Introduction to K-Means Clustering:

K-Means clustering is a popular unsupervised machine learning algorithm used for partitioning a dataset into groups or clusters based on similarity. The goal of K-Means is to divide a set of n data points into k non-overlapping subgroups (clusters) where each data point belongs to the cluster with the nearest mean.

Basic Idea Behind K-Means:

Initialization: Choose k initial centroids (points that represent the centre of each cluster) randomly from the data points.

Assignment: Assign each data point to the cluster whose centroid is the closest in terms of some distance metric, commonly the Euclidean distance.

Update Centroids: Recalculate the centroids of the clusters based on the current assignments.

Repeat: Repeat steps 2 and 3 until convergence, where convergence occurs when the centroids no longer change significantly, or a specified number of iterations is reached.

The algorithm aims to minimise the within-cluster sum of squares, effectively minimising the variance within each cluster. It converges to a solution, but the result may depend on the initial choice of centroids.

Significance and Applications:

- Customer Segmentation: Businesses use K-Means to group customers based on purchasing behaviour, allowing targeted marketing strategies and personalised services.

- Image Segmentation: In computer vision, K-Means is employed to segment images into distinct regions based on pixel intensity or colour similarity.

- Anomaly Detection: K-Means can identify outliers or anomalies in datasets by assigning them to clusters with dissimilar patterns.

- Document Clustering: In natural language processing, K-Means is used to cluster documents, enabling topic modelling and document organisation.

- Biology and Bioinformatics: K-Means is applied in the clustering of biological data, such as gene expression profiles, to discover patterns and relationships.

- Financial Fraud Detection: Identifying fraudulent activities in financial transactions by clustering normal and potentially fraudulent patterns.

- Network Security: Detecting unusual patterns in network traffic to identify potential security threats.
- Retail Inventory Management: K-Means can help optimise inventory by clustering products based on sales patterns.

- Healthcare: Grouping patients with similar medical profiles for personalised treatment plans or disease diagnosis.

- Social Network Analysis: Analysing social network data to identify communities or groups with shared interests.

A. Initialization of Centroids:

- Begin by choosing the number of clusters, k, that you want to identify in the dataset.
- Randomly select k data points from the dataset as initial centroids. These points will serve as the centre of each cluster.

B. Assignment of Data Points to Clusters:

- For each data point in the dataset, calculate its distance to each centroid. Commonly, Euclidean distance is used, but other distance metrics can also be employed.

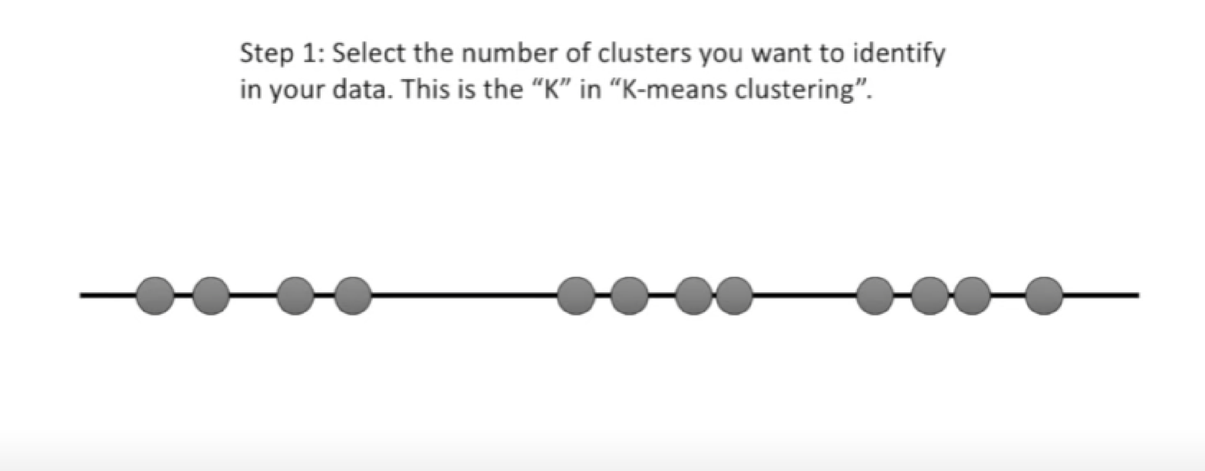
- Assign each data point to the cluster whose centroid is the closest.
- This step essentially groups data points based on their similarity to the centroids.

C. Recalculation of Centroids:

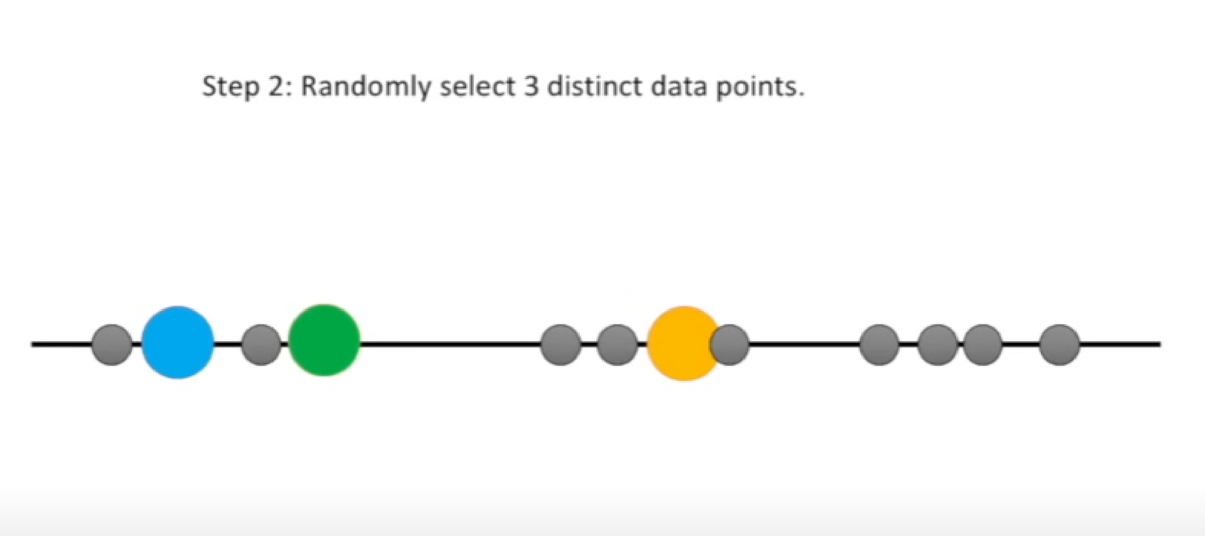
- After assigning all data points to clusters, recalculate the centroids of each cluster by computing the mean of all data points in that cluster.

D. Convergence Criteria:

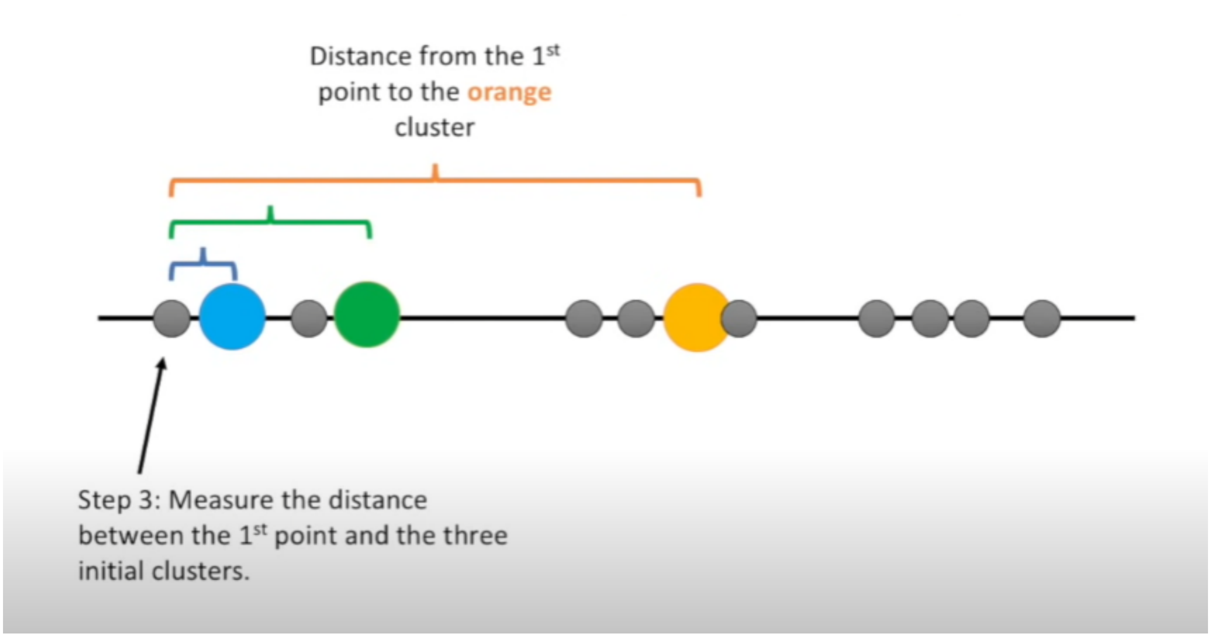
- Repeat the assignment and centroid recalculation steps until convergence is achieved. Convergence typically occurs when one of the following conditions is met:
    - Centroids no longer change significantly between iterations.
    - A fixed number of iterations is reached.
- The algorithm may also stop when a predefined threshold for improvement is met, or when the assignment of data points to clusters remains unchanged between iterations.
- It's important to note that K-Means can converge to a local minimum, meaning the final result depends on the initial choice of centroids. To mitigate this, the algorithm is often run multiple times with different initial centroids, and the best result is chosen based on the lowest sum of squared distances within clusters.



Step 1: Select the number of clusters you want to identify in your data. This is the “K” in “K-means clustering”.

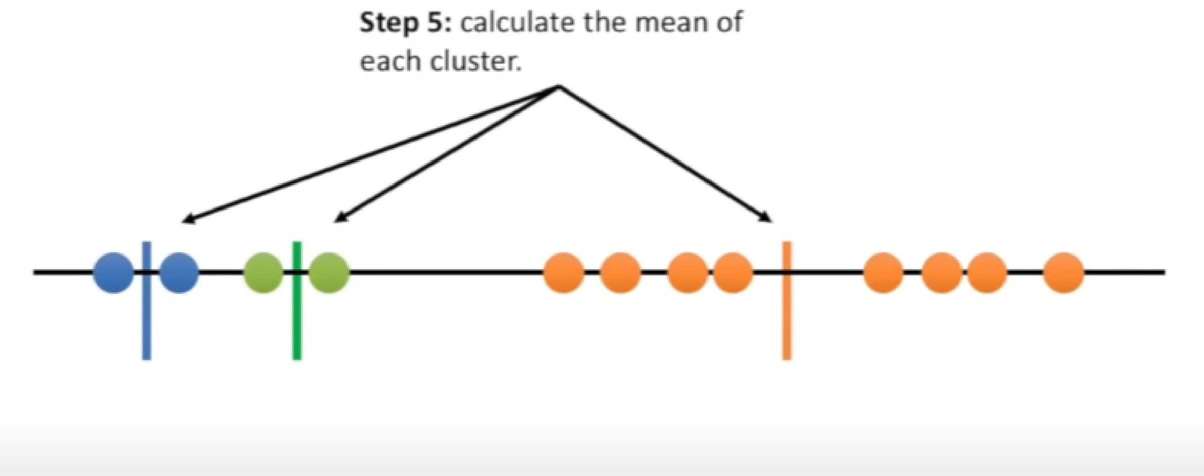


Step 2: Randomly select 3 distinct data points.

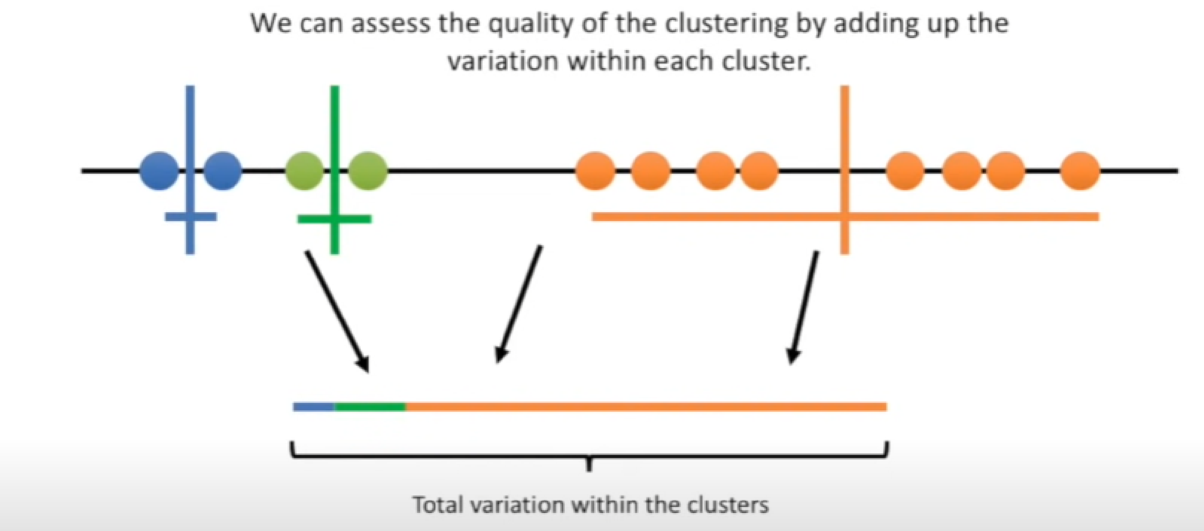


Distance from the 1st point to the orange cluster

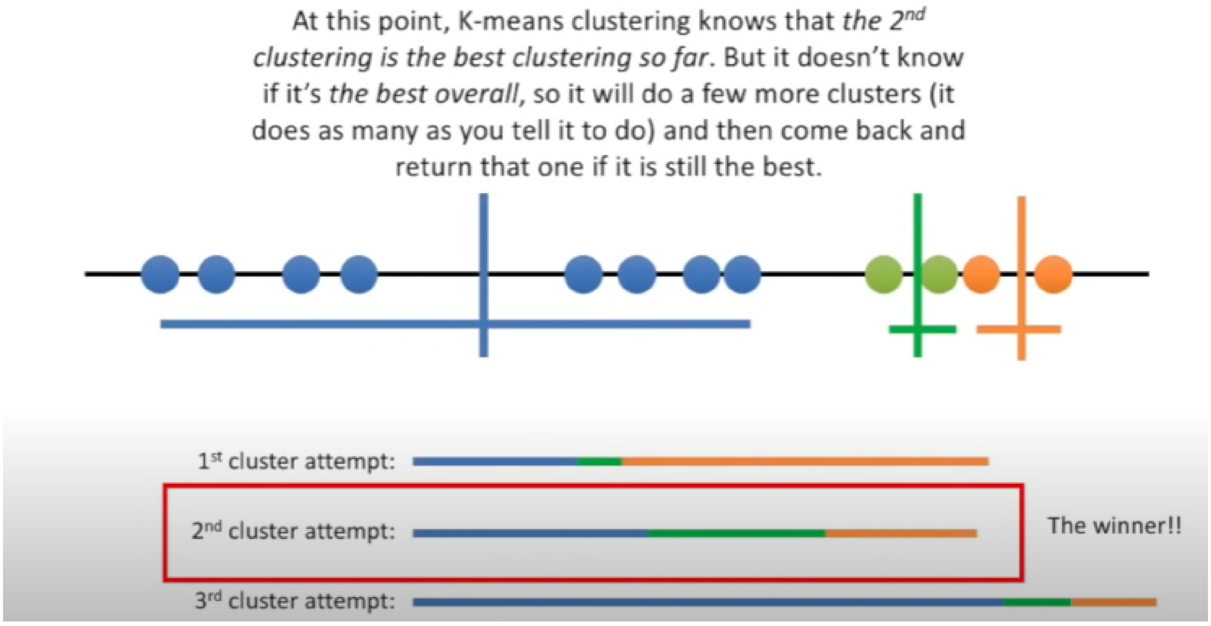
Step 3: Measure the distance between the 1st point and the three initial clusters.



Step 5: calculate the mean of each cluster.



We can assess the quality of the clustering by adding up the variation within each cluster.



At this point, K-means clustering knows that *the 2nd clustering is the best clustering so far*. But it doesn't know if it's *the best overall*, so it will do a few more clusters (it does as many as you tell it to do) and then come back and return that one if it is still the best.