UL02. Clustering

Scan me

Clustering Problem:

- Unsupervised learning: Making sense out of unlabeled data (data description).
- The clustering problem:
  - Given: A set of objects X

    Inter-object distances $D(x, y) = D(y, x) \quad x, y \in X$
  - Output: Partition $P_D(x) = P_D(y) \quad if\ x\ and\ y\ are\ in\ the\ same\ cluster$

Single Linkage Clustering – SLC:

- Consider each object a cluster (n objects).
- Define inter-cluster distance as the distance between the closest points in the two clusters.
- Merge two closest clusters.
- Repeat $n - k$ times to make k clusters.
  - k is an input.
- How you define the inter-cluster distance is a domain knowledge.
- Running time of SLC: $O(n^3)$
- Issues with SLC: It might end up with wrong clusters, depending on the distance definition.

k-Means Clustering:

- Pick k random center points.
- Each center claims its closest points.
- Recompute the centers by averaging the clustered points.
- Repeat until converge.
- k-means in Euclidean space:
  - $P^t(x)$: Partition/cluster of object x.
  - C_i^t: Set of all points in cluster $i = \{x\ such\ that\ P(x) = i\}$
  - $center_i^t$: $\sum_{y \in C_i^t} y \Big/ |C_i|$

$$P^t(x) = argmin_i \left\| x - center_i^{t-1} \right\|_2^2$$

$$center_i^t = \sum_{y \in C_i^t} y \Big/ |C_i|$$

Use the new center to re-compute $P^t(x)$.

- Properties of k-means clustering:
  - Each iteration is polynomial $O(kn)$.
  - Finite (exponential) iterations $O(k^n)$.
  - Error decreases (if ties broken consistently).
  - Can get stuck, if it started with wrong centers. This is similar to converging to a local optima. One solution to this problem is to use random restarts.

Soft Clustering:

- Soft Clustering attaches cluster probability to each point, instead of a specific cluster.
- Assume the data was generated by:
  - Select one of k possible Gaussians (Fixed know variance) uniformly.
  - Sample x_i from that Gaussian.
  - Repeat n times.
- Task: Find a hypothesis $h = \langle \mu_1, \dots, \mu_k \rangle$ ($\mu_1, \dots, \mu_k$ are Gaussian means) that maximizes the probability of the data (Maximum likelihood).
- Maximum Likelihood Gaussian:
  - The Maximum Likelihood mean of the Gaussian μ is the mean of the data.

Expectation Maximization – EM:

- EM assigns a cluster probability to each point.

$$E[Z_{ij}] = \frac{P(x = x_i | \mu = \mu_j)}{\sum_{i=1}^{k} P(x = x_i | \mu = \mu_j)}$$

$$\mu_j = \frac{\sum_i E[Z_{ij}] x_i}{\sum_i E[Z_{ij}]}$$

$$P(x = x_i | \mu = \mu_j) = e^{\frac{-1}{2}\sigma^2 (x_i - \mu_j)^2}$$

  - Use the new μ_j to re-compute $E[Z_{ij}]$.
  - Expectation: $E[Z_{ij}]$ defines the probability that element i was produced by cluster j.
  - Maximization: μ_j defines the mean of cluster j.
- EM properties:
  - Monotonically non-decreasing likelihood: It's not getting worse.
  - Does not converge (practically converge).
  - Will not diverge.
  - Most probably, it will get stuck in a local optima: Use random restarts.
  - Works with any distribution (if E and μ are solvable).

Clustering Properties:

- Richness: For any assignment C of objects to clusters, there's some distance matrix D such that P_D returns that clustering C.
- Scale-invariance: Scaling distances by a positive value doesn't change the clustering:
$$\forall_D \forall_{k>0}\, P_D = P_{kD}$$
- Consistency: shrinking intra-cluster distances (Moving points towards each other) and expanding inter-cluster (Moving clusters away from each other) distances doesn't change the clustering:
$P_D = \acute{P}_D$.
- Impossibility Theorem: There's no clustering algorithm that can achieve these three properties.