

# UVA CS 6316

## – Fall 2015 Graduate: Machine Learning

### Lecture 21: Unsupervised Clustering (II)

Dr. Yanjun Qi

University of Virginia

Department of  
Computer Science

11/11/15

1

## Where are we ? → major sections of this course

- Regression (supervised)
- Classification (supervised)
  - Feature selection
- Unsupervised models
  - Dimension Reduction (PCA)
- Clustering (K-means, GMM/EM, Hierarchical )
- Learning theory
- Graphical models
  - (BN and HMM slides shared)

11/11/15

2

	$X_1$	$X_2$	$X_3$
$S_1$			
$S_2$			
$S_3$			
$S_4$			
$S_5$			
$S_6$			

## An unlabeled Dataset X

a data matrix of  $n$  observations on  $p$  variables  $x_1, x_2, \dots, x_p$

**Unsupervised learning** = learning from raw (unlabeled, unannotated, etc) data, as opposed to supervised data where a classification label of examples is given

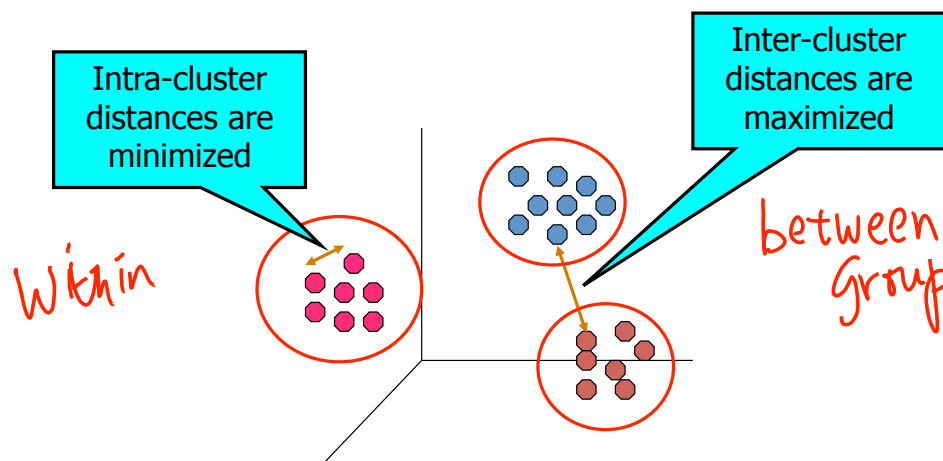
- **Data/points/instances/examples/samples/records:** [ rows ]
- **Features/attributes/dimensions/independent variables/covariates/predictors/regressors:** [ columns ]

11/11/15

3

## What is clustering?

- Find groups (clusters) of data points such that data points in a group will be similar (or related) to one another and different from (or unrelated to) the data points in other groups



11/11/15

4

# Application (I): Search Result Clustering

Partition

Google jaguar Dr. Yanjun Qi / UVA CS 6316 / f15

Web Images News Videos Shopping More Search tools

About 37,200,000 results (0.43 seconds)

**JaguarUSA.com - Jaguar® Convertible Car**  
Ad [www.jaguarusa.com/](http://www.jaguarusa.com/)  
Real Comfort Comes From Control. Schedule Your Test Drive Today.  
Jaguar USA has 1,261,482 followers on Google+

**Build & Price**  
Design A Jaguar Car to Your Driving Style and Personal Tastes.

**Locate A Retailer**  
Find Your New Dream Car At Your Closest Jaguar Retailer Today.

**Naughty Car. Nice Price.**  
Unwrap A Jaguar® Vehicle During Our Winter Sales Event On November 3rd.

**Request A Quote**  
Get A Quote On Your Favorite Model From Your Local Jaguar Retailer.

**Jaguar: Luxury Cars & Sports Cars | Jaguar USA**  
[www.jaguarusa.com/](http://www.jaguarusa.com/) - Jaguar Cars -  
The official home of JAGUAR USA. Our luxury cars feature innovative designs along with legendary performance to deliver one of the top sports cars in the ...  
Models - F-Type - XF - XJ

**Jaguar - Wikipedia, the free encyclopedia**  
[en.wikipedia.org/wiki/Jaguar](http://en.wikipedia.org/wiki/Jaguar) - Wikipedia -  
The jaguar Panthera onca, is a big cat, a feline in the Panthera genus, and is the only Panthera species found in the Americas. The jaguar is the third-largest ...  
Jaguar Cars - Jaguar (disambiguation) - Tapir - List of solitary animals

**Jaguar Cars - Wikipedia, the free encyclopedia**  
[en.wikipedia.org/wiki/Jaguar\\_Cars](http://en.wikipedia.org/wiki/Jaguar_Cars) - Wikipedia -  
Jaguar Cars is a brand of Jaguar Land Rover, a British multinational car manufacturer headquartered in Whitley, Coventry, England, owned by Tata Motors since ...

**Images for jaguar** Report images

More images for jaguar

Brown's Jaquar

11/11/15

5

# Application (II): Navigation

Entertainment in the Yahoo! Directory - Mozilla Firefox

File Edit View History Bookmarks Tools Help

<http://dir.yahoo.com/Entertainment/>

Getting Started Latest Headlines

Yahoo! My Yahoo! Mail Welcome, Guest (Sign In) Directory Home Help

**YAHOO! DIRECTORY** Search:  the Web |  the Directory |  this category

**Entertainment** [Email this page](#) [Suggest a Site](#) [Advanced Search](#)

Directory > Entertainment

**SPONSOR RESULTS**

**Value City Furniture**  
[www.vcf.com](http://www.vcf.com) Quality Home Entertainment Packages Browse Today and Find a Store.

**Entertainment Center Furniture**  
Save 30-60% On A Variety Of Furniture For Any Room Thru 11/13.  
JCPenney.com

**Studiotech Official Site**  
StudioTech Entertainment Furniture. Factory Direct...  
[www.StudioTech.com](http://www.StudioTech.com)

**Bush Entertainment Furniture**  
Save up to 50% factory-direct...  
[www.bushfurniturecolle...](http://www.bushfurniturecolle...)

**CATEGORIES** [Show All](#)

**Top Categories**

- [Music](#) (76772) **NEW**
- [Actors](#) (19211) **NEW**
- [Movies and Film](#) (40031) **NEW**
- [Television Shows](#) (17085) **NEW**
- [Humor](#) (8927)
- [Comics and Animation](#) (6778) **NEW**

**Additional Categories**

- [Amusement and Theme Parks](#) (449)
- [Awards](#) (696)
- [Blogs@](#)
- [Books and Literature@](#)
- [Chats and Forums](#) (47)
- [Comedy](#) (1730)
- [Consumer Electronics](#) (1355) **NEW**
- [Magic](#) (353)
- [News and Media](#) (443)
- [Organizations](#) (33)
- [Performing Arts@](#)
- [Radio@](#)
- [Randomized Things](#) (67)
- [Reviews](#) (32)

Hierarchy

11/11

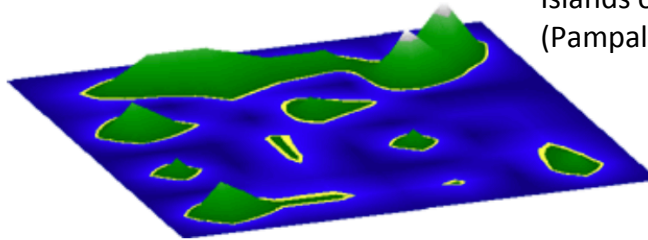
6

# Application (III): Visualization

## Islands of Music

Analysis, Organization, and Visualization of Music Archives

Islands of music  
(Pampalk et al., KDD' 03)

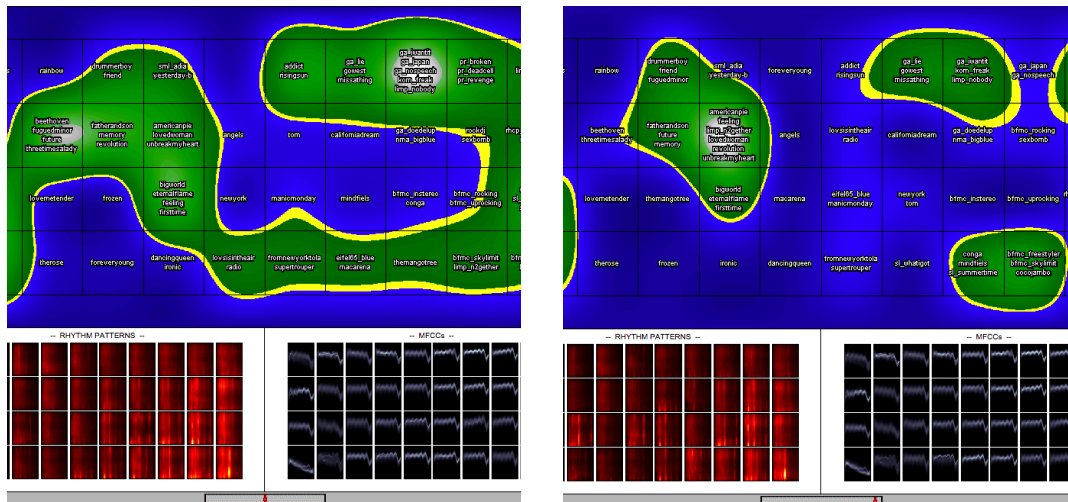


**piece of music:** member of a *music collection* and inhabitant of *islands of music*. Groups of similar pieces of music (also known as *genres*) like to gather around large mountains or small hills depending on the size of the group. Groups which are similar to each other like to live close together. Individuals which are not members of specific groups usually live near the beach and some very individualistic pieces might be found swimming in deep water.

**islands of music:** serve as graphical *user interface* to a music collection and are intended to help the user explore vast amounts of music in an efficient way. Islands of music are generated automatically based on *psychoacoustics models* and *self-organizing maps*. 7

# SOM Application (III): Visualization (feature changes → clusters' change)

Islands of music (Pampalk et al., KDD' 03, [http://www.ofai.at/~elias.pampalk/kdd03/Visualizing Changes in the Structure of Data for Exploratory Feature Selection](http://www.ofai.at/~elias.pampalk/kdd03/Visualizing_Changes_in_the_Structure_of_Data_for_Exploratory_Feature_Selection))



## Roadmap: clustering

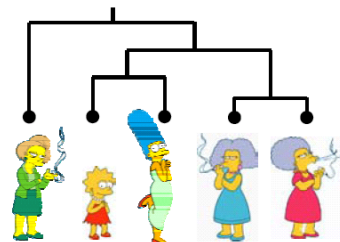
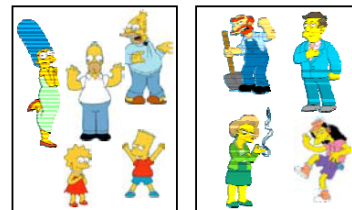
- Definition of "groupness"
- Definition of "similarity/distance"
- Representation for objects
- How many clusters?
- Clustering Algorithms
  - ➔ ▪ Partitional algorithms
    - Hierarchical algorithms
  - Formal foundation and convergence

11/11/15

9

## Clustering Algorithms

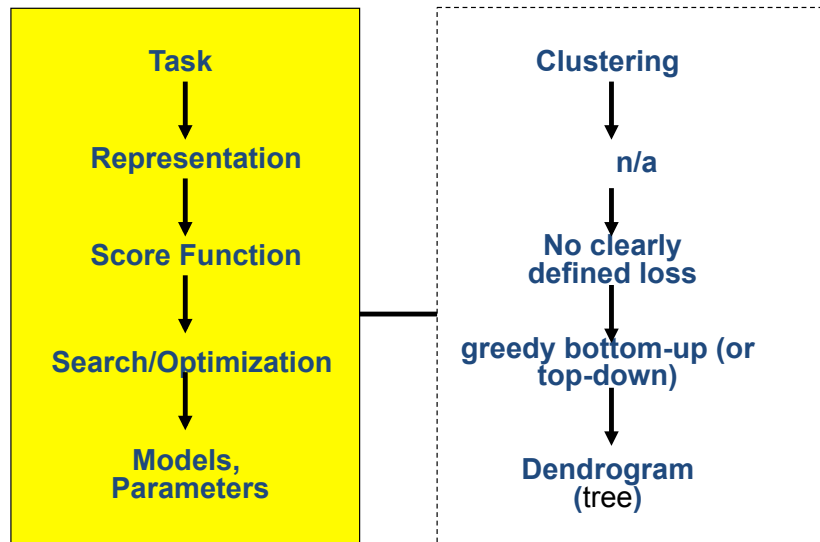
- Partitional algorithms
  - Usually start with a random (partial) partitioning
  - Refine it iteratively
    - K means clustering
    - Mixture-Model based clustering
- Hierarchical algorithms
  - Bottom-up, agglomerative
  - Top-down, divisive



11/11/15

10

## (1) Hierarchical Clustering

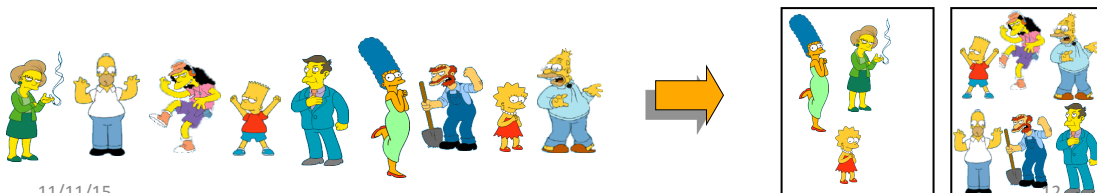


11/11/15

11

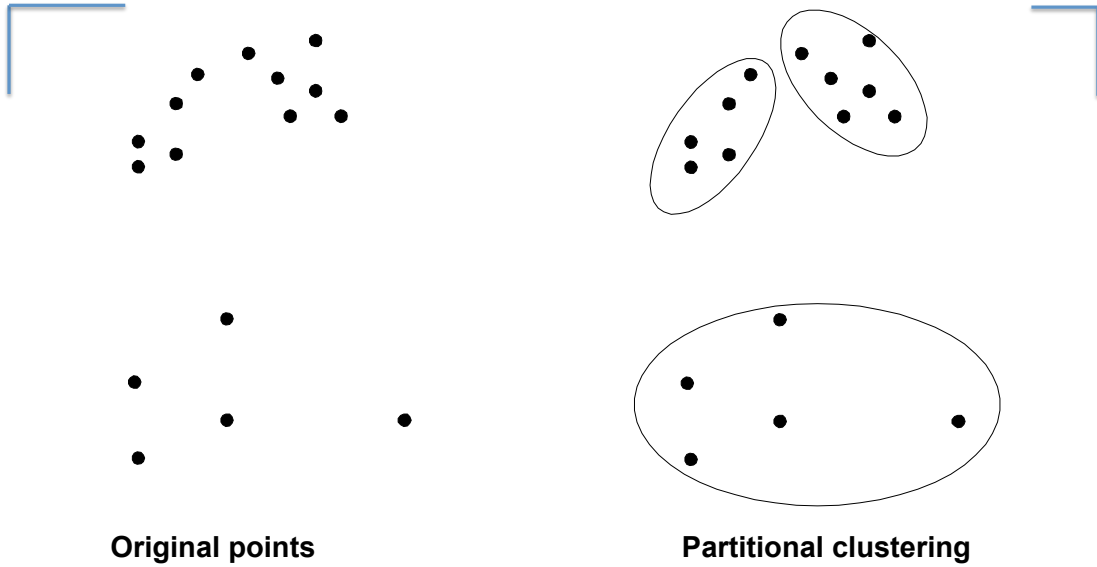
## (2) Partitional Clustering

- Nonhierarchical
- Construct a partition of  $n$  objects into a set of  $K$  clusters
- User has to specify the desired number of clusters  $K$ .

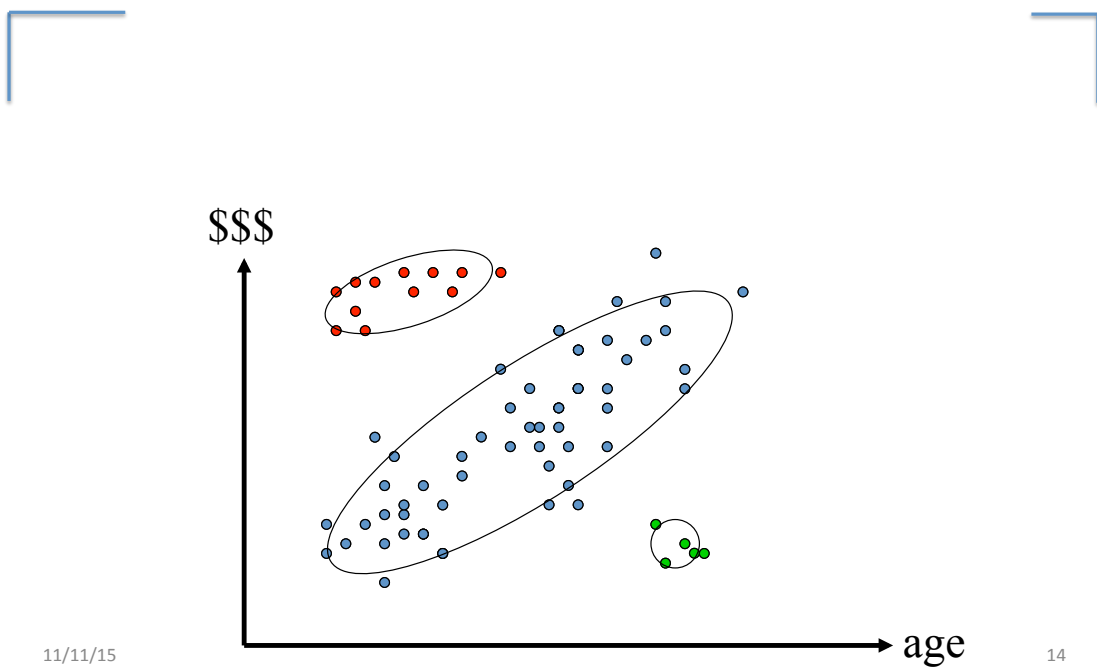


11/11/15

# Partitional clustering (e.g. K=3)

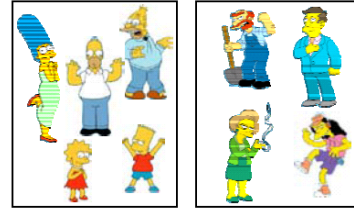


# Partitional clustering (e.g. K=3)



# Clustering Algorithms

- Partitional algorithms
  - Usually start with a random (partial) partitioning
  - Refine it iteratively
- K means clustering
- Mixture-Model based clustering



# Partitioning Algorithms

- Given: a set of objects and the number  $K$
- Find: a partition of  $K$  clusters that optimizes a chosen partitioning criterion
  - Globally optimal: exhaustively enumerate all partitions
  - Effective heuristic methods: K-means and K-medoids algorithms

*too expensive*



# K-Means

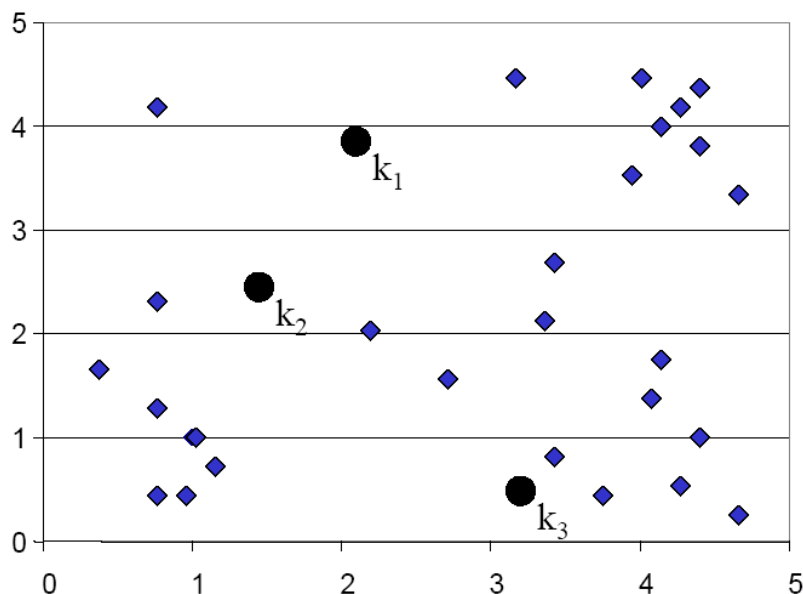
## Algorithm

1. Decide on a value for  $k$ .
2. Initialize the  $k$  cluster centers randomly if necessary.
3. Decide the class memberships of the  $N$  objects by assigning them to the nearest cluster centroids (aka the center of gravity or mean)

$$\vec{\mu}_k = \frac{1}{C_k} \sum_{i \in C_k} \vec{x}_i$$

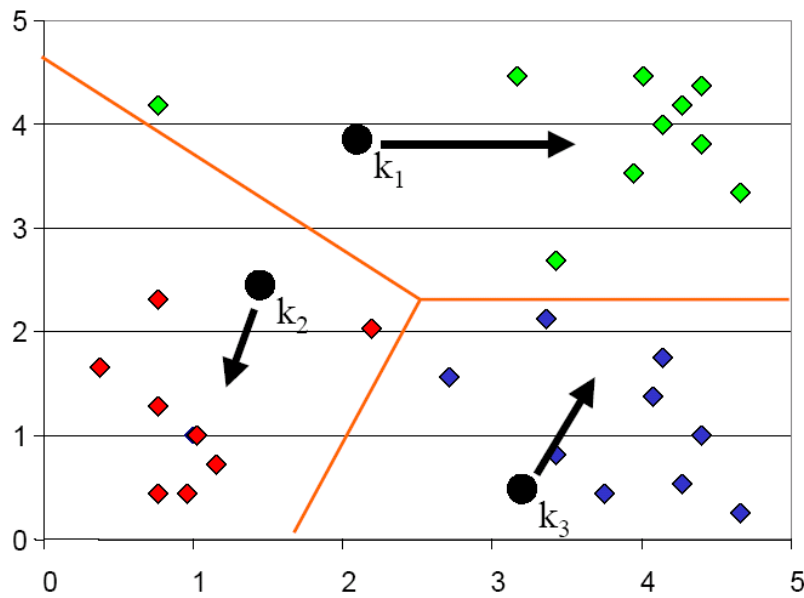
4. Re-estimate the  $k$  cluster centers, by assuming the memberships found above are correct.
5. If none of the  $N$  objects changed membership in the last iteration, exit. Otherwise go to 3.

## K-means Clustering: Step 1 - random guess of cluster centers



## K-means Clustering: Step 2

- Determine the membership of each data points

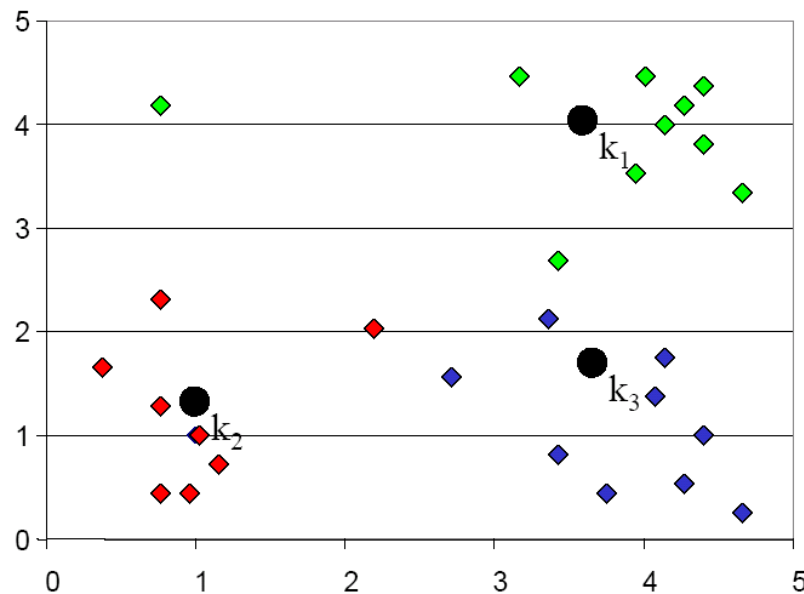


11/11/15

19

## K-means Clustering: Step 3

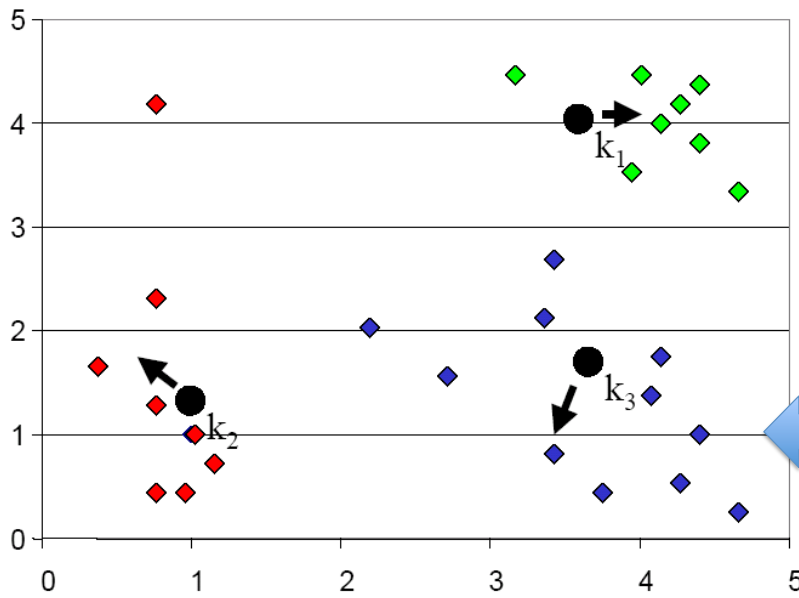
- Adjust the cluster centers



11/11/15

20

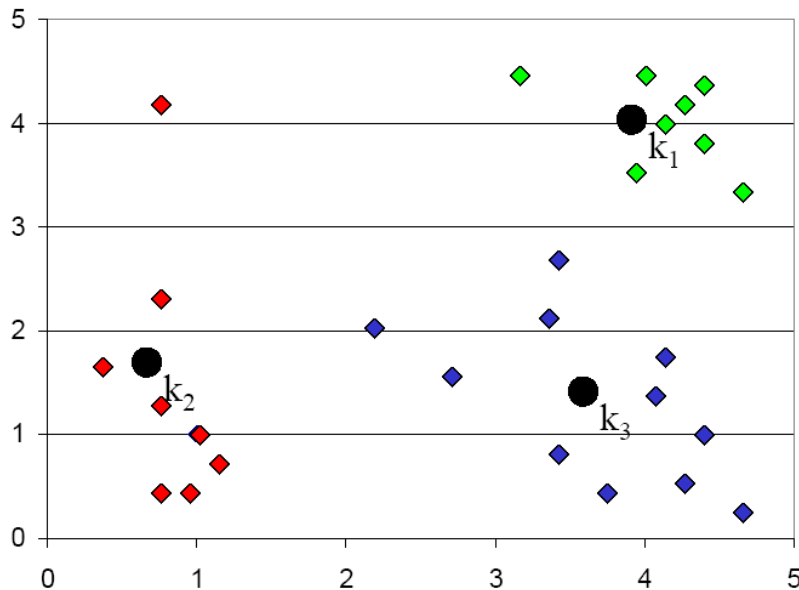
# K-means Clustering: Step 4 - redetermine membership



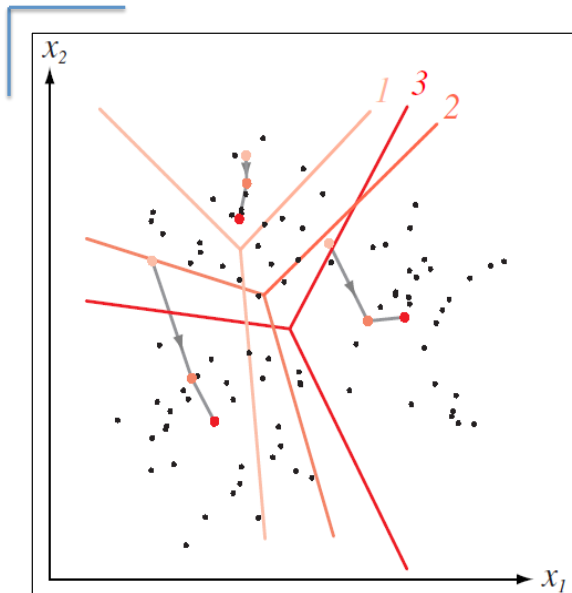
Blue cluster gets more points



# K-means Clustering: Step 5 - readjust cluster centers



# How K-means partitions?



When  $K$  centroids are set/fixed, they partition the whole data space into  $K$  mutually exclusive subspaces to form a partition.

A partition amounts to a

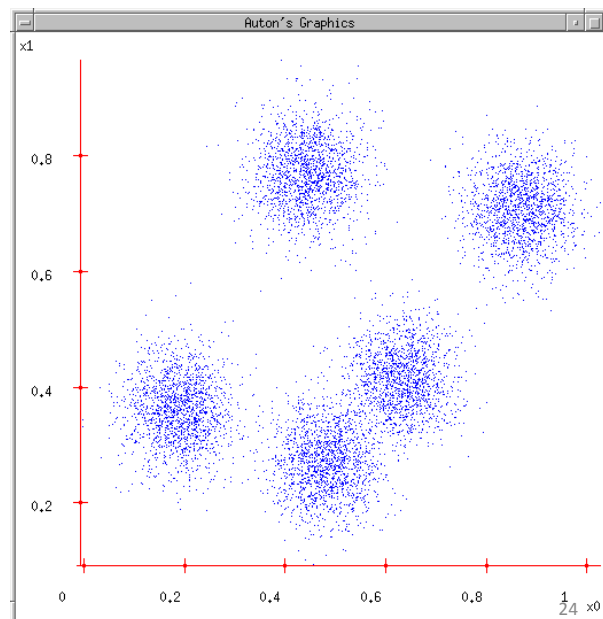
[Voronoi Diagram](#)

Changing positions of centroids leads to a new partitioning.

11/11/15

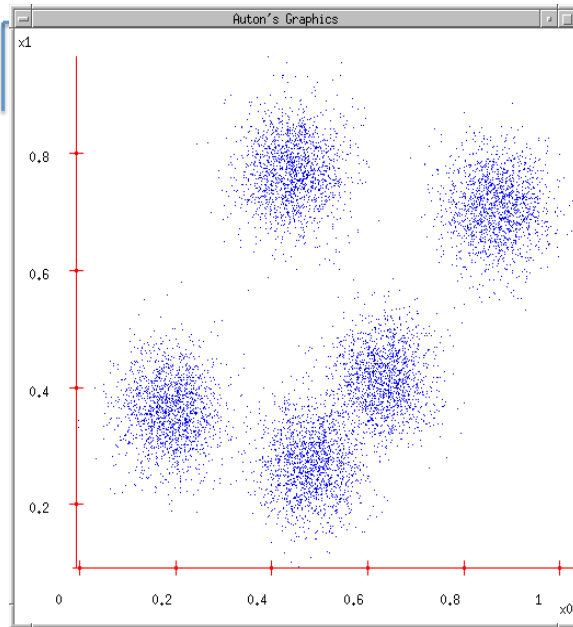
# K-means: another Demo

- K-means
  - Start with a random guess of cluster centers
  - Determine the membership of each data points
  - Adjust the cluster centers



11/11/15

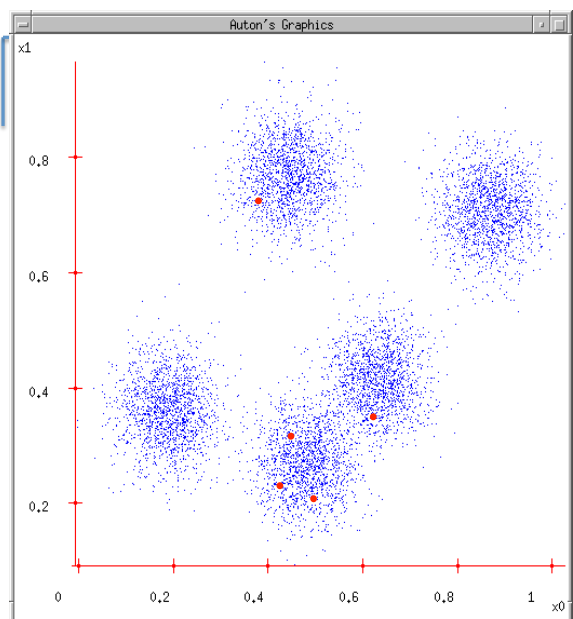
# K-means: another Demo



11/11/15

1. User set up the number of clusters they'd like. (e.g.  $k=5$ )

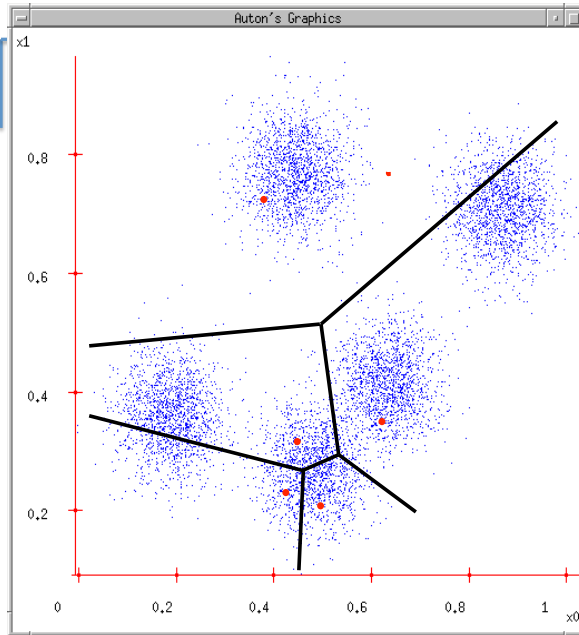
# K-means: another Demo



11/11/15

1. User set up the number of clusters they'd like. (e.g.  $K=5$ )
2. Randomly guess K cluster Center locations

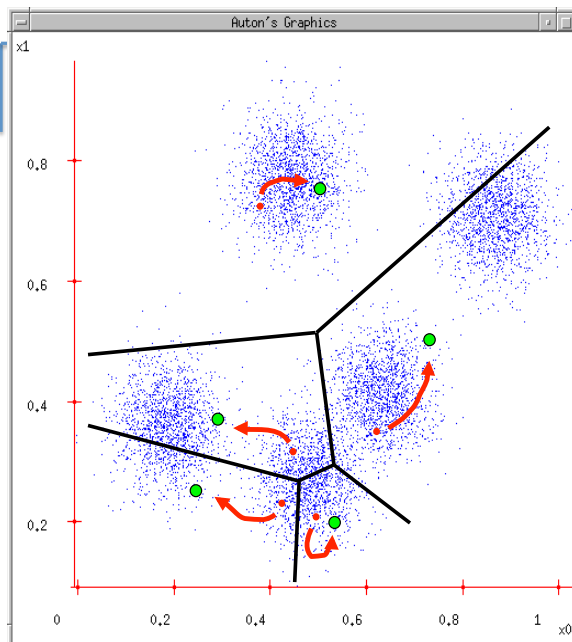
# K-means: another Demo



11/11/15

1. User set up the number of clusters they'd like. (e.g.  $K=5$ )
2. Randomly guess  $K$  cluster Center locations
3. Each data point finds out which Center it's closest to. (Thus each Center "owns" a set of data points)

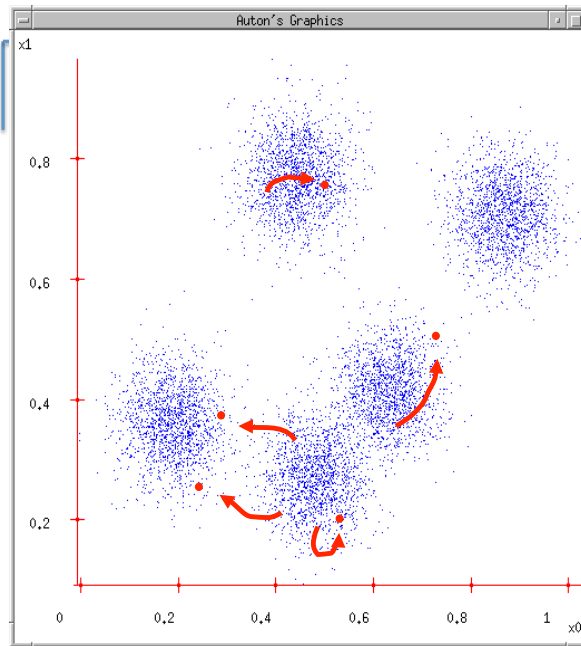
# K-means: another Demo



11/11/15

1. User set up the number of clusters they'd like. (e.g.  $K=5$ )
2. Randomly guess  $K$  cluster centre locations
3. Each data point finds out which centre it's closest to. (Thus each Center "owns" a set of data points)
4. Each centre finds the centroid of the points it owns

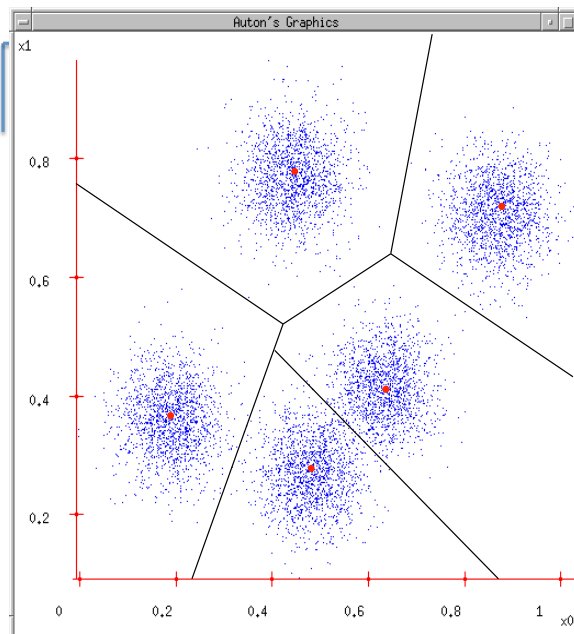
# K-means: another Demo



11/11/15

1. User set up the number of clusters they'd like. (e.g.  $K=5$ )
2. Randomly guess  $K$  cluster centre locations
3. Each data point finds out which centre it's closest to. (Thus each centre "owns" a set of data points)
4. Each centre finds the centroid of the points it owns
5. ...and jumps there

# K-means: another Demo



11/11/15

1. User set up the number of clusters they'd like. (e.g.  $K=5$ )
2. Randomly guess  $K$  cluster centre locations
3. Each data point finds out which centre it's closest to. (Thus each centre "owns" a set of data points)
4. Each centre finds the centroid of the points it owns
5. ...and jumps there
6. ...Repeat until terminated!

# K-means

1. Ask user how many clusters

Computational Complexity:  $O(n)$   
where  $n$  is the number of points?

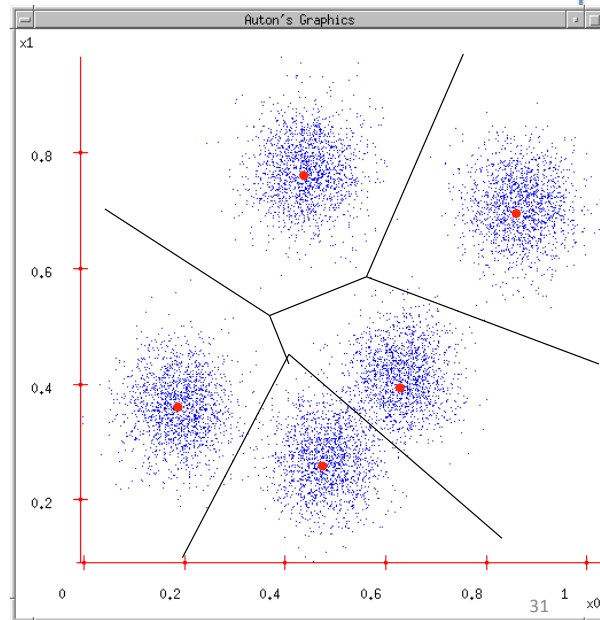
2. Make my guess  $K$  cluster  
Center locations

3. Each datapoint finds out  
which Center it's closest to.

4. Each Center finds the  
centroid of the points it  
owns

Any Computational Problem?

11/11/15



## Time Complexity

- Computing distance between two objs is  $O(p)$  where  $p$  is the dimensionality of the vectors.

Step 3

- Reassigning clusters:  $O(Knp)$  distance computations,

Step 2

- Computing centroids: Each obj gets added once to some centroid:  $O(np)$ .

- Assume these two steps are each done once for  $l$  iterations:  $O(lKnp)$ .

$O(n^3)$  Hierarchical

11/11/15



## Roadmap: clustering

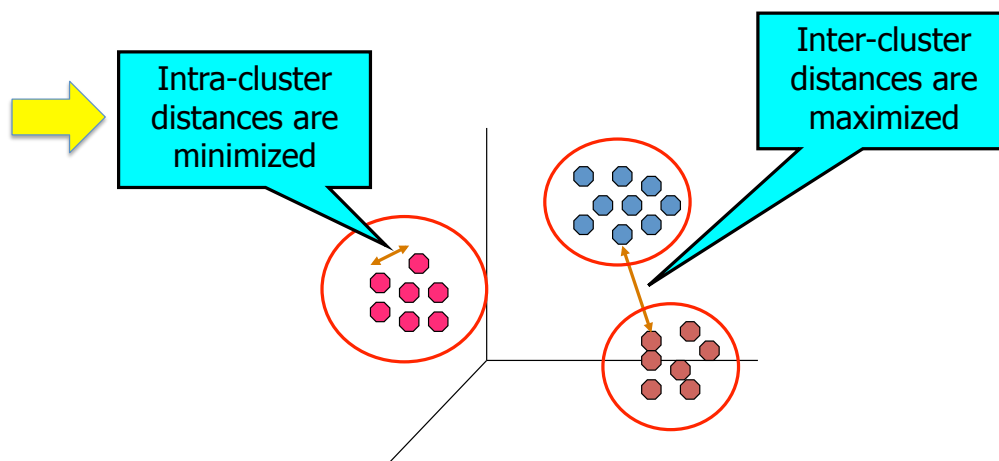
- Definition of "groupness"
- Definition of "similarity/distance"
- Representation for objects
- How many clusters?
- Clustering Algorithms
  - Partitional algorithms
  - Hierarchical algorithms
- ➔ ▪ Formal foundation and convergence

11/11/15

33

## How to Find good Clustering?

- Find groups (clusters) of data points such that data points in a group will be similar (or related) to one another and different from (or unrelated to) the data points in other groups



11/11/15

34

# How to Find good Clustering? E.g.

- Minimize the sum of distance within clusters

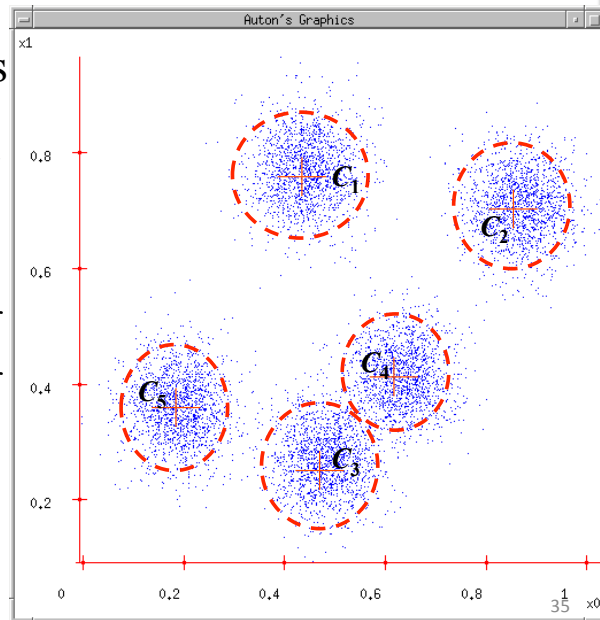
$$\arg \min_{\{\vec{C}_j, m_{i,j}\}} \sum_{j=1}^{k=6} \sum_{i=1}^n m_{i,j} (\vec{x}_i - \vec{C}_j)^2$$

$$m_{i,j} = \begin{cases} 1 & \vec{x}_i \in \text{the } j\text{-th cluster} \\ 0 & \vec{x}_i \notin \text{the } j\text{-th cluster} \end{cases}$$

$$\sum_{j=1}^6 m_{i,j} = 1$$

→ any  $\vec{x}_i \in$  a single cluster

11/11/15



# How to Efficiently Cluster Data?

$$\arg \min_{\{\vec{C}_j, m_{i,j}\}} \sum_{j=1}^6 \sum_{i=1}^n m_{i,j} (\vec{x}_i - \vec{C}_j)^2$$

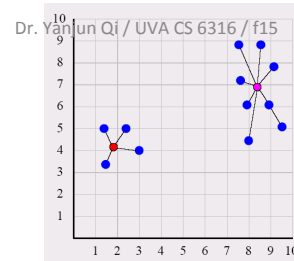
Memberships  $\{m_{i,j}\}$  and centers  $\{C_j\}$  are correlated.

$$\text{Given centers } \{\vec{C}_j\}, m_{i,j} = \begin{cases} 1 & j = \arg \min_k (\vec{x}_i - \vec{C}_k)^2 \\ 0 & \text{otherwise} \end{cases}$$

$$\text{Given memberships } \{m_{i,j}\}, \vec{C}_j = \frac{\sum_{i=1}^n m_{i,j} \vec{x}_i}{\sum_{i=1}^n m_{i,j}}$$

11/11/15

36



# Convergence

- Why should the K-means algorithm ever reach a fixed point?
  - A state in which clusters don't change.
- **K-means is a special case of a general procedure known as the Expectation Maximization (EM) algorithm.**
  - EM is known to converge.
  - Number of iterations could be large.
- **Cluster goodness measure / Loss function to minimize**
  - **sum of squared distances from cluster centroid:**
- Reassignment **monotonically decreases the goodness measure** since each vector is assigned to the closest centroid.

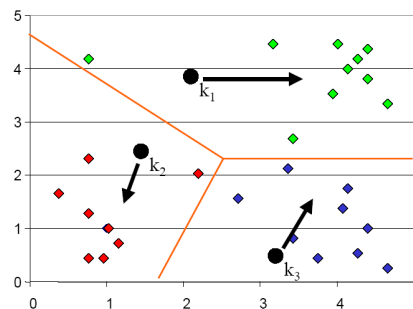
11/11/15

37

Dr. Yanjun Qi / UVA CS 6316 / f15

# Seed Choice

- Results can vary based on random seed selection.

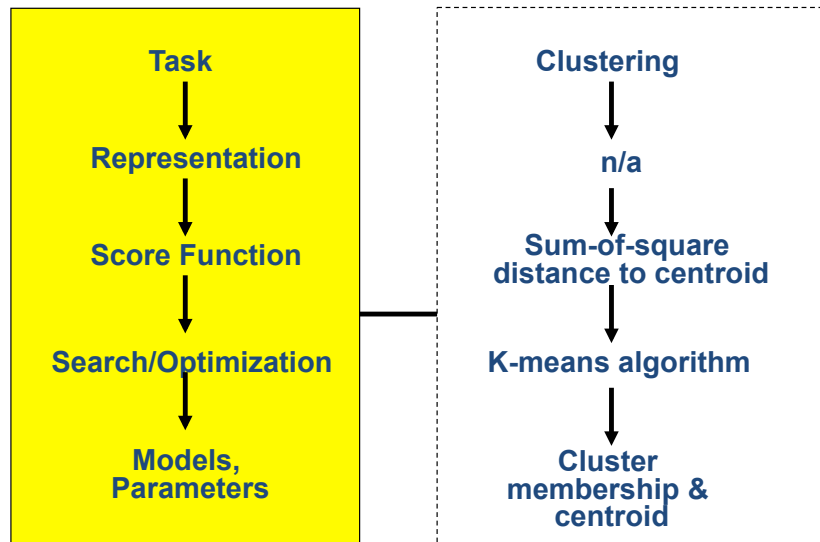


- Some seeds can result in poor convergence rate, or convergence to **sub-optimal clusterings.**
  - Select good seeds using a heuristic (e.g., don't **least similar** to any existing mean)
  - Try out **multiple starting points (very important!!!)**
  - Initialize with the results of **another method.**

11/11/15

38

## (2) K-means Clustering



## Roadmap: clustering

- Definition of "groupness"
- Definition of "similarity/distance"
- Representation for objects
- How many clusters?
- Clustering Algorithms
  - ➔ ▪ Partitional algorithms
    - Hierarchical algorithms
- Formal foundation and convergence

## Other partitioning Methods

- Partitioning around **medoids (PAM)**: instead of averages, use multidim medians as centroids (cluster “prototypes”). Dudoit and Freedland (2002).
- Self-organizing maps **(SOM)**: add an underlying **“topology”** (neighboring structure on a lattice) that relates cluster centroids to one another. Kohonen (1997), Tamayo et al. (1999).
- **Fuzzy k-means**: allow for a “gradation” of points between clusters; soft partitions. Gash and Eisen (2002).
- **Mixture-based clustering**: implemented through an **EM** (Expectation-Maximization) algorithm. This provides **soft partitioning**, and allows for modeling of cluster **centroids and shapes**. Yeung et al. (2001), McLachlan et al. (2002)

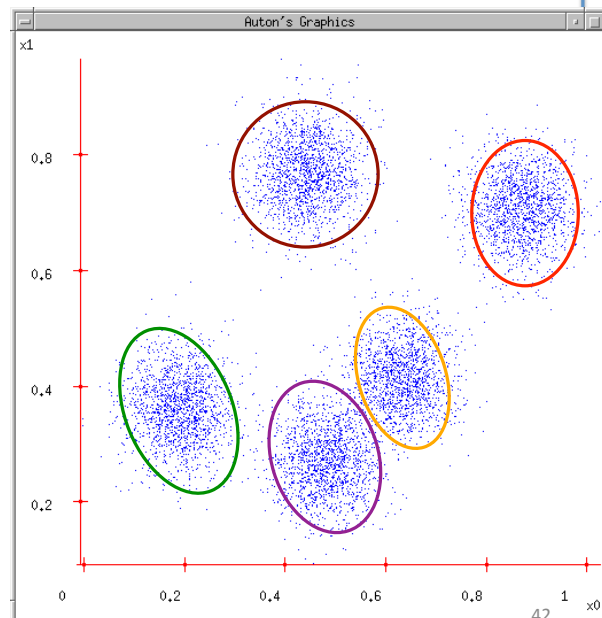
11/11/15

41

## A Gaussian Mixture Model for Clustering

- Assume that data are **generated from a mixture of Gaussian distributions**
- For each Gaussian distribution
  - Center:  $\mu_i$
  - Variance:  $\Sigma_i$  (ignored in the following for simplified equations)
- For each data point
  - Determine membership

$z_{ij}$  : if  $x_i$  belongs to j-th cluster



11/11/15

# Learning a Gaussian Mixture

(with known covariance)

- Probability  $p(x = x_i)$   $\delta = 1, \dots, k$ 

$$p(x = x_i) = \sum_{\mu_j} p(x = x_i, \mu = \mu_j) = \sum_{\mu_j} p(\mu = \mu_j) p(x = x_i | \mu = \mu_j)$$

Total law of probability

11/11/15

43

# Learning a Gaussian Mixture

(with known covariance)

- Probability  $p(x = x_i)$ 

$$p(x = x_i) = \sum_{\mu_j} p(x = x_i, \mu = \mu_j) = \sum_{\mu_j} p(\mu = \mu_j) p(x = x_i | \mu = \mu_j)$$

$$= \sum_{\mu_j} p(\mu = \mu_j) \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{\|x_i - \mu_j\|_2^2}{2\sigma^2}\right)$$

← Assuming
- Log-likelihood of data  $\log p(x_1, x_2, x_3, \dots, x_n) =$ 

$$\sum_{i=1}^n \log p(x = x_i) = \sum_i \log \left[ \sum_{\mu_j} p(\mu = \mu_j) \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{\|x_i - \mu_j\|_2^2}{2\sigma^2}\right) \right]$$
- Apply MLE to find optimal parameters  $\{p(\mu = \mu_j), \mu_j\}_j$   $\delta = 1, \dots, k$

11/11/15

44

# Learning a Gaussian Mixture

(with known covariance)

**E-Step**

*$\mu_j - k$  means*

$$E[z_{ij}] = p(\mu = \mu_j | x = x_i)$$

*membership  
of data  $x_i$   
to cluster  $j$*

$$\begin{aligned} &= \frac{p(x = x_i | \mu = \mu_j) p(\mu = \mu_j)}{\sum_{n=1}^k p(x = x_i | \mu = \mu_n) p(\mu = \mu_n)} \\ &= \frac{e^{-\frac{1}{2\sigma^2}(x_i - \mu_j)^2} p(\mu = \mu_j)}{\sum_{n=1}^k e^{-\frac{1}{2\sigma^2}(x_i - \mu_n)^2} p(\mu = \mu_n)} \end{aligned}$$

11/11/15

45

# Learning a Gaussian Mixture

(with known covariance)

**M-Step**

*$\leftarrow$  mean  $\Rightarrow$  centroid  $\frac{1}{N_j} \sum_{i=1}^n \mu_{ij} x_i$*

$$\mu_j \leftarrow \frac{1}{\sum_{i=1}^n E[z_{ij}]} \sum_{i=1}^n E[z_{ij}] x_i$$

$$p(\mu = \mu_j) \leftarrow \frac{1}{n} \sum_{i=1}^n E[z_{ij}]$$

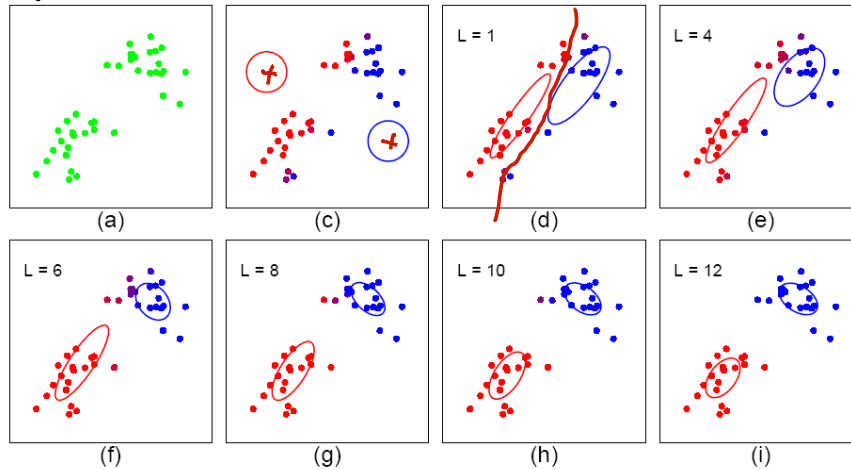
Covariance:  $\Sigma_j$  ( $j$ : 1 to  $K$ ) will also be derived in the M-step under a full setting

11/11/15

46

# Expectation-Maximization for training GMM

- Start:
  - "Guess" the centroid  $m_k$  and covariance  $S_k$  of each of the K clusters
- Loop each cluster, revising both the mean (centroid position) and covariance (shape)



## Recap: K-means iterative learning

$$\arg \min_{\{\bar{C}_j, m_{i,j}\}} \sum_{j=1}^6 \sum_{i=1}^n m_{i,j} (\bar{x}_i - \bar{C}_j)^2$$

Memberships  $\{m_{i,j}\}$  and centers  $\{C_j\}$  are correlated.

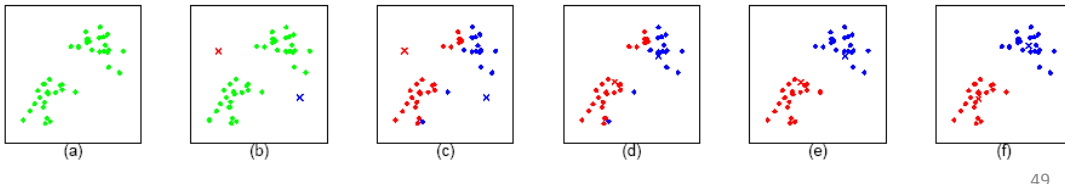
**E-Step** Given centers  $\{\bar{C}_j\}$ ,  $m_{i,j} = \begin{cases} 1 & j = \arg \min_k (\bar{x}_i - \bar{C}_j)^2 \\ 0 & \text{otherwise} \end{cases}$

**M-Step** Given memberships  $\{m_{i,j}\}$ ,  $\bar{C}_j = \frac{\sum_{i=1}^n m_{i,j} \bar{x}_i}{\sum_{i=1}^n m_{i,j}}$

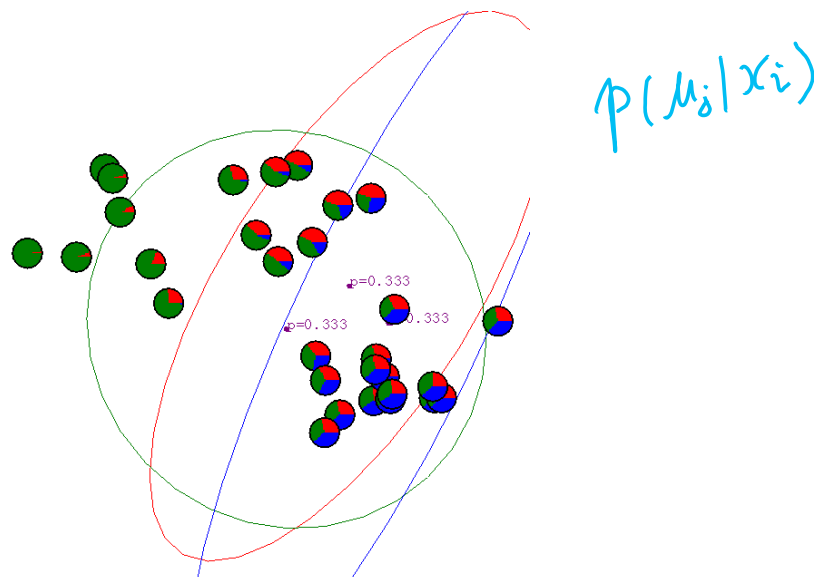


## Compare: K-means

- The EM algorithm for mixtures of Gaussians is like a "soft version" of the K-means algorithm.
- In the K-means "E-step" we do **hard** assignment:
- In the **K-means "M-step"** we update the means as the **weighted sum** of the data, but now the weights are 0 or 1:

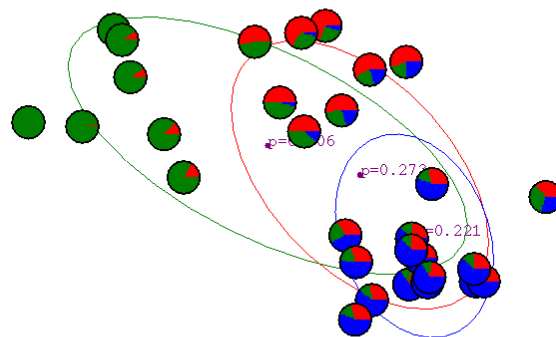


## Gaussian Mixture Example: Start



# After First Iteration

For each point, revising its proportions belonging to each of the K clusters

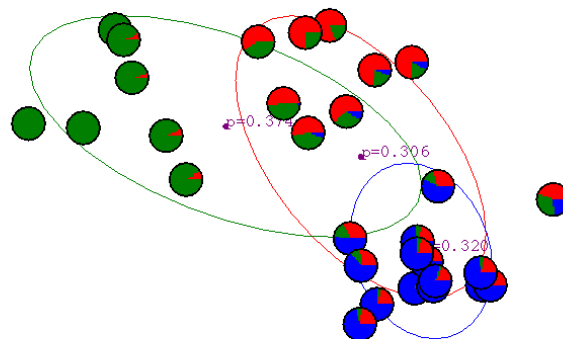


For each cluster, revising its mean (centroid position), covariance (shape) and proportion in the mixture

11/11/15

# After 2nd Iteration

For each point, revising its proportions belonging to each of the K clusters

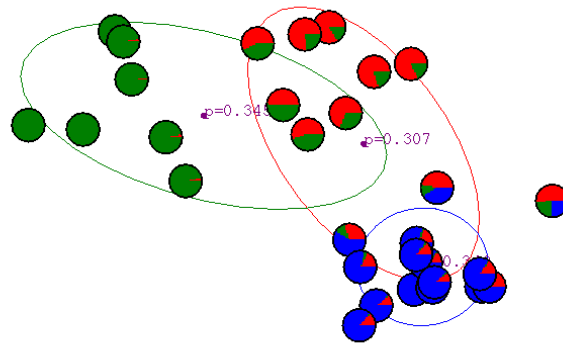


For each cluster, revising its mean (centroid position), covariance (shape) and proportion in the mixture

11/11/15

# After 3rd Iteration

For each point, revising its proportions belonging to each of the K clusters

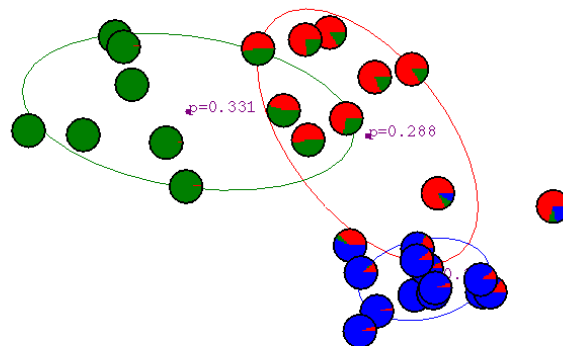


For each cluster, revising its mean (centroid position), covariance (shape) and proportion in the mixture

11/11/15

# After 4th Iteration

For each point, revising its proportions belonging to each of the K clusters

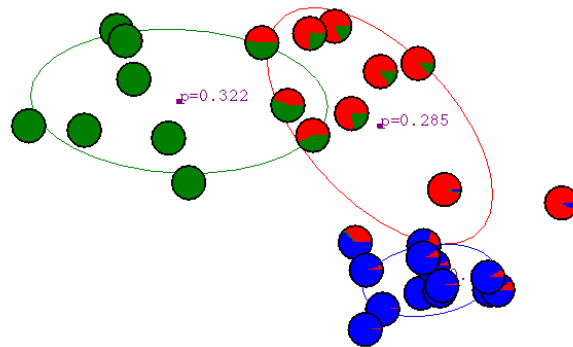


For each cluster, revising its mean (centroid position), covariance (shape) and proportion in the mixture

11/11/15

# After 5th Iteration

For each point, revising its proportions belonging to each of the K clusters

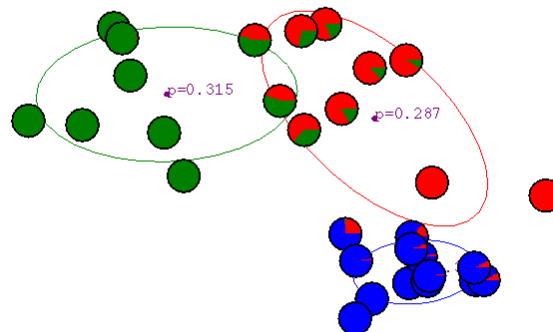


For each cluster, revising its mean (centroid position), covariance (shape) and proportion in the mixture

11/11/15

# After 6th Iteration

For each point, revising its proportions belonging to each of the K clusters

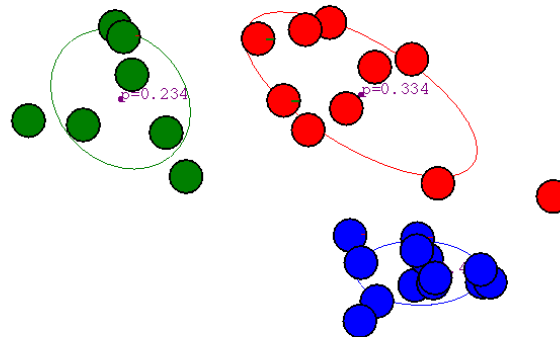


For each cluster, revising its mean (centroid position), covariance (shape) and proportion in the mixture

11/11/15

# After 20th Iteration

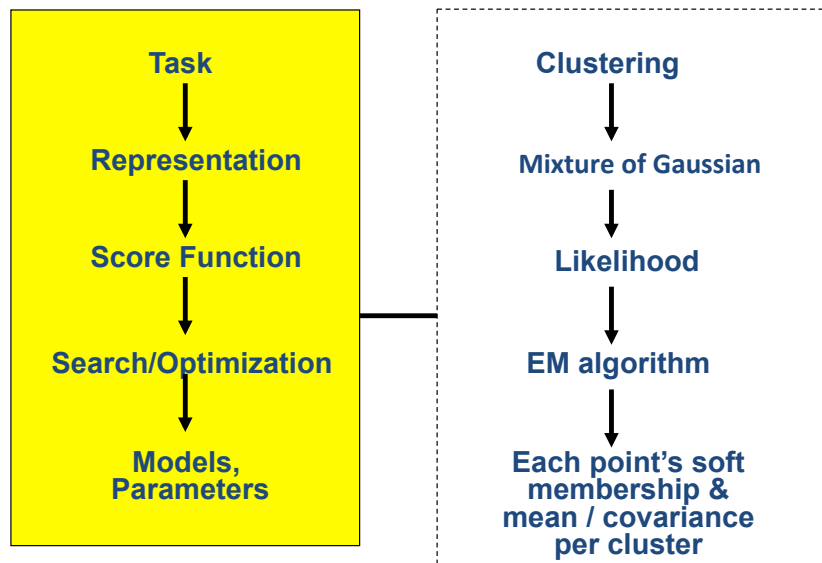
For each point, revising its proportions belonging to each of the K clusters



For each cluster, revising its mean (centroid position), covariance (shape) and proportion in the mixture

11/11/15

## (3) GMM Clustering



$$\sum_i \log p(x = x_i) = \sum_i \log \left[ \sum_{\mu_j} p(\mu = \mu_j) \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{\|x_i - \mu_j\|_2}{2\sigma^2}\right) \right]$$

11/11/15

58

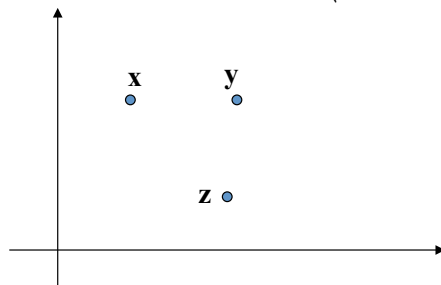
## M-step (more in L23 EM lecture)

$$\Sigma_j^{(t+1)} = \frac{\sum_{i=1}^n E[z_{ij}]^{(t)} (x_i - \mu_j^{(t+1)})(x_i - \mu_j^{(t+1)})^T}{\sum_{i=1}^n E[z_{ij}]^{(t)}}$$

## Problems (I)

- Both k-means and mixture models need to compute centers of clusters and explicit distance measurement
  - Given strange distance measurement, the center of clusters can be hard to compute

E.g.,  $\|\vec{x} - \vec{x}'\|_{\infty} = \max(|x_1 - x_1'|, |x_2 - x_2'|, \dots, |x_p - x_p'|)$

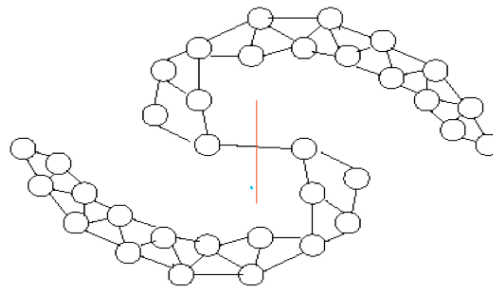


$$\|\mathbf{x} - \mathbf{y}\|_{\infty} = \|\mathbf{x} - \mathbf{z}\|_{\infty}$$

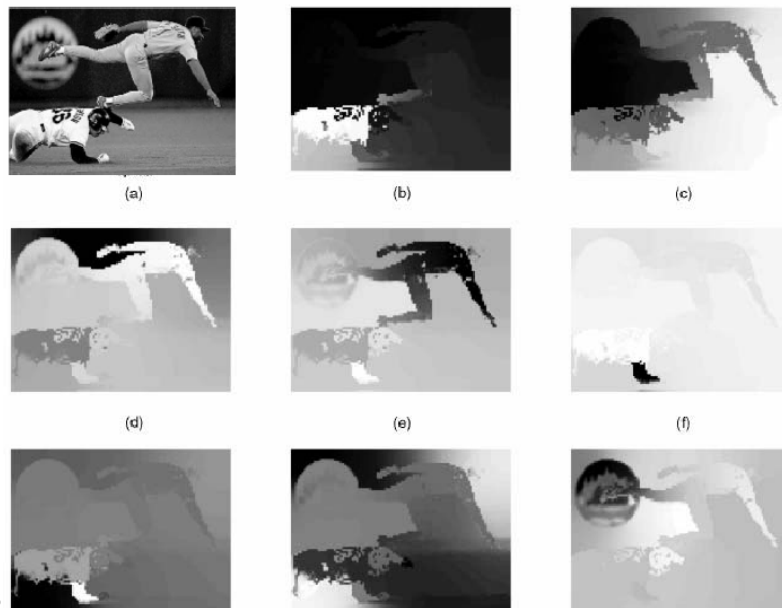
# Problem (II)

- Both k-means and mixture models look for compact clustering structures *tight*
  - In some cases, connected clustering structures are more desirable

Graph based clustering  
e.g. MinCut, Spectral clustering



## e.g. Image Segmentation through minCut



## Roadmap: clustering

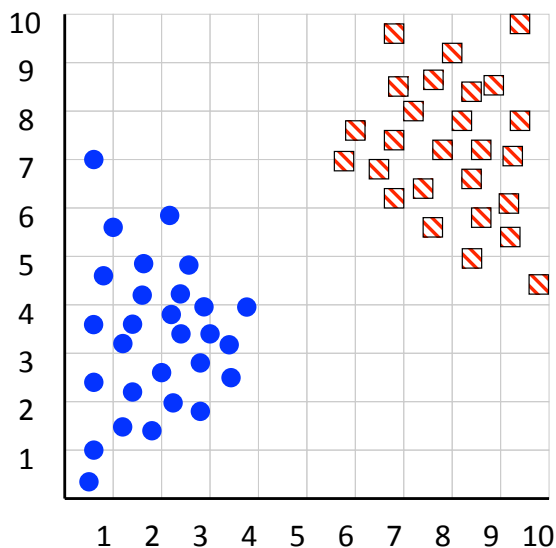
- Definition of "groupness"
- Definition of "similarity/distance"
- Representation for objects
- ➔ ▪ How many clusters?
- Clustering Algorithms
  - Partitional algorithms
  - Hierarchical algorithms
- Formal foundation and convergence

11/11/15

63

## How can we tell the *right* number of clusters?

In general, this is a unsolved problem. However there exist many approximate methods.



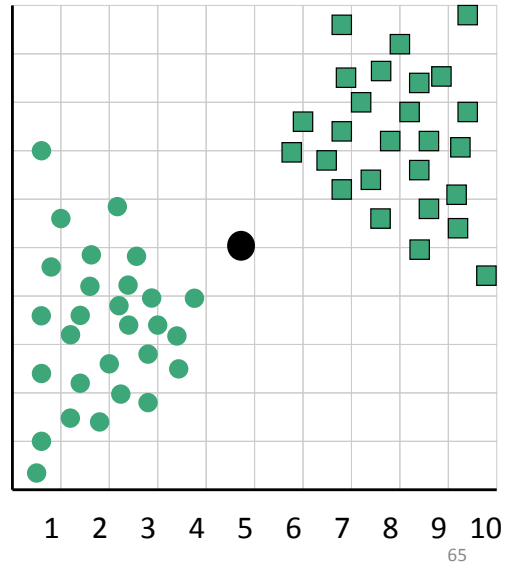
11/11/15

64



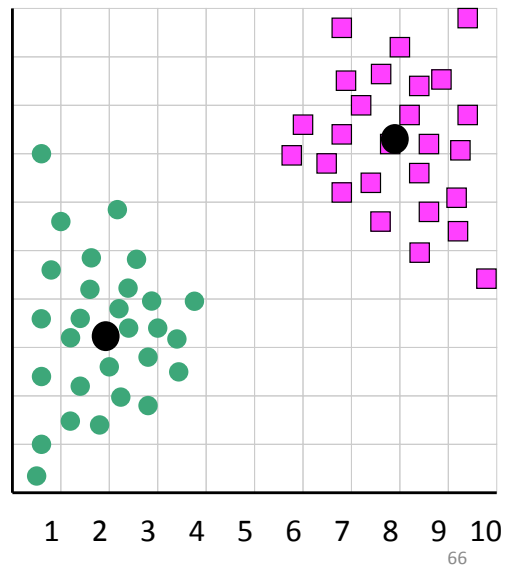
$$\arg \min_{\{\vec{C}_j, m_{i,j}\}} \sum_{j=1}^K \sum_{i=1}^n m_{i,j} (\vec{x}_i - \vec{C}_j)^2$$

When  $k = 1$ , the objective function is 873.0



11/11/15

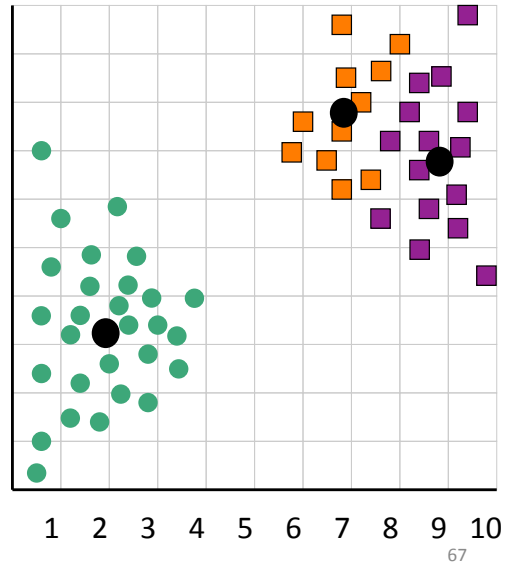
When  $k = 2$ , the objective function is 173.1



11/11/15

When  $k = 3$ , the objective function is 133.6

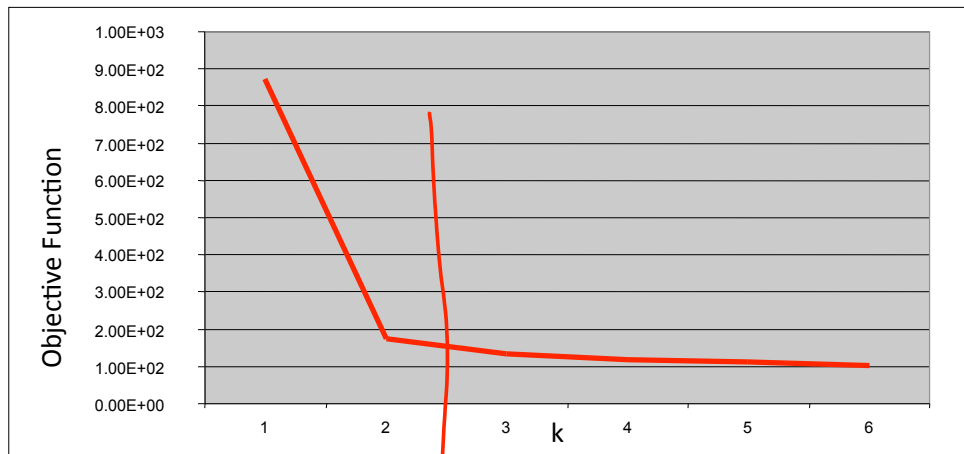
$K=n, obj = 0$



11/11/15

We can plot the objective function values for  $k$  equals 1 to 6...

The abrupt change at  $k = 2$ , is highly suggestive of two clusters in the data. This technique for determining the number of clusters is known as “knee finding” or “elbow finding”.



Note that the results are not always as clear cut as in this toy example

11/11/15

68

# What Is A Good Clustering?

- **Internal** criterion: A good clustering will produce **high quality clusters** in which:
  - the intra-class (that is, intra-cluster) similarity is high *Within*
  - the inter-class similarity is low *between*
  - The measured **quality** of a clustering depends on both the data **representation** and the **similarity** measure used
- **External** criteria for clustering quality
  - Quality measured by its ability to discover some or all of the hidden patterns or latent classes in gold standard data
  - Assesses a clustering **with respect to ground truth**
  - Example:
    - **Purity**
    - entropy of classes in clusters (or mutual information between classes and clusters)

11/11/15

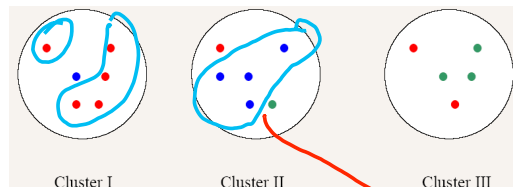
69

## External Evaluation of Cluster Quality, e.g. using purity

- Simple measure: **purity** the ratio between the dominant class in the cluster and the size of cluster
  - Assume data samples with C gold standard classes/groups, while the clustering algorithms produce K clusters,  $\omega_1, \omega_2, \dots, \omega_K$  with  $n_i$  members.

$$Purity(\omega_i) = \frac{1}{n_i} \max_j (n_{ij}) \quad j \in C$$

- Example



Cluster I: Purity =  $1/6 (\max(5, 1, 0)) = 5/6$

Cluster II: Purity =  $1/6 (\max(1, 4, 1)) = 4/6$

Cluster III: Purity =  $1/5 (\max(2, 0, 3)) = 3/5$

11/11/15

70

## References

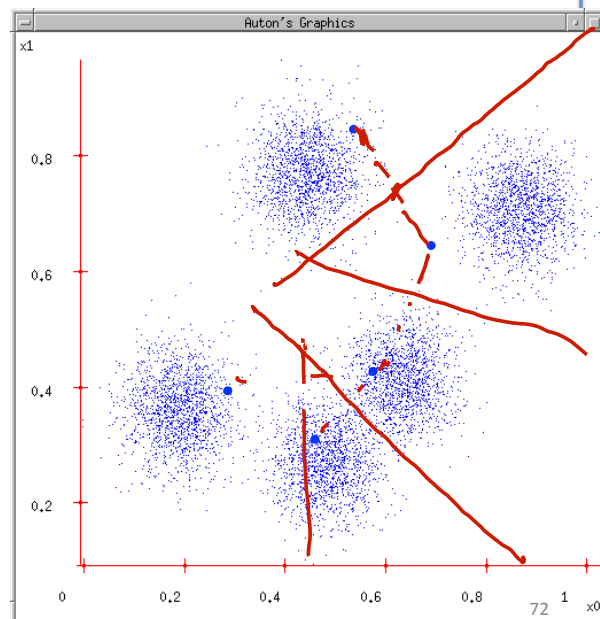
- ❑ Hastie, Trevor, et al. *The elements of statistical learning*. Vol. 2. No. 1. New York: Springer, 2009.
- ❑ Big thanks to Prof. Eric Xing @ CMU for allowing me to reuse some of his slides
- ❑ Big thanks to Prof. Ziv Bar-Joseph @ CMU for allowing me to reuse some of his slides
- ❑ clustering slides from Prof. Rong Jin @ MSU

11/11/15

71

## Extra practice: K-means

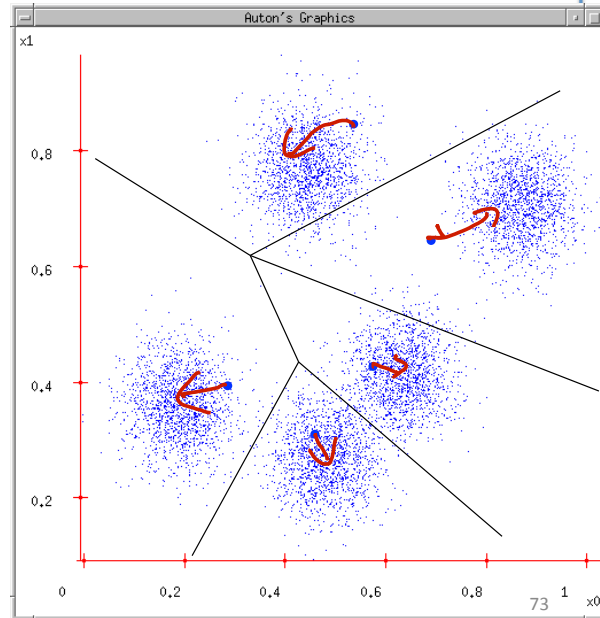
1. Ask user how many clusters they'd like. (e.g.  $k=5$ )
2. Randomly guess  $k$  cluster Center locations



11/11/15

# Extra practice: K-means

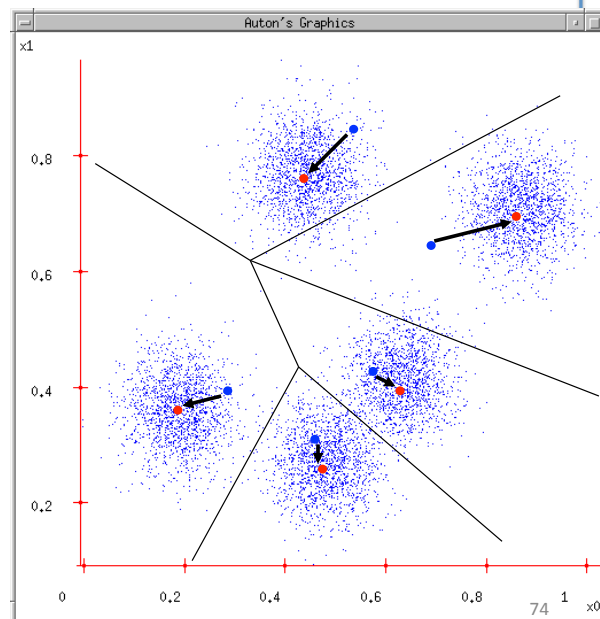
1. Ask user how many clusters they'd like. (e.g.  $k=5$ )
2. Randomly guess  $k$  cluster Center locations
3. Each datapoint finds out which Center it's closest to. (Thus each Center "owns" a set of datapoints)



11/11/15

# Extra practice: K-means

1. Ask user how many clusters they'd like. (e.g.  $k=5$ )
2. Randomly guess  $k$  cluster Center locations
3. Each datapoint finds out which Center it's closest to.
4. Each Center finds the centroid of the points it owns



11/11/15

# K-means: extra practice

1. Ask user how many clusters they'd like. (e.g.  $k=5$ )
2. Randomly guess  $k$  cluster Center locations
3. Each datapoint finds out which Center it's closest to.
4. Each Center finds the centroid of the points it owns

