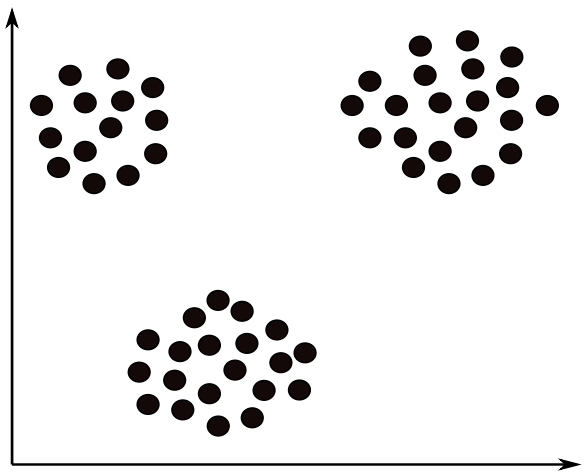
Clustering

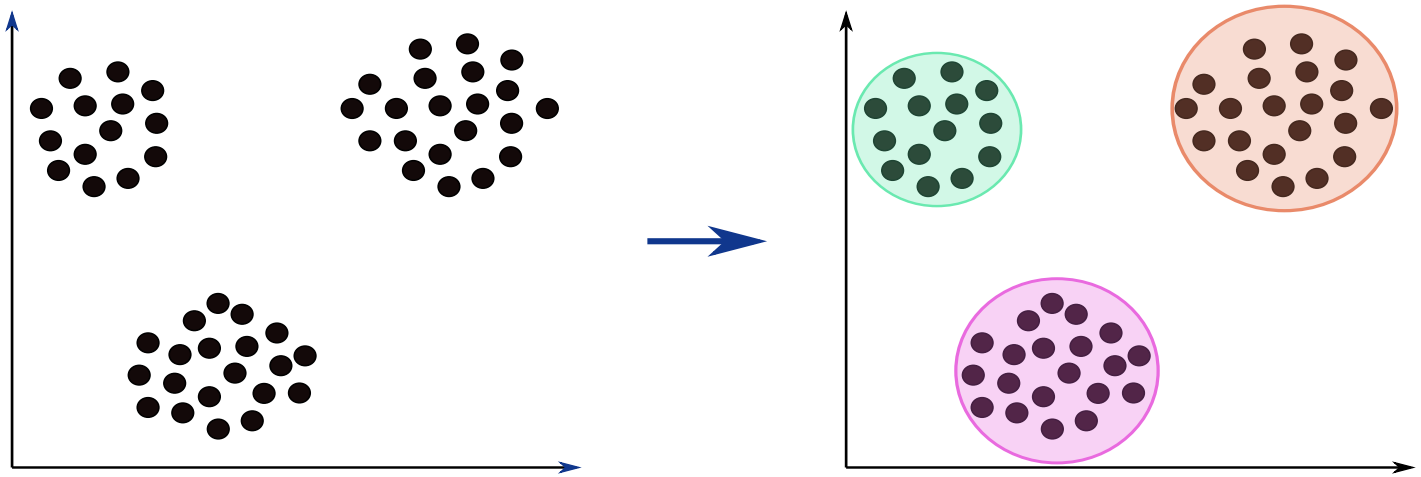
By Van Dinh Tran

November 5, 2023

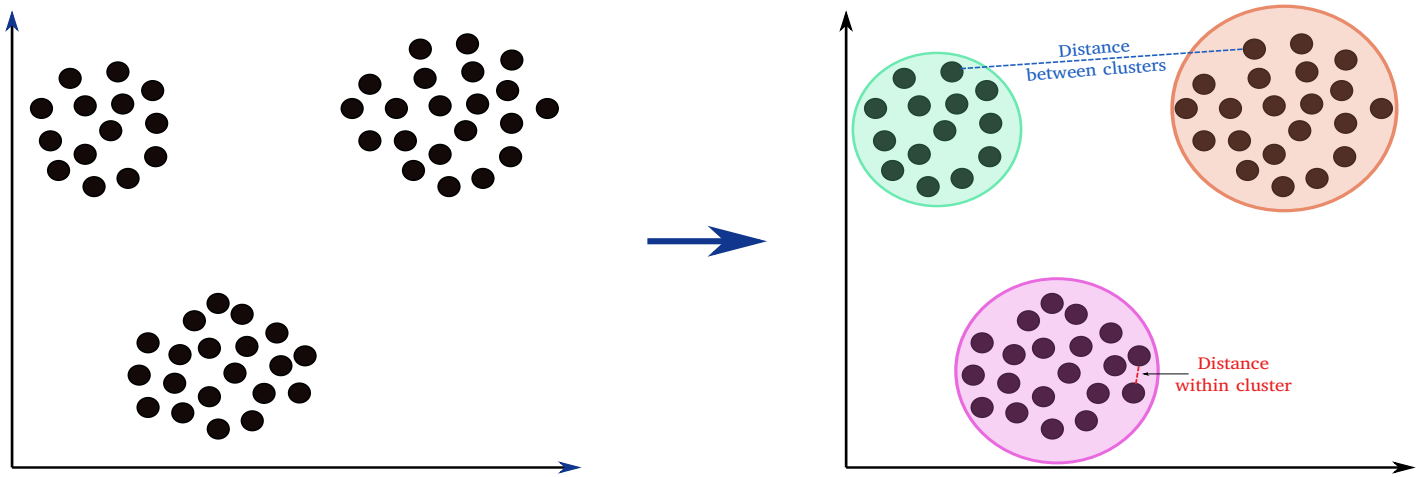
Clustering



Clustering



Clustering



Clustering

Definition: A function f is called a clustering function if it takes a set of data points (D) and a distance metric (d) as input and returns a partition (P) on D.

$$\mathrm{P} = f(d, \mathrm{D})$$

- $\mathrm{P} = \{\mathrm{C}_1, \mathrm{C}_2, \ldots, \mathrm{C}_k\}$ s.t $\mathrm{C}_1 \cup \mathrm{C}_2 \cup \ldots \cup \mathrm{C}_k = \mathrm{D}$; $\mathrm{C}_i \cap \mathrm{C}_j = \Phi$
- C_i are called clusters

Clustering

Definition: A function f is called a clustering function if it takes a set of data points (D) and a distance metric (d) as input and returns a partition (P) on D.

$$\mathrm{P} = f(d, \mathrm{D})$$

- $\mathrm{P} = \{\mathrm{C}_1, \mathrm{C}_2, \ldots, \mathrm{C}_k\}$ s.t $\mathrm{C}_1 \cup \mathrm{C}_2 \cup \ldots \cup \mathrm{C}_k = \mathrm{D}$; $\mathrm{C}_i \cap \mathrm{C}_j = \Phi$
- C_i are called clusters

Three desirable properties of a clustering algorithm.

- Scale invariance: $f(d, \mathrm{D}) = f(\alpha d, \mathrm{D})$, α is a scalar
- Richness: any possible P on D should be a possible outcome of f
- Consistency: If we apply f on D using d' and it reduces within distances and increases between distances, $f(d, \mathrm{D}) = f(d', \mathrm{D}) = \mathrm{P}$.

Clustering

Definition: A function f is called a clustering function if it takes a set of data points (D) and a distance metric (d) as input and returns a partition (P) on D.

$$\mathrm{P} = f(d, \mathrm{D})$$

- $\mathrm{P} = \{\mathrm{C}_1, \mathrm{C}_2, \ldots, \mathrm{C}_k\}$ s.t $\mathrm{C}_1 \cup \mathrm{C}_2 \cup \ldots \cup \mathrm{C}_k = \mathrm{D}$; $\mathrm{C}_i \cap \mathrm{C}_j = \Phi$
- C_i are called clusters

Three desirable properties of a clustering algorithm.

- Scale invariance: $f(d, \mathrm{D}) = f(\alpha d, \mathrm{D})$, α is a scalar
- Richness: any possible P on D should be a possible outcome of f
- Consistency: If we apply f on D using d' and it reduces within distances and increases between distances, $f(d, \mathrm{D}) = f(d', \mathrm{D}) = \mathrm{P}$.

[5ex] **No function that satisfies those three properties**

$\Longrightarrow$ Constraint relaxation

Clustering

- **Common distance metrics**
  - Euclidean distance: $d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$
  - Manhattan distance: $d(x, y) = \sum_{i=1}^{n} |x_i - y_i|$
  - Peason correlation distance: $d(x, y) = 1 - \frac{\sum_{i=1}^{n} (x_i - \hat{x})(y_i - \hat{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \hat{x})^2 \sum_{i=1}^{n} (y_i - \hat{y})^2}}$
  - Kernel-based distance

Clustering

- **Common distance metrics**
  - Euclidean distance: $d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$
  - Manhattan distance: $d(x, y) = \sum_{i=1}^{n} |x_i - y_i|$
  - Peason correlation distance: $d(x, y) = 1 - \frac{\sum_{i=1}^{n} (x_i - \hat{x})(y_i - \hat{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \hat{x})^2 \sum_{i=1}^{n} (y_i - \hat{y})^2}}$
  - Kernel-based distance
- **Metrics to measure clustering performance**
  - Internal metrics
    - Measures of cohesion and separation: Within-cluster sum of squares (WSS) and Between-cluster sum of squares (BSS)
    - Measures of validity: Silhouette score, Davies-Bouldin index, etc
  - External metrics: Rand index, adjusted rand index

Clustering

- **Some types of clustering methods**
  - *Centroid models* (e.g k-means): represents each cluster by a single mean vector, each point is assigned to the cluster whose nearest centroid.
  - *Distribution models*: clusters are modeled using statistical distributions, such as multivariate normal distributions used by the expectation-maximization algorithm.
  - *Connectivity models* (e.g hierarchical clustering): build models based on distance connectivity
  - *Density models* (e.g DBSCAN) defines clusters as connected dense regions in the data space.

K-means

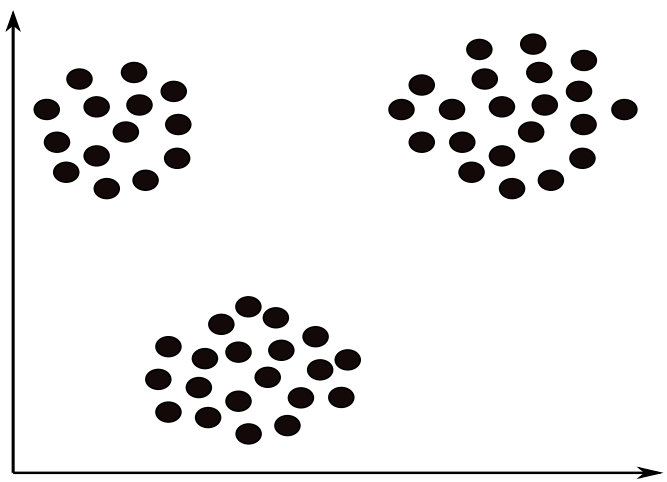
Dataset $D = \{x_i | x_i \in R^n\}$, a distance metric d

1. Specify number of clusters K
2. Initialize K data points as cluster centroids
3. Keep iterating until there is no change to the centroids
   - 3.1 Assign each data point to the closest cluster centroid.
   - 3.2 Compute the centroids for the clusters by taking the average of the all data points that belong to each cluster.

K-means

Dataset $D = \{x_i | x_i \in R^n\}$, a distance metric d

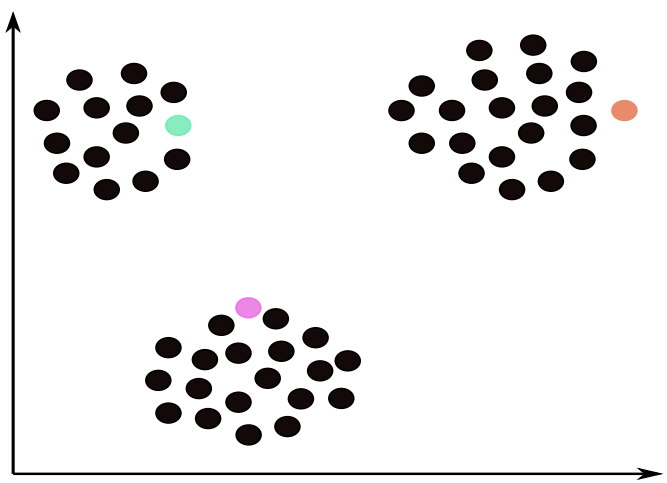
1. Specify number of clusters K
2. Initialize K data points as cluster centroids
3. Keep iterating until there is no change to the centroids
   - 3.1 Assign each data point to the closest cluster centroid.
   - 3.2 Compute the centroids for the clusters by taking the average of the all data points that belong to each cluster.



K-means

Dataset $D = \{x_i | x_i \in R^n\}$, a distance metric d

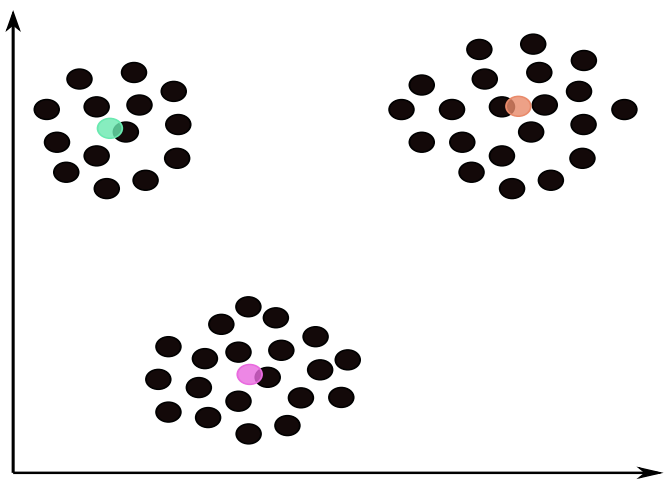
1. Specify number of clusters K
2. Initialize K data points as cluster centroids
3. Keep iterating until there is no change to the centroids
   - 3.1 Assign each data point to the closest cluster centroid.
   - 3.2 Compute the centroids for the clusters by taking the average of the all data points that belong to each cluster.



K-means

Dataset $D = \{x_i | x_i \in R^d\}$, a distance metric d

1. Specify number of clusters K
2. Initialize K data points as cluster centroids
3. Keep iterating until there is no change to the centroids
   - 3.1 Assign each data point to the closest cluster centroid.
   - 3.2 Compute the centroids for the clusters by taking the average of the all data points that belong to each cluster.



K-means

– **Total distances:**

$$
\begin{aligned}
\mathrm{T} &= \sum_{\mathrm{k}=1}^{\mathrm{K}} \sum_{\mathrm{i}\in \mathrm{C_k}} \Big(\sum_{\mathrm{j}\in \mathrm{C_k}} \mathrm{d_{ij}} + \sum_{\mathrm{j}\neq \mathrm{C_k}} \mathrm{d_{ij}} \Big) \\
&= \sum_{\mathrm{k}=1}^{\mathrm{K}} \sum_{\mathrm{i}\in \mathrm{C_k}} \sum_{\mathrm{j}\in \mathrm{C_k}} \mathrm{d_{ij}} + \sum_{\mathrm{k}=1}^{\mathrm{K}} \sum_{\mathrm{i}\in \mathrm{C_k}} \sum_{\mathrm{j}\neq \mathrm{C_k}} \mathrm{d_{ij}} \\
&= \mathrm{T_w} + \mathrm{T_b}
\end{aligned}
$$

K-means

– **Total distances:**

$$
\begin{aligned}
T &= \sum_{k=1}^{K} \sum_{i \in C_k} \Big(\sum_{j \in C_k} d_{ij} + \sum_{j \neq C_k} d_{ij} \Big) \\
&= \sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j \in C_k} d_{ij} + \sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j \neq C_k} d_{ij} \\
&= T_w + T_b
\end{aligned}
$$

$$
\text{Normalized } T_w = \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i,j \in C_k} d_{ij} = \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i,j \in C_k} \|x_i - x_j\|^2 = \sum_{k=1}^{K} \sum_{i \in C_k} \|x_i - \mu_k\|^2
$$

K-means

- **Total distances:**

$$\begin{aligned} T &= \sum_{k=1}^{K} \sum_{i \in C_k} \Big(\sum_{j \in C_k} d_{ij} + \sum_{j \neq C_k} d_{ij} \Big) \\ &= \sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j \in C_k} d_{ij} + \sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j \neq C_k} d_{ij} \\ &= T_w + T_b \end{aligned}$$

$$\text{Normalized } T_w = \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i,j \in C_k} d_{ij} = \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i,j \in C_k} \|x_i - x_j\|^2 = \sum_{k=1}^{K} \sum_{i \in C_k} \|x_i - \mu_k\|^2$$

- **Idea:**

$$(T_w \to \min; T_b \to \max) \Longleftrightarrow T_w \to \min$$

K-means

- **Total distances:**

$$\begin{aligned} T &= \sum_{k=1}^{K} \sum_{i \in C_k} \Big(\sum_{j \in C_k} d_{ij} + \sum_{j \neq C_k} d_{ij} \Big) \\ &= \sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j \in C_k} d_{ij} + \sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j \neq C_k} d_{ij} \\ &= T_w + T_b \end{aligned}$$

$$\text{Normalized } T_w = \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i,j \in C_k} d_{ij} = \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i,j \in C_k} \|x_i - x_j\|^2 = \sum_{k=1}^{K} \sum_{i \in C_k} \|x_i - \mu_k\|^2$$

- **Idea:**

$$(T_w \to \min; T_b \to \max) \iff T_w \to \min$$

- **Objective function:**

$$\underset{\mu, C}{\operatorname{argmin}} \sum_{k=1}^{K} \sum_{i \in C_k} \|x_i - \mu_k\|^2$$

K-means

- **Total distances:**

$$\begin{aligned} T &= \sum_{k=1}^{K} \sum_{i \in C_k} \Big(\sum_{j \in C_k} d_{ij} + \sum_{j \neq C_k} d_{ij} \Big) \\ &= \sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j \in C_k} d_{ij} + \sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j \neq C_k} d_{ij} \\ &= T_w + T_b \end{aligned}$$

$$\text{Normalized } T_w = \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i,j \in C_k} d_{ij} = \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i,j \in C_k} \|x_i - x_j\|^2 = \sum_{k=1}^{K} \sum_{i \in C_k} \|x_i - \mu_k\|^2$$

- **Idea:**

$$(T_w \to \min; T_b \to \max) \Longleftrightarrow T_w \to \min$$

- **Objective function:**

$$\underset{\mu, C}{\operatorname{argmin}} \sum_{k=1}^{K} \sum_{i \in C_k} \|x_i - \mu_k\|^2$$

Note: K-means is guaranteed to converge, but not same result and speed due to different initialization.

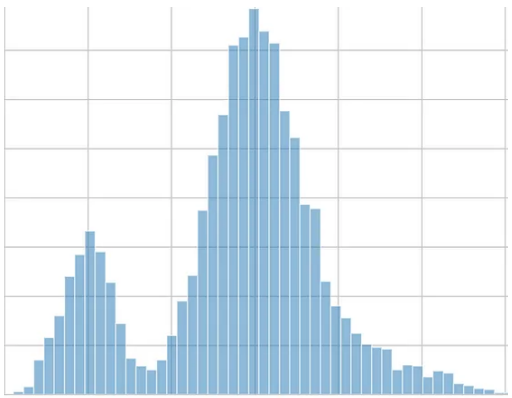
Gaussain mixture models (GMMs)

https://www.naftaliharris.com/blog/visualizing-k-means-clustering/

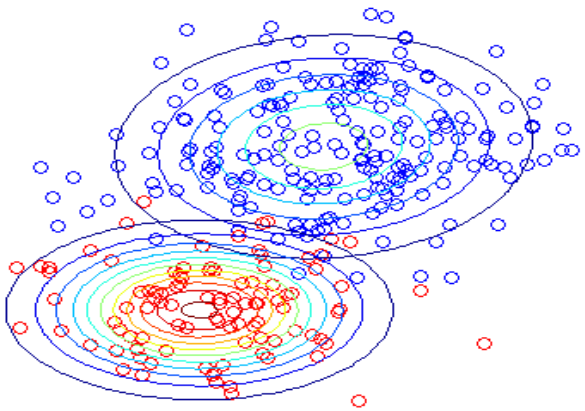
Initialization

- **Classical clustering methods:**
  - Work well with low-dimensional data
  - Poorly perform with high-dimensional data due to the sparsity (curse of dimensionality)
- **Solution with high dimensional data:**
  - Dimension reduction + clustering
  - Autoencoder + clustering (sequential or simultaneous manner)
  - Autoencoder is key for most deep clustering methods

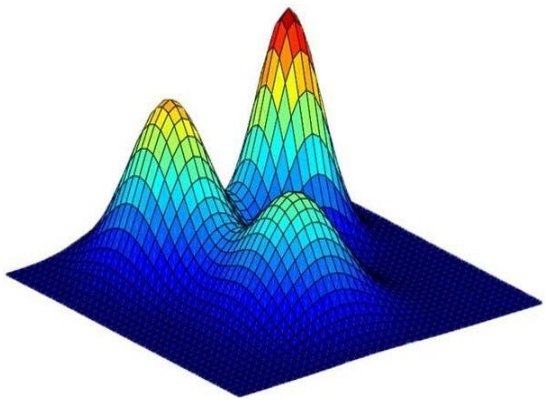
Gaussain mixture models (GMMs)



1D



2D



3D

Gaussain mixture models (GMMs)