CSE 547/
STAT 548

4/7/25

Lecture 3

THANKS for sitting in rows 1 – 6 !!

OH – TBA
L11 – 1, 2, 3
clustering

Clustering · definitions

KDE

Lecture II – Clustering – ~~Part I: Parametric clustering~~ definitions

Marina Meilă
mmp@stat.washington.edu

Department of Statistics
University of Washington

CSE 547/STAT 548
Spring 2025

1. Paradigms for clustering

2. Parametric clustering algorithms (K given)
   - Cost based / hard clustering
   - Model based / soft clustering
   - Outliers

Reading MMDS Ch.: 7.3 K-means HTF Ch.:14.3, Murphy Ch.: 11.[1], 11.2.1-3, 11.3, Ch 25

What is clustering? Problem and Notation

first take

- **Informal definition** **Clustering** = Finding groups in data
- **Notation**

	$\mathcal{D}$	=	$\{x_1, x_2, \ldots x_n\}$ a **data set**
	n	=	number of **data points**
	K	=	number of **clusters** ($K << n$)
	Δ	=	$\{C_1, C_2, \ldots, C_K\}$ a partition of $\mathcal{D}$ into disjoint subsets
	$k(i)$	=	the **label** of point i
	$L(\Delta)$	=	cost (loss) of Δ (to be minimized)

'points

C_k = a cluster

- **Second informal definition** **Clustering** = given n **data points**, separate them into K **clusters**
- Hard vs. soft clusterings
  - **Hard** clustering Δ: an item belongs to only 1 cluster
  - **Soft** clustering $\gamma = \{\gamma_{ki}\}_{k=1:K}^{i=1:n}$

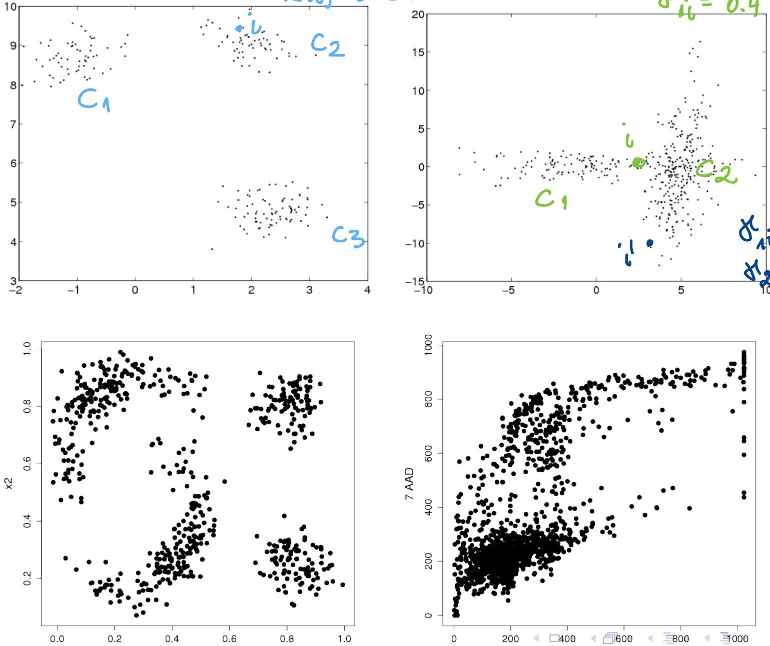
  γ_{ki} = the **degree of membership** of point i to cluster k

$$\sum_k \gamma_{ki} = 1 \quad \text{for all } i$$

(usually associated with a probabilistic model)

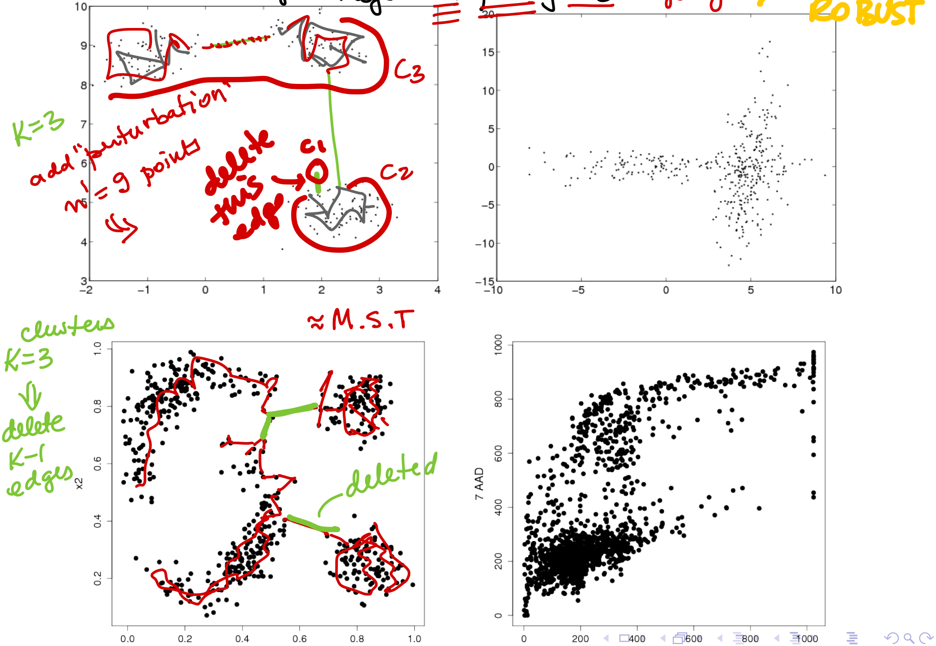
$k(i)=2 \iff x_i \in C_2$

$\gamma_{2i} = 0.6$
$\gamma_{1i} = 0.4$



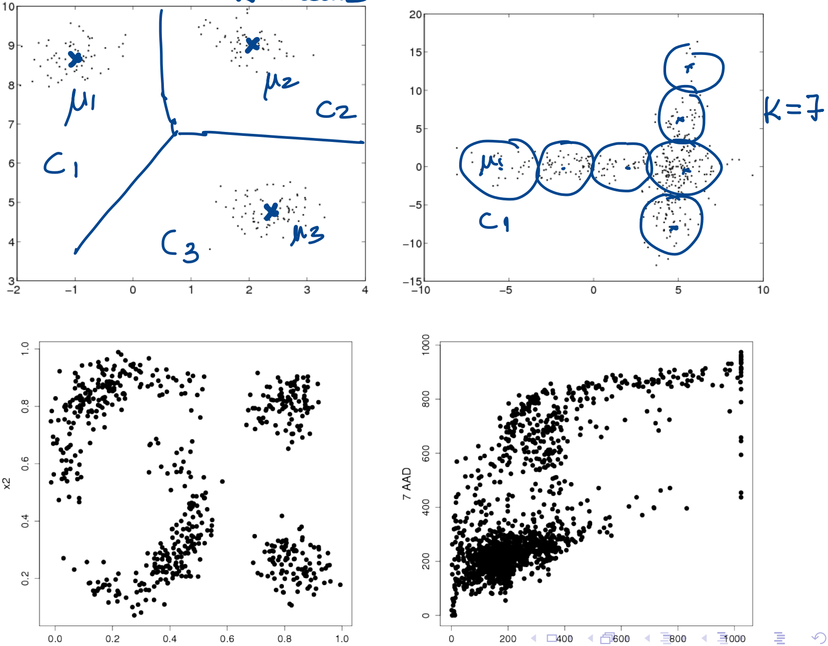
$\gamma_{1i'} \approx 10^{-3}$
$\gamma_{2i'} = 1-10^{-3}$

single linkage ~ Min Spanning Tree ← greedy ⇒ NOT ROBUST



Loss based

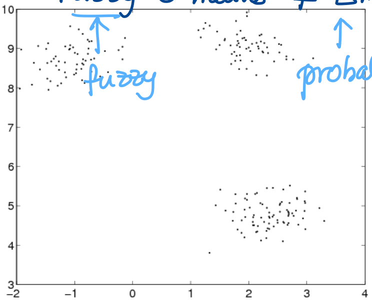
k-means



Fuzzy c-means ≠ EM

fuzzy

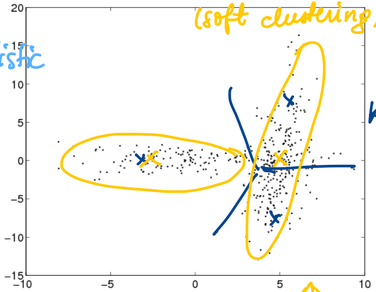
probabilistic



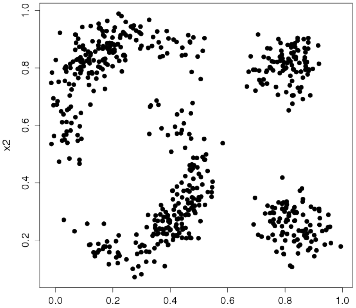
K=2 Mixture model (soft clustering)

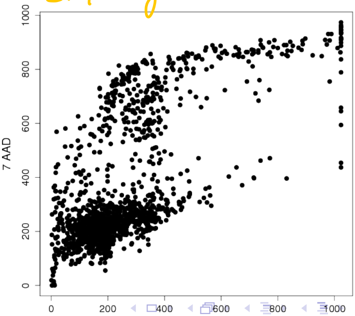
K=3

K-means



EM algorithm





Paradigms

Depend on type of data, type of clustering, type of cost (probabilistic or not), and constraints (about K, shape of clusters)

- Data = vectors $\{x_i\}$ in $\mathbb{R}^d$

Parametric (K known)	Cost based [hard]
	Model based [soft]
Non-parametric (K determined by algorithm)	Dirichlet process mixtures [soft]
	Information bottleneck [soft]
	Modes of distribution [hard]
	Gaussian blurring mean shift? [hard]

K-means, single linkage, min diam.

some constant

$\approx K c$ parameters

→ "complexity" ↗ with n

~~K~~ ↗ with n

K makes no sense

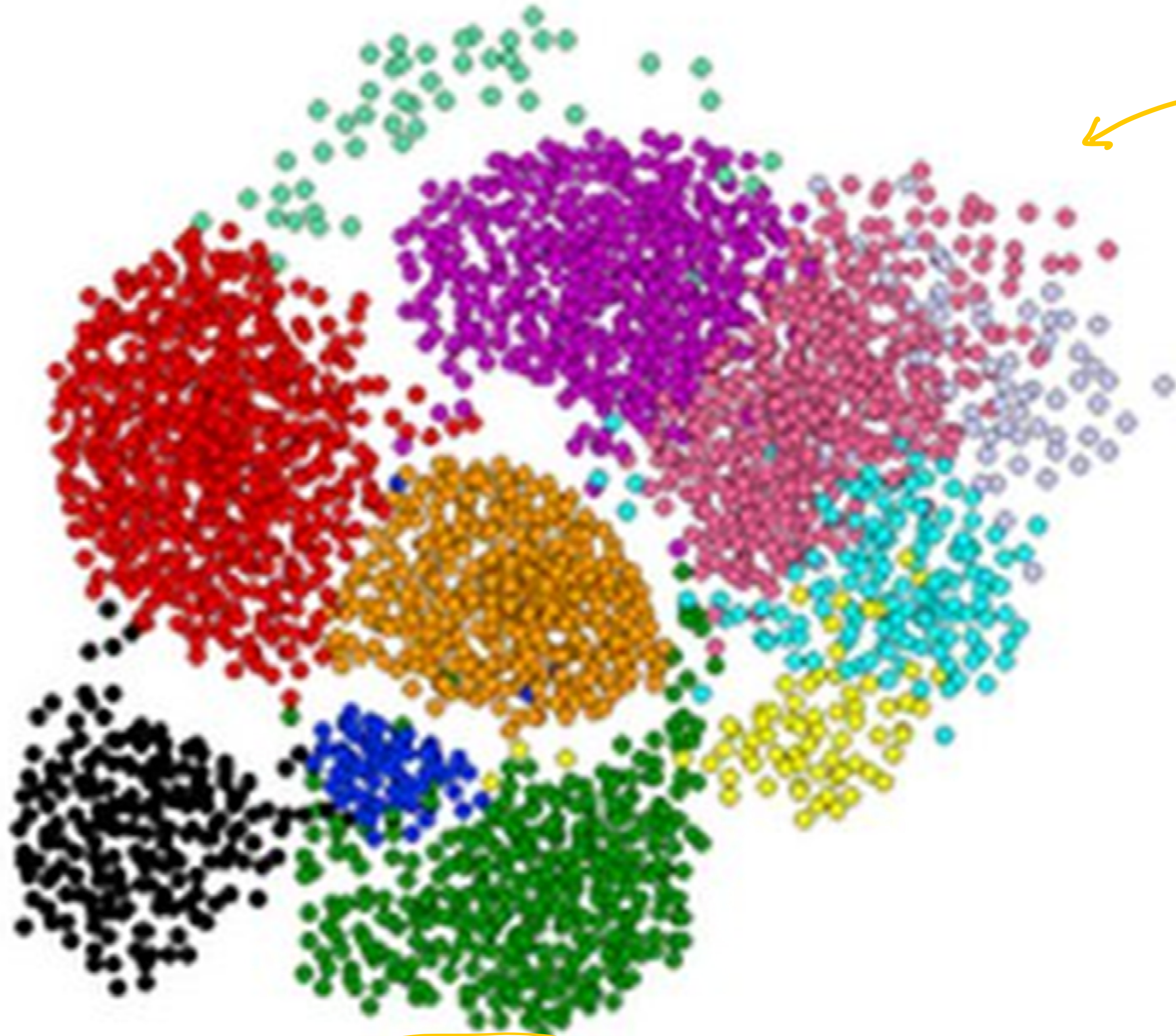
- Data = similarities between pairs of points $[S_{ij}]_{i,j=1:n}$, $S_{ij} = S_{ji} \geq 0$ **Similarity based clustering**

Graph partitioning	spectral clustering [hard, K fixed, cost based]
	typical cuts [hard non-parametric, cost based]
Affinity propagation	[hard/soft non-parametric]

Classification vs Clustering

	Classification	**Clustering**
Cost (or Loss) L	Expectd error	many! (probabilistic or not)
	Supervised	Unsupervised
Generalization	Performance on new data is what matters	Performance on current data is what matters
K	Known	Unknown
"Goal"	Prediction	Exploration Lots of data to explore!
Stage of field	Mature	Still young

Clustering is a hard problem!



MNIST
w.h.p

Why is it hard?

- Clustering in two dimensions looks easy
- Clustering small amounts of data looks easy
- And in most cases, looks are not deceiving

- Many applications involve not 2, but 10 or 10,000 dimensions
- **High-dimensional spaces look different:** Almost all pairs of points are at about the same distance

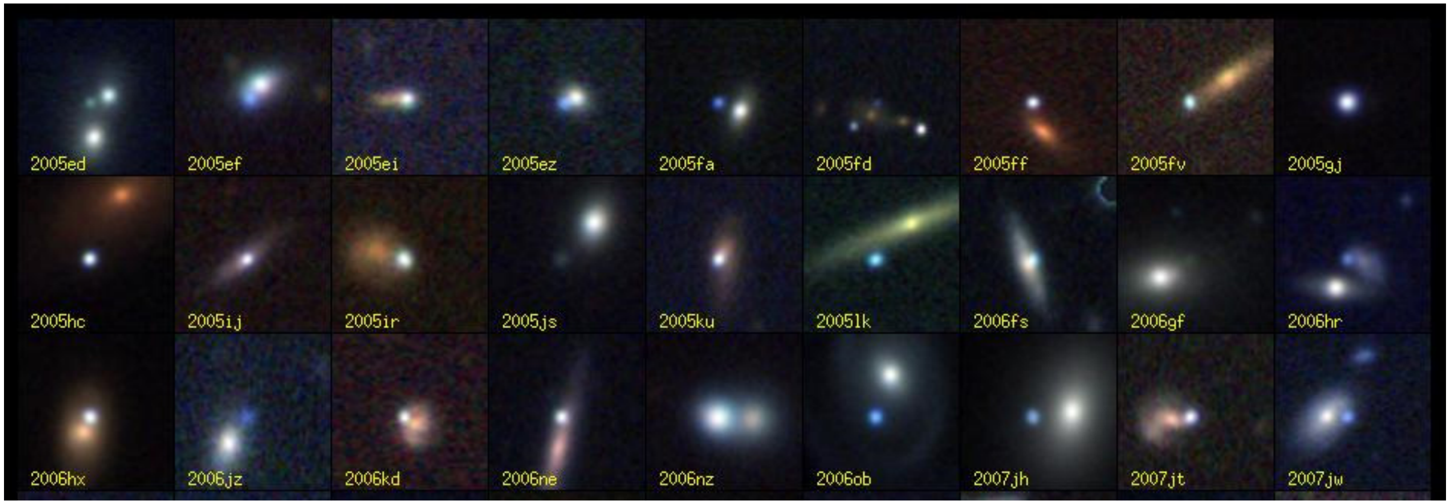
HARD

- need special distance
- loose info? by reducing dim

→ how to verify Δ is good ??

Clustering Problem: Galaxies

- **A catalog of 2 billion "sky objects" represents objects by their radiation in 7 dimensions (frequency bands)**
- **Problem: Cluster into similar objects, e.g., galaxies, nearby stars, quasars, etc.**
- **Sloan Digital Sky Survey**



Clustering Problem: Music CDs

- **Intuitively: Music divides into categories, and customers prefer a few categories**
  - But what are categories really?

- Represent a CD by a set of customers who bought it:

$j = 1:D$

$i = 1:n$

$x_{ij} = 1$ if j listens to i

$x_i = [x_{i1} \cdots x_{iD}] \in \{0,1\}^D$

- Similar CDs have similar sets of customers, and vice-versa

Clustering Problem: Music CDs

Space of all CDs:

- Think of a space with one dim. for each customer
  - Values in a dimension may be 0 or 1 only
  - A CD is a point in this space $(x_1, x_2,\ldots, x_k)$, where $x_i = 1$ iff the i^{th} customer bought the CD
- For Amazon, the dimension is tens of millions

$D \approx 10^7$

- **Task:** Find clusters of similar CDs

Clustering Problem: Documents

$x_{ij} = 1$ if word $j \in$ document i

Finding topics:

- Represent a document by a vector $(x_1, x_2, \ldots, x_k)$, where x_i = 1 iff the i th word (in some order) appears in the document
  - It actually doesn't matter if k is infinite; i.e., we don't limit the set of words
- **Documents with similar sets of words may be about the same topic**

Lecture II – Clustering – Part II: Non-parametric clustering

Marina Meilă
mmp@stat.washington.edu

Department of Statistics
University of Washington

CSE 547/STAT 548
Winter 2022

1. Paradigms for clustering

2. Methods based on non-parametric density estimation

3. Model-based: Dirichlet process mixture models

Methods based on non-parametric density estimation

Kernel density estimation

Idea The clusters are the isolated peaks in the (empirical) data density

- group points by the peak they are under
- some outliers possible
- $K = 1$ possible(no clusters)
- shape and number of clusters K determined by algorithm
- **structural parameters**
  - **smoothness** of the **density estimate**
  - what is a peak

• Kernel regression

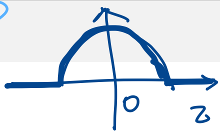
• ... Gaussian processes

... DPM ...

Algorithms

- peak finding algorithms Mean-shift algorithms
- level sets based algorithms
  - Nugent-Stuetzle, Support Vector clustering
- Information Bottleneck **?**

Kernel density estimation



Input
- data $\mathcal{D} \subseteq \mathbb{R}^d$
- **Kernel** function $K(z)$
- parameter **kernel width** h (is a **smoothness parameter**)

Output $f(x)$ a **probability density** over $\mathbb{R}^d$

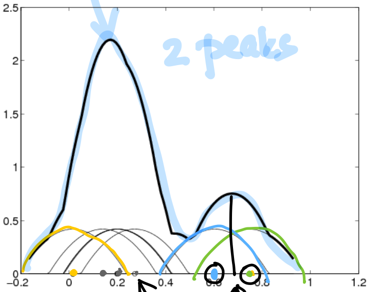
$$f(x) = \frac{1}{nh^d}\sum_{i=1}^{n} K\left(\frac{x-x_i}{h}\right)$$

= avg (bumps)

h = width (bandwidth)

density

Kernel density Estimator (KDE)



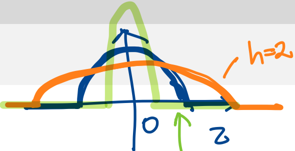
n=6

- f is sum of Gaussians centered on each x_i
- f is smoother (less variation) if h larger
- **caveat:** dimension d can't be too large

x

too far from x

Kernel density estimation



Input
- data $\mathcal{D} \subseteq \mathbb{R}^d$
- **Kernel** function $K(z)$
- parameter **kernel width** h (is a **smoothness parameter**)

Output $f(x)$ a **probability density** over $\mathbb{R}^d$

data

$$f_{h,K()} \qquad f(x) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$$

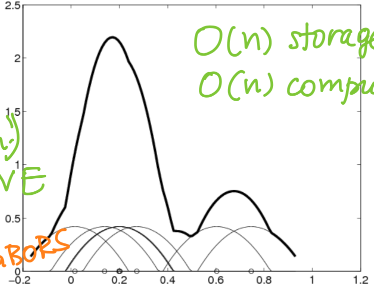
$$\int_{\mathbb{R}^d} K(z)\,dz = 1$$

$$\int_{\mathbb{R}^d} K\left(\frac{z}{h}\right) \cdot \frac{1}{h^d} = 1$$

(by chg. of var.)

- memory-based estimator (depens on $x_{1:n}$) VERY EXPENSIVE
- accelerated by FINDING NEIGBORS

O(n) storage
O(n) computation



d=1

- f is sum of Gaussians centered on each x_i
- f is smoother (less variation) if h larger
- **caveat:** dimension d can't be too large

The kernel function

- Example $K(z) = \frac{1}{(2\pi)^{d/2}} e^{-||z||^2/2}$, $z \in \mathbb{R}^d$ is the Gaussian kernel
- In general
  - $K()$ should represent a density on $\mathbb{R}^d$, i.e $K(z) \geq 0$ for all z and $\int K(z)dz = 1$
  - $K()$ symmetric around 0, decreasing with $||z||$
- In our case, K must be differentiable

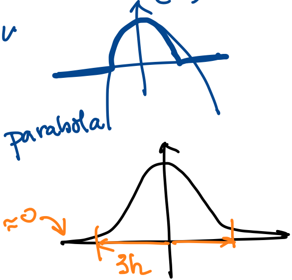
• choice of K not important (statistically)

Gaussian $K(z) = \frac{1}{\sigma^d(\sqrt{2\pi})^d} e^{-\frac{z^2}{2}}$

$f_h(x) = \frac{1}{n h^d} \sum_{i \in} K\left(\frac{x - x^i}{h}\right)$ ← Epanechnikov

Neighbors of x

$= 0$ if $||x - x^i|| \geq h\sqrt{5}$



expensive

improved by f. neighbors

don't know "shape"

adapts to shape of true f

$n \to \infty$

$h \searrow$ with n