The shape of a data set $S \subseteq \mathbb{R}^n$ can be analysed using cluster analysis.

Compute

$$c = \frac{1}{|S|} \sum_{v \in S} v \qquad \text{(Centre)}$$

and for $r > 0$

$$S_r = \{ v \in S : \|v - c\| \geq r \}$$

and for $\varepsilon > 0$ the graph

$G_\varepsilon(S_r)$: vertices $= S_r$

edge $u \text{ —— } v$ if $\|u - v\| \leq \varepsilon$.

For fixed $\varepsilon > 0$ and for

$r_1 > r_2 > r_3 > \dots$

we get a chain of inclusions of graphs

$$G_\varepsilon(S_{r_1}) \subseteq G_\varepsilon(S_{r_2}) \subseteq G_\varepsilon(S_{r_3}) \subseteq \dots$$

For ε in a "stable range" the resulting barcodes for connected components may be informative.

Hierarchical Clustering Algorithms

Given n items to be clustered, and an $n \times n$ distance matrix, do:

1. Start with n clusters, each containing 1 item. Let the distances between clusters equal to the distances between items.
2. Find the closest pair of clusters and merge them, so that you have one fewer clusters.
3. Compute "distances" between the new cluster and each of the remaining old clusters.
4. Repeat 2 and 3 until there is just a single cluster with n items.
5. Return the output as a dendrogram or barcode.

There are various ways to define distance $d(X,Y)$ between clusters X, Y in Step 3.

Single-linkage: $d(X,Y) = \min_{\substack{x \in X \\ y \in Y}} d(x,y)$

Complete-linkage: $d(X,Y) = \max_{\substack{x \in X \\ y \in Y}} d(x,y)$

Average-linkage:

$$d(X,Y) = \frac{1}{|X||Y|} \sum_{\substack{x \in X \\ y \in Y}} d(x,y)$$

Algorithm for single linkage

① Set $m := 0$

$L[0] = 0$

(we'll say the level is $L[m]$ at stage m)

$D_{(0)} = (d_0(i,j))$ The initial $n \times n$ matrix of distances

② while $m < n$ do:

③ find $1 \le t, s \le n-m$ such that

$d_m(t,s) = \min_{1 \le i,j \le n-m} d_m(i,j)$.

Set $L[m+1] = d_m(t,s)$

④ Let $D_{(m+1)} = (d_{m+1}(i,j))$ be the $(n-m-1) \times (n-m-1)$ matrix obtained $D_{(m)}$ by

- delete rows s, row t, columns, column t
- re-index so that old row i' / column i' becomes the new row i / column i
- add a new final row and column with

$d_{m+1}(n-m-1, j) = d_{m+1}(j, n-m-1)$

$= \min(d_m(s,j'), d_m(t,j'))$.

end do

⑤ Return the list $L[0], L[1], \cdots, L[n-1]$ as a barcode.

Example

$D_{(0)} =$

	1	2	3	4	5
1	0	11	10	14	22
2	11	0	3	13	21
3	16	3	0	12	20
4	14	13	12	0	16
5	22	21	20	16	0

$m = 0$

$L = [0]$

---

merge 2 & 3

$m = 1$

$L = [0, 3]$

	1	4	5	6
1	0	14	22	10
4	14	0	16	12
5	22	16	0	20
6	10	12	20	0

---

merge 1 & 6

$m = 2$

$L = [0, 3, 10]$

	4	5	7
4	0	16	12
5	16	0	20
7	12	20	0

---

merge 4 & 7

	5	3
5	0	16
3	16	0

n = 3

L = [0, 3, 10, 12]

---

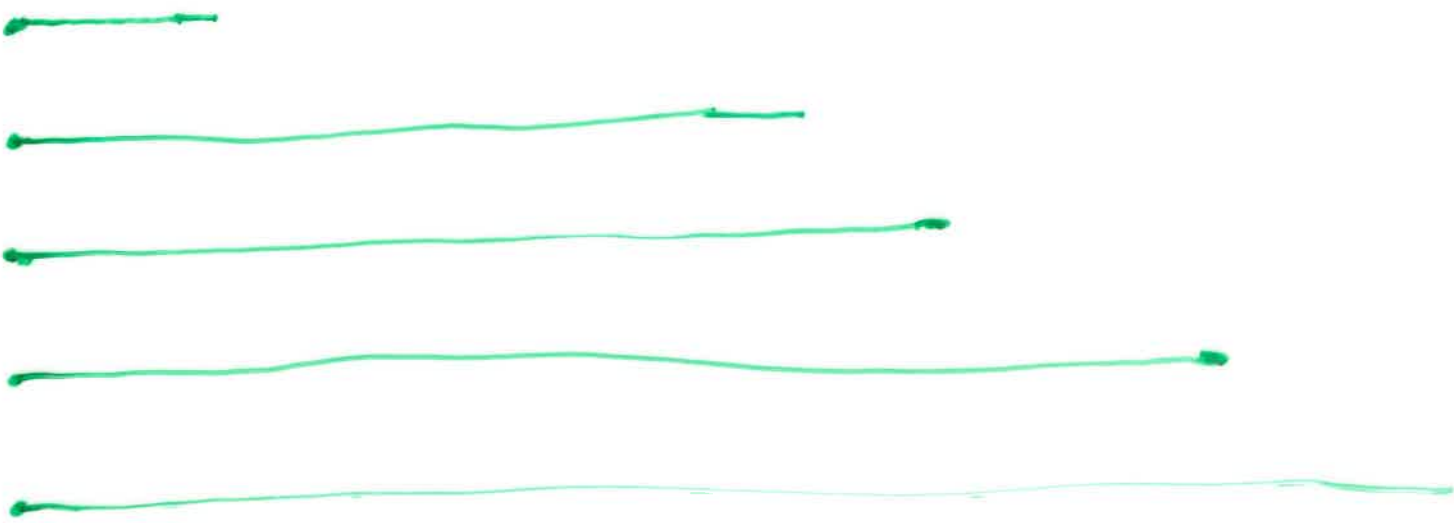
merge 5 & 8

	9
9	0

n = 4

L = [0, 3, 10, 12, 16]

Return the bar code



0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18