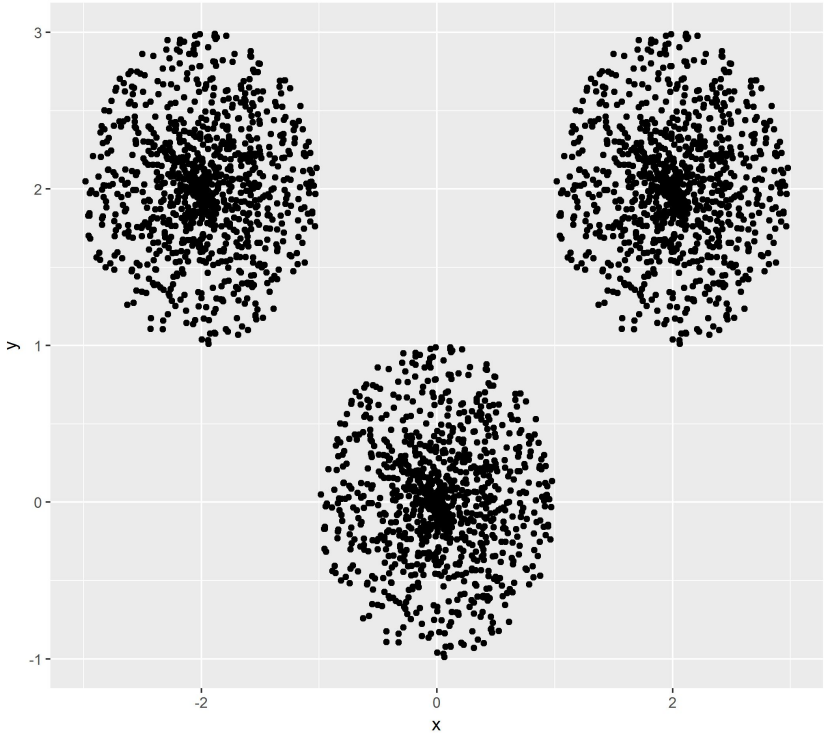
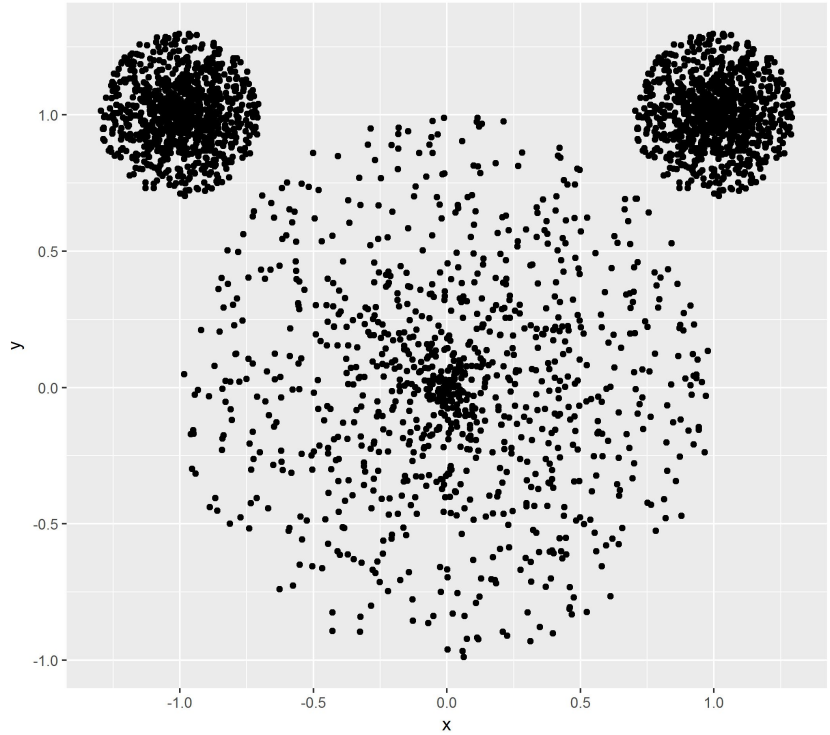
Clustering
(Datos Extensos = High dim. Data?)

Federico Zertuche
DIDE, UTA

What is a cluster?

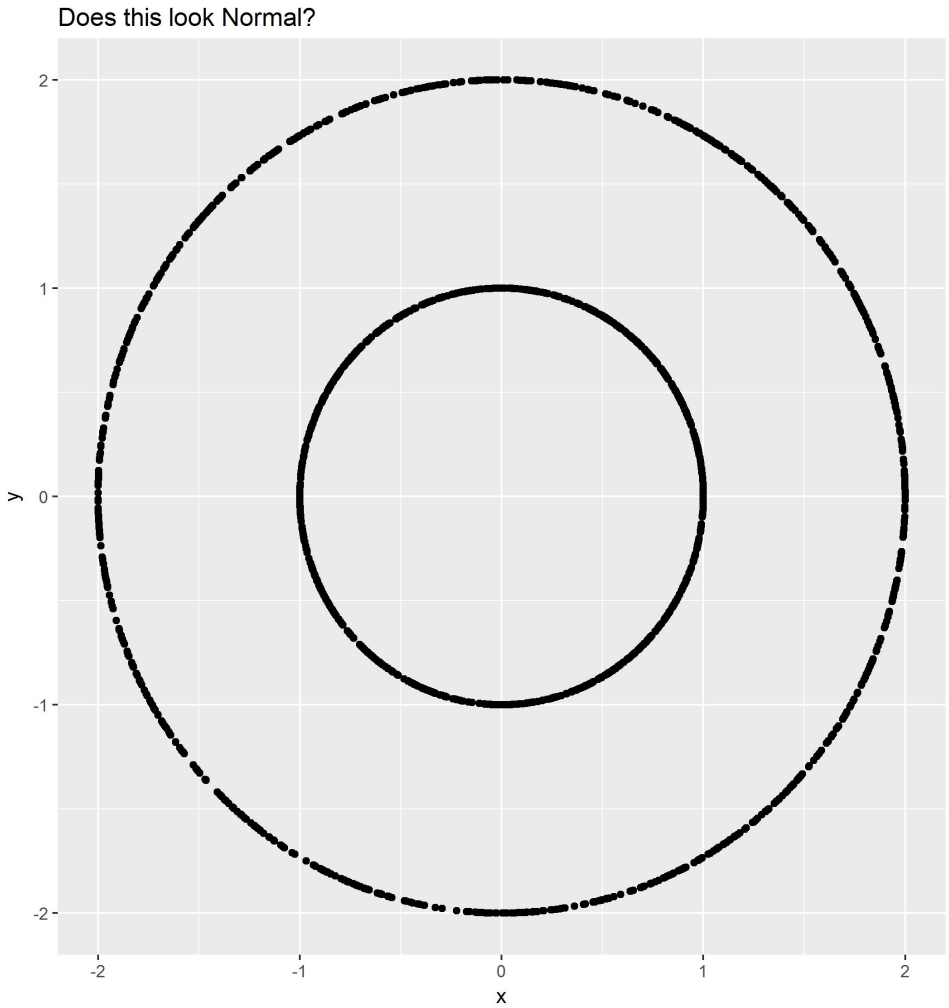




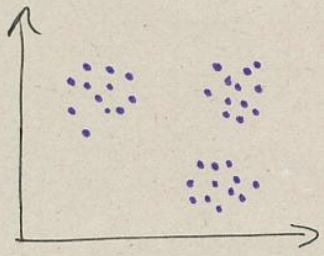
Normal?

(A sample from two Normals in high dimensions)

- Can get unexpected shapes.
- We need a definition (if possible).
- Deal with these shapes (if possible).

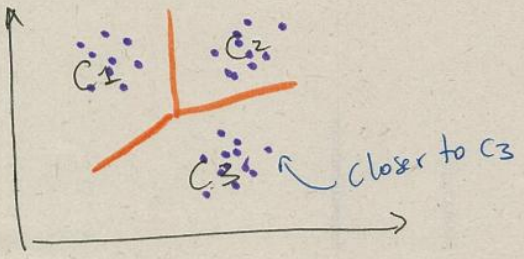


K-means



(K is given)

- Find $\rho = \{c_1, c_2, c_3\}$
- Form Voronoi tesselation



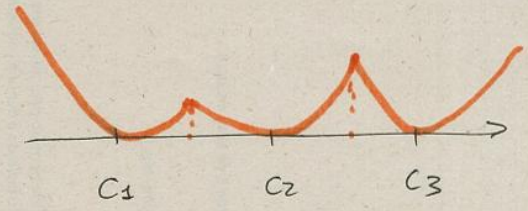
Stats Pt of View

$X_1, \ldots, X_n \sim P$

Given ρ:

$\text{loss} = \mathbb{E}[\min_j \|X - c_j\|^2]$

$R(c) :=$

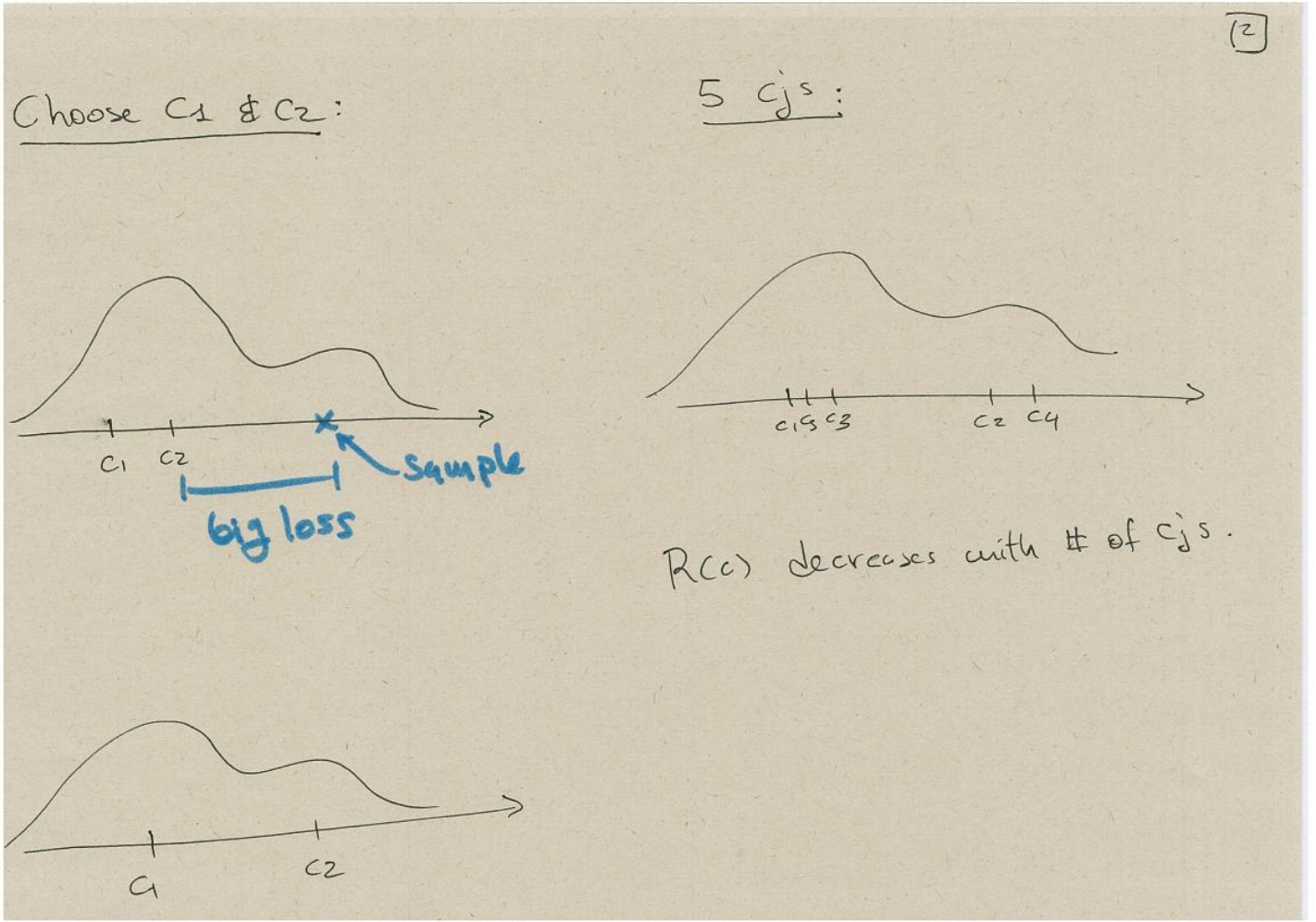


$\underset{c}{\text{minimize}}\ R(c)$

Non-convex optim problem

(In fact NP-Hard)

k-means Guessing the answer



Lloyd's Algo

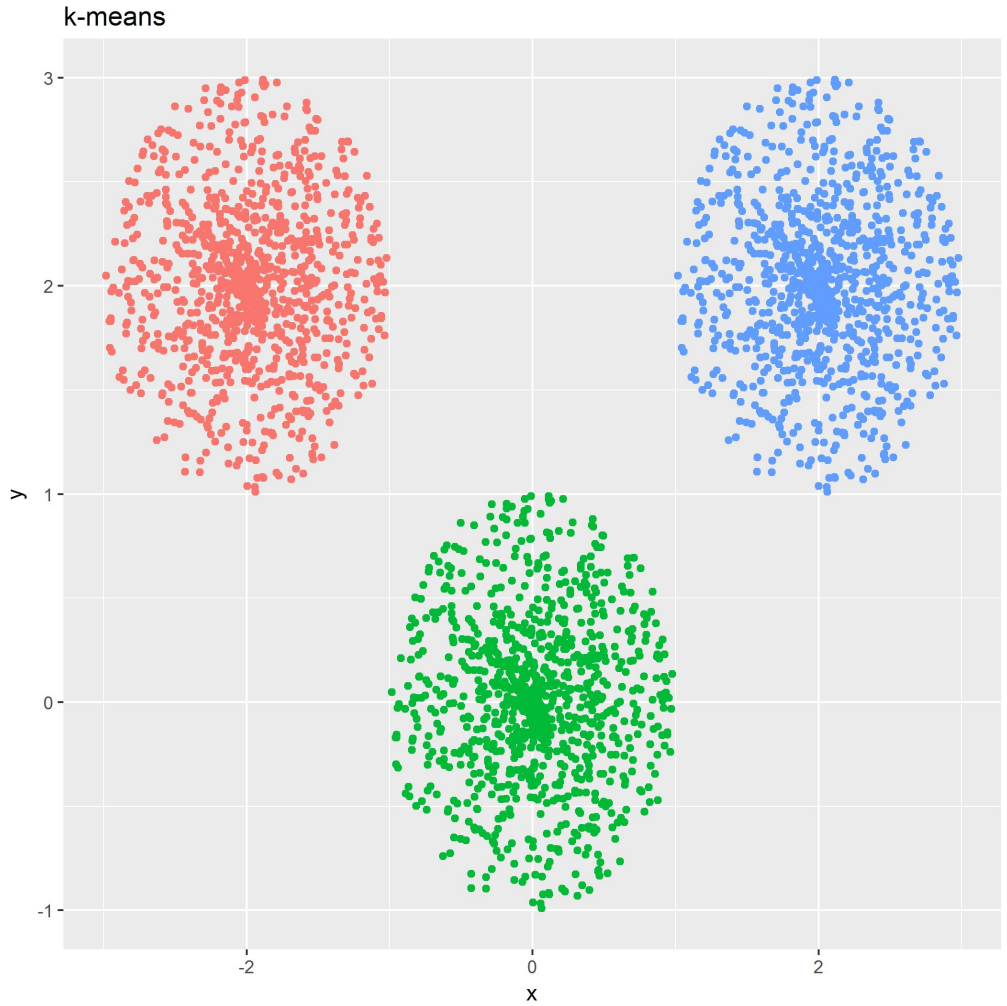
Good Approx:

(Lloyd's algo)

1. Choose $c_1, \ldots, c_k$.
2. Form the clusters.
   Compute $d_j(x_k) = \|x_k - c_j\|^2$
3. Reestimate centers:
   $c_j \leftarrow \frac{1}{|C_j|} \sum_{x_i \in C_j} x_i$

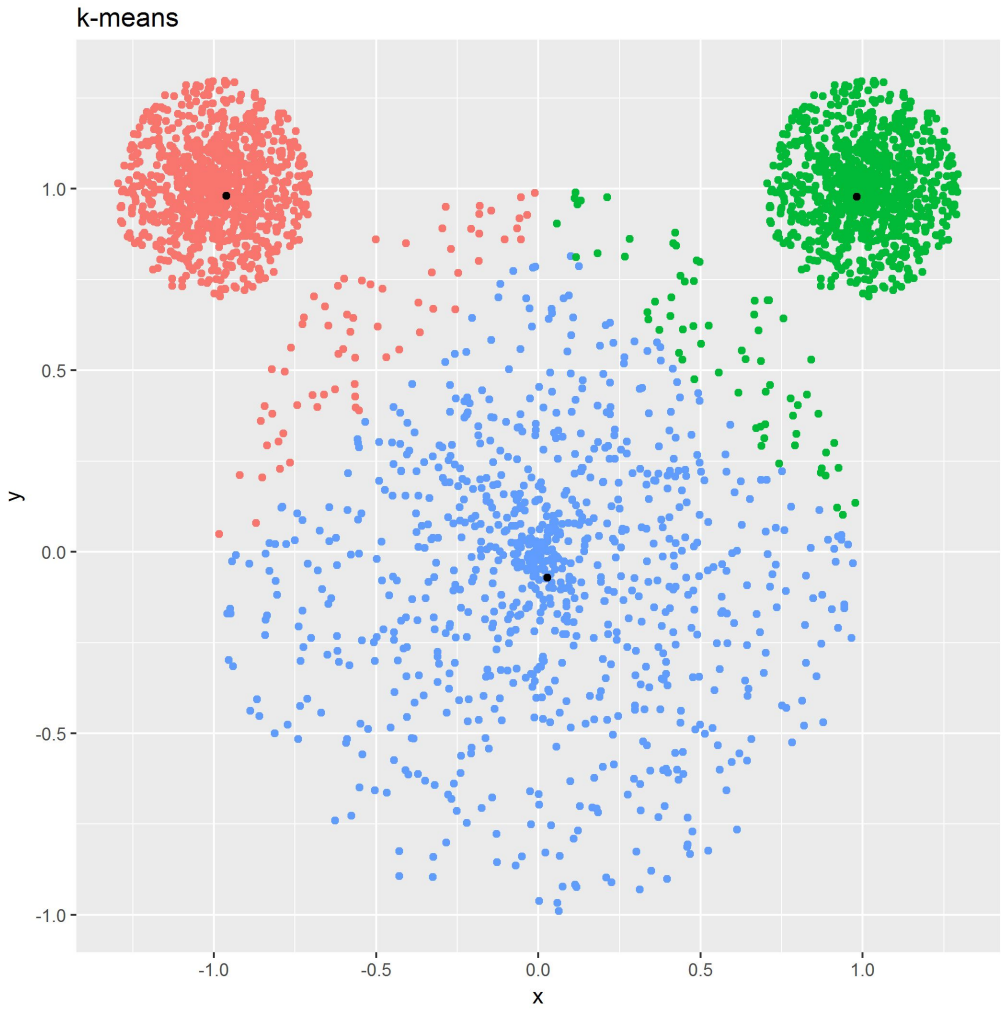
Repeat until convergence.
(local min)

Mickey Mouse Example

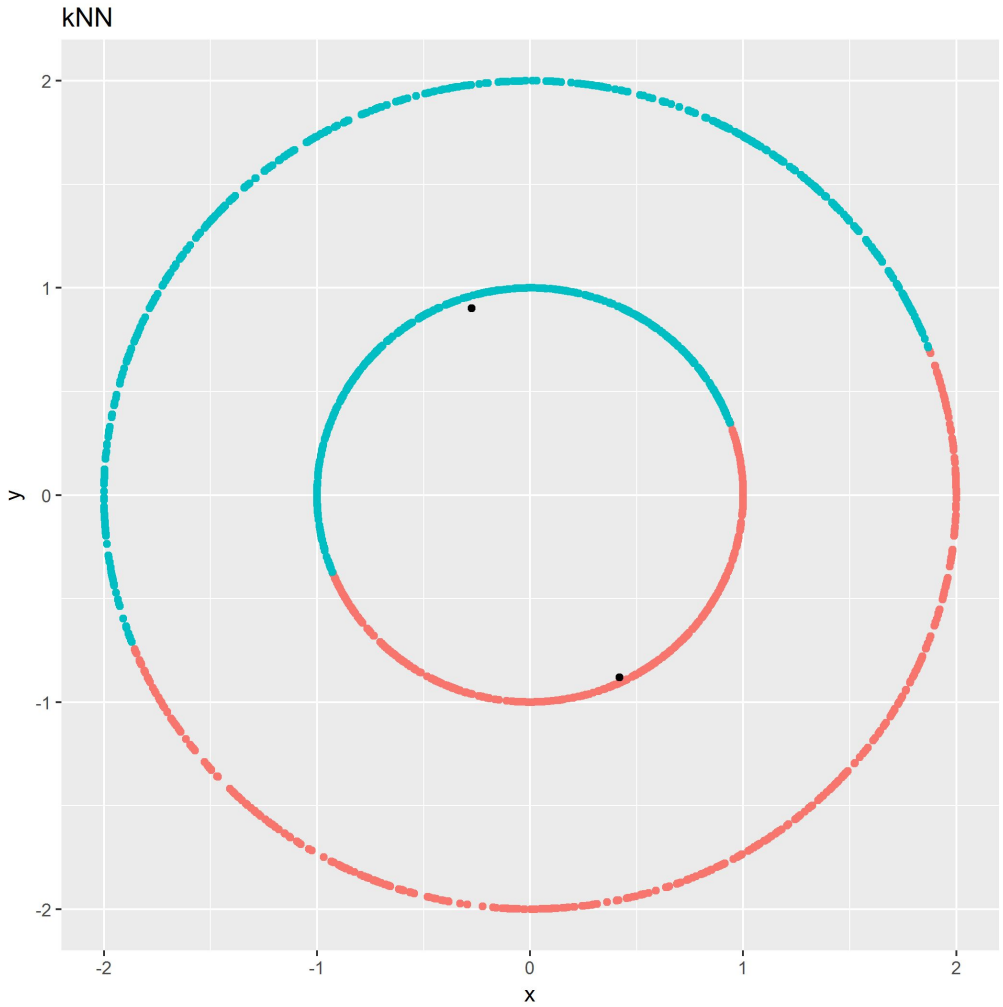


Mickey Mouse Example

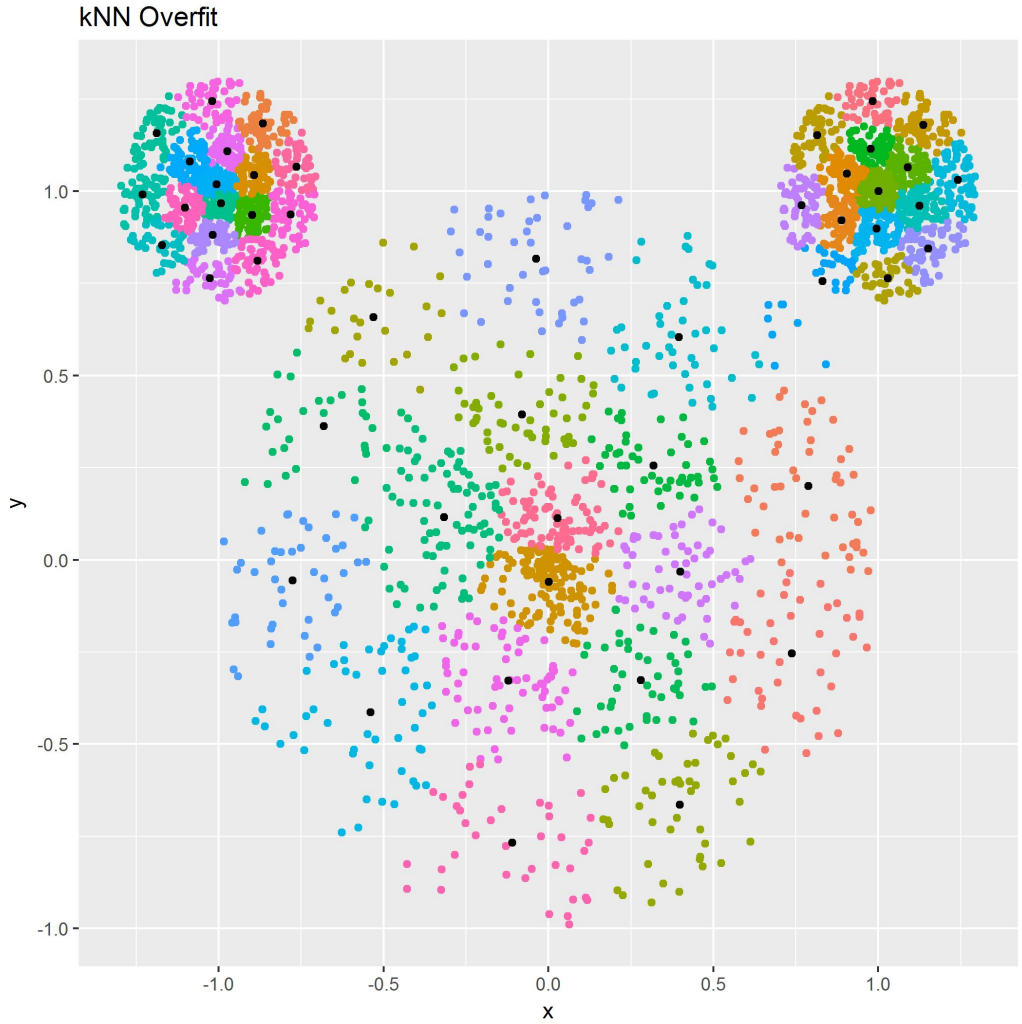
- k-means is supposed to do this.
- k is NOT the # of clusters.
- It makes sense to divide the density as in the fig (otherwise big loss).



Two Rings Example

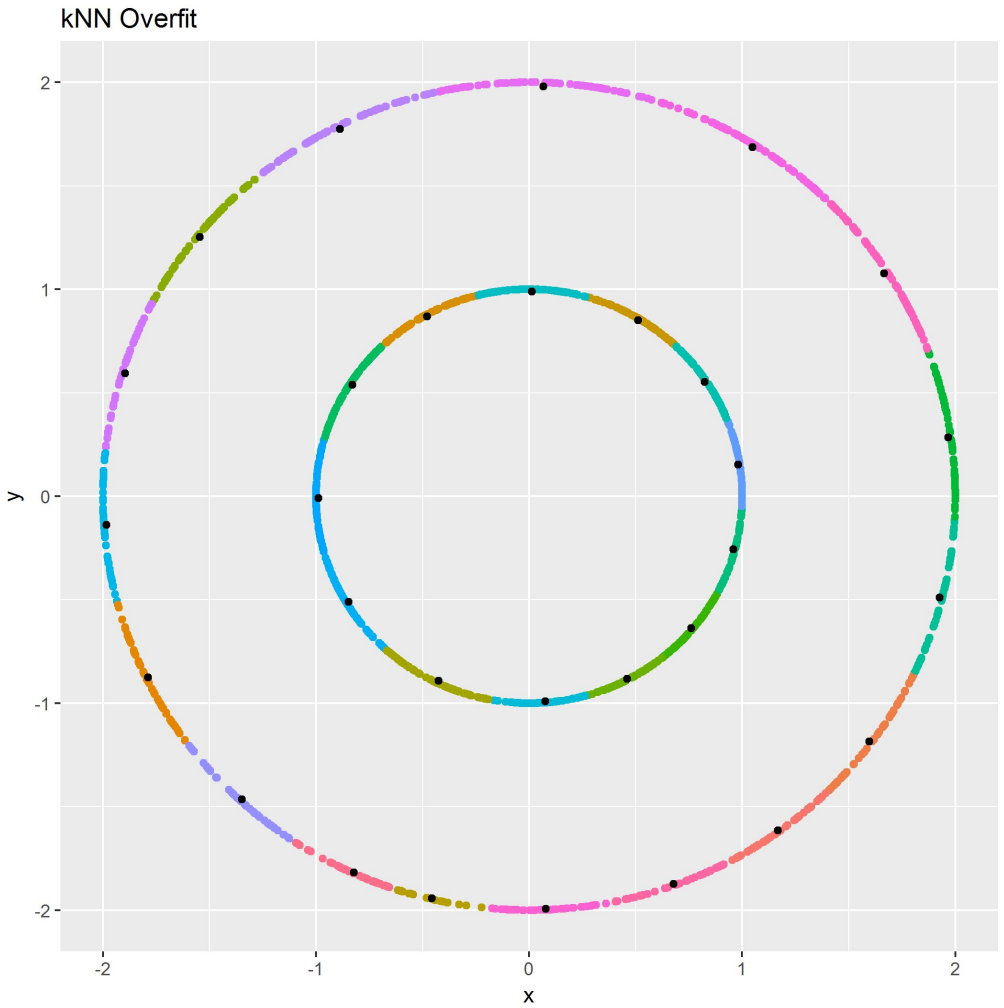


Overfit and Merge (L. Wasserman)



Overfit and Merge (L. Wasserman)

- Open problem:
  - Define this proc.
  - k = n^½?
  - Merging not clear.



Pros and Cons

- Easy to implement.
- Easy to modify.
- Seems to work ok in practice.

- Heuristic (local min).
- No guarantees.
- Not a clear notion of cluster.

Mixture Models (~Generalized-Bayesian k-means)

- Generative point of view.
- Data comes from a mixture of Gaussians.
  - Many parameters O(kd^2)
  - E.M. or Gibbs sampling.

Many unknown things:

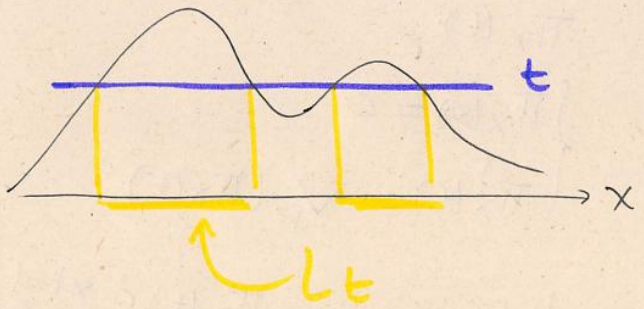
- Identifiability, infinite posteriors, non-intuitive groups, etc.
- Use with -some- caution.

Density-Based Clustering

1. Estimate the density.
2. Use the density to cluster.

Problems in high dimensions when estim. the density.

Level-Set Clustering



$$L_t := \{x : p(x) > t\} = \bigcup_j C_j$$

- High density region is an intuitive notion of cluster

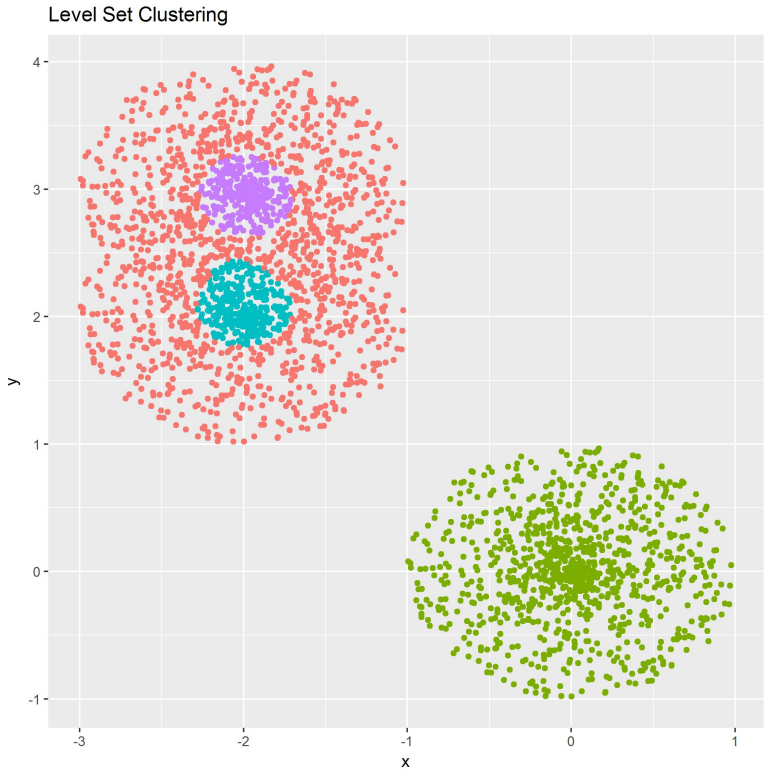
Density tree

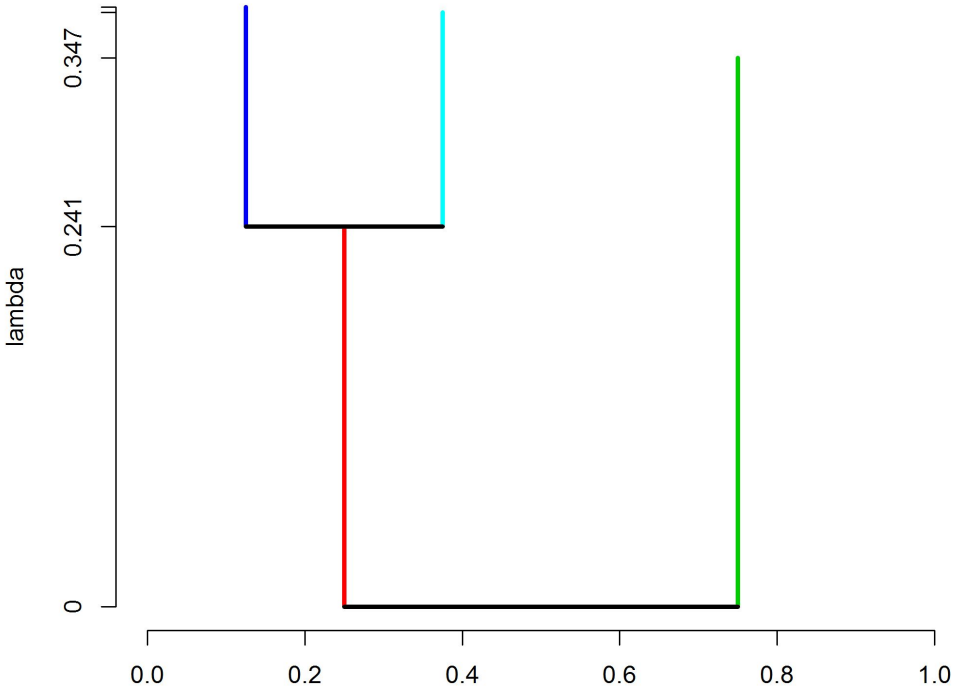
$$C = \bigcup_{t \geq 0} C_t$$

is a tree.

Goal: Build the tree.

Level-Set Clustering

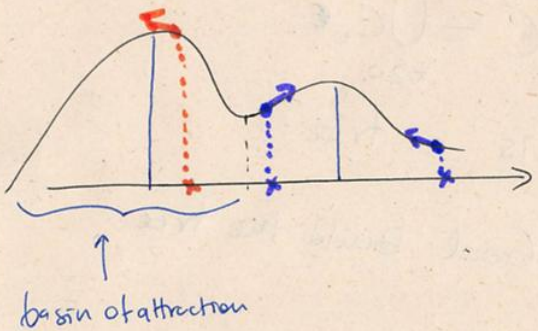




Mean-Shift Clustering

Mode = local max.

1D:



2D:

Define Paths:

$\Pi_x(t)$

$$\begin{cases} \Pi_x(0) = 0 \\ \Pi_x'(t) = \nabla_p(\Pi_x(t)) \end{cases}$$

$$dest(x) = \lim_{t \to \infty} \Pi_x(t) \in \text{set of modes}$$

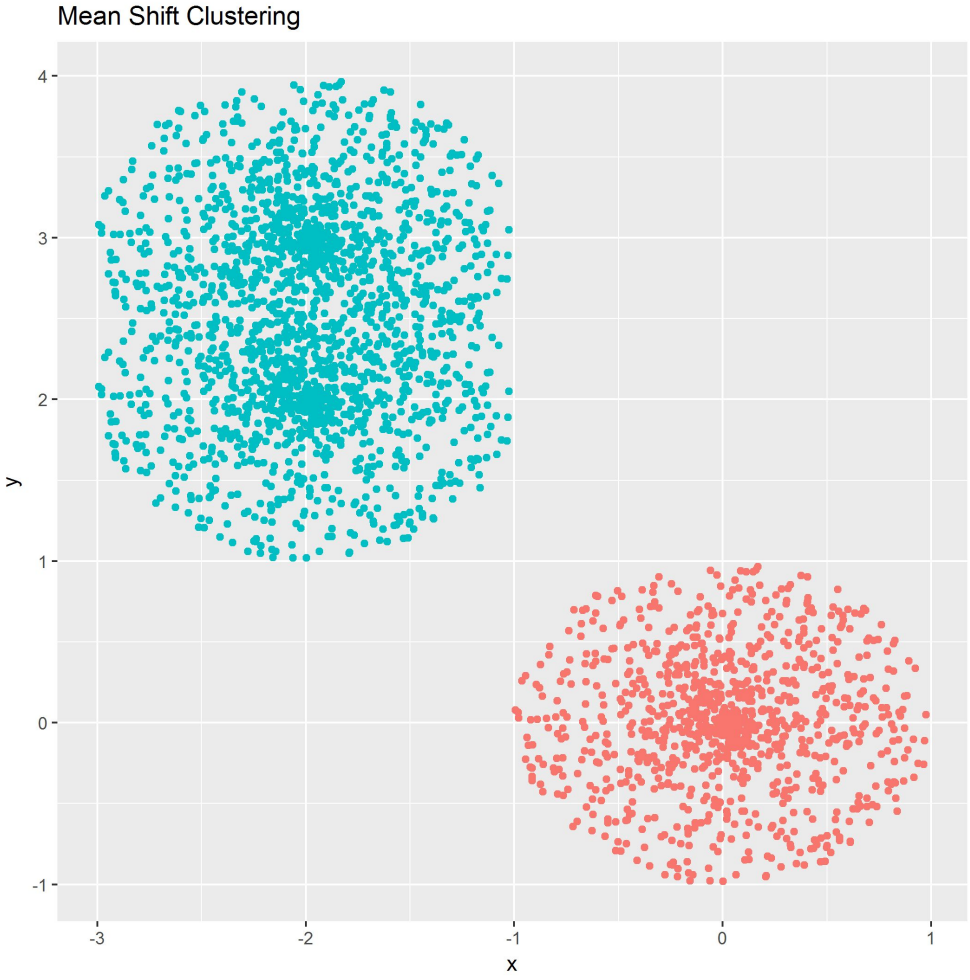
MS-Algo

⓪ Pick any point $a = a^{(0)}$.

①

$$a^{(1)} \leftarrow \frac{\sum_{i=1}^{n} x_i k\left(\frac{x_i - a_k^{(0)}}{h}\right)}{\sum_{i=1}^{n} k\left(\frac{x_i - a_k^{(0)}}{h}\right)}$$

Mean-Shift Clustering



Other Clustering Techniques

- Hierarchical clustering.
  - Not clear what it does.
  - Inconsistent.

- Spectral Clustering.
  - Cool math.
  - Good for graphs.

Spectral Clustering

Graph laplacian

(~ derivative operator for graphs)

$$w_{i,j} = \begin{cases} 1 & \text{if } \|x_i - x_j\| < \varepsilon, \\ 0 & \text{otherwise.} \end{cases}$$

$$L = D + W$$

where

$$D[i,j] = \sum_j w_{ij}$$

$$f := (f_1, \ldots, f_n)$$

$$f^T L f = \frac{1}{2} \sum_{ij} w_{ij} (f_i - f_j)^2$$

small weights to pts. far appart.

Differences on the graph.

eigen(L)

- $0 = \lambda_1 < \lambda_2 < \cdots < \lambda_n$
- $|\lambda_j = 0|$ = # of connected components.

Spectral Clustering

Usually you need to:

- Find the graph

$$w_{ij} = e^{-\|x_i - x_j\|^2 / 2h^2}$$

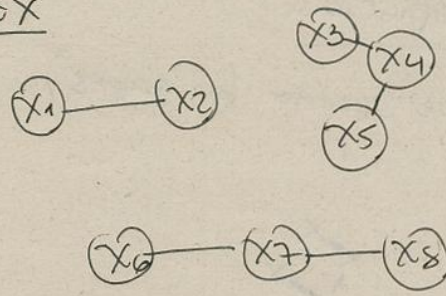
- Def -

$$L = D^{-1} W$$

↑ Like a kernel density estimation.

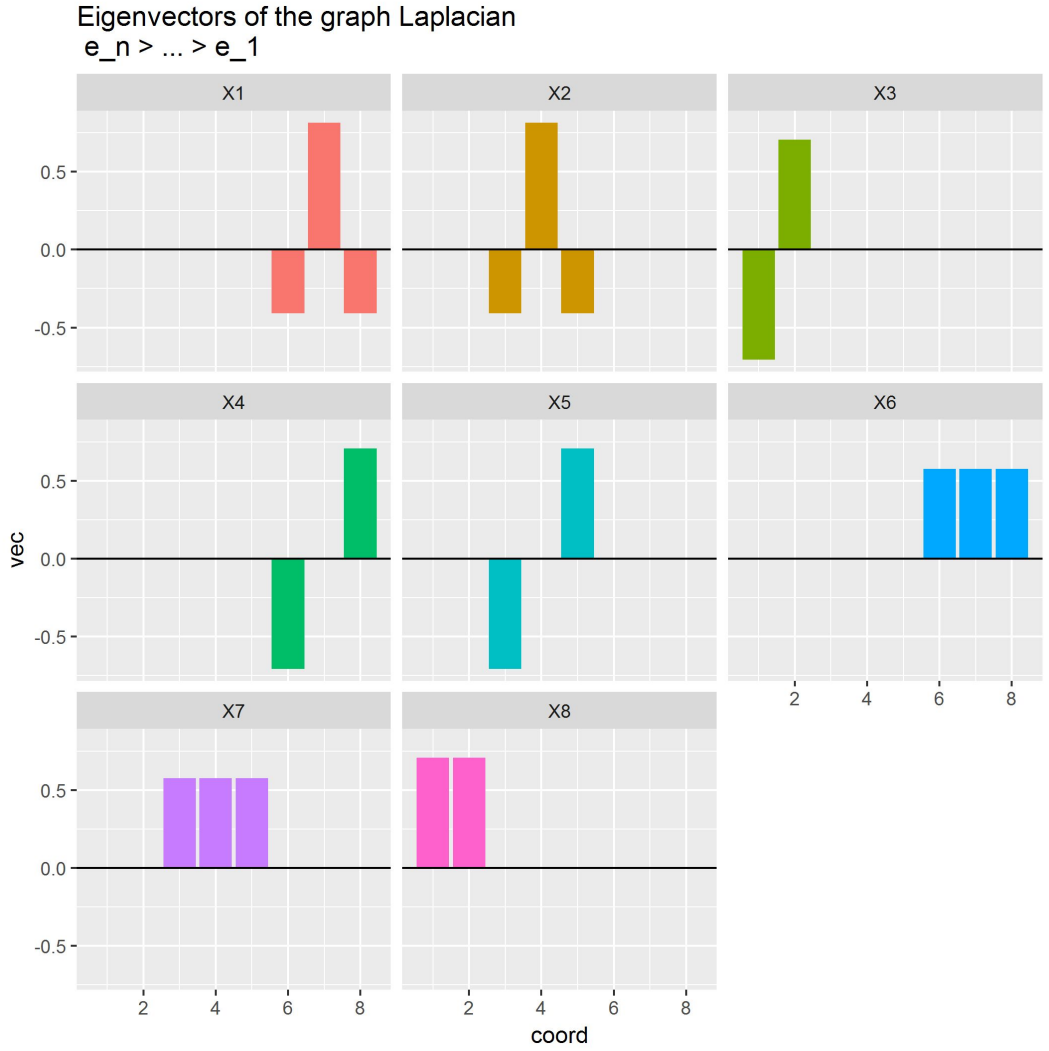
- Pick the first S eigenvects. & run K-means clustering in the subspace

Ex



$$L = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 1 & & & \\ 0 & & 1 & -1 & 0 \\ & & -1 & 2 & -1 \\ & & 0 & -1 & 1 \end{bmatrix}$$

Spectral Clustering



High Dim. Clustering

$X \sim N(0, I)$; $X \in \mathbb{R}^d$

Not hard to show that:

$$\|x\|^2 = d$$

i.e. points lie on rings.

One fix:

Modify the distance

$$\rho(x, y) := \frac{1}{n-2} \sum_{z \neq x, y} \left| \|x - z\| - \|y - z\| \right|$$

[Sarkar, & Ghosh]

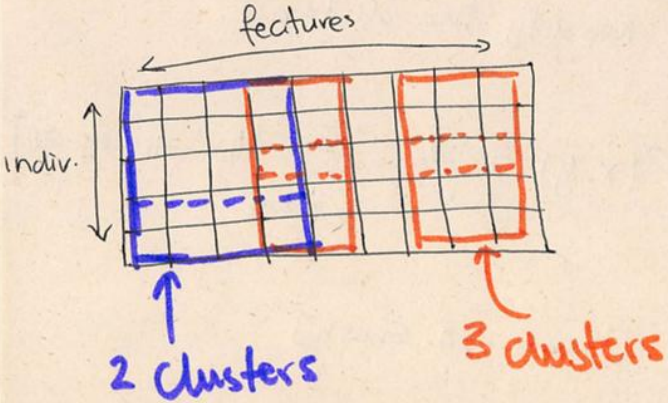
⚠ Proofs assume independence between coordinates...

Open problem:

Adjust k-means to deal with high dim. effects

Mosaics

Find Features that define clusters.



(interesting cluster)

A First Idea:

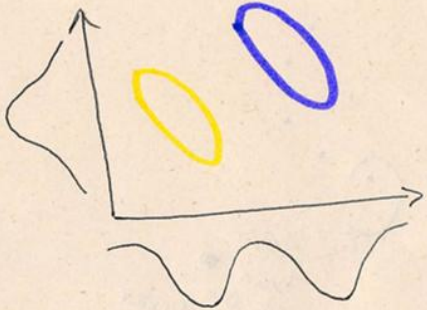
Screening: Are there marginal clusters?

$X = (X[1], \ldots, X[n])$.

$X_1[1], \ldots, X_m[n] \sim F$

H_0: F is unimodal.

H_1: F is not.



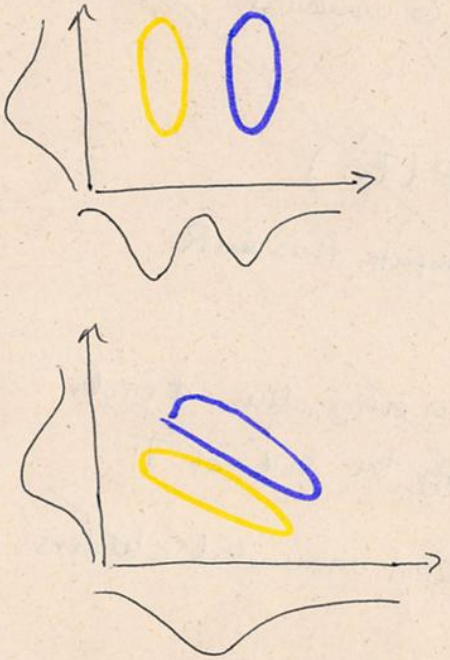
Screening

$$\mathrm{Dip}(F) = \inf_{G \text{ unimodal}} \sup_x |F(x) - G(x)|$$

$$T := \mathrm{Dip}(\hat{F}_n)$$

↑ can compute this in R

- Do this in every dim. & order the T's.
- Select least unimodal?



Alternating Sparse Clustering

$$R_n = \frac{1}{n} \sum_{j=1}^{n} \min_{c} \|X_j - c\|^2$$

$$= \dots =$$

$$= \sum_{j=1}^{K} \frac{1}{|C_j|} \sum_{i,j} \|X_i - X_j\|^2$$

Choose $|S| = L$.

$a = 1, \dots, d$ index of features

$$\delta_a(i,j) := (X_i[a] - X_j[a])^2$$

Rescale so that $\sum_{i,j} \delta(i,j) = 1$

Do k-means with an alternative step:

- Find the centers.
- Find the features

↑

- minimize R_n to find the features.

A Model for Mosaics?

Each object is represented by a vect. of latent features f.

$f_i \longmapsto x_i$

↑ properties are generated by the latent features

N objects ; $F = \begin{bmatrix} f_1^T \\ \vdots \\ f_N^T \end{bmatrix}$

$f_i[j] = \begin{cases} 1 & \text{if } x_i \text{ has prop. } j, \\ 0 & \text{otherwise.} \end{cases}$

Idea:

Det. a a priori dist $P(F)$, compute $P(X|F)$.

Finite-dim model

features

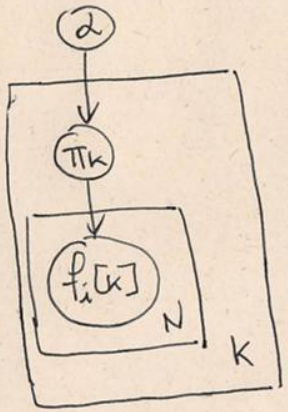
$$P(F|\pi) = \prod_{j=1}^{K} \pi_j^{m_j} (1-\pi_j)^{N-m_j}$$

m_j = # of obj's possessing feature j.

$\pi_j = P(\text{possessing feature } j)$

$P([100100]) = \pi_1 (1-\pi_1)^5 + \cdots + \pi_6^3 (1-\pi_6)$

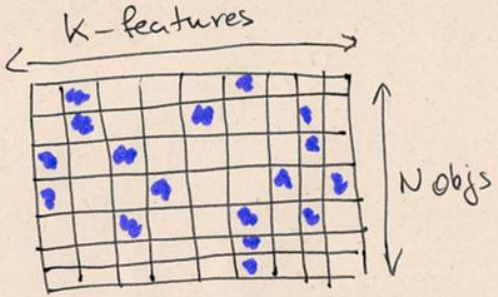
A Model for Mosaics?



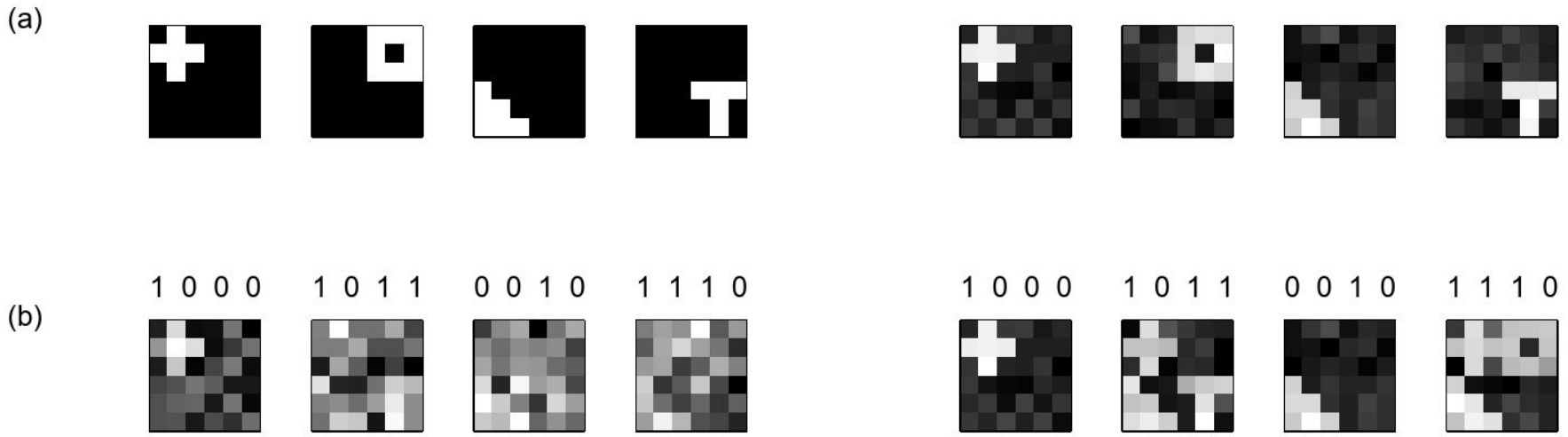
$$\begin{cases} \pi_k \mid \alpha \sim \text{Beta}\left(\frac{\alpha}{K}, 1\right) \\ f[i,k] \mid \pi_k \sim \text{Bern}(\pi_k) \end{cases}$$

Given the model:

$$P(\text{obs} \mid F) \propto P(F \mid \text{obs})\, P(F)$$



A Mosaic Model



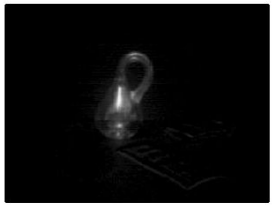
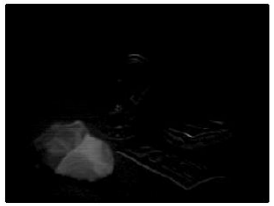
A Mosaic Model

(a)



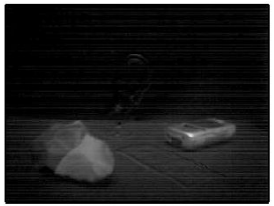

(Positive) (Negative) (Negative) (Negative)

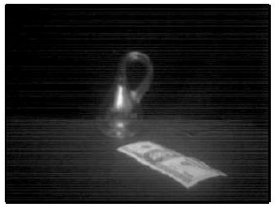
(b)


0 0 0 0 0 1 0 0 1 1 1 0 1 0 1 1

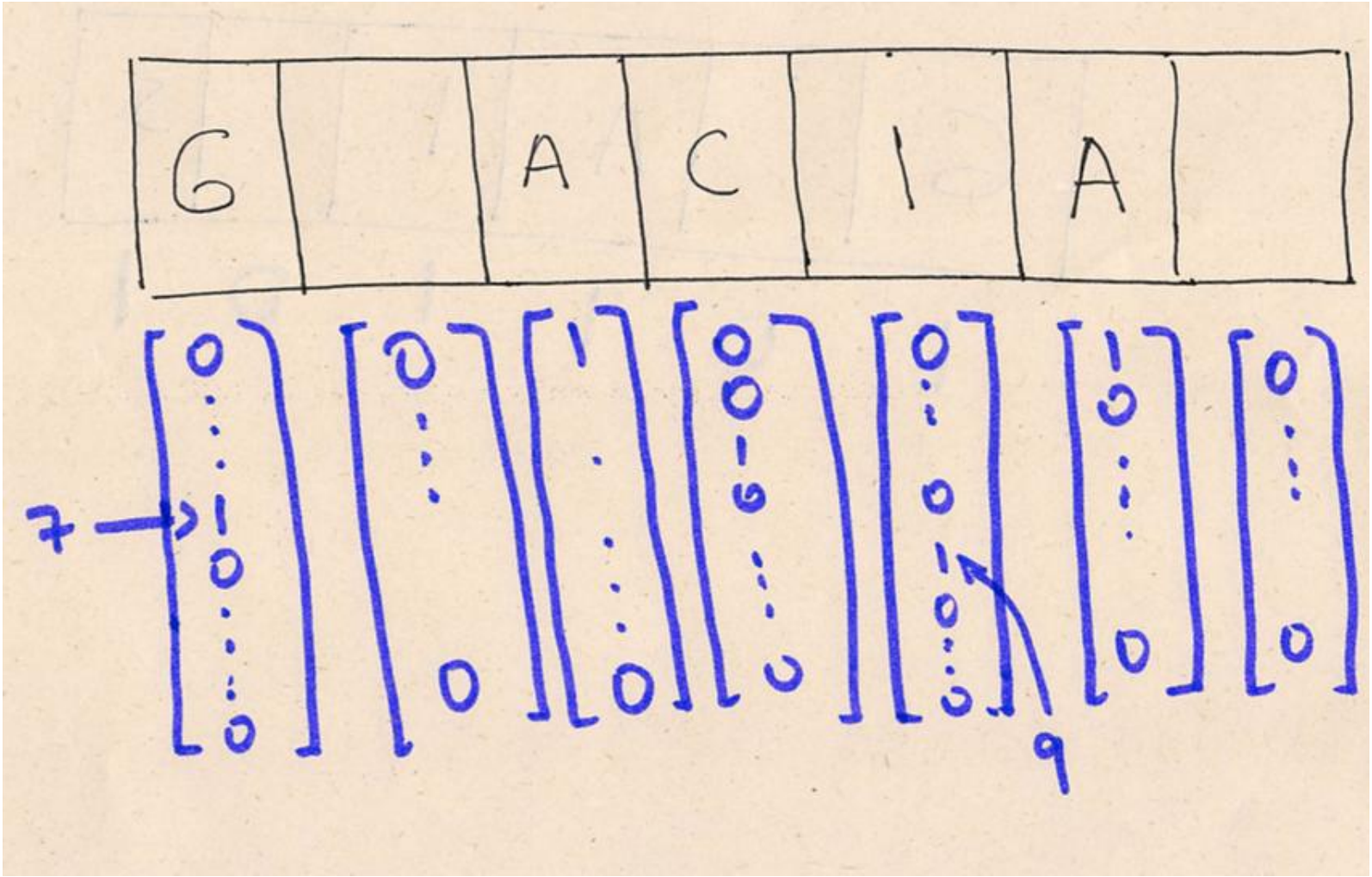
(c)



Some references:

1. *Statistical Machine Learning*, Ryan Tibshirani, Larry Wasserman.
2. *Functional Data Clustering: a Survey*, Julien Jacques, Cristian Preda.
3. *A Simple Approach to Sparse Clustering*, Ery Arias-Castro, Xiao Pu.
4. *On Perfect Clustering of High Dimension, Low sample Size data,* Soham Sarkar, Anil K. Ghosh.
5. *Tree-Structured Stick Breaking Process for Hierarchical Data*, Ryan P. Adams, Zoubin Ghahramani and Michael I. Jordan
6. *The Indian Buffet Process: An Introduction and Review*, Thomas Griffiths and Zoubin Ghahramani.


G
A
C
I
A
7
9