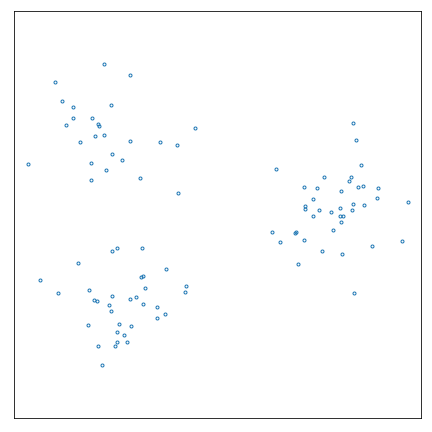
**Solutions for warm-up activity**

1. K-means algorithm
2. A 100-point data set with 3 distinct clusters that can be separated:

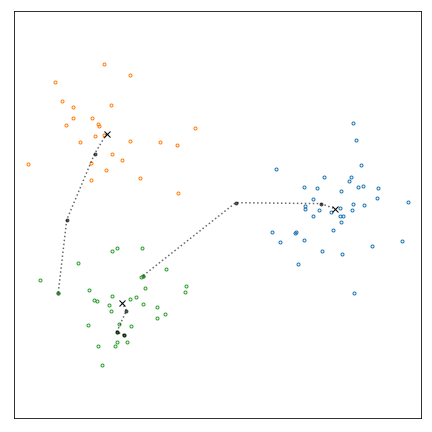


1. Using randomly selected centroids, they were unfortunately all drawn from the same cluster:

A screenshot of a cell phone

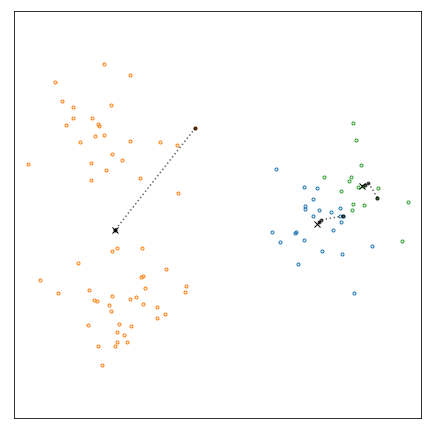
Description automatically generated

1. As can be seen, randomly selected centroids will not necessarily be distributed evenly among the clusters. This might result in more iterations necessary for convergence and perhaps “incorrect” assignment of the clusters as the algorithm finds a local minimum.
2. The initialization step could be improved by allowing the data scientist to select starting centroid locations using informed and educated guesses. This is intuitive if the data set can be inspected visually but requires a more in-depth understanding of the data if the number of dimensions precludes visualization. Alternatively, the dimensionality of the data could be reduced by using a technique like SVD, which could then allow for visualization.
3. For a random centroid selection, the path followed by the cluster centroids and the final cluster assignments (shown by the yellow, green, and blue points) is shown below:



As can be seen, the “correct” clusters were found despite the unfortunate initial centroid configuration. Using the same initial configuration, the centroids will always follow the same path and the final clusters will always be the same. Thus, the algorithm is deterministic once the initial centroids are selected.

1. A different initial configuration of the centroids led to a different and “incorrect” assignment of the final clusters:



In order to avoid local minima, the algorithm could be run using different starting configurations for the centroids and selecting the outcome that produced the minimum coherence. Of course, this will increase the computational time and still not guarantee achieving the global optimum.

1. Mean-shift algorithm
2. The KDE contours for bandwidth = 0.5. More peaks will lead to more clusters.

A close up of text on a white background

Description automatically generated

The KDE contours for bandwidth = 5.0. Fewer peaks will lead to fewer clusters.

A close up of text on a white background

Description automatically generated

1. The lowest bandwidth the mean-shift algorithm converged to 3 clusters for this data set was 1.8.

A close up of a logo

Description automatically generated

The largest bandwidth the mean-shift algorithm converged to 3 clusters for this data set was 3.9.

A close up of text on a white background

Description automatically generated

1. DBSCAN algorithm

A screenshot of a cell phone

Description automatically generated

1. The DBSCAN algorithm “correctly” found the clusters
2. The points not assigned to a cluster are labeled as noise.
3. With , the number of clusters increased, and the points labeled as noise decreased.

A screenshot of a cell phone

Description automatically generated

With , the number of clusters decreased, and the points labeled as noise increased substantially. A whole cluster was re-labeled as noise.

A screenshot of a cell phone

Description automatically generated

1. With , the number of clusters increased, and the points labeled as noise also increased.

A screenshot of a cell phone

Description automatically generated

With , the number of clusters decreased, and the points labeled as noise also decreased.

A screenshot of a cell phone

Description automatically generated

1. With any clustering algorithm, care is required in parameter selection. If inappropriate values of and are selected for the data set to be analyzed, DBSCAN can give poor results.