Introduction to Statistical Machine Learning with Applications in Econometrics

Lecture 7: Moving Beyond Linearity (ISL ch. 7)

Instructor: Ma, Jun

Renmin University of China

November 16, 2022

Moving beyond linearity

- The linearity assumption in the regression model is almost always an approximation.
- If linear approximation to the true function f in the model $Y = f(X) + \epsilon$ is poor, i.e., $\min_b \mathrm{E}\left[(f(X) - X^\top b)^2\right]$ is large, test error could be large, due to large bias.
- We relax the linearity assumption:
  - Polynomial regression;
  - Step functions;
  - Regression splines;
  - Smoothing splines;
  - Generalized additive models.

Polynomial regression

- Assume that $X \in \mathbb{R}$. We replace the linear model $Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$ by a polynomial regression model:

$$Y_i = \beta_0 + \beta_1 X_i + \beta_2 X_i^2 + \cdots + \beta_d X_i^d + \epsilon_i.$$

- The coefficients can be easily estimated by least squares.
- Linear logistic regression can be extended:

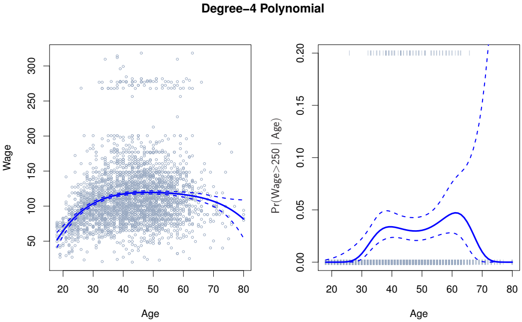
$$\Pr(Y_i = 1 \mid X_i) = \frac{\exp\left(\beta_0 + \beta_1 X_i + \beta_2 X_i^2 + \cdots + \beta_d X_i^d\right)}{1 + \exp\left(\beta_0 + \beta_1 X_i + \beta_2 X_i^2 + \cdots + \beta_d X_i^d\right)}.$$

- The fitted function values at any value x_0:

$$\hat{f}(x_0) = \hat{\beta}_0 + \hat{\beta}_1 x_0 + \hat{\beta}_2 x_0^2 + \cdots + \hat{\beta}_d x_0^4.$$

- $\hat{f}(x_0)$ is a linear function of $\hat{\beta}_0, \hat{\beta}_1, ..., \hat{\beta}_d$. We can get a simple expression for pointwise-variances $\text{Var}\left[\hat{f}(x_0)\right]$ at any value x_0.
- Pointwise standard errors $\text{SE}\left[\hat{f}(x_0)\right]$: estimate of $\sqrt{\text{Var}\left[\hat{f}(x_0)\right]}$.
- Confidence interval: $\left[\hat{f}(x_0) - 2 \cdot \text{SE}\left[\hat{f}(x_0)\right], \hat{f}(x_0) + 2 \cdot \text{SE}\left[\hat{f}(x_0)\right]\right]$.
- We either fix the degree d at some reasonably low value or use cross-validation to choose d.

The `Wage` data



ISL Figure 7.1

- Left: degree-4 polynomial regression of `wage` on `age` (solid blue); an estimated 95 % confidence interval (dashed).
- Right: fitted posterior probability of `wage` > `250` using logistic regression, with a degree-4 polynomial (solid blue); an estimated 95 % confidence interval (dashed).
- For `age` that is close to the boundaries, the prediction of `wage` is highly variable.

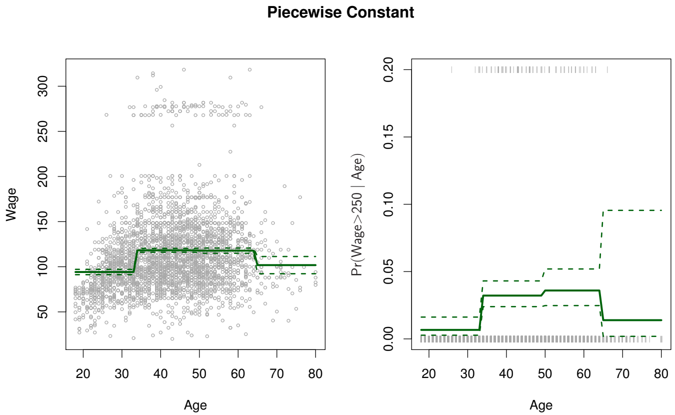
Step functions

- Polynomial regression imposes a global structure on $f(X)$. Step functions avoid imposing such a global structure.
- $c_1, c_2, ..., c_K$: cutpoints in the range of X, then

$$\begin{aligned}
C_0(X) &= 1(X < c_1) \\
C_1(X) &= 1(c_1 \leq X < c_2) \\
C_2(X) &= 1(c_2 \leq X < c_3) \\
\vdots &\quad \vdots \quad \vdots \\
C_{K-1}(X) &= 1(c_{K-1} \leq X < c_K) \\
C_K(X) &= 1(c_K \leq X).
\end{aligned}$$

- We can use least squares to fit a linear model:

$$Y_i = \beta_0 + \beta_1 C_1(X_i) + \beta_2 C_2(X_i) + \cdots + \beta_d C_K(X_i) + \epsilon_i.$$



ISL Figure 7.2

- Use $C_1(X) = 1(X < 35)$, $C_2(X) = 1(35 \leq X < 65)$, $C_3(X) = 1(X \geq 65)$.
- The fitted curve is discontinuous: the predicted wage for X being slightly less than 35 and the predicted wage for X being slightly greater than 35 can be very different.

Piecewise polynomials

- Instead of a single polynomial, we can use different polynomials in regions defined by knots:

$$Y_i = \begin{cases} \beta_{01} + \beta_{11} X_i + \beta_{21} X_i^2 + \beta_{31} X_i^3 + \epsilon_i & \text{if } X_i < c \\ \beta_{02} + \beta_{12} X_i + \beta_{22} X_i^2 + \beta_{32} X_i^3 + \epsilon_i & \text{if } X_i \geq c. \end{cases}$$

- Impose continuity constraint:

$$\beta_{01} + \beta_{11} c + \beta_{21} c^2 + \beta_{31} c^3 = \beta_{02} + \beta_{12} c + \beta_{22} c^2 + \beta_{32} c^3.$$

- Impose continuity constraint on the first derivative:

$$\beta_{11} + 2\beta_{21} c + 3\beta_{31} c^2 = \beta_{12} + 2\beta_{22} c + 3\beta_{32} c^2.$$

Linear splines

- A linear spline with knots at ξ_k, $k = 1, ..., K$ is a piecewise linear polynomial continuous at each knot.
- We can represent this model as

$$Y_i = \beta_0 + \beta_1 b_1 (X_i) + \beta_2 b_2 (X_i) + \cdots + \beta_{K+1} b_{K+1} (X_i) + \epsilon_i,$$

where b_k are basis functions,

$$\begin{aligned} b_1 (X_i) &= X_i \\ b_{k+1} (X_i) &= (X_i - \xi_k)_+ , k = 1, ..., K, \end{aligned}$$

and

$$(X_i - \xi_k)_+ = \begin{cases} X_i - \xi_k & \text{if } x_i > \xi_k \\ 0 & \text{otherwise .} \end{cases}$$

Cubic splines

- A cubic spline with knots at ξ_k, $k = 1, ..., K$ is a piecewise cubic polynomial with continuous derivatives up to order 2 at each knot.
- Represent this model with truncated power basis functions

$$Y_i = \beta_0 + \beta_1 b_1(X_i) + \beta_2 b_2(X_i) + \cdots + \beta_{K+3} b_{K+3}(X_i) + \epsilon_i,$$

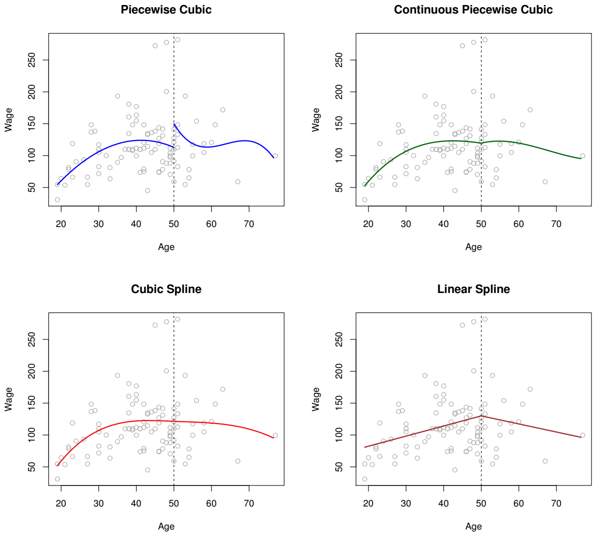
where

$$\begin{aligned} b_1(X_i) &= X_i \\ b_2(X_i) &= X_i^2 \\ b_3(X_i) &= X_i^3 \\ b_{k+3}(X_i) &= (X_i - \xi_k)_+^3, k = 1, ..., K, \end{aligned}$$

and

$$(X_i - \xi_k)_+ = \begin{cases} (X_i - \xi_k)^3 & \text{if } x_i > \xi_k \\ 0 & \text{otherwise .} \end{cases}$$

- A cubic spline with K knots has $K + 4$ parameters or degrees of freedom.

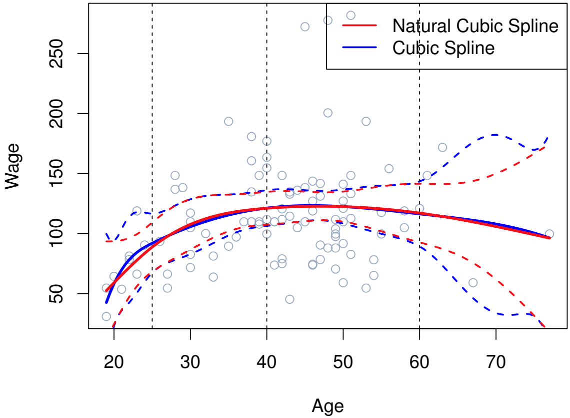


ISL Figure 7.3

- Most human beings are unable to distinguish between a cubic spline and a smooth (infinitely differentiable) function.

Natural spline

- Splines can have high variance when X takes on either a very small or very large value.
- A natural spline is a regression spline with additional 2×2 boundary constraints: the function is required to be linear at the boundary (in the region where X is smaller than the smallest knot, or larger than the largest knot).
- Natural splines generally produce more stable estimates at the boundaries.
- A natural spline with K knots has K degrees of freedom.



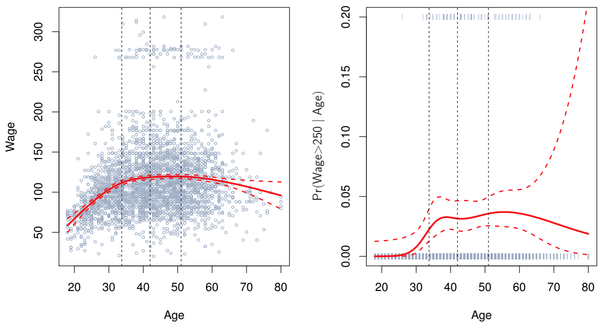
ISL Figure 7.4

- A cubic spline and a natural cubic spline, with three knots. The dashed lines denote the knot locations.
- Narrower confidence intervals reflect lower variances.

Choosing the number and locations of the knots

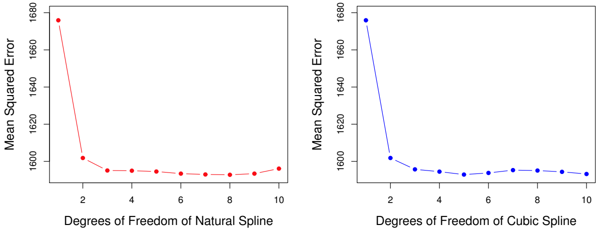
- The regression spline is most flexible in regions that contain a lot of knots, because in those regions the polynomial coefficients can change rapidly.
- One strategy is to place more knots in places where we feel $f(X)$ might vary most rapidly, and to place fewer knots where it seems more stable.
- Another (more objective) strategy is to decide K by cross-validation, the number of knots, and then place them at appropriate quantiles of X.
  - Locations of the knots are estimated. Cross-validation should take such uncertainty into account.
  - Randomly split data into training set Tr and test set Te. Let $\xi_1, \xi_2, ..., \xi_K$ be the knots estimated from the full data, as sample quantiles of X.
    - Wrong way: Use Tr to train the model with knots at $\xi_1, \xi_2, ..., \xi_K$ and then use Te to estimate the test error for the K-knot model.
    - Right way: Use Tr to generate a different list of knots $\tilde{\xi}_1, \tilde{\xi}_2, ..., \tilde{\xi}_K$, which are sample quantiles of X using only data in Tr and then use Te to estimate the test error.

Natural Cubic Spline



ISL Figure 7.5

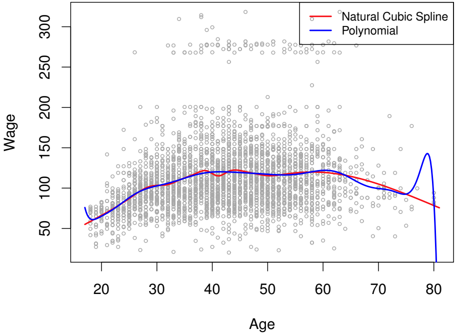
- Fit a natural cubic spline with three knots. The knot locations were chosen automatically as the 25th, 50th, and 75th percentiles of `age`.



ISL Figure 7.6

- Ten-fold cross-validated mean squared errors for splines with various degrees of freedom fit to the `Wage` data.

Comparison to polynomial regression



ISL Figure 7.7

- Comparison of a polynomial and a natural cubic spline with the same degree of freedom.
- Polynomial produces undesirable results at the boundaries, while the natural cubic spline still provides a reasonable fit.

Smoothing splines

- Consider the problem:

$$\min_{g \in \{\text{all functions}\}} \sum_{i=1}^{n} \left(Y_i - g\left(X_i\right)\right)^2 .$$

  If we don't put any constraints on g, then we can always make $RSS = \sum_{i=1}^{n} \left(Y_i - g\left(X_i\right)\right)^2$ zero simply by choosing g such that it interpolates all of the Y_i.
- Such a function would be far too flexible and definitely overfit the data.
- What we really want is a function g that makes RSS small, but that is also smooth.

- The minimizer is known as a smoothing spline:

$$\min_{g \in \mathcal{S}} \sum_{i=1}^{n} (Y_i - g(X_i))^2 + \lambda \int g''(t)^2 \, dt,$$

where $\mathcal{S} = \{\text{all second order differentiable functions}\}$.
- The first term is RSS, and tries to make $g(X_i)$ match Y_i at each X_i.
- The second term is a variability penalty and it is large if g is wiggly.
- The second derivative corresponds to the amount by which the slope is changing. The second derivative of a straight line is zero.
- If g is very smooth, then g' does not vary too much and $\int g''(t)^2 \, dt$ will take a small value.
- λ is a nonnegative tuning parameter. $\lambda \int g''(t)^2 \, dt$ encourages g to be smooth.
- $\lambda \downarrow 0$: the minimizer interpolates the data (large variance); $\lambda \uparrow \infty$: the minimizer will be linear (large bias).

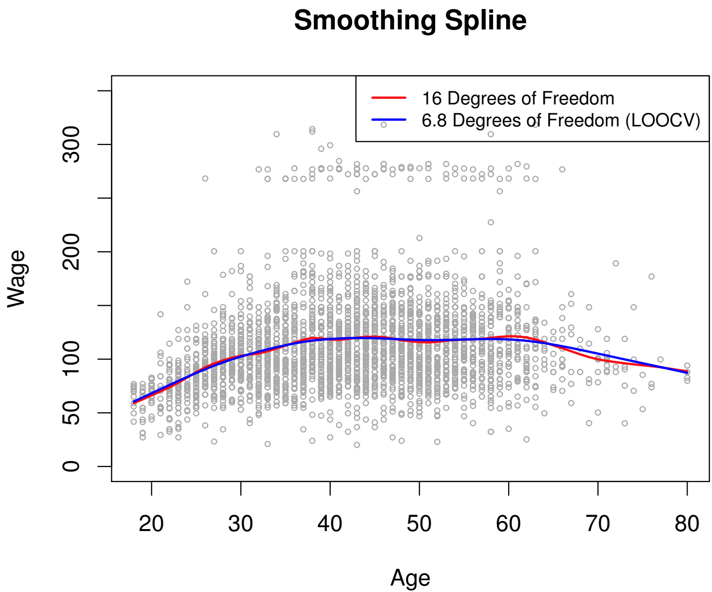
- The solution is a shrunken version of natural cubic spline, with a knot at every unique value of X_i, where λ controls the level of shrinkage.
- Smoothing splines avoid the knot-selection issue, leaving a single λ to be chosen.
- We can find the value of λ that makes the cross-validated RSS as small as possible.
- The leave-one-out cross-validation error (LOOCV) can be computed very efficiently for smoothing splines, with essentially the same cost as computing a single fit.
- The vector of n fitted values can be written as $\hat{\mathbf{g}}_\lambda = \mathbf{S}_\lambda \boldsymbol{Y}$, where $\boldsymbol{Y} = (Y_1, Y_2, ..., Y_n)^\top$ and $\mathbf{S}_\lambda$ is an $n \times n$ matrix that depends on $X_1, X_2, ..., X_n$ and λ.
- The effective degrees of freedom are given by $df_\lambda = \sum_{i=1}^{n} [\mathbf{S}_\lambda]_{ii}$. There is a one-to-one mapping $(0, \infty) \ni \lambda \mapsto df_\lambda$.

LOOCV for smoothing splines

- The LOOCV error is given by

$$RSS_{cv}(\lambda) = \sum_{i=1}^{n} \left(Y_i - \hat{g}_{\lambda}^{(-i)}(X_i) \right)^2 = \sum_{i=1}^{n} \left[\frac{Y_i - \hat{g}_{\lambda}(X_i)}{1 - [\mathbf{S}_{\lambda}]_{ii}} \right]^2,$$

- $\hat{g}_{\lambda}^{(-i)}(X_i)$ indicates the fitted value for this smoothing spline evaluated at X_i, where the fit uses all of the training observations except for the i-th observation.
- $\hat{g}_{\lambda}(X_i)$ indicates the smoothing spline function fit to all of the training observations and evaluated at X_i.
- This formula says that we can compute each of these leave-one-out fits using only $\hat{g}_{\lambda}$.



ISL Figure 7.8

Generalized additive models

- So far in this chapter, we assume a single predictor.
- Generalized additive models (GAMs) allow for flexible nonlinearities in several variables, but retains the additive structure of linear models.
- A natural way to extend the multiple linear regression model is to replace each linear component $\beta_j X_{j,i}$ with a (smooth) nonlinear function $f_j\left(X_{j,i}\right)$:

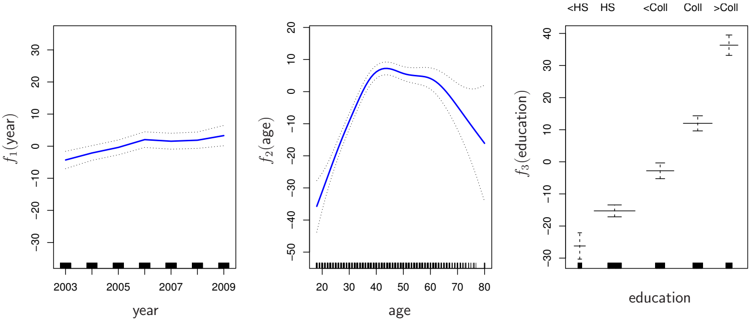
$$Y_i = \beta_0 + f_1\left(X_{1,i}\right) + f_2\left(X_{2,i}\right) + \cdots + f_p\left(X_{p,i}\right) + \epsilon_i .$$

- We can use previous nonlinear methods as building blocks for fitting an additive model.
- GAMs are additive, although low-order interactions such as $X_{1,i} \times X_{2,i}$ can be included as additional predictors.
- Fitting a GAM with cubic splines or natural splines is easily implemented by using multiple least squares regression.
- Fitting a GAM with smoothing splines is not quite simple: in the case of smoothing splines, least squares cannot be used.

- Take, for example, natural splines, and consider the task of fitting the model

$$\texttt{wage} = \beta_0 + f_1\,(\texttt{year}) + f_2\,(\texttt{age}) + f_3\,(\texttt{education}) + \epsilon.$$

- Here `year` and `age` are quantitative variables, and `education` is a qualitative variable with five levels.
- We fit the first two functions using natural splines. We fit the third function using a separate constant for each level, via dummy variables.



ISL Figure 7.12