3/17/26

Lecture 18

Clustering
Mixtures of Gaussian

LVI posted
Q3 Thu 3/26
[HW6 posted]

Lecture VII: Clustering: K-means and Mixtures of Gaussians

Marina Meilă
mmp@uwaterloo.ca

With Thanks to Pascal Poupart & Gautam Kamath
Cheriton School of Computer Science
University of Waterloo

March, 2026

Paradigms for clustering

K-means clustering

Mixtures of Gaussians and the EM algorithm

Special topics in clustering

Reading HTF Ch.: 14.3, Murphy Ch.: Ch 11.[1], 11.2.1-3, 11.3, Ch 25, Bach Ch.:

What is clustering? Problem and Notation

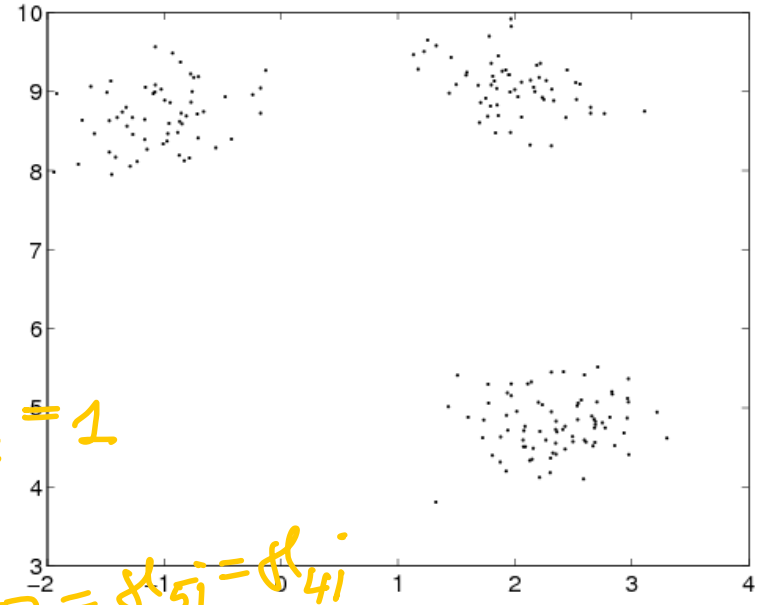
- **Informal definition** Clustering = Finding groups in data
- **Notation**

$\mathcal{D}$	=	$\{\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n\}$ 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
$\mathcal{L}(\Delta)$	=	cost (loss) of Δ (to be minimized)

- **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

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

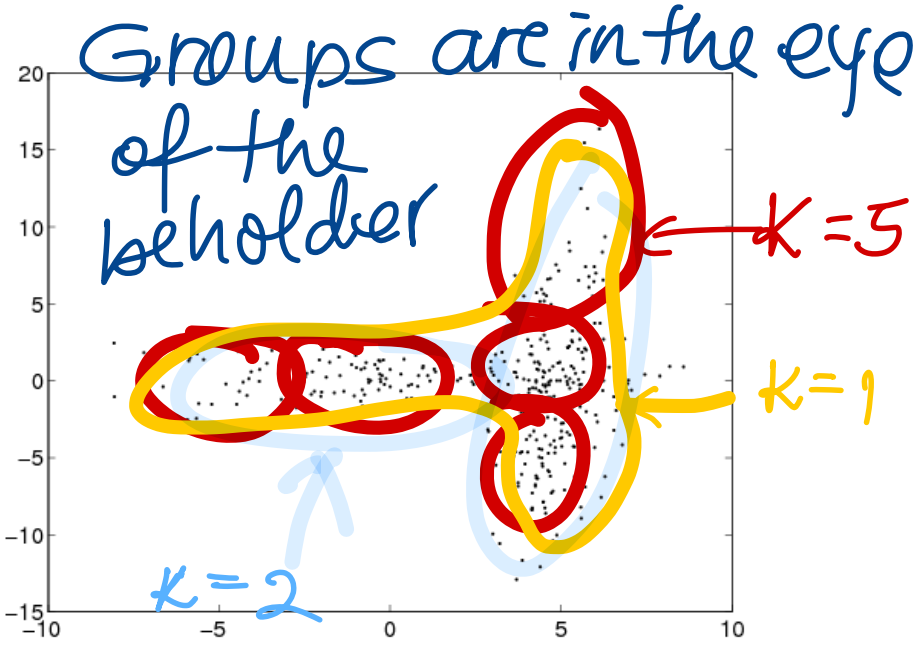
(usually associated with a probabilistic model)

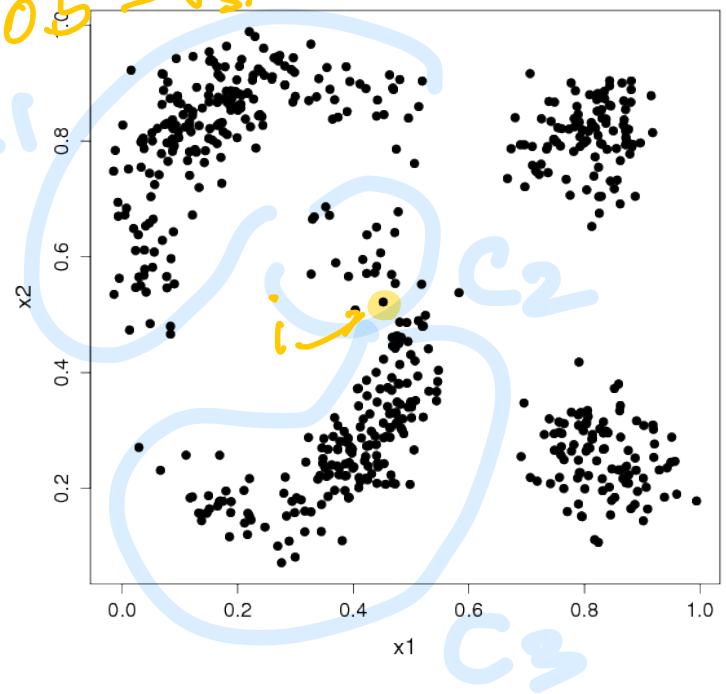


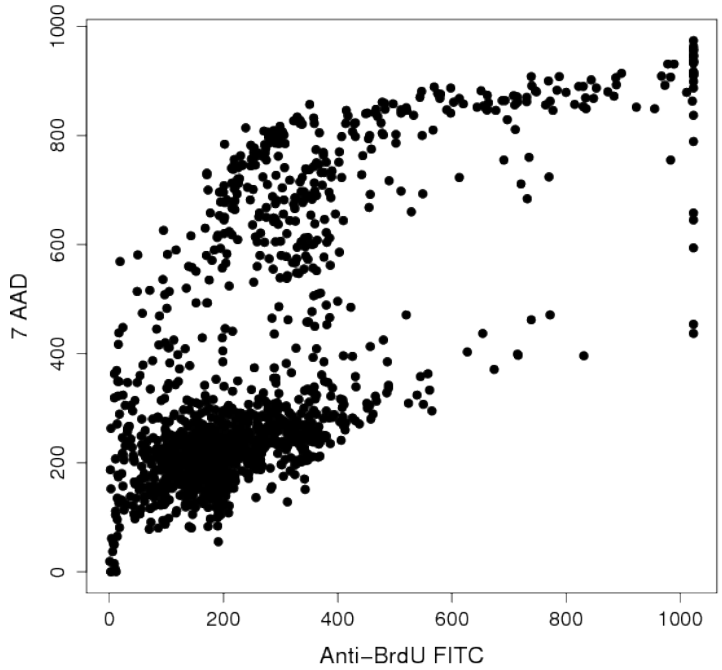
$\sum_{k=1}^{K} \gamma_{ki} = 1$

$\gamma_{1i} = 0 = \gamma_{5i} = \gamma_{4i}$

$\gamma_{2i} = 0.5 = \gamma_{3i}$







(from [?])

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

- 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) $\mathcal{L}$	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

Parametric clustering algorithms

- Cost based
  - Single linkage (min spanning tree)
  - Min diameter
    - Fastest first traversal (HS initialization)
  - K-medians
  - K-means
- Model based (cost is derived from likelihood)
  - EM algorithm
  - "Computer science" / "Probably correct" algorithms

["statistical"]

← Mixture Models

K-means clustering

Algorithm K-Means[?]

Input Data $\mathcal{D} = \{x_i\}_{i=1:n}$, number clusters K

tialize **centers** $\mu_1, \mu_2, \ldots \mu_K \in \mathbb{R}^d$ at random

terate until convergence

1. for $i = 1 : n$ (assign points to clusters $\Rightarrow$ new clustering)

$$k(i) = \operatorname*{argmin}_k ||x_i - \mu_k||$$

2. for $k = 1 : K$ (recalculate centers)

$$\mu_k = \frac{1}{|C_k|} \sum_{i \in C_k} x_i \tag{1}$$

- Convergence
  - if Δ doesn't change at iteration m it will never change after that
  - convergence in finite number of steps to local optimum of cost $\mathcal{L}$ (defined next)
  - therefore, initialization will matter

k-means Algo

C_k represented by $\mu_k \in \mathbb{R}^d$, $k = 1:k$

$x^1 .. x^n \in \mathbb{R}^d$

$x^i \in C_k \Leftrightarrow \|\mu_k - x^i\| = \min_{k'=1:k} \|\mu_{k'} - x^i\|$

$\Delta \leftarrow \{\mu_{1:k}\}$

Definition of clusters

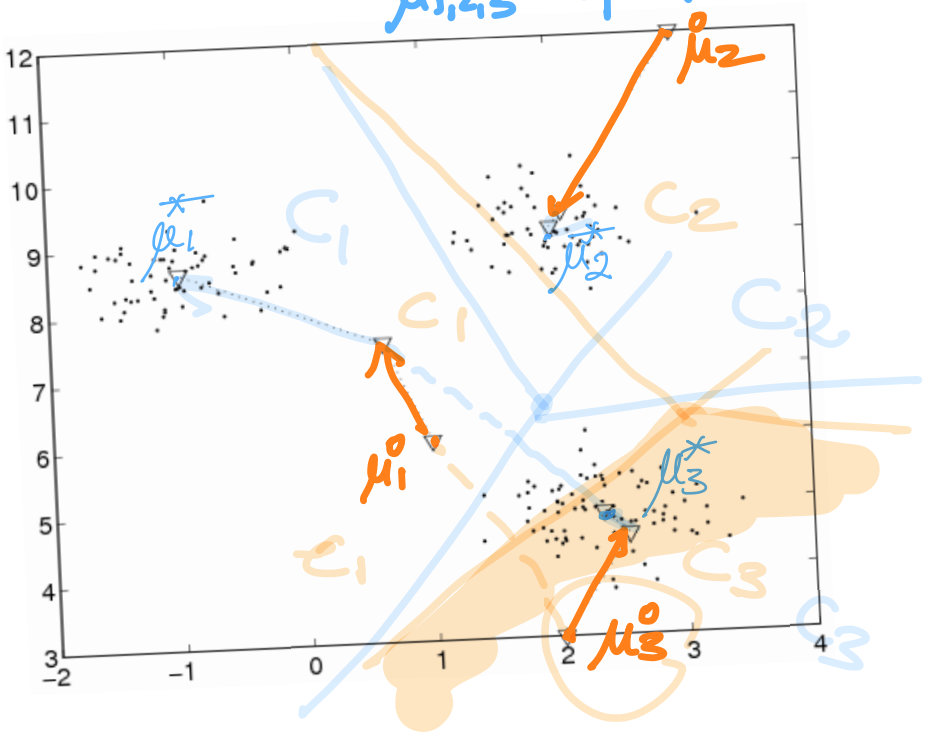
Loss $\mathcal{L}(\Delta) = \sum_{k=1}^{k} \sum_{i \in C_k} \|x^i - \mu_k\|^2$

quadratic loss (distortion)

Best $\Delta = \underset{\mu_{1:k}}{\operatorname{argmin}}\ \mathcal{L}(\Delta)$ ...

$\mu^0_{1,2,3}$ = initial guess

$\mu^*_{1,2,3}$ = opt. for $k=3$



The K-means cost

$$\mathcal{L}(\Delta) = \sum_{k=1}^{K} \sum_{i \in C_k} ||x_i - \mu_k||^2 \tag{2}$$

- K-means solves a **least-squares** problem
- the cost $\mathcal{L}$ is called **quadratic distortion**

Proposition The K-means algorithm decreases $\mathcal{L}(\Delta)$ at every step.

Sketch of proof

- step 1: reassigning the labels can only decrease $\mathcal{L}$
- step 2: reassigning the centers μ_k can only decrease $\mathcal{L}$ because μ_k as given by (1) is the solution to

$$\mu_k = \min_{\mu \in \mathbb{R}^d} \sum_{i \in C_k} ||x_i - \mu||^2 \tag{3}$$

Equivalent and similar cost functions

- The distortion can also be expressed using intracluster distances

$$\mathcal{L}(\Delta) = \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i,j \in C_k} ||x_i - x_j||^2 \tag{4}$$

- **Correlation clustering** is defined as optimizing the related criterion

$$\mathcal{L}(\Delta) = \sum_{k=1}^{K} \sum_{i,j \in C_k} ||x_i - x_j||^2$$

- This cost is equivalent to the (negative) sum of (squared) intercluster distances

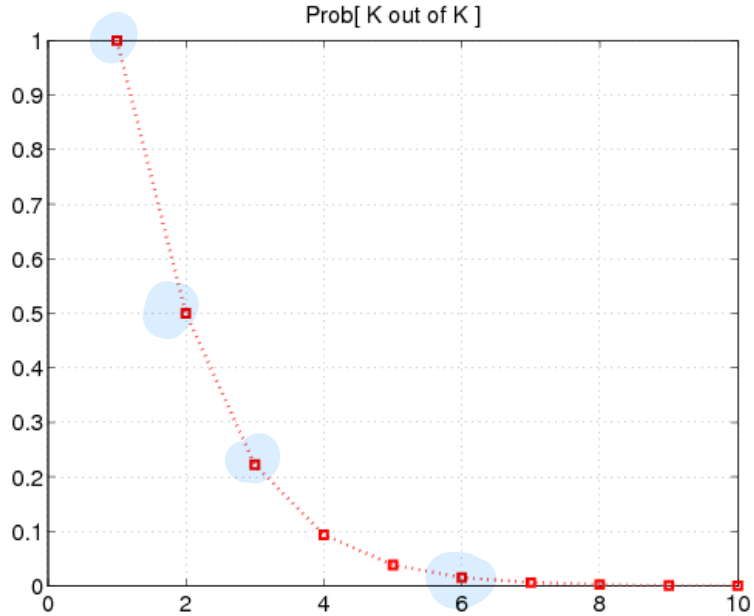
$$\mathcal{L}(\Delta) = -\sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j \notin C_k} ||x_i - x_j||^2 + \text{constant} \tag{5}$$

Proof of (4) Replace μ_k as expressed in (1) in the expression of $\mathcal{L}$, then rearrange the terms

Proof of (5) $\sum_k \sum_{i,j \in C_k} ||x_i - x_j||^2 = \underbrace{\sum_{i=1}^{n} \sum_{j=1}^{n} ||x_i - x_j||^2}_{\text{independent of } \Delta} - \sum_k \sum_{i \in C_k} \sum_{j \notin C_k} ||x_i - x_j||^2$

Initialization of the centroids $\mu_{1:K}$

- Idea 1: start with K points at random
- Idea 2: start with K data points at random
  What's wrong with chosing K data points at random?



  The probability of hitting all K clusters with K samples approaches 0 when $K > 5$
- Idea 3: start with K data points using Fastest First Traversal [?] (greedy simple approach to spread out centers)
- Idea 4: k-means++ [?] (randomized, theoretically backed approach to spread out centers)
- Idea 5: "K-logK" Initialization (start with enough centers to hit all clusters, then prune down to K)
  For EM Algorithm [?], for K-means [?]

The "K-logK" initialization

The K-logK Initialization (see also [?])

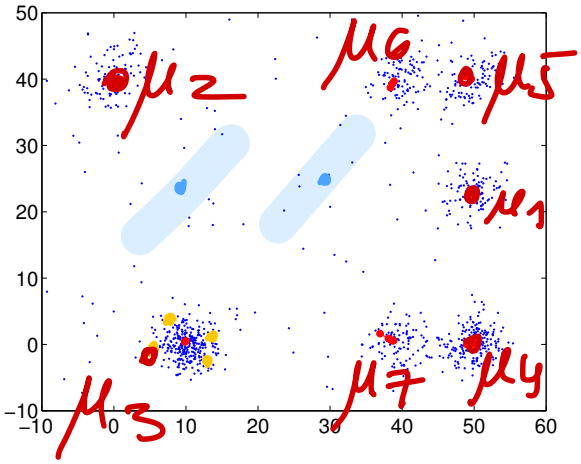
1. pick $\mu^0_{1:K'}$ at random from data set, where $K' = O(K \log K)$
   (this assures that each cluster has at least 1 center w.h.p)
2. run 1 step of K-means
3. remove all centers μ^0_k that have few points, e.g $|C_k| < \frac{n}{eK'}$
4. from the remaining centers ~~select K centers by~~ **Fastest First Traversal**
   - 4.1 pick μ_1 at random from the remaining $\{\mu^0_{1:K'}\}$
   - 4.2 for $k = 2 : K$, $\mu_k \leftarrow \underset{\mu^0_{k'}}{\operatorname{argmax}} \min_{j=1:k-1} ||\mu^0_{k'} - \mu_j||$, i.e next μ_k is furthest away from the already chosen centers
5. continue with the standard **K-means** algorithm

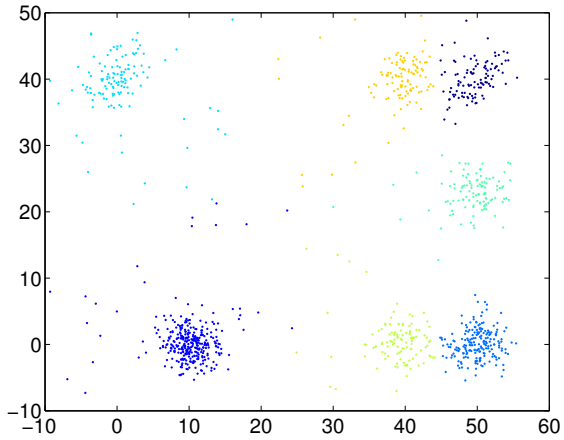
ensure each cluster C has ≥ 1 μ

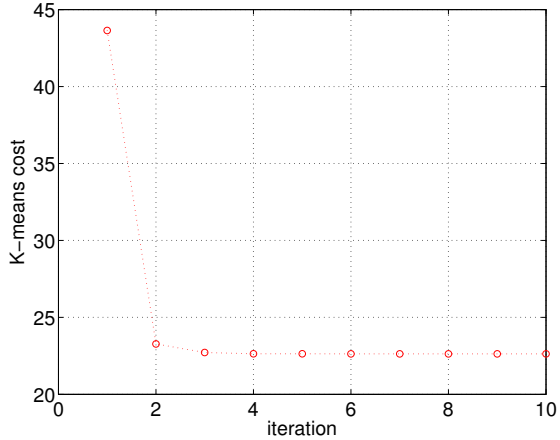
K-means clustering with K-logK Initialization

Example using a mixture of 7 Normal distributions with 100 outliers sampled uniformly

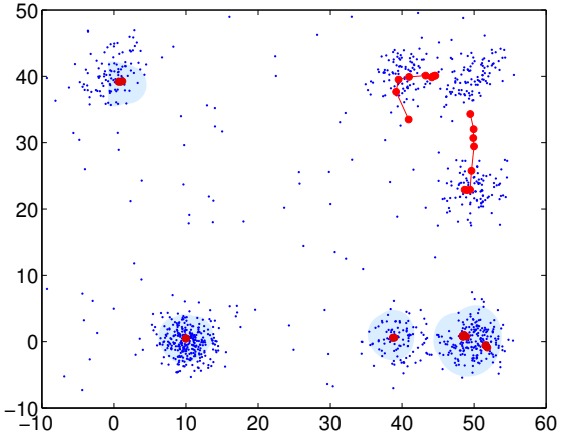
K-LOGK $K = 7$, $T = 100$, $n = 1100$, $c = 1$

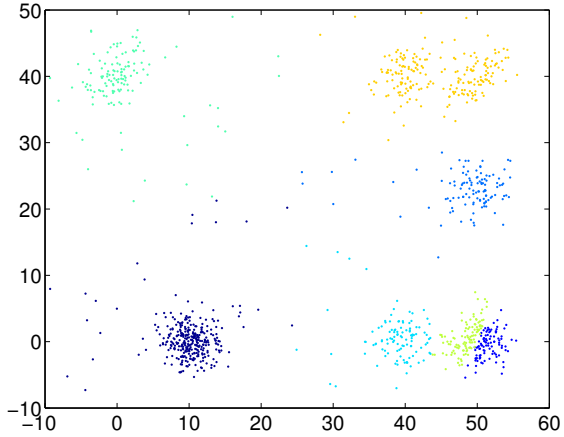


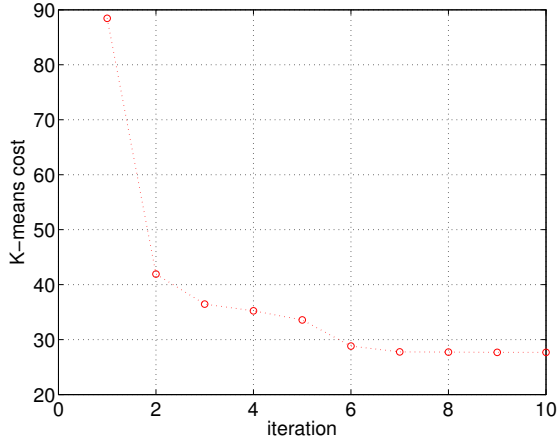




NAIVE $K = 7$ $T = 100$, $n = 1100$





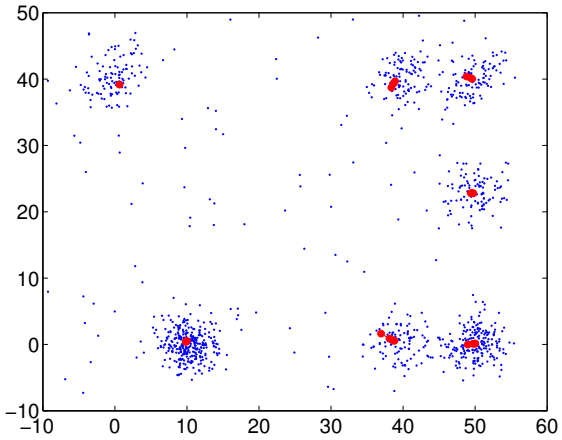


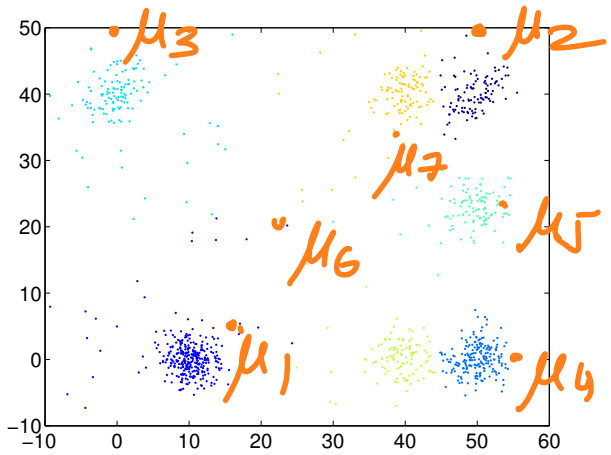
K-means clustering with K-logK Initialization

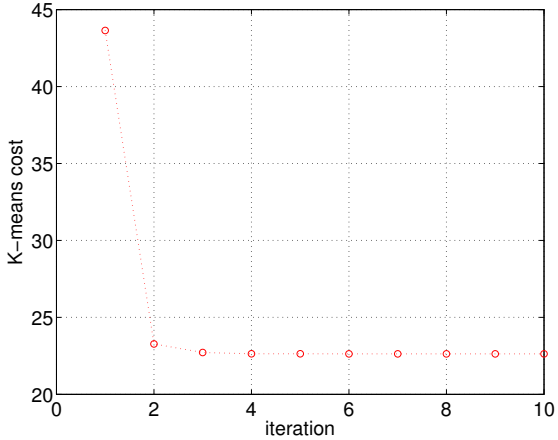
K-means++

- Example using a mixture of 7 Normal distributions with 100 outliers sampled uniformly

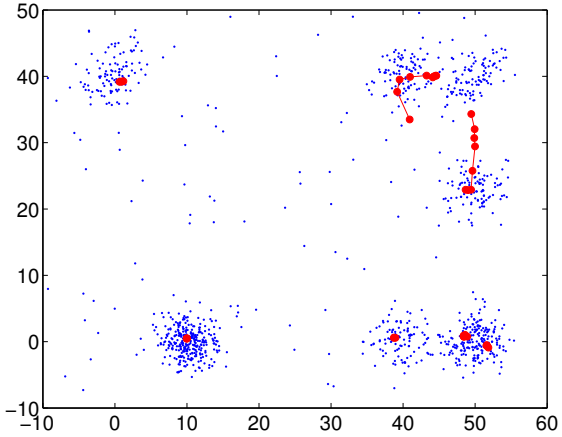
K-LOGK $K = 7$, $T = 100$, $n = 1100$, $c = 1$

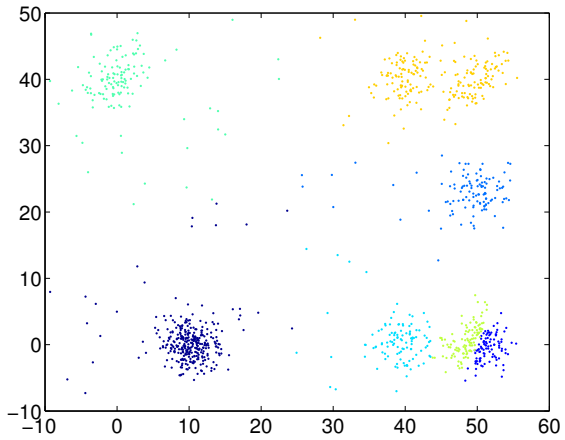


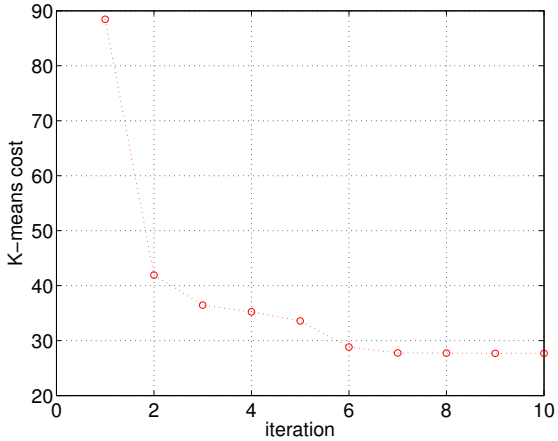




NAIVE $K = 7$ $T = 100$, $n = 1100$







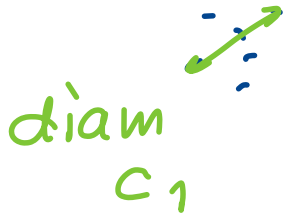
Minimum diameter clustering

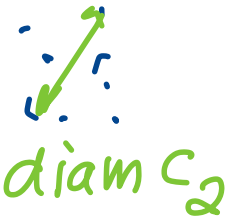
- **Cost** $\mathcal{L}(\Delta) = \max_k \underbrace{\max_{i,j \in C_k} ||x_i - x_j||}_{\text{diameter}}$
  - Mimimize the diameter of the clusters
  - Optimizing this cost is NP-hard
- **Algorithms**
  - **Fastest First Traversal** [?] – a factor 2 approximation for the min cost

For every $\mathcal{D}$, FFT produces a Δ so that

$$\mathcal{L}^{opt} \leq \mathcal{L}(\Delta) \leq 2\mathcal{L}^{opt}$$

  - rediscovered many times





Algorithm Fastest First Traversal

Input Data $\mathcal{D} = \{x_i\}_{i=1:n}$, number clusters K

defines **centers** $\mu_{1:K} \in \mathcal{D}$

(many other clustering algorithms use centers)

1. pick μ_1 at random from $\mathcal{D}$
2. for $k = 2 : K$

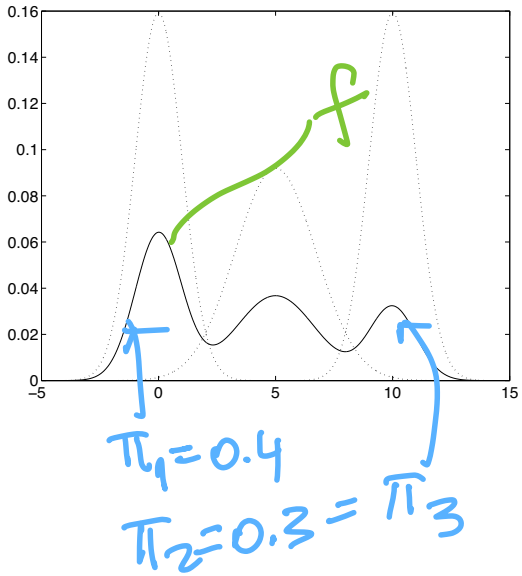
$$\mu_k \leftarrow \underset{\mathcal{D}}{\operatorname{argmax}} \operatorname{distance}(x_i, \{\mu_{1:k-1}\})$$

3. for $i = 1 : n$ (assign points to centers)

$k(i) = k$ if μ_k is the nearest center to x_i

Model based clustering: Mixture models

Mixture in 1D



- The **mixture density**

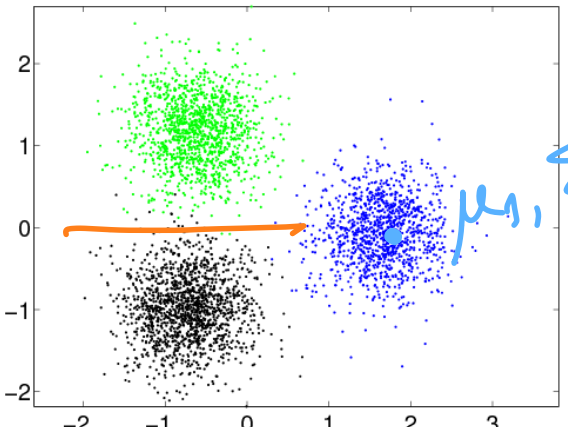
$$f(x) = \sum_{k=1}^{K} \pi_k f_k(x)$$

- $f_k(x)$ = the **components** of the mixture
  - each is a density
  - f called **mixture of Gaussians** if $f_k = Normal_{\mu_k, \Sigma_k}$
- π_k = the **mixing proportions**, $\sum_k = 1^K \pi_k = 1, \quad \pi_k \geq 0.$
- **model parameters** $\theta = (\pi_{1:K}, \mu_{1:K}, \Sigma_{1:K})$
- The **degree of membership** of point i to cluster k

$$\gamma_{ki} \stackrel{\text{def}}{=} P[x_i \in C_k] = \frac{\pi_k f_k(x)}{f(x)} \text{ for } i = 1:n, k = 1:K \tag{8}$$

- depends on x_i and on the model parameters

Mixture in 2D



$\Sigma_{1:3} = \sigma^2 I$

Mixture of Gaussians

$$f(x) = \sum_{k=1}^{K} \pi_k f_k(x)$$

π_k: mixture proportion

$f_k(x)$: component $N(\mu_k, \Sigma_k)$

$f(x)$ = a density on $\mathbb{R}^d$

π = a distribution on $1:K$

$\pi = (\pi_1, \dots \pi_K)$

$\sum_{k=1}^{K} \pi_k = 1$, $\pi_{1:K} \geq 0$

Sampling $x \sim f$

1. $k \sim \pi$ choose a component
2. sample $x \sim f_k$

Output x

$K=2$