EXHIBIT 4
PART

equations for the local-maximum-likelihood estimates $\hat{\mu}_i$, $\hat{\Sigma}_i$, and $\hat{P}(\omega_i)$:

$$\hat{P}(\omega_i) = \frac{1}{n}\sum_{k=1}^{n}\hat{P}(\omega_i \mid \mathbf{x}_k, \hat{\theta}) \tag{14}$$

$$\hat{\mu}_i = \frac{\sum_{k=1}^{n}\hat{P}(\omega_i \mid \mathbf{x}_k, \hat{\theta})\mathbf{x}_k}{\sum_{k=1}^{n}\hat{P}(\omega_i \mid \mathbf{x}_k, \hat{\theta})} \tag{15}$$

$$\Sigma_i = \frac{\sum_{k=1}^{n}\hat{P}(\omega_i \mid \mathbf{x}_k, \hat{\theta})(\mathbf{x}_k - \hat{\mu}_i)(\mathbf{x}_k - \hat{\mu}_i)^t}{\sum_{k=1}^{n}\hat{P}(\omega_i \mid \mathbf{x}_k, \hat{\theta}_i)} \tag{16}$$

where

$$\begin{aligned}\hat{P}(\omega_i \mid \mathbf{x}_k, \hat{\theta}) &= \frac{p(\mathbf{x}_k \mid \omega_i, \hat{\theta}_i)\hat{P}(\omega_i)}{\sum_{j=1}^{c}p(\mathbf{x}_k \mid \omega_j, \hat{\theta}_j)\hat{P}(\omega_j)} \\ &= \frac{|\hat{\Sigma}_i|^{-1/2}\exp[-\frac{1}{2}(\mathbf{x}_k - \hat{\mu}_i)^t\hat{\Sigma}_i^{-1}(\mathbf{x}_k - \hat{\mu}_i)]\hat{P}(\omega_i)}{\sum_{j=1}^{c}|\hat{\Sigma}_j|^{-1/2}\exp[-\frac{1}{2}(\mathbf{x}_k - \hat{\mu}_j)^t\hat{\Sigma}_j^{-1}(\mathbf{x}_k - \hat{\mu}_j)]\hat{P}(\omega_j)}.\end{aligned} \tag{17}$$

While the notation may make these equations appear to be rather formidable, their interpretation is actually quite simple. In the extreme case where $\hat{P}(\omega_i \mid \mathbf{x}_k, \hat{\theta})$ is one when $\mathbf{x}_k$ is from Class ω_i and zero otherwise, $\hat{P}(\omega_i)$ is the fraction of samples from ω_i, $\hat{\mu}_i$ is the mean of those samples, and $\hat{\Sigma}_i$ is the corresponding sample covariance matrix. More generally, $\hat{P}(\omega_i \mid \mathbf{x}_k, \hat{\theta})$ is between zero and one, and all of the samples play some role in the estimates. However, the estimates are basically still frequency ratios, sample means, and sample covariance matrices.

The problems involved in solving these implicit equations are similar to the problems discussed in Section 6.4.1, with the additional complication of having to avoid singular solutions. Of the various techniques that can be used to obtain a solution, the most obvious approach is to use initial estimates to evaluate Eq. (17) for $\hat{P}(\omega_i \mid \mathbf{x}_k, \hat{\theta})$ and then to use Eqs. (14)–(16) to update these estimates. If the initial estimates are very good, having perhaps been obtained from a fairly large set of labelled samples, convergence can be quite rapid. However, the results do depend upon the starting point, and the problem of multiple solutions is always present. Furthermore, the repeated computation and inversion of the sample covariance matrices can be quite time consuming.

Considerable simplification can be obtained it it is possible to assume that the covariance matrices are diagonal. This has the added virtue of reducing the number of unknown parameters, which is very important when the number of samples is not large. If this assumption is too strong, it still may be possible to obtain some simplification by assuming that the c covariance matrices are equal, which also eliminates the problem of singular solutions. The derivation of the appropriate maximum likelihood equations for this case is treated in Problems 5 and 6.

6.4.4 A Simple Approximate Procedure

Of the various techniques that can be used to simplify the computation and accelerate convergence, we shall briefly consider one elementary, approximate method. From Eq. (17), it is clear that the probability $\hat{P}(\omega_i \mid \mathbf{x}_k, \hat{\boldsymbol{\theta}})$ is large when the squared Mahalanobis distance $(\mathbf{x}_k - \hat{\boldsymbol{\mu}}_i)^t \hat{\Sigma}_i^{-1} (\mathbf{x}_k - \hat{\boldsymbol{\mu}}_i)$ is small. Suppose that we merely compute the squared Euclidean distance $\|\mathbf{x}_k - \hat{\boldsymbol{\mu}}_i\|^2$, find the mean $\hat{\boldsymbol{\mu}}_m$ nearest to $\mathbf{x}_k$, and approximate $\hat{P}(\omega_i \mid \mathbf{x}_k, \hat{\boldsymbol{\theta}})$ as

$$\hat{P}(\omega_i \mid \mathbf{x}_k, \hat{\boldsymbol{\theta}}) \approx \begin{cases} 1 & i = m \\ 0 & \text{otherwise.} \end{cases}$$

Then the iterative application of Eq. (15) leads to the following procedure* for finding $\hat{\boldsymbol{\mu}}_1, \ldots, \hat{\boldsymbol{\mu}}_c$:

Procedure: Basic Isodata

1. Choose some initial values for the means $\hat{\boldsymbol{\mu}}_1, \ldots, \hat{\boldsymbol{\mu}}_c$.

Loop:

2. Classify the n samples by assigning them to the class of the closest mean.
3. Recompute the means as the average of the samples in their class.
4. If any mean changed value, go to Loop; otherwise, stop.

This is typical of a class of procedures that are known as *clustering* procedures. Later on we shall place it in the class of iterative optimization procedures, since the means tend to move so as to minimize a squared-error

* Throughout this chapter we shall name and describe various iterative procedures as if they were computer programs. All of these procedures have in fact been programmed, often with much more elaborate provisions for doing such things as breaking ties, avoiding trap states, and allowing more sophisticated terminating conditions. Thus, we occasionally include the word "basic" in their names to emphasize the fact that our interest is limited to explaining essential concepts.

criterion function. At the moment we view it merely as an approximate way to obtain maximum likelihood estimates for the means. The values obtained can be accepted as the answer, or can be used as starting points for the more exact computations.

It is interesting to see how this procedure behaves on the example data in Table 6-1. Figure 6.4 shows the sequence of values for $\hat{\mu}_1$ and $\hat{\mu}_2$ obtained

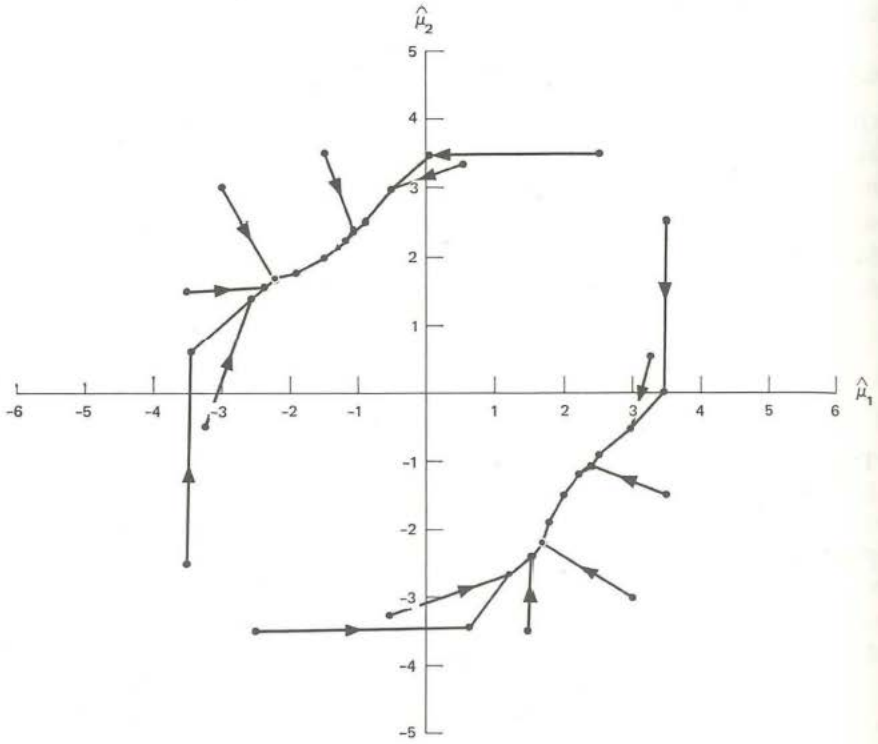

FIGURE 6.4. Trajectories for the Basic Isodata Procedure.

for several different starting points. Since interchanging $\hat{\mu}_1$ and $\hat{\mu}_2$ merely interchanges the labels assigned to the data, the trajectories are symmetric about the line $\hat{\mu}_1 = \hat{\mu}_2$. The trajectories lead either to the point $\hat{\mu}_1 = -2.176$, $\hat{\mu}_2 = 1.684$ or to its image. This is close to the solution found by the maximum likelihood method (viz., $\hat{\mu}_1 = -2.130$ and $\hat{\mu}_2 = 1.668$), and the trajectories show a general resemblance to those shown in Figure 6.3 In general, when the overlap between the component densities is small the maximum likelihood approach and the Isodata procedure can be expected to give similar results.

6.5 UNSUPERVISED BAYESIAN LEARNING

6.5.1 The Bayes Classifier

Maximum likelihood methods do not consider the parameter vector $\boldsymbol{\theta}$ to be random—it is just unknown. Prior knowledge about likely values for $\boldsymbol{\theta}$ is irrelevant, although in practice such knowledge may be used in choosing good starting points for hill-climbing procedures. In this section we shall take a Bayesian approach to unsupervised learning. We shall assume that $\boldsymbol{\theta}$ is a random variable with a known a priori distribution $p(\boldsymbol{\theta})$, and we shall use the samples to compute the a posteriori density $p(\boldsymbol{\theta} \mid \mathcal{X})$. Interestingly enough, the analysis will virtually parallel the analysis of supervised Bayesian learning, showing that the two problems are formally very similar.

We begin with an explicit statement of our basic assumptions. We assume that:

1. The number of classes is known.
2. The a priori probabilities $P(\omega_j)$ for each class are known, $j = 1, \ldots, c$.
3. The forms for the class-conditional probability densities $p(\mathbf{x} \mid \omega_j, \boldsymbol{\theta}_j)$ are known, $j = 1, \ldots, c$, but the parameter vector $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_c)$ is not known.
4. Part of our knowledge about $\boldsymbol{\theta}$ is contained in a known a priori density $p(\boldsymbol{\theta})$.
5. The rest of our knowledge about $\boldsymbol{\theta}$ is contained in a set $\mathcal{X}$ of n samples $\mathbf{x}_1, \ldots, \mathbf{x}_n$ drawn independently from the mixture density

$$p(\mathbf{x} \mid \boldsymbol{\theta}) = \sum_{j=1}^{c} p(\mathbf{x} \mid \omega_j, \boldsymbol{\theta}_j) P(\omega_j). \tag{1}$$

At this point we could go directly to the calculation of $p(\boldsymbol{\theta} \mid \mathcal{X})$. However, let us first see how this density is used to determine the Bayes classifier. Suppose that a state of nature is selected with probability $P(\omega_i)$ and a feature vector $\mathbf{x}$ is selected according to the probability law $p(\mathbf{x} \mid \omega_i, \boldsymbol{\theta}_i)$. To derive the Bayes classifier we must use all of the information at our disposal to compute the a posteriori probability $P(\omega_i \mid \mathbf{x})$. We exhibit the role of the samples explicitly by writing this as $P(\omega_i \mid \mathbf{x}, \mathcal{X})$. By Bayes rule,

$$P(\omega_i \mid \mathbf{x}, \mathcal{X}) = \frac{p(\mathbf{x} \mid \omega_i, \mathcal{X}) P(\omega_i \mid \mathcal{X})}{\sum_{j=1}^{c} p(\mathbf{x} \mid \omega_j, \mathcal{X}) P(\omega_j \mid \mathcal{X})}.$$

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.