Student Name: Sai Vara Prasad Lekkalapudi

Student ID: 23076885

Course: Msc Data Science with Sandwich Placement

GitHub Link:
https://github.com/SaiVaraPrasadL/MachineLearningIndividualAssignment

Table of Contents

A Comparative Study of Distance Metrics in k-Means Clustering

Introduction:

Overview

In machine learning, clustering is an unsupervised learning method that groups related data points together based on similar features. Clustering methods find patterns and structures in the data rather than depending on target variable (independent variables), like supervised learning does.

The effectiveness of clustering depends, where distance metrics such as Euclidean, Manhattan, Cosine, and Mahalanobis distances are used to quantify how similar or different data points are. The goal of clustering is to maximize intra-cluster similarity , while minimizing inter-cluster similarity. Several clustering algorithms exist, each with its own approach to forming clusters. However, k-Means remains one of the most popular due to its simplicity, efficiency, and effectiveness in handling large datasets.

K-Means Clustering:

k-Means clustering is a popular unsupervised learning method that splits a dataset into k distinct groups. A centroid, or center point, represents each cluster in this centroid-based clustering algorithm, and data points are assigned to the nearest centroid based on a chosen distance metric.

Limitations of Different Distance Metrics in K-Means:

The choice of distance measure has a direct effect on cluster formation in k-Means. Although different metrics can produce different results and lead to poor clustering effectiveness, Euclidean distance is the most often used measure. These are a few restrictions on various distance measurements.

a. **Euclidean Distance:**
   - Clusters are spherical and of equal size.
   - Struggles with high-dimensional data due to the curse of dimensionality (Aggarwal et al., 2001).

b. **Manhattan Distance:**
   - Measures distance along axes rather than direct diagonal distance.
   - Performs poorly if clusters have diagonal orientations or varying densities.

c. **Cosine Similarity:**
   - Measures the angle between vectors instead of raw distances.
   - It is not appropriate for compact spherical clusters, but it performs well on text data and high-dimensional sparse data.

Methodology:

K-Means Clustering Implementation:

A centroid-based clustering technique called k-Means divides a dataset into k groups according to a selected distance metric; the following distance metrics are studied in this study:

Dataset Selection & Preprocessing:

- **Dataset:** Mall Customer Segmentation Dataset.
  - **Features:**
    - CustomerID: Unique number of the customer.
    - Gender: Indicates Male or female of the customer.
    - Age: Age of the customer.
    - Annual Income: Indicator of customer buys.
    - Spending Score: Gets the score based on the spending income.

- **Data Preprocessing**
  - By maintaining a zero mean and unit variance, StandardScalar standardization prevents characteristics with greater values such as income from reducing the clustering process.

Choosing the Optimal Number of Clusters (k):

Efficient clustering depends on choosing the appropriate number of clusters. There are two popular approaches:

A. Elbow Method:

- Computes the Within-Cluster Sum of Squares (WCSS) for different values of k.
- An "elbow point" is reached when the gain in variance reduction begins to decline, as seen by plotting WCSS versus k.
- The ideal k is at the "bend" of the curve.

B. Silhouette Score:

- Compares each point's distance from other clusters to how well it fits within its assigned cluster.
- Better clustering is indicated by a higher silhouette score (nearer to 1).
- Formula:

$$s(i) = \frac{b(i) - a(i)}{max\left(a(i), b(i)\right)} \; where:$$

$$a(i) = average\ intra - cluster\ distance\ (cohesion)$$

$$b(i) = average\ nearest - cluster\ distance$$

K-Means with different Distance Metrics:

A. Euclidean Distance:

- The standard metric used in k-Means clustering.
- calculates the straight-line distance between two points.
- clusters are assumed to be spherical and compare size.
- Formula:

$$d(x, y) = \sqrt{\sum(x_i - y_i)^2}$$

B. Manhattan Distance:

- It measures distance using a grid-like path rather than a straight line.
- stronger when the data contains outliers, or extreme values.
- Formula:

$$d(x, y) = \sum|x_i - y_i|$$

C. Cosine Similarity:

- Computes the cosine of the angle between two vectors rather than raw distances.
- Used when grouping shouldn't be impacted by magnitude disparities.
- Good for sparse and high-dimensional data, such text documents.
- Formula:

$$cos(\theta) = \frac{x \cdot y}{||x|| \cdot ||y||}$$

Results and Analysis:

Elbow Method Results for optimal K – Selection

Euclidean Distance (k = 5):

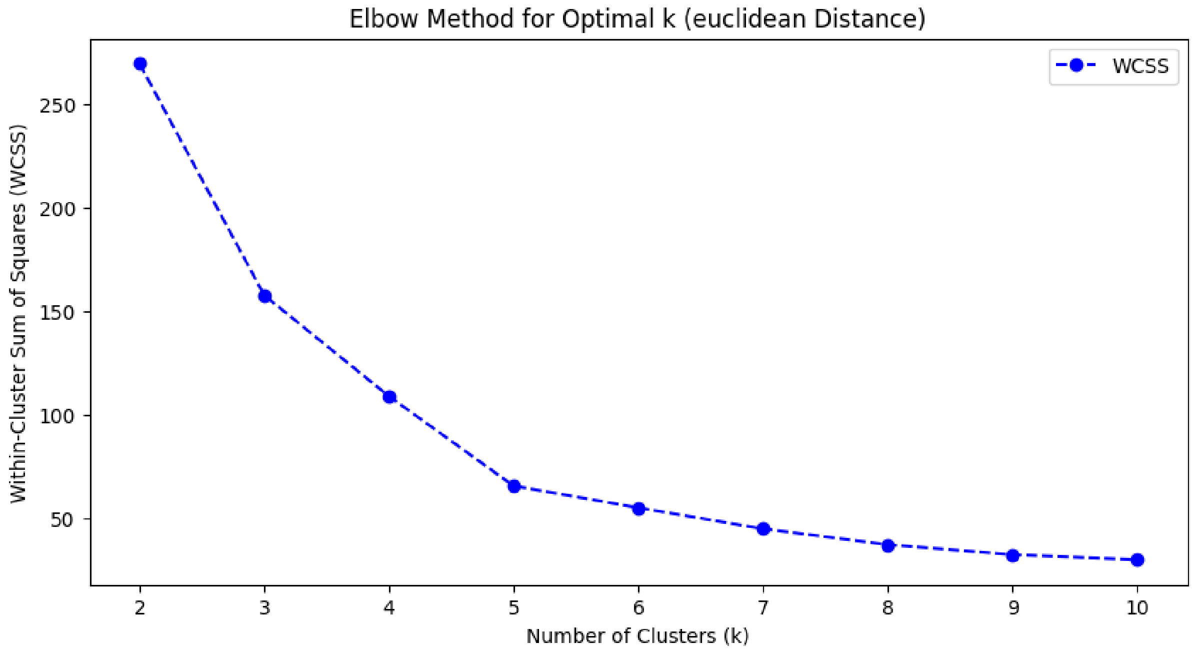

Fig 1: Elbow Method for Euclidean Distance.

- Five clusters offer the optimal trade-off between compactness and separation, as seen by the obvious bend in the elbow plot for Euclidean distance at k = 5.
- This implies that, when evaluated by geometric distance, groups of customers in the dataset naturally form five different segments.

Manhattan Distance (k = 5):

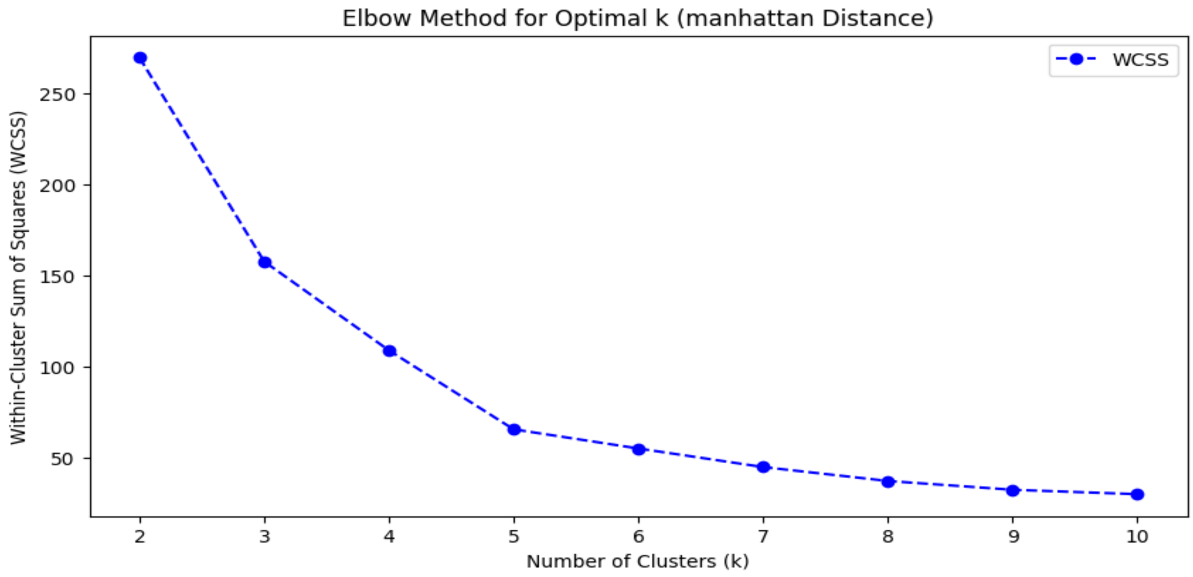

Fig 2: Elbow Method for Manhattan Distance.

- The elbow technique for Manhattan distance offers k = 5 as the ideal value, comparable to Euclidean distance.
- The clusters still follow a similar pattern, but their borders may change somewhat since Manhattan distance evaluates absolute differences rather than squared differences.

Cosine Distance (k = 2):

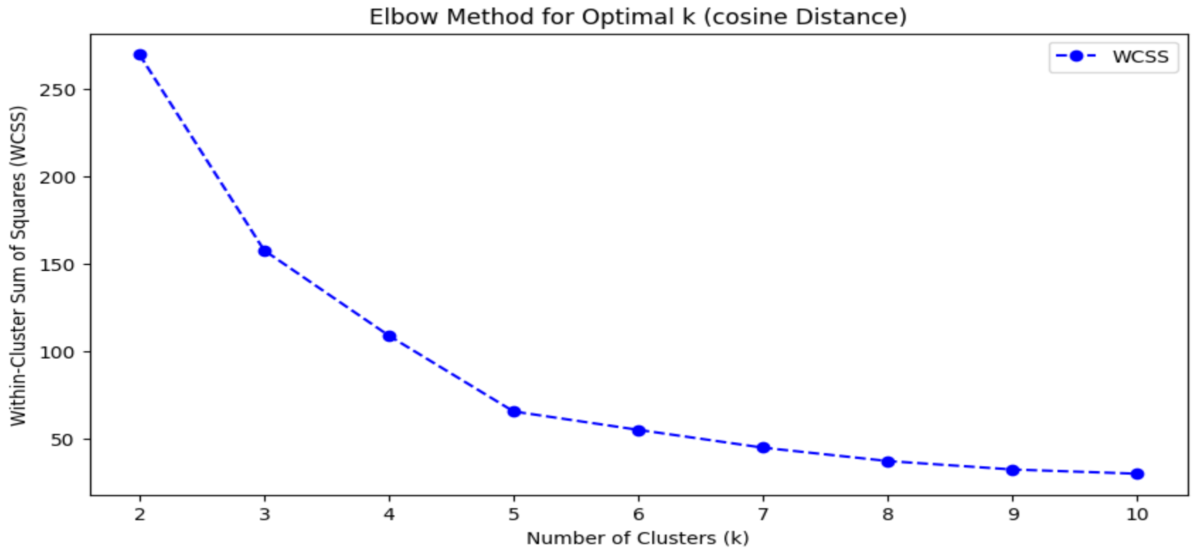

Fig 3: Elbow Method for Cosine Distance

- Based on to the elbow plot for cosine similarity, k = 2 is ideal.

- Data points fall into two major categories according to their orientation rather than their numerical proximity because cosine distance evaluates directional similarity rather than absolute distance.

Silhouette score:

Euclidean Distance Silhouette Score Plot:

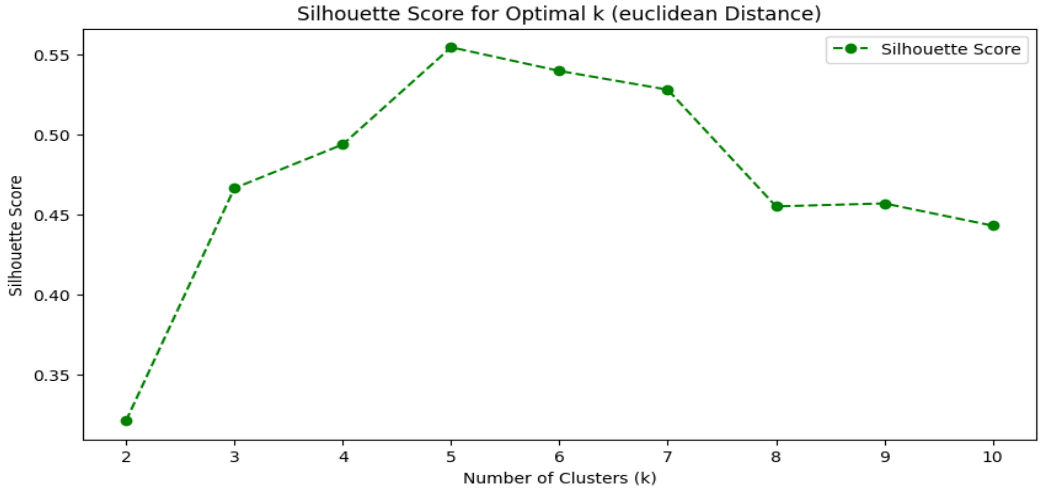

Fig 4: silhouette score using Euclidean distance

- Five clusters offer the finest separation and coherence, as indicated by the highest silhouette score at k = 5.

- The silhouette score falls when k rises over 5, suggesting that larger clusters result in smaller, less significant groupings.

Manhattan Distance Silhouette Score Plot:

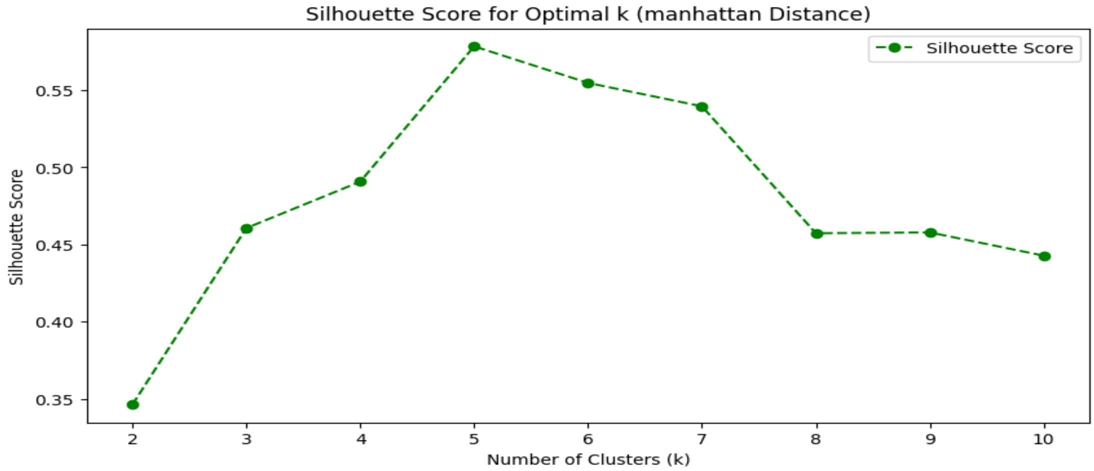

Fig 5: Manhattan distance for silhouette score.

- Five ideal clusters are confirmed by the greatest silhouette score, k = 5, which is comparable to Euclidean distance.

- As additional clusters are added, the silhouette score decreases, indicating a decline in clustering quality, after peaking at k = 5.

Cosine Distance Silhouette Score Plot:

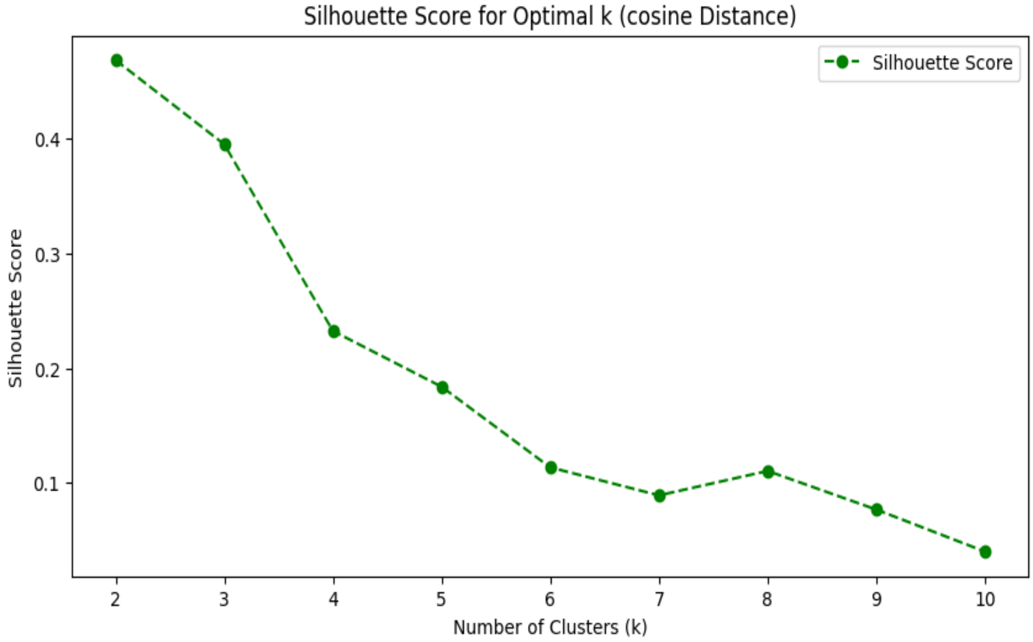

Fig 6: Cosine Distance for silhouette score

- In contrast to Manhattan and Euclidean distances, k = 2 has the greatest silhouette score. This implies that the dataset naturally divides into two major groups rather than five when Cosine similarity is used.

- As K rises, the score drastically decreases, suggesting that more clusters lead to needless fragmentation.

Comparison of Mean Squared Error (MSE) Across Distance Metrics

The Mean Squared Error (MSE), which evaluates how well data points match their assigned cluster centers, is used to assess the accuracy of clustering. Better clustering performance is indicated by a lower MSE.

Distance Metric	Optimal K	MSE	Observations
Euclidean Distance	5	0.1639	Provides strong clustering performance with low error.
Manhattan Distance	5	0.1639	Similar performance to Euclidean, reinforcing cluster consistency.
Cosine Distance	2	0.6742	Significantly higher error, suggesting poor clustering alignment.

Table 1: Comparison of Mean Squared Error

Cluster visualization:

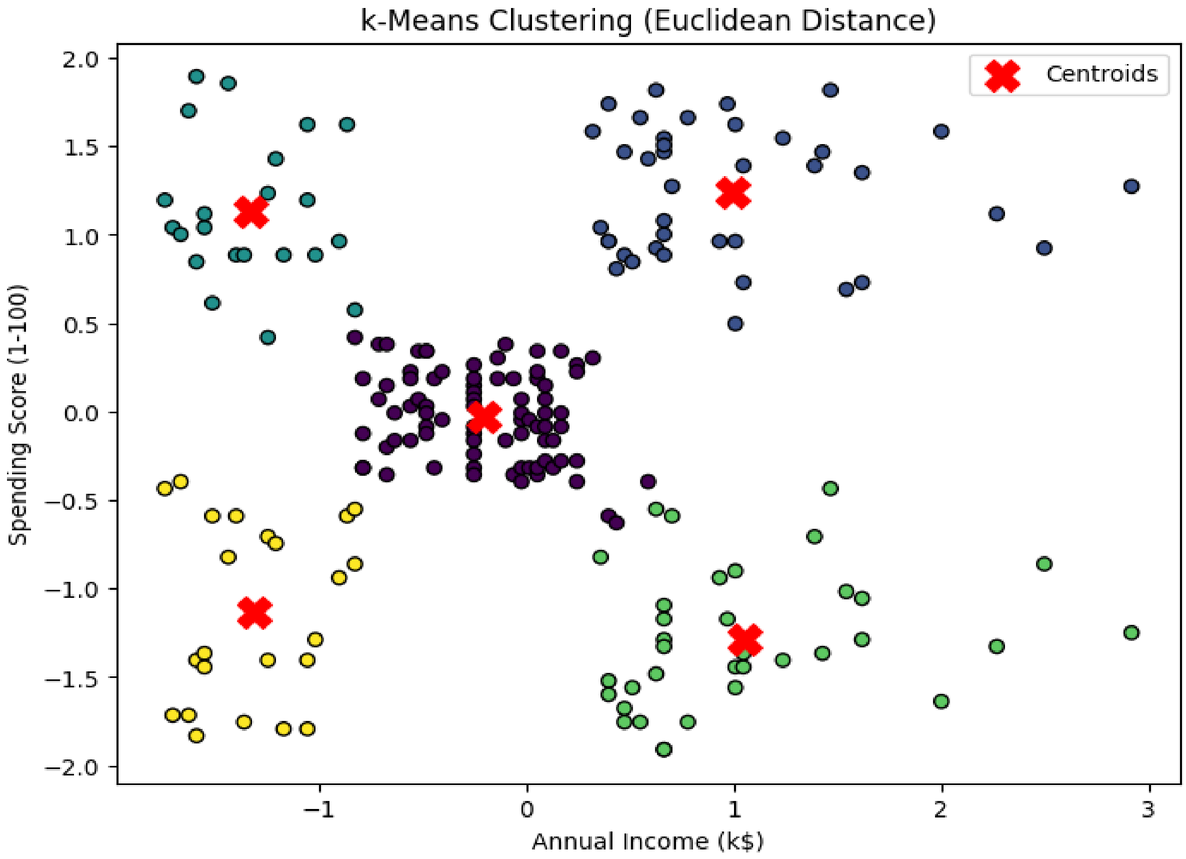

Fig 7: K-Means Clustering using Euclidean Distance.

- Five Clusters formed.
- The Euclidean distance-based clustering shows distinct, compact, and round clusters.
- The centroids, shown by red crosses, are arranged to reduce the sum of squared distances between each data point and the closest cluster center.
- MSE: 0.1639, indicating a well-optimized clustering structure.
- The clusters clearly separate customers based on income and spending score:
    - One cluster includes low-income, high-spending individuals.
    - Another consists of high-income, low-spending individuals.
    - Others are variations of moderate income and spending behaviors.

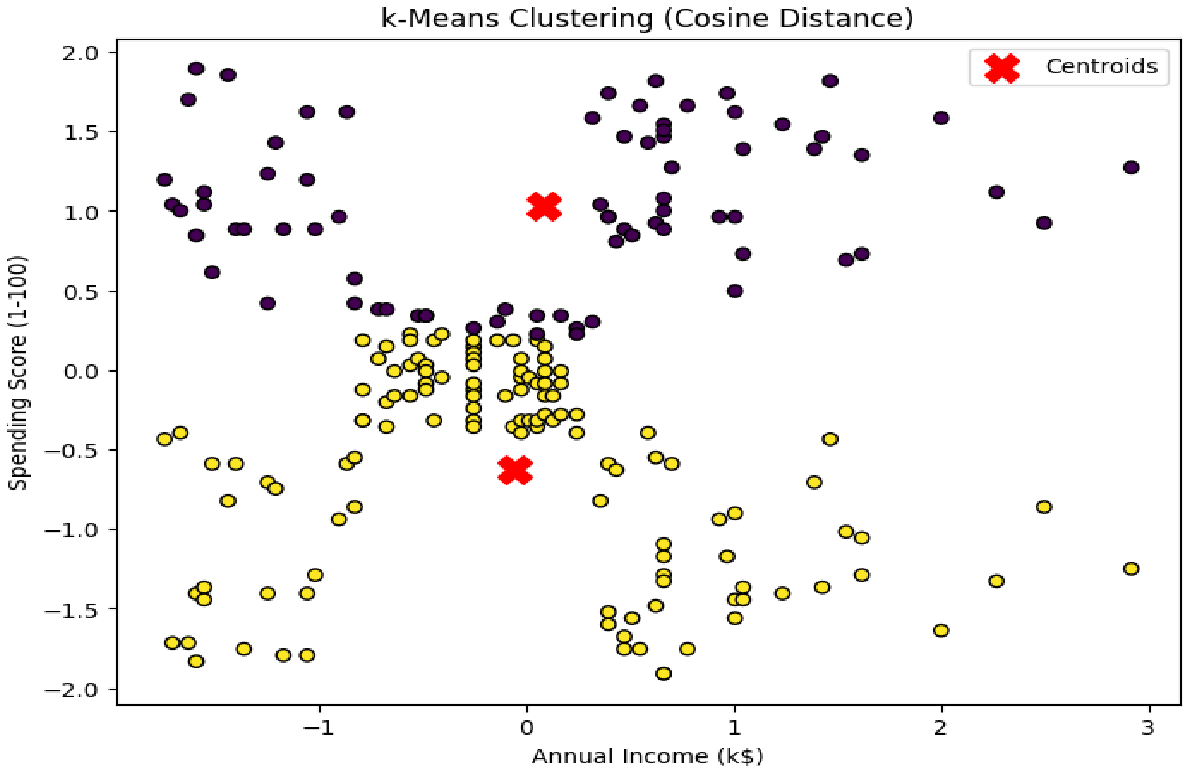

Fig 8: k-Means Clustering Using Cosine Distance Metric.

- Two clusters were formed.
- The cosine similarity divides customers into just two clusters, in contrast to the Manhattan and Euclidean distances.
- Cosine similarity is not a good fit for this dataset since it calculates the angle between vectors rather than their absolute distances.
- Less meaningful segmentation results from the two large clusters' inability to represent the various expenditure and income categories.
- The substantially higher MSE of 0.6742 indicates that this approach is not very effective.

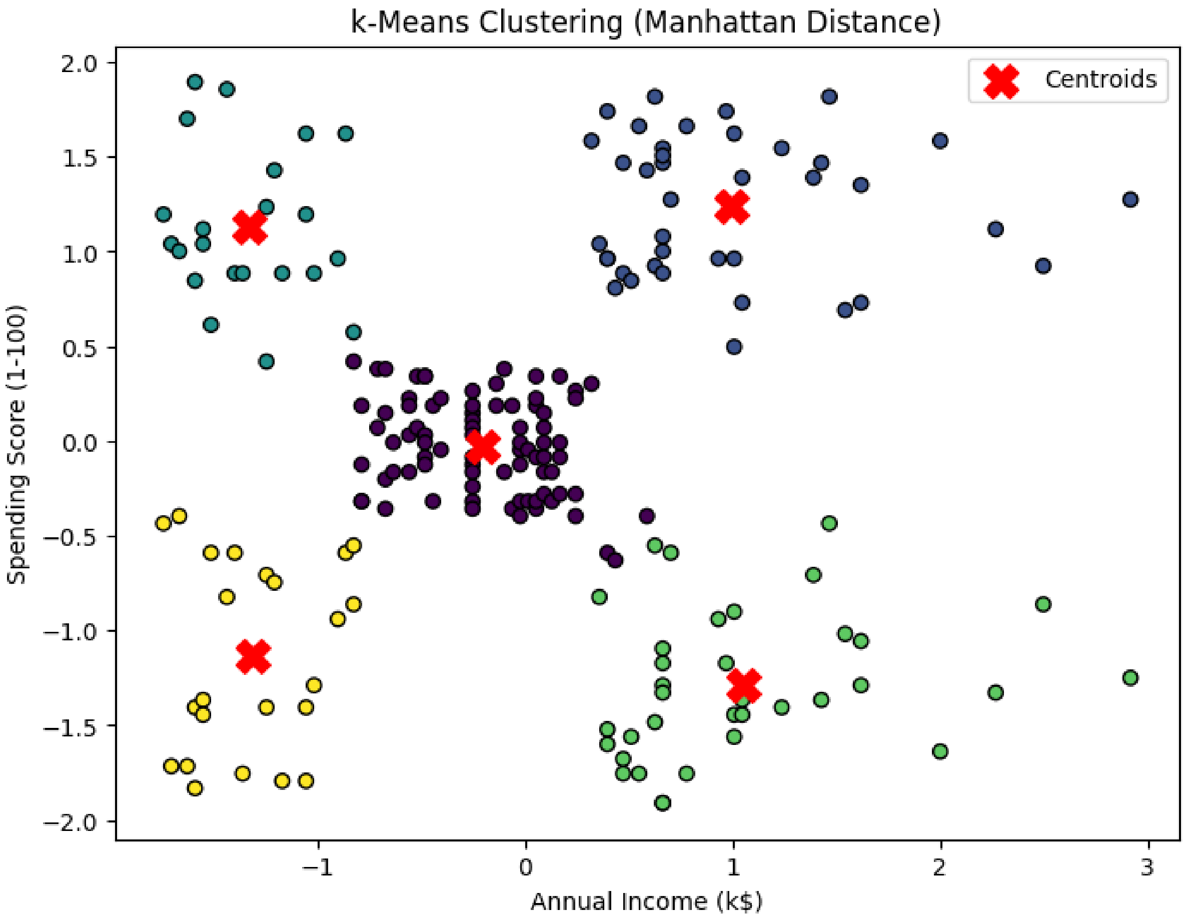

Fig 9: k-Means Clustering Using Manhattan Distance Metric.

- Five clusters were formed.
- Manhattan distance clustering closely resembles Euclidean distance clustering. Because Manhattan distance uses absolute differences to calculate distances, the clusters have a little more grid-like appearance.
- Both the segmentation approach and the centroids are essentially unchanged.
- Euclidean distance and clustering performance are same, as indicated by the MSE of 0.1639.

Challenges of K-means clustering:

In general, k-means is easy and simple to use, but using it is not always the right choice to segment data into groups because it may fail. If the clusters are spherical, then it's the right choice to use. While the groups are of different sizes and densities, then the algorithm may not work well, and it doesn't provide the best results.

In those cases, it's preferable to use other alternative approaches such as DBSCAN, GMM and BIRCH.

Real-time Applications:

Customer Segmentation: It is used for calculating the expenses of a customer in daily life.

Fraud detection: Used for detecting the fraud transactions in banks.

Cybercrime identification: It identifies the crime by a single or the same group of the suspects.

Delivery route optimization: It optimizes the route for the delivery drivers based on the traffic conditions.

Conclusion:

In this study, we evaluated how k-Means clustering for customer segmentation was affected by three distinct distance metrics: cosine, Manhattan, and Euclidean. With five distinct groups and the lowest mean squared error (MSE = 0.1639), the findings show that the Euclidean and Manhattan distances yield the best grouping. Cosine similarity, on the other hand, did not perform well and only created two broad clusters with a substantially larger MSE (0.6742), which means that it is not appropriate for this kind of numerical dataset. Clustering performance is greatly impacted by the distance metric chosen, and the properties of the dataset determine which metric is best. Cosine similarity is better suited for high-dimensional sparse data, such as text analysis, whereas Euclidean and Manhattan distances are best for consumer segmentation. All things considered, this study emphasizes how crucial it is to use the right distance measure in order to ensure accurate and significant clustering findings.

References:

J. Han, M. Kamber and J. Pei, Data Mining Concepts and Techniques, Waltham, USA, 2012.

J. Xu, D. Han, K. Li and H. Jiang, "A K-means algorithm based on characteristics of density", *Comput. Sci. Inf. Syst.*, vol. 17, no. 2, pp. 665-687, 2020.

Jain, A.K. (2010) ‘Data clustering: 50 years beyond K-means’, *Pattern Recognition Letters*, 31(8), pp. 651–666. Available at: https://doi.org/10.1016/j.patrec.2009.09.011 (Accessed: 15 March 2025).

Aggarwal, C.C., Hinneburg, A. and Keim, D.A. (2001) ‘On the surprising behavior of distance metrics in high-dimensional space’, *Proceedings of the 8th International Conference on Database Theory (ICDT)*, pp. 420–434. Available at: https://doi.org/10.1007/3-540-44503-X_27 (Accessed: 15 March 2025).

"COSINE DISTANCE, COSINE SIMILARITY, ANGULAR COSINE DISTANCE, ANGULAR COSINE SIMILARITY". *www.itl.nist.gov*. Retrieved 2020-07-11.

Paul E. Black, "Euclidean distance", in *Dictionary of Algorithms and Data Structures* [online], Paul E. Black, ed. 17 December 2004. (accessed TODAY) Available from: https://www.nist.gov/dads/HTML/euclidndstnc.html

Paul E. Black, "Manhattan distance", in *Dictionary of Algorithms and Data Structures* [online], Paul E. Black, ed. 11 February 2019. (accessed TODAY) Available from: https://www.nist.gov/dads/HTML/manhattanDistance.html