Lecture 12: Clustering (Chapter 7 of Textbook B)

Jinwoo Shin

AI503: Mathematics for AI

Introduction

KAIST AI
Graduate School of AI

- Clustering refers to partitioning a set of objects into subsets according to some desired criterion.
  - ex1) Partition a set of news articles into clusters based on the topics of the articles.
  - ex2) Given a set of pictures of people, one might want to group them into clusters based on who is in the image
- Often it is an important step in making sense of large amounts of data.
- Basic notation
  - n: The number of data points.
  - k: The number of desired clusters.
  - $A = \{a_1, \ldots, a_n\}$: Matrix representation of n data points with rows $a_1, \ldots, a_n$.

Roadmap

KAIST AI
Graduate School of AI

(1) k-Means Clustering

(2) k-Center Clustering

(3) Spectral Clustering

(4) High-Density Clusters

Roadmap

KAIST AI
Graduate School of AI

(1) k-Means Clustering

(2) k-Center Clustering

(3) Spectral Clustering

(4) High-Density Clusters

A Maximum-Likelihood Motivation

- Suppose that the data was generated according to an equal weight mixture of k spherical well-separated Gaussian densities centered at $\mu_1, \mu_2, \ldots, \mu_k$, each with variance one in every direction.
- The density of the mixture is: (data points lie in R^d and $\mu(\mathrm{x})$ is the center nearest to x.)

$$\text{Prob}(\mathrm{x}) = \frac{1}{(2\pi)^{d/2}} \frac{1}{k} \sum_{i=1}^{k} e^{-|\mathrm{x}-\mu_i|^2}$$

- The sum of exponential functions is dominated by the largest. Thus

$$\text{Prob}(\mathrm{x}) \approx \frac{1}{(2\pi)^{d/2} k} e^{-|\mathrm{x}-\mu(\mathrm{x})|^2}$$

A Maximum-Likelihood Motivation

- The likelihood of drawing the sample of points $\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_\mathrm{n}$ from the mixture

$$\frac{1}{k^n} \frac{1}{(2\pi)^{nd/2}} \prod_{i=1}^{n} e^{-\left|\mathrm{x}^{(1)} - \mu\left(\mathrm{x}^{(1)}\right)\right|^2} = ce^{-\sum_{i=1}^{n}\left|\mathrm{x}^{(1)} - \mu\left(\mathrm{x}^{(1)}\right)\right|^2}$$

- Minimizing the sum of squared distances to cluster centers finds the maximum likelihood $\mu_1, \mu_2, \ldots, \mu_\mathrm{n}$, which motivates using the sum of distance squared to the cluster centers.

Structural Properties of the k-Means Objective

KAIST AI
Graduate School of AI

Lemma.

Let $\{\mathrm{a}_1, \mathrm{a}_2, \ldots, \mathrm{a}_\mathrm{n}\}$ be a set of points and $\mathrm{c} = \frac{1}{n}\sum_{i=1}^{n} \mathrm{a}_\mathrm{i}$ is the centroid of the set of points. The sum of the squared distances of the a_i to any point x equals the sum of the squared distances to the centroid of the a_i plus n times the squared distance from x to the centroid. That is,

$$\sum_i |\mathrm{a}_\mathrm{i} - \mathrm{x}|^2 = \sum_i |\mathrm{a}_\mathrm{i} - \mathrm{c}|^2 + n|\mathrm{c} - \mathrm{x}|^2$$

Proof.

$$\begin{aligned}\sum_i |\mathrm{a}_\mathrm{i} - \mathrm{x}|^2 &= \sum_i |\mathrm{a}_\mathrm{i} - \mathrm{c} + \mathrm{c} - \mathrm{x}|^2 \\ &= \sum_i |\mathrm{a}_\mathrm{i} - \mathrm{c}|^2 + 2(\mathrm{c} - \mathrm{x}) \cdot \sum_i (\mathrm{a}_\mathrm{i} - \mathrm{c}) + n|\mathrm{c} - \mathrm{x}|^2\end{aligned}$$

Since c is the centroid, $\sum_i (\mathrm{a}_\mathrm{i} - \mathrm{c}) = 0$. Thus, $\sum_i |\mathrm{a}_\mathrm{i} - \mathrm{x}|^2 = \sum_i |\mathrm{a}_\mathrm{i} - \mathrm{c}|^2 + n|\mathrm{c} - \mathrm{x}|^2$.

Remark. The sum of squared distances of the a_i to a point x is minimized when x is the centroid, which motivates Lloyd's algorithm.

Lloyd's Algorithm

KAIST AI
Graduate School of AI

Lloyd's algorithm

1. Start with k centers.
2. Cluster each point with the center nearest to it.
3. Find the centroid of each cluster and replace the set of old centers with the centroids.
4. Repeat the above two steps until the centers converge according to some criterion, such as the k-means score no longer improving.

- This algorithm always converges to a local minimum of the objective.
- One or more of the clusters can become empty.

Roadmap

KAIST AI
Graduate School of AI

(1) k-Means Clustering

(2) k-Center Clustering

(3) Spectral Clustering

(4) High-Density Clusters

k-Center Clustering

KAIST AI
Graduate School of AI

The Farthest Traversal k-clstering Algorithm

Pick any data point to be the first cluster center. At time t, for $t = 2, 3, ..., k$, pick the farthest data point from any existing cluster center; make it the t^{th} cluster center.

- k-center criterion partitions the points into k clusters so as to minimize the maximum distance of any point to its cluster center.
- Call the maximum distance of any point to its cluster center the radius of the clustering.
- There is a k-clustering of radius r if and only if there are k spheres, each of radius r; which together cover all the points.

Theorem.

If there is a k-clustering of radius $\frac{r}{2}$, then the above algorithm finds a k-clustering with radius at most r.

Proof. Suppose for contradiction that there is some data point x that is distance greater than r from all centers chosen. This means that each new center chosen was distance greater than r from all previous centers, because we could always have chosen x. This implies that we have $k+1$ data points, namely the centers chosen plus x, that are pairwise more than distance r apart. Clearly, no two such points can belong to the same cluster in any k-clustering of radius $\frac{r}{2}$, contradicting the hypothesis.

Roadmap

KAIST AI
Graduate School of AI

(1) k-Means Clustering

(2) k-Center Clustering

(3) Spectral Clustering

(4) High-Density Clusters

Spectral Clustering

KAIST AI
Graduate School of AI

- Let A be a $n \times d$ data matrix with each row a data point and suppose we want to partition the data points into k clusters.
- Spectral clustering refers to a class of clustering algorithms which share the following outline:
  - Find the space V spanned by the top k (right) singular vectors of A.
  - Project data points into V.
  - Cluster the projected points.
- We represent a k-clustering by a $n \times d$ matrix C (same dimensions as A), where row i of C is the center of the cluster to which data point i belongs. So, there are only k distinct rows of C and each other row is a copy of one of these rows.

Spectral Clustering

KAIST AI
Graduate School of AI

The Algorithm

- Find the top k right singular vectors of data matrix A and project rows of A to the space spanned by them to get A_k.
- Select a random row from A_k and form a cluster with all rows of A_k at distance less than $\frac{6k\sigma(C)}{\varepsilon}$ from it, where $\sigma(C) = \|A - C\|_2/\sqrt{n}$.
- Repeat Step 2 k times.

Theorem.

If in a k-clustering C, every pair of centers is separated by at least $15k\sigma(C)/\varepsilon$ and every cluster has at least εn points in it, then with probability at least $1 - \varepsilon$, Spectral Clustering finds a clustering C' that differs from C on at most $\varepsilon^2 n$ points.

Proof. See Page 218-219 of Textbook B.

Roadmap

KAIST AI
Graduate School of AI

(1) k-Means Clustering

(2) k-Center Clustering

(3) Spectral Clustering

(4) High-Density Clusters

High-Density Clusters

Single Linkage: Algorithm begins with each point in its own clusters and then repeatedly merges the two "closet" clusters into one.

Remark

The distance between two clusters is defined as the minimum distance between points in each clusters. That is,

$$d_{\min}(C, C') = \min_{x \in C, y \in C'} d(x, y)$$

High-Density Clusters

KAIST AI
Graduate School of AI

Theorem.

Suppose the desired clustering $C_1^*, \cdots, C_k^*$ satisfies the property that there exists some distance σ such that

(1) Any two data points in different clusters have distance at least σ.

(2) For any cluster C_i^* and any partition of C_i^* into two non-empty sets A and $C_i^* \backslash A$, there exist points on each side of the partition of distance less than σ.

Then, single-linkage will correctly recover the clustering $C_1^*, \cdots, C_k^*$.

Proof. Consider running the algorithm until all pairs of clusters C and $C^{'}$ have $d_{\text{min}}(C, C') \geq \sigma$. At that point, by (2), each target cluster C_i^* will be fully contained within some cluster of the single-linkage algorithm. On the other hand, by (1) and by induction, each cluster C of the single-linkage algorithm will be fully contained within some C_i^* of the target clustering, since any merger of subsets of distinct target clusters would require $d_{\text{min}} \geq \sigma$. Therefore, the single-linkage clusters are indeed the target cluster.

High-Density Clusters

KAIST AI
Graduate School of AI

Robust Linkage: The single-linkage algorithm is fairly brittle. A few points bridging the gap between two different clusters can cause it to do the wrong thing. As a result, there has been significant work developing more robust versions of the algorithm.

Wishart's Algorithm

A ball of radius r is created for each point with the point as center; The radius is gradually increased starting from $r = 0$. The algorithm has a parameter t, when a ball has t or more points, the center of point becomes active. When the two balls with active centers intersect the two center points are connected by an edge. The parameter t prevents a thin string of points between two clusters from causing a spurious merger.

Remark:

$t = 1$, Wishart's algorithm is same as single linkage.

Questions?