JANUARY
FEBRUARY
MARCH
APRIL
MAY
JUN
JULY
AUGUST
SEPTEMBER
INR
DN
SP
NB
D
N
397.66
519.79
604.88
113.92
91.7
221.49
211.27
166.13
177.72
125.91
89.93
41.65
69.77
2,472.26
24.74
32.36
103.95
100.98
99.99
84.81
109
99.47
5,874.00
129.9
134.69
136.24
131.82
135.98
8,019.79
83.48
56.29
92.37
68.02
107.21
78.21
58.34
197.41
69.72
4,106.49
108
187
166
213
12,674.40
128
140
158.00
9,401.00
4,040.81
932.77
413.06
420.23
24,944.01
144.12
241.68
179.77
247.49
301.21
17,879.22
139.72
151.37
175.88
158.17
155.21
181.75
10,730.91
101.18
3,927.28
123.02
183.97
103.66
92.91
145.97
63.51
59.93
52.5
52.04
3,097.61
127.33
105.1
57
113.25
136.65
68.53
5749.89
2,968.46
95.78
528.06
719.49
604.88
550.24

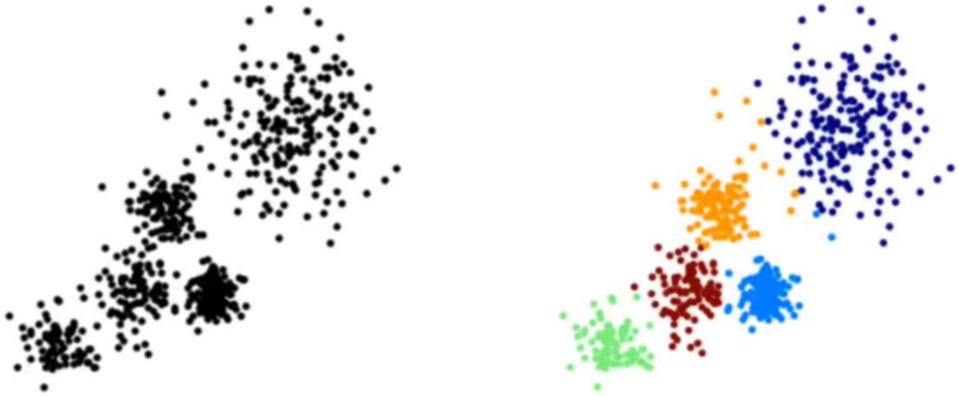
K-Means

K – Means Clustering

- K Means Clustering is an unsupervised learning algorithm that will attempt to group similar clusters together in the data
- What does a typical clustering problem look like?
  - Cluster similar documents
  - Cluster customers based on features
  - Market Segmentation
  - Identify similar physical groups

K – Means Clustering

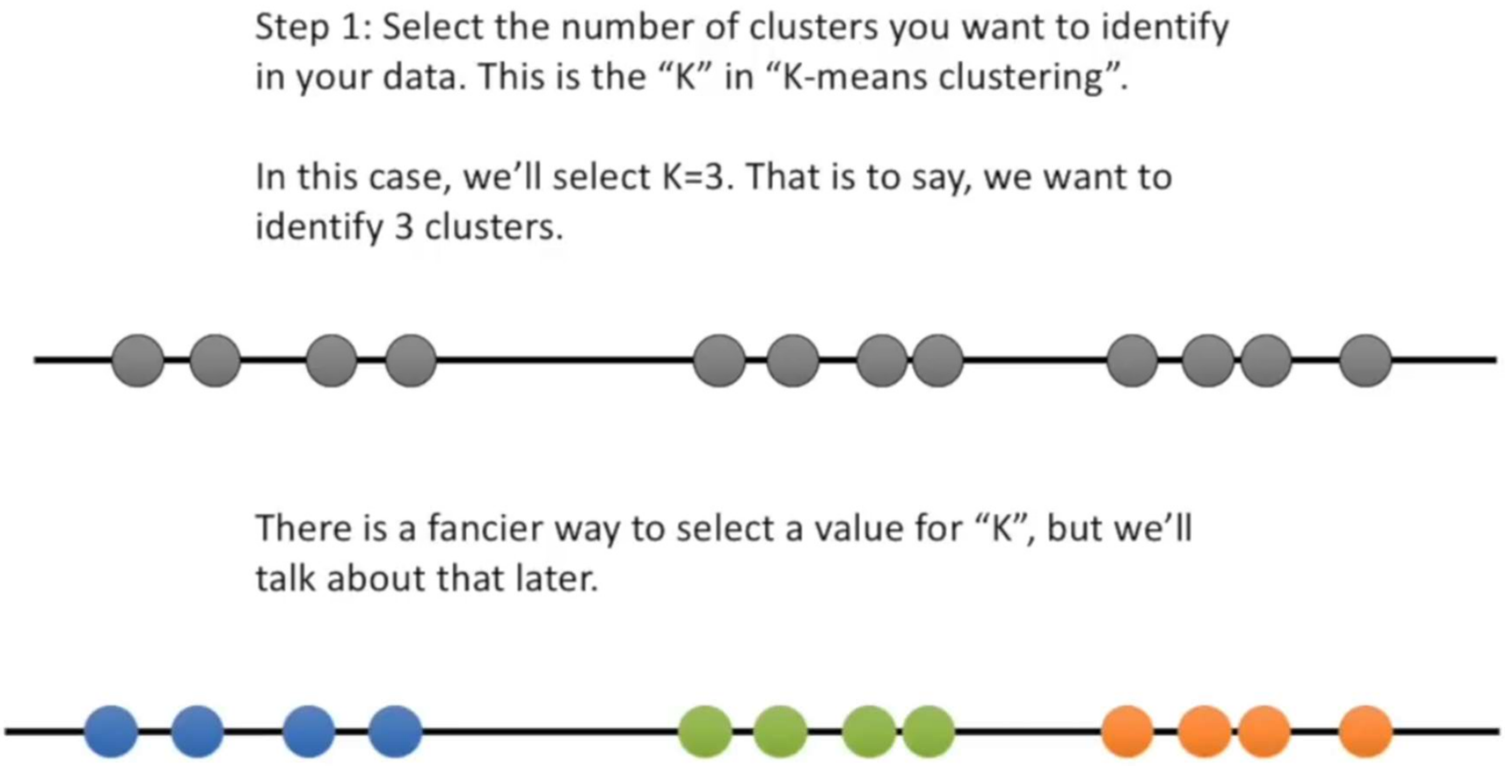
- The overall goal is to divide data into distinct groups such that observations within each group are similar



K –Means Clustering

- The K – Means Algorithm
    - Choose a number of clusters “k”
    - Randomly assign each point to a cluster
    - Until clusters stop changing, repeat the following:
        - For each cluster, compute the cluster centroid by taking the mean vector of the points in the cluster
        - Assign each data point to the cluster for which the centroid is the closest

K - Means



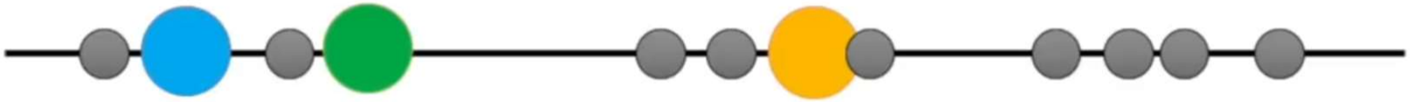
Step 1: Select the number of clusters you want to identify in your data. This is the “K” in “K-means clustering”.

In this case, we’ll select K=3. That is to say, we want to identify 3 clusters.

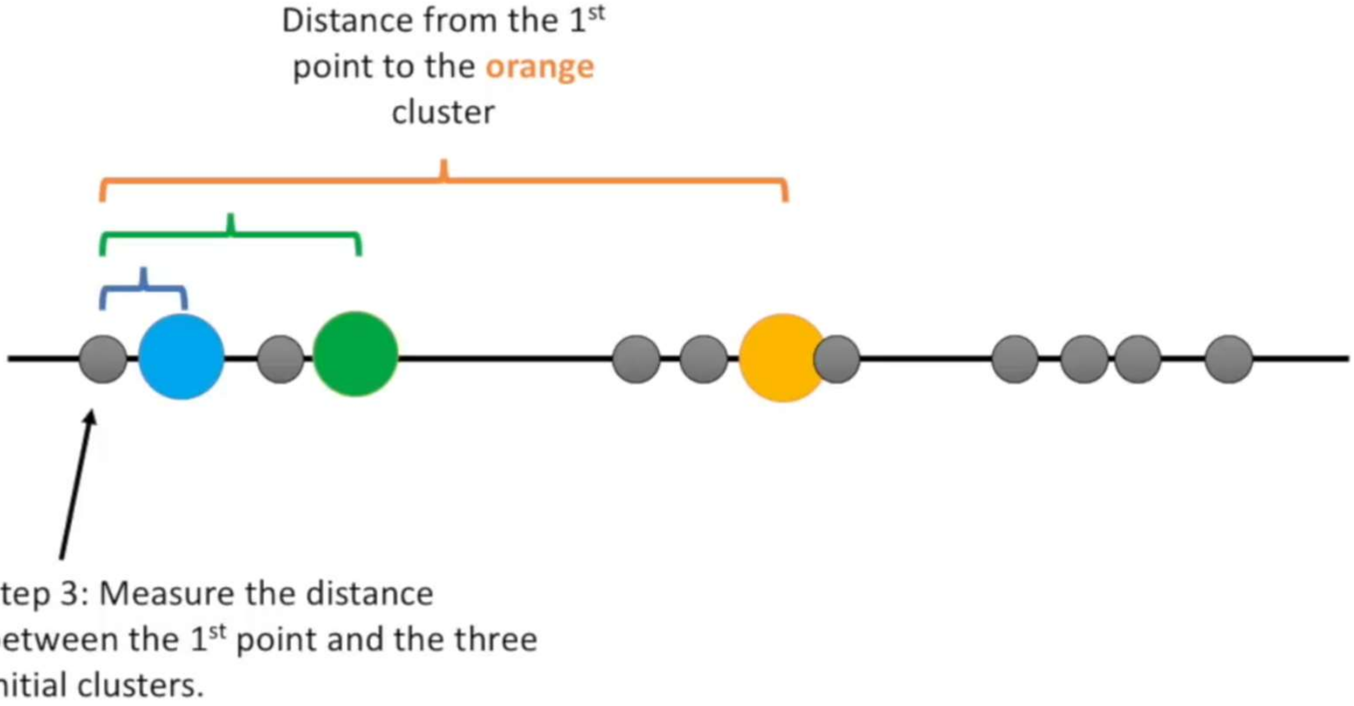
There is a fancier way to select a value for “K”, but we’ll talk about that later.

K - Means

Step 2: Randomly select 3 distinct data points.



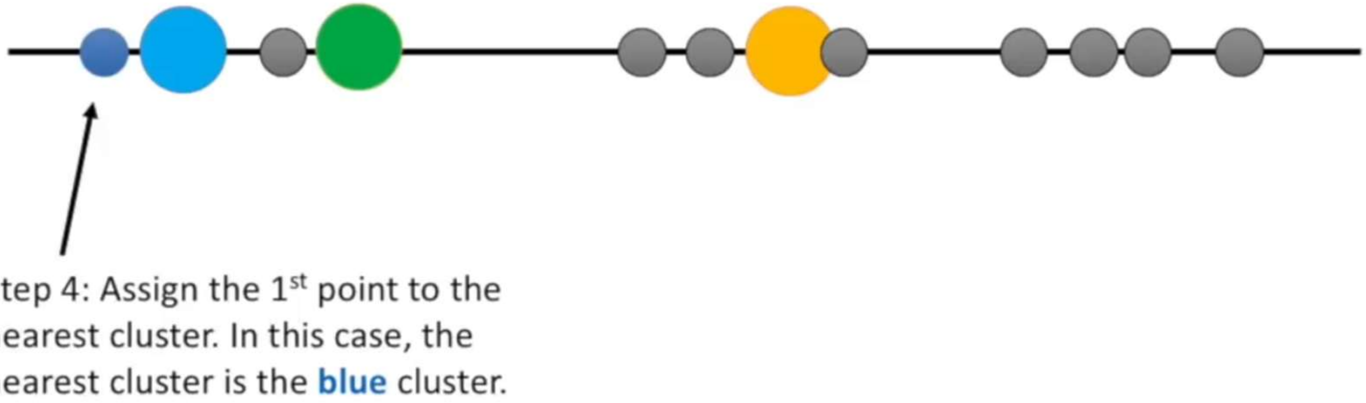
K - Means



Distance from the 1st point to the orange cluster

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

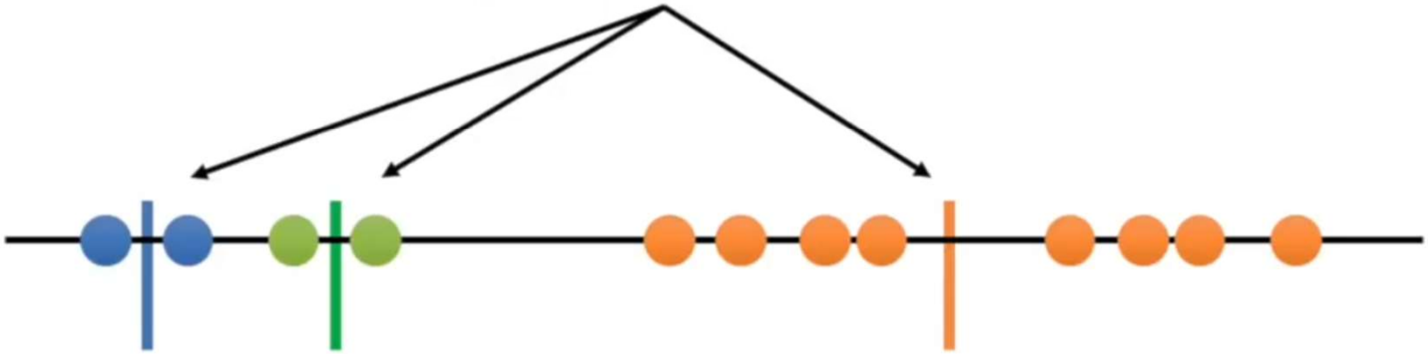
K - Means



Step 4: Assign the 1st point to the nearest cluster. In this case, the nearest cluster is the **blue** cluster.

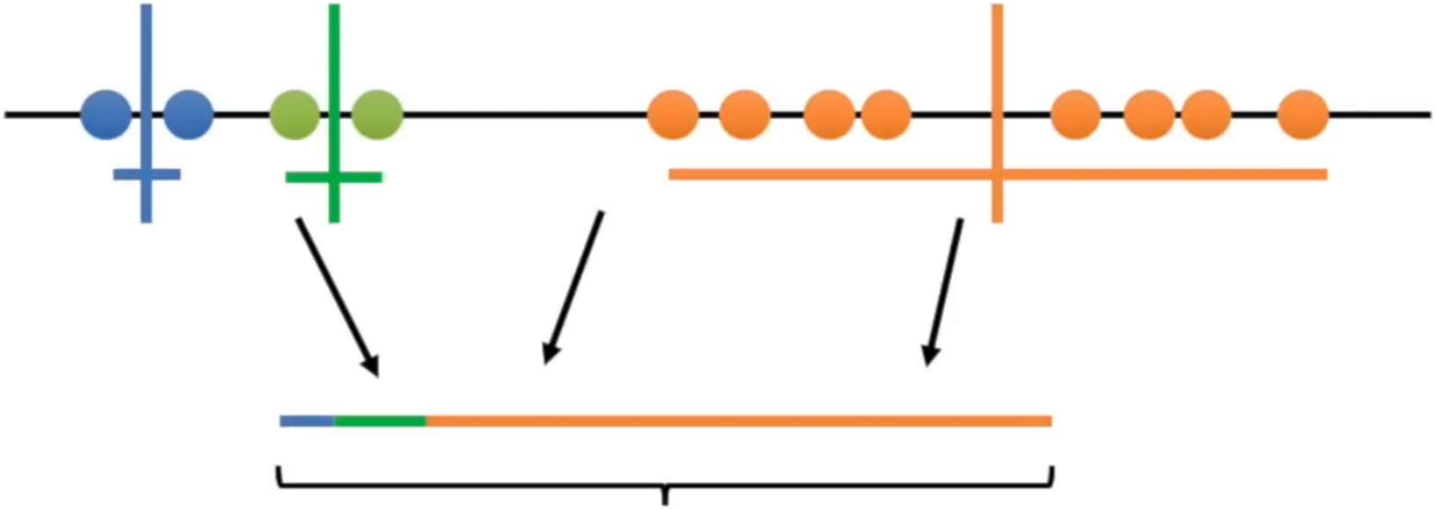
Repeat the steps for the next point

Step 5: calculate the mean of each cluster.



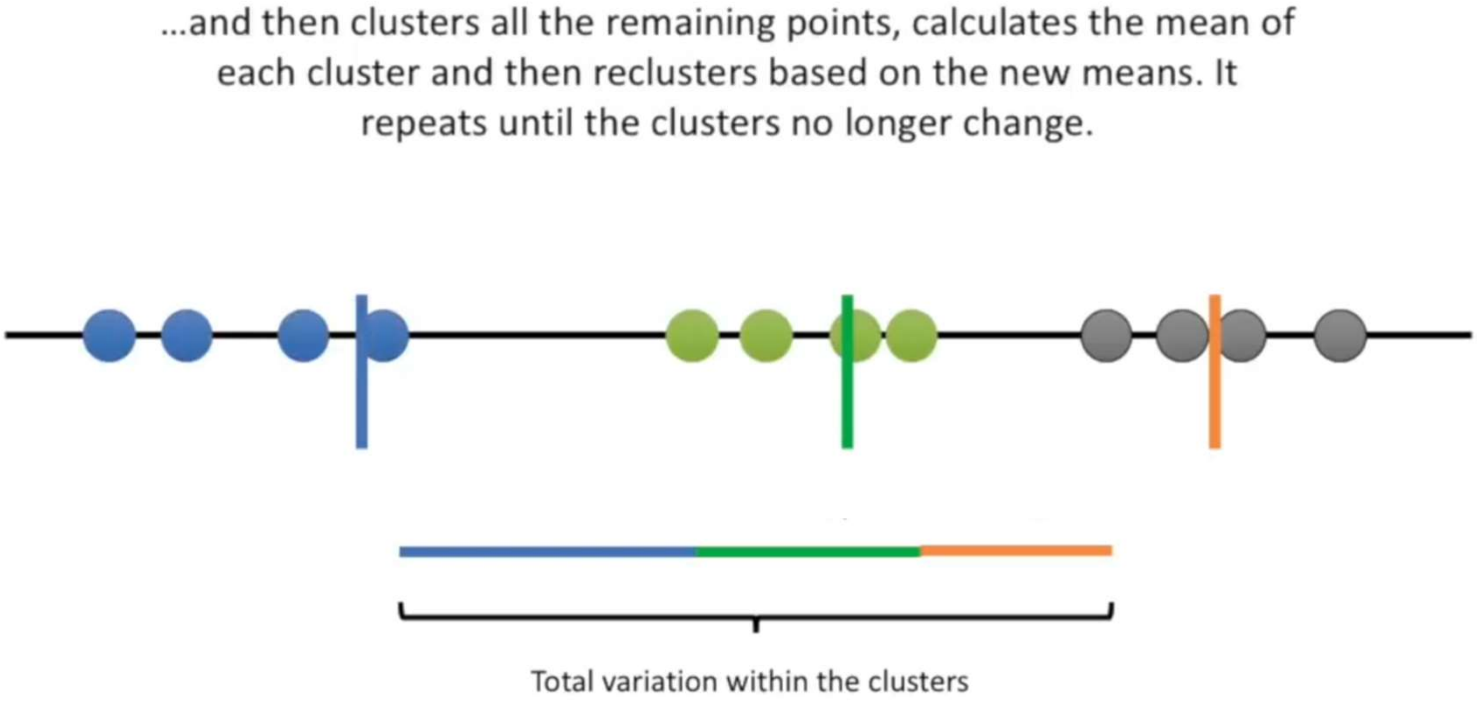
K - Means

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



Total variation within the clusters

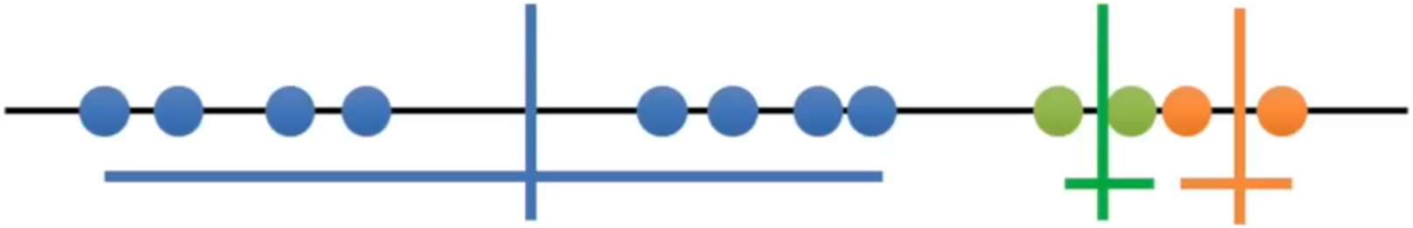
K - Means

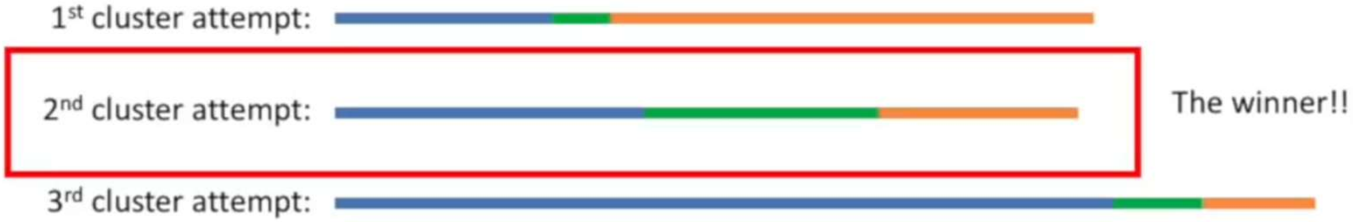


...and then clusters all the remaining points, calculates the mean of each cluster and then reclusters based on the new means. It repeats until the clusters no longer change.

K - Means

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.





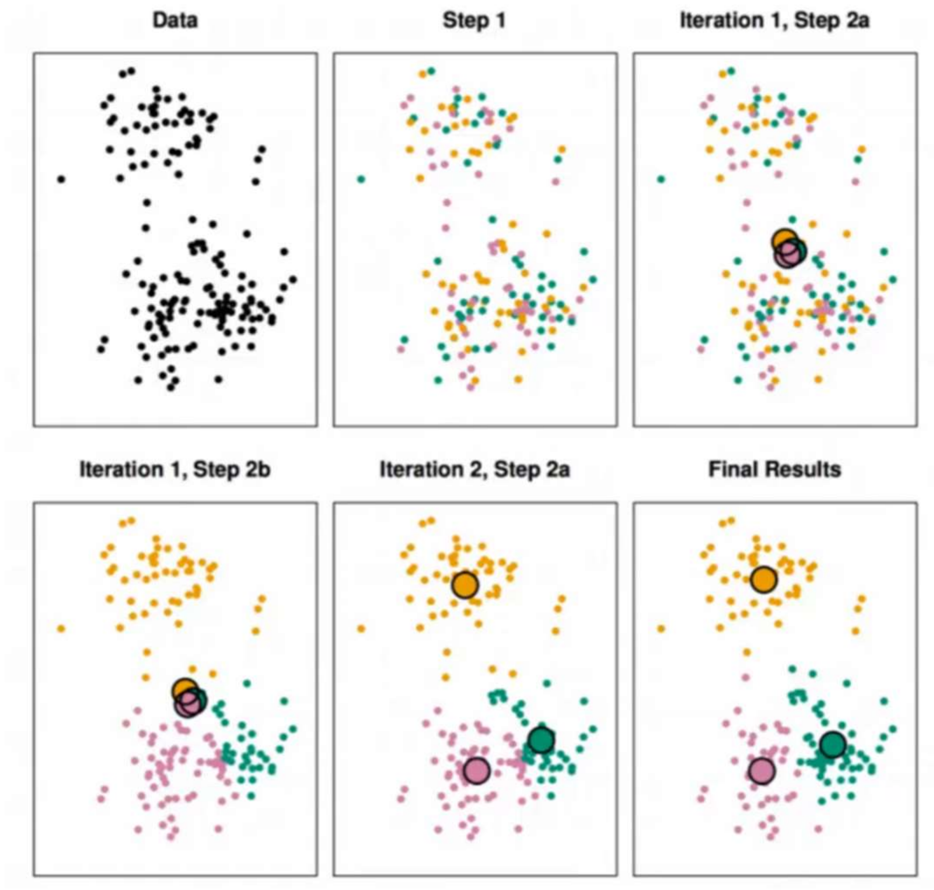
1st cluster attempt:

2nd cluster attempt:

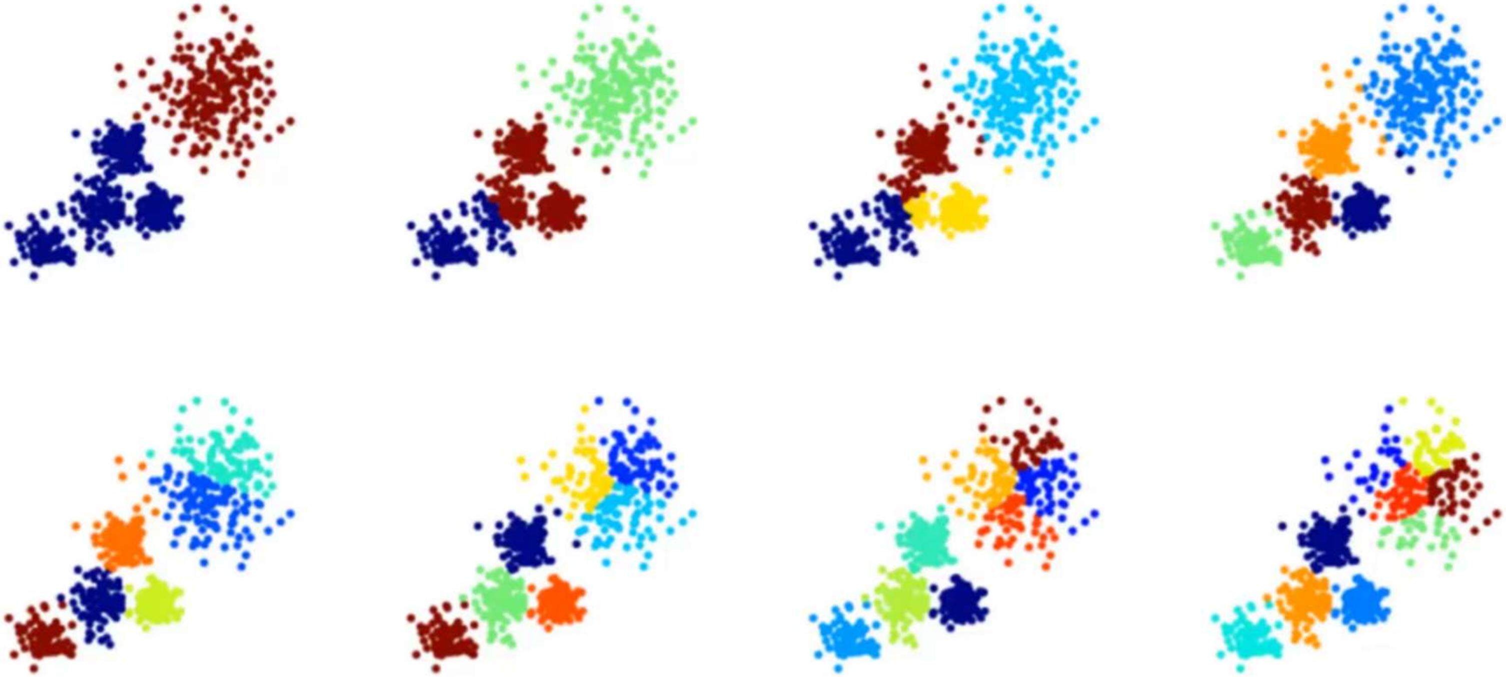
3rd cluster attempt:

The winner!!

K – Means Clustering



Choosing a K Value

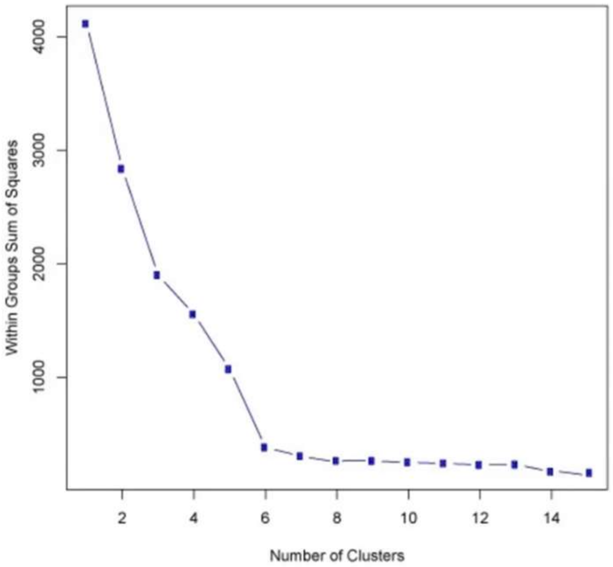


Choosing a K Value

- There is no easy answer for choosing the best K value
- One way is called as the Elbow Method
- First, compute the sum of squared error (SSE) for some values of K (for example: 2, 4, 6, 8…)
- The SSE is defined as the SS distance between each member of the cluster and its centroid
- If you plot k against SSE, you will see error decreases as k gets larger, this is because when the number of cluster increases, they should be smaller, so distortion is also smaller

Choosing K Value

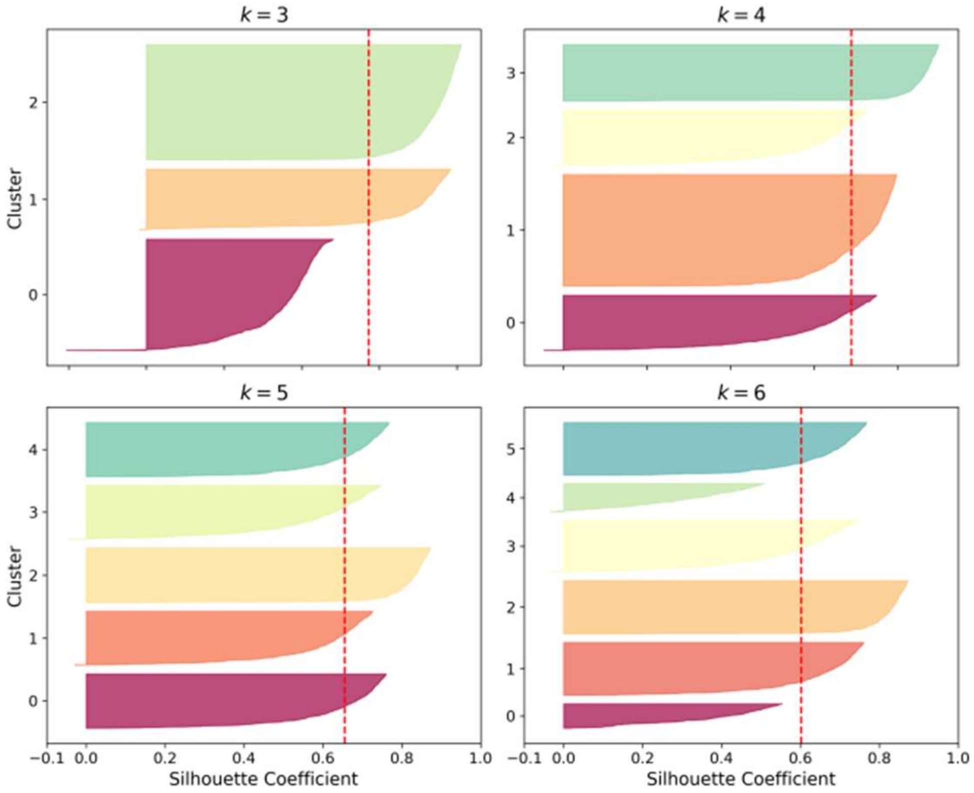
- The idea of the elbow method is to choose the k at which the SSE decreases abruptly
- This produces an elbow effect as shown in the graph:



Silouette score

- Silouette score, which is the mean silhouette coefficient over all the instances.
- An instance's silhouette coefficient is equal to (b – a) / max(a, b) where a is the mean distance to the other instances in the same cluster (it is the mean intra-cluster distance), and b is the mean nearest-cluster distance, that is the mean distance to the instances of the next closest cluster (defined as the one that minimizes b, excluding the instance's own cluster).
- The silhouette coefficient can vary between -1 and +1: a coefficient close to +1 means that the instance is well inside its own cluster and far from other clusters, while a coefficient close to 0 means that it is close to a cluster boundary, and finally a coefficient close to -1 means that the instance may have been assigned to the wrong cluster.

Silhouette Score



We can see that when k=3 and when k=6, we get bad clusters. But when k=4 or k=5, the clusters look pretty good – most instances extend beyond the dashed line, to the right and closer to 1.0.