”UNSUPERVISED LEARNING: K MEANS ALGORITHM”		
Group Number		Deadline
Group 3		**11.04.2023, 23:59 pm**
April 11, 2023		Ac. Year: 2021 - 2022
		Lecturer(s): ”**Moustapha Cisse**”

Unsupervised Learning: K-Means Algorithm.

1 Introduction

When talking about K-Means, we make reference to Clustering. Clustering is a technique that we use in unsupervised learning in order to group data based on some similarities that they have among themselves. We have 2 types of clustering algorithms:

- Direct partitioning: K-Means is part of this technique. In this method, we know in advance the number K of clusters.
- Hierachical clustering: Unlike the direct partitioning, we don’t know in advance the number of clusters K.

The fundamental concept is to update the clustering centers by calculating the mean of the member points, relocate every single point to its new closest center, and repeat this procedure until convergence criteria (which might include an established number of iterations, a difference in the value of the distortion function) are satisfied[1].

In the following work, we will discuss about the differents steps to follow in the implementation of K-Means algorithm where we will mention the techniques used to find the values of K, the techniques used to initialize the centroids and the stopping criteria; We will also talk about the pros and cons of K-Means algorithm and in the last part, we will present some applications of K-Means in our daily life.

2 Implementation of K-Means algorithm

As every algorithm, there are several steps to follow concerning the implementation of K-Means algorithm, the different steps are presented in Figure 1.

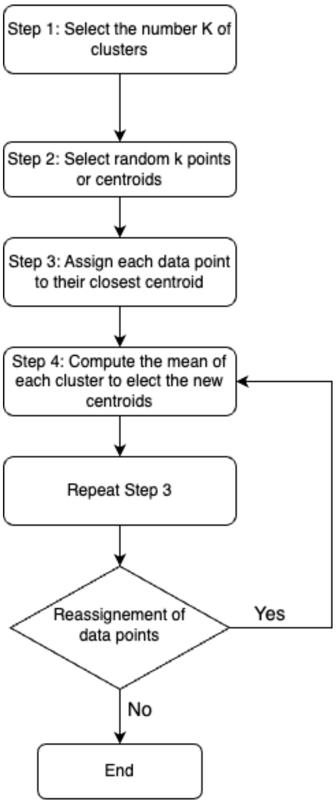

Step 1: Select the number K of clusters
Step 2: Select random k points or centroids
Step 3: Assign each data point to their closest centroid
Step 4: Compute the mean of each cluster to elect the new centroids
Repeat Step 3
Reassignement of data points
Yes
No
End

Figure 1: Organigram of K-Means algorithm

2.1 Selection of the number of clusters

As a prerequisite to apply the K-Means, we need the number of clusters, there are several methods that help us to select the number of clusters:

- The Elbow method
- The Silhouette analysis

2.1.1 The Elbow method

The most well-known approach for determining the ideal cluster size is the elbow method. Elbow is a means to gauge how cohesive a cluster of data that are grouped together because they are similar to one another is. However, this approach has one drawback. The value of cohesiveness will eventually go below zero when K (number of clusters) is increased, therefore it will not be possible to decide if a cluster is good or not. Elbow method suggest that each value of WCSS(Within Cluster Sum of Square) is listed in a graph where the Y-axis label represents the value of WCSS and the X-axis, the value of K. The best value of K is when the graph has a significant bend[2]. The total within-cluster sum of square (WCSS)(Equation 1) measures the compactness of the clustering and we want it to be as small as possible.

$$WCSS = \sum_{i=1}^{k} \sum_{x \in S_i} ||x - \mu_i||^2 \tag{1}$$

The location of a optimal number of clusters in the plot is obtained when we start observing a small decrease of the curve; as shown by Figure 2, the optimal value of K is 3.

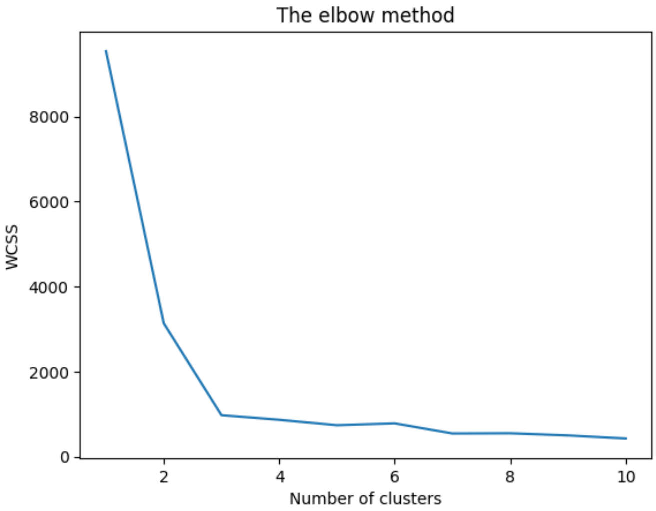


Figure 2: The Elbow method

2.1.2 The Silhouette analysis

Another method we use to find the optimal value of the clusters is the silhouette analysis. This method shows how well objects have been assigned to their cluster. The entire clustering is displayed by combining the silhouette into a single plot, allowing an appreciation of the relative quality of the clusters and an overview of the data configuration. The average silhouette width provides an evaluation of clustering validity, and might be used to select an 'appropriate' number of clusters [3]. To get the silhouette score of a clustering, we compute the silhouette coefficient for each point in the dataset and we take the mean. Let's see how it works. Assume that we have three clusters A, B and C for a dataset $D = \{x_i\}_{i=1}^{n}$. For each object i, $s(i)$ is the silhouette coefficient.

$$a(i) = \frac{\sum_{x \in A} d(i,x)}{|A|} \tag{2}$$

$a(i)$ is the average distance between i and all the points in the same cluster with it.

$$b(i) = \frac{\sum_{x \in B, x \in C} d(i,x)}{|B + C|} \tag{3}$$

$b(i)$ is the average distance between i and all the objects that aren't in the same cluster with it. $d(x, y)$ is the euclidian distance between x and y. Once we have $a(i)$ and $b(i)$, we compute $s(i)$ as show by Equation [4]:

$$s(i) = \frac{b(i) - a(i)}{\max(b(i), a(i))} \tag{4}$$

More explicitely, we can write that as follow:

$$s(i) = \begin{cases} 1 - \frac{a(i)}{b(i)} \text{ if } a(i) < b(i) \\ 0 \text{ if } a(i) = b(i) \\ \frac{b(i)}{a(i)} - 1 \text{ if } a(i) > b(i) \end{cases} \tag{5}$$

From Equation 5, we can easily see that:

$$-1 \leq s(i) \leq 1 \tag{6}$$

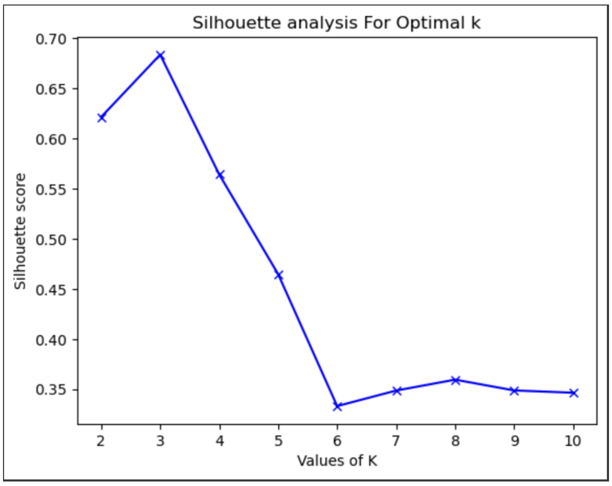


Figure 3: Silhouette analysis

A value close to 1 indicates that the object is well clustered, A value close to 0 informs that the object is very close to the boundary decision and a negative value indicates that the object might have been clustered wrongly. The silhouette score for the clustering is given by

$$S = \frac{\sum_{i=1}^{n} s(i)}{n} \tag{7}$$

Then, we repeat this computation for many values of numbers of clusters. The value of optimal number of cluster is where S has the highest value, in our case k=3 as shown by Figure 3.

2.2 Initialization of the centroids

After selecting, the number of clusters, we have to initialize our centroids, there are several techniques that we may use for centroids initialization like:

- Choose the first k data points: with this technique, we choose the first k data points of our dataset as centroids [1].
- Forgy Initialization: randomly select K instances of a database (seeds) and place the other instances in the cluster indicated by the closest seed [5].
- Random partition Initialization: With this method, we divide the data points randomly into K subsets.
- K means ++ initialization: The first initial cluster centroid is randomly selected by this technique from the data points. The centroids are updated depending on two variables: the squared distance and the probability percentage that draws from points that are close to current cluster centroids [6].

After initializing the centroids, the next step is to assign each data point to their closest centroid, for that we use the euclidean distance given by Equation

$$d_{ij} = ||x_i - x_j||^2 \tag{8}$$

where $||.||$ denotes the euclidean norm. We stop the algorithm when the stopping criteria is satisfied.

2.3 Stopping criteria of K-means algorithm

K-means algorithm is an iterative method. In order to minimize the distorsion within clusters, we iterate many times by moving the positions of centers of clusters. We have essentially three stopping criteria that can be used to stop the K-means algorithm[7].

- Centroids of newly formed clusters do not change: That means as we are moving the centroids during the iteration, when the distance between the current centroid and the previous one is zero for every cluster, we stop the iteration.

- Points remain in the same cluster: We stop also the iteration if we notice that no object is moving from a cluster to another one. we can say that the algorithm is not learning any new pattern, and it is a sign to stop the training.

- Maximum number of iterations is reached: we can stop the training if the maximum number of iterations is reached. Suppose we have set the number of iterations as 300. The process will repeat for 300 iterations before stopping.

The first two criteria confirm that the algorithm has converge.

3 Pros and cons of K-means algorithm

As every algorithm, K-means has some pros and cons that we will present in the following part.

3.1 Pros

- Linear time complexity and can be used with large datasets conveniently:

- Easy to implement

- Easy to interpret the clustering results

3.2 Cons

- Results will differ based on random centroid initialization: with different initialization techniques, we might end up with different results.

- Sensitive to outliers: The mean is easily influenced by extreme values. But, this problem is solved by K-medoids clustering, a variant of K-means which is robust to noises and outliers. It takes one of the point in the cluster as a centroid instead of the mean.

- Assume each cluster has roughly equal number of observations

4 Some applications of K-means algorithm

- Customer segmentation: Clustering helps marketers improve their customer base, work on target areas, and segment customers based on purchase history, interests, or activity monitoring. The classification would help the company target specific clusters of customers for specific campaigns. To do so, they use some keys information as features. We can enumerate: Basic customer information(age, sex,...), Value information, Behavioral information [8].

- Pattern recognition: K-Means have been studied as a tool in patterns recognition in data. For example the basic flowchart of K-Means based PD pattern recognition includes coordinate transformation, K-Means based clustering, ”Means” overlay and PD pattern evaluation. The individual pulses are the input for the pattern recognition of K-Means based pattern recognition. It should be noted that in addition to the PD pulses, the data may contain impulsive noise signals that that resemble the PD pulses in the time domain. After the K-Means based clustering, the next step of PD pattern recognition is performed based on the angles between the different ”means” on the angles between the different ”means”. Before the judgement, the positive and negative ”means” will be superimposed in order to allow to form models[9]
- Data compression: The K-means clustering is used to compress the image data collected. By doing so, the transmission overhead of these images can be reduced significantly.

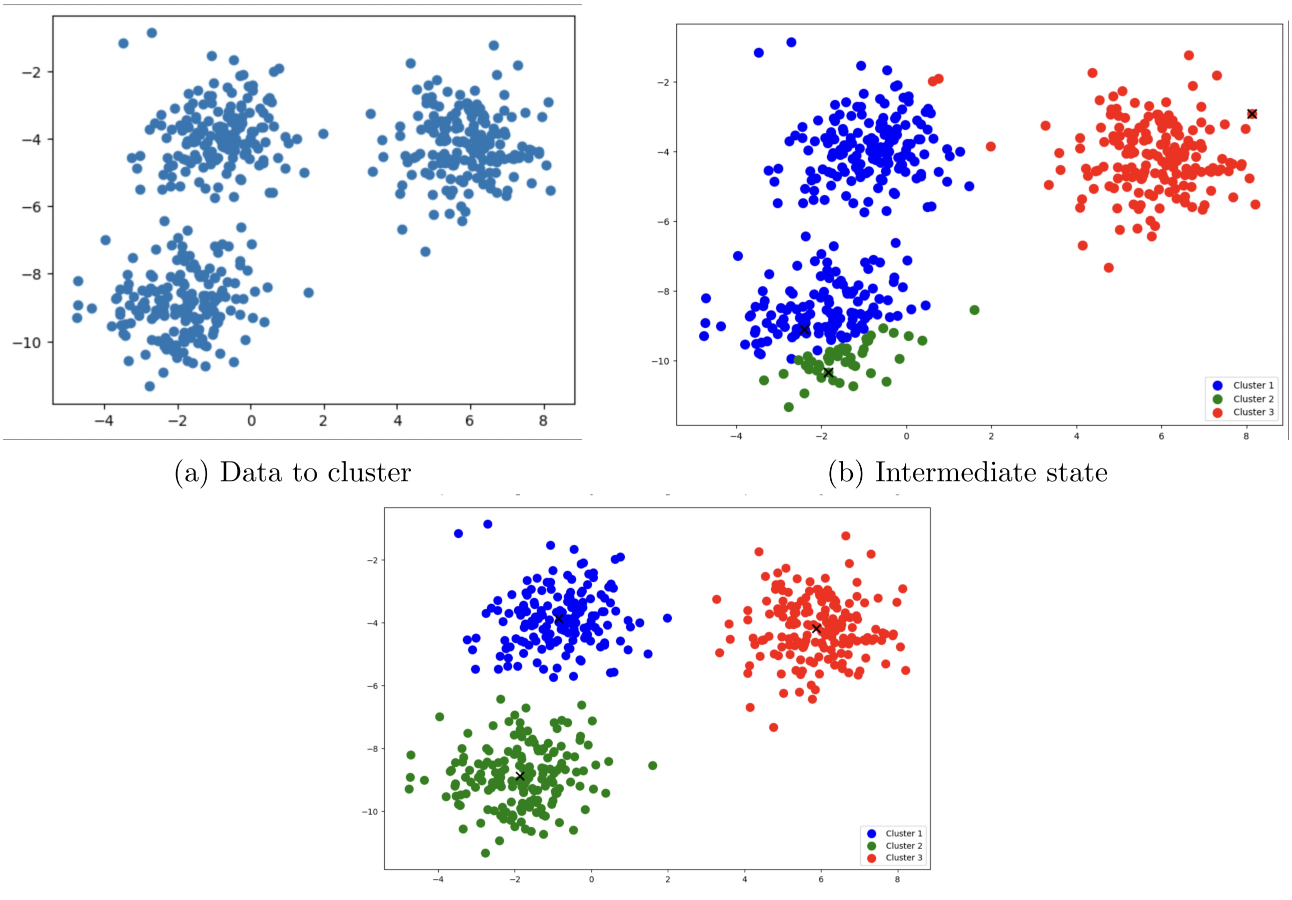
5 Conclusion

We have seen that K-Means is an unsupervised and iterative method [10] part of the Direct partitioning algorithm in the family of clustering. We discuss about the different steps to follow while implementing K-means algorithm and mention the critical part of the implementation like finding the optimal value of K where we talked about the Elbow method and the Silhouette analysis; some initialization techniques like Random Initialization and K means ++. Despite some constraints like the sensitivity to outliers or the differences of results that we might while using different initialization techniques, we can use K-means in many applications of your daily life like customer segmentation, pattern recognition or image compression to name a few.

6 Appendix

Algorithm 1 K-means Clustering Algorithm.

```
Input:
- K: number of clusters
- D: Dataset of N points
Ensure: A set of K clusters
Begin
Initialization
repeat
    for each point $p \in D$ do
        find the nearest center and assign p to the corresponding cluster.
    end for
    update clusters by calculating new centers using the mean of the members
until stop-iteration criteria satisfied
return Clustering results
End.
```



(a) Data to cluster

(b) Intermediate state

(c) Final state

Figure 4: K means Clustering process

References

[1] Xin Jin and Jiawei Han. *K-Means Clustering*, pages 563–564. Springer US, Boston, MA, 2010.

[2] MA Syakur, BK Khotimah, EMS Rochman, and Budi Dwi Satoto. Integration k-means clustering method and elbow method for identification of the best customer profile cluster. In *IOP conference series: materials science and engineering*, volume 336, page 012017. IOP Publishing, 2018.

[3] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

[4] Fei Wang, Hector-Hugo Franco-Penya, John D Kelleher, John Pugh, and Robert Ross. An analysis of the application of simplified silhouette to the evaluation of k-means clustering validity. In *Machine Learning and Data Mining in Pattern Recognition: 13th International Conference, MLDM 2017, New York, NY, USA, July 15-20, 2017, Proceedings 13*, pages 291–305. Springer, 2017.

[5] J.M Pena, J.A Lozano, and P Larranaga. An empirical comparison of four initialization methods for the k-means algorithm. *Pattern Recognition Letters*, 20(10):1027–1040, 1999.

[6] David Arthur and Sergei Vassilvitskii. k-means++: the advantages of careful seeding. In *SODA '07: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.

[7] Kristina P Sinaga and Miin-Shen Yang. Unsupervised k-means clustering algorithm. *IEEE access*, 8:80716–80727, 2020.

[8] Luo Ye, Cai Qiu-ru, Xi Hai-xu, Liu Yi-jun, and Yu Zhi-min. Telecom customer segmentation with k-means clustering. In *2012 7th International Conference on Computer Science & Education (ICCSE)*, pages 648–651. IEEE, 2012.

[9] Xiaosheng Peng, Chengke Zhou, Donald M Hepburn, Martin D Judd, and Wah Hoon Siew. Application of k-means method to pattern recognition in on-line cable partial discharge monitoring. *IEEE Transactions on Dielectrics and Electrical Insulation*, 20(3):754–761, 2013.

[10] Shi Na, Liu Xumin, and Guan Yong. Research on k-means clustering algorithm: An improved k-means clustering algorithm. In *2010 Third International Symposium on Intelligent Information Technology and Security Informatics*, pages 63–67, 2010.

GROUP MEMBERS:

- Honorine Gnonfin
- Binta Sow
- Uriel Nguefack