FTML practical session 8

1^{er} mai 2025

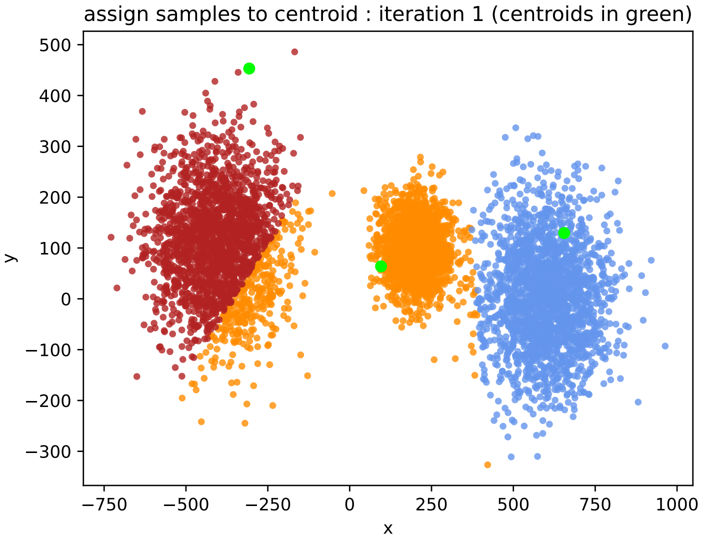

TABLE DES MATIÈRES

1 K-MEANS DEMO

In **k_means_demo/**, you can find an example of k-means algorithm, implemented with numpy, with 3 clusters in dimension 2. If the initialization is kept random, the results will be random but will typically look like figure 1.

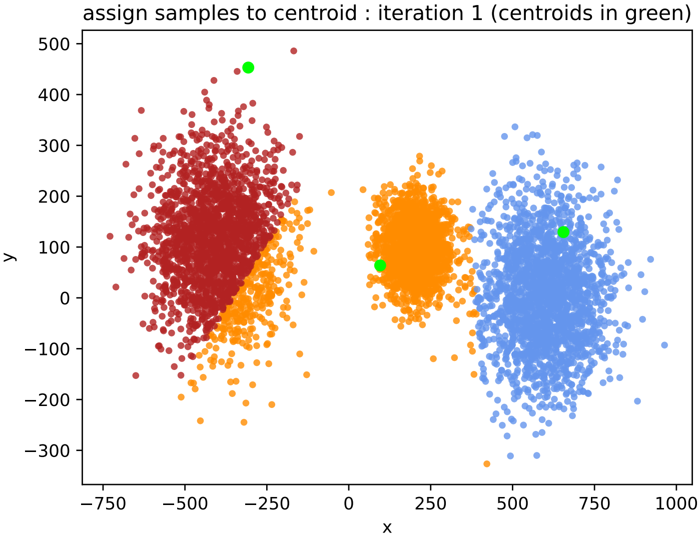

FIGURE 1 – An iteration step in the k-means algorithm

2 NUMBER OF CLUSTERS FOR K-MEANS

In this exercice we study heuristics to find relevant number of clusters for a k-means algorithm.

- folder : **exercice_1_k_means/**
- template file : **k_means_heuristics.py**

A company has gathered data about its customers and would like to identify similar clients, in order to propose relevant products to new clients, based on their features. This can be represented as a clustering problem. The data are stored in **data.npy**. They are 4 dimensional. We would like to study methods to automatically find a relevant number of clusters in this dataset for the K-means algorithm.

2.1 Visual method

Is there a direct, visual way to have an idea of a relevant number of clusters for these 4-dimensional data ?

2.2 Algorithmic heuristics

We would like to see if algorithmic heuristics are consistent with the previous results.

Experiment with the following heuristics, metrics and tools in order to obtain a suggested number of clusters :

- possible metrics to monitor :
  - inertia :
    `https://scikit-learn.org/stable/modules/clustering.html#k-means`

  - silhouette score
    https://scikit-learn.org/stable/modules/clustering.html#silhouette-coefficient
  - Calinksi-Harabasz score :
    https://scikit-learn.org/stable/modules/clustering.html#calinski-harabasz-index
- Knee detection tools :
  - https://github.com/arvkevi/kneed
  - https://www.scikit-yb.org/en/latest/api/cluster/elbow.html

2.3 Changing the data

You can generate the data again in order to study the behavior of the previous heuristics on a dataset with different statistical properties.

3 METEOROLOGICAL DATA : DIMENSIONALITY REDUCTION AND VISUALIZATION

A meteorological station has gathered 1600 data samples in dimension 6, thanks to 6 sensors, that represent various physical measurements (such as wind speed, humidity, temperature, etc). The operators of the station would like to predict the risk of a tempest the next day, but first, they need to reduce the dimensionality of the data, in order to apply a supervised learning algorithm on the reduced data.

The data are stored in the **exercice_2_dimensionality_reduction/data/** folder.

Find a dimensionality reduction method and a dimension (2 or 3), that seems to allow to predict the label based on the projected components only, first by making scatter plots of the projected data, and by coloring the data according to their label. Verify this by training a classifier that learns to predict the labels based on the projections only, with a very good accuracy.

Template files in the folder :

- **main.py**

https://scikit-learn.org/stable/modules/unsupervised_reduction.html

Later in the course, we will also study nonlinear dimensionality reduction methods. Note that for this particular problem, it was also possible to solve it without the first unsupervised learning stage.