K-Means Algorithm and method to choose the optimal K

Duy Khoa Pham

Student ID: 103515617

SWINBURNE UNIVERSITY OF TECHNOLOGY
SCHOOL OF SCIENCE, COMPUTING
AND ENGINEERING TECHNOLOGIES

Abstract – Clustering is a data analysis technique used to identify similar kinds of subgroups in a dataset. Data points (observations) from one subgroup (cluster) are likely similar and different from other subgroups. This report will illustrate the K-Means algorithm, one of the most well-known clustering methods that is based on mathematical analysis, and provide a method of choosing the optimal value for K. Furthermore, the report gives a demonstration of this algorithm's implementation on a 2D dataset and its application in image compression.

1. Introduction

Consider the following scenario: a chief marketing executive wants to create a promotion policy for different groups of customers based on their involvement with the company (e.g., yearly paid money, year of becoming customer, age and occupation, etc.). The company has a lot of data for many customers, but they need to determine a method to divide customers into different groups (clusters). In this case, K-Means clustering can be implemented.

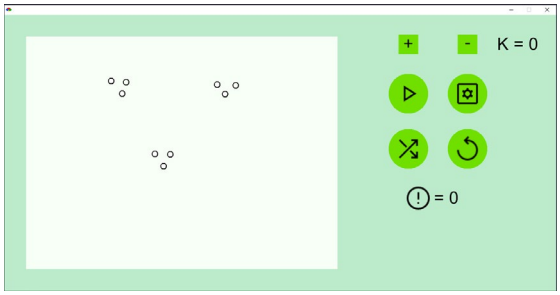

Figure 1. Original data

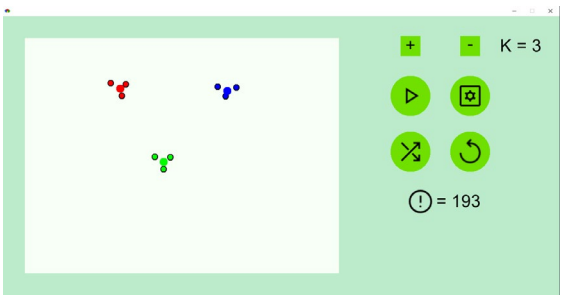

Figure 2. Clustered data

The K-Means algorithm is a basic unsupervised learning algorithm to solve clustering problems. It partitions N observations into K clusters where each observation is assigned to the cluster with the closest mean serving as a prototype of the cluster [1]. To better demonstrate clustering, points in a two-dimensional plane are grouped into 3 subsets.

2. Mathematical model

2.1. Input and Output

Provided N observations: $X = [x_1, x_2, \dots, x_N] \in R^{d \times N}$, in which each observation is an N-dimensional vector and K (< N) is the number of clusters observations are required to be classified into. From the input of N and K, the algorithm needs to calculate each cluster's center point: $m_1, m_2, \dots, m_K \in R^{d \times K}$ and their label.

2.2. Loss function

For labeling, we would use the "One-hot" method. Particularly, with each observation x_i, let its label vector be $y_i = [y_{i1}, y_{i2}, \dots, y_{iK}]$. If x_i belongs to cluster k, $y_{ik} = 1$ and $y_{ij} = 0$, $\forall j \neq k$. Since each observation only belongs to one group, only one element in its label vector will have a value of 1, while the rest will all be set to 0. Therefore, we have this equation:

$$\sum_{k=1}^{K} y_{ik} = 1 \qquad (1)$$

When an observation x_i is assigned to a cluster k, there would be an error margin denoted by its Euclidean distance to the cluster's center $(x_i - m_k)$. This distance will be squared for ease of determining its smallest absolute value. As x_i is in the k cluster (i.e., $y_{ik} = 1$ and $y_{ij} = 0$, $\forall j \neq k$) we have the following:

$$y_{ik}\|x_i - m_k\|^2 = \sum_{j=1}^{K} y_{ij}\|x_i - m_j\|^2 \qquad (2)$$

To optimize the algorithm's loss function, all squared errors of N observations will be summed. Hence, we have the following model to determine the total squared error for a set of observations:

$$f(Y, M) = \sum_{i=1}^{N}\sum_{j=1}^{K} y_{ij}\|x_i - m_j\|^2 \qquad (3)$$

Where:

$Y = [y_1, y_2, \dots, y_N]$ is an observation's label vector matrix

$M = [m_1, m_2, \dots, m_N]$ is the clusters' centerpoint matrix

In other words, from equation (1), we need to optimize the algorithm with regard to minimizing the following output:

$$Y, M = argmin_{Y,M} \sum_{i=1}^{N} \sum_{j=1}^{K} y_{ij} \|x_i - m_j\|^2 \quad (4)$$

2.3. Optimization algorithm for loss function

Equation (4) depends on two variables, Y and M. As a result, there are two approaches to calculating the minimum by alternatively setting a fixed value for one variable and modifying the other.

2.3.1. M fixed, Y variable

As a centerpoint M is already provided, the mathematical problem is simplified. A label vector needs to be determined in order that the total squared error in equation (4) can be minimized.

$$y_i = argmin_{y_i} \sum_{j=1}^{K} y_{ij} \|x_i - m_j\|^2 \quad (5)$$

From equation (1), and given that only one element in the label vector matrix will have the value of y_i = 1, equation (5) can be simplified as well:

$$j = argmin_j \|x_i - m_j\|^2 \quad (6)$$

From the below equation, $\|x_i - m_j\|^2$ represents the squared distance from an observation x_i to its centerpoint m_j. Therefore, it is conclusive that each observation x_i will belong to the cluster whose centerpoint m_j it is closest to.

2.3.2. Y fixed, M variable

With each observation's cluster provided, the optimization problem is now equivalent to mathematically finding a new centerpoint for an established cluster so that the total squared error in equation (4) can be minimized.

$$m_j = argmin_{m_j} \sum_{i=1}^{N} y_{ij} \|x_i - m_j\|^2 \quad (7)$$

As this is a differentiable and convex function on $i \in [1, \mathrm{N}]$, we can take its derivative to find any local optimum and their respective root.

Let $h(m_j) = \sum_{i=1}^{N} y_{ij} \|x_i - m_j\|^2$:

$$\frac{\partial h(m_j)}{\partial (m_j)} = 2 \sum_{i=1}^{N} y_{ij}(x_i - m_j)$$

Solving this derivative for its root:

$$m_j \sum_{i=1}^{N} y_{ij} = \sum_{i=1}^{N} y_{ij} x_i$$

$$\Rightarrow \quad m_j = \frac{\sum_{i=1}^{N} y_{ij} x_i}{\sum_{i=1}^{N} y_{ij}} \quad (8)$$

From equation (1), $y_{ij} = 1$ is conclusively synonymous to x_i belonging to cluster m_j. Therefore, the denominator in equation (8) also represents how many N observations are in the cluster m_j. The nominator, on the other hand, represents the sum of N observations in the cluster m_j. Thus, it is proven that m_j is determined by the average of all observations in a cluster j.

2.4. Synopsis

There are 5 procedural stages:

Stage 1 – Choose an initial K as the number of clusters data points are to be classified into.

Stage 2 – Assign each observation to the cluster whose centerpoint it is closest to.

Stage 3 – Stop the algorithm if the latest iteration produces no output change compared to its immediate predecessor.

Stage 4 – Update each cluster's centerpoint by calculating the average of all observations in that cluster.

Stage 5 – Reiterate the algorithm from stage 2.

2.5 Discussion

Convergent: The loss function produces a positive output whose value will be reduced every time stage 2 is executed. Therefore, it is conclusive that the function is complete and will stop after a finite number of iterations.

Applicable for many types of data: This method is not restricted to solving purely numeric problems. It can also be adapted to work with other types of data (binary, category, etc.).

Other distance metrics: Aside from the Euclidean method, this algorithm could also work with the Manhattan and Minkowski [2] distance model.

Number of clusters (K): A pre-defined value for K is required. In reality, it is oftentimes challenging to estimate an exact value for K. The Elbow method can be applied to solve this problem.

2.6 Elbow method

After the "Within-Cluster-Sum of Squared" (WSS) for each value of K is computed, a clear downward trend for the value of this index can be observed as K increases. The K at which WSS begins to plateau is selected [3].

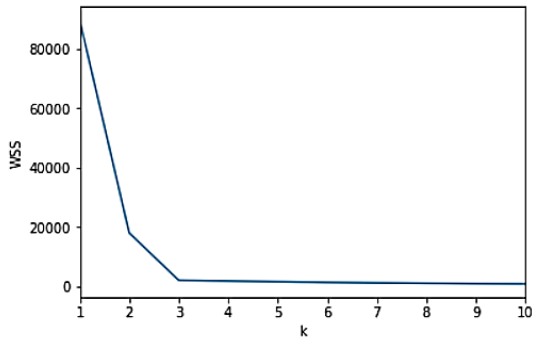

Figure 3. (WSS, k) graph

In Figure 3, the plot for (WSS, K) is resemblant to a human arm, with the elbow at K = 3 being elected as the ideal K selection for this dataset.

3. Application

An application in image compression is illustrated to help better visualize the benefits of K-Means clustering. The first demonstration implements K-Means clustering on a 2D dataset (with x, y representing each point's coordinates on a 2D Euclidean plane), utilizing Ruby and Gosu to build a Graphical User Interface.

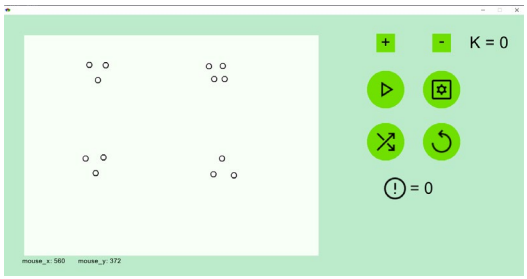

Figure 4. Original data of custom code

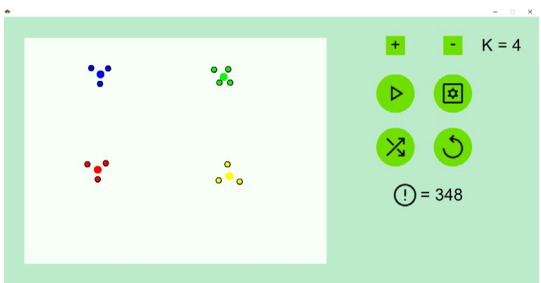

Figure 5. Data clustered into 4 groups

Furthermore, image compression is illustrated with the "matplotlib" and "sklearn" Python libraries to create a small application implementing K-Means clustering which is shown in Figure 6.

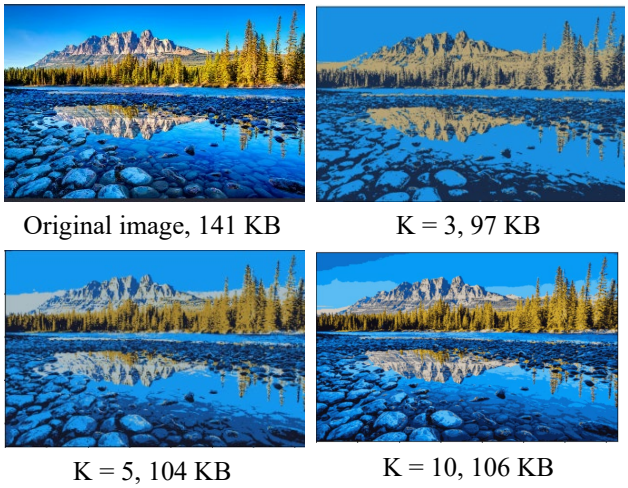

Figure 6. Image compression

4. Future work

As the K-Means clustering algorithm is fairly simple and straightforward, limitations are inevitable. Depending on an initialized cluster's centerpoint, the algorithm could take a significant amount of time to produce output. It is possible, although unlikely, for the algorithm to return an incorrect value (i.e., a local minimum as opposed to a global minimum). Instead of relying on random selection, more advanced techniques could be applied to elect the initial cluster's centerpoint more precisely, reducing time complexity and increasing accuracy.

5. Conclusion

This report evaluates the K-Means algorithm, most prevalently from a mathematical standpoint. It aims to supply sufficient fundamental understanding of K-Means for newcomers, with a demonstration for better visualization. The report briefly introduces the Elbow method which is used to select an initial value for K.

6. Acknowledgment

Gratitude to Tuan Dung Lai for his useful "Basic Artificial Intelligence" course and Huu Tiep Vu for his "Basic Machine Learning" blog.

7. Reference

[1] GeeksforGeeks, 2022, Clustering in Machine Learning, <https://www.geeksforgeeks.org/clustering-in-machine-learning/>

[2] Eric U. Oti, Michael O. Olusola, Oberhiri-Orumah Godwin, Chike H. Nwankwo, New K-means Clustering Method Using Minkowski's Distance as its Metric.

[3] Mahendru K., 2019, How to determine the optimal K for K-Means

8. Appendix

The code for the demonstration in this paper can be found below: https://github.com/EspiusEdwards/Image-Compression-Demo.git.