Data Science
k-means algorithm

Estimating discrete latent factors

Stéphane Marchand-Maillet

Department of Computer Science

UNIVERSITÉ DE GENÈVE
FACULTÉ DES SCIENCES

Master en Sciences Informatiques - Autumn semester

Table of contents

What is the lecture about?

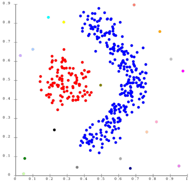
- ⋆ Understand the geometrical and statistical properties of the given data
- ⋆ Analyze the data and develop tools for this analysis
- ⋆ Here, we specifically address the (unsupervised) approach of data clustering
- ⋆ Understand the assumptions made in the design of these tools
- ⋆ Work out the theory (in depth)

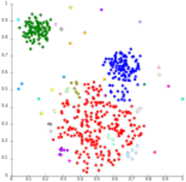
Reading: [2] (chap 9) and [6] (chap 21)

Note: Clustering is similar to (unsupervised) Classification, Density estimation and dual to Outlier detection

Introduction

- ⋆ Data does not generally arise from a simple Gaussian process (i.e, variations of a mean prototype)
- ⋆ The distribution of data generally shows non-uniformity with region of higher density.
- ⋆ Clustering is a unsupervised method that aims at discovering *consistent* groups of data, corresponding to *peaks of data density*
- ⋆ An often-used synonym for clustering (e.g in Signal Processing) is Vector Quantization (VQ) as “multi-dimensional quantization”





Clustering methods

There exists a large number of clustering methods, including:

- ⋆ Hierarchical Agglomerative Clustering
- ⋆ k-means [4], Lloyd's algorithm
- ⋆ Spectral clustering
- ⋆ Community detection
- ⋆ High-dimensional clustering
- ⋆ (see also) *Self-Organizing Maps, Neural Gas*

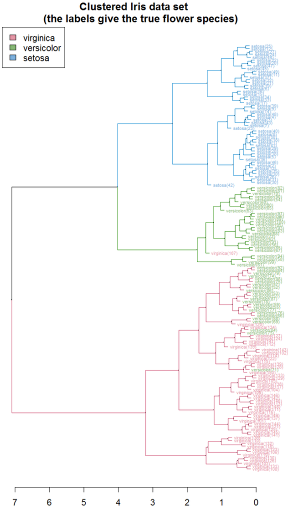
Research on clustering has been active since at least 50 years ([3] and a zillion other surveys on the topic $\longrightarrow$ find your own best)

Hierarchical Agglomerative Clustering

Iterative process:

1. Initialization: each data is a cluster
2. Find the closest pair of clusters
3. Merge these two clusters
4. Iterate from 2. until end

$\Rightarrow$ Data dendrogram φ $\rightarrow$



Closest pair of clusters (set distance):

- ⋆ Distance between centers
- ⋆ Max distance
- ⋆ Min distance

$\Rightarrow$ number of clusters unknown *a priori*

Note: Source: wikipedia / Iris data on UCI ML data repository

k-means strategy

- ⋆ The k-means clustering algorithm postulates a (Euclidean) metric (normed) space over the data
- ⋆ It seeks an unsupervised assignment of the data onto the clusters
- ⋆ It is a hard-assignment algorithm: the assignment is binary: each datum is assigned to one and only one cluster

Model

Given data $\mathcal{X} = \{\boldsymbol{x}_i\}_{i \in [\![N]\!]} \subset \Omega \subseteq \mathbb{R}^D$, given $K \in \mathbb{N}^*$, define:

- ⋆ (latent) binary assignment variables: $\boldsymbol{Z} = \{z_{ik}\}$ with $z_{ik} \in \{\text{false}, \text{true}\} \equiv \{0, 1\}$ for all i and k
- ⋆ cluster representatives: $\boldsymbol{M} = \{\boldsymbol{\mu}_k\}_{k \in [\![K]\!]}$ with $\boldsymbol{\mu}_k \in \mathbb{R}^D$ for all k
- ⋆ Parameters: $\boldsymbol{\theta} = [\boldsymbol{M}, \boldsymbol{Z}]$

k-means loss

k-means seeks the following assignment:

$$\hat{\boldsymbol{\theta}} = [\hat{\boldsymbol{M}}, \hat{\boldsymbol{Z}}] = \underset{\boldsymbol{\theta}=[\boldsymbol{M},\boldsymbol{Z}]}{\operatorname{argmin}} \mathcal{L}(\boldsymbol{\theta}, \mathcal{X})$$

with loss function:

$$\mathcal{L}(\boldsymbol{\theta}, \mathcal{X}) = \sum_{i=1}^{N} \sum_{k=1}^{K} z_{ik} \|\boldsymbol{x}_i - \boldsymbol{\mu}_k\|_2^2$$

Since the assignment is binary, the exact optimization is NP-Hard
$\Rightarrow$ we seek an approximation by coordinate descent

Reminder: Coordinate descent algorithm

This is an alternative minimization algorithm:
Given $\boldsymbol{f} : \mathbb{R}^D \mapsto \mathbb{R}$, we seek

$$\boldsymbol{x}^* = \arg\min_{\boldsymbol{x} \in \mathbb{R}^D} \boldsymbol{f}(\boldsymbol{x})$$

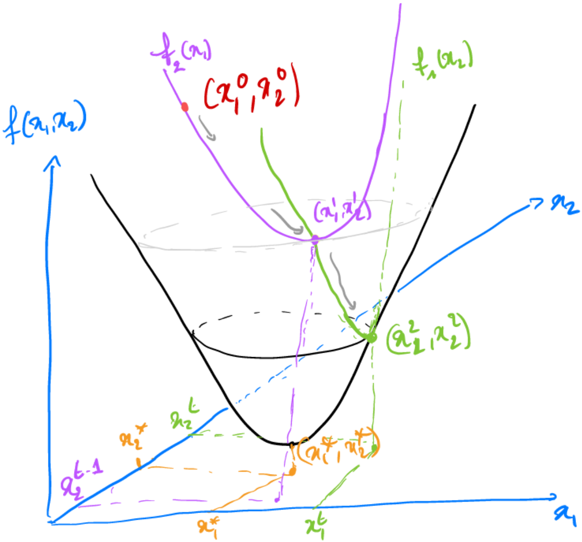
We define $\boldsymbol{f}_d : \mathbb{R}^D \times \mathbb{R} \to \mathbb{R}$ with $d \in [\![D]\!]$ where

$$\boldsymbol{f}_d(\boldsymbol{x}, y) = f([\boldsymbol{x}(1), \cdots, \boldsymbol{x}(d-1), y, \boldsymbol{x}(d+1), \cdots, \boldsymbol{x}(D)]^{\mathsf{T}})$$

and alternatively seek the minimum:

$$\boldsymbol{x}^{(t+1)}(d) = \operatorname*{argmin}_y \boldsymbol{f}_d(\boldsymbol{x}^{(t)}, y)$$

Coordinate descent



Application to k-means

We alternate the optimization of $\boldsymbol{Z}$ and $\boldsymbol{M}$

Let

$$\mathcal{L}_{\boldsymbol{M}}(\boldsymbol{Z}) = \mathcal{L}(\theta, \mathcal{X})|_{\boldsymbol{M}=\boldsymbol{M}}$$

and

$$\mathcal{L}_{\boldsymbol{Z}}(\boldsymbol{M}) = \mathcal{L}(\theta, \mathcal{X})|_{\boldsymbol{Z}=\boldsymbol{Z}}$$

We have:

$$\frac{\partial \mathcal{L}_{\boldsymbol{Z}}}{\partial \boldsymbol{\mu}_k} = -2 \sum_{i=1}^{N} z_{ik}(\boldsymbol{x}_i - \boldsymbol{\mu}_k)$$

$\Rightarrow$ the optimal representation $\boldsymbol{M}$ for a given assignment $\boldsymbol{Z}$ is reached at:

$$\frac{\partial \mathcal{L}_{\boldsymbol{Z}}}{\partial \boldsymbol{\mu}_k} = 0 \qquad \Rightarrow \qquad \boldsymbol{\mu}_k = \frac{\sum_{i=1}^{N} z_{ik} \boldsymbol{x}_i}{\sum_{i=1}^{N} z_{ik}}$$

Hence, cluster representatives are their centers of mass

Updating the assignment

Given a set of cluster representatives $\boldsymbol{M}$, to minimize $\mathcal{L}_{\boldsymbol{M}}(\boldsymbol{Z})$
Recall:

$$\mathcal{L}_{\boldsymbol{M}}(\boldsymbol{Z}) = \left[\sum_{i=1}^{N}\sum_{k=1}^{K} z_{ik}\|\boldsymbol{x}_i - \boldsymbol{\mu}_k\|_2^2\right]_{\boldsymbol{M}=\boldsymbol{M}}$$

It is a sum of positive members $\rightarrow$ we assign:

$$z_{ik} = 1 \qquad \text{only when} \qquad k = \operatorname*{argmin}_m \|\boldsymbol{x}_i - \boldsymbol{\mu}_m\|_2^2$$

($z_{ik} = 0$ otherwise)

Hence, $\mathcal{L}_{\boldsymbol{M}}(\boldsymbol{Z})$ is minimum when every data is assigned to its nearest cluster representative

k-means algorithm

Given data $\mathcal{X} = \{\boldsymbol{x}_i\}_{i \in [\![N]\!]}$ with $\boldsymbol{x}_i \in \mathbb{R}^D$ and given $K \in \mathbb{N}^*$

1. Initialize cluster representatives $\boldsymbol{M}^{(0)}$
2. Given cluster representatives $\boldsymbol{M}^{(t)}$, assignment $\boldsymbol{Z}^{(t)}$ associates each data to its nearest cluster representative
3. Given assignment $\boldsymbol{Z}^{(t)}$, new cluster representatives $\boldsymbol{M}^{(t+1)}$ are centers of mass of data assigned to the clusters
4. Repeat from step 2 until convergence

Convergence is attained when centers of mass do not move much, or when the assignment is stable

Note: Step 2 is similar to an Expectation step, and step 3 is similar to a Maximization step, considering hard-assignment (ref EM algorithm)

Properties

- ⋆ Since it performs alternate minimization of convex functions, it guarantees to decrease the loss at every iteration
- ⋆ Since there is a large (combinatorial) but finite number of assignment, the number of iterations is (large but) finite
- ⋆ Step 2 is equivalent to building the discrete Voronoi diagram of $\mathcal{X}$ with centers $\boldsymbol{M}$ as seeds
- ⋆ Step 2 is similar to an Expectation step, and step 3 is similar to a Maximization step, considering hard-assignment (ref EM algorithm)

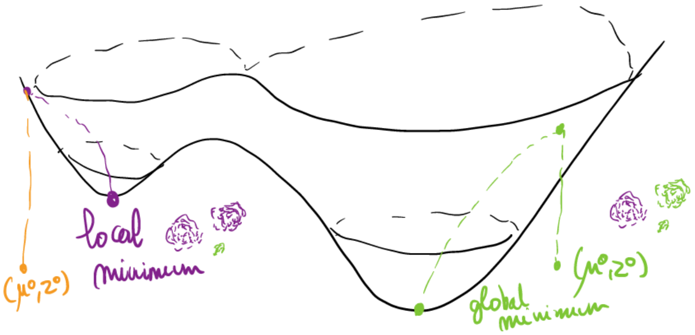
→ The (quality of the) result varies upon initialization

→ Randomization can be useful if computation is fast

otherwise, use heuristic for better initialization

k-means ++

Since the exact optimization (optimal assignment) is NP-Hard, the k-means algorithm reaches a local optimum



The quality of the result depends on the initialization → various strategies

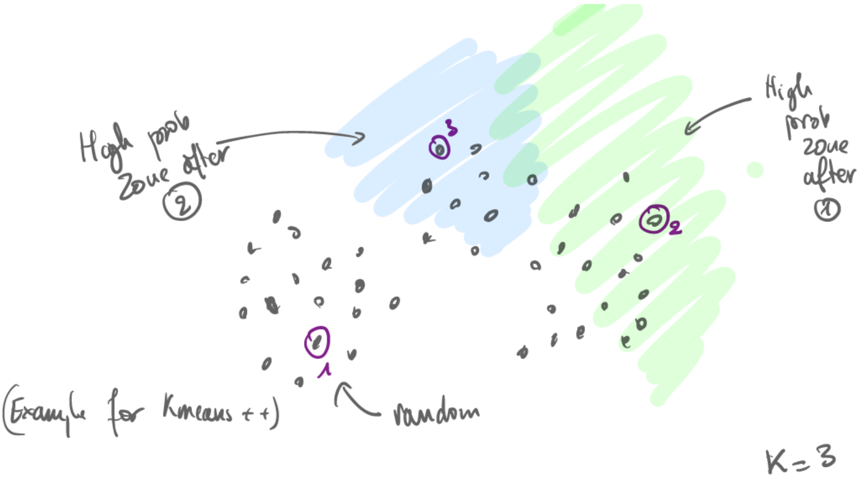
k-means ++

The principle is to maximally spread the initial cluster representatives $\boldsymbol{M}^{(0)}$ over the data [1]
This initialization is known to (i) produce better quality clusters (ii) speed up the convergence of k-means

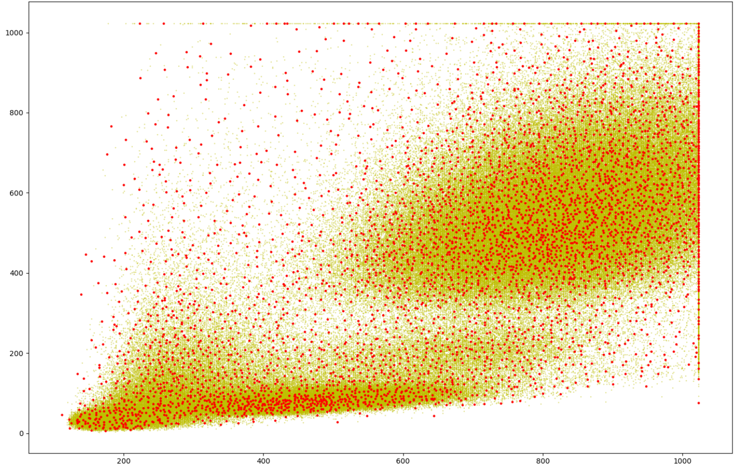
1. Select the first cluster representative $\boldsymbol{\mu}_1^0$ randomly among the data $\mathcal{X}$
2. For all non selected $\boldsymbol{x}_i$, compute $\Delta_i = \|\boldsymbol{x}_i - \boldsymbol{\mu}_m^0\|_2^2$ the distance $\boldsymbol{x}_i$ and its nearest representative $\boldsymbol{\mu}_m^0$ among the k already selected representatives
3. Sample the next representative $\boldsymbol{\mu}_{k+1}^0$ from $\mathcal{X}$ with probability proportional to distribution Δ
4. Repeat from step 2 until K representatives are chosen

Note: this initialization strategy is used by several Data Science packages (MATLAB™, Python© SciKit Learn, R, ...)

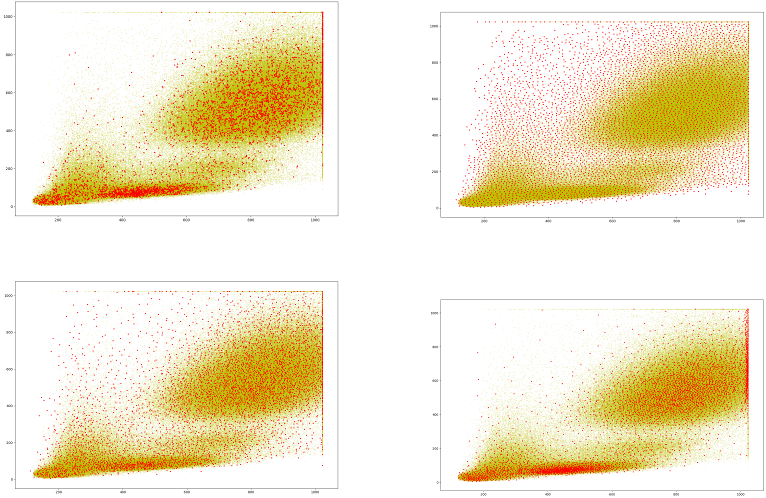
k-means ++


High prob zone after 2
High prob zone after 1
3
2
1
random
(Example for Kmeans ++)
K=3

k-means ++



Data sampling



Reading order: Random, FFT, *k*-means ++, HubHSP [5]

Summary

- ⋆ Clustering is part of the unsupervised family of data modeling techniques
- ⋆ k-means is one of the most popular such techniques
- ⋆ k-means performs a hard assignment
- ⋆ Sound optimization criterion (loss) but NP-Hard

→ the k-means algorithm seeks an approximate solution by coordinate descent (alternate minimization)

- ⋆ Since the loss is not convex, the technique is sensitive to initialization
- ⋆ k-means $++$ is a prior heuristic for initialization that is efficient in practice
- ⋆ Clustering can be constrained (with MUST-LINK and CANNOT-LINK constraints)

Example questions [mostly require formal – mathematical – answers]

- ⋆ Describe formally clustering
- ⋆ In what sense is it an unsupervised technique?
- ⋆ Explain why k-means is intrinsically linked to the Euclidean metric?
- ⋆ Why do we say that k-means considers a Gaussian model for the clusters?
- ⋆ Is the k-means algorithm exact?
- ⋆ What are the principles to initialize k-means?
- ⋆ What is the coordinate descent algorithm?

φ It is strongly advised to develop the algebra contained in this chapter

References I

[1] David Arthur and Sergei Vassilvitskii. K-means++: The advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '07, pages 1027–1035, USA, 2007. Society for Industrial and Applied Mathematics.

[2] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006. (available online).

[3] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: A review. *ACM Comput. Surv.*, 31(3):264–323, September 1999.

[4] J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proc. Fifth Berkeley Sympos. Math. Statist. and Probability (Berkeley, Calif., 1965/66)*, pages Vol. I: Statistics, pp. 281–297. Univ. California Press, Berkeley, Calif., 1967.

[5] Stephane Marchand-Maillet and Edgar Chávez. HubHSP graph: effective data sampling for pivot-based representation strategies. In *15th International Conference on Similarity Search and Applications*, 2022.

[6] Kevin P. Murphy. *Probabilistic Machine Learning: an Introduction*. MIT Press, 2022. (available online).