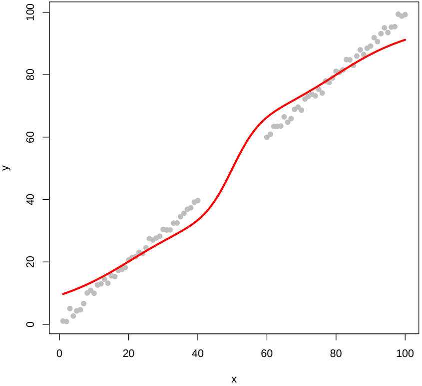
Local regression

Patrick Breheny

October 25

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:



Loess

- 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
- In locally weighted regression, also known as *lowess* or *loess*, we solve a separate weighted least squares problem at each target point x_0:

$$(\hat{\alpha}, \hat{\beta}) = \text{arg min}_{\alpha,\beta} \sum_i K_\lambda(x_0, x_i)(y_i - \alpha - x_i\beta)^2$$

- The estimate is then $\hat{f}(x_0) = \hat{\alpha} + x_0\hat{\beta}$

Loess is a linear smoother

- Let $\mathbf{X}$ denote the $n \times 2$ design matrix with ith row $(1, x_i)$, and $\mathbf{W}$ denote the $n \times n$ diagonal matrix with ith diagonal element $w_i(x_0) = K_\lambda(x_0, x_i)$
- Then

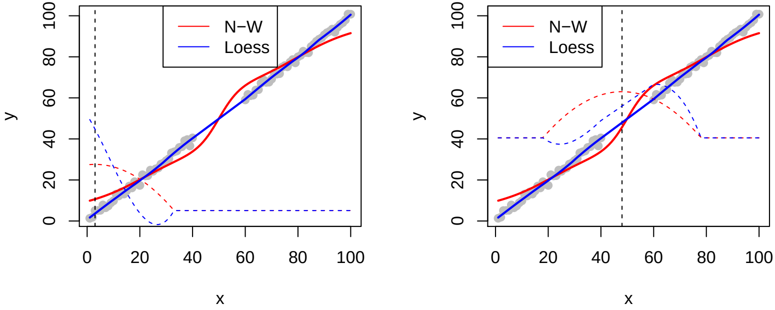
$$\begin{aligned} \hat{f}(x_0) &= \mathbf{x}_i^T [\mathbf{X}'\mathbf{W}\mathbf{X}]^{-1}\mathbf{X}'\mathbf{W}\mathbf{y} \\ &= \sum_i l_i(x_0) y_i, \end{aligned}$$

  where it is important to keep in mind that $\mathbf{W}$ depends implicitly on x_0
- Note that loess is therefore a linear smoother, in the sense that $\hat{\mathbf{y}} = \mathbf{S}\mathbf{y}$

Effective kernels

- Furthermore, the linear smoother in local linear regression is performing weighted local averaging with the weights, determined by $\{l_i(x_0)\}$, forming an *effective kernel* (also called the *equivalent kernel*)
- Before the development of loess, a fair amount of research focused on deriving adaptive modifications to kernels in order to alleviate the bias that we previously discussed
- However, local linear regression automatically modifies the kernel in such a way that this bias is largely eliminated, a phenomenon known as *automatic kernel carpentry*

Automatic kernel carpentry



The smoothing matrix

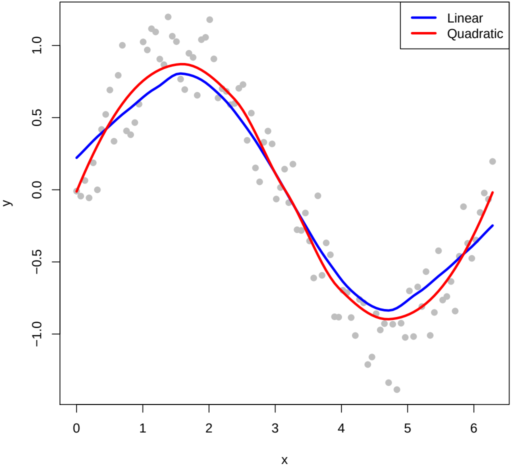
- Furthermore, since loess is a linear smoother, it is easy to carry out cross-validation and generalized cross-validation:

$$CV = \frac{1}{n}\sum_i \left(\frac{y_i - \hat{y}_i}{1 - l_i(x_i)}\right)^2$$

$$GCV = \frac{1}{n}\sum_i \left(\frac{y_i - \hat{y}_i}{1 - tr(\mathbf{S})/n}\right)^2,$$

  where, as we have seen before, we can obtain leave-one-out cross-validation results without refitting our model
- Also as before, $tr(\mathbf{S})$ acts as the effective degrees of freedom

Bias due to local linear fitting



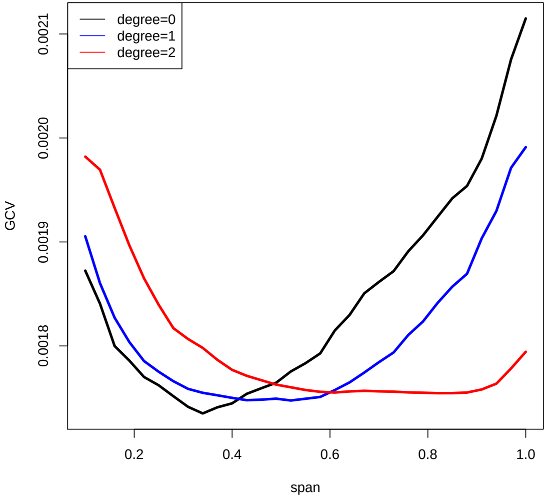
Local linear versus local quadratic fitting

- 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

Constant vs. adaptive λ

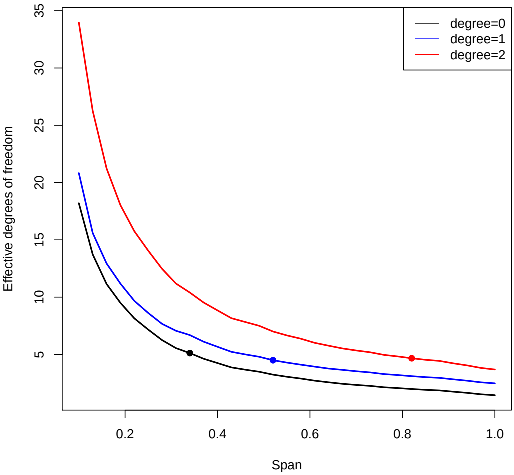
- Our discussion of kernels in the previous lecture featured a constant half-width λ
- An alternative approach, and the one used by default in both SAS and R, is to allow λ to change with $\mathbf{x}_0$ so that the number of points inside $(x_0 - \lambda, x_0 + \lambda)$ remains constant
- The smoothing parameter in loess is therefore readily interpretable: it is the fraction of the sample size used in constructing the local fit at any point x_0

Selection of smoothing parameter

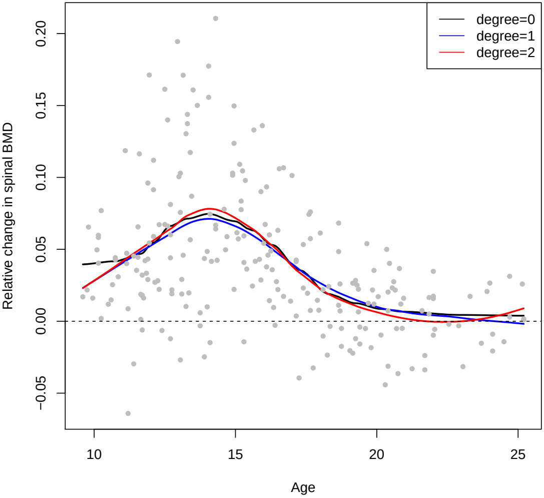


Effective degrees of freedom versus span

Dots indicate optimal smoothing, as chosen by GCV:



Optimal fits for the bone mineral density data



Bone mineral density data – males versus females