EXHIBIT 4
PART

6.10.2.1 THE NEAREST-NEIGHBOR ALGORITHM

Consider first the behavior when $d_{\min}$ is used.* Suppose that we think of the data points as being nodes of a graph, with edges forming a path between nodes in the same subset $\mathscr{X}_i$.† When $d_{\min}$ is used to measure the distance between subsets, the nearest neighbors determine the nearest subsets. The merging of $\mathscr{X}_i$ and $\mathscr{X}_j$ corresponds to adding an edge between the nearest pair of nodes in $\mathscr{X}_i$ and $\mathscr{X}_j$. Since edges linking clusters always go between distinct clusters, the resulting graph never has any closed loops or circuits; in the terminology of graph theory, this procedure generates a *tree*. If it is allowed to continue until all of the subsets are linked, the result is a *spanning tree*, a tree with a path from any node to any other node. Moreover, it can be shown that the sum of the edge lengths of the resulting tree will not exceed the sum of the edge lengths for any other spanning tree for that set of samples. Thus, with the use of $d_{\min}$ as the distance measure, the agglomerative clustering procedure becomes an algorithm for generating a *minimal spanning tree*.

Figure 6.17 shows the results of applying this procedure to the data of Figure 6.16. In all cases the procedure was stopped at $c = 2$; a minimal spanning tree can be obtained by adding the shortest possible edge between the two clusters. In the first case where the clusters are compact and well separated, the obvious clusters are found. In the second case, the presence of a few points located so as to produce a bridge between the clusters results in a rather unexpected grouping into one large, elongated cluster, and one small, compact cluster. This behavior is often called the "chaining effect," and is sometimes considered to be a defect of this distance measure. To the extent that the results are very sensitive to noise or to slight changes in position of the data points, this is certainly a valid criticism. However, as the third case illustrates, this very tendency to form chains can be advantageous if the clusters are elongated or possess elongated limbs.

6.10.2.2 THE FURTHEST-NEIGHBOR ALGORITHM

When $d_{\max}$ is used to measure the distance between subsets, the growth of elongated clusters is discouraged.‡ Application of the procedure can be thought of as producing a graph in which edges connect all of the nodes in

* In the literature, the resulting procedure is often called the *nearest-neighbor* or the *minimum* algorithm. If it is terminated when the distance between nearest clusters exceeds an arbitrary threshold, it is called the *single-linkage* algorithm.

† Although we will not make deep use of graph theory, we assume that the reader has a general familiarity with the subject. A clear, rigorous treatment is given by O. Ore, *Theory of Graphs* (American Math. Soc. Colloquium Publ., Vol. 38, 1962).

‡ In the literature, the resulting procedure is often called the *furthest neighbor* or the *maximum* algorithm. If it is terminated when the distance between nearest clusters exceeds an arbitrary threshold, it is called the *complete-linkage* algorithm.

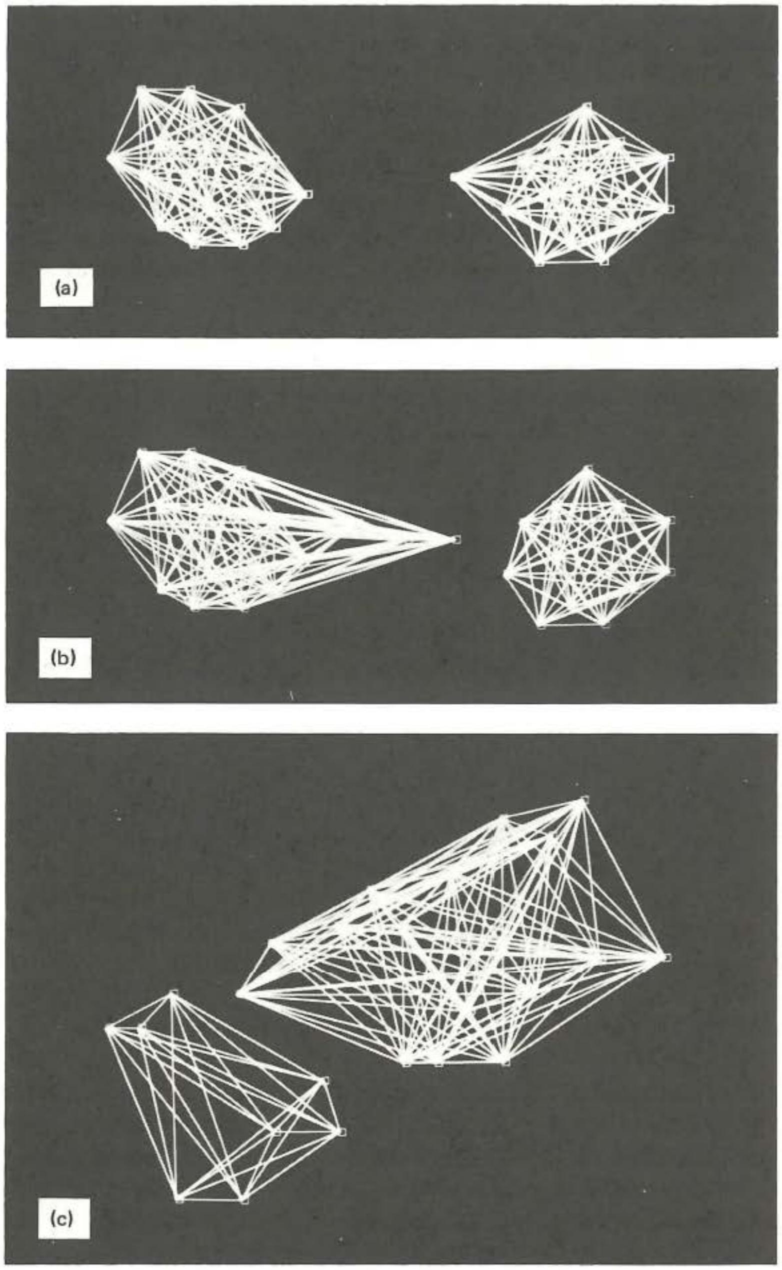

FIGURE 6.18. Results of the furthest-neighbor algorithm.

a cluster. In the terminology of graph theory, every cluster constitutes a *complete* subgraph. The distance between two clusters is determined by the most distant nodes in the two clusters. When the nearest clusters are merged, the graph is changed by adding edges between every pair of nodes in the two clusters. If we define the *diameter* of a cluster as the largest distance between points in the cluster, then the distance between two clusters is merely the diameter of their union. If we define the diameter of a partition as the largest diameter for clusters in the partition, then each iteration increases the diameter of the partition as little as possible. As Figure 6.18 illustrates, this is advantageous when the true clusters are compact and roughly equal in size. However, when this is not the case, as happens with the two elongated clusters, the resulting groupings can be meaningless. This is another example of imposing structure on data rather than finding structure in it.

6.10.2.3 COMPROMISES

The minimum and maximum measures represent two extremes in measuring the distance between clusters. Like all procedures that involve minima or maxima, they tend to be overly sensitive to "mavericks" or "sports" or "outliers" or "wildshots." The use of averaging is an obvious way to ameliorate these problems, and d_{avg} and d_{mean} are natural compromises between d_{min} and d_{max}. Computationally, d_{mean} is the simplest of all of these measures, since the others require computing all $n_i n_j$ pairs of distances $\|\mathbf{x} - \mathbf{x}'\|$. However, a measure such as d_{avg} can be used when the distances $\|\mathbf{x} - \mathbf{x}'\|$ are replaced by similarity measures, where the similarity between mean vectors may be difficult or impossible to define. We leave it to the reader to decide how the use of d_{avg} or d_{mean} might change the way that the points in Figure 6.16 are grouped.

6.10.3 Stepwise-Optimal Hierarchical Clustering

We observed earlier that if clusters are grown by merging the nearest pair of clusters, then the results have a minimum variance flavor. However, when the measure of distance between clusters is chosen arbitrarily, one can rarely assert that the resulting partition extremizes any particular criterion function. In effect, hierarchical clustering defines a cluster as whatever results from applying the clustering procedure. However, with a simple modification it is possible to obtain a stepwise-optimal procedure for extremizing a criterion function. This is done merely by replacing Step 3 of the Basic Agglomerative Clustering Procedure (Section 6.10.2) by

3′. Find the pair of distinct clusters $\mathscr{X}_i$ and $\mathscr{X}_j$ whose merger would increase (or decrease) the criterion function as little as possible.

This assures us that at each iteration we have done the best possible thing, even if it does not guarantee that the final partition is optimal.

We saw earlier that the use of $d_{\max}$ causes the smallest possible stepwise increase in the diameter of the partition. Another simple example is provided by the sum-of-squared-error criterion function J_e. By an analysis very similar to that used in Section 6.9, we find that the pair of clusters whose merger increases J_e as little as possible is the pair for which the "distance"

$$d_e(\mathscr{X}_i, \mathscr{X}_j) = \sqrt{\frac{n_i n_j}{n_i + n_j}}\,\|\mathbf{m}_i - \mathbf{m}_j\|$$

is minimum. Thus, in selecting clusters to be merged, this criterion takes into account the number of samples in each cluster as well as the distance between clusters. In general, the use of d_e tends to favor growth by adding singletons or small clusters to large clusters over merging medium-sized clusters. While the final partition may not minimize J_e, it usually provides a very good starting point for further iterative optimization.

6.10.4 Hierarchical Clustering and Induced Metrics

Suppose that we are unable to supply a metric for our data, but that we can measure a *dissimilarity* value $\delta(\mathbf{x}, \mathbf{x}')$ for every pair of samples, where $\delta(\mathbf{x}, \mathbf{x}') \geq 0$, equality holding if and only if $\mathbf{x} = \mathbf{x}'$. Then agglomerative clustering can still be used, with the understanding that the nearest pair of clusters is the least dissimilar pair. Interestingly enough, if we define the dissimilarity between two clusters by

$$\delta_{\min}(\mathscr{X}_i, \mathscr{X}_j) = \min_{\mathbf{x}\in\mathscr{X}_i, \mathbf{x}'\in\mathscr{X}_j} \delta(\mathbf{x}, \mathbf{x}')$$

or

$$\delta_{\max}(\mathscr{X}_i, \mathscr{X}_j) = \max_{\mathbf{x}\in\mathscr{X}_i, \mathbf{x}'\in\mathscr{X}_j} \delta(\mathbf{x}, \mathbf{x}'),$$

then the hierarchical clustering procedure will induce a distance function for the given set of n samples. Furthermore, the ranking of the distances between samples will be invariant to any monotonic transformation of the dissimilarity values.

To see how this comes about, we begin by defining the *value* v_k for the clustering at level k. For level 1, $v_1 = 0$. For all higher levels, v_k is the minimum dissimilarity between pairs of distinct clusters at level $k - 1$. A moment's reflection will make it clear that with both $\delta_{\min}$ and $\delta_{\max}$ the value v_k either stays the same or increases as k increases. Moreover, we shall assume that no two of the n samples are identical, so that $v_2 > 0$. Thus, $0 = v_1 < v_2 \leq v_3 \leq \cdots \leq v_n$.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase®
Smarter legal research.