CHAPTER 5 - NEW CLUSTERING PROBLEMS

5.1 INTRODUCTION

Classification is such an all-embracing human activity that practically no area of automated data processing can dispense with some form of clustering. Two relatively new areas, which use algorithms adapted from those explored in previous chapters, are studied in sections 5.2 and 5.3.

A clear example of the contiguity-constrained clustering problem (section 5.2) is the grouping of people/areas on the basis of some given set of socio-economic attributes. It might be expected that the objects of analysis which come from major urban areas would be grouped together. Consider now the presence of a contiguity-constraint: the resulting clustering ought to clearly demarcate the urban areas, and instead group with them their respective hinterlands. A good background study in this area is Fischer (1980). Gordon (1980) should be consulted for another perspective on the problem of contiguity constraints in clustering.

The problem of clustering interaction data (section 5.3) is that of handling asymmetric proximities. Slater (1981) or Masser and Scheurwater (1980) provide illustrative studies.

5.2 CONTIGUITY-CONSTRAINED CLUSTERING

One major theme in clustering research over the past two decades has been the automatic classification of quantitatively described objects, without any constraint as to which pairs of such objects might ultimately find themselves in the same class. A second recent trend in clustering work has been where there is such an inherent or an imposed representational constraint. In this section we review general-purpose algorithms which have the function of segmenting (or regionalizing, or zoning) a set of objects, each described by a descriptor vector.

Contiguity-constrained clustering uses proximities between objects, defined in descriptor space, and also takes into account contiguous neighbourhoods. Depending on the application, the contiguous neighbourhood is defined in different ways. In image processing, where the image consists of pixels characterized by grey-level intensity values, the 8 neighbouring pixels (east, north-east, north, etc.) are suitable candidates. Similarly with agricultural data, the terrain which is characterised by crop yields or chemical constituents may be subdivided into square parcels and the neighbourhood of a parcel may be defined as its 8 adjacent parcels. With point patterns, a radius may be used to define the neighbourhood of point i: $N(i) = \{j \mid d_{ij} \leqslant r\}$, and j is said to be contiguous to i; a neighbourhood may alternatively be defined as the k nearest neighbours of an object (see section 4.6 of Chapter 4). In general, when the objects do not comprise the squares of a regular grid, it is convenient to express the contiguity relationship as a binary matrix, with a contiguity value $c_{ij} \in \{0,1\}$ defined on all pairs of objects. Such a matrix can be externally defined by the user, - for example, in the case of contiguities between bordering countries (characterised, perhaps, by socio-economic attributes) or other basic spatial units.

If the stepwise agglomerations in hierarchical clustering are constrained to be between clusters which are contiguous, the problem of inversions (reversals or non-monotonic increase/decrease in cluster criterion value) is likely. This is when $d(q \cup r, s) \not\geq d(q,r)$ for three clusters q,r, and s, where q and r agglomerate to form $q \cup r$, and where the cluster criterion value (e.g. compactness or connectivity) is related to the dissimilarity d between clusters (cf. section 3.2 of Chapter 3). In using the common clustering criteria (e.g. as listed in Table 1, section 3.3), with the restriction that only contiguous clusters can merge, inversions tend to arise since a previously forbidden merger between two very similar classes may be permitted by changes in the contiguity relation. The presence of inversions in a hierarchy is disadvantageous: it makes difficult the interpretation of partitions, and the definition of dissimilarity between classes. Only two of the traditional hierarchical clustering methods appear to be amendable in order to permit agglomerations between contiguous clusters, and simultaneously guarantee that no inversions will arise. These methods are the single and complete linkage methods, which will use two different approaches to the updating of the contiguity relation following each agglomeration.

The contiguity-constrained single linkage method is as follows: at each agglomeration, fuse together the two clusters of least interconnecting dissimilarity, such that this dissimilarity is between a pair of contiguous objects. Initially all clusters are singletons. Each agglomeration in this method is necessarily between a pair of contiguous objects. Therefore, given the contiguity graph where each edge connecting a pair of contiguous objects is weighted by the dissimilarity (in descriptor space) between the objects, it is seen that the minimal spanning tree of the weighted contiguity graph may be obtained and subsequently transformed into the single linkage hierarchy. A simple proof that the

contiguity-constrained single linkage hierarchy cannot present inversions is to replace the dissimilarities between all pairs of non-contiguous objects by some arbitrarily large value. The construction of the single linkage hierarchy on this amended set of dissimilarities is well-defined (in the sense that at all stages the traditional algorithm can be employed and, assuming the contiguity graph is connected, infinite dissimilarities will never be used as cluster criterion - i.e. connectivity-values). As in the case of the usual single linkage method, it has been found that this method has a pronounced tendency to "chain", i.e. to successively agglomerate singletons to one, large cluster in each partition (see Fischer, 1980). Efficient algorithms for constructing a constrained single linkage hierarchy have been examined in section 4.5 of Chapter 4.

An alternative approach for contiguity-based agglomerative clustering allows agglomeration of any pair of clusters such that there exists a contiguity link between at least one member of each of the clusters. Such a definition of contiguity has generally been used in incorporating a contiguity constraint in the minimum variance (Ward's) method. However no way of using this method, in an inversion-free manner, has yet been found. For a review of work in this direction, see Murtagh (1984). Of the major hierarchical methods, only the complete link method excludes the possibility of inversions when constrained in this manner. Before showing this, it may be remarked that the $O(n^2)$ time and $O(n^2)$ space algorithm of section 4.4 (algorithm E) is easily updated to include an additional testing of contiguity whenever a linkage in the NN-chain is created.

Proposition: The complete link method, with the constraint that at least one member of each of the two clusters to be agglomerated be contiguous, is guaranteed not to give rise to inversions.

The proof of this proposition will conclude this section.

Consider three clusters (possibly singletons), q, r, and s. At some stage of the agglomerative construction of the hierarchy, q and r cluster (supposition I); and later, s clusters with q U r directly and not through the intermediary of some cluster, t (supposition II). It is seen that no loss of generality ensues with supposition II, since inversion-free agglomeration of q U r and s implies inversion-free agglomeration of q U r and t, followed by inversion-free agglomeration of q U r U t and s. Also, without loss of generality, assume that $d(q,s) \leqslant d(r,s)$. Three cases may now be considered.

Case I : $d(q,s) \leqslant d(r,s) \leqslant d(q,r)$.
If either (q,s) or (r,s) are contiguous, they should have clustered prior to (q,r): this is contrary to supposition I. If neither (q,s) nor (r,s) are contiguous, then s cannot cluster with q U r, except through both s and q U r being contiguous with some other cluster, t: this is contrary to supposition II.

Case II: $d(q,r) \leqslant d(q,s) \leqslant d(r,s)$.
(q,r) must be contiguous so that they may cluster as supposed. If neither (q,s) nor (r,s) are contiguous, supposition II is not possible. If either are contiguous, then s can cluster with q U r without giving rise to an inversion.

Case III: $d(q,s) \leqslant d(q,r) \leqslant d(r,s)$.
For q and r to cluster before q and s, it must be assumed that q and s are not contigous; supposition I implies that (q,r) is contiguous; and supposition II implies that (r,s) is contiguous.

We may summarize: case I cannot arise; case II presents no problem; and case III is the only case to possibly give rise to an inversion. An

inversion arises in case III if $d(q,r) \not\leq d(q \cup r, s)$. Traditional clustering strategies (cf. Table 1 of section 3.3) define $d(q \cup r, s)$ as a function of $d(q,r)$, $d(q,s)$ and $d(r,s)$. Therefore, choosing $d(q \cup r, s)$ as $\max \{ d(q,s), d(r,s) \}$ precludes an inversion in case III.

5.3 CLUSTERING OF INTERACTION DATA

Interaction flow matrices are square, asymmetric matrices which arise in many of the social sciences. Examples of the flows or interactions involved in such tables are: industrial inputs and outputs for a set of firms or countries; cross-citations for a set of journal articles; internal migrations or trips for a set of geographic regions; or occupational mobility data for a set of occupations of a given population for a set time-period. In some of these applications contiguous clusters may be required, especially in the case of data with geographic location information. Two types of clustering problem may be considered. Consider the case of journey-to-work data, with a given set of zones and associated numbers of cross-boundary journeys. We may wish to ascertain nodal or "central" zones (i.e. those that receive large numbers of workers), or alternatively to carry out a regionalization of the given zones into a smaller set of homogeneous areas. For these two different problems, two approaches have been suggested. A variant of the single linkage method has been proposed for the former problem, - the determining of nodal zones. Faithful representation of the asymmetric character of the interaction matrix is the primary objective, and one disadvantage of this approach is the "chaining" side-effect of single linkage clustering. For the second problem, - creating homogeneous zones - variants of the average linkage method have been used. A disadvantage here is the conflict between the clusters of zones and the often asymmetric characteristics of these zones (i.e. in-flows greater than out-flows or viceversa).

In both cases a standardization of the given flows is carried out, in order to adjust for disproportionate flow in large zones. In the case of the compact clustering, this has been achieved by dividing every element of the flow array by the corresponding row and column sums

(see below). In the case of directed single linkage, analogous frequencies have been obtained. Since the initial, asymmetric data may be standardized directly, an iterated (re) calculation of row (column) sums is carried out until the row (and column) sums are all identical.

The directed linkage procedure involves a generalization of the single linkage method for dealing with asymmetric proximities (here: the standardized flows). The strong components of the directed graph are the sets of mutually reachable nodes (or zones): each node can be reached from another in the same component if there is a series of consistently directed arcs from one to the other. As in the case of the single linkage method, a dendrogram may be constructed, corresponding to the components formed at differing thresholds of proximity. The example shown in Fig. 5.1 is from Tarjan (1983). Note that following each agglomeration, all directed edges from vertices of the new cluster to an outsider vertex may be replaced by the least-weighted edge among them; this, together with a similar updating of in-flows to the new cluster, allows the cluster's vertices to be replaced by a single vertex. Hence, updating after each of the (at most) n-1 agglomerations requires O(n) time. In order to find which pair of vertices to merge, a sorted list of edges may be used (requiring O(m log n) time: i.e. sorting m values, where $m \leqslant n(n-1)/2$). Resulting complexity is then $O(n^2)$ or O(m log n), depending on which term dominates. If m is much less than n, then the O(m log n) algorithm described by Tarjan (1983) should be used.

For constructing a hierarchy of compact clusters, an algorithm proposed by Domengès (1982) is as follows. A symmetric matrix is constructed by summing the $(i,j)^{th}$ elements, for all i and j. Next, the asymmetric matrix is standardized by dividing the $(i,j)^{th}$ element by the product of the associated row and column sums. Finally the sequence of agglomerations takes place by successively seeking the greatest stand-

ardized symmetric flow between regions. Let the symmetric matrix be defined from the given flow matrix by $s_{ij} = f_{ij} + f_{ji}$. When an agglomeration takes place, the (unstandardized) flows to and from the new region, c, equal the sum of flows to and from the sub-clusters a and b : $s_{cc'} = s_{ac'} + s_{bc'}$, for any other region, c'. If s_c and $s_{c'}$ are the totals of rows c and c' (or columns: the matrix has been made symmetric before all agglomerations), then the agglomerations take place on standardized values, s*:

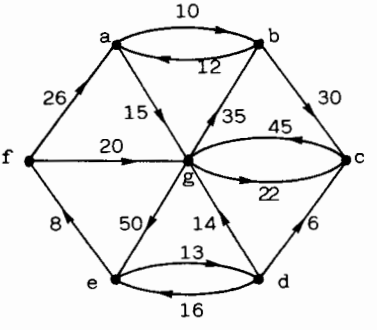
$$s^*_{cc'} = s_{cc'}/s_c s_{c'}$$

$$= (s_{ac'} + s_{bc'})/s_c s_{c'}$$

$$= (s_a s^*_{ac'} + s_b s^*_{bc'})/s_c$$

(simply introducing cancelling terms)

and since $s_c = s_a + s_b$, the above expression ressembles the Lance-Williams update formula for the average linkage (group average or UPGMA) method (cf. Table 1, section 3.3 of Chapter 3).



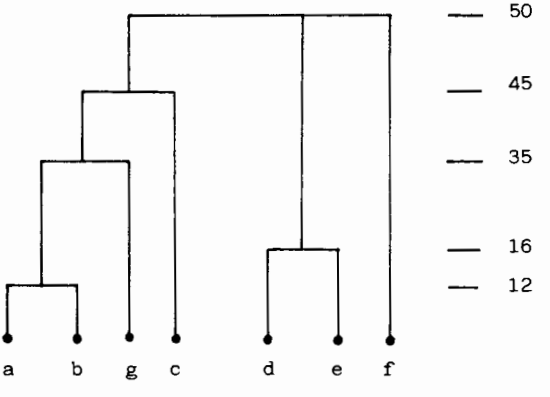


Fig. 5.1 - Example of hierarchical clustering based on strong components at succession of levels.

5.4 REFERENCES

D. DOMENGES, Classification ascendante hiérarchique d'après un critère adapté aux tableaux de flux. Les Cahiers de l'Analyse des Données VII, 169-172 (1982).

M.M. FISCHER, Regional taxonomy. Regional Science and Urban Economics 10, 503-537 (1980).

A.D. GORDON, Methods of constrained classification. In Analyse des Données et Informatique, Edited by R. Tomassone, INRIA, Le Chesnay, pp. 161-171 (1980).

I. MASSER and J. SCHEURWATER, Functional regionalization of spatial interaction data: an evaluation of some suggested strategies. Environment and Planning A 12, 1357-1382 (1980).

F. MURTAGH, A survey of algorithms for contiguity-constrained clustering and related problems. The Computer Journal (in press, 1984).

P.B. SLATER, Combinatorial procedures for structuring internal migration and other transaction flows. Quality and Quantity 15, 179-202 (1981).

R.E. TARJAN, An improved algorithm for hierarchical clustering using strong components. Information Processing Letters 17, 37-41 (1983).