Maximum Likelihood Estimation of Mixture Densities for Binned and Truncated Multivariate Data

Technical Report No. 99–13

Information and Computer Science Department,
University of California, Irvine

I. V. Cadez[1], P. Smyth[1], G. J. McLachlan[2] and C. E. McLaren[3]

[1] Department of Information and Computer Science,
University of California, Irvine, CA 92697, U.S.A.

[2] Department of Mathematics,
The University of Queensland,
Brisbane, Australia

[3]Division of Epidemiology, Department of Medicine,
University of California, Irvine, CA 92697, U.S.A.

March 1999

Abstract

Binning and truncation of data is common in data analysis and machine learning. This paper addresses the problem of fitting mixture densities to multivariate binned and truncated data. The EM approach proposed by McLachlan and Jones (1988) for the univariate case is generalized to multivariate measurements. The multivariate solution requires the evaluation of multidimensional integrals over each bin at each iteration of the EM procedure. Naive implementation of the procedure can lead to computationally inefficient results. To reduce the computational cost a number of straightforward numerical techniques are proposed. Results on simulated data indicate that the proposed methods can achieve significant computational gains with no loss in the accuracy of the final parameter estimates. Furthermore, experimental results suggest that with a sufficient number of bins and data points it is possible to estimate the true underlying density almost as well as if the data had not been binned. The paper concludes with a brief description of an application of this approach to diagnosis of iron deficiency anemia, in the context of binned and truncated bivariate measurements of volume and hemoglobin concentration from an individual's red blood cells.

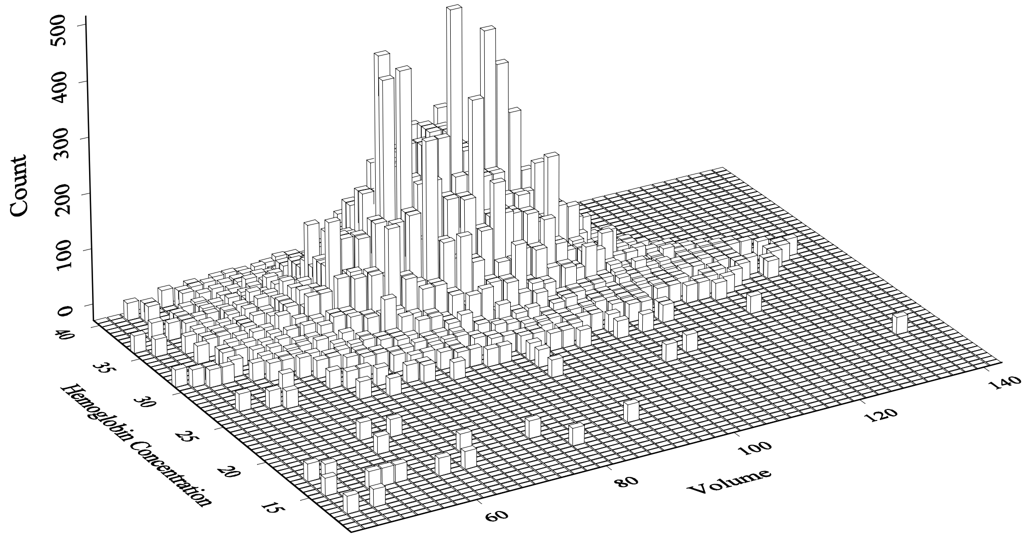

Figure 1: Example of a bivariate histogram for red blood cell data.

1 Introduction

In this paper we address the problem of fitting mixture densities to multivariate binned and truncated data. The problem is motivated by an application in medical diagnosis where blood samples are taken from subjects. Typically each sample contains about 40,000 different red blood cells. The volume and hemoglobin concentration of the red blood cells are measured by a cytometric blood cell counter. It then produces as output a bivariate histogram on a 100 × 100 grid in volume and hemoglobin concentration space (e.g., Figure 1). Each bin contains a count of the number of red blood cells whose volume and hemoglobin concentration fall into that bin. It is known that the data can be truncated, i.e., that the range of machine measurement is less than the actual possible range of volume and hemoglobin concentration values.

We present a general solution to the problem of fitting a multivariate mixture density model to binned and truncated data. Binned and truncated data arise frequently in a variety of application settings. Binning can occur systematically when a measuring instrument has finite resolution, e.g., a digital camera with finite precision for pixel intensity. Binning also may occur intentionally when real-valued variables are quantized to simplify data collection, e.g., binning of a person's age into the ranges 0–10, 10–20, and so forth. Truncation can also easily occur in a practical data collection context, whether due to fundamental limitations on the range of the measurement process or intentionally for other reasons.

For both binning and truncation, one can think of the original "raw" measurements as being masked by the binning and truncation processes, i.e., we do not know the exact location of data points within the bins or how many data points fall outside the measuring range. It is natural to think of this problem as one involving missing data and the Expectation-Maximization (EM) algorithm is an obvious candidate for model fitting in a probabilistic context.

The theory for fitting finite mixture models to univariate binned and truncated data by maximum likelihood via the EM algorithm was developed in McLachlan and Jones (1988). The problem in somewhat simpler form was addressed earlier by Dempster, Laird, and Rubin (1977) when the EM algorithm was originally introduced. The univariate theory of McLachlan and Jones (1988) can be extended in a straightforward manner to cover multivariate data. However, the implementation is subject to exponential time complexity and numerical instability. This requires careful consideration and is the focus of this present paper. In Section 2 we extend the McLachlan and Jones results on univariate mixture estimation to the multivariate case. In Section 3 we present a detailed discussion of the computational and numerical considerations necessary to make the algorithm work in practice. Section 4 discusses experimental results on both simulation data and the afore-mentioned red blood cell data.

2 Basic Theory of EM with Bins and Truncation

We begin with a brief review of the EM algorithm. In the most general form, the EM algorithm is an general procedure for finding maximum likelihood model parameters if some part of the data is missing. For a finite mixture model the underlying assumption (the generative model) is that each data point comes from one of g component distributions. However, this information is hidden in that the identity of the component which generated each point is unknown. If we knew this information, the estimation of the parameters by maximum likelihood would be direct at least for a single normal population; estimate the mean and covariance parameters for each component separately using the data points identified as being from that component. Further, the relative count of data points in each population would be the maximum likelihood estimate of the weight of the components in the mixture model regardless of the component distributions.

We can think of two types of data, the observed data and the missing data. Accordingly, we have the *observed* likelihood (the one we want to maximize), and the *full* likelihood (the one that includes missing data and is typically easier to maximize). The EM algorithm provides a theoretical framework that enables us to iteratively maximize the observed likelihood by maximizing the expected value of the full likelihood. For fitting Gaussian mixtures, the EM iterations are quite straightforward and well-known (see McLachlan and Basford (1988) and Bishop (1995) for tutorial treatments of EM for Gaussian mixtures and see Little and Rubin (1987) or McLachlan and Krishnan (1997) for a discussion of EM in a more general context). With binning and truncation we have two additional sources of hidden information in addition to the hidden component identities for each data point.

McLachlan and Jones (1988) show how to use the EM algorithm for this type of problem. The underlying finite mixture model can be written as:

$$f(x;\Phi) = \sum_{i=1}^{g} \pi_i f_i(x;\theta),$$

where the π_i's are weights for the individual components, the f_i's are the component density functions of the mixture model parametrized by θ, and Φ is the set of all mixture model parameters, $\Phi = \{\pi, \theta\}$. The overall sample space $\mathcal{H}$ is divided into v disjoint subspaces $\mathcal{H}_j$, (bins) of which only the counts on the first r bins are observed, while the counts on last $v - r$ bins are missing. The (observed) likelihood associated with this model (up to irrelevant constant terms) is given by (Jones and McLachlan (1990)):

$$\ln L = \sum_{j=1}^{r} n_j \ln P_j - n \ln P, \tag{1}$$

where n is the total observed count:

$$n = \sum_{j=1}^{r} n_j,$$

and the P's represent integrals of the probability density function (PDF) over bins:

$$\begin{aligned} P_j &\equiv P_j(\Phi) = \int_{\mathcal{H}_j} f(x;\Phi)dx, \\ P &\equiv P(\Phi) = \int_{\mathcal{H}} f(x;\Phi)dx = \sum_{j=1}^{r} P_j \end{aligned}$$

The form of the likelihood function above corresponds to a multinomial distributional assumption on bin occupancy.

To invoke the EM machinery we first define several quantities at the p-th iteration. $\Phi^{(p)}$ and $\theta^{(p)}$ represent current estimates of model parameters. $\mathrm{E}_j^{(p)}[.]$ denotes conditional expectation given that the random variable belongs to the j-th bin, using the current value of the unknown parameter vector (e.g., expected value with respect to the normalized current PDF $f(x;\Phi^{(p)})/P_j(\Phi^{(p)})$). Specifically, for any function $g(x)$:

$$\mathrm{E}_j^{(p)}[g(x)] = \frac{1}{P_j(\Phi^{(p)})} \int_{\mathcal{H}_j} f(x;\Phi^{(p)}) g(x) dx. \tag{2}$$

We also define:

$$\begin{aligned} m_j^{(p)} &= \begin{cases} n_j & j = 1, \ldots, r; \\ n P_j(\Phi^{(p)})/P(\Phi^{(p)}) & j = r+1, \ldots, v; \end{cases} \qquad (3) \\ \tau_i^{(p)}(x) &= \frac{\pi_i f_i(x;\theta^{(p)})}{f(x;\Phi^{(p)})}, \qquad (4) \\ c_i^{(p)} &= \sum_{j=1}^{v} m_j^{(p)} \mathrm{E}_j^{(p)}[\tau_i^{(p)}(x)], \qquad (5) \end{aligned}$$

where all the quantities on the left-hand side (with superscript (p)) depend on the current parameter estimates $\Phi^{(p)}$ and/or $\theta^{(p)}$. Each term has an intuitive interpretation. For example, the m_j's represent a generalization of the bin counts to unobserved data. They are either equal to the actual count in the observed bins (i.e., for $j \leq r$)

or they represent the conditional expected counts for unobserved bins (i.e., $j > r$). The conditional expected count formalizes the notion that if there is (say) 1% of the PDF mass in the unobserved bins, then we should assign them 1% of the total data points. $\tau_i(x)$ is the posterior probability of membership of the i-th component of the mixture model given x being observed on the individual (it represents the relative weight ($\sum_{i=1}^{g} \tau_i(x) = 1$) of each mixture component i at point x). Intuitively it is the probability of data point x "belonging" to component i. c_i is a measure of the overall relative weight of component i. Note that in order to calculate c_i the local relative weight $\tau_i(x)$ is averaged over each bin, weighted by the count in the bin and summed over all bins. This way, each data point within each bin contributes to c_i an *average* local weight for that bin (i.e. $\mathrm{E}_j[\tau_i(x)]$). Compare this to the non-binned data where each data point contributes to c_i the *actual* local weight evaluated at the data point (i.e., $\tau_i(x_k)$, where x_k is the value of the data point).

Next, we use the quantities defined in the last equation to define the E-step and express the closed form solution for the M-step at iteration $(p+1)$:

$$\pi_i^{(p+1)} = \frac{c_i^{(p)}}{\sum_{j=1}^{v} m_j^{(p)}}, \tag{6}$$

$$\mu_i^{(p+1)} = \frac{\sum_{j=1}^{v} m_j^{(p)} \mathrm{E}_j^{(p)}[x\tau_i^{(p)}(x)]}{c_i^{(p)}}, \tag{7}$$

$$\left[\sigma_i^{(p+1)}\right]^2 = \frac{\sum_{j=1}^{v} m_j^{(p)} \mathrm{E}_j^{(p)}[(x-\mu_i^{(p+1)})^2\tau_i^{(p)}(x)]}{c_i^{(p)}}. \tag{8}$$

These equations specify how the component weights (i.e., π's), component means (i.e., μ's) and component standard deviations (i.e., σ's) are updated at each EM step. Note that the main difference here from the standard version of EM (for non-binned data) comes from the fact that we are taking expected values over the bins (i.e., $\mathrm{E}_j^{(p)}[.]$). Here, each data point within each bin contributes the corresponding value averaged over the bin, whereas in the non-binned case each point contributes the same value but evaluated at the data point.

To generalize to the multivariate case, in theory all we need do is generalize Equations (6)-(8) to the vector/covariance cases:

$$\pi_i^{(p+1)} = \frac{c_i^{(p)}}{\sum_{j=1}^{v} m_j^{(p)}}, \tag{9}$$

$$\boldsymbol{\mu}_i^{(p+1)} = \frac{\sum_{j=1}^{v} m_j^{(p)} \mathrm{E}_j^{(p)}[\mathbf{x}\tau_i^{(p)}(\mathbf{x})]}{c_i^{(p)}}, \tag{10}$$

$$\Sigma_i^{(p+1)} = \frac{\sum_{j=1}^{v} m_j^{(p)} \mathrm{E}_j^{(p)}[(\mathbf{x}-\boldsymbol{\mu}_i^{(p+1)})(\mathbf{x}-\boldsymbol{\mu}_i^{(p+1)})^{+}\tau_i^{(p)}(\mathbf{x})]}{c_i^{(p)}}. \tag{11}$$

While the multivariate theory is a straightforward extension of the univariate case, the practical implementation of this theory is considerably more complex due to the fact that the approximation of multi-dimensional integrals is considerably more complex than the univariate case.

Note that the approach above is guaranteed to maximize likelihood as defined by equation (1), irrespective of the form of the selected conditional probability model for missing data given observed data. Different choices of this conditional probability model only lead to different paths in parameter space, but the overall maximum likelihood parameters will be the same. This makes the approach quite general as no additional assumptions about the distribution of the data are required.

3 Computational and Numerical Issues

In this section we discuss our approach to two separate problems that arise in the multivariate case: 1) how to perform a single iteration of the EM algorithm; 2) how to setup a full algorithm that will be both exact and time efficient. The main difficulty in handling binned data (as opposed to having standard, non-binned data) is the evaluation of the different expected values (i.e. $\mathrm{E}_j^{(p)}[.]$) at each EM iteration. As defined by Equation (2), each expected value in equations (9)-(11) requires integration of some function over each of the v bins. These integrals cannot be evaluated analytically for most mixture models (even for Gaussian mixture models). Thus, they have to be evaluated numerically at each EM iteration, considerably complicating the implementation of the EM procedure, especially for multivariate data. To summarize we present some of the difficulties:

- If there are m bins in the univariate space, there are now $O(m^d)$ bins in the d-dimensional space (consider each dimension as having $O(m)$ bins), which represents exponential growth in the number of bins.

- If in the univariate space each numerical integration requires $O(i)$ function evaluations, in multivariate space it will require at least $O(i^d)$ function evaluations for comparable accuracy of the integral. Combined with the exponential growth in the number of bins, this leads to an exponential growth in number of function evaluations. While the underlying exponential complexity cannot be avoided, the overall execution time can greatly benefit from carefully optimized integration schemes.

- The geometry of multivariate space is more complex than the geometry of univariate space. Univariate histograms have natural end-points where the truncation occurs and the unobserved regions have a simple shape. Multivariate histograms typically represent hypercubes and unobserved regions, while still "rectangular," are not of a simple shape any more. For example, for a 2-dimensional histogram there are four sides from which the unobserved regions extend to infinity, but there are also four "wedges" in between these regions.

- For fixed sample size, multivariate histograms are much sparser than their univariate counterparts in terms of counts per bin (i.e., marginals). This sparseness can be leveraged for the purposes of efficient numerical integration.

3.1 Numerical Integration at each EM Iteration

The E step of the EM algorithm consists of finding the expected value of the complete-data log likelihood with respect to the distribution of missing data, while the M step consists of maximizing this expected value with respect to the model parameters Φ. Equations (9)-(11) summarize both steps for a single iteration of the EM algorithm. If there were no expected values in the equations (i.e., no $\mathrm{E}_j^{(p)}[.]$ terms), they would represent a closed form solution for parameter updates. With binned and truncated data, they are almost a closed form solution, but additional integration is still required. One could use any of a variety of Monte Carlo integration techniques for this integration problem. However, the slow convergence of Monte Carlo is undesirable for this problem. Since the functions we are integrating are typically quite smooth across the bins, relatively straightforward numerical integration techniques can be expected to give solutions with a high degree of accuracy.

Multidimensional numerical integration consists of repeated 1-dimensional integrations. For the results in this paper we use Romberg integration (see Thisted (1988) or Press et al., (1992) for details). An important aspect of Romberg integration is selection of the *order* of integration. Lower-order schemes use relatively few function evaluations in the initialization phase, but may converge slowly. Higher-order schemes may take longer at the initialization phase, but converge faster. Thus, order selection can substantially affect the computation time of numerical integration (we will return to this point later). Note that the order only affects the path to convergence of the integration; the final solution is the same for any order given the same pre-specified degree of accuracy.

3.2 Handling Truncated Regions

The next problem that arises in practice concerns the truncated regions (i.e., regions outside the measured grid). If we want to use a mixture model that is naturally defined on the whole space we must define bins to cover regions extending from grid boundaries to ∞. In the 1-dimensional case it suffices to define 2 additional bins: one extending from the last bin to ∞, and the other extending from $-\infty$ to the first bin. In the multivariate case it is more natural to define a single bin

$$\mathcal{H}_{r+1} = \mathcal{H} \setminus \bigcup_{j=1}^{r} \mathcal{H}_j$$

that covers everything but the data grid than to explicitly describe the out-of-grid regions. The reason is that we can calculate all the expected values over the whole space $\mathcal{H}$ without actually doing any integration. With this in mind, we readily write for the integrals over the truncated regions:

$$\int_{\mathcal{H}_{r+1}} f(\mathbf{x};\Phi)d\mathbf{x} = 1 - \sum_{j=1}^{r} P_j(\Phi) \tag{12}$$

$$\int_{\mathcal{H}_{r+1}} f_i(\mathbf{x};\theta)\mathbf{x}d\mathbf{x} = \boldsymbol{\mu}_i - \sum_{j=1}^{r} \int_{\mathcal{H}_j} f_i(\mathbf{x};\theta)\mathbf{x}d\mathbf{x} \tag{13}$$

$$\begin{aligned}\int_{\mathcal{H}_{r+1}} & f_i(\mathbf{x};\theta)(\mathbf{x}-\boldsymbol{\mu}_i)(\mathbf{x}-\boldsymbol{\mu}_i)^+ d\mathbf{x} = \\ &= \Sigma_i - \sum_{j=1}^{r} \int_{\mathcal{H}_j} f_i(\mathbf{x};\theta)(\mathbf{x}-\boldsymbol{\mu}_i)(\mathbf{x}-\boldsymbol{\mu}_i)^+ d\mathbf{x} \qquad (14)\end{aligned}$$

Note that no extra work is required to obtain the integrals on the right-hand side of the equations above. The basic EM Equations (9)-(11) require the calculation of expected values similar to those defined in Equation (2) for each bin. Note, however, that the only difference between those expected values and integrals on the right-hand side of Equations (12)-(14) is the normalizing constant $1/P_j(\Phi)$. Because the normalizing constant does not affect the integration, it suffices to separately record normalized and unnormalized values of integrals for each bin. The normalized values are later used in equations (9)-(11), while the unnormalized values are used in Equations (12)-(14).

For computational efficiency we take advantage of the sparseness of the bin counts. Assume that we want to integrate some function (i.e., the PDF) over the whole grid. Further assume that we require some prespecified accuracy of integration δ. This means that if the relative change of the value of the integral in two consecutive iterations falls below δ we consider the integral to have converged. δ is a small number, typically of the order of 10^{-5} or less. Assume further that we perform integration by integrating over each bin on the grid and by adding up the results. Intuitively, the contribution from some bins will be large (i.e., from the bins with significant PDF mass in them), while the contribution from others will be negligible (i.e., from the bins that contain near zero PDF mass). If the data are sparse, there will be many bins with negligible contributions. The goal is to optimize the time spent on integrating over numerous empty bins that do not significantly contribute to the integral or the accuracy of the integration.

To see how this influences the overall accuracy, consider the following simplified analysis. Let the size of the bins be proportional to H and let the mean height of the PDF be approximately F. Let there be of the order pN bins with relevant PDF mass in them, where $p < 1$ and N is the total number of bins. A rough estimate of the integral over all bins is given by $I \sim FHpN$. Since the accuracy of integration is of order δ, we are tolerating absolute error in integration of order δI. On the other hand, assume that in the irrelevant bins the value of the PDF has height on the order of ϵF, where ϵ is some small number. The estimated contribution of the irrelevant bins to the value of the integral is $I' \sim \epsilon FH(1-p)N$ which is approximately $I' \sim \epsilon/pI$ for sparse data (i.e., p is small compared to 1). The estimated contribution of the irrelevant bins to the absolute error of integration is $\delta' I' = \delta'\epsilon/pI$, where δ' is accuracy of integration within irrelevant bins. Since any integration is as accurate as its least accurate part, in an optimal scheme the contribution to the error of integration from the irrelevant and relevant bins are comparable. In other words, it is suboptimal (as we also confirm experimentally in the result section) to choose δ' any smaller than required by $\delta'\epsilon/p \sim \delta$. This means that integration within any bin with low probability mass (i.e. $\sim \epsilon F$) need not be carried out more accurately than $\delta' \sim \delta p/\epsilon$.

Note that as $\epsilon \to 0$ we can integrate less and less accurately within each bin without hurting the overall integral over the full grid. Note also that as $\epsilon \to 0$ and δ'

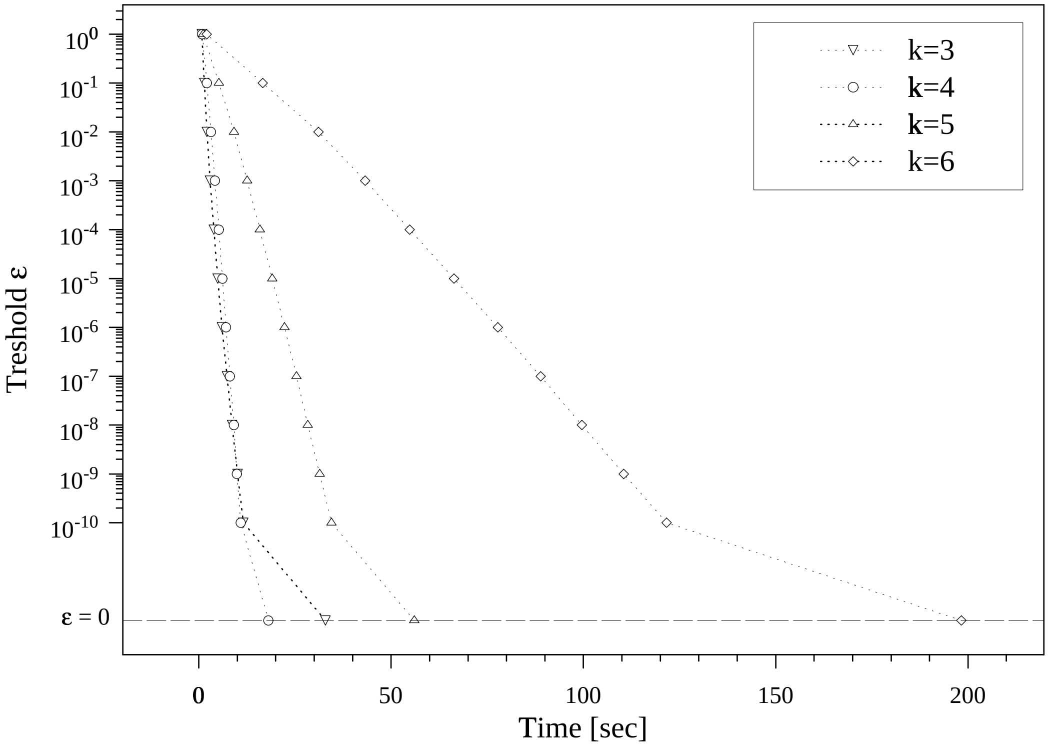

Figure 2: Execution time of a single EM step as a function of the threshold ϵ for several different values k of the Romberg integration order. For $k = 2$, the values were off-scale, e.g., $t(1) = 23, t(0.1) = 363$, etc. Results based on fitting a two-component mixture to 40,000 red blood cell measurements in two dimensions on 100 × 100 bins.

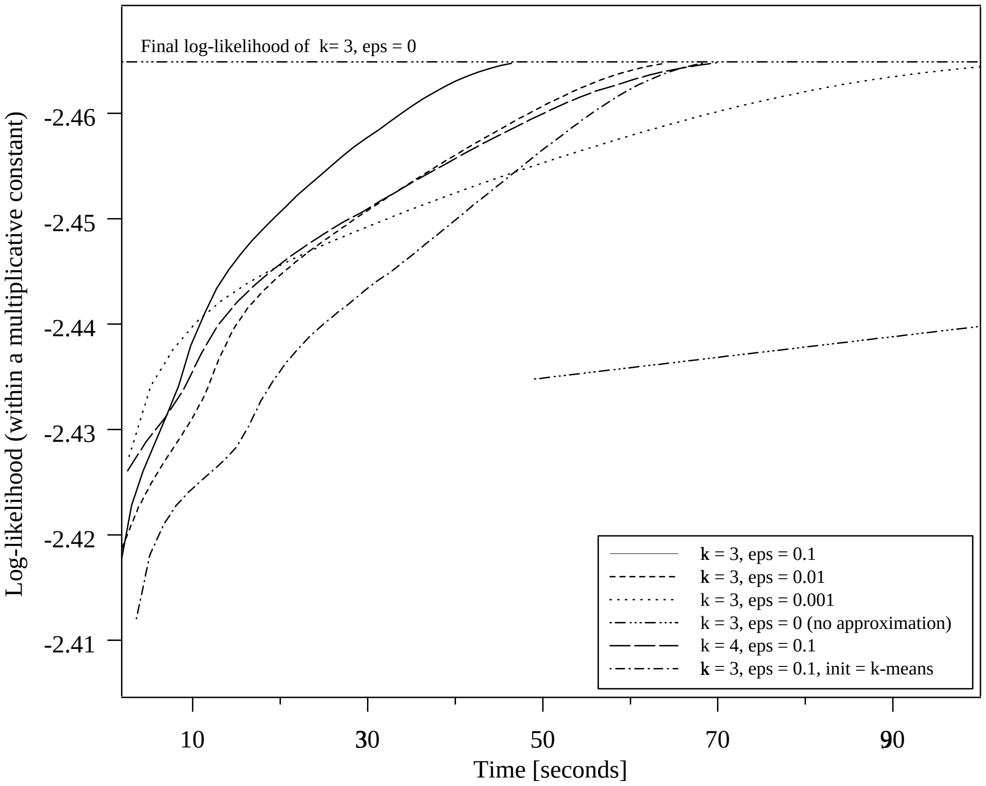

Figure 3: Quality of solution (measured by log-likelihood) as a function of time for different variations on the algorithm.

becomes $o(1)$, we can start using a single iteration of the simplest possible integration scheme and still stay within the allowed limit of δ'. To summarize, given a value for ϵ, the algorithm estimates the average height F of the PDF and for all the bins with PDF values less than ϵF uses a single iteration of a simple and fast integrator. The original behavior is recovered by setting $\epsilon = 0$ (i.e. no bins are integrated "quickly"). This general idea provides a large computational gain with virtually no loss of accuracy (note that δ controls overall accuracy, while ϵ adds only a small correction to δ). For example, we have found that the variability in parameter estimates from using different small values of ϵ is much smaller than the bin size and/or the variability in parameter estimates from different (random) initial conditions.

Figure 2 shows the time required to complete a single EM step for different values of k (the Romberg integration order) and ϵ. The time is minimized for different values of ϵ by using $k = 3$ or $k = 4$, and is greatest for $k = 2$ (off-scale) and $k = 6$, i.e., choosing either too low or too high of an integration order is quite computationally inefficient.

3.3 The Full EM Algorithm

After fine tuning each single EM iteration step above we are able to significantly cut down on the execution time. However, since each step is still computationally intensive, it is desirable to have EM converge as quickly as possible (i.e., to have as few iterations as possible).

With this in mind we use the following additional heuristic. We take a random sample of binned points and randomize the coordinates of each point around the corresponding bin center (we use the uniform distribution within each bin). The EM algorithm for this non-binned and non-truncated data is relatively fast as a closed form solution exists for each EM step (without any integration). Once the EM algorithm converges to a solution in parameter space on this initial data set, we use these parameters as initial starting points for the EM algorithm on the full set of binned and truncated data. This second application of EM (using the methodology described earlier in this paper) refines the initial guesses to a final solution, typically taking just a few iterations. Note that this initialization scheme cannot affect the accuracy of the results, as the log-likelihood on the full set of binned and truncated data is used as the final criterion for convergence.

Figure 3 illustrates the various computational gains. The y axis is the log-likelihood (within a multiplicative constant) of the data and the x axis is computation time. Here we are fitting a two-component mixture on a two-dimensional grid with 100×100 bins of red blood cell counts. k is the order of Romberg integration and ϵ is the threshold for declaring a bin to be small enough for "fast integration" as described earlier. All parameter choices (k, ϵ) result in the same quality of final solution (i.e., all asymmptote to the same log-likelihood eventually) Using no approximation ($\epsilon = 0$) is two orders of magnitude slower than using non-zero ϵ values. Increasing ϵ from 0.001 to 0.1 results in no loss in likelihood but results in faster convergence. Comparing the curves for $k = 3, \epsilon = 0.1$ where k-means is used to initialize the binned algorithm versus the randomized initialization method described earlier, shows about a factor of two gain in convergence time for the randomized initialization.

To summarize, the overall algorithm for fitting mixture models to multivariate binned, truncated data consist of the following steps:

1. Treat the multivariate histogram as a PDF and draw a small number of data points from it (add some counts to all the bins to prevent 0 probabilities in empty bins).

2. Fit a standard mixture model to this sample using the usual EM algorithm for non-binned, non-truncated data.

3. Use the parameter estimates from Step 2, and refine them using the EM algorithm on the full set of binned and truncated data. This consists of iteratively applying equations (9)-(11) for the bins within the grid and applying equations (12)-(14) for the single bin outside the grid until convergence as measured by equation (1).

4 Experimental Results

4.1 EM Methodology

In the experiments below we use the following methods and parameters in the implementation of EM.

- The standard EM algorithm is initialized by running k-means from 10 different initial starting points and choosing the EM solution with the highest likelihood (to avoid poor local maxima).
- K points are randomly drawn from the binned histogram, where K is chosen to be 10% of the number of total data points or 100 points, whichever is greater. Points are drawn using the uniform sampling distribution.
- The binned EM algorithm is initialized by running the standard EM algorithm with 5 random restarts on the K randomly drawn data points.
- To avoid poor local maxima, the binned EM algorithm chooses the solution with the highest likelihood out of solutions from 10 different random initializations.
- Convergence of the standard and binned/truncated EM is judged by a change of less the 0.01% in the log-likelihood, or after a maximum of 20 EM iterations, whichever comes first.
- The order of the Romberg integration is set to 3 and ϵ is set to 10^{-4}.
- The default accuracy of the integration is set to $\delta = 10^{-5}$.

4.2 Simulation Experiments

We simulated data from a two-dimensional mixture of two Gaussians, centered at (-1.5,0) and (1.5,0) with unit covariance matrices. We then varied the number of data points per dimension in steps of 10 from $N = 10$ to 1000, and drew a 10 random samples of size N from the bivariate mixture. In addition we varied the number of bins per dimension in steps of 5 from $B = 5$ to $B = 100$ so that the original unbinned samples were quantized into B^2 bins. The range of the grid extended from (-5,-5) to (5,5) so that truncation was relatively rare.

On the original unbinned samples we ran the standard EM algorithm, and on the binned data we ran the binned version of EM (using for both versions of the EM the parameters and settings described earlier). The purpose of the simulation was to observe the effect of binning and sample size on the quality of the solution. Note that the standard algorithm is typically being given much more information about the data (i.e., the exact locations of the data points) and, thus, on average we expect it to perform better than any algorithm which only has binned data to learn from. To measure solution quality we calculated the Kullback-Leibler (K-L) (or cross-entropy) distance between each estimated density and the true known density. The

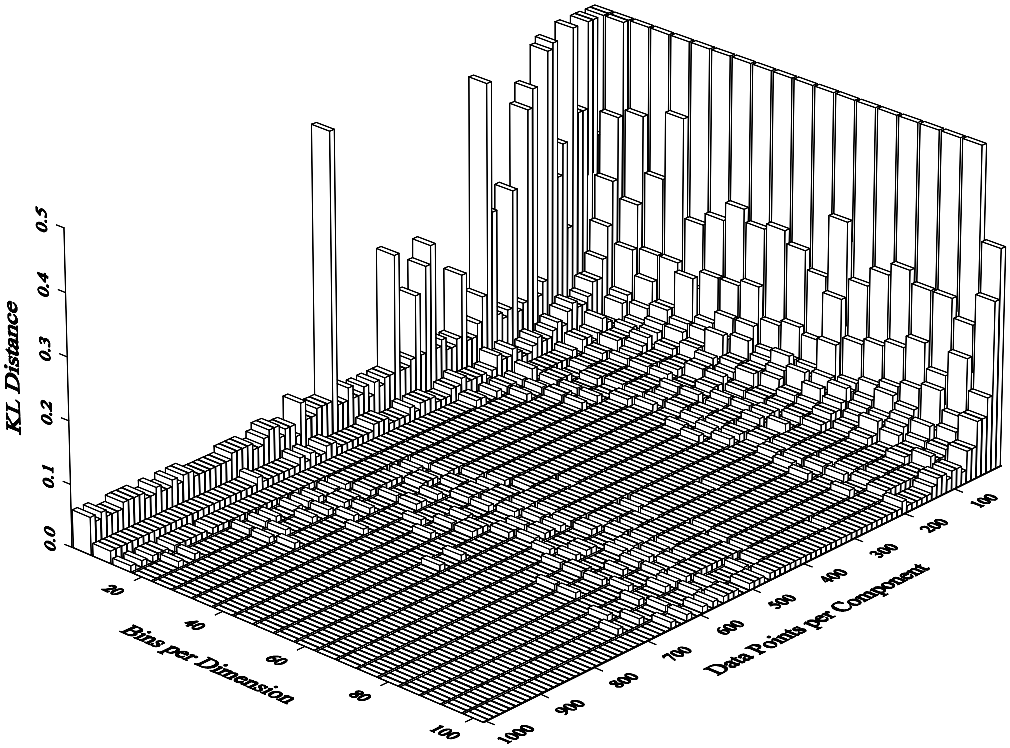

Figure 4: Average KL distance between the estimated density (estimated using the procedure described in this paper) and the true density, as a function of the number of bins and the number of data points.

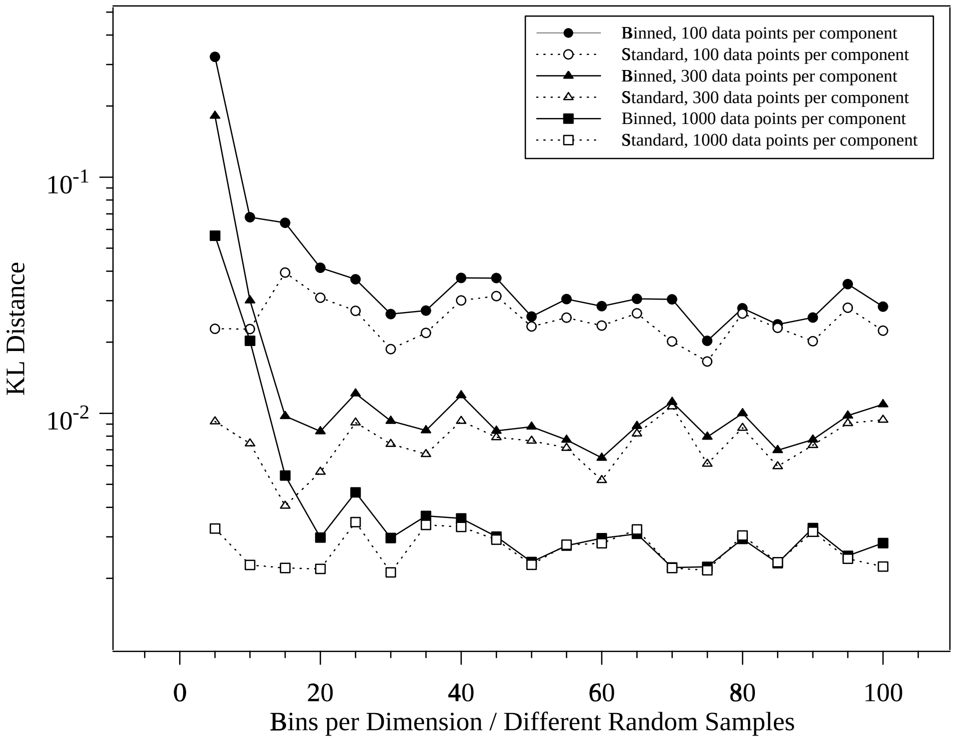

Figure 5: Average KL distance (log-scale) between the estimated densities and the true density as function of the number of bins, for different sample sizes, and compared to standard EM on the unbinned data.

K-L distance is non-negative and is zero if and only if two densities are identical. We calculated the average K-L distance over the 10 samples for each value of N and B, for both the binned and the standard EM algorithms. In total, each of the standard and binned algorithms were run 20,000 different times to generate the reported results.

Figure 4 shows plot of average KL-distance for the binned EM algorithm, as a function of the number of bins and the number of data points. One can clearly see a "plateau" effect in that the KL-distance between solution and the generating true density (a measure of quality of the solution) is relatively close to zero when the number of bins is above 20 and the number of data points is above 500. As a function of N, the number of data points, one sees the typical exponentially decreasing "learning curve,", i.e., solution quality increases roughly in proportion to $N^{-\alpha}$ for some constant α. As a function of bin size B, there appears to be more of a threshold effect: with more than 20 bins the solution quality is again relatively flat as a function of the number of bins. Below $B = 20$ the solutions rapidely decrease in quality (e.g., for $B = 5$ there is a significant degradation).

In Figure 5 we plot the KL distance (log-scale) as a function of bin size, for specific values of N ($N = 100, 300, 1000$), comparing both the standard and binned versions of EM. For each of the 3 values of N, the curves have the same qualitative shape: a rapid improvement in quality as we move from $B = 5$ to $B = 20$, with relatively flat performance (i.e., no sensitivity to B) above $B = 20$. For each of the 3 values of N, the binned EM "tracks" the performance of the standard EM quite closely: the

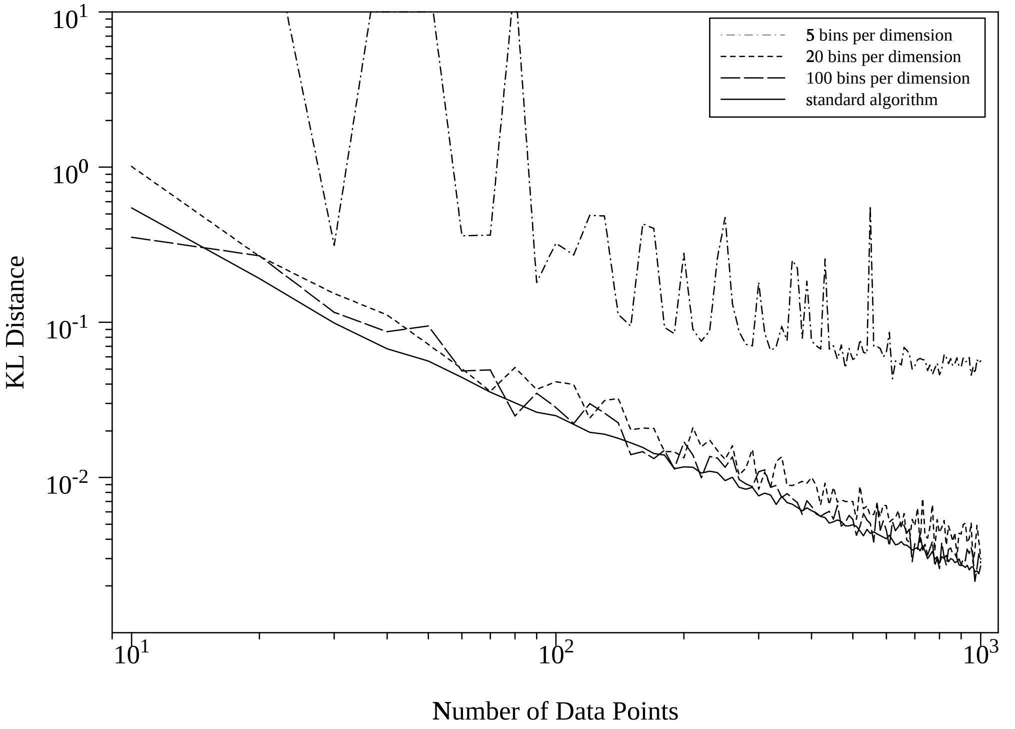

Figure 6: Average KL distance (log-scale) between the estimated densities and the true density as function of sample size, for different numbers of bins, and compared to standard EM on the unbinned data.

difference between the two becomes less as N increases. The variability in the curves is due to the variability in the 10 randomly sampled data sets for each particular value of B and N. Note that for $B \geq 20$ the difference between the binned and standard versions of EM is smaller than the "natural" variability due to random sampling effects.

Figure 6 plots the average KL distance (log-scale) as a function of N, the number of data points per dimension, for specific numbers of bins B. Again we compare the binned algorithm (for various B values) with the standard unbinned algorithm. Overall we see the characteristic exponential decay (linear on a log-log plot) for learning curves as a function of sample size. Again, for $B \geq 20$ the binned EM tracks the standard EM quite closely.

The results suggest (on this particular problem at least) that the EM algorithm for binned data is more sensitive to the number of bins than it is to the number of data points, in terms of comparative performance to EM on unbinned data. Above a certain threshold number of bins (here $B = 20$), the binned version of EM appears to be able to recover the true shape of the densities almost as well as the version of EM which sees the original unbinned data.

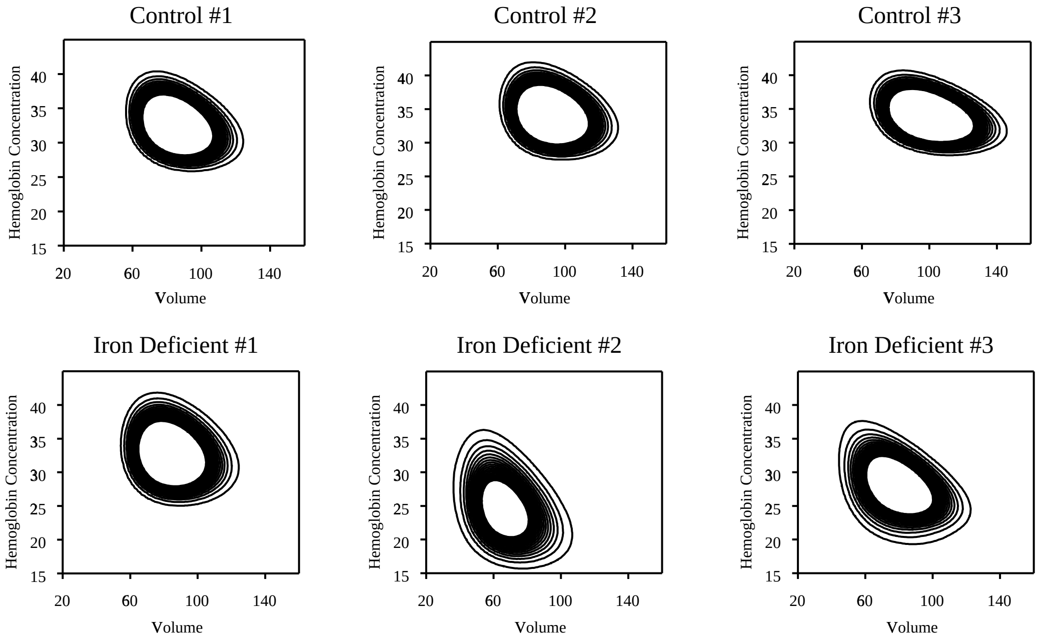

Figure 7: Contour plots from estimated density estimates for three typical control patients and three typical iron deficient anemia patients. The lowest 10% of the probability contours are plotted to emphasize the systematic difference between the two groups.

4.3 Application to Red Blood Cell Data

As mentioned at the beginning of the paper this work was motivated by a real-world application in medical diagnosis based on two-dimensional histograms characterizing red blood cell volume and hemoglobin measurements (see Figure 1).

McLaren (1996) summarizes prior work on this problem: the one-dimensional mixture-fitting algorithm of McLachlan and Jones (1988) was used to fit mixture models to one-dimensional red blood cell volume histograms. Mixture models are particularly useful in this context as a generative model since it is plausible that different components in the model correspond to blood cells in different states. In Cadez et al (1999) we generalized the earlier work of McLaren et al (1991) and McLaren (1996) on one-dimensional volume data to the analysis of two-dimensional volume-hemoglobin histograms. Mixture densities were fit to histograms from 97 control subjects and 83 subjects with iron deficient anemia, using the binned/truncated EM procedure described in the present paper. Figure 3 demonstrated the improvement in computation time which is achievable; the data in Figure 3 are for a 2-component mixture model fit to a control subject with a 2-dimensional histogram of 40,000 red blood cells.

Figure 7 shows contour probability plots of fitted mixture densities for 3 control and 3 iron deficient subjects, where we plot only the lower 10% of the probability density function (since the differences between the two populations are more obvious in the tails). One can clearly see systematic variability within the control and the iron deficient groups, as well as between the two groups. Since the number of bins is relatively large ($B = 100$ in each dimension), as is the number of data points (40,000), the simulation results from the previous section would tend to suggest that these density estimates are likely to be relatively accurate (compared to running EM on unbinned data).

In Cadez et al (1999) we used the parameters of the estimated mixture densities as the basis for supervised classification of subjects into the two groups, with a resulting error rate of about 1.5% in cross-validated experiments. This compares with a cross-validated error rate of about 4% on the same subjects using algorithms such as CART or C5.0 directly on features from the histogram such as univariate means and standard deviations (i.e., using no mixture modeling). Thus, the ability to fit mixture densities to binned and truncated data played a significant role in improved classification performance on this particular problem.

5 Conclusions

The problem of fitting mixture densities to multivariate binned and truncated data was addressed using a generalization of McLachlan and Jones' (1988) EM procedure for the one-dimensional problem. The multivariate EM algorithm requires multivariate numerical integration at each EM iteration. We described a variety of computational and numerical implementation issues which need careful consideration in this context. Simulation results indicate that high quality solutions can be obtained compared to running EM on the "raw" unbinned data, unless the number of bins is relatively small.

Acknowledgements

The contributions of IC and PS to this paper have been supported in part by the National Science Foundation under Grant IRI-9703120. The contribution of CMcL has been supported in part by grants from the National Institutes of Health (R43-HL46037 and R15-HL48349) and a Wellcome Research Travel Grant awarded by the Burroughs Wellcome Fund. We thank Thomas H. Cavanagh for providing laboratory facilities. We are grateful to Dr. Albert Greenbaum for technical assistance.

References

Bishop, C. M. (1995), *Neural Networks for Pattern Recognition*, Oxford, UK: Clarendon Press.

Cadez, I. V., McLaren, C. E., Smyth, P., and McLachlan, G. J. (1999) 'Hierarchical models for screening of iron deficiency anemia,' submitted to ICML-99, January 1999.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977) 'Maximum likelihood from incomplete data via the EM algorithm,' *J. Royal Stat. Soc. B*, 39(1), 1–38.

Jones, P. N. and McLachlan, G. J. (1990) ‘Maximum Likelihood Estimation from Grouped and Truncated Data with Finite Normal Mixture Models,’ *Applied Statistics-Journal of the Royal Statistical Society Series C*, 39(N2):273-282.

Little, R. J. A. and Rubin, D. B. (1987) *Statistical Analysis with Missing Data*, New York: John Wiley.

McLachlan, G. J. and Jones, P. N. (1988) ‘Fitting mixture models to grouped and truncated data via the EM algorithm,‘ *Biometrics*, 44(2):571-8.

McLachlan, G. J., and Krishnan, T. (1997) *The EM Algorithm and Extensions*, New York: John Wiley and Sons.

McLaren, C. E. (1996) ‘Mixture models in haematology: a series of case studies,’ *Statistical Methods in Medical Research*,

McLaren, C. E., Wagstaff, M., Brittenham, G. M., Jacobs, A. (1991) ‘Detection of Two-Component Mixtures of Lognormal Distributions in Grouped, Doubly Truncated Data: Analysis of Red Blood Cell Volume’ Distributions,’ *Biometrics*, 47(2):607-22.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., Flannery, B. P. (1996) *Numerical Recipes in C: the Art of Scientific Computing*, 2nd edition, Cambridge, UK: Cambridge University Press.

Thisted, R. A. (1988) *Elements of Statistical Computing*, London: Chapman and Hall.