Machine Learning

K-means Clustering

Hiroshi Shimodaira

March 2026

Ver. 1.0

Adapted from Kia Nazarpour's slides

Background

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.

Topics

- K-means clustering algorithm
- Mathematical formulation
- Convergence
- Computational complexity
- Voronoi diagram
- Practical issues / limitation with k-means

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)
3. Kevin Murphy, A PMR introduction (Section 21.3)

Problem Statement

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

- Suppose we have a data set $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$ as N observations of a d-dimensional vector $\mathbf{x}$.

Problem Statement

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

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

Problem Statement

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

- Suppose we have a data set $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$ as N observations of a d-dimensional vector $\mathbf{x}$.
- Our goal is to partition the data set into a *known* number of clusters, say K.
- In another word, we'd like to find the cluster label $y_n \in \{1, \ldots, K\}$ for each $\boldsymbol{x}_n$.

Problem Statement

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

- Suppose we have a data set $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$ as N observations of a d-dimensional vector $\mathbf{x}$.
- Our goal is to partition the data set into a *known* number of clusters, say K.
- In another word, we'd like to find the cluster label $y_n \in \{1, \ldots, K\}$ for each $\mathbf{x}_n$.
- This can be also expressed as

$$r_{nk} = \begin{cases} 1 & \text{if } \mathbf{x}_n \text{ is in cluster } k, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

K-means clustering algorithm

1. Pick K random points (from the data set) as cluster centre positions

K-means clustering algorithm

1. Pick K random points (from the data set) as cluster centre positions
2. Assign each point to its nearest centre*

K-means clustering algorithm

1. Pick K random points (from the data set) as cluster centre positions
2. Assign each point to its nearest centre*
3. Move each centre to mean of its assigned points

K-means clustering algorithm

1. Pick K random points (from the data set) as cluster centre positions
2. Assign each point to its nearest centre*
3. Move each centre to mean of its assigned points
4. If centres moved, `goto 2`, otherwise stop.

* In the unlikely event of a tie, break tie in some way.
For example, assign to the centre with smallest index in memory.

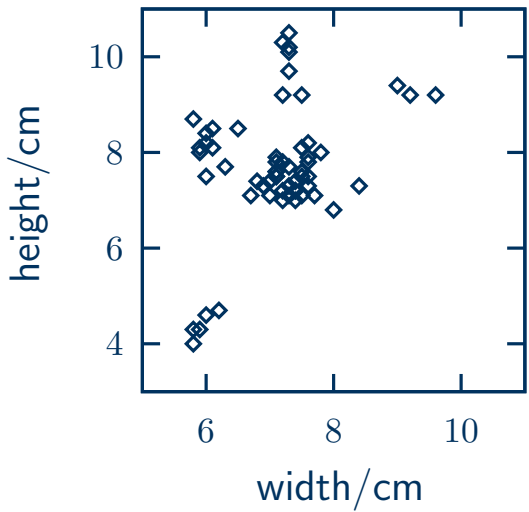
K-means clustering algorithm

1. Pick K random points (from the data set) as cluster centre positions
2. Assign each point to its nearest centre*
3. Move each centre to mean of its assigned points
4. If centres moved, `goto 2`, otherwise stop.

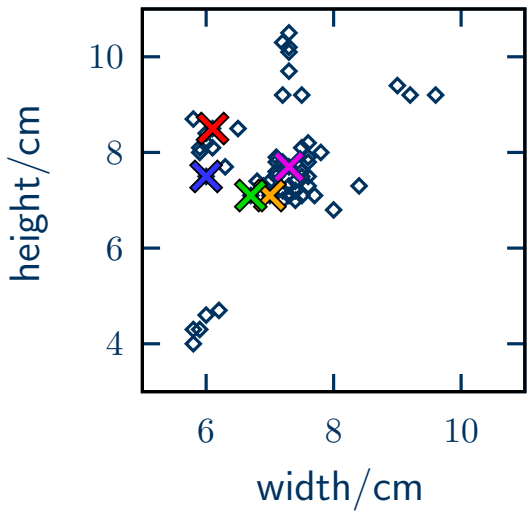
* In the unlikely event of a tie, break tie in some way.
For example, assign to the centre with smallest index in memory.

Computational complexity is $O(NKT)$, where T is the number of iterations.
NB: the original clustering problem has the complexity of (K^N)

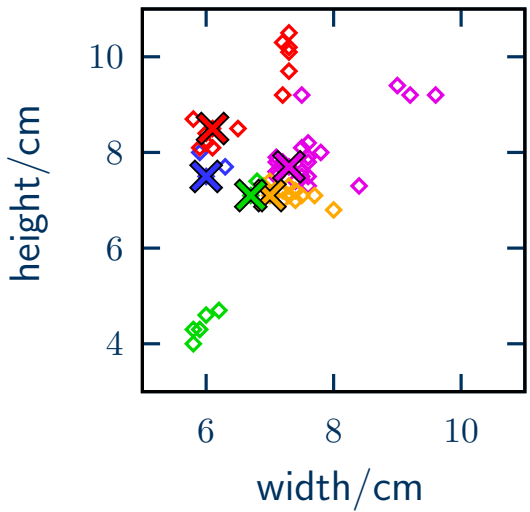
Toy example



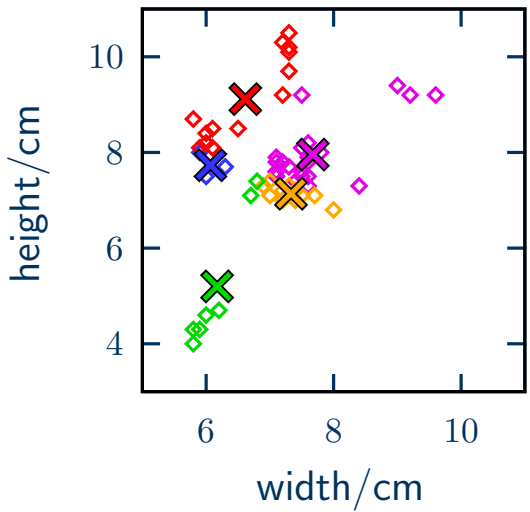
Toy example



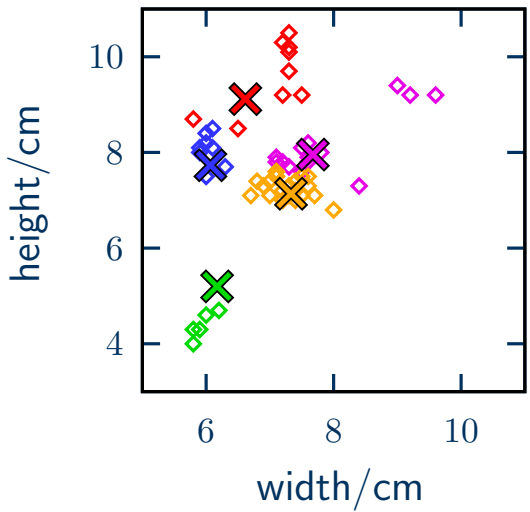
Toy example



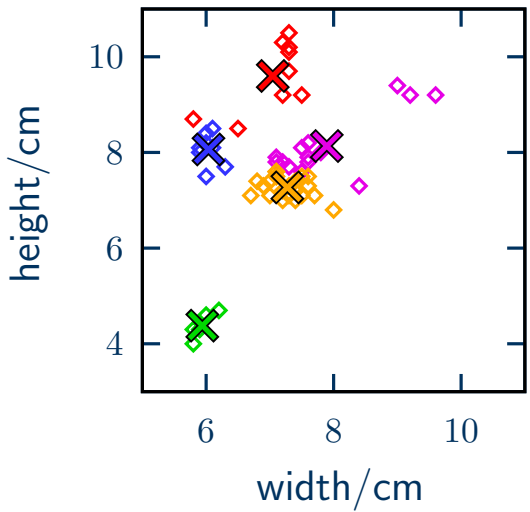
Toy example



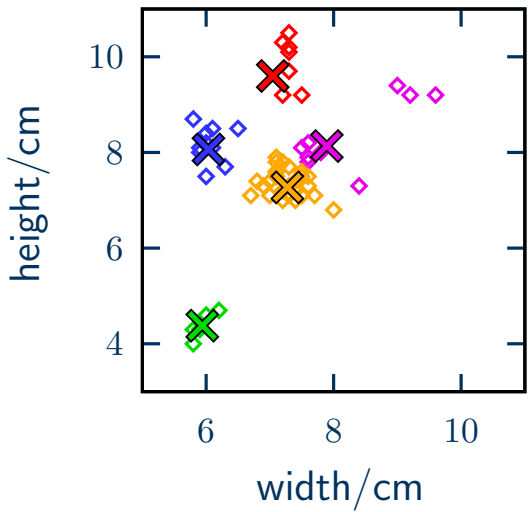
Toy example



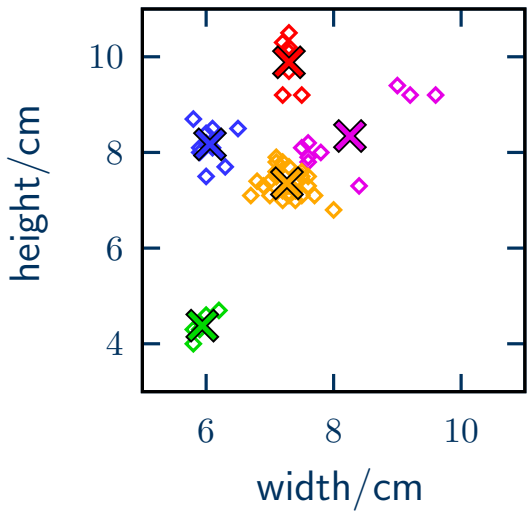
Toy example



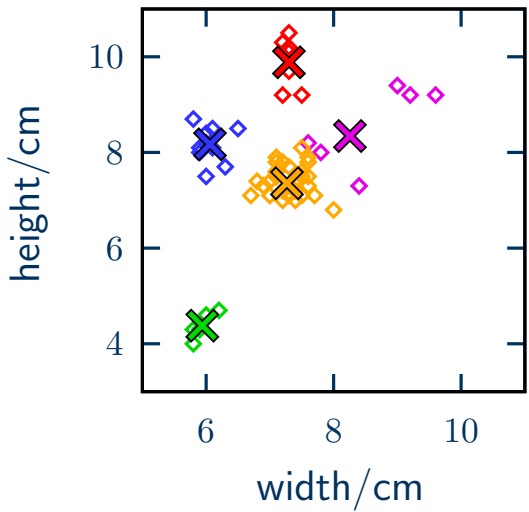
Toy example



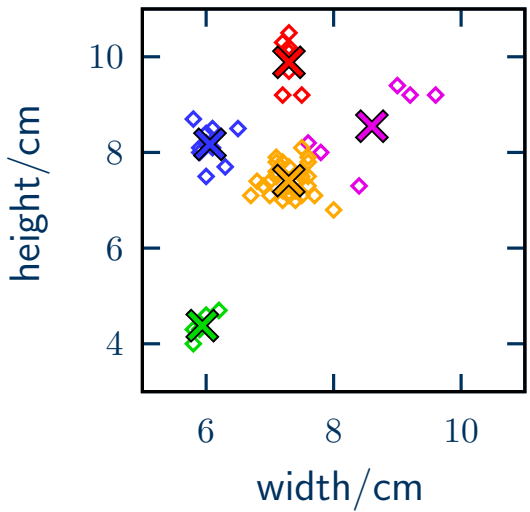
Toy example



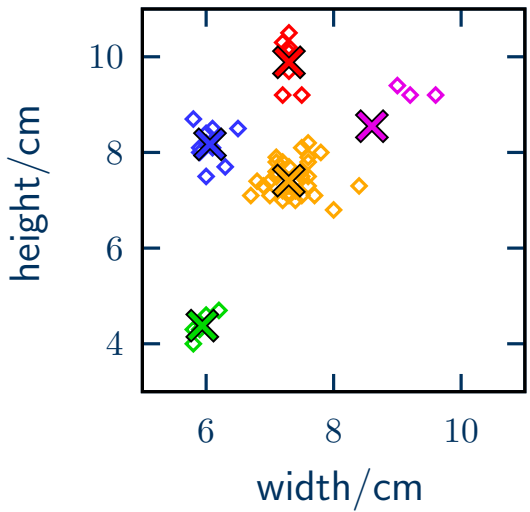
Toy example



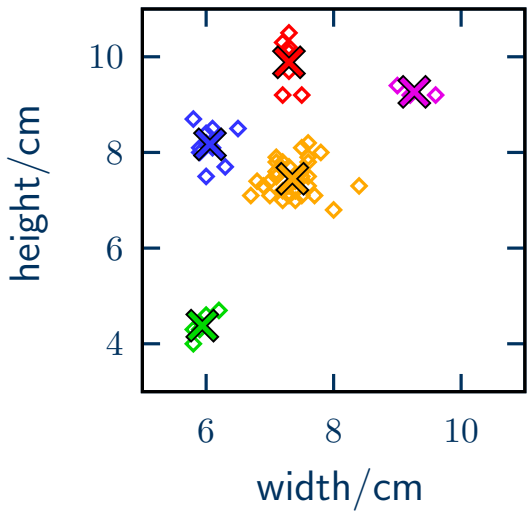
Toy example



Toy example



Toy example



Problem Formulation

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

Problem Formulation

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.

Problem Formulation

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

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.

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

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

Problem Formulation

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.

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

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

- The computation complexity will be $O(K^N)$ – difficult to solve

K-means clustering as an optimisation problem

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

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

We try to find a suboptimal solution using iterative algorithm:

1. Given K, randomly select $\boldsymbol{\mu}_k$ for $k = 1, \ldots, K$.
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 clustering – Step 2

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

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

K-means clustering – Step 2

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

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 clustering – Step 2

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

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 $\|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$.

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

K-means clustering – Step 3

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

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

K-means clustering – Step 3

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

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 clustering – Step 3

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

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 $\dfrac{\delta J}{\delta \boldsymbol{\mu}_k} = 0$.

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

$$= \sum_{n=1}^{N} r_{nk} \mathbf{x}_n - \sum_{n=1}^{N} r_{nk} \boldsymbol{\mu}_k = 0$$

K-means clustering – Step 3

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

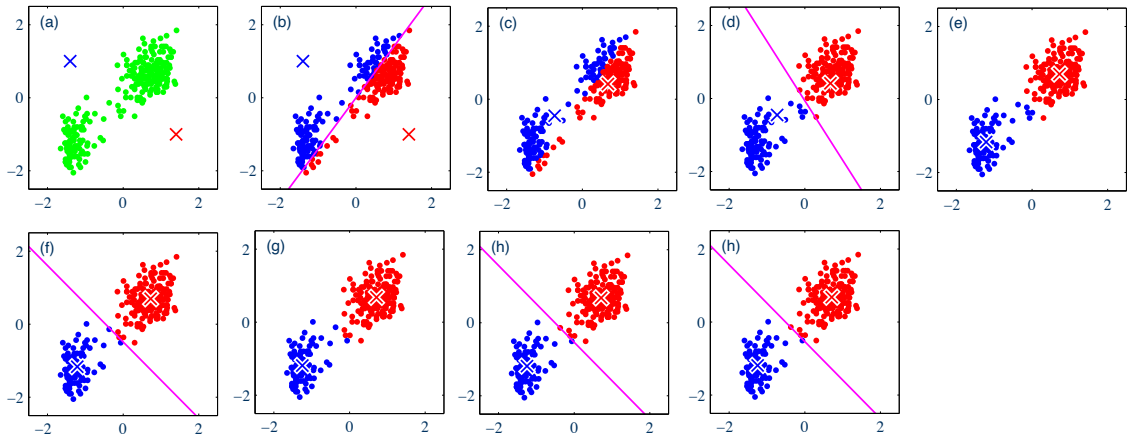
Step 3: Minimise J with respect to $\boldsymbol{\mu}_k$, keeping the r_{nk} fixed.

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 $\dfrac{\delta J}{\delta \boldsymbol{\mu}_k} = 0$.

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

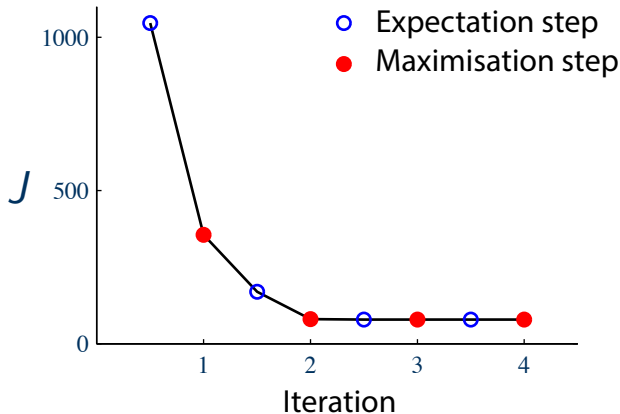
$$= \sum_{n=1}^{N} r_{nk} \mathbf{x}_n - \sum_{n=1}^{N} r_{nk} \boldsymbol{\mu}_k = 0 \qquad \rightarrow \qquad \boldsymbol{\mu}_k = \frac{\sum_n r_{nk} \mathbf{x}_n}{\sum_n r_{nk}}$$

K-means: An example



Bishop Figure 9.1

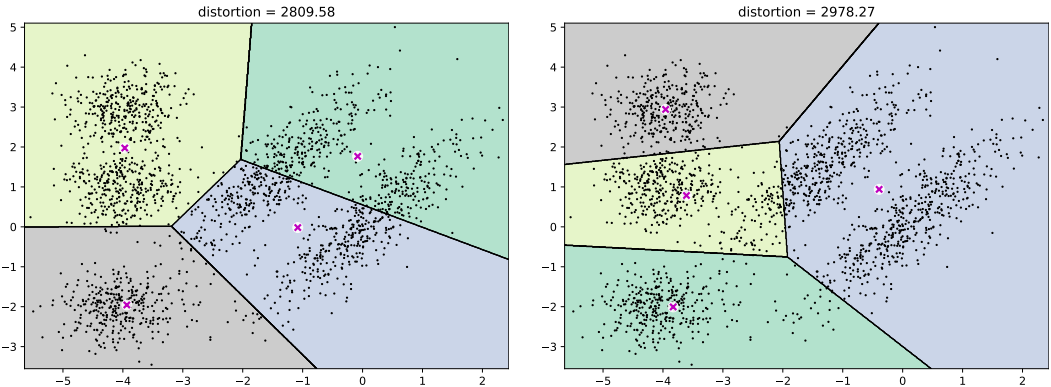
K-means: An example



Bishop Figure 9.2

Voronoi diagram/tessellation

K-means clustering partitions a space into partitions with piecewise hyperplanes.



Credit: K. Murphy, PMR An introduction.

The figures were generated by kmeans_voronoi.ipynb.

K-means for Image Segmentation and Compression

Original Image

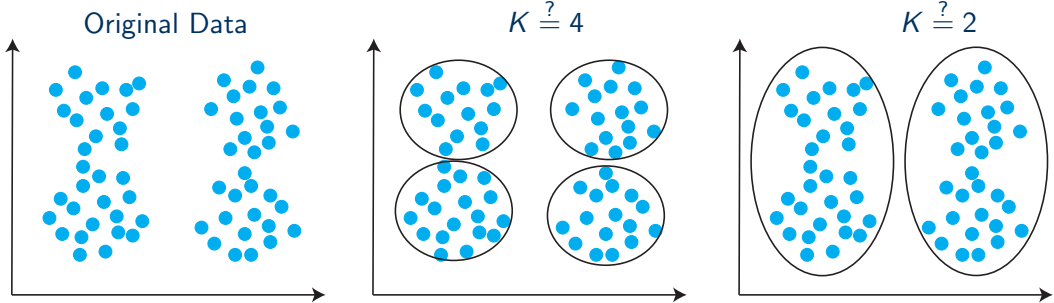
$K = 2$

$K = 3$

$K = 10$

Bishop Figure 9.3

How to choose K?

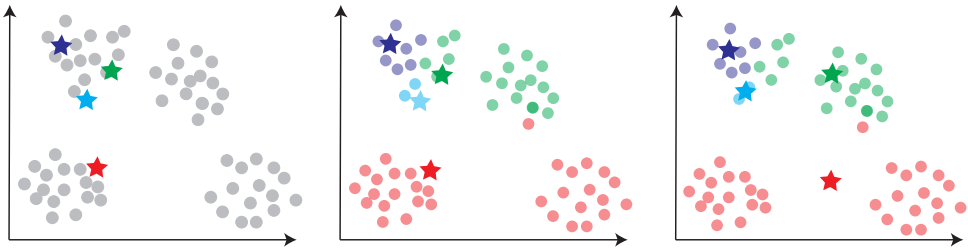


There are several methods for choosing K, including [but not limited to], using domain expertise, elbow and silhouette methods, and gap statistics*.

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

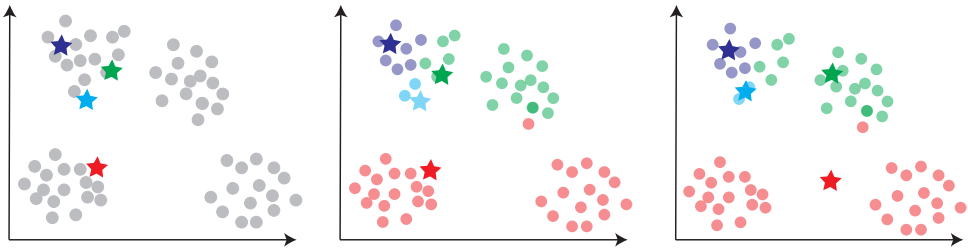
How to initialise μ_k

The K-means algorithm is sensitive to the initialisation of μ_k.



How to initialise μ_k

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



Methods of initialisation:

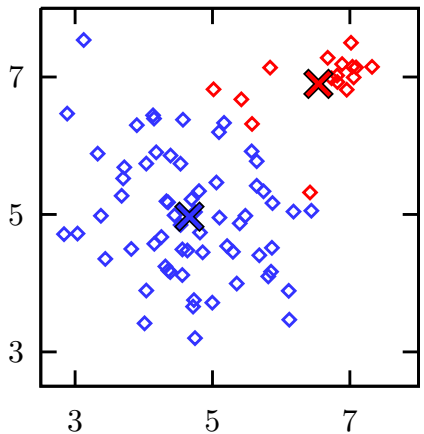
1. Random initialisation (the above case can happen!)
2. Often times, $\boldsymbol{\mu}_k$s are initialised to a subset of data (Forgy initialisation).
3. Repeat clustering for various initial and select the *best* set of $\boldsymbol{\mu}_k$s
4. K-means++ (Arthur and Vassilvitskii, 2007)

Does the algorithm converge?

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

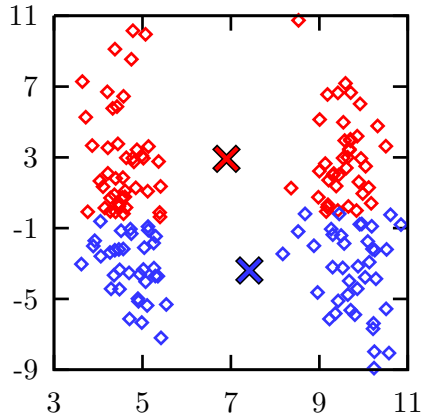
- In Step 2: we've shown $J_{\text{new}} \leq J_{\text{old}}$ by updating r_{nk}.
- In Step 3: we've shown $J_{\text{new}} \leq J_{\text{old}}$ by updating $\boldsymbol{\mu}_k$.
- So, $J_0, J_1, \ldots$ is a monotonically decreasing sequence, which guarantees convergence. However, there is no guarantee of a global optimum.
- In addition, the clustering/assignment problem has the finite number of possible combinations of partitions. This means that the algorithm stops within a finite number of iterations.

Failures of K-means (case 1)



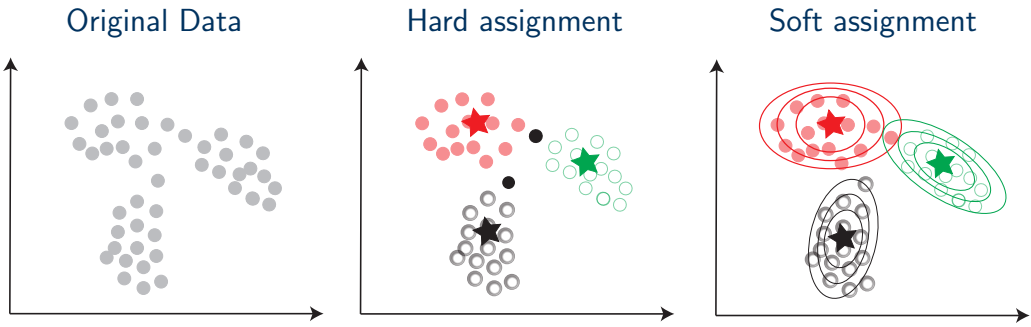
Large clouds pull small clusters off-centre

Failures of K-means (case 2)



Distance needs to be measured sensibly.

Hard assignment vs. Soft assignment



Some notes on k-means clustering

- The k-means algorithm is also referred to as *Lloyd's algorithm*.
- Cluster centres are also referred to as cluster *centroids*.
- *Vector Quantisation (VQ)* – as an application
  It replaces $\mathbf{x}_n \in \mathbb{R}^d$ with an integer k, which represents the index of the cluster where $\mathbf{x}_n$ lies.
  VQ is widely used for image/audio data compression.
- Linde–Buzo–Gray (LBG) algorithm – an iterative VQ, applying k-means while gradually increasing k.

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, and the loss function considers within class distortion only.
4. Hard boundaries!
5. How to generalise it to models that can cluster data of various types and shapes!