annotated version

Machine Learning Course - CS-433

K-Means Clustering

Nov 5, 2019

Last updated on: November 5, 2019

EPFL

Clustering

Clusters are groups of points whose inter-point distances are small compared to the distances outside the cluster.

$x_1, \ldots, x_N$

The goal is to find "prototype" points $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2,, \ldots, \boldsymbol{\mu}_K$ and cluster assignments $z_n \in \{1, 2, \ldots, K\}$ for all $n = 1, 2, \ldots, N$ data vectors $\mathbf{x}_n \in \mathbb{R}^D$.

$\mathbb{R}^D$

specify $K = \#$ groups

K-means clustering

Assume K is known.

$$z_{nk} = \begin{cases} 1 & \text{if } n \text{ assigned to cluster } k \\ 0 & \text{otherwise} \end{cases}$$

$$\min_{\mathbf{z}, \boldsymbol{\mu}} \left(\mathcal{L}(\mathbf{z}, \boldsymbol{\mu}) = \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|_2^2 \right)$$

distance to own representative

$$\text{s.t. } \boldsymbol{\mu}_k \in \mathbb{R}^D, z_{nk} \in \{0, 1\}, \sum_{k=1}^{K} z_{nk} = 1, \forall n$$

$$\text{where } \mathbf{z}_n = [z_{n1}, z_{n2}, \ldots, z_{nK}]^\top$$
$$\mathbf{z} = [\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_N]^\top$$
$$\boldsymbol{\mu} = [\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_K]^\top$$

Is this optimization problem easy?

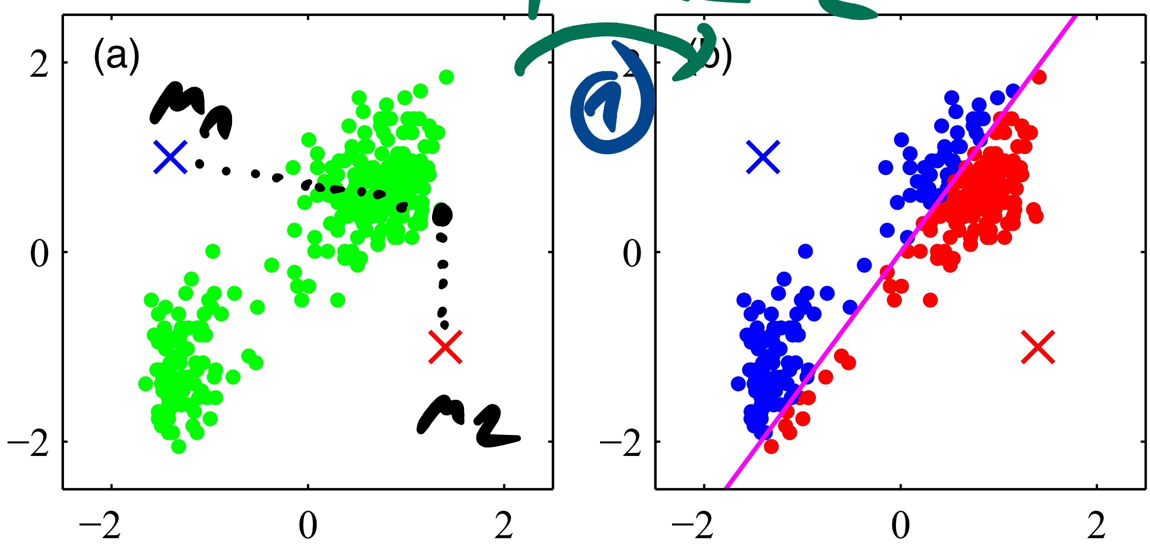
non-convex

Algorithm: Initialize $\boldsymbol{\mu}_k \; \forall k$,
then iterate:

1. For all n, compute $\mathbf{z}_n$ given $\boldsymbol{\mu}$.
2. For all k, compute $\boldsymbol{\mu}_k$ given $\mathbf{z}$.

Step 1: For all n, compute $\mathbf{z}_n$ given $\boldsymbol{\mu}$.

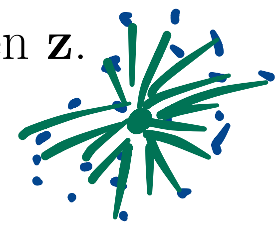
update z



1 assign to closest μ_k

$$z_{nk} = \begin{cases} 1 & \text{if } k = \arg\min_{j=1,2,\ldots K} \|\mathbf{x}_n - \boldsymbol{\mu}_j\|_2^2 \\ 0 & \text{otherwise} \end{cases}$$

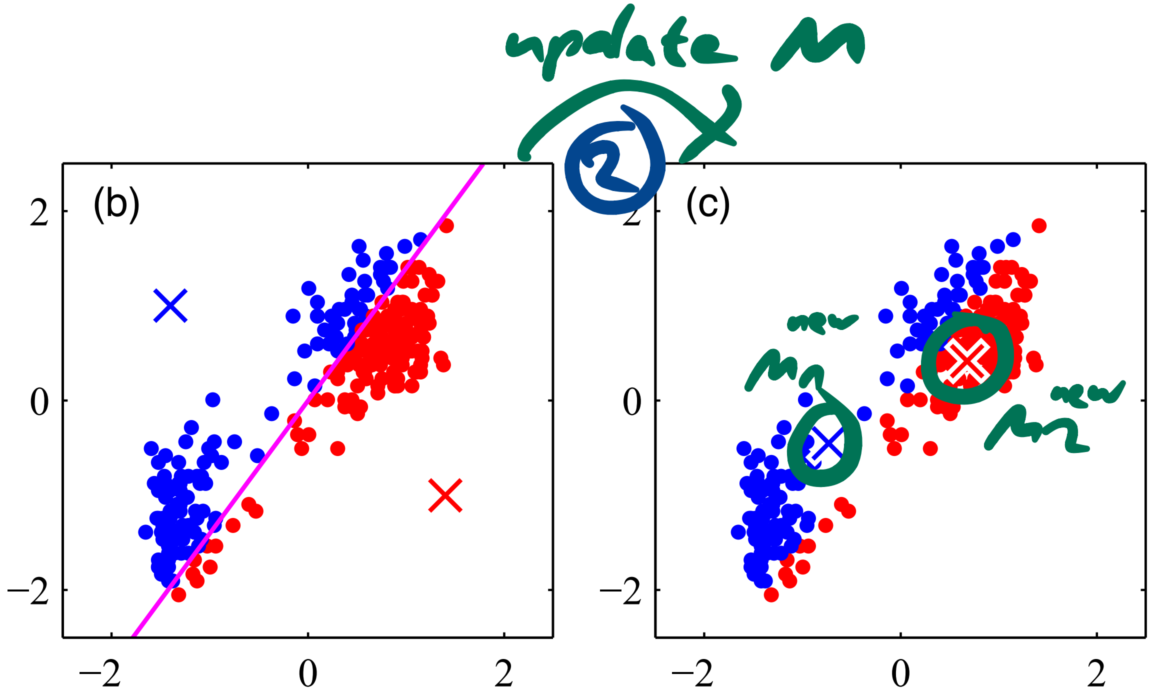
group k



2 **Step 2:** For all k, compute $\boldsymbol{\mu}_k$ given $\mathbf{z}$.
Take derivative w.r.t. $\boldsymbol{\mu}_k$ to get:

$$\boldsymbol{\mu}_k := \frac{\sum_{n=1}^{N} z_{nk}\mathbf{x}_n}{\sum_{n=1}^{N} z_{nk}}$$

Hence, the name 'K-means'.



Summary of K-means

Initialize $\boldsymbol{\mu}_k \, \forall k$, then iterate:

1. For all n, compute $\mathbf{z}_n$ given $\boldsymbol{\mu}$.

$$z_{nk} = \begin{cases} 1 & \text{if } k = \arg\min_j \underbrace{\|\mathbf{x}_n - \boldsymbol{\mu}_j\|_2^2}_{\text{cost } \mathcal{O}(D)} \\ 0 & \text{otherwise} \end{cases}$$

$\mathcal{O}(N \cdot K \cdot D)$

2. For all k, compute $\boldsymbol{\mu}_k$ given $\mathbf{z}$.

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^{N} z_{nk}\mathbf{x}_n}{\sum_{n=1}^{N} z_{nk}}$$

$\mathcal{O}(N \cdot K \cdot D)$

iterations

Convergence to a local optimum is assured since each step decreases the cost (see Bishop, Exercise 9.1).

$\mathcal{L} \geq 0$

$\nabla_{\mu} \mathcal{L}(z, \mu) \stackrel{!}{=} 0$

Coordinate descent

K-means is a coordinate descent algorithm, where, to find $\min_{z,\mu} \mathcal{L}(\mathbf{z}, \boldsymbol{\mu})$, we start with some $\boldsymbol{\mu}^{(0)}$ and repeat the following:

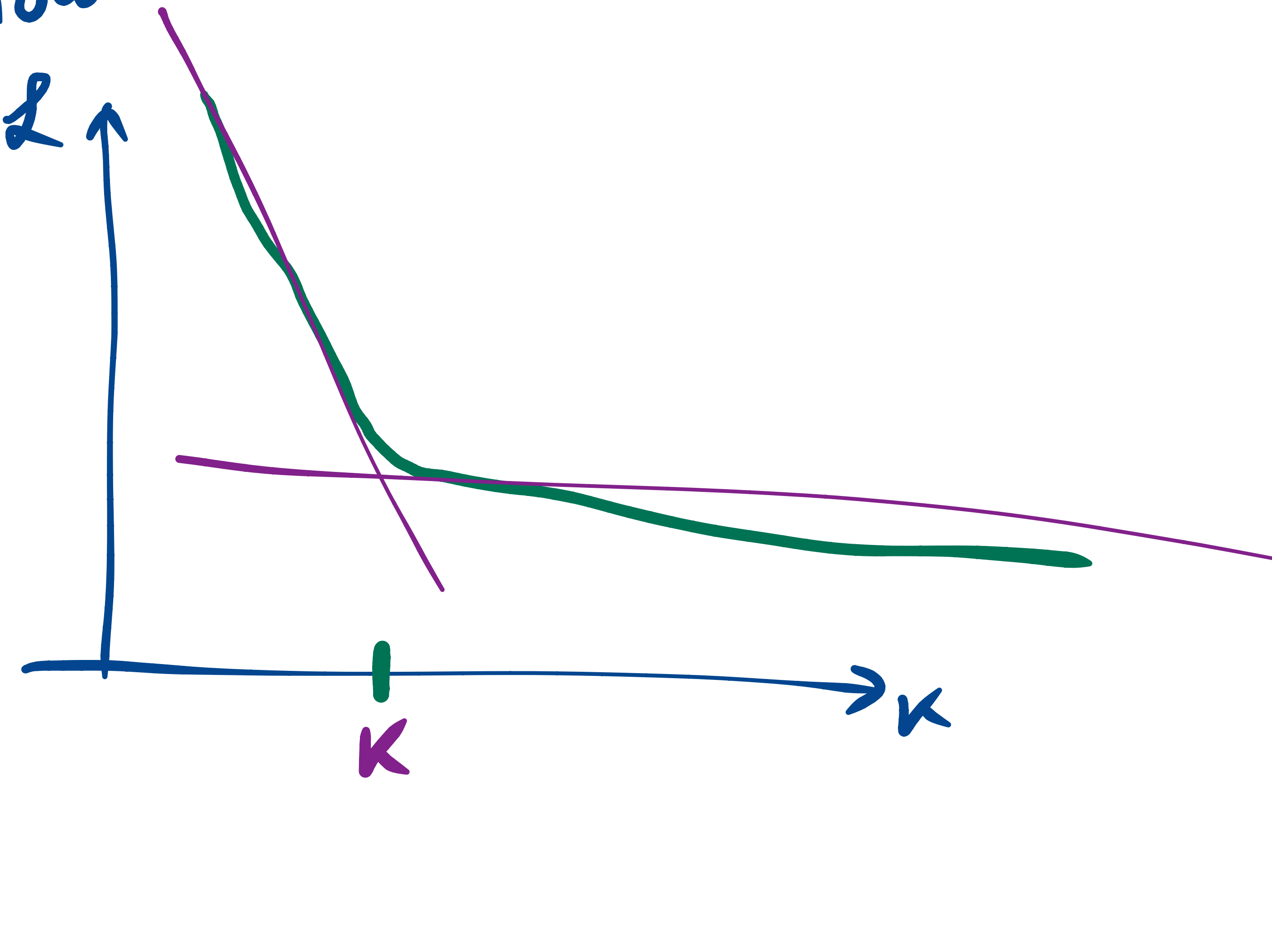
„Block-coordinate descent"

1. $$\mathbf{z}^{(t+1)} := \arg\min_{z} \mathcal{L}(\mathbf{z}, \boldsymbol{\mu}^{(t)})$$

2. $$\boldsymbol{\mu}^{(t+1)} := \arg\min_{\mu} \mathcal{L}(\mathbf{z}^{(t+1)}, \boldsymbol{\mu})$$

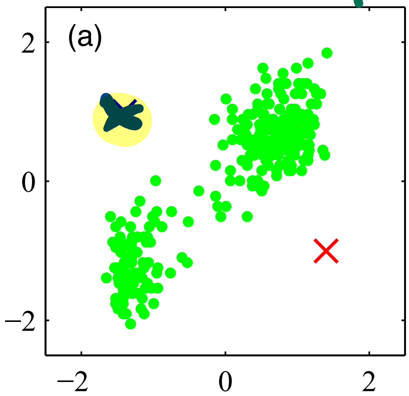
$\nabla_\mu \mathcal{L} \overset{!}{=} 0$

How to set K ?

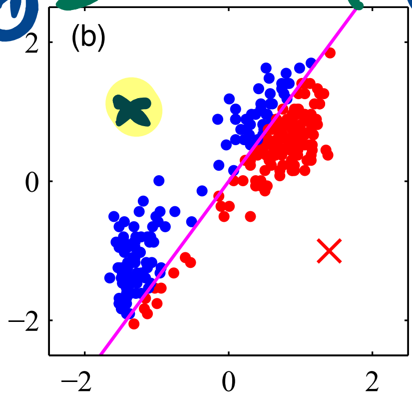


Examples

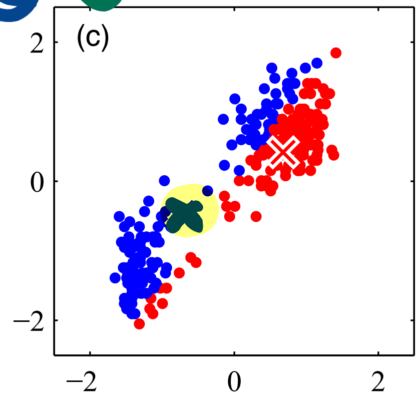
K-means for the “old-faithful” dataset (Bishop’s Figure 9.1)



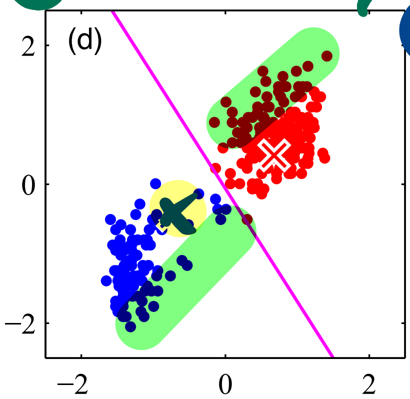
(e) Iteration 0



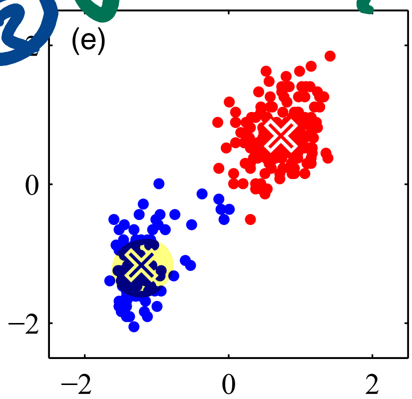
(f) Iteration 1



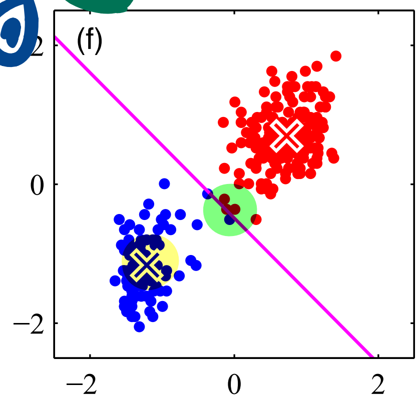
(g) Iteration 1



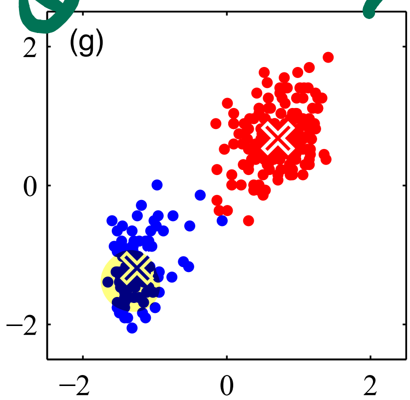
(h) Iteration 2



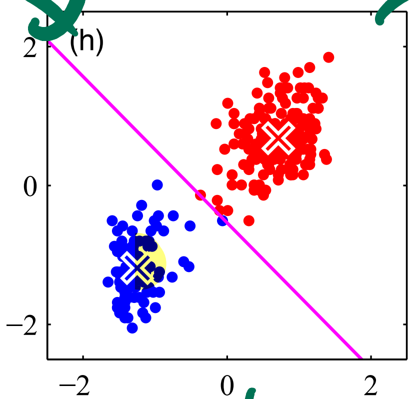
(i) Iteration 2



(j) Iteration 3

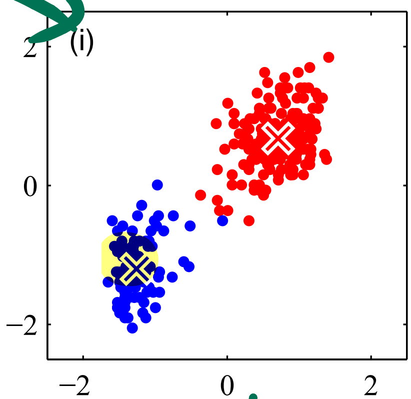


(k) Iteration 3



no change

(l) Iteration 4



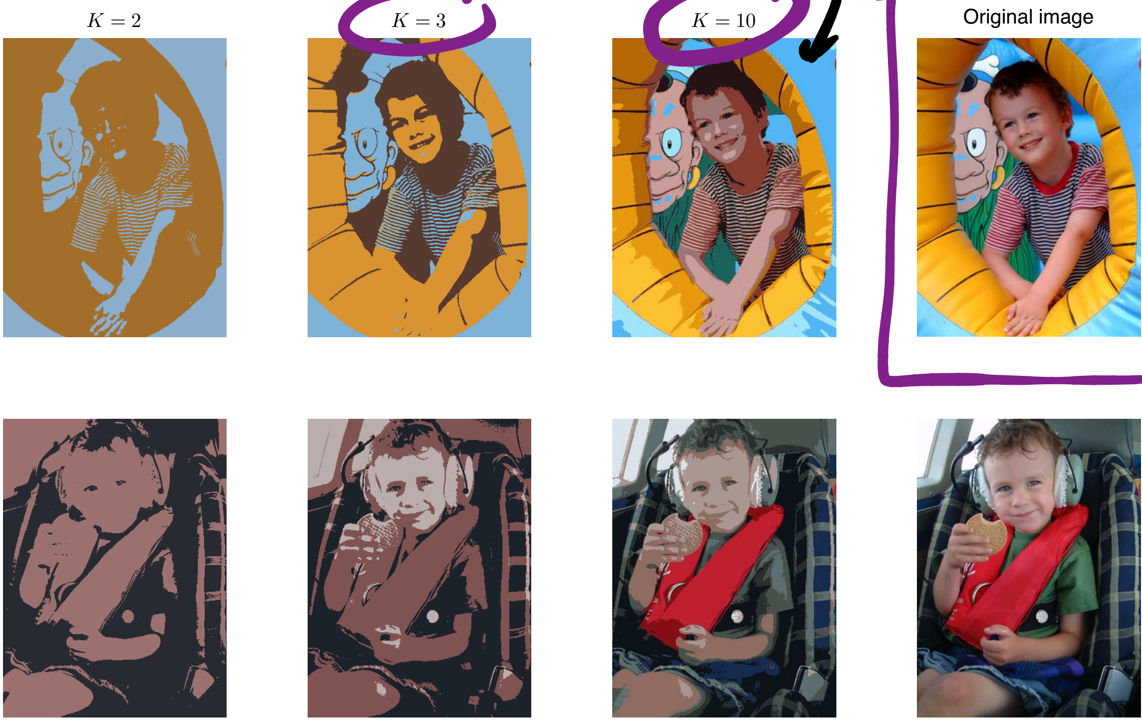
no change

(m) Iteration 4

Data compression for images (this is also known as vector quantization).

$\mu_1 = \boxed{\begin{matrix} k \\ 5 \\ 3 \end{matrix}}$

200



500

100k

Probabilistic model for K-means

Likelihood of X given parameters μ, z

$$p(x_n \mid \mu, z) = \prod_{n=1}^{N} \mathcal{N}(x_n \mid \mu_k, I)$$

$$p(X \mid \mu, z)$$

← assignment for x_n

$$= \prod_{n=1}^{N} \prod_{k=1}^{K} \mathcal{N}(x_n \mid \mu_k, I)^{z_{nk}}$$

$$= \prod_{n=1}^{N} \prod_{k=1}^{N} c \cdot e^{-\frac{1}{2}\|x_n - \mu_k\|^2 \cdot z_{nk}}$$

$$-\log p(X \mid \mu, z) = \sum_n \sum_k \frac{1}{2}\|x_n - \mu_k\|^2 z_{nk} + c'$$

$$= \mathcal{L}(\mu, z)$$

K-means as a Matrix Factorization

Recall the objective

$$\min_{\mathbf{z},\boldsymbol{\mu}} \ \mathcal{L}(\mathbf{z},\boldsymbol{\mu}) = \sum_{n=1}^{N}\sum_{k=1}^{K} z_{nk}\|\mathbf{x}_n - \boldsymbol{\mu}_k\|_2^2$$
$$= \|\mathbf{X}^\top - \mathbf{M}\mathbf{Z}^\top\|_{\text{Frob}}^2$$

entry-wise

$f(M \cdot Z^T)$

$$\text{s.t. } \boldsymbol{\mu}_k \in \mathbb{R}^D,$$
$$z_{nk} \in \{0,1\}, \ \sum_{k=1}^{K} z_{nk} = 1.$$

$$\underset{D\cdot K}{M} = \begin{pmatrix} | & & | \\ \mu_1 & \dots & \mu_K \\ | & & | \end{pmatrix}$$

$$\underset{N\cdot K}{Z} = \begin{pmatrix} — z_1 — \\ — z_N — \end{pmatrix}$$

Issues with K-means

1. Computation can be heavy for large N, D and K.
2. Clusters are forced to be spherical (e.g. cannot be elliptical).
3. Each example can belong to only one cluster ("hard" cluster assignments).