C

Density Estimation

DENSITY estimation is a common problem that occurs in many different fields. We may, for instance, want to determine the likelihood of heart attack for a particular age group given a large collection of medical reports. We may also be interested in the geographical distribution of a particular group of people, using census reports or a telephone survey. Another application could be in predicting the outcome of a natural phenomenon based on observations of its past behavior, such as the eruption interval of a geyser. In the context of computer graphics, we may also wish to determine the intensity of light in a medium based on the distribution of photons in the scene. All of these problems are concerned with the same basic problem of density estimation.

Density estimation is a research area in statistics and has been studied extensively in this field. Density estimation has also been used in the field of computer graphics [Zareski et al., 1995; Shirley et al., 1995; Walter et al., 1997; Walter, 1998]. The volumetric photon mapping technique developed in Chapter 8, as well as the original photon mapping method upon which it is based [Jensen, 1996, 2001; Jensen and Christensen, 1998], can be thought of as a density estimation process and relies on tools developed in statistical modeling.

In this chapter we provide a brief overview of the theory of density estimation as well as review some of the available approaches. More in-depth introduction to these concepts can be found in the classical text by Silverman [1986] as well as Scott [1992]. Other references in the computer graphics literature include Dutré et al. [2006] and Walter [1998].

C.1 Introduction

In density estimation we are interested in determining an unknown function f, given only random samples or observations distributed according to this function. More formally, the goal of density estimation is to infer the probability density function, or PDF, from observations of a random variable. We have already discussed the use of PDFs and random variables in Appendix A with the goal of numerically integrating functions. Density estimation is concerned with a related, but inverse, problem: given a set of random samples, determine what PDF was used to create them.

Density estimation approaches can be broadly classified into two groups: parametric density estimation and non-parametric density estimation.

Parametric Methods. Parametric methods make strict a priori assumptions about the form of the underlying density function. For instance, a parametric approach may assume the random variables have a Gaussian distribution or the PDF is a polynomial of a particular degree. Such assumptions significantly simplify the problem, since only the parameters of the chosen family of functions need to be determined. In the case of a normal distribution, the density estimation process reduces to determining the mean μ and standard deviation σ of the sample points.

Non-Parametric Methods. Oftentimes it is not possible to make such strict assumptions about the form of the underlying density function. Non-parametric approaches are more appropriate in these situations. These techniques make few assumptions about the density function and allow the data to drive the estimation process more directly. In the context of computer graphics, non-parametric approaches are typically the most appropriate, and we will focus on these in the remaining sections.

C.2 Histograms

The simplest form of density estimation is the histogram method. This approach subdivides the domain into bins and counts the number of samples n_b which fall into each bin. The

local probability density is obtained by dividing the number of samples in each bin by the total number of samples N and the bin width h. This can be expressed as

$$\hat{f}(x) = \frac{n_b}{Nh} \quad \text{for } x_b \le x < x_{b+1}, \tag{C.1}$$

where x_b and x_{b+1} are the extents of bin b, and $h = x_{b+1} - x_b$. We use $\hat{f}$ to denote a density estimate of the probability density function f.

The histogram method has a number of advantages. It is easy to implement and provides results which are straightforward to visualize and intuitive to interpret. Also, it is easy to show that $\hat{f}$ is a valid PDF since it is always non-negative and integrates to one over the entire domain. However, histograms have many problems, which motivated the developed of more advanced methods.

One issue with histograms is that the resulting density function is not smooth. In fact, it has zero derivatives everywhere, except at the bin transitions, where its derivative is infinite. This issue can be catastrophic in applications, such as clustering, which rely on following the derivative to find local maxima. In computer graphics, this issue is not as extreme, but the derivative discontinuities can lead to objectionable artifacts, which can be avoided with more advanced techniques.

Another issue with histograms is that the choice of bin transition locations, even when keeping h fixed, can significantly affect the resulting PDF. This extra degree of freedom is completely independent of the underlying data and is simply a side effect of the estimation method itself.

Several approaches have been developed to address some of these concerns. Orthogonal series estimation, discussed in the next section, directly addresses the issue of discontinuity. The naïve estimator and all its generalizations come about by attempting to remove the choice of absolute bin positions.

C.3 Orthogonal Series Estimation

Histograms yield a single average value within each bin, which leads to discontinuities. In order to construct smoother approximations of the underlying PDF, it is possible to directly estimate a higher-order function within each bin. By choosing the functions at neighboring bins to match at the transitions, we can further construct approximations of the PDF which are differentiable everywhere.

To compute higher-order approximations, we decompose the PDF within each bin b as a weighted sum of *basis functions* $\Psi_{b,j}(x)$:

$$\hat{f}(x) = \sum_j f_{b,j}\,\Psi_{b,j}(x). \tag{C.2}$$

The $f_{b,j}$ terms are coefficients which scale the contribution of each basis function j within each bin b. These are defined as the inner product of the density function f and the *dual* basis functions $\tilde{\Psi}_{b,j}$:

$$f_{b,j} = \int f(x)\,\tilde{\Psi}_{b,j}\,dx. \tag{C.3}$$

The dual basis function $\tilde{\Psi}_{b,j}$ are obtained using an orthogonality constraint. Specifically, for any fixed bin b the inner product of $\tilde{\Psi}_{b,j}$ with any of the basis functions $\Psi_{b,j}$ should be:

$$\int \tilde{\Psi}_{b,j}(x)\,\Psi_{b,k}(x)\,dx = \delta_{j,k}. \tag{C.4}$$

The coefficients can be estimated from the random samples using a Monte Carlo estimate of Equation C.3:

$$f_{b,j} \approx \frac{1}{N}\sum_{i=0}^{N-1}\frac{f(x_i)\,\tilde{\Psi}_{b,j}(x_s)}{pdf(x_i)} \tag{C.5}$$

$$\approx \frac{1}{N}\sum_{i=0}^{N-1}\tilde{\Psi}_{b,j}(x_i). \tag{C.6}$$

Relation to Histograms. In the case of a constant basis function within each bin, the orthogonal series estimator reverts to the histogram method. In this case the single dual basis function for bin b is

$$\tilde{\Psi}_b(x) = \begin{cases} \frac{1}{h} & \text{if } x_b \le x < x_{b+1}, \\ 0 & \text{otherwise.} \end{cases} \tag{C.7}$$

It is easy to see that this basis function results in the histogram estimator in Equation C.1.

C.4 Naïve Estimator

Another generalization of the histogram method, which Silverman [1986] calls the *naïve estimator*, addresses the choice of bin locations. The main idea behind this method is to use the estimation point to adaptively determine the bin locations, thereby eliminating it as an extra parameter. This estimator can be written as:

$$\hat{f}(x) = \frac{n_x}{N2h}, \tag{C.8}$$

where n_x is the number of sample points which fall within the interval $[x-h, x+h)$. Comparing the above estimator to Equation C.1 we see that it is equivalent to a histogram where the estimation point x is used as the center of the bin, and the bin width is $2h$. Therefore, the naïve estimator is always globally a valid PDF, i.e., it is non-negative and integrates to one.

It can be informative to rewrite Equation C.8 by introducting the weighting function w:

$$w(t) = \begin{cases} \frac{1}{2} & \text{if } |t| < 1, \\ 0 & \text{otherwise.} \end{cases} \tag{C.9}$$

Using this notation, we can express the naïve estimator as

$$\hat{f}(x) = \frac{1}{Nh} \sum_{i=0}^{N-1} w\left(\frac{x - x_i}{h}\right), \tag{C.10}$$

where x_i are the data samples. In this form, it is easy to see that the naïve estimator places a "box" of width $2h$ and height $(2hN)^{-1}$ at each data point and sums up the contributions. This

interpretation is useful in deriving the kernel estimator, which we discuss in the next section.

C.5 Kernel Estimator

Though the naïve estimator eliminates the problem of choosing the bin locations, it does not address some of the other limitations of the histogram method. The kernel method generalizes the naïve estimator to eliminate the discontinuous nature of the resulting PDF.

By examining Equation C.10 we observe that the reason for discontinuities is due to our particular choice of weighting function w, which has zero derivatives and discontinuities at $|x| = 1$. In fact, it is easy to show that the naïve estimator inherits *all* the differential properties of the weighting function since it is a simple sum. We can therefore improve the smoothness of the estimator by improving the smoothness of the weighting function. We accomplish this by replacing w with a smooth *kernel* function K. Our only restriction is that K must integrate to one:

$$\int_{-\infty}^{\infty} K(t)\, dt = 1. \tag{C.11}$$

With this restriction in place, we can define the kernel estimator as

$$\hat{f}(x) = \frac{1}{Nh} \sum_{i=0}^{N-1} K\Big(\frac{x - x_i}{h}\Big), \tag{C.12}$$

where h controls the amount of smoothing and is called the *window width* or *bandwidth* of the kernel. To make the notation more concise, it is often convenient to set $K_h(t) = (h^{-1})K(h^{-1}t)$ and express the kernel estimator simply as

$$\hat{f}(x) = \frac{1}{N} \sum_{i=0}^{N-1} K_h(x - x_i). \tag{C.13}$$

Typically we will choose K to be a smooth, symmetric function, which has a strong influence at $x = x_i$ and decreased influence as the distance $x - x_i$ increases. Some common examples include the normalized Gaussian kernel, the bi-weight kernel, and the Epanechnikov kernel [Silverman, 1986]. Analogous to our interpretation of the naïve estimator as a sum of

"boxes" at the data point, the kernel estimator is a sum of smooth "bumps" at the data points. Also note that if we use a constant kernel, then the kernel method reverts to the naïve estimator.

The kernel method can also be interpreted as a blurring process, or a convolution operation on the data points. If we consider the data points as Dirac delta impulses, δ, then the kernel method can be written as

$$\hat{f}(x) = K_h(x) \star \left(\frac{1}{N} \sum_{i=0}^{N-1} \delta(x - x_i) \right) \tag{C.14}$$

$$= \int_{-\infty}^{\infty} K_h(x) \left(\frac{1}{N} \sum_{i=0}^{N-1} \delta(x - x_i) \right) dx, \tag{C.15}$$

$$= \frac{1}{N} \sum_{i=0}^{N-1} K_h(x - x_i). \tag{C.16}$$

C.6 Locally Adaptive Estimators

In developing all the estimators so far, we have ignored one important consideration. The window width, or, in the case of the histogram the bin width, plays an important role in the behavior of the resulting density function.

At a high level, the bandwidth determines the amount of smoothing that is performed on the sample data. It is important to set this parameter properly. Choosing a value that is too small will result in under-smoothing and erratic fluctuations in the estimator which are not present in the original PDF. On the other hand, using a very large value may over-smooth the data, potentially eliminating important features of the underlying PDF. In practice it is very hard to choose an optimal bandwidth without knowing something about the density function itself.

More formally, this tradeoff can be expressed as trying to minimize the sum of the global bias and variance of the estimator. As we increase the bandwidth, we are smoothing the true PDF, which reduces variance but introduces bias. In contrast, if we decrease the bandwidth, we decrease the bias but increase variance. When we need to choose a bandwidth parameter for the whole domain of the data, a common technique is to try to minimize the *mean integrated squared*

error:

$$\text{MISE}(\hat{f}) = E\int \left[\hat{f}(x) - f(x)\right]^2 \, dx, \tag{C.17}$$

$$= \int V\left[\hat{f}(x)\right] + \beta\left[\hat{f}(x)\right]^2 \, dx, \tag{C.18}$$

which is the sum of the integrated variance and the integrated squared bias. Computing the MISE exactly requires information about the unknown true density f. However, since the density estimator $\hat{f}$ is a function of a random variable, it is itself a random variable. We can therefore apply the tools from Appendix A to estimate the variance and the bias of $\hat{f}$.

Unfortunately, a single bandwidth may not be optimal for all regions of the domain. For instance, a single bandwidth value may over-smooth features in high density regions and, at the same time, under-smooth regions in the "tails" of the distribution where few samples are present. Locally adaptive methods attempt to address this issue by allowing the bandwidth to change across the domain of the PDF. We will consider two different approaches for adaptively modify the kernel bandwidth [Jones, 1990]. The first class of techniques, called *balloon estimators*, vary the bandwidth based on the evaluation point x. The second set of techniques, called *sample-point estimators*, vary the bandwidth based on the data points x_i.

C.6.1 Balloon Estimator

The general form of the balloon estimator is given by

$$\hat{f}(x) = \frac{1}{Nh(x)} \sum_{i=0}^{N-1} K\left(\frac{x - x_i}{h(x)}\right), \tag{C.19}$$

$$= \frac{1}{N} \sum_{i=0}^{N-1} K_{h_x}(x - x_i), \tag{C.20}$$

where $h(x)$ is the bandwidth as a function of x, the evaluation location.

The balloon estimator was introduced by Loftsgaarden and Quesenberry [1965] in the form of the k^{th} nearest neighbor estimator, which can be written in the form of Equation C.19 by using a constant kernel and setting $h(x) = d_k(x)$ where $d_k(x)$ returns the distance to the k^{th} nearest data point to x. In fact, Silverman [1986] refers to (a slightly more restricted version of)

Equation C.19 as the *generalized k^{th} nearest neighbor estimator*.

The balloon estimator unfortunately suffers from a number of inefficiencies, especially in the univariate case. Firstly, the PDF derived using a balloon estimator will not, in general, integrate to one over the entire domain. If a *global* estimate of the PDF is needed, then this can be problematic. In computer graphics, however, this is not generally a major problem since we typically only care about the *pointwise* behavior of the estimated density. Another problem with the nearest neighbor estimator is that the bandwidth is a discontinuous function, and these discontinuities manifest themselves directly in the resulting PDF. This is true even if the chosen kernel is itself smooth. In computer graphics this can lead to mach banding and other visual artifacts.

C.6.2 Sample-point Estimator

The second type of local bandwidth estimator is called the *sample-point estimator*. The general form of the sample-point estimator is given by

$$\hat{f}(x) = \frac{1}{N}\sum_{i=0}^{N-1}\frac{1}{h(x_i)}K\left(\frac{x-x_i}{h(x_i)}\right), \tag{C.21}$$

$$= \frac{1}{N}\sum_{i=0}^{N-1}K_{h_{x_i}}(x-x_i). \tag{C.22}$$

Note that the only difference to the balloon estimator in Equation C.19 is that the bandwidth $h(x_i)$ is a function of the sample points x_i and not the evaluation point x. The sample-point estimator was first introduced by Breiman et al. [1977] under the name of the *variable kernel method*.

Just as in the regular kernel estimator, the sample-point estimator places a kernel at each data point, but these kernels are allowed to vary in size from one data point to another. Typically, the bandwidth of a data point is chosen based on the local density of samples in the vicinity. This requires performing a *pilot estimate* to compute the local density at each data point and assign the bandwidths.

Sample-point estimators do have a number of benefits over balloon estimators. Firstly, since each kernel is normalized, the estimator itself is a valid PDF, which integrates to one. Furthermore, unlike the balloon estimator, the sample-point estimator inherits all the differential

properties of the kernel functions and can therefore be made fully continuious. Another practical advantage of sample-point estimators is that the extent of each sample point is known *before* density evaluation begins. This often allows for simpler and more efficient data structures to be used during density evaluation. The beam radiance estimate developed in Chapter 8 exploits exactly this property of sample-point estimators.