Lec 14: Local Regression

Weiping Zhang

2019.11.13

Loess: Local Regression

Fitting local polynomials

Multivariate Local Regression

Local linear kernel regression

Local Polynomial Regression

Loess: Local Regression

- Local regression is used to model a relation between a predictor variable and response variable. To keep things simple we will consider the fixed design model. We assume a model of the form

$$Y_i = f(x_i) + \varepsilon_i$$

  where $f(x)$ is an unknown function and ε_i is an error term, representing random errors in the observations or variability from sources not included in the x_i.
- We assume the errors ε_i are i.i.d with mean 0 and finite variance $var(\varepsilon_i) = \sigma^2$.
- We make no global assumptions about the function f but assume that locally it can be well approximated with a member of a simple class of parametric function, e.g. a constant or straight line. Taylor's theorem says that any continuous function can be approximated with polynomial.

Taylor's theorem

We are going to show three forms of Taylor's theorem.

- This is the original. Suppose f is a real function on $[a,b]$, $f^{(K-1)}$ is continuous on $[a,b]$, $f^{(K)}(t)$ is bounded for $t \in (a,b)$ then for any distinct points $x_0 < x_1$ in $[a,b]$ there exists a point x between $x_0 < x < x_1$ such that

$$f(x_1) = f(x_0) + \sum_{k=1}^{K-1} \frac{f^{(k)}(x_0)}{k!}(x_1 - x_0)^k + \frac{f^{(K)}(x)}{K!}(x_1 - x_0)^K.$$

  Notice: if we view $f(x_0) + \sum_{k=1}^{K-1} \frac{f^{(k)}(x_0)}{k!}(x_1 - x_0)^k$ as function of x_1, it's a polynomial in the family of polynomials

$$\mathcal{P}_{K+1} = \{f(x) = a_0 + a_1 x + \cdots + a_K x^K, (a_0, \ldots, a_K)' \in \mathbb{R}^{K+1}\}.$$

- Statistician sometimes use what is called Young's form of Taylor's Theorem: Let f be such that $f^{(K)}(x_0)$ is bounded for x_0 then

$$f(x) = f(x_0) + \sum_{k=1}^{K} \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k + o(|x - x_0|^K), \text{ as } |x - x_0| \to 0.$$

  Notice: again the first two term of the right hand side is in $\mathcal{P}_{K+1}$.

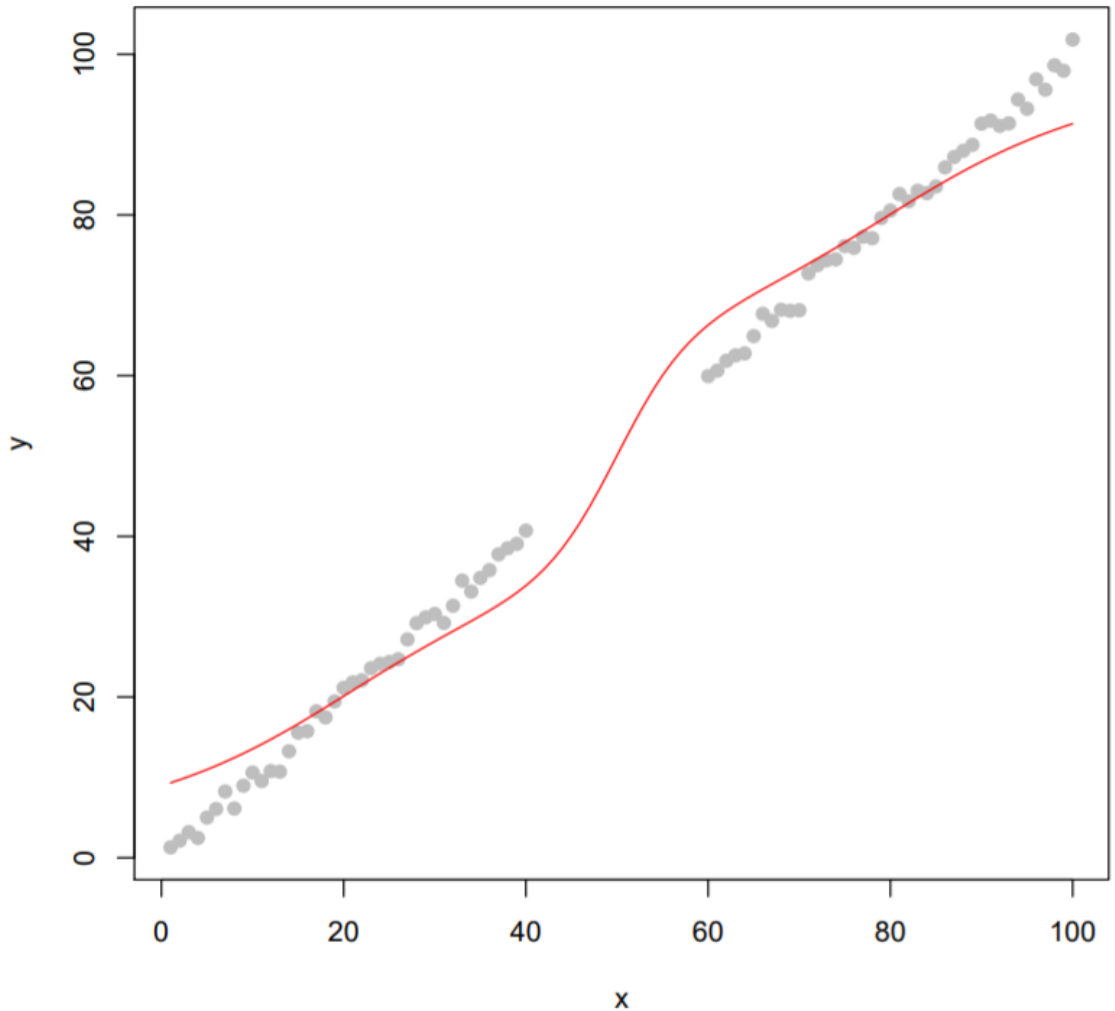
- In some of the asymptotic theory presented in this class we are going to use another refinement of Taylor's theorem called Jackson's Inequality: Suppose f is a real function on $[a, b]$ with K's continuous derivatives then

$$\min_{g \in \mathcal{P}_k} \sup_{x \in [a,b]} |g(x) - f(x)| \leq C \left(\frac{b-a}{2k} \right)^K$$

  with $\mathcal{P}_k$ the linear space of polynomials of degree k.

The problem with kernel weighted averages

Unfortunately, the Nadaraya-Watson kernel estimator suffers from bias, both at the boundaries and in the interior when the x_i's are not uniformly distributed:



Fitting local polynomials

- This arises due to the asymmetry effect of the kernel in these regions. However, we can (up to first order) eliminate this problem by fitting straight lines locally, instead of constants
- We will now define the recipe to obtain a loess smooth for a target covariate x_0.
- The first step in loess is to define a weight function (similar to the kernel K we defined for kernel smoothers). For computational and theoretical purposes we will define this weight function so that only values within a **smoothing window** $[x_0 + h(x_0), x_0 - h(x_0)]$ will be considered in the estimate of $f(x_0)$.
- Notice: In local regression $h(x_0)$ is called the span or bandwidth. It is like the kernel smoother scale parameter h. As will be seen a bit later, in local regression, the span may depend on the target covariate x_0.

- This is easily achieved by considering weight functions that are 0 outside of $[-1, 1]$. For example Tukey's tri-weight function

$$W(u) = \begin{cases} (1 - |u|^3)^3 & |u| \leq 1 \\ 0 & |u| > 1. \end{cases}$$

- The weight sequence is then easily defined by

$$w_i(x_0) = W\left(\frac{x_i - x_0}{h(x_0)}\right)$$

  We define a window by a procedure similar to the k nearest points. We want to include $\alpha \times 100\%$ of the data.

- Within the smoothing window, $f(x)$ is approximated by a polynomial. For example, a quadratic approximation

$$f(x) \approx \beta_0 + \beta_1(x - x_0) + \frac{1}{2}\beta_2(x - x_0)^2 \text{ for } x \in [x_0 - h(x_0), x_0 + h(x_0)].$$

  For continuous function, Taylor's theorem tells us something about how good an approximation this is.

- To obtain the local regression estimate $\hat{f}(x_0)$ we simply find the $\mathbf{b} = (\beta_0, \beta_1, \beta_2)'$ that minimizes

$$\hat{\mathbf{b}} = \arg\min_{\mathbf{b} \in \mathbb{R}^3} \sum_{i=1}^{n} w_i(x_0)[Y_i - \{\beta_0 + \beta_1(x_i - x_0) + \frac{1}{2}\beta_2(x_i - x_0)\}]^2$$

  and define $\hat{f}(x_0) = \hat{\beta}_0$. Notice that the Kernel smoother is a special case of local regression.

Defining the span

- In practice, it is quite common to have the x_i irregularly spaced. If we have a fixed span h then one may have local estimates based on many points and others is very few. For this reason we may want to consider a nearest neighbor strategy to define a span for each target covariate x_0.
- Define $\Delta_i(x_0) = |x_0 - x_i|$, let $\Delta_{(i)}(x_0)$ be the ordered values of such distances. One of the arguments in the local regression function **loess()** (available in the modreg library) is the `span`. A span of α means that for each local fit we want to use $\alpha \times 100\%$ of the data.
- Let q be equal to αn truncated to an integer. Then we define the span $h(x_0) = \Delta_{(q)}(x_0)$. As α increases the estimate becomes smoother.

In the following Figures 1-3 we see loess smooths for the CD4 cell count data using spans of 0.05, 0.25, 0.75, and 0.95. The smooth presented in the Figures are fitting a constant, line, and parabola respectively.

Figure 1: CD4 cell count since seroconversion for HIV infected men.

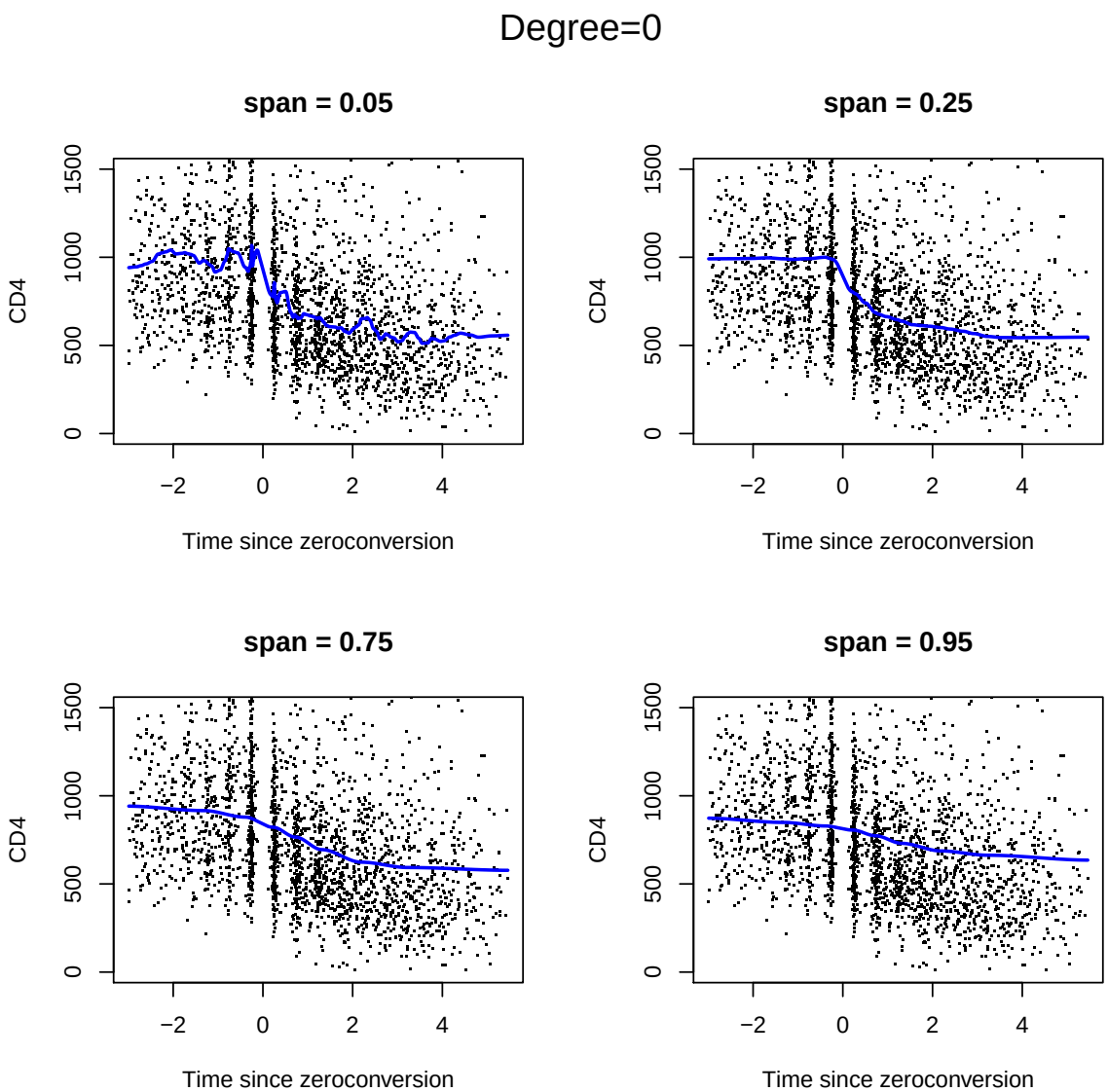


Figure 2: CD4 cell count since seroconversion for HIV infected men.

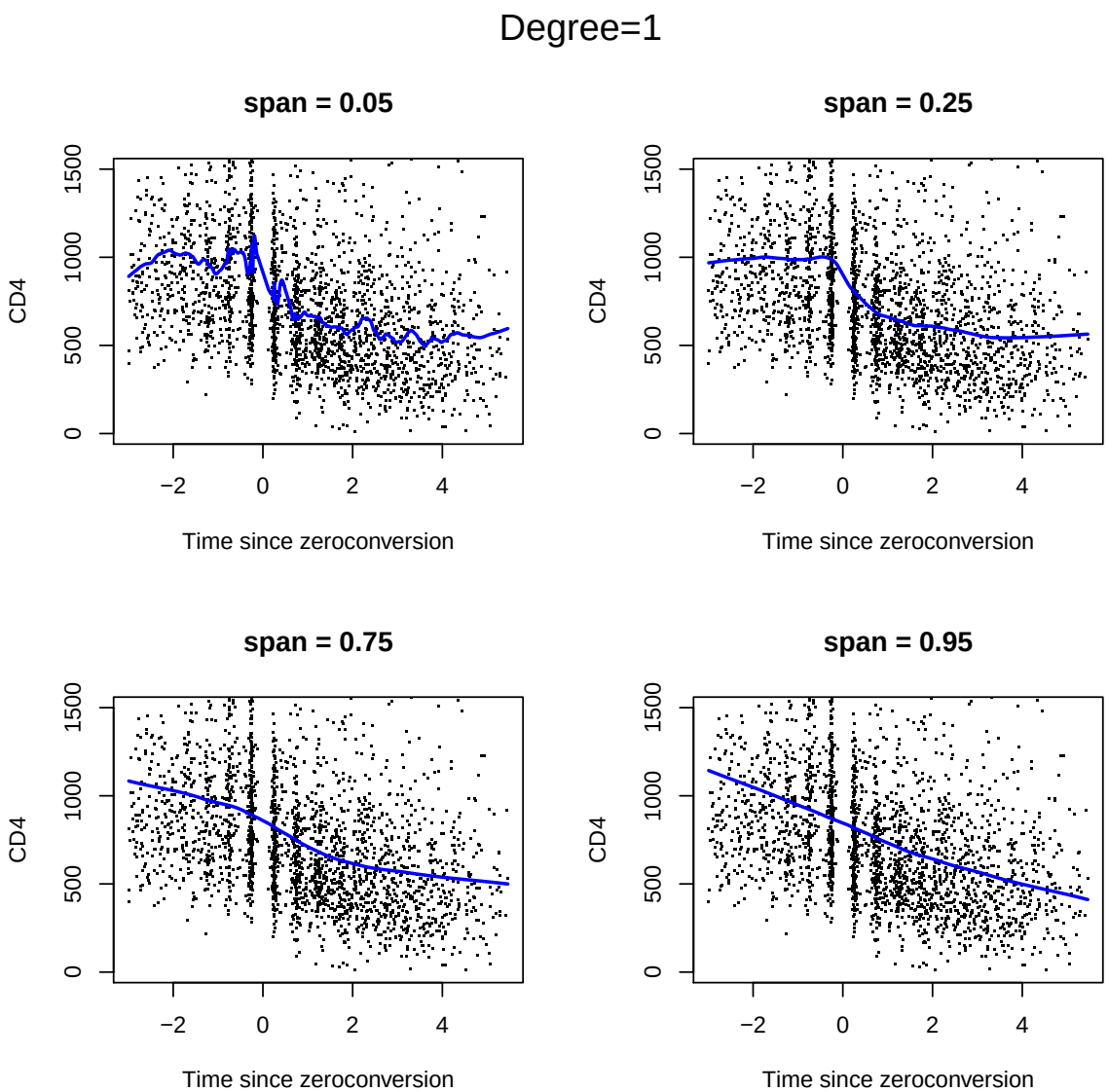
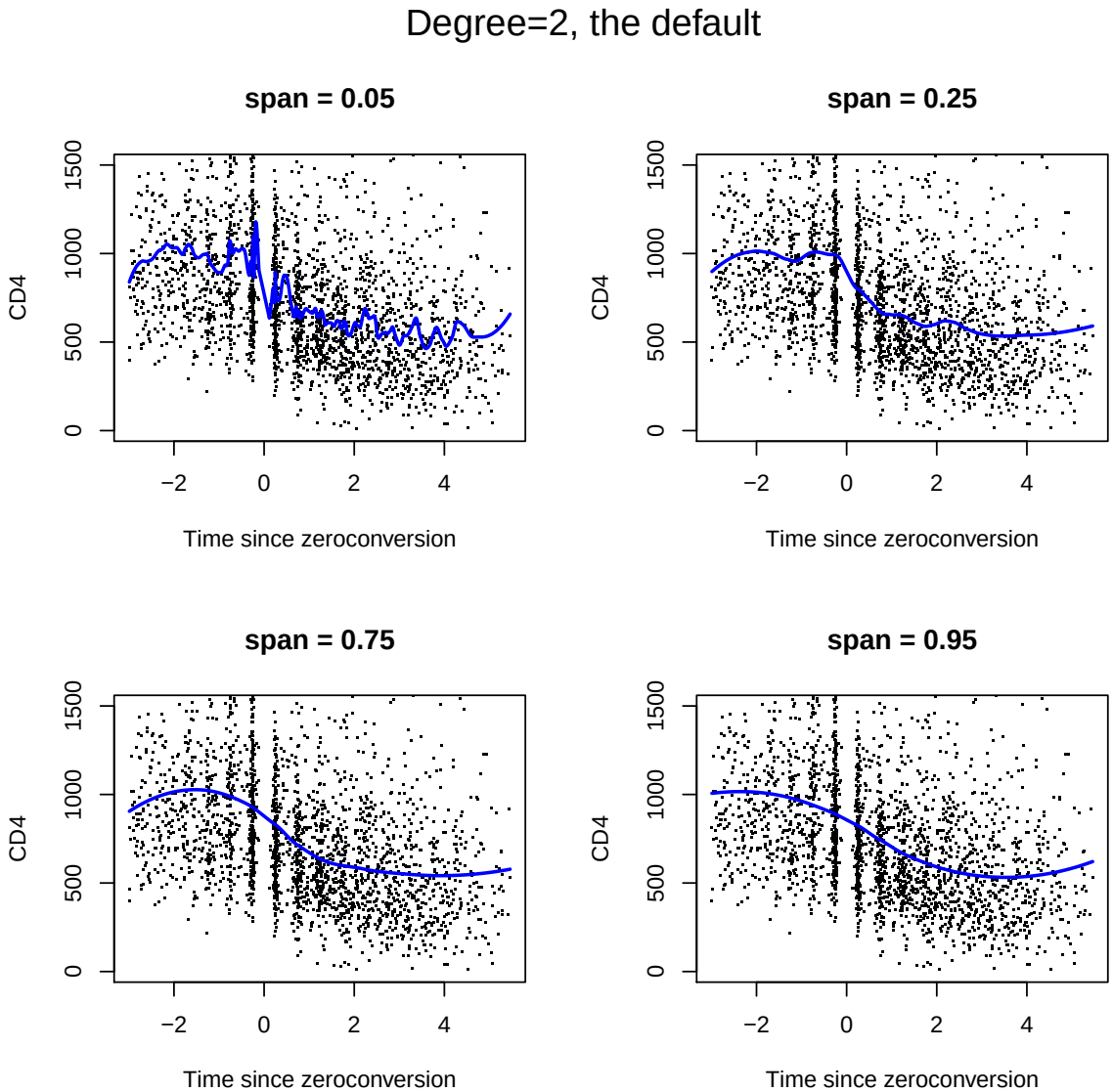


Figure 3: CD4 cell count since seroconversion for HIV infected men.



Symmetric errors and Robust fitting

- If the errors have a symmetric distribution (with long tails), or if there appears to be outliers we can use robust loess.
- We begin with the estimate described above $\hat{f}(x)$. The residuals

$$\hat{\varepsilon}_i = y_i - \hat{f}(x_i)$$

  are computed.
- Let

$$B(u;b) = \begin{cases} \{1-(u/b)^2\}^2 & |u| < b \\ 0 & |u| \geq b \end{cases}$$

  be the bisquare weight function.

- Let $m = \text{median}(|\hat{\varepsilon}_i|)$. The robust weights are

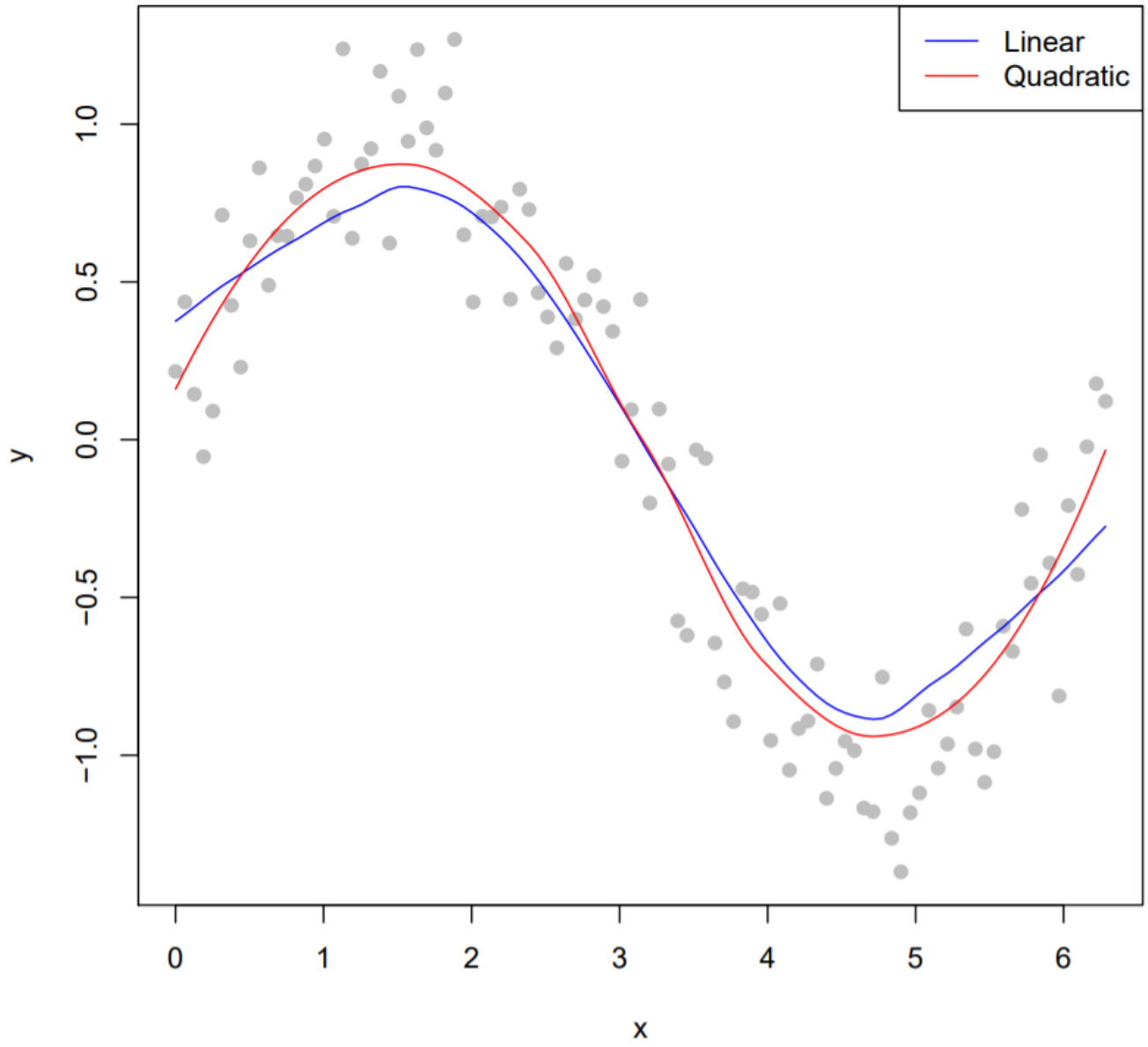
$$r_i = B(\hat{\varepsilon}_i; 6m)$$

- The local regression is repeated but with new weights $r_i w_i(x)$. The robust estimate is the result of repeating the procedure several times.
- If we believe the variance $var(\varepsilon_i) = a_i \sigma^2$ we could also use this double-weight procedure with $r_i = 1/a_i$.

- In R, local linear regression is implemented through the loess function, which uses a formula interface similar to that of other regression functions:

$$fit <- loess(spnbmd \sim age, bmd.data, span = 0.3, degree = 1)$$

- The two key options are
  - span: this is the smoothing parameter which controls the bias-variance tradeof
  - degree: this lets you specify local constant regression (the Nadaraya-Watson estimator from earlier, degree=0), local linear regression (degree=1), or local polynomial fits (degree=2)


Linear
Quadratic
y
x
1.0
0.5
0.0
−0.5
−1.0
0
1
2
3
4
5
6

- As the figure on the previous slide indicates, local linear models tend to be biased in regions of high curvature, a phenomenon referred to as "trimming the hills and filling in the valleys" Higher-order local polynomials correct for this bias, but at the expense of increased variability
- The conventional wisdom on the subject of local linear versus local quadratic fitting says that:
  - Local linear fits tend to be superior at the boundaries
  - Local quadratic fits tend to be superior in the interior
  - Local fitting to higher order polynomials is possible in principle, but rarely necessary in practice

- The other important option in span, which controls the degree of smoothing
- Unlike density, loess does not allow you to choose your own kernel; only the tricube kernel is implemented, and span refers to the proportion of the observations $\{x_i\}$ within its compact support
- Also unlike density, the kernel in loess is adaptive
- Thus, specifying $span = 0.2$ means that the bandwidth of the kernel at x_0 is made just wide enough to include 20% of the x_i values

Multivariate Local Regression

- Because Taylor's theorems also applies to multidimensional functions it is relatively straight forward to extend local regression to cases where we have more than one covariate. For example if we have a regression model for two covariates

$$Y_i = f(x_{i1}, x_{i2}) + \varepsilon_i$$

with $f(x, y)$ unknown. Around a target point $\mathbf{x}_0 = (x_{01}, x_{02})$ a local quadratic approximation is now

$$\begin{aligned} f(x_1, x_2) \approx \beta_0 &+ \beta_1(x_1 - x_{01}) + \beta_2(x_2 - x_{02}) \\ &+ \beta_3(x_1 - x_{01})(x_2 - x_{02}) + \frac{1}{2}\beta_4(x_1 - x_{01})^2 \\ &+ \frac{1}{2}\beta_5(x_2 - x_{02})^2 \end{aligned}$$

- Once we define a distance, between a point $\mathbf{x}$ and $\mathbf{x}_0$, and a span h we can define define weights as in the previous sections:

$$w_i(\mathbf{x}_0) = W\left(\frac{||\mathbf{x}_i - \mathbf{x}_0||}{h}\right).$$

- It makes sense to re-scale x_1 and x_2 so we smooth the same way in both directions. This can be done through the distance function, for example by defining a distance for the space $\mathbb{R}^d$ with

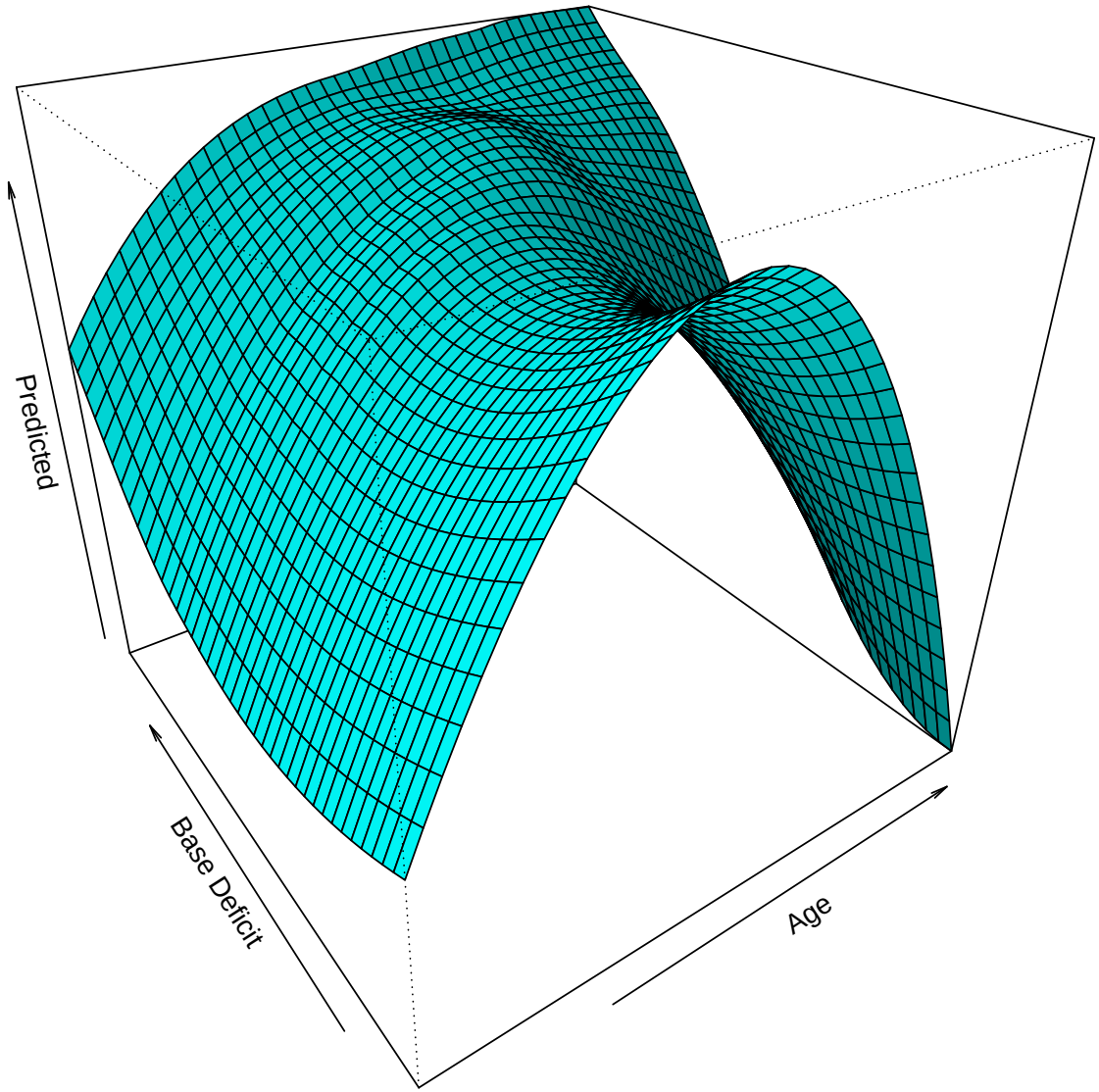
$$||\mathbf{x}||^2 = \sum_{j=1}^{d} (x_j/v_j)^2$$

with v_j a scale for dimension j. A natural choice for these v_j are the standard deviation of the covariates.

Example

- We look at part of the data obtained from a study by Socket et. al. (1987) on the factors affecting patterns of insulin-dependent diabetes mellitus in children.
- The objective was to investigate the dependence of the level of serum C-peptide on various other factors in order to understand the patterns of residual insulin secretion.
- The response measurement is the logarithm of C-peptide concentration (pmol/ml) at diagnosis, and the predictors are age and base deficit, a measure of acidity. In Figure 5 we show a loess two dimensional smooth. Notice that the effect of age is clearly non-linear.

Figure 5: Loess fit for predicting C.Peptide from Base.deficit and Age.



Local linear kernel regression

- Recall that the Nadaraya-Watson estimator of $m(\mathbf{x})$ minimizes:

$$\sum_{i=1}^{n}(y_i - m(x))^2 \mathcal{K}_H(\mathbf{x} - \mathbf{X}_i)$$

  with respect to $m(x)$.

- Stone (1977) and Cleveland (1979) suggested that instead one can minimize

$$\sum_{i=1}^{n}(y_i - m - (\mathbf{X}_i - \mathbf{x})'\beta)^2 \mathcal{K}_H(\mathbf{x} - \mathbf{X}_i)$$

  with respect to m and β.

- Let $M(\mathbf{x}) = (m(\mathbf{x}), \beta(\mathbf{x}))'$ with $\beta(\mathbf{x}) = \partial m(\mathbf{x})/\partial \mathbf{x}$, $W_x = diag[\mathcal{K}_H(\mathbf{x} - \mathbf{X}_1), \ldots, \mathcal{K}_H(\mathbf{x} - \mathbf{X}_n)]$, $Y = (Y_1, \ldots, Y_n)'$,

$$\mathbf{X}_x = \begin{pmatrix} 1 & (\mathbf{X}_1 - \mathbf{x})' \\ 1 & (\mathbf{X}_2 - \mathbf{x})' \\ & \vdots \\ 1 & (\mathbf{X}_n - \mathbf{x})' \end{pmatrix}, \mathbf{X}_x M(\mathbf{x}) = \begin{pmatrix} m(\mathbf{x}) + (\mathbf{X}_1 - \mathbf{x})'\beta(x) \\ m(\mathbf{x}) + (\mathbf{X}_2 - \mathbf{x})'\beta(x) \\ \vdots \\ m(\mathbf{x}) + (\mathbf{X}_n - \mathbf{x})'\beta(x) \end{pmatrix}$$

- Then the above least square objective function can be rewritten as

$$(Y - \mathbf{X}_x M(\mathbf{x}))' W_x (Y - \mathbf{X}_x M(\mathbf{x}))$$

- Therefore, the least square estimator of M is

$$\hat{M}(\mathbf{x}) = (\hat{m}(\mathbf{x}), \hat{\beta}(\mathbf{x}))' = (\mathbf{X}_x' W_x \mathbf{X}_x)^{-1} (\mathbf{X}_x' W_x Y)$$

Theorem

Under some regularity conditions, we have

$$\sqrt{nh_1 \cdots h_d}[\hat{m}(\mathbf{x}) - m(\mathbf{x}) - \frac{\kappa_{21}}{2} \sum_{s=1}^{d} h_s^2 m_{ss}(\mathbf{x})] \rightsquigarrow N(0, \frac{\kappa_{02}^d \sigma^2(\mathbf{x})}{f_X(\mathbf{x})})$$

$$\sqrt{nh_1 \cdots h_d} D[\hat{\beta}(\mathbf{x}) - \beta(\mathbf{x})] \rightsquigarrow N(0, I_d \kappa_{02}^{d-1} \kappa_{22} \sigma^2(\mathbf{x}) / [\kappa_{21}^2 f_X(\mathbf{x})])$$

where $D = diag(h_1, \ldots, h_d)$.

Proof

We sketch the proof here. Let $H = diag(1, h_1, \ldots, h_d)$. Since $Y_i = m(\mathbf{X}_i) + \epsilon_i$, and by the second order Taylor expansion,

$$m(\mathbf{X}_i) = (1, (\mathbf{X}_i - \mathbf{x})^T) M(\mathbf{x}) + \frac{1}{2}(\mathbf{X}_i - \mathbf{x})^T m''(\mathbf{x})(\mathbf{X}_i - \mathbf{x}) + R(\mathbf{x}, \mathbf{X}_i)$$

where $R(\mathbf{x}, \mathbf{X}_i)$ is the remainder, we have

$$\begin{aligned}
\hat{M}(\mathbf{x}) - M(\mathbf{x}) &= (\mathbf{X}_x^T W_x \mathbf{X}_x)^{-1} (\mathbf{X}_x^T W_x [Y - \mathbf{X}_x M(\mathbf{x})]) \\
&= \Big[\sum_{i=1}^{n} \mathcal{K}_H(\mathbf{X}_i - \mathbf{x}) \begin{pmatrix} 1 \\ \mathbf{X}_i - \mathbf{x} \end{pmatrix} (1, (\mathbf{X}_i - \mathbf{x})^T) \Big]^{-1} \\
&\times \sum_{i=1}^{n} \mathcal{K}_H(\mathbf{X}_i - \mathbf{x}) \begin{pmatrix} 1 \\ \mathbf{X}_i - \mathbf{x} \end{pmatrix} \Big\{ \frac{1}{2}(\mathbf{X}_i - \mathbf{x})^T m''(\mathbf{x})(\mathbf{X}_i - \mathbf{x}) \\
&+ \epsilon_i + R(\mathbf{x}, \mathbf{X}_i) \Big\}
\end{aligned}$$

Then

$$\sqrt{nh_1 \cdots h_d} H(\hat{M}(\mathbf{x}) - M(\mathbf{x})) = S_n^{-1}[B_n(\mathbf{x}) + V_n(\mathbf{x}) + R_n(\mathbf{x})],$$

where

$$S_n(\mathbf{x}) = \frac{1}{nh_1 \cdots h_d} \sum_{i=1}^{n} \mathcal{K}_H(\mathbf{X}_i - \mathbf{x}) H^{-1} \begin{pmatrix} 1 \\ \mathbf{X}_i - \mathbf{x} \end{pmatrix} (1, (\mathbf{X}_i - \mathbf{x})^T) H^{-1},$$

$$B_n(\mathbf{x}) = \frac{1}{\sqrt{nh_1 \cdots h_d}} \sum_{i=1}^{n} \mathcal{K}_H(\mathbf{X}_i - \mathbf{x}) H^{-1} \begin{pmatrix} 1 \\ \mathbf{X}_i - \mathbf{x} \end{pmatrix} \times \frac{1}{2} (\mathbf{X}_i - \mathbf{x})^T m''(\mathbf{x}) (\mathbf{X}_i - \mathbf{x}),$$

$$V_n(\mathbf{x}) = \frac{1}{\sqrt{nh_1 \cdots h_d}} \sum_{i=1}^{n} \mathcal{K}_H(\mathbf{X}_i - \mathbf{x}) H^{-1} \begin{pmatrix} 1 \\ \mathbf{X}_i - \mathbf{x} \end{pmatrix} \epsilon_i$$

$$R_n(\mathbf{x}) = \frac{1}{\sqrt{nh_1 \cdots h_d}} \sum_{i=1}^{n} \mathcal{K}_H(\mathbf{X}_i - \mathbf{x}) H^{-1} \begin{pmatrix} 1 \\ \mathbf{X}_i - \mathbf{x} \end{pmatrix} R(\mathbf{x}, \mathbf{X}_i).$$

By the law of large numbers,

$$S_n(\mathbf{x}) \to \begin{pmatrix} 1 & 0 \\ 0 & I_d \kappa_{21} \end{pmatrix} \equiv S(\mathbf{x}).$$

By the Chebyshev's inequality and the Liapounov CLT, we can show

$$B_n(\mathbf{x}) = \sqrt{nh_1 \cdots h_d} \begin{pmatrix} \frac{\kappa_{21} f_X(\mathbf{x})}{2} \sum_{s=1}^{d} h_s^2 m_{ss}(\mathbf{x}) \\ 0 \end{pmatrix} + o_p(1),$$

and

$$\begin{aligned} V_n(\mathbf{x}) &= \frac{1}{\sqrt{nh_1 \cdots h_d}} \sum_{i=1}^{n} \mathcal{K}_H(\mathbf{X}_i - \mathbf{x}) H^{-1} \begin{pmatrix} 1 \\ \mathbf{X}_i - \mathbf{x} \end{pmatrix} \epsilon_i \\ &\rightsquigarrow N\Big(0, \begin{pmatrix} \kappa_{02}^d & 0 \\ 0 & I_d \kappa_{02}^{d-1} \kappa_{22} \end{pmatrix} \sigma^2(\mathbf{x}) f_X(\mathbf{x})\Big). \end{aligned}$$

Noting that $R_n(\mathbf{x})$ is a smaller order term than $B_n(\mathbf{x})$, the conclusion follows from above equations.

Note:

- To estimate $m(\mathbf{x})$ consistently, we only require that $h \to 0$ and $nh_1 \cdots h_d \to \infty$. Nevertheless, for consistent estimation of the derivative of $m(\mathbf{x})$, we need stronger condition: $nh_1 \cdots h_d \sum_{s=1}^{d} h_s^2 \to \infty$.
- The above theorem also says that the convergence rates of $\hat{m}(\mathbf{x})$ and $\hat{\beta}(\mathbf{x})$ are different. This reflects the fact that it is more difficult to estimate the derivatives of a regression function than itself. The above theorem goes through under some weak data dependence conditions. See Masry (1996a, b) for the proof in this case.
- When $m(\mathbf{x}) = \alpha + \beta'\mathbf{x}$, the bias for local linear estimator vanishes so that the local linear estimator becomes unbiased. We could allow $h_s = \infty (s = 1, \ldots, d)$, and it is easy to show in this case that the local linear estimator collapses $\tilde{m}(\mathbf{x}) = \tilde{\alpha} + \tilde{\beta}'\mathbf{x}$, where $\tilde{\alpha}$ and $\tilde{\beta}$ are the OLS estimators of α and β, respectively.

Theorem

Under certain regularity conditions, we have

$$\sup_{x\in\mathcal{S}}|\hat{m}(\mathbf{x})-m(\mathbf{x})| = o\Big(\frac{1}{\sqrt{nh_1\cdots h_d}}\sqrt{lnn}\Big)+o\Big(\sum_{s=1}^{d}h_s^2\Big) \quad a.s.$$

where $\mathcal{S}$ is a compact set on $\mathcal{R}^d$ contained in the support of f_X.

See Masry (1996a,b) for the proof of the above theorem.

Least Squares Cross Validation

Let $\hat{m}_{-i}(\mathbf{X}_i)$ denote the leave-one-out local linear estimator of $m(\mathbf{x})$ at $\mathbf{x} = \mathbf{X}_i$ by using all observations but $(\mathbf{X}_i, Y_i)$. That is, $\hat{M}_{-i}(\mathbf{X}_i) = (\hat{m}_{-i}(\mathbf{X}_i), \hat{\beta}_{-i}(\mathbf{X}_i)^T)^T$ solves the following minimization problem

$$\min_{m,\beta} \sum_{j=1, j\neq i}^{n} [Y_j - m - (\mathbf{X}_j - \mathbf{X}_i)^T \beta]^2 \mathcal{K}_H(\mathbf{X}_j - \mathbf{X}_i).$$

Then the local linear cross-validation approach towards bandwidth selection chooses $h = (h_1, \ldots, h_d)$ to minimize

$$CV_u(h) = \sum_{i=1}^{n} [Y_i - \hat{m}_{-i}(\mathbf{X}_i)]^2 w(\mathbf{X}_i)$$

where w is a weight function.

Given a suitably chosen weighting function $w(\mathbf{x})$, let h_{s0} be the smoothing parameter that minimizes

$$AMISE(h) = \int [\frac{\kappa_{21}}{2} \sum_{s=1}^{d} h_s^2 m_{ss}(\mathbf{x})]^2 w(\mathbf{x}) f_X(\mathbf{x}) d\mathbf{x} + \frac{\kappa_{02}^d \int \sigma^2(\mathbf{x}) w(\mathbf{x}) d\mathbf{x}}{n h_1 \cdots h_d}$$

One can show that

$$h_{s0} = a_s n^{-1/(4+d)}$$

for $s = 1, \ldots, d$. where a_s depends on the unknown function m and its second order derivatives, the density function f_X, the kernel function K and the weighting function w.

We can show that

$$CV_u(h) \cong \int E[\hat{m}(\mathbf{x}) - m(\mathbf{x})]^2 w(\mathbf{x}) f_X(\mathbf{x}) d\mathbf{x}$$

$$= AMISE(h) + o(\sum_{s=1}^{d} h_s^4 + \frac{1}{nh_1 \dots h_d}).$$

Thus minimize $CV_u(h)$ with respect to $h = (h_1, \dots, h_d)$ is equivalent to minimizing $AMISE(h)$. Li and Racine (2003) show that

$$\frac{\hat{h}_s}{h_{s0}} \to 1$$

in probability for $s = 1, \dots, d$. That is, the local linear cross-validation smoothing parameters converge to the AMISE-optimal smoothing parameters. Also, This implies that the rate of convergence of the resulting local linear estimator is the same as the local constant cross-validation case.

Local Polynomial Regression

- A natural extension of the local linear kernel estimator is to fit higher degree of polynomials locally. For notational simplicity, we only consider the univariate case.
- The generalization to multivariate case is straightforward but demands some complicated notation.
- When x is a scalar, a pth order local polynomial kernel estimator is based on the following minimization problem:

$$\min_{\beta_0,\beta_1,\ldots,\beta_p} \sum_{i=1}^{n} [Y_i - \beta_0 - \beta_1(X_i - x) - \cdots - \beta_p(X_i - x)^p]^2 K_h(X_i - x).$$

- Let $\hat{\beta} = (\hat{\beta}_0, \ldots, \hat{\beta}_p)$ denote the values of $\beta_0, \ldots, \beta_p$ that minimize the above problem. Then $\hat{\beta}_0$ estimates $m(x)$, and $s!\hat{\beta}_s$ estimates $m^{(s)}(x)$, the sth derivative of $m(x)$ for $s = 1, \ldots, d$.

- The previous minimizing problem is a standard weighted least squares regression problem. Let $W_x = diag(K_h(X_1 - x), \ldots, K_h(X_n - x))$. Define,

$$Y = \begin{pmatrix} Y_1 \\ \vdots \\ Y_n \end{pmatrix}, \mathbf{X}_s = \begin{pmatrix} 1 & X_1 - x & \cdots & (X_1 - x)^p \\ \vdots & \vdots & & \vdots \\ 1 & X_n - x & \cdots & (X_n - x)^p \end{pmatrix}$$

- Assuming the asymptotic invertibility of $\mathbf{X}_x^T W_x \mathbf{X}_s$, then

$$\hat{\beta} = (\mathbf{X}_x^T W_x \mathbf{X}_x)^{-1} \mathbf{X}_x^T W_x Y$$

- Let $\hat{m}(x; p, h) = \hat{\beta}_0 = e_1^T \hat{\beta}$, where e_1 is the $(p+1) \times 1$ vector with 1 in the first entry and 0 elsewhere.

- Simple explicit formulae exist for the local constant estimator ($p = 0$):

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

- and the local linear estimator ($p = 1$) :

$$\hat{m}(x; 1, h) = \frac{1}{n} \sum_{i=1}^{n} \frac{[s_2(x; h) - s_1(x; h)(X_i - x)] K_h(X_i - x) Y_i}{s_2(x; h) s_0(x; h) - s_1(x; h)^2}$$

  where $s_r(x; h) = \frac{1}{n} \sum_{i=1}^{n} (X_i - x)^r K_h(X_i - x)$.

- Ruppert and Wand (1994) study the leading conditional bias and conditional variance of the above estimator. For brevity, we report their results directly here, even though the derivation is not much involved.

Following Ruppert and Wand (1994), we make the following assumptions:

- The $(X_i, Y_i), i = 1, \ldots, n$ are i.i.d.
- $m(x)$ are $(p+1)$th continuously differentiable at x for odd p or $(p+2)$th continuously differentiable at x for even p. $\sigma^2(x)$ is continuous at x. x is an interior point on the support of f_X. $f_X > 0$.
- The kernel K is symmetric about zero and has compact support on [-1,1].
- As $n \to \infty, h \to 0$ and $nh \to \infty$.

Let $\mu_l(K) = \int_{-1}^{1} z^l K(z) dz$ and N_p be the $(p+1) \times (p+1)$ matrix with (i, j) entry equal to $\mu_{i+j-2}(K)$. Let $M_p(u)$ be the same as N_p but with the first column replaced by $(1, u, \ldots, u^p)'$.

Then the kernel

$$K_{(p)}(u) = \frac{|M_p(u)|}{|N_p|} K(u)$$

is a $(p+1)$th order kernel when p is odd and $(p+2)$th order kernel when p is even.

- For odd p the conditional bias of $\hat{m}(x; p, h)$ is

$$\begin{aligned} &E[\hat{m}(x; p, h) - m(x) | X_1, \ldots, X_n] \\ &= \frac{1}{(p+1)!} h^{p+1} m^{(p+1)}(x) \mu_{p+1}(K_{(p)}) + o_p(h^{p+1}). \end{aligned}$$

- whereas for even p,

$$\begin{aligned} &E[\hat{m}(x; p, h) - m(x) | X_1, \ldots, X_n] \\ &= h^{p+1} \left[\frac{m^{(p+1)}(x) f_X'(x)}{(p+1)! f_X(x)} + \frac{m^{(p+2)}(x)}{(p+2)!} \right] \mu_{p+2}(K_{(p)}) + o_p(h^{p+2}). \end{aligned}$$

- In either case,

$$Var(\hat{m}(x;p,h)|X_1,\ldots,X_n) = \frac{\int K_{(p)}^2(u)du\sigma^2(x)}{nhf_X(x)} + o_p(\frac{1}{nh})$$

Remarks

- First, the degree of local polynomial being fitted determines the order of the bias of $\hat{m}(x;p,h)$.
- Second, a practical problem is the choice of p. Odd degree local polynomial fits have attractive bias and boundary properties (see Wand and Jones, 1995, pp.126-130). These facts suggest the use of either $p = 1$ or $p = 3$ in practice.
- Third, when f is compactly supported, the bias and variance formulas for the local polynomial estimators are quite different for points around the boundary even though the order of bias and variance remains the same as the interior points.

The locfit function

- The basic syntax of model fitting is as follows:

$$fit \leftarrow locfit(spnbmd \sim lp(age, nn = .7, deg = 2))$$

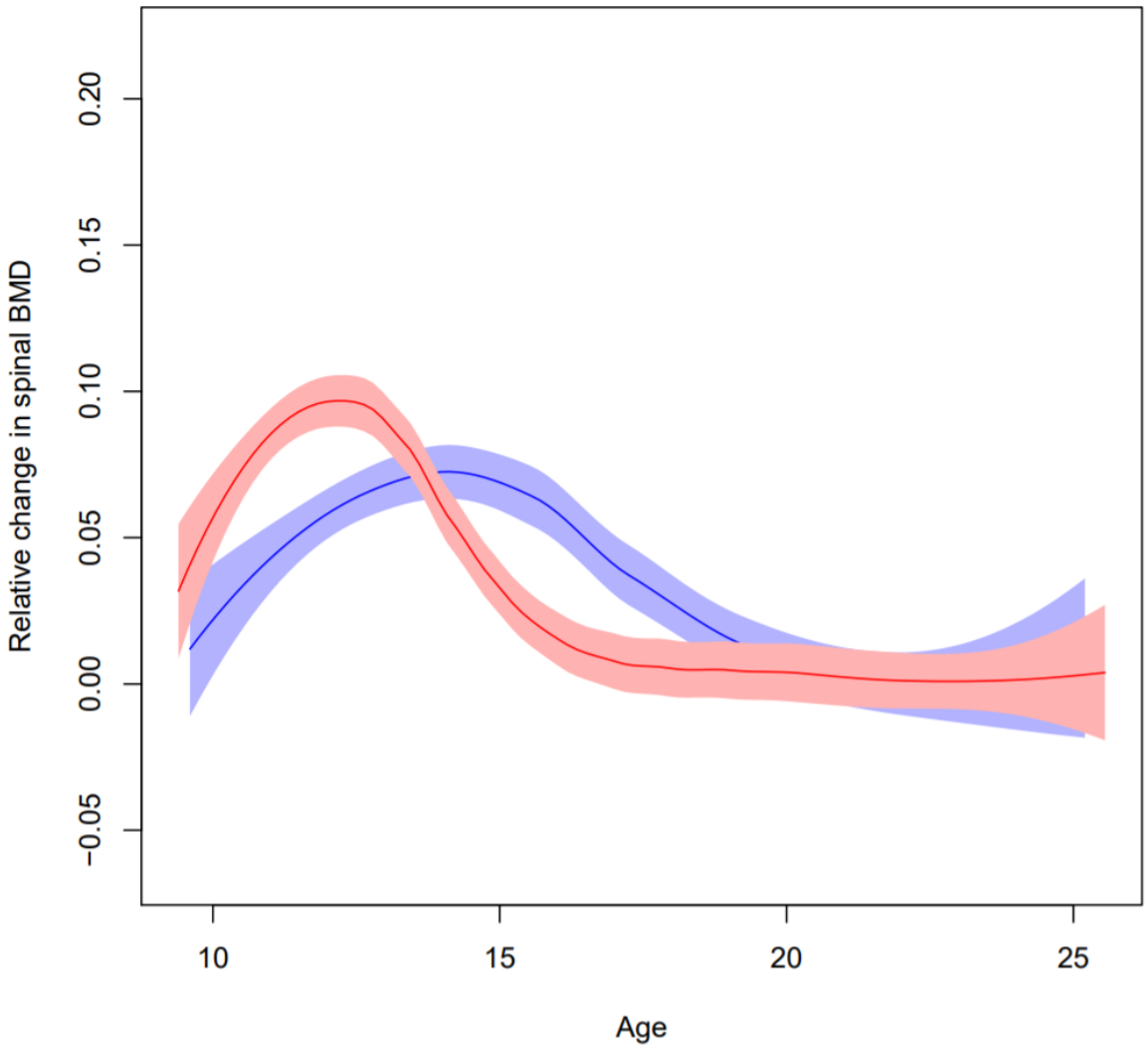
  where lp controls the local polynomial which is fit to the data
- Just like loess, there is a nn parameter (analogous to span), which adaptively determines the bandwidth by setting the number of points in the neighborhood of x0 equal to nn
- There is also a deg parameter, which controls the degree of the local polynomial (like loess, the default is 2)

Confidence intervals in locfit

- The locfit package is very well-developed, and we cannot possibly cover all of its features here
- However, two very important features are its ability to construct pointwise and simultaneous confidence bands:

```
predict(fit,newdata=seq(9,25,len=75),se=TRUE)
scb(spnbmd~lp(age))  # Simultaneous conf. bands
plot(fit,band="global")  # Plot the band
```

where the first line of code returns the value of $\hat{f}(x_0)$ and its standard error for a supplied list of points newdata


Relative change in spinal BMD
0.20
0.15
0.10
0.05
0.00
−0.05
10
15
20
25
Age