COEN281 -- Introduction to Pattern Recognition and Data Mining

Lecture 13: Clustering

Instructor: Dr. Giovanni Seni
GSeni@scu.edu

Department of Computer Engineering
Santa Clara University

Fall/18

Syllabus

Week 1	Introduction; R (Ch.1)
Week 2	Bayesian Decision Theory (Ch.2; DHS: 2.1-2.6, 2.9) Parameter Estimation (DHS: 3.1-3.4)
Week 3	Linear Discriminant Functions (Ch.3&4; DHS: 3.8.2, 5.1-5.8) Regularization (Ch.6; SE: Ch.3)
Week 4	Neural Networks (DHS: 6.1-6.6, 6.8);
Week 5	Support Vector Machines (Ch.9)
Week 6	Decision Trees (Ch. 8.1; DHS: 8.3; Ch 2 SE)
Week 7	Ensemble Methods (Ch. 8.2; SE: Ch 4, 5)
Week 8	**Clustering** (Ch. 10; DHS: 10.6, 10.7) Clustering (DHS: 10.9); How many clusters are there? (DHS: 10.10)
Week 9	Non-metric: Association Rules Collaborative Filtering
Week 10	Text Retrieval; Other topics

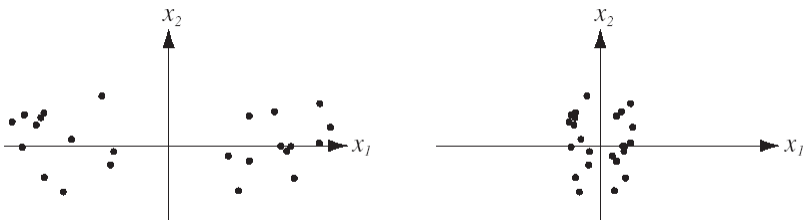
Overview

- Introduction
  - What is Cluster Analysis?
- Distance (and similarity) notion
  - Measures for numerical data
  - Measures for binary data
  - Ordinal, nominal, and mixed data
- Partition-based Clustering
  - Criterion functions
  - *K*-means
  - Unknown *K*

Introduction

What is Cluster Analysis?

- What goes with what?



- Partitioning a data set into groups so that
  - the points in one group are *similar* to each other, and
  - are as *different* as possible from points in other groups

Introduction
What is Cluster Analysis?

- Hinges on a notion of *distance*
- Unsupervised procedure
  - Use unlabeled samples
- Common applications
  - *Segmentation* – partition the data in a way that is "convenient"
    - E.g., shirt dimensions for S/M/L/XL sizes
  - *Exploratory Data Analysis* – gain insight into the nature or structure of the data
    - E.g., do Napa red wines fall into distinct subclasses?

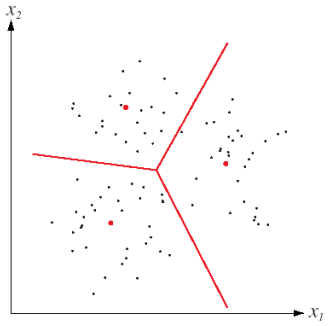
Introduction
What is Cluster Analysis?

- Hinges on a notion of *distance*
- Unsupervised procedure
  - Use unlabeled samples
- Common applications
  - *Segmentation* – partition the data in a way that is "convenient"
    - E.g., shirt dimensions for S/M/L/XL sizes
  - *Exploratory Data Analysis* – gain insight into the nature or structure of the data
    - E.g., do gene expression sample data fall into distinct subclasses?
      do click histories follow similar search patters?

Introduction
Segmentation Example

- Credit card users

$$\mathbf{x} = \begin{bmatrix} \text{type of purchases} \\ \text{total money spent} \\ \text{frequency of card use} \\ \text{locations of use} \\ \dots \end{bmatrix} \Rightarrow$$

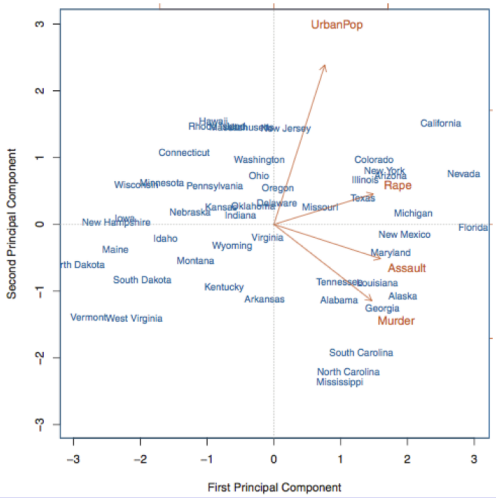


  - Targeted promotional material

- Chain stores – $\mathbf{x}$ = [social neighborhood, size, staff numbers,…]^t
  - Identify similar stores
  - Examine distribution of variables within each group

Introduction
Data Exploration Example

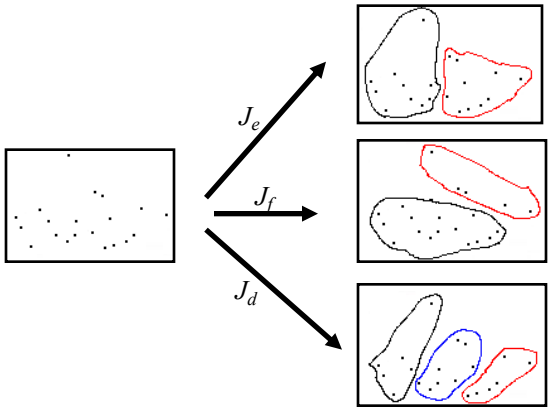
- USArrests data visualization via PCA



Introduction

What is a "good" cluster?

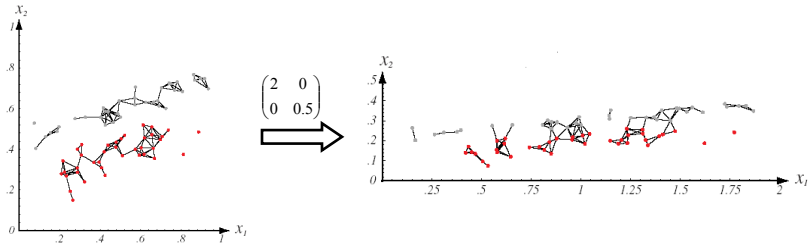
- No direct notion of generalization to a test data set
  - The validity of a clustering is often in the eye of the beholder



Introduction

What is a "good" cluster? (2)

- Invariant to transformations natural to the problem
- Scaling of variables matters
  - E.g., minimum distance method



  - Some variables measure same thing -- e.g., currency, weight, length… better put them in same unit than to re-scale

Introduction

Types of Cluster Analysis Algorithms

- Partition-based
  - Find the optimal partition into a specified number of clusters
  - E.g., K-means
- Hierarchical
  - *Agglomerative* or *divisive* approach
- Density-based
  - Use probabilistic model for underlying clusters
  - E.g., mixture model $p(\mathbf{x}\,|\,\theta) = \sum_{k=1}^{K} p(\mathbf{x}\,|\,\omega_k, \theta_k) P(\omega_k)$

Distance Notion

Measures

- Distance vs. Similarity
  - $d_{ij} = s - s_{ij}$ where S is some notion of perfect similarity (e.g., $S=1$)

  i.e., distance often refers to a dissimilarity measure
- Typically:
  - i) $d_{ij} \geq 0$

    ii) $d_{ii} = 0$

    iii) $d_{ij} = d_{ji}$
  - *metric* if: $d_{ij} \leq d_{ik} + d_{kj}$
  - *ultra-metric* if: $d_{ij} \leq \max[d_{ik}, d_{kj}]$

Distance Notion

Measures for Numerical Data

- Euclidean Distance: $d_{ij} = \sqrt{(x_1^i - x_1^j)^2 + (x_2^i - x_2^j)^2 + \cdots + (x_d^i - x_d^j)^2}$
  - the shortest distance between two points
- Squared Euclidean: $d_{ij} = (x_1^i - x_1^j)^2 + (x_2^i - x_2^j)^2 + \cdots + (x_d^i - x_d^j)^2$
  - cheaper to compute (i.e., sqrt is a monotonic transformation)
- Manhattan Distance: $d_{ij} = \left|x_1^i - x_1^j\right| + \left|x_2^i - x_2^j\right| + \cdots + \left|x_d^i - x_d^j\right|$
  - even cheaper: Δx instead of the diagonal
  - use when distance needs to be discrete rather than continuous (e.g. 6.5 people might not be suitable for analysis!)

Distance Notion

Measures for Numerical Data (2)

- Camberra Metric: $d_{ij} = \frac{\left|x_1^i - x_1^j\right|}{\left|x_1^i\right| + \left|x_1^j\right|} + \frac{\left|x_2^i - x_2^j\right|}{\left|x_2^i\right| + \left|x_2^j\right|} + \cdots + \frac{\left|x_d^i - x_d^j\right|}{\left|x_d^i\right| + \left|x_d^j\right|}$
  - accounts for points relation to the 'origin'
  - meant for non-negative variables only
  - used as a metric for comparing ranked lists
- Correlation Coefficient: $\rho_{ij} = 1 - \left| \frac{\sum_{k=1}^{d} (x_k^i - \mu_{x^i})(x_k^j - \mu_{x^j})}{\left(\sum_{k=1}^{d} (x_k^i - \mu_{x^i})^2 \sum_{k=1}^{d} (x_k^j - \mu_{x^j})^2\right)^{1/2}} \right|$
  - appropriate if the input attributes have vastly different scales
  - 1: perfect similarity, 0: there is no similarity

Distance Notion

Species Abundance Paradox

- Example from ecology: abundance of 3 species a 3 sites

	Species 1	Species 2	Species 3
Site s_1	0	1	1
Site s_2	1	0	0
Site s_3	0	4	8

- Dissimilarity values

	$d(s_1, s_2)$	$d(s_1, s_3)$	$d(s_2, s_3)$
Square Euclidean	3	58	81
Manhattan	3	10	13
Camberra	3	1.378	3

$\Rightarrow$ The choice of an appropriate measure depends on nature of data

Distance Notion

Measures for Binary Data

- Hamming Distance: $d_{ij} = \#\left\{k \mid x_k^i \neq x_k^j\right\}$
  - E.g., **1011101** and **1001001** $\Rightarrow d_{ij} = 2$

- Define

x^i \ x^j	1	0
1	a	b
0	c	d

Name	Dissimilarity	Similarity
Simple Matching	$\frac{b+c}{p}$	$\frac{a+d}{p}$
Jaccard	$\frac{b+c}{a+b+c}$	$\frac{a}{a+b+c}$
Russel Rao	$\frac{b+c+d}{p}$	$\frac{a}{p}$
Dice	$\frac{b+c}{2a+b+c}$	$\frac{2a}{2a+b+c}$

  - SimpleMatching counts 0's, while Jaccard ignores them
  - Dice weights more matched 1's

Distance Notion

Measures for Ordinal & Nominal Data

- *Ordinal* – numerical values but only trust whether $x_k^i < x_k^i$
  - Rank order and normalize: lowest-rank is 0 and highest-rank is 1
  - Conversion to a sequence of binary attributes
    - If feature A has 3 states a_1, a_2, a_3 with $a_1 < a_2 < a_3$ we replace A with three binary features

	B_1	B_2	B_3
a_1	1	0	0
a_2	1	1	0
a_3	1	1	1

- *Nominal* –
  - $d_{ij} = k/d$ - k is # of features in which x_i and x_j have different states
  - Conversion to a sequence of binary attributes

Distance Notion

Measures for Mixed Data

- Divide features into groups: A_n, A_b, A_r, A_o
  - Choose an appropriate dissimilarity measure for each type of feature: d_n, d_b, d_r, d_o
  - Define

$$d_{ij} = d(x^i, x^j) = w_n d_n(x^i, x^j) + w_b d_b(x^i, x^j) + w_r d_r(x^i, x^j) + w_o d_o(x^i, x^j)$$

for some appropriately chosen weight factors

Partition-based Clustering

Overview

- *Task* – partition $D=\{\boldsymbol{x}^1,\ldots,\boldsymbol{x}^n\}$ into k disjoint sets of points $C=\{C_1,\ldots,C_K\}$ such that the points within each set C_k are as "homogeneous" as possible
- *Score function* – captures notion of homogeneity
  e.g., sum of distances between $\boldsymbol{x}^i$ and "centroid" of cluster to which it is assigned
- *Search method* – iterative improvement heuristic
  possible allocations of n objects into K groups: K^n

Partition-based Clustering

Score Functions

- $J(C) = f(wc(C), bc(C))$
  - $wc(C)$ – within cluster variation
    - How compact or tight the clusters are
  - $bc(C)$ – between cluster variation
    - How far from each other clusters are
- Sum-of-Squared-Distances Criterion
  If taking means make sense, $\boldsymbol{\mu}_k = \frac{1}{n_k}\sum_{\mathbf{x}\in C_k}\mathbf{x}$

$$wc(C)=\sum_{k=1}^{K} wc(C_k)=\sum_{k=1}^{K}\sum_{\mathbf{x}\in C_k} d\left(\mathbf{x},\boldsymbol{\mu}_k\right)^2 \qquad bc(C)=\sum_{1\le j<k\le K} d\left(\boldsymbol{\mu}_j,\boldsymbol{\mu}_k\right)^2$$

Partition-based Clustering

Basic Algorithm – *K*-means

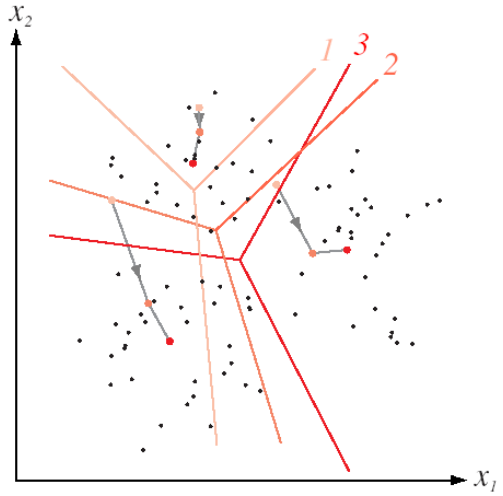
- Greedy approach

Initialize n, K, $\boldsymbol{\mu}_1$, $\boldsymbol{\mu}_2$,…, $\boldsymbol{\mu}_K$
do
  // Step 1: form clusters
  for $k=1,\ldots,K$ do
    $C_k = \{\boldsymbol{x} \in D \mid d(\boldsymbol{\mu}_k, \boldsymbol{x}) \leq d(\boldsymbol{\mu}_j, \boldsymbol{x}) \;\; \forall j \neq k\}$
  end
  // Step 2: compute new cluster centers
  for $k=1,\ldots,K$ do
    $\boldsymbol{\mu}_k$ = *vector mean of the points* C_k
  end
until *no change in* $\boldsymbol{\mu}_k$

Partition-based Clustering

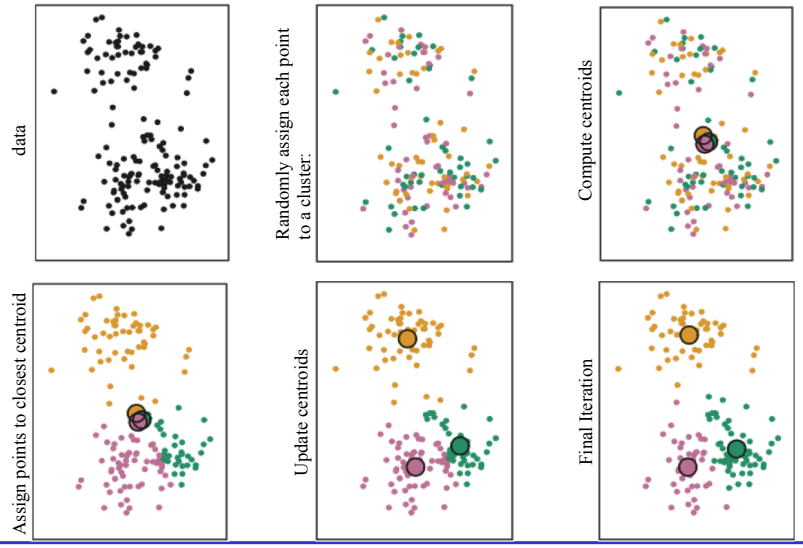
K-means – Example

- 2D data



Partition-based Clustering

K-means – Example (2)



Partition-based Clustering

K-means– Properties

- Complexity $O(KnI)$
  - I : number of iterations. In practice, $I << n$
- Converges to local minima of $J(C)$
  - different initial centers (seeds) can lead to different solution
- Bias towards
  - Spherical clusters
  - Equal-sized clusters

$J(C)$ large

$J(C)$ small

Partition-based Clustering

Scatter Criteria

- *Within-cluster* scatter matrix

$$\mathbf{S}_w = \sum_{k=1}^{K} \mathbf{S}_k \qquad \text{where} \qquad \mathbf{S}_k = \sum_{\mathbf{x} \in C_k} (\mathbf{x} - \boldsymbol{\mu}_k)(\mathbf{x} - \boldsymbol{\mu}_k)^t$$

- *Between-cluster* scatter matrix

$$\mathbf{S}_B = \sum_{k=1}^{K} n_k (\boldsymbol{\mu}_k - \boldsymbol{\mu})(\boldsymbol{\mu}_k - \boldsymbol{\mu})^t$$

- *Total* scatter matrix

$$\mathbf{S}_T = \sum_{\mathbf{x} \in D} (\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})^t = \mathbf{S}_W + \mathbf{S}_B$$

  - $\boldsymbol{S}_T$ does not depend on the partition $\Rightarrow$ there is an exchange between $\boldsymbol{S}_B$ and $\boldsymbol{S}_W$ matrices: $\boldsymbol{S}_B$ goes up as $\boldsymbol{S}_W$ goes down
    - This is fortunate: by minimizing $\boldsymbol{S}_W$ we will also tend to maximize $\boldsymbol{S}_B$

Partition-based Clustering

Scatter Criteria (2)

- *Trace criterion* – $wc(C) = tr[\mathbf{S}_W] = \sum_{k=1}^{K} tr[\mathbf{S}_k]$
  - Measures the square of the scattering radius
  - Note that $tr[\mathbf{S}_W] = \sum_{k=1}^{K} \sum_{\mathbf{x} \in C_k} \|\mathbf{x} - \boldsymbol{\mu}_k\|^2$
  - Because $tr[M] = \sum_{i=1}^{d} \lambda_i$
    - Favors spherical clusters
    - Sensitive to scaling – i.e., alter units in a feature and a different cluster structure may result
  - Tendency to produce roughly equal groups

Partition-based Clustering

Scatter Criteria (3)

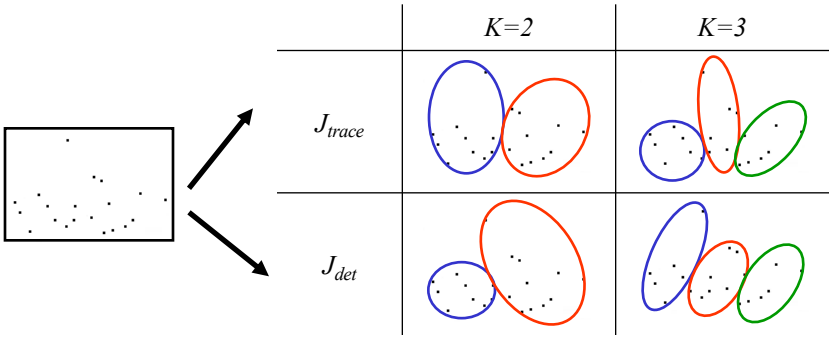
- *Determinant criterion* – $wc(C) = |\mathbf{S}_W| = \sum_{k=1}^{K} |\mathbf{S}_k|$
  - Measures the square of the scattering volume
  - Because $|M| = \prod_{i=1}^{d} \lambda_i$
    - Allows elongated clusters
    - Partition won't change if axes are scaled

      $\Rightarrow$ preferred under conditions where there may be unknown or irrelevant linear transformation of the data
  - Also favors equal-seized groups

Partition-based Clustering

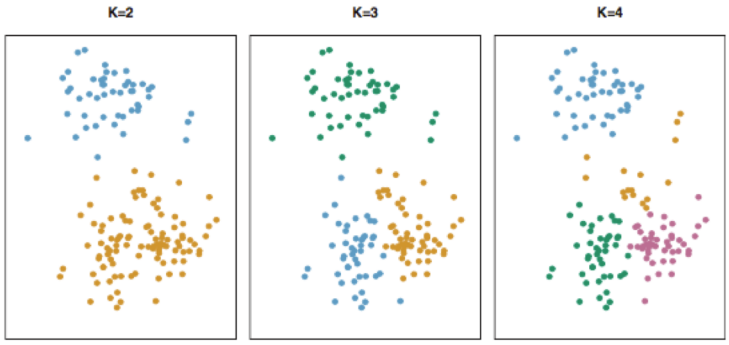
Scatter Criteria (4)

- Differences between *J(C)* become less pronounced for large number of clusters



Partition-based Clustering

Unknown K



- How to choose?

Partition-based Clustering

Unknown K (2)

- Repeat clustering procedure for $K=1, 2, \ldots$ and see how the criterion function J changes
  - Typically, J decreases monotonically
    - Rapidly until $K = \hat{K}$, thereafter more slowly until it reaches zero