Nonparametric Function Estimation

We now consider two related problems, both of which come under the heading of "nonparametric function estimation."

- Density estimation: Estimate $f(x)$, where $P(X \in A) = \int_A f(x)dx$. This is related to the nonparametric estimator of the CDF we studied at the beginning of the course.

- Nonparametric regression: Estimate $f(x)$, where $f(x) = E[Y|X = x]$. This extends the linear regression model we looked at recently.

Finding good estimates in both cases will involve understanding the role of *smoothing* in the *bias-variance tradeoff*.

Let f be the function we're trying to estimate. As with the ECDF $\hat{F}_n(x)$, in studying a particular estimator $\hat{f}_n(x)$, we need to keep track of two things that can vary: the observed data used to construct $\hat{f}_n$, and the value of x at which we evaluate the function.

We will (primarily for tractability) use the integrated squared error (ISE) as our loss function:

$$L(f, \hat{f}_n) = \int [f(x) - \hat{f}_n(x)]^2 dx$$

This gives us a frequentist risk function

$$R(f, \hat{f}_n) = E[L(f, \hat{f}_n)]$$

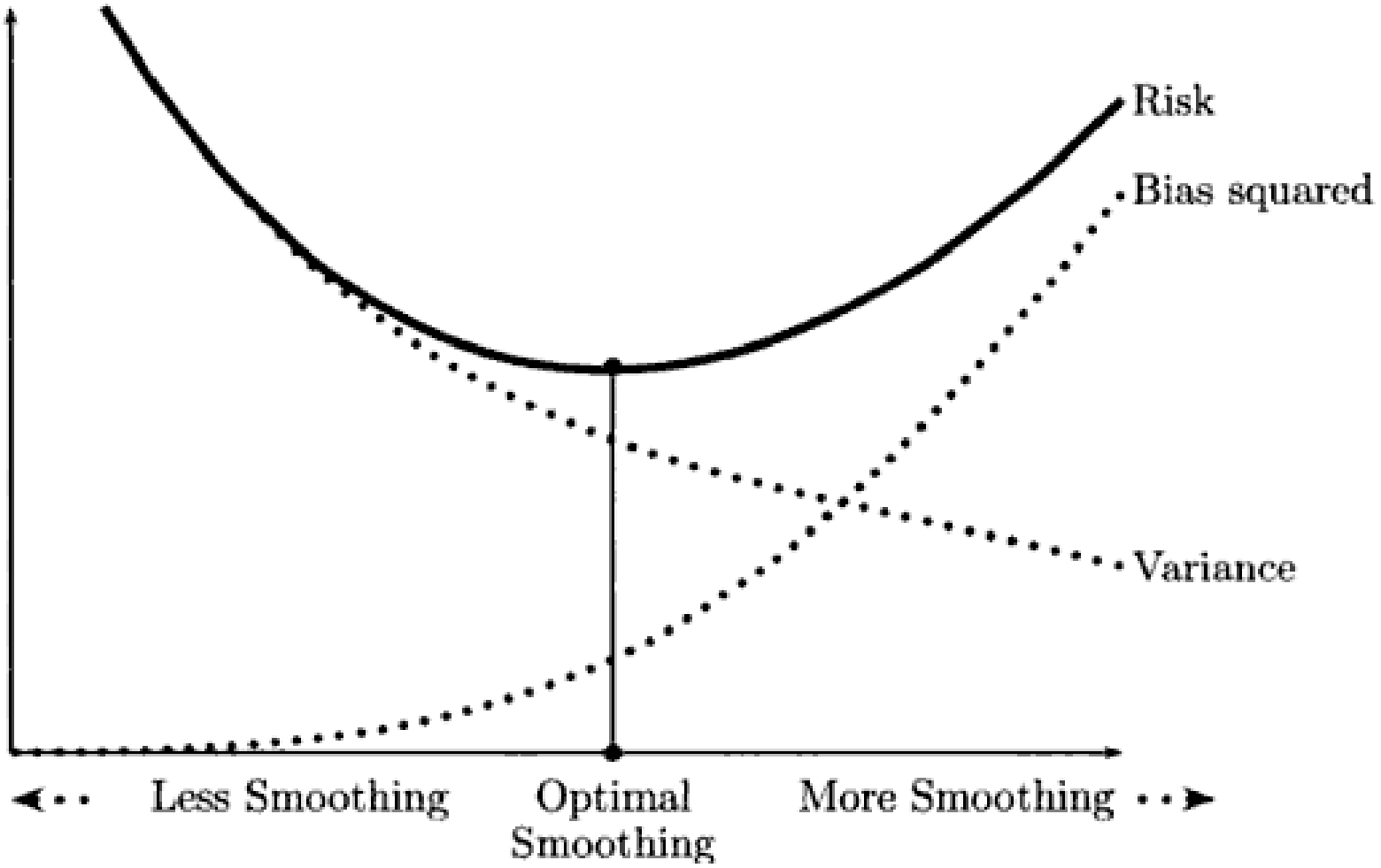
We can rewrite the risk as

$$R(f, \hat{f}_n) = \int b^2(x)dx + \int v(x)dx$$

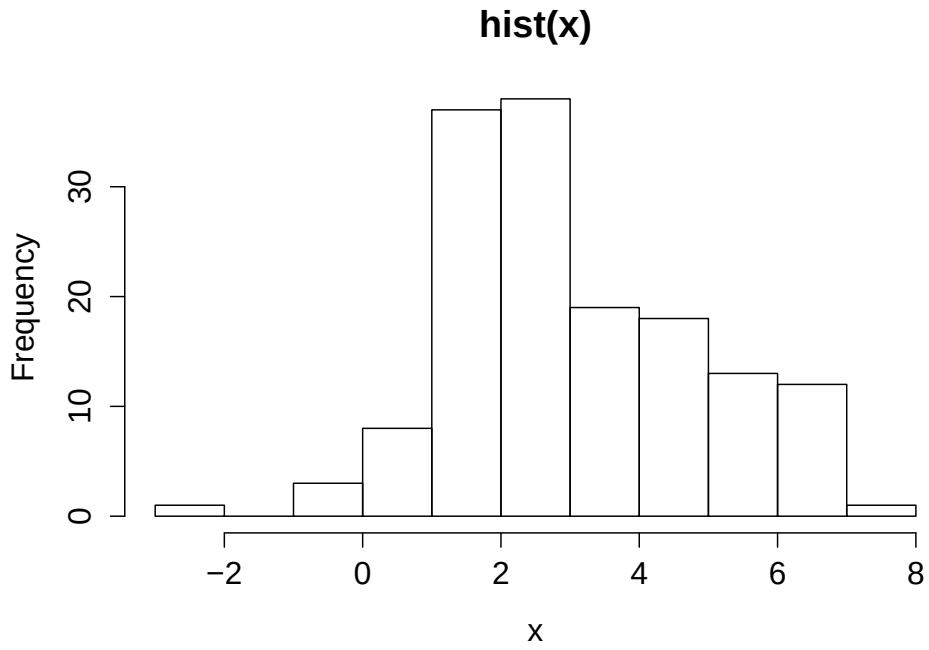
where

$$\begin{aligned} b(x) &= E[\hat{f}_n(x)] - f(x) \\ v(x) &= V[\hat{f}_n(x)] \end{aligned}$$

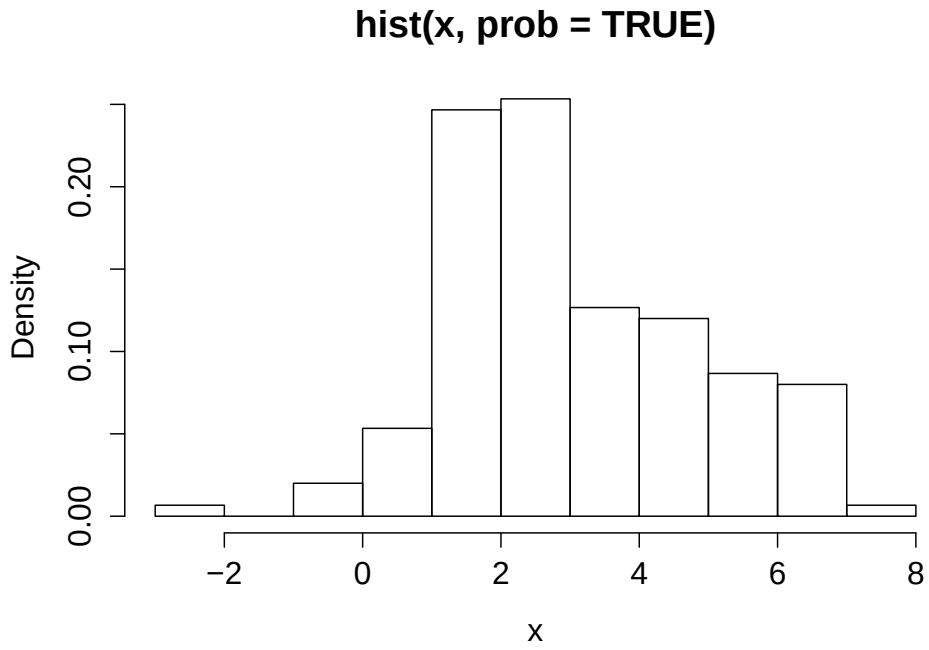
In the cases we will study, $\hat{f}_n$ will depend not only on the data, but also on a choice of *smoothing parameter*. We want to choose the smoothing parameter to minimize $R(f, \hat{f}_n)$. Typically we have that $b(x)$ increases with more smoothing and $v(x)$ decreases, so that the sum $R(f, \hat{f}_n)$ involves a *tradeoff* between bias and variance.


Risk
Bias squared
Variance
Less Smoothing
Optimal Smoothing
More Smoothing

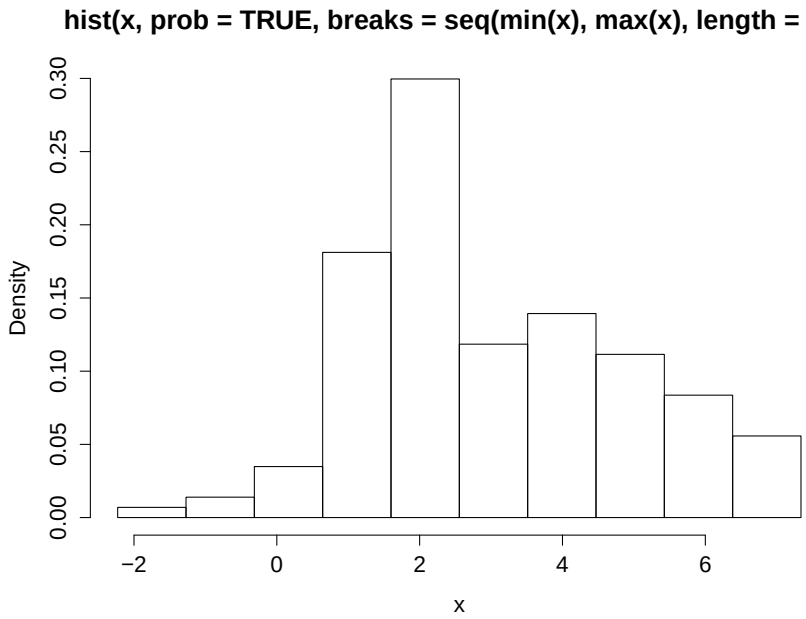
We will start with a simple form of density estimator, which is the histogram. The histogram most of us learn about in introductory statistics divides the range of the data into a number of equally spaced *bins* and then displays the number of observations that fall into each bin.

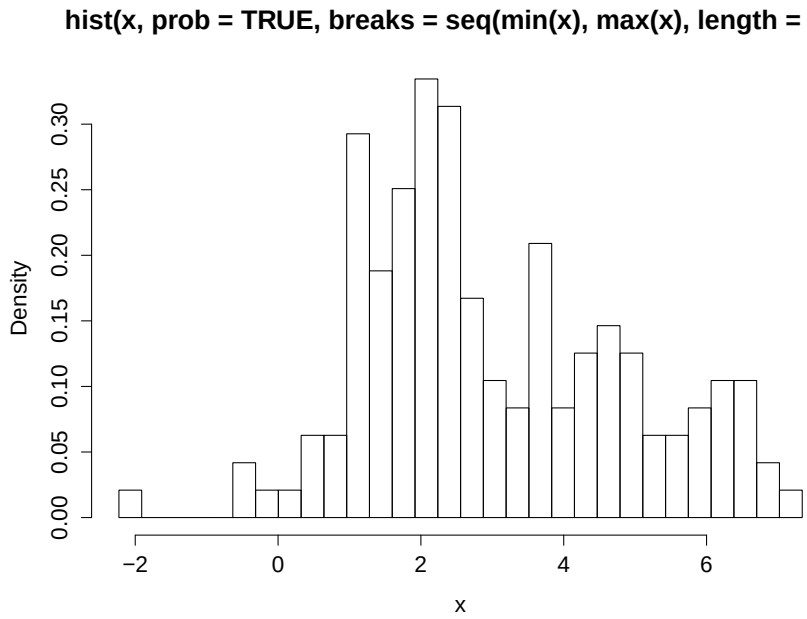


The total area in the bars of the histogram is equal to the number of observations n times the width h of the bins. If we divide the height of each bar by this constant, we get a piecewise constant function that integrates to one.



"Smoothness" in this case is governed by the total number of bins m, or equivalently, by the width of the bins $h = \frac{b-a}{m}$, where $[a, b]$ is the range of the data.





Larger $h \Rightarrow$ more smoothness.

To make this precise, consider observing $X_1, \ldots, X_n$, iid on $[0,1]$, and define bins

$$B_1 = \left[0, \frac{1}{m}\right), B_2 = \left[\frac{1}{m}, \frac{2}{m}\right), \ldots, B_m = \left[\frac{m-1}{m}, 1\right]$$

Then $h = 1/m$ is the binwidth. Let v_j denote the number of observations in bin B_j, and define

$$\begin{aligned} \hat{p}_j &= v_j/n \\ p_j &= \int_{B_j} f(u)du \end{aligned}$$

Note that $\hat{p}_j$ is the plug-in estimator of p_j, with $E[\hat{p}_j] = p_j$ and $V[\hat{p}_j] = p_j(1-p_j)/n$.

Define the histogram estimator of the density f to be

$$\begin{aligned}\hat{f}_n(x) &= \begin{cases} \hat{p}_1/h & x \in B_1 \\ \hat{p}_2/h & x \in B_2 \\ \vdots & \\ \hat{p}_m/h & x \in B_m \end{cases} \\ &= \sum_{j=1}^{m} \frac{\hat{p}_j}{h} I\{x \in B_j\}\end{aligned}$$

The motivation for $\hat{f}$ is that for $x \in B_j$, for small h we have

$$\int_{B_j} f(u)du \approx f(x)h$$

For $x \in B_j$,

$$\begin{aligned} E[\hat{f}_n(x)] &= E[\hat{p}_j]/h = p_j/h \\ V[\hat{f}_n(x)] &= V[\hat{p}_j]/h^2 = p_j(1-p_j)/(nh^2) \end{aligned}$$

From this we can approximate the risk as

$$R(\hat{f}_n, f) \approx \frac{h^2}{12} \int [f'(u)]^2 du + \frac{1}{nh}$$

This is minimized at $h^* = \frac{1}{n^{1/3}} \left(\frac{6}{\int [f'(u)]^2 du} \right)^{1/3}$, for which the risk is $R(\hat{f}_n, f) \approx C/n^{2/3}$. The rate of convergence, $n^{-2/3}$, is slower than for parametric models, for which the rate is typically n^{-1}.

In practice, we do not know h^*, since this depends on the unknown f. Instead we choose h to minimize an *estimate* of the risk. Specifically, note that the loss function, written as a function of h, is

$$\begin{aligned} L(h) &= \int [\hat{f}_n(x) - f(x)]^2 dx \\ &= \int \hat{f}_n^2(x) dx - 2\int \hat{f}_n(x) f(x) dx + \int f^2(x) dx \\ &\equiv J(h) + \int f^2(x) dx \end{aligned}$$

Since $\int f^2(x)dx$ doesn't depend on h, minimizing $E[J(h)]$ is equivalent to minimizing the risk.

The cross-validation estimate of $E[J(h)]$ is

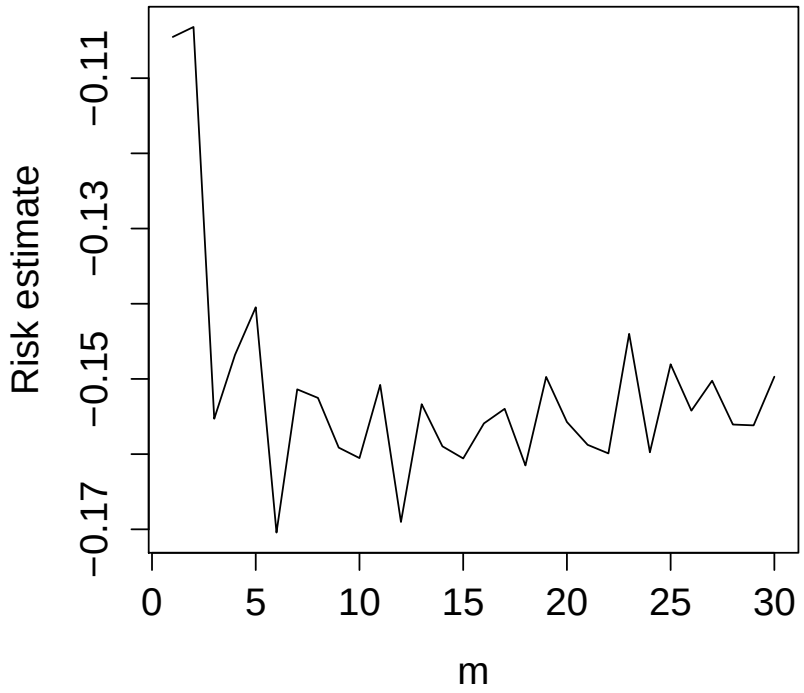
$$\hat{J}_{CV}(h) = \int \hat{f}_n^2(x)dx - \frac{2}{n}\sum_{i=1}^{n} \hat{f}_{(-i)}(X_i)$$

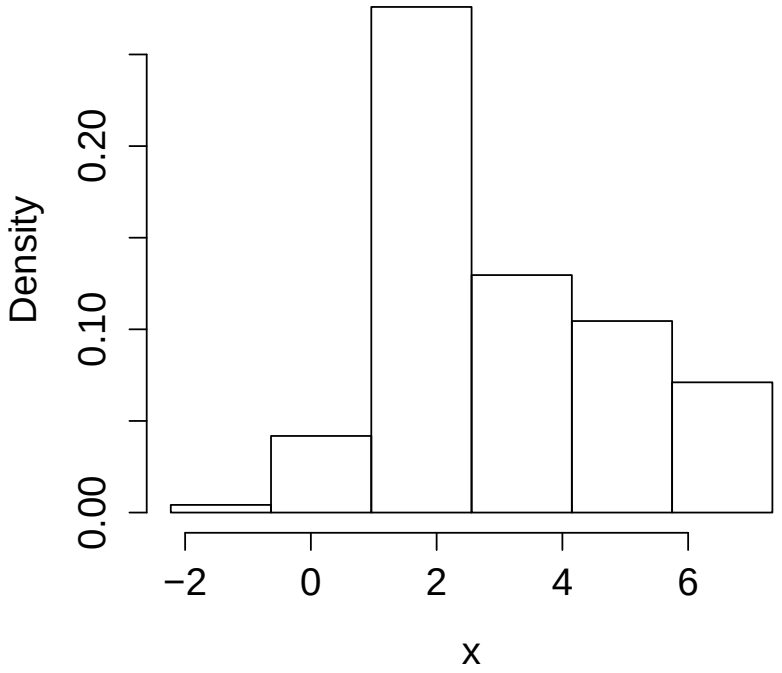
where $\hat{f}_{(-i)}$ is the histogram estimator using all observations except the i^{th} one. Luckily we don't have to recompute the estimator n times, since

$$\hat{J}_{CV}(h) = \frac{2}{(n-1)h} - \frac{n+1}{(n-1)h}\sum_{j=1}^{m} \hat{p}_j^2$$

To do this in practice, look at the number of bins $m = 1/h$, rather than h, since h can only take on the values $h = 1/m$ for integer values of m.

```
jhat <- function(m, x){
  breaks <- seq(min(x), max(x), length = m+1) # m bins
  h <- (breaks[2] - breaks[1])
  n <- length(x)
  phat <- hist(x, breaks = breaks, plot = FALSE)$counts / n
  return(2/((n-1)*h) - (n+1)/((n-1)*h) * sum(phat^2))
}
mvals <- 1:30
risk <- sapply(mvals, jhat, x = x)
par(mfrow = c(1, 2))
plot(mvals, risk, type = "l",
     xlab = "m", ylab = "Risk estimate")
mopt <- mvals[risk==min(risk)]
hist(x, breaks = seq(min(x), max(x), length = mopt+1),
     prob = TRUE, main = "")
```

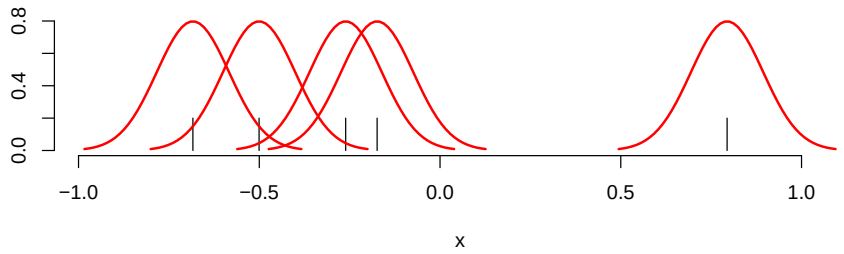

Risk estimate
−0.17
−0.15
−0.13
−0.11
0
5
10
15
20
25
30
m


Density
0.00
0.10
0.20
−2
0
2
4
6
x

A smoother estimate of the density can be obtained using a *kernel density estimator* (KDE).

Consider the estimate of the density corresponding to the ECDF. It puts probability mass $1/n$ on each datapoint. That is, we have a discrete density where each datapoint is equally likely, but values close to the datapoints but not exactly equal have probability zero.

The idea behind the KDE is that each datapoint really gives evidence to nearby values as well, and we should take that $1/n$ mass and "spread it out." We do this using *kernels*.



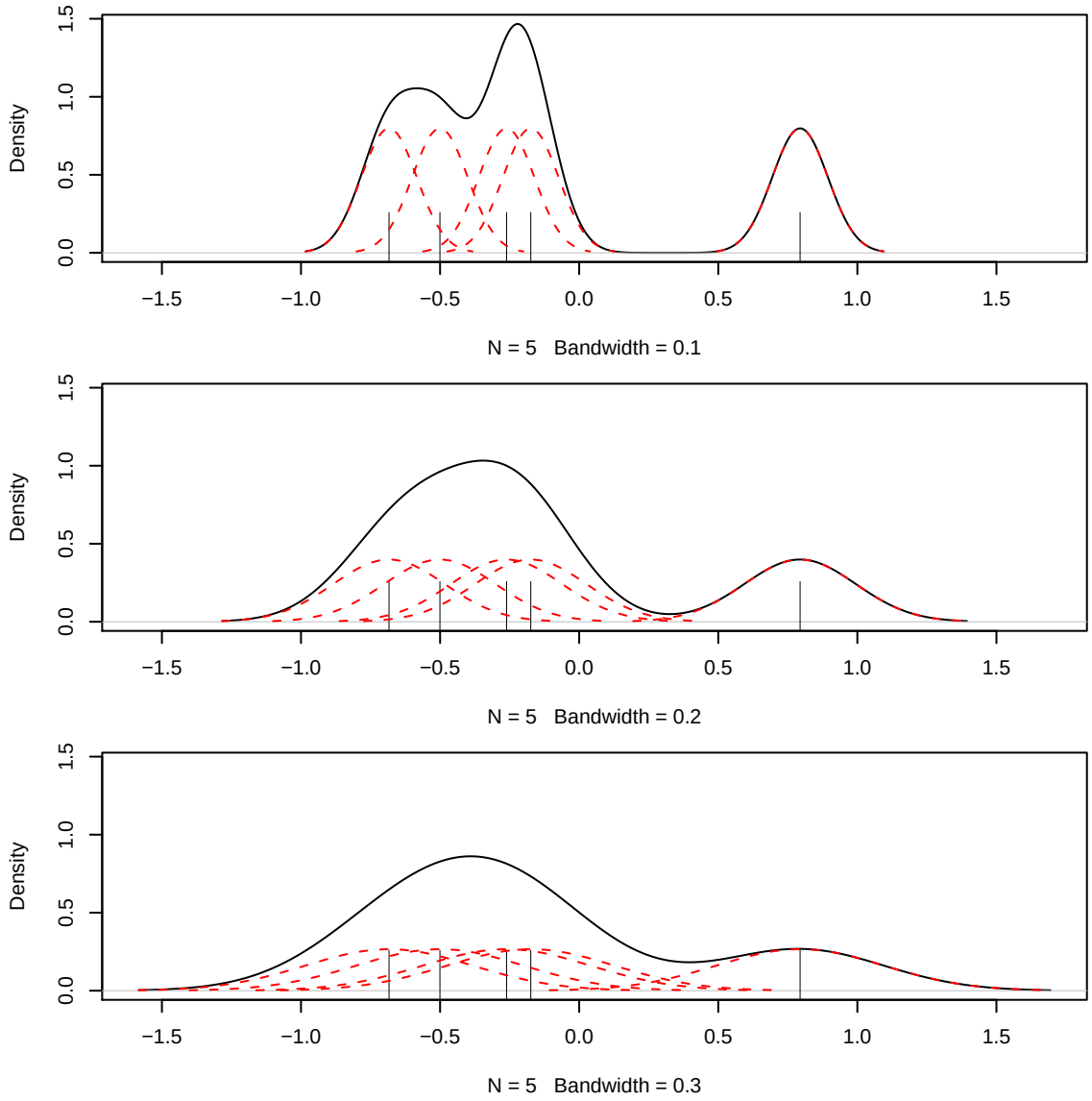
A kernel is a function K such that

- $K(x) \geq 0$
- $\int K(x)dx = 1$
- $\int xK(x)dx = 0$
- $\int x^2K(x)dx \equiv \sigma_K^2 > 0$

The kernel density estimator of f is

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

where h is called the bandwidth. The choice of h matters more than the functional form of K.


Density
0.0
0.5
1.0
1.5
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
N = 5 Bandwidth = 0.1
Density
0.0
0.5
1.0
1.5
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
N = 5 Bandwidth = 0.2
Density
0.0
0.5
1.0
1.5
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
N = 5 Bandwidth = 0.3

As for the histogram estimator, we can approximate the risk. Here it is

$$R(f, \hat{f}_n) \approx \frac{1}{4}\sigma_K^4 h^4 \int [f''(x)]^2 dx + \frac{1}{nh} \int K^2(x) dx$$

Minimizing with respect to h, we find the optimal bandwidth is

$$h^* = \frac{c_1^{-2/5} c_2^{1/5} c_3^{-1/5}}{n^{1/5}}$$

for $c_1 = \sigma_K^2$, $c_2 = \int [K(x)]^2 dx$, and $c_3 = \int [f''(x)]^2 dx$. With this choice of h, the risk is

$$R(f, \hat{f}_n) = \frac{c_4}{n^{4/5}}$$

for another constant c_4. Note the rate of $n^{-4/5}$, which is faster than a histogram but still slower than a parametric estimator.

The constant c_4 on the previous page is

$$\begin{aligned} c_4 &= \frac{5}{4}(\sigma_K^2)^{2/5}\left[\int [K(x)]^2 dx\right]^{4/5}\left[\int [f''(x)]^2 dx\right]^{1/5} \\ &\equiv C(K)\left[\int [f''(x)]^2 dx\right]^{1/5} \end{aligned}$$

Therefore, other things being equal, we should choose the kernel K to minimize $C(K)$. Hodges and Lehmann (1956) showed that this problem is solved by setting $K(x)$ to be the Epanechnikov kernel:

$$K(x) = \begin{cases} \frac{3}{4\sqrt{5}}(1 - x^2/5) & |x| < \sqrt{5} \\ 0 & otherwise \end{cases}$$

However, there is actually very little difference in C for different kernels.

Since f is unknown, we can't compute the optimal h. Instead, we again estimate $E[J(h)] = E[\int \hat{f}_n^2(x)dx - 2\int \hat{f}_n(x)f(x)dx]$ using cross-validation:

$$\hat{J}_{CV}(h) = \int \hat{f}_n^2(x)dx - \frac{2}{n}\sum_{i=1}^{n} \hat{f}_{(-i)}(X_i)$$

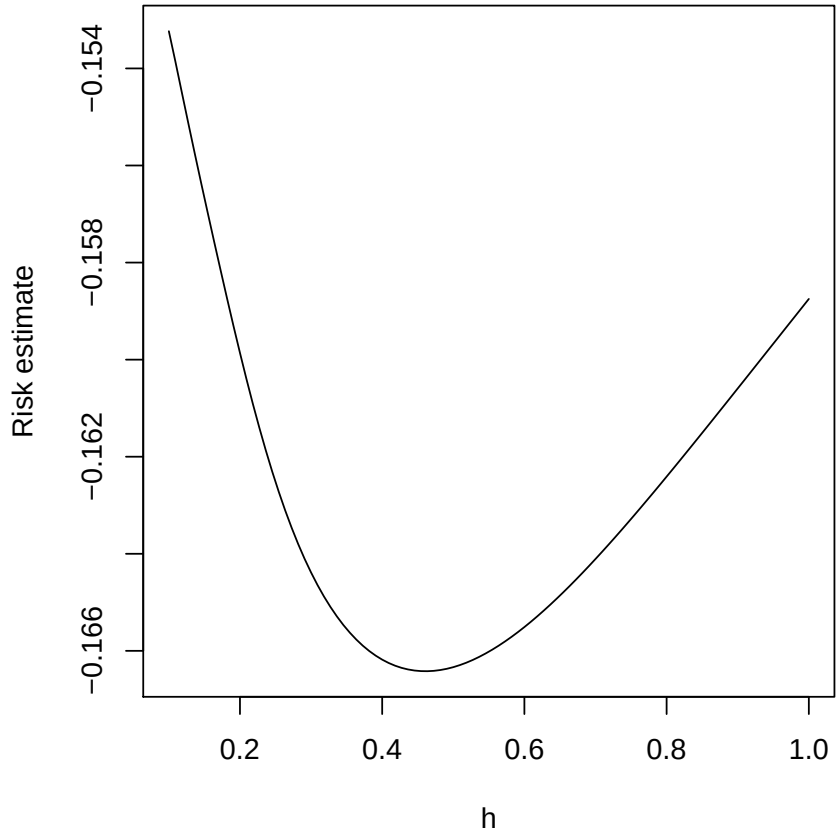
where $\hat{f}_n$ is the KDE with bandwidth h and $\hat{f}_{(-i)}$ is the KDE with bandwidth h computed using all but the i^{th} observation.

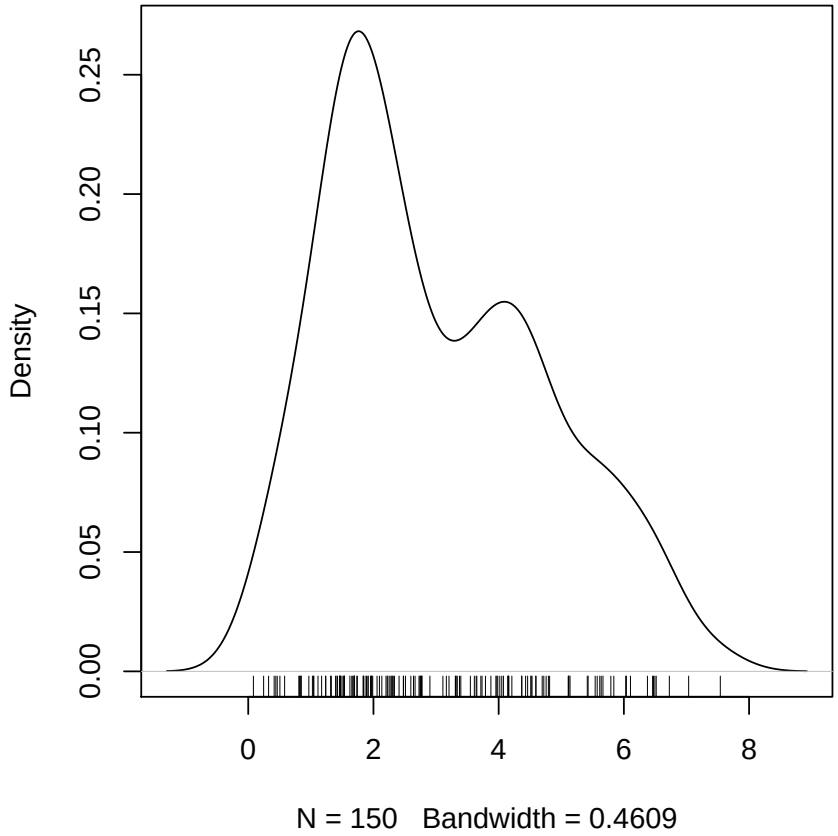
One can show that $\hat{J}(h)$ is unbiased for $E[J(h)]$, and that it may be further approximated using the simpler expression

$$\hat{J}_{CV}(h) \approx \frac{1}{hn^2}\sum_{i=1}^{n}\sum_{j=1}^{n} K^*\left(\frac{X_i - X_j}{h}\right) + \frac{2}{nh}K(0)$$

where $K^*(x) = K^{(2)}(x) - 2K(x)$ and $K^{(2)}(x) = \int K(x-y)K(y)dy$.

```
jhat <- function(h, x){ # defined for normal kernel only
  n <- length(x)
  crossmat <- outer(x, x, "-")/h
  kstar <- dnorm(crossmat, sd = sqrt(2)) - 2*dnorm(crossmat)
  return(sum(kstar)/(h*n^2) + 2*dnorm(0)/(n*h))
}
hseq <- seq(0.1, 1, length = 100)
risk <- sapply(hseq, jhat, x = x)
par(mfrow = c(1, 2))
plot(hseq, risk, type = "l",
     xlab = "h", ylab = "Risk estimate")
h.opt <- optimize(jhat, lower = 0.1,
                  upper = 1, x = x)$minimum
plot(density(x, bw = h.opt), main = "")
rug(x)
```


Risk estimate
−0.154
−0.158
−0.162
−0.166
0.2
0.4
0.6
0.8
1.0
h


Density
0.25
0.20
0.15
0.10
0.05
0.00
0
2
4
6
8
N = 150 Bandwidth = 0.4609