: یک همسایگی حول اون نقطه ایجاد میکنیم و می بینیم چندتا نمونه توی اون همسایگی
هست حداقل سه تا هست حالا این نقطه border نیست چون هیچی نقطه ای از این 3 تا نقطه جز
نواحی core نیست

DBSCAN: Core, Border and Noise Points



Original Points

Eps = 10, MinPts = 4



Point types: **core**,
border and **noise**

حالا بر اساس این سه دسته میاد خوشه ها رو ایجاد میکنه این روش
چطوری داده ها رو با این داستان خوشه بندی بکنیم؟ فیلم رو ببین که توی گوشی داری

پیمایشش: نقاط یکی پس از دیگری تکلیفشون مشخص میشه --> از یک نقطه ای شروع میکنه و
انالیز میکنه که ایا این نقطه، نقطه core است یا نیست و به همین صورت می ره جلو و بهشون
برچسب میده

DBSCAN Algorithm

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

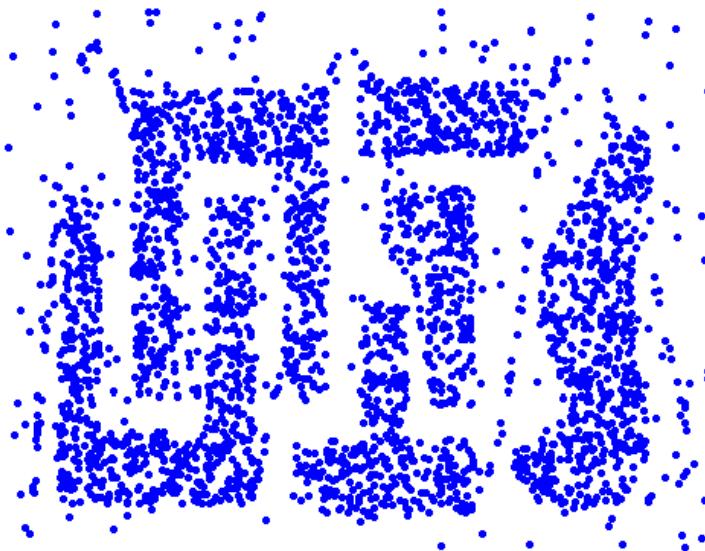
- 1: Label all points as core, border, or noise points.
- 2: Eliminate noise points.
- 3: Put an edge between all core points within a distance Eps of each other.
- 4: Make each group of connected core points into a separate cluster.
- 5: Assign each border point to one of the clusters of its associated core points

-

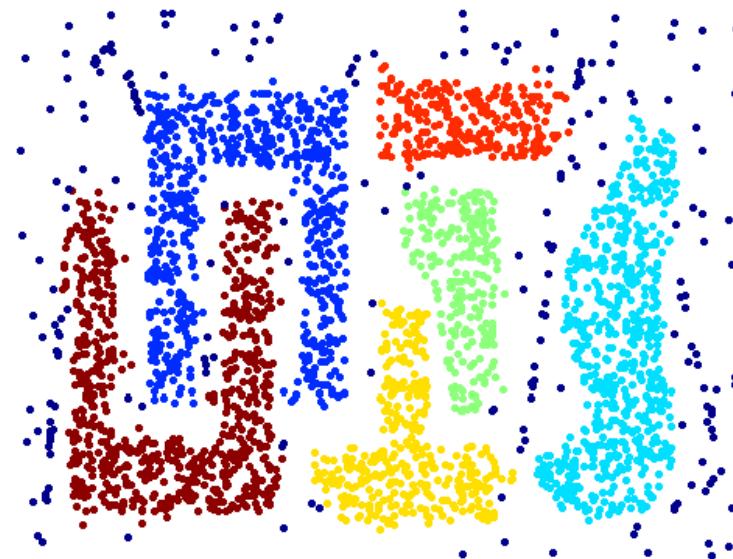
اول سه نقطه رو مشخص میکنه فقط برچسب نمی زنه
بعد نقطه های نویز رو حذف می کنه چون قرار نیست اینارو به عنوان یک خوشه گزارش بکنیم
حالا میخوایم خوشه ها رو نسبت بدیم --> اینجا میاد روی نقطه های core حرکت میکنه و یک خوشه به اینا نسبت میده

این روش نسبت به نویز مقاوم است
اینجا دیگه ما تعداد خوشه تعریف نکردیم و خودش تشخیص داد چه تعداد خوشه توی دیتا است

When DBSCAN Works Well



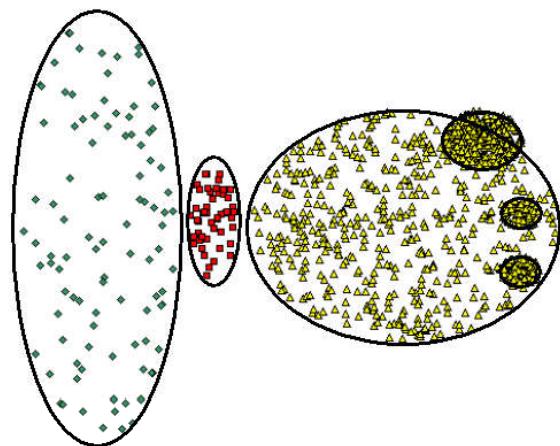
Original Points



Clusters (dark blue points indicate noise)

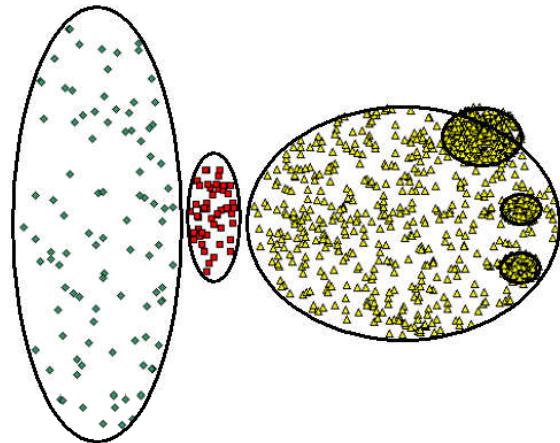
- Can handle clusters of different shapes and sizes
- Resistant to noise

When DBSCAN Does NOT Work Well



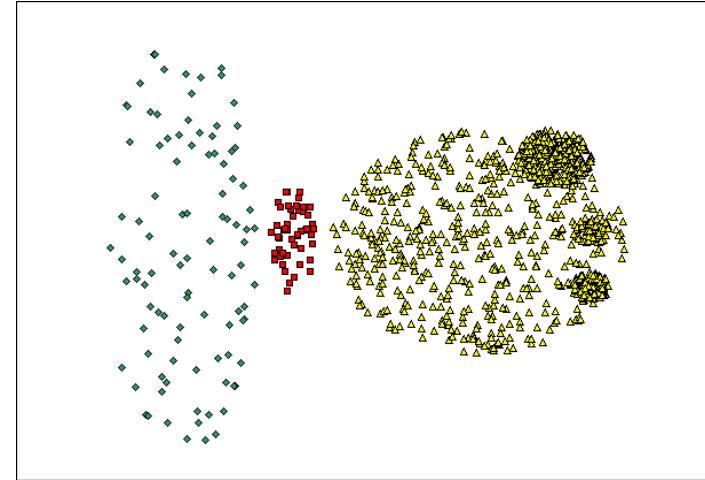
Original Points

When DBSCAN Does NOT Work Well

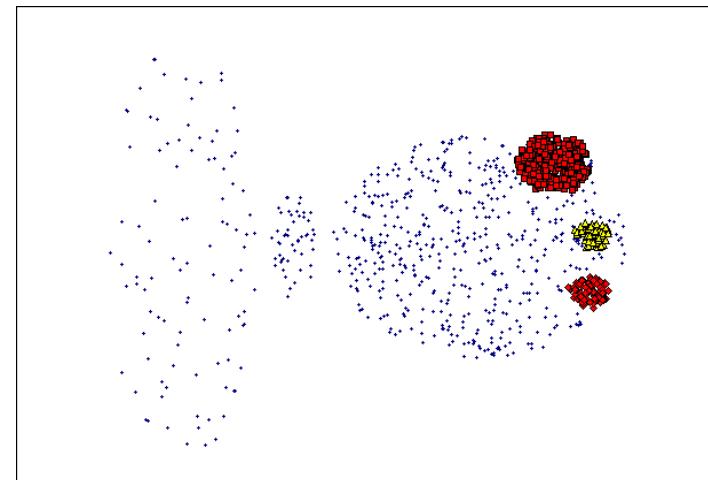


Original Points

- Varying densities
- High-dimensional data



(MinPts=4, Eps=9.92).



(MinPts=4, Eps=9.75)

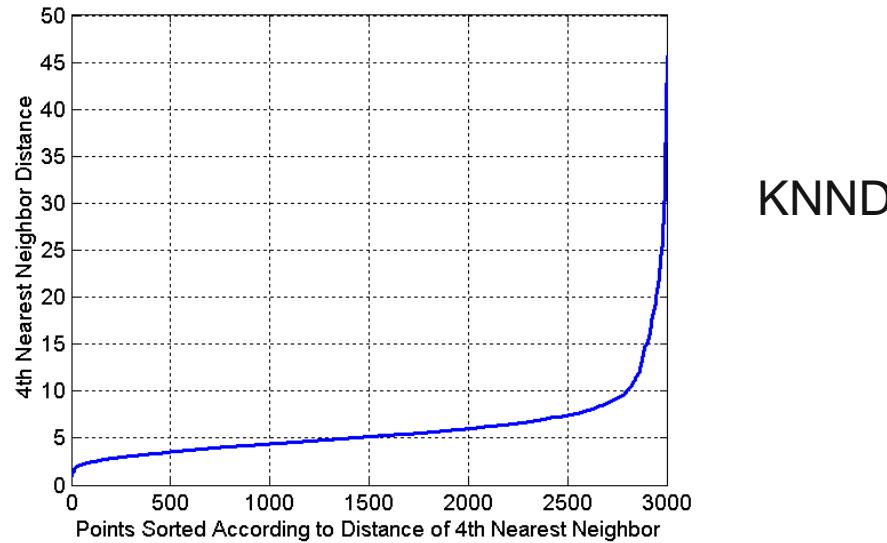
چه موقع هایی DBSCAN خوب نیست؟
چندین خوشه داریم با سطح چگالی های مختلف
اینجا $\text{minpts}=4$ گذاشته و eps رو تغییر داده و مقدار های مختلفی برآش گرفته
نتیجه فرق کرده چرا؟

وقتی ابعاد رو زیاد بکنیم احتمال اینکه یه تعداد داده مشخص توی اون ابعاد ببینیم کم میشه
DBSCAN توی ابعاد بالا چون داریم روی بحث شاعع همسایگی تعریف میکنیم به مشکل
برمیخوره

چگالی های متفاوت
• داده های با ابعاد بالا

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



-
این تجربی است تقریباً ینی یه چیزی که با تجربه به دست اومده
 eps , minpts رو چجوری به دست بیاریم؟

این نمودار یه دیدی از چگالی نمونه ها بهمون میده --> محور X نمونه ها است

CLUSTER EVALUATION

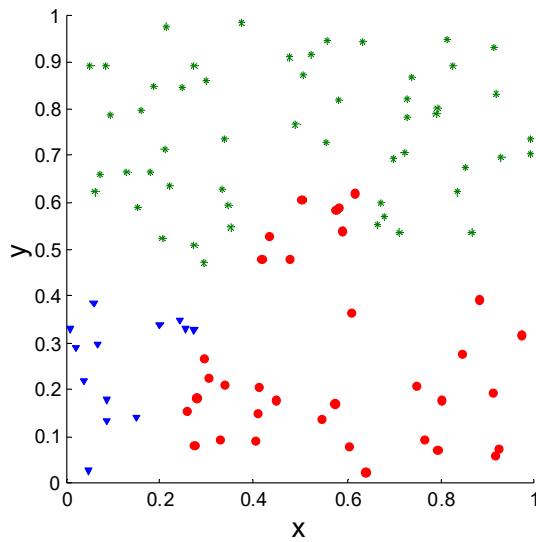
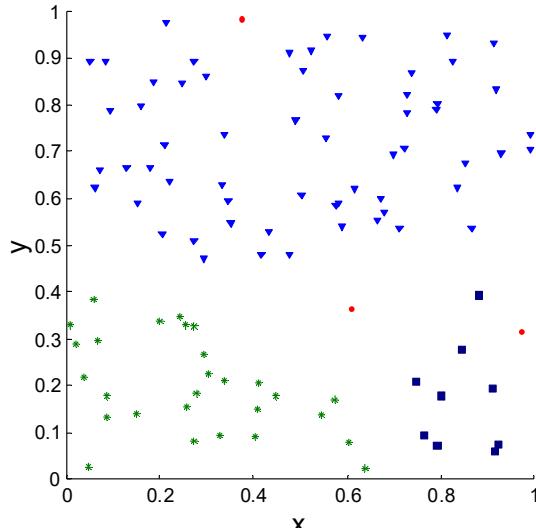
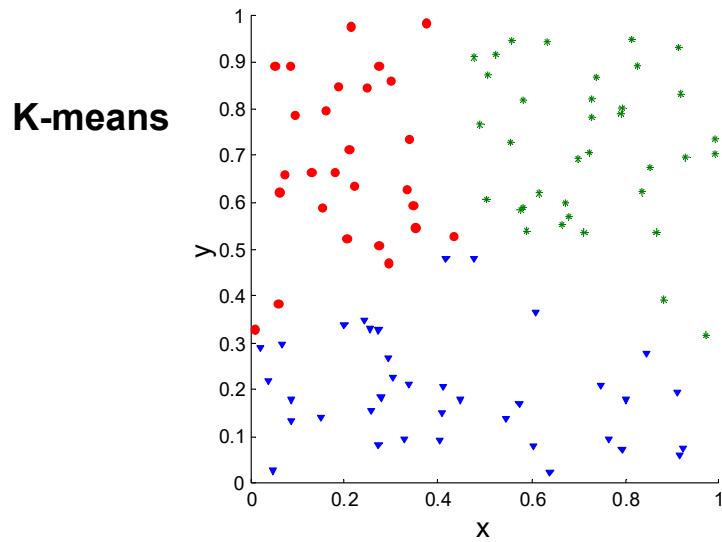
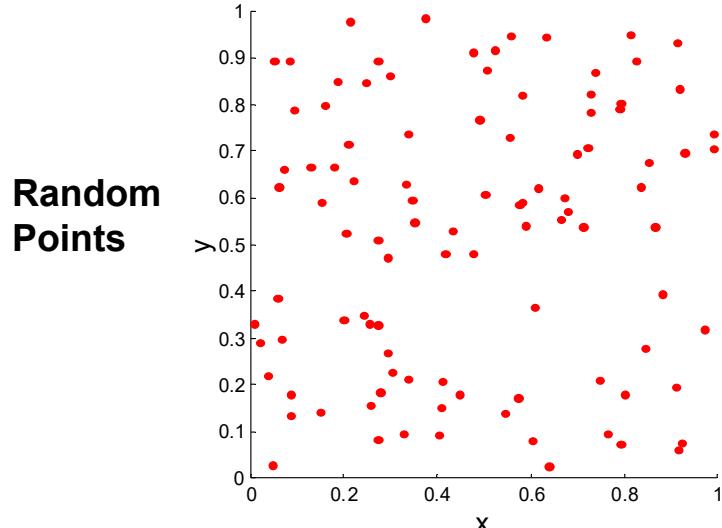
ارزیابی خوش:

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

برای پیدا کردن اینکه یک خوشه خوب است یا نه نیاز داریم با یک متريکی به صورت کمی بگیم
این روش خوشه بندی چقدر خوبه
و همينطور یک معیاري باشه که به نمونه های نويز توجه نکنه که کار رو خراب نکنه
يکسری متريک هايی وجود داره که ميان كيفيت توزيع اين داده ها رو می سنجن

Clusters found in Random Data



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

-
برای اندازه گیری کیفیت خوش بندی دو تا رویکرد وجود داره:
رویکرد Supervised --> یا میای توی بحث تئوری اطلاعات یا برچسب داری
آنتروپی
رویکرد Unsupervised --> اینجا اینو میگیم

Unsupervised Measures: Cohesion and Separation

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

این دو تا متریک اطلاعاتی به ما میدن از نتایج خوشه بندی
خوشه هایی که تو گزارش کردی چقدر بهم شبیه هستن ینی چقدر هم شکل و هم راستا هستن :
Cohesion

فاصله بین خوشه ها یا **Separation**: ایا خوشه های ما به اندازه کافی از هم دور هستن یا نه

X میشه نمونه

mi: میانگین هر خوشه است که فاصله نمونه های هر خوشه رو از مرکزشون می سنجیم
بهمون میگه این روش خوشه بندی نسبت به مرکزها ینی نمونه های داخل هر خوشه نسبت به
مرکزشون چقدر فاصله دارن
ولی

این کاری به نمونه ها نداره بلکه فاصله میانگین خوشه ها رو از هم می سنجه که
هر خوشه چقدر دور افتاده است نسبت به خوشه دیگه
به صورت وزن دار در نظر میگیره
ci اندازه خوشه است که داریم می سنجیم

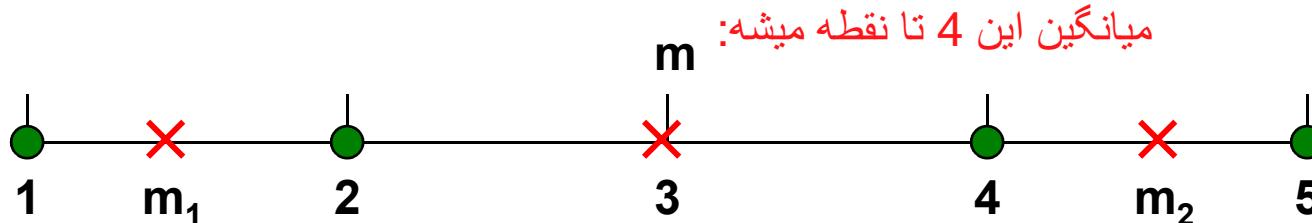
Unsupervised Measures: Cohesion and Separation

- Example: SSE

- $SSB + SSE = \text{constant}$

جمع این دو تا همیشه یک مقدار ثابتی میشے

میانگین این ۴ تا نقطه میشے:



K=1 cluster: $SSE = (1 - 3)^2 + (2 - 3)^2 + (4 - 3)^2 + (5 - 3)^2 = 10$

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

$$Total = 10 + 0 = 10$$

K=2 clusters: $SSE = (1 - 1.5)^2 + (2 - 1.5)^2 + (4 - 4.5)^2 + (5 - 4.5)^2 = 1$

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

$$Total = 1 + 9 = 10$$

اگر این 4 تا نقطه ما یک خوشه باشن می خوایم SSE رو حساب بکنیم:

میانگین کل خوشه ها شده 3
میانگین این خوشه هم شده 3
4 هم تعداد نمونه های توی یک خوشه

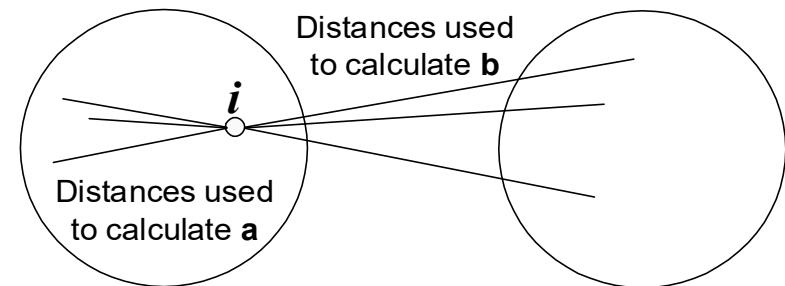
SSE کم بکنیم و SSB زیاد بکنیم --> اینو می خوایم

Unsupervised Measures: Silhouette Coefficient

- Silhouette coefficient combines ideas of both cohesion and separation, but for individual points, as well as clusters and clusterings
- For an individual point, i
 - Calculate a = average distance of i to the points in its cluster
 - Calculate b = min (average distance of i to points in another cluster)
 - The silhouette coefficient for a point is then given by

$$s = (b - a) / \max(a, b)$$

- Value can vary between -1 and 1
- Typically ranges between 0 and 1.
- The closer to 1 the better.



- Can calculate the average silhouette coefficient for a cluster or a clustering

یک الگوریتم داریم به نام Silhouette :

a: فاصله این نمونه از بقیه نمونه ها که ما دوست داریم اینو هی زیاد بکنیم --> اینو زیاد بکنیم
b: فاصله تک تک نمونه هاست --> فاصله هر نمونه از نمونه های هر خوشه که ما دوست داریم
اینو هی کمتر بکنیم --> اینو کم بکنیم

یک نقطه رو بر میداره و میگه این نقطه مال کدوم خوشش و بعد بیا فاصله این نقطه رو از همه هم خوشه ای هاش حساب بکن و میانگینشو بگیر و یک عدد میده این میشه a

b: به ازای هر خوشه ای یک میانگینی می گیریم و از خوشه بعدی هم یه میانگین باز میگیریم و ...
این داره میگه بیا کمینشو بذار اینجا ینی میانگین میانگین نکن و بیا کمینشو بذار ینی نزدیکترین خوشه رو بذار اینجا

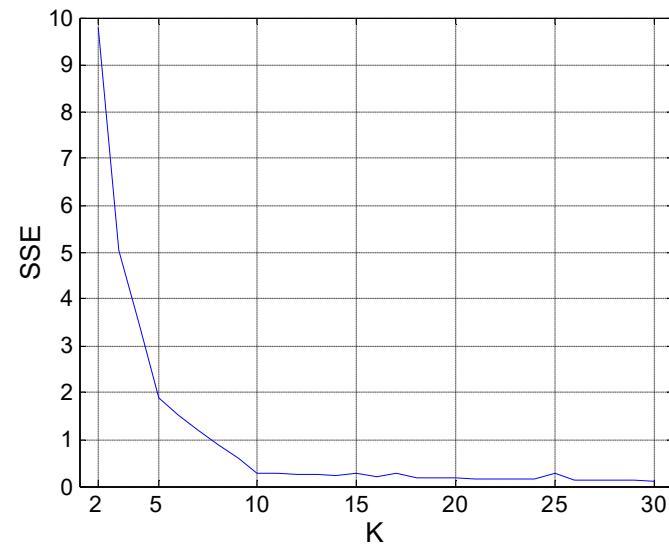
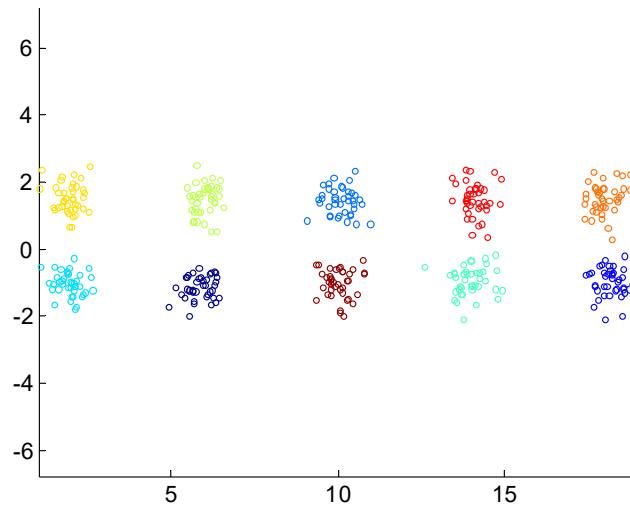
و طبق فرمول توی صفحه این میشه عدد Silhouette این نقطه و برای نقطه بعد هم همین کارو
بکن و به همین صورت تا همه نقاط حساب بشه و این یک عددی بهمون میده

این Silhouette یک رنجی داره:

ماکزیممش یک است و مینیمم -1 --> هر چی b بزرگتر بشه و a کوچیکتر بشه ما اینو میخوایم
و هرچی Silhouette بهتر باشه ما میگیم این روش خوشه بندی بهتر بود نسبت به یک روش دیگه

Determining the Correct Number of Clusters

- SSE is good for comparing two clusterings or two clusters
- SSE can also be used to estimate the number of clusters



تعداد خوشه ها رو چندتا در نظر بگیریم؟

بیا از این SSE کمک بگیر ینی فاصله درون کلاسی

و یک نمودار ایجاد میکنیم به فرم شکل زیر : یک بار تعداد خوشه ها رو میداریم دو و بعد

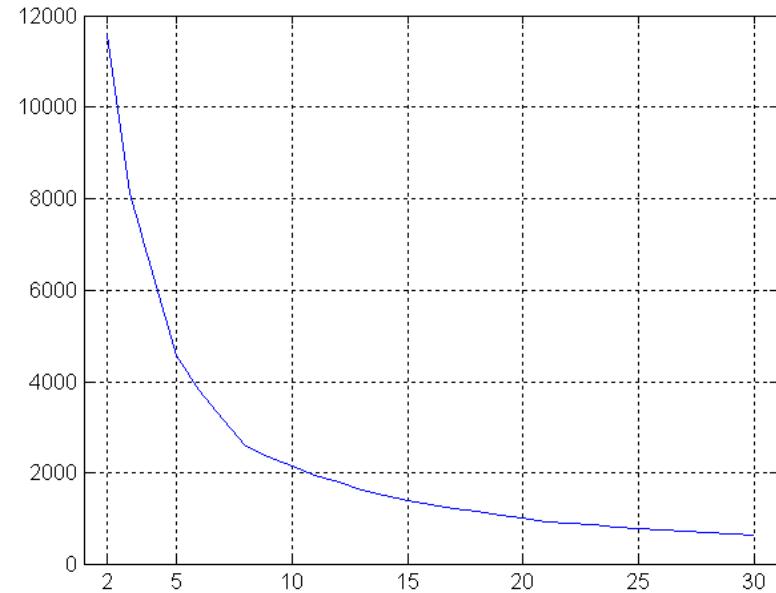
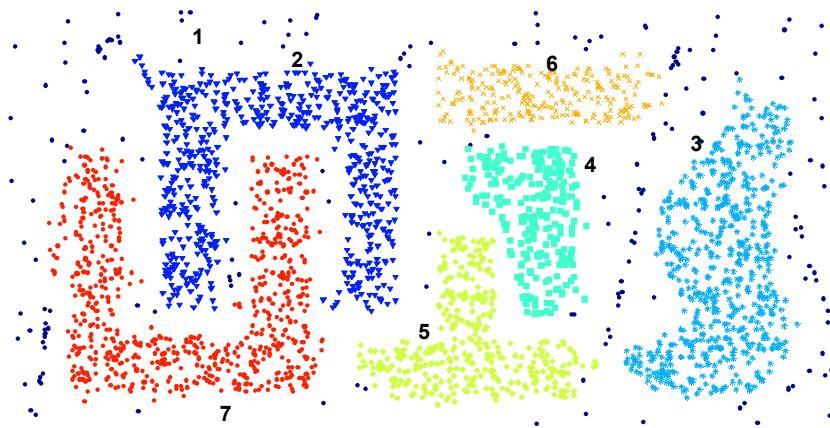
K-means رو اجرا میکنیم و بعد SSE رو اندازه میگیریم و بعد یکبار می ذاریم 5 و بعد

K-means اجرا میکنیم و بعد SSE رو حساب میکنیم و به همین صورت حالا براساس این می

خوایم بدونیم که تعداد خوشه ها رو چندتا بذاریم:

Determining the Correct Number of Clusters

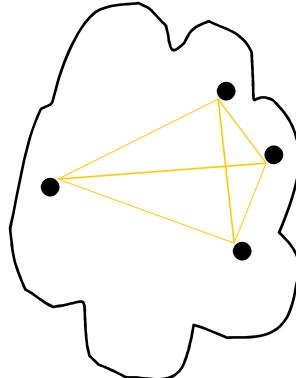
- SSE curve for a more complicated data set



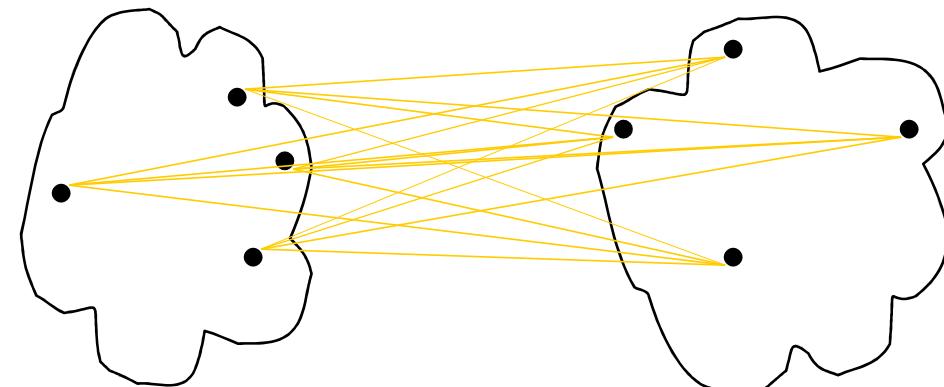
SSE of clusters found using K-means

Unsupervised Measures: Cohesion and Separation

- A proximity graph-based approach can also be used for cohesion and separation.
 - Cluster cohesion is the sum of the weight of all links within a cluster.
 - Cluster separation is the sum of the weights between nodes in the cluster and nodes outside the cluster.



cohesion



separation

Measuring Cluster Validity Via Correlation

- Two matrices
 - Proximity Matrix
 - Ideal Similarity Matrix
 - ◆ One row and one column for each data point
 - ◆ An entry is 1 if the associated pair of points belong to the same cluster
 - ◆ An entry is 0 if the associated pair of points belongs to different clusters
- Compute the correlation between the two matrices
 - Since the matrices are symmetric, only the correlation between $n(n-1) / 2$ entries needs to be calculated.
- High magnitude of correlation indicates that points that belong to the same cluster are close to each other.
 - Correlation may be positive or negative depending on whether the similarity matrix is a similarity or dissimilarity matrix
- Not a good measure for some density or contiguity based clusters.
