Machine Learning, 47, 7–34, 2002


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

IGOR V. CADEZ icadez@ics.uci.edu
PADHRAIC SMYTH smyth@ics.uci.edu
*Department of Information and Computer Science, University of California, Irvine, CA 92697, USA*

GEOFF J. McLACHLAN
*Department of Mathematics, The University of Queensland, Brisbane, Australia*

CHRISTINE E. McLAREN
*Division of Epidemiology, Department of Medicine, University of California, Irvine, CA 92697, USA*

**Editor:** Douglas Fisher

**Abstract.** Binning and truncation of data are 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 (*Biometrics, 44: 2*, 571–578, 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 were not 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.

**Keywords:** EM, binned, truncated, histogram, mixture model, KL-distance, iron deficiency anemia

## 1. Introduction

In this paper we address the problem of fitting mixture densities to multivariate binned and truncated data. The problem is motivated by the problem of developing automated techniques for detection and classification of anemia. Over one billion people in the world are anemic and at risk for major liabilities. In the case of iron deficiency anemia, for example, these liabilities include mental and motor developmental defects in infants and weakness, weight loss, and impaired work performance in adults (Osaki, 1993; Basta et al., 1979; Edgerton et al., 1979). For diagnostic evaluation of anemia and monitoring the response to therapy, blood samples from patients are routinely analyzed to determine the volume

of the red blood cells (RBCs) and the amount of hemoglobin, the oxygen-transporting protein of the red cell. Many anemia-related diseases are known to manifest themselves via fundamental changes in the univariate volume distribution and the univariate hemoglobin concentration of RBCs (Williams et al., 1990; McLaren, 1996).

Automated techniques have been recently developed which can simultaneously measure both volume and hemoglobin concentration of RBCs from a patient's blood sample. Flow cytometric blood cell counting instruments (Technicon H*1, H*2, H*3; Bayer Corporation, White Plains, NY) make measurements using a laser light scattering system to produce as output a bivariate histogram on a $100 \times 100$ grid (known as a *cytogram*) in RBC volume and hemoglobin concentration space (e.g., figure 1). Typically measurements are made on about 40,000 different red blood cells from a blood sample. Each bin in the histogram contains a count of the number of red blood cells whose volume and hemoglobin concentration are in the range defined by the bin. The data can also be truncated, i.e., the range of machine measurement is less than the actual possible range of volume and hemoglobin concentration values.

Current methods to differentiate between disorders of anemia on the basis of RBC measurements are largely based on visual inspection of printed output of an individual's bivariate volume-hemoglobin histogram, as produced by the flow cytometric instrument. In this context it would be highly cost-effective to have the ability to perform automated low-cost accurate diagnostic screening of blood-related disorders using RBC measurements. In Cadez et al. (1999) we presented a classification model for iron deficiency anemia diagnosis which achieved of the order of 98% cross-validated classification accuracy for this problem. An important component of the proposed technique was the fitting of a bivariate normal mixture model to the binned and truncated cytogram data for each individual being classified. In this paper we present a general solution to the problem of fitting a multivariate mixture density model to binned and truncated data.

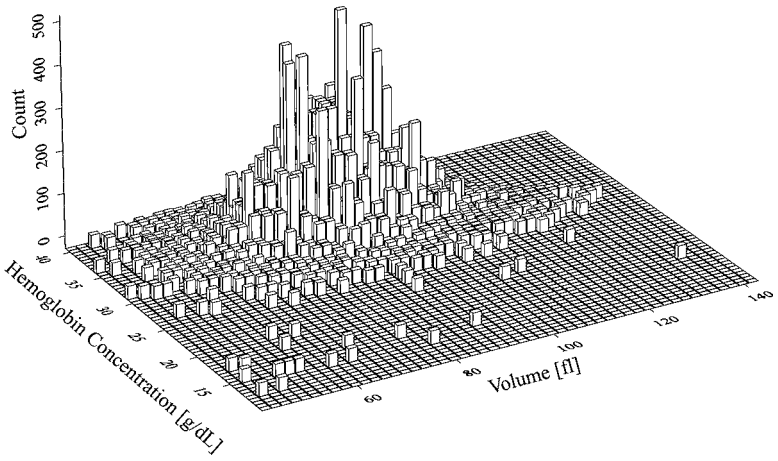



*Figure 1*. Example of a bivariate histogram for red blood cell data.

Data in the form of histograms also play an important role in a variety of other pattern recognition and machine learning problems. For example, in computer vision Swain and Ballard (1991) describe the use of color histograms for object recognition. More recent work in image retrieval relies heavily on the use of color and feature histograms (e.g., Flickner et al., 1995; Maybury, 1997). A number of techniques in approximate querying of databases and in data mining of massive data sets also use histogram representations (e.g., Poosala, 1997; Matias, Vitter, & Wang, 1998; Lee, Kim, & Chung, 1999).

More generally, binned and truncated data arise frequently in a variety of application settings since many measuring instruments produce quantized data. Binning occurs systematically when a measuring instrument has coarse resolution compared to the variance of measured values, 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, i.e., the true values (either within the bin or outside the truncation range) of the points are missing. The Expectation-Maximization (EM) procedure is an obvious candidate for probabilistic model fitting in this context.

The theory of using EM for fitting maximum likelihood finite-mixture models to univariate binned and truncated data 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, although we are unaware of any prior published work which addresses this. The multivariate 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 results of McLachlan and Jones 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 anemia classification problem.

## 2. Basic theory of EM with bins and truncation

We begin with a brief review of the EM procedure. In the most general form, the EM algorithm is an efficient way to find 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 maximum likelihood parameters

would be direct; one could 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 component would be the maximum likelihood estimate of the weight of the components in the mixture model.

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 Bishop, 1995 for a tutorial treatment of EM for Gaussian mixtures and see Little & Rubin, 1987; McLachlan & 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 efficiently 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 $\pi_i$'s are weights for the individual components, the $f_i$'s are the component density functions of the mixture model parameterized by $\theta$, and $\Phi$ is the set of all mixture model parameters, $\Phi = \{\pi, \theta\}$. The overall sample space $\mathcal{H}$ is divided into $v$ disjoint subspaces $\mathcal{H}_j$ (i.e., $v$ bins), of which only the counts on the first $r$ bins are observed, while the counts on the last $v - r$ bins are missing (these are the truncated regions). 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:

$$P_j \equiv P_j(\Phi) = \int_{\mathcal{H}_j} f(x;\Phi)\, dx,$$

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

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 and $\mathrm{E}_j^{(p)}[\cdot]$ represents the expected value within bin $j$ with respect to the (normalized) current PDF $f(x;\Phi^{(p)})/P_j(\Phi^{(p)})$. Specifically, for any function $g(x)$:

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

We also define:

$$m_j^{(p)} = \begin{cases} n_j & j = 1,\ldots,r; \\ nP_j\left(\Phi^{(p)}\right)/P\left(\Phi^{(p)}\right) & j = r+1,\ldots,v; \end{cases} \tag{3}$$

$$\tau_i^{(p)}(x) = \frac{\pi_i f_i\left(x;\theta^{(p)}\right)}{f\left(x;\Phi^{(p)}\right)}, \tag{4}$$

$$c_i^{(p)} = \sum_{j=1}^{v} m_j^{(p)} \mathrm{E}_j^{(p)}\left[\tau_i^{(p)}(x)\right], \tag{5}$$

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 $j \le r$) or they represent the expected count for unobserved bins (i.e., $j > r$). The 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 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 $i$, the current estimate of the expected number in the $i$th mixture component. 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 $(p+1)$:

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

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

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

These equations specify how the component weights (i.e., $\pi$'s), component means (i.e., $\mu$'s) and component standard deviations (i.e., $\sigma$'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., $\mathrm{E}_j^{(p)}[\cdot]$). 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 Eqs. (6)–(8) to the vector/covariance cases:

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

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

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

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 find at least a local maximum of the likelihood as defined by Eq. (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 set up 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., $\mathrm{E}_j^{(p)}[\cdot]$) at each EM iteration. As defined by Eq. (2), each expected value in Eqs. (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 the 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 full 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 $\Phi$. Eqs. (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 $\mathrm{E}_j^{(p)}[\cdot]$ 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 efficiently 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; Press et al., 1992 for details). An important aspect of Romberg integration is selection of the *order k* 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 $\infty$. In the 1-dimensional case it suffices to define 2 additional bins: one extending from the last bin to $\infty$, and the other extending from $-\infty$ to the first bin. In the multivariate case it is more natural to define a single bin

$$\mathcal{H}_{r+1} = \mathcal{H} \Big\backslash \sum_{j=1}^{r} \mathcal{H}_j$$

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:

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

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

$$\int_{\mathcal{H}_{r+1}} f_i(\mathbf{x};\, \theta)(\mathbf{x} - \boldsymbol{\mu}_i)(\mathbf{x} - \boldsymbol{\mu}_i)^+\, d\mathbf{x}$$
$$= \Sigma_i - \sum_{j=1}^{r} \int_{\mathcal{H}_j} f_i(\mathbf{x};\, \theta)(\mathbf{x} - \boldsymbol{\mu}_i)(\mathbf{x} - \boldsymbol{\mu}_i)^+\, d\mathbf{x}. \tag{14}$$

Note that no extra work is required to obtain the integrals on the right-hand side of the equations above. The basic EM Eqs. (9)–(11) require the calculation of expected values similar to those defined in Eq. (2) for each bin. Note, however, that the only difference between those expected values and integrals on the right-hand side of Eqs. (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 Eqs. (9)–(11), while the unnormalized values are used in Eqs. (12)–(14).

For 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 $\delta$. This means that if the relative change of the value of the integral in two consecutive iterations falls below $\delta$ we consider the integral to have converged ($\delta$ is chosen to be 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 computational

resource usage by minimizing 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 $\delta$, we are tolerating absolute error in integration of order $\delta I$. On the other hand, assume that in the irrelevant bins the value of the PDF has height on the order of $\epsilon F$, where $\epsilon$ 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 $\delta'$ 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 to choose $\delta'$ 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 $\delta'$ becomes $o(1)$, we can start using a single iteration of the simplest possible integration scheme and still stay within the allowed limit of $\delta'$. To summarize, given a value for $\epsilon$, the algorithm estimates the average height $F$ of the PDF and for all the bins with PDF values less than $\epsilon F$ 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 $\delta$ controls overall accuracy, while $\epsilon$ adds only a small correction to $\delta$). For example, we have found that the variability in parameter estimates from using different small values of $\epsilon$ 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 $\epsilon$. The time is minimized for different values of $\epsilon$ 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 inefficient.

### *3.3. The complete 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 standard (computationally cheap) EM algorithm on a random sample of the data to provide a good starting point for the more computationally complex binned/truncated version of EM. We take a subset of points within each bin, randomize their coordinates around the bin centers (we use the uniform distribution within each bin) and treat the newly obtained data as non-binned and non-truncated. The standard EM algorithm is

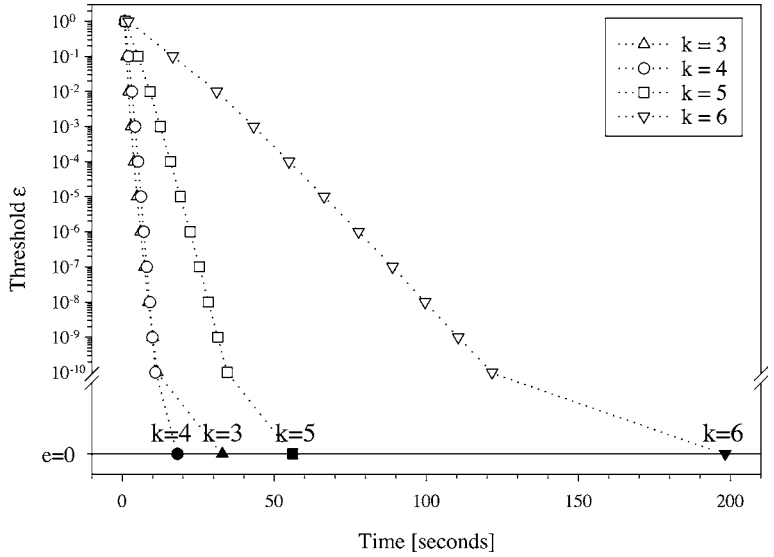



*Figure 2*. Execution time of a single EM step as a function of the threshold $\epsilon$ for several different values $k$ of the Romberg integration order. For $k = 2$, the values were off-scale, e.g., $t(1) = 23$, $t(0.1) = 363$, etc. Results based on fitting a two-component mixture to 40,000 red blood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

relatively fast, as a closed form solution exists for each EM step (without any integration). Once the standard algorithm converges to a solution in parameter space, we use these parameters as initial starting points for the full algorithm (for binned, truncated data) which then refines these 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 full algorithm 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 $100 \times 100$ bins of red blood cell counts; $k$ is the order of Romberg integration and $\epsilon$ is the threshold for declaring a bin to be small enough for "fast integration" as described earlier. All parameter choices $(k, \epsilon)$ result in the same quality of final solution (i.e., all asymptote to the same log-likelihood eventually). Using no approximation ($\epsilon = 0$) is two orders of magnitude slower than using non-zero $\epsilon$ values. Increasing $\epsilon$ from 0.001 to 0.1 results in no loss in likelihood but results in faster convergence. Comparing the curves for $k = 3$, $\epsilon = 0.1$ to the randomized initialization method described earlier, shows about a factor of two gain in convergence time for the randomized initialization. The $k$-means clustering algorithm is used to initialize the binned algorithm, a widely-used approach for initializing the EM algorithm for mixture modeling.

Figure 3 includes the total time required to achieve the specified log likelihoods. The total time consists of both the time required for initialization and the time required to perform several iterations of the EM algorithm until convergence.

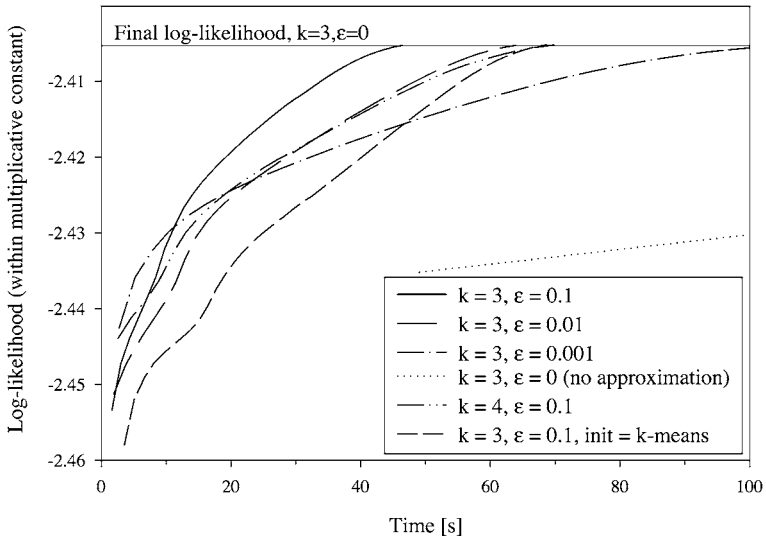



*Figure 3*. Quality of solution (measured by log-likelihood) as a function of time for different variations on the algorithm.

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

- Treat the multivariate histogram as a PDF and draw a small number of data points from it (add some count to all the bins to prevent 0 probabilities in empty bins).
- Fit a standard mixture model using the standard EM algorithm (i.e., for non-binned, non-truncated data).
- Use the parameter estimates from standard mixture modeling and refine them with the full algorithm until convergence. This consists of iteratively applying Eqs. (9)–(11) for the bins within the grid and applying Eqs. (12)–(14) for the single bin outside the grid until convergence as measured by Eq. (1).

## 4. Experimental results with simulated data

In this section we describe three sets of experiments designed to demonstrate various aspects of the algorithm described in this paper. In each set of experiments we simulate data points from a known PDF and then bin them. We vary the number of bins per dimension in steps of 5 from $B = 5$ to $B = 100$ so that the original unbinned samples are quantized into $B^2$ bins. Each experiment is repeated on 10 different samples and the results are averaged to obtain smoother estimates.

On the original unbinned samples we ran the standard EM algorithm, and on the binned data we ran the binned version of EM (using the general technique of Section 3.3 and the parameters and settings described in the following section). The purpose of the simulations was to observe the effect of binning and/or truncation 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 (KL) (or cross-entropy) distance between each estimated density and the true known density. The KL distance is non-negative and is zero if and only if two densities are identical. We calculated the average KL-distance over the 10 samples for each experiment, for both the binned and the standard EM algorithms.

1. The first set of experiments was designed to test the quality of the solution for different numbers of data points drawn from the two–component mixture model shown in figure 4 ($\pi_1 = \pi_2 = 0.5$, $\sigma = 1$, $\mu_1 = (-1.5, 0)$, $\mu_2 = (1.5, 0)$). Figure 4 also shows the grid boundaries we used $(-5, 5) \times (-5, 5)$, i.e., there is almost no truncation except far out in the tails. We varied the number of data points drawn from each of the components in steps of 10 from $N = 100$ to $N = 1000$. In total, each of the standard and binned algorithms were run 20,000 different times (20 different numbers of bins, 100 different numbers of data points, 10 random samples) to generate the reported results.
2. The second set of experiments tests the performance of the algorithm when the component densities are not so well separated. We started with two overlapping components centered at $(-1.5, 0)$ of equal weight ($\pi_1 = \pi_2 = 0.5$) and with unit variance ($\sigma = 1$), and moved them in 20 steps until they were $3\sigma$ apart (as in figure 4). The number of randomly drawn data points was 500 per component. The grid was the same as in the first experiment, i.e, $(-5, 5) \times (-5, 5)$. Figure 4 shows the setup: we started with a mixture PDF that corresponds to the single Gaussian on the left (two completely overlapping component densities), and end up with two separate components shown as component densities.

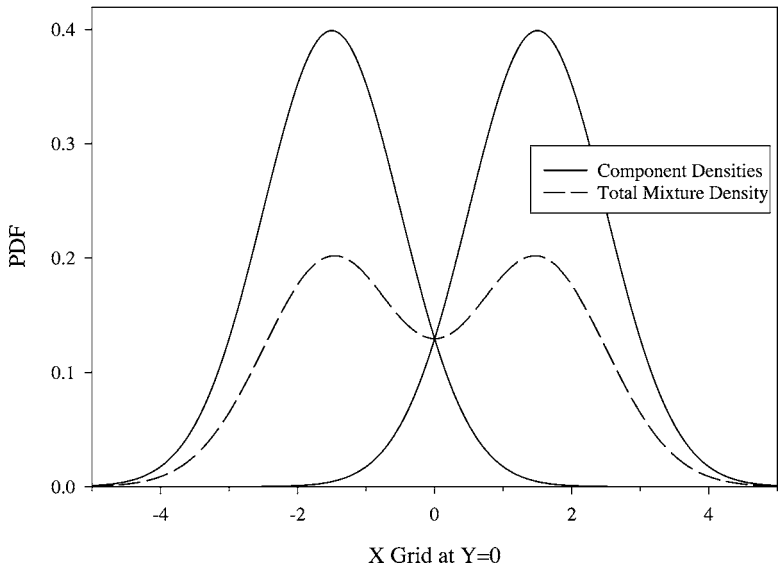



*Figure 4*. Cross-section of the 2-dimensional grid showing two Gaussian components that are $3\sigma$ apart. The total mixture density has $\pi_1 = \pi_2 = 0.5$.

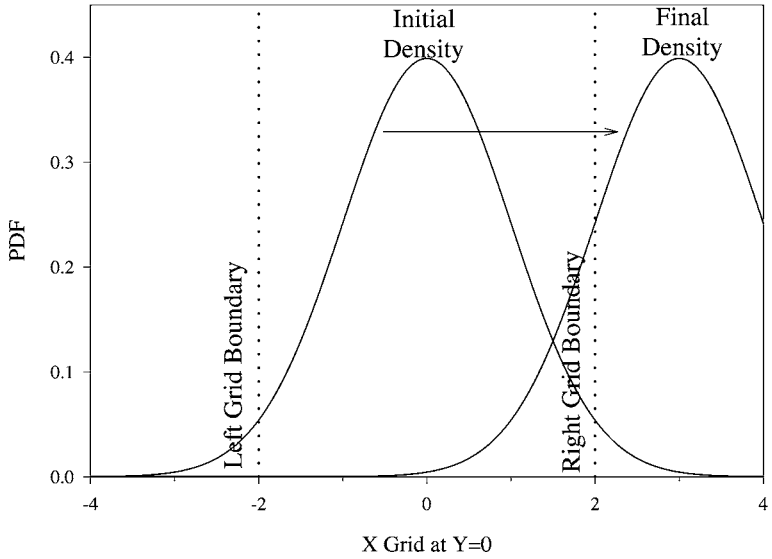



*Figure 5*. Setup for truncation experiments. Cross-section of a 2-dimensional grid showing the initial and final position of a single Gaussian for a set of experiments involving truncation. Note that all points sampled outside the two dotted lines are truncated (i.e., omitted from the training data).

Again, there was very little truncation. Each of the standard and binned algorithms were run 4,000 different times (20 different numbers of bins, 20 different separation of the means, 10 random samples) to generate the reported results.

3. The third set of experiments was designed to test the performance of the algorithm when significant truncation occurs. We decreased the grid size to $(-2, 2) \times (-2, 2)$ and sampled 500 data points from a single Gaussian with unit variance ($\sigma = 1$) which we moved from the center of the grid ($\mu = (0, 0)$) to a point well outside the grid ($\mu = (3, 0)$) in 100 equally spaced steps. Figure 5 shows the initial and final Gaussians together with the truncation boundaries. Each of the standard and binned algorithms were run 20,000 different times (20 different numbers of bins, 100 different positions of the mean, 10 random samples) to generate the reported results.

We also briefly report on the accuracy of PDFs estimated using a standard EM algorithm on data which are randomly sampled from within the bins.

### *4.1. EM methodology*

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

1. $n$ points are randomly drawn from the binned histogram, where $n$ is chosen to be 10% of the number of total data points or 100 points, whichever is greater. Points are drawn using a uniform sampling distribution within each bin.

2. The standard EM algorithm is initialized by the $k$-means algorithm and run until convergence on the $n$ data points from step 1.
3. Step 2 is performed 5 times (5 starts of the standard EM algorithm) and parameters yielding the highest log likelihood are used as the initial parameter guess for the binned algorithm.
4. The binned EM algorithm is initialized by parameters found in step 3 and run until convergence.
5. To avoid poor local maxima, steps 1–4 are repeated 10 times (10 starts of binned algorithm) and the solution with the highest likelihood is reported.
6. Convergence of the standard and binned/truncated EM is judged by a change of less than 0.01% in the log-likelihood, or after a maximum of 20 EM iterations, whichever comes first.
7. The order of the Romberg integration is set to 3 and $\epsilon$ is set to $10^{-4}$. Figure 2 shows execution times of a single EM iteration for several different orders and thresholds $\epsilon$. For $\epsilon = 10^{-4}$ the optimal order of integration (in terms of the execution speed) corresponds to $k = 3$.
8. The default accuracy of the integration is set to $\delta = 10^{-5}$.

### *4.2. Estimation from random samples generated from the binned data*

A simpler baseline approach than the EM approach proposed in this paper would be to estimate the PDF from a random sample from the binned data. Here we briefly discuss the performance of this baseline method relative to our full binned algorithm. Specifically, if the binned data represent $N$ counts in total, one can sample $N' \leq N$ data points, where the number of points from each bin is in proportion to the count for that bin and the points are uniformly distributed within each bin. Standard maximum likelihood estimation is then performed, using EM if there is more than one component in the mixture density. We will refer to this as the *uniform sampling estimation method*. Note that the uniform distribution is likely to be an inappropriate assumption within each bin since the true PDF will typically be non-uniform across a bin. However, the uniform sampling estimation method is nonetheless simple enough that it is worthwhile to investigate as a simple baseline alternative for density estimation on binned data, and indeed this is the method we use for initializing our full binned algorithm, as described earlier.

We generated 500 data points in two dimensions drawn from a zero-mean Gaussian distribution with an identity covariance matrix. Figure 6 shows the estimated PDF corresponding to maximum likelihood parameter estimates on the unbinned original 500 data points (sample mean/covariance). Also shown is the estimated PDF which results from binning the data, and then running the uniform sampling estimation method on 500 sampled points from this binned data. This PDF is then used as initialization for the full binned algorithm (Section 3), and the resulting PDF is also plotted in figure 6.

Figure 6 shows that for both 5 and 10 bins the estimated PDF from the full binned algorithm is much closer to the PDF on unbinned data than is the PDF obtained from the uniform sampling estimation method (in figure 6(b) the full algorithm almost exactly matches the sample PDF). The PDFs estimated using the uniform sampling estimation

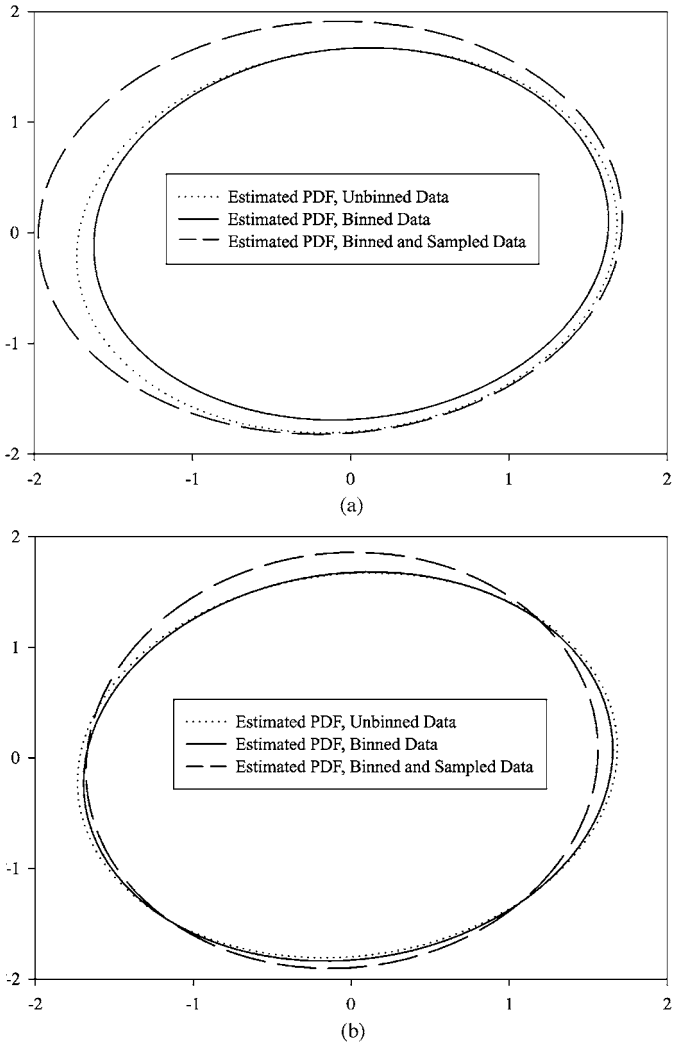



*Figure 6*. Estimated PDFs obtained from original (unbinned) data and PDFs fitted by binned (full) and the uniform random-sample algorithm for (a) 5 bins per dimension and (b) 10 bins per dimension. Plotted lines represent 3-$\sigma$ covariance ellipses.

method had a KL-distance of 0.02 and 0.01 (for 5 and 10 bins respectively) from the PDFs on the unbinned data. The PDFs estimated by the full binned algorithm had KL-distances which were factors of 4 and 20 *smaller* (for 5 and 10 bins respectively) from the PDFs on the unbinned data.

In both plots the PDF obtained from uniform sampling overestimates the variance. Variance overestimation in the case of a true underlying symmetric unimodal distribution can be explained by the bias introduced by uniform random sampling, i.e., the points that are

in the lower-density region of the bin are over-sampled and since they are further from the mean than points in the higher-density region of the bin the overall effect is one of artificial variance inflation relative to the original true PDF which generated the data points.

The results in figure 6 are typical of results obtained in general with the random sampling approach, i.e., variance inflation in the resulting estimates. Of course, as the number of bins increases per dimension the width of the bins decreases, the performance of the uniform sampling estimation method can be expected to asymptotically approach that of the full binned algorithm, and both should approach the performance of PDF estimation on the original unbinned data. In this paper we are primarily interested in the performance of the full binned algorithm, relative to both the PDF estimated from unbinned data and the true underlying PDF. The uniform sampling estimation method typically performs worse than the full binned algorithm: it will be used only for initialization in the experiments which follow and not reported on in any further detail.

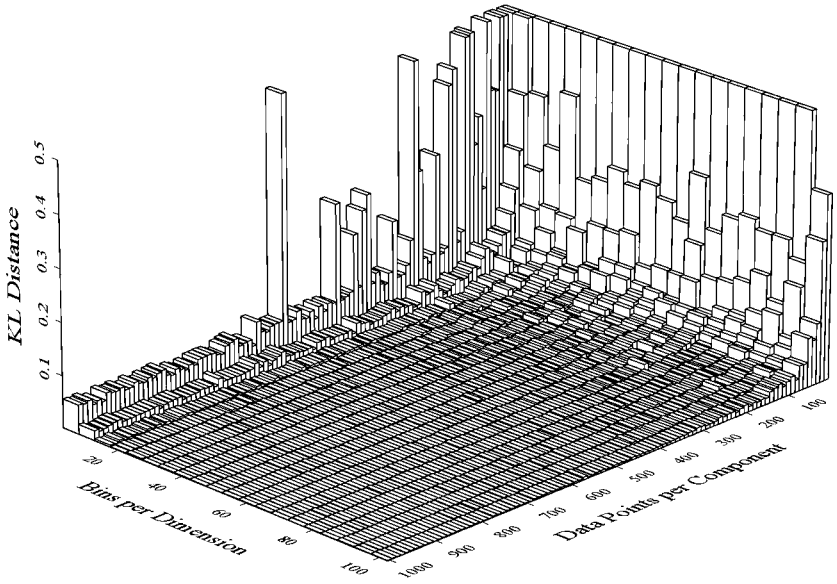
### *4.3. Experiments with different sample sizes*

Figure 7 shows a plot of the average KL-distance for the binned EM algorithm (a) and the corresponding standard deviation (b), 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 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., the solution quality increases roughly in proportion to $N^{-\alpha}$ for some constant $\alpha$. As a function of bin size $B$, there appears to be more of a threshold effect: with more than 20 bins the KL-distance is again relatively flat as a function of the number of bins. Below $B = 20$ the solutions rapidly decrease in quality (e.g., for $B = 5$ there is a significant degradation).

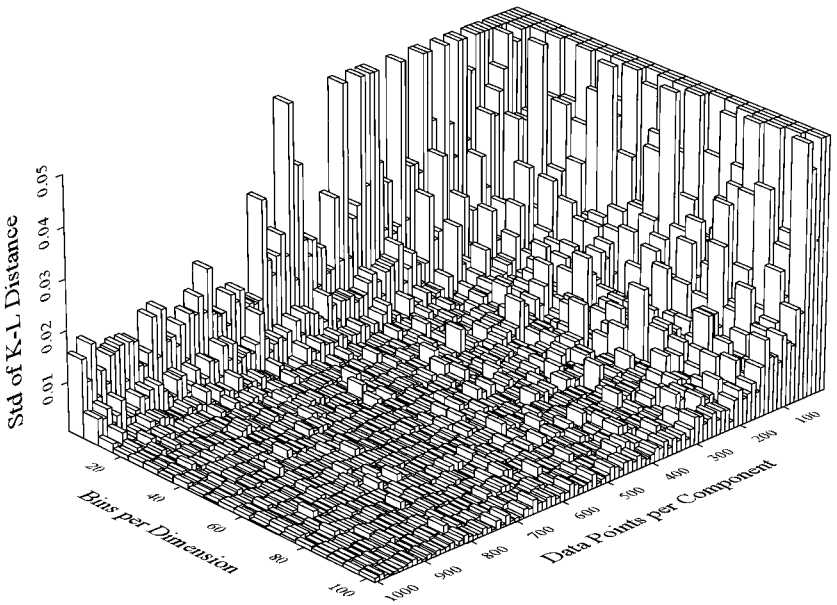
In figure 8 we plot the KL-distance (log-scale) as a function of the bin size, for specific values of $N$ ($N = 100, 300, 1000$), comparing both the standard and binned versions of EM. For each of the 3 values of $N$, the curves have the same qualitative shape: a rapid improvement in quality as we move 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 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 9 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




(a)




(b)

*Figure 7*. (a) 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; (b) Standard deviation of the KL-distance from 10 repeated experiments. Note that the scale is 10 times finer than that of figure (a). The scale in both plots is selected such as to show relevant regions. As a consequence, several bars (for small number of sampled data points) on each plot are truncated.

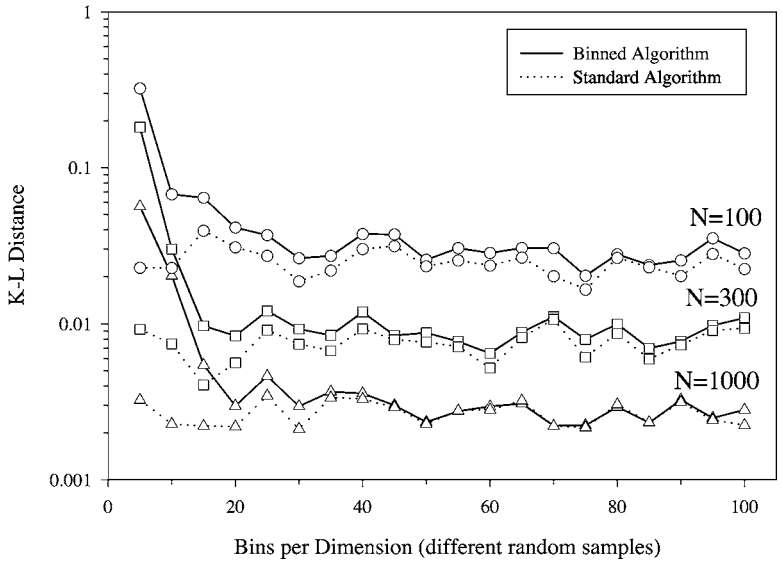



*Figure 8*. Average KL-distance (log-scale) between the estimated densities and the true density as function of the number of bins, for different sample sizes, and compared to standard EM on the unbinned data.

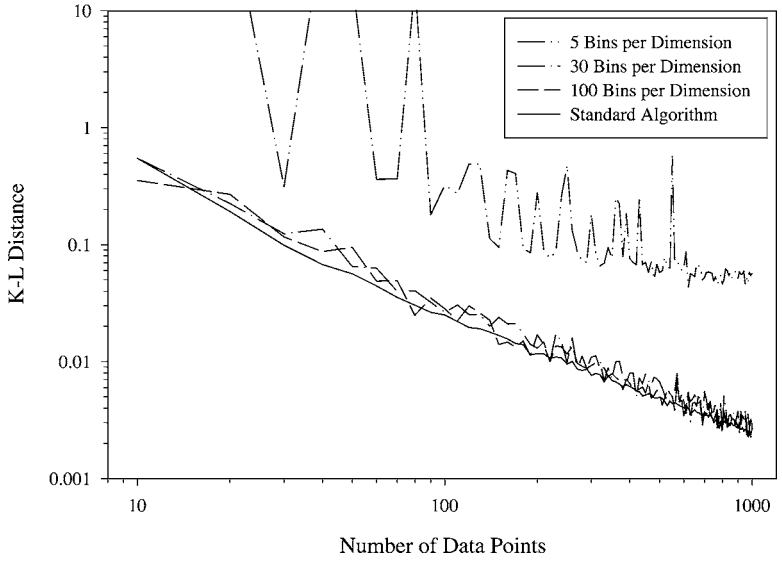



*Figure 9*. Average KL-distance (log-scale) between the estimated densities and the true density as function of sample size, for different numbers of bins, and compared to standard EM on the unbinned data.

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.

### *4.4. Experiments with different separations of mixture components*

Figure 10 shows a plot of the average KL-distance for the binned EM algorithm as a function of the number of bins and the separation of the component densities. It is interesting to note that the quality of the solution is relatively insensitive to the separation of the components and that we only see usual sensitivity on the number of bins per dimension (described in the previous section). However, it is important to note that we measure the KL-distance between the fitted model and the true model, and not the distance between individual components in each of the models. In other words, the *shape* of the fitted model is very close to the true model while the individual components need not necessarily be. We again see a "plateau" effect in that the quality of the solution is relatively close to zero when the number of bins is above 10.

Figure 11 plots the ratio of average KL-distances as a function of separation of means for several specific numbers of bins $B$. We plot the ratio of KL-distances of the standard and binned algorithm to make a comparison. From the plot we again conclude that for small

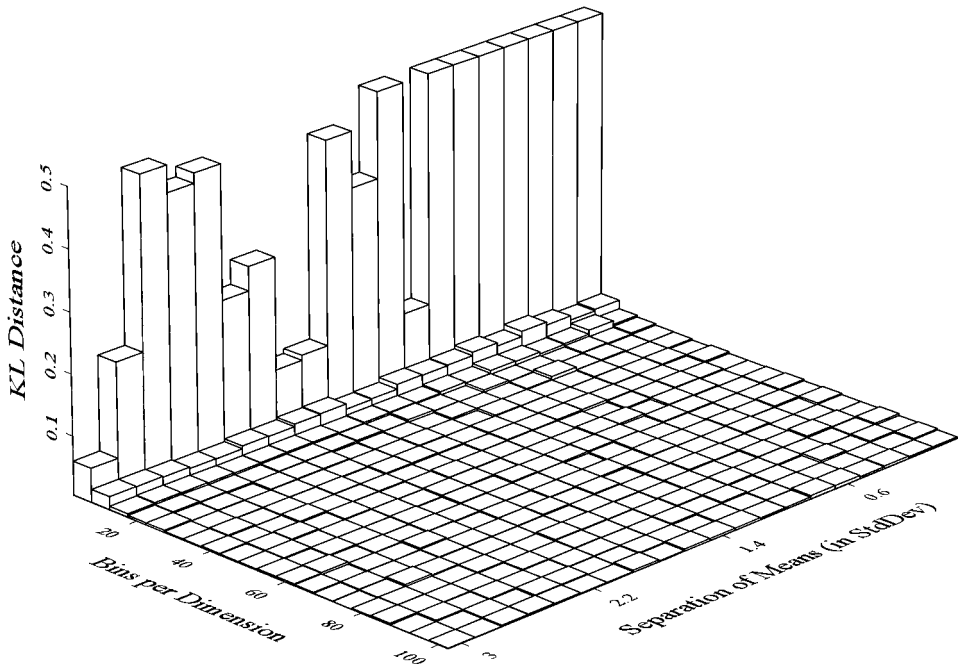



*Figure 10.* Average KL-distance between the estimated density (estimated using the procedure described in this paper) and the true density, as a function of the number of bins and the separation of the means of individual mixture components.

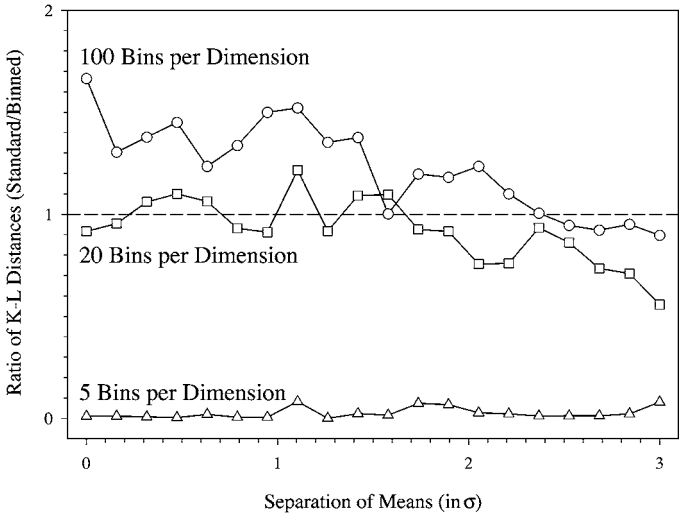



*Figure 11.* Ratio of KL-distances achieved by the standard and binned (full) algorithm as a function of separation of the means of individual mixture components. Regions above a ratio of 1 are the regions where the binned algorithm performs better.

number of bins $B$ the standard algorithm performs significantly better, but the advantage is quickly lost as the number of bins per dimension is above 10. We also note that the binned algorithm performs better when separation of means is small which we attribute to natural smoothing that occurs with binning. The standard algorithm generates noisier estimates when the means are close to each other since it has more information available then the binned algorithm.

Figure 12 shows the ratio of the KL-distances for standard and binned algorithm as a function of number of bins per dimension $B$ for several typical values of the separation of means of mixture components. We see the rapid improvement of binned algorithm as the number of bins increases.

The results in this section suggest that the binned algorithm is sensitive to a very small number of bins (e.g., $B \leq 10$) but then quickly reaches the quality of the standard algorithm as the number of bins increases. The results suggest that inference of mixture components using binned data is typically no more or less sensitive to component overlap than inference using unbinned data.

### *4.5. Experiments with truncation*

In this section we investigate the effects of truncation on the inferred PDF. To isolate the effect, we consider only a single Gaussian PDF. The average KL-distance between the true model and the inferred models (binned and standard algorithms) is linearly increasing as the ratio of truncated points increases (which is we would expect). Since this plot is not particularly informative we omit it (i.e., we do not show the counterpart of

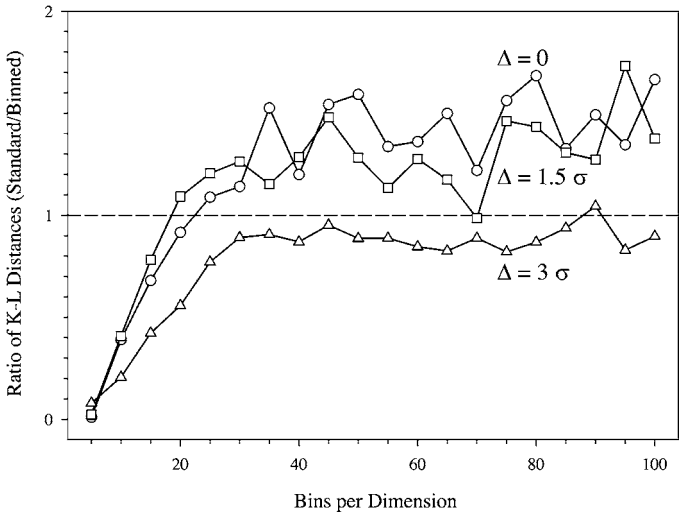



*Figure 12*. Ratio of KL-distances achieved by the standard and binned (full) algorithm as a function of number of bins for several separations of individual components. Regions above a ratio of 1 are the regions where the binned algorithm performs better.

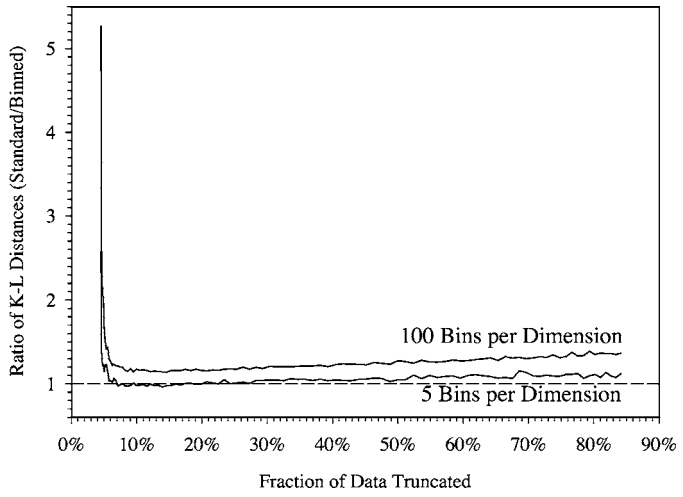



*Figure 13*. Ratio of KL-distances achieved by the standard and binned (full) algorithm as a function of the fraction of truncated data points sampled from a single Gaussian density. Note that the truncation ratio is a function of the position of the mean. Regions above a ratio of 1 are the regions where the binned algorithm performs better.

figures 7 and 10). Instead, we directly compare the performance of the standard and full algorithms.

Figure 13 plots the ratio of the average KL-distances as a function of ratio of truncated points. Note that the standard algorithm we compare our results to cannot handle truncation

(i.e., it is non-binned and non-truncated algorithm and ignores the fact that the data are in fact truncated). On the $x$ axis in figure 13 we show the percent of truncated points which is a function of the position of the single Gaussian PDF. The scale on the $x$ axis is linear in the position of the mean (ranging from 0 to 3) and therefore is not linear in the truncation ratio (hence, we show many tick labels). The plot shows the relative insensitivity of the binned algorithm to the number of bins per dimension, or equivalently, it shows that the effect of truncation is more severe than the effect of binning (compare this to the results in previous sections). Figure 13 also shows that when truncation is small (i.e., less than 5%), the full algorithm can infer the underlying true density much better than the standard algorithm (which typically infers too narrow a PDF). However, as the amount of truncation increases, both algorithms tend to infer poor densities and the difference in quality decreases. The full algorithm still infers a significantly better PDF, especially as the number of bins per component increases.

Figure 14 summarizes the results as a function of bins per component $B$ for several typical truncation ratios. It again shows that the binning effect is significantly smaller than the truncation effect. It also shows that for small truncation effects the full algorithm performs roughly 5 times better than the standard algorithm (as measured by the KL-distance), and that the advantage of being able to account for truncation decreases as the truncation ratio increases (not surprisingly).

The results in this section suggest that the effect of truncation is more significant than the effect of binning (i.e., it has a greater effect on the EM algorithm's ability to infer a PDF).

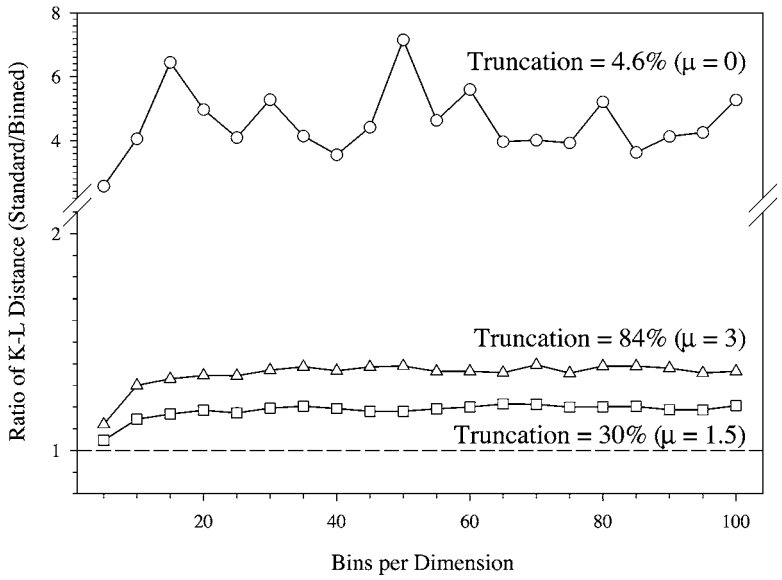



*Figure 14*. Ratio of KL-distances achieved by the standard and binned (full) algorithm as a function of number of bins for several different truncation fractions (positions of the single Gaussian density). Regions above a ratio of 1 are the regions where the binned algorithm performs better.

For small truncation ratios (i.e., less than 5%) it appears to be particularly worthwhile to have an algorithm that can handle truncation properly.

## 5. An application to medical diagnosis

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 (cytograms) characterizing red blood cell volume and hemoglobin measurements (see figure 1). The equipment used to measure red blood cell characteristics produces two-dimensional measurements which are automatically binned and truncated. In this section we discuss how the density estimation methods described earlier in this paper were applied to this data in the context of discriminating healthy individuals from individuals with iron deficiency anemia (Cadez et al., 1999).

The first step in modeling the data is to characterize the two-dimensional volume-hemoglobin distribution. It can be shown that the marginal volume distribution of a single population of RBC is theoretically lognormal. The lognormality comes from the biological mechanism governing the manner by which cells are produced (McLaren, Brittenham, & Hasselblad, 1986). At each "production step" cells divide and have normal variations in their respective volumes. Since the process is repetitive and the effect is multiplicative (i.e., cells divide), the resulting distribution is lognormal. For iron-deficient subjects, the argument follows that the RBC density can be well-approximated as a two-component log-normal mixture (McLaren et al., 1991; McLaren, 1996). Specifically, there are two biological processes that are constantly occurring in a body: 1) red blood cells are produced in the bone marrow; 2) these cells are extruded into the bloodstream and die after about 120 days. For a healthy individual a single population of red blood cells is produced with a mean cell volume and mean hemoglobin concentration within the normal range. In iron deficiency anemia the red blood cells that are produced have decreased volume and decreased hemoglobin, below normal for that of a healthy individual. Thus, with development of the disease, gradually, a second subpopulation of red blood cells begins to emerge and over a period of time, the relative ratio of subpopulations of red cells changes.

Data collection for this study was completed during 1995. A reference sample group of healthy individuals was recruited from staff physicians and hospital employees at the Western Infirmary, Glasgow, Scotland. Patients included in the study were seen on the wards and in the outpatient clinics and referred to the hospital laboratories for a complete blood count. The results described here are based on the 90 Control and 82 Iron Deficient subjects in the study.

Figure 15 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. Note, however, that the first control plot and the first iron deficient plot are visually very similar to each other, underlying the fact that classification of individuals based on their densities is non-trivial. Since the number of bins is relatively large

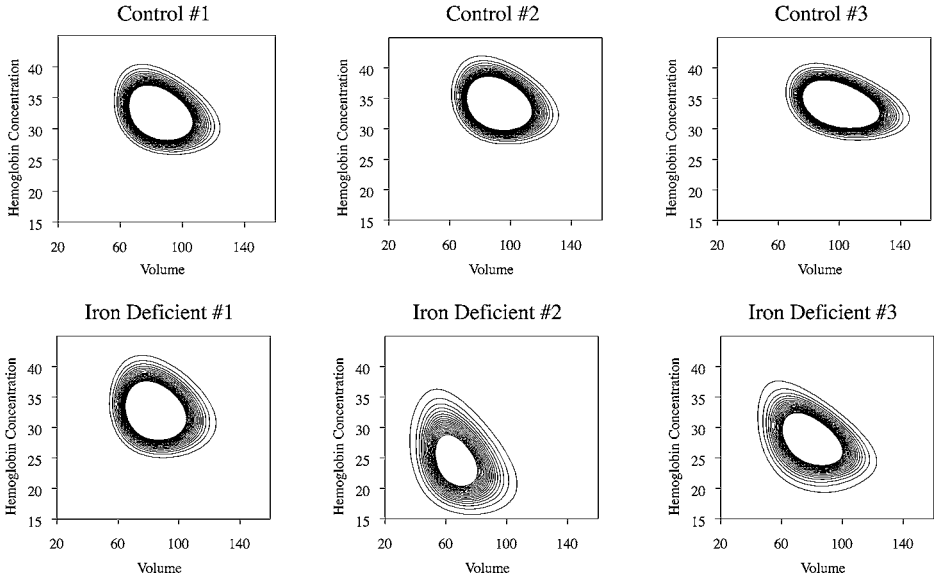



*Figure 15.* Contour plots from estimated density estimates for three control patients and three iron deficient anemia patients. The two plots in the first column show two borderline patients, the two plots in the second column show two typical patients and the two plots in the third column show two patients far from the borderline. The patients are selected such as to show inter and intra group variability. The lowest 10% of the probability contours are plotted to emphasize the systematic difference between the two groups.

($B = 100$ in each dimension), as is the number of data points (40,000), the simulation results from the previous section suggest that these density estimates are likely to be relatively accurate (compared to the results which could have been obtained if the data had not been binned).

We experimented with three different sets of feature representations for the data and two different classification methods (discriminative and probabilistic). The feature representations used were:

1. *Baseline features*: no density estimation at all was performed on each individual's cytogram. Instead a 4-dimensional feature-vector consisting of the mean and variance along each of the volume and hemoglobin dimensions was constructed and individuals were classified in this 4-dimensional space.
2. *11-dimensional parameters*: The EM algorithm described earlier in the paper was used to estimate two-component lognormal mixture model parameters for each of the 172 individuals in the study. The resulting 11-dimensional parameter vector for each individual was then used directly for classification.
3. *9-dimensional parameters*: We reparameterized some of the parameters in the 11-dimensional feature set to reflect a more natural scale for modeling. We used the log-odds of the component weights (rather than the weights directly) and the log of the eigenvalues of the covariance matrices.

Figure 3 demonstrated the decrease in computational resources which can be achieved during the parameter estimation component of classification, using the methods described earlier in this paper. The data in figure 3 are for a two-component mixture model fit to a control subject with a two-dimensional cytogram (histogram) of 40,000 red blood cells.

For classification the discriminative classifiers used were tree-based (C5.0 and CART). The probabilistic classifiers used a hierarchical mixture model; a "low-level" mixture model for each individual's two-dimensional cytogram, and a "high level" population model characterizing the location and variability of individuals in parameter space for each of the two classes. At the individual level, two-component lognormal mixture densities with unconstrained covariance matrices were fit to each individual's cytogram using the EM algorithm for binned data described earlier in this paper. At the population level (modeling a class of individuals in parameter space), we modeled each of the two classes as either a single Normal density or a mixture of two Normals (model selection performed by internal cross-validation). For the 11-dimensional parameter set we used block covariance matrices for the Normal densities, which modeled the mixture weight parameter as independent, allowed full covariance between all 4 means, and full covariance between both sets of the 3 independent covariance parameters for each RBC component. For the 9-dimensional parameter set we again used a block diagonal covariance matrix structure for the Normal densities, allowing full covariance among all 4 means, full covariance between the 2 log-eigenvalues for each RBC component, and allowing the log-odds of the weight to be independent of the other parameters. Classification was subsequently performed using Bayes rule.

For each of the experiments reported below we performed 100 cross-validation runs, where in each run the data were divided into a randomly chosen training set of 80% of the data and a test set consisting of the remaining 20%. Overall performance for each method is reported as the mean and standard deviation of classification accuracy on the test sets over the 100 runs. Note that the parameter estimation (the running of EM to determine the $\hat{\theta}_i$'s) is completely independent of any other data or class labels; it is a purely unsupervised procedure performed on each individual's RBC measurements. Thus, the parameters $\hat{\theta}_i$ were estimated once only, before any cross-validation takes place.

Table 1 summarizes the mean cross-validated error rates and standard errors across the different methods. The discriminative algorithms (C5.0 and CART) were run on the 11 original parameters, the 9 reparameterized parameters, and the 4 "Baseline" features. The hierarchical mixture model (with either 9 or 11 features) had lower error rates than any of the discriminative tree methods (usually on the order of a factor of 2 lower). The 11-parameter hierarchical error rate of 1.65% corresponds to about a 65% decrease in error rate from the error rate of 4.65% of CART on the same features and about a 54% decrease in error rate from the 3.62% error rate of C5.0 on the same 11 features. Thus, for this particular problem, the hierarchical model appears superior in terms of classification accuracy.

For routine clinical classification of RBC data, the decision tree approaches are attractive since classification rules can be clearly described. However, for clinical ranking of subjects based on likelihood of iron deficiency (for example), the hierarchical model approach may be the most useful given its probabilistic basis. In addition, the hierarchical model is intrinsically interesting from a medical research viewpoint. It provides a basis for a complete characterization of blood disorders in hemoglobin-volume space both in terms of typicality

*Table 1.* Means and standard deviations of the cross-validated classification error for each of the different classification methods and feature representations, across 100 runs.

| Method | Features | Mean error rate (%) | Standard deviation |
|---|---|---|---|
| C5.0 | Baseline | 3.32 | 2.92 |
| | 9-Parameters | 3.53 | 2.87 |
| | 11-Parameters | 3.62 | 2.92 |
| CART | Baseline | 3.47 | 3.17 |
| | 9-Parameters | 4.09 | 3.34 |
| | 11-Parameters | 4.65 | 3.43 |
| Hierarchical | 9-Parameters | 1.90 | 1.94 |
| | 11-Parameters | 1.65 | 2.06 |

and variability of individuals within each group, as well as full characterization of group differences.

## 6. Conclusions

The problem of fitting mixture densities to multivariate binned and truncated data was addressed based on the EM procedure for the one-dimensional problem. The theoretical foundations of the multivariate approach follow straightforwardly from the original univariate formulation of McLachlan and Jones (1988). The numerical issues are not so straightforward, however, since 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.

We note that there is an inescapable "curse-of-dimensionality" at work as the number of dimensions $d$ increases, including (1) an exponential increase in the total number of bins, (2) a corresponding decrease in the data per bin for a fixed sample size, (3) numerical evaluation of $d$-dimensional multivariate integrals at each iteration of EM, and (4) an exponential increase in the number of different "tail regions" which must be handled at the corners of the $d$-dimensional histogram for truncated data. Thus, although the theory and algorithms are described here for the general $d$-dimensional case, the methods proposed here are likely to be practical only for low-dimensional problems (this is of course true for any truly unrestricted multivariate density estimation technique).

Simulation results on two-dimensional histograms indicate that relatively high quality solutions can be obtained compared to running EM on the "raw" unbinned data, unless the number of bins is relatively small. These results are interesting since they suggest that the loss of information due to quantization (from a density estimation viewpoint) is minimal once the number of bins being used exceeds 10 or so. Thus, while it is well-known that quantization necessarily incurs a loss of information, the loss was often found to be minimal in the density estimation problems evaluated in this paper. This in turn may provide further justification for the use of histograms in computer vision, information retrieval, and other problems in learning and pattern recognition. In similar experiments with simulated

two-dimensional histograms, truncation produced a systematic deterioration in the quality of the estimated PDFs and was found to typically produce a more severe effect than binning.

The proposed EM algorithm was applied to a real application involving discrimination of individuals with and without anemia based on two-dimensional cytograms characterizing RBC volume and hemoglobin. The ability to fit mixture densities to binned and truncated two-dimensional data in an efficient manner, as described in this paper, is an important part of the overall hierarchical mixture modeling approach for this application.

## Acknowledgments

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 GMcL has been partially supported by a grant from the Australian Research Council. The contribution of CMcL has been supported in part by grants from the National Institutes of Health (R43-HL46037 and R15-HL48349) and a Welcome Research Travel Grant awarded by the Burroughs Welcome Fund. We thank Thomas H. Cavanagh for providing laboratory facilities. We are grateful to Dr. Albert Greenbaum for technical assistance.

## References

Basta, S. S., Soekirman, M. S., Karyada, D., & Scrimshaw, N. S. (1979). Iron deficiency anemia and the productivity of adult males in Indonesia. *American Journal of Clinical Nutrition*, *32*, 916–925.

Bishop, C. M. (1995). *Neural networks for pattern recognition*. Oxford, UK: Clarendon Press.

Cadez, I. V., McLaren, C. E., Smyth, P., & McLachlan, G. J. (1999). Hierarchical models for screening of iron deficiency anemia. In *Proceedings of the International Conference on Machine Learning* (pp. 77–86). Los Gatos, CA: Morgan Kaufmann.

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

Edgerton, V. R., Gardner, G. W., Ohira, Y., Gunawardena, K. A., & Senewiratne, B. (1979). Iron-deficiency anaemia and its effect on worker productivity and activity patterns. *British Medical Journal, 2*, 1546–1549.

Flickner, M., Sawhney, H., Niblack, W., Ashley, J., Huang, Q., Dom, B., Gorkani, M., Hafner, J., Lee, D., Petkovic, D., Steele, D., & Yanker, P. (1995). Query by image and video content. *IEEE Computer*, 23–31.

Jones, P. N., & 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, 39:2*, 273–282.

Lee, J.-H, Kim, D.-H., & Chung, C.-W. (1999). Multi-dimensional selectivity estimation using compressed histogram information. In *Proceedings of SIGMOD 1999*, New York, NY: ACM Press, pp. 205–214.

Little, R. J. A., & Rubin, D. B. (1987). *Statistical analysis with missing data*. New York: John Wiley.

Matias, Y., Vitter, J. S., & Wang, M. (1998). Wavelet-based histograms for selectivity estimation. In *Proceedings of SIGMOD 1998*, New York, NY: ACM Press, pp. 448–459.

Maybury, M. T. (Ed.). (1997) *Intelligent multimedia information retrieval*. Menlo Park, CA: AAAI Press.

McLachlan, G. J., & Jones, P. N. (1988). Fitting mixture models to grouped and truncated data via the EM algorithm. *Biometrics, 44:2*, 571–578.

McLachlan, G. J., & 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, 5:2*, 129–153.

McLaren, C. E., Brittenham, G. M., & Hasselblad, V. (1986). Analysis of the volume of red blood cells: Application of the expectation-maximization algorithm to grouped data from the doubly-truncated lognormal distribution. *Biometrics, 42:1*, 143–158.

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, 47:3*, 607–622.

Osaki, F. A. (1993). Iron deficiency in infancy and childhood. *New England Journal of Medicine, 329*, 190–193.

Poosala, V. (1997). Histogram-based estimation techniques in database systems. Ph.D. Thesis, Computer Science Department, University of Wisconsin at Madison, Madison, WI.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (1992). *Numerical recipes in C: The art of scientific computing*, 2nd edn. Cambridge, UK: Cambridge University Press.

Swain, M. J., & Ballard, D. H. (1991). Colour indexing. *Intl. Journal of Computer Vision, 7:1*, 11–32.

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

Williams, W. J., Beutler, E., Erslev, A. J., & Lichtman, M. A. (1990). *Hematology*, 4th edn. New York: McGraw-Hill, pp. 9–17.

Received February 15, 1999
Revised August 20, 2000
Accepted December 27, 2000
Final manuscript January 27, 2001