A Computationally Efficient Multivariate Maximum-Entropy Density Estimation (MEDE) Technique

Yanni Kouskoulas, Leland E. Pierce*, Senior Member, IEEE*, and Fawwaz T. Ulaby*, Fellow, IEEE*

***Abstract*—Density estimation is the process of taking a set of multivariate data and finding an estimate for the probability density function (pdf) that produced it. One approach for obtaining an accurate estimate of the true density $f(x)$ is to use the polynomial-moment method with Boltzmann–Shannon entropy. Although rigorous mathematically, the method is difficult to implement in practice because the solution involves a large set of simultaneous nonlinear integral equations, one for each moment or joint moment constraint. Solutions available in the literature are generally not easily applicable to multivariate data, nor computationally efficient. In this paper, we take the functional form that was developed in this problem and apply pointwise estimates of the pdf as constraints. These pointwise estimates are transformed into basis coefficients for a set of Legendre polynomials. The procedure is mathematically similar to the multidimensional Fourier transform, although with different basis functions. We apply this technique, called the maximum-entropy density estimation (MEDE) technique, to a series of multivariate datasets.**

***Index Terms*—Adaptive estiamtion, image classification, maximum-entropy methods, probability.**

I. Introduction

THE FUNCTION of a terrain classification algorithm, as applied to a remotely sensed image, is to determine the most likely class identity, from among M possible classes, of a given image pixel on the basis of the observation vector $\boldsymbol{v}$ associated with that pixel. The observation vector usually consists of d measurements (dimensions) made by a multichannel sensor. The decision logic used by the classifier is comprised of a set of decision rules, usually developed on the basis of a d-dimensional data histogram of each of the M classes. In practice, each data histogram consists of a finite number of observations, and therefore it represents an estimate of an underlying probability density function (pdf) that would be found with an infinite number of observations. Conceptually, it is possible to design a classifier that can be optimized to provide the highest statistical classification accuracy possible, provided the pdfs $f_i(x)$ are known for all M classes. Usually, we only have poorly sampled histogram estimates of the density functions, reducing the capability of the classifier to suboptimal performance.

A. Introduction to Density Estimation

Density estimation is the process of taking a set of multidimensional data that were produced by a particular random process and finding the pdf that most likely produced it. For our purposes, we are interested in pdfs that are continuous differentiable functions.

From a mathematical perspective, this is an ill-posed problem; any finite set of data will only *constrain* the solution space, but will not produce a unique solution. There is always some probability that a given density function could produce a particular set of data, and there are many density functions that have a high probability of giving rise to any particular dataset. Given a set of multivariate data $S = \{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_N\}$, there is some probability that a normal density produced it, but there is also some other probability level that an exponential density produced it instead.

Even if we can find a unique solution to this problem, higher dimensional datasets pose an additional challenge because the large amount of data required to produce an estimate of reasonable accuracy. For example, assume it takes a 100 data points to estimate a one-dimensional (1-D) density over a particular domain. The corresponding number of data points required to equivalently fill a five-dimensional domain would be 100^5, or around ten billion. That would require more than 180 GB of dedicated hard drive space to store the sampled data.

As we will see, basing a density estimate on such a large volume of data is sometimes necessary but can give rise to a solution that is so unwieldy in size that it is both difficult to store and time consuming to evaluate. This is a considerable barrier to ease of use.

B. Framework for Understanding Density Estimation Techniques

All density estimation techniques conceptually achieve two objectives: First, they must use the data to to identify the space of solutions that solves the problem. Typically, this begins with the estimation of statistical characteristics of the stochastic process that generated the data. Second, they must apply additional techniques to select the most likely pdf from the space of possible solutions. This phase can include regularization techniques or incorporation of *a priori* information about the problem.

1) Objective 1: Application of Data Toward Density Estimation: The statistical characteristics of a stationary random

Manuscript received October 25 2001; revised July 31, 2003.

The authors are with the Radiation Laboratory, University of Michigan, Ann Arbor, MI 48109-2122 USA (e-mail: ulaby@umich.edu).

Digital Object Identifier 10.1109/TGRS.2003.821068

process can be represented by either an infinite set of joint moments, or by a continuous differentiable pdf. Both representations contain equivalent information, since joint moments can be converted into a pdf, and a pdf can be converted into joint moments.

From the data, we can either estimate a finite set of joint moments or estimate the value of the pdf at a finite set of discrete points in its domain; such estimates are frequently used as the beginning of the process of producing a density estimation, although other techniques exist. Consider a set of random variables $X_1, X_2, \ldots, X_d$ whose statistical relationship is specified by $f(\mathbf{x})$. Also define $\boldsymbol{v}_j = (v_{j1}, v_{j2}, \ldots v_{jd})$ to be the jth of N realizations of the random variables in a dataset.

- We can estimate a finite set of moments and joint moments. If the random variables being observed for each trial are $\{X_1 X_2 X_3 \ldots X_d\}$, the moment $\xi_{\boldsymbol{m}}$ is defined by

$$\xi_{\boldsymbol{m}} = E\left[X_1^{m_1} X_2^{m_2} X_3^{m_3} \ldots X_d^{m_d}\right] \quad (1)$$

where $\boldsymbol{m} = (m_1, m_2, \ldots, m_d)$ is a vector describing which moment or joint moment is being computed. We can estimate the moment $\xi_{\boldsymbol{m}}$ by computing its sample joint moment

$$\frac{1}{N} \sum_{j=1}^{N} \prod_{i=1}^{d} [v_{ji}]^{m_i}. \quad (2)$$

Many parametric techniques rely on estimates of moments or joint moments.

- We can estimate the pdf at discrete points. A pdf evaluated at a discrete point $\mathbf{p}$ would simply be the scalar value given by

$$f(\mathbf{p}). \quad (3)$$

We can estimate this value by computing the fraction of the total dataset $\{\boldsymbol{v}_j\}$ that is sufficiently close to $\mathbf{p}$. If there are n points that are sufficiently close to $\mathbf{p}$, *sufficiently close* encompasses an area A, and there are N total points in the dataset, $f(\mathbf{p})$ can be estimated by

$$\frac{n}{NA}. \quad (4)$$

Techniques for estimating $f(\mathbf{p})$ vary in the definition of the area A and the definition of *sufficiently close*. Histogram techniques always rely on estimates of the density at particular points.

2) Objective 2: Additional Techniques Used to Find a Unique Solution: To complete the density estimation, each technique must narrow a family of solutions to a single unique solution. Each solution may satisfy the requirements that the data places on it equally well, so other information or techniques must be brought to bear.

Wherever possible, density estimation techniques can use *a priori* knowledge of the density's shape and characteristics; techniques that make such assumptions are called parametric techniques, and typically reduce the volume of data required for accurate computation of density estimates by orders of magnitude. For example, if it is known that the character of a particular dataset is Gaussian, density estimation can be achieved by simply computing the sample mean and covariance matrix. This paper focuses on nonparametric density estimation, where little is known and few assumptions can be made about the overall statistical character of a particular dataset.

A common technique is the multidimensional histogram, where the domain of the pdf is divided into regularly spaced and shaped bins, the value of the pdf is computed at the center $(\mathbf{p})$ of each bin as $f(\mathbf{p})$. For all points inside the bin, $f(\mathbf{x}) = f(\mathbf{p})$; the value of the density estimation for the point at the center of the bin is assigned as the value for the rest of the points in the bin.

3) Literature on Density Estimation: Density estimation techniques can broadly be divided into the parametric versus the nonparametric types. The parametric density estimation techniques assume some mathematical model of the density underlying the data, and the method solves for some unknown parameters such as the mean or standard deviation. Nonparametric techniques, on the other hand, attempt to allow the data to specify what shape the pdf should assume.

Some of the mainstream nonparametric techniques are summarized below.

- The histogram approach divides the data domain into bins and measures the relative frequency of data inside each bin. This provides a rough estimate of the local density inside a bin. Histogram bins are typically the same size, but can be variable. Occasionally, this is called a variable partition estimate when the bin sizes are varied.
- The kernel density (KD) estimator is a smooth function that is the sum of many bell-shaped functions, one centered at each data point. KD estimators are introduced and discussed in many books such as [6] and [14].
- The orthogonal series estimator assumes that the histogram can be represented by a weighted sum of orthogonal basis functions, and uses this to estimate the coefficients of that decomposition.
- The maximum penalized likelihood estimator [15] attempts to maximize the likelihood of the density based on the data. Left alone, maximum-likelihood techniques will yield a set of delta functions, so a smoothing requirement is added (Thus, the penalized part of the technique), which yields a density estimate.
- The nearest neighbor estimator assigns densities based on bins centered on the region of interest, which contain a fixed number of points. Thus, the bin sizes vary inversely with the value of the density.

Details of these techniques, as well as a wealth of other sources can be found in [14] and additionally in [6]. These techniques are typically not easy to generalize to higher dimensions in a simple way. Even recent papers such as [5] do not often address multivariate density estimation, although the book by Scott [13] does. Edwin T. Jaynes has written many papers on the concept of maximum entropy, such as [9] and [10]. Other articles on maximum entropy include [7] and [16], while the use of maximum entropy in the context of density estimation has been reported in several studies, namely [1]–[3], [8], and [12]. Most closely related to our investigation is the research by Borwein *et al.* [1] and Ormoneit *et al.* [11]. The former paper demonstrates the technique in up to two dimensions, while the latter only demonstrates the technique on 1-D densities and does not discuss generalization to higher dimensions.

The next sections describe in more detail the kernel density estimation (KDE) technique and the formulation of the entropy moment problem, which we will use as the basis for the maximum-entropy density estimation (MEDE) technique described in this paper.

C. KDE Technique

The KDE technique constructs a density estimate from a set of data by placing a bell-shaped "kernel" centered at each data point $\boldsymbol{v}_j$. In terms of the framework given above, it is a variant of a histogram technique, where $A \rightarrow 0$ and n for any given point of evaluation includes fractional contributions from each data point that is near.

Thus, if the kernel is

$$K(x) = \begin{cases} K_o(1-x^2)^2, & \text{if } -1 < x < 1 \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

then the density estimate is given by

$$f(\mathbf{x}) = \sum_{j=1}^{N} K\left(|\mathbf{x} - \boldsymbol{v}_j|\right). \tag{6}$$

The KDE technique is described in [14]. It is simple, commonly used, and easily extendable to higher dimensions. For these reasons, we will use it as the basis of comparison in this paper. We will also find that it suffers from some drawbacks, notably that it is computationally intensive and tends to show evidence of lobes and bumps that do not necessarily exist in the true pdf. The shape and width of the kernel are important parameters.

D. Entropy Moment Problem

Another approach to density estimation is to formulate the problem as what is known as the *entropy moment problem.* This formulation assumes that we have computed estimates of the moments and joint moments from our dataset, as follows:

$$\xi_{\boldsymbol{m}} = \frac{1}{N} \sum_{j=1}^{N} \prod_{i=1}^{d} [v_{ji}]^{m_i}. \tag{7}$$

The relationship between the pdf and the moments are given in general by

$$\int \prod_{i}^{d} x_i^{m_i} f(\mathbf{x}) d\mathbf{x} = \xi_{\boldsymbol{m}} \tag{8}$$

where the $\xi_{\boldsymbol{m}}$s are the estimated joint moments, and the only unknown in this set of integral equations is $f(\mathbf{x})$. As mentioned before, a finite set of integral equations of this form does not have a unique solution, but rather a family of solutions. We would like to simultaneously satisfy these equations and maximize the entropy so as to produce a unique solution to our problem.

In terms of the framework for understanding density estimation problems given above, this is the application of the data to produce a family of solutions. The specific solution will be chosen by finding the maximum-entropy solution; Appendix A describes the justification for using entropy, and why the maximum-entropy solution is appropriate.

The entropy of $f(x)$ will be written as $H(f(x))$. Because it can be proven that the entropy $H(f(x))$ is a concave function of $f(x)$, we can be confident that there exists an f_m that maximizes the value of $H(f(x))$ and satisfies our moment constraints. To maximize $H(f(x))$ subject to our constraints, we can use the method of Lagrange multipliers to maximize

$$J\left(\mathbf{x}, f(\mathbf{x}), \{\lambda_{\boldsymbol{m}}\}\right) = H\left(f(\mathbf{x})\right) + \sum_{\boldsymbol{m}} \int \lambda_{\boldsymbol{m}} \left[\prod_{i=1}^{d} x_i^{m_i}\right] f(\mathbf{x}) d\mathbf{x} \tag{9}$$

which can be written as

$$J\left(\mathbf{x}, f(\mathbf{x}), \{\lambda_{\boldsymbol{m}}\}\right) = \int \overbrace{\left[-f(\mathbf{x}) \log f(\mathbf{x}) + \sum_{\mathbf{m}} \lambda_{\boldsymbol{m}} \left[\prod_{i=1}^{d} x_i^{m_i}\right] f(\mathbf{x})\right]}^{Q(\mathbf{x}, f(\mathbf{x}))} d\mathbf{x}. \tag{10}$$

Next, we define the integrand as $Q(\mathbf{x}, f(\mathbf{x}))$, and define $f(\mathbf{x}) = f_m(\mathbf{x}) + \epsilon\eta(\mathbf{x})$, where $f_m(\mathbf{x})$ is the function that maximizes $Q(\mathbf{x}, f(\mathbf{x}))$, $\eta(\mathbf{x})$ is any well-behaved nonzero function we choose, and ϵ is a constant. When $\epsilon = 0$, $f(\mathbf{x})$ maximizes the integral of interest. Applying the calculus of variations, we take the partial derivative with respect to ϵ

$$\frac{d}{d\epsilon} \int Q\left(\mathbf{x}, f(\mathbf{x})\right) d\mathbf{x} \bigg|_{\epsilon=0} = 0 \tag{11}$$

$$\text{(by Liebnitz Rule)}, \int \frac{\partial}{\partial\epsilon} Q\left(\mathbf{x}, f(\mathbf{x})\right) d\mathbf{x} \bigg|_{\epsilon=0} = 0 \tag{12}$$

$$\int \left[\frac{\partial}{\partial f(\mathbf{x})} Q\left(\mathbf{x}, f(\mathbf{x})\right)\right] \frac{\partial f(\mathbf{x})}{\partial\epsilon} d\mathbf{x} \bigg|_{\epsilon=0} = 0 \tag{13}$$

$$\int \left[\frac{\partial}{\partial f(\mathbf{x})} Q\left(\mathbf{x}, f(\mathbf{x})\right)\right] \eta(\mathbf{x}) d\mathbf{x} \bigg|_{\epsilon=0} = 0. \tag{14}$$

Note that setting $\epsilon = 0$ does not change the integral at all. Since $\eta(\mathbf{x})$ is an arbitrary well-behaved nonzero function, $f(\mathbf{x})$ must satisfy the condition $(\partial/\partial f(\mathbf{x}))Q(\mathbf{x}, f(\mathbf{x})) = 0$. Solving this equation leads to

$$f(\mathbf{x}) = e^{P(\mathbf{x})} \tag{15}$$

where $P(\mathbf{x}) = \sum_{\boldsymbol{m}} \lambda_{\boldsymbol{m}} [\prod_{i=1}^{d} x_i^{m_i}]$. Note that this polynomial could be a Taylor series expansion of an arbitrary function, and that each term in the polynomial corresponds to a single constraint. Thus, there are as many unknown coefficients $(\lambda_{\boldsymbol{m}})$ as there are integral constraints.

We substitute $f(\mathbf{x}) = e^{P(\mathbf{x})}$ into the original set of equations

$$\int e^{P(\mathbf{x})} d\mathbf{x} = \xi_{\boldsymbol{m}}. \tag{16}$$

The problem of solving these integrals for $\lambda_{\boldsymbol{m}}$ is the entropy moment problem.

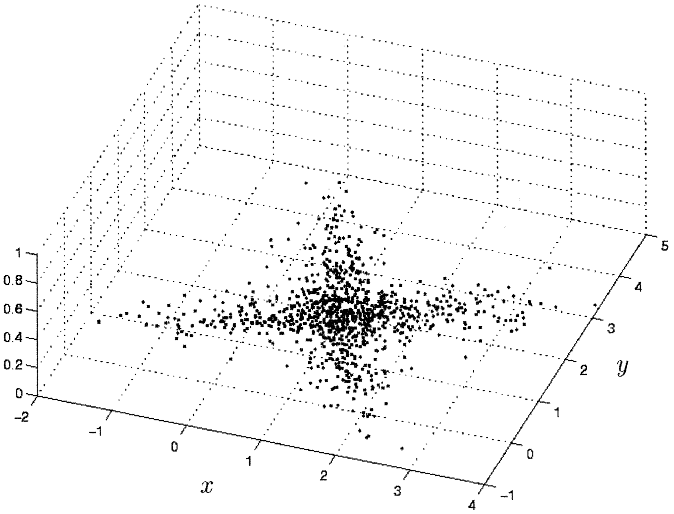


Fig. 1. Dataset of realization from a 2-D density.

II. MEDE: A Novel Solution to the Entropy Moment Problem

Despite the conceptually straightforward treatment of density estimation using maximum-entropy ideas as presented in Section I-D, solving for the coefficients of $P(\mathbf{x})$ is not computationally trivial. This paper proposes a novel solution to the entropy moment problem that uses the functional form of an exponentiated polynomial, but solves for the coefficients by applying alternate constraints. The alternate constraints that we propose are regularly spaced estimates of the pdf throughout its domain; it is reasonable to apply the new set of constraints because the statistical information that they estimate is equivalent to the statistical information in a finite set of moments and joint moments. Applying ideal interpolation to the constraints and then applying a technique that we have called a Legendre transform, we convert the constraints into the coefficients that we seek.

The Legendre transform procedure can be explained by making an analogy with a multidimensional Fourier transform representation. In the terminology of the Fourier transform, we first sample the function in its domain and then transform it to its frequency domain representation. The samples of the function in its domain for our analogy correspond to the "noisy density samples" we just discussed. The basis functions we use, rather than being complex exponentials, are Legendre basis functions. The frequency domain representation in our analogy still corresponds to a weighting of our basis functions, but these weighting coefficients to the Legendre basis functions are exactly the coefficients that we seek in our solution. This technique is computationally efficient, accurate, and easily generalizable to multivariate data.

A. Developing the Constraints

The alternate constraints are simply estimates of the value of the underlying pdf on a grid of points; they can be found by subdividing the domain into hypercubes and counting the fraction of points within each cube. The procedure is exactly the first step of producing a histogram.

These constraints are a series of scalar valued points, one at the center of each bin. The bin size depends on the size and nature of the dataset, the extent or spread of the data throughout measurement space, and the level of detail required in the density estimate. Given no *a priori* knowledge, there is currently no automated optimal way of choosing bin dimensions, so in a practical scenario, we depend on the user's expertise and familiarity with the measurement space and the amount of data available to guide the choice of this parameter.

To produce the estimates, we take each data point and compute the point that represents the center of the histogram bin that contains it. If $\boldsymbol{v}_j$ is our data point, l_d is the minimum value of our domain in dimension d, and s_d is the length of a side of the cube in dimension d, then

$$c_d = \left\lfloor \frac{a_d - l_d}{s_d} \right\rfloor s_d + \frac{s_d}{2} \tag{17}$$

is the dth coordinate of the histogram bin that the point falls in.

Each bin's scalar value is determined by counting the number of points inside it (n), dividing by the total number of points in our dataset (N), and normalizing it so that the whole histogram integrates to one. The value assigned to the point at the center of each bin is given by $(n/N)(\prod_d s_d)^{-1}$.

B. Implementing the Legendre Transform

We would like to find the set of coefficients that produce a pdf that maximizes the entropy, while still taking on the values at the points we have sampled. Earlier, we described this solution as a transformation, since we are converting samples to weighting coefficients for orthogonal basis functions. The approach we are proposing is to examine 1-D slices from the sampled density and expand them in terms of convenient basis functions. The coefficients of these basis functions can easily be collected and expanded in terms of the next dimension. Thus, if we were working with a two-dimensional (2-D) pdf, whose orthogonal axes were x and y, the first expansion produces many densities of the form $e^{P_0+P_1x+\cdots+P_\kappa x^\kappa}$, one for each y coordinate. We can collect these together, and each coefficient can again be expanded in terms of basis functions; so, we could write $P_0 = \sum_n R_{n,0}\Phi_n(y)$, $P_1 = \sum_n R_{n,1}\Phi_n(y)$, etc. For higher dimensional densities, the process can be repeated, expanding the existing coefficients in terms of the next dimension's variable.

Let us be more precise about this algorithm, which does most of the work of converting a dataset into the analytical density that we desire.

C. Reduction Algorithm

The reduction algorithm is used to reduce the histogram to a grid of coefficients that, when used with $f(\mathbf{x}) = e^{P(\mathbf{x})}$, describe a smooth, analytical density function. We will describe the reduction algorithm through a 2-D example, where we used the dataset pictured in Fig. 1. Our first step is to convert the dataset into a histogram representation. Graphically, we can imagine the histogram looking like Fig. 2, and as a sparse numerical representation, we can visualize it as in Table I.

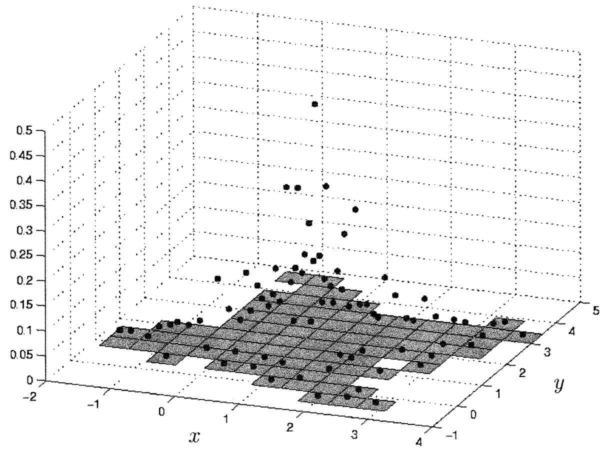


Fig. 2. Graphical histogram representation of our dataset. The height of each point above the domain corresponds to the level of probability that a data point is in the specified bin.

TABLE I
NUMERICAL HISTOGRAM REPRESENTATION OF OUR DATASET

			0.009	0.004								
				0.009	0.004							
			0.128	0.148	0.004	**0.004**					0.014	
			0.049	0.104	0.163	**0.019**	0.014	0.004	0.009	0.004	0.014	0.004
			0.009	0.044	0.307	**0.084**	0.109	0.039	0.004	0.004	0.019	
			0.009	0.034	0.287	**0.198**	0.128	0.049	0.004	0.004	0.004	
	0.019	0.004	0.014	0.054	0.198	**0.480**	0.109	0.024	0.009	0.009		
0.024	0.019	0.054	0.004	0.054	0.114	**0.297**	0.228	0.019		0.039		
0.004	0.004	0.029	0.019	0.059	0.034	**0.109**	0.257	0.074	0.049			
		0.019		0.004	0.009	**0.019**	0.049	0.128				
						0.004	0.004	0.074	0.074			
								0.009	0.019	0.014		

Next, we choose a dimension and take 1-D slices along it, as shown in Fig. 3, plotting each of these slices as shown in the top of Fig. 4. Due to the construction of the histogram, we are implicitly assuming that any unspecified points (i.e., the value of $f(y|X = x)$ when $y = 5$) left out of our description have a value of zero. To build this assumption into our calculations, we will insert samples with zero (or nearly zero) value which bracket any nonzero samples. This "pins" our function, forcing it to be well-behaved and enforcing an implicit assumption of finite support in our domain.

With a sampled 1-D function $f(y_n|X = x)$ taken at values $\{y_n\}$, we can use interpolative techniques (ideal interpolation or even linear interpolation) to come up with an approximation for the underlying continuous function, which we will call $f_a(y|X = x)$. Because we expect the functional form of this pdf to be an exponentiated polynomial, (according to maximum-entropy arguments) we can reasonably represent the logarithm of our density function as a weighted sum of Legendre functions, $\{\Phi_m(y)\}$. Furthermore, because Legendre polynomials are orthogonal, then we can write

$$\int \Phi_n(x)\Phi_m(x)dx = \begin{cases} h_m^2, & m = n \\ 0, & \text{otherwise} \end{cases} \tag{18}$$

and use this to decompose $f_a(y|X = x)$ into a set of coefficients a_n by using

$$a_n = \frac{1}{h_m^2} \int \ln f_a(y|X = x)\Phi_n(y)dy. \tag{19}$$

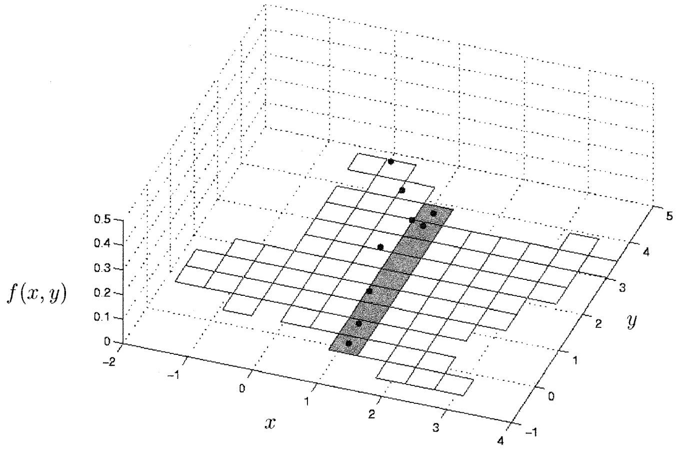


Fig. 3. One-dimensional slice of our histogram.

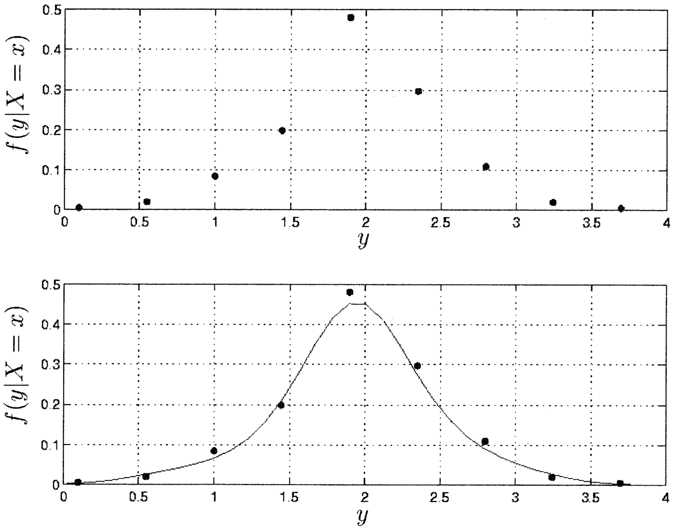


Fig. 4. (Top) Samples of the 1-D slice in Fig. 3. (Bottom) Best set of weighting coefficients for our basis functions so that samples of the final representation match the samples taken from the histogram. This can be thought of as a projection of a function defined by the original samples onto a Legendre basis representation.

With these a_n, we can reconstruct $f_a(y|X = x)$ by writing

$$\ln \tilde{f}(y|X = x) = \sum_i a_i \Phi_i(y) \tag{20}$$

where $\ln \tilde{f}(y|X = x)$ is now the projection of $\ln f_a(y|X = x)$ onto our set of basis functions. This transformation converts samples of a density to coefficients of a Legendre polynomial, which when plotted, closely matches the sampled points as in the lower plot of Fig. 4.

Repeating this procedure for each slice of our pdf would result in Fig. 5, with a tabular representation as in Table II. This entire procedure is repeated d times, one for each dimension of the pdf, taking slices that are geometrically orthogonal to the previous set of slices. During the second and subsequent iterations, the slices can no longer be thought of in the density domain, but instead can be thought of as taken in the domain of Legendre coefficients. Zero values used to "pin" the function in its domain must not be identically zero, but must be values that

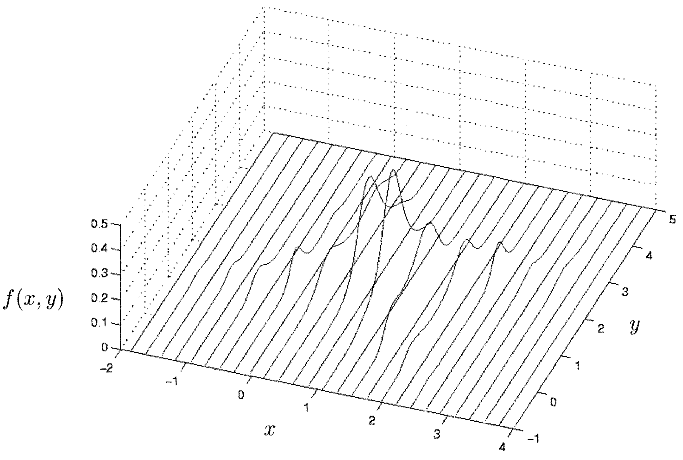


Fig. 5. All slices of histogram bins have been converted into a continuous 1-D functions.

evaluate to zero, once the coefficients are plugged into the appropriate basis functions.

The bold entries in Table II show the next slice during the second iteration of our example. Fig. 6 shows the coefficient slice being plotted, as before and projected into the space of Legendre basis functions.

In our 2-D example, the reduction algorithm gets run only twice, converting our histogram into the grid of numbers in Table III. These numbers are coefficients that describe our pdf, pictured in Fig. 7 as a 2–D surface.

III. Results

In order to evaluate the performance of the MEDE technique, we took various oddly shaped densities that we knew very well, produced randomly generated datasets from them, and then estimated their densities. To select kernel widths (w for the kernel density estimator) and bin sizes (s_d for the MEDE technique) we experimented with various values for these parameters, attempting to find a minimum Kullback–Liebler distance [4] between the estimate and the actual density. The results reflect the best parameter found for each technique, although in practice the actual density would not be known and bin dimensions would have to be chosen by taking into account the amount and density of data in the measurement domain, as well as the detail required in the final estimate. Theoretical comparisons of the storage efficiency and the computational efficiency of the two density estimation techniques were also made.

The results are first presented visually, and then numerically, in table format. For ease of reference, we will call the three test pdfs the X-density, the Y-density and the XV-density. The X- and Y-densities are 2-D densities that are mixtures of bivariate Gaussians. The XV-density is a three-dimensional density that can be thought of as a cylinder whose center is curved, giving it an overall banana-like shape (as shown later in Fig. 10). The cross section of this cylinder is always a mixture of the X- and V-densities such that the cross section is the X-density on one end and the V-density on the other, and half of each in the middle. Appendix B gives the detailed functional forms for each of the densities that were used.

TABLE II
Each Column Is a Vector of Coefficients Weighting Legendre Basis Functions for a Single Slice of the Density. The Bold Entries Represent a Slice of the Coefficients Used in the Second Pass of the Reduction Algorithm

0.0	0.0	0.7	0.4	-0.4	-1.0	-0.1	0.1	-0.1	0.3	0.0	-0.3	0.3	-0.5	-0.3	0.0	0.0
0.0	0.0	1.1	0.0	-0.5	-1.7	-0.4	-0.0	-0.4	-0.6	-0.2	1.1	-0.3	0.6	-1.3	0.0	0.0
0.0	0.0	-1.7	-1.1	0.1	-1.4	0.1	0.6	0.5	0.0	-0.5	-1.5	-1.8	0.3	-0.5	0.0	0.0
0.0	0.0	-0.3	0.8	0.3	2.1	-0.1	-0.3	0.8	-0.2	-0.2	-0.6	0.3	-0.3	1.0	0.0	0.0
0.0	0.0	2.2	1.3	0.6	3.0	-0.0	-0.8	-0.4	-0.5	0.2	1.2	2.1	0.3	1.2	0.0	0.0
0.0	0.0	-0.9	-1.9	-1.0	-0.4	-0.1	-0.4	0.0	1.8	1.2	0.6	-1.0	0.8	-0.2	0.0	0.0
0.0	0.0	-1.9	-0.8	-1.1	-0.6	-0.0	-0.1	-0.3	0.1	0.7	1.2	0.8	-0.9	-1.4	0.0	0.0
0.0	0.0	2.1	2.9	2.4	0.9	1.7	1.0	-0.8	-2.0	-2.6	-3.1	-0.9	-2.0	-0.7	0.0	0.0
0.0	0.0	0.8	-0.3	0.6	-2.1	0.4	0.6	0.5	0.1	-1.0	-2.3	-3.3	0.5	1.0	0.0	0.0
0.0	**0.0**	**-2.8**	**-3.3**	**-3.8**	**-2.9**	**-2.5**	**-0.1**	**0.3**	**-0.1**	**2.5**	**5.2**	**4.3**	**3.4**	**1.6**	**0.0**	**0.0**
0.0	0.0	0.8	2.0	1.2	1.3	-0.8	1.3	1.6	1.8	0.3	0.2	1.4	1.1	0.0	0.0	0.0
0.0	0.0	3.0	3.1	4.5	-0.6	-1.4	-1.8	1.3	3.3	2.4	-0.5	-1.4	-4.2	-2.3	0.0	0.0
0.0	0.0	-4.6	-5.7	-6.2	-7.3	-8.2	-10.0	-10.1	-9.7	-8.0	-6.1	-5.9	-5.9	-3.5	0.0	0.0
0.0	0.0	-1.5	-1.4	-2.3	2.3	2.6	1.4	-1.2	-2.4	-3.3	-2.1	-1.3	2.2	1.3	0.0	0.0
-15.0	-15.0	-11.2	-10.7	-10.0	-7.6	-6.3	-6.6	-6.6	-7.1	-6.9	-8.1	-8.6	-10.1	-11.8	-15.0	-15.0

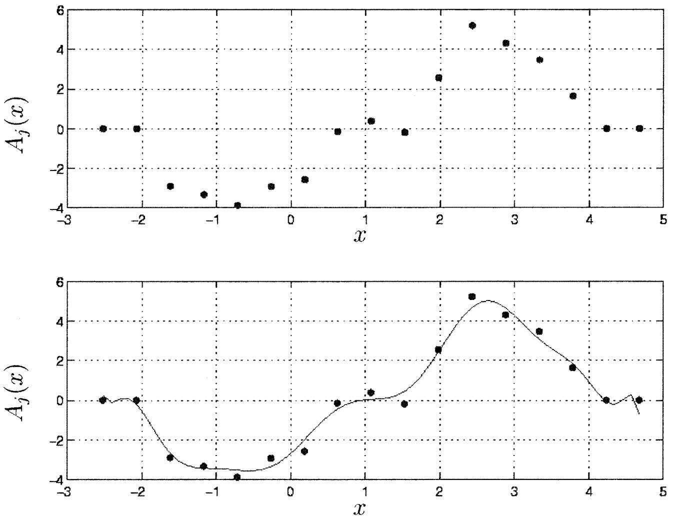


Fig. 6. In the second pass of the reduction algorithm, slices are no longer in the density domain. They are in the coefficient domain. These curves correspond to the entries in bold in Table II.

A. Qualitative Comparison

Fig. 8 shows a comparison using the 2-D analytical V-density, while Fig. 9 shows the same results, but for the X-density. Figs. 8(a) and 9(a) show the analytical densities, as viewed from above, where the different shades of gray differentiate between probability levels, and Figs. 8(b) and 9(b) show datasets each containing 500 randomly generated data points produced from the original density. Using the data points plotted in 8(b) and 9(b), we produced estimates of the underlying density by using both the kernel density estimator, and the maximum-entropy density estimator. The results are plotted in Figs. 8(c) and (d) and 9(c) and (d), respectively. A similar exercise was conducted on the basis of 1000 data points, and the results are displayed in Figs. 8(e) and (f) and 9(e) and (9f).

One of the advantageous features of the MEDE technique that can be observed most clearly when comparing Fig. 9(e) with (f) is that it clearly has fewer spurious lobes than the kernel density estimator. Although the KDE technique can be run with a

TABLE III
FINAL RESULT OF THE REDUCTION ALGORITHM: A GRID OF COEFFICIENTS WHICH WHEN USED WITH OUR FUNCTIONAL FORM GIVEN BY (15) YIELD THE SURFACE IN FIG. 7

-9.8e-00	3.2e-02	-6.5e-00	-9.6e-02	3.5e-01	-1.4e-01	4.6e-01	6.9e-01	5.8e-01	-6.7e-01	8.6e-01	-6.9e-02	-9.6e-01	4.6e-01	-2.8e-01
-3.5e-01	2.5e-01	5.3e-01	1.2e-00	-4.1e-02	-4.0e-00	-3.2e-01	3.5e-00	-1.1e-02	-7.2e-01	8.6e-01	-4.7e-01	-1.2e-00	3.6e-01	5.2e-01
-5.7e-00	2.5e-01	7.0e-00	-3.2e-01	-3.4e-01	7.9e-02	5.2e-01	-2.4e-01	-2.5e-00	1.8e-01	4.9e-01	4.0e-01	6.2e-01	-4.2e-01	2.0e-01
3.3e-01	-1.8e-00	-5.1e-01	1.3e-00	3.0e-01	3.7e-00	-2.6e-01	-4.8e-00	3.5e-01	1.7e-00	-7.0e-01	-5.5e-01	1.0e-00	5.4e-01	-4.3e-01
8.0e-01	-2.1e-01	-7.5e-01	1.0e-01	-4.9e-01	4.6e-01	1.0e-01	-4.7e-01	1.2e-00	1.8e-01	-1.5e-00	-3.4e-01	8.3e-01	4.8e-01	-2.8e-01
1.0e-01	3.0e-00	-3.6e-01	-4.8e-00	6.1e-02	1.0e-00	7.9e-01	1.3e-00	-7.4e-01	-8.4e-01	-1.2e-01	1.0e-00	3.9e-01	-1.2e-00	-3.4e-01
-2.7e-01	-3.2e-01	5.8e-01	8.9e-01	4.0e-01	-8.5e-01	-2.2e-00	-4.9e-02	2.0e-00	7.7e-01	-1.1e-01	-3.0e-01	-7.6e-01	-7.8e-01	5.1e-01
-7.0e-02	-1.8e-00	7.0e-01	3.0e-00	-1.0e-00	-9.5e-01	4.7e-02	1.5e-01	6.8e-01	-1.1e-00	-3.9e-01	9.3e-01	7.5e-02	4.8e-02	2.4e-01
-2.9e-01	4.1e-01	-4.9e-01	-8.5e-01	1.0e-00	5.7e-01	7.3e-01	-1.0e-02	-1.3e-00	-4.8e-01	-6.8e-02	6.2e-01	2.1e-01	-1.7e-01	4.8e-01
-9.7e-02	6.3e-01	-4.5e-01	-1.0e-00	1.1e-00	3.0e-01	-4.9e-01	-2.2e-01	-3.7e-01	1.0e-00	5.4e-01	-1.1e-00	-5.7e-01	3.3e-01	4.3e-01
6.7e-01	-2.3e-01	1.7e-01	1.6e-01	-1.9e-00	2.3e-01	1.2e-00	5.5e-02	-6.9e-01	-3.3e-01	1.0e-00	-3.3e-01	1.2e-01	9.7e-01	-1.4e-00
1.9e-01	-1.4e-01	-5.4e-02	6.5e-01	-3.7e-01	-7.5e-01	2.7e-01	-1.5e-01	-3.5e-02	6.8e-01	-1.0e-01	-2.7e-01	4.8e-01	3.3e-01	-8.0e-01
-4.2e-01	1.0e-01	-8.7e-02	2.9e-01	1.1e-00	-8.8e-01	-8.5e-01	2.1e-01	6.5e-01	6.9e-01	-8.7e-01	-2.4e-01	5.8e-03	-6.6e-01	8.9e-01
-1.6e-01	-5.7e-02	3.4e-01	-5.8e-01	-2.4e-01	1.0e-00	-4.5e-02	2.2e-01	2.6e-01	-1.2e-00	-9.0e-02	4.8e-01	-2.8e-01	-2.4e-01	2.8e-01
-5.9e-02	-1.3e-01	1.3e-01	-1.6e-01	7.6e-02	6.8e-01	-4.6e-01	-1.6e-01	4.5e-01	-6.0e-01	-8.0e-02	3.4e-01	-1.2e-01	2.0e-01	2.5e-01

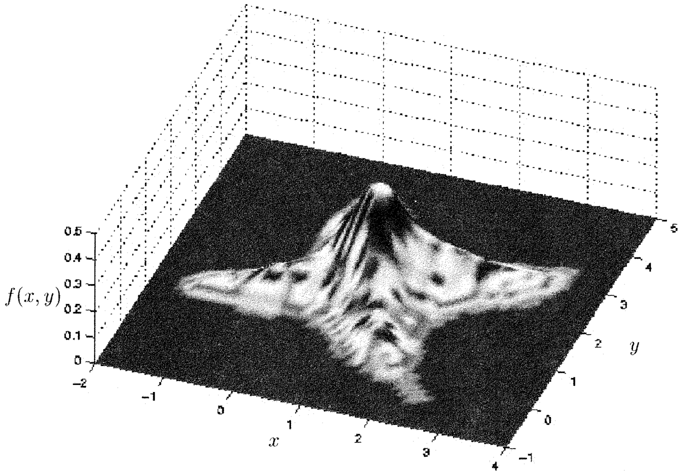


Fig. 7. Estimate of the underlying density for the dataset in Fig. 1.

larger kernel width, effectively smoothing out the lobes that we see, this increases the numerical error between the actual density and the estimate. In Fig. 9(e), the kernel density estimator is responding to the distribution of data in the particular sampling of our pdf, while the MEDE technique better represents the shape of the underlying pdf.

B. Quantitative Comparison

This section quantifies the performance of the maximum-entropy technique as compared with the kernel density technique. We consider various measures of the accuracy of the estimate, as well as computational efficiency and sparseness of representation.

1) Estimate Accuracy: The first concern we have is in measuring how close we have come to the original density we are trying to estimate. There are many ways to measure how different one function is from another. Our most important measure is the Kullback–Liebler distance, because not only is it specifically designed to measure the distance between two pdfs, it is directly related (through Stein's Lemma, discussed in [4]) to the probability of error in certain classification problems. The Kullback–Liebler distance is defined in [4] as

$$D\left(f(x)\|g(x)\right) = \int f(x) \log \frac{f(x)}{g(x)} dx. \tag{21}$$

Another measure of interest is the absolute error fraction between the original density and the estimate, which we define as

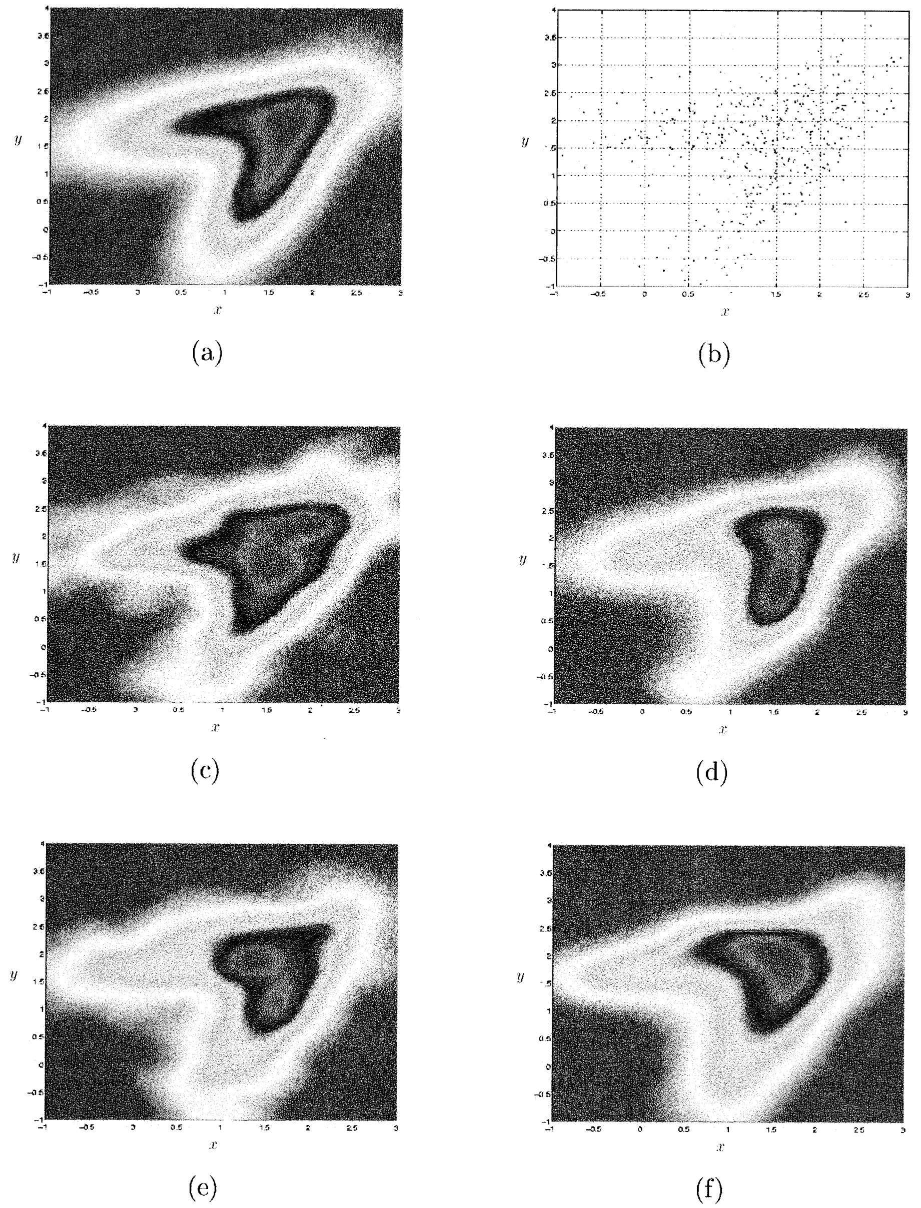
$$\delta = \int |f(x) - g(x)|\, dx. \tag{22}$$

2) Computational Complexity: Once we have estimated our density function, the simplest operation we can perform on it is to evaluate it at a single point. One of the advantages of the maximum-entropy representation is that one can, to some extent, control the complexity of the representation by specifying the number of coefficients to use. This complexity of the representation is directly related to the complexity of evaluating the density at a point. The fewer coefficients, the more computationally efficient, and the sparser the representation. However, as the number of coefficients goes down, so does the ability to represent details in the density. We will compare the maximum-entropy density estimate with the kernel density estimate in terms of computational efficiency by evaluating the density at a particular point. The fundamental unit of computational complexity is the number of multiply-adds, and we will assume that all computations are performed in the most efficient way possible, using table lookups to reduce floating-point operations. We will also assume that we are working with a multivariate dataset in d dimensions that has N points in it. The maximum-entropy estimator has c coefficients to represent the density in each dimension.

Consider a pdf represented by an exponentiated polynomial whose coefficients can be organized as a multidimensional grid with c coefficients on a side. For each evaluation, the maximum-entropy density has to find Legendre polynomials for each order from zero to $c-1$. These can be done by table lookup. It is easiest to imagine this geometrically, where c^d is the number of coefficients we have, and we group 1-D slices of coefficients of size c together. Thus, we have a total of $(c^d/c) = c^{d-1}$ slices. With a total of c multiply-adds per slice, with c^{d-1} slices, we compute a total of $cc^{d-1} = c^d$ multiply-adds. The technique repeats this process $d-1$ more times, as it reduces the dimensionality of the coefficient grid in each iteration. The total computational cost is simply the sum of computations from each dimension, $\sum_{i=1}^{d} c^i$ multiply-adds. On the other hand, the kernel density estimator must compute the distance of the point of interest from each of the data points; each distance computation is d multiply-adds so the total computational cost of a single evaluation is dN multiply adds. Thus, the MEDE technique is more computationally efficient than the kernel density estimator when

$$\sum_{i=1}^{d} c^i < dN. \tag{23}$$

Fig. 11 is a graphical representation of (23). For a particular number of coefficients c, each curve indicates the size of the



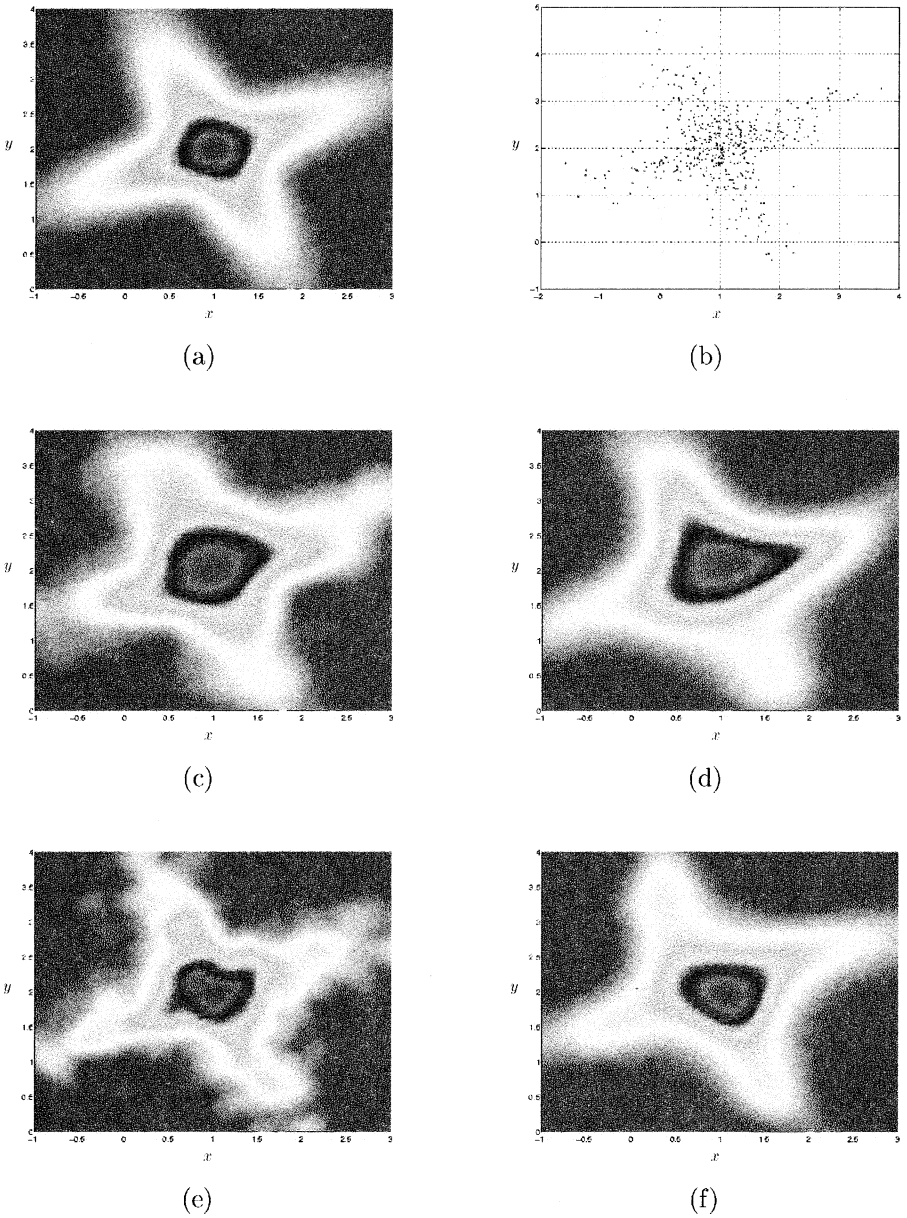
(a) (b) (c) (d) (e) (f)

Fig. 8. Results of various attempts at density estimation using different techniques and different sized datasets. (a) The true underlying smooth, analytical pdf of the V-density. (b) A set of 500 data points produced as samples of the V-density. (c) Kernel density estimate using the dataset in (b). (d) MEDE technique using the dataset in (b). (e) Kernel density estimate using a dataset with 1000 points. (f) MEDE technique using a dataset with 1000 points.

dataset (N) above which the MEDE technique is more efficient than the kernel density estimator.

At this point, the kernel density estimator appears more computationally efficient, because it grows linearly with d, while, the maximum-entropy technique grows exponentially with d. However, one must consider that when using the MEDE technique, c and N should be related, and that c cannot be chosen arbitrarily. It turns out that one can almost always choose c appropriately so that the MEDE technique is the more computationally efficient one.

Once we consider the relationship between c and N, we will see that the MEDE technique is usually more efficient than the kernel density estimator. To incorporate the relationship between c and N into our complexity comparison, we should note that the number of coefficients in each dimension c must grow roughly proportionally with the number of bins in that dimension. The reason for this is that adding a coefficient to a polynomial allows us to represent one more bump in a 1-D curve. The 1-D curve we are referring to is that produced by decomposition of a slice of bins from our sampled histogram. Thus, we can relate how many bumps we should be trying to represent with our sample spacing. In addition to that, on average, there should be some minimum number of points, p per dimension, in each bin in the slice—otherwise, we have chosen an inappropriate bin size. Thus, if there are on average c nonzero bins in a single slice, and we require there to be on average p^d points in each bin (the number of points in a bin should go up with the dimensionality of the problem), then there are cp^d points in each slice. By definition, there are c^{d-1} slices, and we can thus conclude that when performing



(a) (b) (c) (d) (e) (f)

Fig. 9. (a) True underlying smooth analytical pdf of the X-density. (b) A set of 500 data points produced as samples of the X-density. (c) Kernel density estimate using the dataset in (b). (d) MEDE technique using the dataset in (b). (e) Kernel density estimate using a dataset with 1000 points. (f) MEDE technique using a dataset with 1000 points.

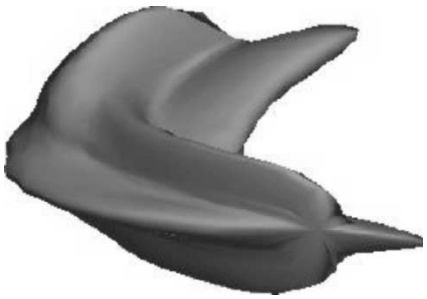


Fig. 10. Equiprobable surface of the XV-density, the solution to $f_{XV}(\mathbf{x}) = 0.012$.

the maximum-entropy density estimate, we must choose c so it satisfies

$$(pc)^d \leq N. \tag{24}$$

The MEDE technique is most computationally inefficient when c is chosen such that equality holds in (24). Accordingly, we assume that $N = (pc)^d$ in (23) and our inequality becomes

$$\sum_{i=1}^{d} c^i < d(pc)^d. \tag{25}$$

The parameter p is the number of data points required in each histogram cube for each dimension of the problem. The larger we make p, the more efficient the MEDE technique will become. Setting $p = 1$ puts the MEDE technique at the greatest disadvantage compared to the KD estimator, but we make this substitution for ease of comparison. Thus, when

$$\sum_{i=1}^{d} c^i < dc^d \tag{26}$$

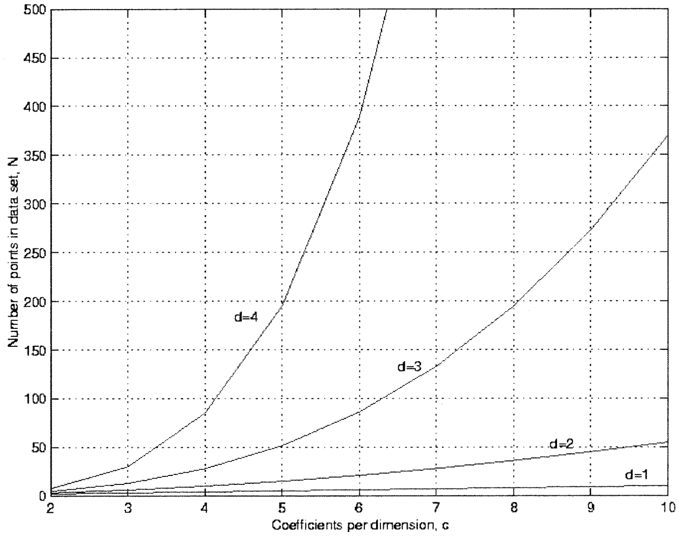


Fig. 11. This plot compares the relative efficiency of the kernel density estimator versus the MEDE technique. For each value of c, each curve indicates the value of N above which the MEDE technique is more efficient than the kernel density estimator.

we are guaranteed that the MEDE technique is more computationally efficient than the kernel density estimator. When d and c are integers greater than one, this is always the case.

The conclusion we can draw from these arguments is that if the dataset for which we are trying to estimate a density is very small and has high dimensionality (a case where one might think that the kernel density estimator would excel, computationally), there is simply not enough information in the dataset about the shape of the pdf to justify having a large number of coefficients in the maximum-entropy density estimate. For appropriately chosen values of c (taking N and d into account), the maximum-entropy density estimator is more computationally efficient than the kernel density estimator. As we will see in Table IV, for the example densities we tested the algorithms on, the maximum-entropy density estimate is two to seven times more computationally efficient than the kernel density estimate.

3) Sparse Representation: As mentioned above, controlling the number of coefficients c in the maximum-entropy estimator controls the sparseness of the representation. This may be important for storing the density estimates, if there are many of them. The kernel density estimator uses the whole dataset as representation, which for a high dimensional dataset with many points, can be unwieldy. This comparison is very similar to the one presented in the preceding section. In this case, the unit of storage is a double-precision floating-point number. The maximum-entropy estimate has c^d double-precision coefficients to represent it, while the kernel density estimator has Nd coefficients that represent it. The maximum-entropy representation is sparser when

$$c^d < dN. \tag{27}$$

TABLE IV
ACCURACY AND EFFICIENCY OF THE MEDE TECHNIQUE COMPARED WITH THE KDE TECHNIQUE. THE FIRST COLUMN INDICATES THE UNDERLYING pdf USED, AND THE SECOND COLUMN LISTS THE METHOD USED TO PERFORM THE DENSITY ESTIMATE ALONG WITH THE NUMBER OF POINTS IN THE DATASET

	Estimator	N	$\int \lvert f-g \rvert dx$	$D(f \parallel g)$	M–A	Stor
V–density (2 dim)	MaxEnt $c=15,\ s_d=0.6$	500	0.1968	0.0367	240	225
	Kernel $w=0.6$		0.1832	0.0462	1,000	1,000
	MaxEnt $c=15,\ s_d=0.6$	1,000	0.1678	0.0330	240	225
	Kernel $w=0.5$		0.1685	0.0370	1,000	2,000
X–density (2 dim)	MaxEnt $c=15,\ s_d=0.7$	500	0.2510	0.0570	240	225
	Kernel $w=0.5$		0.2315	0.0621	1,000	1,000
	MaxEnt $c=15,\ s_d=0.5$	1,000	0.1937	0.0369	240	225
	Kernel $w=0.4$		0.1470	0.0376	1,000	2,000
XV–density (3 dim)	MaxEnt $c=8,\ s_d=1.1$	1,000	0.6284	0.3958	584	512
	MaxEnt $c=10,\ s_d=1.1$		0.5749	0.3643	1,110	1,000
	Kernel $w=1.1$		0.5845	0.5088	3,000	3,000
	MaxEnt $c=12,\ s_d=1.1$	5,000	0.5826	0.3274	1,884	1,726
	Kernel $w=1.0$		0.5584	0.3354	15,000	15,000

Based on the discussion in the preceding sections, we can use the relationship we found between c and N. Substituting $N = c^d$ into the efficiency comparison yields the following condition in order for the maximum-entropy density estimator to have a sparser representation:

$$c^d < dc^d \tag{28}$$

which is always the case for more than one dimension.

4) Numerical Results: We now present numerical results of the estimates that we made, using the synthetically generated known densities described earlier in Section III. While reading Table IV, we keep in mind that the accuracy of our estimate is affected and limited by many variables, including the dimensionality of our data, the shape of the underlying density, and the number of points we are using to do our estimate. The M-A column is the number of multiply-adds it takes to compute the density at a particular point, and the Stor column is the number of double-precision storage units it takes to represent the density (which essentially is a measurement of sparseness of the representation).

The reader should note that Table IV indicates various important trends. These trends show that using the Kullback–Liebler distance as a measure of error, as we increase the complexity of the density estimation problem (either by increasing the dimensionality of our dataset, or by reducing the total samples we are using to make our estimate with), the advantage of using the MEDE technique over the kernel density estimator increases. In some cases, the differences in estimation accuracy between the two techniques are very small. In our examples, MEDE tends to have less error as measured by the Kullback–Liebler distance, while the kernel density estimator tends to produce less error as measured by the absolute error fraction $\int |f - g| dx$.

Table IV also shows an example of the flexibility afforded to us with the MEDE technique, in trading off accuracy for storage and computational efficiency. The second from last row shows results of the density estimate performed with different numbers of coefficients per dimension, namely $c = 8$ and $c = 10$. Choosing $c = 10$ can afford us slightly better accuracy as estimated by the absolute error fraction, significantly better accuracy as estimated by the Kullback–Liebler test, with one-third the storage and run-time requirements.

IV. Conclusion

From the perspective of density estimation, we are successful. Starting with a set of multivariate data, we have produced a smooth, analytical, scalar-valued pdf that is very likely close to the one that created the samples.

- Compared with the kernel density estimator, we have in many cases significantly reduced the number of coefficients, achieving a much sparser representation than before.
- Compared with the kernel density estimator, the maximum-entropy representation is, in many cases, much more computationally efficient to evaluate.
- Compared to the kernel density estimator, the maximum-entropy representation does not introduce spurious high-frequency components into the density being estimated.
- This whole procedure works for multivariate (multidimensional) densities with an arbitrary number of points in their datasets and is only limited by available computational resources.
- Depending on storage or computational requirements, fidelity and efficiency (storage and computational) may be traded off against each other by selecting different numbers of coefficients to use in the representation.

Appendix I
What is Entropy, and Why Maximize it?

Entropy is simply a scalar-valued function of a pdf. Thus, if $f(x)$ is a pdf, entropy is usually written as

$$H\left(f(x)\right) = -\int_{-\infty}^{\infty} f(x) \log f(x) dx.$$

Jaynes gave a succinct description of the utility of entropy in [10]:

> For many decades it has been recognized, or conjectured, that the notion of entropy defines a kind of measure on the space of probability distributions, such that those of high entropy are in some sense favored over others. The basis for this was stated first in a variety of intuitive forms: that distributions of higher entropy represent more "disorder," that they are "smoother," "more probable,"etc., and that they "assume less" according to Shannon's interpretation of entropy as an information measure, etc.

We will apply maximum-entropy techniques to choose the most likely pdf from among the possible densities that satisfy our constraints as described earlier. The densities so selected are smoother, more spread out, and the process does not assume more about the statistics that produced the data than it absolutely has to. In fact, it can be shown that when moment-like constraints are applied to the space of density functions, most of the solutions that satisfy those constraints are clustered around the maximum-entropy point. The proof is called the Concentration Theorem and is presented in [10].

TABLE V
Parameters Used to Define the X-Density and V-Density

Variable	V-density	X-density
ρ_d	0.5	0.8
σ_{xd}	1	1
σ_{yd}	0.5	0.5
μ_{xd}	1	1
μ_{yd}	2	2
ρ_g	0.7	-0.8
σ_{xg}	0.5	0.5
σ_{yg}	1	1
μ_{xg}	1.5	1
μ_{yg}	1	2

Appendix II
Description of X-, V-, and XV-Densities

Both the X- and V-densities are a mixture of two multivariate Gaussians

$$d(x,y) = \frac{1}{2\pi\sqrt{1-\rho_d^2}\sigma_{xd}\sigma_{yd}} e^{R_d(x,y)} \tag{29}$$

$$R_d(x,y) = \frac{-1}{2\left(1-\rho_d^2\right)}\left(\left(\frac{x-\mu_{xd}}{\sigma_{xd}}\right)^2 + \left(\frac{y-\mu_{yd}}{\sigma_{yd}}\right)^2 - \left(\frac{2\rho_d(x-\mu_{xd})(y-\mu_{yd})}{\sigma_{yd}\sigma_{xd}}\right)\right) \tag{30}$$

and

$$g(x,y) = \frac{1}{2\pi\sqrt{1-\rho_g^2}\sigma_{xg}\sigma_{yg}} e^{R_g(x,y)} \tag{31}$$

$$R_g(x,y) = \frac{-1}{2\left(1-\rho_g^2\right)}\left(\left(\frac{x-\mu_{xg}}{\sigma_{xg}}\right)^2 + \left(\frac{y-\mu_{yg}}{\sigma_{yg}}\right)^2 - \left(\frac{2\rho_g(x-\mu_{xg})(y-\mu_{yg})}{\sigma_{yg}\sigma_{xg}}\right)\right) \tag{32}$$

where

$$f(x,y) = \frac{1}{2}\left[d(x,y) + g(x,y)\right] \tag{33}$$

and the constants are defined in Table V.

If $z_l < z < z_u$, the XV-density is given by

$$f_{XV}(x,y,z) = f_X(x-x_o, y-y_o)C(z+6.5) + f_Y(x-x_o, y-y_o)C\left(-(z+2.5)\right) \quad (34)$$

where $z_l = -7.5$, $z_u = -1.5$, and

$$C(z) = \begin{cases} \frac{1}{5}\left(\cos(\pi z)+1\right), & -1 < z < 0 \\ \frac{1}{5}\left(\cos(\frac{\pi}{4}z)+1\right), & 0 < z < 4. \end{cases} \quad (35)$$

x_o and y_o are given by

$$x_o = 2\left(\frac{z-z_l}{z_u-z_l}\right) \quad (36)$$

$$y_o = -12\left(\frac{z-z_l}{z_u-z_l}\right)^2 + 14\left(\frac{z-z_l}{z_u-z_l}\right). \quad (37)$$

Otherwise, if z is outside the given range $f_{XV}(x,y,z) = 0$.

References

[1] J. M. Borwein and W. Huang, "A fast heuristic method for polynomial moment problems with Boltzmann–Shannon entropy," *SIAM J. Optim.*, vol. 5, no. 1, pp. 68–99, 1995.

[2] J. M. Borwein, A. Lewis, and D. Noll, "Maximum entropy reconstruction using derivative information, part 1: Fisher information and convex duality," *Math. Oper. Res.*, vol. 21, no. 2, 1996.

[3] L. Chao, "Multidimensional nonstationary maximum entropy spectral analysis by using neural net," *Math. Geol.*, vol. 31, no. 6, 1999.

[4] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley, 1991.

[5] R. de Bruin, D. Salomé, and W. Schaafsma, "A semi-Bayesian method for nonparametric density estimation," *Comput. Statist. Data Anal.*, vol. 30, pp. 19–30, 1999.

[6] L. Devroye, *A Course in Density Estimation*. Cambridge, MA: Birkhäuser, 1987.

[7] S. F. Gull, "Some misconceptions about entropy," in *Maximum Entropy in Action*. Oxford, U.K.: Clarendon, 1991.

[8] P. Hall and B. Presnell, "Density estimation under constraints," *Amer. Statist. Assoc.*, vol. 8, no. 2, pp. 259–277, 1999.

[9] E. T. Jaynes, "Information theory and statistical mechanics," *Phys. Rev.*, vol. 106, pp. 620–630, 1957.

[10] ——, "On the rationale of maximum-entropy methods," *Proc. IEEE*, vol. 70, pp. 939–952, 1982.

[11] D. Ormoneit and H. White, "An efficient algorithm to compute maximum entropy densities," *Econ. Rev.*, vol. 18, pp. 127–140, 1999.

[12] H. K. Ryu, "Maximum entropy estimation of density and regression functions," *J. Econ.*, vol. 56, pp. 397–440, 1993.

[13] D. W. Scott, *Multivariate Density Estimation*. New York: Wiley, 1992.

[14] B. Silverman, *Density Estimation for Statistics and Data Analysis*. New York: Chapman & Hall, 1986.

[15] J. R. Thompson and R. A. Tapia, *Nonparametric Function Estimation, Modeling, and Simulation*. Philadelphia, PA: SIAM, 1990.

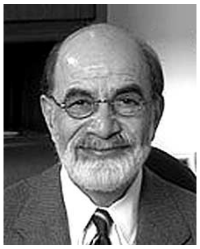
[16] M. Tribus, "Thirty years of information theory," in *The Maximum Entropy Formalism*. Cambridge, MA: MIT Press, 1979.

Yanni Kouskoulas received the B.S., M.S. and Ph.D. degrees from the University of Michigan, Ann Arbor, in 1995, 1996, and 2001, respectively.

He is currently with the Applied Physics Laboratory, Johns Hopkins University, Laurel, MD. His research interests include remote sensing, communications systems, and applied statistics.

Leland E. Pierce (S'85–M'89–SM'01) received the B.S. degrees in both electrical and aerospace engineering in 1983, and the M.S. and Ph.D. degrees in electrical engineering in 1986 and 1991, respectively, all from the University of Michigan, Ann Arbor.

Since 1991, he has been the Head of the Microwave Image Processing Facility within the Radiation Laboratory, Electrical Engineering and Computer Science Department, The University of Michigan, where he is responsible for research into the uses of Polarimetric SAR systems for remote sensing applications, specifically forest canopy parameter inversion.



Fawwaz T. Ulaby (M'68–SM'74–F'80) received the B.S. degree in physics from the American University of Beirut, Lebanon, in 1964, and the M.S.E.E. and Ph.D. degrees in electrical engineering from the University of Texas, Austin, in 1966 and 1968, respectively.

He is the Vice President for Research and Williams Distinguished Professor of electrical engineering and computer science at the University of Michigan, Ann Arbor. His current reserach interests include microwave and millimeter-wave remote sensing, radar systems, and radio wave propagation. He has authored ten books and published more than 500 papers and reports on these subjects.

Dr. Ulaby is the recipient of numerous awards, including the Eta Kappa Nu Association C. Holmes MacDonald Award as "An Outstanding Electrical Engineering Professor in the United States of America for 1975," the IEEE Geoscience and Remote Sensing Distinguished Achievement Award in 1983, the IEEE Centennial Medal in 1984, The American Society of Photogrammetry's Presidential Citation for Meritorious Service in 1984, the NASA Group Achievement Award in 1990, and the 2000 IEEE Electromagnetics Award. He was President of the IEEE Geoscience and Remote Sensing Society from 1980 to 1982, Executive Editor of IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING from 1983 to 1985, and as General Chairman of several international symposia. In 1995, he was elected to membership in the National Academy of Engineering and currently serves as Editor-in-Chief of the PROCEEDINGS OF THE IEEE.