HIERARCHICAL CLUSTERING

Suppose given n observations $X_1, \ldots, X_n$ with values in some $\mathbb{R}^d$. They can be represented by an $n \times d$ matrix. On $\mathbb{R}^d$ we have the usual Euclidean metric. For simplicity, let's assume for now that all the observations are distinct (no ties). Later, adaptations in case there are ties will be mentioned. Let S be the set $\{X_1, ..., X_n\}$.

Clustering can be done in two directions. One is called "agglomerative," where we start with the n singletons $\{X_j\}$ and step by step take unions of clusters. The other is called "divisive," where we first may decompose S two clusters, and then possibly subdivide each cluster into two, and so on. This handout is mainly about agglomerative clustering, but here is one divisive method.

1. A method of divisive clustering when $d = 1$

Apply the dip test to the data vector x (which may contain tied observations). If unimodality is not rejected, then decide there is just one cluster and don't subdivide into further clusters. If unimodality is rejected for x, then estimate a point c from the adapted R software "diprad" ("decomposition point" in its output) at which to subdivide the data into two parts, xlow consisting of $X_i < c$ found by the R code cutlow(x,c) and xhigh consisting of $X_i \geq c$ found by cuthigh(x,c). Then do the dip test on xlow and xhigh, stopping on any branch where unimodality is not rejected but continuing if it is rejected. This method is hierarchical in the sense that if at different stages we have m and m' clusters with $m < m'$, each of the m clusters is a union of some of the m' clusters.

2. Agglomerative hierarchical clustering

We begin with the n singletons $\{X_j\}$ as clusters, then proceed iteratively to join clusters, at each stage taking the union of the two clusters that are closest according to some measure of distance. We will then have decomposed the set C into m clusters for each $m = 1, 2, \ldots, n$. For each $m = 2, \ldots, n-1$, each of the m clusters is a union of some of the $m+1$ clusters, so $m-1$ of those clusters are kept the same and one takes the union of two of the $m+1$ clusters to get one of the m clusters. Any such method is also hierarchical. What values of m are most useful and interesting will depend on the data set.

2.1. **Distances between clusters.** At each stage, one takes the union of the two clusters A, B that are closest according to some measure of distance $D(A, B)$ (which is not usually a metric). For singletons we will have $D(\{x\}, \{y\}) = d(x, y)$ for all methods. (This also holds if there are tied observations, so that $n_{\{u\}} > 1$ for some $u = x$ or y or both. The first stage of clustering by any method is to observe such tied observations and assign the correct $n_{\{u\}}$ to each $u = X_j$ for any $j = 1, ..., n$.)

The agglomerative hierarchical clustering function "hclust" in R provides seven options called "average," "complete," "ward," "single," "mcquitty," "median," and "centroid." These are not very explicitly described in the system documentation, but they are by Murtagh (1983) except for "mcquitty," which is given in other sources. According

to a comment at the beginning of the R source code for hclust, Murtagh in 1992 was the original author of the code.

In 2003, bugs were reported in the code for the "median" and "centroid" methods, which are said to have been fixed later in 2003, but in July 2012, a new bug report was made for the centroid method. See the Appendix. So, it seems best not to use these methods until the bugs are fixed.

One distance between clusters is

$$D_{\min}(A,B) := \min\{d(x,y) : \ x \in A, \ y \in B\}.$$

Clustering by this distance is implemented by the "single" option. Note that $D_{\min}$ does not satisfy the triangle inequality (it is not a metric in the mathematical sense) because we can have $D_{\min}(A,D) > D_{\min}(A,B) + D_{\min}(B,D)$. Use of this can produce long, stringy clusters where points are like beads close to their neighbors on a string, but on a long string, which may not be very far from a different long string. If $d = 1$ all clusters will be 1-dimensional by whatever method, so this behavior of $D_{\min}$ (single) clustering is not particularly a drawback.

Another distance between clusters is defined by

$$D_{\max}(A,B) := \max\{d(x,y) : \ x \in A, \ y \in B\},$$

which is implemented by the "complete" option. Here $D_{\max}$ is also not a metric, in fact $D_{\max}(A,A) > 0$ if A contains more than one point. Clustering by joining the two clusters with smallest $D_{\max}$ produces more compact clusters than does clustering with $D_{\min}$.

A third distance between clusters is the "average" distance defined by

$$D_{\text{ave}}(A,B) = \frac{1}{n_A n_B} \sum_{i,j: \ X_i \in A, X_j \in B} d(X_i, X_j). \tag{1}$$

This is implemented by the "average" option in hclust.

The "centroid" of a cluster A is defined as the sample mean $\overline{X}_A = \sum_{i: \ X_i \in A} X_i / n_A$. The centroid distance is defined by

$$D_{\text{cent}}(A,B) := \|\overline{X}_A - \overline{X}_B\|^2 = d(\overline{X}_A, \overline{X}_B)^2.$$

At this writing (Sept. 29, 2012) the "centroid" method in hclust does not actually implement this (correctly), see the Appendix.

In dimension $d = 1$, for any method of clustering being considered, each cluster will consist of some consecutive order statistics $X_{(j)}, X_{(j+1)}, \ldots, X_{(k)}$. For sets A and B in the line let $A \prec B$ mean that $x < y$ for all $x \in A$ and $y \in B$. Suppose we have disjoint clusters A and B (e.g. any two of the m clusters for a given m). Then $A \prec B$ or $B \prec A$, so suppose $A \prec B$. Then for each $x \in A$ and $y \in B$, $d(x,y) = y - x$, and it's easily seen that (1) reduces to $\overline{X}_B - \overline{X}_A$, the distance between the two cluster sample means. Thus in one dimension (although not in higher dimensions), the distance between centroids is the same as the average distance. But G. Chan in 2007 found differing results of "average" and "centroid" for the 1-dimensional "galaxies" data set ($n = 83$, with 5607 adjoined), further evidence that either hclust may have computed the centroid results wrongly, or it may compute something different.

In dimension 2, consider the 5 points $(-21,-10)$, $(-21,10)$, $(0,0)$, $(22,-1)$, $(22,1)$. For agglomerative hierarchical clustering, by any of the four methods we've considered, one would first join the 4th and 5th points, then the first and second. We then have three clusters, with respective sample means $(-21,0)$, $(0,0)$, and $(22,0)$. The two whose sample means are closest are the first and second. But, to minimize (1), we would join the second and third clusters because $\sqrt{22^2+1} = 22.0227 < \sqrt{21^2+100} = 23.2594$. Thus for dimension $d \geq 2$, minimizing (1) and merging the two clusters whose sample means are closest are not the same.

A further distance measure closely related to the centroid one, is Ward's distance, which satisfies

$$D_{\text{Ward}}(A,B) = \frac{n_A n_B}{n_A + n_B} \|\overline{X}_A - \overline{X}_B\|^2.$$

2.2. **Updating distances.** Given m clusters for $m \geq 3$, we take the union of two of them, say A and B, to get $m-1$ clusters. Before doing the next step we need to update the distances, namely, to find the distance $D(C, A \cup B)$ for each of the $m-2$ clusters C disjoint from A and B. One will need to do this for $m-2 = n-2, n-3, ..., 1$, so one will need to do $\sum_{j=1}^{n-2} j = \frac{1}{2}(n-1)(n-2) = O(n^2)$ updates. To keep computation to a minimum, it will be good if each update can be done by a simple formula based on the already known cluster distances for the m clusters (stored in memory) rather than computing $D(C, A \cup B)$ from scratch from its definition. Updating formulas exist for the five distances mentioned so far. They are instances of "Lance–Williams" formulas given in general by Lance and Williams (1967).

For D_{min}) and D_{max} we have

$$D_{\text{min}}(C, A \cup B) = \min(D_{\text{min}}(C,A), D_{\text{min}}(C,B),$$

$$D_{\text{max}}(C, A \cup B) = \max(D_{\text{max}}(C,A), D_{\text{max}}(C,B).$$

For D_{ave} we have

$$D_{\text{ave}}(C, A \cup B) = \frac{n_A D_{\text{ave}}(C,A) + n_B D_{\text{ave}}(C,B)}{n_A + n_B}. \tag{2}$$

For the centroid distance D_{cent} we have

$$D_{\text{cent}}(C, A \cup B) = \frac{n_A D_{\text{cent}}(C,A) + n_B D_{\text{cent}}(C,B)}{n_A + n_B} - \frac{n_A n_B}{(n_A + n_B)^2} D_{\text{cent}}(A,B),$$

where the first term is of the same form as for D_{ave} but the second is different. For the Ward distance we have

$$D_{\text{Ward}}(C, A \cup B) =$$

$$\frac{(n_A + n_C) D_{\text{Ward}}(C,A) + (n_B + n_C) D_{\text{Ward}}(C,B) - (n_A + n_B) D_{\text{Ward}}(A,B)}{n_A + n_B + n_C}.$$

So the first five distances do have relatively simple update formulas.

2.3. **The median and McQuitty distances.** By analogy with the average distance, one might consider defining the "median" distance between clusters A and B by

$$D_{\text{med}}(A, B) = \text{median}\{d(X_i, X_j) : \ X_i \in A, \ X_j \in B\},$$

the sample median of $n_A n_B$ distances. This appears however not to have a convenient update formula. Consider the following

Example. Let $0 < r < s < t < u < v < w$ where $s - r$ and $t - s$ are small, $u - t$, $v - u$, and $w - v$ are large, and r is still larger. Consider the clusters $C := \{0\}$, $A := \{r, s, t\}$, and $B := \{u, v, w\}$. Then clearly $D_{\text{med}}(C, A) = s$ and $D_{\text{med}}(C, B) = v$. One can check that $D_{\text{med}}(A, B) = v - s$. Clearly, $D_{\text{med}}(C, A \cup B) = (t + u)/2$, which is not a function of v and s. (Here $n_A = n_B = 3$ and $n_C = 1$ are fixed.)

Murtagh (1983) gives, and so very possibly also hclust uses for its "median" option, an update formula related to a "Gower median" (Gower, 1967) for a distance I'll call D_{Gmed}, namely

$$D_{\text{Gmed}}(A, B \cup C) = \frac{1}{2}\left(D_{\text{Gmed}}(C, A) + D_{\text{Gmed}}(C, B)\right) - \frac{1}{4} D_{\text{Gmed}}(A, B).$$

That is certainly a simple formula. The starting formula $D_{\text{Gmed}}(\{x\}, \{y\}) \equiv d(x, y)$ and the update formula completely determine D_{Gmed}. Since it seems to have no close relation with actual medians, it might be better to call the distance the "Gower" distance?

Murtagh (1983) does not mention "McQuitty" as far as I saw. From other sources, it seems that the McQuitty (1966) distance has the even simpler update formula

$$D_{\text{McQ}}(A, B \cup C) = \frac{1}{2}\left(D_{\text{McQ}}(C, A) + D_{\text{McQ}}(C, B)\right).$$

Since weighting clusters by their number of members, as in the update formula (2) for D_{ave}, seems (to me) natural, the advantage of the Gower and McQuitty distances seems to be essentially their easier computation (not having to recall or use numbers n_A, n_B, n_C) for updating, which could be important for possibly large numbers of possibly large data sets. One might use these methods in such a situation in an exploratory way. Then if some of the data sets seemed to have an interesting clustering structure, one could cluster them with another distance such as D_{ave}.

2.4. **A "least squares" rationale for the Ward distances.** The Ward distance resulted from the following considerations. Suppose that the ith cluster A_i contains n_i observations $X_{i1}, ..., X_{in_i}$, which are vectors in a Euclidean space. Let $\overline{X}_i$ be its sample mean, i.e. centroid. One criterion is to try to find m clusters so as to minimize

$$SS_m \ := \ \sum_{i=1}^{m} \sum_{j=1}^{n_i} |X_{ij} - \overline{X}_i|^2.$$

Thus in joining two clusters, we will get $SS_{m-1} > SS_m$ and we'd like to make the difference as small as possible. If we join the ith and kth clusters, the sample mean of the joined cluster will be $\overline{X}_{i,k} := (n_i \overline{X}_i + n_k \overline{X}_k)/(n_i + n_k)$ and it will be shown (by ANOVA-like algebra) that

$$SS_{m-1} - SS_m \ = \ n_i |\overline{X}_i - \overline{X}_{i,k}|^2 + n_k |\overline{X}_k - \overline{X}_{i,k}|^2 = \frac{n_i n_k}{n_i + n_k} |\overline{X}_i - \overline{X}_k|^2. \tag{3}$$

Thus we want to choose $i \neq k$ to minimize the right side, which was already defined above as the Ward distance D_{Ward} between the ith and kth clusters.

To prove (3), we have

$$SS_{m-1} - S_m = \sum_{j=1}^{n_i} |X_{ij} - \overline{X}_{i,k}|^2 - |X_{ij} - \overline{X}_i|^2 + \sum_{j=1}^{n_k} |X_{kj} - \overline{X}_{i,k}|^2 - |X_{kj} - \overline{X}_k|^2$$

$$= n_i(|\overline{X}_{i,k}|^2 - |\overline{X}_i|^2) - 2\sum_{j=1}^{n_i} X_{ij} \cdot (\overline{X}_{i,k} - \overline{X}_i)$$

$$+n_k(|\overline{X}_{i,k}|^2 - |\overline{X}_k|^2) - 2\sum_{j=1}^{n_k} X_{kj} \cdot (\overline{X}_{i,k} - \overline{X}_k)$$

$$= (n_i + n_k)|\overline{X}_{i,k}|^2 - n_i|\overline{X}_i|^2 - n_k|\overline{X}_k|^2 - 2n_i\overline{X}_i \cdot \overline{X}_{i,k} + 2n_i|\overline{X}_i|^2 - 2n_k\overline{X}_k \cdot \overline{X}_{i,k} + 2n_k|\overline{X}_k|^2$$

$$= n_i|\overline{X}_i|^2 + n_k|\overline{X}_k|^2 - (n_i + n_k)||\overline{X}_{i,k}|^2.$$

The middle expression in (3) is easily seen to reduce to the same. The last equation in (3) follows from $\overline{X}_{i,k} = (n_i\overline{X}_i + n_k\overline{X}_k)/(n_i + n_k)$ and simple algebra, so (3) is proved.

2.5. **The one-dimensional case.** In dimension $d = 1$, by any of the methods, each cluster will consist of some consecutive order statistics $X_{(j)}, X_{(j+1)}, \ldots, X_{(k)}$ for some $j \leq k$. For the "single" ($D_{\min}$) method, the divisions between clusters are done in order of the size of the spacings $s_j = X_{(j)} - X_{(j-1)}$ for $j = 2, \ldots, n$. Thus for $m = 2$, if s_j is the largest spacing, the two clusters will be $\{X_{(i)} : \ i < j\}$ and $\{X_{(i)} : \ i \geq j\}$. Then for $m = 3$, one of these two clusters will be decomposed at the next-largest spacing, and so on. For a data set of 83 galaxies' redshifts, namely the 82 observations in "galaxies" from MASS with 5607 adjoined, the results hclust gave by all its 7 methods for $m = 2, \ldots, 7$ found by Gabriel Chan in 2007 have been distributed. (Correcting 26690 to 26960 seemingly would have little effect on the clustering, as the observation would keep its rank relative to the others, and the spacing just below it would remain the 6th largest spacing.) Notably, for $m = 3$, five of the 7 hclust methods gave $8 + 72 + 3$ galaxies, which corresponds reasonably to an astronomical situation with 72 galaxies in the main supercluster, 8 in the foreground, and 3 (intrinsically bright) galaxies in the background. But the "Ward" method makes the split $8 + 38 + 37$. It seems to me rather undesirable to split the main cluster in that way. The McQuitty method gives $8 + 63 + 12$. Looking at the data in the source Postman, Huchra and Geller (1986), and asking about the $9 = 12 - 3$ galaxies which the McQuitty method classifies as in the background but five other agreeing methods do not, the supercluster includes several clusters of galaxies. One of them, Abell 2061 (northern part) has in the same direction two of the nine with velocities 24366 and 26960, but not among the nine, velocities 23263, 23484, 23542, and 23706. It seems arbitrary and unphysical to separate these from 24366 which is also close in direction. The case of 26960 is less clear. In Abell 2061 (southern part), are among the nine, velocities 24285, 24289, and 24717, while in the same sky region, not among the nine, are velocities 23206, 23263, 23538, 23666, and 23711. Again the separation of these, at such a late stage of the McQuitty clustering of the data into 3 or even 2 clusters, seems to violate the unity of the astronomers' Abell 2061 cluster. A more interesting case is Abell 2079, in which region are two of the 9 with velocities 24990 and 25633,

whereas galaxies in the physical cluster seem to have velocities no more than about 22,250. Thus the two might be said to be somewhat in the background of Abell 2079 which they are behind, but they are not in the background of the supercluster when we consider the other cluster A2061. So, the five methods which agree on $8 + 72 + 3$ seem to have correctly identified the 3 galaxies truly in the background of the supercluster.

For dimension $d \geq 2$ the situation seems to be much more complex. There is no sorting of the data as useful as the one in one dimension. One cannot so easily identify candidates $\{X_i\}$ to be clusters for $m = 2$.

3. Details of using the R command hclust

(Cf. Venables and Ripley, p. 217, ignoring "S" lines.) Suppose given a data set, say x, of n points in d-dimensional space, so that x is given by an $n \times d$ matrix. (If $d = 1$ it reduces to a vector.)

The function "dist(x)" will evaluate all the $n(n-1)/2$ distances between pairs of members of x. The object $y = dist(x)$, if displayed on the screen (it does not need to be in general), will be in an array form, with rows of different lengths, column names $1, ..., n-1$, and row names $2, ..., n$. Here y is a vector in R, so that y[j] is defined for $j = 1, ..., n(n-1)/2$ but "y[i,j]" gives an error message for any i and j. If one is going to be using the distances with several methods, one might give dist(x) some name so that it doesn't have to be computed over and over, a concern if n is large. Give a command such as cmx = hclust(dist(x),method = "...") where you insert whichever of the seven available methods you want to use. (Here "cmx" is used the same way as Venables and Ripley use "h." If you want you could replace "m" in "cmx" by a mnemonic for method.)

Then, give the command

plclust(cmx);

then

cutree(cmx,m)

will describe the clustering into m clusters by giving a list of integers from 1 to m. There will be as many occurrences (repetitions) of "j" as there are members of the jth cluster. For 1-dimensional data, if they are first sorted into order statistics, the occurrences of j will be consecutive. Then it's easy to read off how many members each of the smaller clusters has. If there is only one large cluster, one can find how many members it has as n minus the sum of numbers of members in the other clusters.

After giving the "plclust(...)" command, a dendrogram may appear in a window. It can (at least on the math department system) be printed by the command dev.print().

4. Appendix: Bugs in "centroid"

The following are included in an online file called "Bug 4195 – hclust: median, centroid." Peter Kleiweg (September 2003) reported a bug in hclust for the clustering methods "median" and "centroid." Brian Ripley (co-author of Venables and Ripley) in November 2003 said he had fixed the bug. However M. Maechler in April 2012 wrote that "before the fix (in 2003), hclust() was very fast for large n, but it no longer was after the fix." Instead he said he would write another hclust.f [Fortran program] which

would fix the bug and still be fast. Daniel Müllner (2012) wrote a package of programs meant to replace hclust and said to be fast.

Notably, Murtagh (1983, end of §3.2) said that to get fast computation one may have only "approximate" centroid and median algorithms.

Meanwhile, in another file called "Bug 14977 – Problem with centroid method in hclust function," Mateus Teixeira in July 2012 gave an example in one dimension of a data set equivalent by translation to, after sorting, $(-.7, -.1, .6, 1.1, 1.8, 2.5)$. He found that the merges of clusters were all done in the correct order, but that the distances between clusters, other than singletons, were not found correctly. I confirmed this in the version of R on the math department system. In the same file, Jean V. Adams, also in July 2012, confirmed the error and Teixeira's conjecture that the centroids were not being computed correctly. For a cluster containing two numbers a and b, whose centroid should be $(a + b)/2 = .5a + .5b$, hclust actually took $(a + 3b)/4 = .25a + .75b$ instead. So it would not be at all surprising if in other data sets, in one dimension, such as "galaxies" and small modifications of it, hclust with the "centroid" method would assemble clusters the wrong way. Since in dimension 1, the "average" method should be equivalent, one could just use that instead of "centroid." In higher dimensions, it seems one has to wait for the bug(s) to be fixed before using "centroid."

REFERENCES

*Gower, J. C. (1967). A comparison of some methods of cluster analysis. *Biometrics* **23**, 623–637.

Lance, G. N., and Williams, W. T. (1967). A general theory of classificatory sorting strategies. 1. Hierarchical systems. *The Computer Journal* **9**, 373–380.

*McQuitty, L. L. (1966). Single and multiple hierarchical classification by reciprocal pairs and rank order types. *Educational and Psychological Measurement* **26**, 253–.

Müllner, D. (2012). Package 'fastcluster.'
http://math.stanford.edu/∼muellner/fastcluster.html
or
http://cran.r-project.org/web/packages/fastcluster/fastcluster.pdf
Version 1.1.7, dated September 21, 2012.

Murtagh, F. (1983). A survey of recent advances in hierarchical clustering algorithms. *The Computer Journal* **26**, 354–359.

Postman, M., Huchra, J. P., and Geller, M. J. (1986). Probes of large-scale structure in the Corona Borealis Region. *Astronom. J.* **92**, 1238–1247.

* – I have not seen these items in the original