1.2.1 Special cases: local constant and local linear

The local constant regression estimator was due originally to Nadaraya (1964) and Watson (1964), thus it is often referred to as the Nadaraya-Watson estimator.

In this case, the solution to the problem (1.9) has a simple explicit form

$$\widehat{\beta}_0 = \frac{\sum_{i=1}^n W_h(x_i - x_0) y_i}{\sum_{i=1}^n W_h(x_i - x_0)}.$$

Thus, the regression function estimator can be written explicitly as

$$\widehat{g}(x) = \frac{\sum_{i=1}^n W_h(x_i - x) y_i}{\sum_{i=1}^n W_h(x_i - x)}.$$

The local linear regression estimator (i.e. $p = 1$) can be written explicitly as

$$\widehat{g}(x) = \frac{S_2(x)T_0(x) - S_1(x)T_1(x)}{S_2(x)S_0(x) - S_1^2(x)}$$

where

$$S_i(x) = \sum_{j=1}^n W_h(x_j - x)(x_j - x)^i, \quad i \geq 0$$

and

$$T_i(x) = \sum_{j=1}^n W_h(x_j - x)(x_j - x)^i y_j, \quad i \geq 0.$$

Efficient calculation of this estimator is discussed in Wand and Jones (1995), for example.

1.2.2 Asymptotic accuracy

When the regression function, $g(x)$, is nonlinear, kernel regression estimators will be biased. The bias increases with the bandwidth, h. Thus, small bias will be achieved if h is taken to be very small. However, it can be shown that the variance of the estimator increases as h decreases. Thus, we should not take h too small. The best value of h will be the one that trades off bias against variance. Usually, we would prefer a bandwidth which gives smallest mean squared error (pointwise) or mean integrated squared error (MISE) over the interval containing the predictor values. It is usually most convenient to consider asymptotic approximations to these quantities. The asymptotics depend upon the sample size n becoming infinitely large and the bandwidth h becoming very small. We will see that for local constant and local linear regression, the bandwidth should be in the order of $n^{-1/5}$. We will also see that there is additional bias near the boundaries of the data for local constant regression, but there is no such problem for local linear regression.

Bias

For simplicity, we assume that the uniform kernel $W_h(x)$ is in use. We also assume that the regression function, $g(x)$, has three continuous derivatives, and that the predictor values are specified at $x_i = i/(n+1)$, for $i = 1, 2, \ldots, n$. We consider estimation of the regression function on the interval $[0, 1]$. The results that we will obtain here can be derived under more general conditions; the basic idea behind the derivation remains unchanged, but the details may obscure the main ideas.

We consider the Nadaraya-Watson estimator first. It can be expressed as

$$\widehat{g}(x) = \frac{\sum_{i=1}^n W_i g(x_i)}{\sum_{i=1}^n W_i} + \frac{\sum_{i=1}^n W_i \varepsilon_i}{\sum_{i=1}^n W_i} \tag{1.10}$$

where $W_i = W_h(x_i - x)$. By assumption, $E[\varepsilon_i] = 0$. Thus,

$$E[\widehat{g}(x)] = \frac{\sum_{i=1}^n W_i g(x_i)}{\sum_{i=1}^n W_i}$$

Expanding $g(x_i)$ in Taylor series about x, we may deduce that

$$E[\widehat{g}(x)] = g(x) + g'(x)\frac{\sum_{i=1}^n (x_i - x) W_i}{\sum_{i=1}^n W_i} + g''(x)\frac{\sum_{i=1}^n (x_i - x)^2 W_i}{2\sum_{i=1}^n W_i} + R \tag{1.11}$$

where R denotes the remainder term, which can be shown to be of order $O(h^4)$, as $h \to 0$.

This expression can be simplified further. In order to proceed, we recall the following integration property

$$n^{-1}\sum_{i=1}^n (x_i - x)^\ell W_h(x_i - x) = \int_0^1 (y - x)^\ell W_h(y - x)dy + O(n^{-1}). \tag{1.12}$$

Using the case where $\ell = 0$, we deduce that

$$n^{-1}\sum_{i=1}^n W_i = 1 + O(n^{-1})$$

whenever $x \in (h, 1-h)$. For x nearer to the boundary of $[0, 1]$, the leading term of this expansion can be shown to be between .5 and 1.

The cases where $\ell = 1$ and $\ell = 2$ lead to

$$n^{-1}\sum_{i=1}^n (x_i - x) W_i = O(n^{-1})$$

and

$$n^{-1}\sum_{i=1}^n (x_i - x)^2 W_i = h^2 \mu_2(W) + O(n^{-1})$$

whenever $x \in (h, 1-h)$. The first result follows from the symmetry of the kernel. $\mu_2(W)$ denotes the second moment of the kernel[9], viewed as a probability density function.

[9]This is the kernel whose bandwidth is $h = 1$.

Working from (1.11), we can express the bias in $\widehat{g(x)}$ as

$$B(\widehat{g}(x)) = \frac{h^2}{2}\mu_2(W)g''(x) + O(h^4) + O(n^{-1}) \tag{1.13}$$

If x is closer to the boundary, above integral is of order $O(h)$; the symmetry of the kernel no longer helps. For such x, therefore, the bias is of $O(h)$. This is the so-called boundary bias problem of local constant regression. During the 1970's and 1980's, many modifications were suggested to combat this problem. None are as simple and elegant as the solution provided by local linear regression, a discovery of the mid-1990's.

Using the same kinds of approximation techniques as for the local constant case, we can obtain the bias expression (1.11) for local linear regression. However, this expression can obtained without appealing to the symmetry of the kernel. Thus, the expression holds for all $x \in [0, 1]$. Thus, local linear regression has a bias of order $O(h^2)$ throughout the interval $[0, 1]$; there is no additional boundary bias.

Variance

To compute the variance of the Nadaraya-Watson estimator, we may, again, begin with (1.10). However, this time, the first term plays no role. Taking the variance of the second term, we obtain

$$\text{Var}(\widehat{g}(x)) = \frac{\sum_{i=1}^n W_i^2\sigma^2}{(\sum_{i=1}^n W_i)^2}.$$

Arguing as before, this can be written as

$$\text{Var}(\widehat{g}(x)) = \frac{n\int_0^1 W_h^2(x-y)dy\sigma^2}{n^2(\int_0^1 W_h(x-y)dy)^2} + O(n^{-1}).$$

A change of variable in the integrands, and approximating gives

$$\text{Var}(\widehat{g}(x)) = \frac{\sigma^2 R(W)}{nh} + O(n^{-1})$$

where the notation $R(W) = \int W^2(z)dz$ is being used.

For points in the interior of $[0, 1]$, we can then show that the MSE for the Nadaraya-Watson estimator is of the order $O(h^4 + (nh)^{-1})$. Formally minimizing this with respect to h, we can see that $h = O(n^{-1/5})$. This implies that the MSE only decreases at rate $n^{-4/5}$ as n increases; this is slower than the rate n^{-1} for estimators like the sample mean or the slope in simple linear regression.

Similar, but more involved, argumentation leads to the same result for local linear regression.

Bandwidth selection

The quality of the kernel regression estimate depends crucially on the choice of bandwidth. Many proposals for bandwidth selection have been made over the last three decades. None are completely satisfactory. Loader (1999) argues for a choice based on cross-validation, while Wand and Jones (1995) argue for what is called a direct plug-in choice.

A direct plug-in bandwidth is based on the minimizer of the asymptotic mean squared error (or more usually, the mean integrated squared error). A simple calculus argument leads to an optimal bandwidth satisfying

$$h^5 = \frac{\sigma^2 R(W)}{n(\mu_2(W))^2(g''(x))^2}.$$

There are a number of difficulties with this formula. First, σ must be estimated. A number of possibilities exist: usually, a pilot estimate is obtained and the mean residual sum of squares is used.[10] The bigger difficulty is estimation of $g''(x)$. This is usually a harder problem than estimation of $g(x)$ itself. Note, however, that higher order kernel regression estimators can be used to obtain such estimates – simply read off the $\widehat{\beta}_2$ coefficient estimate in local cubic regression, for example. The difficulty here is that a different bandwidth is required to get the best estimate of $g''(x)$. The best bandwidth depends on higher derivatives of $g(x)$. Thus, there is an infinite regress; it is impossible to obtain the best estimate. In practice, a quick and simple bandwidth is used to obtain a high order derivative estimate, and then the procedure described above is carried out. Wand and Jones (1995) give details of the procedure together with a discussion of other bandwidth selectors, including cross-validation. The function `dpill()` in the *KernSmooth* package implements a direct plug-in selector for local linear regression.

1.2.3 Lowess

Cleveland (1979) introduced an alternative form of local polynomial regression which he termed *lowess*.[11] The name stands for Locally Weighted Scatterplot Smoothing.

The basic idea is that of local polynomial regression where the tricube kernel

$$W(t) = \frac{70}{81}(1 - |t|^3)^3 I_{[-1,1]}(t)$$

is used. This kernel is used because it is very smooth at -1 and 1 (i.e. $W_h(t)$ is smooth at $-h$ and h.)

The first real difference between lowess and local polynomial regression comes from the choice of bandwidth. Cleveland proposed a *nearest neighbour bandwidth*. Take $f \in (0, 1]$, and set $r = [nf + .5]$. For each predictor value x_k, let

$$h_k = |x_k - x_\cdot|_{(r)}.$$

In other words, h_k is the rth order statistic of the sample $|x_k - x_1|, |x_k - x_2|, \ldots, |x_k - x_n|$. The local polynomial regression problem (at x_k) is to minimize

$$\sum_{i=1}^{n}(y_i - \sum_{j=0}^{p}\beta_j(x_i - x_k)^j)^2 W_{h_k}(x_i - x_k)$$

[10] Degrees of freedom are computed from the trace of the associated hat matrix.

[11] It is also called *loess*, and there are, in fact, R functions with each name.