

K-means & K-medoids

DRIPTA MJ

Department of Mathematics

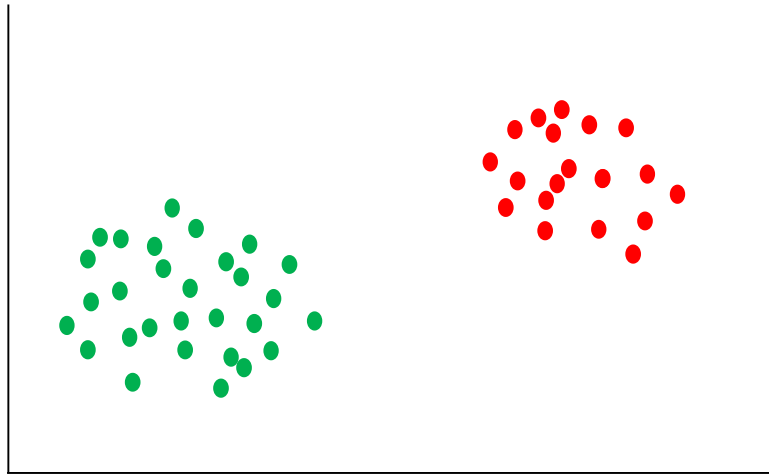
RAMAKRISHNA MISSION VIVEKANANDA EDUCATIONAL AND RESEARCH INSTITUTE
BELUR MATH, INDIA

Machine Learning
CS230

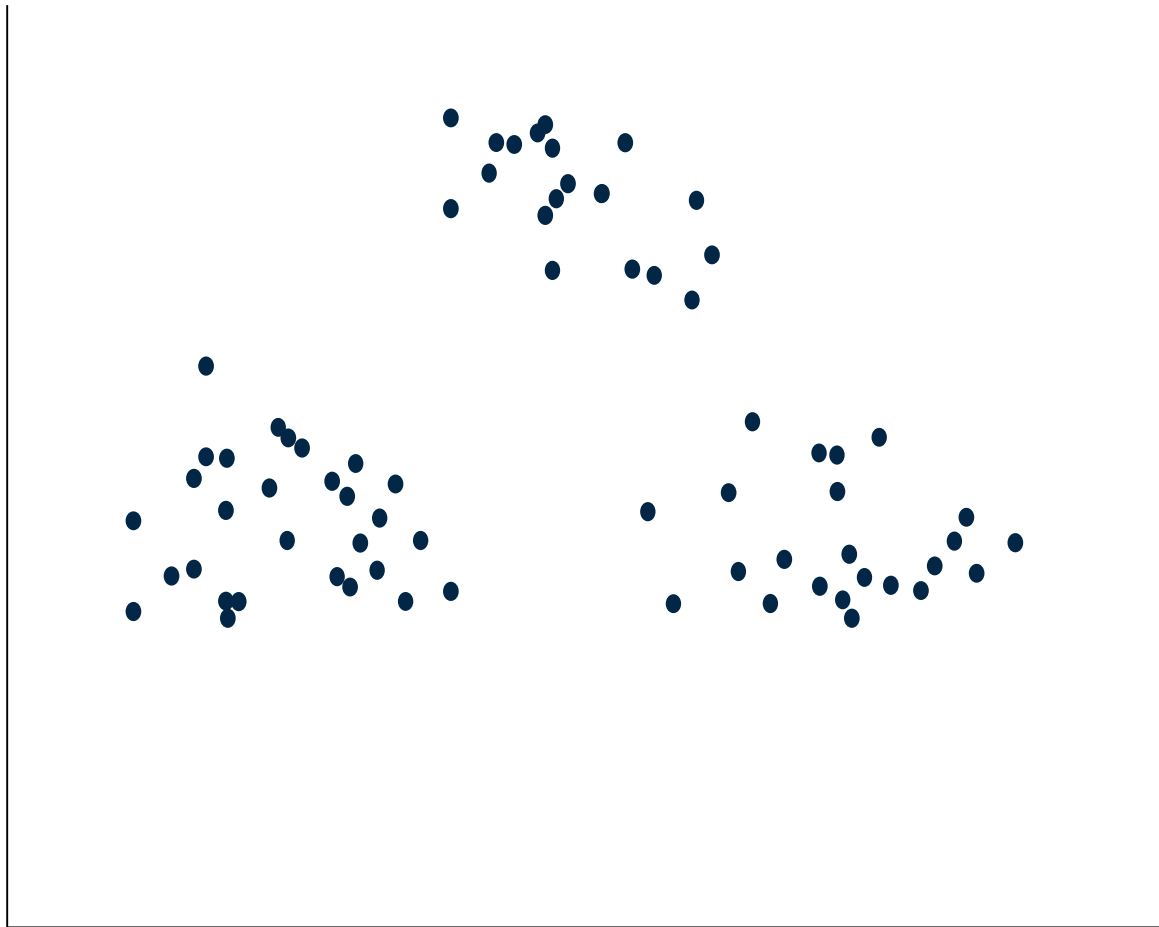
Sem 3, 2018-19

Clustering

- Unsupervised learning: Learn structures in the data as defined by the model.
- Unlabelled data is organized into groups called clusters.
- A cluster contains data items which are “similar”. These data items are dissimilar to data items in other clusters.



Unlabelled data



Notations

- Training dataset comprise N data points:

$$\mathbf{X} = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(N)}\}$$

- Partition data into K clusters:

$$\{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_K\}$$

- Cluster prediction for the n th data point:
 - Label encoding: $z^{(n)} \in \{1, 2, \dots, K\}$.
 - One-hot encoding: $\mathbf{z}^{(n)}$ is a K -dimensional vector with

$$\mathbf{z}_k^{(n)} = \begin{cases} 1 & \text{if } \mathbf{x}^{(n)} \text{ is assigned } \mathcal{C}_k \\ 0 & \text{otherwise} \end{cases}$$

Procedure

- Step 1: Initialize (randomly) the centroids (means) of the K clusters:

$$\{\mu_1, \mu_2, \dots, \mu_K\}.$$

- Step 2: For $n = 1, 2, \dots, N$:

- Compute the distance of the n th data point to all the K centroids, and assign $\mathbf{x}^{(n)}$ to the cluster to which it is the closest:

$$z^{(n)} = \arg \min_k \|\mathbf{x}^{(n)} - \mu_k\|^2$$

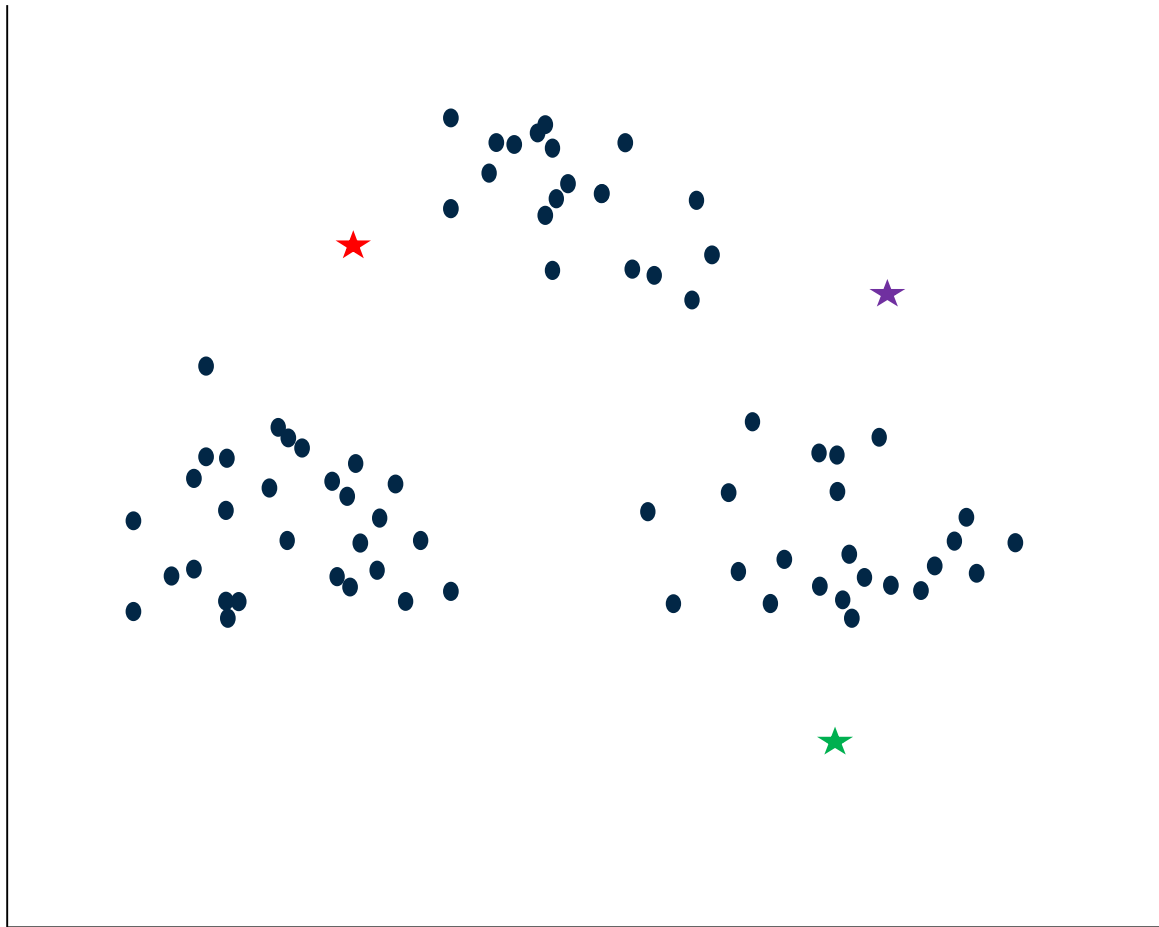
- Step 3: Recompute the cluster centroids with the most recently assigned memberships:

$$\mu_k = \frac{1}{n_k} \sum_{j: \mathbf{x}^{(j)} \in \mathcal{C}_k} \mathbf{x}^{(j)}$$

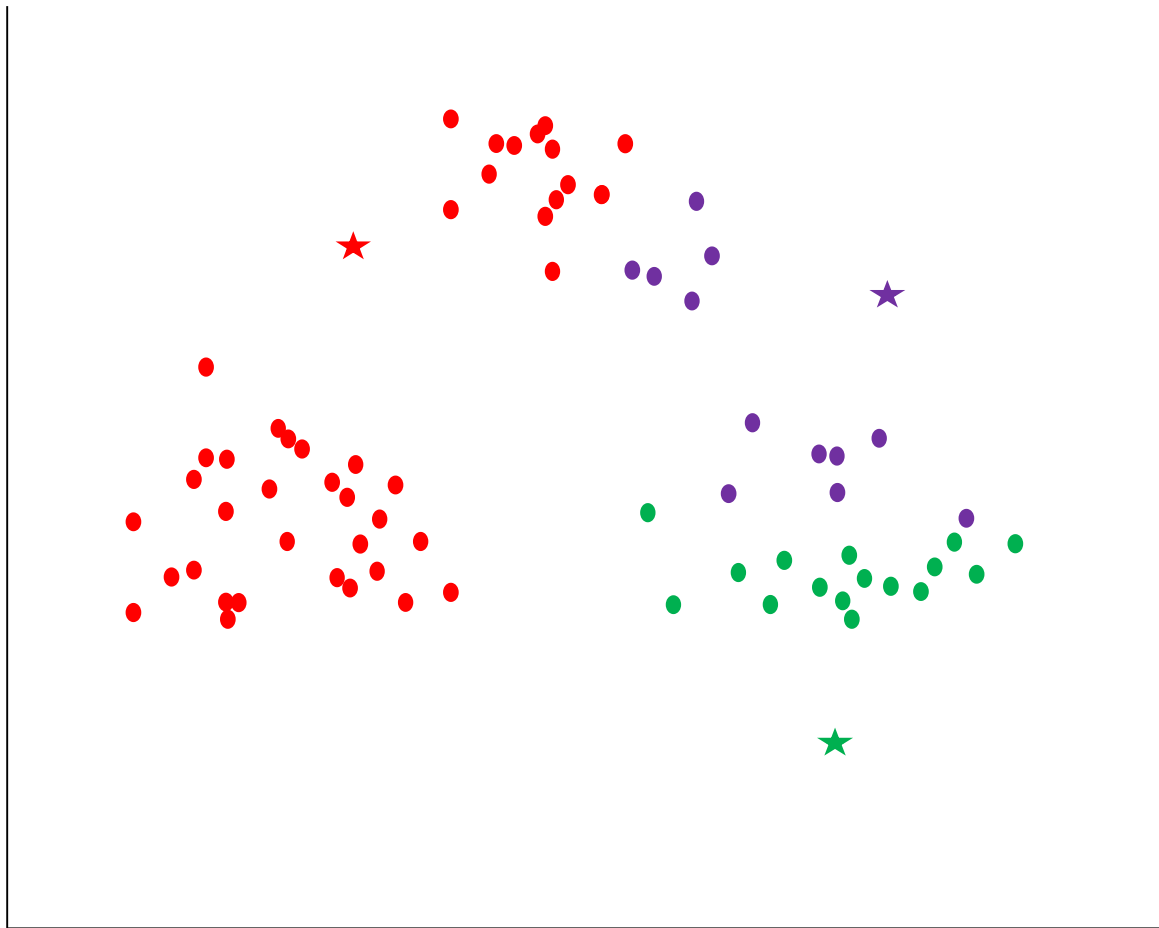
where n_k is the number of points in cluster \mathcal{C}_k .

- If none of the cluster assignments have changed, **STOP**. Else, **REPEAT** from Step 2.

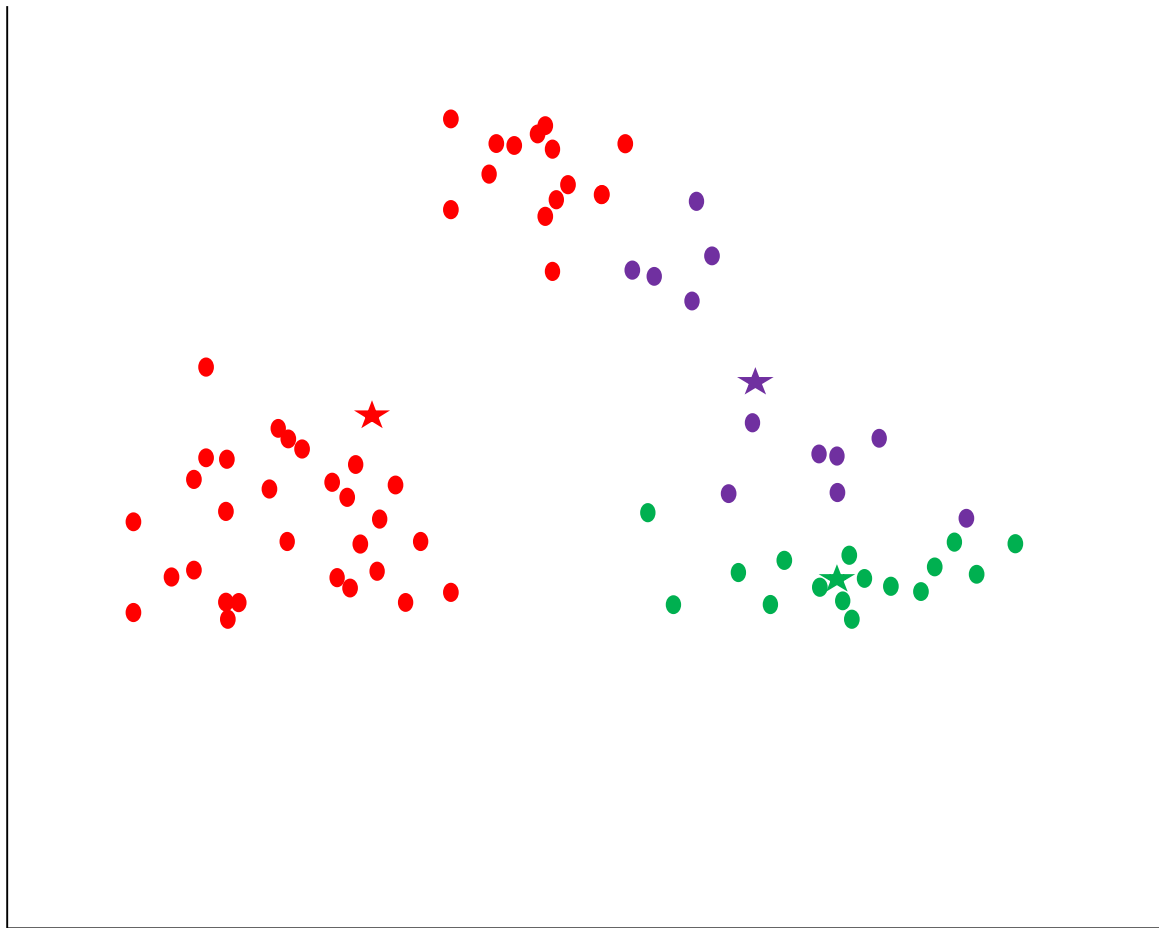
K-means



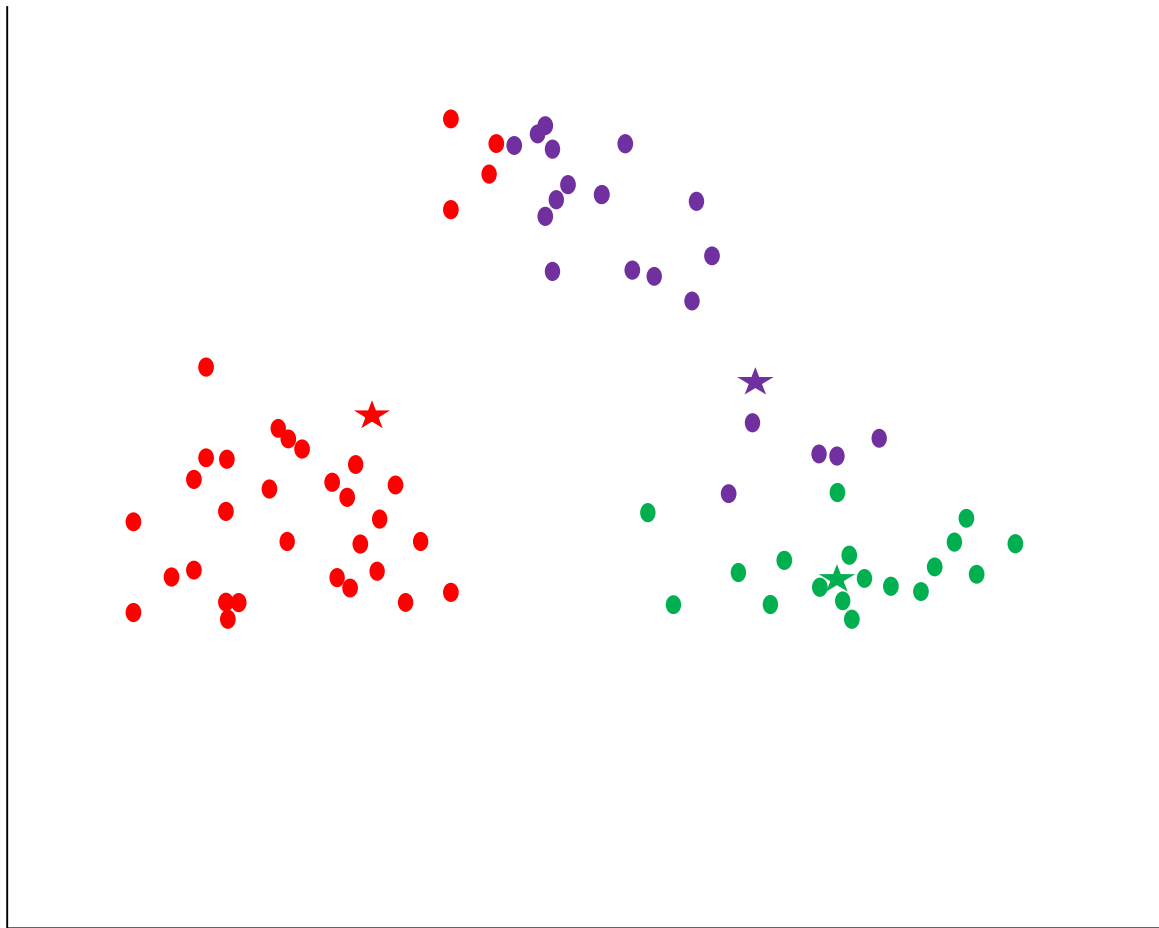
K-means



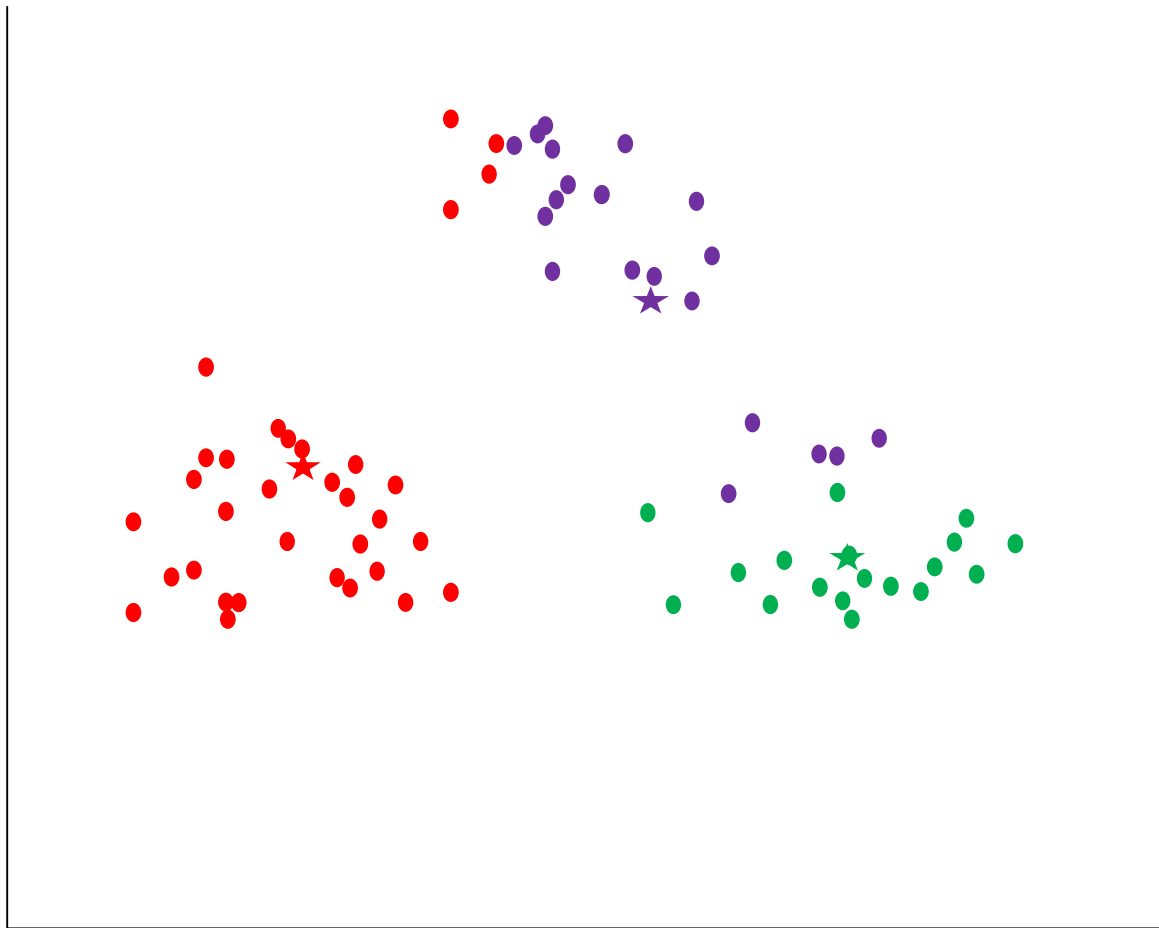
K-means



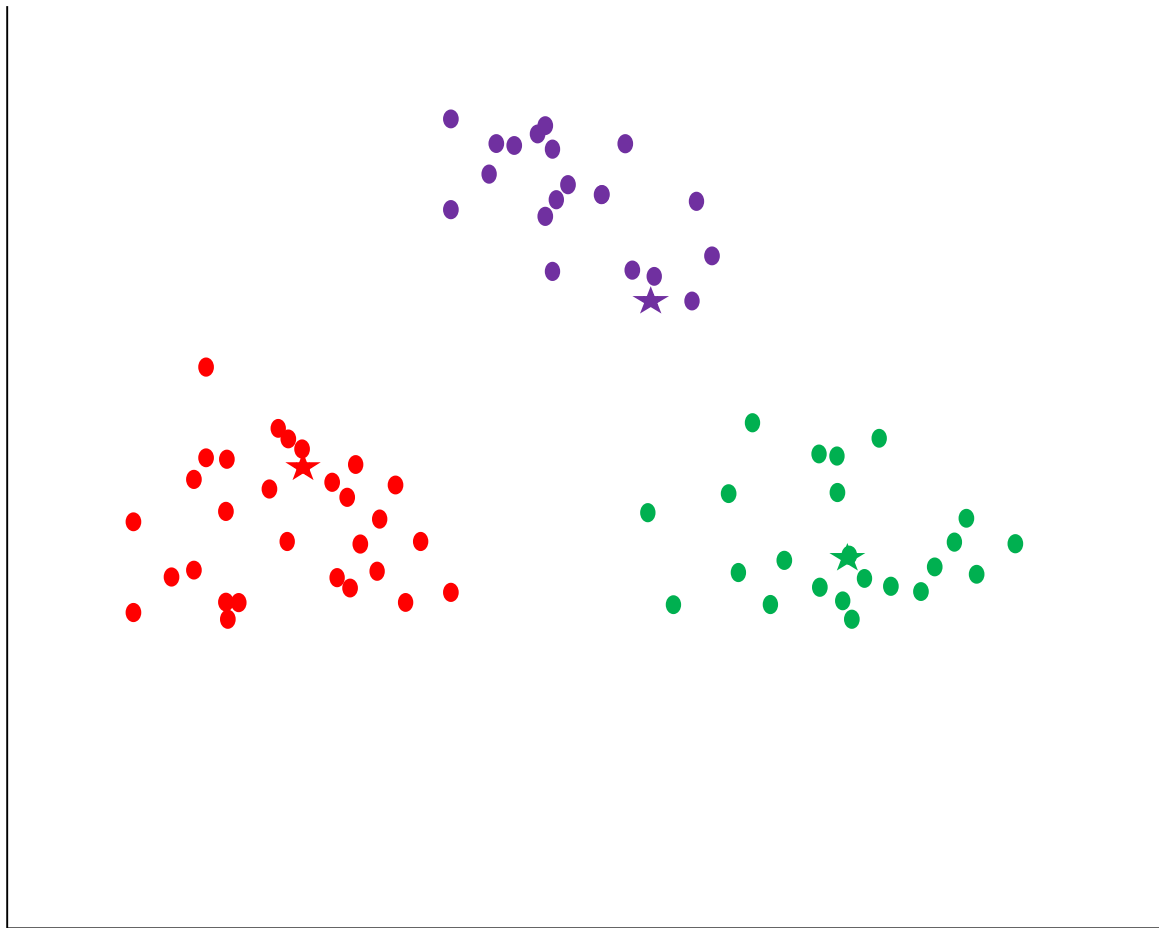
K-means



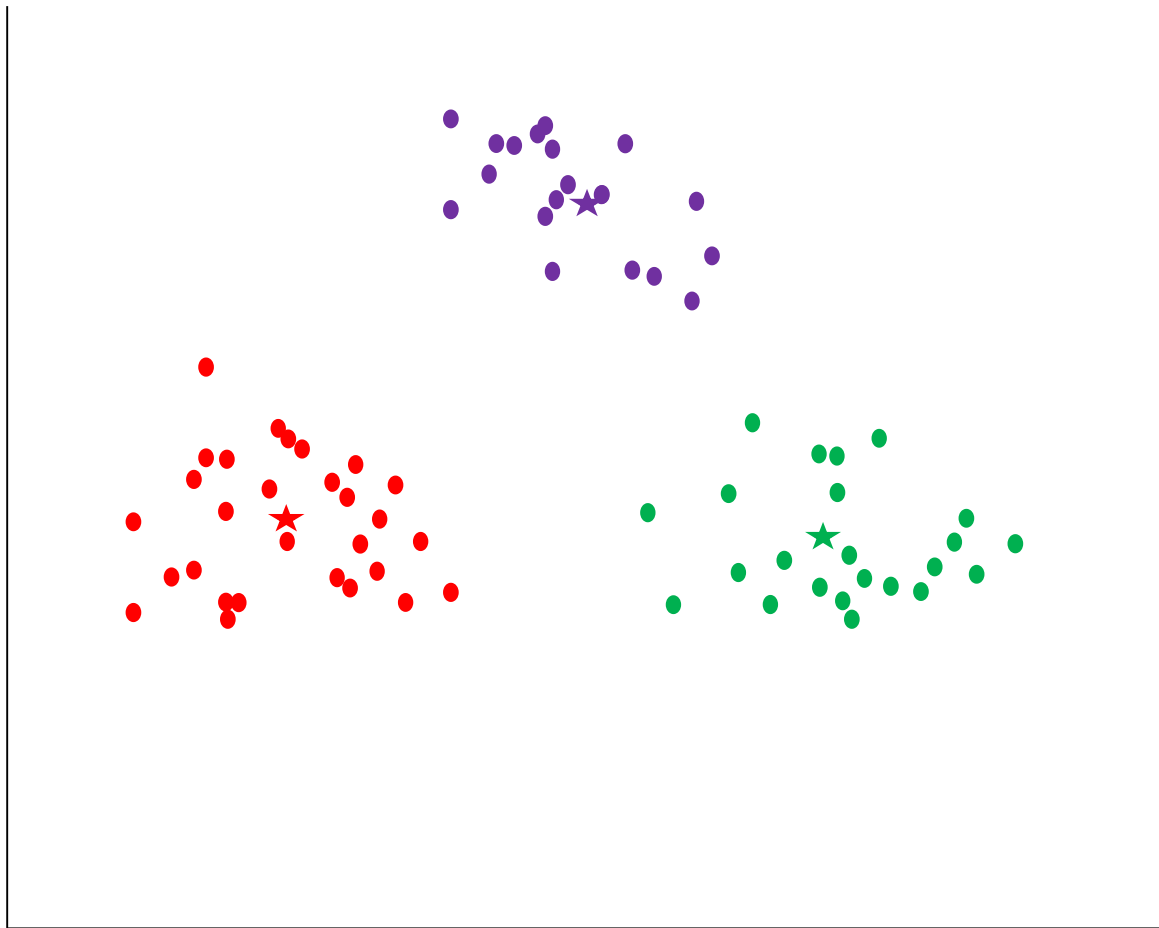
K-means



K-means



K-means



Matrix notation

- Inputs:

$$\mathbf{X} = \begin{bmatrix} (\mathbf{x}^{(1)})^T \\ (\mathbf{x}^{(2)})^T \\ \cdot \\ \cdot \\ (\mathbf{x}^{(N)})^T \end{bmatrix} = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & \cdot & \cdot & x_D^{(1)} \\ x_1^{(2)} & x_2^{(2)} & \cdot & \cdot & x_D^{(2)} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ x_1^{(N)} & \cdot & \cdot & \cdot & x_D^{(N)} \end{bmatrix}$$

- Cluster assignments:

$$\mathbf{Z} = \begin{bmatrix} (\mathbf{z}^{(1)})^T \\ (\mathbf{z}^{(2)})^T \\ \cdot \\ \cdot \\ (\mathbf{z}^{(N)})^T \end{bmatrix} = \begin{bmatrix} z_1^{(1)} & z_2^{(1)} & \cdot & \cdot & z_K^{(1)} \\ z_1^{(2)} & z_2^{(2)} & \cdot & \cdot & z_K^{(2)} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ z_1^{(N)} & \cdot & \cdot & \cdot & z_K^{(N)} \end{bmatrix}$$

Mathematics

- Loss function:

$$\begin{aligned} L(\mathbf{X}, \mathbf{Z}, \boldsymbol{\mu}) &= \sum_{n=1}^N \sum_{k=1}^K z_k^{(n)} \|\mathbf{x}^{(n)} - \mu_k\|^2 \\ &= \|\mathbf{X} - \mathbf{Z}\boldsymbol{\mu}\|_F^2 \end{aligned}$$

- Minimize the loss function $L(\mathbf{X}, \mathbf{Z}, \boldsymbol{\mu})$.
 - Two sets of unknown variables – \mathbf{Z} and $\boldsymbol{\mu}$.
 - Cannot optimize for \mathbf{Z} and $\boldsymbol{\mu}$ at the same time.

- Idea: Perform alternate optimization

- Fixing $\boldsymbol{\mu} = \bar{\boldsymbol{\mu}}$ optimize for \mathbf{Z} , i.e.

$$\bar{\mathbf{Z}} = \arg \min_{\mathbf{Z}} L(\mathbf{X}, \mathbf{Z}, \bar{\boldsymbol{\mu}})$$

- Fixing $\mathbf{Z} = \bar{\mathbf{Z}}$ optimize for $\boldsymbol{\mu}$, i.e.

$$\bar{\boldsymbol{\mu}} = \arg \min_{\boldsymbol{\mu}} L(\mathbf{X}, \bar{\mathbf{Z}}, \boldsymbol{\mu})$$

Alternate optimization – Step 1

- Minimize $L(\mathbf{X}, \mathbf{Z}, \bar{\boldsymbol{\mu}})$ with respect to \mathbf{Z} .

$$\begin{aligned} L(\mathbf{X}, \mathbf{Z}, \bar{\boldsymbol{\mu}}) &= \sum_{n=1}^N \sum_{k=1}^K z_k^{(n)} \|\mathbf{x}^{(n)} - \mu_k\|^2 \\ &= \sum_{k=1}^K z_k^{(1)} \|\mathbf{x}^{(1)} - \mu_k\|^2 + \dots + \sum_{k=1}^K z_k^{(N)} \|\mathbf{x}^{(N)} - \mu_k\|^2 \end{aligned}$$

- Approach: Minimize $L(\mathbf{X}, \mathbf{Z}, \bar{\boldsymbol{\mu}})$ with respect to each $\mathbf{z}^{(n)}$, i.e. minimize each of the above terms separately:

$$\bar{\mathbf{z}}^{(n)} = \arg \min_{\mathbf{z}^{(n)}} \sum_{k=1}^K z_k^{(n)} \|\mathbf{x}^{(n)} - \mu_k\|^2$$

- The above is equivalent to assigning $\mathbf{x}^{(n)}$ to its nearest centroid.

Alternate optimization – Step 2

- Minimize $L(\mathbf{X}, \bar{\mathbf{Z}}, \boldsymbol{\mu})$ with respect to $\boldsymbol{\mu}$.

$$\begin{aligned} L(\mathbf{X}, \bar{\mathbf{Z}}, \boldsymbol{\mu}) &= \sum_{n=1}^N \sum_{k=1}^K z_k^{(n)} \|\mathbf{x}^{(n)} - \mu_k\|^2 \\ &= \sum_{n=1}^N z_1^{(n)} \|\mathbf{x}^{(n)} - \mu_1\|^2 + \dots + \sum_{n=1}^N z_K^{(n)} \|\mathbf{x}^{(n)} - \mu_K\|^2 \end{aligned}$$

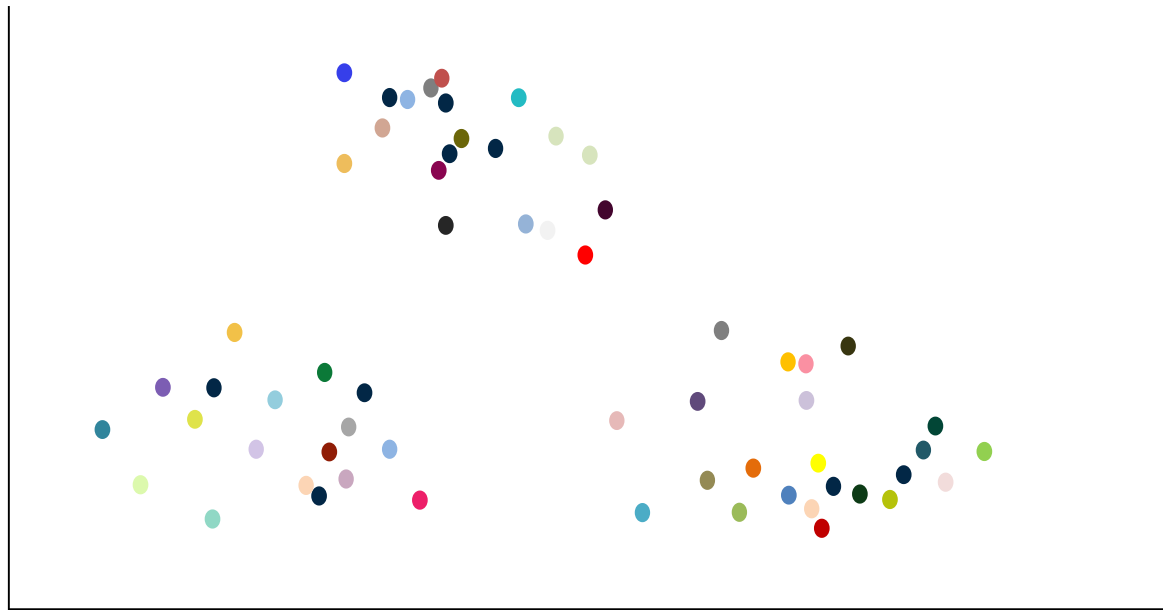
- Approach: Minimize $L(\mathbf{X}, \bar{\mathbf{Z}}, \boldsymbol{\mu})$ with respect to each μ_k separately. So need to minimize each of the above terms separately.
- The optimized value of μ_k is obtained as

$$\bar{\mu}_k = \arg \min_{\mu_k} \sum_{n=1}^N z_k^{(n)} \|\mathbf{x}^{(n)} - \mu_k\|^2$$

- This is equivalent to setting $\bar{\mu}_k$ to be the mean of all the data points in the k th cluster.

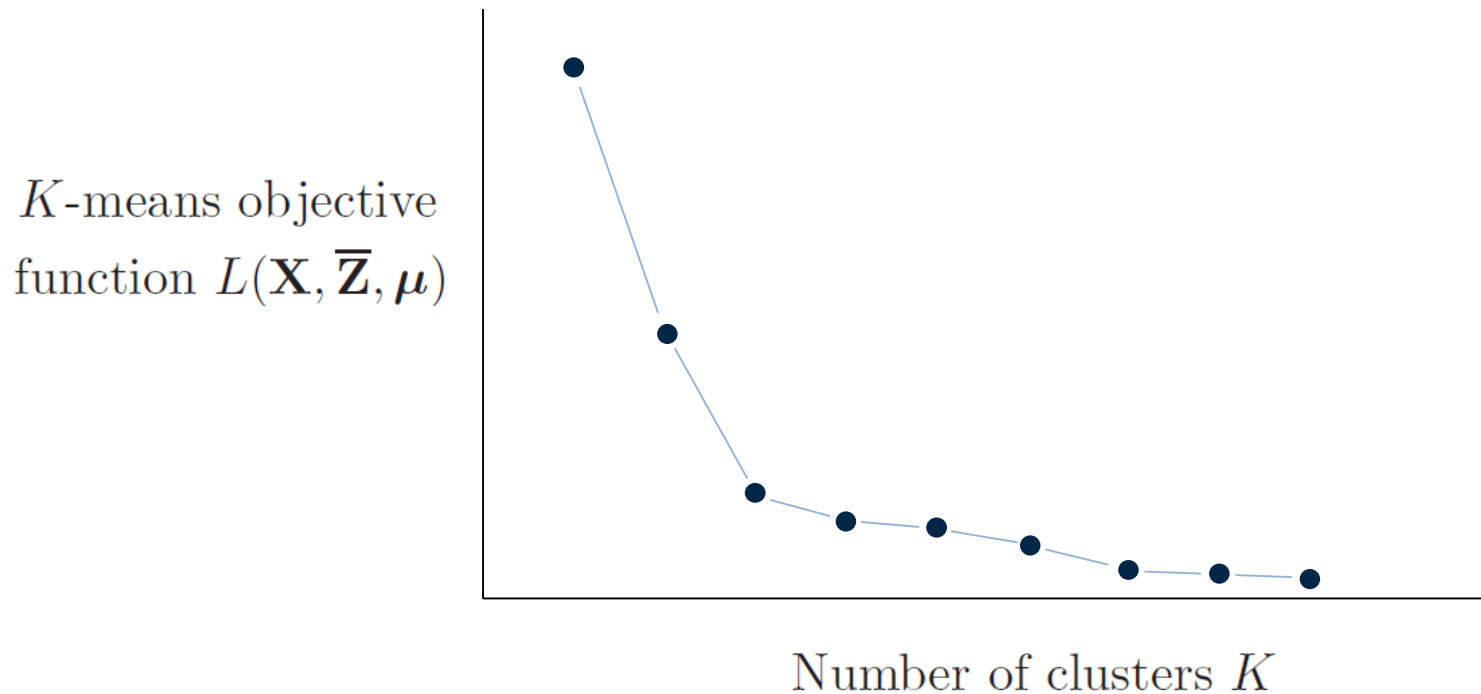
Selecting K value

- As the number of clusters K is increased, the value of the objective function $L(\mathbf{X}, \overline{\mathbf{Z}}, \boldsymbol{\mu})$ decreases.
 - For $K = N$ and $\mu_k = \mathbf{x}^{(k)}$, the value of the objective function $L(\mathbf{X}, \overline{\mathbf{Z}}, \boldsymbol{\mu})$ becomes 0.



Selecting K value

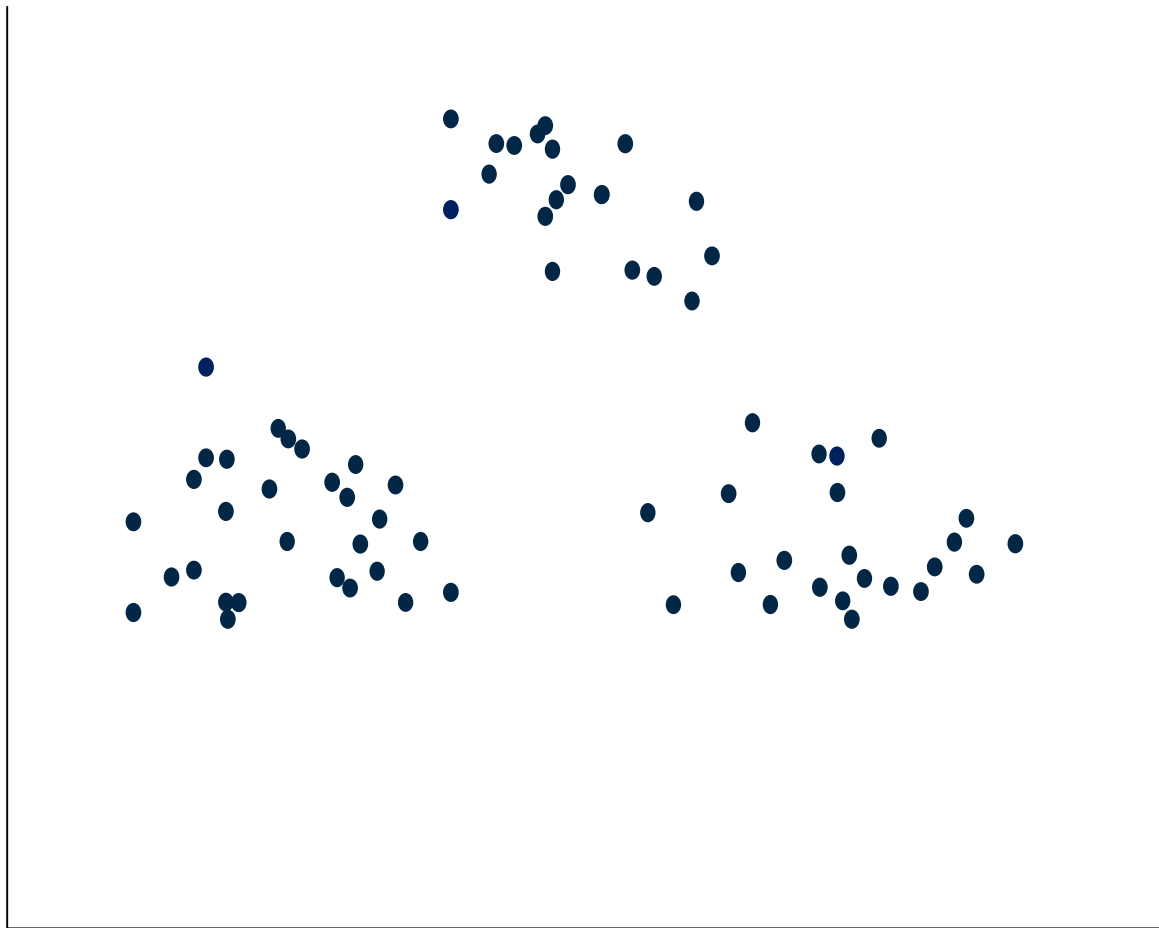
- One approach: Try K -means algorithm for different values of K , and select K to be at the “elbow point” with respect to the variation of $L(\mathbf{X}, \bar{\mathbf{Z}}, \boldsymbol{\mu})$ versus K .



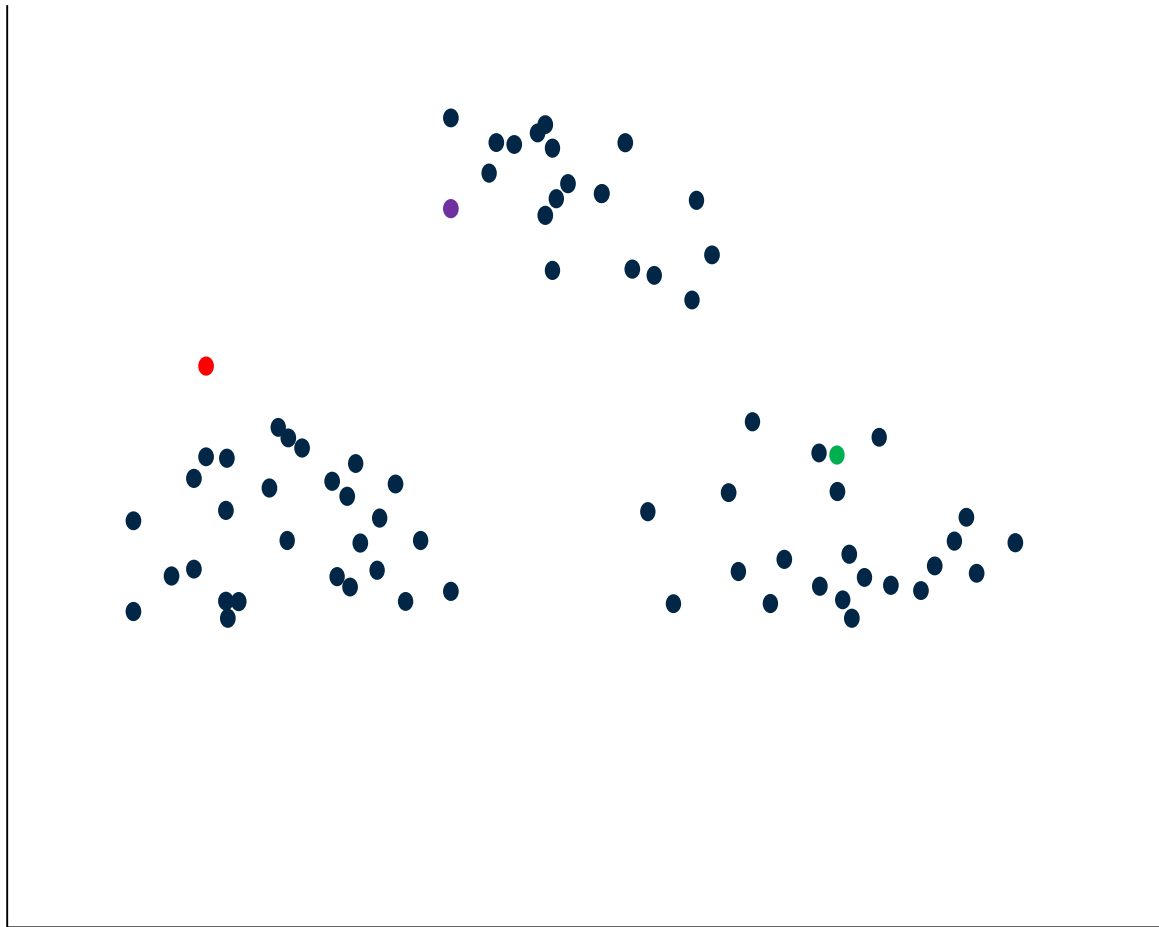
K -medoids

- The center of each cluster is taken to be one of the examples (data points) in the cluster.
 - In contrast, K -means take the mean of a cluster to be its center.
- K -medoids is more robust to outliers and noise.

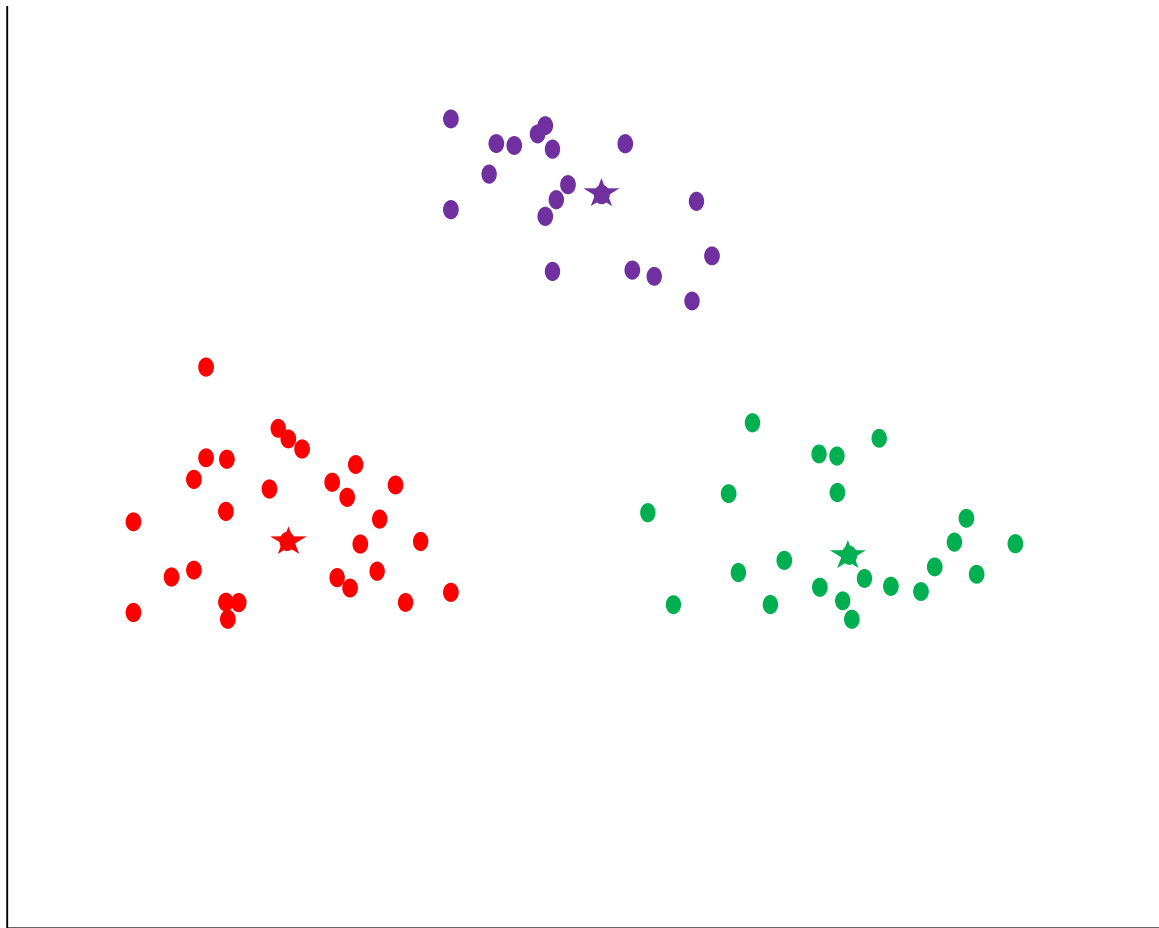
Dataset



K-medoids initialization



K-medoids



Procedure

- Step 1: Initialize the centers of K clusters – one approach is to randomly select K of the given N data points.
- Step 2: For $n = 1, 2, \dots, N$:
 - Compute the distance of the n th data point to all the K centers, and assign $\mathbf{x}^{(n)}$ to the cluster to which it is the closest:

$$z^{(n)} = \arg \min_k \|\mathbf{x}^{(n)} - \mu_k\|^2$$

- Step 3: Recompute the medoid of each cluster. The medoid of the k th cluster is computed as

$$\mu_k = \arg \min_{\mathbf{x}^{(i)} \in \mathcal{C}_k} \sum_{j: \mathbf{x}^{(j)} \in \mathcal{C}_k} \|\mathbf{x}^{(j)} - \mathbf{x}^{(i)}\|^2$$

- If none of the cluster assignments have changed, **STOP**. Else, **REPEAT** from Step 2.