Classification of a set of elements

By M. J. Rose*

The paper describes the use of a computer in some statistical experiments on weakly connected graphs. The work forms part of a statistical approach to some classification problems.

Introduction

In classification problems we are often concerned with trying to classify a set of elements with known characteristics into subsets, such that any two members of the same subset in some sense resemble each other more than members of different subsets. For each element we know which of a given set of properties it has, but division into subsets according to possession or otherwise of particular properties often does not lead to clear-cut non-overlapping groupings. In such cases we can consider every pair of elements, and decide by looking at their characteristics whether or not the members of any pair "resemble" each other (Needham, 1961). The results can be arrayed as a square symmetric matrix whose (i, j)th entry is 1 if elements i and j "resemble" each other, and 0 otherwise. For convenience we take the diagonal entries as zero. The matrix may be considered as the Adjacency matrix of the corresponding graph of the elements, where a point of the graph represents an element and a line indicates that its end-points "resemble" each other (Berge, 1962; Harary *et al.*, 1964–65). In the graph, all lines present have equal importance, are undirected, and loops and multiple lines are not allowed.

The sampling and statistical method for computers outlined below is intended to determine those points and lines which are most likely to be cutpoints and cutsets of the graph. When such points and lines have been found, we remove them from the graph and test by standard procedure whether or not the graph has become disconnected; or equivalently, whether or not the Adjacency matrix, with certain entries removed, has become reducible. One such procedure is given by Needham (1961). If the graph has become disconnected we have succeeded in classifying the elements according to their membership of one or other of the components; if it has not, we repeat the procedure removing at each stage further points and lines until the graph does become disconnected. In connection with the present work, one example used was an adjacency matrix of size 342×342 with non-zero entries about 1 in 10 dense, so that although the corresponding graph could in principle be drawn it would be a far from easy task to detect clumps, non-trivial minimum cutsets, cutpoints and so forth.

* *C.E.I.R. Ltd.*, 30–31 *Newman Street, London*, W.1.

Method of finding points and lines for removal, using a computer

We take a random pair of points, say I and J, determine the shortest path between them on the graph, and store the path in the computer in the form (I, I), (I, K), (K, L), . . . (Z, J),(J, J) where $K, L, \ldots Z$ are the intermediate points on the path. After a statistically sufficient number of such paths have been found and stored, a count is made of how often each line, e.g. (I, K), and each point, e.g. (I, I), has appeared in the paths. From these counts, we can determine four quantities, viz. the number of different points used, the total frequency of all points, the number of different lines used, and the total frequency of all such lines, from which estimates can be obtained for the mean and variance of the count of a typical line, under certain assumptions stated below. Using these estimates a significance threshold can be found, and lines with counts above this are isolated. The expectation is that these lines will be bridges between clumps, or members of minimum cutsets. A test is also carried out to detect significant occurrences of points rather than lines in a similar way. After the significance tests have been done, the cumulative counts are stored and the cycle of path-finding, storing, adding to the cumulative counts, recalculating significance thresholds, and testing is repeated until the significant lines and points become apparent.

Finding the shortest paths

Two numbers I and J are obtained from a random-number generator and are stored as (I, I), (J, J) at the head of two lists held in the computer. The matrix entries in row I are scanned, and lines of the form (K, I) are added to the first list, each K being compared with J to see if there is a direct link (I, J). When all the lines from I have been entered up, the block is terminated and all lines of the form (Z, J) are added to the second list, each Z being compared with entries like K on the first list to see whether there is a path of length two from I to J. When all lines incident to J have been entered up, the block is terminated and the next block on the first list is compiled consisting of entries such as (L, K) which bring in new points such as L which are

irreducibly two steps away from I. Each new point L is compared with points like Z, which are one step away from J, and which, of course, are to be found in the last block only of the second list, to see whether a path of length three exists from I to J. The procedure is repeated, new points on one list being compared as they are found with points in the last block only on the other list, until either a common point is found, in which case the shortest path can be found and stored by tracing back through the two lists, or one list will not have any entries added to it in the construction of a block, in which case no path exists between the two points and hence we may deduce that the graph is not connected. In this case, and in the case when the same number is generated twice by the random-number generator, i.e. when $I = J$, nothing is stored.

If there are more than one equally short paths between two points, the method will only find one, which may be biassed in favour of paths through low-numbered points, if in scanning the matrix entries in a particular row when constructing the above lists, lines to low-numbered points are met first. This trouble, however, may be overcome either by repeating the whole program with significant points relabelled with the highest numbers, or by randomizing the order in which entries in each row of the matrix are scanned.

After a certain number of paths have been found and stored, we move on to the counting and testing procedure.

Counting procedure

Two lists are here constructed, the first of the lines and the points encountered in the path-finding, e.g. (K, L), (I, I), and the second of the corresponding counts. Whenever an entry is read from the store of shortest paths, if it has occurred previously in the first list, its count in the second list is increased by one, while if it has not previously occurred, it is added to the first list and its count set to one. When the testing procedure has been finished, these two lists are stored away while the next group of shortest paths are found, and brought back when the counting routine is re-entered, so that cumulative lists are obtained.

Testing procedure

We need a rational basis for the choice of significance thresholds. Too much sophistication might be out of place here since what we are primarily concerned with is constructing a workable program, which locates significant points and lines only as a first step towards testing for the reducibility of the Adjacency matrix. Furthermore, although the graph may have some suspected structural form, our knowledge of it will often be too slight to be of value, and so we can usefully proceed as follows. We first derive the mean and variance of the number of occurrences of a typical line in, say, r shortest paths, given that there are m points and k lines distributed *randomly* over the $\frac{1}{2}m(m-1)$ possible positions, but not allowing loops or multiple lines (see Erdös and Rényi, 1960).

Let p_i be the probability of choosing two points with a shortest path between them of length i. We can conveniently take $i = 0$ to cover the cases when no path exists and when we choose the same point twice, since in both instances we stored nothing at the path-finding stage. Then the probability of a typical line occurring in this shortest path is i/k, so the expected number of appearances in r shortest paths is $\sum_i \frac{rip_i}{k}$, i.e. $r\mu/k$ where $\mu = \sum_i ip_i$, the mean path length.

The variance of the count

$$= E(\text{number of times line occurs})^2 - \left(\frac{r\mu}{k}\right)^2$$

$$= \sum_m m^2 \binom{r}{m} \left(\frac{\mu}{k}\right)^m \left(1 - \frac{\mu}{k}\right)^{r-m} - \left(\frac{r\mu}{k}\right)^2$$

$$= \frac{r(r-1)\mu^2}{k^2} + \frac{r\mu}{k} - \left(\frac{r\mu}{k}\right)^2$$

$$= \frac{r\mu}{k}\left(1 - \frac{\mu}{k}\right).$$

The standard deviation is thus $\left[\frac{r\mu}{k}\left(1 - \frac{\mu}{k}\right)\right]^{1/2}$ and we take as our threshold value for significance

$$\frac{r\mu}{k} + \alpha_k \left[\frac{r\mu}{k}\left(1 - \frac{\mu}{k}\right)\right]^{1/2},$$

where the α_k depends on the significance level required. In the present work, α_k was taken as $2\frac{1}{2}$, but as explained later, it should increase with k.

The counting procedure incidentally provides us with four quantities: the number of different points such as (I, I), a; the total number of occurrences of all such points, A; the number of different lines which occur at least once, b; and the total number of occurrences of such lines, B. m is estimated by a, k by b, r by $\frac{1}{2}A$, and μ by $B/\frac{1}{2}A = 2B/A$.

Using these estimates, provided by the counting procedure itself, we can evaluate the threshold, then work down the list of counts and register as significant any line whose count is above it.

A test on the counts of points can be derived as follows; we assign the counts of points such as (I, I) to point I but we assign the count of a line such as (K, L) to both point K and point L. We thus count 2 for each endpoint and 2 for each intermediate point.

Count of a typical point for r paths = 2 × number of times
- it is an endpoint of a path + 2 × number of times
- it occurs as an intermediate point of a path = number of times it
- is an endpoint of a path + number of times that group of lines incident to it appear.

E (Count of a typical point for r paths)

$$= \frac{2 \times \text{number of paths}}{\text{number of points}} +$$

$$+\begin{pmatrix}\text{number of lines}\\ \text{incident to point}\end{pmatrix}\cdot\begin{pmatrix}\text{mean frequency of}\\ \text{a typical line}\end{pmatrix}$$

$$=\frac{2r}{m}+\frac{2k}{m}\cdot\frac{r\mu}{k}$$

$$=\frac{2r}{m}(1+\mu).$$

The variance of the line counts was approximately the same as the mean (if $\mu \ll k$), so if we can assume the same is true for the point counts we can take the threshold for point significance as

$$\frac{2r}{m}(1+\mu)+\beta_k\left[\frac{2r}{m}(1+\mu)\right]^{1/2}$$

where β_k depends on the significance level required, and on k. β_k was taken as $2\frac{1}{2}$ in the present work, but as we shall see, a case can be made for setting $\beta_k = \alpha_k\sqrt{2}$. We estimate r, m and μ by $\frac{1}{2}A$, a, and $2B/A$ as above.

When these tests have been done, we go back to the path-finding routine and repeat the whole sequence of operations.

Further remarks on the distribution of line counts

We here give an alternative approach to the statistical problem, which makes rather different assumptions but leads to similar results.

The graph is known to have m points and k lines, and we consider r shortest paths between randomly chosen pairs of points. Each line has a very small chance of occurring in any one path, so that approximately we can take each line count to be a Poisson variable of expectation ν, which is proportional to r. These variables will be only nearly independent, but we shall treat them as if they were. Thus we have k independent Poisson variables $X_1, \ldots, X_k$. The intuitive principle we are working on is that these are nearly a random sample from the Poisson distribution of parameter ν, contaminated by a number of high values from "bridging" lines. We must estimate ν by $\sum_j X_j/k$ and then consider each X_j in relation to this, or equivalently, we consider an independent set of Poisson variables conditional upon their sum; if we put $\gamma_i = \dfrac{X_i}{\sum_j X_j}$, then ignoring the contamination mentioned above, the γ_i's will be the result of assigning $S = \sum_j X_j$ objects randomly and with equal probability $1/k$ to each of k cells. Some remarks about the distribution of $\max \gamma_i$ are made by Koselka (1956) and lead to a significance threshold of

$$\frac{S}{k}+\alpha_k\left[\frac{S}{k}\left(1-\frac{1}{k}\right)\right]^{1/2}$$

which, since $S = B = r\mu$, agrees satisfactorily with the formula given previously when $k \gg \mu$, $k \gg 1$.

α_k is a one-sided (p/k)-significance level for the normal distribution, where p is, say, 0·05. We have a factor $1/k$ appearing here, because if k is very large, some large line counts are to be expected anyway, so that if Z_i is a standardized line count, the expression $pr(\max Z_i > \alpha_k) = p$ gives a more appropriate deviate α_k. If F denotes the normal distribution function, we have $F^k(\alpha_k) = 1 - p$, i.e. $F(\alpha_k) = (1-p)^{1/k} \doteqdot 1 - p/k$, so that α_k is a one-sided (p/k)-significance level.

The fixing of α_k could be carried out automatically if a program were available for inverting the normal distribution integral, but otherwise a value can be decided on beforehand, using only a crude estimate of k.

The treatment of point counts is similar to the above, except that as each endpoint and intermediate point is given a count of 2, the factor β_k in the formula for the threshold corresponds to $\alpha_k\sqrt{2}$, since the variance of twice a Poisson variable is twice its mean.

An interesting possibility in the point count test would be to assign the count of a line such as (K, L) both to point K and to point L, as before, but to subtract the count of a point such as (I, I) from the score of point I. This emphasizes occurrences of a point as an intermediate point of a path by scoring 2 for each, at the expense of occurrences as an endpoint, which are not scored. The significance threshold in this case is $\dfrac{2r}{m}(\mu-1)+\beta_k\left[\dfrac{2r}{m}(\mu-1)\right]^{1/2}$.

Results

On an example of a graph of 21 points and some 36 lines deliberately drawn as two distinct clumps joined only by four lines forming two bridges, the procedure described above successfully isolated the four suspected lines; while on a larger example of a graph of 342 points with lines 1 in 10 dense, some 10 points and 80 lines were shown up as significant after about 1000 paths had been found, though, of course, many more paths ought to be found so that the graph is thoroughly sampled before we can accept the results with any degree of certainty.

There is also a technical reason why r, the number of paths found, must not be too small. We have estimated k by B; this will lead to an appreciable underestimate unless every line is likely to occur at least once. The expected number of occurrences for a typical line is $r\mu/k$ and thus r should exceed, say, $5k/\mu$. If it does not, then it might be necessary to determine k by a complete scanning of the matrix.

Acknowledgements

This work was carried out as part of the Diploma course in the Statistical Laboratory at the University of Cambridge. The programming and computing was done on EDSAC 2 in the University Mathematical Laboratory, by permission of the Director, under the supervision of Dr. R. M. Needham. Helpful comments on the statistical sections of the work were received from Prof. D. G. Kendall. The author is grateful to C.E.I.R. Ltd. for enabling him to complete the work.

References

NEEDHAM, R. M. (1961). Classification and Grouping. Ph.D. Thesis. University of Cambridge.
BERGE, C. (trans. A. Doig) (1962). *Theory of Graphs and its Applications*, London: Methuen.
HARARY, F. et alii, (1964/5). *Structural Models: an introduction to the theory of Directed Graphs.* To be published by Wiley.
ERDÖS, P., and RÉNYI, A. (1960). "Randomly evolved Graphs," *Publ. Math. Inst. Hungarian Academy of Sciences*, Vol. A5, pp. 17–16.
KOZELKA, R. M. (1956). "Approximate upper percentage points for extreme values in multinomial sampling," *Ann. Math. Stat* Vol. 27, pp. 507–512.

Book Review

The Application of Computing Techniques to Automatic Control Systems in Metallurgical Plant, by A. B. CHELYUSTKIN, 1964; 225 pages. (Oxford: *Pergamon Press Ltd.*, 70s.)

Mathematicians and engineers who have been following the magnificent advances in control theory from the Russian school of Pontyagin will be sadly disappointed if they expect to find any applications of this theory. The equipment and techniques described in this book remind one of the first stumbling efforts that have already been made in this country, and one is left with the impression that the full potentiality of computers in industrial control is as far from realization in the Soviet Union as it is here.

The book is divided into two parts. The first and shorter part skims over the subject matter of analogue computers, transducers, digital computers, binary arithmetic, storage devices, and analogue-to-digital conversion equipment, in 56 pages. The remaining 150-odd pages consider some individual applications to a sintering plant, a blast furnace, the combustion of open-hearth furnaces, powering electric arc furnaces, the screwdown and speed in a reversing mill, gauge and tension control in tandem mills. All of these applications seem to be very straightforward and offer no new startling ideas. It is very difficult to judge whether these are merely proposals or descriptions of actual installations. The last few pages make a cursory survey of the possibility of on-line control of a total works by digital computer.

It must, in fairness, be added that the book was originally written in 1960, and it is to be hoped that in the intervening four years a great deal of progress has been made.

K. D. TOCHER