Constraints on Derivatives

In addition to continuity, we often think the functions should be "smooth" — that some number of derivatives should also be continuous.

This also corresponds to a linear constraint on β. For the $K = 2$, $M = 3$ example, we need

$$\begin{aligned}
\beta_1 h_1'(\xi_1) + \beta_4 h_4'(\xi_1) + \beta_7 h_7'(\xi_1) &= \beta_2 h_2'(\xi_1) + \beta_5 h_5'(\xi_1) + \beta_8 h_8'(\xi_1) \\
\beta_2 h_2'(\xi_2) + \beta_5 h_5'(\xi_2) + \beta_8 h_8'(\xi_2) &= \beta_3 h_3'(\xi_2) + \beta_6 h_6'(\xi_2) + \beta_9 h_9'(\xi_2)
\end{aligned}$$

where $h_m'(\xi_j)$ is the derivative of h_m at ξ_j, evaluated just inside the piece (ξ_j will be a knot at one end). So we have

$$h_1'(\xi_j) = h_2'(\xi_j) = h_3'(\xi_j) = 0$$

$$h_4'(\xi_j) = h_5'(\xi_j) = h_6'(\xi_j) = 1$$

$$h_7'(\xi_j) = h_8'(\xi_j) = h_9'(\xi_j) = 2\xi_j$$

Figure 5.2 (bottom left) shows the effect of requiring continuous first derivatives with $M = 4$. Doing the same for second derivatives gives the bottom right plot.

Cubic Splines

The most popular splines are the *cubic splines*, which have order $M = 4$ (hence degree three), with constraints that the function and its first and second derivatives be continuous at the knot locations.

How many basis functions does this require?

- Start with $M(K+1) = 4(K+1) = 4K+4$ of them.
- Subtract $3K$ for the constraints at the K knots.
- That leaves $K+4$ basis functions for the subspace.

The *natural cubic splines* impose the additional constraints that the second and third derivatives at ξ_1 and ξ_K must be zero. This reduces the number of basis functions to K.

With K basis functions, we might expect that fitting these splines would take time proportional to K^3 — to compute $(X^TX)^{-1}$ where X has K columns. But there is a way of choosing basis functions for cubic splines so that this is reduced to time linear in K (see the appendices of Chapter 5).

Fitting Splines Using Few Knots

A natural cubic spline model with K knots has K parameters (one for each basis function). For other splines, the number may be greater than K.

If we fit such a model by least squares, we will overfit unless K is much less than N (the number of training cases). Eg, with natural cubic splines, if $K = N$, our fit will pass exactly through the data points — bad if the data has any noise.

What to do? One approach is to choose some small K, and put the K knots in suitable places — eg, equally spaced quantiles of the data.

How to choose a good value for K? Cross validation is one possibility.

Does Using Few Knots Make Sense?

It doesn’t make sense if you think of the model as your best attempt at representing reality. Usually, we think that the real function can only be approximated well with large K.

One sign of this: As N increases, cross validation will usually choose bigger and bigger values for K.

Using a fairly small value of K does make sense in terms of the bias-variance tradeoff:

- Using a small K might produce a large bias, since the spline can’t approximate the correct function very well.
- Using a large K will produce a large variance, since there are many parameters to fit.
- Some intermediate value for K will minimize the squared bias plus the variance.

But this tradeoff only applies if we are fitting by least-squares or maximum likelihood (eg, for logistic regression).

Smoothing Splines

A more principled approach is to fit spline models by minimizing residual sum of squares plus a penalty — similiar to ridge regression. If we expect the function to be fairly smooth, a suitable penalty will relate to how big its derivatives are.

Here's one such penalized criterion for how good a function f is:

$$\sum_{i=1}^{N} \Big[y_i - f(x_i)\Big]^2 \; + \; \lambda \int \Big[f''(x)\Big]^2 dx$$

The first term is just the usual RSS. The second term is a penalty that is large if the second derivative of f is large.

The constant λ controls how big the penalty is. If $\lambda = 0$, the first term will force the function to pass through the data points. As $\lambda \rightarrow \infty$, the second term will force the function to be a straight line.

Since f here can be *any* function with second derivatives, it seems very hard to find the f that minimizes this criterion.

Remarkably, it turns out that the solution is always a natural cubic spline, with knots at the distinct data points.