Chapter 1. Nonparametric Curve Estimation Part 5

February 5, 2007

1 Stochastic expansion of local linear kernel estimator

Suppose (X_i, Y_i) is a random sample and consider

$$Y_i = m(X_i) + \varepsilon_i$$

If X_i is close to x, then

$$m(X_i) \approx m(x) + m'(x)(X_i - x) + \frac{1}{2}m''(x)(X_i - x)^2$$

and the model can be written as

$$Y_i \approx m(x) + m'(x)(X_i - x) + \frac{1}{2}m''(x)(X_i - x)^2 + \varepsilon_i$$

Recall the estimator is

$$\hat{m}(x) = n^{-1}\sum_{i=1}^{n}\{s_{n,2}(x)K_h(X_i-x) - s_{n,1}(x)K_h(X_i-x)\Big(\frac{X_i - x}{h}\Big)\}Y_i/\{s_{n,2}(x)s_{n,0}(x) - s_{n,1}^2(x)\}$$

where

$$s_{n,k}(x) = n^{-1}\sum_{i=1}^{n} K_h(X_i - x)\Big(\frac{X_i - x}{h}\Big)^k, \quad k = 0, 1, 2, ...$$

Base on these, we have (please prove)

$$\begin{aligned}\hat{m}(x) \quad &\approx \quad m(x) + \frac{1}{2}m''(x)\frac{s_{n,2}^2(x) - s_{n,1}(x)s_{n,3}(x)}{s_{n,2}(x)s_{n,0}(x) - s_{n,1}^2(x)}h^2 \\ &\quad +n^{-1}\sum_{i=1}^{n}\{s_{n,2}(x)K_h(X_i - x) - s_{n,1}K_h(X_i - x)\Big(\frac{X_i - x}{h}\Big)\}\varepsilon_i/\{s_{n,2}(x)s_{n,0}(x) - s_{n,1}^2(x)\}\end{aligned}$$

Theorem 1.1 *Under some conditions, we have*

$$Bias(\hat{m}(x)) = E(\hat{m}(x)) - m(x) \approx \frac{1}{2}m''(x)h^2$$

and

$$Var(\hat{m}(x)) \approx \frac{\sigma^2 d_0}{nhf(x)}$$

where $\sigma^2 = E\varepsilon_i^2$ *and* $d_0 = \int K^2$.

Table 1: Comparison of NW and LL kernel estimators for the inner point

method	Bias	Variance
NW	$\frac{1}{2}c_2 m''(x)h^2 + c_2 f^{-1}(x)f'(x)m'(x)h^2$	$\frac{\sigma^2 d_0}{nhf(x)}$
LL	$\frac{1}{2}c_2 m''(x)h^2$	$\frac{\sigma^2 d_0}{nhf(x)}$

where $c_2 = \int K(v)v^2 dv$.

For the boundary points, LL has even better performance than NW kernel estimator.

The optimal bandwidth that minimizes MSE

$$E\{\hat{m}(x) - m(x)\}^2 = bias^2 + variance \approx \frac{1}{4}c_2^2(m''(x))^2h^4 + \frac{\sigma^2 d_0}{nhf(x)}$$

The optimal bandwidth is

$$h(x) = (d_0/c_2^2)^{1/5}\Big[\frac{\sigma^2}{(m''(x))^2 f(x)}\Big]^{1/5} n^{-1/5}.$$

For Gaussian kernel: $(d_0/c_2^2)^{1/5} = 0.776$; for Epanechnikov kernel: $(d_0/c_2^2)^{1/5} = 1.719$

Theorem 1.2 *Under some conditions, we have*

$$\sqrt{nh}\{\hat{m}(x) - \frac{1}{2}m''(x)h^2\} \to N(0, \frac{\sigma^2 d_0}{f(x)})$$

where $\sigma^2 = E\varepsilon_i^2$ *and* $d_0 = \int K^2$. *If* $nh^5 \to 0$, *then we have the following point-wise 95% confidence band for* $m(x)$

$$[L_n(x), U_n(x)]$$

where

$$L(x) = \hat{m}(x) - 1.96\sqrt{\frac{\hat{\sigma}^2 d_0}{nh\hat{f}(x)}},$$

$$m(x) = \hat{m}(x) + 1.96\sqrt{\frac{\hat{\sigma}^2 d_0}{nh\hat{f}(x)}},$$

and

$$\hat{\sigma}^2 = n^{-1}\sum_{i=1}^{n}(Y_i - \hat{m}(X_i))^2, \quad \hat{f}(x) = n^{-1}\sum_{i=1}^{n} K_h(X_i - x).$$

Example 1.3 (motorcycle) **(data)**

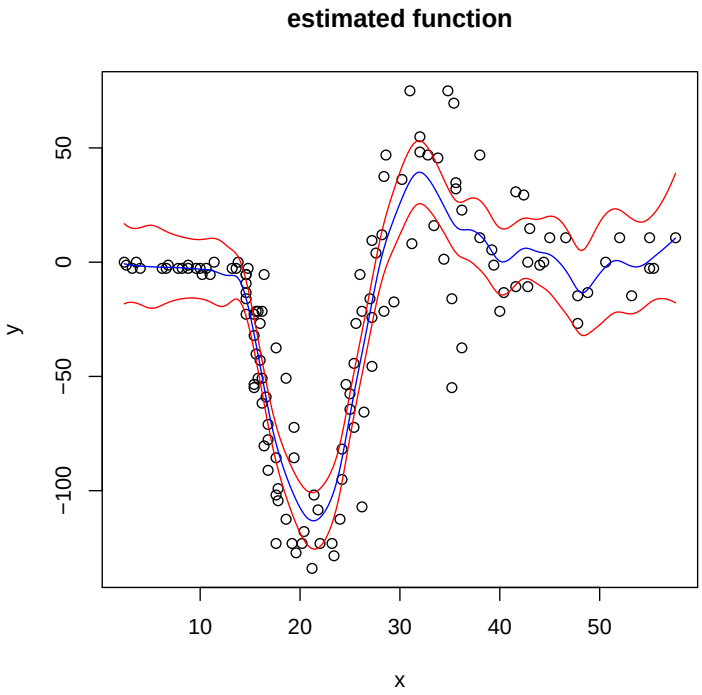

Figure 1: calculation of Example 1.3: local linear estimator of the regression function. "o" denotes the observed values; blue curve denotes the estimated function; the red lines denotes the 95% point-wise confidence band. **(code)**

1.1 bandwidth selection for local linear kernel estimation

The same approaches as for NW kernel estimation can be used here. (please give the details of the cross-validation method for local liner kernel estimation)

2 Estimation of derivatives

Recall that

$$\begin{pmatrix} \hat{m}(x) \\ \hat{m}'(x) \end{pmatrix} = \Big\{ \sum_{i=1}^{n} K_h(X_i - x) \begin{pmatrix} 1 \\ X_i - x \end{pmatrix} \begin{pmatrix} 1 \\ X_i - x \end{pmatrix}^{\top} \Big\}^{-1} \times \sum_{i=1}^{n} K_h(X_i - x) \begin{pmatrix} 1 \\ X_i - x \end{pmatrix} Y_i$$

We can also estimate the derivatives of the regression function. It is proved that under some conditions, we have

$$E\hat{m}'(x) - m(x) \to 0$$

Example 2.1 *Consider*

$$Y = \sin(2\pi X) + 0.2\varepsilon$$

where $X \sim uniform(0,1)$ *and* $\varepsilon \sim N(0,1)$.

. *The true regression function is* $m(x) = \sin(2\pi x)$

. *The true regression derivative function is* $m(x) = 2\pi\cos(2\pi x)$

100 observations are drawn from the model. the estimated results are shown in Fig 2

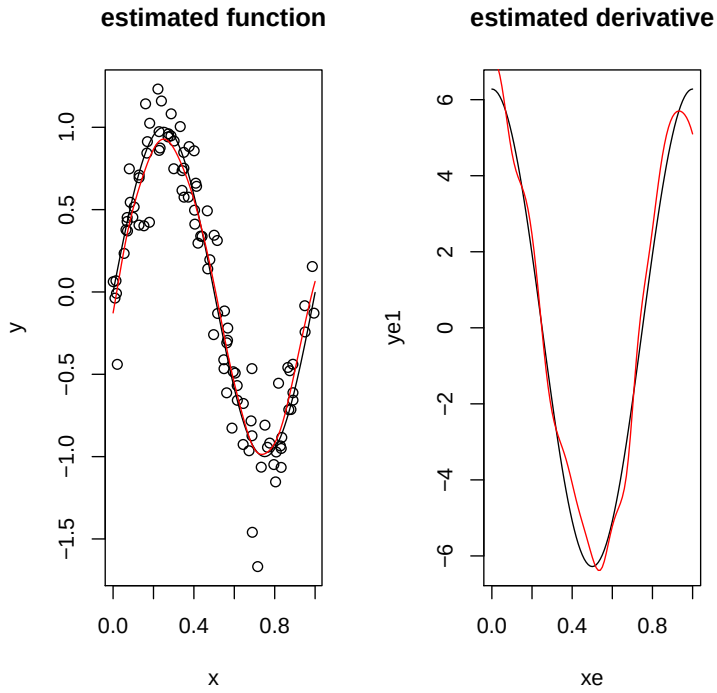

Figure 2: calculation of Example 2.1. In the left panel, black: true function; red: LL kernel estimator. In the right panel, black: true derivative function; red: LL kernel estimated derivative. **(ksLL) (code)**

Consider

$$Y = 4(X - 0.5)^2 + 0.2\varepsilon$$

where $X \sim uniform(0,1)$ *and* $\varepsilon \sim N(0,1)$.

. *The true regression function is* $m(x) = 4(x - 0.5)^2$

. *The true regression derivative function is* $m(x) = 8(x - 0.5)$.

100 observations are drawn from the model. the estimated results are shown in Fig 3

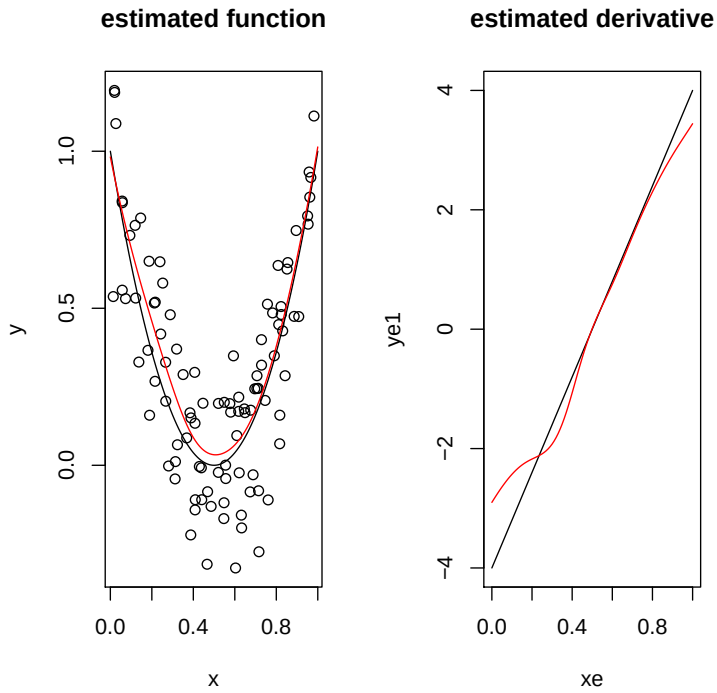

Figure 3: In the left panel, black: true function; red: LL kernel estimator. In the right panel, black: true derivative function; red: LL kernel estimated derivative. **(ksLL) (code)**

3 Local polynomial kernel estimation

Again, consider the conditional expectation of Y given $X = x$. Suppose that $(X_i, Y_i), i = 1, \cdots, n$ are samples.

$$Y_i = m(X_i) + \varepsilon_i$$

For any given point x and any X_i, if X_i is close to x we consider a local polynomial approximation

$$m(X_i) \approx m(x) + m'(x)(X_i - x) + ... + \frac{1}{k!} m^{(k)}(x)(X_i - x)^k.$$

Thus the model is

$$Y_i \approx m(x) + m'(X_i - x) + ... + \frac{1}{k!} m^{(k)}(x)(X_i - x)^k + \varepsilon_i$$

or

$$
\begin{aligned}
Y_1 &\approx m(x) + m'(x)(X_i - x) + ... + \frac{1}{k!}m^{(k)}(x)(X_i - x)^k + \varepsilon_1 \\
Y_2 &\approx m(x) + m'(x)(X_i - x) + ... + \frac{1}{k!}m^{(k)}(x)(X_i - x)^k + \varepsilon_2 \\
&\vdots \\
Y_n &\approx m(x) + m'(x)(X_i - x) + ... + \frac{1}{k!}m^{(k)}(x)(X_i - x)^k + \varepsilon_n
\end{aligned}
$$

We use the following weighted least squares problem to estimate the value $m(x)$ and $m'(x)$.

$$\sum_{i=1}^{n}\{Y_i - m(x) - m'(x)(X_i - x) - ... - \frac{1}{k!}m^{(k)}(x)(X_i - x)^k\}^2 K_h(X_i - x).$$

Let

$$\mathbf{X} = \begin{pmatrix} 1 & X_1 - x & ... & (X_1 - x)^k \\ 1 & X_2 - x & ... & (X_2 - x)^k \\ \cdots & & & \\ 1 & X_n - x & ... & (X_n - x)^k \end{pmatrix}, \mathbf{Y} = \begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix}, \beta = \begin{pmatrix} m(x) \\ m'(x) \\ \frac{1}{2}m''(x) \\ \vdots \\ \frac{1}{k!}m^{(k)}(x) \end{pmatrix}$$

and $\mathbf{W}$ be the diagonal matrix of weights

$$W = diag\{K_h(X_i - x)\}.$$

Then the least squares problem can be written as

$$(Y - \mathbf{X}\beta)^\top \mathbf{W}(\mathbf{Y} - \mathbf{X}\beta)$$

The minimizer to the above problem is

$$\hat{\beta} = \begin{pmatrix} \hat{m}(x) \\ \hat{m}'(x) \\ \frac{1}{2}\hat{m}''(x) \\ \vdots \\ \frac{1}{k!}\hat{m}^{(k)}(x) \end{pmatrix} = \{\mathbf{X}^\top \mathbf{W}\mathbf{X}\}^{-\mathbf{1}}\mathbf{X}^\top \mathbf{W}\mathbf{Y}.$$

Under some conditions, we have

$$Bias = E(\hat{m}(x)) - m(x) \approx C_k(K)m^{(k+1)}(x)h^{k+1}$$

and

$$Var(\hat{m}(x)) \approx D_k(K)\frac{\sigma^2}{nhf(x)}$$

where $C_k(K)$ and $D_k(K)$ depend on kernel function K and the order k.

Remarks: It seems there is a trend: higher order polynomial has faster convergence rate. However, the numerical performance does not show this trend because of computational complexity. In practice, local linear kernel estimation is usually used.

Local polynomial kernel smother can also be used to estimate higher order derivatives.

4 Kernel estimation in multi-dimensional data

Suppose we are interested in the relation between Y and $X = (\mathbf{x}_1, \cdots, \mathbf{x}_p)^\top$. The conditional expectation of Y given $X = x$ is

$$m(x_1, \cdots, x_p) = E(Y|\mathbf{x}_1 = x_1, \cdots, \mathbf{x}_p = x_p)$$

Suppose that we have observations $(X_1, Y_1), \cdots, (X_n, Y_n)$, the Nadaraya-Watson (NW) Kernel Estimator is

$$\hat{m}(x) = \frac{\sum_{i=1}^n K_h(X_i - x)Y_i}{\sum_{i=1}^n K_h(X_i - x)}$$

where $K(.)$ is a multivariate nonnegative function, h is the bandwidth and

$$K_h(X_i - x) = h^{-p}K(\frac{\mathbf{x}_{i1} - x_1}{h}, \cdots, \frac{\mathbf{x}_{ip} - x_p}{h}).$$

Typical kernel functions are

- Epanechnikov kernel: $K(x_1, \cdots, x_p) = \{1 - (x_1^2 + \cdots + \mathbf{x}_p^2)\}I(x_1^2 + \cdots + \mathbf{x}_p^2 < 1)$
- Gaussian kernel: $K(x_1, \cdots, x_p) = (2\pi)^{-p/2}\exp(-\frac{1}{2}(x_1^2 + \cdots + x_p^2))$

The bandwidth can be chosen using cross-validation method. We can also try different values for h and plot the figure. Then chose a bandwidth corresponding to the "best figure"

The local linear Kernel Estimator is to approximate the surface by a plane

$$m(X_i) \approx \alpha_0 + \alpha_1(\mathbf{x}_{i1} - x_1) + ... + \alpha_p(\mathbf{x}_{iq} - x_p).$$

where

$$\alpha_0 = m(x), \alpha_1 = \frac{\partial m(x)}{\partial x_1}, ..., \alpha_p = \frac{\partial m(x)}{\partial x_p}.$$

Similarly to estimation of $m(x)$ and its derivatives in univariate case, we consider the local linear least squares

$$\sum_{i=1}^n \{Y_i - \alpha_0 - \alpha_1(\mathbf{x}_{i1} - x_1) - ... - \alpha_p(\mathbf{x}_{ip} - x_p)\}^2 K_h(X_i - x)$$

The estimator, i.e. the minimizer, is

$$\begin{pmatrix} \hat{\alpha}_0 \\ \hat{\alpha}_1 \\ \vdots \\ \hat{\alpha}_p \end{pmatrix} = (\mathbf{X}^\top \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{W} \mathbf{Y}$$

where

$$\mathbf{X} = \begin{pmatrix} 1 & \mathbf{x}_{i1} - x_1 & ... & \mathbf{x}_{ip} - x_p \\ 1 & \mathbf{x}_{21} - x_1 & ... & \mathbf{x}_{2p} - x_p \\ ... & & & \\ 1 & \mathbf{x}_{n1} - x_1 & ... & \mathbf{x}_{np} - x_p \end{pmatrix}, \ \mathbf{Y} = \begin{pmatrix} Y_1 \\ Y_2 \\ ... \\ Y_n \end{pmatrix},$$

and

$$\mathbf{W} = \begin{pmatrix} K_h(X_1 - x) & 0 & ... & 0 \\ 0 & K_h(X_2 - x) & ... & 0 \\ & ... & & \\ 0 & 0 & ... & K_h(X_n - x) \end{pmatrix}.$$

The estimator of $m(x)$ is then

$$\hat{m}(x) = \hat{\alpha}_0.$$

Under some mild conditions, we have for

$$Bias(\hat{m}(x)) \approx C_{K,m}(x) h^2$$

and

$$var(\hat{m}(x)) \approx \frac{D_{K,m}(x)}{n h^p f(x)}$$

where $C_{K,m}(x)$ and $D_{K,m}(x)$ are two functions of x, depending on K and function m.

It is ease to see the MSE is

$$E\{\hat{m}(x) - m(x)\}^2 \approx C_{K,m}^2(x) h^4 + \frac{D_{K,m}(x)}{n h^p f(x)}$$

The optimal bandwidth that the minimizes the MSE is

$$h \propto c n^{-1/(p+4)}$$

With such a bandwidth, we have

$$E\{\hat{m}(x) - m(x)\}^2 \propto n^{-4/(p+4)}.$$

we have the convergence rate (with n) decrease when the dimension p increase. This is the so-called "curse of dimensionality". Because of this, few people apply the kernel smoothing (or other nonparametric methods) to 3 or higher dimensional data directly.

Example 4.1 *1000 observations are drawn from the model*

$$y = \exp\{-8\mathbf{x}_1^2 - 15(\mathbf{x}_2 - 0.5)^2\} + 0.5\exp\{-8\mathbf{x}_1^2 - 15(\mathbf{x}_2 + 0.5)^2) + 0.1\varepsilon$$

where $\mathbf{x}_1, \mathbf{x}_2 \sim Uniform(-1, 1)$ *and* $\varepsilon \sim N(0, 1)$. *Applying the NW estimator, we have the following estimated surface.*

the estimated results are shown in Fig 4

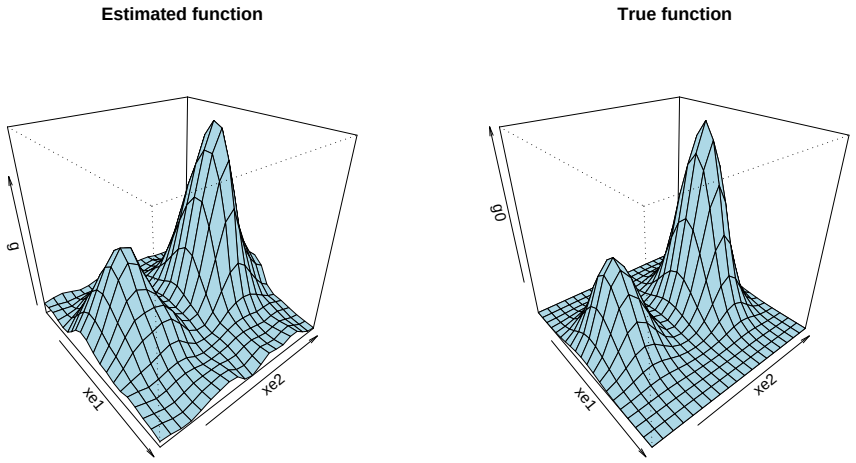

Figure 4: estimated surface of the regression for 4.1. **(ksm) (code)**

Example 4.2 (air pollution in Hong Kong) **(data)**; *Ozone is a second pollutant, i.e. it is generated by chemical reaction of other polluatnts such as* SO_2 *and* NO_2 *with sunlight. Consider the dependency of ozone on* NO_2 *and humidity*

Apply local linear kernel smoothing method, we find the relation as shown Fig 5

From this simulation, we can see that local linear kernel smoothing estimator has better performance at the boundary points than NW local constant estimator.

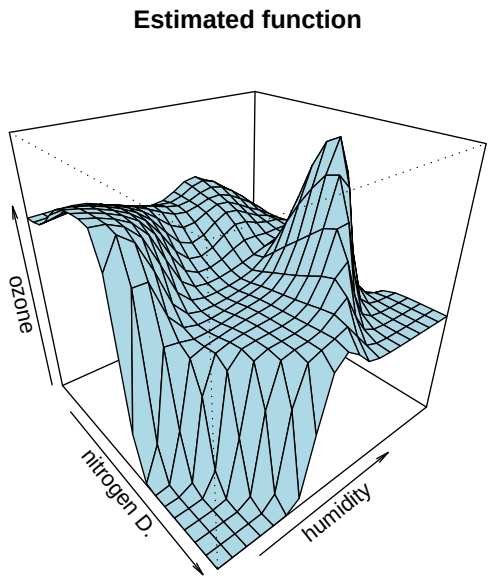

Figure 5: calculation results for Example 4.2. black: true function; cyan: NW estimator; blue: LL kernel estimator. **(ksm) (code)**