Kernel density estimators

One Dimension

From the definition of a probability density, if the random variable X has a density f

$$f(x) = \lim_{h \to 0} \frac{1}{2h} \mathrm{P}(x - h < X < x + h).$$

For any given h, a naive estimator of $\mathrm{P}(x - h < X < x + h)$ is the proportion of the observations $x_1, \dots, x_n$ falling in the interval $(x - h, x + h)$,

$$\hat{f}(x) = \frac{1}{2nh} \sum_{i=1}^{n} I(x_i \in (x - h, x + h));$$

i.e., the number of $x_1, \dots, x_n$ falling in the interval $(x - h, x + h)$ divided by $2nh$. If we introduce a weight function W given by

$$W(x) = \begin{cases} \frac{1}{2} & |x| < 1 \\ 0 & o.w. \end{cases},$$

Then the naive estimator can be written as

$$\hat{f}(x) = \frac{1}{2n} \sum_{i=1}^{n} \frac{1}{h} W\left(\frac{x - x_i}{h}\right).$$

Unfortunately, this estimator is not a continues function and is not particularly satisfactory for practical density estimation. It does, however, lead naturally to the kernel estimator defined by

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) = \frac{1}{nh} \sum_{i=1}^{n} K_h(x - x_i),$$

where K is known as the kernel function and h is the bandwidth or smoothing parameter.
In Statistics, a kernel is a non-negative real-valued integrable function satisfying
$\int_{-\infty}^{\infty} K(x)dx = 1.$
Usually, but not always, the kernel function will be a symmetric density function;
for example, the normal.
In order to compute the MSE of the estimate of $f(.)$, we need the bias and variance of $\hat{f}(.)$. Let X be a random variable having density $f(.)$. Then we have for the specific point $x \in \mathbb{R}$ we have

$$E\big(\hat{f}(x)\big) = E[K_h(x - X)] = \int K_h(x - y) f(y) dy = \int K(z) f(x - hz) dz.$$

Expanding $f(x - hz)$ in a Taylor series about x we obtain

$$f(x - hz) = f(x) - hzf'(x) + \frac{1}{2} h^2 z^2 f''(x) + o(h^2)$$

uniformly in z. This leads to

$$E\big(\hat{f}(x)\big) = f(x) + \frac{1}{2} h^2 f''(x) \int z^2 K(z) dz + o(h^2)$$

where we have used

$$\int K(z)dz = 1, \qquad \int zK(z)dz = 0, \qquad \int z^2 K(z)dz = \sigma_K^2 < \infty.$$

Three commonly used kernel functions are

1. rectangular,
$$K(x) = \begin{cases} \frac{1}{2} & |x| < 1 \\ 0 & o.w. \end{cases} = \frac{1}{2}\, 1_{\{|x|<1\}},$$
2. triangular,
$$K(x) = \begin{cases} 1 - |x| & |x| < 1 \\ 0 & o.w. \end{cases} = (1 - |x|) 1_{\{|x|<1\}},$$
3. Gaussian,
$$K(x) = \frac{1}{\sqrt{2\pi}}\, e^{-\frac{1}{2}x^2}.$$

The three kernel functions are implemented in R as shown in Figure 2.12. For some grid x, the kernel functions are plotted using the R statements in Figure 2.12. The kernel estimator $\hat{f}$ is a sum of "bumps" placed at the observations. The kernel function determines the shape of the bumps, while the window width h determines their width. Figure 2.13 shows the individual bumps $n^{-1}h^{-1}K\left(\frac{x-x_i}{h}\right)$ as well as the estimate $\hat{f}$ obtained by adding them up for an artificial set of data points,

```
rec <- function(x) (abs(x) < 1) * 0.5
tri <- function(x) (abs(x) < 1) * (1 - abs(x))
gauss <- function(x) 1/sqrt(2*pi) * exp(-(x^2)/2)
x <- seq(from = -3, to = 3, by = 0.001)
plot(x, rec(x), type = "l", ylim = c(0,1), lty = 1, ylab = expression(K(x)))
lines(x, tri(x), lty = 2)
lines(x, gauss(x), lty = 3)
legend("topleft", legend = c("Rectangular", "Triangular", "Gaussian"), lty = 1:3,
title = "kernel functions", bty = "n")
```

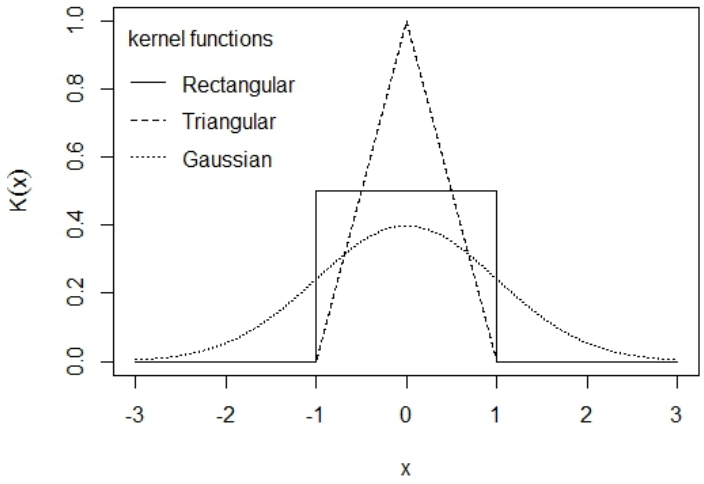

Fig. 2.12. Three commonly used kernel functions.

```
x <- c(0, 1, 1.1, 1.5, 1.9, 2.8, 2.9, 3.5)
n <- length(x)
```

For a grid

```
xgrid <- seq(from = min(x) - 1, to = max(x) + 1, by = 0.01)
```

on the real line, we can compute the contribution of each measurement in x, with $h = 0.4$, by the Gaussian kernel (defined in Figure 2.12, line 3) as follows:

```
h <- 0.4
bumps <- sapply(x, function(a) gauss((xgrid - a)/h)/(n * h))
```

A plot of the individual bumps and their sum, the kernel density estimate $\hat{f}$, is shown in Figure 2.13.

```
plot(xgrid, rowSums(bumps), ylab = expression(hat(f)(x)), type = "l", xlab = "x",
lwd = 2)
rug(x, lwd = 2)
out <- apply(bumps, 2, function(b) lines(xgrid, b))
```

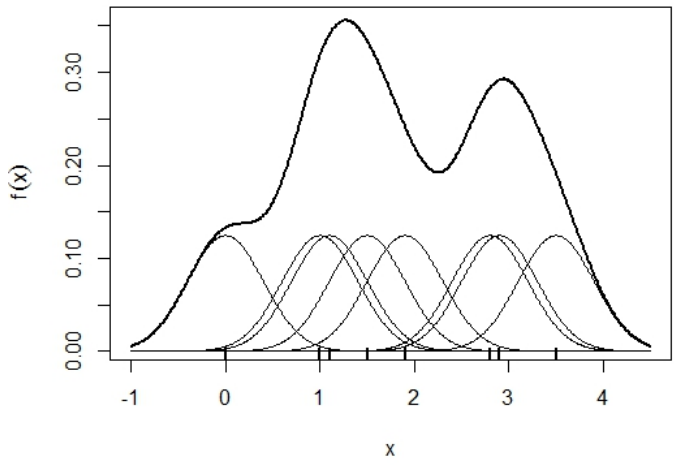

Fig. 2.13. Kernel estimate showing the contributions of Gaussian kernels evaluated for the individual observations with bandwidth h = 0:4.

Some other common kernels:

4. Epanechnikov

$K(x) = \frac{3}{4}(1 - x^2)1_{\{|x|<1\}},$

5. Quartic (biweight)

$K(x) = \frac{15}{16}(1 - x^2)^2 1_{\{|x|<1\}},$

6. Triweight

$K(x) = \frac{35}{32}(1 - x^2)^3 1_{\{|x|<1\}},$

7. Tricube

$K(x) = \frac{70}{81}(1 - |x|^3)^3 1_{\{|x|<1\}},$

8. Cosine

$K(x) = \frac{\pi}{4}\cos\left(\frac{\pi}{2}x\right) 1_{\{|x|<1\}}.$

Two Dimension

The kernel density estimator considered as a sum of "bumps" centered at the observations has a simple extension to two dimensions (and similarly for more than two dimensions). The bivariate estimator for data $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ is defined as

$$\hat{f}(x, y) = \frac{1}{nh_x h_y} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h_x}, \frac{y - y_i}{h_y}\right).$$

In this estimator, each coordinate direction has its own smoothing parameter, h_x or h_y. An alternative is to scale the data equally for both dimensions and use a single smoothing parameter. For bivariate density estimation, a commonly used kernel function is the standard bivariate normal density

$K(x,y) = \frac{1}{2\pi}\, e^{-\frac{1}{2}(x^2+y^2)}$.

Another possibility is the bivariate Epanechnikov kernel given by

$$K(x,y) = \begin{cases} \frac{2}{\pi}\,(1-x^2-y^2) & x^2+y^2<1 \\ 0 & o.w. \end{cases},$$

which is implemented and depicted in Figure 2.14 by using the persp function for plotting in three dimensions.

```
epa <- function(x, y) ((x^2 + y^2) < 1) * 2/pi * (1 - x^2 - y^2)
x <- seq(from = -1.1, to = 1.1, by = 0.05)
 epavals <- sapply(x, function(a) epa(a, x))
persp(x = x, y = x, z = epavals, xlab = "x", ylab = "y", zlab = expression(K(x, y)), theta = -35, axes
= TRUE, box = TRUE)
```

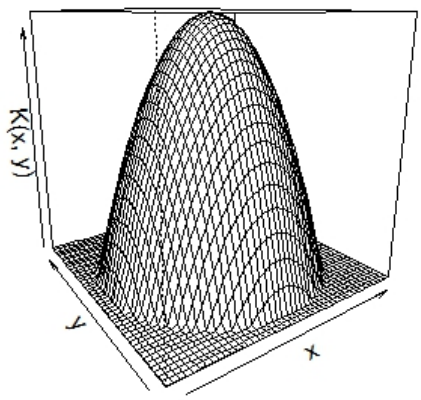

Fig. 2.14. Epanechnikov kernel for a grid between $(-1.1,-1.1)$ and (1.1,1.1).

Our first illustration of enhancing a scatterplot with an estimated bivariate density will involve data from the Hertzsprung-Russell (H-R) diagram of the star cluster CYG OB1, calibrated according to Vanisma and De Greve (1972). The H-R diagram is the basis of the theory of stellar evolution and is essentially a plot of the energy output of stars as measured by the logarithm of their light intensity plotted against the logarithm of their surface temperature. Part of the data is shown in Table 2.1.

Table 2.1: CYGOB1 data. Energy output and surface temperature of star cluster CYG OB1.

Table 2.1: CYGOB1 data (continued).

logst	logli	logst	logli	logst	logli
4.56	5.74	4.42	4.18	3.49	6.29
4.26	4.93	4.23	4.18	4.23	4.34
4.56	5.74	3.49	5.89	4.62	5.62
4.30	5.19	4.29	4.38	4.53	5.10
4.46	5.46	4.29	4.22	4.45	5.22
3.84	4.65	4.42	4.42	4.53	5.18
4.57	5.27	4.49	4.85	4.43	5.57
4.26	5.57	4.38	5.02	4.38	4.62
4.37	5.12	4.42	4.66	4.45	5.06
3.49	5.73	4.29	4.66	4.50	5.34
4.43	5.45	4.38	4.90	4.45	5.34
4.48	5.42	4.22	4.39	4.55	5.54
4.01	4.05	3.48	6.05	4.45	4.98
4.29	4.26	4.38	4.42	4.42	4.50
4.42	4.58	4.56	5.10		

logst	logli	logst	logli	logst	logli
4.37	5.23	4.23	3.94	4.45	5.22

```
logst<-c(4.37, 4.56, 4.26, 4.56, 4.30, 4.46, 3.84, 4.57, 4.26, 4.37, 3.49, 4.43, 4.48, 4.01, 4.29,
4.42, 4.23, 4.42, 4.23, 3.49, 4.29, 4.29, 4.42, 4.49, 4.38, 4.42, 4.29, 4.38, 4.22, 3.48, 4.38,
4.56, 4.45, 3.49, 4.23, 4.62, 4.53, 4.45, 4.53, 4.43, 4.38, 4.45, 4.50, 4.45, 4.55, 4.45, 4.42)
logli<-c(5.23, 5.74, 4.93, 5.74, 5.19, 5.46, 4.65, 5.27, 5.57, 5.12, 5.73, 5.45, 5.42, 4.05, 4.26,
4.58, 3.94, 4.18, 4.18, 5.89, 4.38, 4.22, 4.42, 4.85, 5.02, 4.66, 4.66, 4.90, 4.39, 6.05, 4.42,
5.10, 5.22, 6.29, 4.34, 5.62, 5.10, 5.22, 5.18, 5.57, 4.62, 5.06, 5.34, 5.34, 5.54, 4.98, 4.50)
CYGOB1<-data.frame(logst,logli)
```

A scatterplot of the data enhanced by the contours of the estimated bivariate density, obtained with the function bkde2D()[1] from the package KernSmooth, is shown in Figure 2.15.

```
library("KernSmooth")
CYGOB1d <- bkde2D(CYGOB1, bandwidth = sapply(CYGOB1, dpik))
```

[1] The kernel is the standard bivariate normal density

```
plot(CYGOB1, xlab = "log surface temperature", ylab = "log light intensity")
contour(x = CYGOB1d$x1, y = CYGOB1d$x2, z = CYGOB1d$fhat, add = TRUE)
```

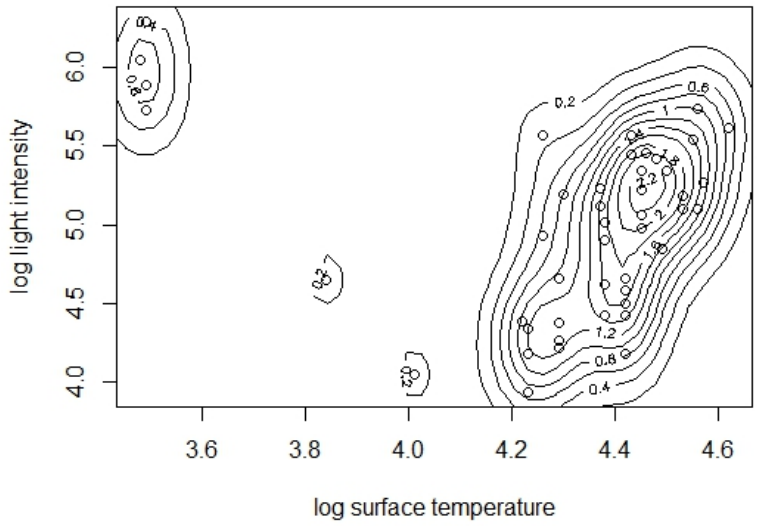

Fig. 2.15. Scatterplot of the log of light intensity and log of surface temperature for the stars in star cluster CYG OB1 showing the estimated bivariate density.

The plot shows the presence of two distinct clusters of stars: the larger cluster consists of stars that have high surface temperatures and a range of light intensities, and the smaller cluster contains stars with low surface temperatures and high light intensities. The bivariate density estimate can also be displayed by means of a perspective plot rather than a contour plot, and this is shown in Figure 2.16.

```
persp(x = CYGOB1d$x1, y = CYGOB1d$x2, z = CYGOB1d$fhat,
 xlab = "log surface temperature",
 ylab = "log light intensity",
 zlab = "density")
```

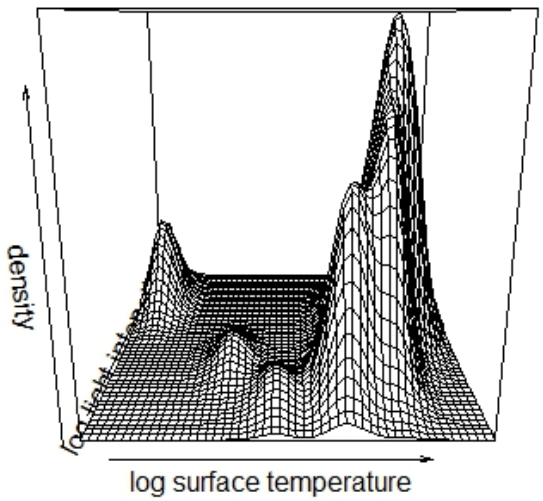

Fig. 2.16. Perspective plot of estimated bivariate density.

This again demonstrates that there are two groups of stars.

Multi Dimension

Kernel density estimation can be easily generalized from univariate to multivariate data, in theory if not always in practice. The general form of the estimator is

$$\hat{f}(\boldsymbol{x}) = \frac{1}{n \det \boldsymbol{H}} \sum_{i=1}^{n} K_q[\boldsymbol{H}^{-1}(\boldsymbol{x} - \boldsymbol{x}_i)],$$

where $\boldsymbol{x} = (x_1, \ldots, x_q)'$, $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{iq})'$, $i = 1, \ldots, n$ are q-vectors; $\boldsymbol{H}$ is the bandwidth (or smoothing) $q \times q$ positive definite matrix and K_q: $\mathbb{R}^q \to \mathbb{R}$ is the kernel function.
A popular technique for generating K_q from a univariate kernel K is by using a product kernel,

$$K_q(\boldsymbol{u}) = \prod_{j=1}^{q} K(u_j).$$