Christian C. Beardah
Mike J. Baxter

MATLAB Routines for Kernel Density Estimation and the Graphical Representation of Archaeological Data

1 Introduction

Histograms are widely used for data presentation in archaeology, but have many potential limitations. They are appropriate for variables where the measurement scale is continuous (e.g., length, height). The scale is divided into a set of contiguous intervals; the frequency count of observations in each interval is obtained; and the count is represented graphically by a bar whose *area* is proportional to the frequency. Although not essential, it is usual for intervals to be defined to be of equal width, in which case the height of a bar is also proportional to the frequency. We shall refer to this common interval width as the *bin-width*.

The choice of bin-width is essentially an arbitrary one. A second arbitrary choice is the starting position of the first interval to contain any data, and we refer to this position as the *origin*. It is well known (e.g., Whallon: 1987) that the appearance of a histogram can depend on both the choice of origin and bin-width. In particular, the archaeological interpretation of a histogram depends on the appearance which can be markedly affected by these two arbitrary choices.

A common use of histograms in archaeology is for comparative purposes; for example, comparing the distribution of the ratio of length to breadth of flint flakes from different contexts. Arguably, histograms are usually inefficient for this kind of purpose, and better methods such as the use of box-and-whisker plots exist (Cleveland 1993). Generalisation of the histogram to display the joint distribution of two variables is sometimes desirable, but is unwieldy and requires lots of data.

Kernel Density Estimates (KDEs), which at their simplest can be thought of as smoothed histograms, avoid many of these problems. They have been little used in archaeology, notwithstanding Orton's (1988) implicit reference to their potential. One reason is undoubtedly that the methodology has not been readily available in the packages used by archaeologists. A possible second reason is that archaeologists may find the mathematics underlying the methodology forbidding.

In this paper, after describing briefly the methodology, routines for implementing KDEs in the MATLAB package, that have been developed by the first author, are described. We illustrate the utility of these routines using several archaeological examples.

2 The Mathematics of KDEs

2.1 Univariate KDEs

Unless otherwise stated the sources for the material in this and the next section are either Wand and Jones (1995) or Silverman (1986).

Given n points $X_1, X_2, \ldots, X_n$ a KDE can be thought of as being obtained by placing a 'bump' at each point and then summing the height of each bump at each point on the X-axis. The shape of the bump is defined by a mathematical function — the kernel, $K(x)$ — that integrates to 1. The spread of the bump is determined by a window- or bandwidth, h, that is analogous to the bin-width of a histogram. $K(x)$ is usually a symmetric probability density function (pdf).

Mathematically, this gives the KDE as

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} K(\frac{x-X_i}{h})$$

Compared to the histogram the shape of $\hat{f}(x)$ does not depend upon the choice of origin, but is affected by the bandwidth h. Large values of h over-smooth, while small values under-smooth the data. Choice of both h and $K(x)$ is discussed later. Generalisations to higher dimensions, d, are relatively direct. For descriptive use only the case d=2 is likely to be of widespread interest, and is considered in the next section.

2.2 Multivariate KDEs

The representation of the KDE as a sum of 'bumps' is easily extended to the higher dimensional case. We shall restrict our attention to the case of bivariate data points of the form (X_i, Y_i). The kernel now becomes a function of two variables, $K(x,y)$, which again integrates to 1 and is usually radially symmetric. (For example, the bivariate normal pdf.) The mathematical representation of the KDE, $\hat{f}(x,y)$, depends, in general, on a 2 by 2 symmetric positive definite matrix, **H**. In this paper we shall only consider the case where **H** is diagonal, i.e.

$$H = \begin{bmatrix} h_1^2 & 0 \\ 0 & h_2^2 \end{bmatrix}$$

With this simplification the representation of the bivariate KDE, $\hat{f}(x,y)$, is given by

$$\hat{f}(x,y) = \frac{1}{nh_1h_2} \sum_{i=1}^{n} K\left(\frac{x\text{-}X_i}{h_1}, \frac{y\text{-}Y_i}{h_2}\right)$$

where h_1 and h_2 are the window-widths in the X and Y directions.

The smoothing parameters h_1 and h_2 control the amount of smoothing in the two co-ordinate directions. If $h_1 = h_2$ then we can think of the 'bumps' of the kernel function as being spherically symmetric (with circular contours). On the other hand, if $h_1 \neq h_2$ then the 'bumps' have elliptical contours with the ellipsoidal axes parallel to the two co-ordinate axes. A further generalisation (not considered here) introduces an off-diagonal value h_3 to the symmetric matrix **H** and allows the ellipsoidal axes to have arbitrary orientation. Whilst taking $h_1 = h_2$ clearly makes understanding and implementation rather more straightforward, the fact that this involves the same amount of smoothing in each co-ordinate direction is regarded as a serious shortcoming (Wand/Jones 1995: 105). In the routines described below the user has the option to interactively vary the smoothing parameters, using one, two or three values of h as discussed above. The default number of smoothing parameters is two.

3 MATLAB Implementation

Here we describe, by way of examples, routines for performing exploratory data analysis using KDEs. These routines have been implemented in MATLAB, a scientific computing environment which has developed a strong user base in Further and Higher Education institutions, particularly in Departments of Mathematics and Engineering. Many such departments have copies of the package available for general use. MATLAB is particularly useful in applications involving numerical matrices and graphical presentation. Multivariate data is most naturally represented as a matrix of values, where columns indicate different components. This matrix representation of data, when coupled with MATLAB's matrix manipulation and programming capabilities, provides a powerful, accessible platform for mathematical and statistical programming and algorithm development.

Powerful graphics facilities are available within the standard package and the Graphical User Interface (GUI) is programmable also. This feature means that software can be designed to be user-friendly, with an assumption of little knowledge on the user's part. Windows, menus, sliders, buttons etc. can be used to create an interface familiar to anyone who has worked within a Windows environment, and quickly learned by those who have not. We have taken advantage of these features to develop a suite of routines allowing the user to interactively vary the kernel function, the smoothing parameter(s) and various aspects of the graphical depiction of the resulting KDEs, including contouring in the bivariate case. The use of mathematical packages such as MATLAB to create such Windows based software is a new and hitherto underexploited option for users with specific applications in mind. While a significant amount of effort must be invested in the production of such routines, we believe that the portable and re-usable nature of the software justifies this effort.

All of the figures in the remainder of section 3 were generated either using these routines exclusively, or in combination with basic MATLAB commands for plotting multiple images (fig. 1).

3.1 Example: The univariate case

In practice the choice of kernel function makes little difference to the appearance of the KDE. Figure 1 shows four KDEs generated using the same value of h, yet with different kernel functions. The names of the kernels are given in the graphs and their mathematical definitions in Silverman (1986). These data represent the rim diameters of 60 Bronze Age cups from Italy (Source: Baxter 1994: 233-234), based upon Lukesh and Howe (1978)).

Each of the kernels used in figure 1 has *bounded support*, meaning that the kernel function is non-zero only for the range $x \in [-1,1]$. In practice this makes the 'bumps' that form the KDE spread out rather less than the more commonly used normal kernel, which has *infinite support*. Compare the KDEs of figure 1 with those of figure 2(c), also obtained using $h = 2.5$ but using the normal kernel. It is clear that the KDE obtained using the normal kernel oversmooths relative to those KDEs produced with finite support kernel functions (for the same value of h).

In contrast to variation of the kernel function, the degree of smoothing (controlled by h) is of crucial importance in density estimation. If h is too large we 'oversmooth', erasing detail. If h is too small we 'undersmooth', and fail to filter out spurious detail. Several methods of automatically choosing an *optimal* (in some sense) value of h exist, though a subjective choice is often equally valid.

Some simplistic methods of automatically choosing h depend upon an assumption of normality in the data. If this assumption is not valid 'oversmoothing' often results. This explains the oversmoothing apparent in figure 2(c), which was obtained using the rim diameter data; a near 'optimal' value of h and the normal kernel. It is clear from figure 1 that this data is far from normal in structure. For the reasons outlined above, it is important to have the facility to

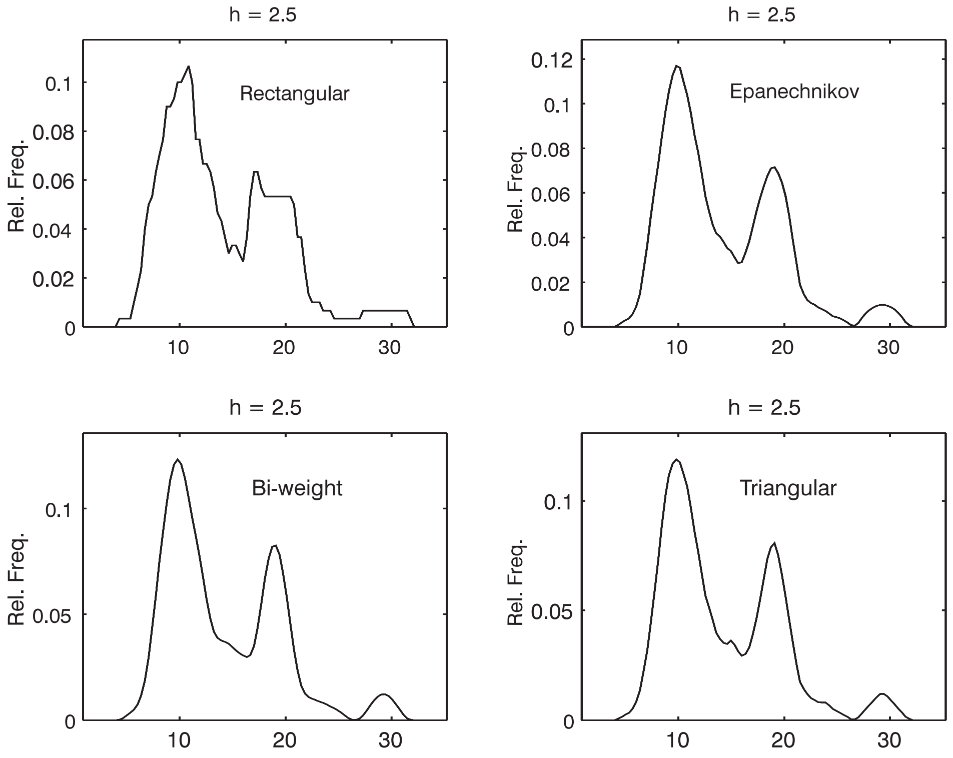

Figure 1. The effect of varying the kernel function.

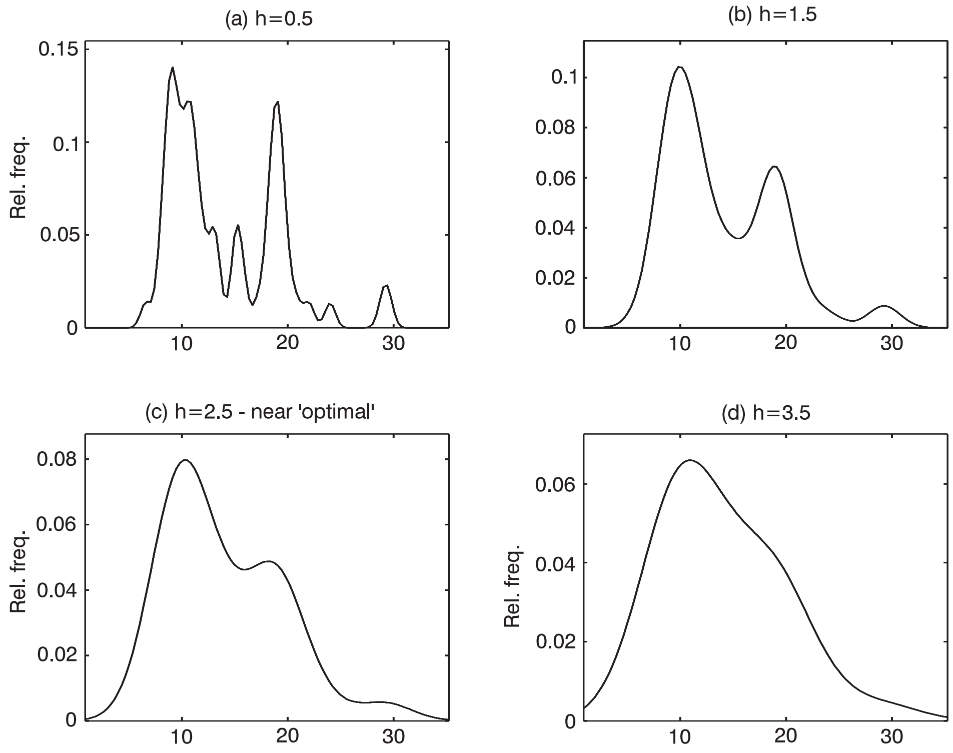

Figure 2. The effect of varying h.

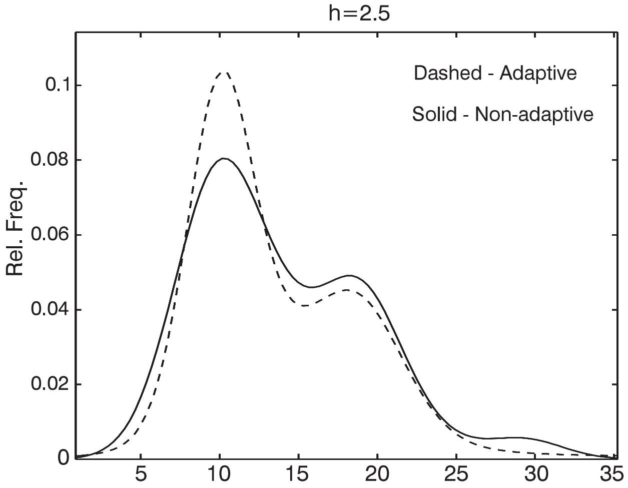

Figure 3. Adaptive Kernel Density Estimation.

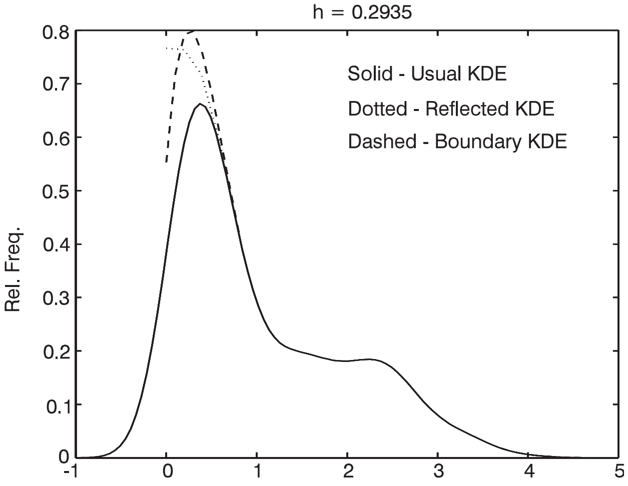

Figure 4. Bounded and Unbounded KDEs.

interactively vary the smoothing parameter *h*. Where an 'optimal' value is automatically used by a routine, it is sensible to reduce this value and recompute the KDE. The boundary between over- and undersmoothing is quite large in our experience, and a visual inspection of KDEs obtained using various values of *h* should quickly lead to a satisfactory value of *h* being found. In this respect it is helpful to be able to overlay several KDEs on the same axes, or to use subplots as in figures 1 and 2. Each of these methods is supported.

3.2 Adaptive methods

The basic idea of *adaptive* methods is identical to that described above, i.e. we construct a KDE by placing kernel functions at the observed data points. The difference is that here we allow the smoothing parameter, *h*, to vary from one data point to the next. More specifically, we use a larger value of *h* for observations in regions of low density, in particular for observations in the tails of the distribution. The intention is to reduce the effect of outliers on the KDE.

This procedure requires that we can first identify data points which lie in regions of low density. This can be achieved by initially computing a *pilot estimate* of the KDE by the standard methods of section 2. An adaptive KDE can then be constructed based upon this information. See Silverman (1986: 100-110) for a detailed discussion. Figure 3 shows both adaptive and non-adaptive KDEs for the cup diameter data and the normal kernel.

3.3 Bounded data

If the data represents some measured quantity, for example the rim diameter data considered above, then it makes little sense to use a density estimate which is positive for negative values of *x*. However, if the data set includes data points near zero, it is inevitable that the kernel or 'bump' associated with such data points will stray into the region where *x* is negative. This is especially true of the normal kernel function, since it has infinite support.

A natural, simplistic way of dealing with this situation is to reflect the part of the KDE to the left of zero in the line $x = 0$. Figure 4 shows three KDEs. The solid curve was produced using the normal kernel and an 'optimal' value of *h* as described in section 2 above. This density estimate has the undesirable property that it overlaps the line $x = 0$. In contrast, the KDE represented by the dotted curve in figure 4 was produced by reflecting the appropriate portion of the solid curve in the line $x = 0$. The data in this case represents the Na_2O content of a sample of 361 fragments of French medieval glass. Clearly this quantity cannot be negative.

More advanced methods of dealing with so-called 'bounded' data exist. In particular, there are classes of 'boundary' kernel functions which take into account the proximity of the boundary and ensure that it is not crossed. These boundary kernels have the unusual property that $K(x)$ may be negative for some *x*. In addition to the simple reflection method, we have implemented a boundary kernel method as described in Jones (1993). In figure 4 the broken line represents such a KDE.

3.4 The bivariate case

Just as in the univariate case the choice of kernel function makes little difference to the appearance of the final KDE, though for completeness we have provided a

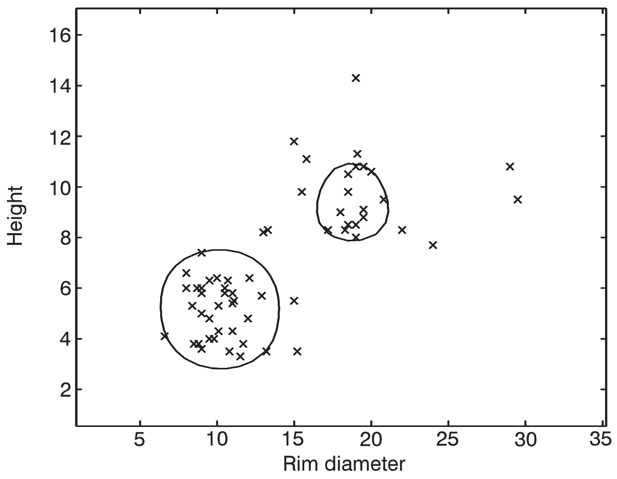

Figure 5. A 75% contour for the Bronze Age cup data.

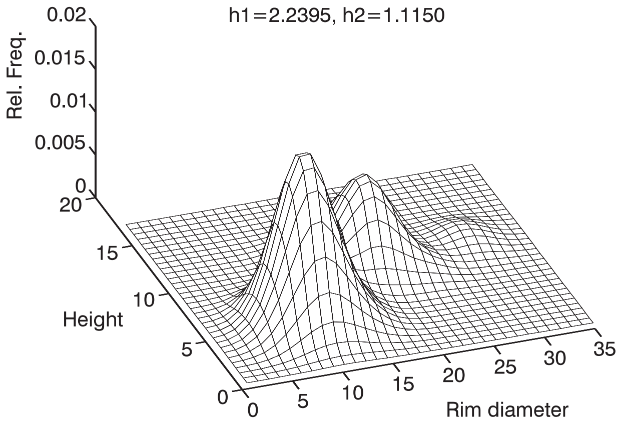

Figure 6. The KDE used to generate the contour of figure 5.

choice of four commonly used bivariate kernel functions. On the other hand, the choice of smoothing parameters again *does* have a significant effect. Our routines automatically choose values for h_1 and h_2 based upon the univariate method of selection for each of the two components considered separately. However, interactive subjective choices by the user are also supported.

An important use of bivariate KDEs is in contouring. Since a KDE is a *function*, we can apply standard contouring methods based upon the height of the function. In addition, we have found some useful applications of a new contouring method reported in Bowman and Foster (1993) (see Baxter/Beardah 1995; Baxter *et al*. 1994 for more details). This method consists of forming a KDE, then ranking the data points by descending density as estimated by the KDE. The contour enclosing *p*% of the data is then formed by drawing a contour line corresponding to the value of the kernel estimate of a data point *p*% through the ordered list. Since this technique involves calculation of the KDE at each of the data points it can be computationally expensive for large data sets. Figure 5 shows a 75% contour overlaid upon a scatter plot of data representing the rim diameter and overall height of 60 Bronze Age cups from Italy. The contour encloses the 'most dense' 75% of the data set. Figure 6 shows the bivariate KDE (obtained using the bivariate normal kernel and 'optimal' values for h_1 and h_2) which was used to generate the contour shown in figure 5.

Our routines allow interactive variation of the smoothing parameters h_1 and h_2 as well as the type of kernel function. In addition, the resulting KDEs and percentage contour plots can be viewed from any angle by means of sliders.

Acknowledgements

We conclude by acknowledging the contribution of Miss K. Bibby to this work and by inviting interested parties to obtain the software by contacting the first named author.

references

Baxter, M.J. 1994 *Exploratory Multivariate Analysis in Archaeology*. Edinburgh: Edinburgh University Press.

Baxter, M.J.
C.C. Beardah
I.C. Freestone
1994 A Statistical Re-Analysis of Some French Medieval Glass Compositional Data With an Application. *Department of Mathematics Research report* 7/94. Nottingham Trent University.

Baxter, M.J.
C.C. Beardah
1995 Graphical Presentation of Results from Principal Components Analysis. In: J. Huggett/ N. Ryan (eds), *Computer Applications and Quantitive Methods in Archaeology 1994*, 63-67, BAR International Series 600, Oxford: Tempus Reparatum.

Bowman, A.
P. Foster
1993 Density Based Exploration of Bivariate Data, *Statistics and Computing* 3, 171-177.

Cleveland, W.S. 1993 *Visualising Data*. New Jersey: Hobart Press.

Jones, M.C. 1993 Simple Boundary Correction for Kernel Density Estimation, *Statistics and Computing* 3, 135-146.

Lukesh, S.S.
S. Howe
1978 Protoapennine vs. Subapennine: Mathematical Distinction Between Two Ceramic Phases, *Journal of Field Archaeology* 5, 339-347.

Orton, C.R. 1988 Review of Quantitative Research in Archaeology, M.S. Aldenderfer (ed.), *Antiquity* 62, 597-598.

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

Silverman, B. 1986 *Density Estimation for Statistics and Data Analysis*. London: Chapman and Hall.

Wand, M.P.
M.C. Jones
1995 *Kernel Smoothing*. London: Chapman and Hall.

Whallon, R. 1987 Simple Statistics. In: M.S. Aldenderfer (ed.), *Quantitative Research in Archaeology: Progress and Prospects*, 135-150, London: Sage.

Christian C. Beardah and Mike J. Baxter
Department of Mathematics, Statistics and Operational Research
The Nottingham Trent University
Nottingham NG11 8NS
United Kingdom
e-mail: ccb@maths.ntu.ac.uk
mat3beardcc@newvax.ntu.ac.uk