How the Choice of *k* Affects k-Means Clustering Performance and Cluster Quality

An Intuitive and Visual Tutorial Using Customer Segmentation Data

Student name: Dharanidhar Beere
Student ID:24077147
Github Link: https://github.com/DharanidharBeere/mall-customers-kmeans-tutorial.git

---

1. Introduction

Clustering is a fundamental task in unsupervised machine learning, where the goal is to discover natural groupings within data without using labelled examples. One of the most widely used clustering algorithms is **k-Means**, favoured for its simplicity, speed, and effectiveness on many real-world problems such as customer segmentation, image compression, and anomaly detection.

Despite its popularity, k-Means requires the user to specify one crucial hyperparameter in advance: the **number of clusters, k**. The choice of k strongly influences the quality and interpretability of the resulting clusters. If k is too small, distinct groups may be forced together (under-clustering). If k is too large, the model may produce artificial or meaningless splits (over-clustering).

This tutorial demonstrates, in a practical and visual way, **how different values of k affect clustering behaviour and performance**. Using a real customer dataset, we analyse clustering quality using the **Elbow Method** and the **Silhouette Score**, and we visualise how cluster structure changes as k varies. By the end of this tutorial, the reader will be able to:

- Understand how k-Means works,
- Recognise the effects of poor choices of k,
- Apply quantitative methods to select an appropriate value of k,
- Interpret clustering results critically.

---

2. How k-Means Works (Intuitive Explanation)

The k-Means algorithm aims to partition a dataset into **k distinct groups**, where each data point belongs to the cluster with the nearest centroid. A **centroid** is the average position of all points assigned to a cluster.

The algorithm proceeds through the following steps:

1. **Initialisation**
   k initial centroids are selected (usually randomly or using k-Means++ initialisation).
2. **Assignment Step**
   Each data point is assigned to the nearest centroid using a distance measure, typically **Euclidean distance**.
3. **Update Step**
   New centroids are computed as the mean of all points assigned to each cluster.
4. **Iteration**
   Steps 2 and 3 repeat until the centroids no longer change significantly or a maximum number of iterations is reached.

Mathematically, k-Means minimises the following objective function:

$$J = \sum_{i=1}^{k} \sum_{x \in C_i} \| x - \mu_i \|^2$$

where $C_i$is the set of points in cluster i and $\mu_i$is the centroid of cluster i. This represents the **within-cluster sum of squared distances**, often referred to as **inertia**.

k-Means converges to a **local minimum**, not necessarily the global optimum. For this reason, the algorithm is usually run multiple times with different starting positions.

---

3. Why Choosing *k* Is Crucial

The value of k fundamentally defines the structure of the solution. Poor selection leads to misleading or unusable results.

Under-Clustering (k too small)

- Distinct groups are merged.
- Important patterns are lost.
- High within-cluster variance.

- Poor representation of the data structure.

Over-Clustering (k too large)

- Natural groups are fractured.
- Noise and outliers may form their own clusters.
- Reduced interpretability.
- Risk of modelling artefacts rather than real structure.

Selecting k is therefore a **model-selection problem**, not merely a technical setting.

---

4. Dataset and Preprocessing

Dataset

This tutorial uses the **Mall Customer Segmentation Dataset**, which contains information about customers including:

- Age
- Annual income
- Spending score (1–100)

This dataset is well suited to clustering because:

- It has no labels (truly unsupervised),
- The variables are continuous,
- The context (customer behaviour) is intuitively interpretable.

Feature Selection

For visualisation and clarity, **Annual Income** and **Spending Score** are selected as the two main features for clustering.

Feature Scaling

k-Means relies on distance computations, so feature scaling is essential. Variables are standardised using **z-score normalisation**:

$$z = \frac{x - \mu}{\sigma}$$

If scaling is ignored, features with larger numeric ranges dominate the clustering process.

---

5. Experimental Method

To examine how k affects clustering behaviour, the following procedure is applied:

- The dataset is standardised using StandardScaler.
- k-Means++ initialisation is used to improve centroid placement.
- k is tested across the range **k = 2 to k = 10**.
- For each k:
    - **Inertia** is recorded.
    - **Silhouette score** is computed.
    - Cluster assignments are visualised.

Evaluation Metrics

Inertia

Inertia measures how compact the clusters are. Lower inertia indicates tighter clusters. However, inertia **always decreases** as k increases, so it cannot alone determine the best k.

Silhouette Score

The silhouette score measures how well each data point matches its own cluster compared to other clusters:

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

where:

- a = average intra-cluster distance
- b = average nearest-cluster distance

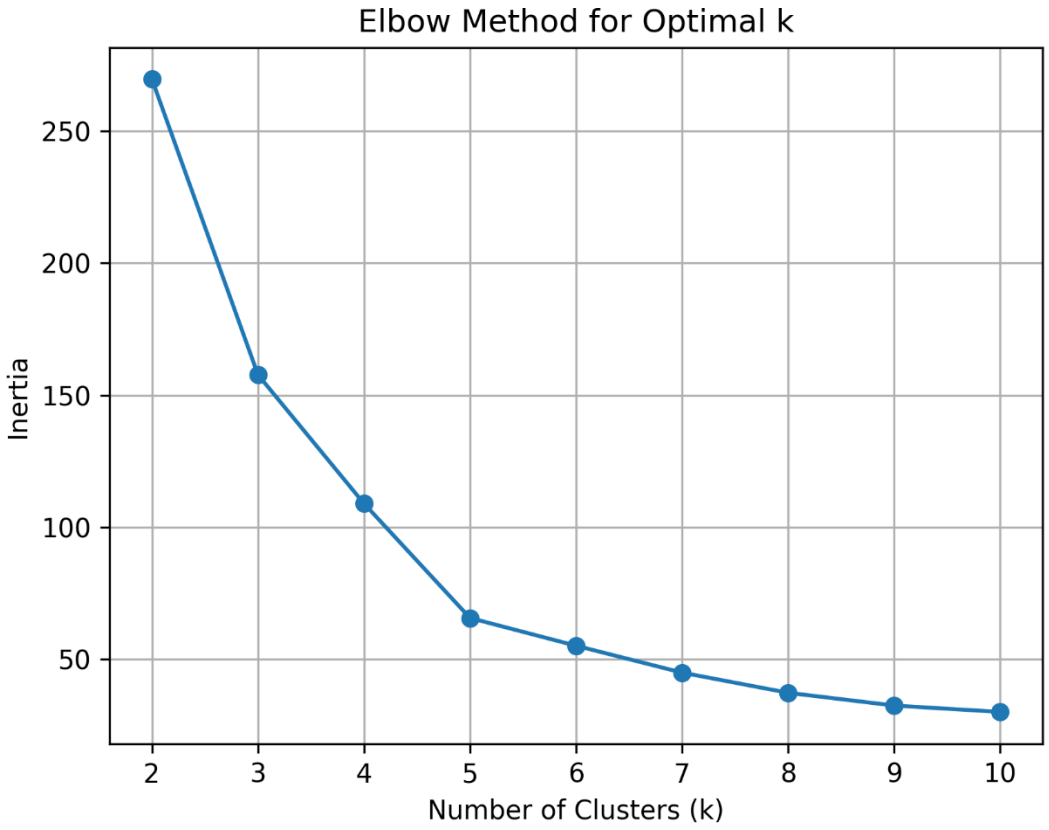
The silhouette score lies between –1 and 1. Higher values indicate better clustering.

---

6. Results and Visual Analysis

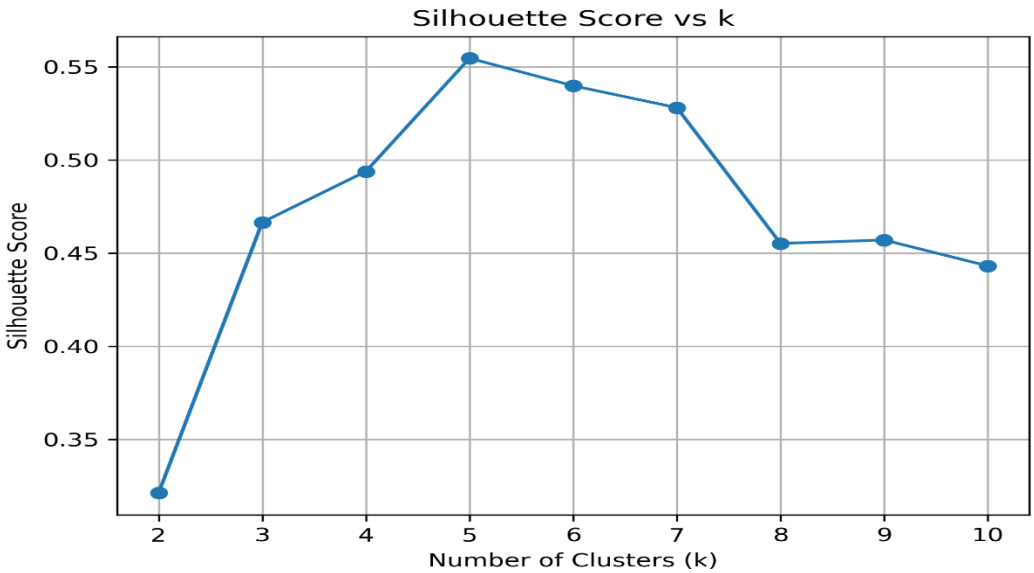
6.1 Elbow Method

The inertia curve shows a sharp decrease up to approximately **k = 5**, after which the rate of improvement slows significantly. This change in curvature is referred to as the **elbow point** and suggests that k ≈ 5 provides a good balance between model complexity and cluster compactness.



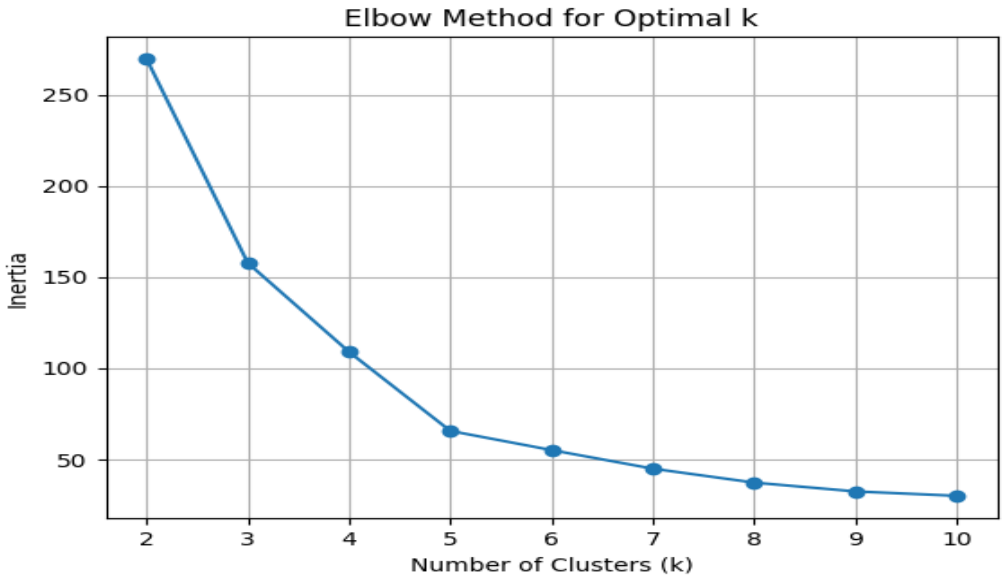
6.2 Silhouette Analysis

The silhouette scores peak around **k = 4 or k = 5**, indicating strong separation between clusters at those values. Very small k values yield lower scores due to forced merging of distinct groups. Larger k values cause silhouette scores to drop as clusters become fragmented.

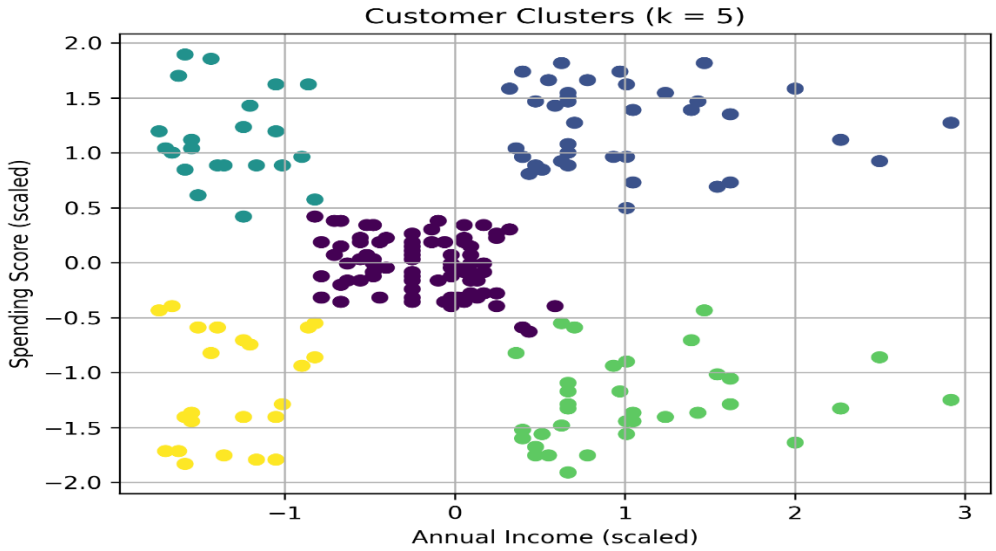


6.3 Cluster Visualisation

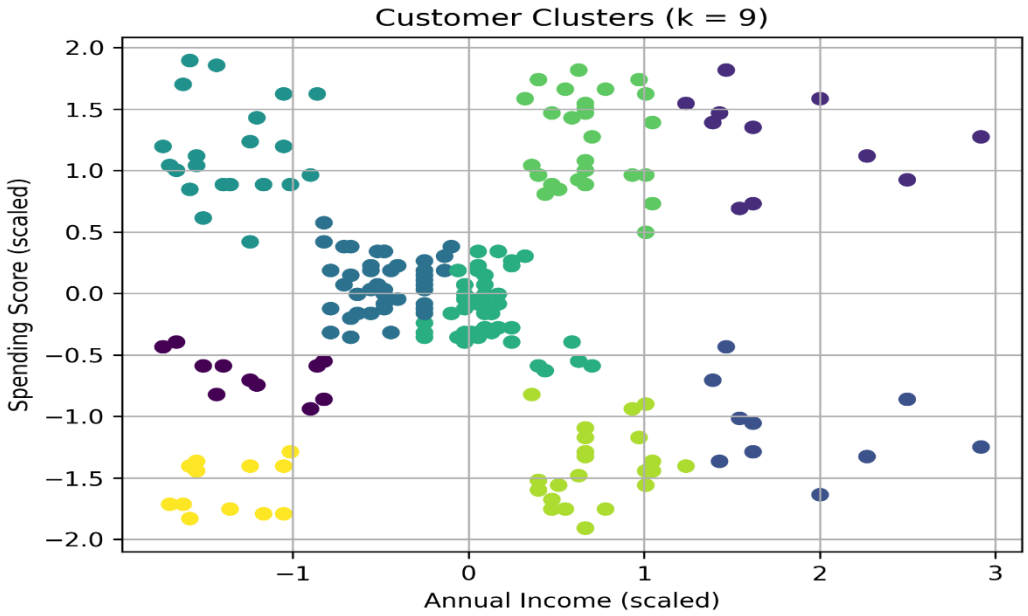
- **k = 2:**
  The dataset is divided into two very broad groups. High-spending and low-spending customers are mixed within clusters, demonstrating under-clustering.



- **k = 5 (optimal):**
  Clear, well-separated customer segments emerge. Groups correspond to intuitive behavioural patterns such as high-income/high-spending customers and low-income/low-spending customers.



- **k = 9:**
  Several clusters contain very few points. The model begins to capture noise rather than meaningful structure, showing the effect of over-clustering.



These visual results confirm the quantitative findings from the silhouette and elbow methods. **Silhouette Score (Random Init)**: 0.5546571631111091, **Silhouette Score (k-Means++ Init):** 0.5546571631111091

---

7. Discussion

Both evaluation methods indicate that **k ≈ 5** is an appropriate choice for this dataset. However, it is important to note that:

- There is rarely a single “correct” value of k.
- Optimal k depends on both statistical performance and **practical interpretability**.
- Domain knowledge should always complement numerical metrics.

k-Means also assumes spherical, equally sized clusters and performs poorly on non-convex cluster shapes. Furthermore, its sensitivity to outliers and initial centroid placement limits its robustness in complex real-world datasets.

---

8. Limitations of k-Means

- Requires the user to specify k in advance.
- Sensitive to feature scaling.
- Sensitive to outliers.
- Struggles with non-spherical and overlapping clusters.

- Converges to local minima.

These limitations motivate alternative clustering methods such as DBSCAN and Gaussian Mixture Models.

---

9. Conclusion

This tutorial demonstrated how the number of clusters, k, fundamentally affects the behaviour and quality of k-Means clustering. By systematically varying k and using both the elbow method and silhouette analysis, it was shown that:

- Too small k values lead to under-clustering,
- Too large k values lead to over-clustering,
- An intermediate value (k ≈ 5) provides the best balance for the customer dataset studied.

The key takeaway is that **k should never be chosen arbitrarily**. Instead, both quantitative validation techniques and domain-specific reasoning must be applied to select a value that is both statistically sound and practically meaningful.

---

10. Accessibility Considerations

This tutorial uses:

- Colour-blind friendly plotting schemes,
- Clearly labelled and captioned figures,
- Large readable fonts,
- Full textual explanations for all visuals.

These measures ensure that the material is accessible to users with visual impairments and compatible with screen readers.

---

11. References (Starter List – Will Be Fully Formatted in Step 4)

MacQueen, J. (1967). *Some methods for classification and analysis of multivariate observations*.
Jain, A. K. (2010). *Data clustering: 50 years beyond k-means*.
Pedregosa et al. (2011). *Scikit-learn: Machine Learning in Python*.
UCI Machine Learning Repository – Mall Customers Dataset.