Machine Learning

K-means Clustering

Kia Nazarpour

Context

1. Often times we need to analyse data for which we do not have their labels.
2. How can we find any structure in a collection of unlabelled data?
3. Clustering is an established category of methods for organising objects into groups whose members are similar in some way.

Learning Outcomes

1. Understand the key motivations behind clustering and its challenges.
2. Implement the K-means algorithm.
3. Solve the maths of the K-means algorithm.
4. Analyse when/how/why the simple K-means method can fail.
5. Understand the notion of hard and soft clustering, introducing briefly the notion of mixture models.

References:

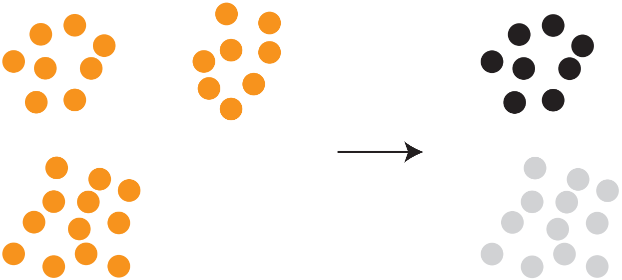
1. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2008. (Section 9.1)
2. Hastie *et al.*, *The Elements of Statistical Learning*, Springer, 2017. (Section 14.3.6)

Problem Statement

Aim: Identify clusters of data points in a multi-dimensional space.

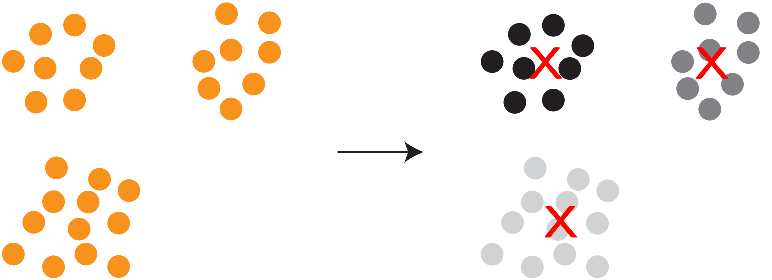
- Suppose we have data set $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$ as N observations of a d-dimensional variable $\mathbf{x}$.
- Our goal is to partition data set into a *known* number of clusters, say K.

Problem Statement



$K = 3$

Problem Statement



We can formalise the idea by introducing d-dimensional vectors $\boldsymbol{\mu}_{k\in\{1,\cdots,K\}}$ to represent each cluster.

The vectors $\boldsymbol{\mu}_{1:3}$ are shown by X.

Problem Formulation

Specific goal: Given a K, find an assignment of data points to clusters and the set of vectors $\{\boldsymbol{\mu}_k\}$ to represent these cluster.

The assignment rule ($r_{nk} = 1$ if $\mathbf{x}_n$ is in cluster k) and all $\boldsymbol{\mu}_k$s are unknown.

Ideally, we want the points in each cluster to be close to each other and far from points in other clusters.

Problem Formulation

Specific goal: Given a K, find an assignment of data points to clusters and the set of vectors $\{\boldsymbol{\mu}_k\}$ to represent these cluster.

The assignment rule ($r_{nk} = 1$ if $\mathbf{x}_n$ is in cluster k) and all $\boldsymbol{\mu}_k$s are unknown.

Ideally, we want the points in each cluster to be close to each other and far from points in other clusters.

A proposal: Minimise the *distortion function*, i.e., the sum of the squared distances of each data point to its closest vector $\boldsymbol{\mu}_k$.

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

K-means Solution

A proposal: Minimise the *distortion function*, i.e., the sum of the squared distances of each data point to its closest vector $\boldsymbol{\mu}_k$.

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

1. Given K, randomly select $\boldsymbol{\mu}_{k=1,\cdots,K}$
2. Minimise J with respect to r_{nk}, keeping the $\boldsymbol{\mu}_k$ fixed.
3. Minimise J with respect to $\boldsymbol{\mu}_k$, keeping the r_{nk} fixed.
4. Repeat steps 2 (*Expectation*) and 3 (*Maximisation*) steps until convergence, that is, $\Delta J < \epsilon$.

K-means Solution

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

Step 2: Minimise J with respect to r_{nk}, keeping the $\boldsymbol{\mu}_k$ fixed.

K-means Solution

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

Step 2: Minimise J with respect to r_{nk}, keeping the $\boldsymbol{\mu}_k$ fixed.

J is a linear function of r_{nk}. Also terms with n are independent.

K-means Solution

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

Step 2: Minimise J with respect to r_{nk}, keeping the $\boldsymbol{\mu}_k$ fixed.

J is a linear function of r_{nk}. Also terms with n are independent.

Simply, $r_{nk} = 1$ for the closest cluster k, i.e. whichever k that gives the smallest value of $\|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$.

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

K-means Solution

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

Step 3: Minimise J with respect to $\boldsymbol{\mu}_k$, keeping the r_{nk} fixed.

K-means Solution

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

Step 3: Minimise J with respect to $\boldsymbol{\mu}_k$, keeping the r_{nk} fixed.

J is a quadratic function of $\boldsymbol{\mu}_k$ and can be minimised by setting its derivative with respect to $\boldsymbol{\mu}_k$ to zero, that is $\dfrac{\delta J}{\delta \boldsymbol{\mu}_k} = 0$.

K-means Solution

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

Step 3: Minimise J with respect to $\boldsymbol{\mu}_k$, keeping the r_{nk} fixed.

J is a quadratic function of $\boldsymbol{\mu}_k$ and can be minimised by setting its derivative with respect to $\boldsymbol{\mu}_k$ to zero, that is $\dfrac{\delta J}{\delta \boldsymbol{\mu}_k} = 0$.

$$\begin{aligned}
\frac{\delta J}{\delta \boldsymbol{\mu}_k} &= \frac{\delta \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2}{\delta \boldsymbol{\mu}_k} = \sum_{n=1}^{N} r_{nk} \times (-1) \times 2(\mathbf{x}_n - \boldsymbol{\mu}_k) = 0 \\
&= \sum_{n=1}^{N} r_{nk} \mathbf{x}_n - \sum_{n=1}^{N} r_{nk} \boldsymbol{\mu}_k = 0
\end{aligned}$$

K-means Solution

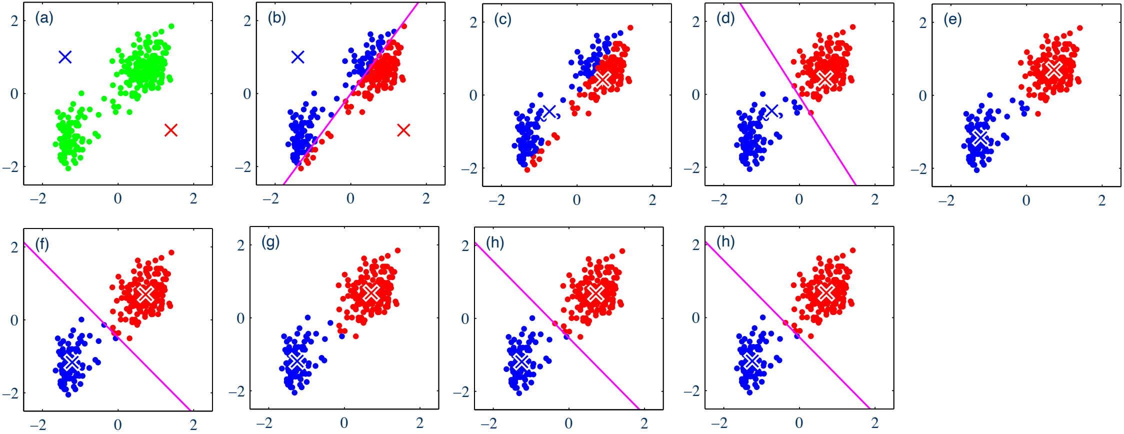
$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

Step 3: Minimise J with respect to $\boldsymbol{\mu}_k$, keeping the r_{nk} fixed.

J is a quadratic function of $\boldsymbol{\mu}_k$ and can be minimised by setting its derivative with respect to $\boldsymbol{\mu}_k$ to zero, that is $\dfrac{\delta J}{\delta \boldsymbol{\mu}_k} = 0$.

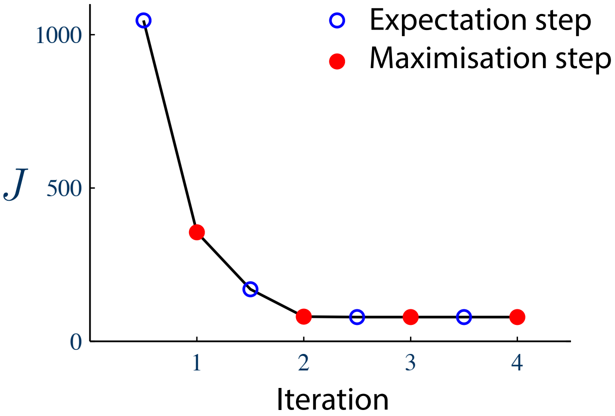
$$\begin{aligned}
\frac{\delta J}{\delta \boldsymbol{\mu}_k} &= \frac{\delta \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2}{\delta \boldsymbol{\mu}_k} = \sum_{n=1}^{N} r_{nk} \times (-1) \times 2(\mathbf{x}_n - \boldsymbol{\mu}_k) = 0 \\
&= \sum_{n=1}^{N} r_{nk} \mathbf{x}_n - \sum_{n=1}^{N} r_{nk} \boldsymbol{\mu}_k = 0 \quad \rightarrow \quad \boldsymbol{\mu}_k = \frac{\sum_n r_{nk} \mathbf{x}_n}{\sum_n r_{nk}}
\end{aligned}$$

K-means: An example



Bishop Figure 9.1

K-means: An example



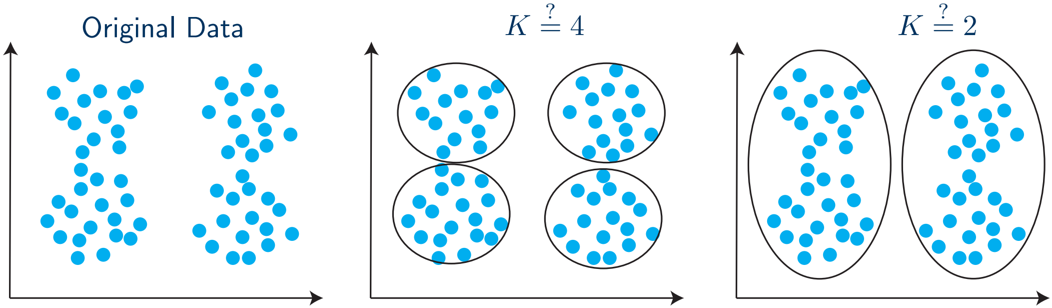
Bishop Figure 9.2

K-means for Image Segmentation and Compression

Original Image | $K = 2$ | $K = 3$ | $K = 10$

Bishop Figure 9.3

How to choose K?

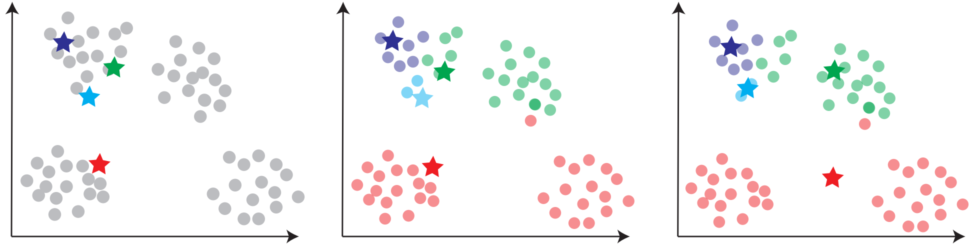


There are several methods for choosing K, including [but not limited to], using domain expertise, elbow and silhouette methods, and gap statistics*.

*Tibshirani *et al. J. R. Statist. Soc. B.* (2001) 63:411-423.

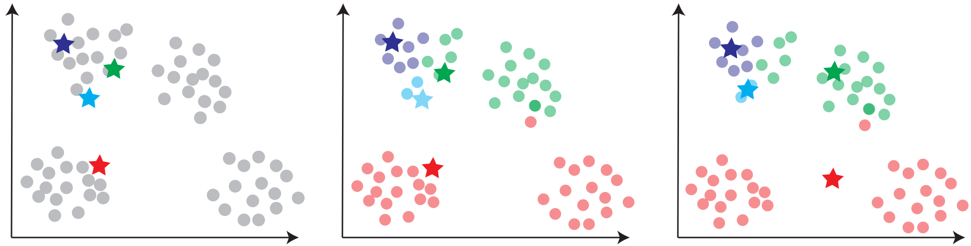
How to initialise $\boldsymbol{\mu}_k$

The K-means algorithm is sensitive to the initialisation of $\boldsymbol{\mu}_k$.



How to initialise $\boldsymbol{\mu}_k$

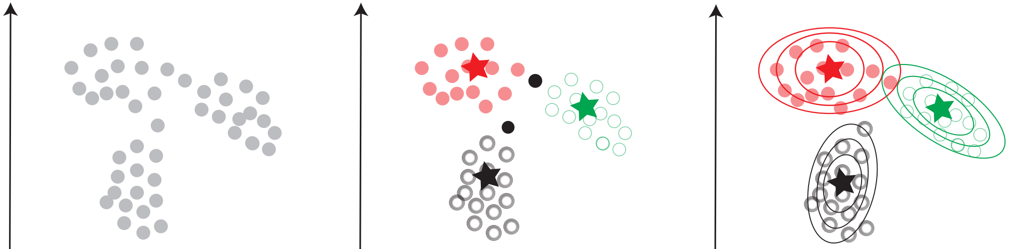
The K-means algorithm is sensitive to the initialisation of $\boldsymbol{\mu}_k$.



Methods of initialisation:

1. Random initialisation (the above case can happen!)
2. Often times, $\boldsymbol{\mu}_k$s are initialised to a subset of data (Forgy initialisation).
3. Repeat clustering for various initial and select the *best* set of $\boldsymbol{\mu}_k$s
4. K-means++ (Arthur and Vassilvitskii, 2007)

Hard assignment vs. Soft assignment



K-means: Summary

1. A simple unsupervised method that enables clustering of data
2. Poses no great computational complexity
3. Too crude to assume a cluster can be represented with a single point and a simple distance metric
4. Hard boundaries!
5. How to generalise it to models that can cluster data of various types and shapes!