**Table of** **contents**

[1. Introduction 2](#_Toc125389702)

[2. Unsupervised Machine learning and K-means Clustering 2](#_Toc125389703)

[3.Implementation of K-means clustering 4](#_Toc125389704)

[4. Results and Analysis 5](#_Toc125389705)

[5.Stopping Criteria for K-Means Clustering 9](#_Toc125389706)

[6.Discussions and conclusions 9](#_Toc125389707)

[7. References 10](#_Toc125389708)

[8. Appendix 11](#_Toc125389709)

**Table of figures**

[Figure 1 - K-means clustering algorithm 3](#_Toc125395994)

[Figure 2-Standard deviation 4](#_Toc125395995)

[Figure 3 -3 cluster silhouette 5](#_Toc125395996)

[Figure 4-3 cluster centroid. 5](#_Toc125395997)

[Figure 5 - 4 cluster silhouette. 6](#_Toc125395998)

[Figure 6 -4 cluster centroid 6](#_Toc125395999)

[Figure 7 - 5 cluster silhouette. 7](#_Toc125396000)

[Figure 8-histrogram 8](#_Toc125396001)

[Figure 9 - Optimal number showing 8](#_Toc125396002)

**Unsupervised Machine Learning and K-means clustering**

# 1. Introduction

“Unsupervised learning” is a method of “machine learning” where the consumers aren't required to supervise over model. it permits the model to work alone effectively and identify prior unrecognized information and patterns. It primarily handles with unlabeled data. Comparing to “supervised learning”, “Unsupervised learning” algorithms enable users to conduct out more advanced processing tasks. “Unsupervised learning”, is usually unpredictable than other techniques of “natural learning”. Algorithms for “unsupervised learning” include “neural networks, anomaly detection, and clustering”.

In terms of “unsupervised learning”, clustering seems to be the core concept. The major focus is on identifying a pattern or structure in a group of unorganized data. Unsupervised Learning, when there are any natural clusters or groupings in the data, clustering algorithms may process them and recognize them. Additionally, can change number of clusters that algorithms need to identify. That provides the granularity of such groupings to be modified.

Among the unsupervised algorithms that uses input data which excludes labeled responses is K-Means clustering. The clustering concerns in machine learning as well as the data science are resolved using the unsupervised learning algorithm K-Means Clustering. In this study, will understand what the K-means clustering method is, how it operates, and how to develop it in MATLAB.

# 2. Unsupervised Machine learning and K-means Clustering

In data science, clustering seems to be a useful method. It's a method for determining cluster structure in a dataset which is differentiated by the maximum level of similarities inside a cluster as well as the best performance of dissimilarity across clusters. The very first clustering method utilized by biologists & social scientists was hierarchical clustering, but cluster analysis evolved into a subfield of statistical multivariate analysis. As terms of machine learning, it's also unsupervised. Clustering techniques can be classified broadly as nonparametric approaches as well as statistical model-based approaches from a statistical perspective. The variables are derived from a “mixture probability model”, so a “mixture possibility” approach to clustering is utilized according to the probability model-based approaches.

Partitional approaches typically presumptively assume that perhaps the training dataset may be portrayed by finite cluster prototypes with unique objective functions. Therefore, it is crucial for partition algorithms to provide the dissimilarity (or separation) among a point as well as a cluster prototype. The k-means algorithm is acknowledged as the most established as well as widely used partitioning algorithm. K-means clustering has received a significant amount of attention in the literature and has been used in a wide range of relevant fields. These k-means clustering algorithms must be provided with a set of clusters up front because initializations frequently have an impact on clusters. The clustering concerns in machine learning and the data science were resolved by using unsupervised learning algorithm.K-Means Clustering. The algorithm's fundamental objective is to decrease the total distances among each data point as well as its corresponding clusters.

The algorithm begins with an unlabeled dataset as its input, partitions it into k clusters, and afterwards continues the procedure till it running out of clusters for using. Within that algorithm, the k parameter must be predetermined.

The 2 significant functions of the “k-means clustering algorithm” are:

* Utilizes an iterative method for selecting the better value for K center points or centroids.
* Align every data point only with nearest k-center. A cluster is formed by the data points that are nearby for a specific k-center.

As a result, each cluster is distinct from the other ones as well as contains data points with some similarities.

The K-means Clustering Algorithm is illustrated in the figure below:

Diagram

Description automatically generated

Figure 1 - K-means clustering algorithm

# 3.Implementation of K-means clustering

When going through the implementation using the function "gen\_kmeansdata" to generate a data matrix with 10750058 observations. The "rng default" command is used to set the random number generator to its default settings., determin the number of rows and columns of the data matrix "X" using the size function.

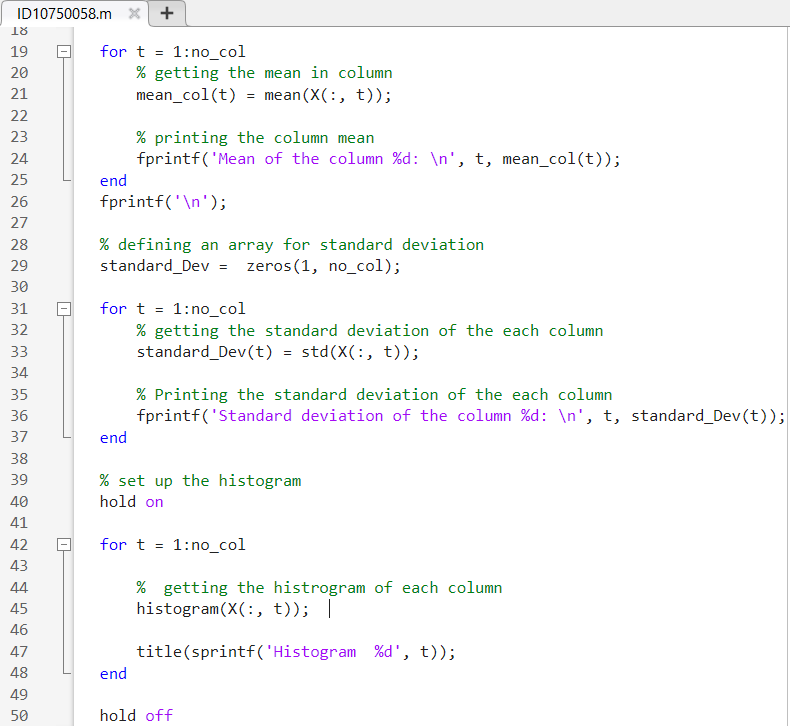


Figure 2-Standard deviation

After that defines an array X that contains the values from 3 to 5 and an array "value" that will store the mean silhouette scores for each value. For each value, it performs k-means clustering on the matrix X with the current value of K. It also plots the silhouette scores as a function of K and then it repeats the loop but this time it plots the clusters and the cluster centroids with different colors. using the color map hsv(j) where j is the current value of K. It also adds a title and legend for each plot.

# 4. Results and Analysis

The range across each data point in 1 cluster and the data points in its closet clusters is shown on a silhouette plot. Its measure the range from 0 to -1.

**When K=3**

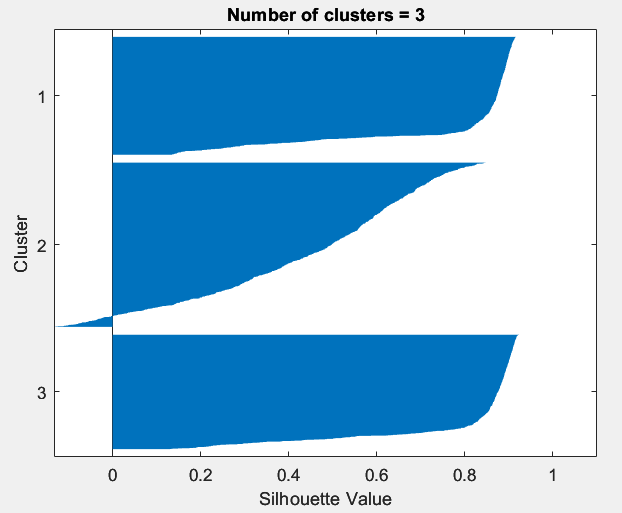


Figure -3 cluster silhouette

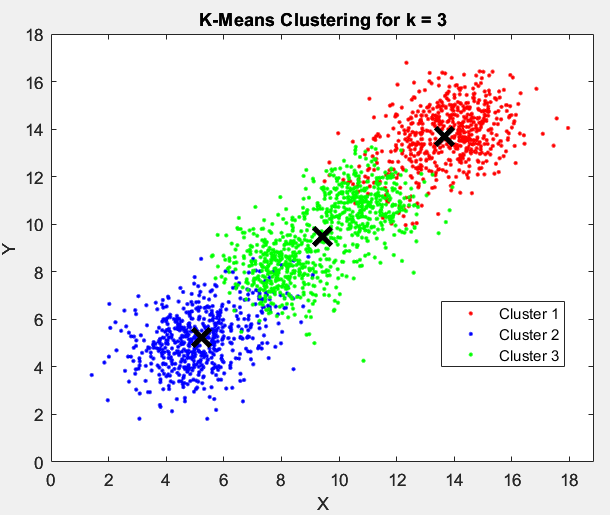


Figure -3 cluster centroid.

However, the 3rd cluster contains two points with low silhouette values, the third clusters having few points with negative values, but the 3 values are 0.9911499 ,0.86014 and 0.878334 they also close.

**When K=4**

Chart

Description automatically generated with medium confidence

Figure - 4 cluster silhouette.

Chart, scatter chart

Description automatically generated

Figure -4 cluster centroid

Going through figures 5 and 6, the outcome appears to be much more acceptable and positive, without negative points in the four-cluster silhouette value. Each value is displayed as 0.89476, 0.85997, 0.87954, and 0.88966 and all the results may so close to 1.

**When K=5**

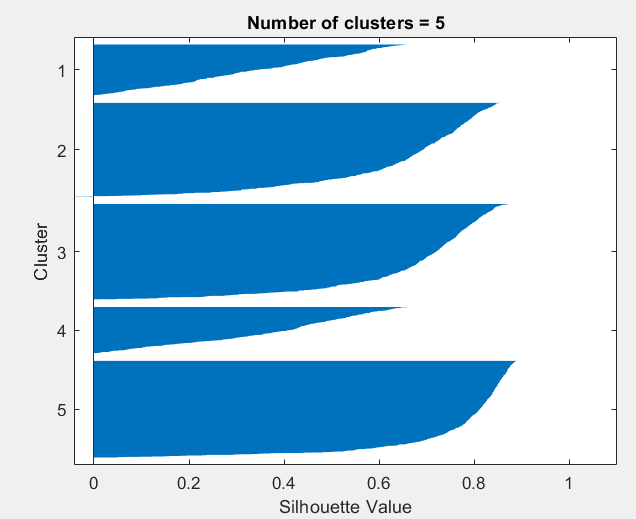
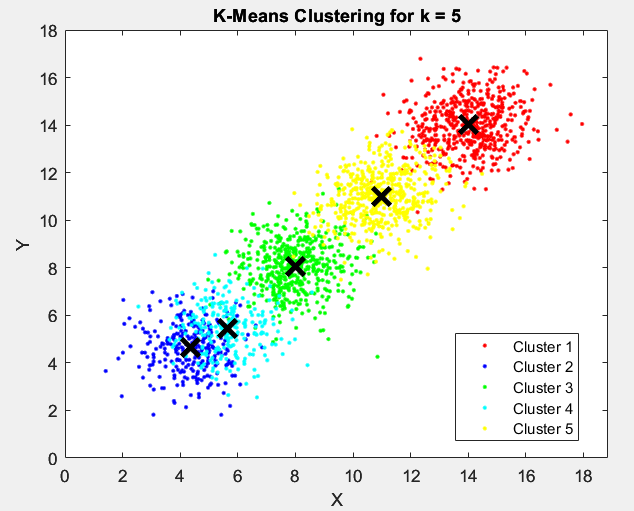
****

Figure - 5 cluster silhouette.

****

In figure 7 Two values appear low in the figure 7 silhouette, at 0.662378 and 0.655723 respectively. The 5 cluster is not seeming positive, as shown by the other 3 values that are close to one.

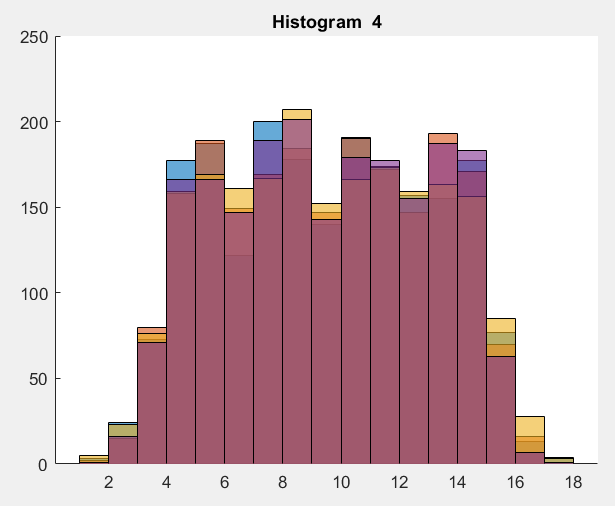


Figure -histrogram

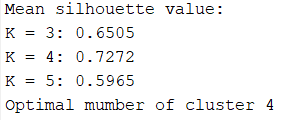
In figure 8 shows the histogram of cluster 4

Chart, line chart

Description automatically generated

Figure - Optimal number showing

As the optimal number of cluster shows in the figure 9,with evaluate of the k means



# 5.Stopping Criteria for K-Means Clustering

The K-means algorithm able to be stopped basically via adopting one of 3 stopping factors:

* Centroids of newly formed clusters do not change
* Points remain in the same cluster
* Maximum iterations are completed.

Unless the centroids of freshly generated clusters would not be modifying, we able to stop the algorithm. when the centroids for every one of the clusters remain the same even after several iterations, then algorithm isn't really going to be catching up any new patterns, then training may stop. When the points are remained in the exact same cluster though after training the algorithm over several iterations, that is another obvious indication that should stop the training phase.

# 6.Discussions and conclusions

The k-means algorithm is acknowledged as partitioning algorithm. It's an iterative method that splits the unlabeled data into k distinct clusters, each of includes a one dataset & contains a collection of features. K-Means clustering has received a significant amount of attention in the literature and has been used in a wide range of relevant fields. Code used to analyze data for the student ID as 10750058, Using Covariance matrix and correlation matrix random variable made up of observations and rows represent observations. As result data points get 2052. Using silhouette function passing original data. From that got optimal number of clusters as four by using silhouette score finding. And the K-means algorithm can be stopped basically via adopting one of 3 stopping factors.

As limitations of k-means clustering it is a form of clustering algorithm that focuses on the fact that all clusters are spherical. Not that all clustering algorithms make this relatively strong assumption. In circumstances when clusters typically have uneven shapes, it does not work as well. And also, there’s several drawbacks and limitations as responsive to scale, when categorical characteristics are required in the dataset, difficult to include categorical variables don't really function as well., Sensitive to outliers in contrast to some other clustering methods, k-means clustering incorporates all of the data points in a cluster. The algorithm is therefore somewhat susceptible to significant outliers., This indicates that if the starting conditions are changed, we might not achieve the same results because the seed choice is sensitive., Choosing the number of clusters and battling high dimensional data also more challenging.

# 7. References

1. ‌www.mathworks.com. (n.d.). k-Means Clustering - MATLAB & Simulink. [online] Available at: [https://www.mathworks.com/help/stats/k-means-clustering.html](https://www.mathworks.com/help/stats/k-means-clustering.html%20) .
2. Ellis, C. (2022). When to use K-means clustering. [online] Crunching the Data. Available at: [https://crunchingthedata.com/when-to-use-k-means-clustering/.](https://crunchingthedata.com/when-to-use-k-means-clustering/)

‌

# 8. Appendix

rng default;

% creates the data matrix using k-means

X = gen\_kmeansdata(10750058);

disp(X)

% determine the row size & columns

size(X,1)

size(X,2)

% getting the column amount

no\_col = size(X, 2);

% mean of the column

mean\_col = zeros(1, no\_col);

for t = 1:no\_col

% getting the mean in column

mean\_col(t) = mean(X(:, t));

% printing the column mean

fprintf('Mean of the column %d: \n', t, mean\_col(t));

end

fprintf('\n');

% defining an array for standard deviation

standard\_Dev = zeros(1, no\_col);

for t = 1:no\_col

% getting the standard deviation of the each column

standard\_Dev(t) = std(X(:, t));

% Printing the standard deviation of the each column

fprintf('Standard deviation of the column %d: \n', t, standard\_Dev(t));

end

% set up the histogram

hold on

for t = 1:no\_col

% getting the histrogram of each column

histogram(X(:, t));

title(sprintf('Histogram %d', t));

end

hold off

covariance = cov(X); %covariance of the matrix

correlation = corrcoef(X); %correlation of the matrix

% loop for repeating different k-means value

K = 3:5;

% defining an array for silhouette values

value = zeros(size(K));

for t = 1:length(K)

% getting the current value

[c\_idx, centroids] = kmeans(X, K(t));

% getting silhouette score

sil = silhouette(X, c\_idx);

value(t) = mean(sil);

figure;

silhouette(X, c\_idx);

title(sprintf('Number of clusters = %d', K(t)));

end

fprintf('Mean silhouette value:\n');

for t = 1:length(K)

fprintf('K = %d: %.4f\n', K(t), value(t));

end

% finding the optimal number

[~, max\_index] = max(value);

fprintf('Optimal mumber of cluster %d\n', K(max\_index))

% plotting the mean silhoutte score

plot(K, value, '-o')

xlabel('Number of clusters (K)')

ylabel('Mean silhouette value')

% setting the values for K

K = 3:5;

for j = K

% Using k-means

[c\_idx, centroids] = kmeans(X, j);

% getting the colour map

cmap = colormap(hsv(j));

% ploting the clusters & the cluster centroids with colours

figure;

gscatter(X(:,1), X(:,2), c\_idx, 'rbgcy','...',7)

% ploting the clusters & the cluster centroids

hold on;

plot(centroids(:,1), centroids(:,2), 'kx', 'MarkerSize', 15, 'LineWidth', 3);

title(sprintf('K\_Means Clustering for the k = %d', j));

leg\_strings = cell(1, j);

for t = 1:j

leg\_strings{t} = sprintf('Cluster %d', t);

end

legend(leg\_strings);

xlabel('X');% mark the x-axis

ylabel('Y');% mark the y-axis

hold off;

end