

CSI 4142 Introduction to Data Science

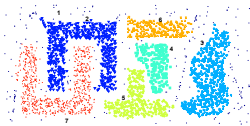
Unsupervised learning (cluster analysis)

(Slides by HL Viktor © based on Chapters 8 and 9 of Han et. al. and Chapter 7 of Data mining by Tan et. al.)

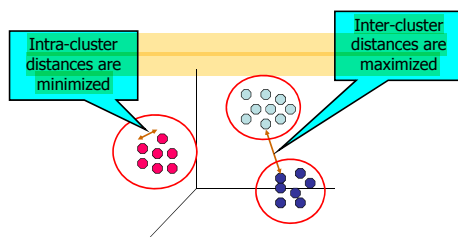
1

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*, ...)
 - grouping similar data objects into clusters
- **Unsupervised learning**: no predefined classes (i.e., *learning by observations* vs. learning by examples: supervised)

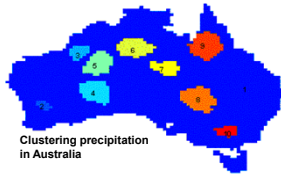


What is Cluster Analysis?



Examples

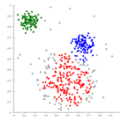
- Geospatial data: an early adopter
- Customer profiling
- Document clustering
- City planning, etc.



4

Quality: What Is Good Clustering?

- A **good clustering** method will produce high quality clusters
 - high **intra-class** similarity: **cohesive** within clusters
 - low **inter-class** similarity: **distinctive** between clusters
- The **quality** of a clustering method depends on
 - the **similarity measure** used by the method
 - its **implementation**, and
 - Its ability to discover some or all of the **hidden patterns**



5

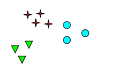
Notion of a Cluster can be Ambiguous



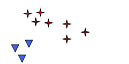
How many clusters?



Six Clusters



Two Clusters



Four Clusters

Measures for “Similarity and Dissimilarity”

- **Similarity**
 - Numerical measure of how alike two data objects are
 - Value is higher when objects are more alike
 - Often falls in the range [0,1], [0.1, 0.9] or [-1, 1]
- **Dissimilarity** (e.g., distance)
 - Numerical measure of how different two data objects are
 - Lower when objects are more alike
 - Minimum dissimilarity is often 0
 - Upper limit varies
- **Proximity** refers to a similarity or dissimilarity

7

Data Matrix and Dissimilarity Matrix

- **Data matrix**
 - n data points with p dimensions
 - Two modes
- $$\begin{bmatrix} x_{11} & \dots & x_{1f} & \dots & x_{1p} \\ \dots & \dots & \dots & \dots & \dots \\ x_{i1} & \dots & x_{if} & \dots & x_{ip} \\ \dots & \dots & \dots & \dots & \dots \\ x_{n1} & \dots & x_{nf} & \dots & x_{np} \end{bmatrix}$$
- **Dissimilarity matrix**
 - n data points, but registers only the distance
 - A triangular matrix
 - Single mode
- $$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ \vdots & \vdots & \vdots & \ddots & \\ d(n,1) & d(n,2) & \dots & \dots & 0 \end{bmatrix}$$

8

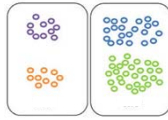
Data Matrix and Dissimilarity Matrix

- **Data matrix**
 - n data points with p = 3 dimensions
- | Age | Income | Gender |
|------|-----------|--------|
| 29 | 1,000,000 | F |
| 23 | 30,000 | F |
| 28 | 8,000,000 | M |
| | | |
- **Dissimilarity matrix**
 - n data points, but registers only the distance
 - A triangular matrix
 - How do we measure distance?
- $$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ \vdots & \vdots & \vdots & \ddots & \\ d(n,1) & d(n,2) & \dots & \dots & 0 \end{bmatrix}$$

9

Proximity Measure for Nominal Attributes

- Can take 2 or more states
- E.g. Color:** orange, blue, green, purple
- Typical Method: Use binary attributes
create a new binary attribute for each of the M nominal states



10

Ordinal Variables

- An ordinal variable can be discrete or continuous
- Order is important, e.g., rank (assistant, associate, full, chair)
- Can be treated like interval-scaled
 - replace x_{if} by their rank $r_{if} \in \{1, \dots, M_f\}$
 - map the range of each variable onto e.g. $[0, 1]$ by replacing i -the object in the f -th variable by

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

Original	Rank	Scaled
Assistant	1	0
Associate	2	0.33
Full	3	0.67
Chair	4	1

11

Proximity Measure for Binary Attributes

- A contingency table for binary data

		Object j		
		1	0	sum
Object i	1	q	r	$q + r$
	0	s	t	$s + t$
sum		$q + s$	$r + t$	p

- Symmetric (equal importance): $d(i, j) = \frac{r+s}{q+r+s+t}$
- Asymmetric (not equal importance): $d(i, j) = \frac{r+s}{q+r+s}$
- Jaccard coefficient (asymmetric): $sim_{Jaccard}(i, j) = \frac{q}{q+r+s}$

12

Dissimilarity between Binary Variables

- Example (Medical diagnosis)

	1	0	sum
1	q	r	q+r
0	s	t	s+t
sum	q+s	r+t	p

Name	Gender	Fever	Cough	Test-1	Test-2	Test-3	Test-4
Jack	M	1	0	1	0	0	0
Mary	F	1	0	1	0	1	0
Jim	M	1	1	0	0	0	0

Assumptions:

- Gender is a symmetric attribute (equally important)...
- The remaining attributes are asymmetric binary (P more important than N); Let the values Y and P be 1, and the value N set to 0

$$d(jack, mary) = \frac{0+1}{2+0+1} = 0.33$$

$$d(i, j) = \frac{r+s}{q+r+s}$$

$$d(jack, jim) = \frac{1+1}{1+1+1} = 0.67$$

$$d(jim, mary) = \frac{1+2}{1+1+2} = 0.75$$

13

Distance on Numeric Data: Minkowski Distance

- Minkowski distance**: A popular distance measure

$$d(i, j) = \sqrt[h]{|x_{i1} - x_{j1}|^h + |x_{i2} - x_{j2}|^h + \dots + |x_{ip} - x_{jp}|^h}$$

where $i = (x_{i1}, x_{i2}, \dots, x_{ip})$ and $j = (x_{j1}, x_{j2}, \dots, x_{jp})$ are two p -dimensional data objects, and h is the order (the distance so defined is also called L - h norm)

- Properties
 - $d(i, j) > 0$ if $i \neq j$, and $d(i, i) = 0$ (Positive definiteness)
 - $d(i, j) = d(j, i)$ (Symmetry)
 - $d(i, j) \leq d(i, k) + d(k, j)$ (Triangle Inequality)
- A distance that satisfies these properties is a **metric**

14

Special Cases of Minkowski Distance

- $h = 1$: **Manhattan** (city block, L_1 norm) distance
 - E.g., the Hamming distance: the number of bits that are different between two binary vectors

$$d(i, j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \dots + |x_{ip} - x_{jp}|$$

- $h = 2$: **Euclidean** (norm) distance

$$d(i, j) = \sqrt{(|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \dots + |x_{ip} - x_{jp}|^2)}$$

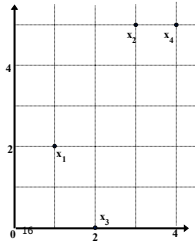
- $h \rightarrow \infty$: **"supremum"** (L_{\max} norm, L_{∞} norm) distance.
 - This is the maximum difference between any component (attribute) of the vectors

$$d(i, j) = \lim_{h \rightarrow \infty} \left(\sum_{f=1}^p |x_{if} - x_{jf}|^h \right)^{\frac{1}{h}} = \max_f |x_{if} - x_{jf}|$$

15

Example: Minkowski Distance

point	attribute 1	attribute 2
x1	1	2
x2	3	5
x3	2	0
x4	4	5



Manhattan (L_1)

L	x1	x2	x3	x4
x1	0			
x2	5	0		
x3	3	6	0	
x4	6	1	7	0

Euclidean (L_2)

L2	x1	x2	x3	x4
x1	0			
x2	3.61	0		
x3	2.24	5.1	0	
x4	4.24	1	5.39	0

Supremum

L_∞	x1	x2	x3	x4
x1	0			
x2	3	0		
x3	2	5	0	
x4	3	1	5	0

Dissimilarity Matrices

Attributes of Mixed Type

- A database may contain all attribute types
 - Nominal, symmetric binary, asymmetric binary, numeric, ordinal
- One may use a **weighted formula to combine their effects**

$$d(i, j) = \frac{\sum_{f=1}^p \delta_{ij}^{(f)} d_{ij}^{(f)}}{\sum_{f=1}^p \delta_{ij}^{(f)}}$$

- f is binary or nominal:
 - $d_{ij}^{(f)} = 0$ if $x_{if} = x_{jf}$, or $d_{ij}^{(f)} = 1$ otherwise
- f is numeric: use the normalized distance
- f is ordinal

- Compute ranks r_{if} and
- Treat z_{if} as interval-scaled

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

17

More about cluster analysis methods

- Partitioning: k-means algorithm
- Measuring tendency
- Density-based: DBSCAN method
- Model-based: EM technique
- Curse of dimensionality
- Subspace clustering (bi-clustering)

18

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)

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

- 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

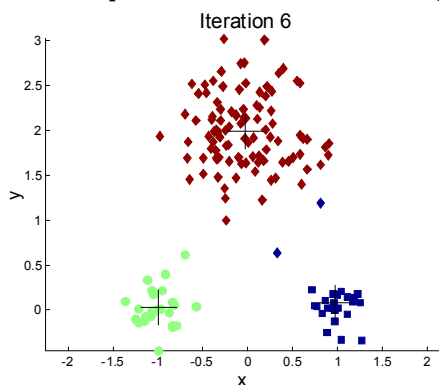
19

The K -Means Clustering Method

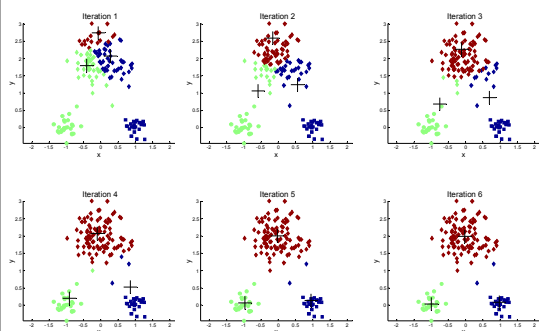
- Given k , the k -means algorithm is implemented in four steps:
 - Partition objects into k nonempty subsets
 - Compute seed points as the centroids of the clusters of the current partitioning (the centroid is the center, i.e., **mean point**, of the cluster)
 - Assign each object to the cluster with the nearest seed point
 - Go back to Step 2, stop when the assignment does not change

20

Example of K-means Clustering



Example of K-means Clustering



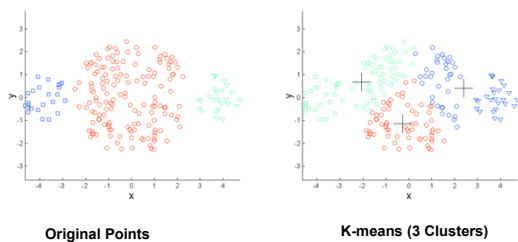
Comments on the *K-Means* Method

- **Strength: Efficient:** $O(tkn)$, where n is # objects, k is # clusters, and t is # iterations. Normally, $k, t \ll n$.
 - Comparing: PAM: $O(k(n-k)^2)$, CLARA: $O(ks^2 + k(n-k))$
- **Comment:** Often terminates at a *local optimal*.
- **Weakness**
 - Applicable only to objects in a continuous n -dimensional space
 - Using the k -modes method for categorical data
 - In comparison, k -medoids can be applied to a wide range of data
 - Need to specify k , the number of clusters, in advance
 - Sensitive to noisy data and outliers
 - Not suitable to discover clusters with non-convex shapes

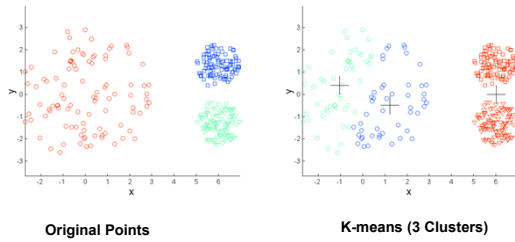


23

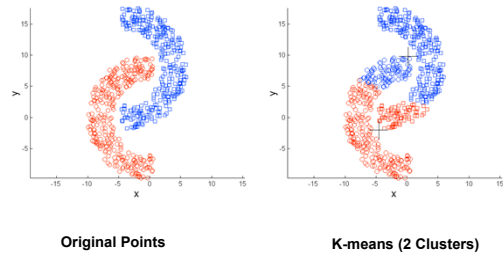
Limitations of K-means: Differing Sizes



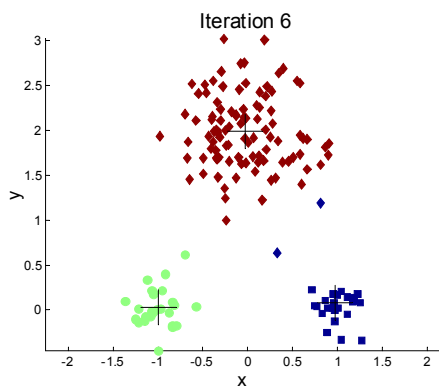
Limitations of K-means: Differing Density

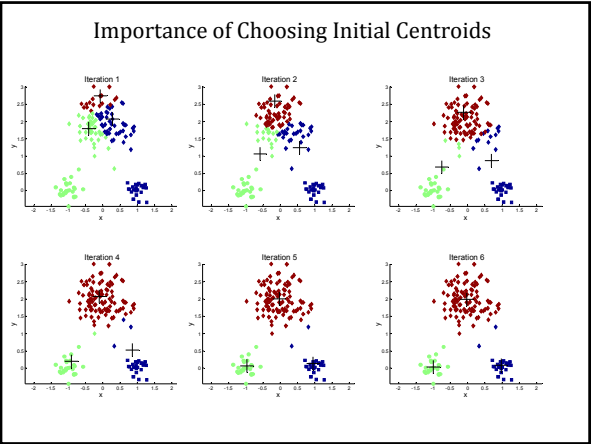


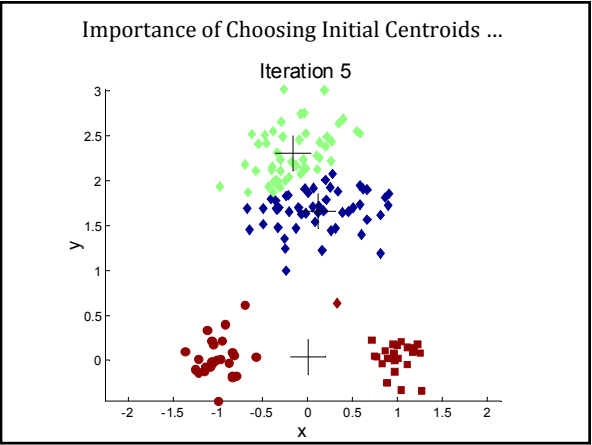
Limitations of K-means: Non-globular Shapes

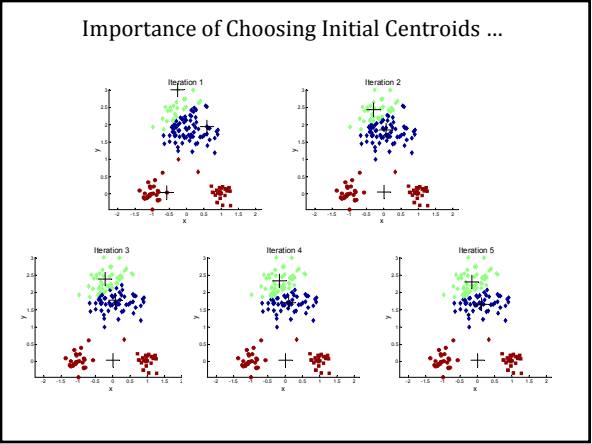


Importance of Choosing Initial Centroids









Solutions to Initial Centroids Problem

- Multiple runs
 - Helps, but probability is not on your side
- Select more than k initial centroids and then select among these initial centroids
 - Select most widely separated
- Post-processing
- Generate a larger number of clusters and then perform a merge

Evaluating clusters

- Do we actually have clusters that may be found?
- What does a cluster correspond to?
- Is the number of clusters correct?

32

Assessing Clustering Tendency

- Assess if non-random structure exists in the data by measuring the probability that the data is generated by a uniform data distribution
- Test spatial randomness by statistic test: Hopkins Statistic
 - Given a dataset D regarded as a sample of a random variable o , determine how far away o is from being uniformly distributed in the data space
 - Sample n points, p_1, \dots, p_n , uniformly from D . For each p_i find its nearest neighbor in D : $x_i = \min\{\text{dist}(p_i, v)\}$ where v in D
 - Sample n points, q_1, \dots, q_n , uniformly from D . For each q_i find its nearest neighbor in $D - \{q_i\}$: $y_i = \min\{\text{dist}(q_i, v)\}$ where v in D and $v \neq q_i$
 - Calculate the Hopkins Statistic: $H = \frac{\sum_{i=1}^n y_i}{\sum_{i=1}^n x_i + \sum_{i=1}^n y_i}$
 - If D is uniformly distributed, $\sum x_i$ and $\sum y_i$ will be close to each other and H is close to 0.5. If D is highly skewed, H is close to 0

33

Determine the Number of Clusters

- Empirical method
 - # of clusters $\approx \sqrt{n/2}$ for a dataset of n points
- Elbow method
 - Use the turning point in the curve of sum of within cluster variance w.r.t the # of clusters
- Cross validation method
 - Divide a given data set into m parts
 - Use $m - 1$ parts to obtain a clustering model
 - Use the remaining part to test the quality of the clustering
 - E.g., For each point in the test set, find the closest centroid, and use the sum of squared distance between all points in the test set and the closest centroids to measure how well the model fits the test set
 - For any $k > 0$, repeat it m times, compare the overall quality measure w.r.t. different k 's, and find # of clusters that fits the data the best



34

Evaluating Clusters: k-means

- Sum of Squared Error (SSE)
 - For each point, the error is the distance to the nearest cluster
 - 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 representative point for cluster C_i
 - can show that m_i corresponds to the center (mean) of the cluster
 - Given two sets of clusters, we prefer the one with the smallest error
 - One easy way to reduce SSE is to increase K , the number of clusters
 - A good clustering with smaller K can have a lower SSE than a poor clustering with higher K

Measuring Clustering Quality

- Two methods: extrinsic vs. intrinsic
- **Extrinsic**: supervised, i.e., the ground truth is available
 - Compare a clustering against the ground truth using certain clustering quality measure
 - Ex. BCubed precision and recall metrics
- **Intrinsic**: unsupervised, i.e., the ground truth is unavailable
 - Evaluate the goodness of a clustering by considering how well the clusters are separated, and how compact the clusters are
 - Ex. Silhouette coefficient

36

Measuring Clustering Quality: Extrinsic Methods

- Clustering quality measure: $Q(C, C_g)$, for a clustering C given the ground truth C_g .
- Q is good if it satisfies the following 4 essential criteria
 - Cluster homogeneity: the purer, the better
 - Cluster completeness: should assign objects belong to the same category in the ground truth to the same cluster
 - Rag bag: putting a heterogeneous object into a pure cluster should be penalized more than putting it into a *rag bag* (i.e., “miscellaneous” or “other” category)
 - Small cluster preservation: splitting a small category into pieces is more harmful than splitting a large category into pieces

37

Final Comment on Cluster Validity

“The validation of clustering structures is the most difficult and frustrating part of cluster analysis.

Without a strong effort in this direction, cluster analysis will remain a black art accessible only to those true believers who have experience and great courage.”

Algorithms for Clustering Data, Jain and Dubes

Beyond k-means...

39

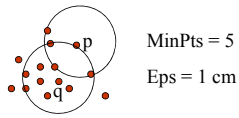
Density-Based Clustering Methods

- Clustering based on **density** (local cluster criterion), such as density-connected points
- Major features:
 - Discover clusters of arbitrary shape
 - Handle noise
 - One scan
 - Need density parameters as termination condition
- Several interesting studies:
 - **DBSCAN**: Ester, et al. (KDD'96)
 - **OPTICS**: Ankerst, et al (SIGMOD'99).
 - **DENCLUE**: Hinneburg & D. Keim (KDD'98)
 - **CLIQUE**: Agrawal, et al. (SIGMOD'98) (more grid-based)

40

Density-Based Clustering: Basic Concepts

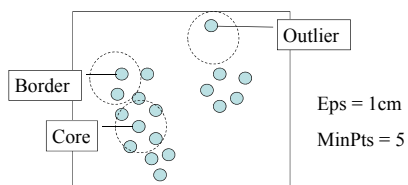
- Two parameters:
 - **Eps**: Maximum radius of the neighbourhood
 - **MinPts**: Minimum number of points in an Eps-neighbourhood of that point
- $N_{Eps}(p)$: $\{q \text{ belongs to } D \mid \text{dist}(p,q) \leq Eps\}$
- **Directly density-reachable**: A point p is directly density-reachable from a point q w.r.t. $Eps, MinPts$ if
 - p belongs to $N_{Eps}(q)$
 - core point condition:
 $|N_{Eps}(q)| \geq MinPts$



41

DBSCAN: Density-Based Spatial Clustering of Applications with Noise

- Relies on a *density-based* notion of cluster: A *cluster* is defined as a maximal set of density-connected points
- Discovers clusters of arbitrary shape in databases with noise



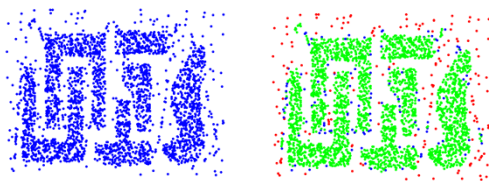
42

DBSCAN: The Algorithm

- Arbitrary select a point p
- Retrieve all points density-reachable from p w.r.t. Eps and $MinPts$
- If p is a core point, a cluster is formed
- If p is a border point, no points are density-reachable from p and DBSCAN visits the next point of the database
- Continue the process until all of the points have been processed

43

DBSCAN: Core, Border and Noise Points

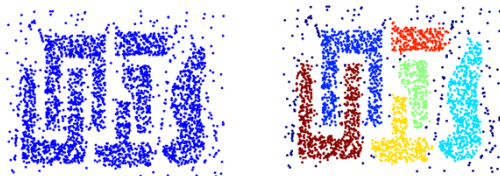


Original Points

Point types: core,
border and noise

$Eps = 10$, $MinPts = 4$

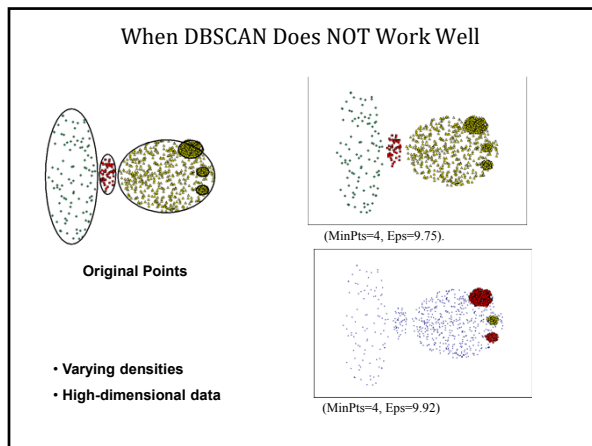
When DBSCAN Works Well

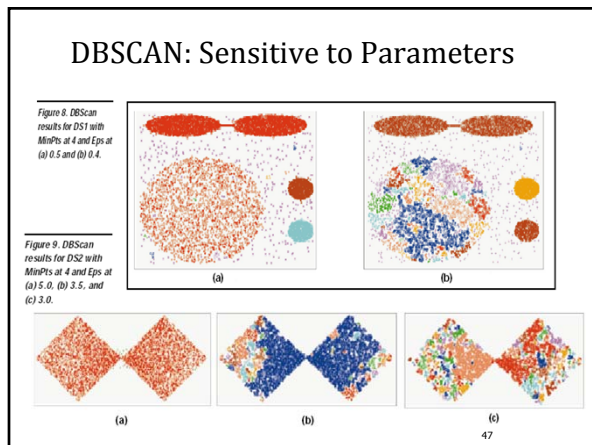


Original Points

Clusters

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






Fuzzy Set and Fuzzy Cluster

- Some applications may need for fuzzy or soft cluster assignment
 - Ex. An e-game could belong to both entertainment and software
- Methods: fuzzy clusters and probabilistic model-based clusters
- Fuzzy cluster: A fuzzy set $S: F_S: X \rightarrow [0, 1]$ (value between 0 and 1)
- Example: Popularity of cameras is defined as a fuzzy mapping

Camera	Sales (units)
A	50
B	1320
C	860
D	270

$$\text{Pop}(o) = \begin{cases} 1 & \text{if 1,000 or more units of } o \text{ are sold} \\ \frac{i}{1000} & \text{if } i \text{ (} i < 1000 \text{) units of } o \text{ are sold} \end{cases}$$

- Then, $A(0.05), B(1), C(0.86), D(0.27)$



48

Fuzzy (Soft) Clustering

- Example: Let cluster features be
 - C_1 : "digital camera" and "lens"
 - C_2 : "computer"
- Fuzzy clustering
 - k fuzzy clusters C_1, \dots, C_k , represented as a partition matrix $M = [w_{ij}]$
 - P1: for each object o_i and cluster C_j , $0 \leq w_{ij} \leq 1$ (fuzzy set)
 - P2: for each object o_i , $\sum_{j=1}^k w_{ij} = 1$ equal participation in the clustering
 - P3: for each cluster C_j , $0 < \sum_{i=1}^n w_{ij} < n$ ensures there is no empty cluster

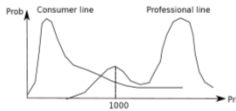
Review-id	Keywords
R_1	digital camera, lens
R_2	digital camera
R_3	lens
R_4	digital camera, lens, computer
R_5	computer, CPU
R_6	computer, computer game

$$M = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}$$

49

Probabilistic Model-Based Clustering

- Cluster analysis is to find hidden categories.
- A hidden category (i.e., *probabilistic cluster*) is a distribution over the data space, which can be mathematically represented using a probability density function (or distribution function).
- Ex. 2 categories for digital cameras sold
 - consumer line vs. professional line
 - density functions f_1, f_2 for C_1, C_2
 - obtained by probabilistic clustering
- A **mixture model** assumes that a set of observed objects is a mixture of instances from multiple probabilistic clusters, and conceptually each observed object is generated independently
- Our task**: infer a set of k probabilistic clusters that is **mostly likely** to generate D using the above data generation process



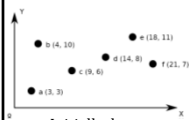
50

The EM (Expectation Maximization) Algorithm

- A framework to approach maximum likelihood or maximum a posteriori estimates of parameters in statistical models.
 - E-step** assigns objects to clusters according to the current fuzzy clustering or parameters of probabilistic clusters
 - M-step** finds the new clustering or parameters that maximize the sum of squared error (SSE) or the expected likelihood

51

Fuzzy Clustering Using the EM Algorithm



Iteration	E-step	M-step
1	$M^T = \begin{bmatrix} 1 & 0 & 0.48 & 0.42 & 0.41 & 0.47 \\ 0 & 1 & 0.52 & 0.58 & 0.59 & 0.53 \end{bmatrix}$	$c_1 = (8.47, 5.12)$ $c_2 = (10.42, 8.99)$
2	$M^T = \begin{bmatrix} 0.73 & 0.49 & 0.91 & 0.26 & 0.33 & 0.42 \\ 0.27 & 0.51 & 0.09 & 0.74 & 0.67 & 0.58 \end{bmatrix}$	$c_1 = (8.51, 6.11)$ $c_2 = (14.42, 8.69)$
3	$M^T = \begin{bmatrix} 0.80 & 0.76 & 0.99 & 0.02 & 0.14 & 0.23 \\ 0.20 & 0.24 & 0.01 & 0.98 & 0.86 & 0.77 \end{bmatrix}$	$c_1 = (6.40, 6.24)$ $c_2 = (16.55, 8.64)$

- Initially, let $c_1 = a$ and $c_2 = b$
- 1st E-step: assign o to c_i , w. wt = $\frac{\frac{1}{dist(o, c_1)^2}}{\frac{1}{dist(o, c_1)^2} + \frac{1}{dist(o, c_2)^2}} = \frac{dist(o, c_2)^2}{dist(o, c_1)^2 + dist(o, c_2)^2}$
 $w_{c, c_1} = \frac{41}{45+41} = 0.48$
- 1st M-step: recalculate the centroids according to the partition matrix, minimizing the sum of squared error (SSE)

$$c_1 = \frac{\sum_{\text{each point } o} w_{o, c_1}^2 o}{\sum_{\text{each point } o} w_{o, c_1}^2} = \frac{(1^2 \times 3 + 0^2 \times 4 + 0.48^2 \times 9 + 0.42^2 \times 14 + 0.41^2 \times 18 + 0.47^2 \times 21)}{(1^2 + 0^2 + 0.48^2 + 0.42^2 + 0.41^2 + 0.47^2)}$$

$$c_2 = \frac{\sum_{\text{each point } o} w_{o, c_2}^2 o}{\sum_{\text{each point } o} w_{o, c_2}^2} = \frac{(1^2 \times 3 + 0^2 \times 4 + 0.48^2 \times 6 + 0.42^2 \times 8 + 0.41^2 \times 11 + 0.47^2 \times 7)}{(1^2 + 0^2 + 0.48^2 + 0.42^2 + 0.41^2 + 0.47^2)}$$

$$= (8.47, 5.12)$$
- Iteratively calculate this until the cluster centers converge or the change is small enough

Clustering High-Dimensional Data

- Clustering high-dimensional data (How high is high-D in clustering?)
 - Many applications: text documents, DNA micro-array data
 - Major challenges:
 - Many irrelevant dimensions may mask clusters
 - Distance measure becomes meaningless—due to equi-distance
 - Clusters may exist only in some subspaces
- Methods
 - Subspace-clustering:** Search for clusters existing in subspaces of the given high dimensional data space
 - CLIQUE, ProClus, and bi-clustering approaches
 - Dimensionality reduction approaches:** Construct a much lower dimensional space and search for clusters there (may construct new dimensions by combining some dimensions in the original data)
 - Dimensionality reduction methods and spectral clustering

53

Traditional Distance Measures May Not Be Effective on High-D Data!

- Traditional distance measure could be dominated by noises in many dimensions
- Ex. Which pairs of customers are more similar?

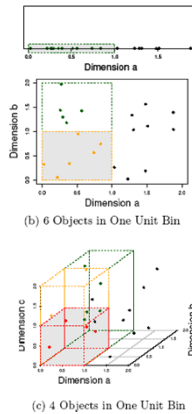
Customer	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}
Ada	1	0	0	0	0	0	0	0	0	0
Bob	0	0	0	0	0	0	0	0	0	1
Cathy	1	0	0	0	1	0	0	0	0	1

- By Euclidean distance, we get,
 $dist(Ada, Bob) = dist(Bob, Cathy) = dist(Ada, Cathy) = \sqrt{2}$
 - despite Ada and Cathy "look" more similar



The Curse of Dimensionality

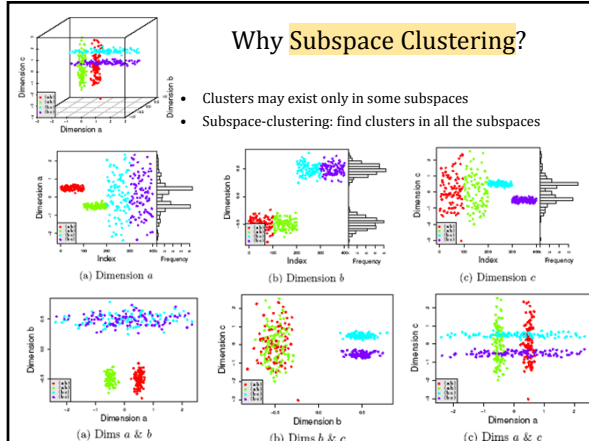
- Data in only one dimension is relatively packed
- Adding a dimension “stretch” the points across that dimension, making them further apart
- Adding more dimensions will make the points further apart
- High dimensional data is extremely sparse
- Distance measure may be meaningless



55

Why Subspace Clustering?

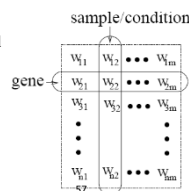
- Clusters may exist only in some subspaces
- Subspace-clustering: find clusters in all the subspaces



Subspace Clustering: Bi-Clustering Methods

- Bi-clustering: Cluster both objects and attributes simultaneously (treat objs and attrs in symmetric way)
- Four requirements:
 - Only a small set of objects participate in a cluster
 - A cluster only involves a small number of attributes
 - An object may participate in multiple clusters, or does not participate in any cluster at all
 - An attribute may be involved in multiple clusters, or is not involved in any cluster at all
- Ex. Clustering customers and products

	products			
customers	w_{11}	w_{12}	\dots	w_{1m}
	w_{21}	w_{22}	\dots	w_{2m}
	\dots	\dots	\dots	\dots
	w_{n1}	w_{n2}	\dots	w_{nm}



57

Types of Bi-clusters

- Let $A = \{a_1, \dots, a_n\}$ be a set of genes, $B = \{b_1, \dots, b_n\}$ a set of conditions
- A bi-cluster: A **submatrix** where genes and conditions follow some consistent **patterns**
- 4 types of bi-clusters (ideal cases)
 - Bi-clusters with constant values:
 - for any i in I and j in J , $e_{ij} = c$
 - Bi-clusters with constant values on **rows**:
 - $e_{ij} = c + \alpha_i$
 - Also, it can be constant values on **columns**
 - Bi-clusters with **coherent values** (aka. **pattern-based clusters**)
 - $e_{ij} = c + \alpha_i + \beta_j$
 - Bi-clusters with **coherent evolutions on rows**
 - $e_{ij} (e_{i1j1} - e_{i1j2})(e_{i2j1} - e_{i2j2}) \geq 0$
 - i.e., only interested in the up- or down- regulated changes across genes or conditions without constraining on the exact values

10	10	10	10	10
20	20	20	20	20
50	50	50	50	50
0	0	0	0	0

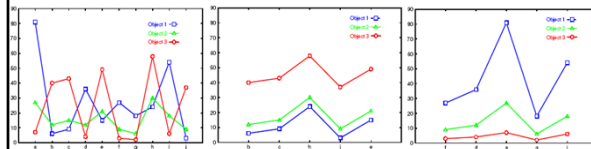
10	50	30	70	20
20	60	40	80	30
50	90	70	110	60
0	40	20	60	10

10	50	30	70	20
20	100	50	1000	30
50	100	90	120	80
0	80	20	100	10

58

Bi-Clustering for Micro-Array Data Analysis

- Left figure: Micro-array "raw" data shows 3 genes and their values in a **multi-D** space: Difficult to find their patterns
- Right two: Some **subsets of dimensions** form nice **shift** and **scaling** patterns
- No globally defined similarity/distance measure
- Clusters may not be exclusive
 - An object can appear in multiple clusters



Summary

- Cluster analysis** (unsupervised learning) groups objects based on their **similarity** and has wide applications
- Measure of similarity computed for **various types of data**
- Measure of similarity crucial to success
- Quality of clustering results evaluated in various ways
- Some Issues:
 - Shapes and sizes of clusters
 - Setting parameters
 - Scalability and high dimensionality
 - Time to construct the clusters
 - Overlapping clusters and other constraints
 - Interpretability

60

Next

FINDING OUTLIERS

61
