Machine Learning Course - CS-433

K-Means Clustering

Nov 15, 2018

minor changes by Martin Jaggi 2016

minor changes by Martin Jaggi 2017

changes by Ruediger Urbanke 2018

Last updated: November 13, 2018

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Clustering

Clusters are groups of points so that the distances within the clusters (groups) are small compared to the distances between the clusters (groups).
The clusters are typically defined by finding some "centers" $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2,, \ldots, \boldsymbol{\mu}_K$. Each such center then defines one cluster via an assignement mapping $z_n \in \{1, 2, \ldots, K\}$ for all N data vectors $\mathbf{x}_n \in \mathbb{R}^D$.

K-means clustering

The perhaps best-known clustering algorithm is the K-means algorithm. Assume that K is known. The optimization that leads to the clusters is the following:

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

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

$$\text{where } \mathbf{z}_n = [z_{n1}, z_{n2}, \ldots, z_{nK}]^\top$$

$$\mathbf{z} = [\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_N]^\top$$

$$\boldsymbol{\mu} = [\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_K]^\top$$

Let us discuss this in some more detail. For fixed centers $\boldsymbol{\mu}_k$, the cost is minimized if we map each sample to its nearest center, where we measure distance in terms of Euclidean distance. This is accomplished by means of the indicator variables z_{nk}, $z_{nk} \in \{0, 1\}$, where for each sample n the sum

$\sum_{k=1}^{K} z_{nk} = 1$. In words, each sample is assigned exactly to one center (cluster) and this is indicated by setting the corresponding indicator variable s_{nk} to 1 and all the other ones $z_{nk'}$ to 0. In addition we still have to minimize over the best choices for the centers.
The above description leads very naturally to an algorithm.
Algorithm: Initialize $\boldsymbol{\mu}_k \, \forall k$.
Iterate:

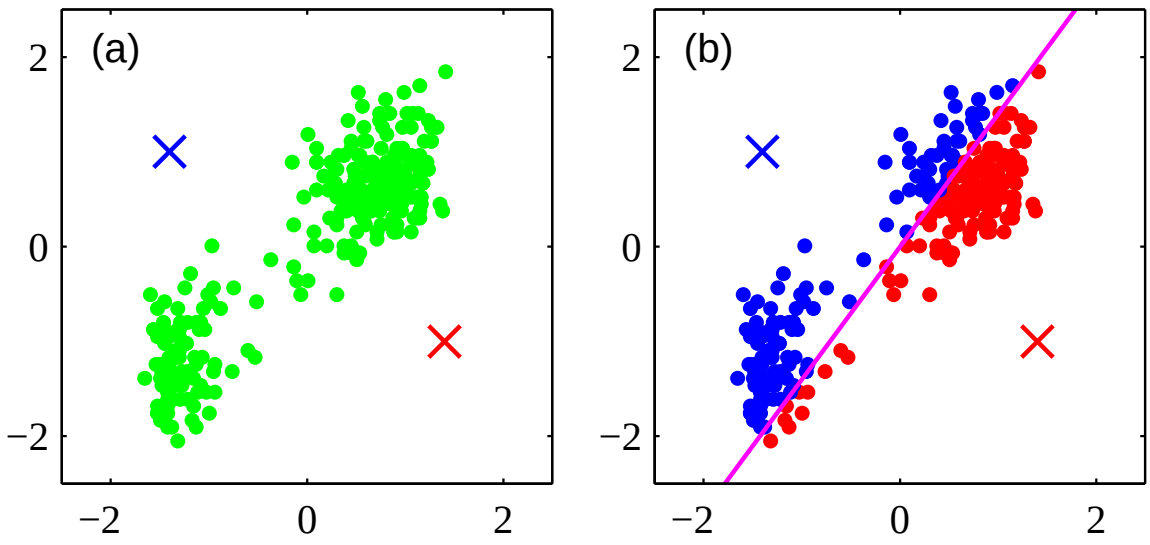
1. For all n, compute $\mathbf{z}_n$ given $\boldsymbol{\mu}$.

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

As we discussed, step (1) is clear. Once we fixed the centers the best assignments is to map each point to the nearest center. But (2) is equally natural. If we have fixed assignements then it is easy to compute the best centers. This can be formally seen by taking derivatives of the cost function w.r.t. $\boldsymbol{\mu}_k$ and solving for the centers. We get:

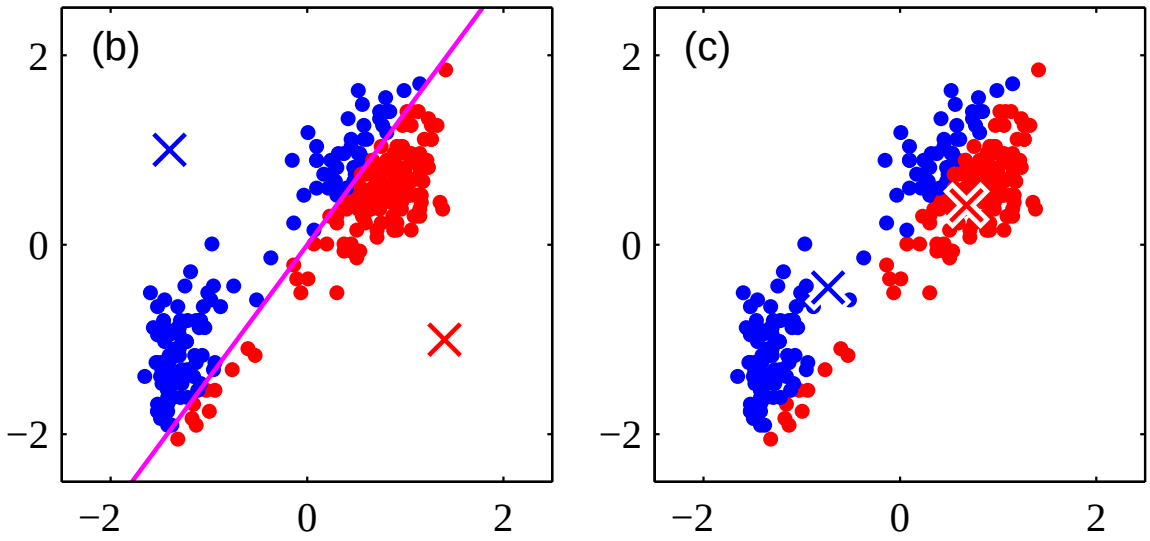
$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^{N} z_{nk} \mathbf{x}_n}{\sum_{n=1}^{N} z_{nk}}$$

Hence, the best centers are just the means of each cluster. Hence, the name 'K-means'.
Let us look at the following example. Figure (a) shows a set of points. We have $K = 2$ and two initial centers are shown.
Step 1: Given the centers, in the first step we compute the optimal assignments. I.e., for all n, compute $\mathbf{z}_n$ given $\boldsymbol{\mu}$. The result is shown in Figure (b) via a color-coding. We map each point to its closest center.



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

Step 2: In the second step we now freeze the assignments and re-compute the best centers for each cluster. I.e., for all k, compute $\boldsymbol{\mu}_k$ given $\mathbf{z}$. As we mentioned, this computation



is just the computation of the mean of each cluster. T

Summary of K-means

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

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

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

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

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

Convergence to a local optimum is assured since each step decreases the cost (see Bishop, Exercise 9.1). But note that we are not guaranteed to reach the globally optimal solution with this iterative algorithm.

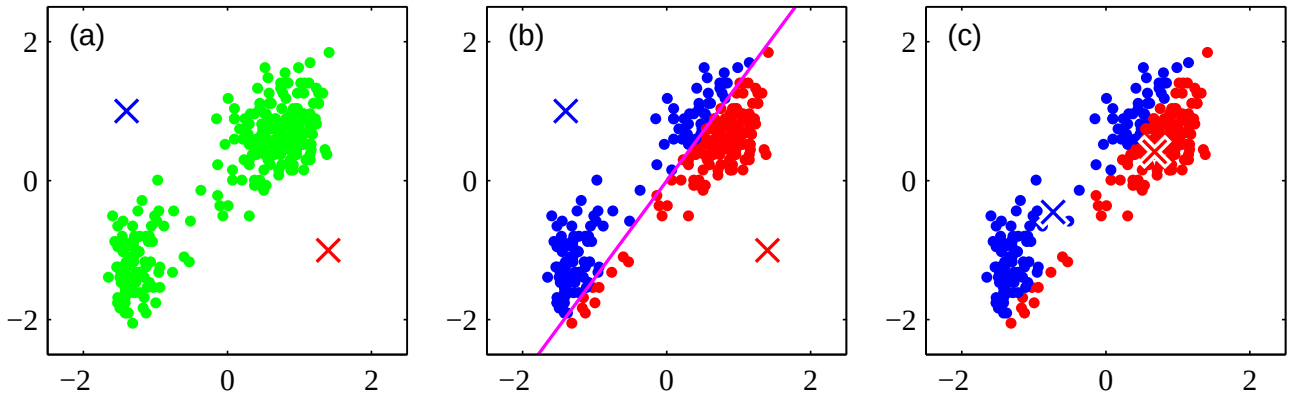
Coordinate descent

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

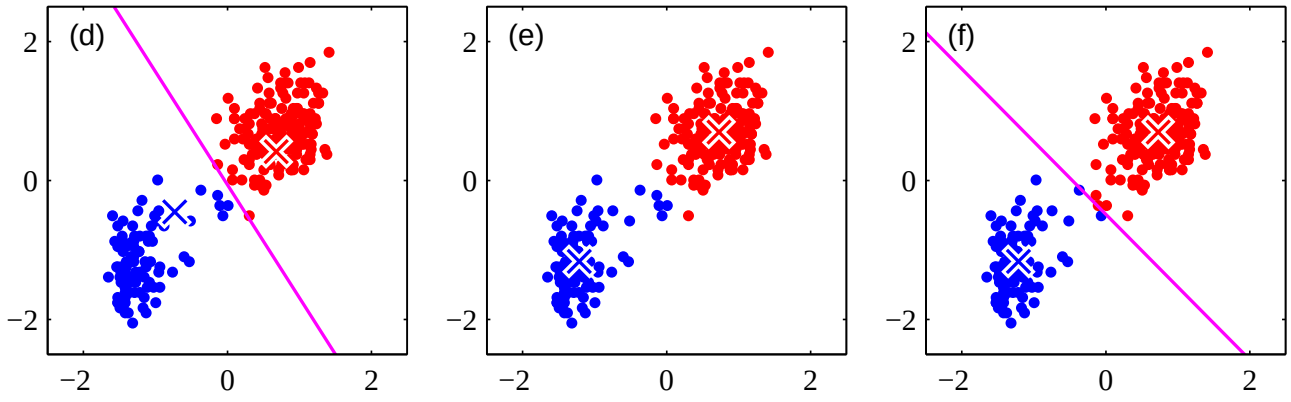
$$\mathbf{z}^{(t+1)} := \arg\min_{\boldsymbol{z}} \; \mathcal{L}(\mathbf{z}, \boldsymbol{\mu}^{(t)})$$
$$\boldsymbol{\mu}^{(t+1)} := \arg\min_{\boldsymbol{\mu}} \; \mathcal{L}(\mathbf{z}^{(t+1)}, \boldsymbol{\mu})$$

Examples

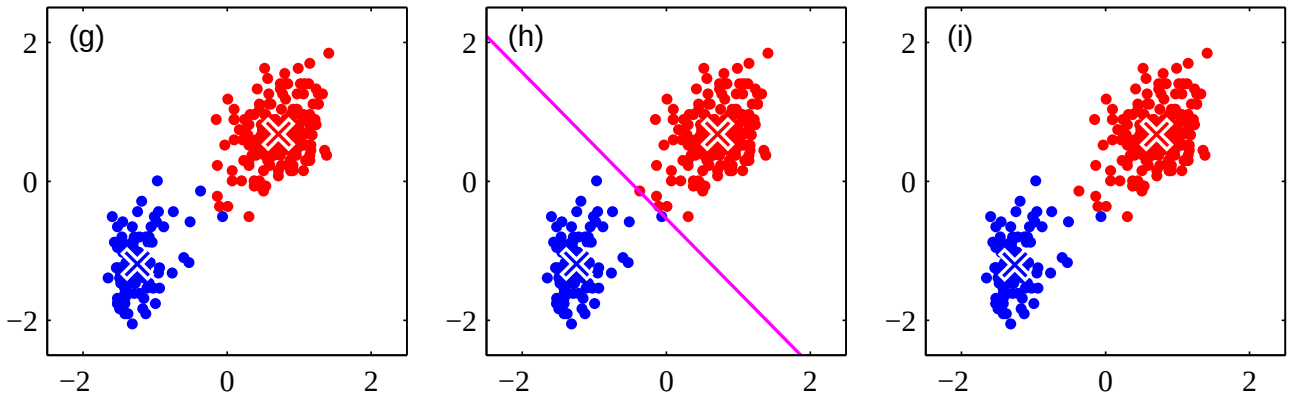
K-means for the "old-faithful" dataset (Bishop's Figure 9.1)



(e) Iteration 0 (f) Iteration 1 (g) Iteration 1



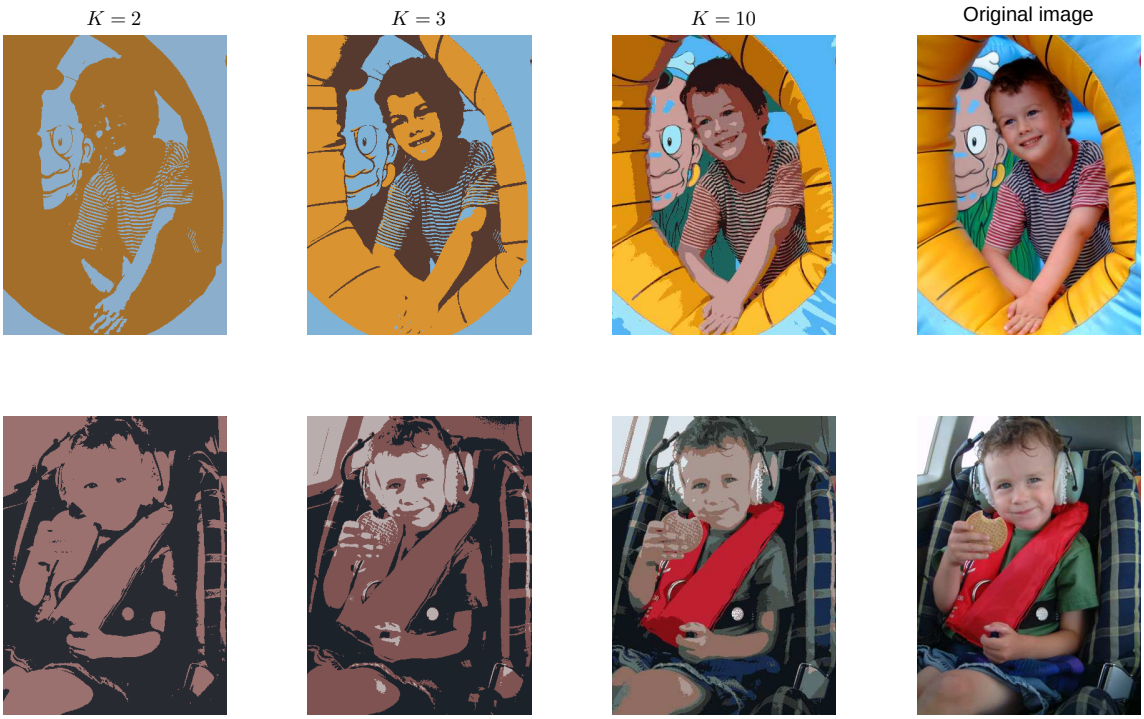
(h) Iteration 2 (i) Iteration 2 (j) Iteration 3



(k) Iteration 3 (l) Iteration 4 (m) Iteration 4

To consider a second example. In the figures below the colors

contained in the photo were quantized to a small number $K = 2, 3, \cdots$. And these "center" colors were chosen by clustering. This is a form of data compression. It is known the signal processing community as vector quantization.



Probabilistic model for K-means

So far we have presented K-means using a geometric reasoning. But as so many other models in ML, it also has a probabilistic interpretation.

Assume that, conditioned that a point is associated to cluster k, we consider it a sample from the D-dimensional Gaussian with mean $\boldsymbol{\mu}_k$ and covariance matrix $\mathbf{I}$. I.e., the likelihood of a sample $\mathbf{x}$ given the cluster assignment $\mathbf{z}$ and the centers

$\boldsymbol{\mu} = \{\boldsymbol{\mu}_K\}_{k=1}^K$ is

$$p(\mathbf{x}|\mathbf{z}, \boldsymbol{\mu}) = \prod_{k=1}^{K} \left[\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \mathbf{I})\right]^{z_k}.$$

Then the likelihood associated to a whole data set $S_{\text{train}} = \{\mathbf{x}_k\}_{k=1}^K$.

$$p(\mathbf{X}|\mathbf{z}, \boldsymbol{\mu}) = \prod_{n=1}^{N} \prod_{k=1}^{K} \left[\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \mathbf{I})\right]^{z_{nk}}.$$

Taking, as always, the log and multiplying with -1 in order to get a minimization problem this gives us

$$-\log p(\mathbf{x}|\mathbf{z}, \boldsymbol{\mu}) = \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} \|\mathbf{x} - \boldsymbol{\mu}\|_2^2,$$

which corresponds exactly to our original formulation.

K-means as a Matrix Factorization

K-means can be interpreted as a matrix factorization problem. We will learn much more about the matrix factorization problem in a future lecture but the set-up is very natural.

We can write:

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

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

As always $\mathbf{X}$ is the $N \times D$ data matrix whose rows are the individual featue vectors. Hence $\mathbf{X}^\top$ has dimensions $D \times N$. The matrix $\mathbf{Z}$ is the $N \times K$ matrix whose rows are the indicator vectors $\mathbf{z}_n$. And $\mathbf{M}$ is the real-valued matrix of dimension $D \times K$ whose columns are the centers $\boldsymbol{\mu}_k$. Note that $\mathbf{M}\mathbf{Z}^\top$ is a $D \times N$ matrix that contains in its n-th column one of the K centers $\boldsymbol{\mu}_k$. Therefore $\|\mathbf{X}^\top - \mathbf{M}\mathbf{Z}^\top\|$ is a $D \times N$ matrix that contains in its n-th column the difference between the n-th sample and the center that we assigned it to. The Frobenius norm now squares all these entries and sums them up. This is hence the total distance. In order to have a short Frobenius norm our task hence is to "factorize" the matrix $\mathbf{X}^\top$ in the form $\mathbf{M}\mathbf{Z}^\top$ where the matrix $\mathbf{Z}$ is restricted to entries from $\{0,1\}$ and each row of $\mathbf{Z}$ contains exactly a single 1. This is why we say that this is a matrix factorization formulation.

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).