Pattern Recognition Letters 15 (1994) 27–34 January 1994

Dynamic clustering for time incremental data

B.B. Chaudhuri

Electronics and Communication Sciences Unit, Indian Statistical Institute, 203 B.T. Road, Calcutta 700 035, India

Abstract

Chaudhuri, B.B., Dynamic clustering for time incremental data, Pattern Recognition Letters 15 (1994) 27–34.

The aim of this article is to propose a model for on-line data clustering when a new subset of data accumulates after an interval of time. The new data may be absorbed in the old clusters or form new clusters or appear as stray data. The absorbed data may cause the clusters to grow so that two grown clusters may merge to form a single cluster. On the other hand, a large number of absorbed data may change the density profile of a cluster so that it should be split into two or more clusters. Procedures to compute these situations are proposed.

Keywords. Clustering, data analysis, pattern recognition.

1. Introduction

Numerous data clustering algorithms are available in the literature [1,4,5]. Most algorithms are static or off line in the sense that the data to be clustered are all available at a time. In contrast, there can be a steady arrival of new data, and it may be necessary to find a dynamic (or on line) clustering algorithm that updates the clustering results after each n new data are obtained.

On the other hand, it is possible to convert an off-line situation of N data into an on-line situation where we start with clustering $N_0 < N$ data and then add n data at each instant picked randomly from the rest. If for a few instants the clustering results do not change appreciably, we can assume that the process has stabilized and declare it as the final outcome. This approach may be useful when (a) N is very large and the clustering algorithm cost increases non-linearly with N, and (b) it is necessary to know the number and types of clusters generated by the data, rather than placing each datum to one of the clusters.

Unfortunately, we have not come across any dynamic (or on-line) clustering algorithm in the sense stated above. Use of a smaller subset of data for seed point evaluation is reported, as in Astrahan [2], but a dynamic cluster updating concept is absent in his procedure. Our approach is based on a few simple observations about how new data can perturb the existing clusters and how these observations can be quantitatively evaluated. In this connection, some basic principles of data clustering have been clarified.

2. Principles of dynamic clustering

A close look at how data appear to be clustered in the Euclidean space will reveal that clustering is a relativistic phenomenon. Here, we present some points

that may be useful for designing clustering algorithms in general.

(i) Whether a set of data S will appear clustered depends on the window of space in which the data set is defined. If S is the only data set in the window and if S appears as clustered then the window size is much bigger than that required to enclose S.

(ii) If a set of data S appears as clustered then the 'density' of data will show a single plateau or a single top and no prominent valley.

(iii) If there exists another cluster T in the window of space and if S appears clustered then either one or more of the following should be true:

(a) Number of data in S is comparable with that in T.

(b) Average spacing (density) of data in S is comparable with that in T.

(c) Distance between S and T is large compared to the average spacing of data in S and T.

If the number of data in S is very small compared to that in T, it may not be wise to consider S as a cluster. Similarly, if the average spacing of data in S is very large compared to T the data of S may appear as stray un-clustered data. On the other hand, if the distance between S and T is not large compared to the average spacing of data in S and T then they do not appear as distinct clusters. Then either a part or the whole of S gets absorbed in T (i.e., appears as part of T).

There is interplay of the characteristics (a)–(c). Depending on the situation, one characteristic predominates or aids or obstructs the other in the cluster perception.

(iv) Subclusters may be seen in S when 'viewed closely'. But when viewed from a distance, S may appear as a single cluster. This principle supports the hierarchical agglomerative or divisive clustering procedures in the literature.

In the light of the above principles we can analyze the set of new data arriving in the space of existing data. For convenience, we introduce a dummy time variable so that at time t there are N data in the observation space. At time $t+\Delta t$ n new data have arrived.

Due to arrival of new data one or more of the following situations may arise.

(a) *Absorption of data.* The new data may be absorbed in the existing clusters and hence the clustering remains unchanged.

(b) *Merging of clusters.* Two or more existing clusters may be merged or connected into one by the new data.

(c) *New cluster formation.* Separate new clusters may be formed.

(d) *Stray data.* The data appear as stray un-clustered data.

When a large number of new data have arrived, say, after $r\Delta t$, $r \gg 1$, another situation may occur.

(e) One cluster may appear as two or more clusters because a large number of new data have been absorbed in the cluster.

Examples of the above situations are illustrated in Figures 1(A) and (B). It may be understood that a new datum is absorbed if it is very close to a cluster. The cluster may grow because of absorbance of new data. If the new data make two (or more) clusters to 'grow' then the clusters may be merged into one. Stray data are those on which the clustering technique does not succeed.

3. Computation of situations

At first, we consider the problems of computing situations (a)–(d). It is understood that testing of situation (a) should follow the situations (b) and (c). Situation (d) is automatically tested while testing situation (c). Situation (e) is a difficult one, involving the problem of finding clusters in a cluster, which we shall discuss later on.

Absorption of data

Absorption of data in an existing cluster is based on the following principles.

(i) A new datum p is likely to be absorbed in the cluster nearest (most similar) to it. Let C_i be the cluster to which p is nearest.

The distance of C_i from p may be defined as the minimum of Euclidean distances of all points of C_i from p.

(ii) Let q_0 be the datum in C_i nearest to p. Then p will be absorbed in C_i if the nearest neighbor distances of points q_i in the neighborhood of q_0 are similar to the distance between p and q_0.

More specifically, let $q_1, q_2, \ldots, q_m$ be the m nearest neighbors of q_0 in C_i. For each q_i, $i=0, \ldots, m$, find the

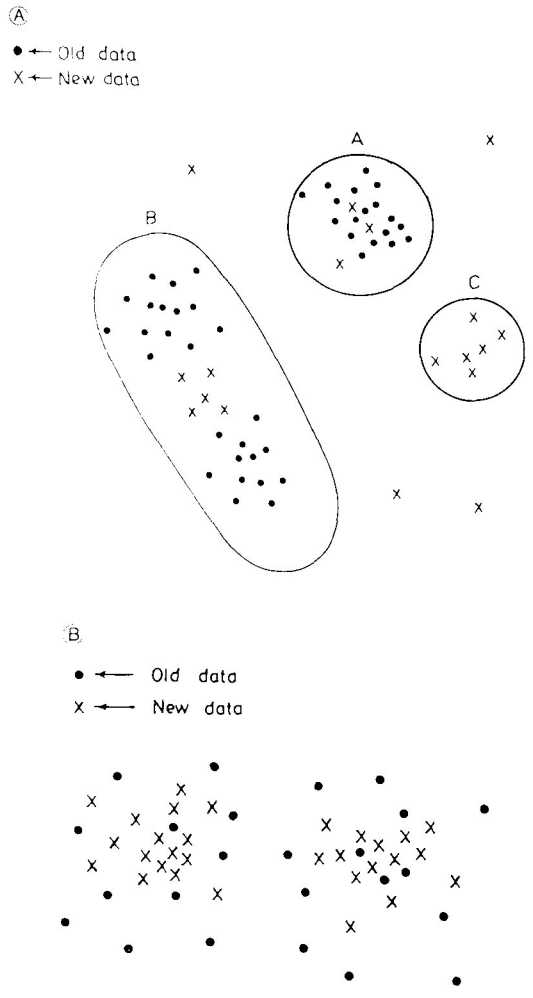

Figure 1. Situation in dynamic clustering. (A) Absorbed data A, Merged clusters B, New cluster C. The remaining data are stray data. (B) The new data transforms a single cluster into two.

distance d_i from its nearest neighbor in C_i. Let

$$\bar{d} = \frac{1}{m+1} \sum_{i=0}^{m} d_i . \qquad (1)$$

Let the distance between p and q_0 be d. Then absorb p in C_i if $d \simeq \bar{d}$.

We can choose m as the smallest integer greater than 10% of the number of data in C_i. The procedure is illustrated in Figure 2 where p is the new data currently being tested for absorption. q_0, q_1, ..., q_4 are its 5 nearest neighbors in the cluster. Now, the 1-nearest neighbor of q_0 is q_1 while the 1-nearest neighbor of q_1

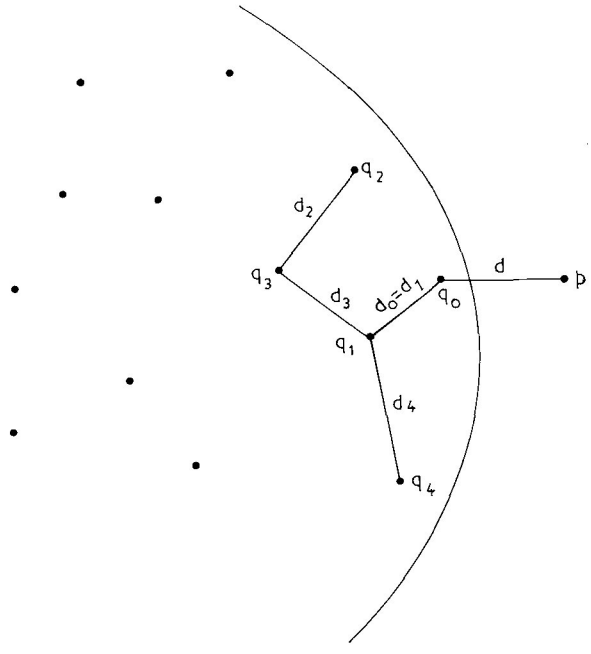

Figure 2. Principle of absorbing a new data p.

is q_0. The 1-nearest neighbors of q_2, q_3 and q_4 are q_3, q_1 and q_1, respectively. The corresponding d_i, $i=0, ..., 4$, are shown in Figure 2. Equation (1) is used to compute $\bar{d}$.

Note that a datum q rejected for absorption at the current iteration can be absorbed later when C_i has grown because of absorption of other data. The algorithm should stop at the iteration when not a single new datum is absorbed in a cluster. For n data being considered, therefore, the worst number of iterations is n.

Merging of clusters

Merging of clusters is based on the following principles.

(i) Only the clusters absorbing new data at the previous stage are considered for merging.

(ii) If the distance between one cluster surviving test (i) above and any other cluster is reduced (as compared to the distance that existed before data absorption), then only these two clusters are considered for merging. The distance will not get reduced if a cluster does not grow because of absorption. The

distance between two clusters C_i and C_j may be defined as the minimum of Euclidean distances between one point of C_i and one point of C_j.

(iii) For two clusters C_i and C_j considered for merging let $p \in C_i$ and $q \in C_j$ be such that the distance between p and q is minimum. Thus, p and q are the points where the two clusters are closest to each other and they could be merged around these points.

(iv) The behavior of C_i and C_j should be identical around p and q if merging is agreed.

More specifically, consider m nearest neighbors of p and m nearest neighbors of q in $C_i \cup C_j$. If the two clusters are homogeneous at p and q we would expect that approximately $m/2$ nearest neighbors will come from C_i and $m/2$ nearest neighbors will come from C_j in both the cases. On the other hand, if the two clusters are distinct and not to be merged, then in case of p, most of the m nearest neighbors will come from C_i while in case of q, most of the m nearest neighbors will come from C_j. Let $m_{p,i}$ be the number of nearest neighbors of p coming from C_i and so on. Then $m_{p,i}+m_{p,j}=m=m_{q,i}+m_{q,j}$. Consider the quantity

$$J=\frac{m_{p,j}}{m}+\frac{m_{q,i}}{m}. \tag{2}$$

The quantity should tend to 1 if C_i and C_j should be merged while it should tend to 0 if C_i and C_j are not to be merged. We can use a threshold $0<T<1$ to decide whether the two clusters should be merged or not. Thus, if $J \geq T$, the two clusters should be merged.

The situation is illustrated in Figure 3 where in Figure 3(A) $J=0$ and in Figure 3(B) $J=2/5+2/5=4/5$.

Detection of new cluster and stray data

The data not absorbed in any existing cluster can form new clusters. One of the many existing clustering algorithms may be used for the purpose. However, since there exist (except at the beginning, i.e., $t=0$) some clusters in the observation window, the new clusters should be compatible with the existing ones. More specifically, as stated in principle (iii) of Section 2 we should expect that:

(i) The density of new clusters should not be much less than the existing ones.

(ii) The number of data in a new cluster should not be much less than the existing ones.

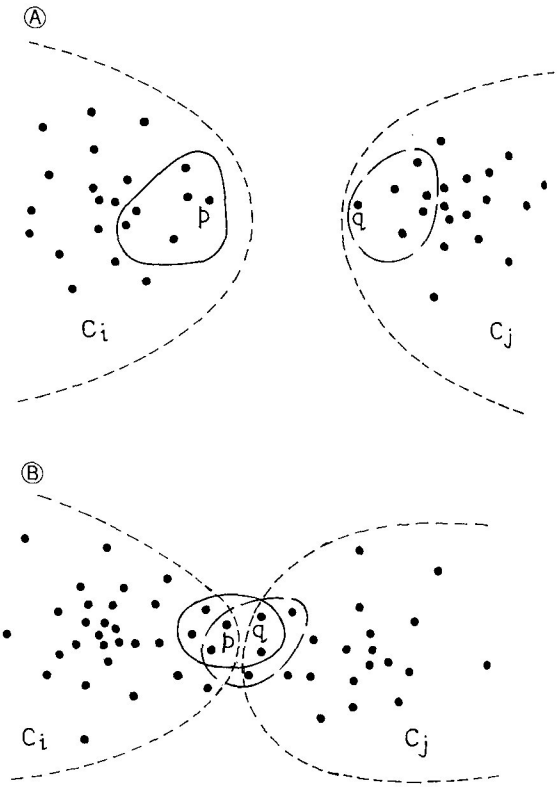

Figure 3. Test for merging of two clusters. (A) When two clusters should not be merged. (B) When two clusters should be merged.

We use the following algorithm to detect new clusters.

(a) Find the minimum spanning tree of the new data which are not absorbed in the previous stage.

(b) Find the average of distances of each datum and its nearest neighbor in the existing clusters. Let it be $\bar{d}_0$. Delete the edges with length greater than $\alpha \bar{d}_0$ where $\alpha > 1$. The minimum spanning tree is now converted into a set of subtrees.

(c) If the number of nodes in a subtree is less than a pre-specified number m_0 then consider all data of the subtree as stray data. Otherwise, a subtree denotes a new cluster.

The stray data and the new data arrived after Δt are considered as the data for clustering for the next interval. Thus, if n data arrive after Δt, the total number of data to be examined is $n+n_S(t)$ where $n_S(t)$ is the number of stray data left at the tth instant.

Splitting of cluster

As stated in situation (e) in Section 2, a large number of new absorbed data may change the structure of a cluster so that it should be split into two or more clusters. We consider that the splitting criterion is examined at an interval of $r\Delta t$.

A splitting test should be conducted on the clusters that absorb a 'sufficient' number of new data during the interval. There exist a few algorithms that perform splitting at some stage of the clustering procedure. Popular among them are the ISODATA [3] and its variants. In ISODATA, a cluster is split if it has too many data and an unusually large variance along the dimension with largest spread. The algorithm does not look at the data density valley region and hence may not be adequate on many occasions. In fact, judgment of a cluster splitting situation is difficult and computationally unattractive. Thus, there exist very few reports involving a divisive hierarchical clustering procedure [6].

We propose the cluster splitting algorithm based on the following principles.

1. The density at each datum in the cluster is computed. An estimate of the density at a point p may be obtained either by a kernel-based approach or by the k-nearest neighbor approach [7].

2. Let the highest and lowest density points be μ_h and μ_l. Consider a fraction β of $\mu_h - \mu_l$. Let $\mu_0 = \beta(\mu_h - \mu_l)$. Discard all points with density less than $\mu_l + \mu_0$ and generate the minimum spanning tree with the remaining data. Find the average of edge lengths of the tree. If the maximum edge length is much larger than the average edge length and if by deleting the edge with maximum length two sub-trees with a reasonable number of nodes are generated, then consider each subtree as a separate cluster.

4. An experiment

An experiment is presented here to illustrate the procedures of dynamic clustering. The initialization stage corresponds to a set of data initially available for clustering. This stage is called the initial clustering stage. After some time, a set of new data is available and the dynamic clustering procedure is started. Figure 4(A) depicts such a situation where an old (initial) datum is denoted by a dot while a new datum is denoted by a cross. The initial clustering procedure adopted here is based on computing the minimum spanning tree of the data. Edges larger than 1.5 times the average edge length are deleted from the spanning tree. In the remaining structure, a tree with 5 or more nodes (data) is declared to represent a cluster. As a result, clusters A and B are formed. All remaining data are stray data. So, at the stage of Figure 4(A) both the stray and new data are subject to dynamic clustering. At this stage, a new cluster C is formed and some data are absorbed by clusters A and B. Figure 4(B) shows the situation when some more data have arrived. In this case, two new clusters D and E are formed while A, B and C have absorbed some data. The value of α for new cluster formation is 2.0. Next, we show the situation of Figure 4(C) where the new data caused two clusters C and D to be merged into one. The threshold T for cluster merging is 0.5. Other clusters absorbed some data. Figure 4(D) is the situation when, after a reasonable period since initial clustering, a test of cluster splitting is made on each of the clusters. The test is successful on cluster C of the previous stage which is now split into clusters C and D. The value of β used in the procedure is 0.1.

One of the advantages of the dynamic clustering procedure is that the results are less sensitive to the parameters used for clustering. If a cluster is not detected or if two clusters are not merged at some stage due to an inadequate choice of the parameters, the results may be modified at a later stage when new data have arrived.

5. Discussions

The situations (a)–(e) described in Section 2 are general for a wide variety of data. Even when the data are in the form of 'evidences' where the clusters may be viewed as 'concepts', the situations (a)–(e) have clear interpretations. For example, new evidences (i) may support an existing concept (similar to absorption of data), (ii) may merge two concepts into a broader one (similar to merging of clusters), (iii) may make a single concept contradictory so that two or more concepts are needed to explain the evidences (splitting of one cluster into two or more), (iv) may

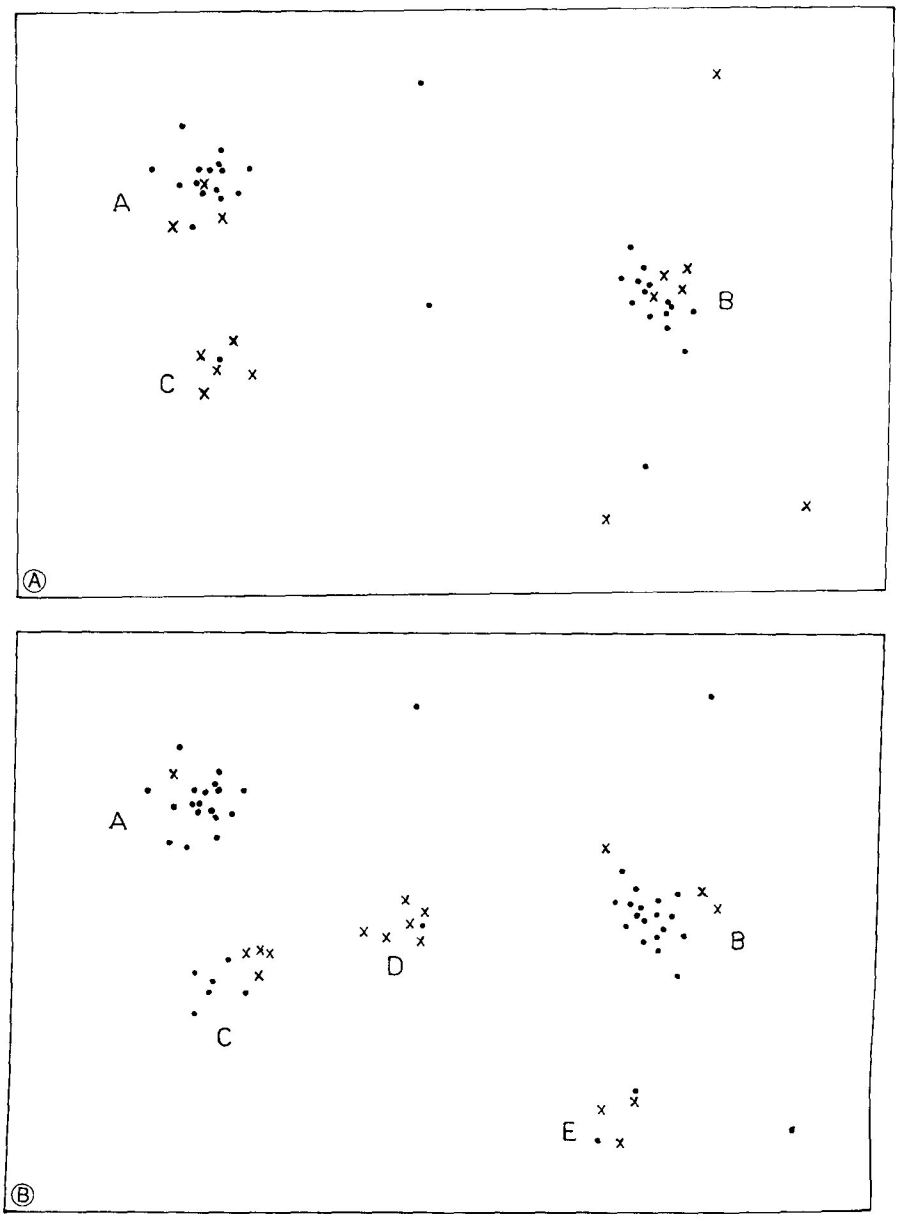

Figure 4. A dynamic clustering session. (See text for details.)

form a separate new concept (similar to new cluster formation) or (v) may not be strong enough to form a new concept (stray data).

Computation of the situations may depend on the type of the problem. It is possible to propose algorithms to compute the situations that are different from those presented in this article. However, in the absence of any specific knowledge about a problem, we stressed data driven algorithms that need few parameter specifications. These algorithms and their underlying concepts may be applied for conventional clustering of off-line data. For example, the algorithms for merging and splitting may be combined to propose a split and merge clustering algorithm.

It is hoped that this article will stimulate further interest on incremental data clustering problems.

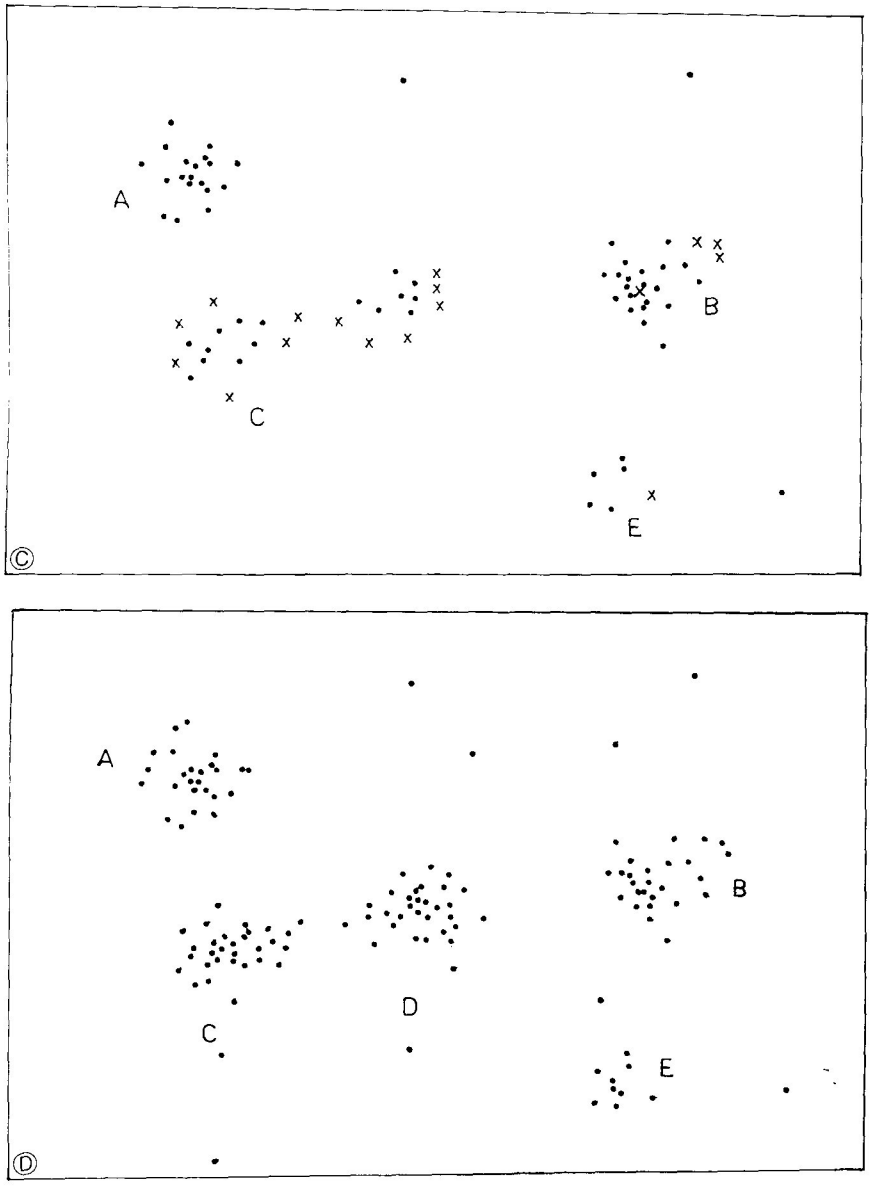

Figure 4 (contd.).

Acknowledgment

The author wishes to thank Mr. S. Chackraborty, Mr. N. Sarkar, and Mrs. M. De for secretarial help.

References

[1] Anderberg, M.R. (1973). *Cluster Analysis for Applications*. Academic Press, New York.

[2] Astrahan, M.M. (1970). Speech analysis by clustering or the hyperphoneme method. Stanford A.I. Project Men. AIM-124, AD 709067, Stanford Univ., CA.

[3] Ball, G.H. and D.J. Hall (1964). Some fundamental concepts and synthesis procedures for pattern recognition preprocessors. *Proc. Internat. Conf. on Microwaves, Circuit Theory and Information Theory*, Sept. 1964, Tokyo.

[4] Hartigan, J.A. (1975). *Clustering Algorithms*. Wiley, New York.

[5] Jain, A.K. and R.C. Dubes (1988). *Algorithms for Clustering Data*. Prentice-Hall, Englewood Cliffs, NJ.

[6] Jambu, M. and M.O. Lebeaux (1983). *Cluster Analysis and Data Analysis*. North-Holland, Amsterdam, 171–175.

[7] Prakasa Rao, B.L.S. (1983). *Nonparametric Functional Estimation*. Academic Press, New York.