CSE 291: Unsupervised learning **Spring 2008**

Lecture 5 — Finding meaningful clusters in data

So far we've been in the *vector quantization* mindset, where we want to approximate a data set by a small number of representatives, and the quality of the approximation is measured by a precise distortion function. Now we move to a different use of clustering: to discover interesting structure in data.

This immediately seems fuzzy and ill-defined, and it is unclear how to make a principled choice among the plethora of clustering heuristics that present themselves. But if we step back and take a slightly more abstract view, some interesting conclusions and directions emerge.

We'll assume that we have a data set S, along with some notion of interpoint distances. For the latter, we use the following formalism.

Definition 1. $d : S \times S \to \mathbb{R}$ is a *distance function* if it satisfies the following two properties:

- $d(x, y) \geq 0$, with equality if and only if $x = y$
- $d(x, y) = d(y, x)$.

In particular, any metric is a distance function, although a distance function need not satisfy the triangle inequality.

Given only S and d, what is a reasonable way to generate a clustering?

5.1 Kleinberg's axiomatic framework for clustering

Kleinberg formalizes a *clustering function* as a mapping that takes as input a distance function d on n points and returns a partition Γ of $[n] = \{1, 2, \ldots, n\}$; if this mapping is f, we will write $f(d) = \Gamma$. For instance,

$$f\left(\begin{bmatrix} 0 & 10 & 10 \\ 10 & 0 & 1 \\ 10 & 1 & 0 \end{bmatrix}\right) = (\{1\}, \{2, 3\})$$

sends a distance function on three points to a clustering with two clusters. Notice that the clustering function has to choose the number of clusters.

5.1.1 Axioms for reasonable clustering functions

What properties might one expect a good clustering function f to possess? Here are some possibilities.

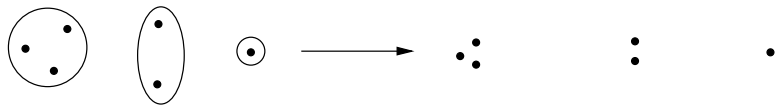
- *Scale invariance.* For any $\alpha > 0$ and any d, $f(\alpha \cdot d) = f(d)$.
- *Richness.* Range$(f) = \{$all possible partitions of $[n]\}$.
- *Consistency.* If $f(d) = \Gamma$ and d' is a Γ-enhancing transformation of d, then $f(d') = \Gamma$.

For the last property, a Γ-enhancing transformation is one that makes a distance function even more suggestive of the clustering Γ. More precisely:

Definition 2. d' is a Γ-*enhancing transformation* of d if

$$\begin{aligned} d'(i,j) &\leq d(i,j) \quad \text{for } i,j \text{ in the same cluster of } \Gamma \\ d'(i,j) &\geq d(i,j) \quad \text{for } i,j \text{ in different clusters of } \Gamma \end{aligned}$$

For instance, in the figure below, the left side shows a set of six points (with Euclidean distances d) and a particular clustering Γ consisting of three clusters. The right side shows a modified configuration whose distance function d' is a Γ-transformation of d.



What kinds of algorithms satisfy the axioms above? Here are three illustrative examples.

1. Run single linkage till you get k clusters.

   This satisfies scale invariance and consistency, but not richness.

2. Run single linkage till distances exceed $c \cdot \max_{i,j} d(i,j)$, where c is some constant.

   This satisfies scale invariance and richness but not consistency.

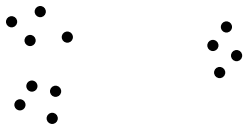
3. Run single linkage until distances exceed some threshold r.

   This satisfies richness and consistency but not scale invariance.

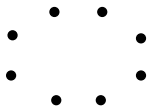
In fact, Kleinberg shows that there is no clustering function that satisfies all three axioms.

5.1.2 An impossibility result

The basic problem is that there is no consistent way to choose a level of granularity. Does the following picture contain 2 clusters, or 3, or 12?



Let's see how this problem makes it impossible to satisfy all three axioms. Suppose we have eight points whose interpoint distances d look like this:



This looks like one cluster, so suppose $f(d) = \Gamma$ consists of a single cluster. Then a Γ-enhancing transformation yields the following d' in which the four points initially on the left are now almost on top of each other, and likewise with the four points initially on the right.

Rescaling, we get d'':

Now, consistency and scale invariance would enforce $f(d'') = f(d') = f(d)$. But d looks like one cluster while d'' looks like two!

To make this intuition more precise, we introduce a key concept.

Definition 3. Partition Γ' is a *refinement* of Γ (written $\Gamma' \preceq \Gamma$) if for every $C' \in \Gamma'$, there is some $C \in \Gamma$ such that $C' \subset C$.

In other words, Γ can be obtained by merging clusters of Γ'. For instance:

Γ Γ'

Theorem 4. *Pick any $\Gamma_0 \preceq \Gamma_1$ with $\Gamma_0 \neq \Gamma_1$. If f satisfies scale invariance and consistency then Γ_0 and Γ_1 cannot both be in range(f).*

Proof. Suppose, on the contrary, that $f(d_0) = \Gamma_0$ and $f(d_1) = \Gamma_1$. We'll arrive at a contradiction.

Let $a_0 = \min\{d_0(i,j) : i \neq j\}$ and $b_0 = \max\{d_0(i,j)\}$; likewise, let a_1 and b_1 be the smallest and largest values of d_1. Now construct a new distance function d:

$$d(i,j) = \begin{cases} \frac{a_0 a_1}{b_0} & \text{if } i,j \text{ in same cluster of } \Gamma_0 \\ a_1 & \text{if } i,j \text{ in same cluster of } \Gamma_1 \text{ but not } \Gamma_0 \\ b_1 & \text{if } i,j \text{ in different clusters of } \Gamma_1 \end{cases}$$

Distance function d is a Γ_1-enhancing transformation of d_1. Therefore, $f(d) = f(d_1) = \Gamma_1$. But now consider distance function $d' = (b_0/a_1) \cdot d$; notice that

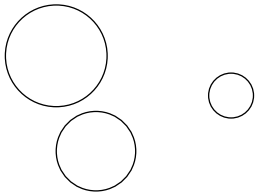
$$d'(i,j) = \begin{cases} a_0 & \text{if } i,j \text{ in same cluster of } \Gamma_0 \\ \geq b_0 & \text{if } i,j \text{ in different clusters of } \Gamma_0 \end{cases}$$

Therefore d' is a Γ_0-enhancing transformation of d_0 and $f(d') = f(d_0) = \Gamma_0$. But d' is also a scaled version of d, so we need $f(d) = f(d')$, a contradiction. □

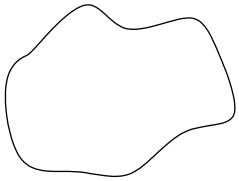
Problem 1. Develop an axiomatic framework for *hierarchical* clustering. This will circumvent the difficulty of automatically choosing a level of granularity.

5.2 Finding prominent clusterings

In some situations, we primarily want a clustering algorithm to find prominent groups in the data, *if such groups exist*. If, on the other hand, the data has no well-defined clusters, then it doesn't matter what the algorithm does; it shouldn't kill itself trying to optimize some cost function, for instance. These two situations are depicted in the cartoon below.



Meaningful structure



No structure

This style of analysis can potentially avoid the negative results (NP-hardness) associated with vector quantization. But how should "well-defined clustering" be formalized? The recent literature offers at least two alternatives.

1. *Probabilistic models.* In this case, it is assumed that the data come from a mixture model whose component distributions are (i) from some well-behaved family (such as Gaussians, or logconcave measures) and (ii) are reasonably well-separated from each other. The goal is to recover this mixture model from the data. The assumption itself is unreasonable, but it is a useful lower bound, in the sense that any algorithm that fails this criterion is really pretty bad. Also, this kind of analysis has given many insights into the geometry of high-dimensional clusters.
2. *Clusterings defined by simple properties.* The latest framework of this kind is due to Balcan, Blum, and Vempala, and we turn to it next.

Suppose we are given a data set S, and some associated distance function $d : S \times S \to \mathbb{R}$. If S has a prominent clustering $S = C_1^* \cup C_2^* \cup \cdots \cup C_k^*$ (where the C_i^* are disjoint), we'd like to discover it. We will define a prominent clustering to be one in which, roughly, "points in the same cluster are closer together than points in different clusters." It turns out that there are many ways to make this kind of property precise, with subtle differences between them. We'll consider three possible such notions.

First some notation. For $x \in S$, let $C^*(x) \in \{C_1^*, \ldots, C_k^*\}$ be the "true" cluster containing x. And for $S_1, S_2 \subset S$, define the average distance between these two sets to be

$$d_{\text{avg}}(S_1, S_2) = \frac{1}{|S_1| \cdot |S_2|} \sum_{x \in S_1} \sum_{y \in S_2} d(x, y).$$

In contrast, recall that for $x \in S$, we define $d(x, S) = \min_{y \in S} d(x, y)$.

Notion 1: STRICT ORDERING

Clustering $\{C_1^*, \ldots, C_k^*\}$ satisfies the *strict ordering* property if:

> For all $x \in S$, $y \in C^*(x)$, and $z \notin C^*(x)$, we have $d(x, y) < d(x, z)$.

See Figure 5.1 for an example.

Are commonly-used clustering algorithms certain to recover such a clustering? Well, first of all, we don't have advance knowledge of the number of clusters, so hierarchical schemes are a good bet. Let's look at the usual possibilities: single linkage, complete linkage, and the three variants of average linkage.

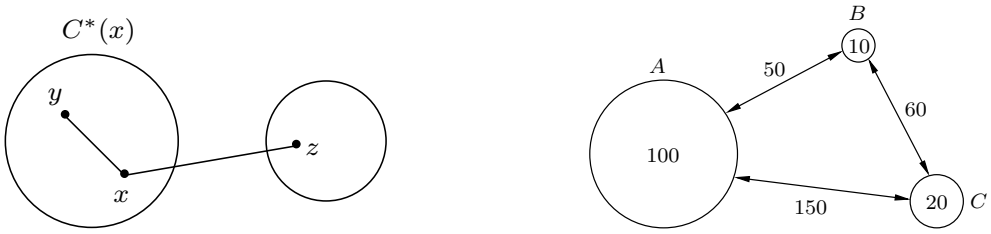


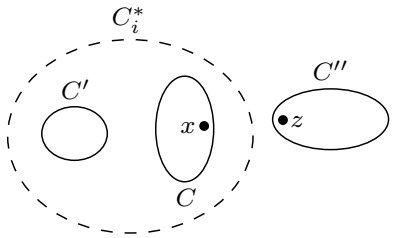
Figure 5.1. Left: strict ordering says $d(x, y) < d(x, z)$. Right: Consider a data set $S = A \cup B \cup C$ where, as shown in the figure, all distances within A are 100, all distances within B are 10, all distances between A and B are 50, and so on. Can you find all clusterings of S that satisfy strict ordering?

Single linkage works well in this setting, in the sense that the desired clustering is guaranteed to correspond to some pruning of the tree.

Claim 5. *If the clustering $C_1^*, \ldots, C_k^*$ satisfies strict ordering, then it corresponds to some pruning of the single linkage tree.*

Proof. We just need to show that each C_i^* is a subtree of the final single linkage tree.

Suppose that at some stage during the linkage process, the current set of clusters includes $C, C' \subset C_i^*$ and $C'' \subset S \setminus C_i^*$. We'll show that C will always be merged with C' in preference to C''; applying this result repeatedly, we can conclude that there must be a time at which C_i^* is one of the current clusters.



Let the closest distance between C and C'' be from $x \in C$ to $z \in C''$. By strict ordering, we know that $d(x, y) < d(x, z)$ for any $y \in C'$. Therefore, C will be merged with C' in preference to C''. □

The *complete linkage* tree also has a pruning corresponding to $\{C_1^*, \ldots, C_k^*\}$, as does the variant of *average linkage* in which the distance between two current clusters C and C' is taken to $d_{\text{avg}}(C, C')$. In either case, the argument is much the same as for single linkage.

Interestingly, the other two variants of average linkage fail! Let's consider the one in which the distance between clusters C and C' is defined as $\|\text{mean}(C) - \text{mean}(C')\|^2$. Here's a data set S consisting of just four points in $\mathbb{R}^3$:

$$A_1 = (0, 0, 1),\ A_2 = (0, 0, -1),\ B_1 = (c, 1, 0),\ B_2 = (c, -1, 0),$$

where c is some constant with $\sqrt{2} < c < \sqrt{3}$. The clustering $(\{A_1, A_2\}, \{B_1, B_2\})$ satisfies strict ordering. However, average linkage will start by merging A_1 and A_2 (say), and will then merge $\{A_1, A_2\}$ with one of the B_i's instead of merging B_1 with B_2!

Ward's method of average linkage, which defines the distance between clusters C and C' to be $(|C| \cdot |C'|/(|C| + |C'|))\|\text{mean}(C) - \text{mean}(C')\|^2$, runs into similar problems. Modify the example above to include a fifth point $B_3 = B_1$. Ward will first merge B_1 and B_3; then A_1 with A_2; and then $\{A_1, A_2\}$ with B_2!

Notion 2: AVERAGE ATTRACTION

Strict ordering is rather a strong property to demand of a clustering. A weaker notion is γ-*average attraction*, which says simply that

For all $x \in S$, and any true cluster $C' \neq C^*(x)$, we have $d_{\text{avg}}(x, C^*(x)) \leq d_{\text{avg}}(x, C') - \gamma$.

However, it turns out that a data set can have many different clusterings that satisfy this property, that are not necessarily nestable in a hierarchical manner.

Claim 6. *Pick any $\gamma > 0$. There exist data sets for which the set of clusterings satisfying γ-average attraction cannot be resolved into a hierarchy.*

Proof. Consider a data set S divided into subsets $S_1, S_2, \ldots, S_{1/\gamma}$, such that

$$d(x, y) = \begin{cases} 0 & \text{within } S_i \\ 1 & \text{between } S_i \text{ and } S_j, j \neq i \end{cases}$$

Consider *any* clustering that results from selected merges of these regions. If $C(x)$ denotes the cluster containing point x, we have that

$$d_{\text{avg}}(x, C(x)) = \frac{|C(x)| - \gamma|S|}{|C(x)|} \leq 1 - \gamma$$

whereas for any other cluster $C' \neq C(x)$, we have $d_{\text{avg}}(x, C') = 1$. Thus the clustering satisfies average attraction.

But no tree can contain all clusterings obtainable by merging combinations of the S_i. □

Notion 3: STRONG STABILITY

Clustering $\{C_1^*, \ldots, C_k^*\}$ satisfies *strong stability* if

For any true clusters $C \neq C'$ and subsets $A \subset C, A' \subset C'$, we have $d_{\text{avg}}(A, C \setminus A) < d_{\text{avg}}(A, A')$.

Strict ordering implies strong stability, and therefore we can already rule out the two variants of average linkage that use cluster means. Interestingly, in this case neither single linkage nor complete linkage works either (can you find such examples?), but the remaining form of average linkage does.

Problem 2. Run experiments on "real" data to determine the extent to which the desired clusters satisfy the above three properties.

Problem 3. What are other natural clustering properties?