SPATIAL CLUSTERING METHODS

Emmanuel Kwame Ayanful
10658737

Department of Mathematics
School of Physical and Mathematical Sciences
University of Ghana, Legon

July 21, 2022

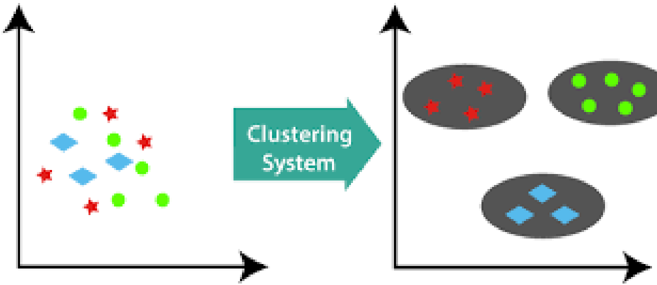
Introduction

- Machine learning is a branch of Artificial intelligence that provide systems the ability to learn from experience without being explicitly programmed.
- There are three (3) learning methods in Machine learning namely:
    - Supervised learning
    - Unsupervised learning
    - Reinforcement learning

Clustering I

Definition

Data clustering may be described as the problem of identifying subgroups of data in such a way that data points in the same subgroup (cluster) are highly similar while data points in other clusters are quite distinct.



Clustering II

There are five(5) types of clustering methods:

- Partitioning clustering
    - K-means clustering
    - K-medoids clustering
- Hierarchical clustering
    - Agglomerative clustering
    - Divisive clustering
- Fuzzy clustering
    - Fuzzy C-means clustering
- Density-based clustering
    - DBSCAN
- Model-based clustering
    - Gaussian-mixture model clustering

Proximity measure I

Euclidean Norm

Given a vector $x \in R^n$, the Euclidean norm is defined as the square root of the sum of absolute squares of its elements.

$$\|x\| = \sqrt{\sum_{i=1}^{n} x_i^2}$$

Euclidean distance

Given two vectors $p, q \in R^n$. The Euclidean distance from p to q is defined as the Euclidean norm on $p - q$ expressed as:

$$d = \|p - q\| = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}$$

Proximity measure II

Squared Euclidean Distance

The squared Euclidean distance uses the same equation as the Euclidean distance but does not take the square root.

$$d = \|x - y\|_2^2 = \sum_{i=1}^{n} (x_i - y_i)^2$$

K-means Clustering Technique I

Definition

K-means clustering is a partitional clustering technique that attempts to partition or divide the datasets into a number of pre-defined distinct non-overlapping subgroups where each data point belongs to one and only one group.

K-means as an optimization problem

The Objective Function

The main objective function of the K-Means algorithm is given by:

$$
\begin{aligned}
J &= \sum_{j=1}^{k} \sum_{i: x_i \in j} \left\| x_i - \mu_j \right\|^2 \\
&= \sum_{j=1}^{K} \sum_{i=1}^{n} w_{ij} \left\| x_i - \mu_j \right\|^2
\end{aligned}
$$

where x_i is the i^{th} data point, μ_j is the center of the j^{th} cluster and
$w_{ij} = \begin{cases} 1, & \text{if data point } x_i \text{ is assigned to cluster } j. \\ 0, & \text{otherwise.} \end{cases}$

K-Means as an optimization problem (Cont.) I

- The problem here is a minimization problem in two parts. We first want to minimize J w.r.t w_{ij} and treat μ_j as a constant. Then we minimize J w.r.t μ_j and treat w_{ij} as a constant
- Choose the optimal w_{ij} for fixed μ_j. We call this step the expectation step (E-step).
- Choose the optimal μ_j for fixed w_{ij}. We call this step the maximization step (M-step).

Expectation step

Here, we minimize J by holding μ_k constant and optimizing w_{ij}

$$w_{ij} = \begin{cases} 1, & \text{if } j = \arg\min_l \|x_i - \mu_l\|^2. \\ 0, & \text{otherwise.} \end{cases}$$

That is, the data point x_n is assigned to the closest cluster with centroid μ_k with respect to the sum of squared Euclidean distance.

Maximization Step I

We continue by taking the partial derivative of J with respect to μ_j given as:

$$\frac{\partial J}{\partial \mu_j} = \frac{\partial \sum_{i=1}^{n} w_{ij} \|x_i - \mu_j\|^2}{\partial \mu_j}$$

But

$$\begin{aligned} \|x_i - \mu_j\|^2 &= (x_i - \mu_j)^T (x_i - \mu_j) \\ &= x_i^T x_i - x_i^T \mu_j - \mu_j^T x_i + \mu_j^T \mu_j \\ &= x_i^T x_i - 2x_i^T \mu_j + \mu_j^T \mu_j \end{aligned}$$

Maximization Step II

So

$$
\begin{aligned}
\frac{\partial J}{\partial \mu_j} &= \frac{\partial \sum_{i=1}^{n} w_{ij}(x_i^T x_i - 2x_i^T \mu_j + \mu_j^T \mu_j)}{\partial \mu_j} \\
&= \sum_{i=1}^{n} w_{ij}\left(\frac{\partial x_i^T x_i}{\partial \mu_j} - 2\frac{\partial x_i^T \mu_j}{\partial \mu_j} + \frac{\partial \mu_j^T \mu_j}{\partial \mu_j}\right) \\
&= \sum_{i=1}^{n} w_{ij}(-2x_i + 2\mu_j) \\
&= -2\sum_{i=1}^{n} w_{ij} x_i + 2\mu_j \sum_{i=1}^{n} w_{ij}
\end{aligned}
$$

Maximization Step III

Setting $\frac{\partial J}{\partial \mu_j} = 0$

$$
\begin{aligned}
\Rightarrow \quad & -2\sum_{i=1}^{n} w_{ij}x_i + 2\mu_j \sum_{i=1}^{n} w_{ij} = 0 \\
\Rightarrow \quad & -2\sum_{i=1}^{n} w_{ij}x_i = -2\mu_j \sum_{i=1}^{n} w_{ij} \\
\Rightarrow \quad & \mu_j = \frac{\sum_{i=1}^{n} w_{ij}x_i}{\sum_{i=1}^{n} w_{ij}}
\end{aligned}
$$

Now we let

$$\sum_{i=1}^{n} w_{ij} = n_j$$

Then

$$\mu_j = \frac{\sum_{i:x_i \in j} x_i}{n_j}$$

Maximization Step IV

The matrix of second derivatives is given as:

$$\frac{\partial^2 J}{\partial \mu_j^2} = \frac{\partial \sum_{i=1}^{n} w_{ij}(-2x_i + 2\mu_j)}{\partial \mu_j}$$

$$= 2\sum_{i=1}^{n} w_{ij} I > 0$$

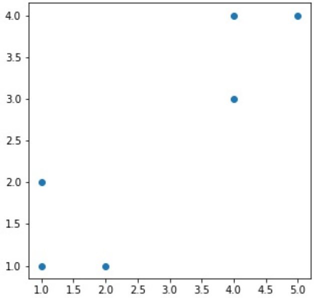
The Algorithm I

1. Specify the number of clusters k and choose k initial centroids.
2. Compute the sum of squared distances between data points and all centroids.
3. Assign each data point to the closest cluster based on the smallest distance to cluster's centroids.
4. Recompute the centroids for each cluster by taking the average of all data points that belongs to the cluster.
5. Repeat step 2, 3, and 4 until no data point changes cluster or centroids do not change values.

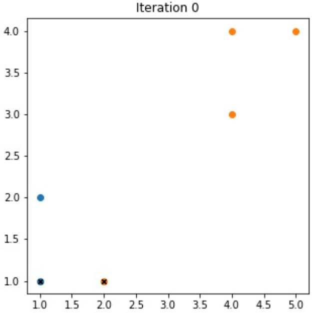
The Algorithm II

Machine A	1	1
Machine B	2	1
Machine C	4	3
Machine D	5	4
Machine E	1	2
Machine F	4	4

Table: A simple dataset to illustrate the kmeans Algorithm

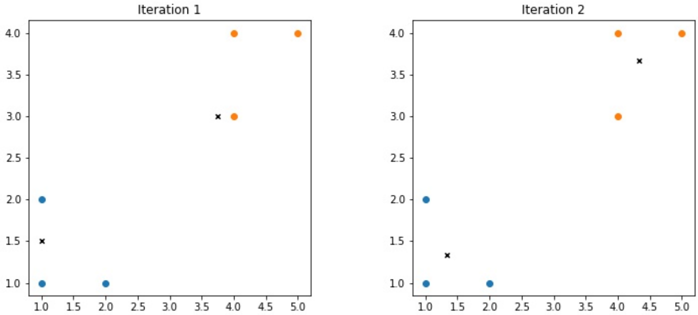


(a) Original



(b) Iteration 0

The Algorithm IV



(c) Iteration 1 (d) Iteration 2

Figure: Plots showing the iteration process of the kmeans.

Applications

- Image segmentation

(a) Original

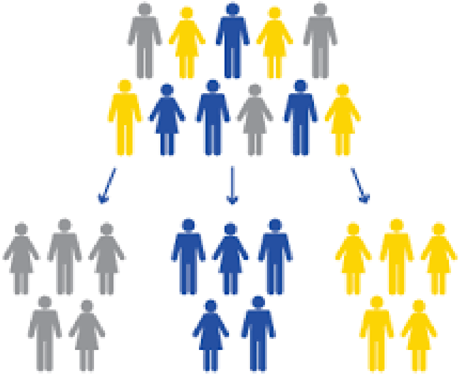
(b) Using 5 clusters

(c) Using 12 clusters

Figure: Plots showing image segments using kmeans.

Applications Contd.

- Marketing and Sales



- Spam filter